User's Manual

8

# 78K0/Kx2-L

## User's Manual: Hardware

## 8-Bit Single-Chip Microcontrollers

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

Renesas Electronics
www.renesas.com

Rev.4.00 Sep 2010

# 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.

NOTES FOR CMOS DEVICES

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

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

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

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

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

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

# How to Use This Manual

**Readers**

This manual is intended for user engineers who wish to understand the functions of the 78K0/Kx2-L microcontrollers and design and develop application systems and programs for these devices.
The target products are as follows.

- 78K0/KY2-L: μPD78F0550, 78F0551, 78F0552, 78F0555, 78F0556, 78F0557
- 78K0/KA2-L: μPD78F0560, 78F0561, 78F0562, 78F0565, 78F0566, 78F0567
- 78K0/KB2-L: μPD78F0571, 78F0572, 78F0573, 78F0576, 78F0577, 78F0578
- 78K0/KC2-L: μPD78F0581, 78F0582, 78F0583, 78F0586, 78F0587, 78F0588

**Purpose**

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

**Organization**

The manual for the 78K0/Kx2-L microcontrollers is separated into two parts: this manual and the instructions edition (common to the 78K0 microcontrollers).

| **78K0/Kx2-L User's Manual (This Manual)** | **78K/0 Series User's Manual Instructions** |
|---|---|
| • Pin functions<br>• Internal block functions<br>• Interrupts<br>• Other on-chip peripheral functions<br>• Electrical specifications | • CPU functions<br>• Instruction set<br>• Explanation of each instruction |

**How to Read This Manual**

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

- To gain a general understanding of functions:
  → Read this manual in the order of the **CONTENTS**. The mark "<R>" shows major revised points. The revised points can be easily searched by copying an "<R>" in the PDF file and specifying it in the "Find what:" field.
- How to interpret the register format:
  → For a bit number enclosed in angle brackets, the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.
- When you know a register name and want to confirm its details:
  → Refer to **APPENDIX B REGISTER INDEX**.
- To know details of the 78K0 microcontroller instructions:
  → Refer to the separate document **78K/0 Series Instructions User's Manual (U12326E)**.

**Conventions**

| | | |
|---|---|---|
| Data significance: | Higher digits on the left and lower digits on the right | |
| Active low representations: | $\overline{\times\times\times}$ (overscore over pin and signal name) | |
| **Note**: | Footnote for item marked with **Note** in the text | |
| **Caution**: | Information requiring particular attention | |
| **Remark**: | Supplementary information | |
| Numerical representations: | Binary | ···×××× or ××××B |
| | Decimal | ···×××× |
| | Hexadecimal | ···××××H |

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

**Documents Related to Devices**

| Document Name | Document No. |
|---|---|
| 78K0/Kx2-L User's Manual | This manual |
| 78K0/Kx2-L Application Note Setting for Low Power Consumption Operation | U19612E |
| 78K/0 Series User's Manual Instructions | U12326E |
| 78K0 Microcontrollers User's Manual Self Programming Library Type 01 | U18274E |
| 78K0 Microcontrollers Self Programming Library Type 01 Ver. 3.10 Operating Precautions (Notification Document) | ZUD-CD-09-0122 |
| 78K0 Microcontrollers User's Manual EEPROM™ Emulation Library Type 01 | U18275E |
| 78K0 Microcontrollers EEPROM Emulation Library Type 01 Ver.2.10 Operating Precautions (Notification Document) | ZUD-CD-09-0165 |

**Documents Related to Development Tools (Hardware) (User's Manual)**

| Document Name | | Document No. |
|---|---|---|
| QB-MINI2 On-Chip Debug Emulator with Programming Function | | U18371E |
| QB-Programmer Programming GUI | Operation | U18527E |

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

| Document Name | Document No. |
|---|---|
| PG-FP5 Flash Memory Programmer | U18865E |

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

**Documents Related to Development Tools (Software)**

| Document Name | | Document No. |
|---|---|---|
| RA78K0 Ver.3.80 Assembler Package User's Manual[Note 1] | Operation | U17199E |
| | Language | U17198E |
| | Structured Assembly Language | U17197E |
| 78K0 Assembler Package RA78K0 Ver.4.01 Operating Precautions (Notification Document)[Note 1] | | ZUD-CD-07-0181-E |
| CC78K0 Ver.3.70 C Compiler User's Manual[Note 2] | Operation | U17201E |
| | Language | U17200E |
| 78K0 C Compiler CC78K0 Ver. 4.00 Operating Precautions (Notification Document)[Note 2] | | ZUD-CD-07-0103-E |
| SM+ System Simulator User's Manual | Operation | U18601E |
| | User Open Interface | U18212E |
| ID78K0-QB Ver.2.94 Integrated Debugger User's Manual | Operation | U18330E |
| ID78K0-QB Ver.3.00 Integrated Debugger User's Manual | Operation | U18492E |
| PM plus Ver.5.20[Note 3] User's Manual | | U16934E |
| PM+ Ver.6.30[Note 4] User's Manual | | U18416E |

**Notes 1.** This document is installed into the PC together with the tool when installing RA78K0 Ver. 4.01. For descriptions not included in "78K0 Assembler Package RA78K0 Ver. 4.01 Operating Precautions", refer to the user's manual of RA78K0 Ver. 3.80.

**2.** This document is installed into the PC together with the tool when installing CC78K0 Ver. 4.00. For descriptions not included in "78K0 C Compiler CC78K0 Ver. 4.00 Operating Precautions", refer to the user's manual of CC78K0 Ver. 3.70.

**3.** PM plus Ver. 5.20 is the integrated development environment included with RA78K0 Ver. 3.80.

**4.** PM+ Ver. 6.30 is the integrated development environment included with RA78K0 Ver. 4.01. Software tool (assembler, C compiler, debugger, and simulator) products of different versions can be managed.

**Other Documents**

| Document Name | Document No. |
|---|---|
| SEMICONDUCTOR SELECTION GUIDE – Products and Packages – | X13769X |
| Semiconductor Device Mount Manual | **Note** |
| Quality Grades on NEC Semiconductor Devices | C11531E |
| NEC Semiconductor Device Reliability/Quality Control System | C10983E |
| Guide to Prevent Damage for Semiconductor Devices by Electrostatic Discharge (ESD) | C11892E |

<R> **Note** See the "Semiconductor Device Mount Manual" website (http://www2.renesas.com/pkg/en/mount/index.html).

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

All trademarks and registered trademarks are the property of their respective owners.
EEPROM is a trademark of Renesas Electronics Corporation.
Windows is a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.
SuperFlash is a registered trademark of Silicon Storage Technology, Inc. in several countries including the United States and Japan.

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

**CONTENTS**


**CHAPTER 1 OUTLINE ..... 1**

**1.1 Features ..... 1**
**1.2 Ordering Information ..... 4**
**1.3 Pin Configuration (Top View) ..... 6**
1.3.1 78K0/KY2-L ..... 6
1.3.2 78K0/KA2-L ..... 7
1.3.3 78K0/KB2-L ..... 11
1.3.4 78K0/KC2-L ..... 12
**1.4 Block Diagram ..... 18**
1.4.1 78K0/KY2-L ..... 18
1.4.2 78K0/KA2-L ..... 19
1.4.3 78K0/KB2-L ..... 22
1.4.4 78K0/KC2-L ..... 23
**1.5 Outline of Functions ..... 24**

**CHAPTER 2 PIN FUNCTIONS ..... 26**

**2.1 Pin Function List ..... 26**
2.1.1 78K0/KY2-L ..... 27
2.1.2 78K0/KA2-L ..... 29
2.1.3 78K0/KB2-L ..... 34
2.1.4 78K0/KC2-L ..... 37
**2.2 Description of Pin Functions ..... 42**
2.2.1 P00 to P02 (port 0) ..... 42
2.2.2 P10 to P17 (port 1) ..... 43
2.2.3 P20 to P27 (port 2) ..... 44
2.2.4 P30 to P37 (port 3) ..... 45
2.2.5 P40 to P42 (port 4) ..... 47
2.2.6 P60 to P63 (port 6) ..... 48
2.2.7 P70 to P75 (port 7) ..... 49
2.2.8 P120 to P125 (port 12) ..... 50
2.2.9 $AV_{REF}$, $AV_{SS}$, $V_{DD}$, $V_{SS}$ ..... 52
2.2.10 REGC, IC0, IC ..... 52
**2.3 Pin I/O Circuits and Recommended Connection of Unused Pins ..... 54**

**CHAPTER 3 CPU ARCHITECTURE ..... 64**

**3.1 Memory Space ..... 64**
3.1.1 Internal program memory space ..... 69
3.1.2 Internal data memory space ..... 71
3.1.3 Special function register (SFR) area ..... 71
3.1.4 Data memory addressing ..... 72
**3.2 Processor Registers ..... 76**
3.2.1 Control registers ..... 76
3.2.2 General-purpose registers ..... 80
3.2.3 Special function registers (SFRs) ..... 81
**3.3 Instruction Address Addressing ..... 106**

3.3.1 Relative addressing ... 106
3.3.2 Immediate addressing ... 107
3.3.3 Table indirect addressing ... 108
3.3.4 Register addressing ... 109
**3.4 Operand Address Addressing ... 109**
3.4.1 Implied addressing ... 109
3.4.2 Register addressing ... 110
3.4.3 Direct addressing ... 111
3.4.4 Short direct addressing ... 112
3.4.5 Special function register (SFR) addressing ... 113
3.4.6 Register indirect addressing ... 114
3.4.7 Based addressing ... 115
3.4.8 Based indexed addressing ... 116
3.4.9 Stack addressing ... 117

**CHAPTER 4 PORT FUNCTIONS ... 118**

**4.1 Port Functions ... 118**
**4.2 Port Configuration ... 125**
4.2.1 Port 0 ... 126
4.2.2 Port 1 ... 129
4.2.3 Port 2 ... 141
4.2.4 Port 3 ... 147
4.2.5 Port 4 ... 152
4.2.6 Port 6 ... 155
4.2.7 Port 7 ... 160
4.2.8 Port 12 ... 162
**4.3 Registers Controlling Port Function ... 167**
**4.4 Port Function Operations ... 185**
4.4.1 Writing to I/O port ... 185
4.4.2 Reading from I/O port ... 185
4.4.3 Operations on I/O port ... 185
**4.5 Settings of Port Mode Register and Output Latch When Using Alternate Function ... 186**
**4.6 Cautions on 1-Bit Manipulation Instruction for Port Register n (Pn) ... 197**

**CHAPTER 5 CLOCK GENERATOR ... 198**

**5.1 Functions of Clock Generator ... 198**
**5.2 Configuration of Clock Generator ... 199**
**5.3 Registers Controlling Clock Generator ... 202**
**5.4 System Clock Oscillator ... 213**
5.4.1 X1 oscillator ... 213
5.4.2 XT1 oscillator ... 213
5.4.3 When subsystem clock is not used ... 216
5.4.4 Internal high-speed oscillator ... 216
5.4.5 Internal low-speed oscillator ... 216
5.4.6 Prescaler ... 216
**5.5 Clock Generator Operation ... 217**
**5.6 Controlling Clock ... 220**
5.6.1 Example of controlling high-speed system clock ... 220
5.6.2 Example of controlling internal high-speed oscillation clock ... 223

5.6.3 Example of controlling subsystem clock ..... 226
5.6.4 Example of controlling internal low-speed oscillation clock ..... 228
5.6.5 Clocks supplied to CPU and peripheral hardware ..... 229
5.6.6 CPU clock status transition diagram ..... 230
5.6.7 Condition before changing CPU clock and processing after changing CPU clock ..... 236
5.6.8 Time required for switchover of CPU clock and main system clock ..... 237
5.6.9 Conditions before clock oscillation is stopped ..... 239
5.6.10 Peripheral hardware and source clocks ..... 240

**CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00 ..... 241**

**6.1 Functions of 16-Bit Timer/Event Counter 00 ..... 241**
**6.2 Configuration of 16-Bit Timer/Event Counter 00 ..... 242**
**6.3 Registers Controlling 16-Bit Timer/Event Counter 00 ..... 248**
**6.4 Operation of 16-Bit Timer/Event Counter 00 ..... 257**
6.4.1 Interval timer operation ..... 257
6.4.2 Square-wave output operation ..... 260
6.4.3 External event counter operation ..... 263
6.4.4 Operation in clear & start mode entered by TI000 pin valid edge input ..... 267
6.4.5 Free-running timer operation ..... 280
6.4.6 PPG output operation ..... 289
6.4.7 One-shot pulse output operation ..... 293
6.4.8 Pulse width measurement operation ..... 298
**6.5 Special Use of TM00 ..... 306**
6.5.1 Rewriting CR010 during TM00 operation ..... 306
6.5.2 Setting LVS00 and LVR00 ..... 306
**6.6 Cautions for 16-Bit Timer/Event Counter 00 ..... 308**

**CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50 AND 51 ..... 313**

**7.1 Functions of 8-Bit Timer/Event Counters 50 and 51 ..... 313**
**7.2 Configuration of 8-Bit Timer/Event Counters 50 and 51 ..... 314**
**7.3 Registers Controlling 8-Bit Timer/Event Counters 50 and 51 ..... 318**
**7.4 Operations of 8-Bit Timer/Event Counters 50 and 51 ..... 326**
7.4.1 Operation as interval timer ..... 326
7.4.2 Operation as external event counter ..... 328
7.4.3 Square-wave output operation ..... 329
7.4.4 PWM output operation ..... 330
**7.5 Cautions for 8-Bit Timer/Event Counters 50 and 51 ..... 334**

**CHAPTER 8 8-BIT TIMERS H0 AND H1 ..... 335**

**8.1 Functions of 8-Bit Timers H0 and H1 ..... 335**
**8.2 Configuration of 8-Bit Timers H0 and H1 ..... 335**
**8.3 Registers Controlling 8-Bit Timers H0 and H1 ..... 339**
**8.4 Operation of 8-Bit Timers H0 and H1 ..... 347**
8.4.1 Operation as interval timer/square-wave output ..... 347
8.4.2 Operation as PWM output ..... 350
8.4.3 Carrier generator operation (8-bit timer H1 only) ..... 356

**CHAPTER 9 WATCHDOG TIMER ..... 363**

**9.1 Functions of Watchdog Timer ..... 363**
**9.2 Configuration of Watchdog Timer ..... 364**
**9.3 Register Controlling Watchdog Timer ..... 365**
**9.4 Operation of Watchdog Timer ..... 366**
9.4.1 Controlling operation of watchdog timer ..... 366
9.4.2 Setting overflow time of watchdog timer ..... 367
9.4.3 Setting window open period of watchdog timer ..... 368

**CHAPTER 10 REAL-TIME COUNTER ..... 370**

**10.1 Functions of Real-Time Counter ..... 370**
**10.2 Configuration of Real-Time Counter ..... 370**
**10.3 Registers Controlling Real-Time Counter ..... 372**
**10.4 Real-Time Counter Operation ..... 386**
10.4.1 Starting operation of real-time counter ..... 386
10.4.2 Shifting to STOP mode after starting operation ..... 387
10.4.3 Reading/writing real-time counter ..... 388
10.4.4 Setting alarm of real-time counter ..... 390
10.4.5 1 Hz output of real-time counter ..... 391
10.4.6 32.768 kHz output of real-time counter ..... 391
10.4.7 512 Hz, 16.384 kHz output of real-time counter ..... 392
10.4.8 Example of watch error correction of real-time counter ..... 393

**CHAPTER 11 CLOCK OUTPUT CONTROLLER ..... 398**

**11.1 Functions of Clock Output Controller ..... 398**
**11.2 Configuration of Clock Output Controller ..... 398**
**11.3 Registers Controlling Clock Output Controller ..... 399**
**11.4 Operations of Clock Output Controller ..... 400**

**CHAPTER 12 A/D CONVERTER ..... 401**

**12.1 Function of A/D Converter ..... 401**
**12.2 Configuration of A/D Converter ..... 403**
**12.3 Registers Used in A/D Converter ..... 405**
**12.4 A/D Converter Operations ..... 422**
12.4.1 Basic operations of A/D converter ..... 422
12.4.2 Input voltage and conversion results ..... 424
12.4.3 A/D converter operation mode ..... 426
**12.5 How to Read A/D Converter Characteristics Table ..... 428**
**12.6 Cautions for A/D Converter ..... 430**

**CHAPTER 13 OPERATIONAL AMPLIFIERS ..... 434**

**13.1 Function of Operational Amplifier ..... 434**
**13.2 Configuration of Operational Amplifier ..... 435**
**13.3 Registers Used in Operational Amplifier ..... 436**
**13.4 Operational Amplifier Operations ..... 445**
13.4.1 Single AMP mode (operational amplifiers 0 and 1) ..... 445
13.4.2 PGA (Programmable gain amplifier) mode (operational amplifier 0 only) ..... 445

**CHAPTER 14 SERIAL INTERFACE UART6 ..... 446**

**14.1 Functions of Serial Interface UART6 ..... 446**
**14.2 Configuration of Serial Interface UART6 ..... 451**
**14.3 Registers Controlling Serial Interface UART6 ..... 454**
**14.4 Operation of Serial Interface UART6 ..... 465**
14.4.1 Operation stop mode ..... 465
14.4.2 Asynchronous serial interface (UART) mode ..... 466
14.4.3 Dedicated baud rate generator ..... 480
14.4.4 Calculation of baud rate ..... 482

**CHAPTER 15 SERIAL INTERFACE IICA ..... 487**

**15.1 Functions of Serial Interface IICA ..... 487**
**15.2 Configuration of Serial Interface IICA ..... 490**
**15.3 Registers Controlling Serial Interface IICA ..... 492**
**15.4 $I^2C$ Bus Mode Functions ..... 505**
15.4.1 Pin configuration ..... 505
15.4.2 Setting transfer clock by using IICWL and IICWH registers ..... 506
**15.5 $I^2C$ Bus Definitions and Control Methods ..... 507**
15.5.1 Start conditions ..... 507
15.5.2 Addresses ..... 508
15.5.3 Transfer direction specification ..... 508
15.5.4 Acknowledge ($\overline{ACK}$) ..... 509
15.5.5 Stop condition ..... 510
15.5.6 Wait ..... 511
15.5.7 Canceling wait ..... 513
15.5.8 Interrupt request (INTIICA0) generation timing and wait control ..... 514
15.5.9 Address match detection method ..... 515
15.5.10 Error detection ..... 515
15.5.11 Extension code ..... 515
15.5.12 Arbitration ..... 516
15.5.13 Wakeup function ..... 518
15.5.14 Communication reservation ..... 521
15.5.15 Cautions ..... 525
15.5.16 Communication operations ..... 526
15.5.17 Timing of $I^2C$ interrupt request (INTIICA0) occurrence ..... 534
**15.6 Timing Charts ..... 555**

**CHAPTER 16 SERIAL INTERFACES CSI10 AND CSI11 ..... 562**

**16.1 Functions of Serial Interfaces CSI10 and CSI11 ..... 562**
**16.2 Configuration of Serial Interfaces CSI10 and CSI11 ..... 563**
**16.3 Registers Controlling Serial Interfaces CSI10 and CSI11 ..... 566**
**16.4 Operation of Serial Interfaces CSI10 and CSI11 ..... 576**
16.4.1 Operation stop mode ..... 576
16.4.2 3-wire serial I/O mode ..... 577

**CHAPTER 17 INTERRUPT FUNCTIONS ..... 591**

**17.1 Interrupt Function Types ..... 591**
**17.2 Interrupt Sources and Configuration ..... 591**

**17.3 Registers Controlling Interrupt Functions ..... 596**
**17.4 Interrupt Servicing Operations ..... 629**
17.4.1 Maskable interrupt acknowledgment ..... 629
17.4.2 Software interrupt request acknowledgment ..... 631
17.4.3 Multiple interrupt servicing ..... 632
17.4.4 Interrupt request hold ..... 635

**CHAPTER 18 KEY INTERRUPT FUNCTION ..... 636**

**18.1 Functions of Key Interrupt ..... 636**
**18.2 Configuration of Key Interrupt ..... 637**
**18.3 Register Controlling Key Interrupt ..... 638**

**CHAPTER 19 STANDBY FUNCTION ..... 639**

**19.1 Standby Function and Configuration ..... 639**
19.1.1 Standby function ..... 639
19.1.2 Registers controlling standby function ..... 640
**19.2 Standby Function Operation ..... 642**
19.2.1 HALT mode ..... 642
19.2.2 STOP mode ..... 647

**CHAPTER 20 RESET FUNCTION ..... 655**

**20.1 Register for Confirming Reset Source ..... 664**

**CHAPTER 21 POWER-ON-CLEAR CIRCUIT ..... 665**

**21.1 Functions of Power-on-Clear Circuit ..... 665**
**21.2 Configuration of Power-on-Clear Circuit ..... 666**
**21.3 Operation of Power-on-Clear Circuit ..... 666**
**21.4 Cautions for Power-on-Clear Circuit ..... 669**

**CHAPTER 22 LOW-VOLTAGE DETECTOR ..... 671**

**22.1 Functions of Low-Voltage Detector ..... 671**
**22.2 Configuration of Low-Voltage Detector ..... 672**
**22.3 Registers Controlling Low-Voltage Detector ..... 672**
**22.4 Operation of Low-Voltage Detector ..... 676**
22.4.1 When used as reset ..... 678
22.4.2 When used as interrupt ..... 683
**22.5 Cautions for Low-Voltage Detector ..... 688**

**CHAPTER 23 REGULATOR ..... 691**

**23.1 Regulator Overview ..... 691**
**23.2 Register Controlling Regulator ..... 691**
**23.3 Cautions for Self Programming ..... 692**

**CHAPTER 24 OPTION BYTE ..... 693**

**24.1 Functions of Option Bytes ..... 693**
**24.2 Format of Option Byte ..... 694**

**CHAPTER 25 FLASH MEMORY ..... 699**

**25.1 Internal Memory Size Switching Register ..... 699**
**25.2 Writing with Flash Memory Programmer ..... 700**
**25.3 Programming Environment ..... 701**
**25.4 Connection of Pins on Board ..... 702**
25.4.1 TOOL pins ..... 702
25.4.2 $\overline{\text{RESET}}$ pin ..... 703
25.4.3 Port pins ..... 703
25.4.4 REGC pin ..... 703
25.4.5 Other signal pins ..... 703
25.4.6 Power supply ..... 703
25.4.7 On-board writing when connecting crystal/ceramic resonator ..... 704
**25.5 Programming Method ..... 705**
25.5.1 Controlling flash memory ..... 705
25.5.2 Flash memory programming mode ..... 705
25.5.3 Communication commands ..... 705
**25.6 Security Settings ..... 707**
**25.7 Processing Time for Each Command When PG-FP5 Is Used (Reference) ..... 709**
**25.8 Flash Memory Programming by Self Programming ..... 712**
25.8.1 Register controlling self programming mode ..... 713
25.8.2 Flow of self programming (Rewriting Flash Memory) ..... 713
25.8.3 Boot swap function ..... 715
**25.9 Creating ROM Code to Place Order for Previously Written Product ..... 717**
25.9.1 Procedure for using ROM code to place an order ..... 717

**CHAPTER 26 ON-CHIP DEBUG FUNCTION ..... 718**

**26.1 Connecting QB-MINI2 to 78K0/Kx2-L Microcontrollers ..... 718**
**26.2 On-Chip Debug Security ID ..... 721**
**26.3 Securing of User Resources ..... 722**

**CHAPTER 27 INSTRUCTION SET ..... 723**

**27.1 Conventions Used in Operation List ..... 723**
27.1.1 Operand identifiers and specification methods ..... 723
27.1.2 Description of operation column ..... 724
27.1.3 Description of flag operation column ..... 724
**27.2 Operation List ..... 725**
**27.3 Instructions Listed by Addressing Type ..... 733**

**CHAPTER 28 ELECTRICAL SPECIFICATIONS ..... 736**

**CHAPTER 29 PACKAGE DRAWINGS ..... 769**

**29.1 78K0/KY2-L ..... 769**
**29.2 78K0/KA2-L ..... 770**
**29.3 78K0/KB2-L ..... 773**
**29.4 78K0/KC2-L ..... 774**

**CHAPTER 30 RECOMMENDED SOLDERING CONDITIONS ..... 777**

**CHAPTER 31 CAUTIONS FOR WAIT ..... 778**

**31.1 Cautions for Wait ..... 778**
**31.2 Peripheral Hardware That Generates Wait ..... 779**

**APPENDIX A DEVELOPMENT TOOLS ..... 780**

**A.1 Software Package ..... 783**
**A.2 Language Processing Software ..... 783**
**A.3 Flash Memory Programming Tools ..... 784**
A.3.1 When using flash memory programmer PG-FP5 and FL-PR5 ..... 784
A.3.2 When using on-chip debug emulator with programming function QB-MINI2 ..... 784
**A.4 Debugging Tools (Hardware) ..... 785**
A.4.1 When using in-circuit emulator ..... 785
A.4.2 When using on-chip debug emulator with programming function QB-MINI2 ..... 785
**A.5 Debugging Tools (Software) ..... 785**

**APPENDIX B REGISTER INDEX ..... 786**

**B.1 Register Index (In Alphabetical Order with Respect to Register Names) ..... 786**
**B.2 Register Index (In Alphabetical Order with Respect to Register Symbol) ..... 790**

**APPENDIX C REVISION HISTORY ..... 794**

**C.1 Major Revisions in This Edition ..... 794**
**C.2 Revision History of Preceding Editions ..... 798**

# CHAPTER 1 OUTLINE

## 1.1 Features

- 78K0 CPU core
- I/O ports, ROM and RAM capacities

| Item / Products | I/O ports | Program Memory (Flash Memory) | Data Memory (Internal High-Speed RAM) |
|---|---|---|---|
| 78K0/KY2-L (16 pins) | 12 (CMOS I/O: 9, CMOS input: 3) | 4 KB to 16 KB | 384 bytes to 768 bytes |
| 78K0/KA2-L (20 pins) | 16 (CMOS I/O: 13, CMOS input: 3) | | |
| <R> 78K0/KA2-L (25 pins) | 21 (CMOS I/O: 18, CMOS input: 3) | | |
| <R> 78K0/KA2-L (32 pins) | 25 (CMOS I/O: 22, CMOS input: 3) | | |
| 78K0/KB2-L (30 pins) | 24 (CMOS I/O: 21, CMOS input: 3) | 8 KB to 32 KB | 512 bytes to 1 KB |
| <R> 78K0/KC2-L (40 pins) | 34 (CMOS I/O: 29, CMOS input: 5) | | |
| 78K0/KC2-L (44 pins) | 38 (CMOS I/O: 33, CMOS input: 5) | | |
| 78K0/KC2-L (48 pins) | 42 (CMOS I/O: 37, CMOS input: 5) | | |

- Low power consumption ($V_{DD}$ = 3.0 V)
  - Internal high-speed oscillation mode: 220 $\mu$A (TYP.) (at $f_{CPU}$ = 1 MHz operation)
  - <R> STOP mode: 0.58 $\mu$A (TYP.) (at $f_{IL}$ = 30 kHz operation)
  - <R> Subsystem clock and HALT mode: 0.98 $\mu$A (at $f_{SUB}$ = 32.768 kHz operation) ∗ 78K0/KC2-L only
- Clock
  - High-speed system clock … Selected from the following three sources
    - Ceramic/crystal oscillator: 1 to 10 MHz
    - External clock: 1 to 10 MHz
    - Internal high-speed oscillator: 4 MHz ± 2 % (–20 to +70°C), or 8 MHz ± 3 %(–40 to +85°C)
  - Low-speed system oscillator 30 kHz ± 10 % … Watchdog timer, timer clock in intermittent operation
  - Subsystem clock: Clock to operate the real-time counter mainly (32.768 kHz)
- Power-on-clear (POC) circuit
- Low-voltage detector (LVI) (An interrupt/reset (selectable) is generated when the detection voltage is reached))
  - Detection voltage: Selectable from sixteen levels between 1.91 and 4.22 V
- Single-power-supply flash memory
  - Flash self programming enabled
  - Software protection function: Protected from outside party copying (no flash reading command)
- Safety function
  - Watchdog timer operated by clock independent from CPU
    … A hang-up can be detected even if the system clock stops
  - Supply voltage drop detectable by LVI
    … Appropriate processing can be executed before the supply voltage drops below the operation voltage
  - Equipped with option byte function
    … Important system operation settings set in hardware

○ Timer

- 16-bit timer/event counter … PPG output, capture input, external event counter input
- 8-bit timer H … PWM output
- 8-bit timer/event counter 5 … PWM output, external event counter input
- Watchdog timer … Operable with low-speed internal oscillation clock
- Real-time counter … Available to count up in year, month, week, day, hour, minute, and second units

| Item / Products | 16-bit timer/event counter | 8-bit timer | Watchdog timer | Real-time counter |
|---|---|---|---|---|
| 78K0/KY2-L (16 pins) | 1 ch | Timer H: 1 ch<br>Timer 5: 1 ch | 1 ch | – |
| 78K0/KA2-L (20 pins) | | | | |
| <R> 78K0/KA2-L (25 pins) | | | | |
| <R> 78K0/KA2-L (32 pins) | | | | |
| 78K0/KB2-L (30 pins) | | Timer H: 2 ch<br>Timer 5: 2 ch | | |
| <R> 78K0/KC2-L (40 pins) | | | | 1 ch |
| 78K0/KC2-L (44 pins) | | | | |
| 78K0/KC2-L (48 pins) | | | | |

○ Serial interface

- UART … Asynchronous 2-wire serial interface
- IICA … Clock synchronous 2-wire serial interface, multimaster supported, standby can be released upon address match in slave mode
- CSI … Clock synchronous 3-wire serial interface

| Item / Products | UART | IIC | CSI |
|---|---|---|---|
| 78K0/KY2-L (16 pins) | 1 ch | 1 ch | – |
| 78K0/KA2-L (20 pins) | | | |
| <R> 78K0/KA2-L (25 pins) | | | 1 ch (CSI11[Note]) |
| <R> 78K0/KA2-L (32 pins) | | | |
| 78K0/KB2-L (30 pins) | | | 1 ch (CSI10) |
| <R> 78K0/KC2-L (40 pins) | | | 2 ch (CSI10, CSI11) |
| 78K0/KC2-L (44 pins) | | | |
| 78K0/KC2-L (48 pins) | | | 2 ch (CSI10, CSI11[Note]) |

**Note** Can control by an enabled signal, when using CSI11 in the slave mode.

○ 10-bit resolution A/D conversion

- 78K0/KY2-L: 4 ch
- <R> 78K0/KA2-L (20 pins): 6 ch, 78K0/KA2-L (25 pins): 7 ch
- <R> 78K0/KA2-L (32 pins): 11 ch
- 78K0/KB2-L: 7 ch
- <R> 78K0/KC2-L (40 pins): 10 ch, 78K0/KC2-L (44 pins, 48 pins): 11 ch

○ Operational amplifier (products with operational amplifier only)

- 78K0/KY2-L, 78K0/KA2-L: 1 ch
- 78K0/KB2-L, 78K0/KC2-L: 2 ch

- On-chip debug function …Available to control for the target device, and to reference memory
- Assembler and C language supported
- Development tools
  - Support for full-function emulator (IECUBE), and simplified emulator (MINICUBE2)
- Power supply voltage: $V_{DD}$ = 1.8 to 5.5 V
- Operating ambient temperature: $T_A$ = –40 to +85°C

## <R>1.2 Ordering Information

**[Part Number]**

μPD78F05 x y ΔΔ - ××× -AX

| | Semiconductor | |
|---|---|---|
| -AX | Lead-free | Product contains no lead in any area (Terminal finish is Ni/Pd/Au plating) |

| x | ΔΔ - ××× | Package Type |
|---|---|---|
| 5 | MA-FAA | 16-pin plastic SSOP (5.72 mm (225)) |
| 6 | MC-CAA | 20-pin plastic SSOP (7.62 mm (300)) |
| | FC-2N2 | 25-pin plastic FLGA (3x3) |
| | K8-3B4 | 32-pin plastic WQFN (5x5) |
| 7 | MC-CAB | 30-pin plastic SSOP (7.62 mm (300)) |
| 8 | K8-4B4 | 40-pin plastic WQFN (6x6) |
| | GB-GAF | 44-pin plastic LQFP (10x10) |
| | GA-GAM | 48-pin plastic LQFP (fine pitch) (7x7) |

| y | Flash Memory Capacity | High-speed RAM Capacity | Operational amplifier |
|---|---|---|---|
| 0 | 4 KB | 384 bytes | Not mounted |
| 1 | 8 KB | 512 bytes | |
| 2 | 16 KB | 768 bytes | |
| 3 | 32 KB | 1 KB | |
| 5 | 4 KB | 384 bytes | Mounted |
| 6 | 8 KB | 512 bytes | |
| 7 | 16 KB | 768 bytes | |
| 8 | 32 KB | 1 KB | |

| | Product Type |
|---|---|
| F | Flash memory version |

**[Example of Part Number]**

μPD78F05 5 0 MA-FAA -AX

- Lead-free
- 16-pin plastic SSOP (5.72 mm)
- High-speed RAM: 384 bytes, flash memory: 4 KB, operational amplifier: not mounted
- Flash memory version

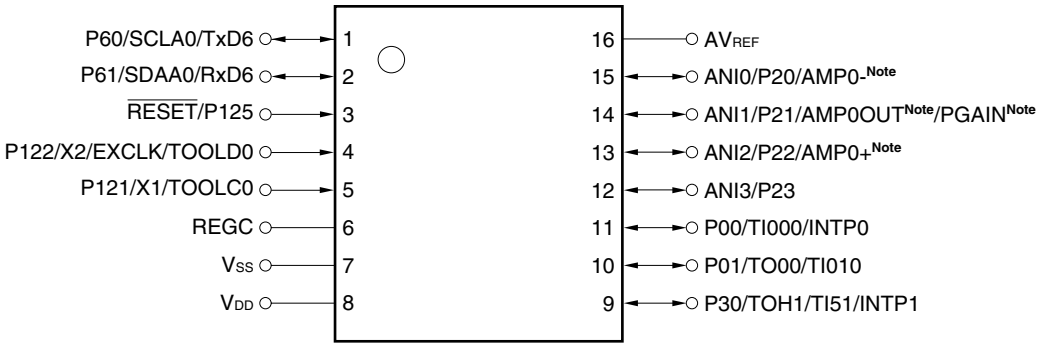
**[List of Part Number]**

| 78K0/Kx2-L Microcontrollers | Package | Part Number |
|---|---|---|
| 78K0/KY2-L | 16-pin plastic SSOP (5.72 mm (225)) | μPD78F0550MA-FAA-AX, 78F0551MA-FAA-AX, 78F0552MA-FAA-AX, 78F0555MA-FAA-AX, 78F0556MA-FAA-AX, 78F0557MA-FAA-AX |
| 78K0/KA2-L | 20-pin plastic SSOP (7.62 mm (300)) | μPD78F0560MC-CAA-AX, 78F0561MC-CAA-AX, 78F0562MC-CAA-AX, 78F0565MC-CAA-AX, 78F0566MC-CAA-AX, 78F0567MC-CAA-AX |
| | 25-pin plastic FLGA (3x3) | μPD78F0560FC-2N2-A, 78F0561FC-2N2-A, 78F0562FC-2N2-A, 78F0565FC-2N2-A, 78F0566FC-2N2-A, 78F0567FC-2N2-A |
| | 32-pin plastic WQFN (5x5) | μPD78F0560K8-3B4-AX, 78F0561K8-3B4-AX, 78F0562K8-3B4-AX, 78F0565K8-3B4-AX, 78F0566K8-3B4-AX, 78F0567K8-3B4-AX |
| 78K0/KB2-L | 30-pin plastic SSOP (7.62 mm (300)) | μPD78F0571MC-CAB-AX, 78F0572MC-CAB-AX, 78F0573MC-CAB-AX, 78F0576MC-CAB-AX, 78F0577MC-CAB-AX, 78F0578MC-CAB-AX |
| 78K0/KC2-L | 40-pin plastic WQFN (6x6) | μPD78F0581K8-4B4-AX, 78F0582K8-4B4-AX, 78F0583K8-4B4-AX, 78F0586K8-4B4-AX, 78F0587K8-4B4-AX, 78F0588K8-4B4-AX |
| | 44-pin plastic LQFP (10x10) | μPD78F0581GB-GAF-AX, 78F0582GB-GAF-AX, 78F0583GB-GAF-AX, 78F0586GB-GAF-AX, 78F0587GB-GAF-AX, 78F0588GB-GAF-AX |
| | 48-pin plastic LQFP (fine pitch) (7x7) | μPD78F0581GA-GAM-AX, 78F0582GA-GAM-AX, 78F0583GA-GAM-AX, 78F0586GA-GAM-AX, 78F0587GA-GAM-AX, 78F0588GA-GAM-AX |

## 1.3 Pin Configuration (Top View)

### 1.3.1 78K0/KY2-L

- 16-pin plastic SSOP (5.72 mm (225))




| | | | |
|---|---|---|---|
| AMP0-Note, AMP0+Note: | Amplifier Input | P121, P122, P125 : | Port 12 |
| AMP0OUTNote: | Amplifier Output | REGC : | Regulator Capacitance |
| PGAINNote: | Programmable Gain | RESET : | Reset |
| | Amplifier Input | RxD6 : | Receive Data |
| ANI0 to ANI3 : | Analog Input | SCLA0 : | Serial Clock Input/Output |
| AVREF : | Analog Reference | SDAA0 : | Serial Data Input/Output |
| | Voltage | TI000, TI010, TI51 : | Timer Input |
| EXCLK : | External Clock Input | TO00, TOH1 : | Timer Output |
| | (Main System Clock) | TOOLC0 : | Clock Input for Tool |
| INTP0, INTP1 : | External Interrupt | TOOLD0 : | Data Input/Output for Tool |
| | Input | TxD6 : | Transmit Data |
| P00, P01 : | Port 0 | VDD : | Power Supply |
| P20 to P23 : | Port 2 | VSS : | Ground |
| P30 : | Port 3 | X1, X2 : | Crystal Oscillator |
| P60, P61 : | Port 6 | | (Main System Clock) |

**Note** μPD78F0555, 78F0556, 78F0557 (products with operational amplifier) only

**Cautions 1. VSS functions alternately as the ground potential of the A/D converter. Be sure to connect VSS to a stabilized GND (= 0 V).**

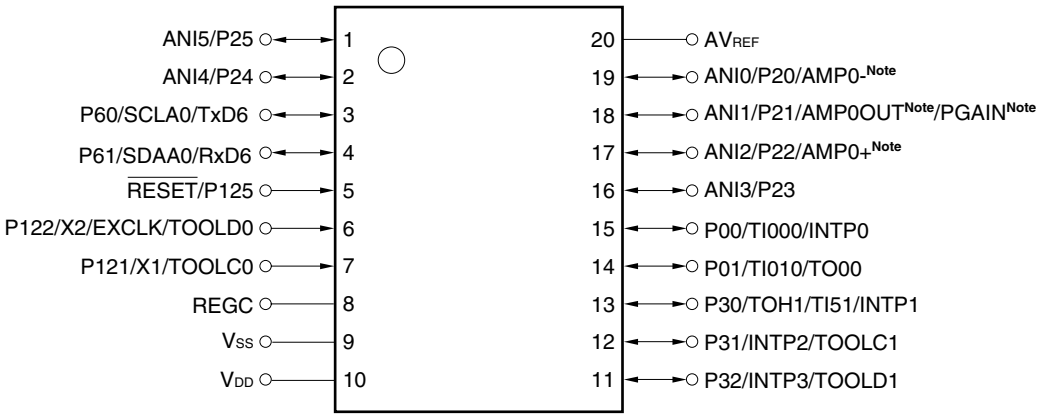
**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 are set in the analog input mode after release of reset.**

**4. RESET/P125 immediately after release of reset is set in the external reset input.**

### 1.3.2 78K0/KA2-L

**(1) 20-pin plastic SSOP (7.62 mm (300))**




| | | | |
|---|---|---|---|
| AMP0-Note, AMP0+Note : | Amplifier Input | P121, P122, P125 : | Port 12 |
| AMP0OUTNote : | Amplifier Output | REGC : | Regulator Capacitance |
| PGAINNote : | Programmable Gain | $\overline{\text{RESET}}$ : | Reset |
| | Amplifier Input | RxD6 : | Receive Data |
| ANI0 to ANI5 : | Analog Input | SCLA0 : | Serial Clock Input/Output |
| AVREF : | Analog Reference | SDAA0 : | Serial Data Input/Output |
| | Voltage | TI000, TI010, TI51 : | Timer Input |
| EXCLK : | External Clock Input | TO00, TOH1 : | Timer Output |
| | (Main System Clock) | TOOLC0, TOOLC1 : | Clock Input for Tool |
| INTP0 to INTP3 : | External Interrupt | TOOLD0, TOOLD1 : | Data Input/Output for Tool |
| | Input | TxD6 : | Transmit Data |
| P00, P01 : | Port 0 | VDD : | Power Supply |
| P20 to P25 : | Port 2 | VSS : | Ground |
| P30 to P32 : | Port 3 | X1, X2 : | Crystal Oscillator |
| P60, P61 : | Port 6 | | (Main System Clock) |

**Note** $\mu$PD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

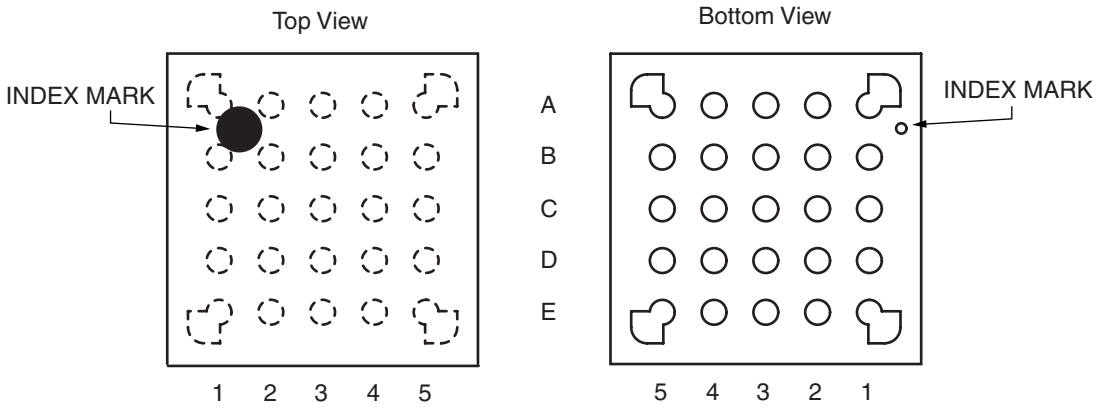
**Cautions 1. VSS functions alternately as the ground potential of the A/D converter. Be sure to connect VSS to a stabilized GND (= 0 V).**

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

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI5/P25 are set in the analog input mode after release of reset.**

**4. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

<R> **(2) 25-pin plastic FLGA (3x3) (1/2)**




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| A | $V_{SS}$ | $V_{DD}$ | $\overline{RESET}$/ P125 | P61/RXD6 /SDAA0 | ANI4/P24 |
| B | REGC | P121/X1/TOOLC0 (/TI000)(/INTP0) | P122/X2/EXCLK /TOOLD0 | P60/TXD6/SCLA0 | ANI6/P26 |
| C | P35/$\overline{SCK11}$ | P36/SI11 | P37/SO11 | P02/$\overline{SSI11}$/INTP5 | ANI5/P25 |
| D | P33 | P00/TI000/INTP0 (/TOH1)(/TI51) | ANI3/P23 | ANI2/P22 /AMP0+Note | ANI0/P20 /AMP0-Note |
| E | P34/INTP4 (/TOH1)(/TI51) | P32/INTP3 /TOOLD1 | P31/INTP2 /TOOLC1 | ANI1/P21/ AMP0OUTNote /PGAINNote | $AV_{REF}$ |

AMP0-Note, AMP0+Note : Amplifier Input
AMP0OUTNote : Amplifier Output
PGAINNote : Programmable Gain Amplifier Input
ANI0 to ANI6 : Analog Input
$AV_{REF}$ : Analog Reference Voltage
EXCLK : External Clock Input (Main System Clock)
INTP0, INTP2 to INTP5 : External Interrupt Input
P00, P02 : Port 0
P20 to P26 : Port 2
P31 to P37 : Port 3
P60, P61 : Port 6
P121, P122, P125 : Port 12
REGC : Regulator Capacitance
$\overline{RESET}$ : Reset
RxD6 : Receive Data
$\overline{SCK11}$ : Serial Clock Input/Output
SCLA0 : Serial Clock Input/Output
SDAA0 : Serial Data Input/Output
SI11 : Serial Data Input
SO11 : Serial Data Output
$\overline{SSI11}$ : Serial Interface Chip Select Input
TI000, TI51 : Timer Input
TOH1 : Timer Output
TOOLC0, TOOLC1 : Clock Input for Tool
TOOLD0, TOOLD1 : Data Input/Output for Tool
TxD6 : Transmit Data
$V_{DD}$ : Power Supply
$V_{SS}$ : Ground
X1, X2 : Crystal Oscillator (Main System Clock)

**(2) 25-pin plastic FLGA (3x3) (2/2)**

**Note** $\mu$PD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

**Cautions 1. $V_{SS}$ functions alternately as the ground potential of the A/D converter. Be sure to connect $V_{SS}$ to a stabilized GND (= 0 V).**

**2. Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F).**

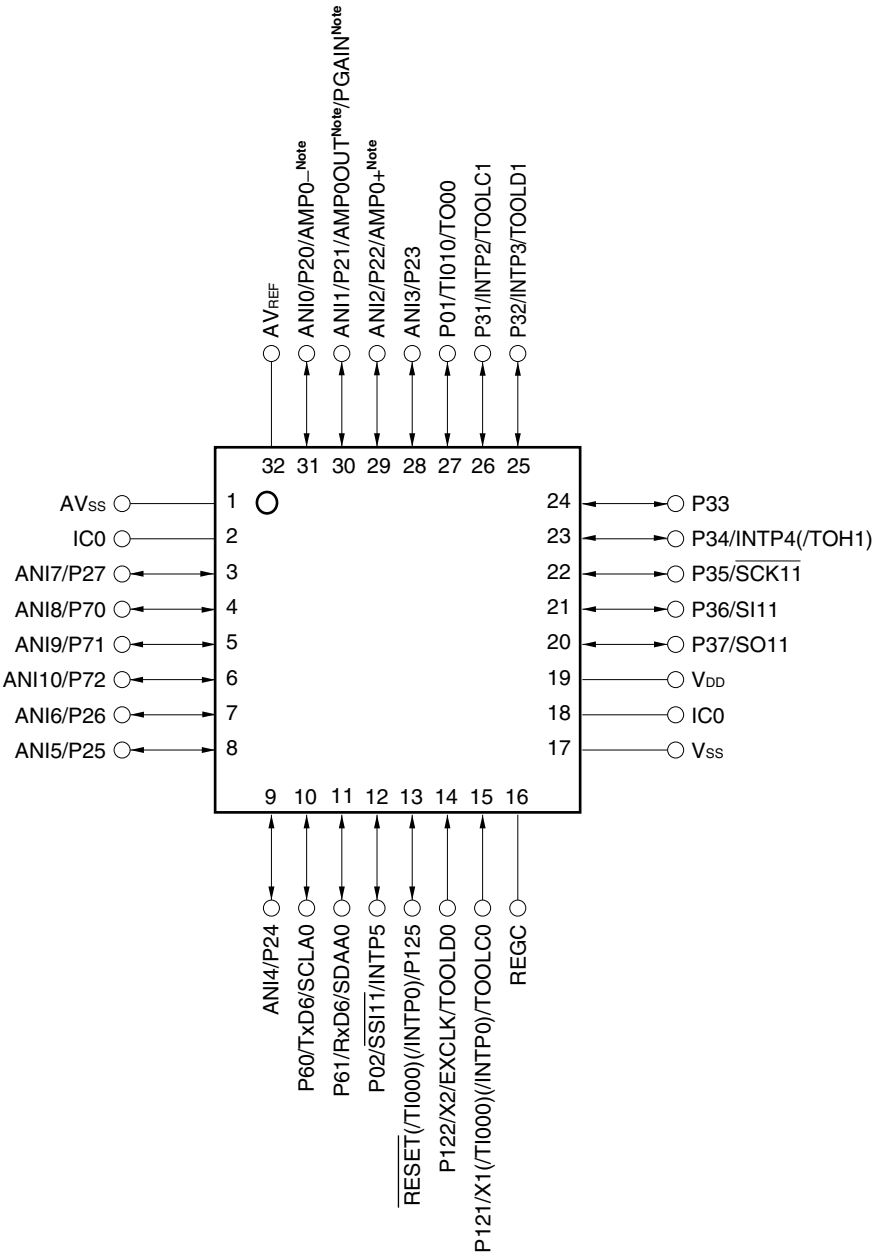
**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI6/P26 are set in the analog input mode after release of reset.**

**4. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

**5. Set P30 and P01 to output mode (PM30 = PM01 = 0) by using software after release of reset.**

**Remark** Functions in parentheses ( ) in the figure above can be assigned by setting the port alternate switch control register (MUXSEL).

<R> **(3) 32-pin plastic WQFN (5x5) (1/2)**



<R> **(3) 32-pin plastic WQFN (5x5) (2/2)**

| | | | |
|---|---|---|---|
| AMP0-[Note], AMP0+[Note] | : Amplifier Input | $\overline{\text{RESET}}$ | : Reset |
| AMP0OUT[Note] | : Amplifier Output | RxD6 | : Receive Data |
| PGAIN[Note] | : Programmable Gain Amplifier Input | $\overline{\text{SCK11}}$ | : Serial Clock Input/Output |
| | | SCLA0 | : Serial Clock Input/Output |
| ANI0 to ANI10 | : Analog Input | SDAA0 | : Serial Data Input/Output |
| $AV_{REF}$ | : Analog Reference Voltage | SI11 | : Serial Data Input |
| $AV_{SS}$ | : Analog Ground | SO11 | : Serial Data Output |
| EXCLK | : External Clock Input (Main System Clock) | $\overline{\text{SSI11}}$ | : Serial Interface Chip Select Input |
| IC0 | : Internally Connected | TI000, TI010 | : Timer Input |
| INTP0, INTP2 to INTP5 | : External Interrupt Input | TO00, TOH1 | : Timer Output |
| P01, P02 | : Port 0 | TOOLC0, TOOLC1 | : Clock Input for Tool |
| P20 to P27 | : Port 2 | TOOLD0, TOOLD1 | : Data Input/Output for Tool |
| P31 to P37 | : Port 3 | TxD6 | : Transmit Data |
| P60, P61 | : Port 6 | $V_{DD}$ | : Power Supply |
| P70 to P72 | : Port 7 | $V_{SS}$ | : Ground |
| P121, P122, P125 | : Port 12 | X1, X2 | : Crystal Oscillator (Main System Clock) |
| REGC | : Regulator Capacitance | | |

**Note** $\mu$PD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

**Cautions 1. Connect directly IC0 (Internally Connected) to $V_{SS}$.**

**2. $V_{SS}$ functions alternately as the ground potential of the A/D converter. Be sure to connect $V_{SS}$ to a stabilized GND (= 0 V).**

**3. Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F).**

**4. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, ANI3/P23 to ANI7/P27, and ANI8/P70 to ANI10/P72 are set in the analog input mode after release of reset.**

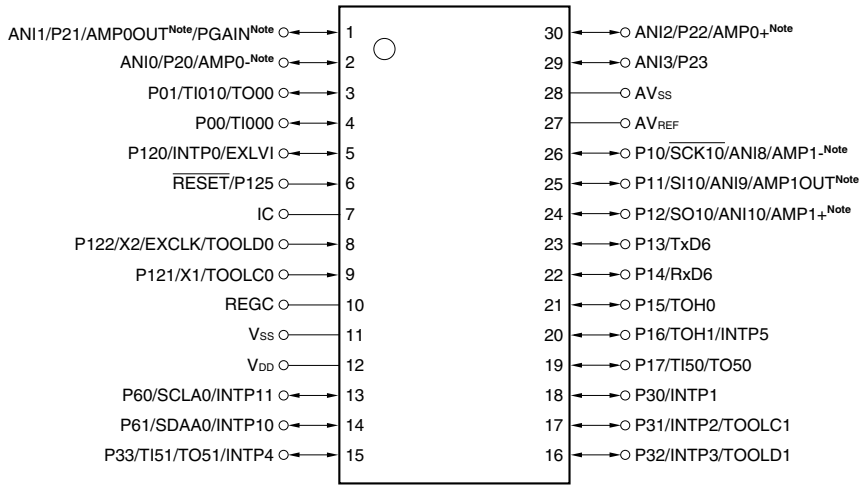
**5. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

**6. Set P30 to output mode (PM30 = 0) by using software after release of reset.**

**Remark** Functions in parentheses ( ) in the figure above can be assigned by setting the port alternate switch control register (MUXSEL).

### 1.3.3 78K0/KB2-L

- 30-pin plastic SSOP (7.62 mm (300))



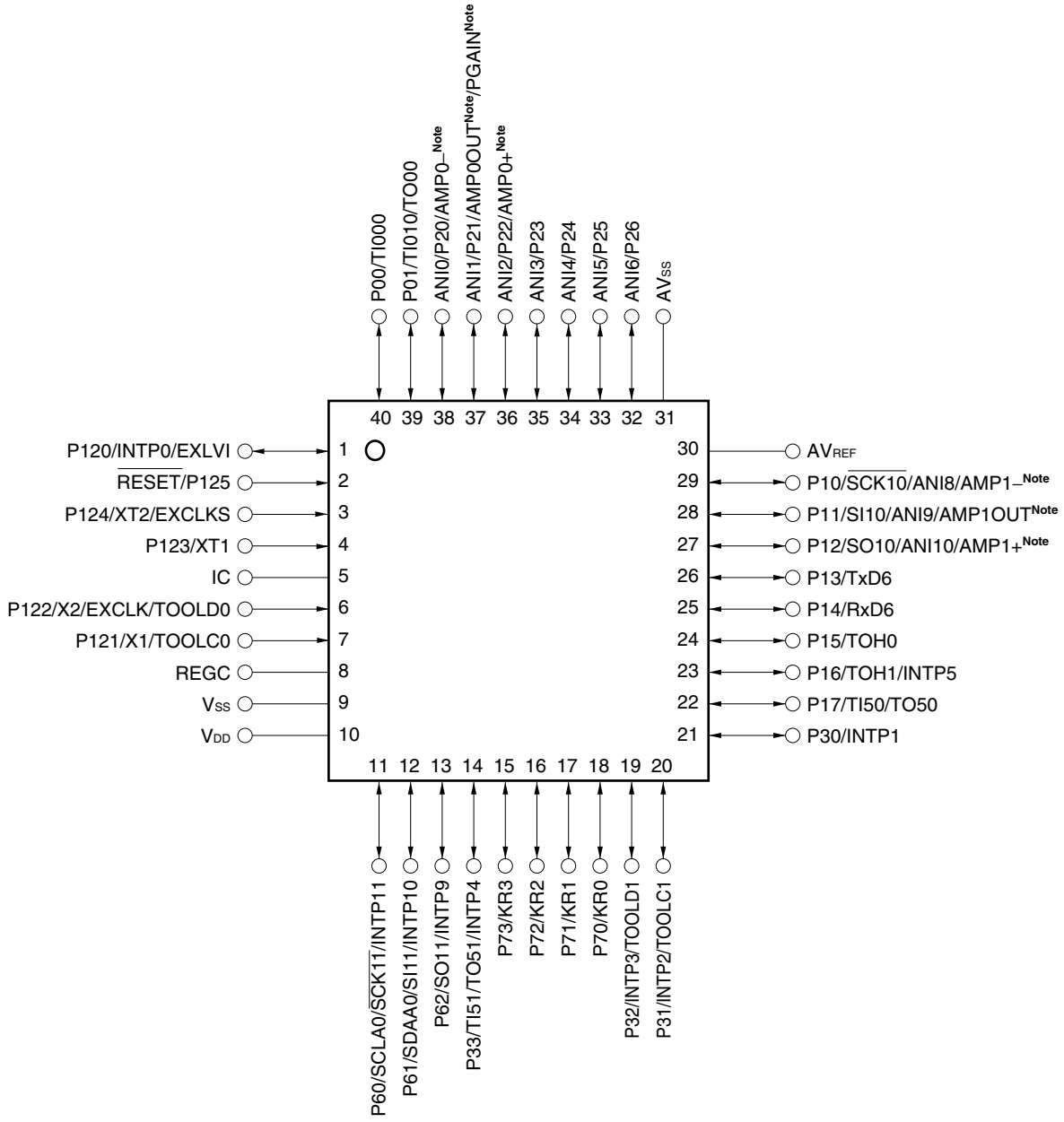

| | | | |
|---|---|---|---|
| AMP0- Note, AMP0+ Note, | | P20 to P23 : | Port 2 |
| AMP1- Note, AMP1+ Note : | Amplifier Input | P30 to P33 : | Port 3 |
| AMP0OUT Note, | | P60, P61 : | Port 6 |
| AMP1OUT Note : | Amplifier Output | P120 to P122, P125 : | Port 12 |
| PGAIN Note : | Programmable Gain | REGC : | Regulator Capacitance |
| | Amplifier Input | RESET : | Reset |
| ANI0 to ANI3, | | RxD6 : | Receive Data |
| ANI8 to ANI10 : | Analog Input | SCLA0, SCK10 : | Serial Clock Input/Output |
| AVREF : | Analog Reference | SDAA0 : | Serial Data Input/Output |
| | Voltage | SI10 : | Serial Data Input |
| AVSS : | Analog Ground | SO10 : | Serial Data Output |
| EXCLK : | External Clock Input | TI000, TI010, TI50, TI51 : | Timer Input |
| | (Main System Clock) | TO00, TO50, TO51, | |
| EXLVI : | External potential Input | TOH0, TOH1 : | Timer Output |
| | for Low-voltage detector | TOOLC0, TOOLC1 : | Clock Input for Tool |
| IC : | Internally Connected | TOOLD0, TOOLD1 : | Data Input/Output for Tool |
| INTP0 to INTP5, | | TxD6 : | Transmit Data |
| INTP10, INTP11 : | External Interrupt Input | VDD : | Power Supply |
| P00, P01 : | Port 0 | VSS : | Ground |
| P10 to P17 : | Port 1 | X1, X2 : | Crystal Oscillator |
| | | | (Main System Clock) |

**Note** μPD78F0576, 78F0577, 78F0578 (products with operational amplifier) only

**Cautions 1. Leave the IC (Internally Connected) pin open.**
**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).**
**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 are set in the analog input mode, P10/ANI8/AMP1-/SCK10, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**
**4. RESET/P125 immediately after release of reset is set in the external reset input.**

### 1.3.4 78K0/KC2-L

<R> **(1) 40-pin plastic WQFN (6x6) (1/2)**




**Note** μPD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**Cautions 1. Leave the IC (Internally Connected) pin open.**

**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI6/P26 are set in the analog input mode, P10/ANI8/AMP1-/SCK10, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**4. RESET/P125 immediately after release of reset is set in the external reset input.**

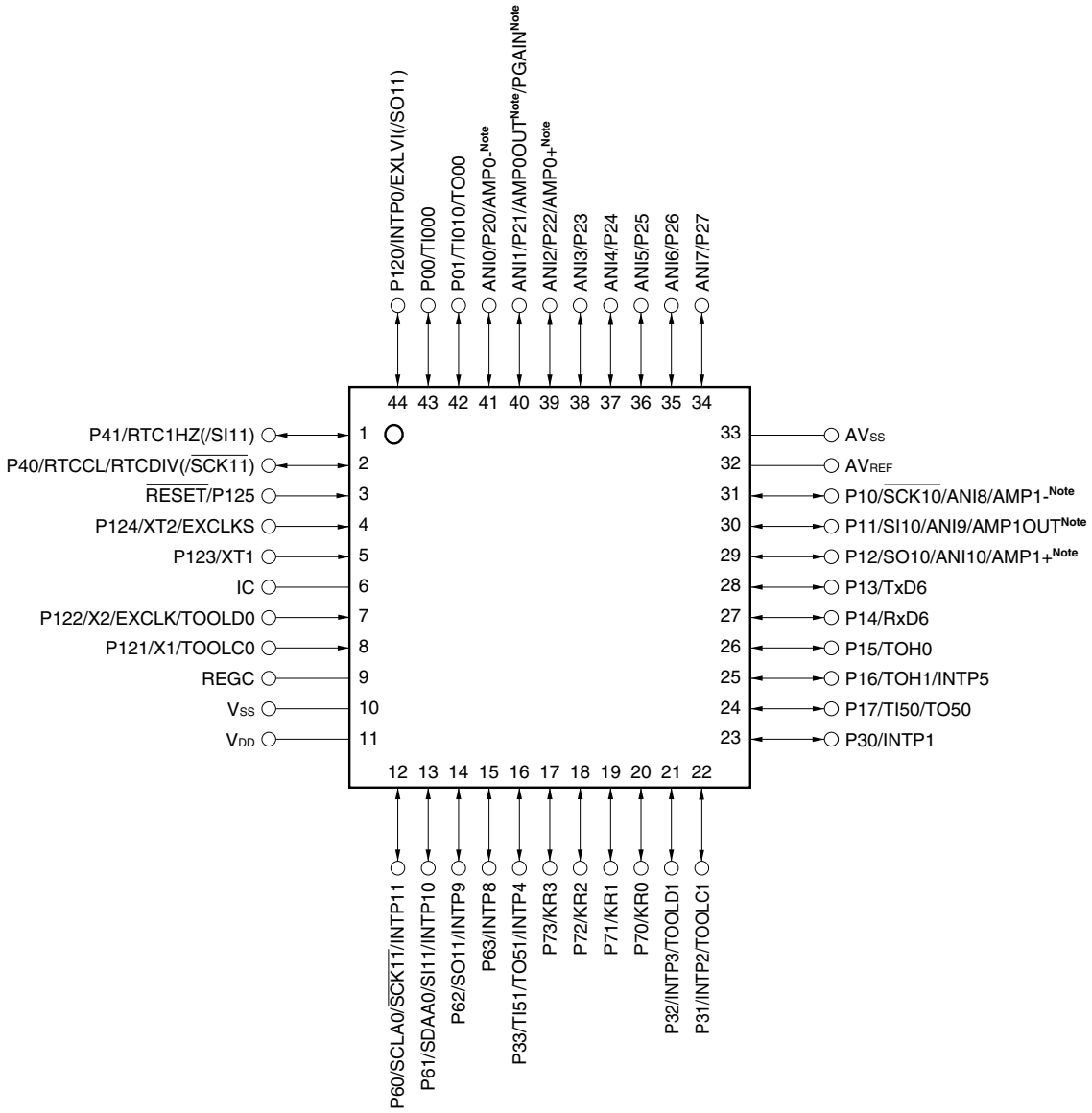
**5. Set P40, P41, and P63 to output mode (PM40 = PM41 = PM63 = 0) by using software after release of reset.**

<R> **(1) 40-pin plastic WQFN (6x6) (2/2)**

| | | | |
|---|---|---|---|
| AMP0-[Note], AMP0+[Note], | : Amplifier Input | REGC | : Regulator Capacitance |
| AMP1-[Note], AMP1+[Note], | | $\overline{\text{RESET}}$ | : Reset |
| AMP0OUT[Note], | : Amplifier Output | RxD6 | : Receive Data |
| AMP1OUT[Note] | | SCLA0, $\overline{\text{SCK10}}$, $\overline{\text{SCK11}}$ | : Serial Clock Input/Output |
| PGAIN[Note] | : Programmable Gain Amplifier Input | SDAA0 | : Serial Data Input/Output |
| | | SI10, SI11 | : Serial Data Input |
| ANI0 to ANI6, ANI8 to ANI10 | : Analog Input | SO10, SO11 | : Serial Data Output |
| $AV_{REF}$ | : Analog Reference Voltage | TI000, TI010, TI50, TI51 | : Timer Input |
| | | TO00, TO50, TO51, TOH0, TOH1 | : Timer Output |
| $AV_{SS}$ | : Analog Ground | | |
| EXCLK | : External Clock Input (Main System Clock) | TOOLC0, TOOLC1 | : Clock Input for Tool |
| | | TOOLD0, TOOLD1 | : Data Input/Output for Tool |
| EXCLKS | : External Clock Input (Subsystem Clock) | TxD6 | : Transmit Data |
| | | $V_{DD}$ | : Power Supply |
| EXLVI | : External potential Input for Low-voltage detector | $V_{SS}$ | : Ground |
| | | X1, X2 | : Crystal Oscillator (Main System Clock) |
| IC | : Internally Connected | | |
| INTP0 to INTP5, INTP9 to INTP11 | : External Interrupt Input | XT1, XT2 | : Crystal Oscillator (Subsystem Clock) |
| KR0 to KR3 | : Key Return | | |
| P00, P01 | : Port 0 | | |
| P10 to P17 | : Port 1 | | |
| P20 to P26 | : Port 2 | | |
| P30 to P33 | : Port 3 | | |
| P60 to P62 | : Port 6 | | |
| P70 to P73 | : Port 7 | | |
| P120 to P125 | : Port 12 | | |

**Note** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**(2) 44-pin plastic LQFP (10x10) (1/2)**




**Note** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**Cautions 1. Leave the IC (Internally Connected) pin open.**

**2. Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI7/P27 are set in the analog input mode, P10/ANI8/AMP1-/$\overline{\text{SCK10}}$, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**4. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

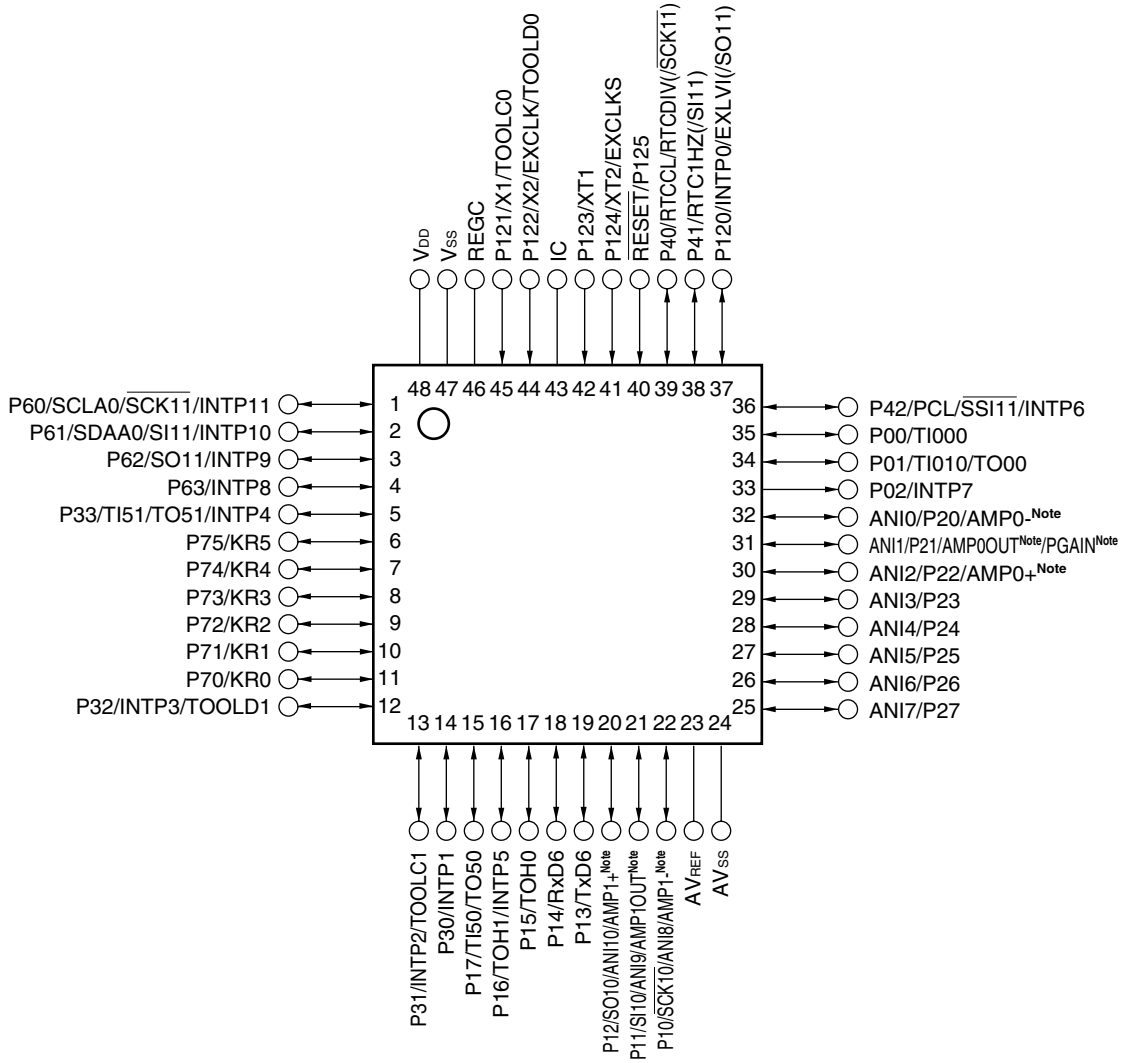
**Remark** Functions in parentheses ( ) in the figure above can be assigned by setting the port alternate switch control register (MUXSEL).

### (2) 44-pin plastic LQFP (10x10) (2/2)

| | | | |
|---|---|---|---|
| AMP0-[Note], AMP0+[Note], AMP1-[Note], AMP1+[Note] : | Amplifier Input | REGC: | Regulator Capacitance |
| AMP0OUT[Note], AMP1OUT[Note] : | Amplifier Output | $\overline{\text{RESET}}$ : | Reset |
| PGAIN[Note] : | Programmable Gain Amplifier Input | RTC1HZ : | Real-time Counter Correction Clock (1 Hz) Output |
| ANI0 to ANI10 : | Analog Input | RTCCL : | Real-time Counter Clock (32 kHz Original Oscillation) Output |
| $AV_{REF}$ : | Analog Reference Voltage | RTCDIV : | Real-time Counter Clock (32 kHz Divided Frequency) Output |
| $AV_{SS}$ : | Analog Ground | RxD6 : | Receive Data |
| EXCLK : | External Clock Input (Main System Clock) | SCLA0, $\overline{\text{SCK10}}$, $\overline{\text{SCK11}}$ : | Serial Clock Input/Output |
| EXCLKS : | External Clock Input (Subsystem Clock) | SDAA0 : | Serial Data Input/Output |
| EXLVI : | External potential Input for Low-voltage detector | SI10, SI11 : | Serial Data Input |
| IC : | Internally Connected | SO10, SO11 : | Serial Data Output |
| INTP0 to INTP5, INTP8 to INTP11 : | External Interrupt Input | TI000, TI010, TI50, TI51 : TO00, TO50, TO51, TOH0, TOH1 : | Timer Input Timer Output |
| KR0 to KR3 : | Key Return | TOOLC0, TOOLC1 : | Clock Input for Tool |
| P00, P01 : | Port 0 | TOOLD0, TOOLD1 : | Data Input/Output for Tool |
| P10 to P17 : | Port 1 | TxD6 : | Transmit Data |
| P20 to P27 : | Port 2 | $V_{DD}$ : | Power Supply |
| P30 to P33 : | Port 3 | $V_{SS}$ : | Ground |
| P40, P41 : | Port 4 | X1, X2 : | Crystal Oscillator (Main System Clock) |
| P60 to P63 : | Port 6 | | |
| P70 to P73 : | Port 7 | XT1, XT2 : | Crystal Oscillator (Subsystem Clock) |
| P120 to P125 : | Port 12 | | |

**Note** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**(3) 48-pin plastic LQFP (fine pitch) (7x7) (1/2)**




**Note** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**Cautions 1. Leave the IC (Internally Connected) pin open.**

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

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI7/P27 are set in the analog input mode, P10/ANI8/AMP1-/SCK10, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**4. RESET/P125 immediately after release of reset is set in the external reset input.**

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

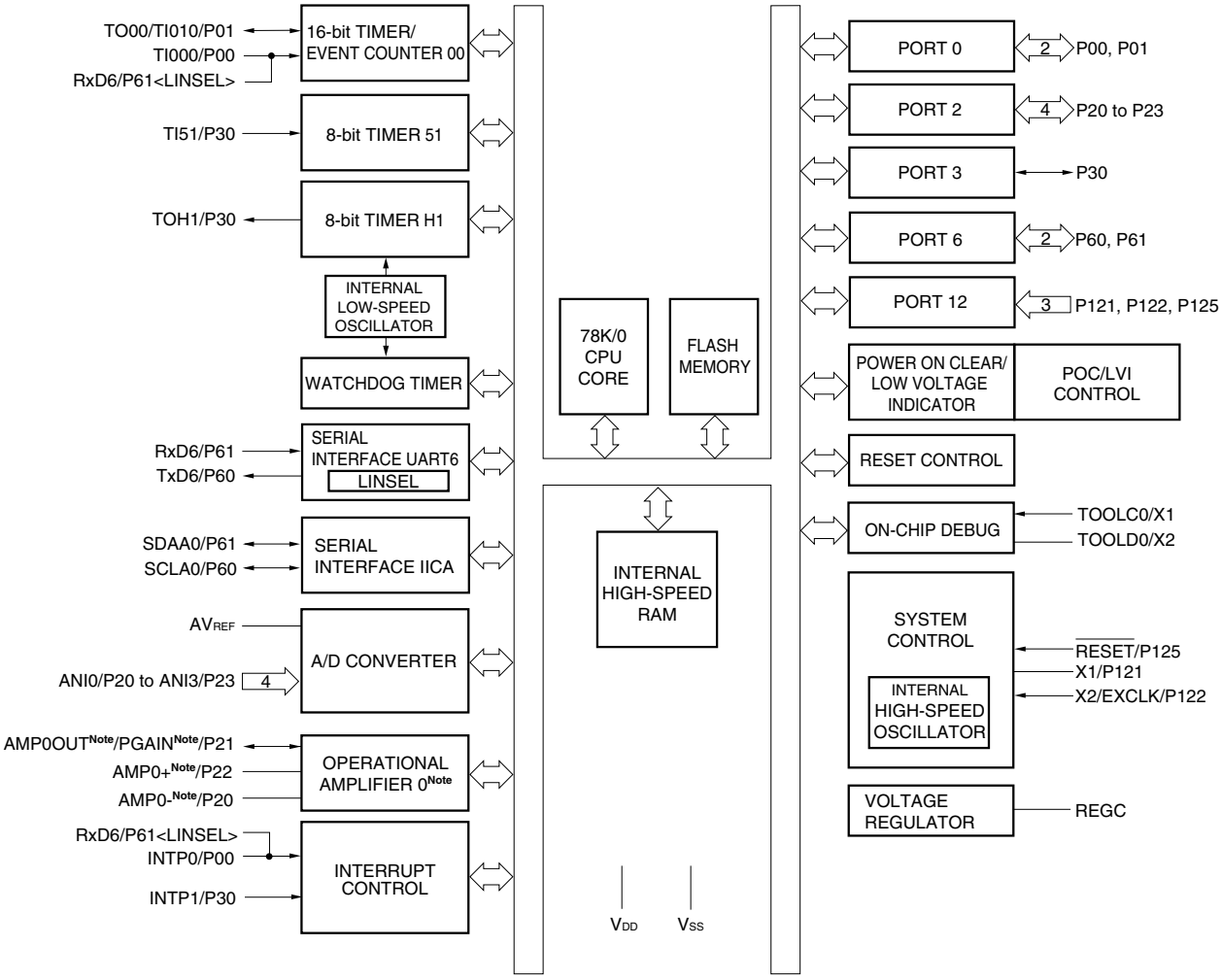
### (3) 48-pin plastic LQFP (fine pitch) (7x7) (2/2)

| | | | |
|---|---|---|---|
| AMP0-[Note], AMP0+[Note], AMP1-[Note], AMP1+[Note] : | Amplifier Input | REGC : | Regulator Capacitance |
| AMP0OUT[Note], AMP1OUT[Note] : | Amplifier Output | $\overline{\text{RESET}}$ : | Reset |
| PGAIN[Note] : | Programmable Gain Amplifier Input | RTC1HZ : | Real-time Counter Correction Clock (1 Hz) Output |
| ANI0-ANI10 : | Analog Input | RTCCL : | Real-time Counter Clock (32 kHz Original Oscillation) Output |
| $AV_{REF}$ : | Analog Reference Voltage | RTCDIV : | Real-time Counter Clock (32 kHz Divided Frequency) Output |
| $AV_{SS}$ : | Analog Ground | | |
| EXCLK : | External Clock Input (Main System Clock) | RxD6 : | Receive Data |
| EXCLKS : | External Clock Input (Subsystem Clock) | SCLA0, $\overline{\text{SCK10}}$, $\overline{\text{SCK11}}$ : | Serial Clock Input/Output |
| | | SDAA0 : | Serial Data Input/Output |
| EXLVI : | External potential Input for Low-voltage detector | SI10, SI11 : | Serial Data Input |
| | | SO10, SO11 : | Serial Data Output |
| IC : | Internally Connected | $\overline{\text{SSI11}}$ : | Serial Interface Chip Select Input |
| INTP0 to INTP11 : | External Interrupt Input | TI000, TI010, TI50, TI51 : | Timer Input |
| KR0 to KR5 : | Key Return | TO00, TO50, TO51, TOH0, TOH1 : | Timer Output |
| P00 to P02 : | Port 0 | TOOLC0, TOOLC1 : | Clock Input for Tool |
| P10 to P17 : | Port 1 | TOOLD0, TOOLD1 : | Data Input/Output for Tool |
| P20 to P27 : | Port 2 | TxD6 : | Transmit Data |
| P30 to P33 : | Port 3 | $V_{DD}$ : | Power Supply |
| P40 to P42 : | Port 4 | $V_{SS}$ : | Ground |
| P60 to P63 : | Port 6 | X1, X2 : | Crystal Oscillator (Main System Clock) |
| P70 to P75 : | Port 7 | | |
| P120 to P125 : | Port 12 | XT1, XT2 : | Crystal Oscillator (Subsystem Clock) |
| PCL : | Programmble Clock Output | | |

**Note** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

## 1.4 Block Diagram

### 1.4.1 78K0/KY2-L




**Note** $\mu$PD78F0555, 78F0556, 78F0557 (products with operational amplifier) only

**Cautions 1. VSS functions alternately as the ground potential of the A/D converter. Be sure to connect VSS to a stabilized GND (= 0 V).**

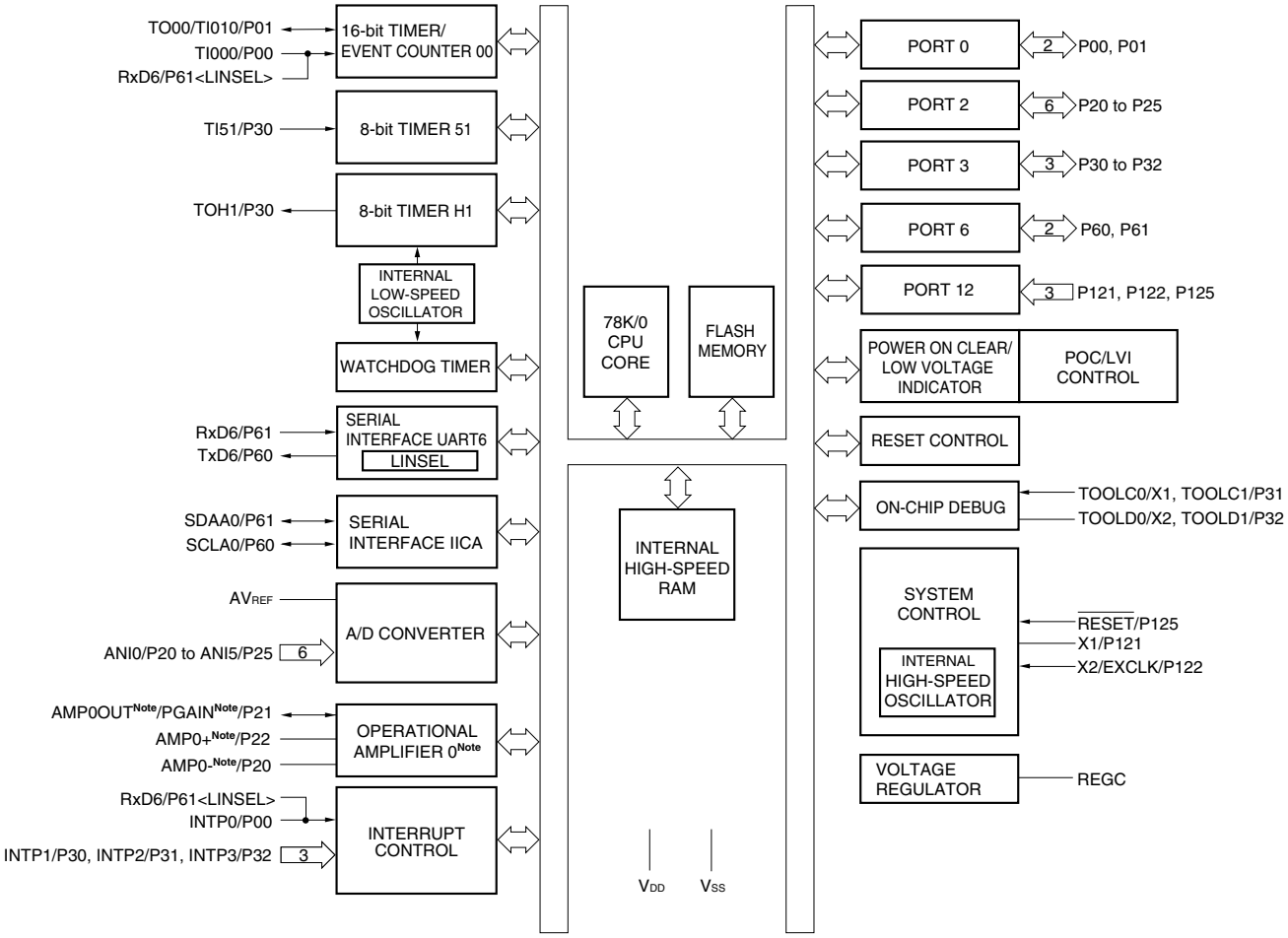
**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 $\mu$F).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 are set in the analog input mode after release of reset.**

**4. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

### 1.4.2 78K0/KA2-L

#### (1) 20-pin products




**Note** μPD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

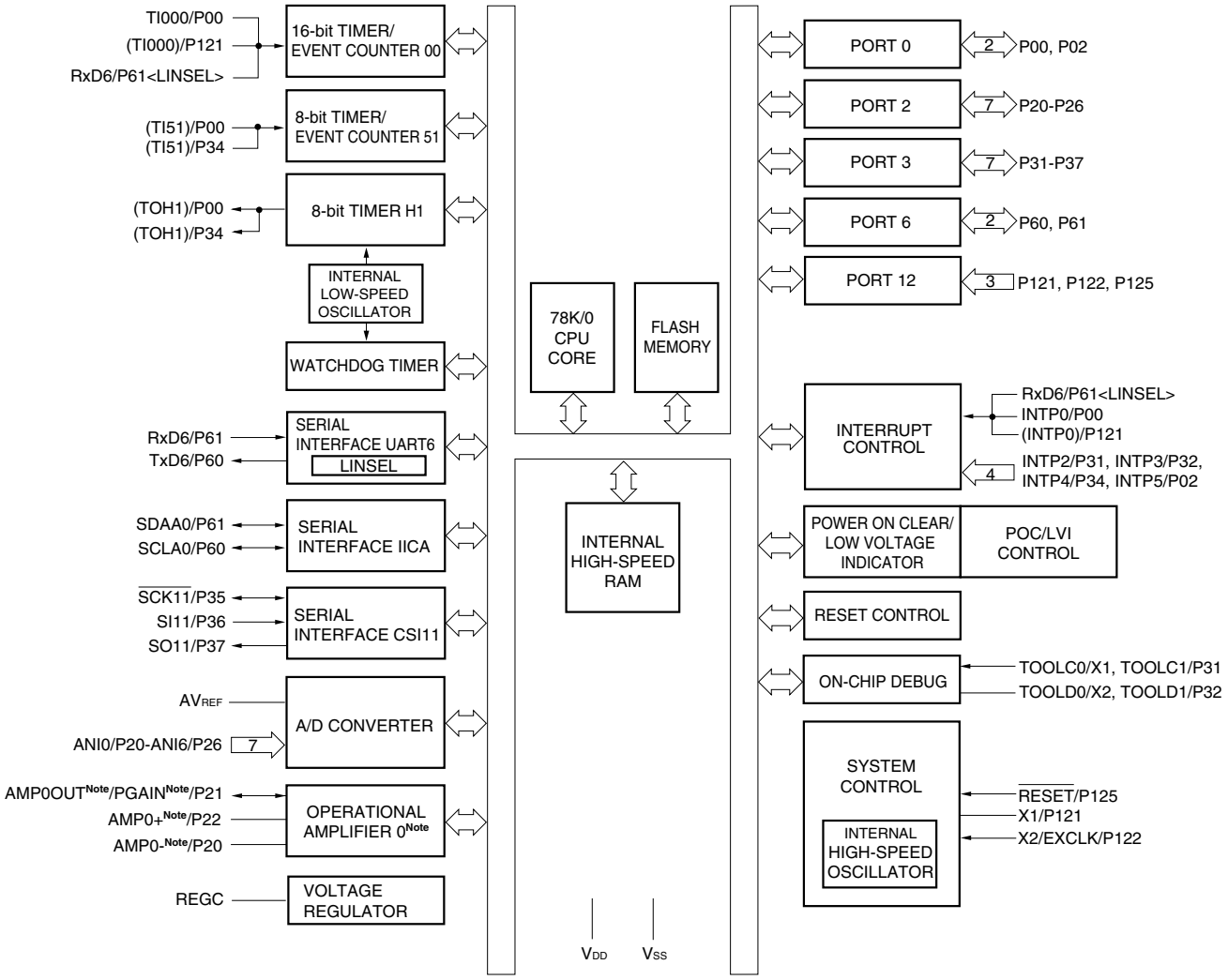
**Cautions 1. VSS functions alternately as the ground potential of the A/D converter. Be sure to connect VSS to a stabilized GND (= 0 V).**

**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI5/P25 are set in the analog input mode after release of reset.**

**4. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

<R> **(2) 25-pin products**




**Note** μPD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

**Cautions 1. VSS functions alternately as the ground potential of the A/D converter. Be sure to connect VSS to a stabilized GND (= 0 V).**

**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).**

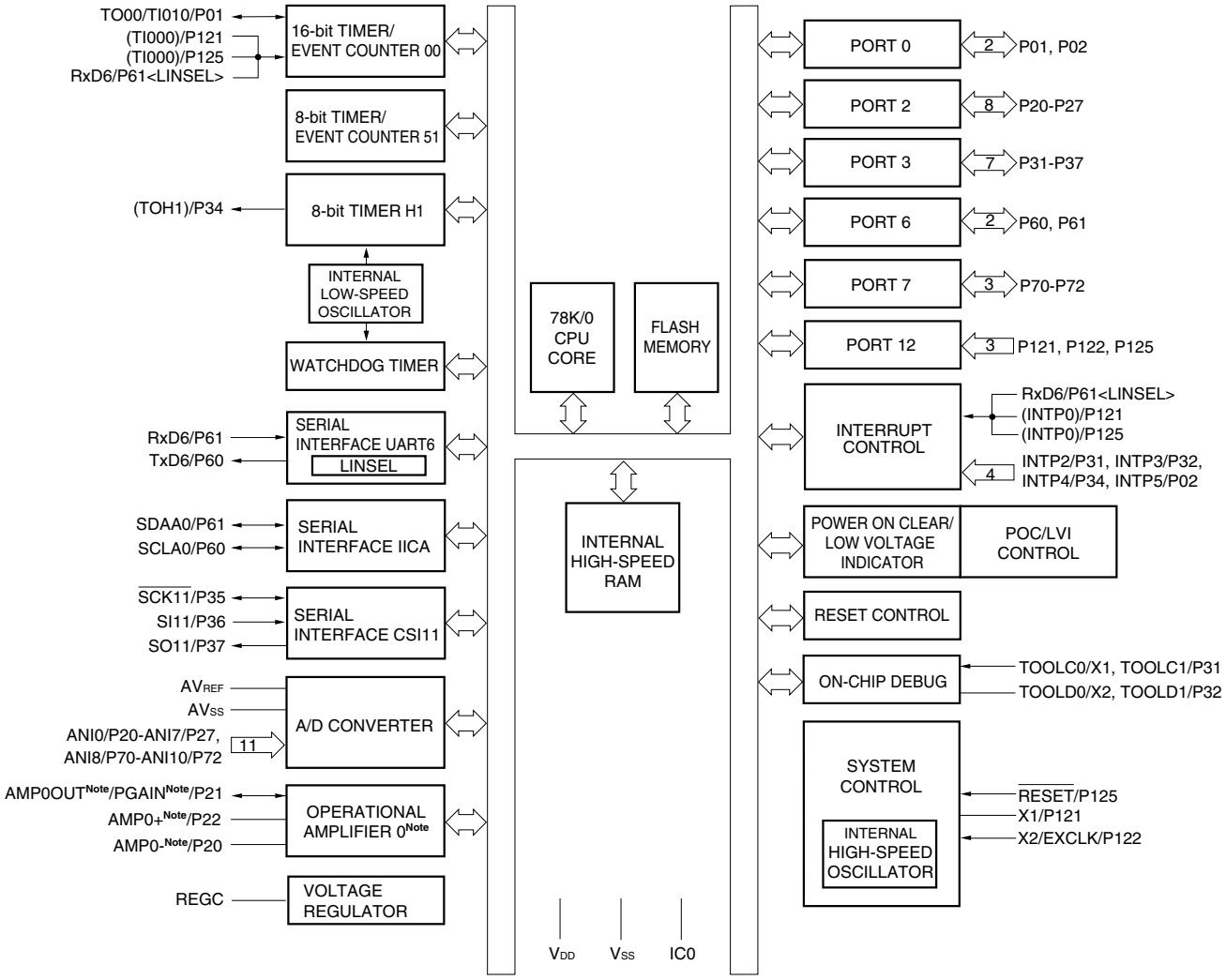
**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI6/P26 are set in the analog input mode after release of reset.**

**4. RESET/P125 immediately after release of reset is set in the external reset input.**

**5. Set P30 and P01 to output mode (PM30 = PM01 = 0) by using software after release of reset.**

**Remark** Functions in parentheses ( ) in the figure above can be assigned by setting the port alternate switch control register (MUXSEL).

<R> **(3) 32-pin products**




**Note** $\mu$PD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

**Cautions 1. Connect directly IC0 (Internally Connected) to $V_{SS}$.**

**2. $V_{SS}$ functions alternately as the ground potential of the A/D converter. Be sure to connect $V_{SS}$ to a stabilized GND (= 0 V).**

**3. Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F).**

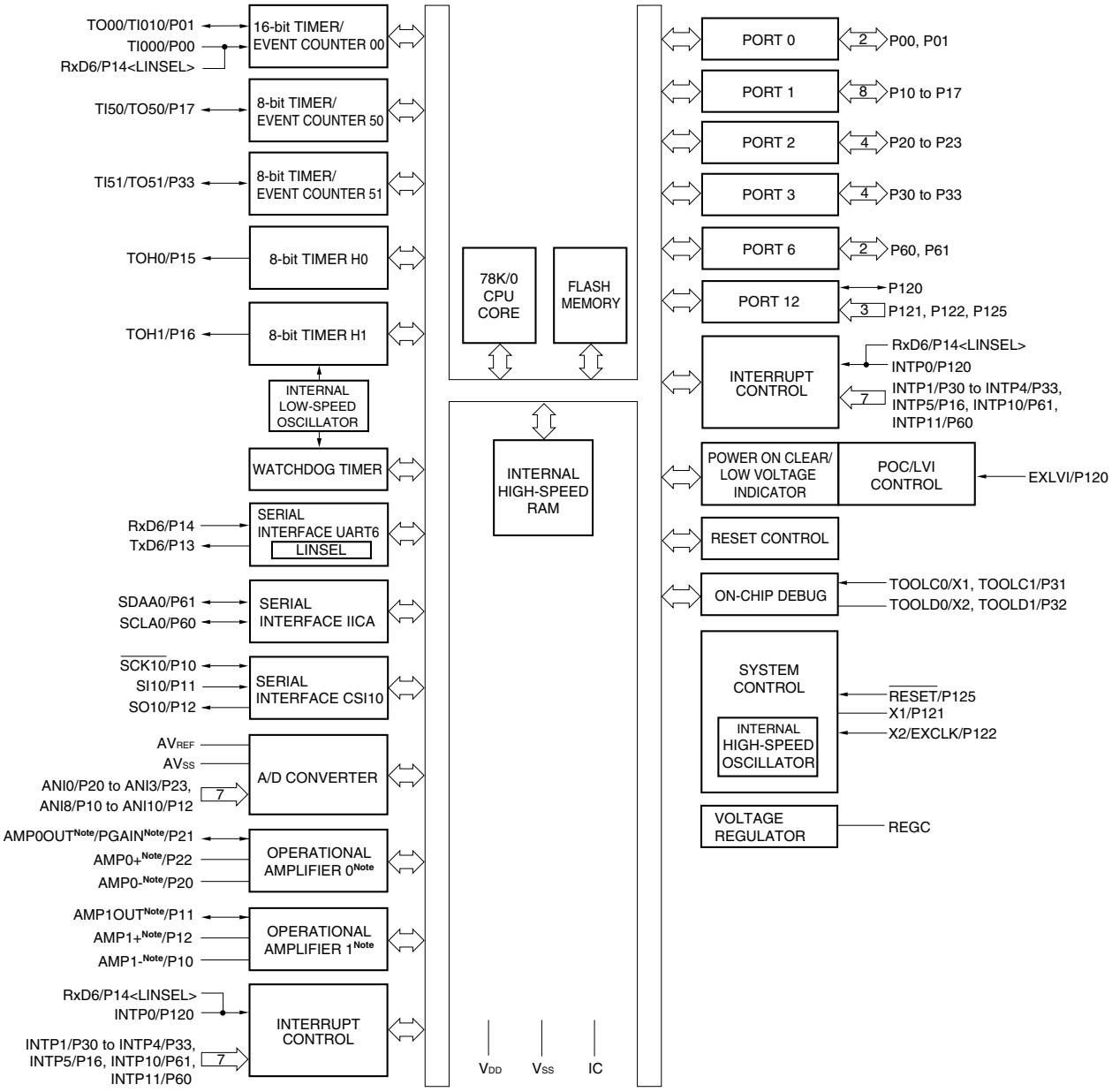
**4. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, ANI3/P23 to ANI7/P27, and ANI8/P70 to ANI10/P72 are set in the analog input mode after release of reset.**

**5. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

**6. Set P30 to output mode (PM30 = 0) by using software after release of reset.**

**Remark** Functions in parentheses ( ) in the figure above can be assigned by setting the port alternate switch control register (MUXSEL).

### 1.4.3 78K0/KB2-L




**Note** $\mu$PD78F0576, 78F0577, 78F0578 (products with operational amplifier) only

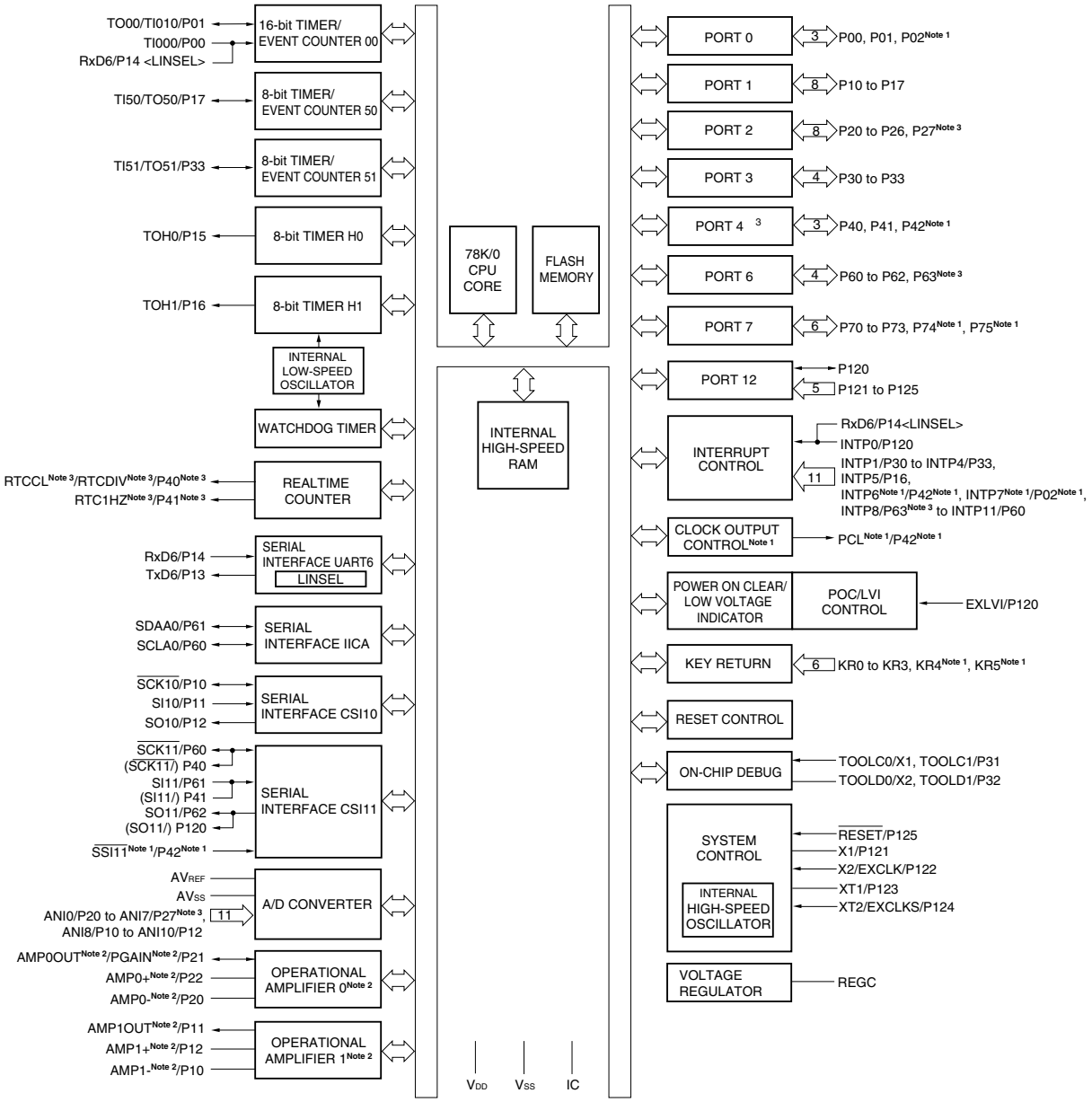
**Cautions 1. Leave the IC (Internally Connected) pin open.**

**2. Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 are set in the analog input mode, P10/ANI8/AMP1-/$\overline{\text{SCK10}}$, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**4. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.**

### <R> 1.4.4 78K0/KC2-L




**Notes 1.** 48-pin products only

**2.** μPD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**Cautions 1. Leave the IC (Internally Connected) pin open.**

**2. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).**

**3. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI7/P27 are set in the analog input mode, P10/ANI8/AMP1-/SCK10, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**4. RESET/P125 immediately after release of reset is set in the external reset input.**

**5. For 40-pin products, set P40, P41, and P63 to output mode (PM40 = PM41 = PM63 = 0) by using software after release of reset.**

**Remark** Functions in parentheses ( ) in the figure above can be assigned by setting the port alternate switch control register (MUXSEL).

## <R>1.5 Outline of Functions

(1/2)

| Item | | | 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| Internal memory | | Flash memory (self-programming supported ) | 4 KB to 16 KB | | | | 8 KB to 32 KB | | | |
| | | High-Speed RAM | 384 bytes to 768 bytes | | | | 512 bytes to 1 KB | | | |
| Memory space | | | 64 KB | | | | | | | |
| Clock | Main | High-speed system (crystal/ceramic oscillation, external clock input) | 1 to 10 MHz: $V_{DD}$ = 2.7 to 5.5 V/1 to 5 MHz: $V_{DD}$ = 1.8 to 5.5 V | | | | | | | |
| | | Internal high-speed oscillation | 4 MHz ± 2 % ($T_A$ = –20 to +70°C), or 8 MHz ± 3 % ($T_A$ = –40 to +85°C): $V_{DD}$ = 1.8 to 5.5 V | | | | | | | |
| | Subsystem (crystal oscillation, external clock input) | | – | | | | | 32.768 kHz (TYP.): $V_{DD}$ = 1.8 to 5.5 V | | |
| | Internal low-speed oscillation | | 30 kHz ± 10 %: $V_{DD}$ = 2.7 to 5.5 V, 30 kHz ± 15 %: $V_{DD}$ = 1.8 to 5.5 V | | | | | | | |
| General-purpose registers | | | 8 bits × 32 registers (8 bits × 8 registers × 4 banks) | | | | | | | |
| Instruction set | | | • 8-bit operation, 16-bit operation<br>• Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits)<br>• Bit manipulate (set, reset, test, and Boolean operation)<br>• BCD adjust, etc. | | | | | | | |
| I/O ports (total) | | | 12 | 16 | 21 | 25 | 24 | 34 | 38 | 42 |
| | CMOS I/O | | 9 | 13 | 18 | 22 | 21 | 29 | 33 | 37 |
| | CMOS input | | 3 | 3 | 3 | 3 | 3 | 5 | 5 | 5 |
| Timer | 16 bits (TM0) | | 1 ch (PPG output: 1, capture input: 2) | | | | | | | |
| | 8 bits (TM5) | | 1 ch | | | | 2 ch (PWM output: 2) | | | |
| | 8 bits (TMH) | | 1 ch (PWM output: 1) | | | | 2 ch (PWM output: 2) | | | |
| | Watchdog (WDT) | | 1 ch | | | | | | | |
| | Real-time counter | | – | | | | | 1 ch (RTC output : None) | 1 ch (RTC output: 2) | |
| Clock output | | | – | | | | | – | – | 1 |

(2/2)

| Item | | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | | | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| Serial interface | UART | 1 ch | | | | | | | |
| | IICA | 1 ch | | | | | | | |
| | CSI | – | – | 1 ch (CSI11[Note]) | | 1 ch (CSI10) | 2 ch (CSI10, CSI11[Note]) | | |
| 10-bit A/D converter ($AV_{REF}$ = 1.8 to 5.5 V) | | 4 ch | 6 ch | 7 ch | 11 ch | 7 ch | 10 ch | 11 ch | 11 ch |
| Operational amplifier (Products with operational amplifier) | | 1 ch ($V_{DD}$ = 2.2 to 5.5 v) | | | | 2 ch ($V_{DD}$ = 2.2 to 5.5 v) | | | |
| Vectored interrupt Internal sources | External | 2 | 4 | 5 | 5 | 8 | 10 | 11 | 13 |
| | Internal | 10 | 10 | 11 | 11 | 13 | 17 | | |
| Key interrupt | | – | | | | | 4 | 4 | 6 |
| Reset | | • Reset using $\overline{\text{RESET}}$ pin<br>• Internal reset by watchdog timer<br>• Internal reset by power-on-clear<br>• Internal reset by low-voltage detector | | | | | | | |
| On-chip debug function | | Provided | | | | | | | |
| Power supply voltage | | $V_{DD}$ = 1.8 to 5.5 V | | | | | | | |
| Operating ambient temperature | | $T_A$ = –40 to +85°C | | | | | | | |
| Package | | 16-pin plastic SSOP (5.72 mm (225)) | • 20-pin plastic SSOP (7.62 mm (300))<br>• 25-pin plastic FLGA (3x3)<br>• 32-pin plastic WQFN (5x5) | | | 30-pin plastic SSOP (7.62 mm (300)) | • 40-pin plastic WQFN (6x6)<br>• 44-pin plastic LQFP (10x10)<br>• 48-pin plastic LQFP (fine pitch) (7x7) | | |

**Note** The 78K0/KA2-L (25-pin and 32-pin products) and 78K0/KC2-L (48-pin products) can be controlled by an enabled signal, when using CSI11 in the slave mode.

# CHAPTER 2 PIN FUNCTIONS

## 2.1 Pin Function List

There are two types of pin I/O buffer power supplies: $AV_{REF}$ and $V_{DD}$. The relationship between these power supplies and the pins is shown below.

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

| Power Supply | Corresponding Pins |
|---|---|
| $AV_{REF}$ | P20 to P27[Note] |
| $V_{DD}$ | Pins other than P20 to P27[Note] |

**Note** 78K0/KY2-L: P20 to P23
78K0/KA2-L (20 pins): P20 to P25
<R> 78K0/KA2-L (25 pins): P20 to P26
<R> 78K0/KA2-L (32 pins): P20 to P27, P70 to P72
78K0/KB2-L: P20 to P23
<R> 78K0/KC2-L (40 pins): P20 to P26
78K0/KC2-L (44 pins, 48 pins): P20 to P27

### 2.1.1 78K0/KY2-L

#### (1) Port functions: 78K0/KY2-L

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000/INTP0 |
| P01 | | | | TO00/TI010 |
| P20 | I/O | Port 2.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0-Note |
| P21 | | | | ANI1/AMP0OUTNote/<br>PGAINNote |
| P22 | | | | ANI2/AMP0+Note |
| P23 | | | | ANI3 |
| P30 | I/O | Port 3.<br>1-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TOH1/TI51/INTP1 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/TxD6 |
| P61 | | | | SDAA0/RxD6 |
| P121 | Input | Port 12.<br>3-bit input-only port.<br>For only P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| <R> P125 | | | Reset input | $\overline{\text{RESET}}$ |

**Note** $\mu$PD78F0555, 78F0556, and 78F0557 (products with operational amplifier) only

### (2) Non-port functions : 78K0/KY2-L

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| ANI0 | Input | A/D converter analog input | Analog input | P20/AMP0-Note |
| ANI1 | | | | P21/AMP0OUTNote / PGAINNote |
| ANI2 | | | | P22/AMP0+Note |
| ANI3 | | | | P23 |
| AMP0-Note | Input | Operational amplifier 0 input | Analog input | P20/ANI0 |
| AMP0+Note | | | | P22/ANI2 |
| AMP0OUTNote | Output | Operational amplifier 0 output | | P21/ANI1/PGAINNote |
| PGAINNote | Input | PGA (programmable gain amplifier) input | Analog input | P21/ANI1/ AMP0OUTNote |
| INTP0 | Input | External interrupt request input for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified | Input port | P00/TI000 |
| INTP1 | | | | P30/TOH1/TI51 |
| REGC | – | Connecting regulator output (2.0 V/2.4 V) stabilization capacitance for internal operation. Connect to $V_{SS}$ via a capacitor (0.47 to 1 μF). | – | – |
| <R> $\overline{RESET}$ | Input | System reset input | Reset input | P125 |
| RxD6 | Input | Serial data input to UART6 | Input port | P61/SDAA0 |
| TxD6 | Output | Serial data output from UART6 | | P60/SCLA0 |
| SCLA0 | I/O | Clock input/output for $I^2C$ | Input port | P60/TxD6 |
| SDAA0 | | Serial data I/O for $I^2C$ | | P61/RxD6 |
| TI000 | Input | External count clock input to 16-bit timer/event counter 00 Capture trigger input to capture registers (CR000, CR010) of 16-bit timer/event counter 00 | Input port | P00/INTP0 |
| TI010 | | Capture trigger input to capture register (CR000) of 16-bit timer/event counter 00 | | P01/TO00 |
| TI51 | Input | External count clock input to 8-bit timer/event counter 51 | Input port | P30/TOH1/INTP1 |
| TO00 | Output | 16-bit timer/event counter 00 output | Input port | P01/TI010 |
| TOH1 | Output | 8-bit timer H1 output | Input port | P30/TI51/INTP1 |
| X1 | – | Connecting resonator for main system clock | Input port | P121/TOOLC0 |
| X2 | | | | P122/EXCLK/TOOLD0 |
| EXCLK | Input | External clock input for main system clock | Input port | P122/X2/TOOLD0 |
| $V_{DD}$ | – | Positive power supply for pins other than port 2 | – | – |
| $AV_{REF}$ | | A/D converter reference voltage input and positive power supply for port 2 and A/D converter | | |
| $V_{SS}$ | – | Ground potential | – | – |
| TOOLC0 | Input | Clock input for flash memory programmer/on-chip debugger | Input port | P121/X1 |
| TOOLD0 | I/O | Data I/O for flash memory programmer/on-chip debugger | | P122/X2/EXCLK |

**Note** μPD78F0555, 78F0556, and 78F0557 (products with operational amplifier) only

### 2.1.2 78K0/KA2-L

#### (1) Port functions: 78K0/KA2-L (20 pins)

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000/INTP0 |
| P01 | | | | TO00/TI010 |
| P20 | I/O | Port 2.<br>6-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0-Note |
| P21 | | | | ANI1/AMP0OUTNote/PGAINNote |
| P22 | | | | ANI2/AMP0+Note |
| P23 | | | | ANI3 |
| P24 | | | | ANI4 |
| P25 | | | | ANI5 |
| P30 | I/O | Port 3.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TOH1/TI51/INTP1 |
| P31 | | | | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/TxD6 |
| P61 | | | | SDAA0/RxD6 |
| P121 | Input | Port 12.<br>3-bit input-only port.<br>For only P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| <R> P125 | | | Reset input | $\overline{\text{RESET}}$ |

**Note** μPD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

#### (2) Non-port functions : 78K0/KA2-L (20 pins) (1/2)

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| ANI0 | Input | A/D converter analog input | Analog input | P20/AMP0-Note |
| ANI1 | | | | P21/AMP0OUTNote/PGAINNote |
| ANI2 | | | | P22/AMP0+Note |
| ANI3 | | | | P23 |
| ANI4 | | | | P24 |
| ANI5 | | | | P25 |

**Note** μPD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

### (2) Non-port functions: 78K0/KA2-L (20 pins) (2/2)

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| AMP0-[Note] | Input | Operational amplifier 0 input | Analog input | P20/ANI0 |
| AMP0+[Note] | | | | P22/ANI2 |
| AMP0OUT[Note] | Output | Operational amplifier 0 output | | P21/ANI1/PGAIN[Note] |
| PGAIN[Note] | Input | PGA (programmable gain amplifier) input | Analog input | P21/ANI1/ AMP0OUT[Note] |
| INTP0 | Input | External interrupt request input for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified | Input port | P00/TI000 |
| INTP1 | | | | P30/TOH1/TI51 |
| INTP2 | | | | P31/TOOLC1 |
| INTP3 | | | | P32/TOOLD1 |
| REGC | – | Connecting regulator output (2.0 V/2.4 V) stabilization capacitance for internal operation. Connect to $V_{SS}$ via a capacitor (0.47 to 1 μF). | – | – |
| <R> $\overline{\text{RESET}}$ | Input | System reset input | Reset input | P125 |
| RxD6 | Input | Serial data input to UART6 | Input port | P61/SDAA0 |
| TxD6 | Output | Serial data output from UART6 | | P60/SCLA0 |
| SCLA0 | I/O | Clock input/output for $I^2C$ | Input port | P60/TxD6 |
| SDAA0 | | Serial data I/O for $I^2C$ | | P61/RxD6 |
| TI000 | Input | External count clock input to 16-bit timer/event counter 00 Capture trigger input to capture registers (CR000, CR010) of 16-bit timer/event counter 00 | Input port | P00/INTP0 |
| TI010 | | Capture trigger input to capture register (CR000) of 16-bit timer/event counter 00 | | P01/TO00 |
| TI51 | Input | External count clock input to 8-bit timer/event counter 51 | Input port | P30/TOH1/INTP1 |
| TO00 | Output | 16-bit timer/event counter 00 output | Input port | P01/TI010 |
| TOH1 | Output | 8-bit timer H1 output | Input port | P30/TI51/INTP1 |
| X1 | – | Connecting resonator for main system clock | Input port | P121/TOOLC0 |
| X2 | | | | P122/EXCLK/TOOLD0 |
| EXCLK | Input | External clock input for main system clock | Input port | P122/X2/TOOLD0 |
| $V_{DD}$ | – | Positive power supply for pins other than port 2 | – | – |
| $AV_{REF}$ | | A/D converter reference voltage input and positive power supply for port 2 and A/D converter | | |
| $V_{SS}$ | – | Ground potential | – | – |
| TOOLC0 | Input | Clock input for flash memory programmer/on-chip debugger | Input port | P121/X1 |
| TOOLC1 | | | | P31/INTP2 |
| TOOLD0 | I/O | Data I/O for flash memory programmer/on-chip debugger | | P122/X2/EXCLK |
| TOOLD1 | | | | P32/INTP3 |

**Note** μPD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

### <R> (3) Port functions: 78K0/KA2-L (25, 32 pins)

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 Note 1 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000 Note 1/INTP0 Note 1 (/TOH1) Note 1 (/TI51) Note 1 |
| P01 Note 2 | | | | TO00 Note 2/TI010 Note 2 |
| P02 | | | | $\overline{SSI11}$/INTP5 |
| P20 | I/O | Port 2.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0- Note 3 |
| P21 | | | | ANI1/AMP0OUT Note 3/ PGAIN Note 3 |
| P22 | | | | ANI2/AMP0+ Note 3 |
| P23 | | | | ANI3 |
| P24 | | | | ANI4 |
| P25 | | | | ANI5 |
| P26 | | | | ANI6 |
| P27 Note 2 | | | | ANI7 Note 2 |
| P31 | I/O | Port 3.<br>7-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P33 | | | | – |
| P34 | | | | INTP4(/TOH1) (/TI51) Note 1 |
| P35 | | | | $\overline{SCK11}$ |
| P36 | | | | SI11 |
| P37 | | | | SO11 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TxD6/SCLA0 |
| P61 | | | | RxD6/SDAA0 |
| P70 Note 2 | I/O | Port 7.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI8 Note 2 |
| P71 Note 2 | | | | ANI9 Note 2 |
| P72 Note 2 | | | | ANI10 Note 2 |
| P121 | Input | Port 12.<br>3-bit I/O port.<br>For only P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | X1/TOOLC0 (/TI000)(/INTP0) |
| P122 | | | | X2/EXCLK/ TOOLD0 |
| P125 | | | Reset input | $\overline{RESET}$(/TI000) Note 2 (/INTP0) Note 2 |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** μPD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

<R> **(4) Non-port functions: 78K0/KA2-L (25, 32 pins) (1/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| ANI0 | Input | A/D converter analog input | Analog input | P20/AMP0-Note 3 |
| ANI1 | | | | P21/AMP0OUTNote 3/ PGAINNote 3 |
| ANI2 | | | | P22/AMP0+Note 3 |
| ANI3 to ANI6 | | | | P23 to P26 |
| ANI7Note 2 | | | | P27Note 2 |
| ANI8Note 2 | | | | P70Note 2 |
| ANI9Note 2 | | | | P71Note 2 |
| ANI10Note 2 | | | | P72Note 2 |
| AMP0-Note 3 | Input | Operational amplifier 0 input | Analog input | P20/ANI0 |
| AMP0+Note 3 | | | | P22/ANI2 |
| AMP0OUTNote 3 | Output | Operational amplifier 0 output | Analog input | P21/ANI1/PGAINNote 3 |
| PGAINNote 3 | Input | PGA (programmable gain amplifier) input | Analog input | P21/ANI1/ AMP0OUTNote 3 |
| INTP0Note 1 | Input | External interrupt request input for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified | Input port | P00Note 1/TI000Note 1 (/TOH1)Note 1 (/TI51)Note 1 |
| (INTP0) | | | | P121/X1/TOOLC0 (/TI000) |
| (INTP0)Note 2 | | | | $\overline{\text{RESET}}$/P125 (/TI000)Note 2 |
| INTP2 | | | | P31/TOOLC1 |
| INTP3 | | | | P32/TOOLD1 |
| INTP4 | | | | P34/I(/TOH1)(/TI51)Note 1 |
| INTP5 | | | | P02/$\overline{\text{SSI11}}$ |
| REGC | – | Connecting regulator output (2.0 V/2.4 V) stabilization capacitance for internal operation. Connect to Vss via a capacitor (0.47 to 1 μF). | – | – |
| $\overline{\text{RESET}}$ | Input | System reset input | Reset input | P125(/TI000)Note 2 (/INTP0)Note 2 |
| RxD6 | Input | Serial data input to UART6 | Input port | P61/SDAA0 |
| TxD6 | Output | Serial data output from UART6 | | P60/SCLA0 |
| SCLA0 | I/O | Clock input/output for I²C | Input port | P60/TxD6 |
| SDAA0 | | Serial data I/O for I²C | | P61/RxD6 |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** μPD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

<R>**(4) Non-port functions: 78K0/KA2-L (25, 32 pins) (2/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| $\overline{SCK11}$ | I/O | Clock input/output for CSI10 | Input port | P35 |
| SI11 | Input | Serial data input to CSI10 | | P36 |
| SO11 | Output | Serial data output from CSI10 | | P37 |
| $\overline{SSI11}$ | Input | Chip select input to CSI11 | | P02/INTP5 |
| TI000 Note 1 | Input | External count clock input to 16-bit timer/event counter 00<br>Capture trigger input to capture registers (CR000, CR010) of 16-bit timer/event counter 00 | Input port | P00 Note 1/INTP0 Note 1 (/TOH1) Note 1 (/TI51) Note 1 |
| (TI000) | | | | P121/TOOLC0 (/INTP0) |
| (TI000) Note 2 | | | | $\overline{RESET}$/P125 (/INTP0) Note 2 |
| TI010 Note 2 | | Capture trigger input to capture register (CR000) of 16-bit timer/event counter 00 | | P01 Note 2/TO00 Note 2 |
| (TI50) Note 1 | Input | External count clock input to 8-bit timer/event counter 51 | Input port | P34/INTP4(/TOH1) |
| TO00 Note 2 | Output | 16-bit timer/event counter 00 output | Input port | P01 Note 2/TI010 Note 2 |
| (TOH1) | Output | 8-bit timer H1 output | Input port | P34/INTP4(/TI51) Note 1 |
| X1 | – | Connecting resonator for main system clock | Input port | P121/TOOLC0 |
| X2 | | | | P122/EXCLK/TOOLD0 |
| EXCLK | Input | External clock input for main system clock | Input port | P122/X2/TOOLD0 |
| $V_{DD}$ | – | Positive power supply for pins other than port 2 | – | – |
| $AV_{REF}$ | | A/D converter reference voltage input and positive power supply for port 2 and A/D converter | | |
| $V_{SS}$ | – | Ground potential. For 32-pin products, ground potential for pins other than port 2 | – | – |
| $AV_{SS}$ Note 2 | | Ground potential for port 2 and A/D converter | | |
| TOOLC0 | Input | Clock input for flash memory programmer/on-chip debugger | Input port | P121/X1 (/TI000) (/INTP0) |
| TOOLC1 | | | | P31/INTP2 |
| TOOLD0 | I/O | Data I/O for flash memory programmer/on-chip debugger | | P122/X2/EXCLK |
| TOOLD1 | | | | P32/INTP3 |
| IC0 Note 2 | – | Internally connected. Connect directly to $V_{SS}$. | – | – |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** $\mu$PD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

### 2.1.3 78K0/KB2-L

#### (1) Port functions: 78K0/KB2-L

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000 |
| P01 | | | | TI010/TO00 |
| P10 | I/O | Port 1.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | ANI8/AMP1-Note/$\overline{\text{SCK10}}$ |
| P11 | | | | ANI9/AMP1OUTNote/SI10 |
| P12 | | | | ANI10/AMP1+Note/SO10 |
| P13 | | | | TxD6 |
| P14 | | | | RxD6 |
| P15 | | | | TOH0 |
| P16 | | | | TOH1/INTP5 |
| P17 | | | | TI50/TO50 |
| P20 | I/O | Port 2.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0-Note |
| P21 | | | | ANI1/AMP0OUTNote/PGAINNote |
| P22 | | | | ANI2/AMP0+Note |
| P23 | | | | ANI3 |
| P30 | I/O | Port 3.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | INTP1 |
| P31 | | | | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P33 | | | | TI51/TO51/INTP4 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/INTP11 |
| P61 | | | | SDAA0/INTP10 |
| P120 | I/O | Port 12.<br>1-bit I/O port and 3-bit input port.<br>For only P120 and P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | EXLVI/INTP0 |
| P121 | Input | | | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| <R> P125 | | | Reset input | $\overline{\text{RESET}}$ |

**Note** $\mu$PD78F0576, 78F0577, and 78F0578 (products with operational amplifier) only

**(2) Non-port functions: 78K0/KB2-L (1/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| ANI0 | Input | A/D converter analog input | Analog input | P20/AMP0-Note |
| ANI1 | | | | P21/AMP0OUTNote/PGAINNote |
| ANI2 | | | | P22/AMP0+Note |
| ANI3 | | | | P23 |
| ANI8 | | | Input port | P10/$\overline{SCK10}$/AMP1-Note |
| ANI9 | | | | P11/SI10/AMP1OUTNote |
| ANI10 | | | | P12/SO10/AMP1+Note |
| AMP0-Note | Input | Operational amplifier 0 input | Analog input | P20/ANI0 |
| AMP0+Note | | | | P22/ANI2 |
| AMP1-Note | | Operational amplifier 1 input | Input port | P10/ANI8/$\overline{SCK10}$ |
| AMP1+Note | | | | P12/ANI10/SO10 |
| AMP0OUTNote | Output | Operational amplifier 0 output | Analog input | P21/ANI1/PGAINNote |
| AMP1OUTNote | | Operational amplifier 1 output | Input port | P11/ANI9/SI10 |
| PGAINNote | Input | PGA (programmable gain amplifier) input | Analog input | P21/ANI1/AMP0OUTNote |
| EXLVI | Input | Potential input for external low-voltage detection | Input port | P120/INTP0 |
| INTP0 | Input | External interrupt request input for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified | Input port | P120/EXLVI |
| INTP1 | | | | P30 |
| INTP2 | | | | P31/TOOLC1 |
| INTP3 | | | | P32/TOOLD1 |
| INTP4 | | | | P33/TI51/TO51 |
| INTP5 | | | | P16/TOH1 |
| INTP10 | | | | P61/SDAA0 |
| INTP11 | | | | P60/SCLA0 |
| REGC | – | Connecting regulator output (2.0 V/2.4 V) stabilization capacitance for internal operation. Connect to $V_{SS}$ via a capacitor (0.47 to 1 μF). | – | – |
| <R> $\overline{RESET}$ | Input | System reset input | Reset input | P125 |
| RxD6 | Input | Serial data input to UART6 | Input port | P14 |
| TxD6 | Output | Serial data output from UART6 | | P13 |
| SCLA0 | I/O | Clock input/output for $I^2C$ | Input port | P60/INTP11 |
| SDAA0 | | Serial data I/O for $I^2C$ | | P61/INTP10 |

**Note** μPD78F0576, 78F0577, and 78F0578 (products with operational amplifier) only

**(2) Non-port functions: 78K0/KB2-L (2/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| $\overline{SCK10}$ | I/O | Clock input/output for CSI10 | Input port | P10/ANI8/AMP1-Note |
| SI10 | Input | Serial data input to CSI10 | | P11/ANI9/AMP1OUTNote |
| SO10 | Output | Serial data output from CSI10 | | P12/ANI10/AMP1+Note |
| TI000 | Input | External count clock input to 16-bit timer/event counter 00<br>Capture trigger input to capture registers (CR000, CR010) of 16-bit timer/event counter 00 | Input port | P00 |
| TI010 | | Capture trigger input to capture register (CR000) of 16-bit timer/event counter 00 | | P01/TO00 |
| TI50 | Input | External count clock input to 8-bit timer/event counter 50 | Input port | P17/TO50 |
| TI51 | | External count clock input to 8-bit timer/event counter 51 | | P33/TO51/INTP4 |
| TO00 | Output | 16-bit timer/event counter 00 output | Input port | P01/TI010 |
| TO50 | Output | 8-bit timer/event counter 50 output | Input port | P17/TI50 |
| TO51 | | 8-bit timer/event counter 51 output | | P33/TI51/INTP4 |
| TOH0 | Output | 8-bit timer H0 output | Input port | P15 |
| TOH1 | | 8-bit timer H1 output | | P16/INTP5 |
| X1 | – | Connecting resonator for main system clock | Input port | P121/TOOLC0 |
| X2 | | | | P122/EXCLK/TOOLD0 |
| EXCLK | Input | External clock input for main system clock | Input port | P122/X2/TOOLD0 |
| $V_{DD}$ | – | Positive power supply for pins other than port 2 | – | – |
| $AV_{REF}$ | | A/D converter reference voltage input and positive power supply for port 2 and A/D converter | | |
| $V_{SS}$ | – | Ground potential for pins other than port 2 | – | – |
| $AV_{SS}$ | | Ground potential for port 2 and A/D converter | | |
| TOOLC0 | Input | Clock input for flash memory programmer/on-chip debugger | Input port | P121/X1 |
| TOOLC1 | | | | P31/INTP2 |
| TOOLD0 | I/O | Data I/O for flash memory programmer/on-chip debugger | | P122/X2/EXCLK |
| TOOLD1 | | | | P32/INTP3/TOH1 |
| IC | – | Internally connected. Leave open. | – | – |

**Note** μPD78F0576, 78F0577, and 78F0578 (products with operational amplifier) only

### 2.1.4 78K0/KC2-L

<R>**(1) Port functions: 78K0/KC2-L (1/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000 |
| P01 | | | | TI010/TO00 |
| P02 Note 1 | | | | INTP7 Note 1 |
| P10 | I/O | Port 1.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | ANI8/AMP1- Note 2 /$\overline{\text{SCK10}}$ |
| P11 | | | | ANI9/AMP1OUT Note 2 /SI10 |
| P12 | | | | ANI10/AMP1+ Note 2 /SO10 |
| P13 | | | | TxD6 |
| P14 | | | | RxD6 |
| P15 | | | | TOH0 |
| P16 | | | | TOH1/INTP5 |
| P17 | | | | TI50/TO50 |
| P20 | I/O | Port 2.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0- Note 2 |
| P21 | | | | ANI1/AMP0OUT Note 2 /PGAIN Note 2 |
| P22 | | | | ANI2/AMP0+ Note 2 |
| P23 | | | | ANI3 |
| P24 | | | | ANI4 |
| P25 | | | | ANI5 |
| P26 | | | | ANI6 |
| P27 Note 3 | | | | ANI7 Note 3 |
| P30 | I/O | Port 3.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | INTP1 |
| P31 | | | | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P33 | | | | TI51/TO51/INTP4 |
| P40 Note 3 | I/O | Port 4. Note 3<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | RTCCL Note 3 /RTCDIV Note 3 (/$\overline{\text{SCK11}}$) Note 3 |
| P41 Note 3 | | | | RTC1HZ Note 3 (/SI11) Note 3 |
| P42 Note 1 | | | | PCL Note 1 /$\overline{\text{SSI11}}$ Note 1 /INTP6 Note 1 |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, and 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

### (1) Port functions: 78K0/KC2-L (2/2)

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P60 | I/O | Port 6.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input of P60 and P61 can be set to SMBus input buffer in 1-bit units.<br>Output of P60 to P63 can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/$\overline{SCK11}$/ INTP11 |
| P61 | | | | SDAA0/SI11/INTP10 |
| P62 | | | | SO11/INTP9 |
| P63 Note 3 | | | | INTP8 Note 3 |
| P70 | I/O | Port 7.<br>6-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | KR0 |
| P71 | | | | KR1 |
| P72 | | | | KR2 |
| P73 | | | | KR3 |
| P74 Note 1 | | | | KR4 Note 1 |
| P75 Note 1 | | | | KR5 Note 1 |
| P120 | I/O | Port 12.<br>1-bit I/O port and 5-bit input port.<br>For only P120 and P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | EXLVI/INTP0 (/SO11) Note 3 |
| P121 | Input | | | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| P123 | | | | XT1 |
| P124 | | | | XT2/EXCLKS |
| P125 | | | Reset input | $\overline{RESET}$ |

### <R> (2) Non-port functions : 78K0/KC2-L (1/4)

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| ANI0 | Input | A/D converter analog input | Analog input | P20/AMP0- Note 2 |
| ANI1 | | | | P21/AMP0OUT Note 2 / PGAIN Note 2 |
| ANI2 | | | | P22/AMP0+ Note 2 |
| ANI3 | | | | P23 |
| ANI4 | | | | P24 |
| ANI5 | | | | P25 |
| ANI6 | | | | P26 |
| ANI7 Note 3 | | | | P27 Note 3 |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, and 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

<R> **(2) Non-port functions : 78K0/KC2-L (2/4)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| ANI8 | Input | A/D converter analog input | Input port | P10/$\overline{\text{SCK10}}$/AMP1-[Note 2] |
| ANI9 | | | | P11/SI10/ AMP1OUT[Note 2] |
| ANI10 | | | | P12/SO10/AMP1+[Note 2] |
| AMP0-[Note 2] | | Operational amplifier 0 input | Analog input | P20/ANI0 |
| AMP0+[Note 2] | | | | P22/ANI2 |
| AMP1-[Note 2] | | Operational amplifier 1 input | Input port | P10/ANI8/$\overline{\text{SCK10}}$ |
| AMP1+[Note 2] | | | | P12/ANI10/SO10 |
| AMP0OUT[Note 2] | Output | Operational amplifier 0 output | Analog input | P21/ANI1/PGAIN[Note 2] |
| AMP1OUT[Note 2] | | Operational amplifier 1 output | Input port | P11/ANI9/SI10 |
| PGAIN[Note 2] | Input | PGA (programmable gain amplifier) input | Analog input | P21/ANI1/ AMP0OUT[Note 2] |
| EXLVI | Input | Potential input for external low-voltage detection | Input port | P120/INTP0 (/SO11)[Note 3] |
| INTP0 | Input | External interrupt request input for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified | Input port | P120/EXLVI (/SO11)[Note 3] |
| INTP1 | | | | P30 |
| INTP2 | | | | P31/TOOLC1 |
| INTP3 | | | | P32/TOOLD1 |
| INTP4 | | | | P33/TI51/TO51 |
| INTP5 | | | | P16/TOH1 |
| INTP6[Note 1] | | | | P42[Note 1]/PCL[Note 1]/ $\overline{\text{SSI11}}$[Note 1] |
| INTP7[Note 1] | | | | P02[Note 1] |
| INTP8[Note 3] | | | | P63[Note 3] |
| INTP9 | | | | P62/SO11 |
| INTP10 | | | | P61/SDAA0/SI11 |
| INTP11 | | | | P60/SCLA0/$\overline{\text{SCK11}}$ |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, and 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

<R> **(2) Non-port functions : 78K0/KC2-L (3/4)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| KR0 to KR3 | Input | Key interrupt input | Input port | P70 to P73 |
| KR4[Note 1], KR5[Note 1] | | | | P74[Note 1], P75[Note 1] |
| PCL[Note 1] | Output | Clock output (for output of high-speed system clock, subsystem clock) | Input port | P42[Note 1]/$\overline{\text{SSI11}}$[Note 1]/ INTP6[Note 1] |
| RTCDIV[Note 3] | Output | Real-time counter clock (32 kHz divided frequency) output | Input port | P40[Note 3]/RTCCL[Note 3] (/$\overline{\text{SCK11}}$)[Note 3] |
| RTCCL[Note 3] | | Real-time counter clock (32 kHz original oscillation) output | | P40[Note 3]/RTCDIV[Note 3] (/$\overline{\text{SCK11}}$)[Note 3] |
| RTC1HZ[Note 3] | | Real-time counter correction clock (1 Hz) output | | P41[Note 3] (/SI11)[Note 3] |
| REGC | – | Connecting regulator output (2.0 V/2.4V) stabilization capacitance for internal operation. Connect to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F). | – | – |
| $\overline{\text{RESET}}$ | Input | System reset input | Reset input | P125 |
| RxD6 | Input | Serial data input to UART6 | Input port | P14 |
| TxD6 | Output | Serial data output from UART6 | Input port | P13 |
| SCLA0 | I/O | Clock input/output for $I^2C$ | Input port | P60/$\overline{\text{SCK11}}$/INTP11 |
| SDAA0 | I/O | Serial data I/O for $I^2C$ | Input port | P61/SI11/INTP10 |
| $\overline{\text{SCK10}}$ | I/O | Clock input/output for CSI10 | Input port | P10/ANI8/AMP1-[Note 2] |
| $\overline{\text{SCK11}}$ | | Clock input/output for CSI11 | Input port | P60/SCLA0/INTP11 |
| (/$\overline{\text{SCK11}}$)[Note 3] | | | | P40[Note 3]/RTCCL[Note 3]/ RTCDIV[Note 3] |
| SI10 | Input | Serial data input to CSI10 | Input port | P11/ANI9/ AMP1OUT[Note 2] |
| SI11 | | Serial data input to CSI11 | Input port | P61/SDAA0/INTP10 |
| (SI11)[Note 3] | | | | P41[Note 3]/RTC1HZ[Note 3] |
| SO10 | Output | Serial data output from CSI10 | Input port | P12/ANI10/AMP1+[Note 2] |
| SO11 | | Serial data output from CSI11 | Input port | P62/INTP9 |
| (SO11)[Note 3] | | | | P120/INTP0/EXLVI |
| $\overline{\text{SSI11}}$[Note 1] | Input | Chip select input to CSI11 | Input port | P42[Note 1]/PCL[Note 1]/ INTP6[Note 1] |
| TI000 | Input | External count clock input to 16-bit timer/event counter 00 Capture trigger input to capture registers (CR000, CR010) of 16-bit timer/event counter 00 | Input port | P00 |
| TI010 | | Capture trigger input to capture register (CR000) of 16-bit timer/event counter 00 | | P01/TO00 |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, and 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

**(2) Non-port functions: 78K0/KC2-L (4/4)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| TI50 | Input | External count clock input to 8-bit timer/event counter 50 | Input port | P17/TO50 |
| TI51 | | External count clock input to 8-bit timer/event counter 51 | | P33/TO51/INTP4 |
| TO00 | Output | 16-bit timer/event counter 00 output | Input port | P01/TI010 |
| TO50 | Output | 8-bit timer/event counter 50 output | Input port | P17/TI50 |
| TO51 | | 8-bit timer/event counter 51 output | | P33/TI51/INTP4 |
| TOH0 | Output | 8-bit timer H0 output | Input port | P15 |
| TOH1 | | 8-bit timer H1 output | | P16/INTP5 |
| X1 | – | Connecting resonator for main system clock | Input port | P121/TOOLC0 |
| X2 | | | | P122/EXCLK/TOOLD0 |
| EXCLK | Input | External clock input for main system clock | Input port | P122/X2/TOOLD0 |
| XT1 | – | Connecting resonator for subsystem clock | Input port | P123 |
| XT2 | | | | P124/EXCLKS |
| EXCLKS | Input | External clock input for subsystem clock | Input port | P124/XT2 |
| $V_{DD}$ | – | Positive power supply for pins other than port 2 | – | – |
| $AV_{REF}$ | – | A/D converter reference voltage input and positive power supply for port 2 and A/D converter | – | – |
| $V_{SS}$ | – | Ground potential for pins other than port 2 | – | – |
| $AV_{SS}$ | | Ground potential for port 2 and A/D converter | | |
| TOOLC0 | Input | Clock input for flash memory programmer/on-chip debugger | Input port | P121/X1 |
| TOOLC1 | | | | P31/INTP2 |
| TOOLD0 | I/O | Data I/O for flash memory programmer/on-chip debugger | | P122/X2/EXCLK |
| TOOLD1 | | | | P32/INTP3 |
| IC | – | Internally connected. Leave open. | – | – |

## 2.2 Description of Pin Functions

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

### 2.2.1 P00 to P02 (port 0)

P00 to P02 function as an I/O port. These pins also function as timer I/O, external interrupt request input, and chip select input of serial interface.

The timer I/O can be assigned to P00 of the 78K0/KA2-L (25-pin products) by setting the port alternate switch control register (MUXSEL).

<R>

| 78K0/KY2-L (μPD78F057x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| P00/TI000/ INTP0 | P00/TI000/ INTP0 | P00/TI000/ INTP0(/TOH1) (/TI51) | – | P00/TI000 | P00/TI000 | P00/TI000 | P00/TI000 |
| P01/TO00/ TI010 | P01/TO00/ TI010 | – | P01/TO00/ TI010 | P01/TO00/ TI010 | P01/TO00/ TI010 | P01/TO00 /TI010 | P01/TO00/ TI010 |
| – | – | P02/$\overline{\text{SSI11}}$/ INTP5 | P02/$\overline{\text{SSI11}}$/ INTP5 | – | – | – | P02/INTP7 |

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P00 to P02 function as an I/O port. P00 to P02 can be set to input or output port in 1-bit units using port mode register 0 (PM0). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 0 (PU0).

**(2) Control mode**

P00 to P02 function as timer I/O, external interrupt request input, and chip select input of serial interface.

**(a) TI000**

This is a pin for inputting an external count clock to 16-bit timer/event counter 00 and is also for inputting a capture trigger signal to the capture registers (CR000, CR010) of 16-bit timer/event counter 00.

**(b) TI010**

This is a pin for inputting a capture trigger signal to the capture register (CR000) of 16-bit timer/event counter 00.

**(c) TO00**

This is a timer output pin of 16-bit timer/event counter 00.

**(d) INTP0, INTP5, INTP7**

These are external interrupt request input pins for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.

**(e) TOH1**

This is a timer output pin of 8-bit timer H1

**(f) TI51**

This is a pin for inputting an external count clock to 8-bit timer/event counter 51.

**(g) $\overline{\text{SSI11}}$**

This is a chip select input pin of serial interface CSI11.

### 2.2.2 P10 to P17 (port 1)

P10 to P17 function as an I/O port. These pins also function as pins for A/D converter analog input, operational amplifier I/O, external interrupt request input, serial interface data I/O, clock I/O, and timer I/O.

<R>

| 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40, 44, 48 Pins |
| – | – | P10/ANI8/AMP1-Note/$\overline{\text{SCK10}}$ | P10/ANI8/AMP1-Note/$\overline{\text{SCK10}}$ |
| – | – | P11/ANI9/AMP1OUTNote/SI10 | P11/ANI9/AMP1OUTNote/SI10 |
| – | – | P12/ANI10/AMP1+Note/SO10 | P12/ANI10/AMP1+Note/SO10 |
| – | – | P13/TxD6 | P13/TxD6 |
| – | – | P14/RxD6 | P14/RxD6 |
| – | – | P15/TOH0 | P15/TOH0 |
| – | – | P16/TOH1/INTP5 | P16/TOH1/INTP5 |
| – | – | P17/TI50/TO50 | P17/TI50/TO50 |

**Note** Products with operational amplifier only

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P10 to P17 function as an I/O port. P10 to P17 can be set to input or output port in 1-bit units using port mode register 1 (PM1). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 1 (PU1).

**(2) Control mode**

P10 to P17 function as A/D converter analog input, operational amplifier I/O, external interrupt request input, serial interface data I/O, clock I/O, and timer I/O.

**(a) ANI8 to ANI10**

These are A/D converter analog input pins. When using these pins as analog input pins, refer to **(5) ANI0/P20 to ANI7/P27, ANI8/P10 to ANI10/P12** in **12.6 Cautions for A/D Converter**.

**Cautions 1. ANI8/P10 to ANI10/P12 are set in the digital input mode after release of reset.**

<R> **2. Make the $AV_{REF}$ pin the same potential as the $V_{DD}$ pin when ANI8 to ANI10 are used.**

**(b) AMP1+, AMP1-**

These are operational amplifier 1 input pins.

**(c) AMP1OUT**

This is an operational amplifier 1 output pin.

**(d) SI10**

This is a serial data input pin of serial interface CSI10.

**(e) SO10**

This is a serial data output pin of serial interface CSI10.

**(f) $\overline{\text{SCK10}}$**

This is a serial clock I/O pin of serial interface CSI10.

**(g) RxD6**

This is a serial data input pin of serial interface UART6.

**(h) TxD6**

This is a serial data output pin of serial interface UART6.

**(i) TI50**

This is a pin for inputting an external count clock to 8-bit timer/event counter 50.

**(j) TO50**

This is a timer output pin of 8-it timer/event counter 50.

**(k) TOH0, TOH1**

These are a timer output pins of 8-bit timers H0 and H1.

**(l) INTP5**

This is an external interrupt request input pin for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.

### 2.2.3 P20 to P27 (port 2)

P20 to P27 function as an I/O port. These pins also function as pins for A/D converter analog input, operational amplifier I/O, and PGA input.

<R>

| 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note |
| P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote |
| P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note |
| P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 |
| – | P24/ANI4 | P24/ANI4 | P24/ANI4 | – | P24/ANI4 | P24/ANI4 | P24/ANI4 |
| – | P25/ANI5 | P25/ANI5 | P25/ANI5 | – | P25/ANI5 | P25/ANI5 | P25/ANI5 |
| – | – | P26/ANI6 | P26/ANI6 | – | P26/ANI6 | P26/ANI6 | P26/ANI6 |
| – | – | – | P27/ANI7 | – | – | P27/ANI7 | P27/ANI7 |

**Note** Products with operational amplifier only

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P20 to P27 function as an I/O port. P20 to P27 can be set to input or output port in 1-bit units using port mode register 2 (PM2).

**(2) Control mode**

P20 to P27 function as A/D converter analog input, operational amplifier I/O, and PGA input.

**(a) ANI0 to ANI7**

These are A/D converter analog input pins. When using these pins as analog input pins, refer to **(5) ANI0/P20 to ANI7/P27 and ANI8/P10 to ANI10/P12** in **12.6 Cautions for A/D Converter**.

**(b) AMP0+, AMP0-**

These are operational amplifier 0 input pins.

**(c) AMP0OUT**

This is an operational amplifier 0 output pin.

**(d) PGAIN**

This is a PGA (Programmable gain amplifier) input pin.

**Caution ANI0/P20 to ANI7/P27 are set in the analog input mode after release of reset.**

### 2.2.4 P30 to P37 (port 3)

P30 to P37 function as an I/O port. These pins also function as pins for external interrupt request input, timer I/O, clock input and data I/O for flash memory programmer/on-chip debugger, and clock I/O and data I/O for serial interface.

The timer I/O can be assigned to P34 of the 78K0/KA2-L (25-pin and 32-pin products) by setting the port alternate switch control register (MUXSEL).

<R>

| 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| P30/TOH1/ TI51/INTP1 | P30/TOH1/ TI51/INTP1 | – | – | P30/INTP1 | P30/INTP1 | P30/INTP1 | P30/INTP1 |
| – | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 |
| – | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 |
| – | – | P33 | P33 | P33/TI51/ TO51/INTP4 | P33/TI51/ TO51/INTP4 | P33/TI51/ TO51/INTP4 | P33/TI51/ TO51/INTP4 |
| – | – | P34/INTP4 (/TOH1) (/TI51) | P34/INTP4 (/TOH1) | – | – | – | – |
| – | – | P35/$\overline{\text{SCK11}}$ | P35/$\overline{\text{SCK11}}$ | – | – | – | – |
| – | – | P36/SI11 | P36/SI11 | – | – | – | – |
| – | – | P37/SO11 | P37/SO11 | – | – | – | – |

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P30 to P37 function as an I/O port. P30 to P37 can be set to input or output port in 1-bit units using port mode register 3 (PM3). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 3 (PU3).

**(2) Control mode**

P30 to P37 function as external interrupt request input, timer I/O, clock input and data I/O for flash memory programmer/on-chip debugger, and clock I/O and data I/O for serial interface.

**(a) INTP1 to INTP4**

These are external interrupt request input pins for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.

**(b) TI51**

This is an external count clock input pin to 8-bit timer/event counter 51.

**(c) TO51**

This is a timer output pin from 8-bit timer/event counter 51.

**(d) TOH1**

This is a timer output pin of 8-bit timer H1.

**(e) TOOLC1**

This is a clock input pin for flash memory programmer/on-chip debugger.

**(f) TOOLD1**

This is a data I/O pin for flash memory programmer/on-chip debugger.

**(g) $\overline{\text{SCK11}}$**

This is a serial clock I/O pin of serial interface CSI11.

**(h) SI11**

This is a serial data input pin of serial interface CSI11.

**(i) SO11**

This is a serial data output pin of serial interface CSI11.

**Remark** For how to connect a flash memory programmer using TOOLC1/P31, TOOLD1/P32, refer to **CHAPTER 25 FLASH MEMORY**. For how to connect TOOLC1/P31, TOOLD1/P32 and an on-chip debug emulator, refer to **CHAPTER 26 ON-CHIP DEBUG FUNCTION**.

### 2.2.5 P40 to P42 (port 4)

P40 to P42 function as an I/O port. These pins also function as pins for external interrupt request input, real-time counter clock output, real-time counter correction clock output, and chip select input of serial interface.

The clock I/O and data input of the serial interface can be assigned to P40 and P41 of the 78K0/KC2-L (44-pin and 48-pin products) respectively by setting the port alternate switch control register (MUXSEL).

<R>

| 78K0/KY2-L (μPD78F057x) | 78K0/KA2-L (μPD78F056x) | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| – | – | – | – | P40/RTCCL/ RTCDIV(/$\overline{\text{SCK11}}$) | P40/RTCCL/ RTCDIV(/$\overline{\text{SCK11}}$) |
| – | – | – | – | P41/RTC1HZ (/SI11) | P41/RTC1HZ (/SI11) |
| – | – | – | – | – | P42/PCL/$\overline{\text{SSI11}}$/ INTP6 |

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P40 to P42 function as an I/O port. P40 to P42 can be set to input or output port in 1-bit units using port mode register 4 (PM4). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 4 (PU4).

**(2) Control mode**

P40 to P42 function as external interrupt request input, real-time counter clock output, real-time counter correction clock output, and serial interface clock I/O, data input, and chip select input.

**(a) RTCDIV**

This is the real-time counter clock (32 kHz division) output pin.

**(b) RTCCL**

This is the real-time counter clock (32 kHz original oscillation) output pin.

**(c) RTC1HZ**

This is the real-time counter correction clock (1 Hz) output pin.

**(d) INTP6**

This is an external interrupt request input pin for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.

**(e) PCL**

This is a clock output pin.

**(f) $\overline{\text{SCK11}}$**

This is a serial clock I/O pin of serial interface CSI11.

**(g) SI11**

This is a serial data input pin of serial interface CSI11.

**(h) $\overline{\text{SSI11}}$**

This is a chip select input pin of serial interface CSI11.

### 2.2.6 P60 to P63 (port 6)

P60 to P63 function as an I/O port. These pins also function as pins for serial interface data I/O, clock I/O, and external interrupt request input.

Input to the P60 and P61 pins can be specified through a normal input buffer or an SMBus input buffer in 1-bit units, using port input mode register 6 (PIM6).

Output from the P60 to P63 pins can be specified as normal CMOS output or N-ch open-drain output ($V_{DD}$ tolerance) in 1-bit units, using port output mode register 6 (POM6).

<R>

| 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | |
|---|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40 Pins | 44, 48 Pins |
| P60/SCLA0/TxD6 | P60/SCLA0/TxD6 | P60/SCLA0/INTP11 | P60/SCLA0/$\overline{\text{SCK11}}$/ INTP11 | P60/SCLA0/$\overline{\text{SCK11}}$/ INTP11 |
| P61/SDAA0/RxD6 | P61/SDAA0/RxD6 | P61/SDAA0/INTP10 | P61/SDAA0/SI11/ INTP10 | P61/SDAA0/SI11/ INTP10 |
| – | – | – | P62/SO11/INTP9 | P62/SO11/INTP9 |
| – | – | – | – | P63/INTP8 |

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P60 to P63 function as an I/O port. P60 to P63 can be set to input port or output port in 1-bit units using port mode register 6 (PM6). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 6 (PU6).

**(2) Control mode**

P60 to P63 function as serial interface data I/O and clock I/O.

**(a) SDAA0**

This is a serial data I/O pin for serial interface IICA.

**(b) SCLA0**

This is a serial clock I/O pin for serial interface IICA.

**(c) RxD6**

This is a serial data input pin for serial interface UART6.

**(d) TxD6**

This is a serial data output pin for serial interface UART6.

**(e) $\overline{\text{SCK11}}$**

This is a serial clock I/O pin for serial interface CSI11.

**(f) SI11**

This is a serial data input pin for serial interface CSI11.

**(g) SO11**

This is a serial data output pin for serial interface CSI11.

**(h) INTP8 to INTP11**

These are external interrupt request input pins for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.

### 2.2.7 P70 to P75 (port 7)

P70 to P75 function as an I/O port. These pins also function as pins for A/D converter analog input and key interrupt input pins.

<R>

| 78K0/KY2-L (μPD78F057x) | 78K0/KA2-L (μPD78F056x) | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | |
|---|---|---|---|---|---|
| 16 Pins | 20, 25 Pins | 32 Pins | 30 Pins | 40, 44 Pins | 48 Pins |
| – | – | P70/ANI8 | – | P70/KR0 | P70/KR0 |
| – | – | P71/ANI9 | – | P71/KR1 | P71/KR1 |
| – | – | P72/ANI10 | – | P72/KR2 | P72/KR2 |
| – | – | – | – | P73/KR3 | P73/KR3 |
| – | – | – | – | – | P74/KR4 |
| – | – | – | – | – | P75/KR5 |

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P70 to P75 function as an I/O port. P70 to P75 can be set to input or output port in 1-bit units using port mode register 7 (PM7). Use of an on-chip pull-up resistor can be specified by pull-up resistor option register 7 (PU7) in 78K0/KC2-L.

**(2) Control mode**

P70 to P75 function as pins for A/D converter analog input and key interrupt input pins.

**(a) ANI8 to ANI10**

These are the A/D converter analog input pins. When using this pin as analog input pin, refer to **(5) ANI0/P20 to ANI7/P27, ANI8/P10 to ANI10/P12, and ANI8/P70 to ANI10/P72** in **12.6 Cautions for A/D Converter**.

**(b) KR0 to KR5**

These are the key interrupt input pins

### 2.2.8 P120 to P125 (port 12)

P120 functions as an I/O port. P121 to P125 function as an Input port. These pins also function as pins for external interrupt request input, potential input for external low-voltage detection, connecting resonator for main system clock, connecting resonator for subsystem clock, external clock input for main system clock, external clock input for subsystem clock, external reset input, and clock input and data I/O for flash memory programmer/on-chip debugger.

Set bit 5 (RSTM) of the reset pin mode register (RSTMASK) to 1 when using P125/$\overline{\text{RESET}}$ as an input port, and clear RSTM to 0 when using P125/$\overline{\text{RESET}}$ as an external reset input.

<R> Furthermore, the timer input and external interrupt request input can be assigned to P121 of the 78K0/KA2-L (25-pin products) and P121 and P125 of the 78K0/KA2-L (32-pin products) by setting the port alternate switch control register (MUXSEL). The data output of the serial interface can be assigned to P120 of the 78K0/KC2-L (44-pin and 48-pin products) by setting the port alternate switch control register (MUXSEL).

<R>

| 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | |
|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44, 48 Pins |
| – | – | – | – | P120/EXLVI/ INTP0 | P120/EXLVI/ INTP0 | P120/EXLVI/ INTP0(/SO11) |
| P121/X1/ TOOLC0 | P121/X1/ TOOLC0 | P121/X1/ TOOLC0 (/TI000) (/INTP0) | P121/X1/ TOOLC0 (/TI000) (/INTP0) | P121/X1/ TOOLC0 | P121/X1/ TOOLC0 | P121/X1/ TOOLC0 |
| P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 |
| – | – | – | – | – | P123/XT1 | P123/XT1 |
| – | – | – | – | – | P124/XT2/ EXCLKS | P124/XT2/ EXCLKS |
| P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ (/TI000) (/INTP0) | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ |

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

The following operation modes can be specified in 1-bit units.

**(1) Port mode**

P120 to P125 function as an I/O port. P120 to P125 can be set to input or output port using port mode register 12 (PM12). Only for P120 and P125, use of an on-chip pull-up resistor can be specified by pull-up resistor option register 12 (PU12).

**(2) Control mode**

P120 to P125 function as pins for external interrupt request input, potential input for external low-voltage detection, connecting resonator for main system clock, connecting resonator for subsystem clock, external clock input for main system clock, external clock input for subsystem clock, external reset input, and clock input and data I/O for flash memory programmer/on-chip debugger.

**(a) INTP0**

This functions as an external interrupt request input (INTP0) for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.

**(b) EXLVI**

This is a potential input pin for external low-voltage detection.

**(c) X1, X2**

These are pins for connecting a resonator for main system clock.

**(d) EXCLK**

This is an external clock input pin for main system clock.

**(e) XT1, XT2**

These are pins for connecting a resonator for subsystem clock.

**(f) EXCLKS**

This is an external clock input pin for subsystem clock.

**(g) SO11**

This is a serial data output pin of serial interface CSI11.

**(h) $\overline{\text{RESET}}$**

This is an active-low system reset input pin.

**(i) TOOLC0**

This is a clock input pin for flash memory programmer/on-chip debugger.

**(j) TOOLD0**

This is a data I/O pin for flash memory programmer/on-chip debugger.

**(k) TI000**

This is a pin for inputting an external count clock to 16-bit timer/event counter 00 and is also for inputting a capture trigger signal to the capture registers (CR000, CR010) of 16-bit timer/event counter 00.

<R> **Caution Because $\overline{\text{RESET}}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

**Remark** For how to connect a flash memory programmer using TOOLC0/X1, TOOLD0/X2, refer to **CHAPTER 25 FLASH MEMORY**. For how to connect TOOLC0/X1, TOOLD0/X2 and an on-chip debug emulator, refer to **CHAPTER 26 ON-CHIP DEBUG FUNCTION**.

### 2.2.9 $AV_{REF}$, $AV_{SS}$, $V_{DD}$, $V_{SS}$

These are the power supply/ground pins.

<R>

| 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40, 44, 48 Pins |
| $AV_{REF}$ | $AV_{REF}$ | $AV_{REF}$ | $AV_{REF}$ |
| – | – | $AV_{SS}$ | $AV_{SS}$ |
| $V_{DD}$ | $V_{DD}$ | $V_{DD}$ | $V_{DD}$ |
| $V_{SS}$ | $V_{SS}$ | $V_{SS}$ | $V_{SS}$ |

**(a) $AV_{REF}$**

This is the A/D converter reference voltage input pin and the positive power supply pin of port 2 and A/D converter.

<R> This is also the positive power supply pin of port 7 in the 78K0/KA2-L (32 pins).

When the A/D converter is not used, connect this pin directly to $V_{DD}$[Note].

**Note** Make the $AV_{REF}$ pin the same potential as the $V_{DD}$ pin when port 2 is used as a digital port.

**(b) $AV_{SS}$**

This is a ground potential pin of A/D converter and port 2. Even when the A/D converter is not used, always use this pin with the same potential as the $V_{SS}$ pin.

**(c) $V_{DD}$**

$V_{DD}$ is a positive power supply pin.

**(d) $V_{SS}$**

$V_{SS}$ is a ground potential pin[Note].

**Note** In the 78K0/KY2-L and 78K0/KA2-L, $V_{SS}$ functions alternately as the ground potential of the A/D converter. Be sure to connect $V_{SS}$ to a stabilized GND (= 0 V).

### 2.2.10 REGC, IC0, IC

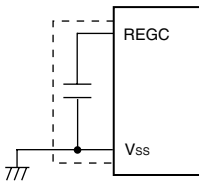
This is a pin for connecting regulator output stabilization capacitance for internal operation and an internally connected pin.

<R>

| 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|---|
| 16 Pins | 20, 25 Pins | 32 Pins | 30 Pins | 40, 44, 48 Pins |
| REGC | REGC | REGC | REGC | REGC |
| – | – | IC0 | IC | IC |

**(a) REGC**

This is a pin for connecting regulator output (2.0 V/2.4 V) stabilization capacitance for internal operation. Connect this pin to $V_{SS}$ via a capacitor (0.47 to 1 $\mu$F). However, when using the STOP mode that has been entered since operation of the internal high-speed oscillation clock and external main system clock, 0.47 $\mu$F is recommended. Also, use a capacitor with good characteristics, since it is used to stabilize internal voltage.




**Caution Keep the wiring length as short as possible for the broken-line part in the above figure.**

**(b) IC0**

This is an internally connected pin. Connect directly to $V_{SS}$.

**(c) IC**

This is an internally connected pin. Leave open.

## 2.3 Pin I/O Circuits and Recommended Connection of Unused Pins

Tables 2-2 to 2-6 show the types of pin I/O circuits and the recommended connections of unused pins.
Refer to **Figure 2-1** for the configuration of the I/O circuit of each type.

**Table 2-2. Pin I/O Circuit Types (78K0/KY2-L)**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P00/TI000/INTP0 | 5-AQ | I/O | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P01/TO00/TI010 | | | |
| ANI0/P20/AMP0-Note 1 | 11-P | | <Digital input setting><br>Independently connect to $AV_{REF}$ or $V_{SS}$ via a resistor.<br><Digital output setting and analog input setting ><br>Leave open. Note 2 |
| <R> ANI1/P21/AMP0OUTNote 1/ PGAINNote 1 | 11-O | | |
| ANI2/P22/AMP0+Note 1 | 11-N | | |
| ANI3/P23 | 11-G | | |
| P30/TOH1/TI51/INTP1 | 5-AQ | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P60/SCLA0/TxD6 | 5-AS | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave this pin open at low-level output after clearing the output latch of the port to 0. |
| P61/SDAA0/RxD6 | | | |
| P121/X1/TOOLC0Note 3 | 37-A | Input | Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor. |
| P122/X2/EXCLK/ TOOLD0Note 3 | | | |
| $\overline{\text{RESET}}$/P125 | 42-A | | Connect directly to $V_{DD}$ or via a resistor. |
| $AV_{REF}$ | – | – | Connect directly to $V_{DD}$. |

**Notes 1.** $\mu$PD78F0555, 78F0556, and 78F0557 (products with operational amplifier) only

<R> **2.** If this pin is left open when specified as an analog input pin, the input voltage level might become undefined. It is therefore recommended to leave this pin open after specifying it as a digital output pin.

**3.** Use recommended connection above in input port mode (refer to **Figure 5-3 Format of Clock Operation Mode Select Register (OSCCTL)**) when these pins are not used.

**Cautions 1. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 are set in the analog input mode after release of reset.**

<R> **2. Because $\overline{\text{RESET}}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

**Table 2-3. Pin I/O Circuit Types (78K0/KA2-L (20-pin products))**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P00/TI000/INTP0 | 5-AQ | I/O | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P01/TO00/TI010 | | | |
| ANI0/P20/AMP0-[Note 1] | 11-P | | <Digital input setting><br>Independently connect to $AV_{REF}$ or $V_{SS}$ via a resistor.<br><Digital output setting and analog input setting ><br>Leave open.[Note 2] |
| <R> ANI1/P21/AMP0OUT[Note 1]/ PGAIN[Note 1] | 11-O | | |
| ANI2/P22/AMP0+[Note 1] | 11-N | | |
| ANI3/P23 | 11-G | | |
| ANI4/P24 | | | |
| ANI5/P25 | | | |
| P30/TOH1/TI51/INTP1 | 5-AQ | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P31/INTP2/TOOLC1 | | | |
| P32/TOH1/INTP3/TOOLD1 | | | |
| P60/SCLA0/TxD6 | 5-AS | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave this pin open at low-level output after clearing the output latch of the port to 0. |
| P61/SDAA0/RxD6 | | | |
| P121/X1/TOOLC0[Note 3] | 37-A | Input | Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor. |
| P122/X2/EXCLK/TOOLD0[Note 3] | | | |
| $\overline{\text{RESET}}$/P125 | 42-A | Input | Connect directly to $V_{DD}$ or via a resistor. |
| $AV_{REF}$ | – | – | Connect directly to $V_{DD}$. |

**Notes 1.** $\mu$PD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

<R> **2.** If this pin is left open when specified as an analog input pin, the input voltage level might become undefined. It is therefore recommended to leave this pin open after specifying it as a digital output pin.

**3.** Use recommended connection above in input port mode (refer to **Figure 5-3 Format of Clock Operation Mode Select Register (OSCCTL)**) when these pins are not used.

**Cautions 1. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI5/P25 are set in the analog input mode after release of reset.**

<R> **2. Because $\overline{\text{RESET}}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

<R> **Table 2-4. Pin I/O Circuit Types (78K0/KA2-L (25-pin and 32-pin products))**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P00 Note 1/TI000 Note 1/ INTP0 Note 1 | 5-AQ | I/O | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P01 Note 2/TO00 Note 2/ TI010 Note 2 | | | |
| P02/$\overline{SSI11}$/INTP5 | | | |
| ANI0/P20/AMP0- Note 3 | 11-P | | <Digital input setting><br>Independently connect to $AV_{REF}$ or $V_{SS}$ via a resistor.<br><Digital output setting and analog input setting ><br>Leave open. Note 4 |
| ANI1/P21/AMP0OUT Note 3/ PGAIN Note 3 | 11-O | | |
| ANI2/P22/AMP0+ Note 3 | 11-N | | |
| ANI3/P23 to ANI6/P26 | 11-G | | |
| ANI7/P27 Note 2 | | | |
| P31/INTP2/TOOLC1 | 5-AQ | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P32/TOH1/INTP3/TOOLD1 | | | |
| P33 | | | |
| P34/INTP4 (/TOH1)(/TI51) Note 1 | | | |
| P35/$\overline{SCK11}$ | | | |
| P36/SI11 | | | |
| P37/SO11 | | | |
| P60/SCLA0/TxD6 | 5-AS | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave this pin open at low-level output after clearing the output latch of the port to 0. |
| P61/SDAA0/RxD6 | | | |
| ANI8 Note 2/P70 Note 2 | 11-G | | <Digital input setting><br>Independently connect to $AV_{REF}$ or $V_{SS}$ via a resistor.<br><Digital output setting and analog input setting ><br>Leave open. Note 4 |
| ANI9 Note 2/P71 Note 2 | | | |
| ANI10 Note 2/P72 Note 2 | | | |
| P121/X1/TOOLC0 Note 5 (/TI000)(/INTP0) | 37-A | Input | Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor. |
| P122/X2/EXCLK/ TOOLD0 Note 5 | | | |
| $\overline{RESET}$/P125 (/TI000) Note 2 (/INTP0) Note 2 | 42-A | | Connect directly to $V_{DD}$ or via a resistor. |
| $AV_{REF}$ | – | – | Connect directly to $V_{DD}$. |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** μPD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

**4.** If this pin is left open when specified as an analog input pin, the input voltage level might become undefined. It is therefore recommended to leave this pin open after specifying it as a digital output pin.

**5.** Use recommended connection above in input port mode (refer to **Figure 5-3 Format of Clock Operation Mode Select Register (OSCCTL)**) when these pins are not used.

**Cautions 1. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, ANI3/P23 to ANI5/P25, and ANI8/P70 to ANI10/P72 are set in the analog input mode after release of reset.**

**2. Because $\overline{RESET}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

**Table 2-5. Pin I/O Circuit Types (78K0/KB2-L)**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P00/TI000 | 5-AQ | I/O | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P01/TO00/TI010 | | | |
| P10/ANI8/ANP1-[Note 1]/$\overline{SCK10}$ | 11-L | | |
| P11/ANI9/ANP1OUT[Note 1]/SI10 | 11-M | | |
| P12/ANI10/ANP1+[Note 1]/SO10 | 11-K | | |
| P13/TxD6 | 5-AG | | |
| P14/RxD6 | 5-AQ | | |
| P15/TOH0 | 5-AG | | |
| P16/TOH1/INTP5 | 5-AQ | | |
| P17/TI50/TO50 | | | |
| ANI0/P20/AMP0-[Note 1] | 11-P | | <Digital input setting><br>Independently connect to $AV_{REF}$ or $AV_{SS}$ via a resistor.<br><Digital output setting and analog input setting><br>Leave open.[Note 2] |
| ANI1/P21/AMP0OUT[Note 1]/PGAIN[Note 1] | 11-O | | |
| ANI2/P22/AMP0+[Note 1] | 11-N | | |
| ANI3/P23 | 11-G | | |
| P30/INTP1 | 5-AQ | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P31/INTP2/TOOLC1 | | | |
| P32/INTP3/TOOLD1 | | | |
| P33/TI51/TO51/INTP4 | | | |
| P60/SCLA0/INTP11 | 5-AS | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave this pin open at low-level output after clearing the output latch of the port to 0. |
| P61/SDAA0/INTP10 | | | |
| P120/EXLVI/INTP0 | 5-AQ | | Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor. |
| P121/X1/TOOLC0[Note 3] | 37-A | Input | |
| P122/X2/EXCLK/TOOLD0[Note 3] | | | |
| $\overline{RESET}$/P125 | 42-A | Input | Connect directly to $V_{DD}$ or via a resistor. |
| $AV_{REF}$ | – | – | Connect directly to $V_{DD}$.[Note 4] |
| $AV_{SS}$ | – | – | Connect directly to $V_{SS}$. |

**Notes 1.** $\mu$PD78F0576, 78F0577, and 78F0578 (products with operational amplifier) only

**2.** If this pin is left open when specified as an analog input pin, the input voltage level might become undefined. It is therefore recommended to leave this pin open after specifying it as a digital output pin.

**3.** Use recommended connection above in input port mode (refer to **Figure 5-3 Format of Clock Operation Mode Select Register (OSCCTL)**) when these pins are not used.

**4.** When port 2 is used as the digital port pins, make $AV_{REF}$ the same potential as $V_{DD}$.

**Cautions 1. ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 are set in the analog input mode, P10/ANI8/AMP1-/$\overline{SCK10}$, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**2. Because $\overline{RESET}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

<R> **Table 2-6. Pin I/O Circuit Types (78K0/KC2-L) (1/2)**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P00/TI000 | 5-AQ | I/O | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P01/TO00/TI010 | | | |
| P02[Note 1]/INTP7[Note 1] | | | |
| P10/ANI8/ANP1-[Note 2]/$\overline{SCK10}$ | 11-L | | |
| P11/ANI9/ANP1OUT[Note 2]/SI10 | 11-M | | |
| P12/ANI10/ANP1+[Note 2]/SO10 | 11-K | | |
| P13/TxD6 | 5-AG | | |
| P14/RxD6 | 5-AQ | | |
| P15/TOH0 | 5-AG | | |
| P16/TOH1/INTP5 | 5-AQ | | |
| P17/TI50/TO50 | | | |
| ANI0/P20/AMP0-[Note 2] | 11-P | | < Digital input setting><br>Independently connect to $AV_{REF}$ or $AV_{SS}$ via a resistor.<br><Digital output setting and analog input setting><br>Leave open.[Note 3] |
| ANI1/P21/AMP0OUT[Note 2]/PGAIN[Note 2] | 11-O | | |
| ANI2/P22/AMP0+[Note 2] | 11-N | | |
| ANI3/P23 to ANI6/P26 | 11-G | | |
| ANI7[Note 4]/P27[Note 4] | | | |
| P30/INTP1 | 5-AQ | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P31/INTP2/TOOLC1 | | | |
| P32/INTP3/TOOLD1 | | | |
| P33/TI51/TO51/INTP4 | | | |
| P40[Note 4]/RTCCL[Note 4]/RTCDIV[Note 4](/$\overline{SCK11}$)[Note 4] | | | |
| P41[Note 4]/RTC1HZ[Note 4] (/SI11)[Note 4] | | | |
| P42[Note 1]/PCL[Note 1]/$\overline{SSI11}$[Note 1]/INTP6[Note 1] | | | |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, and 78F0588 (products with operational amplifier) only

**3.** If this pin is left open when specified as an analog input pin, the input voltage level might become undefined. It is therefore recommended to leave this pin open after specifying it as a digital output pin.

**4.** 44-pin and 48-pin products only

**Caution ANI0/P20/AMP0-, ANI1/P21/AMP0OUT/PGAIN, ANI2/P22/AMP0+, and ANI3/P23 to ANI7/P27 are set in the analog input mode, P10/ANI8/AMP1-/$\overline{SCK10}$, P11/ANI9/AMP1OUT/SI10, and P12/ANI10/AMP1+/SO10 are set in the digital input mode after release of reset.**

**Remark** Functions in parentheses ( ) in the table above can be assigned by setting the port alternate switch control register (MUXSEL).

<R> **Table 2-6. Pin I/O Circuit Types (78K0/KC2-L) (2/2)**

| Pin Name | I/O Circuit Type | I/O | Recommended Connection of Unused Pins |
|---|---|---|---|
| P60/SCLA0/$\overline{\text{SCK11}}$/INTP11 | 5-AS | I/O | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave this pin open at low-level output after clearing the output latch of the port to 0. |
| P61/SDAA0/SI11/INTP10 | | | |
| P62/SO11/INTP9 | 5-AR | | |
| P63 Note 3/INTP8 Note 3 | | | |
| P70/KR0 | 5-AQ | | Input: Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.<br>Output: Leave open. |
| P71/KR1 | | | |
| P72/KR2 | | | |
| P73/KR3 | | | |
| P74 Note 1/KR4 Note 1 | | | |
| P75 Note 1/KR5 Note 1 | | | |
| P120/EXLVI/INTP0 (/SO11) Note 3 | | | Independently connect to $V_{DD}$ or $V_{SS}$ via a resistor. |
| P121/X1/TOOLC0 Note 2 | 37-A | Input | |
| P122/X2/EXCLK/ TOOLD0 Note 2 | | | |
| P123/XT1 Note 2 | | | |
| P124/XT2/EXCLKS Note 2 | | | |
| $\overline{\text{RESET}}$/P125 | 42-A | Input | Connect directly to $V_{DD}$ or via a resistor. |
| $AV_{REF}$ | – | – | Connect directly to $V_{DD}$. Note 4 |
| $AV_{SS}$ | – | – | Connect directly to $V_{SS}$. |

**Notes 1.** 48-pin products only

**2.** Use recommended connection above in input port mode (refer to **Figure 5-4 Format of Clock Operation Mode Select Register (OSCCTL)**) when these pins are not used.

**3.** 44-pin and 48-pin products only

**4.** When port 2 is used as the digital port pins, make $AV_{REF}$ the same potential as $V_{DD}$.

**Caution Because $\overline{\text{RESET}}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

**Remark** Functions in parentheses ( ) in the table above can be assigned by setting the port alternate switch control register (MUXSEL).

**Figure 2-1. Pin I/O Circuit List (1/4)**

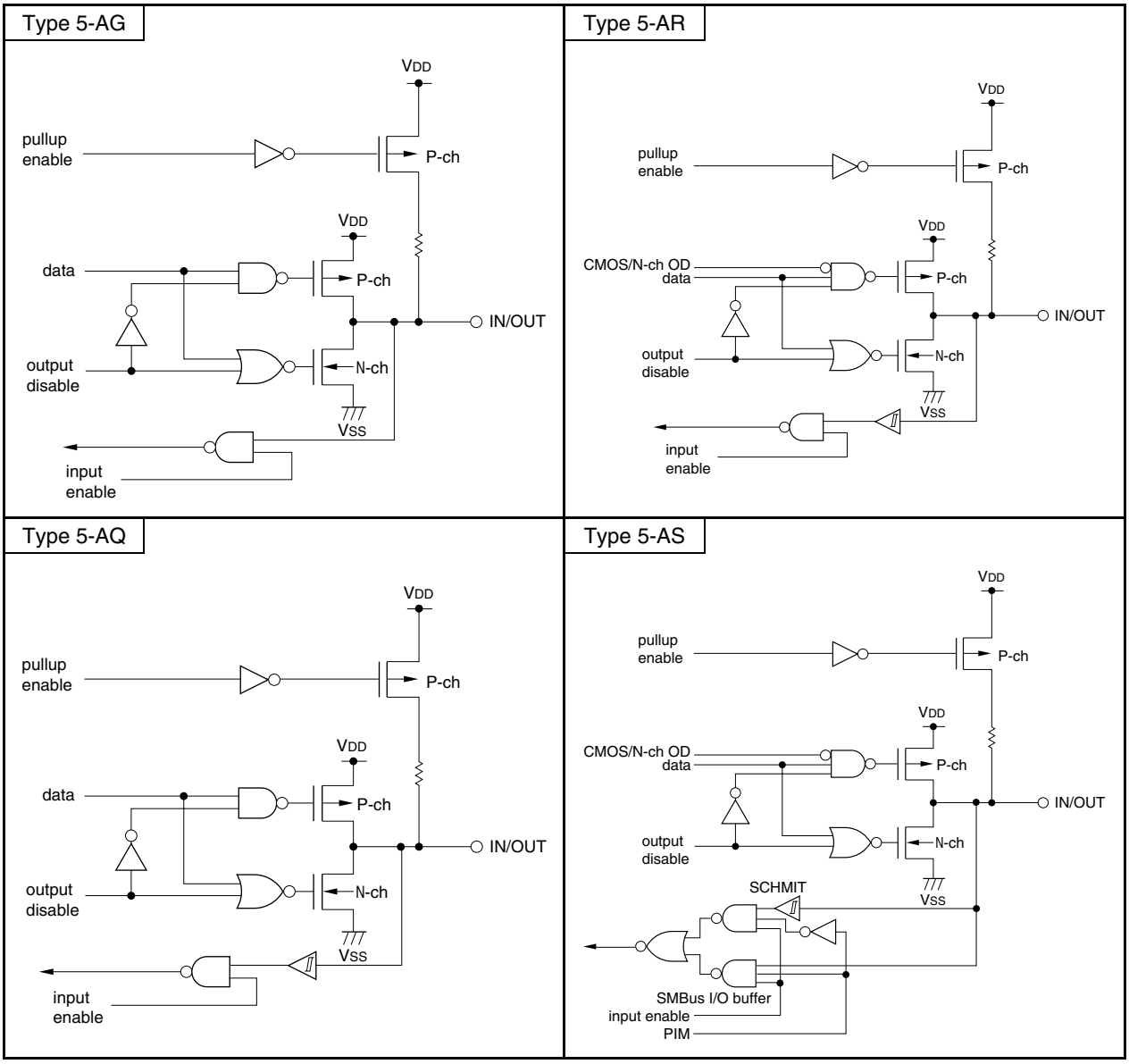


**Figure 2-1. Pin I/O Circuit List (2/4)**

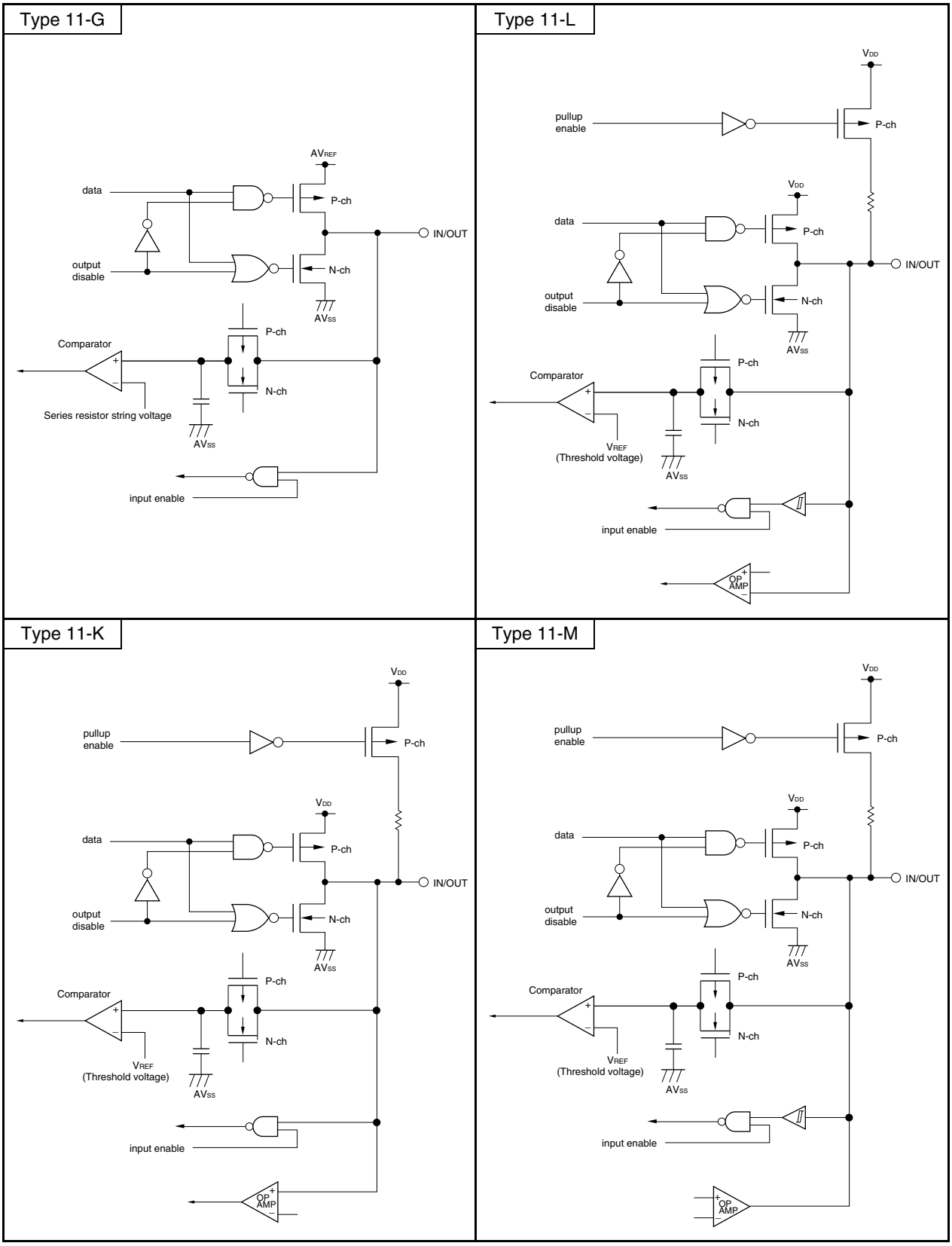
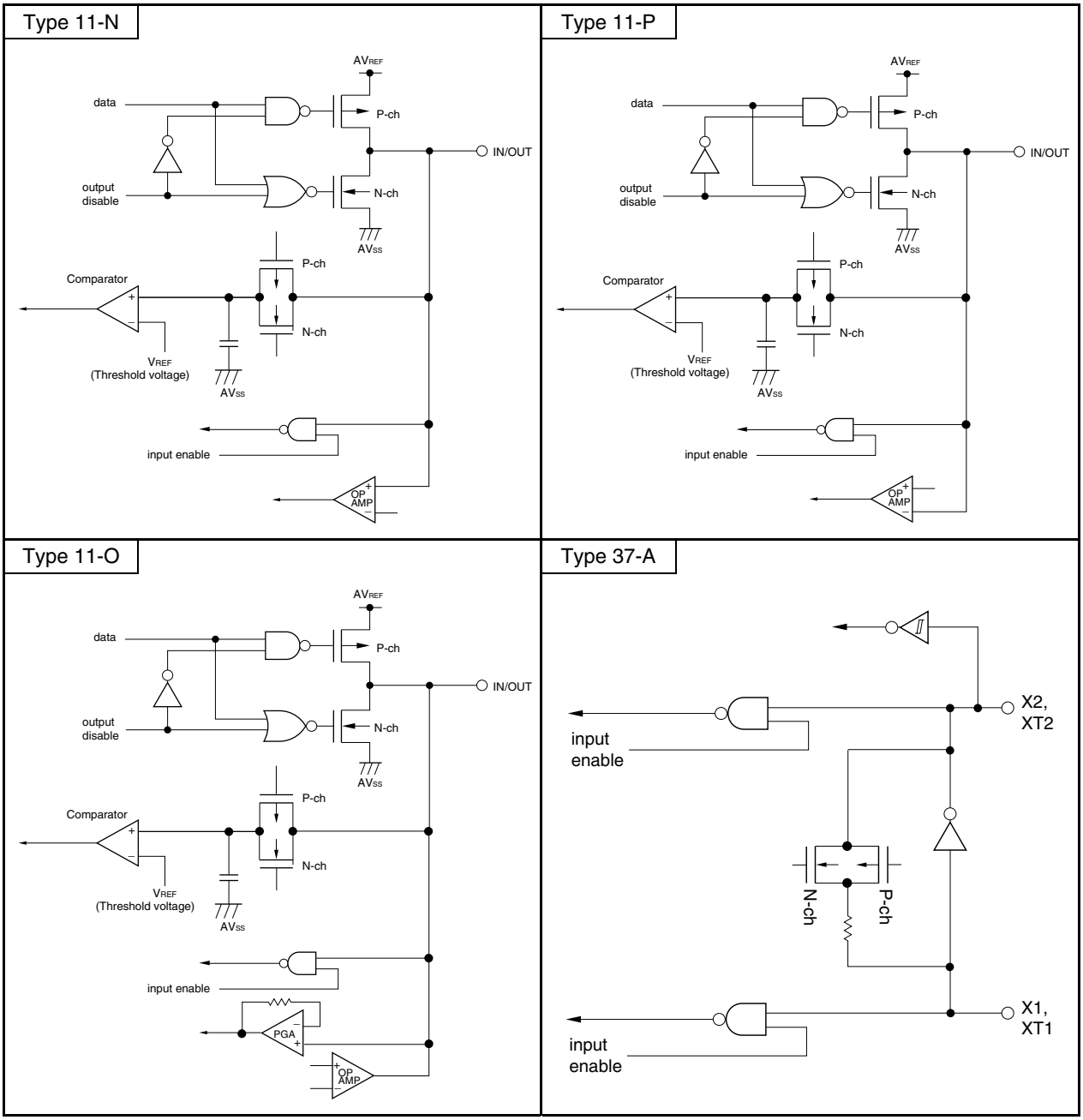
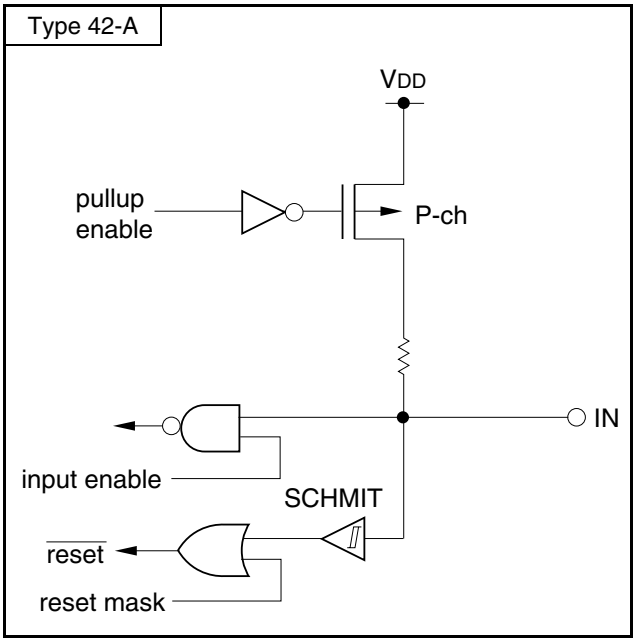


**Figure 2-1. Pin I/O Circuit List (3/4)**



<R>

**Figure 2-1. Pin I/O Circuit List (4/4)**



# CHAPTER 3 CPU ARCHITECTURE

## 3.1 Memory Space

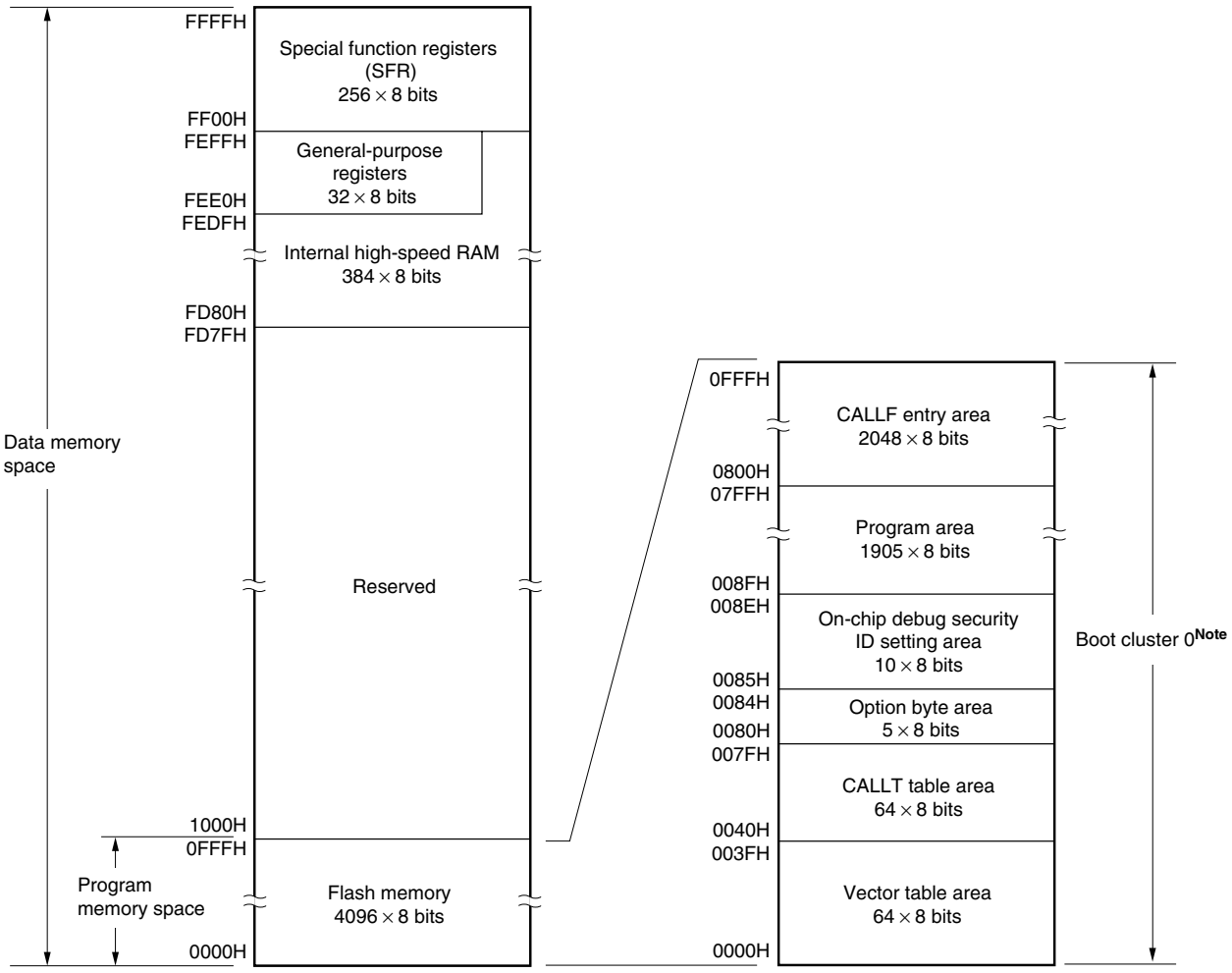
Products in the 78K0/Kx2-L microcontrollers can access a 64 KB memory space. Figures 3-1 to 3-4 show the memory maps.

**Caution Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated below after release of reset.**

**Table 3-1. Set Values of Internal Memory Size Switching Register (IMS)**

| Products | | | | IMS | ROM Capacity | Internal High-Speed RAM Capacity |
|---|---|---|---|---|---|---|
| 78K0/KY2-L | 78K0/KA2-L | 78K0/KB2-L | 78K0/KC2-L | | | |
| μPD78F0550, 78F0555 | μPD78F0560, 78F0565 | – | – | 61H | 4 KB | 384 bytes |
| μPD78F0551, 78F0556 | μPD78F0561, 78F0566 | μPD78F0571, 78F0576 | μPD78F0581, 78F0586 | 42H | 8 KB | 512 bytes |
| μPD78F0552, 78F0557 | μPD78F0562, 78F0567 | μPD78F0572, 78F0577 | μPD78F0582, 78F0587 | 04H | 16 KB | 768 bytes |
| – | – | μPD78F0573, 78F0578 | μPD78F0583, 78F0588 | C8H | 32 KB | 1 KB |

**Figure 3-1. Memory Map (μPD78F0550, 78F0555, 78F0560, 78F0565)**




**Note** Writing boot cluster 0 can be prohibited depending on the setting of security (refer to **25.6 Security Settings**).

**Remark** The flash memory is divided into blocks (one block = 1 KB). For the address values and block numbers, refer to **Table 3-2 Correspondence Between Address Values and Block Numbers in Flash Memory**.

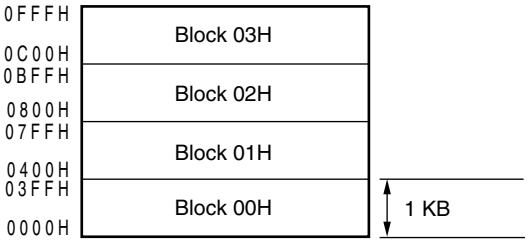
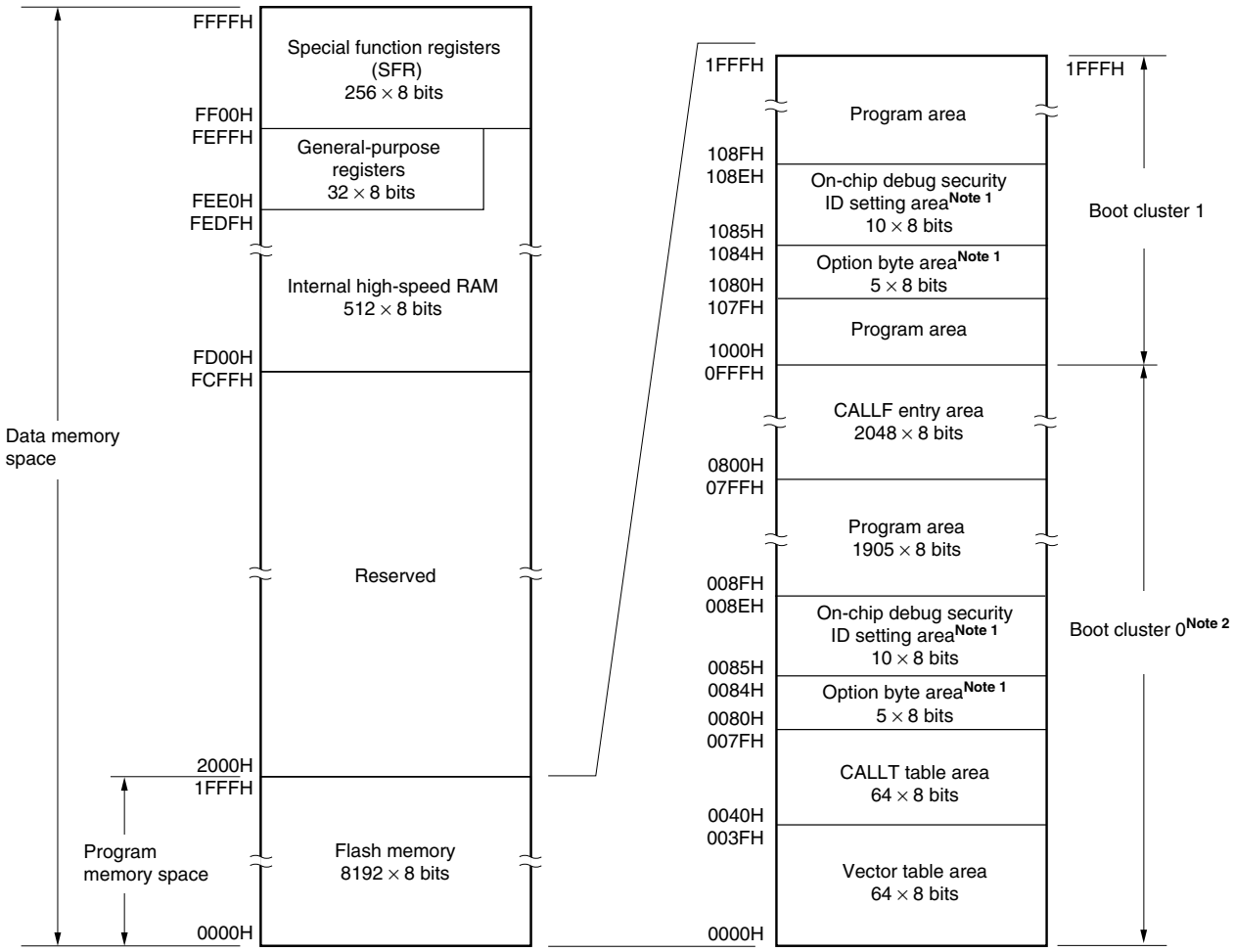


**Figure 3-2. Memory Map (μPD78F0551, 78F0556, 78F0561, 78F0566, 78F0571, 78F0576, 78F0581, 78F0586)**




**Notes 1.** When boot swap is not used: Set the option bytes to 0080H to 0084H, and the on-chip debug security IDs to 0085H to 008EH.

When boot swap is used: Set the option bytes to 0080H to 0084H and 1080H to 1084H, and the on-chip debug security IDs to 0085H to 008EH and 1085H to 108EH.

**2.** Writing boot cluster 0 can be prohibited depending on the setting of security (refer to **25.6 Security Settings**).

**Remark** The flash memory is divided into blocks (one block = 1 KB). For the address values and block numbers, refer to **Table 3-2 Correspondence Between Address Values and Block Numbers in Flash Memory**.

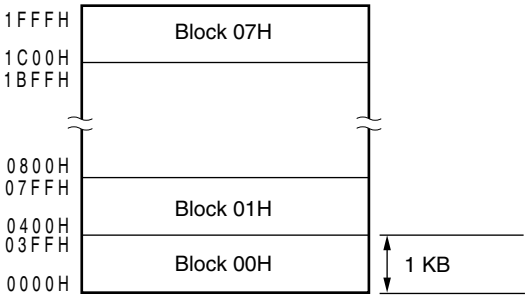
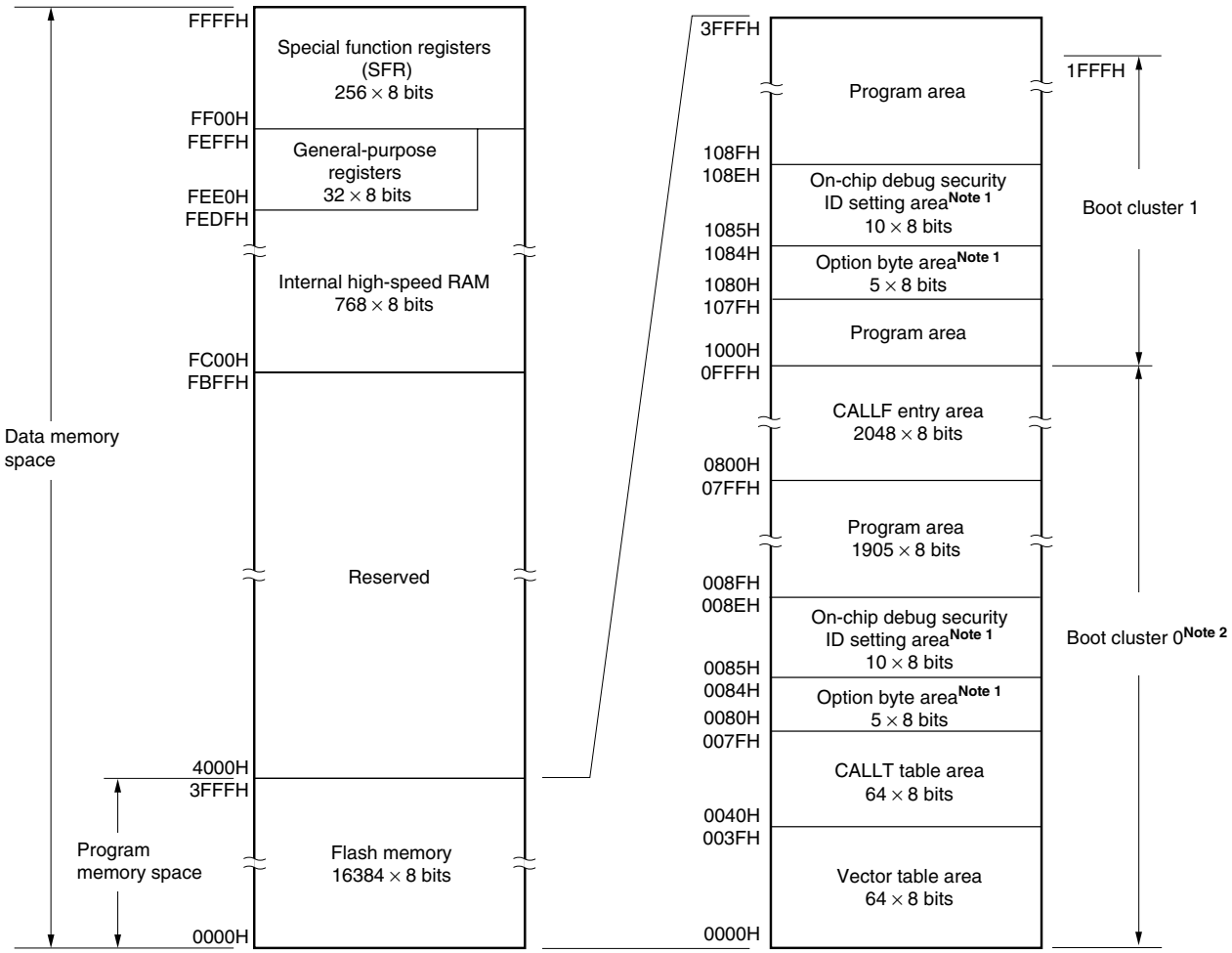


**Figure 3-3. Memory Map (μPD78F0552, 78F0557, 78F0562, 78F0567, 78F0572, 78F0577, 78F0582, 78F0587)**




**Notes 1.** When boot swap is not used: Set the option bytes to 0080H to 0084H, and the on-chip debug security IDs to 0085H to 008EH.

When boot swap is used: Set the option bytes to 0080H to 0084H and 1080H to 1084H, and the on-chip debug security IDs to 0085H to 008EH and 1085H to 108EH.

**2.** Writing boot cluster 0 can be prohibited depending on the setting of security (refer to **25.6 Security Settings**).

**Remark** The flash memory is divided into blocks (one block = 1 KB). For the address values and block numbers, refer to **Table 3-2 Correspondence Between Address Values and Block Numbers in Flash Memory**.

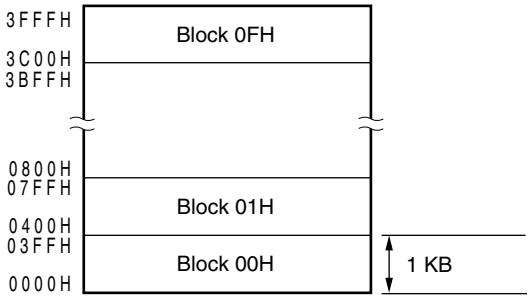
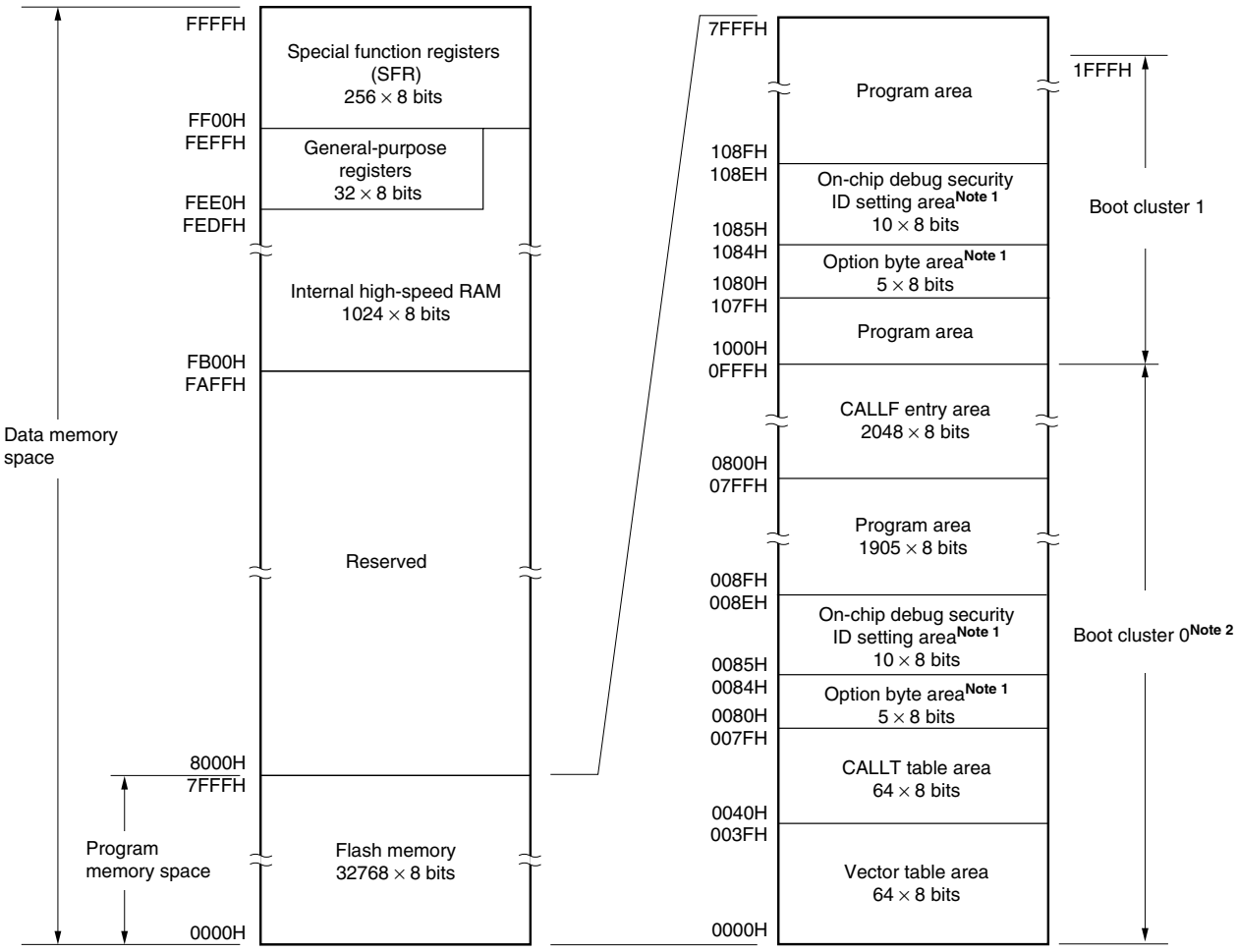


**Figure 3-4. Memory Map (μPD78F0573, 78F0578, 78F0583, 78F0588)**

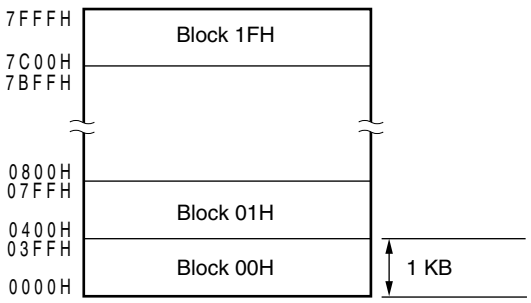



**Notes 1.** When boot swap is not used: Set the option bytes to 0080H to 0084H, and the on-chip debug security IDs to 0085H to 008EH.

When boot swap is used: Set the option bytes to 0080H to 0084H and 1080H to 1084H, and the on-chip debug security IDs to 0085H to 008EH and 1085H to 108EH.

**2.** Writing boot cluster 0 can be prohibited depending on the setting of security (refer to **25.6 Security Settings**).

**Remark** The flash memory is divided into blocks (one block = 1 KB). For the address values and block numbers, refer to **Table 3-2 Correspondence Between Address Values and Block Numbers in Flash Memory**.



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

**Table 3-2. Correspondence Between Address Values and Block Numbers in Flash Memory**

| Address Value | Block Number | Address Value | Block Number |
|---|---|---|---|
| 0000H to 03FFH | 00H | 4000H to 43FFH | 10H |
| 0400H to 07FFH | 01H | 4400H to 47FFH | 11H |
| 0800H to 0BFFH | 02H | 4800H to 4BFFH | 12H |
| 0C00H to 0FFFH | 03H | 4C00H to 4FFFH | 13H |
| 1000H to 13FFH | 04H | 5000H to 53FFH | 14H |
| 1400H to 17FFH | 05H | 5400H to 57FFH | 15H |
| 1800H to 1BFFH | 06H | 5800H to 5BFFH | 16H |
| 1C00H to 1FFFH | 07H | 5C00H to 5FFFH | 17H |
| 2000H to 23FFH | 08H | 6000H to 63FFH | 18H |
| 2400H to 27FFH | 09H | 6400H to 67FFH | 19H |
| 2800H to 2BFFH | 0AH | 6800H to 6BFFH | 1AH |
| 2C00H to 2FFFH | 0BH | 6C00H to 6FFFH | 1BH |
| 3000H to 33FFH | 0CH | 7000H to 73FFH | 1CH |
| 3400H to 37FFH | 0DH | 7400H to 77FFH | 1DH |
| 3800H to 3BFFH | 0EH | 7800H to 7BFFH | 1EH |
| 3C00H to 3FFFH | 0FH | 7C00H to 7FFFH | 1FH |

**Remark** μPD78F05x0, 78F05x5 (x = 5, 6): Block numbers 00H to 03H
μPD78F05x1, 78F05x6 (x = 5 to 8): Block numbers 00H to 07H
μPD78F05x2, 78F05x7 (x = 5 to 8): Block numbers 00H to 0FH
μPD78F05x3, 78F05x8 (x = 7, 8): Block numbers 00H to 1FH

### 3.1.1 Internal program memory space

The internal program memory space stores the program and table data. Normally, it is addressed with the program counter (PC).

78K0/Kx2-L microcontrollers incorporate internal ROM (flash memory), as shown below.

**Table 3-3. Internal ROM Capacity**

| Product | | | | Internal ROM | |
|---|---|---|---|---|---|
| 78K0/KY2-L | 78K0/KA2-L | 78K0/KB2-L | 78K0/KC2-L | Structure | Capacity |
| μPD78F0550, 78F0555 | μPD78F0560, 78F0565 | – | – | Flash memory | 4096 × 8 bits (0000H to 0FFFH) |
| μPD78F0551, 78F0556 | μPD78F0561, 78F0566 | μPD78F0571, 78F0576 | μPD78F0581, 78F0586 | | 8192 × 8 bits (0000H to 1FFFH) |
| μPD78F0552, 78F0557 | μPD78F0562, 78F0567 | μPD78F0572, 78F0577 | μPD78F0582, 78F0587 | | 16384 × 8 bits (0000H to 3FFFH) |
| – | – | μPD78F0573, 78F0578 | μPD78F0583, 78F0588 | | 32768 × 8 bits (0000H to 7FFFH) |

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

**(1) Vector table area**

The 64-byte area 0000H to 003FH is reserved as a vector table area. The program start addresses for branch upon reset or generation of each interrupt request are stored in the vector table area.

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

<R>

**Table 3-4. Vector Table**

| Vector Table Address | Interrupt Source | 78K0/KY2-L | 78K0/KA2-L | | | 78K0/KB2-L | 78K0/KC2-L | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| 0000H | $\overline{\text{RESET}}$ input, POC, LVI, WDT | √ | √ | √ | √ | √ | √ | √ | √ |
| 0004H | INTLVI | √ | √ | √ | √ | √ | √ | √ | √ |
| 0006H | INTP0 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0008H | INTP1 | √ | √ | – | – | √ | √ | √ | √ |
| 000AH | INTP2 | – | √ | √ | √ | √ | √ | √ | √ |
| 000CH | INTP3 | – | √ | √ | √ | √ | √ | √ | √ |
| 000EH | INTP4 | – | – | √ | √ | √ | √ | √ | √ |
| 0010H | INTP5 | – | – | √ | √ | √ | √ | √ | √ |
| 0012H | INTSRE6 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0014H | INTSR6 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0016H | INTST6 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0018H | INTCSI10 | – | – | – | – | √ | √ | √ | √ |
| | INTCSI11 | – | – | √ | √ | – | – | – | – |
| 001AH | INTTMH1 | √ | √ | √ | √ | √ | √ | √ | √ |
| 001CH | INTTMH0 | – | – | – | – | √ | √ | √ | √ |
| 001EH | INTTM50 | – | – | – | – | √ | √ | √ | √ |
| 0020H | INTTM000 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0022H | INTTM010 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0024H | INTAD | √ | √ | √ | √ | √ | √ | √ | √ |
| 0026H | INTP6 | – | – | – | – | – | – | – | √ |
| 0028H | INTRTCI | – | – | – | – | – | √ | √ | √ |
| 002AH | INTTM51 | √ | √ | √ | √ | √ | √ | √ | √ |
| 002CH | INTKR | – | – | – | – | – | √ | √ | √ |
| 002EH | INTRTC | – | – | – | – | – | √ | √ | √ |
| 0030H | INTP7 | – | – | – | – | – | – | – | √ |
| 0032H | INTP8 | – | – | – | – | – | – | √ | √ |
| 0034H | INTIICA0 | √ | √ | √ | √ | √ | √ | √ | √ |
| 0036H | INTCSI11 | – | – | – | – | – | √ | √ | √ |
| 0038H | INTP9 | – | – | – | – | – | √ | √ | √ |
| 003AH | INTP10 | – | – | – | – | √ | √ | √ | √ |
| 003CH | INTP11 | – | – | – | – | √ | √ | √ | √ |
| 003EH | BRK | √ | √ | √ | √ | √ | √ | √ | √ |

**Remark** √: Mounted, –: Not mounted

**(2) CALLT instruction table area**

The 64-byte area 0040H to 007FH can store the subroutine entry address of a 1-byte call instruction (CALLT).

**(3) Option byte area**

A 5-byte area of 0080H to 0084H and 1080H to 1084H can be used as an option byte area. Set the option byte at 0080H to 0084H when the boot swap is not used, and at 0080H to 0084H and 1080H to 1084H when the boot swap is used. For details, refer to **CHAPTER 24 OPTION BYTE**.

**(4) On-chip debug security ID setting area**

A 10-byte area of 0085H to 008EH and 1085H to 108EH can be used as an on-chip debug security ID setting area. Set the on-chip debug security ID of 10 bytes at 0085H to 008EH when the boot swap is not used and at 0085H to 008EH and 1085H to 108EH when the boot swap is used. For details, refer to **CHAPTER 26 ON-CHIP DEBUG FUNCTION**.

**(5) CALLF instruction entry area**

The area 0800H to 0FFFH can perform a direct subroutine call with a 2-byte call instruction (CALLF).

### 3.1.2 Internal data memory space

78K0/Kx2-L microcontrollers incorporate the following RAMs.

**(1) Internal high-speed RAM**

**Table 3-5. Internal High-Speed RAM Capacity**

| Product | | | | Internal High-Speed RAM |
|---|---|---|---|---|
| 78K0/KY2-L | 78K0/KA2-L | 78K0/KB2-L | 78K0/KC2-L | |
| μPD78F0550, 78F0555 | μPD78F0560, 78F0565 | – | – | 384 × 8 bits (FD80H to FEFFH) |
| μPD78F0551, 78F0556 | μPD78F0561, 78F0566 | μPD78F0571, 78F0576 | μPD78F0581, 78F0586 | 512 × 8 bits (FD00H to FEFFH) |
| μPD78F0552, 78F0557 | μPD78F0562, 78F0567 | μPD78F0572, 78F0577 | μPD78F0582, 78F0587 | 768 × 8 bits (FC00H to FEFFH) |
| – | – | μPD78F0573, 78F0578 | μPD78F0583, 78F0588 | 1024 × 8 bits (FB00H to FEFFH) |

The 32-byte area FEE0H to FEFFH is assigned to four general-purpose register banks consisting of eight 8-bit registers per bank.

This area cannot be used as a program area in which instructions are written and executed.

The internal high-speed RAM can also be used as a stack memory.

### 3.1.3 Special function register (SFR) area

On-chip peripheral hardware special function registers (SFRs) are allocated in the area FF00H to FFFFH (refer to **Tables 3-6 to 3-9 Special Function Register List** in **3.2.3 Special function registers (SFRs)**).

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

### 3.1.4 Data memory addressing

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

Several addressing modes are provided for addressing the memory relevant to the execution of instructions for the 78K0/Kx2-L microcontrollers, based on operability and other considerations. For areas containing data memory in particular, special addressing methods designed for the functions of special function registers (SFR) and general-purpose registers are available for use. Figures 3-5 to 3-8 show correspondence between data memory and addressing. For details of each addressing mode, refer to **3.4 Operand Address Addressing**.

**Figure 3-5. Correspondence Between Data Memory and Addressing**
**(μPD78F0550, 78F0555, 78F0560, 78F0565)**

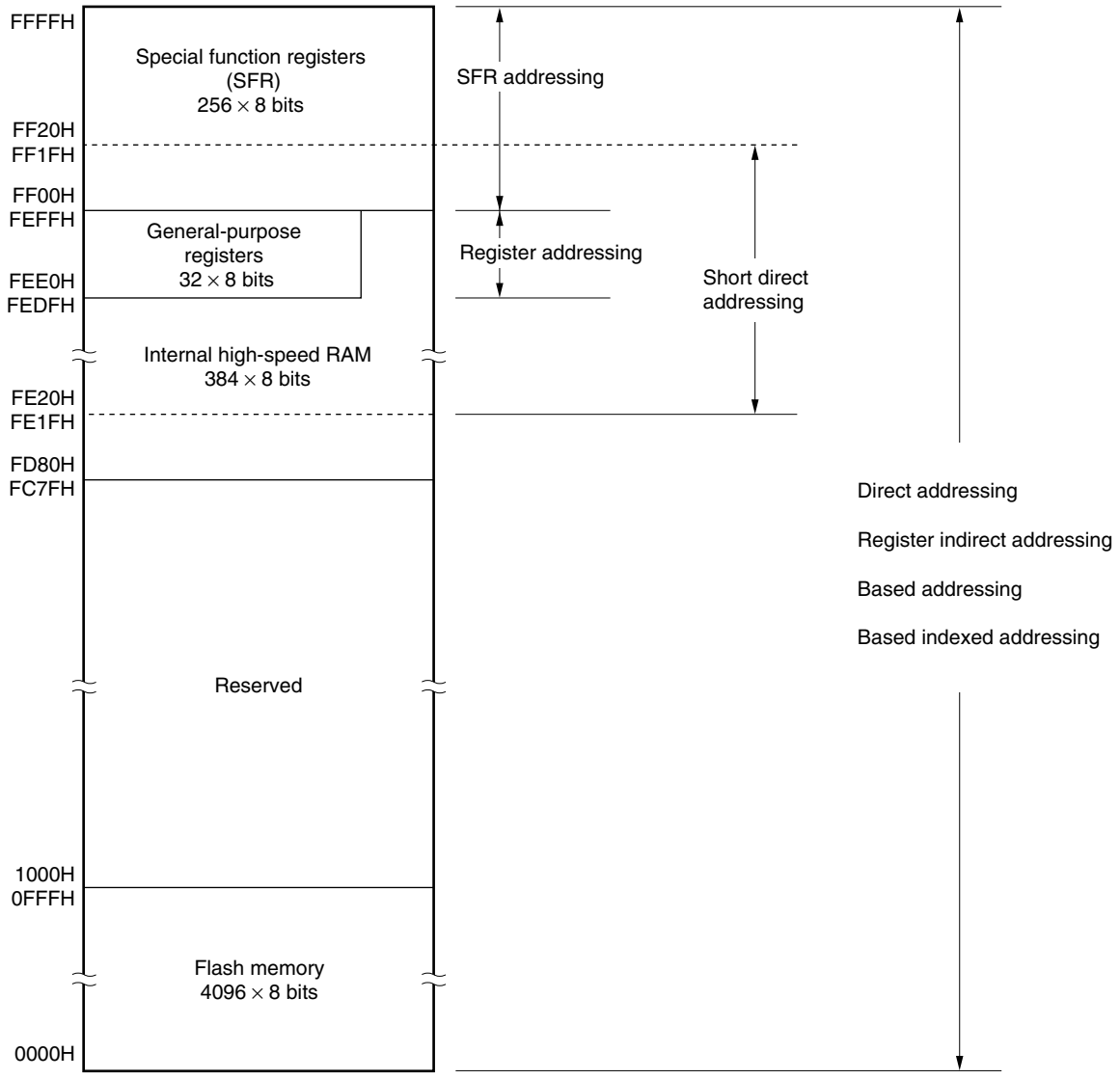


**Figure 3-6. Correspondence Between Data Memory and Addressing**
**($\mu$PD78F0551, 78F0556, 78F0561, 78F0566, 78F0571, 78F0576, 78F0581, 78F0586)**

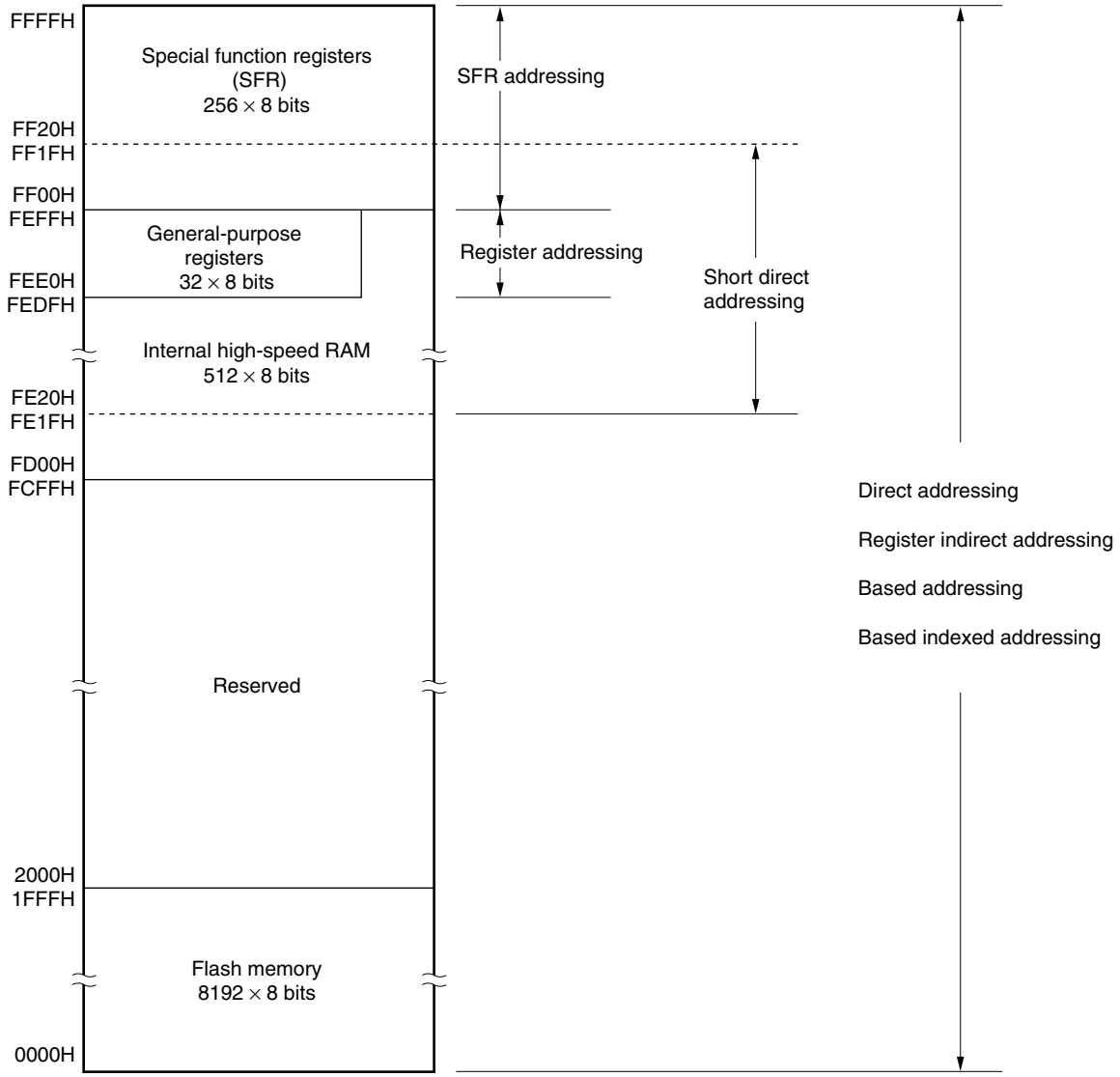


**Figure 3-7. Correspondence Between Data Memory and Addressing**
**($\mu$PD78F0552, 78F0557, 78F0562, 78F0567, 78F0572, 78F0577, 78F0582, 78F0587)**

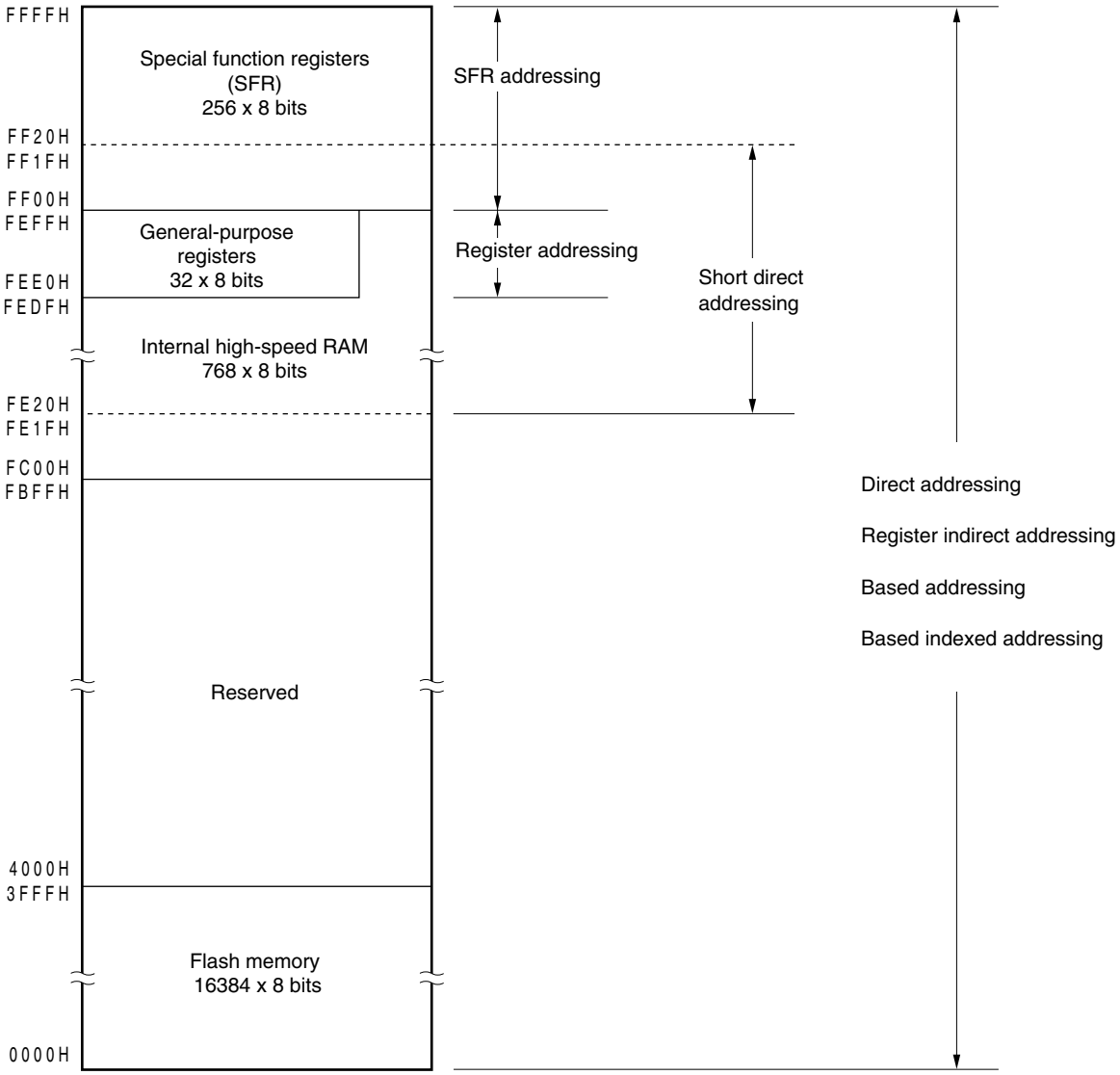
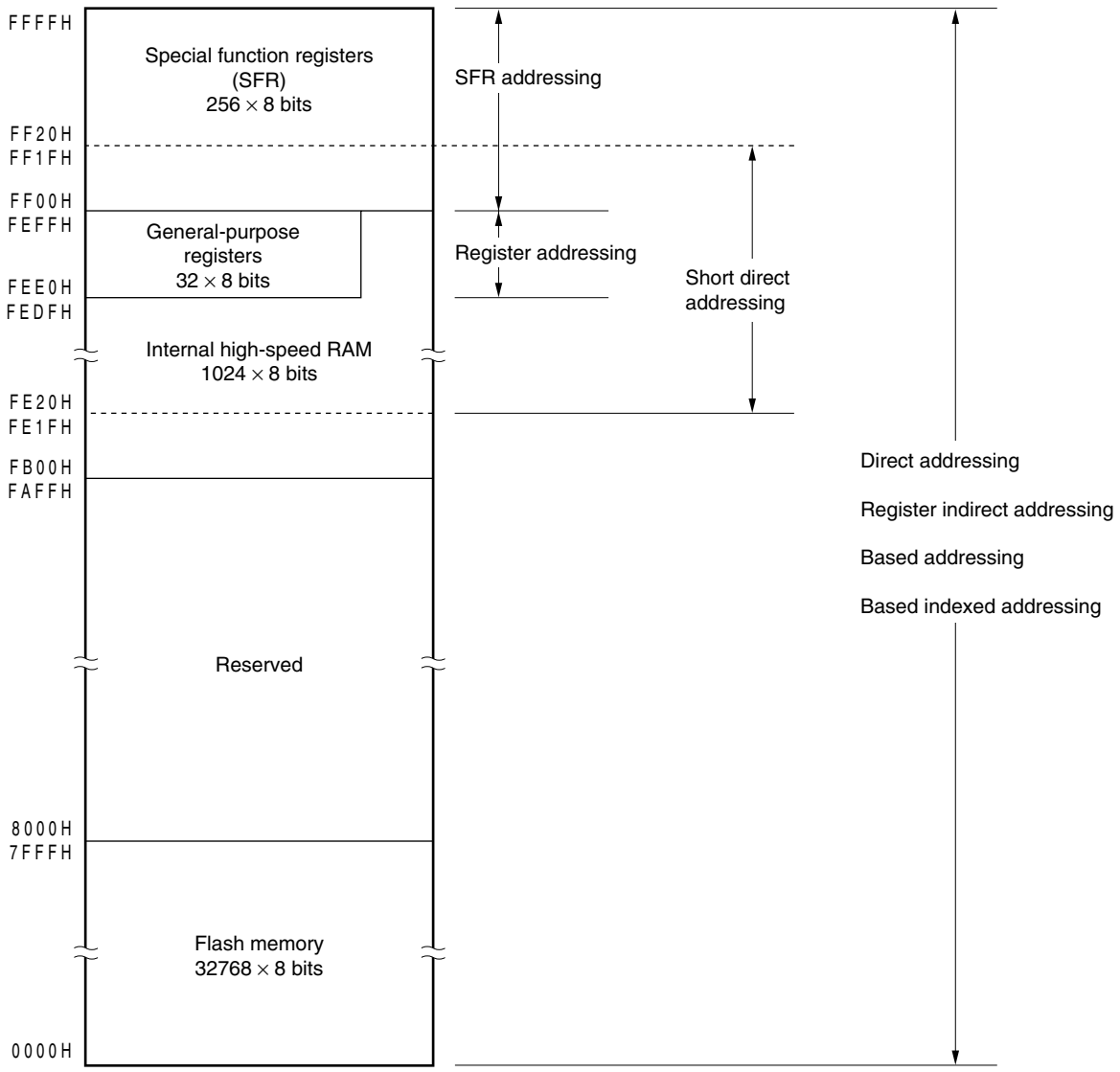


**Figure 3-8. Correspondence Between Data Memory and Addressing**
**(μPD78F0573, 78F0578, 78F0583, 78F0588)**



## 3.2 Processor Registers

The 78K0/Kx2-L microcontrollers incorporate the following processor registers.

### 3.2.1 Control registers

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

**(1) Program counter (PC)**

The program counter is a 16-bit register that holds the address information of the next program to be executed.

In normal operation, PC is automatically incremented according to the number of bytes of the instruction to be fetched.

When a branch instruction is executed, immediate data and register contents are set.

Reset signal generation sets the reset vector table values at addresses 0000H and 0001H to the program counter.

**Figure 3-9. Format of Program Counter**

| | 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC | PC15 | PC14 | PC13 | PC12 | PC11 | PC10 | PC9 | PC8 | PC7 | PC6 | PC5 | PC4 | PC3 | PC2 | PC1 | PC0 |

**(2) Program status word (PSW)**

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

Program status word contents are stored in the stack area upon vectored interrupt request acknowledge or PUSH PSW instruction execution and are restored upon execution of the RETB, RETI and POP PSW instructions.

Reset signal generation sets PSW to 02H.

**Figure 3-10. Format of Program Status Word**

| | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|
| PSW | IE | Z | RBS1 | AC | RBS0 | 0 | ISP | CY |

**(a) Interrupt enable flag (IE)**

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

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

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

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

**(b) Zero flag (Z)**

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

**(c) Register bank select flags (RBS0 and RBS1)**

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

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

**(d) Auxiliary carry flag (AC)**

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

**(e) In-service priority flag (ISP)**

This flag manages the priority of acknowledgeable maskable vectored interrupts. When this flag is 0, low-level vectored interrupt requests specified by a priority specification flag register (PR0L, PR0H, PR1L, PR1H) (refer to **17.3 (3) Priority specification flag registers (PR0L, PR0H, PR1L, PR1H)**) can not be acknowledged. Actual request acknowledgment is controlled by the interrupt enable flag (IE).

**(f) Carry flag (CY)**

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

**(3) Stack pointer (SP)**

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

**Figure 3-11. Format of Stack Pointer**

| | 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SP | SP15 | SP14 | SP13 | SP12 | SP11 | SP10 | SP9 | SP8 | SP7 | SP6 | SP5 | SP4 | SP3 | SP2 | SP1 | SP0 |

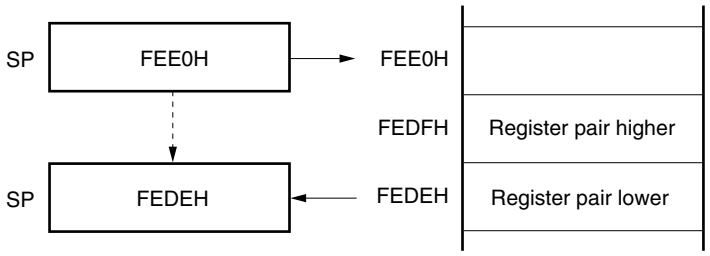
The SP is decremented ahead of write (save) to the stack memory and is incremented after read (restored) from the stack memory.

Each stack operation saves/restores data as shown in Figures 3-12 and 3-13.

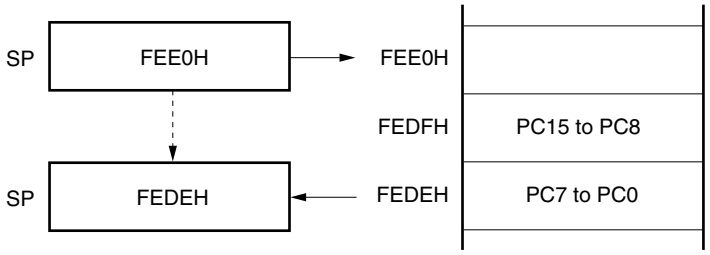
**Caution Since reset signal generation makes the SP contents undefined, be sure to initialize the SP before using the stack.**

**Figure 3-12. Data to Be Saved to Stack Memory**

**(a) PUSH rp instruction (when SP = FEE0H)**




**(b) CALL, CALLF, CALLT instructions (when SP = FEE0H)**




**(c) Interrupt, BRK instructions (when SP = FEE0H)**

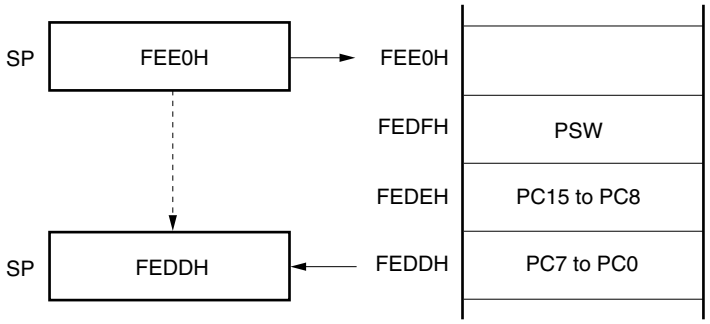
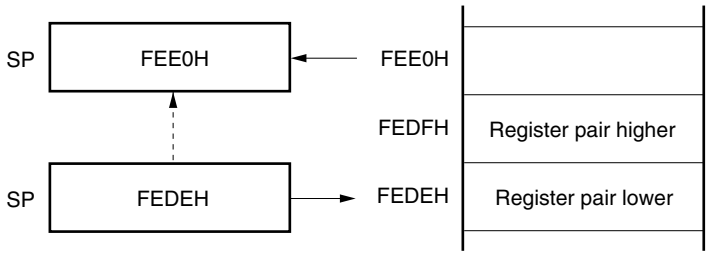
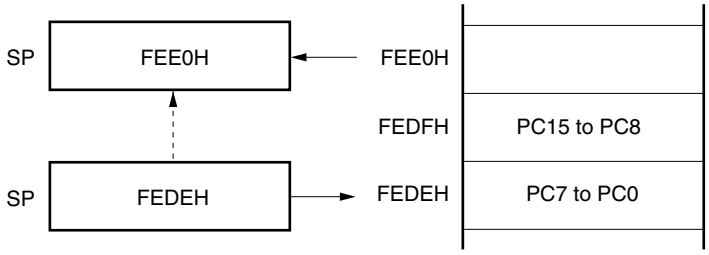


**Figure 3-13. Data to Be Restored from Stack Memory**

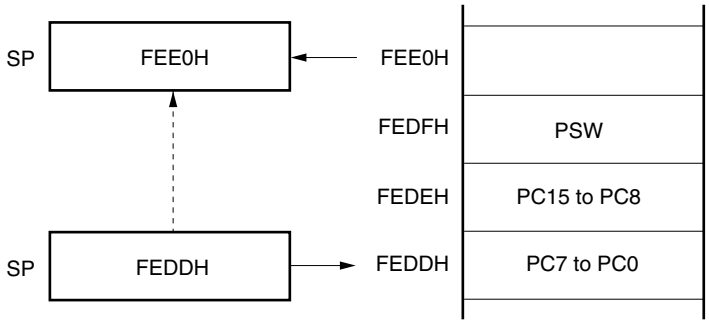
**(a) POP rp instruction (when SP = FEDEH)**




**(b) RET instruction (when SP = FEDEH)**




**(c) RETI, RETB instructions (when SP = FEDDH)**



### 3.2.2 General-purpose registers

General-purpose registers are mapped at particular addresses (FEE0H to FEFFH) of the data memory. The general-purpose registers consists of 4 banks, each bank consisting of eight 8-bit registers (X, A, C, B, E, D, L, and H).

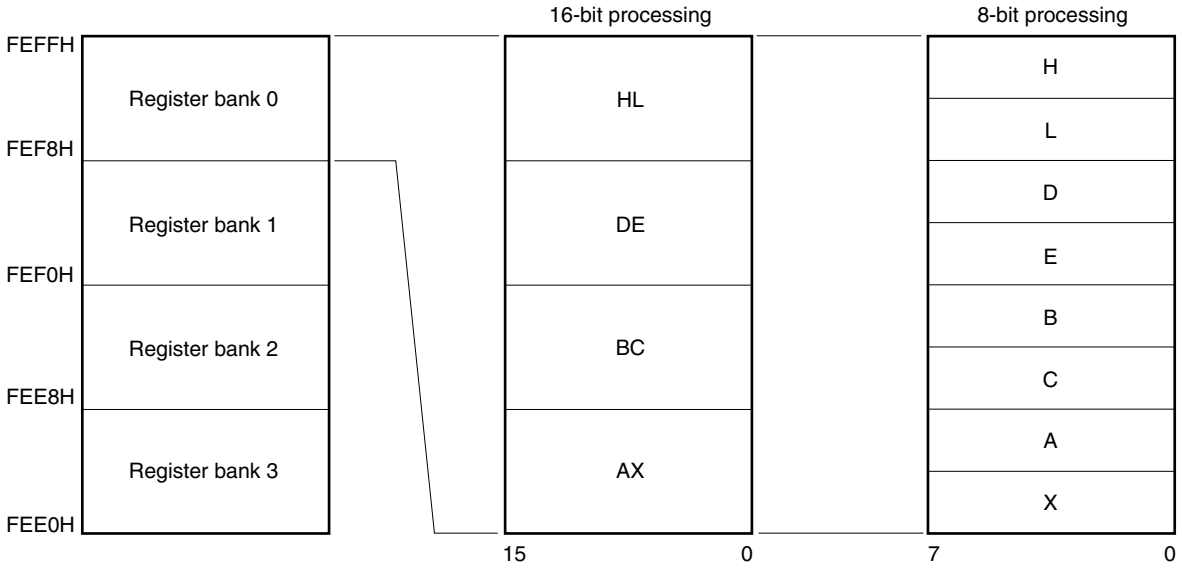
Each register can be used as an 8-bit register, and two 8-bit registers can also be used in a pair as a 16-bit register (AX, BC, DE, and HL).

These registers can be described in terms of function names (X, A, C, B, E, D, L, H, AX, BC, DE, and HL) and absolute names (R0 to R7 and RP0 to RP3).

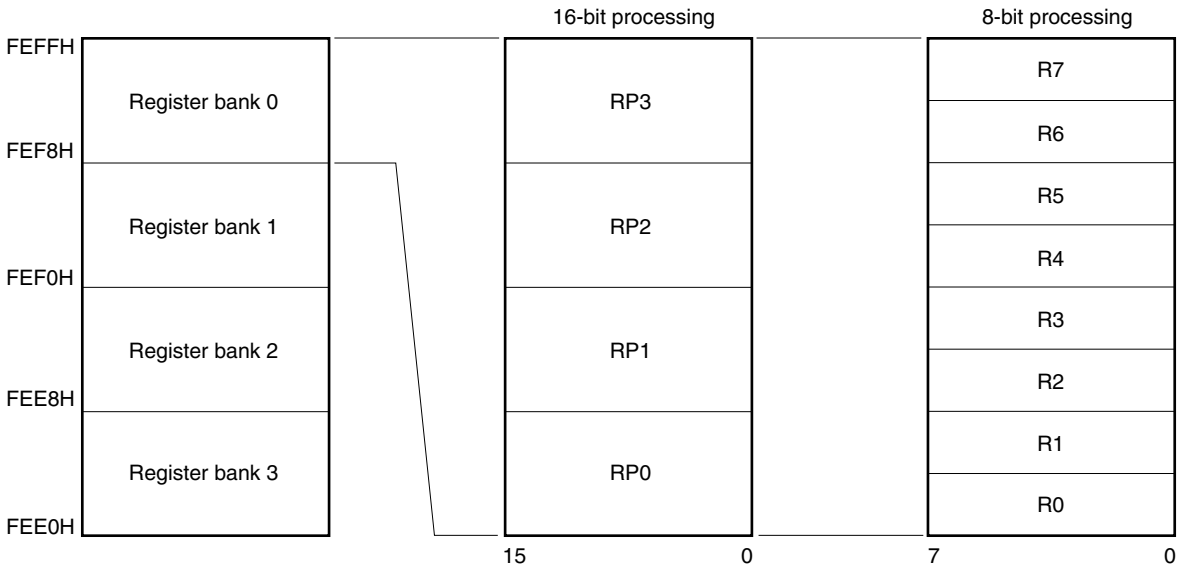
Register banks to be used for instruction execution are set by the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interrupts for each bank.

**Figure 3-14. Configuration of General-Purpose Registers**

**(a) Function name**




**(b) Absolute name**



### 3.2.3 Special function registers (SFRs)

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

SFRs are allocated to the FF00H to FFFFH area.

Special function registers can be manipulated like general-purpose registers, using operation, transfer, and bit manipulation instructions. The manipulatable bit units, 1, 8, and 16, depend on the special function register type.

Each manipulation bit unit can be specified as follows.

- 1-bit manipulation
  Describe the symbol reserved by the assembler for the 1-bit manipulation instruction operand (sfr.bit).
  This manipulation can also be specified with an address.
- 8-bit manipulation
  Describe the symbol reserved by the assembler for the 8-bit manipulation instruction operand (sfr).
  This manipulation can also be specified with an address.
- 16-bit manipulation
  Describe the symbol reserved by the assembler for the 16-bit manipulation instruction operand (sfrp).
  When specifying an address, describe an even address.

Tables 3-6 to 3-9 give lists of the special function registers. The meanings of items in the table are as follows.

- Symbol
  Symbol indicating the address of a special function register. It is a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0. When using the RA78K0, ID78K0-QB, and system simulator, symbols can be written as an instruction operand.
- R/W
  Indicates whether the corresponding special function register can be read or written.
  R/W: Read/write enable
  R: Read only
  W: Write only
- Manipulatable bit units
  Indicates the manipulatable bit unit (1, 8, or 16). "–" indicates a bit unit for which manipulation is not possible.
- After reset
  Indicates each register status upon reset signal generation.

**Table 3-6. Special Function Register List: 78K0/KY2-L (1/4)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF00H | P0 | 0 | 0 | 0 | 0 | 0 | 0 | P01 | P00 | R/W | √ | √ | – | 00H | 172 |
| FF01H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF02H | P2 | 0 | 0 | 0 | 0 | P23 | P22 | P21 | P20 | R/W | √ | √ | – | 00H | 172 |
| FF03H | P3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | P30 | R/W | √ | √ | – | 00H | 172 |
| FF04H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF05H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF06H | P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | R/W | √ | √ | – | 00H | 172 |
| FF07H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF08H | AD ADCRL | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF09H | CR | 0 | 0 | 0 | 0 | 0 | 0 | – | – | R | – | – | √ | 0000H | 410 |
| FF0AH | RXB6 | – | – | – | – | – | – | – | – | R | – | √ | – | FFH | 452 |
| FF0BH | TXB6 | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 453 |
| FF0CH | P12 | 0 | 0 | P125 | 0 | 0 | P122 | P121 | 0 | R | √ | √ | – | 00H | 172 |
| FF0DH | ADCRH | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF0EH | ADS | 0 | <ADOAS> Note | 0 | 0 | 0 | 0 | <ADS1> | <ADS0> | R/W | √ | √ | – | 00H | 412, 439 |
| FF0FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF10H | TM00 | – | – | – | – | – | – | – | – | R | – | – | √ | 0000H | 243 |
| FF11H | | – | – | – | – | – | – | – | – | | | | | | |
| FF12H | CR000 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF13H | | – | – | – | – | – | – | – | – | | | | | | |
| FF14H | CR010 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF15H | | – | – | – | – | – | – | – | – | | | | | | |
| FF16H to FF19H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1AH | CMP01 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1BH | CMP11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1CH to FF1EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1FH | TM51 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |
| FF20H | PM0 | 1 | 1 | 1 | 1 | 1 | 1 | PM01 | PM00 | R/W | √ | √ | – | FFH | 167, 256 |
| FF21H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF22H | PM2 | 1 | 1 | 1 | 1 | PM23 | PM22 | PM21 | PM20 | R/W | √ | √ | – | FFH | 167, 415, 440 |
| FF23H | PM3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM30 | R/W | √ | √ | – | FFH | 167, 324, 345 |
| FF24H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF25H | – | – | – | – | – | – | – | – | – | | – | – | – | – | – |

**Note** This bit is incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-6. Special Function Register List: 78K0/KY2-L (2/4)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF26H | PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | R/W | √ | √ | – | FFH | 167, 463, 504, 573 |
| FF27H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF28H | ADM0 | <ADCS> | 0 | FR2 | FR1 | FR0 | LV1 | LV0 | <ADCE> | R/W | √ | √ | – | 00H | 405 |
| FF29H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2AH | POM6 | 0 | 0 | 0 | 0 | 0 | 0 | POM61 | POM60 | R/W | √ | √ | – | 00H | 180, 464, 504 |
| FF2BH | FPCTL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <FLMD PUP> | R/W | √ | √ | – | 00H | 713 |
| FF2CH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2DH | RSTMASK | 0 | 0 | RSTM | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 180 |
| FF2EH | ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 | R/W | √ | √ | – | 00H | 181, 413, 437 |
| FF2FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF30H | PU0 | 0 | 0 | 0 | 0 | 0 | 0 | PU01 | PU00 | R/W | √ | √ | – | 00H | 177 |
| FF31H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF32H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF33H | PU3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | PU30 | R/W | √ | √ | – | 00H | 177 |
| FF34H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF35H | | | | | | | | | | | | | | | |
| FF36H | PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | R/W | √ | √ | – | 00H | 177 |
| FF37H to FF3BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF3CH | PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 20H | 177 |
| FF3DH | RMC | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 691 |
| FF3EH | PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 | R/W | √ | √ | – | 00H | 179, 503 |
| FF3FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF40H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF41H | CR51 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |
| FF42H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF43H | TMC51 | <TCE51> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 320 |
| FF44H to FF47H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF48H | EGPCTL0 | 0 | 0 | 0 | 0 | 0 | 0 | EGP1 | EGP0 | R/W | √ | √ | – | 00H | 619 |
| FF49H | EGNCTL0 | 0 | 0 | 0 | 0 | 0 | 0 | EGN1 | EGN0 | R/W | √ | √ | – | 00H | 619 |
| FF4AH to FF4EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF4FH | ISC | 0 | 0 | 0 | 0 | 0 | 0 | ISC1 | ISC0 | R/W | √ | √ | – | 00H | 463 |
| FF50H | ASIM6 | <POWER6> | <TXE6> | <RXE6> | PS61 | PS60 | CL6 | SL6 | ISRM6 | R/W | √ | √ | – | 01H | 454 |

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-6. Special Function Register List: 78K0/KY2-L (3/4)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF51H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF52H | | | | | | | | | | | | | | | |
| FF53H | ASIS6 | 0 | 0 | 0 | 0 | 0 | PE6 | FE6 | OVE6 | R | – | √ | – | 00H | 457 |
| FF54H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF55H | ASIF6 | 0 | 0 | 0 | 0 | 0 | 0 | TXBF6 | TXSF6 | R | – | √ | – | 00H | 458 |
| FF56H | CKSR6 | 0 | 0 | 0 | 0 | TPS63 | TPS62 | TPS61 | TPS60 | R/W | – | √ | – | 00H | 458 |
| FF57H | BRGC6 | MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 | R/W | – | √ | – | FFH | 460 |
| FF58H | ASICL6 | <SBRF6> | <SBRT6> | SBTT6 | SBL62 | SBL61 | SBL60 | DIR6 | TXDLV6 | R/W | √ | √ | – | 16H | 461 |
| FF59H to FF5FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF60H | AMP0M[Note1] | <OPAMP0E> | <PGAEN> | 0 | 0 | 0 | 0 | AMP0VG1 | AMP0VG0 | R/W | √ | √ | – | 00H | 436 |
| FF61H to FF6BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF6CH | TMHMD1 | <TMHE1> | CKS12 | CKS11 | CKS10 | TMMD11 | TMMD10 | <TOLEV1> | <TOEN1> | R/W | √ | √ | – | 00H | 339 |
| FF6DH | TMCYC1 | 0 | 0 | 0 | 0 | 0 | RMC1 | NRZB1 | <NRZ1> | R/W | √ | √ | – | 00H | 343 |
| FF6EH to FF8BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF8CH | TCL51 | 0 | 0 | 0 | 0 | 0 | TCL512 | TCL511 | TCL510 | R/W | √ | √ | – | 00H | 318 |
| FF8DH to FF98H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF99H | WDTE | – | – | – | – | – | – | – | – | R/W | – | √ | – | 1AH/9AH[Note2] | 365 |
| FF9AH to FF9EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF9FH | OSCCTL | <EXCLK> | <OSCSEL> | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 202 |
| FFA0H | RCM | <RSTS> | 0 | 0 | 0 | 0 | 0 | <LSRSTOP> | <RSTOP> | R/W | √ | √ | – | 80H[Note3] | 207 |
| FFA1H | MCM | 0 | 0 | 0 | 0 | 0 | <XSEL> | <MCS> | <MCM0> | R/W | √ | √ | – | 00H | 209 |
| FFA2H | MOC | <MSTOP> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 80H | 208 |
| FFA3H | OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | R | √ | √ | – | 00H | 210, 640 |
| FFA4H | OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 | R/W | – | √ | – | 05H | 211, 641 |
| FFA5H | IICA | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 490 |
| FFA6H | SVA0 | – | – | – | – | – | – | – | 0 | R/W | – | √ | – | 00H | 490 |

**Notes 1.** This register is incorporated only in products with operational amplifier.

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

**3.** The value of this register is 00H immediately after a reset release but automatically changes to 80H after oscillation accuracy stabilization of high-speed internal oscillator has been waited.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-6. Special Function Register List: 78K0/KY2-L (4/4)**

| Address | Symbol | | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FFA7H | IICACTL0 | | <IICE0> | <LREL0> | <WREL0> | <SPIE0> | <WTIM0> | <ACKE0> | <STT0> | <SPT0> | R/W | √ | √ | – | 00H | 492 |
| FFA8H | IICACTL1 | | <WUP> | 0 | <CLD0> | <DAD0> | <SMC0> | <DFC0> | 0 | 0 | R/W | √ | √ | – | 00H | 501 |
| FFA9H | IICAF0 | | <STCF> | <IICBSY> | 0 | 0 | 0 | 0 | <STCEN> | <IICRSV> | R/W | √ | √ | – | 00H | 499 |
| FFAAH | IICAS0 | | <MSTS0> | <ALD0> | <EXC0> | <COI0> | <TRC0> | <ACKD0> | <STD0> | <SPD0> | R | √ | √ | – | 00H | 497 |
| FFABH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFACH | RESF | | 0 | 0 | 0 | WDTRF | 0 | 0 | 0 | LVIRF | R | – | √ | – | 00H[Note1] | 664 |
| FFADH | IICWL | | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAEH | IICWH | | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAFH to FFB9H | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFBAH | TMC00 | | 0 | 0 | 0 | 0 | TMC003 | TMC002 | TMC001 | <OVF00> | R/W | √ | √ | – | 00H | 248 |
| FFBBH | PRM00 | | ES110 | ES100 | ES010 | ES000 | 0 | 0 | PRM001 | PRM000 | R/W | √ | √ | – | 00H | 253 |
| FFBCH | CRC00 | | 0 | 0 | 0 | 0 | 0 | CRC002 | CRC001 | CRC000 | R/W | √ | √ | – | 00H | 249 |
| FFBDH | TOC00 | | 0 | <OSPT00> | <OSPE00> | TOC004 | <LVS00> | <LVR00> | TOC001 | <TOE00> | R/W | √ | √ | – | 00H | 251 |
| FFBEH | LVIM | | <LVION> | 0 | 0 | 0 | 0 | 0 | <LVIMD> | <LVIF> | R/W | √ | √ | – | 00H[Note2] | 672 |
| FFBFH | LVIS | | 0 | 0 | 0 | 0 | LVIS3 | LVIS2 | LVIS1 | LVIS0 | R/W | √ | √ | – | 00H[Note3] | 675 |
| FFC0H to FFDFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFE0H | IF0 | IF0L | <SREIF6> | 0 | 0 | 0 | 0 | <PIF1> | <PIF0> | <LVIIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE1H | | IF0H | <TMIF010> | <TMIF000> | 0 | 0 | <TMIFH1> | 0 | <STIF6> | <SRIF6> | R/W | √ | √ | | 00H | 598 |
| FFE2H | IF1 | IF1L | 0 | 0 | 0 | 0 | <TMIF51> | 0 | 0 | <ADIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE3H | | IF1H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <IICAIF0> | R/W | √ | √ | | 00H | 598 |
| FFE4H | MK0 | MK0L | <SREMK6> | 1 | 1 | 1 | 1 | <PMK1> | <PMK0> | <LVIMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE5H | | MK0H | <TMMK010> | <TMMK000> | 1 | 1 | <TMMKH1> | 1 | <STMK6> | <SRMK6> | R/W | √ | √ | | FFH | 606 |
| FFE6H | MK1 | MK1L | 1 | 1 | 1 | 1 | <TMMK51> | 1 | 1 | <ADMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE7H | | MK1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | <IICAMK0> | R/W | √ | √ | | FFH | 606 |
| FFE8H | PR0 | PR0L | <SREPR6> | 1 | 1 | 1 | 1 | <PPR1> | <PPR0> | <LVIPR> | R/W | √ | √ | √ | FFH | 613 |
| FFE9H | | PR0H | <TMPR010> | <TMPR000> | 1 | 1 | <TMPRH1> | 1 | <STPR6> | <SRPR6> | R/W | √ | √ | | FFH | 613 |
| FFEAH | PR1 | PR1L | 1 | 1 | 1 | 1 | <TMPR51> | 1 | 1 | <ADPR> | R/W | √ | √ | √ | FFH | 613 |
| FFEBH | | PR1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | <IICAPR0> | R/W | √ | √ | | FFH | 613 |
| FFECH to FFEFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFF0H | IMS | | RAM2 | RAM1 | RAM0 | 0 | ROM3 | ROM2 | ROM1 | ROM0 | R/W | – | √ | – | CFH[Note4] | 699 |
| FFF1H to FFFAH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFFBH | PCC | | 0 | 0 | 0 | 0 | 0 | PCC2 | PCC1 | PCC0 | R/W | √ | √ | – | 01H | 204 |

**Notes 1.** The reset value of RESF varies depending on the reset source.

**2.** The reset values of LVIM vary depending on the reset source and setting of option byte.

**3.** The reset values of LVIS vary depending on the reset source.

**4.** Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated in Table 3-1 after release of reset.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-7. Special Function Register List: 78K0/KA2-L (20-pin products) (1/4)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF00H | P0 | 0 | 0 | 0 | 0 | 0 | 0 | P01 | P00 | R/W | √ | √ | – | 00H | 172 |
| FF01H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF02H | P2 | 0 | 0 | P25 | P24 | P23 | P22 | P21 | P20 | R/W | √ | √ | – | 00H | 172 |
| FF03H | P3 | 0 | 0 | 0 | 0 | 0 | P32 | P31 | P30 | R/W | √ | √ | – | 00H | 172 |
| FF04H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF05H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF06H | P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | R/W | √ | √ | – | 00H | 172 |
| FF07H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF08H | AD ADCRL | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF09H | CR | 0 | 0 | 0 | 0 | 0 | 0 | – | – | R | – | – | √ | 0000H | 410 |
| FF0AH | RXB6 | – | – | – | – | – | – | – | – | R | – | √ | – | FFH | 452 |
| FF0BH | TXB6 | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 453 |
| FF0CH | P12 | 0 | 0 | P125 | 0 | 0 | P122 | P121 | 0 | R | √ | √ | – | 00H | 172 |
| FF0DH | ADCRH | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF0EH | ADS | 0 | <ADOAS> Note | 0 | 0 | 0 | <ADS2> | <ADS1> | <ADS0> | R/W | √ | √ | – | 00H | 412, 439 |
| FF0FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF10H | TM00 | – | – | – | – | – | – | – | – | R | – | – | √ | 0000H | 243 |
| FF11H | | – | – | – | – | – | – | – | – | | | | | | |
| FF12H | CR000 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF13H | | – | – | – | – | – | – | – | – | | | | | | |
| FF14H | CR010 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF15H | | – | – | – | – | – | – | – | – | | | | | | |
| FF16H to FF19H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1AH | CMP01 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1BH | CMP11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1CH to FF1EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1FH | TM51 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |
| FF20H | PM0 | 1 | 1 | 1 | 1 | 1 | 1 | PM01 | PM00 | R/W | √ | √ | – | FFH | 167, 256 |
| FF21H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF22H | PM2 | 1 | 1 | PM25 | PM24 | PM23 | PM22 | PM21 | PM20 | R/W | √ | √ | – | FFH | 167, 415, 440 |
| FF23H | PM3 | 1 | 1 | 1 | 1 | 1 | PM32 | PM31 | PM30 | R/W | √ | √ | – | FFH | 167, 324, 345 |
| FF24H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF25H | – | – | – | – | – | – | – | – | – | | – | – | – | – | – |

**Note** This bit is incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-7. Special Function Register List: 78K0/KA2-L (20-pin products) (2/4)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF26H | PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | R/W | √ | √ | – | FFH | 167, 463, 504, 573 |
| FF27H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF28H | ADM0 | <ADCS> | 0 | FR2 | FR1 | FR0 | LV1 | LV0 | <ADCE> | R/W | √ | √ | – | 00H | 405 |
| FF29H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2AH | POM6 | 0 | 0 | 0 | 0 | 0 | 0 | POM61 | POM60 | R/W | √ | √ | – | 00H | 180, 464, 504 |
| FF2BH | FPCTL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <FLMDPUP> | R/W | √ | √ | – | 00H | 713 |
| FF2CH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2DH | RSTMASK | 0 | 0 | RSTM | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 180 |
| FF2EH | ADPC0 | 0 | 0 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 | R/W | √ | √ | – | 00H | 181, 413, 437 |
| FF2FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF30H | PU0 | 0 | 0 | 0 | 0 | 0 | 0 | PU01 | PU00 | R/W | √ | √ | – | 00H | 177 |
| FF31H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF32H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF33H | PU3 | 0 | 0 | 0 | 0 | 0 | PU32 | PU31 | PU30 | R/W | √ | √ | – | 00H | 177 |
| FF34H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF35H | | | | | | | | | | | | | | | |
| FF36H | PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | R/W | √ | √ | – | 00H | 177 |
| FF37H to FF3BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF3CH | PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 20H | 177 |
| FF3DH | RMC | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 691 |
| FF3EH | PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 | R/W | √ | √ | – | 00H | 179, 503 |
| FF3FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF40H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF41H | CR51 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |
| FF42H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF43H | TMC51 | <TCE51> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 320 |
| FF44H to FF47H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF48H | EGPCTL0 | 0 | 0 | 0 | 0 | EGP3 | EGP2 | EGP1 | EGP0 | R/W | √ | √ | – | 00H | 619 |
| FF49H | EGNCTL0 | 0 | 0 | 0 | 0 | EGN3 | EGN2 | EGN1 | EGN0 | R/W | √ | √ | – | 00H | 619 |
| FF4AH to FF4EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF4FH | ISC | 0 | 0 | 0 | 0 | 0 | 0 | ISC1 | ISC0 | R/W | √ | √ | – | 00H | 463 |
| FF50H | ASIM6 | <POWER6> | <TXE6> | <RXE6> | PS61 | PS60 | CL6 | SL6 | ISRM6 | R/W | √ | √ | – | 01H | 454 |

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-7. Special Function Register List: 78K0/KA2-L (20-pin products) (3/4)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF51H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF52H | | | | | | | | | | | | | | | |
| FF53H | ASIS6 | 0 | 0 | 0 | 0 | 0 | PE6 | FE6 | OVE6 | R | – | √ | – | 00H | 457 |
| FF54H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF55H | ASIF6 | 0 | 0 | 0 | 0 | 0 | 0 | TXBF6 | TXSF6 | R | – | √ | – | 00H | 458 |
| FF56H | CKSR6 | 0 | 0 | 0 | 0 | TPS63 | TPS62 | TPS61 | TPS60 | R/W | – | √ | – | 00H | 458 |
| FF57H | BRGC6 | MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 | R/W | – | √ | – | FFH | 460 |
| FF58H | ASICL6 | <SBRF6> | <SBRT6> | SBTT6 | SBL62 | SBL61 | SBL60 | DIR6 | TXDLV6 | R/W | √ | √ | – | 16H | 461 |
| FF59H to FF5FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF60H | AMP0M[Note1] | <OPAMP0E> | <PGAEN> | 0 | 0 | 0 | 0 | AMP0VG1 | AMP0VG0 | R/W | √ | √ | – | 00H | 436 |
| FF61H to FF6BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF6CH | TMHMD1 | <TMHE1> | CKS12 | CKS11 | CKS10 | TMMD11 | TMMD10 | <TOLEV1> | <TOEN1> | R/W | √ | √ | – | 00H | 339 |
| FF6DH | TMCYC1 | 0 | 0 | 0 | 0 | 0 | RMC1 | NRZB1 | <NRZ1> | R/W | √ | √ | – | 00H | 343 |
| FF6EH to FF8BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF8CH | TCL51 | 0 | 0 | 0 | 0 | 0 | TCL512 | TCL511 | TCL510 | R/W | √ | √ | – | 00H | 318 |
| FF8DH to FF98H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF99H | WDTE | – | – | – | – | – | – | – | – | R/W | – | √ | – | 1AH/9AH[Note2] | 365 |
| FF9AH to FF9EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF9FH | OSCCTL | <EXCLK> | <OSCSEL> | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 202 |
| FFA0H | RCM | <RSTS> | 0 | 0 | 0 | 0 | 0 | <LSRSTOP> | <RSTOP> | R/W | √ | √ | – | 80H[Note3] | 207 |
| FFA1H | MCM | 0 | 0 | 0 | 0 | 0 | <XSEL> | <MCS> | <MCM0> | R/W | √ | √ | – | 00H | 209 |
| FFA2H | MOC | <MSTOP> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 80H | 208 |
| FFA3H | OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | R | √ | √ | – | 00H | 210, 640 |
| FFA4H | OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 | R/W | – | √ | – | 05H | 211, 641 |
| FFA5H | IICA | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 490 |
| FFA6H | SVA0 | – | – | – | – | – | – | – | 0 | R/W | – | √ | – | 00H | 490 |

**Notes 1.** This register is incorporated only in products with operational amplifier.

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

**3.** The value of this register is 00H immediately after a reset release but automatically changes to 80H after oscillation accuracy stabilization of high-speed internal oscillator has been waited.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-7. Special Function Register List: 78K0/KA2-L (20-pin products) (4/4)**

| Address | Symbol | | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FFA7H | IICACTL0 | | <IICE0> | <LREL0> | <WREL0> | <SPIE0> | <WTIM0> | <ACKE0> | <STT0> | <SPT0> | R/W | √ | √ | – | 00H | 492 |
| FFA8H | IICACTL1 | | <WUP> | 0 | <CLD0> | <DAD0> | <SMC0> | <DFC0> | 0 | 0 | R/W | √ | √ | – | 00H | 501 |
| FFA9H | IICAF0 | | <STCF> | <IICBSY> | 0 | 0 | 0 | 0 | <STCEN> | <IICRSV> | R/W | √ | √ | – | 00H | 499 |
| FFAAH | IICAS0 | | <MSTS0> | <ALD0> | <EXC0> | <COI0> | <TRC0> | <ACKD0> | <STD0> | <SPD0> | R | √ | √ | – | 00H | 497 |
| FFABH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFACH | RESF | | 0 | 0 | 0 | WDTRF | 0 | 0 | 0 | LVIRF | R | – | √ | – | 00H[Note1] | 664 |
| FFADH | IICWL | | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAEH | IICWH | | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAFH to FFB9H | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFBAH | TMC00 | | 0 | 0 | 0 | 0 | TMC003 | TMC002 | TMC001 | <OVF00> | R/W | √ | √ | – | 00H | 248 |
| FFBBH | PRM00 | | ES110 | ES100 | ES010 | ES000 | 0 | 0 | PRM001 | PRM000 | R/W | √ | √ | – | 00H | 253 |
| FFBCH | CRC00 | | 0 | 0 | 0 | 0 | 0 | CRC002 | CRC001 | CRC000 | R/W | √ | √ | – | 00H | 249 |
| FFBDH | TOC00 | | 0 | <OSPT00> | <OSPE00> | TOC004 | <LVS00> | <LVR00> | TOC001 | <TOE00> | R/W | √ | √ | – | 00H | 251 |
| FFBEH | LVIM | | <LVION> | 0 | 0 | 0 | 0 | 0 | <LVIMD> | <LVIF> | R/W | √ | √ | – | 00H[Note2] | 672 |
| FFBFH | LVIS | | 0 | 0 | 0 | 0 | LVIS3 | LVIS2 | LVIS1 | LVIS0 | R/W | √ | √ | – | 00H[Note3] | 675 |
| FFC0H to FFDFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFE0H | IF0 | IF0L | <SREIF6> | 0 | 0 | <PIF3> | <PIF2> | <PIF1> | <PIF0> | <LVIIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE1H | | IF0H | <TMIF010> | <TMIF000> | 0 | 0 | <TMIFH1> | 0 | <STIF6> | <SRIF6> | R/W | √ | √ | | 00H | 598 |
| FFE2H | IF1 | IF1L | 0 | 0 | 0 | 0 | <TMIF51> | 0 | 0 | <ADIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE3H | | IF1H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <IICAIF0> | R/W | √ | √ | | 00H | 598 |
| FFE4H | MK0 | MK0L | <SREMK6> | 1 | 1 | <PMK3> | <PMK2> | <PMK1> | <PMK0> | <LVIMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE5H | | MK0H | <TMMK010> | <TMMK000> | 1 | 1 | <TMMKH1> | 1 | <STMK6> | <SRMK6> | R/W | √ | √ | | FFH | 606 |
| FFE6H | MK1 | MK1L | 1 | 1 | 1 | 1 | <TMMK51> | 1 | 1 | <ADMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE7H | | MK1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | <IICAMK0> | R/W | √ | √ | | FFH | 606 |
| FFE8H | PR0 | PR0L | <SREMK6> | 1 | 1 | <PPR3> | <PPR2> | <PPR1> | <PPR0> | <LVIPR> | R/W | √ | √ | √ | FFH | 613 |
| FFE9H | | PR0H | <TMPR010> | <TMPR000> | 1 | 1 | <TMPRH1> | 1 | <STPR6> | <SRPR6> | R/W | √ | √ | | FFH | 613 |
| FFEAH | PR1 | PR1L | 1 | 1 | 1 | 1 | <TMPR51> | 1 | 1 | <ADPR> | R/W | √ | √ | √ | FFH | 613 |
| FFEBH | | PR1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | <IICAPR0> | R/W | √ | √ | | FFH | 613 |
| FFECH to FFEFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFF0H | IMS | | RAM2 | RAM1 | RAM0 | 0 | ROM3 | ROM2 | ROM1 | ROM0 | R/W | – | √ | – | CFH[Note4] | 699 |
| FFF1H to FFFAH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFFBH | PCC | | 0 | 0 | 0 | 0 | 0 | PCC2 | PCC1 | PCC0 | R/W | √ | √ | – | 01H | 204 |

**Notes 1.** The reset value of RESF varies depending on the reset source.

**2.** The reset values of LVIM vary depending on the reset source and setting of option byte.

**3.** The reset values of LVIS vary depending on the reset source.

**4.** Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated in Table 3-1 after release of reset.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-8. Special Function Register List: 78K0/KA2-L (25-pin and 32-pin products) (1/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF00H | P0 | 0 | 0 | 0 | 0 | 0 | P02 | P01 Note 2 | P00 Note 1 | R/W | √ | √ | – | 00H | 172 |
| FF01H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF02H | P2 | P27 Note 2 | P26 | P25 | P24 | P23 | P22 | P21 | P20 | R/W | √ | √ | – | 00H | 172 |
| FF03H | P3 | P37 | P36 | P35 | P34 | P33 | P32 | P31 | 0 | R/W | √ | √ | – | 00H | 172 |
| FF04H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF05H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF06H | P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | R/W | √ | √ | – | 00H | 172 |
| FF07H | P7 Note 2 | 0 | 0 | 0 | 0 | 0 | P72 Note 2 | P71 Note 2 | P70 Note 2 | R/W | √ | √ | – | 00H | 172 |
| FF08H | AD ADCRL | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF09H | CR | 0 | 0 | 0 | 0 | 0 | 0 | – | – | R | – | – | √ | 0000H | 410 |
| FF0AH | RXB6 | – | – | – | – | – | – | – | – | R | – | √ | – | FFH | 452 |
| FF0BH | TXB6 | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 453 |
| FF0CH | P12 | 0 | 0 | P125 | 0 | 0 | P122 | P121 | 0 | R | √ | √ | – | 00H | 172 |
| FF0DH | ADCRH | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF0EH | ADS | 0 | <ADOAS> Note 3 | 0 | 0 | <ADS3> Note 3 | <ADS2> | <ADS1> | <ADS0> | R/W | √ | √ | – | 00H | 412, 439 |
| FF0FH | SIO11 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 566 |
| FF10H | TM00 | – | – | – | – | – | – | – | – | R | – | – | √ | 0000H | 243 |
| FF11H | | – | – | – | – | – | – | – | – | | | | | | |
| FF12H | CR000 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF13H | | – | – | – | – | – | – | – | – | | | | | | |
| FF14H | CR010 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF15H | | – | – | – | – | – | – | – | – | | | | | | |
| FF16H to FF19H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1AH | CMP01 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1BH | CMP11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1CH to FF1EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1FH | TM51 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |
| FF20H | PM0 | 1 | 1 | 1 | 1 | 1 | PM02 | PM01 Note 2 | PM00 Note 1 | R/W | √ | √ | – | FFH | 167, 256 |
| FF21H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF22H | PM2 | PM27 Note 2 | PM26 | PM25 | PM24 | PM23 | PM22 | PM21 | PM20 | R/W | √ | √ | – | FFH | 167, 415, 440 |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** This bit is incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-8. Special Function Register List: 78K0/KA2-L (25-pin and 32-pin products) (2/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF23H | PM3 | PM37 | PM36 | PM35 | PM34 | PM33 | PM32 | PM31 | 1 | R/W | √ | √ | – | FFH | 167, 324, 345 |
| FF24H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF25H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF26H | PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | R/W | √ | √ | – | FFH | 167, 463, 504, 573 |
| FF27H | P7 Note2 | 1 | 1 | 1 | 1 | 1 | P72 Note 2 | P71 Note 2 | P70 Note 2 | R/W | √ | √ | – | 00H | 167 |
| FF28H | ADM0 | <ADCS> | 0 | FR2 | FR1 | FR0 | LV1 | LV0 | <ADCE> | R/W | √ | √ | – | 00H | 405 |
| FF29H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2AH | POM6 | 0 | 0 | 0 | 0 | 0 | 0 | POM61 | POM60 | R/W | √ | √ | – | 00H | 180, 464, 504 |
| FF2BH | FPCTL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <FLMD PUP> | R/W | √ | √ | – | 00H | 713 |
| FF2CH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2DH | RSTMASK | 0 | 0 | RSTM | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 180 |
| FF2EH | ADPC0 | ADPCS7 Note 2 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 | R/W | √ | √ | – | 00H | 181, 413, 437 |
| FF2FH | ADPC1 Note 2 | 0 | 0 | 0 | 0 | 0 | ADPCS10 Note 2 | ADPCS9 Note 2 | ADPCS8 Note 2 | R/W | √ | √ | – | 00H | |
| FF30H | PU0 | 0 | 0 | 0 | 0 | 0 | PU02 | PU01 Note 2 | PU00 Note 1 | R/W | √ | √ | – | 00H | 177 |
| FF31H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF32H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF33H | PU3 | PU37 | PU36 | PU35 | PU34 | PU33 | PU32 | PU31 | 0 | R/W | √ | √ | – | 00H | 177 |
| FF34H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF35H | | | | | | | | | | | | | | | |
| FF36H | PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | R/W | √ | √ | – | 00H | 177 |
| FF37H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF38H | | | | | | | | | | | | | | | |
| FF39H | MUXSEL | <INTP0SEL1> Note 2 | <INTP0SEL0> | <TM00SEL1> Note 2 | <TM00SEL0> | <TM5SEL1> Note 1 | <TM5SEL0> Note 1 | <TMHSEL1> Note 1 | <TMHSEL0> | R/W | √ | √ | – | 00H | 183, 572 |
| FF3AH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF3BH | | | | | | | | | | | | | | | |
| FF3CH | PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 20H | 177 |
| FF3DH | RMC | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 691 |
| FF3EH | PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 | R/W | √ | √ | – | 00H | 179, 503 |
| FF3FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF40H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF41H | CR51 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |

**Notes** **1.** 25-pin products only

**2.** 32-pin products only

**3.** This bit is incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-8. Special Function Register List: 78K0/KA2-L (25-pin and 32-pin products) (3/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF42H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF43H | TMC51 | <TCE51> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 320 |
| FF44H to FF47H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF48H | EGPCTL0 | 0 | 0 | EGP5 | EGP4 | EGP3 | EGP2 | 0 | EGP0 | R/W | √ | √ | – | 00H | 619 |
| FF49H | EGNCTL0 | 0 | 0 | EGN5 | EGN4 | EGN3 | EGN2 | 0 | EGN0 | R/W | √ | √ | – | 00H | 619 |
| FF4AH to FF4EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF4FH | ISC | 0 | 0 | 0 | 0 | 0 | 0 | ISC1 | ISC0 | R/W | √ | √ | – | 00H | 463 |
| FF50H | ASIM6 | <POWER6> | <TXE6> | <RXE6> | PS61 | PS60 | CL6 | SL6 | ISRM6 | R/W | √ | √ | – | 01H | 454 |
| FF51H FF52H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF53H | ASIS6 | 0 | 0 | 0 | 0 | 0 | PE6 | FE6 | OVE6 | R | – | √ | – | 00H | 457 |
| FF54H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF55H | ASIF6 | 0 | 0 | 0 | 0 | 0 | 0 | TXBF6 | TXSF6 | R | – | √ | – | 00H | 458 |
| FF56H | CKSR6 | 0 | 0 | 0 | 0 | TPS63 | TPS62 | TPS61 | TPS60 | R/W | – | √ | – | 00H | 458 |
| FF57H | BRGC6 | MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 | R/W | – | √ | – | FFH | 460 |
| FF58H | ASICL6 | <SBRF6> | <SBRT6> | SBTT6 | SBL62 | SBL61 | SBL60 | DIR6 | TXDLV6 | R/W | √ | √ | – | 16H | 461 |
| FF59H to FF5FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF60H | AMP0M Note 1 | <OPAMP0E> | <PGAEN> | 0 | 0 | 0 | 0 | <AMP0VG1> | <AMP0VG0> | R/W | √ | √ | – | 00H | 436 |
| FF61H to FF6BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF6CH | TMHMD1 | <TMHE1> | CKS12 | CKS11 | CKS10 | TMMD11 | TMMD10 | <TOLEV1> | <TOEN1> | R/W | √ | √ | – | 00H | 339 |
| FF6DH | TMCYC1 | 0 | 0 | 0 | 0 | 0 | RMC1 | NRZB1 | <NRZ1> | R/W | √ | √ | – | 00H | 343 |
| FF6EH to FF7BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF7CH | SOTB11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 565 |
| FF7DH to FF87H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF88H | CSIM11 | <CSIE11> | TRMD11 | SSE11 | DIR11 | 0 | 0 | 0 | CSOT11 | R/W | √ | √ | – | 00H | 566 |
| FF89H | CSIC11 | 0 | 0 | 0 | CKP11 | DAP11 | CKS102 | CKS101 | CKS100 | R/W | √ | √ | – | 00H | 569 |
| FF8AH to FF8BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF8CH | TCL51 | 0 | 0 | 0 | 0 | 0 | TCL512 | TCL511 | TCL510 | R/W | √ | √ | – | 00H | 318 |

**Notes 1.** This bit is incorporated only in products with operational amplifier y

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

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-8. Special Function Register List: 78K0/KA2-L (25-pin and 32-pin products) (4/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF8DH to FF98H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF99H | WDTE | – | – | – | – | – | – | – | – | R/W | – | √ | – | 1AH/ 9AH Note1 | 365 |
| FF9AH to FF9EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF9FH | OSCCTL | <EXCLK> | <OSCSEL> | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 202 |
| FFA0H | RCM | <RSTS> | 0 | 0 | 0 | 0 | 0 | <LSRSTOP> | <RSTOP> | R/W | √ | √ | – | 80H Note2 | 207 |
| FFA1H | MCM | 0 | 0 | 0 | 0 | 0 | <XSEL> | <MCS> | <MCM0> | R/W | √ | √ | – | 00H | 209 |
| FFA2H | MOC | <MSTOP> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 80H | 208 |
| FFA3H | OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | R | √ | √ | – | 00H | 210, 640 |
| FFA4H | OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 | R/W | – | √ | – | 05H | 211, 641 |
| FFA5H | IICA | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 490 |
| FFA6H | SVA0 | – | – | – | – | – | – | – | 0 | R/W | – | √ | – | 00H | 490 |
| FFA7H | IICACTL0 | <IICE0> | <LREL0> | <WREL0> | <SPIE0> | <WTIM0> | <ACKE0> | <STT0> | <SPT0> | R/W | √ | √ | – | 00H | 492 |
| FFA8H | IICACTL1 | <WUP> | 0 | <CLD0> | <DAD0> | <SMC0> | <DFC0> | 0 | 0 | R/W | √ | √ | – | 00H | 501 |
| FFA9H | IICAF0 | <STCF> | <IICBSY> | 0 | 0 | 0 | 0 | <STCEN> | <IICRSV> | R/W | √ | √ | – | 00H | 499 |
| FFAAH | IICAS0 | <MSTS0> | <ALD0> | <EXC0> | <COI0> | <TRC0> | <ACKD0> | <STD0> | <SPD0> | R | √ | √ | – | 00H | 497 |
| FFABH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFACH | RESF | 0 | 0 | 0 | WDTRF | 0 | 0 | 0 | LVIRF | R | – | √ | – | 00H Note3 | 664 |
| FFADH | IICWL | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAEH | IICWH | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAFH to FFB9H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFBAH | TMC00 | 0 | 0 | 0 | 0 | TMC003 | TMC002 | TMC001 | <OVF00> | R/W | √ | √ | – | 00H | 248 |
| FFBBH | PRM00 | ES110 | ES100 | ES010 | ES000 | 0 | 0 | PRM001 | PRM000 | R/W | √ | √ | – | 00H | 253 |
| FFBCH | CRC00 | 0 | 0 | 0 | 0 | 0 | CRC002 | CRC001 | CRC000 | R/W | √ | √ | – | 00H | 249 |
| FFBDH | TOC00 Note4 | 0 | <OSPT00> | <OSPE00> | TOC004 | <LVS00> | <LVR00> | TOC001 | <TOE00> | R/W | √ | √ | – | 00H | 251 |
| FFBEH | LVIM | <LVION> | 0 | 0 | 0 | 0 | 0 | <LVIMD> | <LVIF> | R/W | √ | √ | – | 00H Note5 | 672 |
| FFBFH | LVIS | 0 | 0 | 0 | 0 | LVIS3 | LVIS2 | LVIS1 | LVIS0 | R/W | √ | √ | – | 00H Note6 | 675 |

**Notes 1.** The reset value of WDTE is determined by setting of option byte.

**2.** The value of this register is 00H immediately after a reset release but automatically changes to 80H after oscillation accuracy stabilization of high-speed internal oscillator has been waited.

**3.** The reset value of RESF varies depending on the reset source.

**4.** 32-pin products only

**5.** The reset values of LVIM vary depending on the reset source and setting of option byte.

**6.** The reset value of LVIS varies depending on the reset source.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-8. Special Function Register List: 78K0/KA2-L (25-pin and 32-pin products) (5/5)**

| Address | Symbol | | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FFC0H to FFDFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFE0H | IF0 | IF0L | <SREIF6> | <PIF5> | <PIF4> | <PIF3> | <PIF2> | 0 | <PIF0> | <LVIIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE1H | | IF0H | <TMIF010> | <TMIF000> | 0 | 0 | <TMIFH1> | <CSIIF10> | <STIF6> | <SRIF6> | R/W | √ | √ | | 00H | 598 |
| FFE2H | IF1 | IF1L | 0 | 0 | 0 | 0 | <TMIF51> | 0 | 0 | <ADIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE3H | | IF1H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <IICAIF0> | R/W | √ | √ | | 00H | 598 |
| FFE4H | MK0 | MK0L | <SREMK6> | <PMK5> | <PMK4> | <PMK3> | <PMK2> | 1 | <PMK0> | <LVIMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE5H | | MK0H | <TMMK010> | <TMMK000> | 1 | 1 | <TMMKH1> | <CSIMK10 | <STMK6> | <SRMK6> | R/W | √ | √ | | FFH | 606 |
| FFE6H | MK1 | MK1L | 1 | 1 | 1 | 1 | <TMMK51> | 1 | 1 | <ADMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE7H | | MK1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | <IICAMK0> | R/W | √ | √ | | FFH | 606 |
| FFE8H | PR0 | PR0L | <SREMK6> | <PPR5> | <PPR4> | <PPR3> | <PPR2> | 1 | <PPR0> | <LVIPR> | R/W | √ | √ | √ | FFH | 613 |
| FFE9H | | PR0H | <TMPR010> | <TMPR000> | 1 | 1 | <TMPRH1> | <CSIPR10> | <STPR6> | <SRPR6> | R/W | √ | √ | | FFH | 613 |
| FFEAH | PR1 | PR1L | 1 | 1 | 1 | 1 | <TMPR51> | 1 | 1 | <ADPR> | R/W | √ | √ | √ | FFH | 613 |
| FFEBH | | PR1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | <IICAPR0> | R/W | √ | √ | | FFH | 613 |
| FFECH to FFEFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFF0H | IMS | | RAM2 | RAM1 | RAM0 | 0 | ROM3 | ROM2 | ROM1 | ROM0 | R/W | – | √ | – | CFH[Note] | 699 |
| FFF1H to FFFAH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFFBH | PCC | | 0 | 0 | 0 | 0 | 0 | PCC2 | PCC1 | PCC0 | R/W | √ | √ | – | 01H | 204 |

**Note** Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated in Table 3-1 after release of reset.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-9. Special Function Register List: 78K0/KB2-L (1/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF00H | P0 | 0 | 0 | 0 | 0 | 0 | 0 | P01 | P00 | R/W | √ | √ | – | 00H | 172 |
| FF01H | P1 | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 | R/W | √ | √ | – | 00H | 172 |
| FF02H | P2 | 0 | 0 | 0 | 0 | P23 | P22 | P21 | P20 | R/W | √ | √ | – | 00H | 172 |
| FF03H | P3 | 0 | 0 | 0 | 0 | P33 | P32 | P31 | P30 | R/W | √ | √ | – | 00H | 172 |
| FF04H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF05H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF06H | P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | R/W | √ | √ | – | 00H | 172 |
| FF07H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF08H | AD ADCRL | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF09H | CR | 0 | 0 | 0 | 0 | 0 | 0 | – | – | R | – | – | √ | 0000H | 410 |
| FF0AH | RXB6 | – | – | – | – | – | – | – | – | R | – | √ | – | FFH | 452 |
| FF0BH | TXB6 | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 453 |
| FF0CH | P12 | 0 | 0 | P125 | 0 | 0 | P122 | P121 | P120 | R Note1 | √ | √ | – | 00H | 172 |
| FF0DH | ADCRH | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF0EH | ADS | 0 | <ADOAS> Note2 | 0 | 0 | <ADS3> | <ADS2> | <ADS1> | <ADS0> | R/W | √ | √ | – | 00H | 412, 439 |
| FF0FH | SIO10 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 566 |
| FF10H | TM00 | – | – | – | – | – | – | – | – | R | – | – | √ | 0000H | 243 |
| FF11H | | – | – | – | – | – | – | – | – | | | | | | |
| FF12H | CR000 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF13H | | – | – | – | – | – | – | – | – | | | | | | |
| FF14H | CR010 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF15H | | – | – | – | – | – | – | – | – | | | | | | |
| FF16H | TM50 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |
| FF17H | CR50 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |
| FF18H | CMP00 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF19H | CMP10 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1AH | CMP01 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1BH | CMP11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1CH to FF1EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1FH | TM51 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |
| FF20H | PM0 | 1 | 1 | 1 | 1 | 1 | 1 | PM01 | PM00 | R/W | √ | √ | – | FFH | 167, 256 |
| FF21H | PM1 | PM17 | PM16 | PM15 | PM14 | PM13 | PM12 | PM11 | PM10 | R/W | √ | √ | – | FFH | 167, 324, 345, 415, 440, 463, 573 |
| FF22H | PM2 | 1 | 1 | 1 | 1 | PM23 | PM22 | PM21 | PM20 | R/W | √ | √ | – | FFH | 167, 415, 440 |

**Notes 1.** Only P120 is R/W.

**2.** This bit is incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-9. Special Function Register List: 78K0/KB2-L (2/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF23H | PM3 | 1 | 1 | 1 | 1 | PM33 | PM32 | PM31 | PM30 | R/W | √ | √ | – | FFH | 167, 324, 345 |
| FF24H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF25H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF26H | PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | R/W | √ | √ | – | FFH | 167, 463, 504, 573 |
| FF27H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF28H | ADM0 | <ADCS> | 0 | FR2 | FR1 | FR0 | LV1 | LV0 | <ADCE> | R/W | √ | √ | – | 00H | 405 |
| FF29H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2AH | POM6 | 0 | 0 | 0 | 0 | 0 | 0 | POM61 | POM60 | R/W | √ | √ | – | 00H | 180, 464, 504 |
| FF2BH | FPCTL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <FLMDPUP> | R/W | √ | √ | – | 00H | 713 |
| FF2CH | PM12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM120 | R/W | √ | √ | – | FFH | 167, 573, 676 |
| FF2DH | RSTMASK | 0 | 0 | RSTM | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 180 |
| FF2EH | ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 | R/W | √ | √ | – | 00H | 181, 413, 437 |
| FF2FH | ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 | R/W | √ | √ | – | 07H | |
| FF30H | PU0 | 0 | 0 | 0 | 0 | 0 | 0 | PU01 | PU00 | R/W | √ | √ | – | 00H | 177 |
| FF31H | PU1 | PU17 | PU16 | PU15 | PU14 | PU13 | PU12 | PU11 | PU10 | R/W | √ | √ | – | 00H | 177 |
| FF32H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF33H | PU3 | 0 | 0 | 0 | 0 | PU33 | PU32 | PU31 | PU30 | R/W | √ | √ | – | 00H | 177 |
| FF34H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF35H | | | | | | | | | | | | | | | |
| FF36H | PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | R/W | √ | √ | – | 00H | 177 |
| FF37H to FF3BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF3CH | PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | PU120 | R/W | √ | √ | – | 20H | 177 |
| FF3DH | RMC | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 691 |
| FF3EH | PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 | R/W | √ | √ | – | 00H | 179, 503 |
| FF3FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF40H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF41H | CR51 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |
| FF42H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF43H | TMC51 | <TCE51> | TMC516 | 0 | 0 | <LVS51> | <LVR51> | <TMC511> | <TOE51> | R/W | √ | √ | – | 00H | 320 |
| FF44H to FF47H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF48H | EGPCTL0 | 0 | 0 | EGP5 | EGP4 | EGP3 | EGP2 | EGP1 | EGP0 | R/W | √ | √ | – | 00H | 619 |
| FF49H | EGNCTL0 | 0 | 0 | EGN5 | EGN4 | EGN3 | EGN2 | EGN1 | EGN0 | R/W | √ | √ | – | 00H | 619 |
| FF4AH | EGPCTL1 | 0 | 0 | 0 | 0 | EGP11 | EGP10 | 0 | 0 | R/W | √ | √ | – | 00H | 619 |
| FF4BH | EGNCTL1 | 0 | 0 | 0 | 0 | EGN11 | EGN10 | 0 | 0 | R/W | √ | √ | – | 00H | 619 |

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-9. Special Function Register List: 78K0/KB2-L (3/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF4CH to FF4EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF4FH | ISC | 0 | 0 | 0 | 0 | 0 | 0 | ISC1 | ISC0 | R/W | √ | √ | – | 00H | 463 |
| FF50H | ASIM6 | <POWER6> | <TXE6> | <RXE6> | PS61 | PS60 | CL6 | SL6 | ISRM6 | R/W | √ | √ | – | 01H | 454 |
| FF51H FF52H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF53H | ASIS6 | 0 | 0 | 0 | 0 | 0 | PE6 | FE6 | OVE6 | R | – | √ | – | 00H | 457 |
| FF54H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF55H | ASIF6 | 0 | 0 | 0 | 0 | 0 | 0 | TXBF6 | TXSF6 | R | – | √ | – | 00H | 458 |
| FF56H | CKSR6 | 0 | 0 | 0 | 0 | TPS63 | TPS62 | TPS61 | TPS60 | R/W | – | √ | – | 00H | 458 |
| FF57H | BRGC6 | MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 | R/W | – | √ | – | FFH | 460 |
| FF58H | ASICL6 | <SBRF6> | <SBRT6> | SBTT6 | SBL62 | SBL61 | SBL60 | DIR6 | TXDLV6 | R/W | √ | √ | – | 16H | 461 |
| FF59H to FF5FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF60H | AMP0M[Note] | <OPAMP0E> | <PGAEN> | 0 | 0 | 0 | 0 | AMP0VG1 | AMP0VG0 | R/W | √ | √ | – | 00H | 436 |
| FF61H | AMP1M[Note] | <OPAMP1E> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 436 |
| FF62H to FF68H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF69H | TMHMD0 | <TMHE0> | CKS02 | CKS01 | CKS00 | TMMD01 | TMMD00 | <TOLEV0> | <TOEN0> | R/W | √ | √ | – | 00H | 339 |
| FF6AH | TCL50 | 0 | 0 | 0 | 0 | 0 | TCL502 | TCL501 | TCL500 | R/W | √ | √ | – | 00H | 318 |
| FF6BH | TMC50 | <TCE50> | TMC506 | 0 | 0 | <LVS50> | <LVR50> | TMC501 | <TOE50> | R/W | √ | √ | – | 00H | 320 |
| FF6CH | TMHMD1 | <TMHE1> | CKS12 | CKS11 | CKS10 | TMMD11 | TMMD10 | <TOLEV1> | <TOEN1> | R/W | √ | √ | – | 00H | 339 |
| FF6DH | TMCYC1 | 0 | 0 | 0 | 0 | 0 | RMC1 | NRZB1 | <NRZ1> | R/W | √ | √ | – | 00H | 343 |
| FF6EH to FF7FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF80H | CSIM10 | <CSIE10> | TRMD10 | 0 | DIR10 | 0 | 0 | 0 | CSOT10 | R/W | √ | √ | – | 00H | 566 |
| FF81H | CSIC10 | 0 | 0 | 0 | CKP10 | DAP10 | CKS102 | CKS101 | CKS100 | R/W | √ | √ | – | 00H | 569 |
| FF82H FF83H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF84H | SOTB10 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 565 |

**Note** These registers are incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-9. Special Function Register List: 78K0/KB2-L (4/5)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF85H to FF8BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF8CH | TCL51 | 0 | 0 | 0 | 0 | 0 | TCL512 | TCL511 | TCL510 | R/W | √ | √ | – | 00H | 318 |
| FF8DH to FF98H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF99H | WDTE | – | – | – | – | – | – | – | – | R/W | – | √ | – | 1AH/ 9AH[Note1] | 365 |
| FF9AH to FF9EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF9FH | OSCCTL | <EXCLK> | <OSCSEL> | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 202 |
| FFA0H | RCM | <RSTS> | 0 | 0 | 0 | 0 | 0 | <LSRSTOP> | <RSTOP> | R/W | √ | √ | – | 80H[Note2] | 207 |
| FFA1H | MCM | 0 | 0 | 0 | 0 | 0 | <XSEL> | <MCS> | <MCM0> | R/W | √ | √ | – | 00H | 209 |
| FFA2H | MOC | <MSTOP> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 80H | 208 |
| FFA3H | OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | R | √ | √ | – | 00H | 210, 640 |
| FFA4H | OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 | R/W | – | √ | – | 05H | 211, 641 |
| FFA5H | IICA | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 490 |
| FFA6H | SVA0 | – | – | – | – | – | – | – | 0 | R/W | – | √ | – | 00H | 490 |
| FFA7H | IICACTL0 | <IICE0> | <LREL0> | <WREL0> | <SPIE0> | <WTIM0> | <ACKE0> | <STT0> | <SPT0> | R/W | √ | √ | – | 00H | 492 |
| FFA8H | IICACTL1 | <WUP> | 0 | <CLD0> | <DAD0> | <SMC0> | <DFC0> | 0 | 0 | R/W | √ | √ | – | 00H | 501 |
| FFA9H | IICAF0 | <STCF> | <IICBSY> | 0 | 0 | 0 | 0 | <STCEN> | <IICRSV> | R/W | √ | √ | – | 00H | 499 |
| FFAAH | IICAS0 | <MSTS0> | <ALD0> | <EXC0> | <COI0> | <TRC0> | <ACKD0> | <STD0> | <SPD0> | R | √ | √ | – | 00H | 497 |
| FFABH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFACH | RESF | 0 | 0 | 0 | WDTRF | 0 | 0 | 0 | LVIRF | R | – | √ | – | 00H[Note3] | 664 |
| FFADH | IICWL | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAEH | IICWH | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |

**Notes 1.** The reset value of WDTE is determined by setting of option byte.

**2.** The value of this register is 00H immediately after a reset release but automatically changes to 80H after oscillation accuracy stabilization of high-speed internal oscillator has been waited.

**3.** The reset value of RESF varies depending on the reset source.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-9. Special Function Register List: 78K0/KB2-L (5/5)**

| Address | Symbol | | Bit No. 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | R/W | Number of Bits Manipulated Simultaneously 1 | 8 | 16 | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FFAFH to FFB9H | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFBAH | TMC00 | | 0 | 0 | 0 | 0 | TMC003 | TMC002 | TMC001 | <OVF00> | R/W | √ | √ | – | 00H | 248 |
| FFBBH | PRM00 | | ES110 | ES100 | ES010 | ES000 | 0 | 0 | PRM001 | PRM000 | R/W | √ | √ | – | 00H | 253 |
| FFBCH | CRC00 | | 0 | 0 | 0 | 0 | 0 | CRC002 | CRC001 | CRC000 | R/W | √ | √ | – | 00H | 249 |
| FFBDH | TOC00 | | 0 | <OSPT00> | <OSPE00> | TOC004 | <LVS00> | <LVR00> | TOC001 | <TOE00> | R/W | √ | √ | – | 00H | 251 |
| FFBEH | LVIM | | <LVION> | 0 | 0 | 0 | 0 | <LVISEL> | <LVIMD> | <LVIF> | R/W | √ | √ | – | 00H[Note1] | 672 |
| FFBFH | LVIS | | 0 | 0 | 0 | 0 | LVIS3 | LVIS2 | LVIS1 | LVIS0 | R/W | √ | √ | – | 00H[Note2] | 675 |
| FFC0H to FFDFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFE0H | IF0 | IF0L | <SREIF6> | <PIF5> | <PIF4> | <PIF3> | <PIF2> | <PIF1> | <PIF0> | <LVIIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE1H | | IF0H | <TMIF010> | <TMIF000> | <TMIF50> | <TMIFH0> | <TMIFH1> | <CSIIF10> | <STIF6> | <SRIF6> | R/W | √ | √ | | 00H | 598 |
| FFE2H | IF1 | IF1L | 0 | 0 | 0 | 0 | <TMIF51> | 0 | 0 | <ADIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE3H | | IF1H | 0 | 0 | 0 | <PIF11> | <PIF10> | 0 | 0 | <IICAIF0> | R/W | √ | √ | | 00H | 598 |
| FFE4H | MK0 | MK0L | <SREMK6> | <PMK5> | <PMK4> | <PMK3> | <PMK2> | <PMK1> | <PMK0> | <LVIMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE5H | | MK0H | <TMMK010> | <TMMK000> | <TMMK50> | <TMMKH0> | <TMMKH1> | <CSIMK10> | <STMK6> | <SRMK6> | R/W | √ | √ | | FFH | 606 |
| FFE6H | MK1 | MK1L | 1 | 1 | 1 | 1 | <TMMK51> | 1 | 1 | <ADMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE7H | | MK1H | 1 | 1 | 1 | <PMK11> | <PMK10> | 1 | 1 | <IICAMK0> | R/W | √ | √ | | FFH | 606 |
| FFE8H | PR0 | PR0L | <SREMK6> | <PPR5> | <PPR4> | <PPR3> | <PPR2> | <PPR1> | <PPR0> | <LVIPR> | R/W | √ | √ | √ | FFH | 613 |
| FFE9H | | PR0H | <TMPR010> | <TMPR000> | <TMPR50> | <TMPRH0> | <TMPRH1> | <CSIPR10> | <STPR6> | <SRPR6> | R/W | √ | √ | | FFH | 613 |
| FFEAH | PR1 | PR1L | 1 | 1 | 1 | 1 | <TMPR51> | 1 | 1 | <ADPR> | R/W | √ | √ | √ | FFH | 613 |
| FFEBH | | PR1H | 1 | 1 | 1 | <PPR11> | <PPR10> | 1 | 1 | <IICAPR0> | R/W | √ | √ | | FFH | 613 |
| FFECH to FFEFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFF0H | IMS | | RAM2 | RAM1 | RAM0 | 0 | ROM3 | ROM2 | ROM1 | ROM0 | R/W | – | √ | – | CFH[Note3] | 699 |
| FFF1H to FFFAH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFFBH | PCC | | 0 | 0 | 0 | 0 | 0 | PCC2 | PCC1 | PCC0 | R/W | √ | √ | – | 01H | 204 |

**Notes 1.** The reset values of LVIM vary depending on the reset source and setting of option byte.

**2.** The reset values of LVIS vary depending on the reset source.

**3.** Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated in Table 3-1 after release of reset.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-10. Special Function Register List: 78K0/KC2-L (1/6)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF00H | P0 | 0 | 0 | 0 | 0 | 0 | P02 Note1 | P01 | P00 | R/W | √ | √ | – | 00H | 172 |
| FF01H | P1 | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 | R/W | √ | √ | – | 00H | 172 |
| FF02H | P2 | P27 Note2 | P26 | P25 | P24 | P23 | P22 | P21 | P20 | R/W | √ | √ | – | 00H | 172 |
| FF03H | P3 | 0 | 0 | 0 | 0 | P33 | P32 | P31 | P30 | R/W | √ | √ | – | 00H | 172 |
| FF04H | P4 Note2 | 0 | 0 | 0 | 0 | 0 | P42 Note1 | P41 Note2 | P40 Note2 | R/W | √ | √ | – | 00H | 172 |
| FF05H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF06H | P6 | 0 | 0 | 0 | 0 | P63 Note2 | P62 | P61 | P60 | R/W | √ | √ | – | 00H | 172 |
| FF07H | P7 | 0 | 0 | P75 Note1 | P74 Note1 | P73 | P72 | P71 | P70 | R/W | √ | √ | – | 00H | 172 |
| FF08H | AD ADCRL | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF09H | CR | 0 | 0 | 0 | 0 | 0 | 0 | – | – | R | – | – | √ | 0000H | 410 |
| FF0AH | RXB6 | – | – | – | – | – | – | – | – | R | – | √ | – | FFH | 452 |
| FF0BH | TXB6 | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 453 |
| FF0CH | P12 | 0 | 0 | P125 | P124 | P123 | P122 | P121 | P120 | R Note3 | √ | √ | – | 00H | 172 |
| FF0DH | ADCRH | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 411 |
| FF0EH | ADS | 0 | <ADOAS> Note4 | 0 | 0 | <ADS3> | <ADS2> | <ADS1> | <ADS0> | R/W | √ | √ | – | 00H | 412, 439 |
| FF0FH | SIO10 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 566 |
| FF10H | TM00 | – | – | – | – | – | – | – | – | R | – | – | √ | 0000H | 243 |
| FF11H | | – | – | – | – | – | – | – | – | | | | | | |
| FF12H | CR000 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF13H | | – | – | – | – | – | – | – | – | | | | | | |
| FF14H | CR010 | – | – | – | – | – | – | – | – | R/W | – | – | √ | 0000H | 244 |
| FF15H | | – | – | – | – | – | – | – | – | | | | | | |
| FF16H | TM50 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |
| FF17H | CR50 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |
| FF18H | CMP00 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF19H | CMP10 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1AH | CMP01 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1BH | CMP11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 338 |
| FF1CH to FF1EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF1FH | TM51 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 317 |

**Notes 1.** 48-pin products only.

**2.** 44-pin and 48-pin products only.

**3.** Only P120 is R/W.

**4.** This bit is incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-10. Special Function Register List: 78K0/KC2-L (2/6)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF20H | PM0 | 1 | 1 | 1 | 1 | 1 | PM02 Note 1 | PM01 | PM00 | R/W | √ | √ | – | FFH | 167, 256 |
| FF21H | PM1 | PM17 | PM16 | PM15 | PM14 | PM13 | PM12 | PM11 | PM10 | R/W | √ | √ | – | FFH | 167, 324, 345, 415, 440, 463, 573 |
| FF22H | PM2 | PM27 Note2 | PM26 | PM25 | PM24 | PM23 | PM22 | PM21 | PM20 | R/W | √ | √ | – | FFH | 167, 415, 440 |
| FF23H | PM3 | 1 | 1 | 1 | 1 | PM33 | PM32 | PM31 | PM30 | R/W | √ | √ | – | FFH | 167, 324, 345 |
| FF24H | PM4 Note2 | 1 | 1 | 1 | 1 | 1 | PM42 Note 1 | PM41 Note2 | PM40 Note2 | R/W | √ | √ | – | FFH | 167, 385, 400, 573 |
| FF25H | PER0 | RTCEN | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 212, 373 |
| FF26H | PM6 | 1 | 1 | 1 | 1 | PM63 Note2 | PM62 | PM61 | PM60 | R/W | √ | √ | – | FFH | 167, 463, 504, 573 |
| FF27H | PM7 | 1 | 1 | PM75 Note 1 | PM74 Note 1 | PM73 | PM72 | PM71 | PM70 | R/W | √ | √ | – | FFH | 167 |
| FF28H | ADM0 | <ADCS> | 0 | FR2 | FR1 | FR0 | LV1 | LV0 | <ADCE> | R/W | √ | √ | – | 00H | 405 |
| FF29H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF2AH | POM6 | 0 | 0 | 0 | 0 | POM63 Note2 | POM62 | POM61 | POM60 | R/W | √ | √ | – | 00H | 180, 464, 504 |
| FF2BH | FPCTL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | <FLMD PUP> | R/W | √ | √ | – | 00H | 713 |
| FF2CH | PM12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM120 | R/W | √ | √ | – | FFH | 167, 573, 676 |
| FF2DH | RSTMASK | 0 | 0 | RSTM | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 180 |
| FF2EH | ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 | R/W | √ | √ | – | 00H | 181, 413, 437 |
| FF2FH | ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 | R/W | √ | √ | – | 07H | |
| FF30H | PU0 | 0 | 0 | 0 | 0 | 0 | PU02 Note 1 | PU01 | PU00 | R/W | √ | √ | – | 00H | 177 |
| FF31H | PU1 | PU17 | PU16 | PU15 | PU14 | PU13 | PU12 | PU11 | PU10 | R/W | √ | √ | – | 00H | 177 |
| FF32H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF33H | PU3 | 0 | 0 | 0 | 0 | PU33 | PU32 | PU31 | PU30 | R/W | √ | √ | – | 00H | 177 |
| FF34H | PU4 Note2 | 0 | 0 | 0 | 0 | 0 | PU42 Note 1 | PU41 Note2 | PU40 Note2 | R/W | √ | √ | – | 00H | 177 |
| FF35H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF36H | PU6 | 0 | 0 | 0 | 0 | PU63 Note2 | PU62 | PU61 | PU60 | R/W | √ | √ | – | 00H | 177 |
| FF37H | PU7 | 0 | 0 | PU75 Note 1 | PU74 Note 1 | PU73 | PU72 | PU71 | PU70 | R/W | √ | √ | – | 00H | 177 |
| FF38H to FF3BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF3CH | PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | PU120 | R/W | √ | √ | – | 20H | 177 |
| FF3DH | RMC | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 691 |
| FF3EH | PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 | R/W | √ | √ | – | 00H | 179, 503 |
| FF3FH | MUXSEL Note2 | 0 | 0 | 0 | 0 | 0 | CSISEL | 0 | 0 | R/W | √ | √ | – | 00H | 183, 572 |

**Notes 1.** 48-pin products only.

**2.** 44-pin and 48-pin products only.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-10. Special Function Register List: 78K0/KC2-L (3/6)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF40H | CKS Note1 | 0 | 0 | 0 | <CLOE> | CCS3 | CCS2 | CCS1 | CCS0 | R/W | √ | √ | – | 00H | 399 |
| FF41H | CR51 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 317 |
| FF42H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF43H | TMC51 | <TCE51> | TMC516 | 0 | 0 | <LVS51> | <LVR51> | TMC511 | <TOE51> | R/W | √ | √ | – | 00H | 320 |
| FF44H to FF47H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF48H | EGPCTL0 | EGP7 Note1 | EGP6 Note1 | EGP5 | EGP4 | EGP3 | EGP2 | EGP1 | EGP0 | R/W | √ | √ | – | 00H | 619 |
| FF49H | EGNCTL0 | EGN7 Note1 | EGN6 Note1 | EGN5 | EGN4 | EGN3 | EGN2 | EGN1 | EGN0 | R/W | √ | √ | – | 00H | 619 |
| FF4AH | EGPCTL1 | 0 | 0 | 0 | 0 | EGP11 | EGP10 | EGP9 | EGP8 Note2 | R/W | √ | √ | – | 00H | 619 |
| FF4BH | EGNCTL1 | 0 | 0 | 0 | 0 | EGN11 | EGN10 | EGN9 | EGN8 Note2 | R/W | √ | √ | – | 00H | 619 |
| FF4CH to FF4EH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF4FH | ISC | 0 | 0 | 0 | 0 | 0 | 0 | ISC1 | ISC0 | R/W | √ | √ | – | 00H | 463 |
| FF50H | ASIM6 | <POWER6> | <TXE6> | <RXE6> | PS61 | PS60 | CL6 | SL6 | ISRM6 | R/W | √ | √ | – | 01H | 454 |
| FF51H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF52H | | | | | | | | | | | | | | | |
| FF53H | ASIS6 | 0 | 0 | 0 | 0 | 0 | PE6 | FE6 | OVE6 | R | – | √ | – | 00H | 457 |
| FF54H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF55H | ASIF6 | 0 | 0 | 0 | 0 | 0 | 0 | TXBF6 | TXSF6 | R | – | √ | – | 00H | 458 |
| FF56H | CKSR6 | 0 | 0 | 0 | 0 | TPS63 | TPS62 | TPS61 | TPS60 | R/W | – | √ | – | 00H | 458 |
| FF57H | BRGC6 | MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 | R/W | – | √ | – | FFH | 460 |
| FF58H | ASICL6 | <SBRF6> | <SBRT6> | SBTT6 | SBL62 | SBL61 | SBL60 | DIR6 | TXDLV6 | R/W | √ | √ | – | 16H | 461 |
| FF59H to FF5FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF60H | AMP0M Note3 | <OPAMP0E> | <PGAEN> | 0 | 0 | 0 | 0 | AMP0VG1 | AMP0VG0 | R/W | √ | √ | – | 00H | 436 |
| FF61H | AMP1M Note3 | <OPAMP1E> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 00H | 436 |
| FF62H to FF68H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF69H | TMHMD0 | <TMHE0> | CKS02 | CKS01 | CKS00 | TMMD01 | TMMD00 | <TOLEV0> | <TOEN0> | R/W | √ | √ | – | 00H | 339 |
| FF6AH | TCL50 | 0 | 0 | 0 | 0 | 0 | TCL502 | TCL501 | TCL500 | R/W | √ | √ | – | 00H | 318 |

**Notes 1.** 48-pin products only.

**2.** 44-pin and 48-pin products only.

**3.** These registers are incorporated only in products with operational amplifier.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

**Table 3-10. Special Function Register List: 78K0/KC2-L (4/6)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF6BH | TMC50 | <TCE50> | TMC506 | 0 | 0 | <LVS50> | <LVR50> | TMC501 | <TOE50> | R/W | √ | √ | – | 00H | 320 |
| FF6CH | TMHMD1 | <TMHE1> | CKS12 | CKS11 | CKS10 | TMMD11 | TMMD10 | <TOLEV1> | <TOEN1> | R/W | √ | √ | – | 00H | 339 |
| FF6DH | TMCYC1 | 0 | 0 | 0 | 0 | 0 | RMC1 | NRZB1 | <NRZ1> | R/W | √ | √ | – | 00H | 343 |
| FF6EH | KRM | 0 | 0 | KRM5 Note1 | KRM4 Note1 | KRM3 | KRM2 | KRM1 | KRM0 | R/W | √ | √ | – | 00H | 638 |
| FF6FH | RTCC2 | <RINTE> | <RCLOE2> | <RCKDIV> | 0 | 0 | ICT2 | ICT1 | ICT0 | R/W | √ | √ | – | 00H | 377 |
| FF70H to FF79H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF7AH | SIO11 | – | – | – | – | – | – | – | – | R | – | √ | – | 00H | 566 |
| FF7BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF7CH | SOTB11 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 565 |
| FF7DH to FF7FH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF80H | CSIM10 | <CSIE10> | TRMD10 | 0 | DIR10 | 0 | 0 | 0 | CSOT10 | R/W | √ | √ | – | 00H | 566 |
| FF81H | CSIC10 | 0 | 0 | 0 | CKP10 | DAP10 | CKS102 | CKS101 | CKS100 | R/W | √ | √ | – | 00H | 569 |
| FF82H FF83H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF84H | SOTB10 | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 565 |
| FF85H to FF87H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF88H | CSIM11 | <CSIE11> | TRMD11 | SSE11 Note1 | DIR11 | 0 | 0 | 0 | CSOT11 | R/W | √ | √ | – | 00H | 566 |
| FF89H | CSIC11 | 0 | 0 | 0 | CKP11 | DAP11 | CKS112 | CKS111 | CKS110 | R/W | √ | √ | – | 00H | 569 |
| FF8AH to FF8BH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF8CH | TCL51 | 0 | 0 | 0 | 0 | 0 | TCL512 | TCL511 | TCL510 | R/W | √ | √ | – | 00H | 318 |
| FF8DH to FF98H | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FF99H | WDTE | – | – | – | – | – | – | – | – | R/W | – | √ | – | 1AH/9AH Note2 | 365 |
| FF9AH | ALARMWM | 0 | WM40 | WM20 | WM10 | WM8 | WM4 | WM2 | WM1 | R/W | – | √ | – | 00H | 384 |
| FF9BH | ALARMWH | 0 | 0 | WH20 | WH10 | WH8 | WH4 | WH2 | WH1 | R/W | – | √ | – | 12H | 384 |
| FF9CH | ALARMWW | 0 | WW6 | WW5 | WW4 | WW3 | WW2 | WW1 | WW0 | R/W | – | √ | – | 00H | 384 |

**Notes 1.** 48-pin products only.

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

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-10. Special Function Register List: 78K0/KC2-L (5/6)**

| Address | Symbol | Bit No. | | | | | | | | R/W | Number of Bits Manipulated Simultaneously | | | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | 1 | 8 | 16 | | |
| FF9DH | RTCC0 | <RTCE> | 0 | <RCLOE1> | <RCLOE0> | AMPM | CT2 | CT1 | CT0 | R/W | √ | √ | – | 00H | 373 |
| FF9EH | RTCC1 | <WALE> | <WALIE> | 0 | <WAFG> | <RIFG> | 0 | <RWST> | <RWAIT> | R/W | √ | √ | – | 00H | 375 |
| FF9FH | OSCCTL | <EXCLK> | <OSCSEL> | <EXCLKS> | <OSCSELS> | 0 | <RSWOSC> | <AMPHXT> | 0 | R/W | √ | √ | – | 00H | 202 |
| FFA0H | RCM | <RSTS> | 0 | 0 | 0 | 0 | 0 | <LSRSTOP> | <RSTOP> | R/W | √ | √ | – | 80H Note1 | 207 |
| FFA1H | MCM | 0 | 0 | 0 | 0 | 0 | <XSEL> | <MCS> | <MCM0> | R/W | √ | √ | – | 00H | 209 |
| FFA2H | MOC | <MSTOP> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R/W | √ | √ | – | 80H | 208 |
| FFA3H | OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | R | √ | √ | – | 00H | 210, 640 |
| FFA4H | OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 | R/W | – | √ | – | 05H | 211, 641 |
| FFA5H | IICA | – | – | – | – | – | – | – | – | R/W | – | √ | – | 00H | 490 |
| FFA6H | SVA0 | – | – | – | – | – | – | – | 0 | R/W | – | √ | – | 00H | 490 |
| FFA7H | IICACTL0 | <IICE0> | <LREL0> | <WREL0> | <SPIE0> | <WTIM0> | <ACKE0> | <STT0> | <SPT0> | R/W | √ | √ | – | 00H | 492 |
| FFA8H | IICACTL1 | <WUP> | 0 | <CLD0> | <DAD0> | <SMC0> | <DFC0> | 0 | 0 | R/W | √ | √ | – | 00H | 501 |
| FFA9H | IICAF0 | <STCF> | <IICBSY> | 0 | 0 | 0 | 0 | <STCEN> | <IICRSV> | R/W | √ | √ | – | 00H | 499 |
| FFAAH | IICAS0 | <MSTS0> | <ALD0> | <EXC0> | <COI0> | <TRC0> | <ACKD0> | <STD0> | <SPD0> | R | √ | √ | – | 00H | 497 |
| FFABH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFACH | RESF | 0 | 0 | 0 | WDTRF | 0 | 0 | 0 | LVIRF | R | – | √ | – | 00H Note2 | 664 |
| FFADH | IICWL | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAEH | IICWH | – | – | – | – | – | – | – | – | R/W | – | √ | – | FFH | 503 |
| FFAFH | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFB0H | RSUBC | – | – | – | – | – | – | – | – | R | – | – | √ | 0000H | 378 |
| FFB1H | | – | – | – | – | – | – | – | – | | | | | | |
| FFB2H | SEC | 0 | SEC40 | SEC20 | SEC10 | SEC8 | SEC4 | SEC2 | SEC1 | R/W | – | √ | – | 00H | 378 |
| FFB3H | MIN | 0 | MIN40 | MIN20 | MIN10 | MIN8 | MIN4 | MIN2 | MIN1 | R/W | – | √ | – | 00H | 379 |
| FFB4H | HOUR | 0 | 0 | HOUR20 | HOUR10 | HOUR8 | HOUR4 | HOUR2 | HOUR1 | R/W | – | √ | – | 12H | 379 |
| FFB5H | WEEK | 0 | 0 | 0 | 0 | 0 | WEEK4 | WEEK2 | WEEK1 | R/W | – | √ | – | 00H | 381 |
| FFB6H | DAY | 0 | 0 | DAY20 | DAY10 | DAY8 | DAY4 | DAY2 | DAY1 | R/W | – | √ | – | 01H | 380 |
| FFB7H | MONTH | 0 | 0 | 0 | MONTH10 | MONTH8 | MONTH4 | MONTH2 | MONTH1 | R/W | – | √ | – | 01H | 382 |
| FFB8H | YEAR | YEAR80 | YEAR40 | YEAR20 | YEAR10 | YEAR8 | YEAR4 | YEAR2 | YEAR1 | R/W | – | √ | – | 00H | 382 |
| FFB9H | SUBCUD | DEV | F6 | F5 | F4 | F3 | F2 | F1 | F0 | R/W | – | √ | – | 00H | 383 |

**Notes 1.** The value of this register is 00H immediately after a reset release but automatically changes to 80H after oscillation accuracy stabilization of high-speed internal oscillator has been waited.

**2.** The reset value of RESF varies depending on the reset source.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

<R> **Table 3-10. Special Function Register List: 78K0/KC2-L (6/6)**

| Address | Symbol | | Bit No. 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | R/W | Number of Bits Manipulated Simultaneously 1 | 8 | 16 | After Reset | Reference page |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FFBAH | TMC00 | | 0 | 0 | 0 | 0 | TMC003 | TMC002 | TMC001 | <OVF00> | R/W | √ | √ | – | 00H | 248 |
| FFBBH | PRM00 | | ES110 | ES100 | ES010 | ES000 | 0 | 0 | PRM001 | PRM000 | R/W | √ | √ | – | 00H | 253 |
| FFBCH | CRC00 | | 0 | 0 | 0 | 0 | 0 | CRC002 | CRC001 | CRC000 | R/W | √ | √ | – | 00H | 249 |
| FFBDH | TOC00 | | 0 | <OSPT00> | <OSPE00> | TOC004 | <LVS00> | <LVR00> | TOC001 | <TOE00> | R/W | √ | √ | – | 00H | 251 |
| FFBEH | LVIM | | <LVION> | 0 | 0 | 0 | 0 | <LVISEL> | <LVIMD> | <LVIF> | R/W | √ | √ | – | 00H Note1 | 672 |
| FFBFH | LVIS | | 0 | 0 | 0 | 0 | LVIS3 | LVIS2 | LVIS1 | LVIS0 | R/W | √ | √ | – | 00H Note2 | 675 |
| FFC0H to FFDFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFE0H | IF0 | IF0L | <SREIF6> | <PIF5> | <PIF4> | <PIF3> | <PIF2> | <PIF1> | <PIF0> | <LVIIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE1H | | IF0H | <TMIF010> | <TMIF000> | <TMIF50> | <TMIFH0> | <TMIFH1> | <CSIIF10> | <STIF6> | <SRIF6> | R/W | √ | √ | | 00H | 598 |
| FFE2H | IF1 | IF1L | <PIF8> Note4 | <PIF7> Note3 | <RTCIF> | <KRIF> | <TMIF51> | <RTCIIF> | <PIF6> Note3 | <ADIF> | R/W | √ | √ | √ | 00H | 598 |
| FFE3H | | IF1H | 0 | 0 | 0 | <PIF11> | <PIF10> | <PIF9> | <CSIIF11> | <IICAIF0> | R/W | √ | √ | | 00H | 598 |
| FFE4H | MK0 | MK0L | <SREMK6> | <PMK5> | <PMK4> | <PMK3> | <PMK2> | <PMK1> | <PMK0> | <LVIMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE5H | | MK0H | <TMMK010> | <TMMK000> | <TMMK50> | <TMMKH0> | <TMMKH1> | <CSIMK10 | <STMK6> | <SRMK6> | R/W | √ | √ | | FFH | 606 |
| FFE6H | MK1 | MK1L | <PMK8> Note4 | <PMK7> Note3 | <RTCMK> | <KRMK> | <TMMK51> | <RTCIMK> | <PMK6> Note3 | <ADMK> | R/W | √ | √ | √ | FFH | 606 |
| FFE7H | | MK1H | 1 | 1 | 1 | <PMK11> | <PMK10> | <PMK9> | <CSIMK11> | <IICAMK0> | R/W | √ | √ | | FFH | 606 |
| FFE8H | PR0 | PR0L | <SREPR6> | <PPR5> | <PPR4> | <PPR3> | <PPR2> | <PPR1> | <PPR0> | <LVIPR> | R/W | √ | √ | √ | FFH | 613 |
| FFE9H | | PR0H | <TMPR010> | <TMPR000> | <TMPR50> | <TMPRH0> | <TMPRH1> | <CSIPR10> | <STPR6> | <SRPR6> | R/W | √ | √ | | FFH | 613 |
| FFEAH | PR1 | PR1L | <PPR8> Note4 | <PPR7> Note3 | <RTCPR> | <KRPR> | <TMPR51> | <RTCIPR> | <PPR6> Note3 | <ADPR> | R/W | √ | √ | √ | FFH | 613 |
| FFEBH | | PR1H | 1 | 1 | 1 | <PPR11> | <PPR10> | <PPR9> | <CSIPR11> | <IICAPR0> | R/W | √ | √ | | FFH | 613 |
| FFECH to FFEFH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFF0H | IMS | | RAM2 | RAM1 | RAM0 | 0 | ROM3 | ROM2 | ROM1 | ROM0 | R/W | – | √ | – | CFH Note5 | 699 |
| FFF1H to FFFAH | – | | – | – | – | – | – | – | – | – | – | – | – | – | – | – |
| FFFBH | PCC | | 0 | XTSTART | <CLS> | <CSS> | 0 | PCC2 | PCC1 | PCC0 | R/W | √ | √ | – | 01H | 204 |

**Notes 1.** The reset values of LVIM vary depending on the reset source and setting of option byte.

**2.** The reset values of LVIS vary depending on the reset source.

**3.** 48-pin products only.

**4.** 44-pin and 48-pin products only.

**5.** Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated in Table 3-1 after release of reset.

**Remark** For a bit name enclosed in angle brackets (<>), the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.

## 3.3 Instruction Address Addressing

An instruction address is determined by contents of the program counter (PC) and is normally incremented (+1 for each byte) automatically according to the number of bytes of an instruction to be fetched each time another instruction is executed. When a branch instruction is executed, the branch destination information is set to PC and branched by the following addressing (for details of instructions, refer to the **78K/0 Series Instructions User's Manual (U12326E)**).

### 3.3.1 Relative addressing

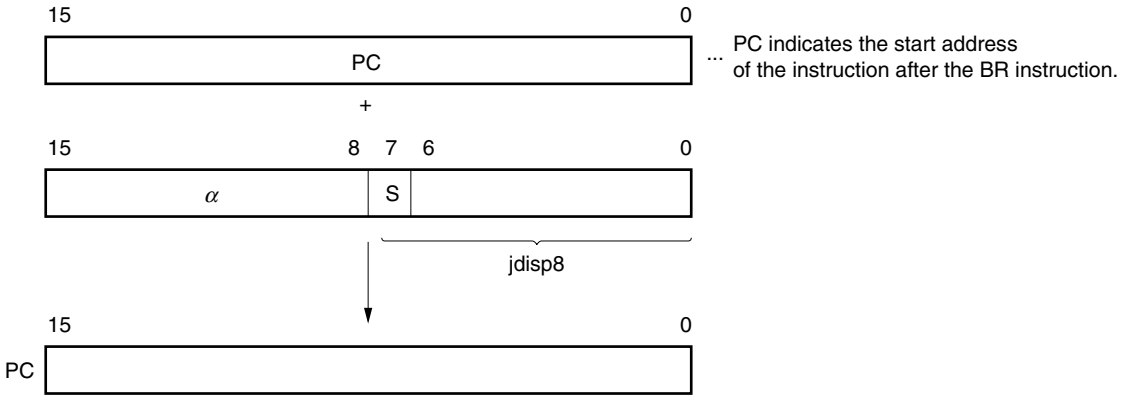
**[Function]**

The value obtained by adding 8-bit immediate data (displacement value: jdisp8) of an instruction code to the start address of the following instruction is transferred to the program counter (PC) and branched. The displacement value is treated as signed two's complement data (−128 to +127) and bit 7 becomes a sign bit.

In other words, relative addressing consists of relative branching from the start address of the following instruction to the −128 to +127 range.

This function is carried out when the BR $addr16 instruction or a conditional branch instruction is executed.

**[Illustration]**




When S = 0, all bits of $\alpha$ are 0.
When S = 1, all bits of $\alpha$ are 1.

### 3.3.2 Immediate addressing

**[Function]**

Immediate data in the instruction word is transferred to the program counter (PC) and branched.

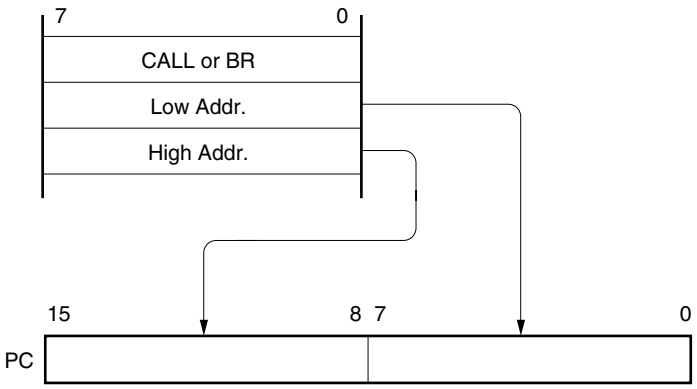
This function is carried out when the CALL !addr16 or BR !addr16 or CALLF !addr11 instruction is executed.

CALL !addr16 and BR !addr16 instructions can be branched to the entire memory space.

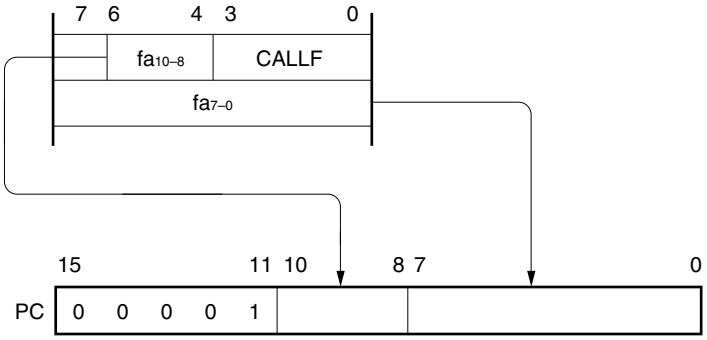
The CALLF !addr11 instruction is branched to the 0800H to 0FFFH area.

**[Illustration]**

In the case of CALL !addr16 and BR !addr16 instructions




In the case of CALLF !addr11 instruction



### 3.3.3 Table indirect addressing

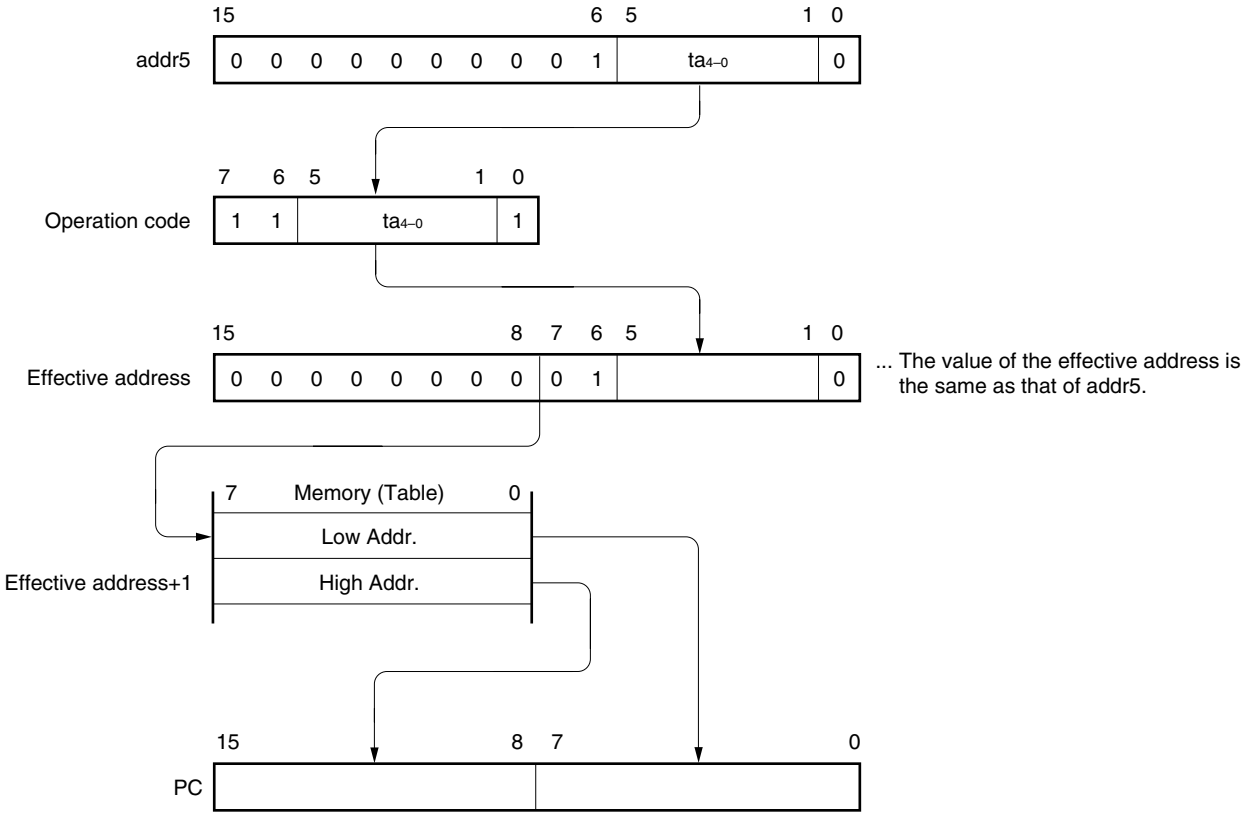
**[Function]**

Table contents (branch destination address) of the particular location to be addressed by bits 1 to 5 of the immediate data of an operation code are transferred to the program counter (PC) and branched.

This function is carried out when the CALLT [addr5] instruction is executed.

This instruction references the address that is indicated by addr5 and is stored in the memory table from 0040H to 007FH, and allows branching to the entire memory space.

**[Illustration]**



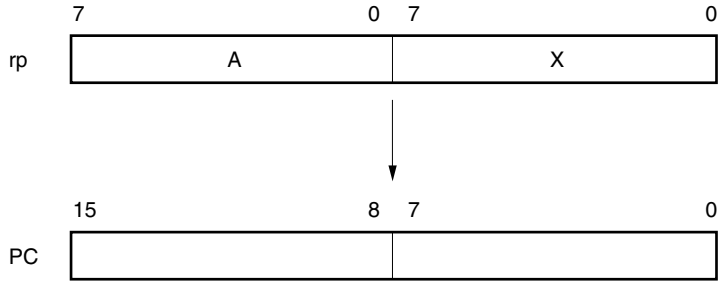
### 3.3.4 Register addressing

**[Function]**

Register pair (AX) contents to be specified with an instruction word are transferred to the program counter (PC) and branched.

This function is carried out when the BR AX instruction is executed.

**[Illustration]**




## 3.4 Operand Address Addressing

The following methods are available to specify the register and memory (addressing) to undergo manipulation during instruction execution.

### 3.4.1 Implied addressing

**[Function]**

The register that functions as an accumulator (A and AX) among the general-purpose registers is automatically (implicitly) addressed.

Of the 78K0/Kx2-L microcontroller instruction words, the following instructions employ implied addressing.

| Instruction | Register to Be Specified by Implied Addressing |
|---|---|
| MULU | A register for multiplicand and AX register for product storage |
| DIVUW | AX register for dividend and quotient storage |
| ADJBA/ADJBS | A register for storage of numeric values that become decimal correction targets |
| ROR4/ROL4 | A register for storage of digit data that undergoes digit rotation |

**[Operand format]**

Because implied addressing can be automatically determined with an instruction, no particular operand format is necessary.

**[Description example]**

In the case of MULU X

With an 8-bit × 8-bit multiply instruction, the product of the A register and X register is stored in AX. In this example, the A and AX registers are specified by implied addressing.

### 3.4.2 Register addressing

**[Function]**

The general-purpose register to be specified is accessed as an operand with the register bank select flags (RBS0 to RBS1) and the register specify codes of an operation code.

Register addressing is carried out when an instruction with the following operand format is executed. When an 8-bit register is specified, one of the eight registers is specified with 3 bits in the operation code.

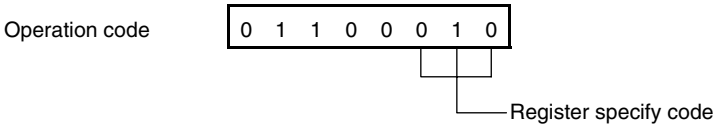
**[Operand format]**

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

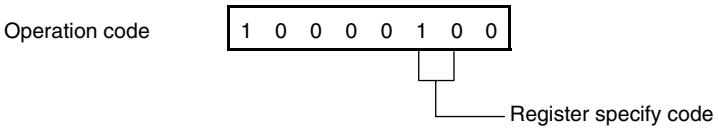
'r' and 'rp' can be described by absolute names (R0 to R7 and RP0 to RP3) as well as function names (X, A, C, B, E, D, L, H, AX, BC, DE, and HL).

**[Description example]**

MOV A, C; when selecting C register as r




INCW DE; when selecting DE register pair as rp



### 3.4.3 Direct addressing

**[Function]**

The memory to be manipulated is directly addressed with immediate data in an instruction word becoming an operand address.

This addressing can be carried out for all of the memory spaces.

**[Operand format]**

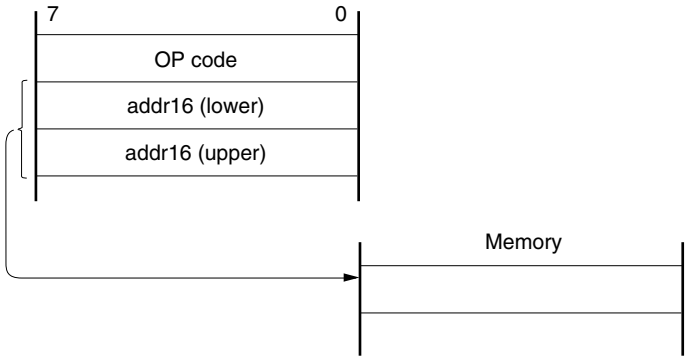
| Identifier | Description |
|---|---|
| addr16 | Label or 16-bit immediate data |

**[Description example]**

MOV A, !0FE00H; when setting !addr16 to FE00H

| | | |
|---|---|---|
| Operation code | 1 0 0 0 1 1 1 0 | OP code |
| | 0 0 0 0 0 0 0 0 | 00H |
| | 1 1 1 1 1 1 1 0 | FEH |

**[Illustration]**



### 3.4.4 Short direct addressing

**[Function]**

The memory to be manipulated in the fixed space is directly addressed with 8-bit data in an instruction word.

This addressing is applied to the 256-byte space FE20H to FF1FH. Internal high-speed RAM and special function registers (SFRs) are mapped at FE20H to FEFFH and FF00H to FF1FH, respectively.

The SFR area (FF00H to FF1FH) where short direct addressing is applied is a part of the overall SFR area. Ports that are frequently accessed in a program and compare and capture registers of the timer/event counter are mapped in this area, allowing SFRs to be manipulated with a small number of bytes and clocks.

When 8-bit immediate data is at 20H to FFH, bit 8 of an effective address is set to 0. When it is at 00H to 1FH, bit 8 is set to 1. Refer to the **[Illustration]** shown below.

**[Operand format]**

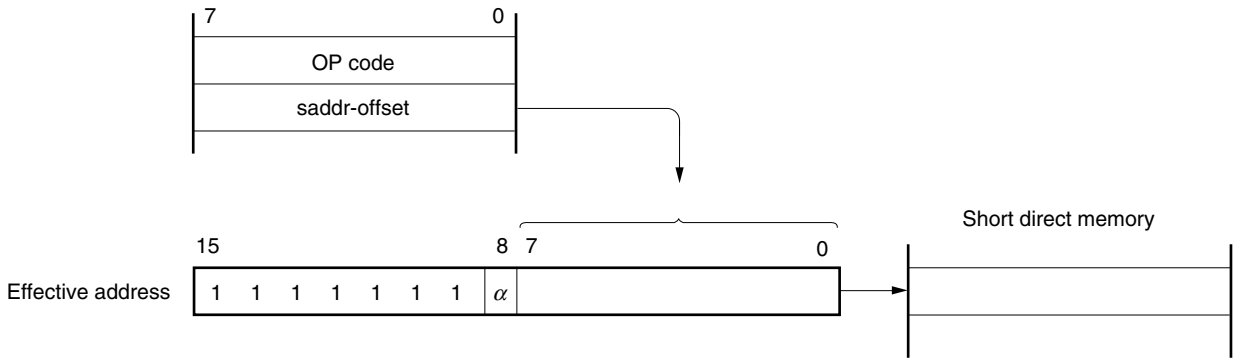
| Identifier | Description |
|---|---|
| saddr | Immediate data that indicate label or FE20H to FF1FH |
| saddrp | Immediate data that indicate label or FE20H to FF1FH (even address only) |

**[Description example]**

```
LB1 EQU 0FE30H   ; Defines FE30H by LB1.
      :
MOV LB1, A       ; When LB1 indicates FE30H of the saddr area and the value of register A is transferred to that
                   address
```

| | | |
|---|---|---|
| Operation code | 1 1 1 1 0 0 1 0 | OP code |
| | 0 0 1 1 0 0 0 0 | 30H (saddr-offset) |

**[Illustration]**




When 8-bit immediate data is 20H to FFH, $\alpha = 0$

When 8-bit immediate data is 00H to 1FH, $\alpha = 1$

### 3.4.5 Special function register (SFR) addressing

**[Function]**

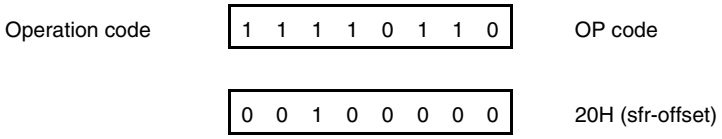
A memory-mapped special function register (SFR) is addressed with 8-bit immediate data in an instruction word. This addressing is applied to the 240-byte spaces FF00H to FFCFH and FFE0H to FFFFH. However, the SFRs mapped at FF00H to FF1FH can be accessed with short direct addressing.

**[Operand format]**

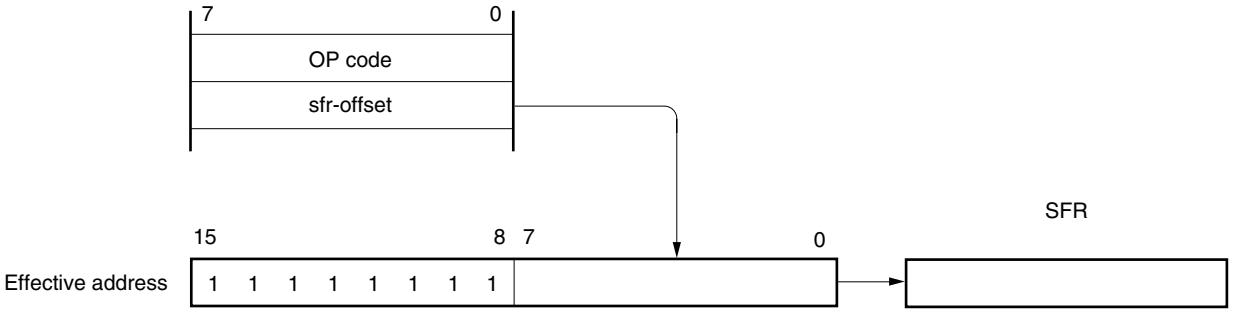
| Identifier | Description |
|---|---|
| sfr | Special function register name |
| sfrp | 16-bit manipulatable special function register name (even address only) |

**[Description example]**

MOV PM0, A; when selecting PM0 (FF20H) as sfr




**[Illustration]**



### 3.4.6 Register indirect addressing

**[Function]**

Register pair contents specified by a register pair specify code in an instruction word and by a register bank select flag (RBS0 and RBS1) serve as an operand address for addressing the memory.

This addressing can be carried out for all of the memory spaces.

**[Operand format]**

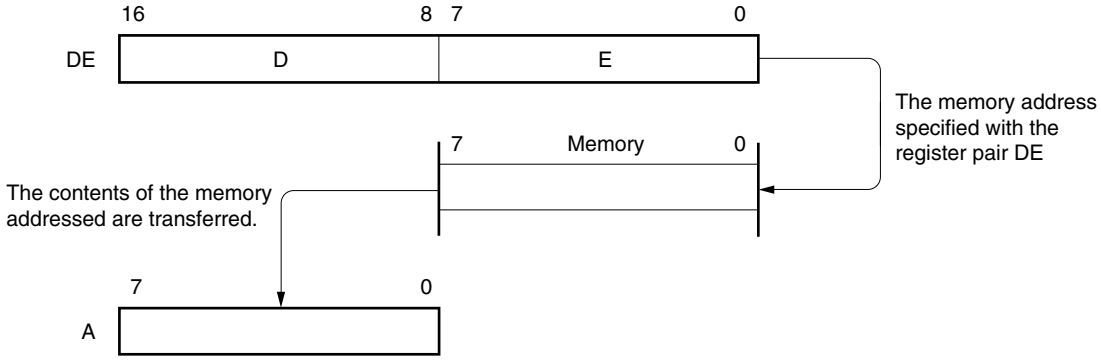
| Identifier | Description |
|---|---|
| – | [DE], [HL] |

**[Description example]**

MOV A, [DE]; when selecting [DE] as register pair

| Operation code | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|

**[Illustration]**



### 3.4.7 Based addressing

**[Function]**

8-bit immediate data is added as offset data to the contents of the base register, that is, the HL register pair in the register bank specified by the register bank select flag (RBS0 and RBS1), and the sum is used to address the memory. Addition is performed by expanding the offset data as a positive number to 16 bits. A carry from the 16th bit is ignored.

This addressing can be carried out for all of the memory spaces.

**[Operand format]**

| Identifier | Description |
|---|---|
| – | [HL + byte] |

**[Description example]**

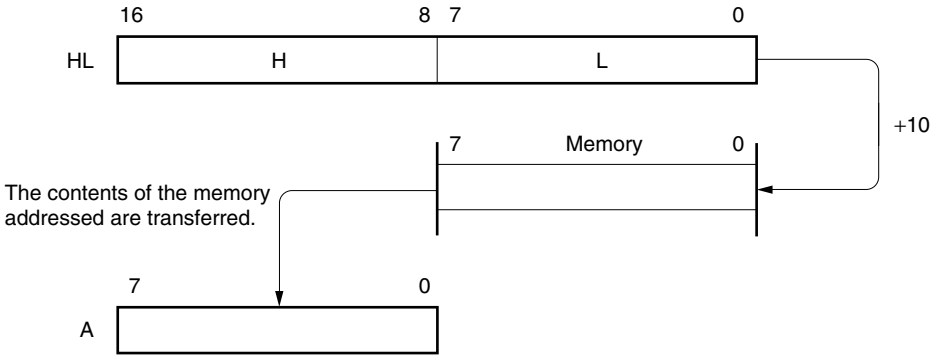
MOV A, [HL + 10H]; when setting byte to 10H

Operation code

| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|

**[Illustration]**



### 3.4.8 Based indexed addressing

**[Function]**

The B or C register contents specified in an instruction word are added to the contents of the base register, that is, the HL register pair in the register bank specified by the register bank select flag (RBS0 and RBS1), and the sum is used to address the memory. Addition is performed by expanding the B or C register contents as a positive number to 16 bits. A carry from the 16th bit is ignored.

This addressing can be carried out for all of the memory spaces.

**[Operand format]**

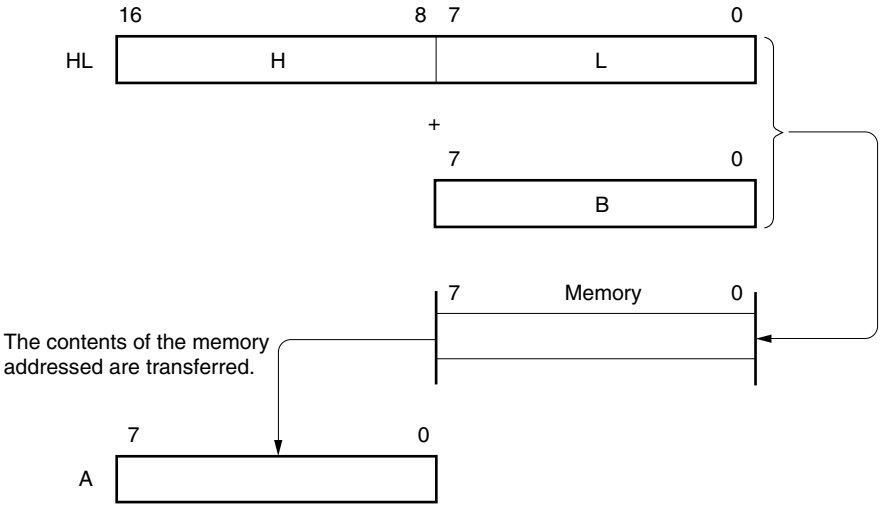
| Identifier | Description |
|---|---|
| – | [HL + B], [HL + C] |

**[Description example]**

MOV A, [HL +B]; when selecting B register

| Operation code | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|

**[Illustration]**



### 3.4.9 Stack addressing

**[Function]**

The stack area is indirectly addressed with the stack pointer (SP) contents.

This addressing method is automatically employed when the PUSH, POP, subroutine call and return instructions are executed or the register is saved/reset upon generation of an interrupt request.

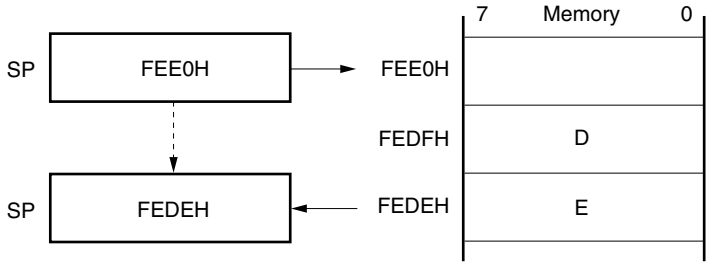
With stack addressing, only the internal high-speed RAM area can be accessed.

**[Description example]**

PUSH DE; when saving DE register

| Operation code | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|

**[Illustration]**



# CHAPTER 4 PORT FUNCTIONS

## 4.1 Port Functions

There are two types of pin I/O buffer power supplies: $AV_{REF}$ and $V_{DD}$. The relationship between these power supplies and the pins is shown below.

**Table 4-1. Pin I/O Buffer Power Supplies**

| Power Supply | Corresponding Pins |
|---|---|
| $AV_{REF}$ | P20 to P27[Note] |
| $V_{DD}$ | Pins other than P20 to P27[Note] |

**Note** 78K0/KY2-L: P20 to P23
78K0/KA2-L (20 pins): P20 to P25
<R> 78K0/KA2-L (25 pins): P20 to P26
<R> 78K0/KA2-L (32 pins): P20 to P27, P70 to P72
78K0/KB2-L: P20 to P23
<R> 78K0/KC2-L (40 pins): P20 to P26
78K0/KC2-L (44 pins, 48 pins): P20 to P27

78K0/Kx2-L microcontrollers are provided with digital I/O ports, which enable variety of control operations. The functions of each port are shown in Tables 4-2 to 4-6.

In addition to the function as digital I/O ports, these ports have several alternate functions. For details of the alternate functions, refer to **CHAPTER 2 PIN FUNCTIONS**.

**Table 4-2. Port Functions (78K0/KY2-L)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000/INTP0 |
| P01 | | | | TO00/TI010 |
| P20 | I/O | Port 2.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0-Note |
| P21 | | | | ANI1/AMP0OUTNote/ PGAINNote |
| P22 | | | | ANI2/AMP0+Note |
| P23 | | | | ANI3 |
| P30 | I/O | Port 3.<br>1-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TOH1/TI51/INTP1 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/TxD6 |
| P61 | | | | SDAA0/RxD6 |
| P121 | Input | Port 12.<br>3-bit input-only port.<br>For only P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| P125 | | | Reset input | $\overline{\text{RESET}}$ |

**Note** $\mu$PD78F0555, 78F0556, and 78F0557 (products with operational amplifier) only

**Table 4-3. Port Functions (78K0/KA2-L (20-pin products))**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000/INTP0 |
| P01 | | | | TO00/TI010 |
| P20 | I/O | Port 2.<br>6-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0-Note |
| P21 | | | | ANI1/AMP0OUTNote/ PGAINNote |
| P22 | | | | ANI2/AMP0+Note |
| P23 | | | | ANI3 |
| P24 | | | | ANI4 |
| P25 | | | | ANI5 |
| P30 | I/O | Port 3.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TOH1/TI51/INTP1 |
| P31 | | | | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/TxD6 |
| P61 | | | | SDAA0/RxD6 |
| P121 | Input | Port 12.<br>3-bit input-only port.<br>For only P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| P125 | | | Reset input | $\overline{\text{RESET}}$ |

**Note** $\mu$PD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

<R> **Table 4-4. Port Functions (78K0/KA2-L (25-pin and 32-pin products))**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 Note 1 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000 Note 1/INTP0 Note 1 (/TOH1) Note 1 (/TI51) Note 1 |
| P01 Note 2 | | | | TO00 Note 2/TI010 Note 2 |
| P02 | | | | $\overline{SSI11}$/INTP5 |
| P20 | I/O | Port 2.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0- Note 3 |
| P21 | | | | ANI1/AMP0OUT Note 3/ PGAIN Note 3 |
| P22 | | | | ANI2/AMP0+ Note 3 |
| P23 | | | | ANI3 |
| P24 | | | | ANI4 |
| P25 | | | | ANI5 |
| P26 | | | | ANI6 |
| P27 Note 2 | | | | ANI7 Note 2 |
| P31 | I/O | Port 3.<br>7-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P33 | | | | – |
| P34 | | | | INTP4(/TOH1) (/TI51) Note 1 |
| P35 | | | | $\overline{SCK11}$ |
| P36 | | | | SI11 |
| P37 | | | | SO11 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TxD6/SCLA0 |
| P61 | | | | RxD6/SDAA0 |
| P70 Note 2 | I/O | Port 7.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI8 Note 2 |
| P71 Note 2 | | | | ANI9 Note 2 |
| P72 Note 2 | | | | ANI10 Note 2 |
| P121 | Input | Port 12.<br>3-bit I/O port.<br>For only P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | X1/TOOLC0 (/TI000)(/INTP0) |
| P122 | | | | X2/EXCLK/ TOOLD0 |
| P125 | | | Reset input | $\overline{RESET}$(/TI000) Note 2 (/INTP0) Note 2 |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** $\mu$PD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

**Table 4-5. Port Functions (78K0/KB2-L)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000 |
| P01 | | | | TI010/TO00 |
| P10 | I/O | Port 1.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | ANI8/AMP1-Note/$\overline{SCK10}$ |
| P11 | | | | ANI9/AMP1OUTNote/SI10 |
| P12 | | | | ANI10/AMP1+Note/ SO10 |
| P13 | | | | TxD6 |
| P14 | | | | RxD6 |
| P15 | | | | TOH0 |
| P16 | | | | TOH1/INTP5 |
| P17 | | | | TI50/TO50 |
| P20 | I/O | Port 2.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0-Note |
| P21 | | | | ANI1/AMP0OUTNote/ PGAINNote |
| P22 | | | | ANI2/AMP0+Note |
| P23 | | | | ANI3 |
| P30 | I/O | Port 3.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | INTP1 |
| P31 | | | | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P33 | | | | TI51/TO51/INTP4 |
| P60 | I/O | Port 6.<br>2-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input can be set to SMBus input buffer in 1-bit units.<br>Output can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/INTP11 |
| P61 | | | | SDAA0/INTP10 |
| P120 | I/O | Port 12.<br>1-bit I/O port and 3-bit input port.<br>For only P120 and P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | EXLVI/INTP0 |
| P121 | Input | | | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| P125 | | | Reset input | $\overline{RESET}$ |

**Note** $\mu$PD78F0576, 78F0577, and 78F0578 (products with operational amplifier) only

**Table 4-6. Port Functions (78K0/KC2-L) (1/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P00 | I/O | Port 0.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | TI000 |
| P01 | | | | TI010/TO00 |
| P02 Note 1 | | | | INTP7 Note 1 |
| P10 | I/O | Port 1.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | ANI8/AMP1- Note 2 /$\overline{\text{SCK10}}$ |
| P11 | | | | ANI9/AMP1OUT Note 2 /SI10 |
| P12 | | | | ANI10/AMP1+ Note 2 /SO10 |
| P13 | | | | TxD6 |
| P14 | | | | RxD6 |
| P15 | | | | TOH0 |
| P16 | | | | TOH1/INTP5 |
| P17 | | | | TI50/TO50 |
| P20 | I/O | Port 2.<br>8-bit I/O port.<br>Input/output can be specified in 1-bit units. | Analog input | ANI0/AMP0- Note 2 |
| P21 | | | | ANI1/AMP0OUT Note 2 /PGAIN Note 2 |
| P22 | | | | ANI2/AMP0+ Note 2 |
| P23 | | | | ANI3 |
| P24 | | | | ANI4 |
| P25 | | | | ANI5 |
| P26 | | | | ANI6 |
| P27 Note 3 | | | | ANI7 Note 3 |
| P30 | I/O | Port 3.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | INTP1 |
| P31 | | | | INTP2/TOOLC1 |
| P32 | | | | INTP3/TOOLD1 |
| P33 | | | | TI51/TO51/INTP4 |
| P40 Note 3 | I/O | Port 4.<br>3-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | RTCCL Note 3 /RTCDIV Note 3 (/$\overline{\text{SCK11}}$) Note 3 |
| P41 Note 3 | | | | RTC1HZ Note 3 (/SI11) Note 3 |
| P42 Note 1 | | | | PCL Note 1 /$\overline{\text{SSI11}}$ Note 1 /INTP6 Note 1 |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, and 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

**Table 4-6. Port Functions (78K0/KC2-L) (2/2)**

| Function Name | I/O | Function | After Reset | Alternate Function |
|---|---|---|---|---|
| P60 | I/O | Port 6.<br>4-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Input of P60 and P61 can be set to SMBus input buffer in 1-bit units.<br>Output of P60 to P63 can be set to N-ch open-drain output ($V_{DD}$ tolerance).<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | SCLA0/$\overline{\text{SCK11}}$/ INTP11 |
| P61 | | | | SDAA0/SI11/INTP10 |
| P62 | | | | SO11/INTP9 |
| P63[Note 2] | | | | INTP8[Note 2] |
| P70 | I/O | Port 7.<br>6-bit I/O port.<br>Input/output can be specified in 1-bit units.<br>Use of an on-chip pull-up resistor can be specified by a software setting. | Input port | KR0 |
| P71 | | | | KR1 |
| P72 | | | | KR2 |
| P73 | | | | KR3 |
| P74[Note 1] | | | | KR4[Note 1] |
| P75[Note 1] | | | | KR5[Note 1] |
| P120 | I/O | Port 12.<br>1-bit I/O port and 5-bit input port.<br>For only P120 and P125, use of an on-chip pull-up resistor can be specified by a software setting. | Input port | EXLVI/INTP0 (/SO11)[Note 2] |
| P121 | Input | | | X1/TOOLC0 |
| P122 | | | | X2/EXCLK/TOOLD0 |
| P123 | | | | XT1 |
| P124 | | | | XT2/EXCLKS |
| P125 | | | Reset input | $\overline{\text{RESET}}$ |

**Notes 1.** 48-pin products only

**2.** 44-pin and 48-pin products only

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

## 4.2 Port Configuration

Ports include the following hardware.

<R>

**Table 4-7. Port Configuration**

| Item | Configuration |
|---|---|
| Control registers | Port mode registers (PMxx): PM0, PM1[Note 1], PM2, PM3, PM4[Note 2], PM6, PM7[Note 3], PM12[Note 1]<br>Port registers (Pxx): P0, P1[Note 1], P2, P3, P4[Note 2], P6, P7[Note 3], P12<br>Pull-up resistor option registers (PUxx): PU0, PU1[Note 1], PU3, PU4[Note 2], PU6, PU7[Note 4], PU12<br>Port input mode register 6 (PIM6)<br>Port output mode register 6 (POM6)<br>Reset pin mode register (RSTMASK)<br>A/D port configuration register 0 (ADPC0)<br>A/D port configuration register 1 (ADPC1)[Note 1]<br>Port alternate switch control register (MUXSEL)[Note 5] |
| Port | • 78K0/KY2-L: Total: 12 (CMOS I/O: 9, CMOS input: 3)<br>• 20-pin products of 78K0/KA2-L: Total: 16 (CMOS I/O: 13, CMOS input: 3)<br>• 25-pin products of 78K0/KA2-L: Total: 21 (CMOS I/O: 18, CMOS input: 3)<br>• 32-pin products of 78K0/KA2-L: Total: 25 (CMOS I/O: 22, CMOS input: 3)<br>• 78K0/KB2-L: Total: 24 (CMOS I/O: 21, CMOS input: 3)<br>• 40-pin products of 78K0/KC2-L: Total: 34 (CMOS I/O: 29, CMOS input: 5)<br>• 44-pin products of 78K0/KC2-L: Total: 38 (CMOS I/O: 33, CMOS input: 5)<br>• 48-pin products of 78K0/KC2-L: Total: 42 (CMOS I/O: 37, CMOS input: 5) |
| Pull-up resistor | • 78K0/KY2-L: Total: 6<br>• 20-pin products of 78K0/KA2-L: Total: 8<br>• 25-pin products of 78K0/KA2-L: Total: 12<br>• 32-pin products of 78K0/KA2-L: Total: 12<br>• 78K0/KB2-L: Total: 18<br>• 40-pin products of 78K0/KC2-L: Total: 22<br>• 44-pin products of 78K0/KC2-L: Total: 26<br>• 48-pin products of 78K0/KC2-L: Total: 30 |

**Notes 1.** 78K0/KB2-L and 78K0/KC2-L only

**2.** 78K0/KC2-L (44-pin and 48-pin products) only

**3.** 78K0/KA2-L (32-pin products) and 78K0/KC2-L only

**4.** 78K0/KC2-L only

**5.** 78K0/KA2-L (25-pin and 32-pin products) and 78K0/KC2-L (44-pin and 48-pin products) only

### 4.2.1 Port 0

<R>

| 78K0/KY2-L (μPD78F057x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| P00/TI000/ INTP0 | P00/TI000/ INTP0 | P00/TI000/ INTP0(/TOH1) (/TI51) | – | P00/TI000 | P00/TI000 | P00/TI000 | P00/TI000 |
| P01/TO00/ TI010 | P01/TO00/ TI010 | – | P01/TO00/ TI010 | P01/TO00/ TI010 | P01/TO00/ TI010 | P01/TO00 /TI010 | P01/TO00/ TI010 |
| – | – | P02/$\overline{\text{SSI11}}$/ INTP5 | P02/$\overline{\text{SSI11}}$/ INTP5 | – | – | – | P02/INTP7 |

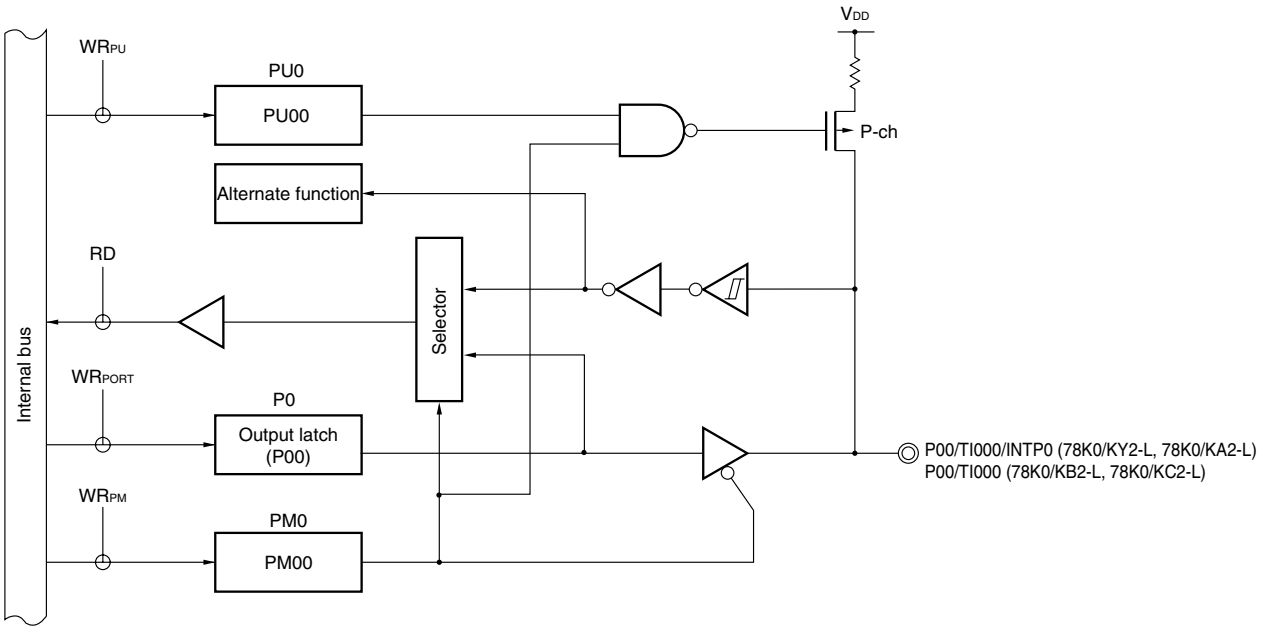
Port 0 is an I/O port with an output latch. Port 0 can be set to the input mode or output mode in 1-bit units using port mode register 0 (PM0). When the P00 to P02 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 0 (PU0).

This port can also be used for timer I/O, external interrupt request input, and chip select input of serial interface. The timer I/O can be assigned to P00 of the 78K0/KA2-L (25-pin products) by setting the port alternate switch control register (MUXSEL).

Reset signal generation sets port 0 to input mode.

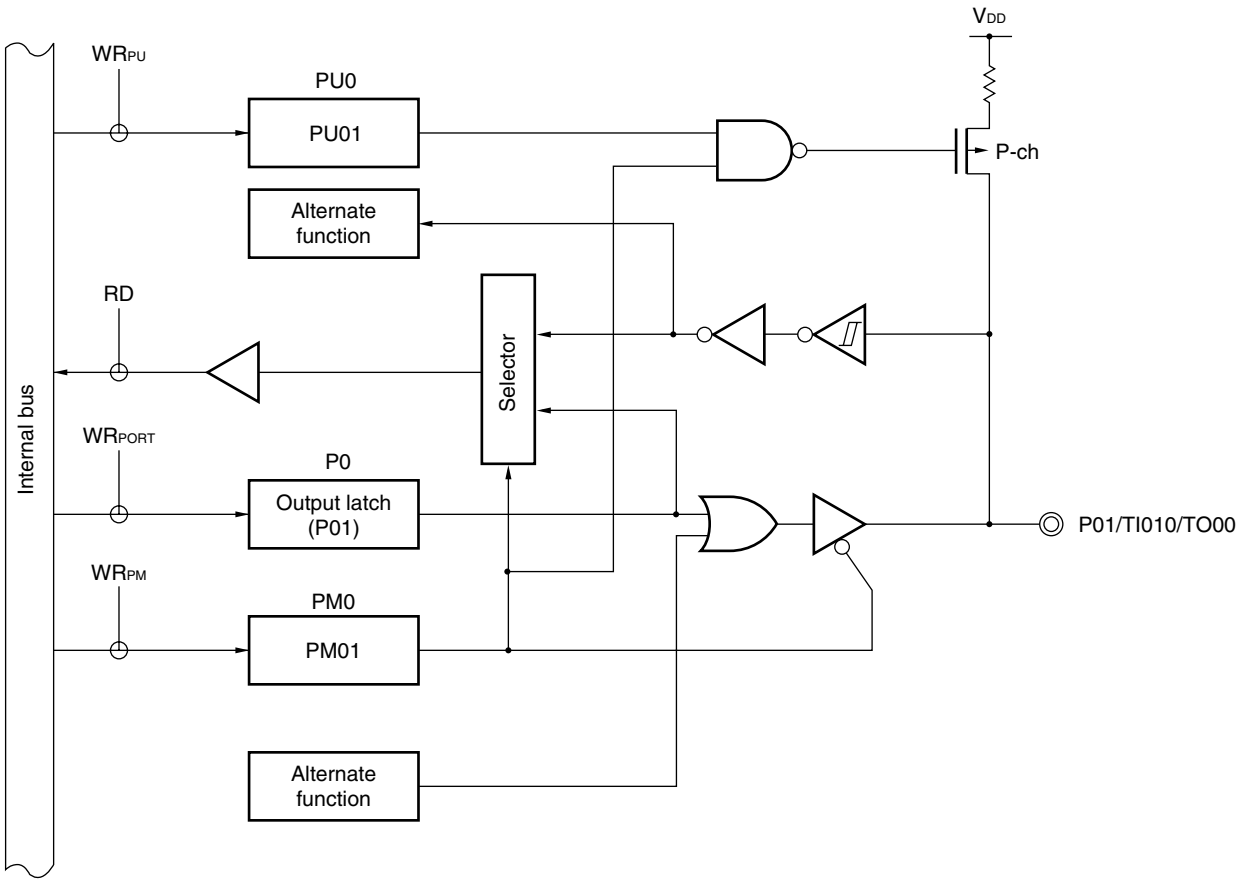
Figures 4-1 to 4-3 show block diagrams of port 0.

**Figure 4-1. Block Diagram of P00**




P0: Port register 0
PU0: Pull-up resistor option register 0
PM0: Port mode register 0
RD: Read signal
WR××: Write signal

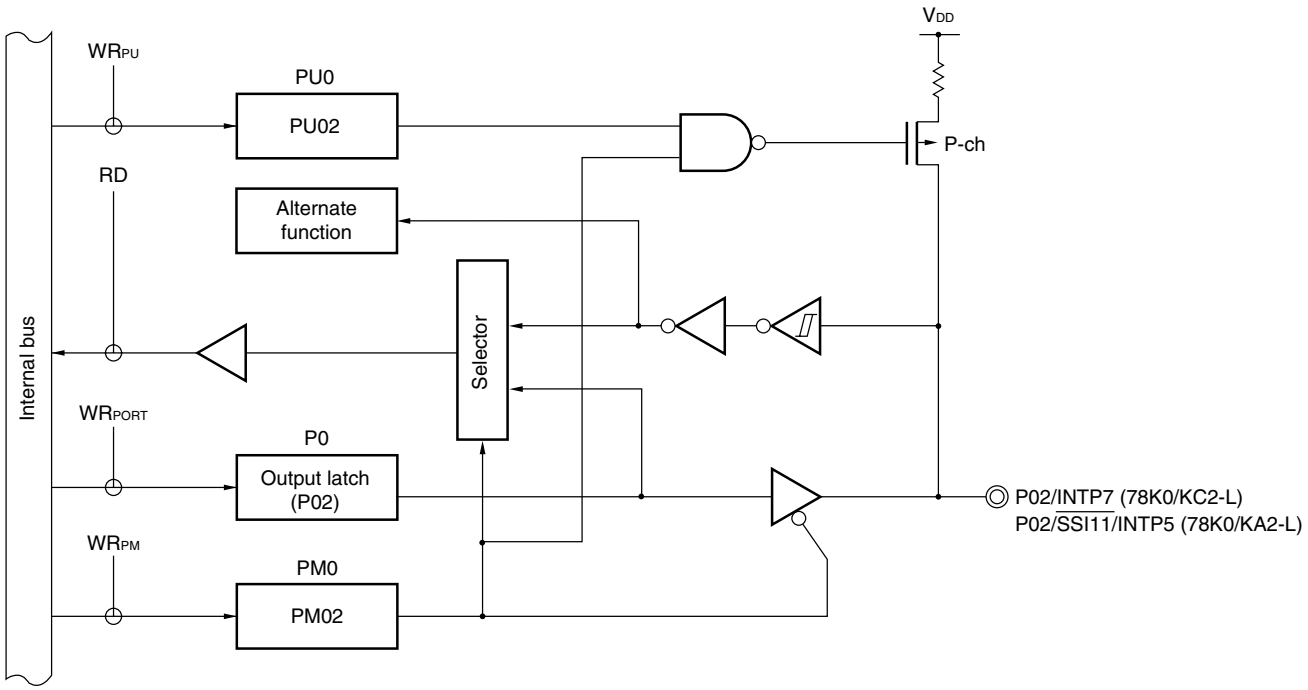
**Figure 4-2. Block Diagram of P01**




P0: Port register 0
PU0: Pull-up resistor option register 0
PM0: Port mode register 0
RD: Read signal
WR××: Write signal

<R>

**Figure 4-3. Block Diagram of P02**




P0: Port register 0
PU0: Pull-up resistor option register 0
PM0: Port mode register 0
RD: Read signal
WR××: Write signal

### 4.2.2 Port 1

<R>

| 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40, 44, 48 Pins |
| – | – | P10/ANI8/AMP1-Note/$\overline{SCK10}$ | P10/ANI8/AMP1-Note/$\overline{SCK10}$ |
| – | – | P11/ANI9/AMP1OUTNote/SI10 | P11/ANI9/AMP1OUTNote/SI10 |
| – | – | P12/ANI10/AMP1+Note/SO10 | P12/ANI10/AMP1+Note/SO10 |
| – | – | P13/TxD6 | P13/TxD6 |
| – | – | P14/RxD6 | P14/RxD6 |
| – | – | P15/TOH0 | P15/TOH0 |
| – | – | P16/TOH1/INTP5 | P16/TOH1/INTP5 |
| – | – | P17/TI50/TO50 | P17/TI50/TO50 |

**Note** Products with operational amplifier only

Port 1 is an I/O port with an output latch. Port 1 can be set to the input mode or output mode in 1-bit units using port mode register 1 (PM1). When the P10 to P17 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 1 (PU1).

This port can also be used for A/D converter analog input, operational amplifier I/O, external interrupt request input, serial interface data I/O, clock I/O, and timer I/O.

When using P10/ANI8/AMP1-, P11/ANI9/AMP1OUT, or P12/ANI10/AMP1+, set the registers according to the pin function to be used (refer to **Tables 4-8** and **4-9**).

**Table 4-8. Setting Functions of P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins**

| ADPC1 Register | PM1 Register | OPAMP1E bit | ADS Register (n = 8, 10) | P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins |
|---|---|---|---|---|
| Digital I/O selection | Input mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital input |
| | Output mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital output |
| Analog input selection | Input mode | 0 | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Operational amplifier 1 input |
| | Output mode | – | – | Setting prohibited |

**Remark** ADPC1: A/D port configuration register 1
PM1: Port mode register 1
OPAMP1E: Bit 7 of operational amplifier 1 control register (AMP1M)
ADS: Analog input channel specification register

**Table 4-9. Setting Functions of P11/ANI9/AMP1OUT Pin**

| ADPC1 Register | PM1 Register | OPAMP1E bit | ADS Register | P11/ANI9/AMP1OUT Pin |
|---|---|---|---|---|
| Digital I/O selection | Input mode | 0 | Selects ANI9. | Setting prohibited |
| | | | Does not select ANI9. | Digital input |
| | | 1 | – | Setting prohibited |
| | Output mode | 0 | Selects ANI9. | Setting prohibited |
| | | | Does not select ANI9. | Digital output |
| | | 1 | – | Setting prohibited |
| Analog I/O selection | Input mode | 0 | Selects ANI9. | Analog input (to be converted into digital signals) |
| | | | Does not select ANI9. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANI9. | Operational amplifier 1 output (to be converted into digital signals) |
| | | | Does not select ANI9. | Operational amplifier 1 output (not to be converted into digital signals) |
| | Output mode | – | – | Setting prohibited |

**Remark** ADPC1: A/D port configuration register 1
PM1: Port mode register 1
OPAMP1E: Bit 7 of operational amplifier 1 control register (AMP1M)
ADS: Analog input channel specification register

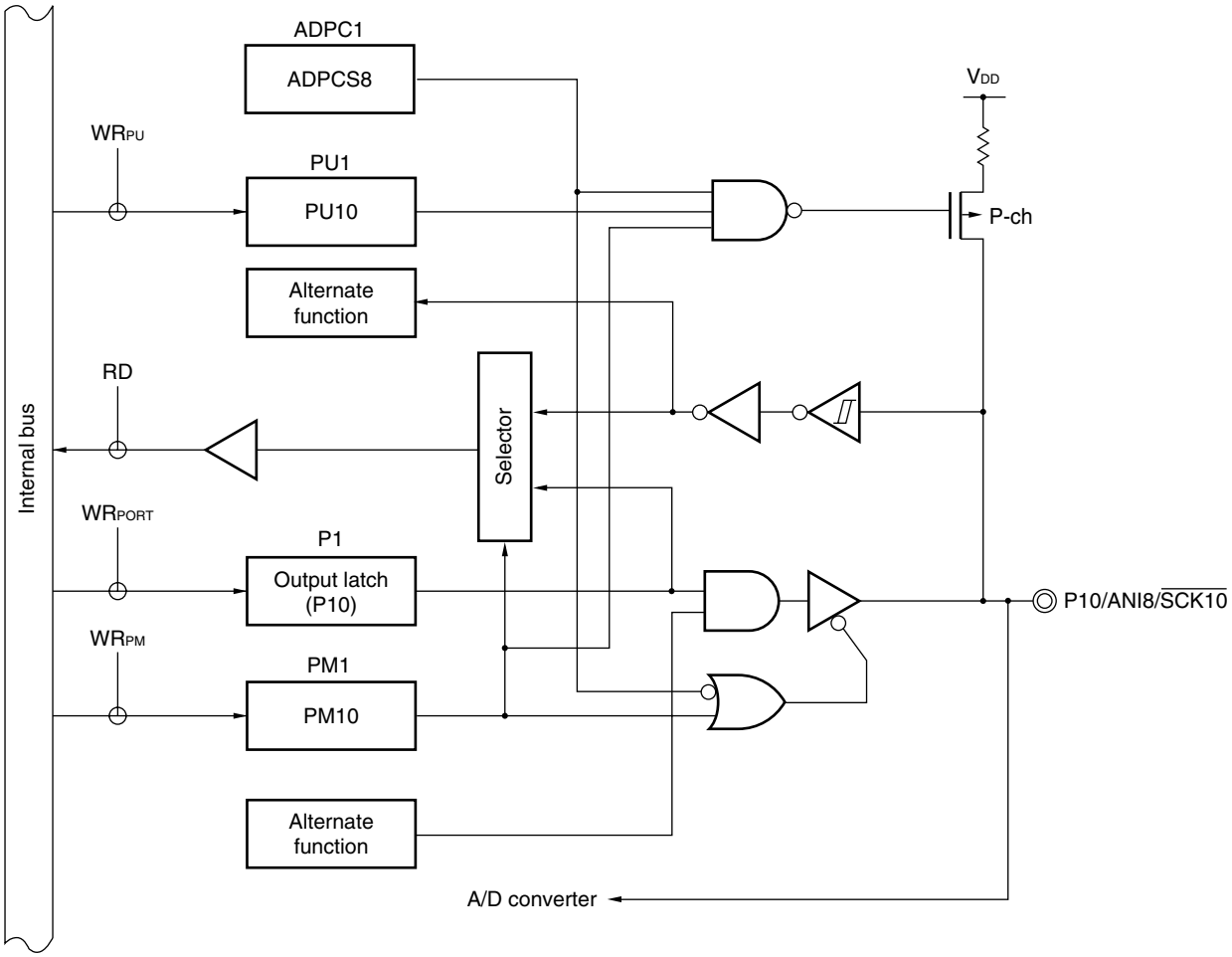
Reset signal generation sets port 1 to digital input.
Figures 4-4 to 4-10 show block diagrams of port 1.

**Cautions 1. To use P10/$\overline{\text{SCK10}}$ and P12/SO10 of 78K0/KB2-L and 78K0/KC2-L as general-purpose ports, set serial operation mode register 10 (CSIM10) and serial clock selection register 10 (CSIC10) to the default status (00H).**

**2. To use P13/TxD6 of 78K0/KB2-L and 78K0/KC2-L as general-purpose port, clear bit 0 (TXDLV6) of asynchronous serial interface control register 6 (ASICL6) to 0 (normal output of TxD6).**

<R> **Figure 4-4. Block Diagram of P10 (1/2)**

**(1) Products without operational amplifier of 78K0/KB2-L and 78K0/KC2-L**

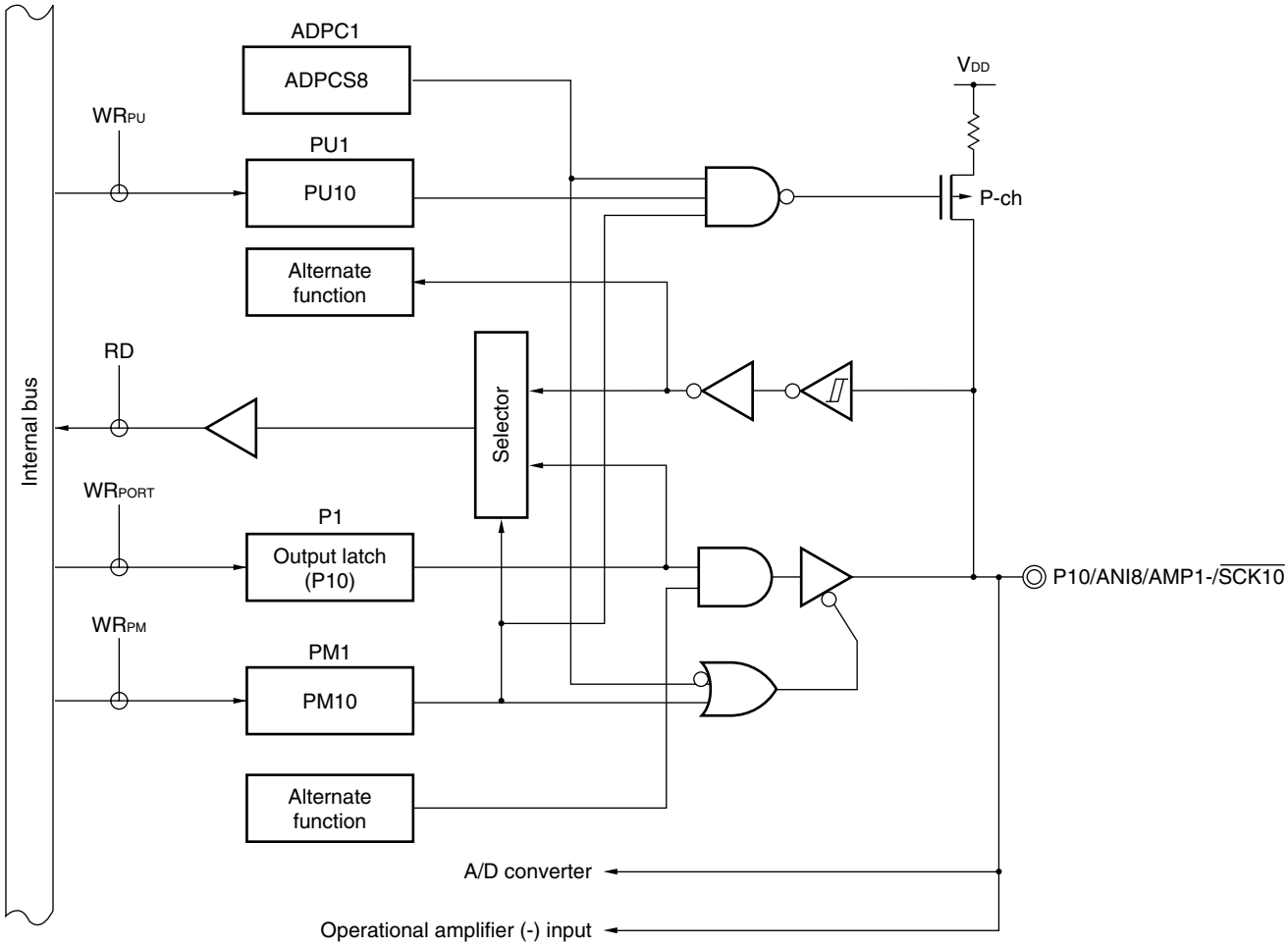



P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
ADPC1: A/D port configuration register 1
RD: Read signal
WR××: Write signal

<R>

**Figure 4-4. Block Diagram of P10 (2/2)**

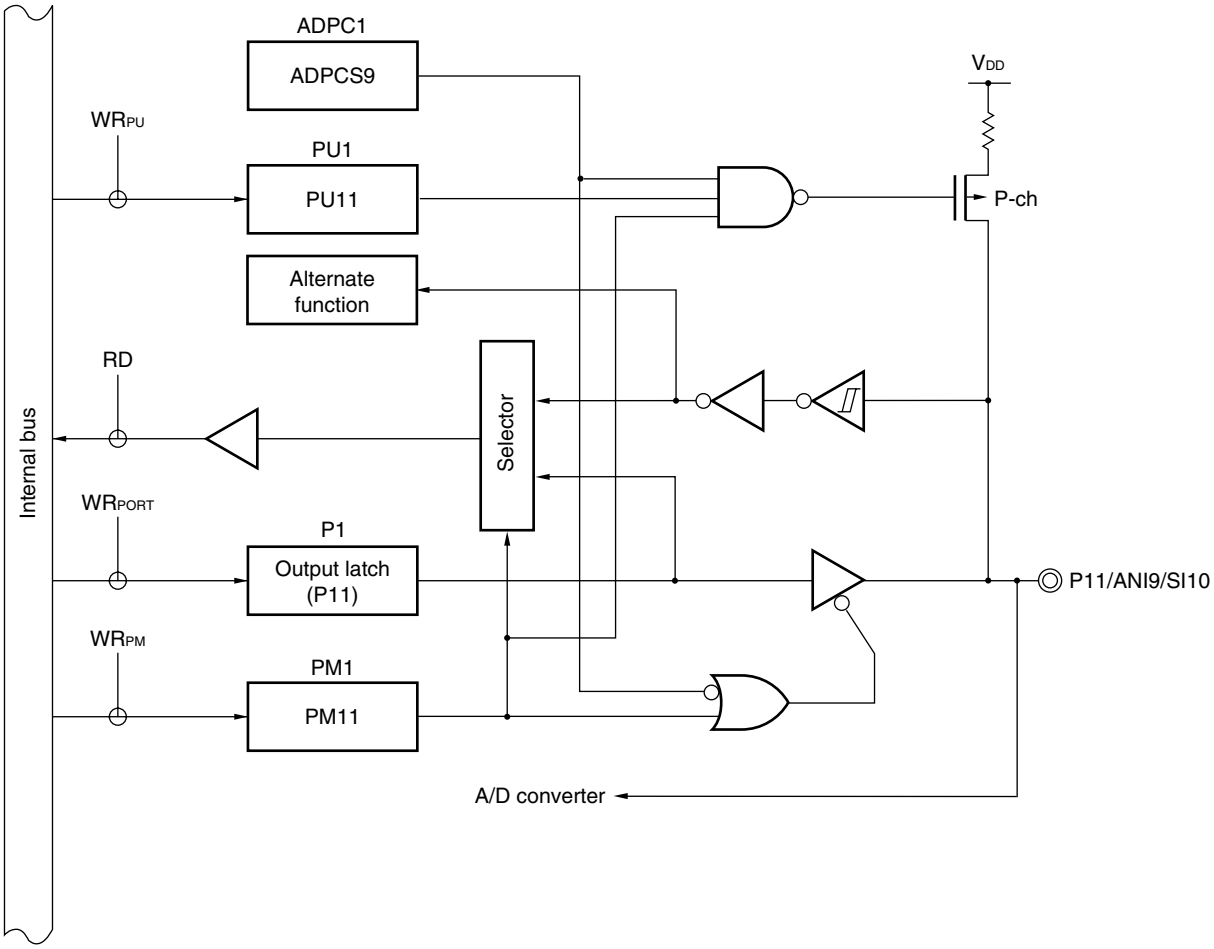
**(2) Products with operational amplifier of 78K0/KB2-L and 78K0/KC2-L**




P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
ADPC1: A/D port configuration register 1
RD: Read signal
WR××: Write signal

<R> **Figure 4-5. Block Diagram of P11 (1/2)**

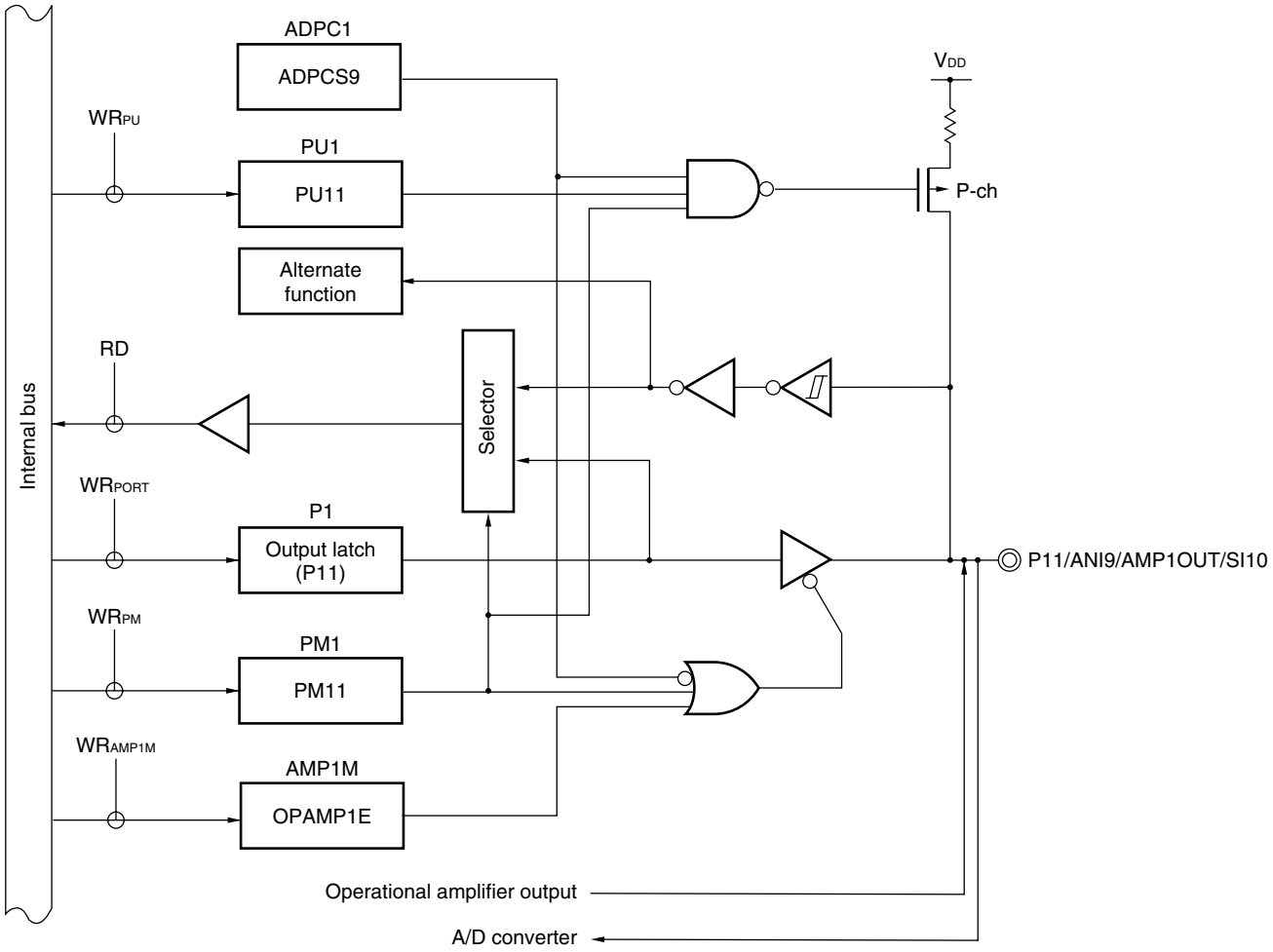
**(1) Products without operational amplifier of 78K0/KB2-L and 78K0/KC2-L**




P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
ADPC1: A/D port configuration register 1
RD: Read signal
WR××: Write signal

<R> **Figure 4-5. Block Diagram of P11 (2/2)**

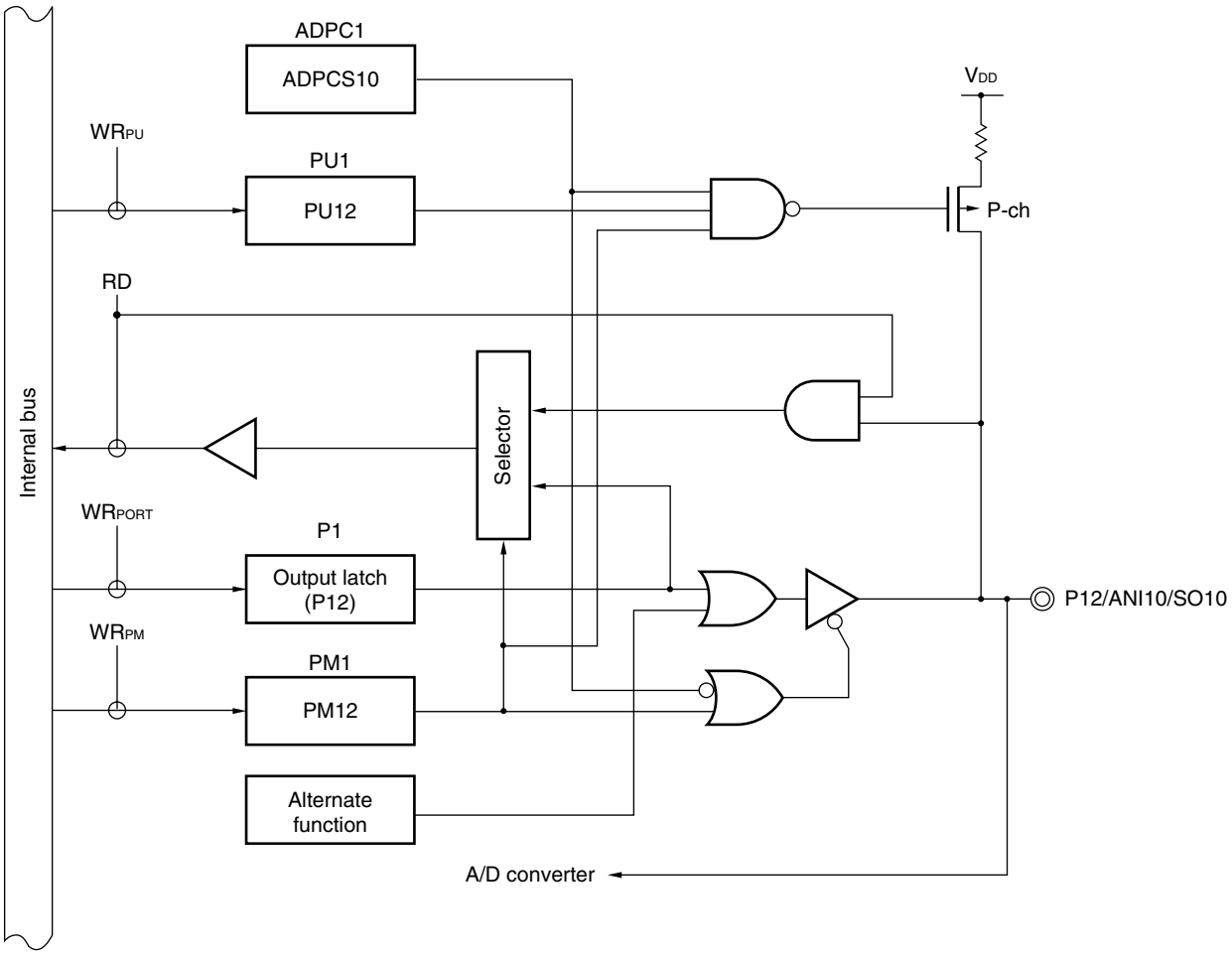
**(2) Products with operational amplifier of 78K0/KB2-L and 78K0/KC2-L**




P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
ADPC1: A/D port configuration register 1
RD: Read signal
WR××: Write signal

<R> **Figure 4-6. Block Diagram of P12 (1/2)**

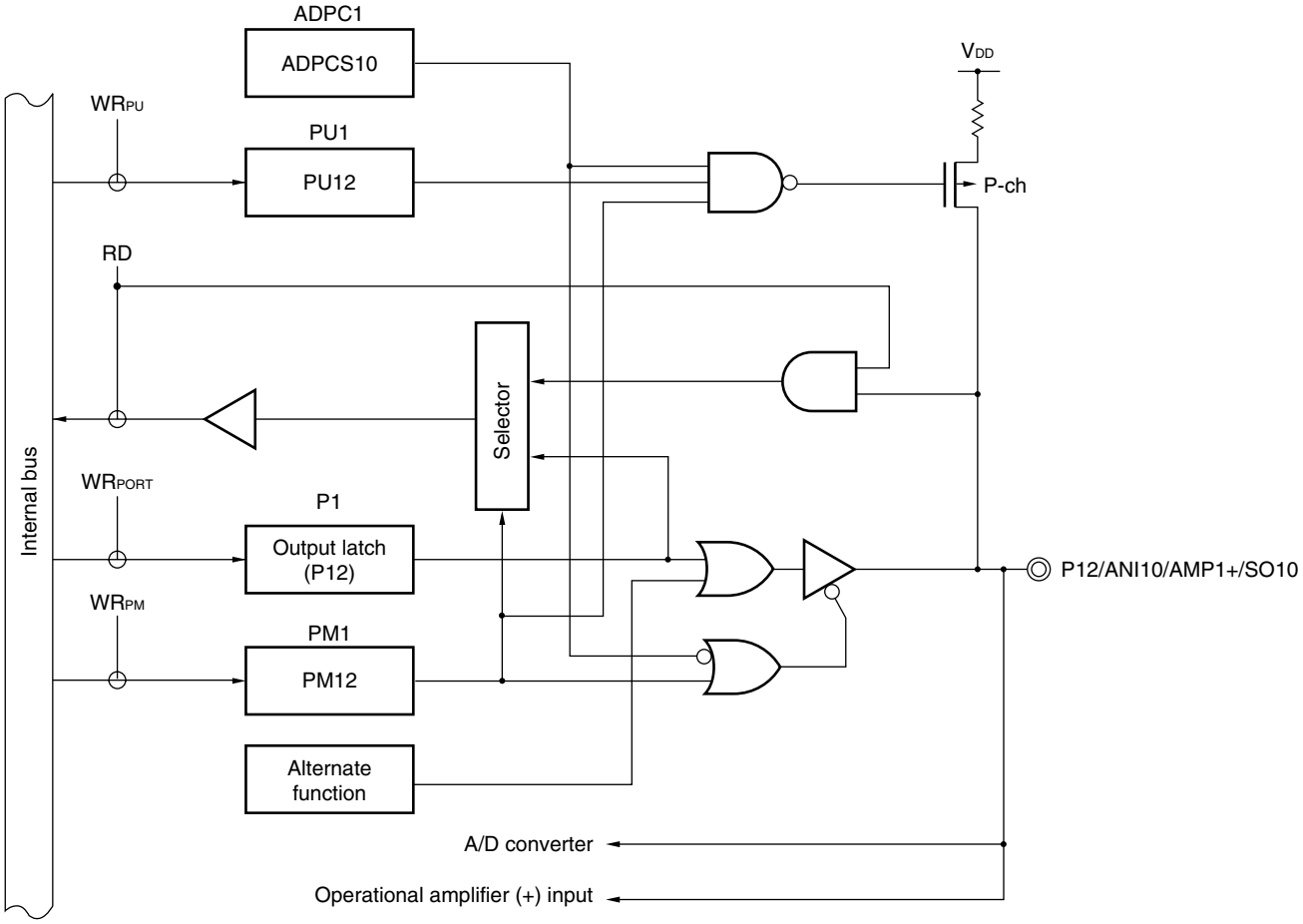
**(1) Products without operational amplifier of 78K0/KB2-L and 78K0/KC2-L**




P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
ADPC1: A/D port configuration register 1
RD: Read signal
WR××: Write signal

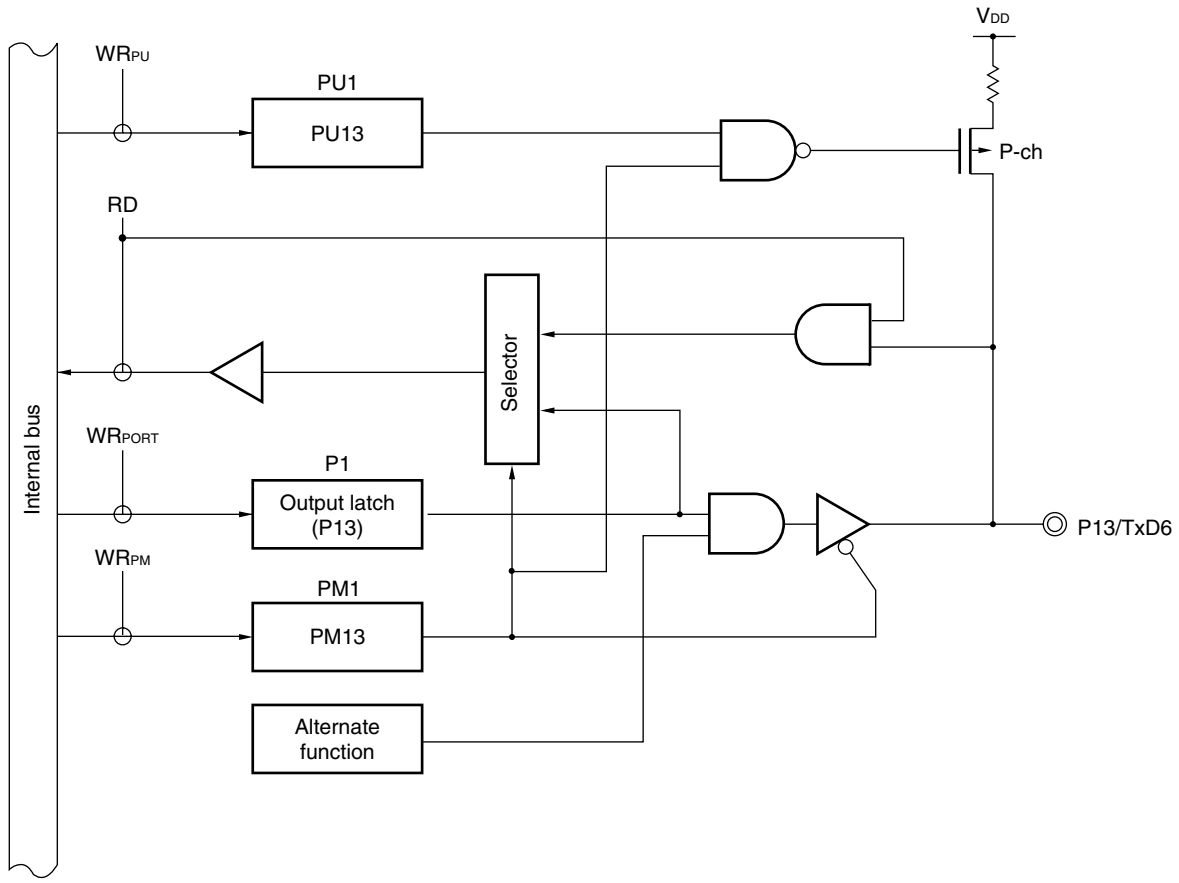
<R> **Figure 4-6. Block Diagram of P12 (2/2)**

**(2) Products with operational amplifier of 78K0/KB2-L and 78K0/KC2-L**



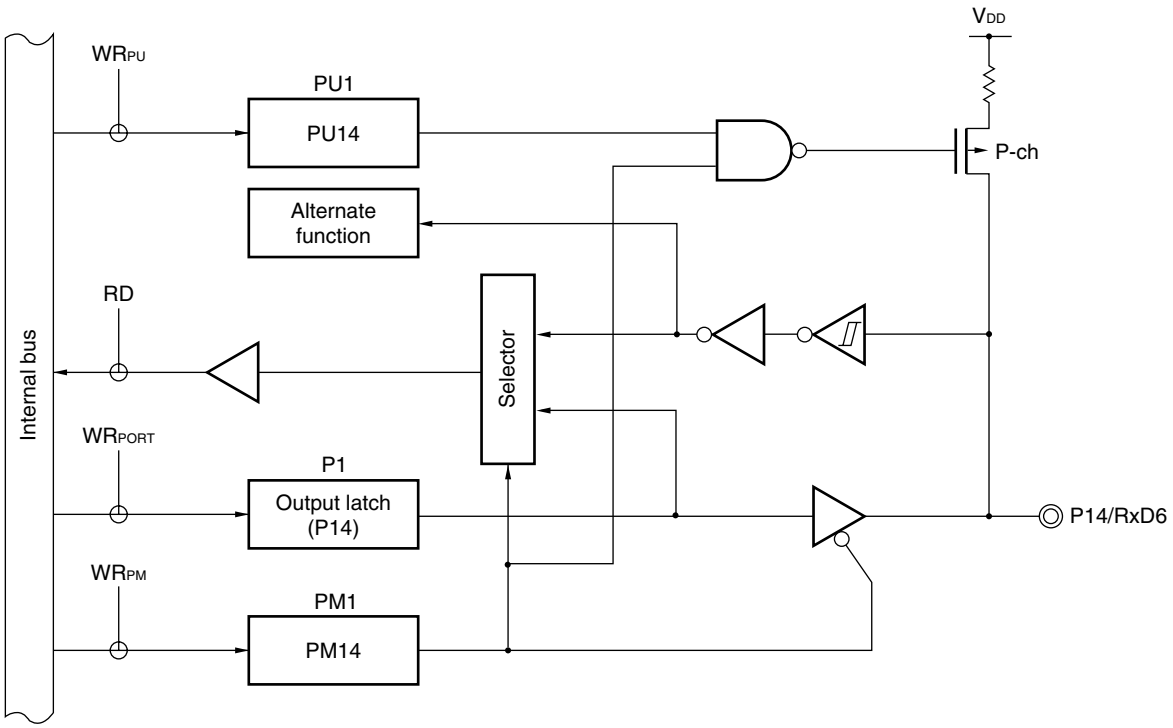

P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
ADPC1: A/D port configuration register 1
RD: Read signal
WR××: Write signal

**Figure 4-7. Block Diagram of P13**



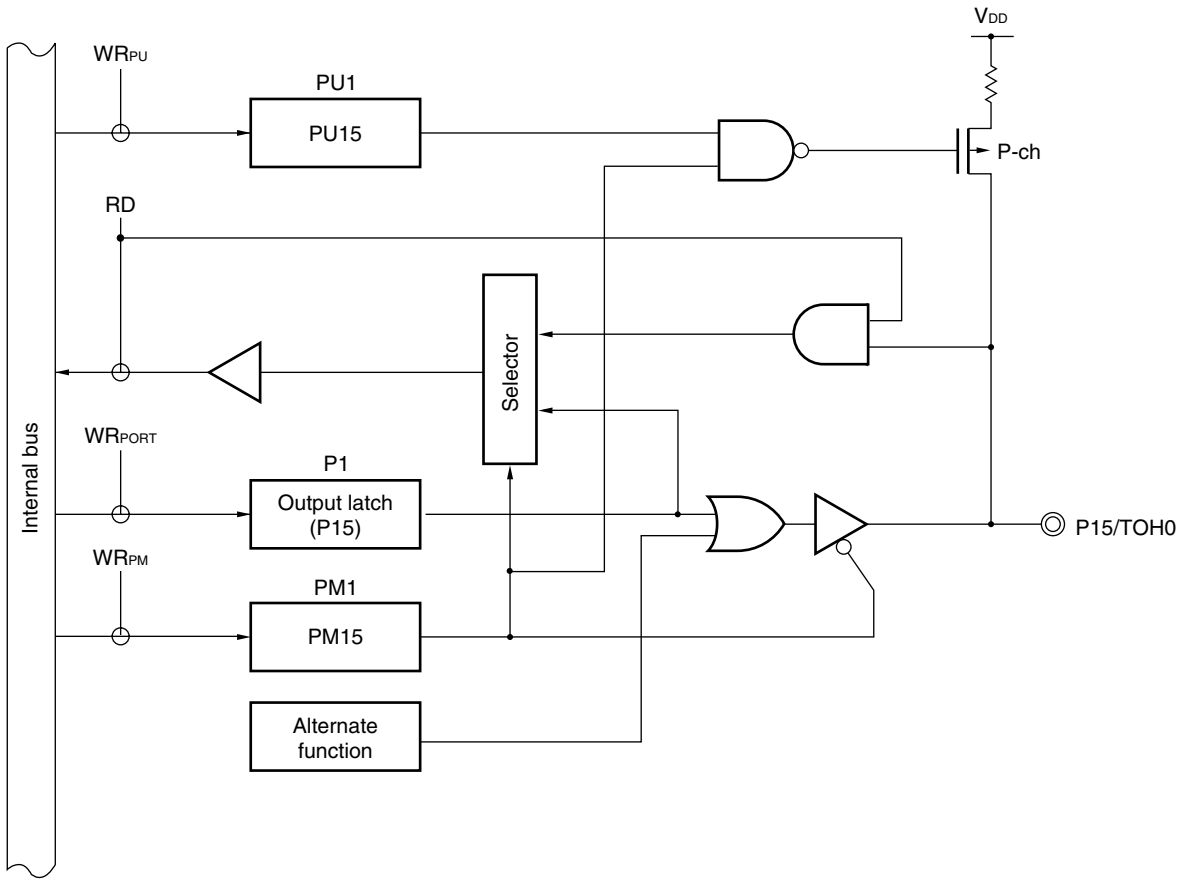

P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
RD: Read signal
WR××: Write signal

**Figure 4-8. Block Diagram of P14**



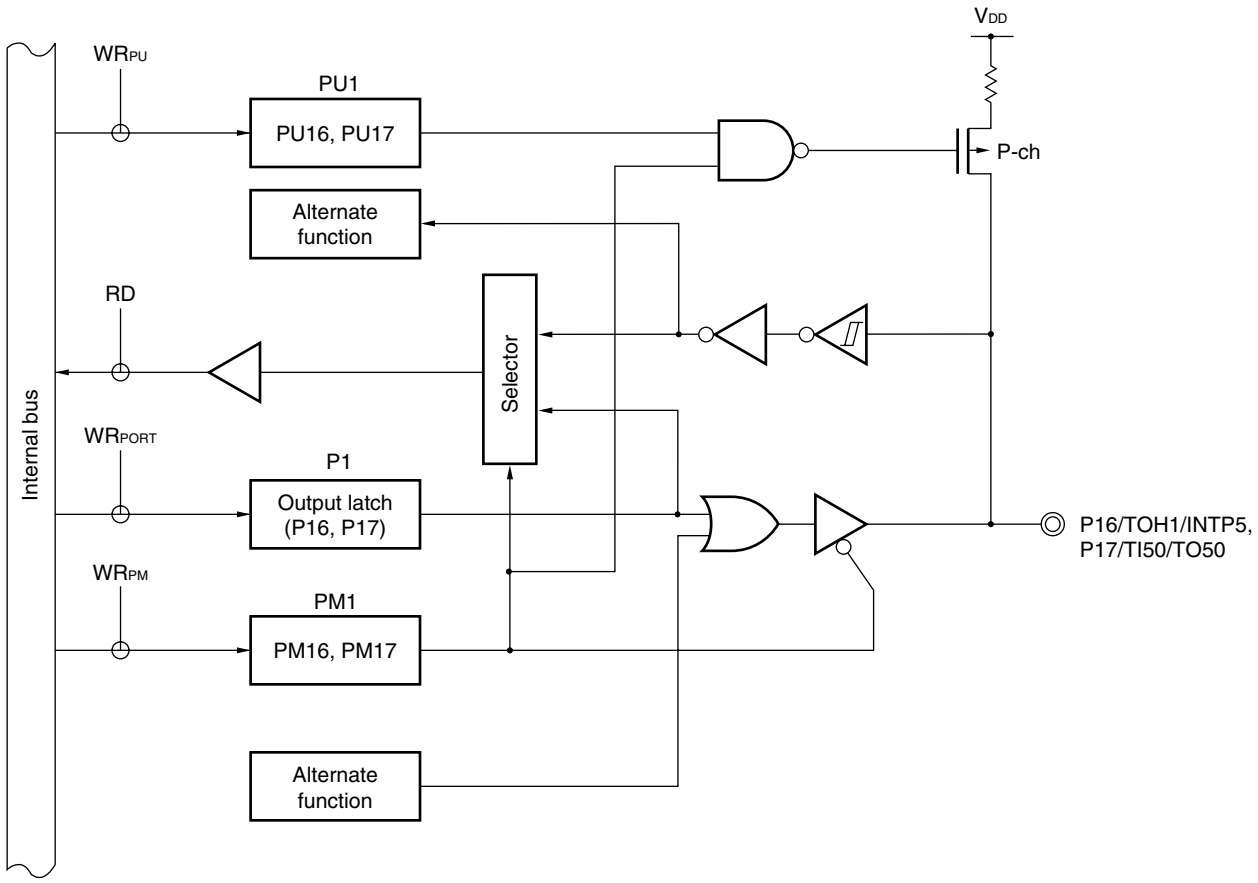

P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
RD: Read signal
WR××: Write signal

**Figure 4-9. Block Diagram of P15**




P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
RD: Read signal
WR××: Write signal

**Figure 4-10. Block Diagram of P16 and P17**




P1: Port register 1
PU1: Pull-up resistor option register 1
PM1: Port mode register 1
RD: Read signal
WR××: Write signal

### 4.2.3 Port 2

<R>

| 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note | P20/ANI0/ AMP0-Note |
| P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote | P21/ANI1/ AMP0OUTNote/ PGAINNote |
| P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note | P22/ANI2/ AMP0+Note |
| P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 | P23/ANI3 |
| – | P24/ANI4 | P24/ANI4 | P24/ANI4 | – | P24/ANI4 | P24/ANI4 | P24/ANI4 |
| – | P25/ANI5 | P25/ANI5 | P25/ANI5 | – | P25/ANI5 | P25/ANI5 | P25/ANI5 |
| – | – | P26/ANI6 | P26/ANI6 | – | P26/ANI6 | P26/ANI6 | P26/ANI6 |
| – | – | – | P27/ANI7 | – | – | P27/ANI7 | P27/ANI7 |

**Note** Products with operational amplifier only

Port 2 is an I/O port with an output latch. Port 2 can be set to the input mode or output mode in 1-bit units using port mode register 2 (PM2).

This port can also be used for A/D converter analog input, operational amplifier I/O, and PGA (Programmable Gain Amplifier) input.

When using P20/AMP0-/ANI0 to P27/ANI7, set the registers according to the pin function to be used (refer to **Tables 4-10** to **4-12**).

To use P20/AMP0-/ANI0 to P27/ANI7 as a digital input or a digital output, it is recommended to select a pin to use starting with the furthest pin from $AV_{REF}$ (for example, the P20/AMP0-/ANI0 pin in the 78K0/KC2-L). To use P20/AMP0-/ANI0 to P27/ANI7 as an analog input, it is recommended to select a pin to use starting with the closest pin to $AV_{SS}$ (for example, the P27/ANI7 pin in the 78K0/KC2-L (44-pin and 48-pin products)).

**Table 4-10. Setting Functions of P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins**

| ADPC0 Register | PM2 Register | OPAMP0E bit | ADS Register (n = 0, 2) | P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins |
|---|---|---|---|---|
| Digital I/O selection | Input mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital input |
| | Output mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital output |
| Analog input selection | Input mode | 0 | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Operational amplifier 0 input |
| | Output mode | – | – | Setting prohibited |

**Remark** ADPC0: A/D port configuration register 0
PM2: Port mode register 2
OPAMP0E: Bit 7 of operational amplifier 0 control register (AMP0M)
ADS: Analog input channel specification register

<R> **Table 4-11. Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin**

| ADPC0 Register | PM2 Register | OPAMP0E bit | PGAEN bit | ADS Register | P21/ANI1/AMP0OUT/PGAIN Pin |
|---|---|---|---|---|---|
| Digital I/O selection | Input mode | 0 | – | Selects ANI1. | Setting prohibited |
| | | | | Does not select ANI1. | Digital input |
| | | 1 | – | – | Setting prohibited |
| | Output mode | 0 | – | Selects ANI1. | Setting prohibited |
| | | | | Does not select ANI1. | Digital output |
| | | 1 | – | – | Setting prohibited |
| Analog I/O selection | Input mode | 0 | 0 | Selects ANI1. | Analog input (to be converted into digital signals) |
| | | | | Does not select ANI1. | Analog input (not to be converted into digital signals) |
| | | 0 | 1 | Selects PGAOUT. | PGA input (PGA output is converted into digital signals) |
| | | | | Selects ANI1. | PGA input (to be converted into digital signals) |
| | | | | Does not select PGAOUT and ANI1. | PGA input (not to be converted into digital signals) |
| | | 1 | 0 | Selects ANI1. | Operational amplifier 0 output (to be converted into digital signals) |
| | | | | Does not select ANI1. | Operational amplifier 0 output (not to be converted into digital signals) |
| | | 1 | 1 | Selects PGAOUT. | Operational amplifier 0 output and PGA input (PGA output is converted into digital signals) |
| | | | | Selects ANI1. | Operational amplifier 0 output (to be converted into digital signals) |
| | | | | Does not select PGAOUT and ANI1. | Operational amplifier 0 output (not to be converted into digital signals) |
| | Output mode | – | | – | Setting prohibited |

**Table 4-12. Setting Functions of P23/ANI3 to P27/ANI7 Pins**

| ADPC0 Register | PM2 Register | ADS Register(n = 3 to 7) | P23/ANI3 to P27/ANI7 Pins |
|---|---|---|---|
| Digital I/O selection | Input mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital input |
| | Output mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital output |
| Analog input selection | Input mode | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | Output mode | – | Setting prohibited |

**Remark** ADPC0: A/D port configuration register 0
PM2: Port mode register 2
OPAMP0E: Bit 7 of operational amplifier 0 control register (AMP0M)
PGAEN: Bit 6 of AMP0M
ADS: Analog input channel specification register

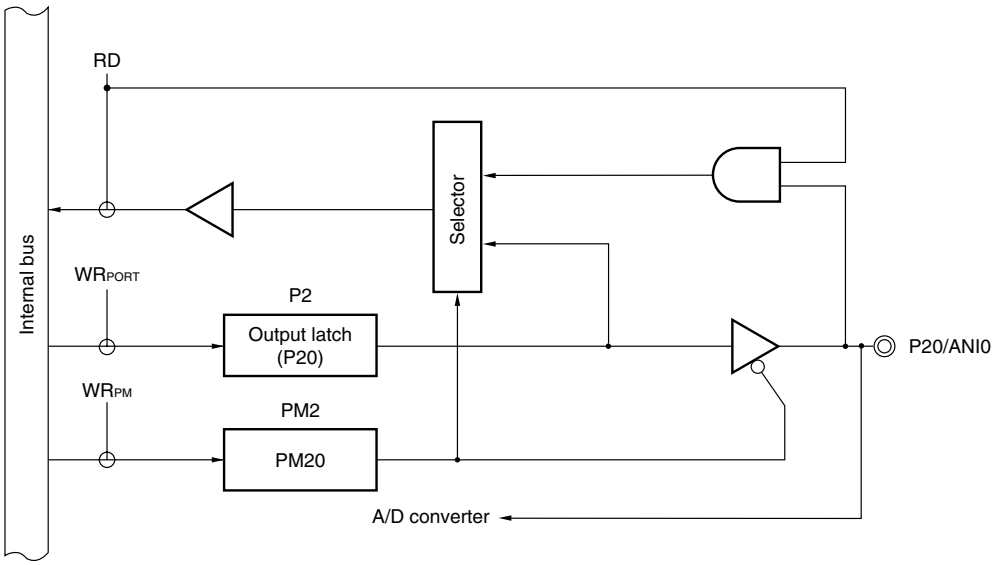
Reset signal generation sets port 2 to analog input.

Figures 4-11 to 4-14 show block diagrams of port 2.

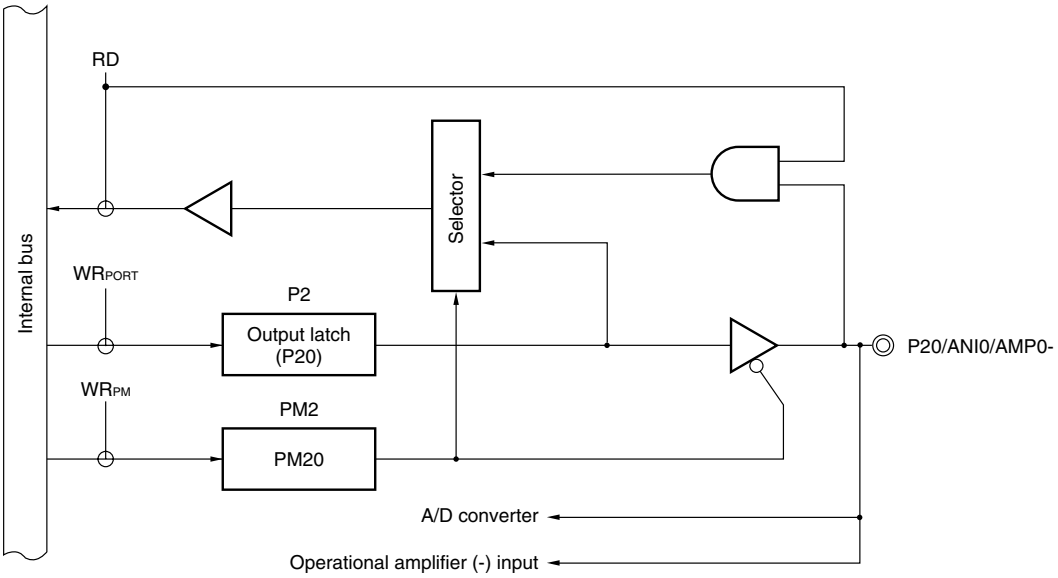
**Caution Make the $AV_{REF}$ pin the same potential as the $V_{DD}$ pin when port 2 is used as a digital port.**

**Figure 4-11. Block Diagram of P20**

**(1) Products without operational amplifier**



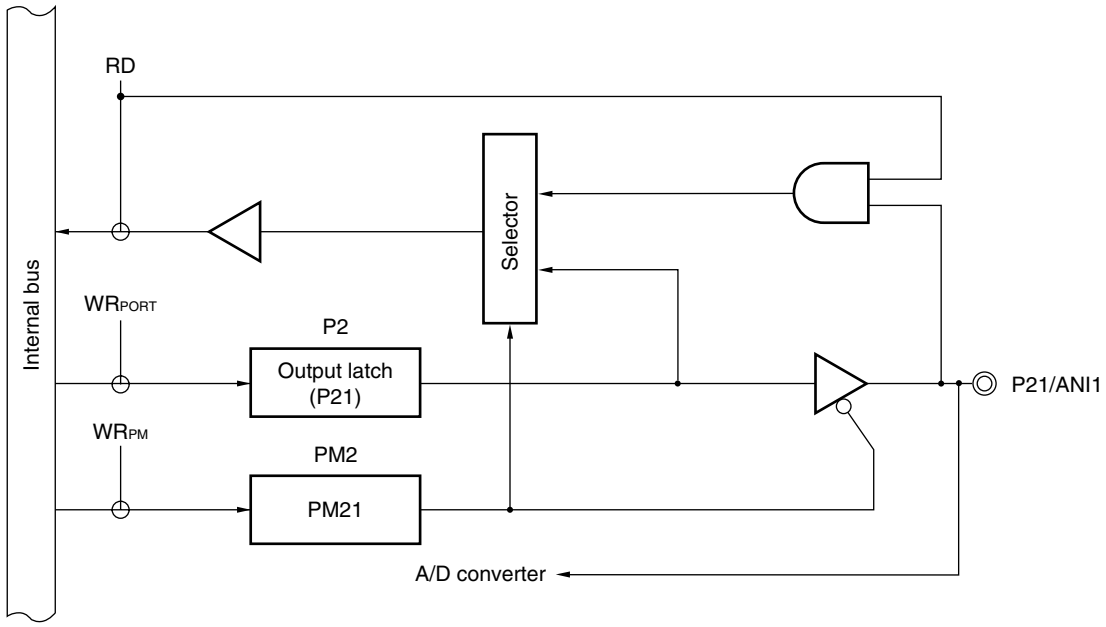

**(2) Products with operational amplifier**



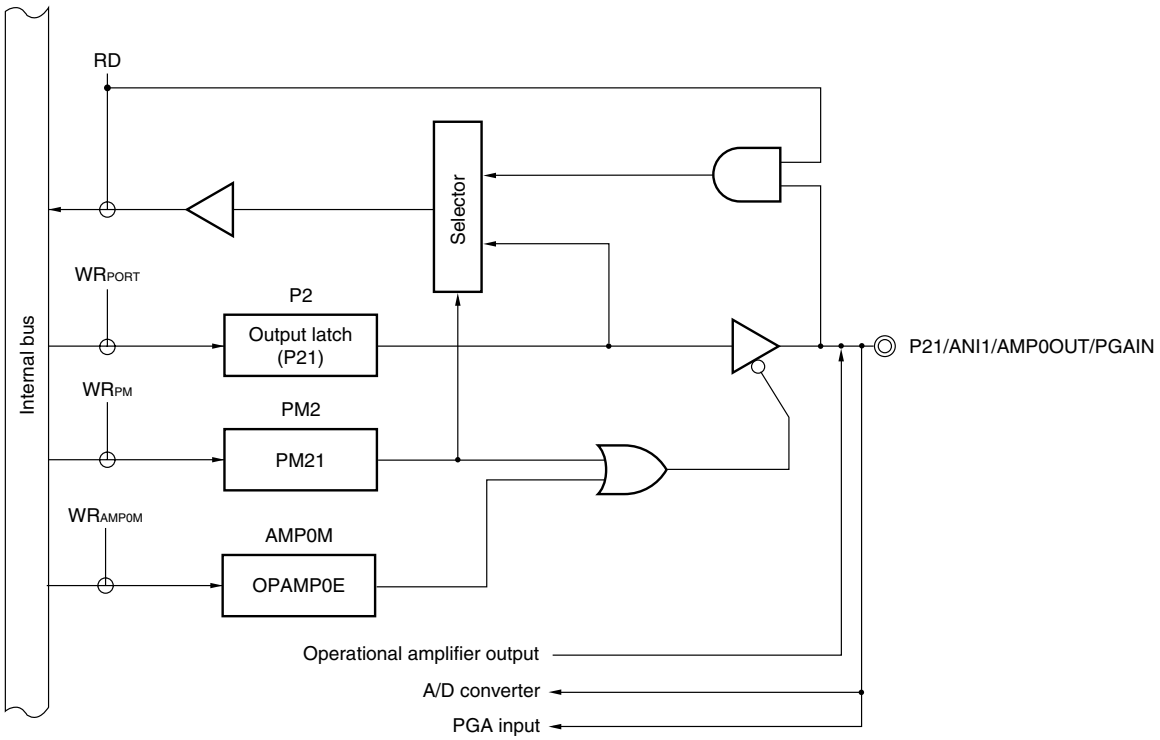

P2: Port register 2
PM2: Port mode register 2
RD: Read signal
WR××: Write signal

**Figure 4-12. Block Diagram of P21**

**(1) Products without operational amplifier**




<R> **(2) Products with operational amplifier**

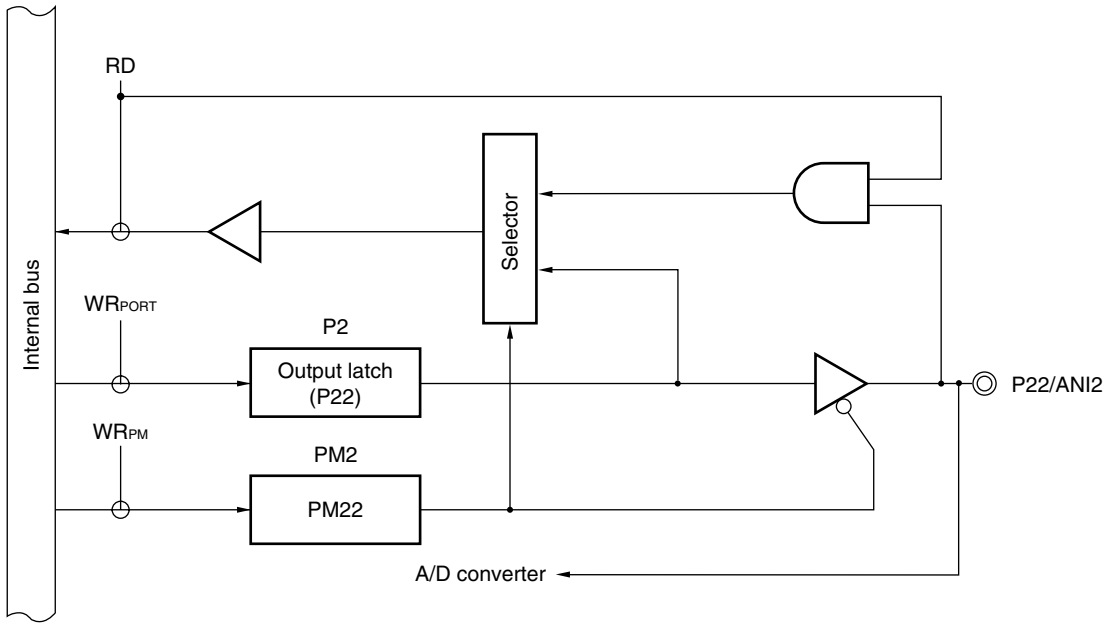



P2: Port register 2
PM2: Port mode register 2
RD: Read signal
WR××: Write signal

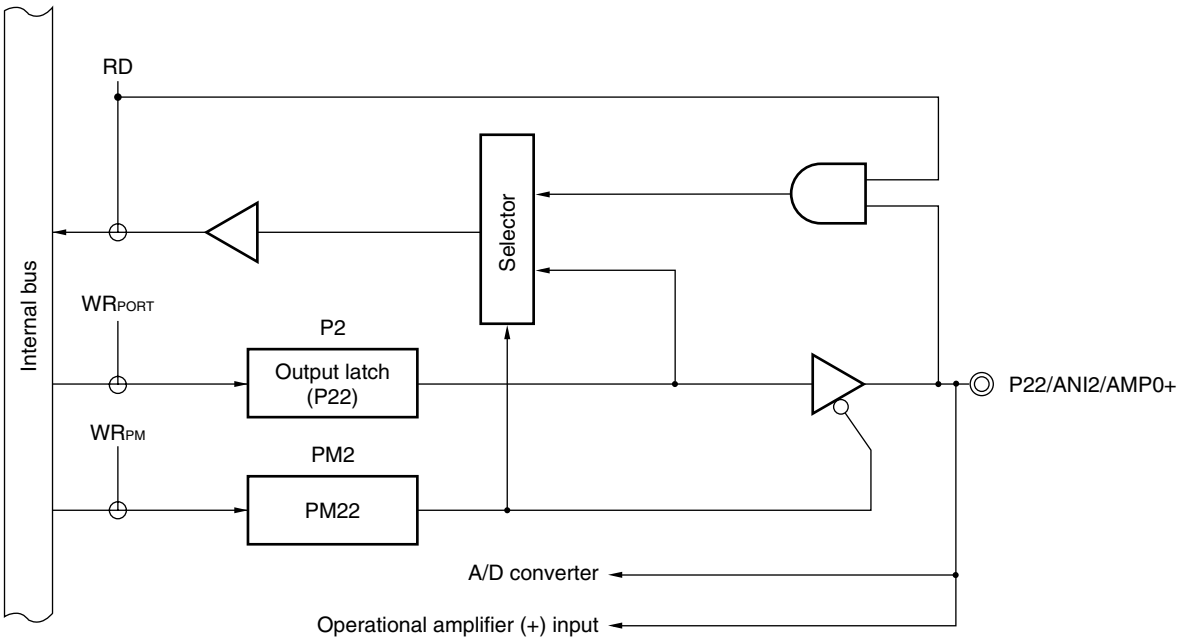
**Remark** PGA: Programmable Gain Amplifier

**Figure 4-13. Block Diagram of P22**

**(1) Products without operational amplifier**

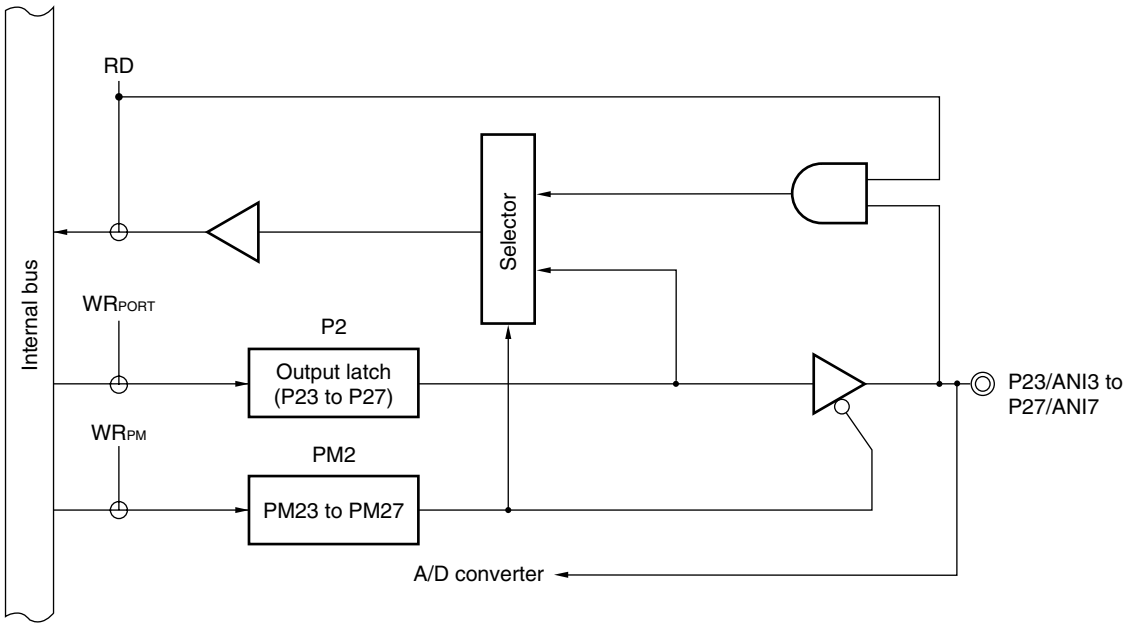



**(2) Products with operational amplifier**




P2: Port register 2
PM2: Port mode register 2
RD: Read signal
WR××: Write signal

**Figure 4-14. Block Diagram of P23-P27**




P2: Port register 2
PM2: Port mode register 2
RD: Read signal
WR××: Write signal

### 4.2.4 Port 3

<R>

| 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| P30/TOH1/ TI51/INTP1 | P30/TOH1/ TI51/INTP1 | – | – | P30/INTP1 | P30/INTP1 | P30/INTP1 | P30/INTP1 |
| – | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 | P31/INTP2/ TOOLC1 |
| – | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 | P32/INTP3/ TOOLD1 |
| – | – | P33 | P33 | P33/TI51/ TO51/INTP4 | P33/TI51/ TO51/INTP4 | P33/TI51/ TO51/INTP4 | P33/TI51/ TO51/INTP4 |
| – | – | P34/INTP4 (/TOH1) (/TI51) | P34/INTP4 (/TOH1) | – | – | – | – |
| – | – | P35/$\overline{\text{SCK11}}$ | P35/$\overline{\text{SCK11}}$ | – | – | – | – |
| – | – | P36/SI11 | P36/SI11 | – | – | – | – |
| – | – | P37/SO11 | P37/SO11 | – | – | – | – |

Port 3 is an I/O port with an output latch. Port 3 can be set to the input mode or output mode in 1-bit units using port mode register 3 (PM3). When the P30 to P37 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 3 (PU3).

This port can also be used for external interrupt request input, timer I/O, clock input and data I/O for flash memory programmer/on-chip debugger, and clock I/O and data I/O for serial interface.

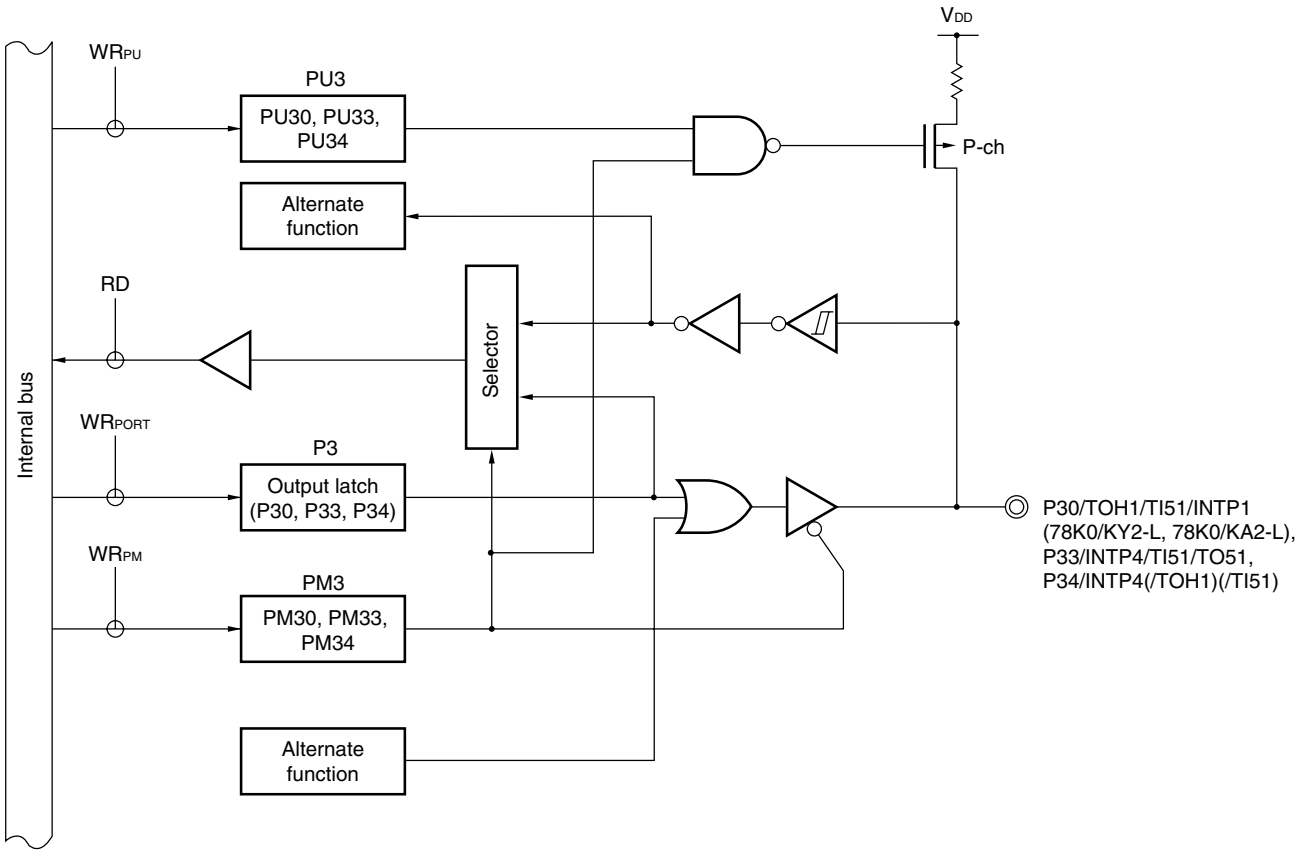
The timer I/O can be assigned to P34 of the 78K0/KA2-L (25-pin and 32-pin products) by setting the port alternate switch control register (MUXSEL).

Reset signal generation sets port 3 to input mode.

Figures 4-15 to 4-18 show block diagrams of port 3.

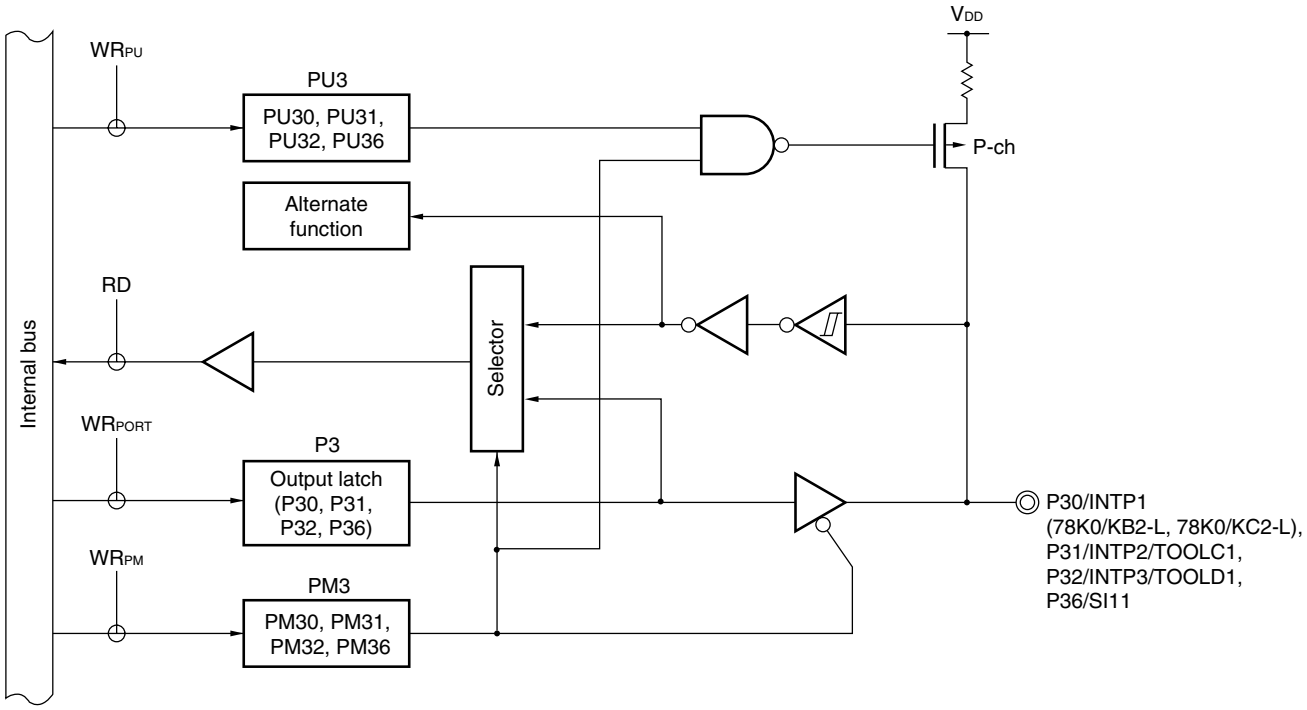
**Remark** For how to connect a flash memory programmer using TOOLC1/P31, TOOLD1/P32, refer to **CHAPTER 25 FLASH MEMORY**. For how to connect TOOLC1/P31, TOOLD1/P32 and an on-chip debug emulator, refer to **CHAPTER 26 ON-CHIP DEBUG FUNCTION**.

**Figure 4-15. Block Diagram of P30 (78K0/KY2-L, 78K0/KA2-L), P33, P34**



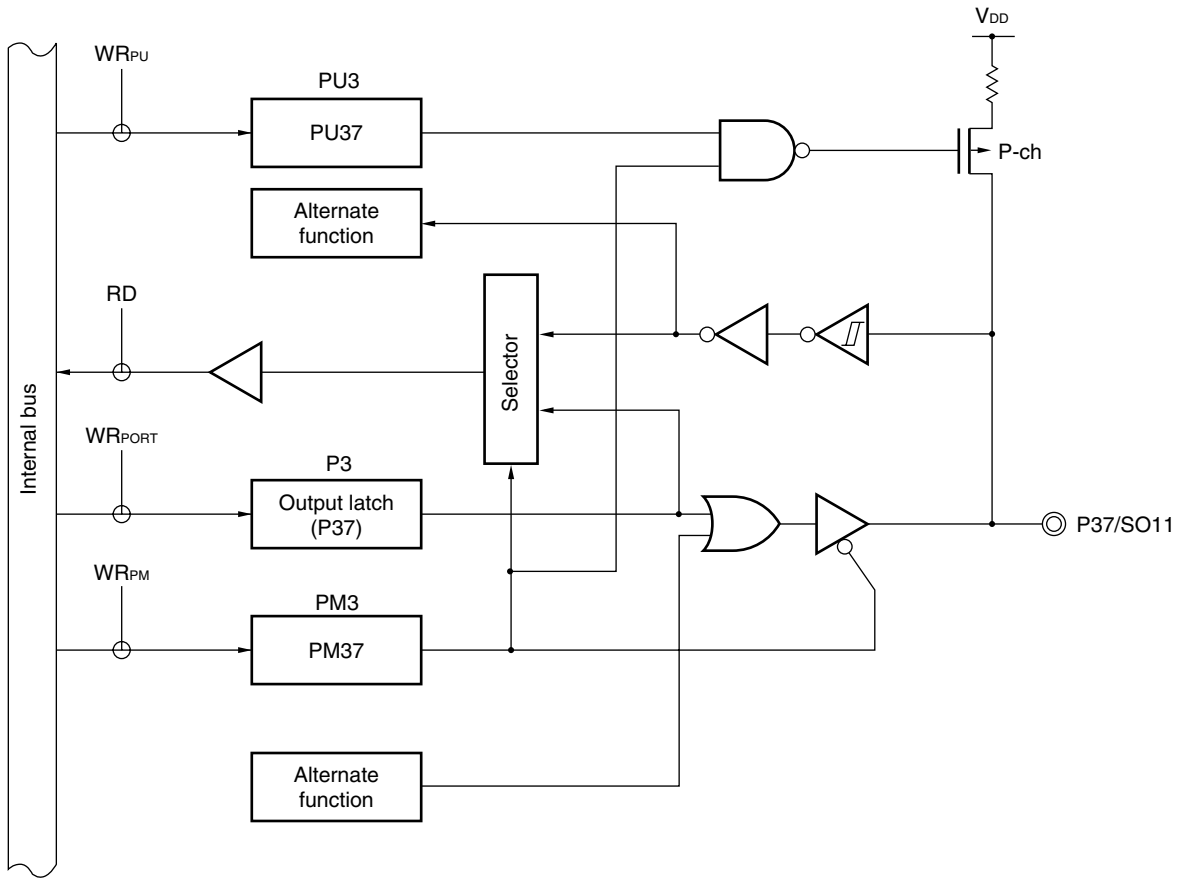

P3: Port register 3
PU3: Pull-up resistor option register 3
PM3: Port mode register 3
RD: Read signal
WR××: Write signal

**Figure 4-16. Block Diagram of P30 (78K0/KB2-L, 78K0/KC2-L), P31, P32, P36**



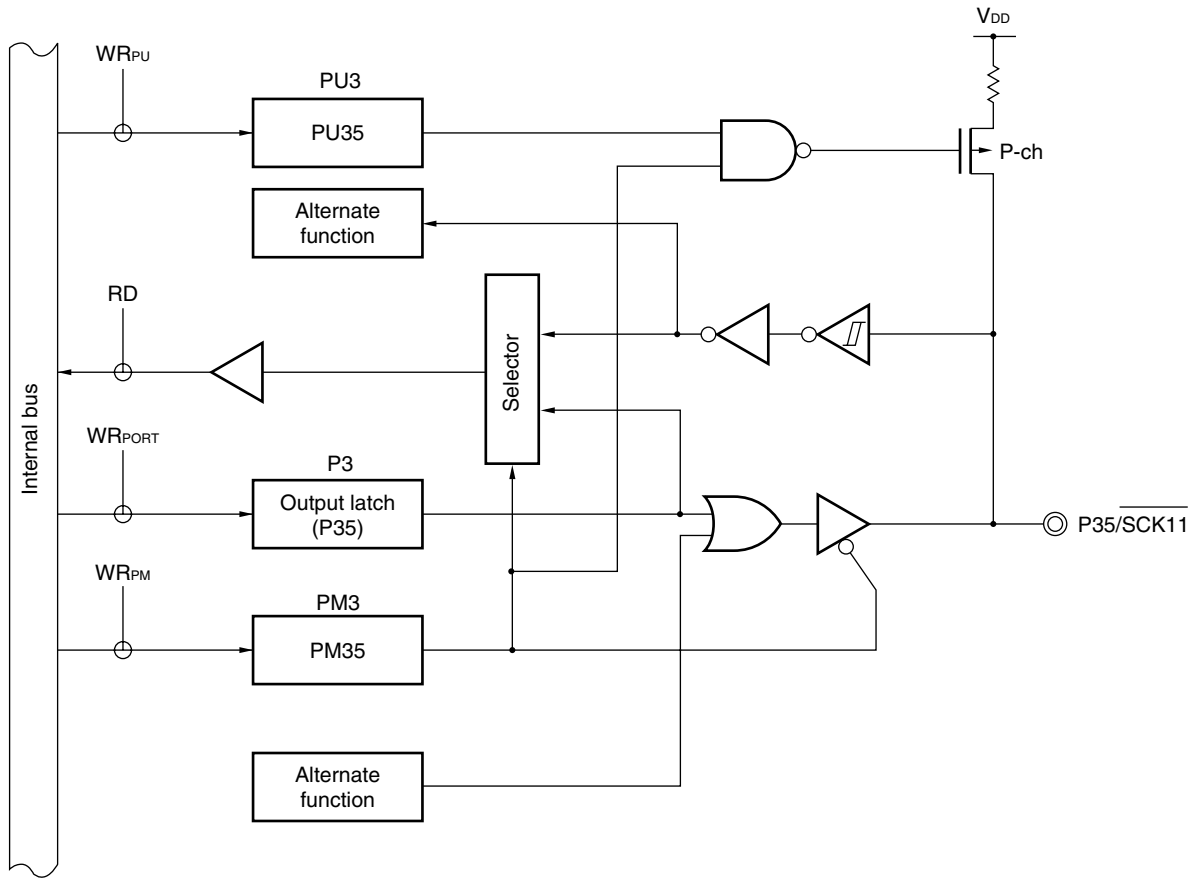

P3: Port register 3
PU3: Pull-up resistor option register 3
PM3: Port mode register 3
RD: Read signal
WR××: Write signal

**Figure 4-17. Block Diagram of P37**




P3: Port register 3
PU3: Pull-up resistor option register 3
PM3: Port mode register 3
RD: Read signal
WR××: Write signal

**Figure 4-18. Block Diagram of P35**




P3: Port register 3
PU3: Pull-up resistor option register 3
PM3: Port mode register 3
RD: Read signal
WR××: Write signal

### 4.2.5 Port 4

<R>

| 78K0/KY2-L ($\mu$PD78F057x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | | |
|---|---|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| – | – | – | – | P40/RTCCL/ RTCDIV(/$\overline{\text{SCK11}}$) | P40/RTCCL/ RTCDIV(/$\overline{\text{SCK11}}$) |
| – | – | – | – | P41/RTC1HZ (/SI11) | P41/RTC1HZ (/SI11) |
| – | – | – | – | – | P42/PCL/$\overline{\text{SSI11}}$/ INTP6 |

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

Port 4 is an I/O port with an output latch. Port 4 can be set to the input mode or output mode in 1-bit units using port mode register 4 (PM4). When the P40 to P42 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 4 (PU4).

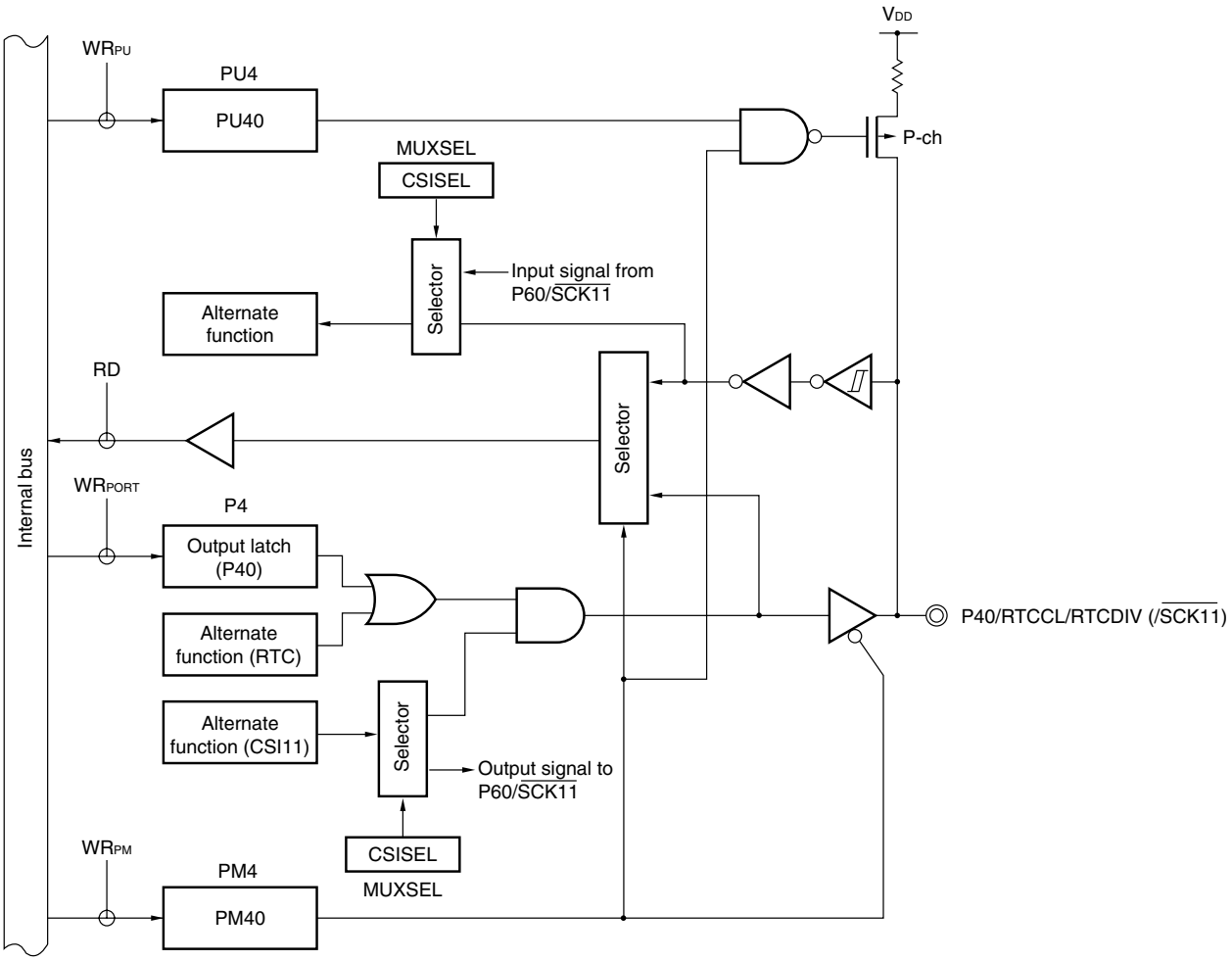
This port can also be used for external interrupt request input, real-time counter clock output, real-time counter correction clock output, and chip select input of serial interface.

The clock I/O and data input of the serial interface can be assigned to P40 and P41 of the 78K0/KC2-L (44-pin and 48-pin products) respectively by setting the port alternate switch control register (MUXSEL).

Reset signal generation sets port 4 to input mode.

Figures 4-19 to 4-21 show block diagrams of port 4.

**Figure 4-19. Block Diagram of P40**




P4: Port register 4
PU4: Pull-up resistor option register 4
PM4: Port mode register 4
MUXSEL: Port alternate switch control register
RD: Read signal
WR××: Write signal

**Figure 4-20. Block Diagram of P41**

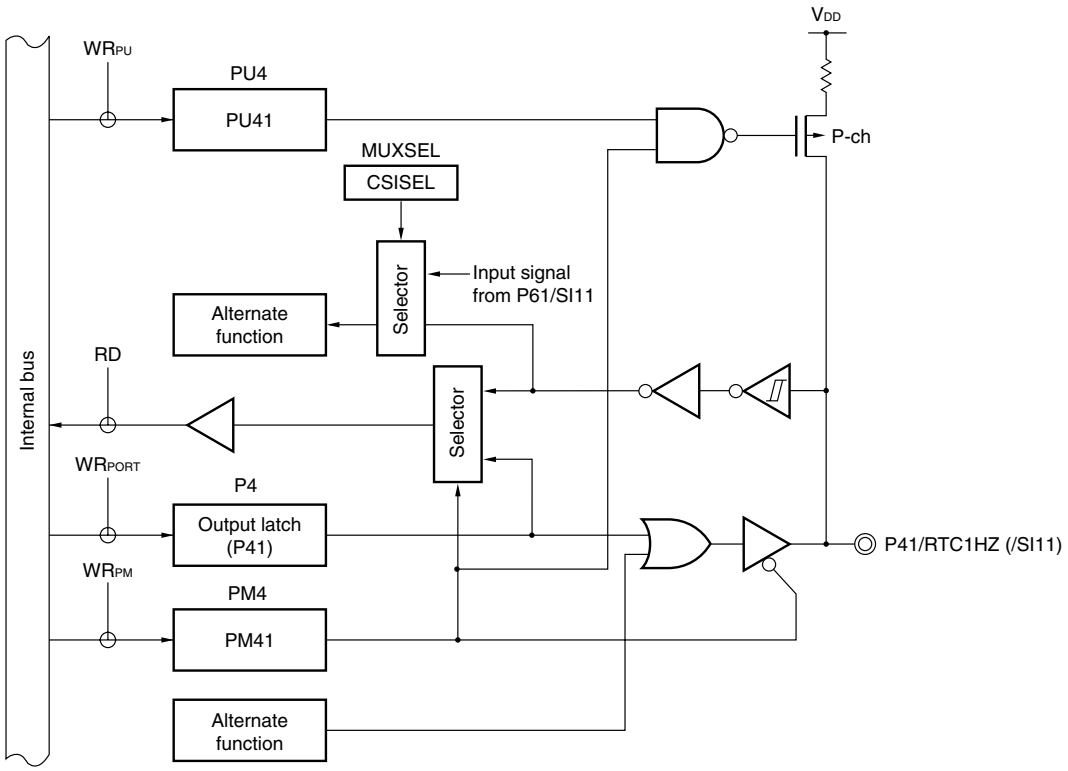
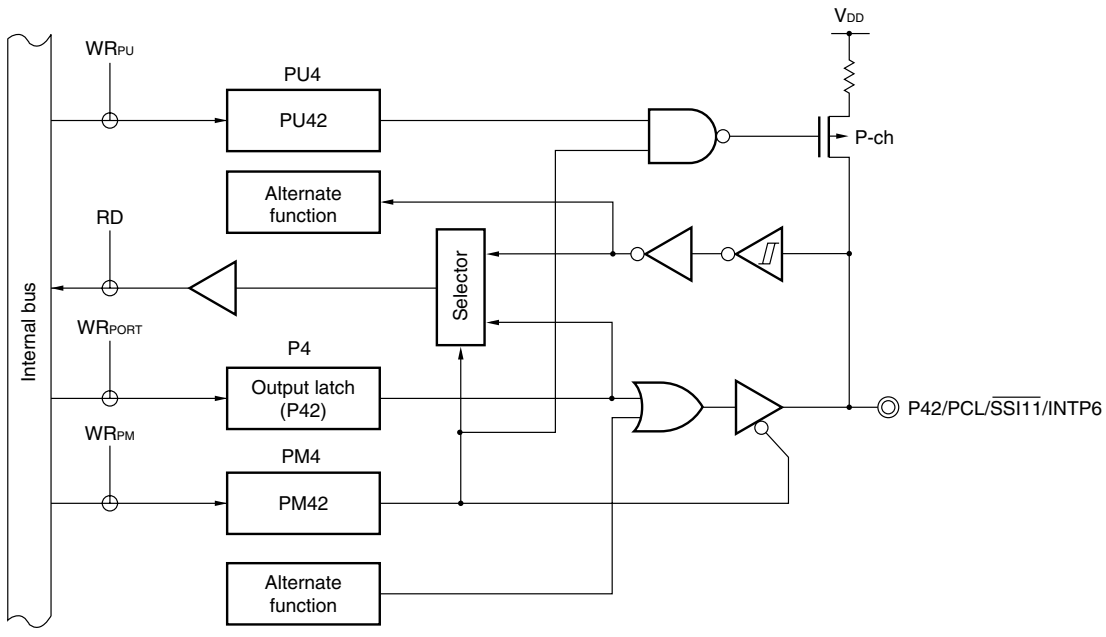



**Figure 4-21. Block Diagram of P42**




P4: Port register 4
PU4: Pull-up resistor option register 4
PM4: Port mode register 4
MUXSEL: Port alternate switch control register
RD: Read signal
WR××: Write signal

### 4.2.6 Port 6

<R>

| 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | |
|---|---|---|---|---|
| 16 Pins | 20, 25, 32 Pins | 30 Pins | 40 Pins | 44, 48 Pins |
| P60/SCLA0/TxD6 | P60/SCLA0/TxD6 | P60/SCLA0/INTP11 | P60/SCLA0/$\overline{\text{SCK11}}$/INTP11 | P60/SCLA0/$\overline{\text{SCK11}}$/INTP11 |
| P61/SDAA0/RxD6 | P61/SDAA0/RxD6 | P61/SDAA0/INTP10 | P61/SDAA0/SI11/INTP10 | P61/SDAA0/SI11/INTP10 |
| – | – | – | P62/SO11/INTP9 | P62/SO11/INTP9 |
| – | – | – | – | P63/INTP8 |

Port 6 is an I/O port with an output latch. Port 6 can be set to the input mode or output mode in 1-bit units using port mode register 6 (PM6). When the P60 to P63 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 6 (PU6).

Input to the P60 and P61 pins can be specified through a normal input buffer or an SMBus input buffer in 1-bit units, using port input mode register 6 (PIM6).

Output from the P60 to P63 pins can be specified as normal CMOS output or N-ch open-drain output ($V_{DD}$ tolerance) in 1-bit units, using port output mode register 6 (POM6).

This port can also be used for serial interface data I/O, clock I/O, and external clock input.

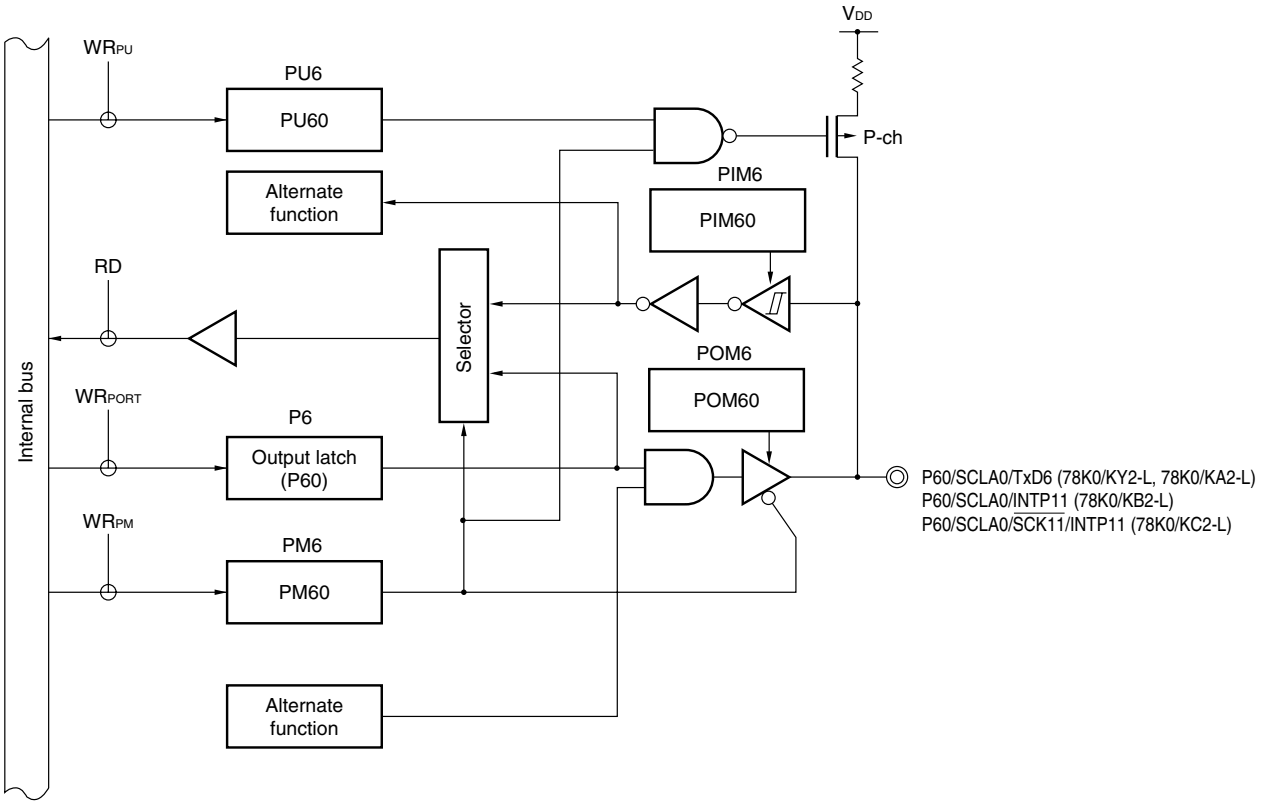
Reset signal generation sets port 6 to input mode.

**Cautions 1. To use P60/SCLA0/$\overline{\text{SCK11}}$/INTP11 and P62/SO11/INTP9 of 78K0/KC2-L as general-purpose ports, set serial operation mode register 11 (CSIM11) and serial clock selection register 11 (CSIC11) to the default status (00H).**

**2. To use P60/SCLA0/TxD6 of 78K0/KY2-L and 78K0/KA2-L as general-purpose port, clear bit 0 (TXDLV6) of asynchronous serial interface control register 6 (ASICL6) to 0 (normal output of TxD6).**

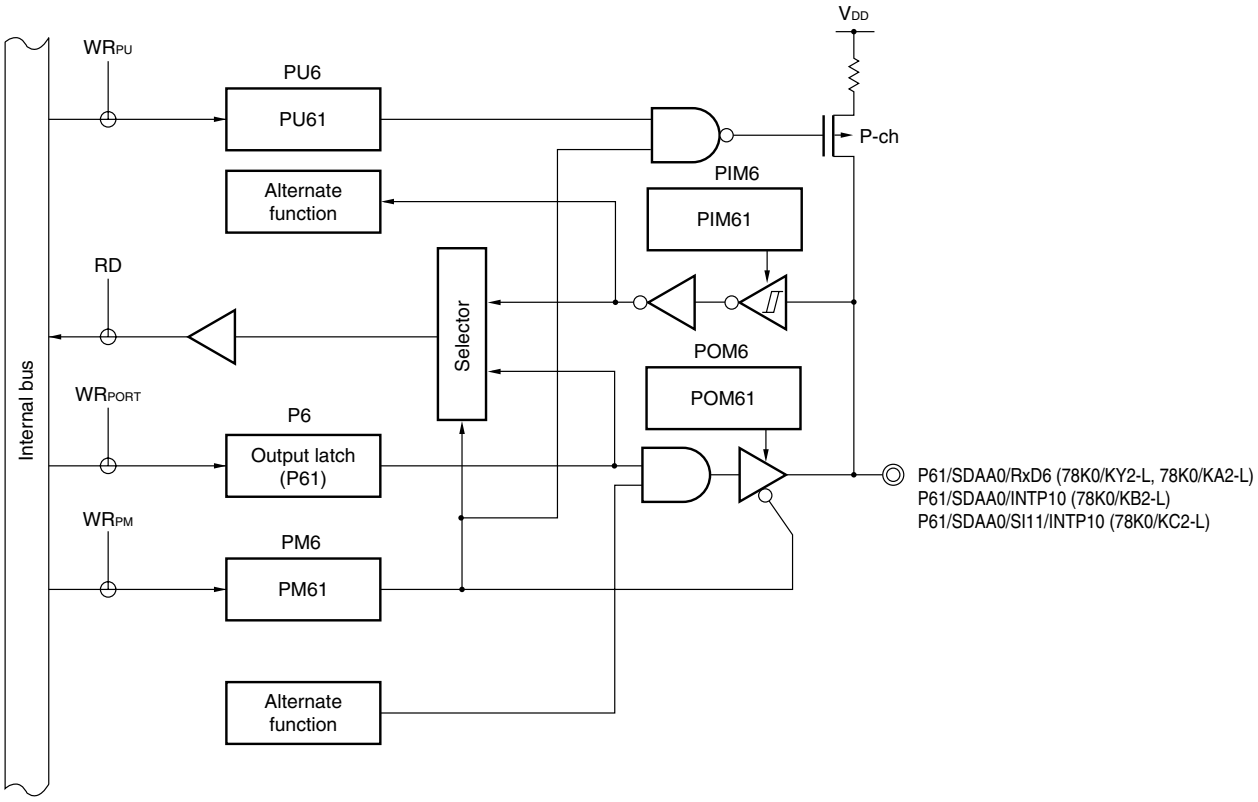
Figures 4-22 to 4-25 show block diagrams of port 6.

**Figure 4-22. Block Diagram of P60**



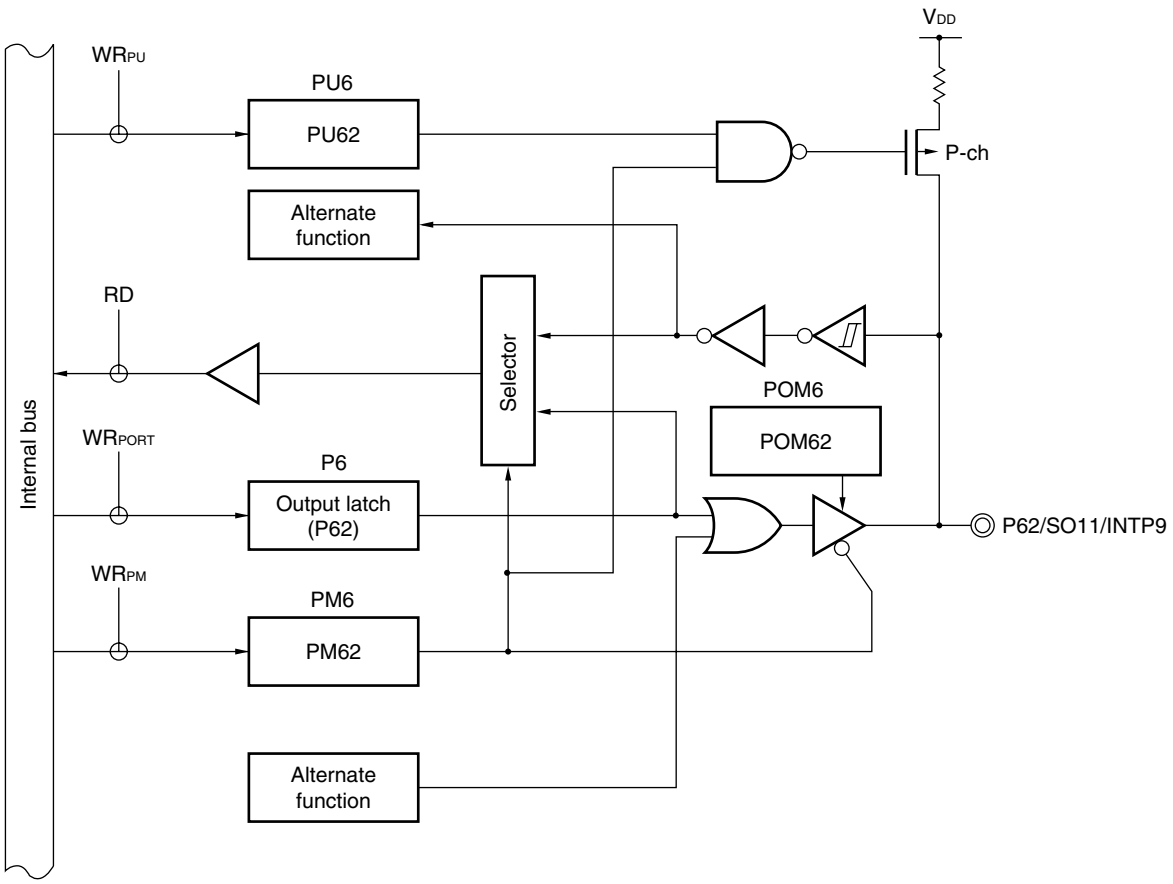

P6: Port register 6
PU6: Pull-up resistor option register 6
PM6: Port mode register 6
PIM6: Port input mode register 6
POM6: Port output mode register 6
RD: Read signal
WR××: Write signal

**Figure 4-23. Block Diagram of P61**



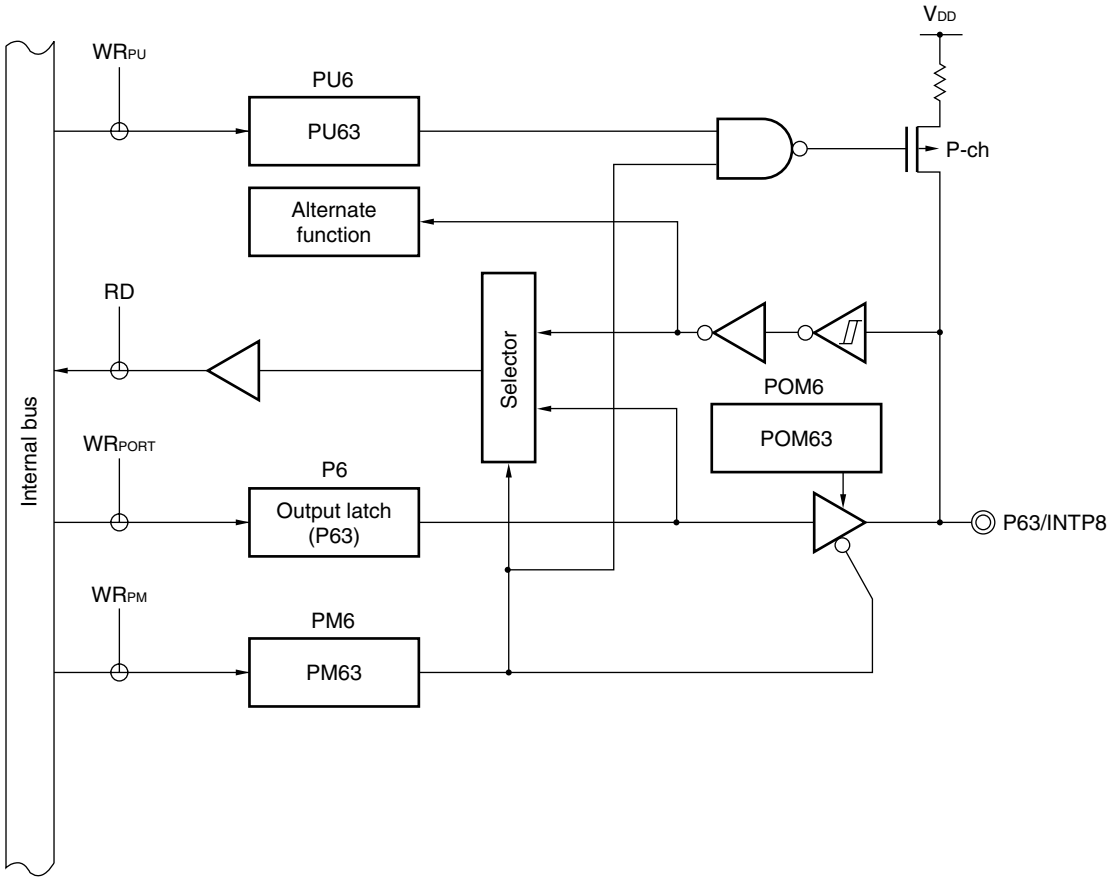

P6: Port register 6
PU6: Pull-up resistor option register 6
PM6: Port mode register 6
PIM6: Port input mode register 6
POM6: Port output mode register 6
RD: Read signal
WR××: Write signal

**Figure 4-24. Block Diagram of P62**




P6: Port register 6
PU6: Pull-up resistor option register 6
PM6: Port mode register 6
POM6: Port output mode register 6
RD: Read signal
WR××: Write signal

**Figure 4-25. Block Diagram of P63**




P6: Port register 6
PU6: Pull-up resistor option register 6
PM6: Port mode register 6
POM6: Port output mode register 6
RD: Read signal
WR××: Write signal

### 4.2.7 Port 7

<R>

| 78K0/KY2-L ($\mu$PD78F057x) | 78K0/KA2-L ($\mu$PD78F056x) | | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | |
|---|---|---|---|---|---|
| 16 Pins | 20, 25 Pins | 32 Pins | 30 Pins | 40, 44 Pins | 48 Pins |
| – | – | P70/ANI8 | – | P70/KR0 | P70/KR0 |
| – | – | P71/ANI9 | – | P71/KR1 | P71/KR1 |
| – | – | P72/ANI10 | – | P72/KR2 | P72/KR2 |
| – | – | – | – | P73/KR3 | P73/KR3 |
| – | – | – | – | – | P74/KR4 |
| – | – | – | – | – | P75/KR5 |

Port 7 is an I/O port with an output latch. Port 7 can be set to the input mode or output mode in 1-bit units using port mode register 7 (PM7). When the P70 to P75 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 7 (PU7) in 78K0/KC2-L.

<R> This port can also be used for A/D converter analog input and key interrupt input pins.

When using P70/ANI8 to P72/ANI10, set the registers according to the pin function to be used (refer to **Tables 4-13**).

<R> **Table 4-13. Setting Functions of P70/ANI8 to P72/ANI10 Pins**

| ADPC1 register | PM7 register | ADS register (n = 8 to 10) | P70/ANI8 to P72/ANI10 Pins |
|---|---|---|---|
| Digital I/O selection | Input mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital input |
| | Output mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital output |
| Analog input selection | Input mode | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | Output mode | – | Setting prohibited |

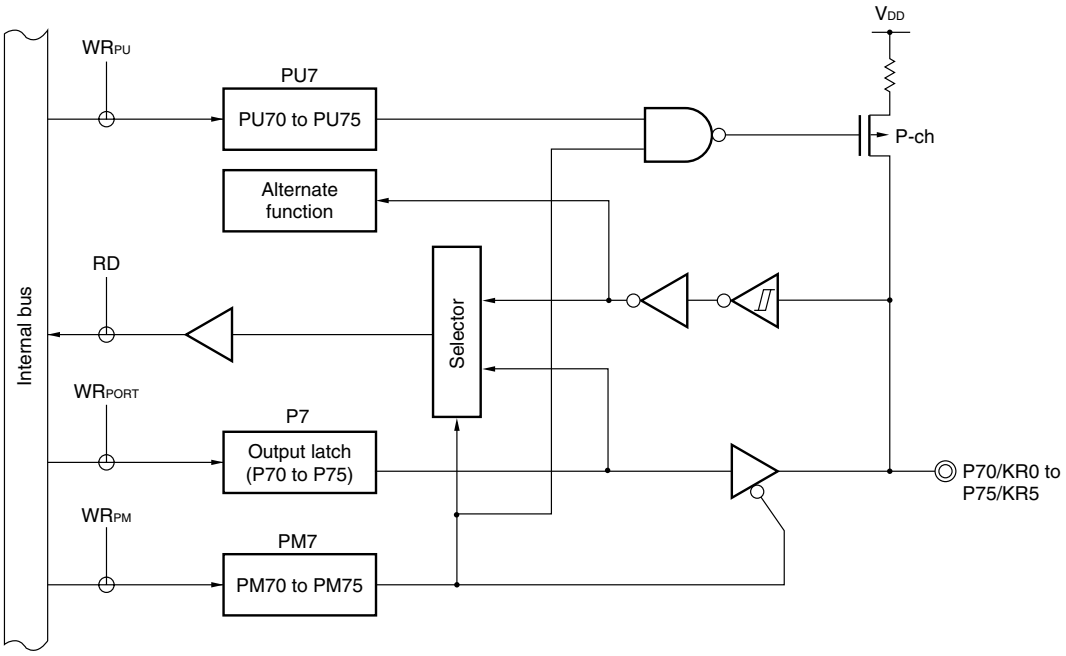
**Remark** ADPC1: A/D port configuration register 1
PM7: Port mode register 7
ADS: Analog input channel specification register

Reset signal generation sets port 7 to input mode.

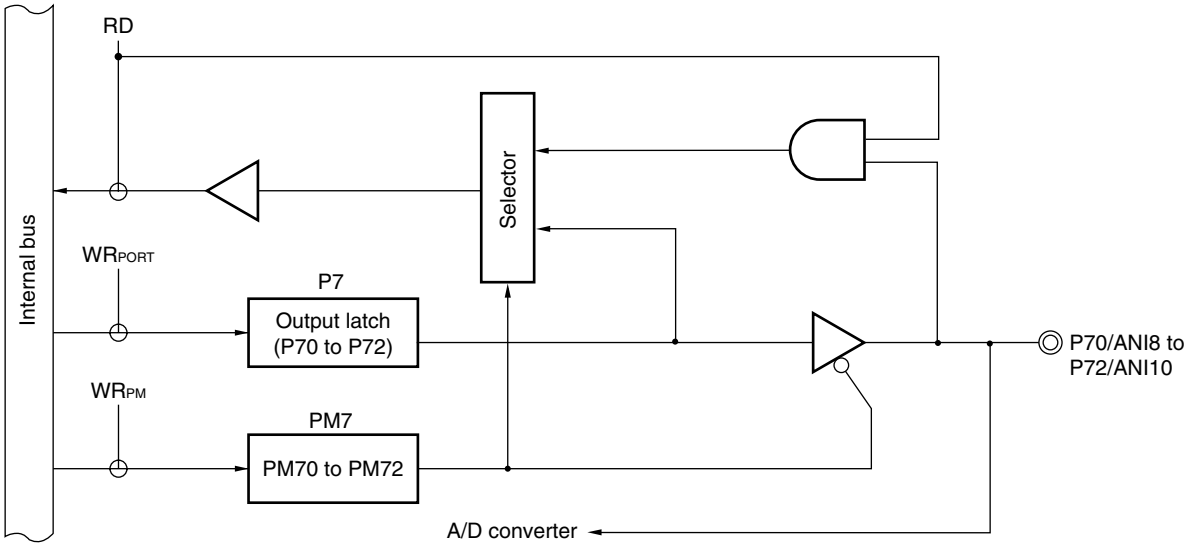
Figure 4-26 shows a block diagram of port 7.

**Figure 4-26. Block Diagram of P70 to P75**

**(1) 78K0/KC2-L**




<R> **(2) 78K0/KA2-L (32-pin products)**




P7: Port register 7
PU7: Pull-up resistor option register 7
PM7: Port mode register 7
RD: Read signal
WR××: Write signal

### 4.2.8 Port 12

<R>

| 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | |
|---|---|---|---|---|---|---|
| 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44, 48 Pins |
| – | – | – | – | P120/EXLVI/ INTP0 | P120/EXLVI/ INTP0 | P120/EXLVI/ INTP0(/SO11) |
| P121/X1/ TOOLC0 | P121/X1/ TOOLC0 | P121/X1/ TOOLC0 (/TI000) (/INTP0) | P121/X1/ TOOLC0 (/TI000) (/INTP0) | P121/X1/ TOOLC0 | P121/X1/ TOOLC0 | P121/X1/ TOOLC0 |
| P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 | P122/X2/ EXCLK/ TOOLD0 |
| – | – | – | – | – | P123/XT1 | P123/XT1 |
| – | – | – | – | – | P124/XT2/ EXCLKS | P124/XT2/ EXCLKS |
| P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ (/TI000) (/INTP0) | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ | P125/$\overline{\text{RESET}}$ |

**Remark** Functions in parentheses ( ) can be assigned by setting the port alternate switch control register (MUXSEL).

P120 functions as an I/O port with an output latch. P120 can be set to the input mode or output mode in 1-bit units using port mode register 12 (PM12).

P121 to P125 function as an Input port.

When used as an input port for P120 and P125, use of an on-chip pull-up resistor can be specified by pull-up resistor option register 12 (PU12).

This port can also be used as pins for external interrupt request input, potential input for external low-voltage detection, connecting resonator for main system clock, connecting resonator for subsystem clock, external clock input for main system clock, external clock input for subsystem clock, external reset input, and clock input and data I/O for flash memory programmer/on-chip debugger.

<R> The timer input and external interrupt request input can be assigned to P121 of the 78K0/KA2-L (25-pin products) and P121 and P125 of the 78K0/KA2-L (32-pin products) by setting the port alternate switch control register (MUXSEL). The data output of the serial interface can be assigned to P120 of the 78K0/KC2-L (44-pin and 48-pin products) by setting the port alternate switch control register (MUXSEL).

Set bit 5 (RSTM) of the reset pin mode register (RSTMASK) to 1 when using P125/$\overline{\text{RESET}}$ as an input port, and clear RSTM to 0 when using P125/$\overline{\text{RESET}}$ as an external reset input.

Reset signal generation sets port 12 to input mode.

Figures 4-27 to 4-29 show block diagrams of port 12.

**Cautions 1. When using the P121 to P124 pins to connect a resonator for the main system clock (X1, X2) or subsystem clock (XT1, XT2), or to input an external clock for the main system clock (EXCLK) or subsystem clock (EXCLKS), the X1 oscillation mode, XT1 oscillation mode, or external clock input mode must be set by using the clock operation mode select register (OSCCTL) (for details, refer to 5.3 (1) Clock operation mode select register (OSCCTL) and (3) Setting of operation mode for subsystem clock pin). The reset value of OSCCTL is 00H (all of the P121 to P124 pins are Input port pins).**

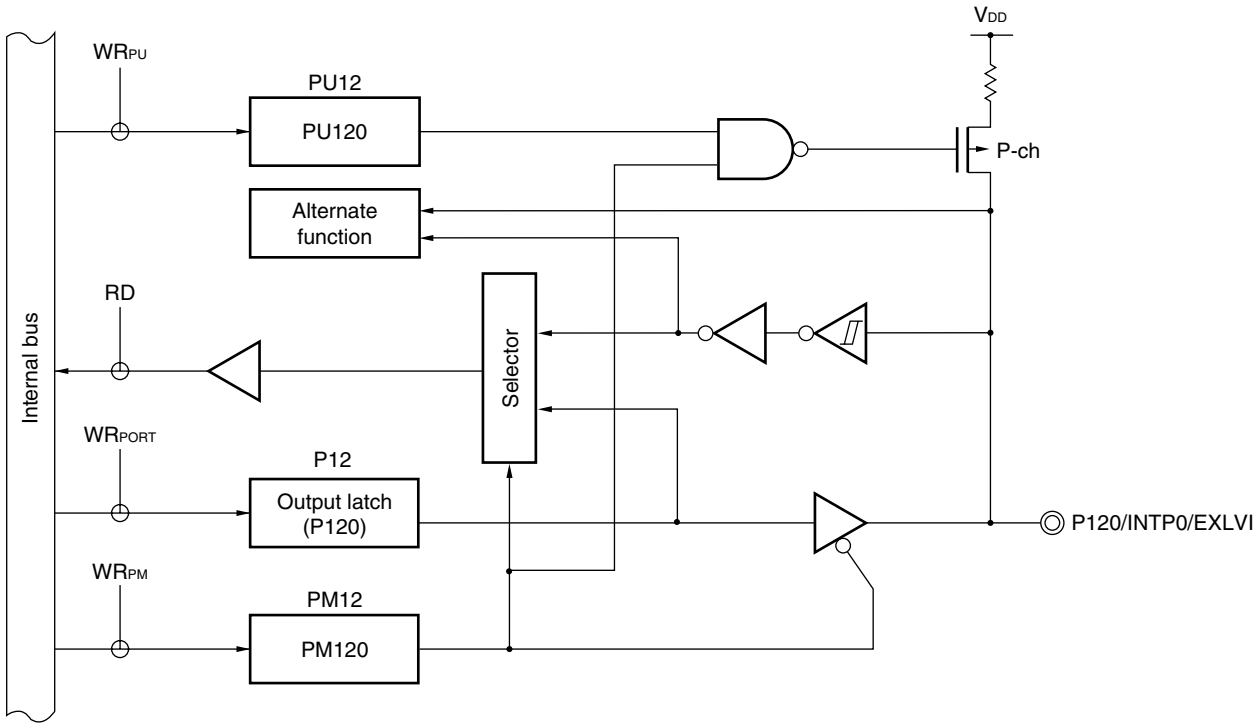
**2. $\overline{\text{RESET}}$/P125 is set in an external reset input after a reset release.**

<R> **3. Because $\overline{\text{RESET}}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

<R> **4. If using P120 to input the potential for an external low-voltage detector, connect P120 to the measured voltage source through a resistor. Do not apply a voltage of $V_{DD}$ or more to P120.**

**Figure 4-27. Block Diagram of P120 (1/2)**

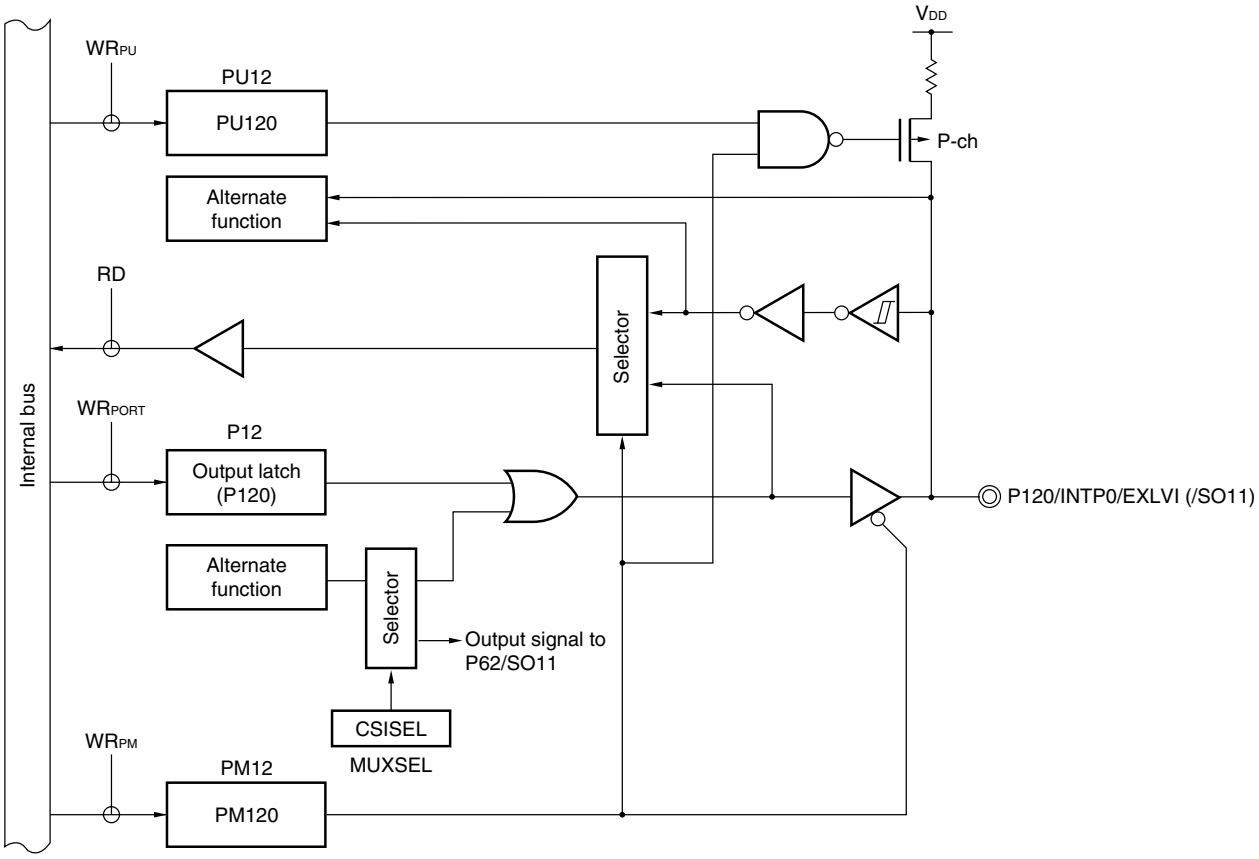
**(1) 78K0/KB2-L**




P12: Port register 12
PU12: Pull-up resistor option register 12
PM12: Port mode register 12
RD: Read signal
WR××: Write signal

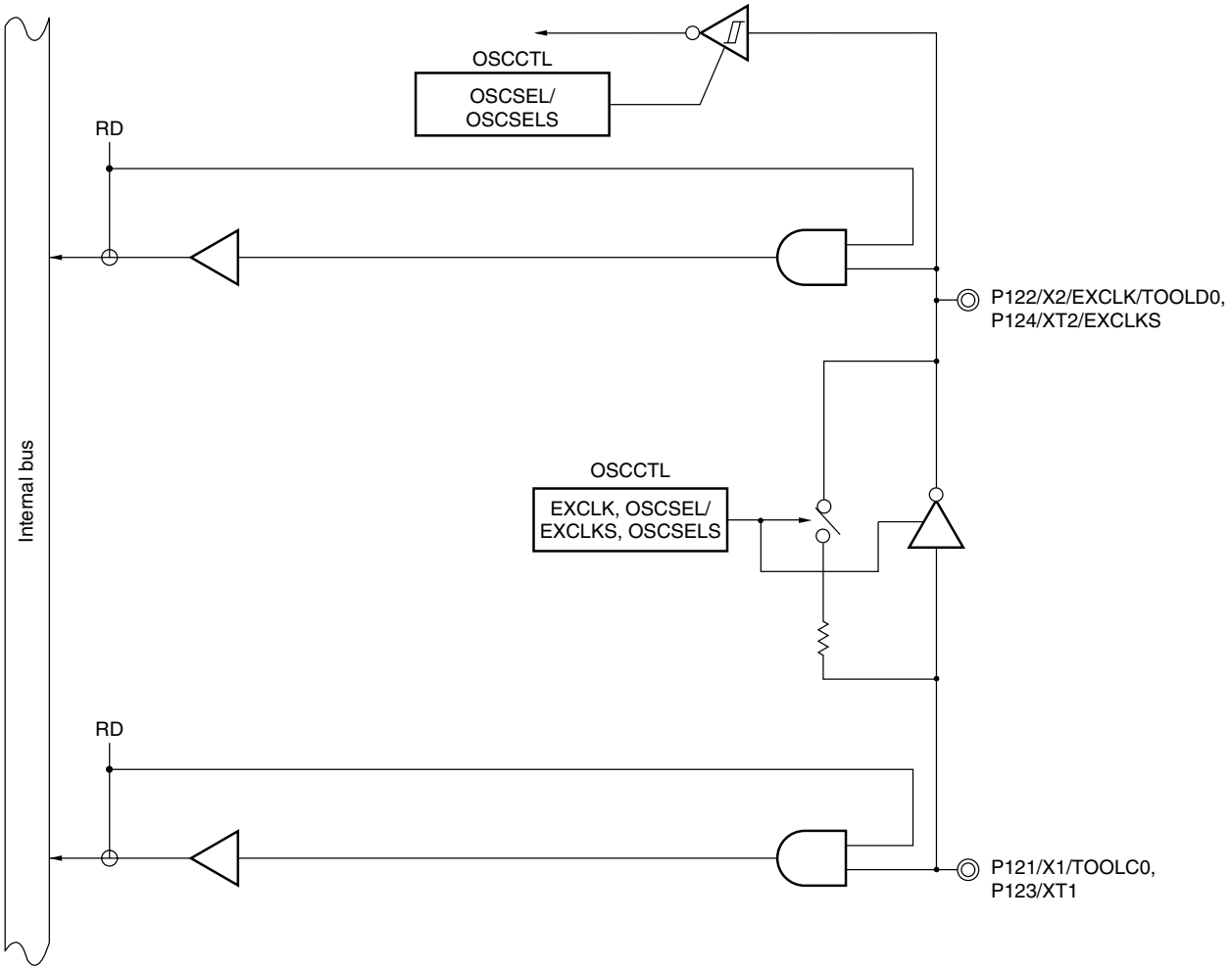
**Figure 4-27. Block Diagram of P120 (2/2)**

**(2) 78K0/KC2-L**



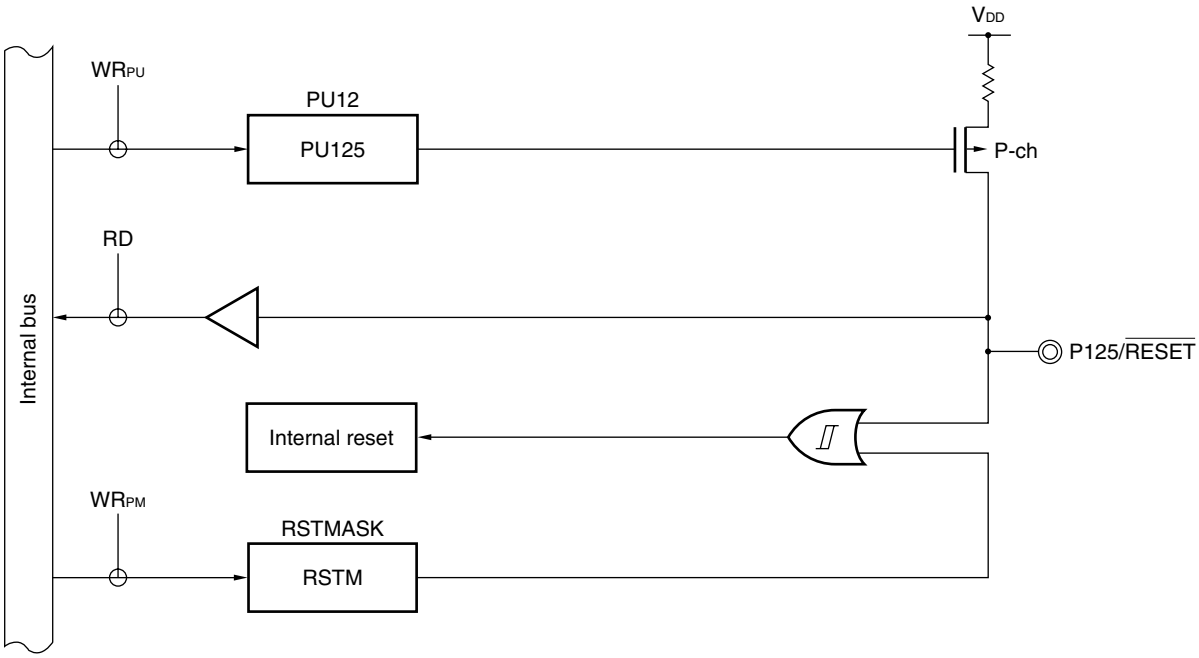

P12: Port register 12
PU12: Pull-up resistor option register 12
PM12: Port mode register 12
MUXSEL: Port alternate switch control register
RD: Read signal
WR××: Write signal

**Figure 4-28. Block Diagram of P121 to P124**




OSCCTL: Clock operation mode select register
RD: Read signal
WR××: Write signal

**Figure 4-29. Block Diagram of P125**




PU12: Pull-up resistor option register 12
RD: Read signal
WR××: Write signal
RSTMASK: Reset pin mode register

<R> **Caution Because $\overline{\text{RESET}}$/P125 is set in the external reset input immediately after release of reset, if a reset signal is generated during low level input, the reset status continues until the input rises to the high level.**

**Remark** After reset, the external reset function and the pull-up resistor are enabled (RSTM = 0, PU125 = 1). Set RSTM bit to 1 when using as a port function.

## 4.3 Registers Controlling Port Function

Port functions are controlled by the following eight types of registers.

- Port mode registers (PMxx)
- Port registers (Pxx)
- Pull-up resistor option registers (PUxx)
- Port input mode register 6 (PIM6)
- Port output mode register 6 (POM6)
- Reset pin mode register (RSTMASK)
- A/D port configuration registers 0, 1 (ADPC0, ADPC1)
- Port alternate switch control register (MUXSEL)

### (1) Port mode registers (PMxx)

These registers specify input or output mode for the port in 1-bit units.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

When port pins are used as alternate-function pins, set the port mode register by referencing **4.5 Settings of Port Mode Register and Output Latch When Using Alternate Function**.

**Figure 4-30. Format of Port Mode Register (78K0/KY2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | 1 | PM01 | PM00 | FF20H | FFH | R/W |
| PM2 | 1 | 1 | 1 | 1 | PM23Note | PM22Note | PM21Note | PM20Note | FF22H | FFH | R/W |
| PM3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM30 | FF23H | FFH | R/W |
| PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | FF26H | FFH | R/W |

| PMmn | Pmn pin I/O mode selection (m = 0, 2, 3, 6; n = 0 to 3) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Note** If this pin is set as an analog input by using the ADPC0 register, be sure to set it to input mode.

**Caution Be sure to set bits 2 to 7 of PM0, bits 4 to 7 of PM2, bits 1 to 7 of PM3, bits 2 to 7 of PM6 to 1.**

**Figure 4-31. Format of Port Mode Register (78K0/KA2-L (20-pin products))**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | 1 | PM01 | PM00 | FF20H | FFH | R/W |
| PM2 | 1 | 1 | PM25[Note] | PM24[Note] | PM23[Note] | PM22[Note] | PM21[Note] | PM20[Note] | FF22H | FFH | R/W |
| PM3 | 1 | 1 | 1 | 1 | 1 | PM32 | PM31 | PM30 | FF23H | FFH | R/W |
| PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | FF26H | FFH | R/W |

| PMmn | Pmn pin I/O mode selection<br>(m = 0, 2, 3, 6; n = 0 to 5) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Note** If this pin is set as an analog input by using the ADPC0 register, be sure to set it to input mode.

**Caution Be sure to set bits 2 to 7 of PM0, bits 6, 7 of PM2, bits 3 to 7 of PM3, bits 2 to 7 of PM6 to 1.**

<R>

**Figure 4-32. Format of Port Mode Register (78K0/KA2-L (25-pin and 32-pin products)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | PM02 | PM01 Note 4 | PM00 Note 1 | FF20H | FFH | R/W |
| PM2 | PM27 Notes 2, 3 | PM26 Note 3 | PM25 Note 3 | PM24 Note 3 | PM23 Note 3 | PM22 Note 3 | PM21 Note 3 | PM20 Note 3 | FF22H | FFH | R/W |
| PM3 | PM37 | PM36 | PM35 | PM34 | PM33 | PM32 | PM31 | PM30 Notes 4, 5 | FF23H | FFH | R/W |
| PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | FF26H | FFH | R/W |
| PM7 Note 2 | 1 | 1 | 1 | 1 | 1 | PM72 Notes 2, 3 | PM71 Notes 2, 3 | PM70 Notes 2, 3 | FF27H | FFH | R/W |

| PMmn | Pmn pin I/O mode selection<br>(m = 0, 2, 3, 6, 7; n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** If this pin is set as an analog input by using the ADPC0 and ADPC1 registers, be sure to set it to input mode.

**4.** For 25-pin products, clear it to 0.

**5.** For 32-pin products, clear it to 0.

**Cautions 1. For 25-pin products, be sure to set bits 1 and 3 to 7 of PM0, bit 7 of PM2, and bits 2 to 7 of PM6 to 1. Set P30 and P01 to output mode (PM30 = PM01 = 0) by using software after release of reset.**

**2. For 32-pin products, be sure to set bits 0 and 3 to 7 of PM0, bits 2 to 7 of PM6, and bits 3 to 7 of PM7 to 1. Set P30 to output mode (PM30 = 0) by using software after release of reset.**

**Figure 4-33. Format of Port Mode Register (78K0/KB2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | 1 | PM01 | PM00 | FF20H | FFH | R/W |
| PM1 | PM17 | PM16 | PM15 | PM14 | PM13 | PM12Note | PM11Note | PM10Note | FF21H | FFH | R/W |
| PM2 | 1 | 1 | 1 | 1 | PM23Note | PM22Note | PM21Note | PM20Note | FF22H | FFH | R/W |
| PM3 | 1 | 1 | 1 | 1 | PM33 | PM32 | PM31 | PM30 | FF23H | FFH | R/W |
| PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 | FF26H | FFH | R/W |
| PM12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM120 | FF2CH | FFH | R/W |

| PMmn | Pmn pin I/O mode selection<br>(m = 0 to 3, 6, 12; n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Note** If this pin is set as an analog input by using the ADPC1 or ADPC0 register, be sure to set it to input mode.

**Caution Be sure to set bits 2 to 7 of PM0, bits 4 to 7 of PM2, bits 4 to 7 of PM3, bits 2 to 7 of PM6, bits 1 to 7 of PM12 to 1.**

<R> **Figure 4-34. Format of Port Mode Register (78K0/KC2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | PM02 Note 1 | PM01 | PM00 | FF20H | FFH | R/W |
| PM1 | PM17 | PM16 | PM15 | PM14 | PM13 | PM12 Note 2 | PM11 Note 2 | PM10 Note 2 | FF21H | FFH | R/W |
| PM2 | PM27 Note 2, 3 | PM26 Note 2 | PM25 Note 2 | PM24 Note 2 | PM23 Note 2 | PM22 Note 2 | PM21 Note 2 | PM20 Note 2 | FF22H | FFH | R/W |
| PM3 | 1 | 1 | 1 | 1 | PM33 | PM32 | PM31 | PM30 | FF23H | FFH | R/W |
| PM4 Note 3 | 1 | 1 | 1 | 1 | 1 | PM42 Note 1 | PM41 Note 3 | PM40 Note 3 | FF24H | FFH | R/W |
| PM6 | 1 | 1 | 1 | 1 | PM63 Note 3 | PM62 | PM61 | PM60 | FF26H | FFH | R/W |
| PM7 | 1 | 1 | PM75 Note 1 | PM74 Note 1 | PM73 | PM72 | PM71 | PM70 | FF27H | FFH | R/W |
| PM12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM120 | FF2CH | FFH | R/W |

| PMmn | Pmn pin I/O mode selection<br>(m = 0 to 4, 6, 7, 12; n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Notes 1.** 48-pin products only

**2.** If this pin is set as an analog input by using the ADPC1 or ADPC0 register, be sure to set it to input mode.

**3.** For 40-pin products, clear it to 0.

**Cautions 1. For 40-pin products, be sure to set bits 2 to 7 of PM0, bit 7 of PM2, bits 4 to 7 of PM3, bits 2 to 7 of PM4, bits 3 to 7 of PM6, bits 4 to 7 of PM7, and bits 1 to 7 of PM12 to 1. Set P40, P41, and P63 to output mode (PM40 = PM41 = PM63 = 0) by using software after release of reset.**

**2. For the 44-pin products, be sure to set bits 2 to 7 of PM0, bits 4 to 7 of PM3, bits 2 to 7 of PM4, bits 4 to 7 of PM6, bits 4 to 7 of PM7, and bits 1 to 7 of PM12 to "1".**

**3. For the 48-pin products, be sure to set bits 3 to 7 of PM0, bits 4 to 7 of PM3, bits 3 to 7 of PM4, bits 4 to 7 of PM6, bits 6 and 7 of PM7, and bits 1 to 7 of PM12 to "1".**

**(2) Port registers (Pxx)**

These registers write the data that is output from the chip when data is output from a port.

If the data is read in the input mode, the pin level is read. If it is read in the output mode, the output latch value is read.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Figure 4-35. Format of Port Register (78K0/KY2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P0 | 0 | 0 | 0 | 0 | 0 | 0 | P01 | P00 | FF00H | 00H (output latch) | R/W |
| P2 | 0 | 0 | 0 | 0 | P23Note 1 | P22Note 1 | P21Note 1 | P20Note 1 | FF02H | 00H (output latch) | R/W |
| P3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | P30 | FF03H | 00H (output latch) | R/W |
| P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | FF06H | 00H (output latch) | R/W |
| P12 | 0 | 0 | P125 | 0 | 0 | P122Note 2 | P121Note 2 | 0 | FF0CH | 00H | R |

| Pmn | m = 0, 2, 3, 6, 12; n = 0 to 3, 5 | |
|---|---|---|
| | Output data control (in output mode) | Input data read (in input mode) |
| 0 | Output 0 | Input low level |
| 1 | Output 1 | Input high level |

**Notes 1.** If this pin is set as an analog input and to input mode, do not access the output latch.

**2.** “0” is always read from the output latch of the pin in the X1 oscillation mode or external clock input mode.

**Figure 4-36. Format of Port Register (78K0/KA2-L (20-pin products))**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P0 | 0 | 0 | 0 | 0 | 0 | 0 | P01 | P00 | FF00H | 00H (output latch) | R/W |
| P2 | 0 | 0 | P25Note 1 | P24Note 1 | P23Note 1 | P22Note 1 | P21Note 1 | P20Note 1 | FF02H | 00H (output latch) | R/W |
| P3 | 0 | 0 | 0 | 0 | 0 | P32 | P31 | P30 | FF03H | 00H (output latch) | R/W |
| P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | FF06H | 00H (output latch) | R/W |
| P12 | 0 | 0 | P125 | 0 | 0 | P122Note 2 | P121Note 2 | 0 | FF0CH | 00H | R |

| Pmn | m = 0, 2, 3, 6, 12; n = 0 to 5 | |
|---|---|---|
| | Output data control (in output mode) | Input data read (in input mode) |
| 0 | Output 0 | Input low level |
| 1 | Output 1 | Input high level |

**Notes 1.** If this pin is set as an analog input and to input mode, do not access the output latch.

**2.** “0” is always read from the output latch of the pin in the X1 oscillation mode or external clock input mode.

<R> **Figure 4-37. Format of Port Register (78K0/KA2-L (25-pin and 32-pin products))**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P0 | 0 | 0 | 0 | 0 | 0 | P02 | P01 Note 2 | P00 Note 1 | FF00H | 00H (output latch) | R/W |
| P2 | P27 Note 2, 3 | P26 Note 3 | P25 Note 3 | P24 Note 3 | P23 Note 3 | P22 Note 3 | P21 Note 3 | P20 Note 3 | FF02H | 00H (output latch) | R/W |
| P3 | P37 | P36 | P35 | P34 | P33 | P32 | P31 | 0 | FF00H | 00H (output latch) | R/W |
| P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | FF00H | 00H (output latch) | R/W |
| P7 Note 2 | 0 | 0 | 0 | 0 | 0 | P72 Note 2, 3 | P71 Note 2, 3 | P70 Note 2, 3 | FF07H | 00H (output latch) | R/W |
| P12 | 0 | 0 | P125 | 0 | 0 | P122 Note 4 | P121 Note 4 | 0 | FF0CH | 00H | R |

| Pmn | m = 0, 2, 3, 6, 7, 12; n = 0 to 7 | |
|---|---|---|
| | Output data control (in output mode) | Input data read (in input mode) |
| 0 | Output 0 | Input low level |
| 1 | Output 1 | Input high level |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** If this pin is set as an analog input and to input mode, do not access the output latch.

**4.** “0” is always read from the output latch of the pin in the X1 oscillation mode or external clock input mode.

**Figure 4-38. Format of Port Register (78K0/KB2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P0 | 0 | 0 | 0 | 0 | 0 | 0 | P01 | P00 | FF00H | 00H (output latch) | R/W |
| P1 | P17 | P16 | P15 | P14 | P13 | P12Note 1 | P11Note 1 | P10Note 1 | FF01H | 00H (output latch) | R/W |
| P2 | 0 | 0 | 0 | 0 | P23Note 1 | P22Note 1 | P21Note 1 | P20Note 1 | FF02H | 00H (output latch) | R/W |
| P3 | 0 | 0 | 0 | 0 | P33 | P32 | P31 | P30 | FF03H | 00H (output latch) | R/W |
| P6 | 0 | 0 | 0 | 0 | 0 | 0 | P61 | P60 | FF06H | 00H (output latch) | R/W |
| P12 | 0 | 0 | P125 | 0 | 0 | P122Note 2 | P121Note 2 | P120 | FF0CH | 00H (output latch) | R/WNote 3 |

| Pmn | m = 0 to 3, 6, 12; n = 0 to 7 | |
|---|---|---|
| | Output data control (in output mode) | Input data read (in input mode) |
| 0 | Output 0 | Input low level |
| 1 | Output 1 | Input high level |

**Notes 1.** If this pin is set as an analog input and to input mode, do not access the output latch.

**2.** “0” is always read from the output latch of the pin in the X1 oscillation mode or external clock input mode.

**3.** P121, P122, and P125 are read-only.

<R>

**Figure 4-39. Format of Port Register (78K0/KC2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P0 | 0 | 0 | 0 | 0 | 0 | P02Note 1 | P01 | P00 | FF00H | 00H (output latch) | R/W |
| P1 | P17 | P16 | P15 | P14 | P13 | P12Note 2 | P11Note 2 | P10Note 2 | FF01H | 00H (output latch) | R/W |
| P2 | P27Note 2, 3 | P26Note 2 | P25Note 2 | P24Note 2 | P23Note 2 | P22Note 2 | P21Note 2 | P20Note 2 | FF02H | 00H (output latch) | R/W |
| P3 | 0 | 0 | 0 | 0 | P33 | P32 | P31 | P30 | FF03H | 00H (output latch) | R/W |
| P4Note 3 | 0 | 0 | 0 | 0 | 0 | P42Note 1 | P41Note 3 | P40Note 3 | FF04H | 00H (output latch) | R/W |
| P6 | 0 | 0 | 0 | 0 | P63Note 3 | P62 | P61 | P60 | FF06H | 00H (output latch) | R/W |
| P7 | 0 | 0 | P75Note 1 | P74Note 1 | P73 | P72 | P71 | P70 | FF07H | 00H (output latch) | R/W |
| P12 | 0 | 0 | P125 | P124Note 4 | P123Note 4 | P122Note 4 | P121Note 4 | P120 | FF0CH | 00H (output latch) | R/WNote 5 |

| Pmn | m = 0 to 4, 6, 7, 12; n = 0 to 7 | |
|---|---|---|
| | Output data control (in output mode) | Input data read (in input mode) |
| 0 | Output 0 | Input low level |
| 1 | Output 1 | Input high level |

**Notes 1.** 48-pin products only

**2.** If this pin is set as an analog input and to input mode, do not access the output latch.

**3.** 44-pin and 48-pin products y

**4.** “0” is always read from the output latch of the pin in the X1 oscillation mode, XT1 oscillation mode, or external clock input mode.

**5.** P121 to P125 are read-only.

**(3) Pull-up resistor option registers (PUxx)**

These registers specify whether the on-chip pull-up resistors are to be used or not. On-chip pull-up resistors can be used in 1-bit units only for the bits set to input mode of the pins to which the use of an on-chip pull-up resistor has been specified in these registers. On-chip pull-up resistors cannot be connected to bits set to output mode and bits used as alternate-function output pins, regardless of the settings of these registers.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H (sets only PU12 to 20H).

**Figure 4-40. Format of Pull-up Resistor Option Register (78K0/KY2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PU0 | 0 | 0 | 0 | 0 | 0 | 0 | PU01 | PU00 | FF30H | 00H | R/W |
| PU3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | PU30 | FF33H | 00H | R/W |
| PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | FF36H | 00H | R/W |
| PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | 0 | FF3CH | 20H | R/W |

| PUmn | Pmn pin on-chip pull-up resistor selection (m = 0, 3, 6, 12; n = 0, 1, 5) |
|---|---|
| 0 | On-chip pull-up resistor not connected |
| 1 | On-chip pull-up resistor connected |

**Figure 4-41. Format of Pull-up Resistor Option Register (78K0/KA2-L (20-pin products))**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PU0 | 0 | 0 | 0 | 0 | 0 | 0 | PU01 | PU00 | FF30H | 00H | R/W |
| PU3 | 0 | 0 | 0 | 0 | 0 | PU32 | PU31 | PU30 | FF33H | 00H | R/W |
| PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | FF36H | 00H | R/W |
| PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | 0 | FF3CH | 20H | R/W |

| PUmn | Pmn pin on-chip pull-up resistor selection (m = 0, 3, 6, 12; n = 0 to 2, 5) |
|---|---|
| 0 | On-chip pull-up resistor not connected |
| 1 | On-chip pull-up resistor connected |

**Figure 4-42. Format of Pull-up Resistor Option Register (78K0/KA2-L (25-pin and 32-pin products))**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PU0 | 0 | 0 | 0 | 0 | 0 | PU02 | PU01[Note 2] | PU00[Note 1] | FF30H | 00H | R/W |
| PU3 | PU37 | PU36 | PU35 | PU34 | PU33 | PU32 | PU31 | 0 | FF33H | 00H | R/W |
| PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | FF36H | 00H | R/W |
| PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | 0 | FF3CH | 20H | R/W |

| PUmn | Pmn pin on-chip pull-up resistor selection (m = 0, 3, 6, 12; n = 0 to 7) |
|---|---|
| 0 | On-chip pull-up resistor not connected |
| 1 | On-chip pull-up resistor connected |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**Figure 4-43. Format of Pull-up Resistor Option Register (78K0/KB2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PU0 | 0 | 0 | 0 | 0 | 0 | 0 | PU01 | PU00 | FF30H | 00H | R/W |
| PU1 | PU17 | PU16 | PU15 | PU14 | PU13 | PU12 | PU11 | PU10 | FF31H | 00H | R/W |
| PU3 | 0 | 0 | 0 | 0 | PU33 | PU32 | PU31 | PU30 | FF33H | 00H | R/W |
| PU6 | 0 | 0 | 0 | 0 | 0 | 0 | PU61 | PU60 | FF36H | 00H | R/W |
| PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | PU120 | FF3CH | 20H | R/W |

| PUmn | Pmn pin on-chip pull-up resistor selection (m = 0, 1, 3, 6, 12; n = 0 to 7) |
|---|---|
| 0 | On-chip pull-up resistor not connected |
| 1 | On-chip pull-up resistor connected |

<R>

**Figure 4-44. Format of Pull-up Resistor Option Register (78K0/KC2-L)**

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Address | After reset | R/W |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PU0 | 0 | 0 | 0 | 0 | 0 | PU02 Note 1 | PU01 | PU00 | FF30H | 00H | R/W |
| PU1 | PU17 | PU16 | PU15 | PU14 | PU13 | PU12 | PU11 | PU10 | FF31H | 00H | R/W |
| PU3 | 0 | 0 | 0 | 0 | PU33 | PU32 | PU31 | PU30 | FF33H | 00H | R/W |
| PU4 Note 2 | 0 | 0 | 0 | 0 | 0 | PU42 Note 1 | PU41 Note 2 | PU40 Note 2 | FF34H | 00H | R/W |
| PU6 | 0 | 0 | 0 | 0 | PU63 Note 2 | PU62 | PU61 | PU60 | FF36H | 00H | R/W |
| PU7 | 0 | 0 | PU75 Note 1 | PU74 Note 1 | PU73 | PU72 | PU71 | PU70 | FF37H | 00H | R/W |
| PU12 | 0 | 0 | PU125 | 0 | 0 | 0 | 0 | PU120 | FF3CH | 20H | R/W |

| PUmn | Pmn pin on-chip pull-up resistor selection (m = 0, 1, 3, 4, 6, 7, 12; n = 0 to 7) |
|---|---|
| 0 | On-chip pull-up resistor not connected |
| 1 | On-chip pull-up resistor connected |

**Notes 1.** 48-pin products only

**2.** 44-pin and 48-pin products only

**(4) Port input mode register 6 (PIM6)**

This register sets the input buffer of P60 and P61 in 1-bit units.

When using an input compliant with the SMBus specifications in $I^2C$ communication, set PIM60 and PIM61 to 1.

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

Reset signal generation clears this register to 00H.

**Figure 4-45. Format of Port Input Mode Register 6 (PIM6)**

Address: FF3EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 |

| PIM6n | P6n pin input buffer selection (n = 0, 1) |
|---|---|
| 0 | Normal input (Schmitt) buffer |
| 1 | SMBus input buffer |

**(5) Port output mode register 6 (POM6)**

This register sets the output mode of P60 to P63 in 1-bit units.

During $I^2C$ communication, set POM60 and POM61 to 1.

<R> In the 78K0/KY2-L and 78K0/KA2-L, clear POM60 to 0 when using the P60/TxD6/SCLA0 pin as the data output of serial interface UART6.

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

Reset signal generation clears this register to 00H.

**Figure 4-46. Format of Port Output Mode Register 6 (POM6)**

Address: FF2AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| POM6 | 0 | 0 | 0 | 0 | POM63Note | POM62Note | POM61 | POM60 |

| POM6n | P6n pin output mode selection (n = 0 to 3) |
|---|---|
| 0 | Normal output (CMOS output) mode |
| 1 | N-ch open drain output ($V_{DD}$ tolerance) mode |

**Note** 78K0/KC2-L only

**(6) Reset pin mode register (RSTMASK)**

This register sets the pin function of $\overline{\text{RESET}}$/P125 (external reset input/input-dedicated port).

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

Reset signal generation clears this register to 00H.

**Figure 4-47. Format of Reset Pin Mode Register (RSTMASK)**

Address: FF2DH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RSTMASK | 0 | 0 | RSTM | 0 | 0 | 0 | 0 | 0 |

| RSTM | $\overline{\text{RESET}}$/P125 pin function selection |
|---|---|
| 0 | Using as external reset input ($\overline{\text{RESET}}$) |
| 1 | Using as input-dedicated port (P125) |

**(7) A/D port configuration registers 0, 1[Note] (ADPC0, ADPC1[Note])**

ADPC0 switches the P20/AMP0-/ANI0 to P27/ANI7 pins to digital I/O or analog I/O of port. Each bit of ADPC0 corresponds to a pin of port 2 and can be specified in 1-bit units.

ADPC1[Note] switches the P10/AMP1-/ANI8 to P12/AMP1+/ANI10 or P70/ANI8 to P72/ANI10 pins to digital I/O or analog I/O of port. Each bit of ADPC1 corresponds to a pin of P10 to P12 in port 1 or P70 to P72 in port 7 and can be specified in 1-bit units.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears ADPC0 to 00H, clears ADPC1 of the 78K0/KA2-L (32-pin products) to 00H, and sets ADPC1 of the 78K0/KB2-L and 78K0/KC2-L to 07H

**Note** 78K0/KA2-L (32-pin products), 78K0/KB2-L and 78K0/KC2-L only

**Figure 4-48. Format of A/D Port Configuration Register 0, 1 (ADPC0, ADPC1) (1/2)**

**(a) 78K0/KY2-L**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(b) 78K0/KA2-L (20-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(c) 78K0/KA2-L (25-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(d) 78K0/KA2-L (32-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

**Figure 4-48. Format of A/D Port Configuration Register 0, 1 (ADPC0, ADPC1) (2/2)**

**(e) 78K0/KB2-L**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

**(f) 78K0/KC2-L (40-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

**(g) 78K0/KC2-L (44-pin and 48-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

| ADPCSn | Digital I/O or analog I/O selection (n = 0 to 10) |
|---|---|
| 0 | Analog I/O |
| 1 | Digital I/O |

**Cautions 1. Set the pin set to analog input to the input mode by using port mode register 1, 2, and 7 (PM1, PM2, and PM7).**

**2. If data is written to ADPC0 and ADPC1, a wait cycle is generated. Do not write data to ADPC0 and ADPC1 when the peripheral hardware clock is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(8) Port alternate switch control register (MUXSEL)[Note]**

MUXSEL of 78K0/KA2-L (25-pin products) assigns TOH1, TI51, TI000, and INTP0 pins. By default, INTP0 and TI000 are assigned to P00, while TI51 and TOH1 have no assignment setting.

MUXSEL of 78K0/KA2-L (32-pin products) assigns TOH1, TI000, and INTP0 pins. By default, INTP0 and TI000 and TOH1 have no assignment setting.

MUXSEL of 78K0/KC2-L (44-pin and 48-pin products) assigns the pin function to be used with serial interface CSI11. $\overline{\text{SCK11}}$ is assigned to P60, SI11 to P61, and SO11 to P62 by default.

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

Reset signal generation clears MUXSEL to 00H.

**Figure 4-49. Format of Port Alternate Switch Control Register (MUXSEL) (1/2)**

**(1) 78K0/KA2-L (25-pin products)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | 0 | INTP0SEL0 | 0 | TM00SEL0 | TM5SEL1 | TM5SEL0 | TMHSEL1 | TMHSEL0 |

| INTP0SEL0 | External interrupt request input (INTP0) pin function assignment |
|---|---|
| 0 | Assign INTP0 to the P00 pin as the alternate function. |
| 1 | Assign INTP0 to the P121 pin as the alternate function. |

| TM00SEL0 | 16-bit timer 00 input (TI000) pin function assignment |
|---|---|
| 0 | Assign TI000 to the P00 pin as the alternate function. |
| 1 | Assign TI000 to the P121 pin as the alternate function. |

| TM5SEL1 | TM5SEL0 | 8-bit timer 51 input (TI51) pin function assignment |
|---|---|---|
| 0 | 0 | No TI51 function assignment. |
| 0 | 1 | Assign TI51 to the P34 pin as the alternate function. |
| 1 | 0 | Assign TI51 to the P00 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

| TMHSEL1 | TMHSEL0 | 8-bit timer H1 output (TOH1) pin function assignment |
|---|---|---|
| 0 | 0 | No TOH1 function assignment. |
| 0 | 1 | Assign TOH1 to the P34 pin as the alternate function. |
| 1 | 0 | Assign TOH1 to the P00 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

**Figure 4-49. Format of Port Alternate Switch Control Register (MUXSEL) (2/2)**

**(2) 78K0/KA2-L (32-pin products)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | INTP0SEL1 | INTP0SEL0 | TM00SEL1 | TM00SEL0 | 0 | 0 | 0 | TMHSEL0 |

| INTP0SEL1 | INTP0SEL0 | External interrupt input (INTP0) pin function assignment |
|---|---|---|
| 0 | 0 | No INTP0 function assignment. |
| 0 | 1 | Assign INTP0 to the P121 pin as the alternate function. |
| 1 | 0 | Assign INTP0 to the P125 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

| TM00SEL1 | TM00SEL0 | 16-bit timer 00 input (TI000) pin function assignment |
|---|---|---|
| 0 | 0 | No TI000 function assignment. |
| 0 | 1 | Assign TI000 to the P121 pin as the alternate function. |
| 1 | 0 | Assign TI000 to the P125 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

| TMHSEL0 | 8-bit timer H1 output (TOH1) pin function assignment |
|---|---|
| 0 | No TOH1 function assignment. |
| 1 | Assign TOH1 to the P34 pin as the alternate function. |

**(3) 78K0/KC2-L (44-pin and 48-pin products)**

Address: FF3FH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | 0 | 0 | 0 | 0 | 0 | CSISEL | 0 | 0 |

| CSISEL | Pin function assignment to be used with serial interface CSI11 |
|---|---|
| 0 | $\overline{SCK11}$/P60, SI11/P61, SO11/P62 |
| 1 | $\overline{SCK11}$/P40, SI11/P41, SO11/P120 |

## 4.4 Port Function Operations

Port operations differ depending on whether the input or output mode is set, as shown below.

### 4.4.1 Writing to I/O port

**(1) Output mode**

A value is written to the output latch by a transfer instruction, and the output latch contents are output from the pin.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

**(2) Input mode**

A value is written to the output latch by a transfer instruction, but since the output buffer is off, the pin status does not change.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

### 4.4.2 Reading from I/O port

**(1) Output mode**

The output latch contents are read by a transfer instruction. The output latch contents do not change.

**(2) Input mode**

The pin status is read by a transfer instruction. The output latch contents do not change.

### 4.4.3 Operations on I/O port

**(1) Output mode**

An operation is performed on the output latch contents, and the result is written to the output latch. The output latch contents are output from the pins.
Once data is written to the output latch, it is retained until data is written to the output latch again.
The data of the output latch is cleared when a reset signal is generated.

**(2) Input mode**

The pin level is read and an operation is performed on its contents. The result of the operation is written to the output latch, but since the output buffer is off, the pin status does not change.
The data of the output latch is cleared when a reset signal is generated.

## 4.5 Settings of Port Mode Register and Output Latch When Using Alternate Function

To use the alternate function of a port pin, set the port mode register and output latch as shown in **Tables 4-14** to **4-18**.

**Table 4-14. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KY2-L) (1/2)**

| Pin Name | Alternate Function | | PM×× | P×× |
|---|---|---|---|---|
| | Function Name | I/O | | |
| P00 | TI000 | Input | 1 | × |
| | INTP0 | Input | 1 | × |
| P01 | TI010 | Input | 1 | × |
| | TO00 | Output | 0 | 0 |
| P20 | ANI0Note 1 | Input | 1 | × |
| | AMP0-Notes 1, 2 | Input | 1 | × |
| P21 | ANI1Note 1 | Input | 1 | × |
| | AMP0OUTNotes 1, 2 | Output | 1 | × |
| | PGAINNotes 1, 2 | Input | 1 | × |
| P22 | ANI2Note 1 | Input | 1 | × |
| | AMP0+Notes 1, 2 | Input | 1 | × |
| P23 | ANI3Note 1 | Input | 1 | × |
| P30 | INTP1 | Input | 1 | × |
| | TI51 | Input | 1 | × |
| | TOH1 | Output | 0 | 0 |
| P60 | SCLA0Notes 3, 4 | I/O | 0 | 1 |
| | TxD6Note 5 | Output | 0 | 1 |
| P61 | SDAA0Notes 3, 4 | I/O | 0 | 1 |
| | RxD6 | Input | 1 | × |

**Notes 1.** The pin function can be selected by using ADPC0 register, PM2 register, ADS register, OPAMP0E bit, and PGAIN bit. Refer to **Tables 4-10** to **4-12** of **4.2.3 Port 2**.

**2.** $\mu$PD78F0555, 78F0556, 78F0557 (products with operational amplifier) only

**3.** During $I^2C$ communication, set SCLA0 and SDAA0 to N-ch open drain output ($V_{DD}$ tolerance) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**4.** When using an input compliant with the SMBus specifications in $I^2C$ communication, select the SMBus input buffer by using PIM6 register (refer to **4.3 (4) Port input mode register 6 (PIM6)**).

<R> **5.** During UART communication, set TxD6 to normal output (CMOS output) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

**Table 4-14. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KY2-L) (2/2)**

| Pin Name | Alternate Function | | PM×× | P×× |
|---|---|---|---|---|
| | Function Name | I/O | | |
| P121 | X1[Note 1] | – | × | × |
| | TOOLC0 | Input | × | × |
| P122 | X2[Note 1] | – | × | × |
| | EXCLK[Note 1] | Input | × | × |
| | TOOLD0 | I/O | × | × |
| P125 | $\overline{\text{RESET}}$[Note 2] | Input | × | × |

**Notes 1.** When using the P121 and P122 pins to connect a resonator for the main system clock (X1, X2) or to input an external clock for the main system clock (EXCLK), the X1 oscillation mode or external clock input mode must be set by using OSCCTL register (for details, refer to **5.3 (1) Clock operation mode select register (OSCCTL)**). The reset value of OSCCTL is 00H (both P121 and P122 are input port pins).

**2.** Clear RSTM bit (bit 5 of RSTMASK register) to 0 when using P125 as an external reset input ($\overline{\text{RESET}}$).

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

**Table 4-15. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KA2-L (20-pin products)) (1/2)**

| Pin Name | Alternate Function | | PM×× | P×× |
|---|---|---|---|---|
| | Function Name | I/O | | |
| P00 | TI000 | Input | 1 | × |
| | INTP0 | Input | 1 | × |
| P01 | TI010 | Input | 1 | × |
| | TO00 | Output | 0 | 0 |
| P20 | ANI0[Note 1] | Input | 1 | × |
| | AMP0-[Notes 1, 2] | Input | 1 | × |
| P21 | ANI1[Note 1] | Input | 1 | × |
| | AMP0OUT[Notes 1, 2] | Output | 1 | × |
| | PGAIN[Notes 1, 2] | Input | 1 | × |
| P22 | ANI2[Note 1] | Input | 1 | × |
| | AMP0+[Notes 1, 2] | Input | 1 | × |
| P23 to P25 | ANI3 to ANI5[Note 1] | Input | 1 | × |
| P30 | INTP1 | Input | 1 | × |
| | TI51 | Input | 1 | × |
| | TOH1 | Output | 0 | 0 |
| P31 | INTP2 | Input | 1 | × |
| | TOOLC1 | Input | × | × |
| P32 | INTP3 | Input | 1 | × |
| | TOOLD1 | I/O | × | × |
| P60 | SCLA0[Notes 3, 4] | I/O | 0 | 1 |
| | TxD6[Note 5] | Output | 0 | 1 |
| P61 | SDAA0[Notes 3, 4] | I/O | 0 | 1 |
| | RxD6 | Input | 1 | × |

**Notes 1.** The pin function can be selected by using ADPC0 register, PM2 register, ADS register, OPAMP0E bit, and PGAIN bit. Refer to **Tables 4-10** to **4-12** of **4.2.3 Port 2**.

**2.** $\mu$PD78F0565, 78F0566, 78F0567 (products with operational amplifier) only

**3.** During $I^2C$ communication, set SCLA0 and SDAA0 to N-ch open drain output ($V_{DD}$ tolerance) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**4.** When using an input compliant with the SMBus specifications in $I^2C$ communication, select the SMBus input buffer by using PIM6 register (refer to **4.3 (4) Port input mode register 6 (PIM6)**).

**5.** During UART communication, set TxD6 to normal output (CMOS output) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

**Table 4-15. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KA2-L (20-pin products)) (2/2)**

| Pin Name | Alternate Function | | PM×× | P×× |
|---|---|---|---|---|
| | Function Name | I/O | | |
| P121 | X1[Note 1] | – | × | × |
| | TOOLC0 | Input | × | × |
| P122 | X2[Note 1] | – | × | × |
| | EXCLK[Note 1] | Input | × | × |
| | TOOLD0 | I/O | × | × |
| P125 | $\overline{\text{RESET}}$[Note 2] | Input | × | × |

**Notes 1.** When using the P121 and P122 pins to connect a resonator for the main system clock (X1, X2) or to input an external clock for the main system clock (EXCLK), the X1 oscillation mode or external clock input mode must be set by using OSCCTL register (for details, refer to **5.3 (1) Clock operation mode select register (OSCCTL)**). The reset value of OSCCTL is 00H (both P121 and P122 are input port pins).

**2.** Clear RSTM bit (bit 5 of RSTMASK register) to 0 when using P125 as an external reset input ($\overline{\text{RESET}}$).

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

<R> **Table 4-16. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KA2-L (25-pin and 32-pin product)) (1/2)**

| Pin Name | Alternate Function | | MUXSEL | PM×× | P×× |
|---|---|---|---|---|---|
| | Function Name | I/O | | | |
| P00[Note 1] | TI000[Note 1] | Input | TM00SEL0 = 0 | 1 | × |
| | INTP0[Note 1] | Input | INTP0SEL0 = 0 | 1 | × |
| | (TOH1)[Note 1] | Output | TMHSEL1, TMHSEL0 = 1, 0 | 0 | 0 |
| | (TI51)[Note 1] | Input | TM5SEL1, TM5SEL0 = 1, 0 | 1 | × |
| P01[Note 2] | TI010[Note 2] | Input | – | 1 | × |
| | TO00[Note 2] | Output | – | 0 | 0 |
| P20 | ANI0[Note 3] | Input | – | 1 | × |
| | AMP0-[Notes 3, 4] | Input | – | 1 | × |
| P21 | ANI1[Note 3] | Input | – | 1 | × |
| | AMP0OUT[Notes 3, 4] | Output | – | 1 | × |
| | PGAIN[Notes 3, 4] | Input | – | 1 | × |
| P22 | ANI2[Note 3] | Input | – | 1 | × |
| | AMP0+[Notes 3, 4] | Input | – | 1 | × |
| P23 to P26 | ANI3 to ANI6[Note 3] | Input | – | 1 | × |
| P27[Note 2] | ANI7[Notes 2, 3] | Input | – | 1 | × |
| P31 | INTP2 | Input | – | 1 | × |
| | TOOLC1 | Input | – | × | × |
| P32 | INTP3 | Input | – | 1 | × |
| | TOOLD1 | Input | – | × | × |
| P34 | INTP4 | I/O | – | 1 | × |
| | (TOH1) | Output | TMHSEL1[Note 1], TMHSEL0 = 0, 1 | 0 | 0 |
| | (TI51)[Note 1] | Input | TM5SEL1, TM5SEL0 = 0, 1 | 1 | × |
| P35 | $\overline{\text{SCK11}}$ | Input | – | 1 | × |
| | | Output | – | 0 | 1 |
| P36 | SI11 | Input | – | 1 | × |
| P37 | SO11 | Input | – | 0 | 0 |

**Notes 1.** 25-pin products only

**2.** 32-pin products only

**3.** The pin function can be selected by using ADPC0 register, PM2 register, ADS register, OPAMP0E bit, and PGAIN bit. Refer to **Tables 4-10** to **4-12** of **4.2.3 Port 2**.

**4.** $\mu$PD78F0565, 78F0566, and 78F0567 (products with operational amplifier) only

**Remarks 1.** ×: Don't care
PM××: Port mode register
P××: Port output latch

**2.** Functions in parentheses ( ) can be assigned by setting MUXSEL register.

<R> **Table 4-16. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KA2-L (25-pin and 32-pin products)) (2/2)**

| Pin Name | Alternate Function | | MUXSEL | PM×× | P×× |
|---|---|---|---|---|---|
| | Function Name | I/O | | | |
| P60 | SCLA0 Notes 1, 2 | I/O | – | 0 | 1 |
| | TxD6 Note 3 | Output | – | 0 | 1 |
| P61 | SDAA0 Notes 1, 2 | I/O | – | 0 | 1 |
| | RxD6 | Input | – | 1 | × |
| P70 to P72 Note 4 | ANI8 to ANI10 Notes 4, 5 | Input | – | 1 | × |
| P121 | X1 Note 6 | – | – | × | × |
| | TOOLC0 | Input | – | × | × |
| | (TI000) | Input | TM00SEL1 Note 4, TM00SEL0 = 0, 1 | × | × |
| | (INTP0) | Input | INTP0SEL1 Note 4, INTP0SEL0 = 0, 1 | × | × |
| P122 | X2 Note 6 | – | – | × | × |
| | EXCLK Note 6 | Input | – | × | × |
| | TOOLD0 | I/O | – | × | × |
| P125 | $\overline{\text{RESET}}$ Note 7 | Input | – | × | × |
| | (TI000) Note 4 | Input | TM00SEL1, TM00SEL0 = 1, 0 | × | × |
| | (INTP0) Note 4 | Input | INTP0SEL1, INTP0SEL0 = 1, 0 | × | × |

**Notes 1.** During $I^2C$ communication, set SCLA0 and SDAA0 to N-ch open drain output ($V_{DD}$ tolerance) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**2.** When using an input compliant with the SMBus specifications in $I^2C$ communication, select the SMBus input buffer by using PIM6 register (refer to **4.3 (4) Port input mode register 6 (PIM6)**).

**3.** During UART communication, set TxD6 to normal output (CMOS output) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**4.** 32-pin products only

**5.** The pin function can be selected by using ADPC1 register, PM7 register, and ADS register. Refer to **Table 4-13** of **4.2.7 Port 7**.

**6.** When using the P121 and P122 pins to connect a resonator for the main system clock (X1, X2) or to input an external clock for the main system clock (EXCLK), the X1 oscillation mode or external clock input mode must be set by using OSCCTL register (for details, refer to **5.3 (1) Clock operation mode select register (OSCCTL)**). The reset value of OSCCTL is 00H (both P121 and P122 are input port pins).

**7.** Clear RSTM bit (bit 5 of RSTMASK register) to 0 when using P125 as an external reset input ($\overline{\text{RESET}}$).

**Remarks 1.** ×: Don't care
PM××: Port mode register
P××: Port output latch

**2.** Functions in parentheses ( ) can be assigned by setting MUXSEL register.

**Table 4-17. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KB2-L) (1/2)**

| Pin Name | Alternate Function | | PM×× | P×× |
|---|---|---|---|---|
| | Function Name | I/O | | |
| P00 | TI000 | Input | 1 | × |
| P01 | TI010 | Input | 1 | × |
| | TO00 | Output | 0 | 0 |
| P10 | ANI8Note 2 | Input | 1 | × |
| | AMP1-Notes 1, 2 | Input | 1 | × |
| | $\overline{SCK10}$ | Input | 1 | × |
| | | Output | 0 | 1 |
| P11 | ANI9Note 2 | Input | 1 | × |
| | AMP1OUTNotes 1, 2 | Output | 1 | × |
| | SI10 | Input | 1 | × |
| P12 | ANI10Note 2 | Input | 1 | × |
| | AMP1+Notes 1, 2 | Input | 1 | × |
| | SI10 | Output | 0 | 0 |
| P13 | TxD6 | Output | 0 | 1 |
| P14 | RxD6 | Input | 1 | × |
| P15 | TOH0 | Output | 0 | 0 |
| P16 | TOH1 | Output | 0 | 0 |
| | INTP5 | Input | 1 | × |
| P17 | TI50 | Input | 1 | × |
| | TO50 | Output | 0 | 0 |
| P20 | ANI0Note 3 | Input | 1 | × |
| | AMP0-Notes 1, 3 | Input | 1 | × |
| P21 | ANI1Note 3 | Input | 1 | × |
| | AMP0OUTNotes 1, 3 | Output | 1 | × |
| | PGAINNotes 1, 3 | Input | 1 | × |
| P22 | ANI2Note 3 | Input | 1 | × |
| | AMP0+Notes 1, 3 | Input | 1 | × |
| P23 | ANI3Note 3 | Input | 1 | × |

**Notes 1** $\mu$PD78F0576, 78F0577, 78F0578 (products with operational amplifier) only

**2.** The pin function can be selected by using ADPC1 register, PM1 register, ADS register, and OPAMP1E bit. Refer to **Tables 4-8** and **4-9** of **4.2.2 Port 1**.

**3.** The pin function can be selected by using ADPC0 register, PM2 register, ADS register, OPAMP0E bit, and PGAIN bit. Refer to **Tables 4-10** to **4-12** of **4.2.3 Port 2**.

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

**Table 4-17. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KB2-L) (2/2)**

| Pin Name | Alternate Function | | PM×× | P×× |
|---|---|---|---|---|
| | Function Name | I/O | | |
| P30 | INTP1 | Input | 1 | × |
| P31 | INTP2 | Input | 1 | × |
| | TOOLC1 | Input | × | × |
| P32 | INTP3 | Input | 1 | × |
| | TOOLD1 | I/O | × | × |
| P33 | INTP4 | Input | 1 | × |
| | TI51 | Input | 1 | × |
| | TO51 | Output | 0 | 0 |
| P60 | SCLA0Notes 1, 2 | I/O | 0 | 1 |
| | INTP11 | Input | 1 | × |
| P61 | SDAA0Notes 1, 2 | I/O | 0 | 1 |
| | INTP10 | Input | 1 | × |
| P120 | INTP0 | Input | 1 | × |
| | EXLVI | Input | 1 | × |
| P121 | X1Note 3 | – | × | × |
| | TOOLC0 | Input | × | × |
| P122 | X2Note 3 | – | × | × |
| | EXCLKNote 3 | Input | × | × |
| | TOOLD0 | I/O | × | × |
| P125 | $\overline{\text{RESET}}$Note 4 | Input | × | × |

**Notes 1.** During $I^2C$ communication, set SCLA0 and SDAA0 to N-ch open drain output ($V_{DD}$ tolerance) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**2.** When using an input compliant with the SMBus specifications in $I^2C$ communication, select the SMBus input buffer by using PIM6 register (refer to **4.3 (4) Port input mode register 6 (PIM6)**).

**3.** When using the P121 and P122 pins to connect a resonator for the main system clock (X1, X2) or to input an external clock for the main system clock (EXCLK), the X1 oscillation mode or external clock input mode must be set by using OSCCTL register (for details, refer to **5.3 (1) Clock operation mode select register (OSCCTL)**). The reset value of OSCCTL is 00H (both P121 and P122 are input port pins).

**4.** Clear RSTM bit (bit 5 of RSTMASK register) to 0 when using P125 as an external reset input ($\overline{\text{RESET}}$).

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

<R> **Table 4-18. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KC2-L) (1/3)**

| Pin Name | Alternate Function | | MUXSEL | PM×× | P×× |
|---|---|---|---|---|---|
| | Function Name | I/O | | | |
| P00 | TI000 | Input | – | 1 | × |
| P01 | TI010 | Input | – | 1 | × |
| | TO00 | Output | – | 0 | 0 |
| P02[Note 1] | INTP7[Note 1] | Input | – | 1 | × |
| P10 | ANI8[Note 3] | Input | – | 1 | × |
| | AMP1-[Notes 2, 3] | Input | – | 1 | × |
| | $\overline{\text{SCK10}}$ | Input | – | 1 | × |
| | | Output | – | 0 | 1 |
| P11 | ANI9[Note 3] | Input | – | 1 | × |
| | AMP1OUT[Notes 2, 3] | Output | – | 1 | × |
| | SI10 | Input | – | 1 | × |
| P12 | ANI10[Note 3] | Input | – | 1 | × |
| | AMP1+[Notes 2, 3] | Input | – | 1 | × |
| | SO10 | Output | – | 0 | 0 |
| P13 | TxD6 | Output | – | 0 | 1 |
| P14 | RxD6 | Input | – | 1 | × |
| P15 | TOH0 | Output | – | 0 | 0 |
| P16 | TOH1 | Output | – | 0 | 0 |
| | INTP5 | Input | – | 1 | × |
| P17 | TI50 | Input | – | 1 | × |
| | TO50 | Output | – | 0 | 0 |

**Notes 1.** 48-pin products only

**2.** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**3.** The pin function can be selected by using ADPC1 register, PM1 register, ADS register, and OPAMP1E bit. Refer to **Tables 4-8** and **4-9** of **4.2.2 Port 1**.

**Remark** ×: Don't care
PM××: Port mode register
P××: Port output latch

<R> **Table 4-18. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KC2-L) (2/3)**

| Pin Name | Alternate Function | | MUXSEL | PM×× | P×× |
|---|---|---|---|---|---|
| | Function Name | I/O | | | |
| P20 | ANI0[Note 1] | Input | – | 1 | × |
| | AMP0-[Notes 1, 2] | Input | – | 1 | × |
| P21 | ANI1[Note 1] | Input | – | 1 | × |
| | AMP0OUT[Notes 1, 2] | Output | – | 1 | × |
| | PGAIN[Notes 1, 2] | Input | – | 1 | × |
| P22 | ANI2[Note 1] | Input | – | 1 | × |
| | AMP0+[Notes 1, 2] | Input | – | 1 | × |
| P23 to P26, P27[Note 3] | ANI3 to ANI6[Note 1], ANI7[Notes 1, 3] | Input | – | 1 | × |
| P30 | INTP1 | Input | – | 1 | × |
| P31 | INTP2 | Input | – | 1 | × |
| | TOOLC1 | Input | – | × | × |
| P32 | INTP3 | Input | – | 1 | × |
| | TOOLD1 | I/O | – | × | × |
| P33 | INTP4 | Input | – | 1 | × |
| | TI51 | Input | – | 1 | × |
| | TO51 | Output | – | 0 | 0 |
| P40[Note 3] | RTCCL[Note 3] | Output | – | 0 | 0 |
| | RTCDIV[Note 3] | Output | – | 0 | 0 |
| | ($\overline{SCK11}$)[Note 3] | Input | CSISEL = 1 | 1 | × |
| | | Output | CSISEL = 1 | 0 | 1 |
| P41[Note 3] | RTC1HZ[Note 3] | Output | – | 0 | 0 |
| | (SI11)[Note 3] | Input | CSISEL = 1 | 1 | × |
| P42[Note 4] | PCL[Note 4] | Output | – | 0 | 0 |
| | $\overline{SSI11}$[Note 4] | Input | – | 1 | × |
| | INTP6[Note 4] | Input | – | 1 | × |
| P60 | SCLA0[Notes 5, 6] | I/O | – | 0 | 1 |
| | $\overline{SCK11}$ | Input | CSISEL = 0 | 1 | × |
| | | Output | CSISEL = 0 | 0 | 1 |
| | INTP11 | Input | – | 1 | × |

**Notes 1.** The pin function can be selected by using ADPC0 register, PM2 register, ADS register, OPAMP0E bit, and PGAIN bit. Refer to **Tables 4-10** to **4-12** of **4.2.3 Port 2**.

**2.** $\mu$PD78F0586, 78F0587, 78F0588 (products with operational amplifier) only

**3.** 44-pin and 48-pin products only

**4.** 48-pin products only

**5.** During $I^2C$ communication, set SCLA0 and SDAA0 to N-ch open drain output ($V_{DD}$ tolerance) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**6.** When using an input compliant with the SMBus specifications in $I^2C$ communication, select the SMBus input buffer by using PIM6 register (refer to **4.3 (4) Port input mode register 6 (PIM6)**).

**Remarks 1.** ×: Don't care
PM××: Port mode register
P××: Port output latch

**2.** Functions in parentheses ( ) can be assigned by setting MUXSEL register.

<R> **Table 4-18. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KC2-L) (3/3)**

| Pin Name | Alternate Function | | MUXSEL | PM×× | P×× |
|---|---|---|---|---|---|
| | Function Name | I/O | | | |
| P61 | SDAA0[Notes 1, 2] | I/O | – | 0 | 1 |
| | SI11 | Input | CSISEL = 0 | 1 | × |
| | INTP10 | Input | – | 1 | × |
| P62 | SO11 | Output | CSISEL = 0 | 0 | 0 |
| | INTP9 | Input | – | 1 | × |
| P63[Note 3] | INTP8[Note 3] | Input | – | 1 | × |
| P70 to P73, P74[Note 4], P75[Note 4] | KR0 to KR3, KR4[Note 4], KR5[Note 4] | Input | – | 1 | × |
| P120 | INTP0 | Input | – | 1 | × |
| | EXLVI | Input | – | 1 | × |
| | (SO11)[Note 3] | Output | CSISEL = 1 | 0 | 0 |
| P121 | X1[Note 5] | – | – | × | × |
| | TOOLC0 | Input | – | × | × |
| P122 | X2[Note 5] | – | – | × | × |
| | EXCLK[Note 5] | Input | – | × | × |
| | TOOLD0 | I/O | – | × | × |
| P123 | XT1[Note 5] | – | – | × | × |
| P124 | XT2[Note 5] | – | – | × | × |
| | EXCLKS[Note 5] | Input | – | × | × |
| P125 | $\overline{\text{RESET}}$[Note 6] | Input | – | × | × |

**Notes 1.** During $I^2C$ communication, set SCLA0 and SDAA0 to N-ch open drain output ($V_{DD}$ tolerance) mode by using POM6 register (refer to **4.3 (5) Port output mode register 6 (POM6)**).

**2.** When using an input compliant with the SMBus specifications in $I^2C$ communication, select the SMBus input buffer by using PIM6 register (refer to **4.3 (4) Port input mode register 6 (PIM6)**).

**3.** 44-pin and 48-pin products only

**4.** 48-pin products only

**5.** When using the P121 to P124 pins to connect a resonator for the main system clock (X1, X2) or subsystem clock (XT1, XT2), or to input an external clock for the main system clock (EXCLK) or subsystem clock (EXCLKS), the X1 oscillation mode, XT1 oscillation mode, or external clock input mode must be set by using OSCCTL register (for details, refer to **5.3 (1) Clock operation mode select register (OSCCTL)** and **(3) Setting of operation mode for subsystem clock pin**). The reset value of OSCCTL is 00H (all of the P121 to P124 pins are Input port pins).

**6.** Clear RSTM bit (bit 5 of RSTMASK register) to 0 when using P125 as an external reset input ($\overline{\text{RESET}}$).

**Remarks 1.** ×: Don't care
PM××: Port mode register
P××: Port output latch

**2.** Functions in parentheses ( ) can be assigned by setting MUXSEL register.

## 4.6 Cautions on 1-Bit Manipulation Instruction for Port Register n (Pn)

When a 1-bit manipulation instruction is executed on a port that provides both input and output functions, the output latch value of an input port that is not subject to manipulation may be written in addition to the targeted bit.

Therefore, it is recommended to rewrite the output latch when switching a port from input mode to output mode.

<Example> When P10 is an output port, P11 to P17 are input ports (all pin statuses are high level), and the port latch value of port 1 is 00H, if the output of output port P10 is changed from low level to high level via a 1-bit manipulation instruction, the output latch value of port 1 is FFH.

Explanation: The targets of writing to and reading from the Pn register of a port whose PMnm bit is 1 are the output latch and pin status, respectively.

A 1-bit manipulation instruction is executed in the following order in the 78K0/Kx2-L microcontrollers.

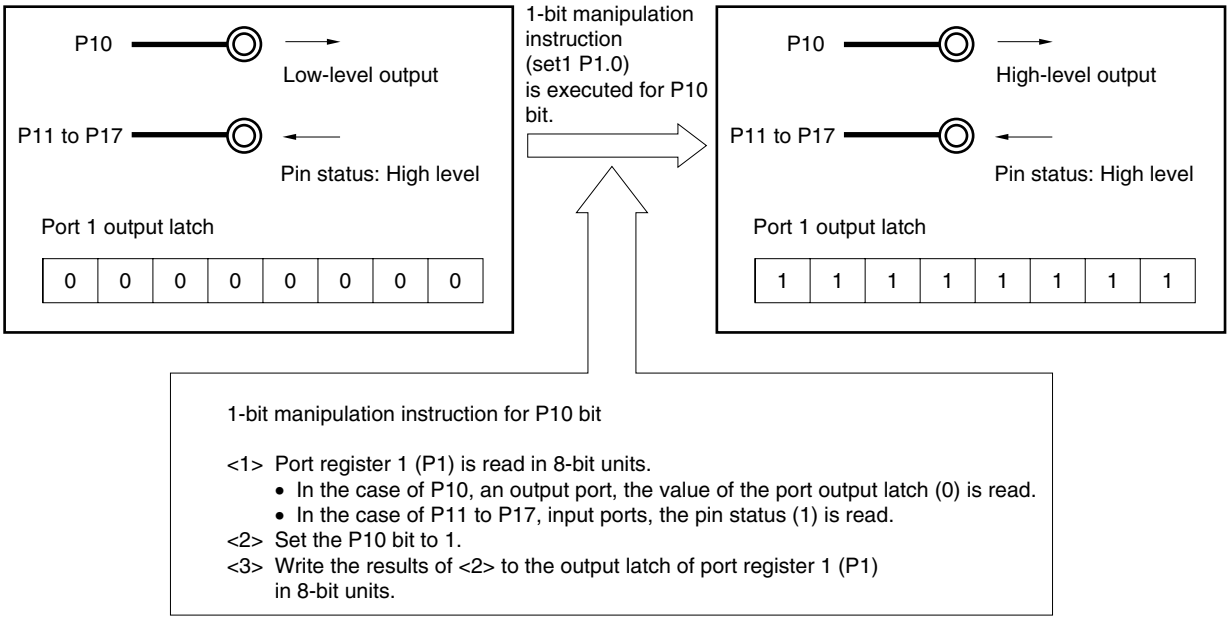
<1> The Pn register is read in 8-bit units.
<2> The targeted one bit is manipulated.
<3> The Pn register is written in 8-bit units.

In step <1>, the output latch value (0) of P10, which is an output port, is read, while the pin statuses of P11 to P17, which are input ports, are read. If the pin statuses of P11 to P17 are high level at this time, the read value is FEH.

The value is changed to FFH by the manipulation in <2>.

FFH is written to the output latch by the manipulation in <3>.

**Figure 4-50 1-Bit Manipulation Instruction (P10)**




**Remark** The following instructions are 1-bit manipulation instructions.

- MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1

# CHAPTER 5 CLOCK GENERATOR

## 5.1 Functions of Clock Generator

The clock generator generates the clock to be supplied to the CPU and peripheral hardware.
The following three kinds of system clocks and clock oscillators are selectable.

**(1) Main system clock**

**<1> X1 oscillator**

This circuit oscillates a clock of $f_X$ = 1 to 10 MHz by connecting a resonator to X1 and X2.
Oscillation can be stopped by executing the STOP instruction or using the main OSC control register (MOC).

**<2> Internal high-speed oscillator**

This circuit oscillates a clock of $f_{IH}$ = 4 MHz (TYP.)/8 MHz (TYP.). After a reset release, the CPU always starts operating with this internal high-speed oscillation clock. Oscillation can be stopped by executing the STOP instruction or using the internal oscillation mode register (RCM).

An external main system clock ($f_{EXCLK}$ = 1 to 10 MHz) can also be supplied from the EXCLK/X2/P122 pin. An external main system clock input can be disabled by executing the STOP instruction or using RCM.
As the main system clock, a high-speed system clock (X1 clock or external main system clock) or internal high-speed oscillation clock can be selected by using the main clock mode register (MCM).

**(2) Subsystem clock[Note]**

- **Subsystem clock oscillator**

  This circuit oscillates at a frequency of $f_{XT}$ = 32.768 kHz by connecting a 32.768 kHz resonator across XT1 and XT2. Oscillation can be stopped by using the processor clock control register (PCC) and clock operation mode select register (OSCCTL).

An external subsystem clock ($f_{EXCLKS}$ = 32.768 kHz) can also be supplied from the EXCLKS/XT2/P124 pin. An external subsystem clock input can be disabled by setting PCC and OSCCTL.

**Note** 78K0/KC2-L only

**Remark** $f_X$: X1 clock oscillation frequency
$f_{IH}$: Internal high-speed oscillation clock frequency
$f_{EXCLK}$: External main system clock frequency
$f_{XT}$: XT1 clock oscillation frequency
$f_{EXCLKS}$: External subsystem clock frequency

**(3) Internal low-speed oscillation clock (clock for watchdog timer)**

- **Internal low-speed oscillator**

  This circuit oscillates a clock of $f_{IL}$ = 30 kHz (TYP.). After a reset release, the internal low-speed oscillation clock always starts operating.

  Oscillation can be stopped by using the internal oscillation mode register (RCM) when “internal low-speed oscillator can be stopped by software” is set by option byte.

  The internal low-speed oscillation clock cannot be used as the CPU clock. The following hardware operates with the internal low-speed oscillation clock.

  - Watchdog timer
  - 8-bit timer H1 (when $f_{IL}$, $f_{IL}/2^6$, or $f_{IL}/2^{15}$ is selected)

**Remark** $f_{IL}$: Internal low-speed oscillation clock frequency

## 5.2 Configuration of Clock Generator

The clock generator includes the following hardware.

**Table 5-1. Configuration of Clock Generator**

| Item | Configuration |
|---|---|
| Control registers | Clock operation mode select register (OSCCTL)<br>Processor clock control register (PCC)<br>Internal oscillation mode register (RCM)<br>Main OSC control register (MOC)<br>Main clock mode register (MCM)<br>Oscillation stabilization time counter status register (OSTC)<br>Oscillation stabilization time select register (OSTS) |
| Oscillators | X1 oscillator<br>XT1 oscillator[Note]<br>Internal high-speed oscillator<br>Internal low-speed oscillator |

**Note** 78K0/KC2-L only

**Figure 5-1. Block Diagram of Clock Generator (78K0/KY2-L, 78K0/KA2-L, 78K0/KB2-L)**

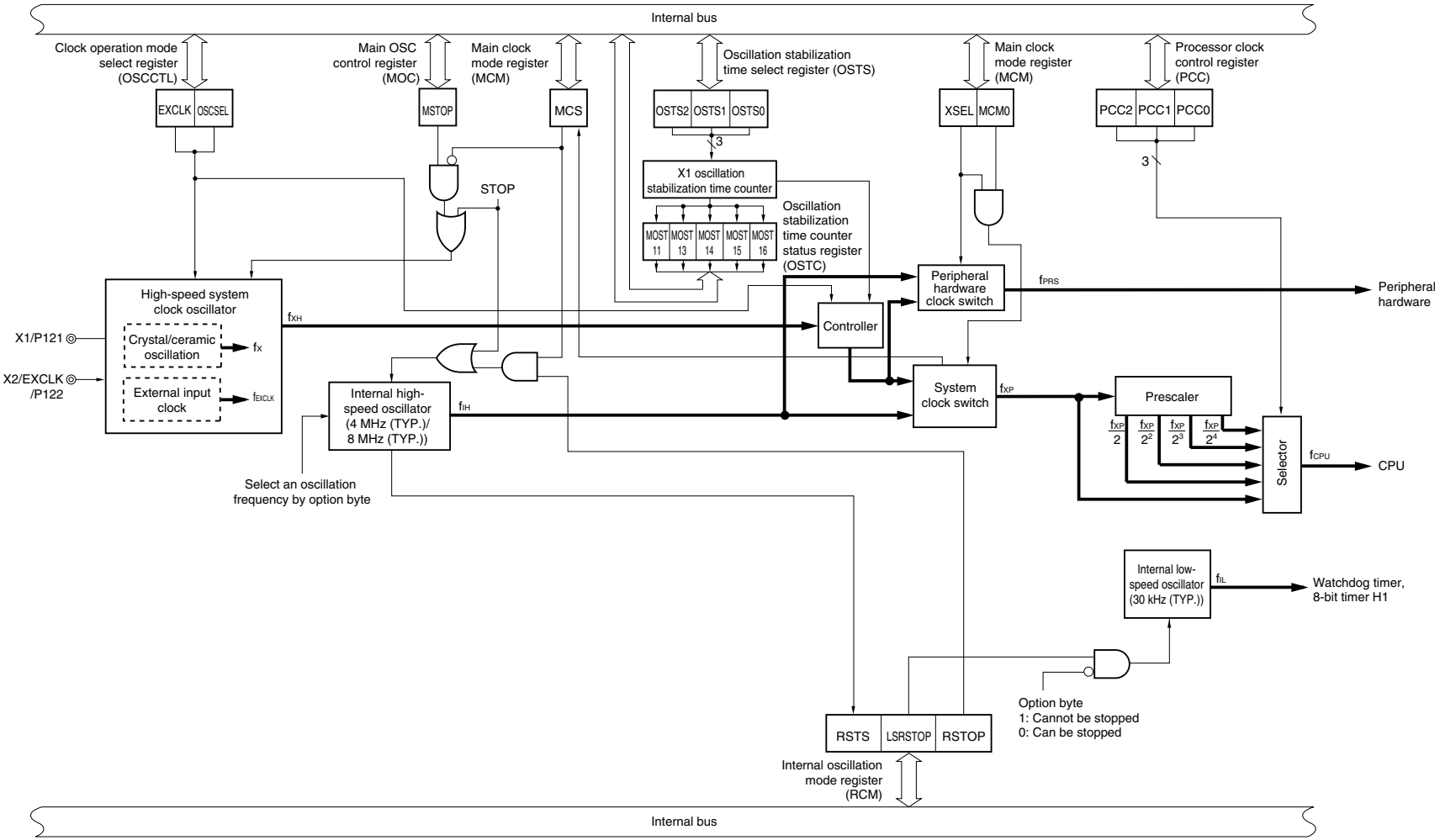
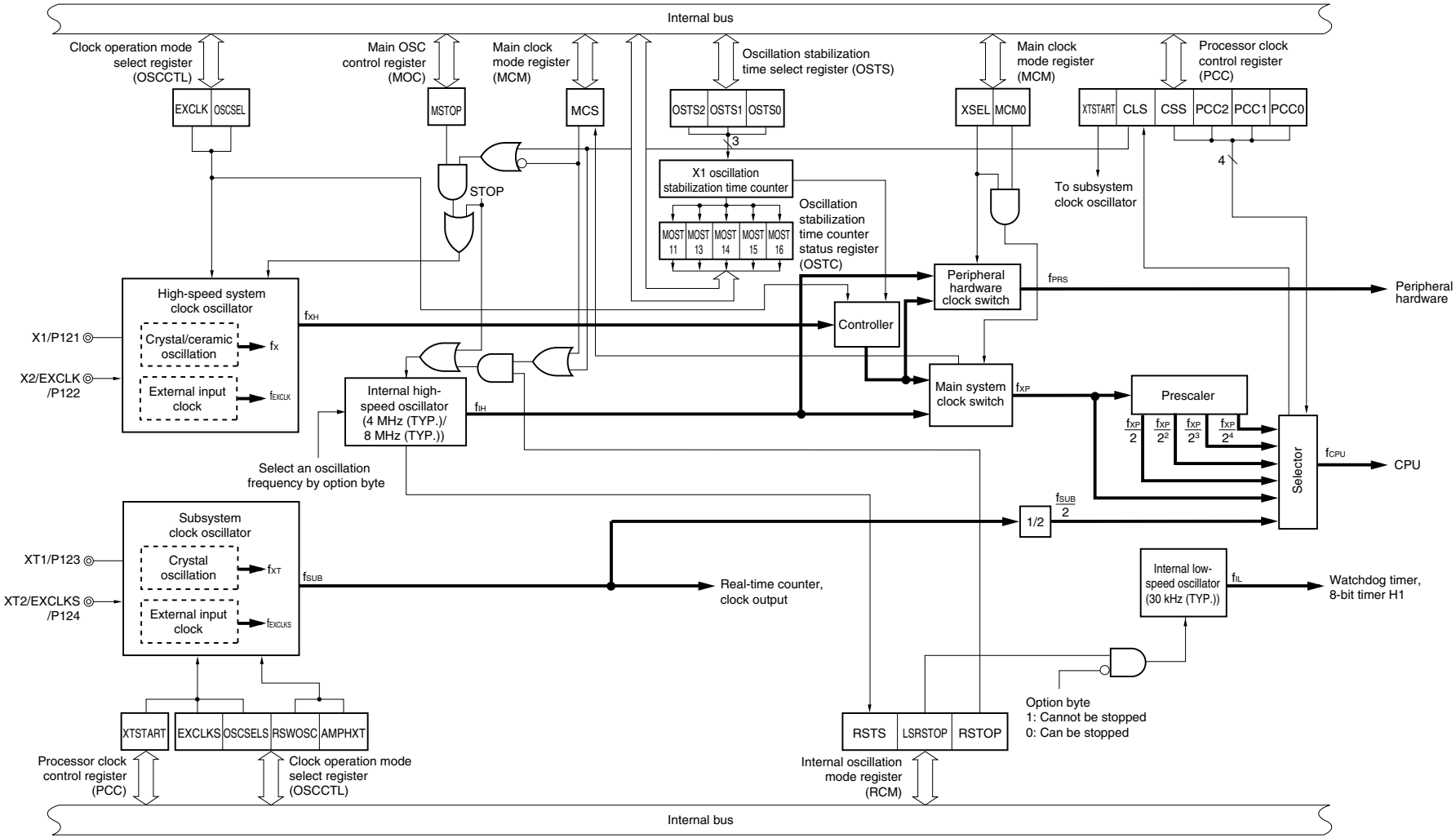


**Figure 5-2. Block Diagram of Clock Generator (78K0/KC2-L)**



**Remark** $f_X$: X1 clock oscillation frequency
$f_{IH}$: Internal high-speed oscillation clock frequency
$f_{EXCLK}$: External main system clock frequency
$f_{XH}$: High-speed system clock frequency
$f_{XP}$: Main system clock frequency
$f_{PRS}$: Peripheral hardware clock frequency
$f_{CPU}$: CPU clock frequency
$f_{XT}$: XT1 clock oscillation frequency
$f_{EXCLKS}$: External subsystem clock frequency
$f_{SUB}$: Subsystem clock frequency
$f_{IL}$: Internal low-speed oscillation clock frequency

## 5.3 Registers Controlling Clock Generator

The following eight registers are used to control the clock generator.

- Clock operation mode select register (OSCCTL)
- Processor clock control register (PCC)
- Internal oscillation mode register (RCM)
- Main OSC control register (MOC)
- Main clock mode register (MCM)
- Oscillation stabilization time counter status register (OSTC)
- Oscillation stabilization time select register (OSTS)
- Peripheral enable register 0 (PER0)

### (1) Clock operation mode select register (OSCCTL)

This register selects the operation modes of the high-speed system and subsystem clocks.
OSCCTL can be set by a 1-bit or 8-bit memory manipulation instruction.
Reset signal generation clears this register to 00H.

**Figure 5-3. Format of Clock Operation Mode Select Register (OSCCTL)**
**(78K0/KY2-L, 78K0/KA2-L, 78K0/KB2-L)**

Address: FF9FH After reset: 00H R/W

| Symbol | <7> | <6> | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| OSCCTL | EXCLK | OSCSEL | 0 | 0 | 0 | 0 | 0 | 0 |

| EXCLK | OSCSEL | High-speed system clock pin operation mode | P121/X1 pin | P122/X2/EXCLK pin |
|---|---|---|---|---|
| 0 | 0 | Input port mode | Input port | |
| 0 | 1 | X1 oscillation mode | Crystal/ceramic resonator connection | |
| 1 | 0 | Input port mode | Input port | |
| 1 | 1 | External clock input mode | Input port | External clock input |

**Cautions 1. To change the value of EXCLK and OSCSEL, be sure to confirm that bit 7 (MSTOP) of the main OSC control register (MOC) is 1 (the X1 oscillator stops or the external clock from the EXCLK pin is disabled).**
**2. Be sure to clear bits 0 to 5 to 0.**

**Figure 5-4. Format of Clock Operation Mode Select Register (OSCCTL)**
**(78K0/KC2-L)**

Address: FF9FH After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | 3 | <2> | <1> | 0 |
|---|---|---|---|---|---|---|---|---|
| OSCCTL | EXCLK | OSCSEL | EXCLKS[Note] | OSCSELS[Note] | 0 | RSWOSC | AMPHXT | 0 |

| EXCLK | OSCSEL | High-speed system clock pin operation mode | P121/X1 pin | P122/X2/EXCLK pin |
|---|---|---|---|---|
| 0 | 0 | Input port mode | Input port | |
| 0 | 1 | X1 oscillation mode | Crystal/ceramic resonator connection | |
| 1 | 0 | Input port mode | Input port | |
| 1 | 1 | External clock input mode | Input port | External clock input |

| RSWOSC | AMPHXT | XT1 oscillator oscillation mode selection |
|---|---|---|
| 0 | 0 | Low power consumption oscillation (default) |
| 0 | 1 | Normal oscillation |
| 1 | × | Ultra-low power consumption oscillation |

**Note** EXCLKS and OSCSELS are used in combination with XTSTART (bit 6 of the processor clock control register (PCC)). Refer to **(3) Setting of operation mode for subsystem clock pin**.

**Cautions 1. To change the value of EXCLK and OSCSEL, be sure to confirm that bit 7 (MSTOP) of the main OSC control register (MOC) is 1 (the X1 oscillator stops or the external clock from the EXCLK pin is disabled).**
**2. Be sure to clear bits 0 and 3 to 0.**

**Caution 3. The XT1 oscillator is a circuit with low amplification in order to achieve low-power consumption. Note the following points when designing the circuit.**

- **Pins and circuit boards include parasitic capacitance. Therefore, perform oscillation evaluation using a circuit board to be actually used and confirm that there are no problems.**
- **Use the recommended resonator, which will be described in CHAPTER 28 ELECTRICAL SPECIFICATIONS after it is evaluated, when using the XT1 oscillator in the ultra-low power consumption oscillation (RSWOSC = 1).**
- **Make the wiring between the XT1 and XT2 pins and the resonators as short as possible, and minimize the parasitic capacitance and wiring resistance. Note this particularly when the ultra-low power consumption oscillation (RSWOSC = 1) is selected.**
- **Configure the circuit of the circuit board, using material with little wiring resistance.**
- **Place a ground pattern that has the same potential as $V_{SS}$ as much as possible near the XT1 oscillator.**
- **Be sure that the signal lines between the XT1 and XT2 pins, and the resonators do not cross with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **The impedance between the XT1 and XT2 pins may drop and oscillation may be disturbed due to moisture absorption of the circuit board in a high-humidity environment or dew condensation on the board. When using the circuit board in such an environment, take measures to damp-proof the circuit board, such as by coating.**
- **When coating the circuit board, use material that does not cause capacitance or leakage between the XT1 and XT2 pins.**

### (2) Processor clock control register (PCC)

This register is used to select the CPU clock, the division ratio, and operation mode for subsystem clock.

PCC is set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets PCC to 01H.

**Figure 5-5. Format of Processor Clock Control Register (PCC) (78K0/KY2-L, 78K0/KA2-L, 78K0/KB2-L)**

Address: FFFBH After reset: 01H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PCC | 0 | 0 | 0 | 0 | 0 | PCC2 | PCC1 | PCC0 |

| PCC2 | PCC1 | PCC0 | CPU clock ($f_{CPU}$) selection |
|---|---|---|---|
| 0 | 0 | 0 | $f_{XP}$ |
| 0 | 0 | 1 | $f_{XP}$/2 (default) |
| 0 | 1 | 0 | $f_{XP}/2^2$ |
| 0 | 1 | 1 | $f_{XP}/2^3$ |
| 1 | 0 | 0 | $f_{XP}/2^4$ |
| Other than above | | | Setting prohibited |

**Cautions 1. Be sure to clear bits 3 to 7 to 0.**

**2. The peripheral hardware clock ($f_{PRS}$) is not divided when the division ratio of the PCC is set.**

**Remark** $f_{XP}$: Main system clock oscillation frequency

**Figure 5-6. Format of Processor Clock Control Register (PCC)**
**(78K0/KC2-L)**

Address: FFFBH After reset: 01H R/W[Note 1]

| Symbol | 7 | 6 | <5> | <4> | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PCC | 0 | XTSTART[Note 2] | CLS | CSS | 0 | PCC2 | PCC1 | PCC0 |

| CLS | CPU clock status |
|---|---|
| 0 | Main system clock |
| 1 | Subsystem clock |

| CSS | PCC2 | PCC1 | PCC0 | CPU clock ($f_{CPU}$) selection |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $f_{XP}$ |
| | 0 | 0 | 1 | $f_{XP}$/2 (default) |
| | 0 | 1 | 0 | $f_{XP}/2^2$ |
| | 0 | 1 | 1 | $f_{XP}/2^3$ |
| | 1 | 0 | 0 | $f_{XP}/2^4$ |
| 1 | 0 | 0 | 0 | $f_{SUB}$/2 |
| | 0 | 0 | 1 | |
| | 0 | 1 | 0 | |
| | 0 | 1 | 1 | |
| | 1 | 0 | 0 | |
| Other than above | | | | Setting prohibited |

**Notes 1.** Bit 5 is read-only.

**2.** XTSTART is used in combination with EXCLKS and OSCSELS (bits 5 and 4 of the clock operation mode select register (OSCCTL)). Refer to **(3) Setting of operation mode for subsystem clock pin**.

**Cautions 1. Be sure to clear bits 3 and 7 to "0".**

**2. The peripheral hardware clock ($f_{PRS}$) is not divided when the division ratio of the PCC is set.**

**Remark** $f_{XP}$: Main system clock oscillation frequency
$f_{SUB}$: Subsystem clock oscillation frequency

The fastest instruction can be executed in 2 clocks of the CPU clock in the 78K0/Kx2-L microcontrollers. Therefore, the relationship between the CPU clock ($f_{CPU}$) and the minimum instruction execution time is as shown in Table 5-2.

**Table 5-2. Relationship between CPU Clock and Minimum Instruction Execution Time**

| CPU Clock ($f_{CPU}$) | Minimum Instruction Execution Time: $2/f_{CPU}$ | | | |
|---|---|---|---|---|
| | Main System Clock | | | Subsystem Clock[Note 2] |
| | High-Speed System Clock[Note 1] | Internal High-Speed Oscillation Clock[Note 1] | | |
| | At 10 MHz Operation | At 8 MHz (TYP.) Operation | At 4 MHz (TYP.) Operation | At 32.768 kHz Operation |
| $f_{XP}$ | 0.2 μs | 0.25 μs (TYP.) | 0.5 μs (TYP.) | – |
| $f_{XP}/2$ | 0.4 μs | 0.5 μs (TYP.) | 1.0 μs (TYP.) | – |
| $f_{XP}/2^2$ | 0.8 μs | 1.0 μs (TYP.) | 2.0 μs (TYP.) | – |
| $f_{XP}/2^3$ | 1.6 μs | 2.0 μs (TYP.) | 4.0 μs (TYP.) | – |
| $f_{XP}/2^4$ | 3.2 μs | 4.0 μs (TYP.) | 8.0 μs (TYP.) | – |
| $f_{SUB}/2$[Note 2] | – | – | | 122.1 μs |

**Notes 1.** The main clock mode register (MCM) is used to set the main system clock supplied to CPU clock (high-speed system clock/internal high-speed oscillation clock) (refer to **Figures 5-5 and 5-6**).

**2.** 78K0/KC2-L only

**(3) Setting of operation mode for subsystem clock pin**

The operation mode for the subsystem clock pin[Note] can be set by using bit 6 (XTSTART) of the processor clock control register (PCC) and bits 5 and 4 (EXCLKS, OSCSELS) of the clock operation mode select register (OSCCTL) in combination.

**Note** 78K0/KC2-L only

**Table 5-3. Setting of Operation Mode for Subsystem Clock Pin (78K0/KC2-L)**

| PCC | OSCCTL | | Subsystem Clock Pin Operation Mode | P123/XT1 Pin | P124/XT2/EXCLKS Pin |
|---|---|---|---|---|---|
| Bit 6 | Bit 5 | Bit 4 | | | |
| XTSTART | EXCLKS | OSCSELS | | | |
| 0 | 0 | 0 | Input port mode | Input port | |
| 0 | 0 | 1 | XT1 oscillation mode | Crystal resonator connection | |
| 0 | 1 | 0 | Input port mode | Input port | |
| 0 | 1 | 1 | External clock input mode | Input port | External clock input |
| 1 | × | × | XT1 oscillation mode | Crystal resonator connection | |

**Caution Confirm that bit 5 (CLS) of the processor clock control register (PCC) is 0 (CPU is operating with main system clock) when changing the current values of XTSTART, EXCLKS, and OSCSELS.**

**Remark** ×: don't care

**(4) Internal oscillation mode register (RCM)**

This register sets the operation mode of internal oscillator.

RCM can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 80H[Note 1].

**Figure 5-7. Format of Internal Oscillation Mode Register (RCM)**

Address: FFA0H After reset: 80H[Note 1] R/W[Note 2]

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| RCM | RSTS | 0 | 0 | 0 | 0 | 0 | LSRSTOP | RSTOP |

| RSTS | Status of internal high-speed oscillator |
|---|---|
| 0 | Waiting for accuracy stabilization of internal high-speed oscillator |
| 1 | Stability operating of internal high-speed oscillator |

| LSRSTOP | Internal low-speed oscillator oscillating/stopped |
|---|---|
| 0 | Internal low-speed oscillator oscillating |
| 1 | Internal low-speed oscillator stopped |

| RSTOP | Internal high-speed oscillator oscillating/stopped |
|---|---|
| 0 | Internal high-speed oscillator oscillating |
| 1 | Internal high-speed oscillator stopped |

**Notes 1.** The value of this register is 00H immediately after a reset release but automatically changes to 80H after internal high-speed oscillator has been stabilized.

**2.** Bit 7 is read-only.

**Caution When setting RSTOP to 1, be sure to confirm that the CPU operates with a clock other than the internal high-speed oscillation clock. Specifically, set under either of the following conditions.**

**<1> 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L**

- **When MCS = 1 (when CPU operates with the high-speed system clock)**

**<2> 78K0/KC2-L**

- **When MCS = 1 (when CPU operates with the high-speed system clock)**
- **When CLS = 1 (when CPU operates with the subsystem clock)**

**In addition, stop peripheral hardware that is operating on the internal high-speed oscillation clock before setting RSTOP to 1.**

**(5) Main OSC control register (MOC)**

This register selects the operation mode of the high-speed system clock.

This register is used to stop the X1 oscillator or to disable an external clock input from the EXCLK pin when the CPU operates with a clock other than the high-speed system clock.

MOC can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 80H.

**Figure 5-8. Format of Main OSC Control Register (MOC)**

Address: FFA2H After reset: 80H R/W

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

| MSTOP | Control of high-speed system clock operation | |
|---|---|---|
| | X1 oscillation mode | External clock input mode |
| 0 | X1 oscillator operating | External clock from EXCLK pin is enabled |
| 1 | X1 oscillator stopped | External clock from EXCLK pin is disabled |

**Cautions 1. Clear MSTOP to 0 while the regulator mode control register (RMC) is 00H.**

**2. When setting MSTOP to 1, be sure to confirm that the CPU operates with a clock other than the high-speed system clock. Specifically, set under either of the following conditions.**

**<1> 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L**

- **When MCS = 0 (when CPU operates with the internal high-speed oscillation clock)**

**<2> 78K0/KC2-L**

- **When MCS = 0 (when CPU operates with the internal high-speed oscillation clock)**
- **When CLS = 1 (when CPU operates with the subsystem clock)**

**In addition, stop peripheral hardware that is operating on the high-speed system clock before setting MSTOP to 1.**

**3. Do not clear MSTOP to 0 while bit 6 (OSCSEL) of the clock operation mode select register (OSCCTL) is 0 (input port mode).**

**4. The peripheral hardware cannot operate when the peripheral hardware clock is stopped. To resume the operation of the peripheral hardware after the peripheral hardware clock has been stopped, initialize the peripheral hardware.**

**(6) Main clock mode register (MCM)**

This register selects the main system clock supplied to CPU clock and clock supplied to peripheral hardware clock.

MCM can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 5-9. Format of Main Clock Mode Register (MCM)**

Address: FFA1H After reset: 00H R/W[Note]

| Symbol | 7 | 6 | 5 | 4 | 3 | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MCM | 0 | 0 | 0 | 0 | 0 | XSEL | MCS | MCM0 |

| XSEL | MCM0 | Selection of clock supplied to main system clock and peripheral hardware | |
|---|---|---|---|
| | | Main system clock ($f_{XP}$) | Peripheral hardware clock ($f_{PRS}$) |
| 0 | 0 | Internal high-speed oscillation clock ($f_{IH}$) | Internal high-speed oscillation clock ($f_{IH}$) |
| 0 | 1 | | |
| 1 | 0 | | High-speed system clock ($f_{XH}$) |
| 1 | 1 | High-speed system clock ($f_{XH}$) | |

| MCS | Main system clock status |
|---|---|
| 0 | Operates with internal high-speed oscillation clock |
| 1 | Operates with high-speed system clock |

**Note** Bit 1 is read-only.

**Cautions 1. XSEL can be changed only once after a reset release.**

**2. Do not rewrite MCM0 when the CPU clock operates with the subsystem clock.**

**3. A clock other than $f_{PRS}$ is supplied to the following peripheral functions regardless of the setting of XSEL and MCM0.**

- **Watchdog timer (operates with internal low-speed oscillation clock)**
- **When "$f_{IL}$", "$f_{IL}/2^6$", or "$f_{IL}/2^{15}$" is selected as the count clock for 8-bit timer H1 (operates with internal low-speed oscillation clock)**
- **Peripheral hardware selects the external clock as the clock source (Except when the external count clock of TM00 is selected (TI000 pin valid edge))**

**(7) Oscillation stabilization time counter status register (OSTC)**

This is the register that indicates the count status of the X1 clock oscillation stabilization time counter. When X1 clock oscillation starts with the internal high-speed oscillation clock or subsystem clock used as the CPU clock, the X1 clock oscillation stabilization time can be checked.

OSTC can be read by a 1-bit or 8-bit memory manipulation instruction.

When reset is released (reset by $\overline{\text{RESET}}$ input, POC, LVI, and WDT), the STOP instruction and MSTOP (bit 7 of MOC register) = 1 clear OSTC to 00H.

**Figure 5-10. Format of Oscillation Stabilization Time Counter Status Register (OSTC)**

Address: FFA3H After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 |

| MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | Oscillation stabilization time status | |
|---|---|---|---|---|---|---|
| | | | | | | fx = 10 MHz |
| 1 | 0 | 0 | 0 | 0 | $2^{11}$/fx min. | 204.8 μs min. |
| 1 | 1 | 0 | 0 | 0 | $2^{13}$/fx min. | 819.2 μs min. |
| 1 | 1 | 1 | 0 | 0 | $2^{14}$/fx min. | 1.64 ms min. |
| 1 | 1 | 1 | 1 | 0 | $2^{15}$/fx min. | 3.27 ms min. |
| 1 | 1 | 1 | 1 | 1 | $2^{16}$/fx min. | 6.55 ms min. |

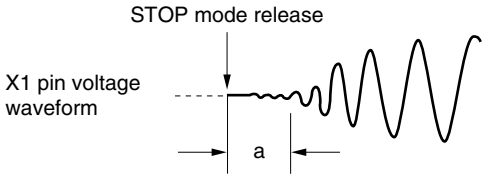
**Cautions 1. After the above time has elapsed, the bits are set to 1 in order from MOST11 and remain 1.**

**2. The oscillation stabilization time counter counts up to the oscillation stabilization time set by OSTS. If the STOP mode is entered and then released while the internal high-speed oscillation clock is being used as the CPU clock, set the oscillation stabilization time as follows.**

- **Desired OSTC oscillation stabilization time ≤ Oscillation stabilization time set by OSTS**

**Note, therefore, that only the status up to the oscillation stabilization time set by OSTS is set to OSTC after STOP mode is released.**

**3. The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).**




**Remark** fx: X1 clock oscillation frequency

**(8) Oscillation stabilization time select register (OSTS)**

This register is used to select the X1 clock oscillation stabilization wait time when the STOP mode is released.

When the X1 clock is selected as the CPU clock, the operation waits for the time set using OSTS after the STOP mode is released.

When the internal high-speed oscillation clock is selected as the CPU clock, confirm with OSTC that the desired oscillation stabilization time has elapsed after the STOP mode is released. The oscillation stabilization time can be checked up to the time set using OSTC.

OSTS can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets OSTS to 05H.

**Figure 5-11. Format of Oscillation Stabilization Time Select Register (OSTS)**

Address: FFA4H After reset: 05H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 |

| OSTS2 | OSTS1 | OSTS0 | Oscillation stabilization time selection | |
|---|---|---|---|---|
| | | | | fx = 10 MHz |
| 0 | 0 | 1 | $2^{11}$/fx | 204.8 μs |
| 0 | 1 | 0 | $2^{13}$/fx | 819.2 μs |
| 0 | 1 | 1 | $2^{14}$/fx | 1.64 ms |
| 1 | 0 | 0 | $2^{15}$/fx | 3.27 ms |
| 1 | 0 | 1 | $2^{16}$/fx | 6.55 ms |
| Other than above | | | Setting prohibited | |

**Cautions 1. To set the STOP mode when the X1 clock is used as the CPU clock, set OSTS before executing the STOP instruction.**

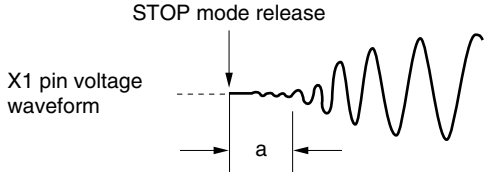
**2. Do not change the value of the OSTS register during the X1 clock oscillation stabilization time.**

**3. The oscillation stabilization time counter counts up to the oscillation stabilization time set by OSTS. If the STOP mode is entered and then released while the internal high-speed oscillation clock is being used as the CPU clock, set the oscillation stabilization time as follows.**

- **Desired OSTC oscillation stabilization time ≤ Oscillation stabilization time set by OSTS**

**Note, therefore, that only the status up to the oscillation stabilization time set by OSTS is set to OSTC after STOP mode is released.**

**4. The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).**




**Remark** fx: X1 clock oscillation frequency

**(9) Peripheral enable register 0 (PER0)[Note]**

<R> This register controls the clock supplied to peripheral functions other than the real-time counter. By stopping the clock supplied to such peripheral functions, the power consumption can be reduced.

PER0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Note** 78K0/KC2-L only

**Figure 5-12. Format of Peripheral Enable Register 0 (PER0)**

Address: FF25H After reset: 00H R/W

<R>

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PER0 | RTCEN | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| RTCEN | Control of real-time counter (RTC) input clock supply |
|---|---|
| 0 | Sub HALT low power consumption mode |
| 1 | Sub HALT normal mode[Note] |

**Note** To output the subsystem clock by using the PCL function while in the subsystem clock HALT mode, set RTCEN to 1.

**Caution Be sure to clear bits 0 to 6 of PER0 to "0".**

## 5.4 System Clock Oscillator

### 5.4.1 X1 oscillator

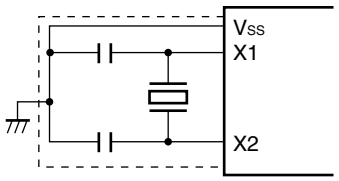
The X1 oscillator oscillates with a crystal resonator or ceramic resonator (1 to 10 MHz) connected to the X1 and X2 pins.

An external clock can also be input. In this case, input the clock signal to the EXCLK pin.

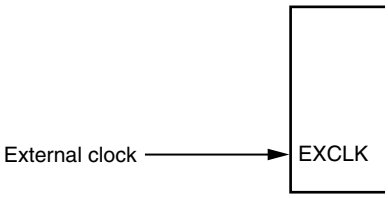
Figure 5-13 shows an example of the external circuit of the X1 oscillator.

**Figure 5-13. Example of External Circuit of X1 Oscillator**

**(a) Crystal or ceramic oscillation**




**(b) External clock**




Cautions are listed on the next page.

### 5.4.2 XT1 oscillator

The XT1 oscillator[Note] oscillates with a crystal resonator (standard: 32.768 kHz) connected to the XT1 and XT2 pins.

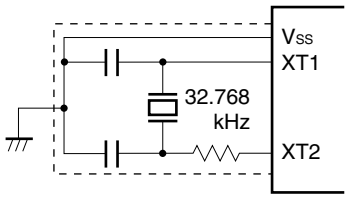
An external clock can also be input. In this case, input the clock signal to the EXCLKS pin.

Figure 5-14 shows an example of the external circuit of the XT1 oscillator.

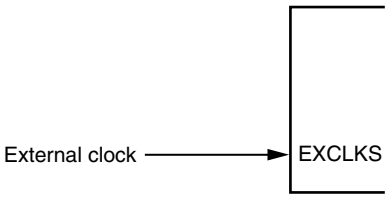
**Note** 78K0/KC2-L only

**Figure 5-14. Example of External Circuit of XT1 Oscillator**

**(a) Crystal oscillation**




**(b) External clock**




Cautions are listed on the next page.

**Caution 1. When using the X1 oscillator and XT1 oscillator, wire as follows in the area enclosed by the broken lines in the Figures 5-13 and 5-14 to avoid an adverse effect from wiring capacitance.**

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as $V_{SS}$. Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

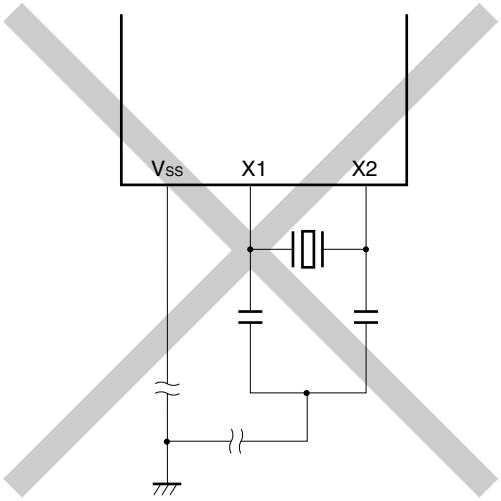
**The XT1 oscillator is a circuit with low amplification in order to achieve low-power consumption. Note the following points when designing the circuit.**

- **Pins and circuit boards include parasitic capacitance. Therefore, perform oscillation evaluation using a circuit board to be actually used and confirm that there are no problems.**
- **Use the recommended resonator, which will be described in CHAPTER 28 ELECTRICAL SPECIFICATIONS after it is evaluated, when using the XT1 oscillator in the ultra-low power consumption oscillation (RSWOSC = 1).**
- **Make the wiring between the XT1 and XT2 pins and the resonators as short as possible, and minimize the parasitic capacitance and wiring resistance. Note this particularly when the ultra-low power consumption oscillation (RSWOSC = 1) is selected.**
- **Configure the circuit of the circuit board, using material with little wiring resistance.**
- **Place a ground pattern that has the same potential as $V_{SS}$ as much as possible near the XT1 oscillator.**
- **Be sure that the signal lines between the XT1 and XT2 pins, and the resonators do not cross with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **The impedance between the XT1 and XT2 pins may drop and oscillation may be disturbed due to moisture absorption of the circuit board in a high-humidity environment or dew condensation on the board. When using the circuit board in such an environment, take measures to damp-proof the circuit board, such as by coating.**
- **When coating the circuit board, use material that does not cause capacitance or leakage between the XT1 and XT2 pins.**

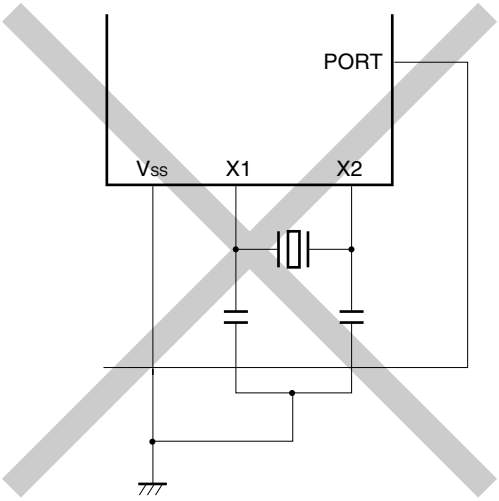
Figure 5-15 shows examples of incorrect resonator connection.

**Figure 5-15. Examples of Incorrect Resonator Connection (1/2)**

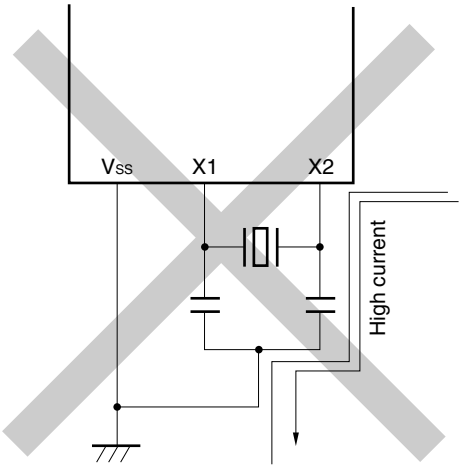
**(a) Too long wiring**



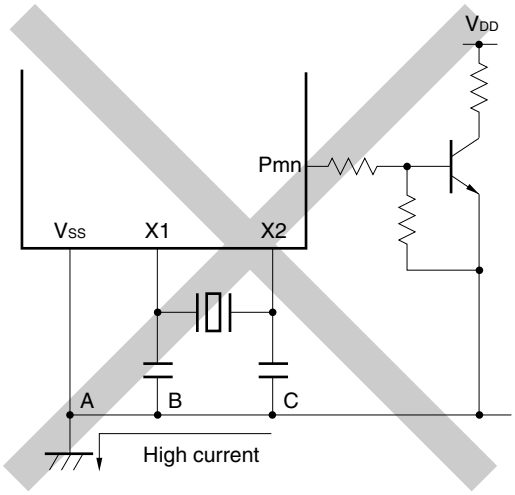

**(b) Crossed signal line**




**(c) Wiring near high alternating current**



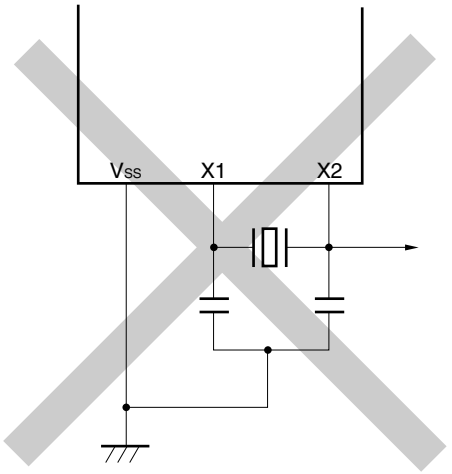

**(d) Current flowing through ground line of oscillator (potential at points A, B, and C fluctuates)**




**Remark** When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

**Figure 5-15. Examples of Incorrect Resonator Connection (2/2)**

**(e) Signals are fetched**




**Remark** When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

**Caution 2. When X2 and XT1 are wired in parallel, the crosstalk noise of X2 may increase with XT1, resulting in malfunctioning.**

### 5.4.3 When subsystem clock is not used

If it is not necessary to use the subsystem clock[Note] for low power consumption operations, watch operations, etc., or if not using the subsystem clock as an Input port, set the XT1 and XT2 pins to Input mode (OSCSELS = 0) and independently connect to $V_{DD}$ or $V_{SS}$ via a resistor.

**Note** 78K0/KC2-L only

### 5.4.4 Internal high-speed oscillator

The internal high-speed oscillator is incorporated in the 78K0/Kx2-L microcontrollers. Oscillation can be controlled by the internal oscillation mode register (RCM).

After a reset release, the internal high-speed oscillator automatically starts oscillation.

Internal high-speed oscillation clock frequency (4 MHz (TYP.)/8 MHz (TYP.)) can be set by the option byte.

### 5.4.5 Internal low-speed oscillator

The internal low-speed oscillator is incorporated in the 78K0/Kx2-L microcontrollers.

The internal low-speed oscillation clock is only used as the watchdog timer and the clock of 8-bit timer H1. The internal low-speed oscillation clock cannot be used as the CPU clock.

“Can be stopped by software” or “Cannot be stopped” can be selected by the option byte. When “Can be stopped by software” is set, oscillation can be controlled by the internal oscillation mode register (RCM).

After a reset release, the internal low-speed oscillator automatically starts oscillation, and the watchdog timer is driven (30 kHz (TYP.)) if the watchdog timer operation is enabled using the option byte.

### 5.4.6 Prescaler

The prescaler generates the CPU clock by dividing the main system clock when the main system clock is selected as the clock to be supplied to the CPU.

## 5.5 Clock Generator Operation

The clock generator generates the following clocks and controls the operation modes of the CPU, such as standby mode (refer to **Figures 5-1** and **5-2**).

- Main system clock $f_{XP}$
  - High-speed system clock $f_{XH}$
    X1 clock $f_X$
    External main system clock $f_{EXCLK}$
  - Internal high-speed oscillation clock $f_{IH}$
- Subsystem clock $f_{SUB}$ Note
  - XT1 clock $f_{XT}$
  - External subsystem clock $f_{EXCLKS}$
- Internal low-speed oscillation clock $f_{IL}$
- CPU clock $f_{CPU}$
- Peripheral hardware clock $f_{PRS}$

**Note** 78K0/KC2-L only

The CPU starts operation when the internal high-speed oscillator starts outputting after a reset release in the 78K0/Kx2-L microcontrollers, thus enabling the following.

**(1) Enhancement of security function**

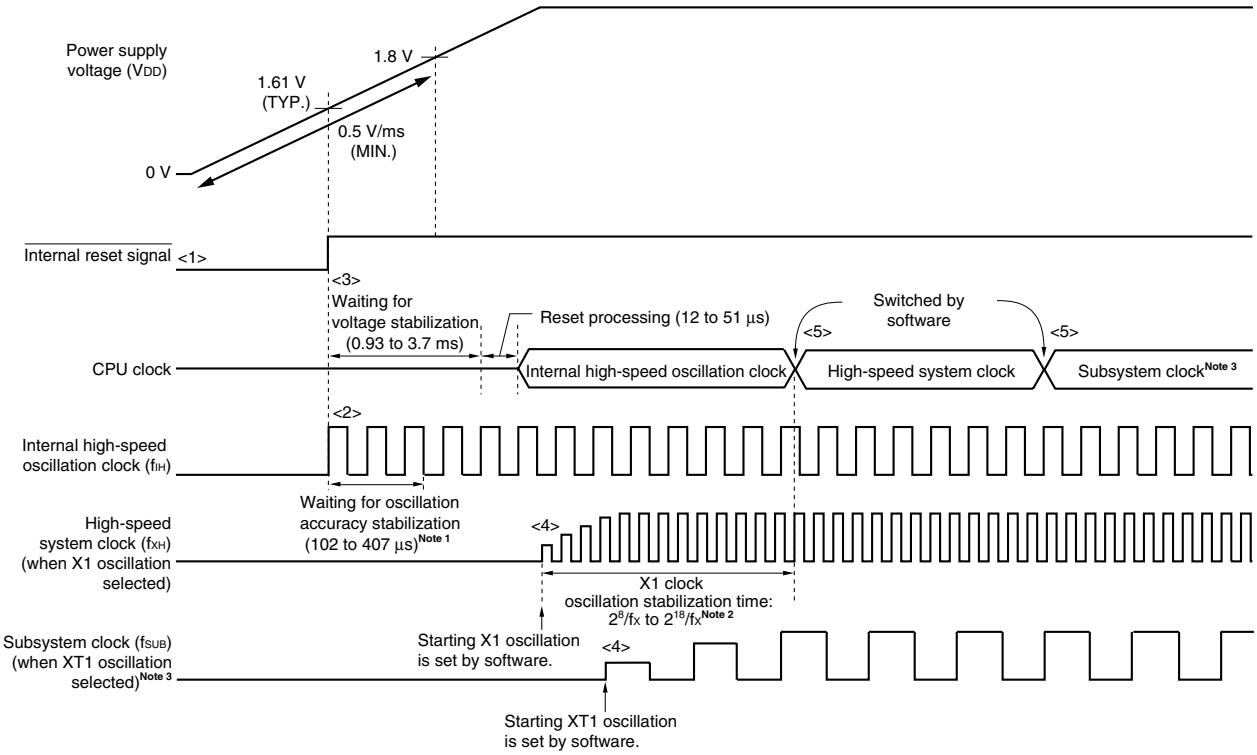
When the X1 clock is set as the CPU clock by the default setting, the device cannot operate if the X1 clock is damaged or badly connected and therefore does not operate after reset is released. However, the start clock of the CPU is the internal high-speed oscillation clock, so the device can be started by the internal high-speed oscillation clock after a reset release. Consequently, the system can be safely shut down by performing a minimum operation, such as acknowledging a reset source by software or performing safety processing when there is a malfunction.

**(2) Improvement of performance**

Because the CPU can be started without waiting for the X1 clock oscillation stabilization time, the total performance can be improved.

When the power supply voltage is turned on, the clock generator operation is shown in Figures 5-16 and 5-17.

**Figure 5-16. Clock Generator Operation When Power Supply Voltage Is Turned On (When LVI Default Start Function Stopped Is Set (Option Byte: LVISTART = 0))**




<1> When the power is turned on, an internal reset signal is generated by the power-on-clear (POC) circuit.

<2> When the power supply voltage exceeds 1.61 V (TYP.), the reset is released and the internal high-speed oscillator automatically starts oscillation.

<3> When the power supply voltage rises with a slope of 0.5 V/ms (MIN.), the CPU starts operation on the internal high-speed oscillation clock after the reset is released and after the stabilization times for the voltage of the power supply and regulator have elapsed, and then reset processing is performed.

<4> Set the start of oscillation of the X1 or XT1 clock via software (refer to **(1)** in **5.6.1 Example of controlling high-speed system clock** and **(1)** in **5.6.3 Example of controlling subsystem clock**).

<5> When switching the CPU clock to the X1 or XT1 clock, wait for the clock oscillation to stabilize, and then set switching via software (refer to **(3)** in **5.6.1 Example of controlling high-speed system clock** and **(3)** in **5.6.3 Example of controlling subsystem clock**).

**Notes 1.** The internal voltage stabilization time includes the oscillation accuracy stabilization time of the internal high-speed oscillation clock.

**2.** When releasing a reset (above figure) or releasing STOP mode while the CPU is operating on the internal high-speed oscillation clock, confirm the oscillation stabilization time for the X1 clock using the oscillation stabilization time counter status register (OSTC). If the CPU operates on the high-speed system clock (X1 oscillation), set the oscillation stabilization time when releasing STOP mode using the oscillation stabilization time select register (OSTS).

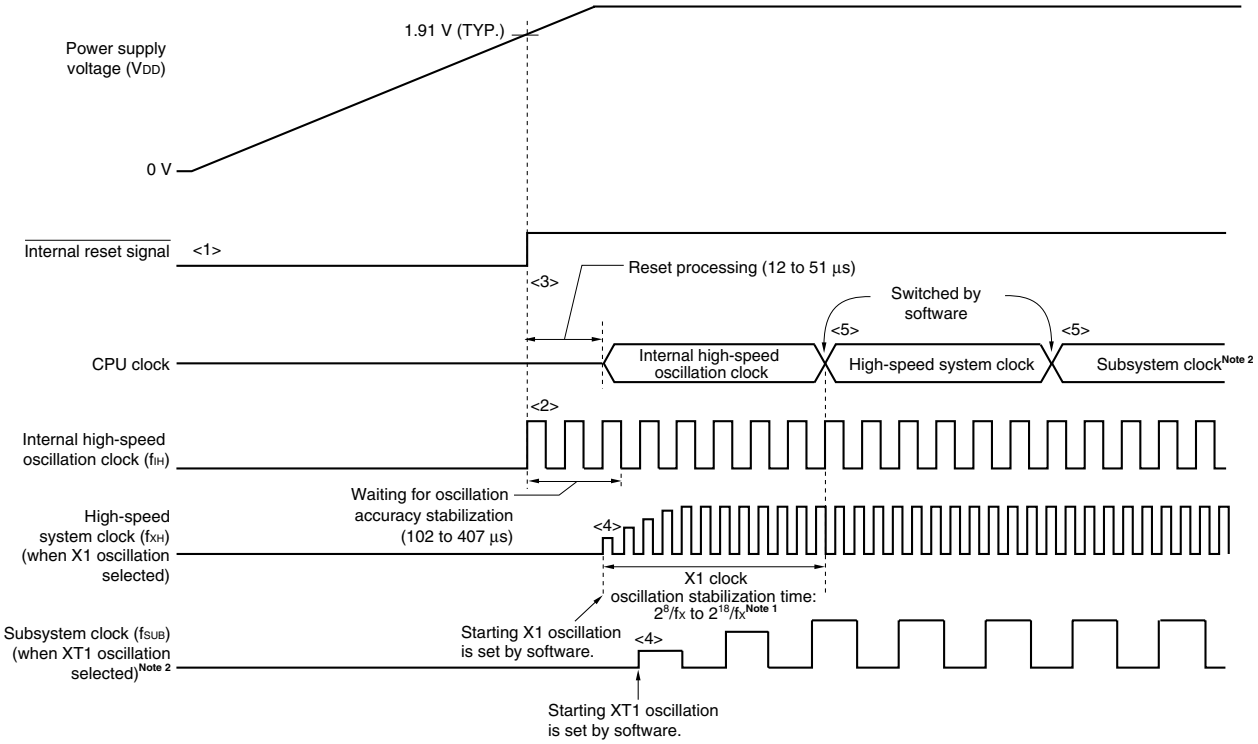
**3.** 78K0/KC2-L only

**Cautions 1. If the voltage rises with a slope of less than 0.5 V/ms (MIN.) from power application until the voltage reaches 1.8 V, input a low level to the $\overline{\text{RESET}}$ pin from power application until the voltage reaches 1.8 V, or set the LVI default start function enabled by using the option byte (LVISTART = 1) (refer to Figure 5-17). When a low level has been input to the $\overline{\text{RESET}}$ pin until the voltage reaches 1.8 V, the CPU operates with the same timing as <2> and thereafter in Figure 5-16, after the reset has been released by the $\overline{\text{RESET}}$ pin.**

**2. It is not necessary to wait for the oscillation stabilization time when an external clock input from the EXCLK and EXCLKS pins is used.**

**Remark** While the microcontroller is operating, a clock that is not used as the CPU clock can be stopped via software settings. The internal high-speed oscillation clock and high-speed system clock can be stopped by executing the STOP instruction (refer to **(4)** in **5.6.1 Example of controlling high-speed system clock**, **(3)** in **5.6.2 Example of controlling internal high-speed oscillation clock**, and **(4)** in **5.6.3 Example of controlling subsystem clock**).

**Figure 5-17. Clock Generator Operation When Power Supply Voltage Is Turned On (When LVI Default Start Function Enabled Is Set (Option Byte: LVISTART = 1))**




<1> When the power is turned on, an internal reset signal is generated by the power-on-clear (POC) circuit.

<2> When the power supply voltage exceeds 1.91 V (TYP.), the reset is released and the internal high-speed oscillator automatically starts oscillation.

<3> After the reset is released and reset processing is performed, the CPU starts operation on the internal high-speed oscillation clock.

<4> Set the start of oscillation of the X1 or XT1 clock via software (refer to **(1)** in **5.6.1 Example of controlling high-speed system clock** and **(1)** in **5.6.3 Example of controlling subsystem clock**).

<5> When switching the CPU clock to the X1 or XT1 clock, wait for the clock oscillation to stabilize, and then set switching via software (refer to **(3)** in **5.6.1 Example of controlling high-speed system clock** and **(3)** in **5.6.3 Example of controlling subsystem clock**).

**Notes 1.** When releasing a reset (above figure) or releasing STOP mode while the CPU is operating on the internal high-speed oscillation clock, confirm the oscillation stabilization time for the X1 clock using the oscillation stabilization time counter status register (OSTC). If the CPU operates on the high-speed system clock (X1 oscillation), set the oscillation stabilization time when releasing STOP mode using the oscillation stabilization time select register (OSTS).

**2.** 78K0/KC2-L only

**Cautions 1. A voltage oscillation stabilization time (0.93 to 3.7 ms) is required after the supply voltage reaches 1.61 V (TYP.). If the supply voltage rises from 1.61 V (TYP.) to 1.91 V (TYP.) within the power supply oscillation stabilization time, the power supply oscillation stabilization time is automatically generated before reset processing.**

**2. It is not necessary to wait for the oscillation stabilization time when an external clock input from the EXCLK and EXCLKS pins is used.**

**Remark** While the microcontroller is operating, a clock that is not used as the CPU clock can be stopped via software settings. The internal high-speed oscillation clock and high-speed system clock can be stopped by executing the STOP instruction (refer to **(4)** in **5.6.1 Example of controlling high-speed system clock**, **(3)** in **5.6.2 Example of controlling internal high-speed oscillation clock**, and **(4)** in **5.6.3 Example of controlling subsystem clock**).

## 5.6 Controlling Clock

### 5.6.1 Example of controlling high-speed system clock

The following two types of high-speed system clocks are available.

- X1 clock: Crystal/ceramic resonator is connected across the X1 and X2 pins.
- External main system clock: External clock is input to the EXCLK pin.

When the high-speed system clock is not used, the X1/P121 and X2/EXCLK/P122 pins can be used as input port pins.

**Caution The X1/P121 and X2/EXCLK/P122 pins are in the input port mode after a reset release.**

The following describes examples of setting procedures for the following cases.

(1) When oscillating X1 clock

(2) When using external main system clock

(3) When using high-speed system clock as CPU clock and peripheral hardware clock

(4) When stopping high-speed system clock

**(1) Example of setting procedure when oscillating the X1 clock**

<1> Setting P121/X1 and P122/X2/EXCLK pins and selecting X1 clock or external clock (OSCCTL register)
When EXCLK is cleared to 0 and OSCSEL is set to 1, the mode is switched from port mode to X1 oscillation mode.

| EXCLK | OSCSEL | Operation Mode of High-Speed System Clock Pin | P121/X1 Pin | P122/X2/EXCLK Pin |
|---|---|---|---|---|
| 0 | 1 | X1 oscillation mode | Crystal/ceramic resonator connection | |

<2> Controlling oscillation of X1 clock (MOC register)
If MSTOP is cleared to 0, the X1 oscillator starts oscillating.

<3> Waiting for the stabilization of the oscillation of X1 clock
Check the OSTC register and wait for the necessary time.
During the wait time, other software processing can be executed with the internal high-speed oscillation clock.

**Cautions 1. Do not change the value of EXCLK and OSCSEL while the X1 clock is operating.**
**2. Set the X1 clock after the supply voltage has reached the operable voltage of the clock to be used (refer to CHAPTER 28 ELECTRICAL SPECIFICATIONS).**

**(2) Example of setting procedure when using the external main system clock**

<1> Setting P121/X1 and P122/X2/EXCLK pins and selecting operation mode (OSCCTL register)
When EXCLK and OSCSEL are set to 1, the mode is switched from port mode to external clock input mode.

| EXCLK | OSCSEL | Operation Mode of High-Speed System Clock Pin | P121/X1 Pin | P122/X2/EXCLK Pin |
|---|---|---|---|---|
| 1 | 1 | External clock input mode | Input port | External clock input |

<2> Controlling external main system clock input (MOC register)
When MSTOP is cleared to 0, the input of the external main system clock is enabled.

**Cautions 1. Do not change the value of EXCLK and OSCSEL while the external main system clock is operating.**
**2. Set the external main system clock after the supply voltage has reached the operable voltage of the clock to be used (refer to CHAPTER 28 ELECTRICAL SPECIFICATIONS).**

**(3) Example of setting procedure when using high-speed system clock as CPU clock and peripheral hardware clock**

<1> Setting high-speed system clock oscillation[Note]
(Refer to **5.6.1 (1) Example of setting procedure when oscillating the X1 clock** and **(2) Example of setting procedure when using the external main system clock.**)

**Note** The setting of <1> is not necessary when high-speed system clock is already operating.

<2> Setting the high-speed system clock as the main system clock (MCM register)

When XSEL and MCM0 are set to 1, the high-speed system clock is supplied as the main system clock and peripheral hardware clock.

| XSEL | MCM0 | Selection of Main System Clock and Clock Supplied to Peripheral Hardware | |
|---|---|---|---|
| | | Main System Clock ($f_{XP}$) | Peripheral Hardware Clock ($f_{PRS}$) |
| 1 | 1 | High-speed system clock ($f_{XH}$) | High-speed system clock ($f_{XH}$) |

**Caution If the high-speed system clock is selected as the main system clock, a clock other than the high-speed system clock cannot be set as the peripheral hardware clock.**

<3> Setting the main system clock as the CPU clock and selecting the division ratio (PCC register)

When CSS is cleared to 0, the main system clock is supplied to the CPU. To select the CPU clock division ratio, use PCC0, PCC1, and PCC2.

| CSS | PCC2 | PCC1 | PCC0 | CPU Clock ($f_{CPU}$) Selection |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $f_{XP}$ |
| | 0 | 0 | 1 | $f_{XP}$/2 (default) |
| | 0 | 1 | 0 | $f_{XP}/2^2$ |
| | 0 | 1 | 1 | $f_{XP}/2^3$ |
| | 1 | 0 | 0 | $f_{XP}/2^4$ |
| | Other than above | | | Setting prohibited |

### (4) Example of setting procedure when stopping the high-speed system clock

The high-speed system clock can be stopped in the following two ways.

- Executing the STOP instruction and stopping the X1 oscillation (disabling clock input if the external clock is used)
- Setting MSTOP to 1 and stopping the X1 oscillation (disabling clock input if the external clock is used)

#### (a) To execute a STOP instruction

<1> Setting to stop peripheral hardware

Stop peripheral hardware that cannot be used in the STOP mode (for peripheral hardware that cannot be used in STOP mode, refer to **CHAPTER 19 STANDBY FUNCTION**).

<2> Setting the X1 clock oscillation stabilization time after standby release

When the CPU is operating on the X1 clock, set the value of the OSTS register before the STOP instruction is executed.

<3> Executing the STOP instruction

When the STOP instruction is executed, the system is placed in the STOP mode and X1 oscillation is stopped (the input of the external clock is disabled).

**(b) To stop X1 oscillation (disabling external clock input) by setting MSTOP to 1**

<1> Confirming the CPU clock status (PCC and MCM registers)

Confirm with CLS and MCS that the CPU is operating on a clock other than the high-speed system clock. When CLS = 0 and MCS = 1, the high-speed system clock is supplied to the CPU, so change the CPU clock to a clock other than the high-speed system clock.

- 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L

| MCS | CPU Clock Status |
|---|---|
| 0 | Internal high-speed oscillation clock |
| 1 | High-speed system clock |

- 78K0/KC2-L

| CLS | MCS | CPU Clock Status |
|---|---|---|
| 0 | 0 | Internal high-speed oscillation clock |
| 0 | 1 | High-speed system clock |
| 1 | × | Subsystem clock |

<2> Stopping the high-speed system clock (MOC register)

When MSTOP is set to 1, X1 oscillation is stopped (the input of the external clock is disabled).

**Caution Be sure to confirm that MCS = 0 or CLS = 1 when setting MSTOP to 1. In addition, stop peripheral hardware that is operating on the high-speed system clock.**

### 5.6.2 Example of controlling internal high-speed oscillation clock

The following describes examples of clock setting procedures for the following cases.

(1) When restarting oscillation of the internal high-speed oscillation clock
(2) When using internal high-speed oscillation clock as CPU clock, and internal high-speed oscillation clock or high-speed system clock as peripheral hardware clock
(3) When stopping the internal high-speed oscillation clock

**(1) Example of setting procedure when restarting oscillation of the internal high-speed oscillation clock[Note 1]**

<1> Setting restart of oscillation of the internal high-speed oscillation clock (RCM register)

When RSTOP is cleared to 0, the internal high-speed oscillation clock starts operating.

<2> Waiting for the oscillation accuracy stabilization time of internal high-speed oscillation clock (RCM register)

Wait until RSTS is set to 1[Note 2].

**Notes 1.** After a reset release, the internal high-speed oscillator automatically starts oscillating and the internal high-speed oscillation clock is selected as the CPU clock.

**2.** This wait time is not necessary if high accuracy is not necessary for the CPU clock and peripheral hardware clock.

**(2) Example of setting procedure when using internal high-speed oscillation clock as CPU clock, and internal high-speed oscillation clock or high-speed system clock as peripheral hardware clock**

<1> • Restarting oscillation of the internal high-speed oscillation clock[Note]
(Refer to **5.6.2 (1) Example of setting procedure when restarting oscillation of the internal high-speed oscillation clock**).

• Oscillating the high-speed system clock[Note]
(This setting is required when using the high-speed system clock as the peripheral hardware clock. Refer to **5.6.1 (1) Example of setting procedure when oscillating the X1 clock** and **(2) Example of setting procedure when using the external main system clock.**)

**Note** The setting of <1> is not necessary when the internal high-speed oscillation clock or high-speed system clock is already operating.

<2> Selecting the clock supplied as the main system clock and peripheral hardware clock (MCM register)
Set the main system clock and peripheral hardware clock using XSEL and MCM0.

| XSEL | MCM0 | Selection of Main System Clock and Clock Supplied to Peripheral Hardware | |
|---|---|---|---|
| | | Main System Clock ($f_{XP}$) | Peripheral Hardware Clock ($f_{PRS}$) |
| 0 | 0 | Internal high-speed oscillation clock ($f_{IH}$) | Internal high-speed oscillation clock ($f_{IH}$) |
| 0 | 1 | | |
| 1 | 0 | | High-speed system clock ($f_{XH}$) |

<3> Selecting the CPU clock division ratio (PCC register)
When CSS is cleared to 0, the main system clock is supplied to the CPU. To select the CPU clock division ratio, use PCC0, PCC1, and PCC2.

| CSS | PCC2 | PCC1 | PCC0 | CPU Clock ($f_{CPU}$) Selection |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $f_{XP}$ |
| | 0 | 0 | 1 | $f_{XP}/2$ (default) |
| | 0 | 1 | 0 | $f_{XP}/2^2$ |
| | 0 | 1 | 1 | $f_{XP}/2^3$ |
| | 1 | 0 | 0 | $f_{XP}/2^4$ |
| | Other than above | | | Setting prohibited |

**(3) Example of setting procedure when stopping the internal high-speed oscillation clock**

The internal high-speed oscillation clock can be stopped in the following two ways.

- Executing the STOP instruction to set the STOP mode
- Setting RSTOP to 1 and stopping the internal high-speed oscillation clock

**(a) To execute a STOP instruction**

<1> Setting of peripheral hardware

Stop peripheral hardware that cannot be used in the STOP mode (for peripheral hardware that cannot be used in STOP mode, refer to **CHAPTER 19 STANDBY FUNCTION**).

<2> Setting the X1 clock oscillation stabilization time after standby release

When the CPU is operating on the X1 clock, set the value of the OSTS register before the STOP instruction is executed. To operate the CPU immediately after the STOP mode has been released, set MCM0 to 0, switch the CPU clock to the internal high-speed oscillation clock, and check that RSTS is 1.

<3> Executing the STOP instruction

When the STOP instruction is executed, the system is placed in the STOP mode and internal high-speed oscillation clock is stopped.

**(b) To stop internal high-speed oscillation clock by setting RSTOP to 1**

<1> Confirming the CPU clock status (PCC and MCM registers)

Confirm with CLS and MCS that the CPU is operating on a clock other than the internal high-speed oscillation clock.

When CLS = 0 and MCS = 0, the internal high-speed oscillation clock is supplied to the CPU, so change the CPU clock to a clock other than the internal high-speed oscillation clock.

- 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L

| MCS | CPU Clock Status |
|---|---|
| 0 | Internal high-speed oscillation clock |
| 1 | High-speed system clock |

- 78K0/KC2-L

| CLS | MCS | CPU Clock Status |
|---|---|---|
| 0 | 0 | Internal high-speed oscillation clock |
| 0 | 1 | High-speed system clock |
| 1 | × | Subsystem clock |

<2> Stopping the internal high-speed oscillation clock (RCM register)

When RSTOP is set to 1, internal high-speed oscillation clock is stopped.

**Caution Be sure to confirm that MCS = 1 or CLS = 1 when setting RSTOP to 1. In addition, stop peripheral hardware that is operating on the internal high-speed oscillation clock.**

### 5.6.3 Example of controlling subsystem clock

The following two types of subsystem clocks[Note] are available.

- XT1 clock: Crystal/ceramic resonator is connected across the XT1 and XT2 pins.
- External subsystem clock: External clock is input to the EXCLKS pin.

When the subsystem clock is not used, the XT1/P123 and XT2/EXCLKS/P124 pins can be used as input port pins.

**Note** 78K0/KC2-L only

**Cautions 1. The XT1/P123 and XT2/EXCLKS/P124 pins are in the input port mode after a reset release.**

**2. Do not start the peripheral hardware operation with the external clock from peripheral hardware pins when the internal high-speed oscillation clock and high-speed system clock are stopped while the CPU operates with the subsystem clock, or when in the STOP mode.**

The following describes examples of setting procedures for the following cases.

(1) When oscillating XT1 clock
(2) When using external subsystem clock
(3) When using subsystem clock as CPU clock
(4) When stopping subsystem clock

#### (1) Example of setting procedure when oscillating the XT1 clock

<1> Setting XT1 and XT2 pins and selecting operation mode (PCC and OSCCTL registers)
When a value is specified for XTSTART and EXCLKS and OSCSELS are set to the values below, the system switches from the port mode to the XT1 oscillation mode. set as any of the following, the mode is switched from port mode to XT1 oscillation mode.

| XTSTART | EXCLKS | OSCSELS | Operation Mode of Subsystem Clock Pin | P123/XT1 Pin | P124/XT2/ EXCLKS Pin |
|---|---|---|---|---|---|
| 0 | 0 | 1 | XT1 oscillation mode | Crystal/ceramic resonator connection | |
| 1 | × | × | | | |

**Remark** ×: don't care

<2> Waiting for the stabilization of the subsystem clock oscillation
Wait for the oscillation stabilization time of the subsystem clock by software, using a timer function.

**Caution Do not change the value of XTSTART, EXCLKS, and OSCSELS while the subsystem clock is operating.**

#### (2) Example of setting procedure when using the external subsystem clock

<1> Setting XT1 and XT2 pins, selecting XT1 clock/external clock and controlling oscillation (PCC and OSCCTL registers)
When XTSTART is cleared to 0 and EXCLKS and OSCSELS are set to 1, the mode is switched from port mode to external clock input mode. In this case, input the external clock to the EXCLKS/XT2/P124 pins.

| XTSTART | EXCLKS | OSCSELS | Operation Mode of Subsystem Clock Pin | P123/XT1 Pin | P124/XT2/ EXCLKS Pin |
|---|---|---|---|---|---|
| 0 | 1 | 1 | External clock input mode | Input port | External clock input |

**Caution Do not change the value of XTSTART, EXCLKS, and OSCSELS while the subsystem clock is operating.**

**(3) Example of setting procedure when using the subsystem clock as the CPU clock**

<1> Setting subsystem clock oscillation[Note]

(Refer to **5.6.3 (1) Example of setting procedure when oscillating the XT1 clock** and **(2) Example of setting procedure when using the external subsystem clock**.)

**Note** The setting of <1> is not necessary when while the subsystem clock is operating.

<2> Switching the CPU clock (PCC register)

When CSS is set to 1, the subsystem clock is supplied to the CPU.

| CSS | PCC2 | PCC1 | PCC0 | CPU Clock ($f_{CPU}$) Selection |
|---|---|---|---|---|
| 1 | 0 | 0 | 0 | $f_{SUB}$ |
| | 0 | 0 | 1 | |
| | 0 | 1 | 0 | |
| | 0 | 1 | 1 | |
| | 1 | 0 | 0 | |
| | Other than above | | | Setting prohibited |

**(4) Example of setting procedure when stopping the subsystem clock**

<1> Confirming the CPU clock status (PCC and MCM registers)

Confirm with CLS and MCS that the CPU is operating on a clock other than the subsystem clock.

When CLS = 1, the subsystem clock is supplied to the CPU, so change the CPU clock to a clock other than the subsystem clock.

| CLS | MCS | CPU Clock Status |
|---|---|---|
| 0 | 0 | Internal high-speed oscillation clock |
| 0 | 1 | High-speed system clock |
| 1 | × | Subsystem clock |

<2> Stopping the subsystem clock (OSCCTL register)

When OSCSELS is cleared to 0, XT1 oscillation is stopped (the input of the external clock is disabled).

**Cautions 1. Be sure to confirm that CLS = 0 when clearing OSCSELS to 0. In addition, stop the watch timer if it is operating on the subsystem clock.**

**2. The subsystem clock oscillation cannot be stopped using the STOP instruction.**

### 5.6.4 Example of controlling internal low-speed oscillation clock

The internal low-speed oscillation clock cannot be used as the CPU clock.

Only the following peripheral hardware can operate with this clock.

- Watchdog timer
- 8-bit timer H1 (if $f_{IL}$ is selected as the count clock)

In addition, the following operation modes can be selected by the option byte.

- Internal low-speed oscillator cannot be stopped
- Internal low-speed oscillator can be stopped by software

The internal low-speed oscillator automatically starts oscillation after a reset release, and the watchdog timer is driven (30 kHz (TYP.)) if the watchdog timer operation has been enabled by the option byte.

**(1) Example of setting procedure when stopping the internal low-speed oscillation clock**

<1> Setting LSRSTOP to 1 (RCM register)

When LSRSTOP is set to 1, the internal low-speed oscillation clock is stopped.

**(2) Example of setting procedure when restarting oscillation of the internal low-speed oscillation clock**

<1> Clearing LSRSTOP to 0 (RCM register)

When LSRSTOP is cleared to 0, the internal low-speed oscillation clock is restarted.

**Caution If "Internal low-speed oscillator cannot be stopped" is selected by the option byte, oscillation of the internal low-speed oscillation clock cannot be controlled.**

### 5.6.5 Clocks supplied to CPU and peripheral hardware

The following table shows the relation among the clocks supplied to the CPU and peripheral hardware, and setting of registers.

**Table 5-4. Clocks Supplied to CPU and Peripheral Hardware, and Register Setting (78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L)**

| Supplied Clock | | XSEL | MCM0 | EXCLK |
|---|---|---|---|---|
| Clock Supplied to CPU | Clock Supplied to Peripheral Hardware | | | |
| Internal high-speed oscillation clock | | 0 | × | × |
| Internal high-speed oscillation clock | X1 clock | 1 | 0 | 0 |
| | External main system clock | 1 | 0 | 1 |
| X1 clock | | 1 | 1 | 0 |
| External main system clock | | 1 | 1 | 1 |

**Remarks 1.** The 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L are not provided with a subsystem clock.

**2.** XSEL: Bit 2 of the main clock mode register (MCM)
MCM0: Bit 0 of MCM
EXCLK: Bit 7 of the clock operation mode select register (OSCCTL)
×: don't care

**Table 5-5. Clocks Supplied to CPU and Peripheral Hardware, and Register Setting (78K0/KC2-L)**

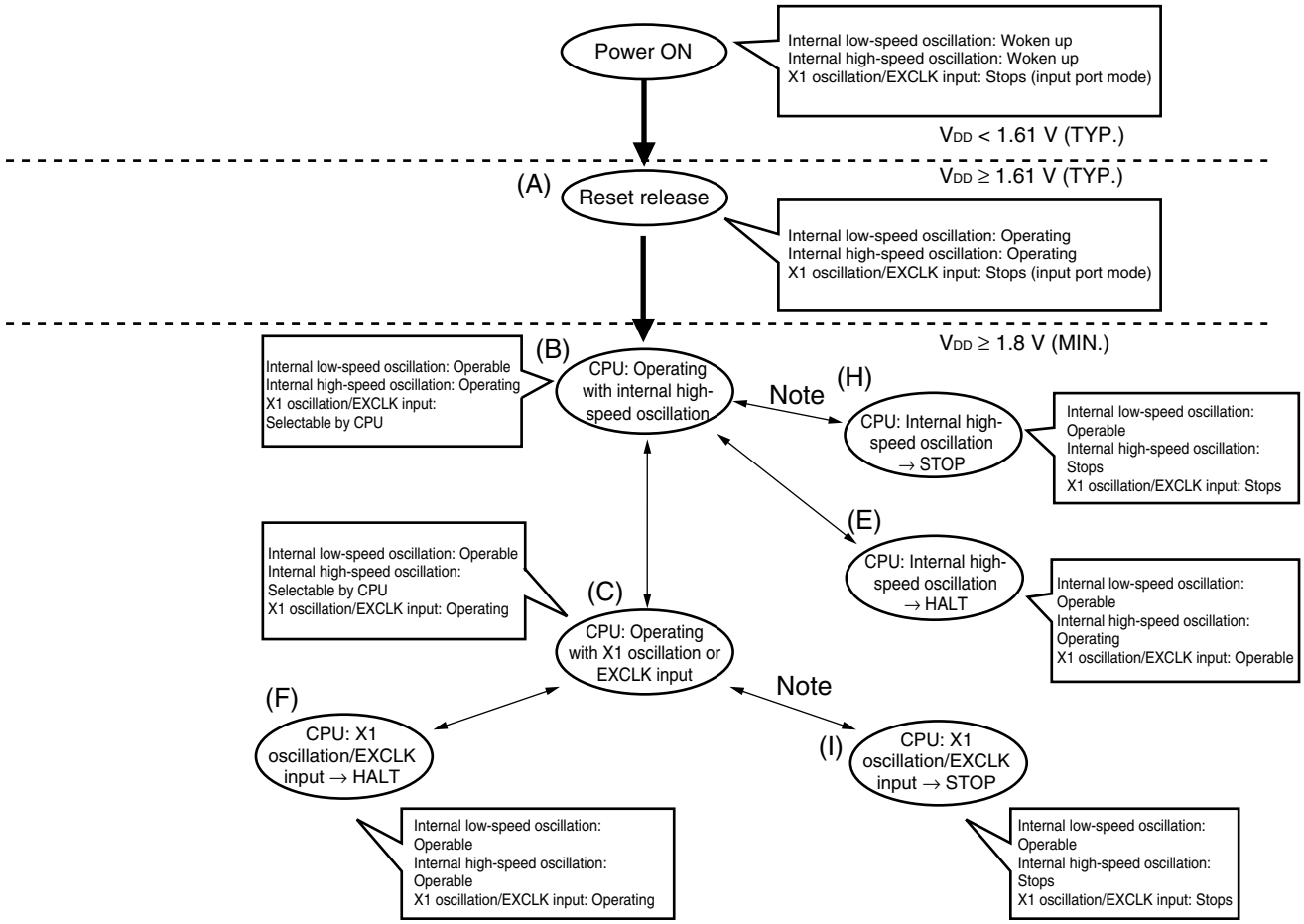
| Supplied Clock | | XSEL | CSS | MCM0 | EXCLK |
|---|---|---|---|---|---|
| Clock Supplied to CPU | Clock Supplied to Peripheral Hardware | | | | |
| Internal high-speed oscillation clock | | 0 | 0 | × | × |
| Internal high-speed oscillation clock | X1 clock | 1 | 0 | 0 | 0 |
| | External main system clock | 1 | 0 | 0 | 1 |
| X1 clock | | 1 | 0 | 1 | 0 |
| External main system clock | | 1 | 0 | 1 | 1 |
| Subsystem clock | Internal high-speed oscillation clock | 0 | 1 | × | × |
| | X1 clock | 1 | 1 | 0 | 0 |
| | | 1 | 1 | 1 | 0 |
| | External main system clock | 1 | 1 | 0 | 1 |
| | | 1 | 1 | 1 | 1 |

**Remark** XSEL: Bit 2 of the main clock mode register (MCM)
CSS: Bit 4 of the processor clock control register (PCC)
MCM0: Bit 0 of MCM
EXCLK: Bit 7 of the clock operation mode select register (OSCCTL)
×: don't care

### 5.6.6 CPU clock status transition diagram

Figures 5-18 and 5-19 show the CPU clock status transition diagram of this product.

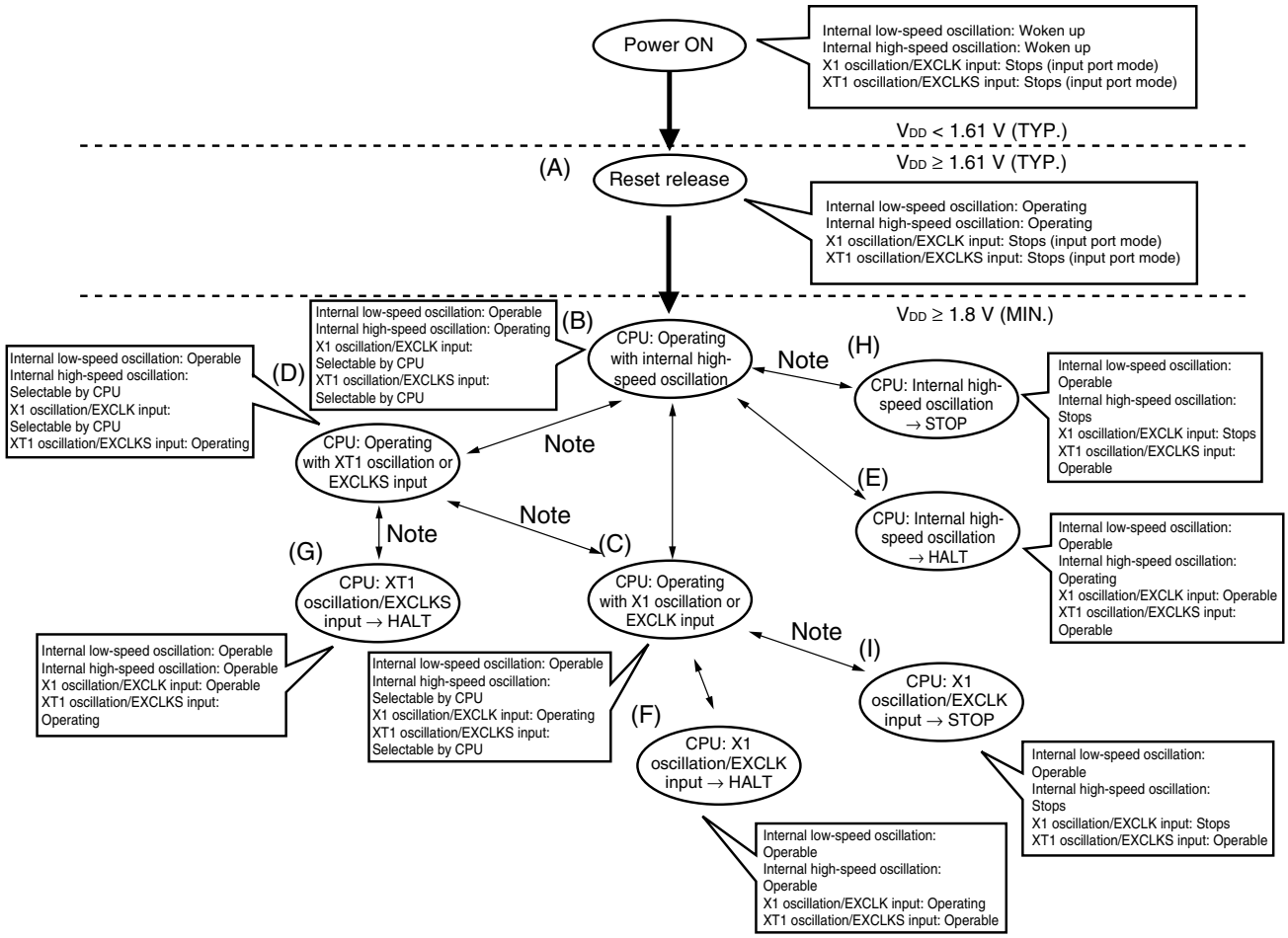
**Figure 5-18. CPU Clock Status Transition Diagram (When LVI Default Start Mode Function Stopped Is Set (Option Byte: LVISTART = 0), 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L)**




<R> **Note** When transitioning to the STOP mode, it is possible to achieve low power consumption by setting RMC = 56H first.

**Remark** When LVI default start function enabled is set (option byte: LVISTART = 1), the CPU clock status changes to (A) in the above figure when the supply voltage exceeds 1.91 V (TYP.), and to (B) after reset processing (12 to 51 $\mu$s).

**Figure 5-19. CPU Clock Status Transition Diagram (When LVI Default Start Mode Function Stopped Is Set (Option Byte: LVISTART = 0), 78K0/KC2-L)**




<R> **Note** When transitioning to the STOP mode, subsystem clock operation mode, and subsystem clock HALT mode, it is possible to achieve low power consumption by setting RMC = 56H first.

**Remark** When LVI default start function enabled is set (option byte: LVISTART = 1), the CPU clock status changes to (A) in the above figure when the supply voltage exceeds 1.91 V (TYP.), and to (B) after reset processing (12 to 51 $\mu$s).

Table 5-6 shows transition of the CPU clock and examples of setting the SFR registers.

**Table 5-6. CPU Clock Transition and SFR Register Setting Examples (1/4)**

**(1) CPU operating with internal high-speed oscillation clock (B) after reset release (A)**

| Status Transition | SFR Register Setting |
|---|---|
| (A) → (B) | SFR registers do not have to be set (default status after reset release). |

**(2) CPU operating with high-speed system clock (C) after reset release (A)**

(The CPU operates with the internal high-speed oscillation clock (B) immediately after a reset release.)

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | EXCLK | OSCSEL | MSTOP | OSTC Register | XSEL | MCM0 |
|---|---|---|---|---|---|---|
| (A) → (B) → (C) (X1 clock) | 0 | 1 | 0 | Must be checked | 1 | 1 |
| (A) → (B) → (C) (external main system clock) | 1 | 1 | 0 | Must not be checked | 1 | 1 |

**Caution Set the clock after the supply voltage has reached the operable voltage of the clock to be set (refer to CHAPTER 28 ELECTRICAL SPECIFICATIONS).**

**(3) CPU operating with subsystem clock (D) after reset release (A)[Note]**

(The CPU operates with the internal high-speed oscillation clock (B) immediately after a reset release.)

**Note** 78K0/KC2-L only

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | XTSTART | EXCLKS | OSCSELS | Waiting for Oscillation Stabilization | CSS |
|---|---|---|---|---|---|
| (A) → (B) → (D) (XT1 clock) | 0 | 0 | 1 | Necessary | 1 |
| | 1 | × | × | | |
| (A) → (B) → (D) (external subsystem clock) | 0 | 1 | 1 | Unnecessary | 1 |

**Remarks 1.** (A) to (I) in Table 5-6 correspond to (A) to (I) in Figures 5-18 and 5-19.

**2.** EXCLK, OSCSEL, EXCLKS, OSCSELS: Bits 7 to 4 of the clock operation mode select register (OSCCTL)
MSTOP: Bit 7 of the main OSC control register (MOC)
XSEL, MCM0: Bits 2 and 0 of the main clock mode register (MCM)
XTSTART, CSS: Bits 6 and 4 of the processor clock control register (PCC)
×: Don't care

**Table 5-6. CPU Clock Transition and SFR Register Setting Examples (2/4)**

**(4) CPU clock changing from internal high-speed oscillation clock (B) to high-speed system clock (C)**

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | EXCLK | OSCSEL | MSTOP | OSTC Register | XSEL[Note] | MCM0 |
|---|---|---|---|---|---|---|
| (B) → (C) (X1 clock) | 0 | 1 | 0 | Must be checked | 1 | 1 |
| (B) → (C) (external main system clock) | 1 | 1 | 0 | Must not be checked | 1 | 1 |

Unnecessary if these registers are already set

Unnecessary if the CPU is operating with the high-speed system clock

**Note** The value of this flag can be changed only once after a reset release. This setting is not necessary if it has already been set.

**Caution Set the clock after the supply voltage has reached the operable voltage of the clock to be set (refer to CHAPTER 28 ELECTRICAL SPECIFICATIONS).**

**(5) CPU clock changing from internal high-speed oscillation clock (B) to subsystem clock (D)[Note]**

**Note** 78K0/KC2-L only

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | XTSTART | EXCLKS | OSCSELS | Waiting for Oscillation Stabilization | CSS |
|---|---|---|---|---|---|
| (B) → (D) (XT1 clock) | 0 | 0 | 1 | Necessary | 1 |
| | 1 | × | × | | |
| (B) → (D) (external subsystem clock) | 0 | 1 | 1 | Unnecessary | 1 |

Unnecessary if the CPU is operating with the subsystem clock

**Remarks 1.** (A) to (I) in Table 5-6 correspond to (A) to (I) in Figures 5-18 and 5-19.

**2.** EXCLK, OSCSEL, EXCLKS, OSCSELS: Bits 7 to 4 of the clock operation mode select register (OSCCTL)
MSTOP: Bit 7 of the main OSC control register (MOC)
XSEL, MCM0: Bits 2 and 0 of the main clock mode register (MCM)
XTSTART, CSS: Bits 6 and 4 of the processor clock control register (PCC)
×: Don't care

**Table 5-6. CPU Clock Transition and SFR Register Setting Examples (3/4)**

**(6) CPU clock changing from high-speed system clock (C) to internal high-speed oscillation clock (B)**

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | RSTOP | RSTS | MCM0 |
|---|---|---|---|
| (C) → (B) | 0 | Confirm this flag is 1. | 0 |

Unnecessary if the CPU is operating with the internal high-speed oscillation clock

**(7) CPU clock changing from high-speed system clock (C) to subsystem clock (D)[Note]**

**Note** 78K0/KC2-L only

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | XTSTART | EXCLKS | OSCSELS | Waiting for Oscillation Stabilization | CSS |
|---|---|---|---|---|---|
| (C) → (D) (XT1 clock) | 0 | 0 | 1 | Necessary | 1 |
| | 1 | × | × | | |
| (C) → (D) (external subsystem clock) | 0 | 1 | 1 | Unnecessary | 1 |

Unnecessary if the CPU is operating with the subsystem clock

**(8) CPU clock changing from subsystem clock (D) to internal high-speed oscillation clock (B)[Note]**

**Note** 78K0/KC2-L only

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | RSTOP | RSTS | MCM0 | CSS |
|---|---|---|---|---|
| (D) → (B) | 0 | Confirm this flag is 1. | 0 | 0 |

Unnecessary if the CPU is operating with the internal high-speed oscillation clock

↑

Unnecessary if XSEL is 0

**Remarks 1.** (A) to (I) in Table 5-6 correspond to (A) to (I) in Figure 5-18 and 5-19.

**2.** MCM0: Bit 0 of the main clock mode register (MCM)
EXCLKS, OSCSELS: Bits 5 and 4 of the clock operation mode select register (OSCCTL)
RSTS, RSTOP: Bits 7 and 0 of the internal oscillation mode register (RCM)
XTSTART, CSS: Bits 6 and 4 of the processor clock control register (PCC)
×: Don't care

**Table 5-6. CPU Clock Transition and SFR Register Setting Examples (4/4)**

**(9) CPU clock changing from subsystem clock (D) to high-speed system clock (C)[Note]**

**Note** 78K0/KC2-L only

(Setting sequence of SFR registers) →

| Setting Flag of SFR Register / Status Transition | EXCLK | OSCSEL | MSTOP | OSTC Register | XSEL[Note] | MCM0 | CSS |
|---|---|---|---|---|---|---|---|
| (D) → (C) (X1 clock) | 0 | 1 | 0 | Must be checked | 1 | 1 | 0 |
| (D) → (C) (external main system clock) | 1 | 1 | 0 | Must not be checked | 1 | 1 | 0 |
| | Unnecessary if these registers are already set | | Unnecessary if the CPU is operating with the high-speed system clock | | Unnecessary if this register is already set | | |

**Note** The value of this flag can be changed only once after a reset release. This setting is not necessary if it has already been set.

**Caution Set the clock after the supply voltage has reached the operable voltage of the clock to be set (refer to CHAPTER 28 ELECTRICAL SPECIFICATIONS).**

**(10) • HALT mode (E) set while CPU is operating with internal high-speed oscillation clock (B)**
**• HALT mode (F) set while CPU is operating with high-speed system clock (C)**
**• HALT mode (G) set while CPU is operating with subsystem clock (D)[Note]**

| Status Transition | Setting |
|---|---|
| (B) → (E)<br>(C) → (F)<br>(D) → (G)[Note] | Executing HALT instruction |

**Note** 78K0/KC2-L only

**(11) • STOP mode (H) set while CPU is operating with internal high-speed oscillation clock (B)**
**• STOP mode (I) set while CPU is operating with high-speed system clock (C)**

(Setting sequence) →

| Status Transition | Setting | |
|---|---|---|
| (B) → (H)<br>(C) → (I) | Stopping peripheral functions that cannot operate in STOP mode | Executing STOP instruction |

<R> **Note** When transitioning to the STOP mode, it is possible to achieve low power consumption by setting RMC = 56H first.

**Remarks 1.** (A) to (I) in Table 5-6 correspond to (A) to (I) in Figures 5-18 and 5-19.

**2.** EXCLK, OSCSEL: Bits 7 and 6 of the clock operation mode select register (OSCCTL)
MSTOP: Bit 7 of the main OSC control register (MOC)
XSEL, MCM0: Bits 2 and 0 of the main clock mode register (MCM)
CSS: Bit 4 of the processor clock control register (PCC)

### 5.6.7 Condition before changing CPU clock and processing after changing CPU clock

Condition before changing the CPU clock and processing after changing the CPU clock are shown below.

**Table 5-7. Changing CPU Clock**

| | CPU Clock | | Condition Before Change | Processing After Change |
|---|---|---|---|---|
| | Before Change | After Change | | |
| KY2-L, KA2-L, KB2-L, KC2-L | Internal high-speed oscillation clock | X1 clock | Stabilization of X1 oscillation<br>• MSTOP = 0, OSCSEL = 1, EXCLK = 0<br>• After elapse of oscillation stabilization time | Internal high-speed oscillator can be stopped (RSTOP = 1). |
| | | External main system clock | Enabling input of external clock from EXCLK pin<br>• MSTOP = 0, OSCSEL = 1, EXCLK = 1 | Internal high-speed oscillator can be stopped (RSTOP = 1). |
| | X1 clock | Internal high-speed oscillation clock | Oscillation of internal high-speed oscillator<br>• RSTOP = 0 | X1 oscillation can be stopped (MSTOP = 1). |
| | External main system clock | | | External main system clock input can be disabled (MSTOP = 1). |
| KC2-L | Internal high-speed oscillation clock | XT1 clock | Stabilization of XT1 oscillation<br>• XTSTART = 0, EXCLKS = 0, OSCSELS = 1, or XTSTART = 1<br>• After elapse of oscillation stabilization time | Operating current can be reduced by stopping internal high-speed oscillator (RSTOP = 1). |
| | X1 clock | | | X1 oscillation can be stopped (MSTOP = 1). |
| | External main system clock | | | External main system clock input can be disabled (MSTOP = 1). |
| | Internal high-speed oscillation clock | External subsystem clock | Enabling input of external clock from EXCLKS pin<br>• XTSTART = 0, EXCLKS = 1, OSCSELS = 1 | Operating current can be reduced by stopping internal high-speed oscillator (RSTOP = 1). |
| | X1 clock | | | X1 oscillation can be stopped (MSTOP = 1). |
| | External main system clock | | | External main system clock input can be disabled (MSTOP = 1). |
| | XT1 clock, external subsystem clock | Internal high-speed oscillation clock | Oscillation of internal high-speed oscillator and selection of internal high-speed oscillation clock as main system clock<br>• RSTOP = 0, MCS = 0 | XT1 oscillation can be stopped or external subsystem clock input can be disabled (OSCSELS = 0). |
| | | X1 clock | Stabilization of X1 oscillation and selection of high-speed system clock as main system clock<br>• MSTOP = 0, OSCSEL = 1, EXCLK = 0<br>• After elapse of oscillation stabilization time<br>• MCS = 1 | XT1 oscillation can be stopped or external subsystem clock input can be disabled (OSCSELS = 0). |
| | | External main system clock | Enabling input of external clock from EXCLK pin and selection of high-speed system clock as main system clock<br>• MSTOP = 0, OSCSEL = 1, EXCLK = 1<br>• MCS = 1 | XT1 oscillation can be stopped or external subsystem clock input can be disabled (OSCSELS = 0). |

**Remark** Only 78K0/KC2-L is provided with a subsystem clock.

### 5.6.8 Time required for switchover of CPU clock and main system clock

By setting bits 0 to 2 (PCC0 to PCC2) and bit 4 (CSS) of the processor clock control register (PCC), the CPU clock can be switched (between the main system clock and the subsystem clock) and the division ratio of the main system clock can be changed.

The actual switchover operation is not performed immediately after rewriting to PCC; operation continues on the pre-switchover clock for several clocks (refer to **Tables 5-8** and **5-9**).

Whether the CPU is operating on the main system clock or the subsystem clock[Note] can be ascertained using bit 5 (CLS) of the PCC register.

**Note** 78K0/KC2-L only

**Table 5-8. Time Required for Switchover of CPU Clock and Main System Clock Cycle Division Factor (78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L)**

| Set Value Before Switchover | | | Set Value After Switchover | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PCC2 | PCC1 | PCC0 | PCC2 | PCC1 | PCC0 | PCC2 | PCC1 | PCC0 | PCC2 | PCC1 | PCC0 | PCC2 | PCC1 | PCC0 | PCC2 | PCC1 | PCC0 |
| | | | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | | | | 16 clocks | | | 16 clocks | | | 16 clocks | | | 16 clocks | | |
| 0 | 0 | 1 | 8 clocks | | | | | | 8 clocks | | | 8 clocks | | | 8 clocks | | |
| 0 | 1 | 0 | 4 clocks | | | 4 clocks | | | | | | 4 clocks | | | 4 clocks | | |
| 0 | 1 | 1 | 2 clocks | | | 2 clocks | | | 2 clocks | | | | | | 2 clocks | | |
| 1 | 0 | 0 | 1 clock | | | 1 clock | | | 1 clock | | | 1 clock | | | | | |

**Remark** The number of clocks listed in Table 5-8 is the number of CPU clocks before switchover.

**Table 5-9. Time Required for Switchover of CPU Clock and Main System Clock Cycle Division Factor (78K0/KC2-L)**

| Set Value Before Switchover | | | | Set Value After Switchover | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CSS | PCC2 | PCC1 | PCC0 | CSS | PCC2 | PCC1 | PCC0 | CSS | PCC2 | PCC1 | PCC0 | CSS | PCC2 | PCC1 | PCC0 | CSS | PCC2 | PCC1 | PCC0 | CSS | PCC2 | PCC1 | PCC0 | CSS | PCC2 | PCC1 | PCC0 |
| | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | × | × | × |
| 0 | 0 | 0 | 0 | | | | | 16 clocks | | | | 16 clocks | | | | 16 clocks | | | | 16 clocks | | | | $f_{XP}/f_{SUB}$ clocks | | | |
| | 0 | 0 | 1 | 8 clocks | | | | | | | | 8 clocks | | | | 8 clocks | | | | 8 clocks | | | | $f_{XP}/2f_{SUB}$ clocks | | | |
| | 0 | 1 | 0 | 4 clocks | | | | 4 clocks | | | | | | | | 4 clocks | | | | 4 clocks | | | | $f_{XP}/4f_{SUB}$ clocks | | | |
| | 0 | 1 | 1 | 2 clocks | | | | 2 clocks | | | | 2 clocks | | | | | | | | 2 clocks | | | | $f_{XP}/8f_{SUB}$ clocks | | | |
| | 1 | 0 | 0 | 1 clock | | | | 1 clock | | | | 1 clock | | | | 1 clock | | | | | | | | $f_{XP}/16f_{SUB}$ clocks | | | |
| 1 | × | × | × | 2 clocks | | | | 2 clocks | | | | 2 clocks | | | | 2 clocks | | | | 2 clocks | | | | | | | |

**Caution Selection of the main system clock cycle division factor (PCC0 to PCC2) and switchover from the main system clock to the subsystem clock (changing CSS from 0 to 1) should not be set simultaneously.**
**Simultaneous setting is possible, however, for selection of the main system clock cycle division factor (PCC0 to PCC2) and switchover from the subsystem clock to the main system clock (changing CSS from 1 to 0).**

**Remark 1.** The number of clocks listed in Table 5-9 is the number of CPU clocks before switchover.

**Remark 2.** When switching the CPU clock from the main system clock to the subsystem clock, calculate the number of clocks by rounding up to the next clock and discarding the decimal portion, as shown below.

**Example** When switching CPU clock from $f_{XP}$ to $f_{SUB}$ (@ oscillation with $f_{XP}$ = 10 MHz, $f_{SUB}$ = 32.768 kHz)

$f_{XP}/f_{SUB}$ = 10000/32.768 ≅ 305.1 → 306 clocks

By setting bit 0 (MCM0) of the main clock mode register (MCM), the main system clock can be switched (between the internal high-speed oscillation clock and the high-speed system clock).

The actual switchover operation is not performed immediately after rewriting to MCM0; operation continues on the pre-switchover clock for several clocks (refer to **Table 5-10**).

Whether the CPU is operating on the internal high-speed oscillation clock or the high-speed system clock can be ascertained using bit 1 (MCS) of MCM.

**Table 5-10. Maximum Time Required for Main System Clock Switchover**

| Set Value Before Switchover | Set Value After Switchover | |
|---|---|---|
| MCM0 | MCM0 | |
| | 0 | 1 |
| 0 | | 1 + $2f_{IH}/f_{XH}$ clock |
| 1 | 1 + $2f_{XH}/f_{IH}$ clock | |

**Cautions 1. When switching the internal high-speed oscillation clock to the high-speed system clock, bit 2 (XSEL) of MCM must be set to 1 in advance. The value of XSEL can be changed only once after a reset release.**

**2. Do not rewrite MCM0 when the CPU clock operates with the subsystem clock.**

**Remarks 1.** The number of clocks listed in Table 5-10 is the number of main system clocks before switchover.

**2.** Calculate the number of clocks in Table 5-10 by removing the decimal portion.

**Example** When switching the main system clock from the internal high-speed oscillation clock to the high-speed system clock (@ oscillation with $f_{IH}$ = 8 MHz, $f_{XH}$ = 10 MHz)

1 + $2f_{IH}/f_{XH}$ = 1 + 2 × 8/10 = 1 + 2 × 0.8 = 1 + 1.6 = 2.6 → 2 clocks

### 5.6.9 Conditions before clock oscillation is stopped

The following lists the register flag settings for stopping the clock oscillation (disabling external clock input) and conditions before the clock oscillation is stopped.

**Table 5-11. Conditions Before the Clock Oscillation Is Stopped and Flag Settings (78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L)**

| Clock | Conditions Before Clock Oscillation Is Stopped (External Clock Input Disabled) | Flag Settings of SFR Register |
|---|---|---|
| Internal high-speed oscillation clock | MCS = 1<br>(The CPU is operating on the high-speed system clock) | RSTOP = 1 |
| X1 clock | MCS = 0<br>(The CPU is operating on the internal high-speed oscillation clock) | MSTOP = 1 |
| External main system clock | | |

**Table 5-12. Conditions Before the Clock Oscillation Is Stopped and Flag Settings (78K0/KC2-L)**

| Clock | Conditions Before Clock Oscillation Is Stopped (External Clock Input Disabled) | Flag Settings of SFR Register |
|---|---|---|
| Internal high-speed oscillation clock | MCS = 1 or CLS = 1<br>(The CPU is operating on a clock other than the internal high-speed oscillation clock) | RSTOP = 1 |
| X1 clock | MCS = 0 or CLS = 1<br>(The CPU is operating on a clock other than the high-speed system clock) | MSTOP = 1 |
| External main system clock | | |
| XT1 clock | CLS = 0<br>(The CPU is operating on a clock other than the subsystem clock) | OSCSELS = 0 |
| External subsystem clock | | |

### 5.6.10 Peripheral hardware and source clocks

The following lists peripheral hardware and source clocks incorporated in the 78K0/Kx2-L microcontrollers.

**Remark** The peripheral hardware depends on the product. Refer to **1.4 Block Diagram** and **1.5 Outline of Functions**.

**Table 5-13. Peripheral Hardware and Source Clocks**

| Peripheral Hardware \ Source Clock | | Peripheral Hardware Clock ($f_{PRS}$) | Subsystem Clock ($f_{SUB}$)[Note 1] | Internal Low-Speed Oscillation Clock ($f_{IL}$) | TM50 Output | External Clock from Peripheral Hardware Pins |
|---|---|---|---|---|---|---|
| 16-bit timer/event counter 00 | | Y | N | N | N | Y (TI000 pin)[Note 2] |
| 8-bit timer/ event counter | 50 | Y | N | N | N | Y (TI50 pin)[Note 2] |
| | 51 | Y | N | N | N | Y (TI51 pin)[Note 2] |
| 8-bit timer | H0 | Y | N | N | Y | N |
| | H1 | Y | N | Y | N | N |
| Real-time counter | | N | Y | N | N | N |
| Watchdog timer | | N | N | Y | N | N |
| Clock output | | Y | Y | N | N | N |
| A/D converter | | Y | N | N | N | N |
| Serial interface | UART6 | Y | N | N | Y | N |
| | CSI10 | Y | N | N | N | Y ($\overline{SCK10}$ pin)[Note 2] |
| | CSI11 | Y | N | N | N | Y ($\overline{SCK11}$ pin)[Note 2] |
| | IICA | Y | N | N | N | Y (SCLA0 pin)[Note 2] |

**Notes 1.** 78K0/KC2-L only

**2.** Do not start the peripheral hardware operation with the external clock from peripheral hardware pins when the internal high-speed oscillation clock and high-speed system clock are stopped while the CPU operates with the subsystem clock, or when in the STOP mode.

**Remark** Y: Can be selected, N: Cannot be selected

# CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00

<R>

| Item | 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) |
|---|---|---|---|---|---|---|
| | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40, 44, 48 Pins |
| 16-bit timer/event counter 00 | √ | √ | √ | √ | √ | √ |
| Timer I/O pin | input: 1, output: 1<br>or<br>input: 2 | input: 1, output: 1<br>or<br>input: 2 | input: 1,<br>output: none | input: 1<br>or<br>output 1<br>or<br>input: 2<br>or<br>input: 1, output: 1 | input: 1, output: 1<br>or<br>input: 2 | input: 1, output: 1<br>or<br>input: 2 |

**Remark** √: Mounted, –: Not mounted

## 6.1 Functions of 16-Bit Timer/Event Counter 00

16-bit timer/event counter 00 is mounted onto all 78K0/Kx2-L microcontroller products.

16-bit timer/event counter 00 has the following functions.

**(1) Interval timer**

16-bit timer/event counter 00 generates an interrupt request at the preset time interval.

**(2) Square-wave output**

16-bit timer/event counter 00 can output a square wave with any selected frequency.

**(3) External event counter**

16-bit timer/event counter 00 can measure the number of pulses of an externally input signal.

**(4) One-shot pulse output**

16-bit timer event counter 00 can output a one-shot pulse whose output pulse width can be set freely.

**(5) PPG output**

16-bit timer/event counter 00 can output a rectangular wave whose frequency and output pulse width can be set freely.

**(6) Pulse width measurement**

16-bit timer/event counter 00 can measure the pulse width of an externally input signal.

## 6.2 Configuration of 16-Bit Timer/Event Counter 00

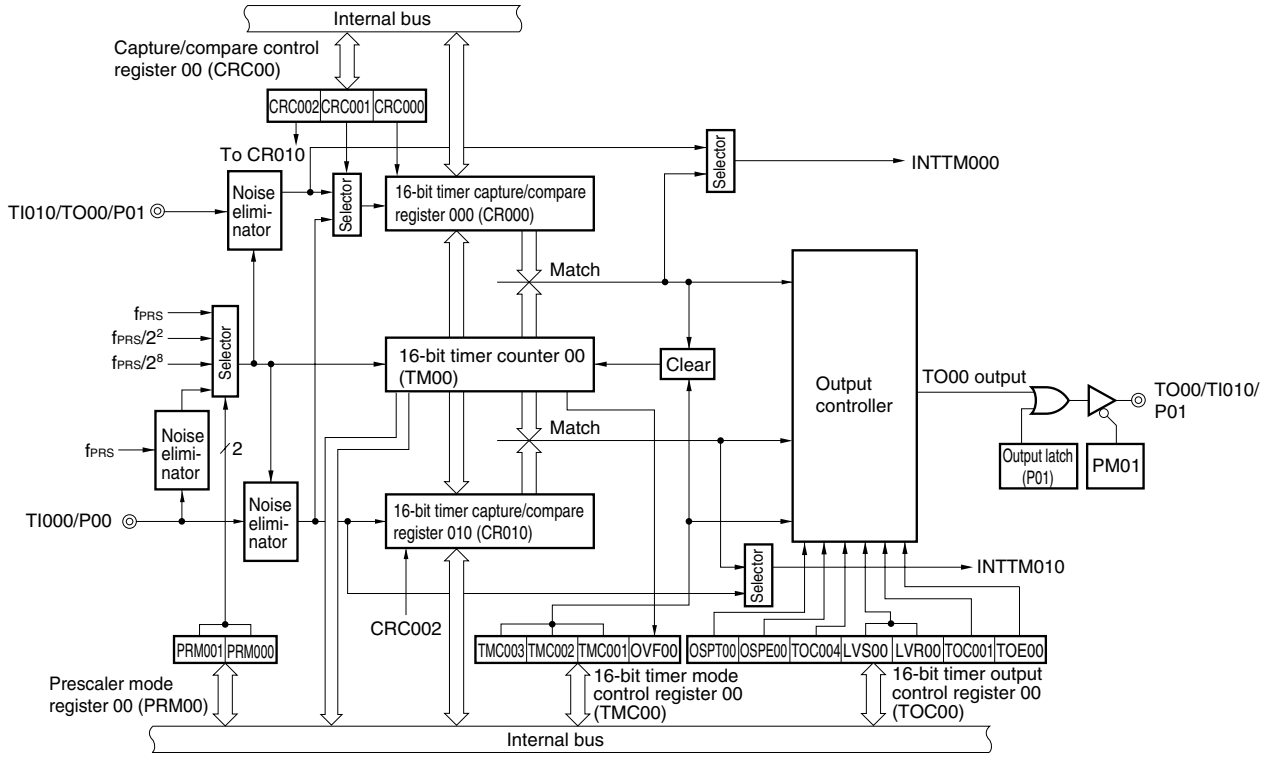
16-bit timer/event counter 00 includes the following hardware.

**Table 6-1. Configuration of 16-Bit Timer/Event Counter 00**

| Item | Configuration |
|---|---|
| Time/counter | 16-bit timer counter 00 (TM00) |
| Register | 16-bit timer capture/compare registers 000, 010 (CR000, CR010) |
| Timer input | TI000, TI010 |
| Timer output | TO00, output controller |
| Control registers | 16-bit timer mode control register 00 (TMC00)<br>16-bit timer capture/compare control register 00 (CRC00)<br>16-bit timer output control register 00 (TOC00)<br>Prescaler mode register 00 (PRM00)<br>Port alternate switch control register (MUXSEL) Note<br>Port mode register 0 (PM0)<br>Port register 0 (P0) |

**Note** 78K0/KA2-L (25-pin and 32-pin products) only

**Figure 6-1. Block Diagram of 16-Bit Timer/Event Counter 00**




**Remark** 78K0/KY2-L, 78K0/KA2-L (20 pins) : TI000/INTP0/P00, TI010/TO00/P01
78K0/KA2-L (25 pins) : TI000/INTP0/P00 or (TI000)/(INTP0)/P121
78K0/KA2-L (32 pins) : TI010/TO00/P01, (TI000)/(INTP0)/P121 or (TI000)/(INTP0)/$\overline{\text{RESET}}$/P125
78K0/KB2-L, 78K0/KC2-L: TI000/P00, TI010/TO00/P01

**Cautions 1. The valid edge of TI010 and timer output (TO00) cannot be used for the P01 pin at the same time. Select either of the functions.**

**2. If clearing of bits 3 and 2 (TMC003 and TMC002) of 16-bit timer mode control register 00 (TMC00) to 00 and input of the capture trigger conflict, then the captured data is undefined.**

**3. To change the mode from the capture mode to the comparison mode, first clear the TMC003 and TMC002 bits to 00, and then change the setting.**

**A value that has been once captured remains stored in CR000 unless the device is reset. If the mode has been changed to the comparison mode, be sure to set a comparison value.**

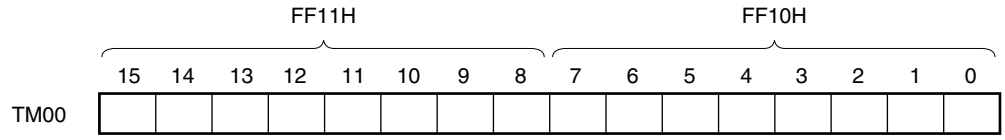
### (1) 16-bit timer counter 00 (TM00)

TM00 is a 16-bit read-only register that counts count pulses.

The counter is incremented in synchronization with the rising edge of the count clock.

**Figure 6-2. Format of 16-Bit Timer Counter 00 (TM00)**

Address: FF10H, FF11H After reset: 0000H R




The count value of TM00 can be read by reading TM00 when the value of bits 3 and 2 (TMC003 and TMC002) of 16-bit timer mode control register 00 (TMC00) is other than 00. The value of TM00 is 0000H if it is read when TMC003 and TMC002 = 00.

The count value is reset to 0000H in the following cases.

- At reset signal generation
- If TMC003 and TMC002 are cleared to 00
- If the valid edge of the TI000 pin is input in the mode in which the clear & start occurs when inputting the valid edge to the TI000 pin
- If TM00 and CR000 match in the mode in which the clear & start occurs when TM00 and CR000 match
- OSPT00 is set to 1 in one-shot pulse output mode or the valid edge is input to the TI000 pin

**Caution Even if TM00 is read, the value is not captured by CR010.**

**(2) 16-bit timer capture/compare register 000 (CR000), 16-bit timer capture/compare register 010 (CR010)**

CR000 and CR010 are 16-bit registers that are used with a capture function or comparison function selected by using CRC00.

Change the value of CR000 while the timer is stopped (TMC003 and TMC002 = 00).

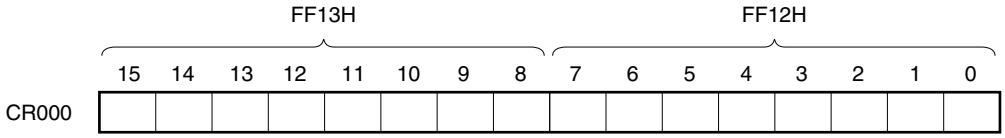
The value of CR010 can be changed during operation if the value has been set in a specific way. For details, refer to **6.5.1 Rewriting CR010 during TM00 operation**.

These registers can be read or written in 16-bit units.

Reset signal generation clears these registers to 0000H.

**Figure 6-3. Format of 16-Bit Timer Capture/Compare Register 000 (CR000)**

Address: FF12H, FF13H After reset: 0000H R/W




**(i) When CR000 is used as a compare register**

The value set in CR000 is constantly compared with the TM00 count value, and an interrupt request signal (INTTM000) is generated if they match. The value is held until CR000 is rewritten.

**Caution CR000 does not perform the capture operation when it is set in the comparison mode, even if a capture trigger is input to it.**

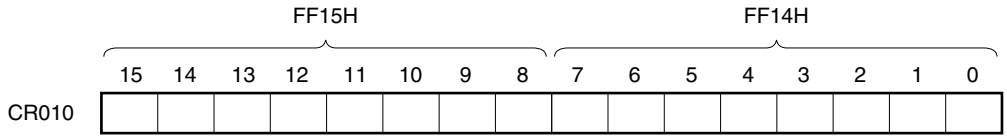
**(ii) When CR000 is used as a capture register**

The count value of TM00 is captured to CR000 when a capture trigger is input.

As the capture trigger, an edge of a phase reverse to that of the TI000 pin or the valid edge of the TI010 pin can be selected by using CRC00 or PRM00.

**Figure 6-4. Format of 16-Bit Timer Capture/Compare Register 010 (CR010)**

Address: FF14H, FF15H After reset: 0000H R/W




**(i) When CR010 is used as a compare register**

The value set in CR010 is constantly compared with the TM00 count value, and an interrupt request signal (INTTM010) is generated if they match.

**Caution CR010 does not perform the capture operation when it is set in the comparison mode, even if a capture trigger is input to it.**

**(ii) When CR010 is used as a capture register**

The count value of TM00 is captured to CR010 when a capture trigger is input.

It is possible to select the valid edge of the TI000 pin as the capture trigger. The TI000 pin valid edge is set by PRM00.

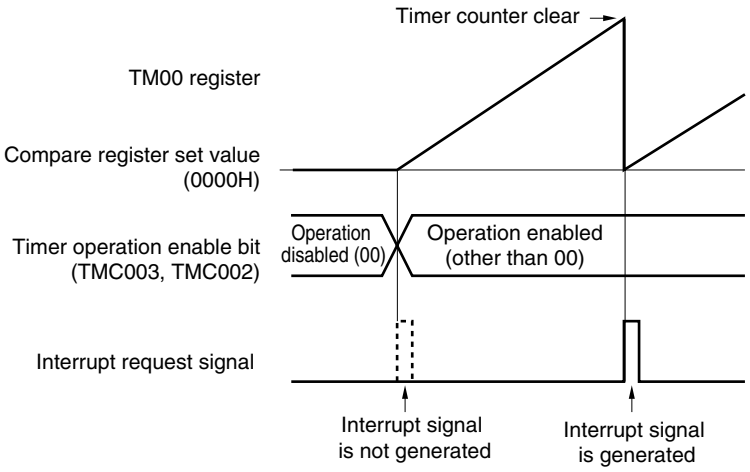
**(iii) Setting range when CR000 or CR010 is used as a compare register**

When CR000 or CR010 is used as a compare register, set it as shown below.

| Operation | CR000 Register Setting Range | CR010 Register Setting Range |
|---|---|---|
| Operation as interval timer | 0000H < N ≤ FFFFH | 0000H[Note] ≤ M ≤ FFFFH<br>Normally, this setting is not used. Mask the match interrupt signal (INTTM010). |
| Operation as square-wave output | | |
| Operation as external event counter | | |
| Operation in the clear & start mode entered by TI000 pin valid edge input | 0000H[Note] ≤ N ≤ FFFFH | 0000H[Note] ≤ M ≤ FFFFH |
| Operation as free-running timer | | |
| Operation as PPG output | M < N ≤ FFFFH | 0000H[Note] ≤ M < N |
| Operation as one-shot pulse output | 0000H[Note] ≤ N ≤ FFFFH (N ≠ M) | 0000H[Note] ≤ M ≤ FFFFH (M ≠ N) |

**Note** When 0000H is set, a match interrupt immediately after the timer operation does not occur and timer output is not changed, and the first match timing is as follows. A match interrupt occurs at the timing when the timer counter (TM00 register) is changed from 0000H to 0001H.

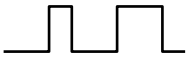
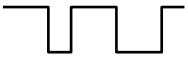
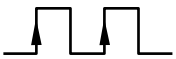
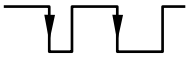
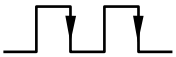
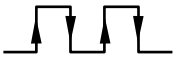
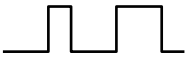
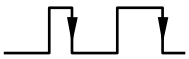
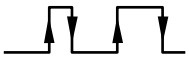
- When the timer counter is cleared due to overflow
- When the timer counter is cleared due to TI000 pin valid edge (when clear & start mode is entered by TI000 pin valid edge input)
- When the timer counter is cleared due to compare match (when clear & start mode is entered by match between TM00 and CR000 (CR000 = other than 0000H, CR010 = 0000H))




**Remarks 1.** N: CR000 register set value, M: CR010 register set value

**2.** For details of the operation enable bits (bits 3 and 2 (TMC003 and TMC002)), refer to **6.3 (1) 16-bit timer mode control register 00 (TMC00)**.

**Table 6-2. Capture Operation of CR000 and CR010**

<table>
<tr><th>External Input Signal / Capture Operation</th><th colspan="2">TI000 Pin Input </th><th colspan="2">TI010 Pin Input </th></tr>
<tr><td rowspan="5">Capture operation of CR000</td><td rowspan="4">CRC001 = 1<br>TI000 pin input<br>(reverse phase)<br></td><td>Set values of ES010 and ES000<br>Position of edge to be captured</td><td rowspan="4">CRC001 bit = 0<br>TI010 pin input<br></td><td>Set values of ES110 and ES100<br>Position of edge to be captured</td></tr>
<tr><td>01: Rising </td><td>01: Rising </td></tr>
<tr><td>00: Falling </td><td>00: Falling </td></tr>
<tr><td>11: Both edges<br>(cannot be captured)</td><td>11: Both edges </td></tr>
<tr><td>Interrupt signal</td><td>INTTM000 signal is not generated even if value is captured.</td><td>Interrupt signal</td><td>INTTM000 signal is generated each time value is captured.</td></tr>
<tr><td rowspan="5">Capture operation of CR010</td><td rowspan="4">TI000 pin input[Note]<br></td><td>Set values of ES010 and ES000<br>Position of edge to be captured</td><td colspan="2" rowspan="5"></td></tr>
<tr><td>01: Rising </td></tr>
<tr><td>00: Falling </td></tr>
<tr><td>11: Both edges </td></tr>
<tr><td>Interrupt signal</td><td>INTTM010 signal is generated each time value is captured.</td></tr>
</table>

**Note** The capture operation of CR010 is not affected by the setting of the CRC001 bit.

**Caution To capture the count value of the TM00 register to the CR000 register by using the phase reverse to that input to the TI000 pin, the interrupt request signal (INTTM000) is not generated after the value has been captured. If the valid edge is detected on the TI010 pin during this operation, the capture operation is not performed but the INTTM000 signal is generated as an external interrupt signal. To not use the external interrupt, mask the INTTM000 signal.**

**Remark** CRC001: Refer to **6.3 (2) Capture/compare control register 00 (CRC00)**.
ES110, ES100, ES010, ES000: Refer to **6.3 (4) Prescaler mode register 00 (PRM00)**.

## 6.3 Registers Controlling 16-Bit Timer/Event Counter 00

Registers used to control 16-bit timer/event counter 00 are shown below.

- 16-bit timer mode control register 00 (TMC00)
- Capture/compare control register 00 (CRC00)
- 16-bit timer output control register 00 (TOC00)
- Prescaler mode register 00 (PRM00)
- Port alternate switch control register (MUXSEL)
- Port mode register 0 (PM0)
- Port register 0 (P0)

### (1) 16-bit timer mode control register 00 (TMC00)

TMC00 is an 8-bit register that sets the 16-bit timer/event counter 00 operation mode, TM00 clear mode, and output timing, and detects an overflow.

Rewriting TMC00 is prohibited during operation (when TMC003 and TMC002 = other than 00). However, it can be changed when TMC003 and TMC002 are cleared to 00 (stopping operation) and when OVF00 is cleared to 0.

TMC00 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears TMC00 to 00H.

**Caution 16-bit timer/event counter 00 starts operation at the moment TMC003 and TMC002 are set to values other than 00 (operation stop mode), respectively. Set TMC003 and TMC002 to 00 to stop the operation.**

**Figure 6-5. Format of 16-Bit Timer Mode Control Register 00 (TMC00)**

Address: FFBAH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| TMC00 | 0 | 0 | 0 | 0 | TMC003 | TMC002 | TMC001 | OVF00 |

| TMC003 | TMC002 | Operation enable of 16-bit timer/event counter 00 |
|---|---|---|
| 0 | 0 | Disables 16-bit timer/event counter 00 operation. Stops supplying operating clock. Clears 16-bit timer counter 00 (TM00). |
| 0 | 1 | Free-running timer mode |
| 1 | 0 | Clear & start mode entered by TI000 pin valid edge input[Note] |
| 1 | 1 | Clear & start mode entered upon a match between TM00 and CR000 |

| TMC001 | Condition to reverse timer output (TO00) |
|---|---|
| 0 | • Match between TM00 and CR000 or match between TM00 and CR010 |
| 1 | • Match between TM00 and CR000 or match between TM00 and CR010<br>• Trigger input of TI000 pin valid edge |

| OVF00 | TM00 overflow flag |
|---|---|
| Clear (0) | Clears OVF00 to 0 or TMC003 and TMC002 = 00 |
| Set (1) | Overflow occurs. |
| OVF00 is set to 1 when the value of TM00 changes from FFFFH to 0000H in all the operation modes (free-running timer mode, clear & start mode entered by TI000 pin valid edge input, and clear & start mode entered upon a match between TM00 and CR000).<br>It can also be set to 1 by writing 1 to OVF00. | |

**Note** The TI000 pin valid edge is set by bits 5 and 4 (ES010, ES000) of prescaler mode register 00 (PRM00).

**(2) Capture/compare control register 00 (CRC00)**

CRC00 is the register that controls the operation of CR000 and CR010.

Changing the value of CRC00 is prohibited during operation (when TMC003 and TMC002 = other than 00).

CRC00 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears CRC00 to 00H.

**Figure 6-6. Format of Capture/Compare Control Register 00 (CRC00)**

Address: FFBCH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CRC00 | 0 | 0 | 0 | 0 | 0 | CRC002 | CRC001 | CRC000 |

| CRC002 | CR010 operating mode selection |
|---|---|
| 0 | Operates as compare register |
| 1 | Operates as capture register |

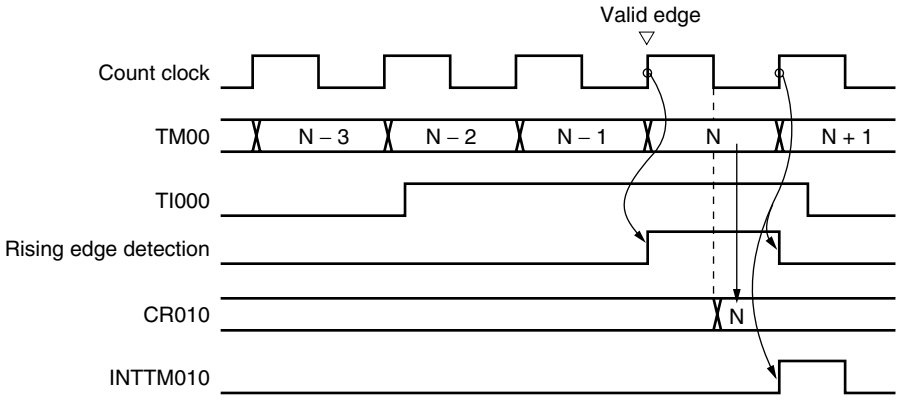
| CRC001 | CR000 capture trigger selection |
|---|---|
| 0 | Captures on valid edge of TI010 pin |
| 1 | Captures on valid edge of TI000 pin by reverse phase[Note] |
| The valid edge of the TI010 and TI000 pin is set by PRM00.<br>If ES010 and ES000 are set to 11 (both edges) when CRC001 is 1, the valid edge of the TI000 pin cannot be detected. | |

| CRC000 | CR000 operating mode selection |
|---|---|
| 0 | Operates as compare register |
| 1 | Operates as capture register |
| If TMC003 and TMC002 are set to 11 (clear & start mode entered upon a match between TM00 and CR000), be sure to set CRC000 to 0. | |

**Note** When the valid edge is detected from the TI010 pin, the capture operation is not performed but the INTTM000 signal is generated as an external interrupt signal.

**Caution To ensure that the capture operation is performed properly, the capture trigger requires a pulse two cycles longer than the count clock selected by prescaler mode register 00 (PRM00).**

**Figure 6-7. Example of CR010 Capture Operation (When Rising Edge Is Specified)**



**(3) 16-bit timer output control register 00 (TOC00)**

TOC00 is an 8-bit register that controls the TO00 output.

TOC00 can be rewritten while only OSPT00 is operating (when TMC003 and TMC002 = other than 00). Rewriting the other bits is prohibited during operation.

However, TOC004 can be rewritten during timer operation as a means to rewrite CR010 (refer to **6.5.1 Rewriting CR010 during TM00 operation**).

TOC00 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears TOC00 to 00H.

**Caution Be sure to set TOC00 using the following procedure.**

**<1> Set TOC004 and TOC001 to 1.**

**<2> Set only TOE00 to 1.**

**<3> Set either of LVS00 or LVR00 to 1.**

**Figure 6-8. Format of 16-Bit Timer Output Control Register 00 (TOC00)**

Address: FFBDH After reset: 00H R/W

| Symbol | 7 | <6> | <5> | 4 | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| TOC00 | 0 | OSPT00 | OSPE00 | TOC004 | LVS00 | LVR00 | TOC001 | TOE00 |

| OSPT00 | One-shot pulse output trigger via software |
|---|---|
| 0 | – |
| 1 | One-shot pulse output |
| The value of this bit is always “0” when it is read. Do not set this bit to 1 in a mode other than the one-shot pulse output mode.<br>If it is set to 1, TM00 is cleared and started. | |

| OSPE00 | One-shot pulse output operation control |
|---|---|
| 0 | Successive pulse output |
| 1 | One-shot pulse output |
| One-shot pulse output operates correctly in the free-running timer mode or clear & start mode entered by TI000 pin valid edge input.<br>The one-shot pulse cannot be output in the clear & start mode entered upon a match between TM00 and CR000. | |

| TOC004 | TO00 output control on match between CR010 and TM00 |
|---|---|
| 0 | Disables inversion operation |
| 1 | Enables inversion operation |
| The interrupt signal (INTTM010) is generated even when TOC004 = 0. | |

| LVS00 | LVR00 | Setting of TO00 output status |
|---|---|---|
| 0 | 0 | No change |
| 0 | 1 | Initial value of TO00 output is low level (TO00 output is cleared to 0). |
| 1 | 0 | Initial value of TO00 output is high level (TO00 output is set to 1). |
| 1 | 1 | Setting prohibited |

- LVS00 and LVR00 can be used to set the initial value of the TO00 output level. If the initial value does not have to be set, leave LVS00 and LVR00 as 00.
- Be sure to set LVS00 and LVR00 when TOE00 = 1.
  LVS00, LVR00, and TOE00 being simultaneously set to 1 is prohibited.
- LVS00 and LVR00 are trigger bits. By setting these bits to 1, the initial value of the TO00 output level can be set. Even if these bits are cleared to 0, TO00 output is not affected.
- The values of LVS00 and LVR00 are always 0 when they are read.
- For how to set LVS00 and LVR00, refer to **6.5.2 Setting LVS00 and LVR00**.
- The actual TO00/TI010/P01 pin output is determined depending on PM01 and P01, besides TO00 output.

| TOC001 | TO00 output control on match between CR000 and TM00 |
|---|---|
| 0 | Disables inversion operation |
| 1 | Enables inversion operation |
| The interrupt signal (INTTM000) is generated even when TOC001 = 0. | |

| TOE00 | TO00 output control |
|---|---|
| 0 | Disables output (TO00 output fixed to low level) |
| 1 | Enables output |

**(4) Prescaler mode register 00 (PRM00)**

PRM00 is the register that sets the TM00 count clock and TI000 and TI010 pin input valid edges.

Rewriting PRM00 is prohibited during operation (when TMC003 and TMC002 = other than 00).

PRM00 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears PRM00 to 00H.

**Cautions 1. Do not apply the following setting when setting the PRM001 and PRM000 bits to 11 (to specify the valid edge of the TI000 pin as a count clock).**

- **Clear & start mode entered by the TI000 pin valid edge**
- **Setting the TI000 pin as a capture trigger**

**2. If the operation of the 16-bit timer/event counter 00 is enabled when the TI000 or TI010 pin is at high level and when the valid edge of the TI000 or TI010 pin is specified to be the rising edge or both edges, the high level of the TI000 or TI010 pin is detected as a rising edge. Note this when the TI000 or TI010 pin is pulled up. However, the rising edge is not detected when the timer operation has been once stopped and then is enabled again.**

**3. The valid edge of TI010 and timer output (TO00) cannot be used for the P01 pin at the same time. Select either of the functions.**

**Figure 6-9. Format of Prescaler Mode Register 00 (PRM00)**

Address: FFBBH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PRM00 | ES110 | ES100 | ES010 | ES000 | 0 | 0 | PRM001 | PRM000 |

| ES110 | ES100 | TI010 pin valid edge selection |
|---|---|---|
| 0 | 0 | Falling edge |
| 0 | 1 | Rising edge |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Both falling and rising edges |

| ES010 | ES000 | TI000 pin valid edge selection |
|---|---|---|
| 0 | 0 | Falling edge |
| 0 | 1 | Rising edge |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Both falling and rising edges |

| PRM001 | PRM000 | Count clock selection[Note 1] | | | |
|---|---|---|---|---|---|
| | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 1 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz |
| 1 | 0 | $f_{PRS}/2^8$ | 7.81 kHz | 19.53 kHz | 39.06 kHz |
| 1 | 1 | TI000 valid edge[Notes 2, 3] | | | |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.
- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** The external clock from the TI000 pin requires a pulse longer than twice the cycle of the peripheral hardware clock ($f_{PRS}$).

**3.** Do not start timer operation with the external clock from the TI000 pin when in the STOP mode.

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

<R> **(5) Port alternate switch control register (MUXSEL) (78K0/KA2-L (25-pin and 32-pin products) only)**

MUXSEL of 78K0/KA2-L (25-pin products) assigns TOH1, TI51, TI000, and INTP0 pins. y default, INTP0 and TI000 are assigned to P00, while TI51 and TOH1 have no assignment setting.

MUXSEL of 78K0/KA2-L (32-pin products) assigns TOH1, TI000, and INTP0 pins. By default, INTP0 and TI000 and TOH1 have no assignment setting.

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

Reset signal generation clears MUXSEL to 00H.

**Figure 6-10. Format of Port Alternate Switch Control Register (MUXSEL)**

**(1) 78K0/KA2-L (25-pin products only)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | 0 | INTP0SEL0 | 0 | TM00SEL0 | TM5SEL1 | TM5SEL0 | TMHSEL1 | TMHSEL0 |

| TM00SEL0 | 16-bit timer 00 input (TI000) pin function assignment |
|---|---|
| 0 | Assign TI000 to the P00 pin as the alternate function. |
| 1 | Assign TI000 to the P121 pin as the alternate function. |

**(2) 78K0/KA2-L (32-pin products only)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | INTP0SEL1 | INTP0SEL0 | TM00SEL1 | TM00SEL0 | 0 | 0 | 0 | TMHSEL0 |

| TM00SEL1 | TM00SEL0 | 16-bit timer 00 input (TI000) pin function assignment |
|---|---|---|
| 0 | 0 | No TI000 function assignment. |
| 0 | 1 | Assign TI000 to the P121 pin as the alternate function. |
| 1 | 0 | Assign TI000 to the P125 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

**(6) Port mode register 0 (PM0)**

This register sets port 0 input/output in 1-bit units.

When using the P01/TO00/TI010 pin for timer output, set PM01 and the output latches of P01 to 0.

When using the P00/TI000 (P00/TI000/INTP0 in the 78K0/KY2-L and 78K0/KA2-L) and P01/TI010/TO00 pins for timer input, set PM00 and PM01 to 1. At this time, the output latches of P00 and P01 may be 0 or 1.

PM0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets PM0 to FFH.

**Figure 6-11. Format of Port Mode Register 0 (PM0)**

Address: FF20H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | PM02 | PM01 | PM00 |

| PM0n | P0n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 0 of 48-pin products (78K0/KC2-L). For the format of port mode register 0 of other products, refer to **(1) Port mode registers (PMxx)** in **4.3 Registers Controlling Port Function**.

## 6.4 Operation of 16-Bit Timer/Event Counter 00

### 6.4.1 Interval timer operation

If bits 3 and 2 (TMC003 and TMC002) of the 16-bit timer mode control register (TMC00) are set to 11 (clear & start mode entered upon a match between TM00 and CR000), the count operation is started in synchronization with the count clock.

When the value of TM00 later matches the value of CR000, TM00 is cleared to 0000H and a match interrupt signal (INTTM000) is generated. This INTTM000 signal enables TM00 to operate as an interval timer.

**Remarks 1.** For the setting of I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 6-12. Block Diagram of Interval Timer Operation**

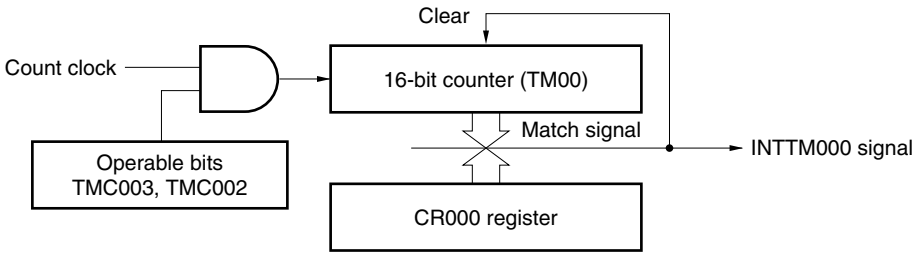



**Figure 6-13. Basic Timing Example of Interval Timer Operation**

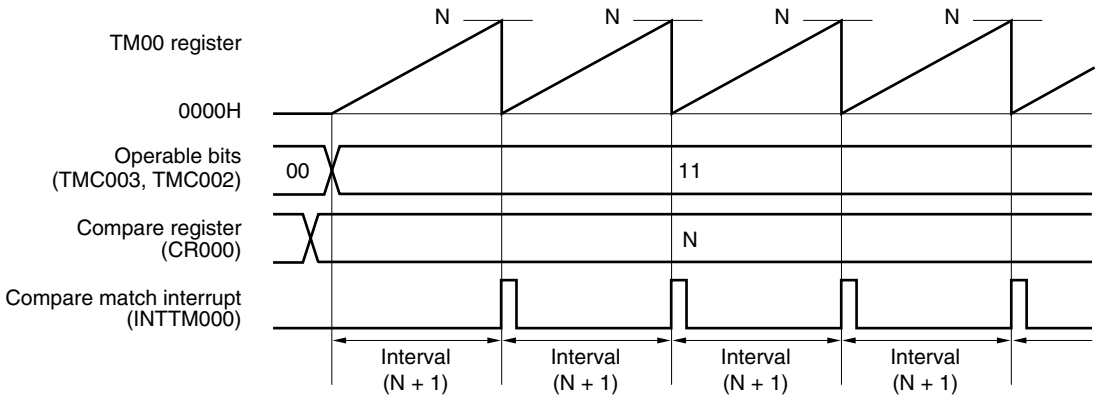


**Figure 6-14. Example of Register Settings for Interval Timer Operation**

**(a) 16-bit timer mode control register 00 (TMC00)**

| | | | | TMC003 | TMC002 | TMC001 | OVF00 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

Clears and starts on match between TM00 and CR000.

**(b) Capture/compare control register 00 (CRC00)**

| | | | | | CRC002 | CRC001 | CRC000 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CR000 used as compare register

**(c) 16-bit timer output control register 00 (TOC00)**

| | OSPT00 | OSPE00 | TOC004 | LVS00 | LVR00 | TOC001 | TOE00 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**(d) Prescaler mode register 00 (PRM00)**

| ES110 | ES100 | ES010 | ES000 | 3 | 2 | PRM001 | PRM000 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0/1 | 0/1 |

Selects count clock

**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

If M is set to CR000, the interval time is as follows.

- Interval time = (M + 1) × Count clock cycle

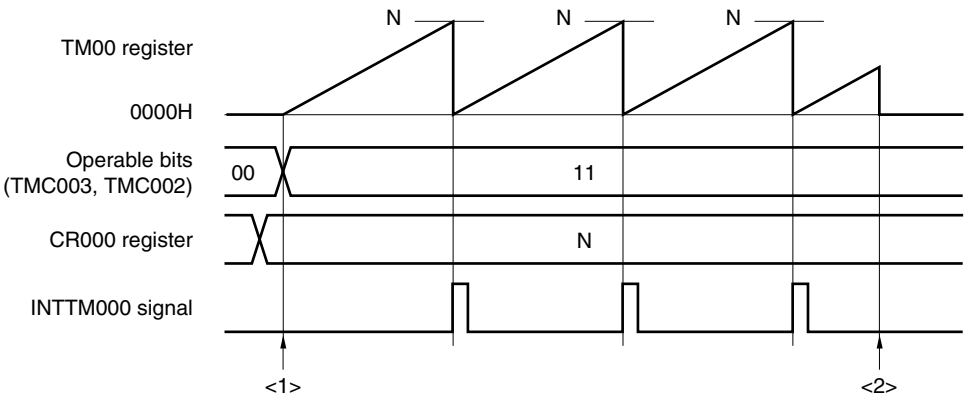
Setting CR000 to 0000H is prohibited.

**(g) 16-bit capture/compare register 010 (CR010)**

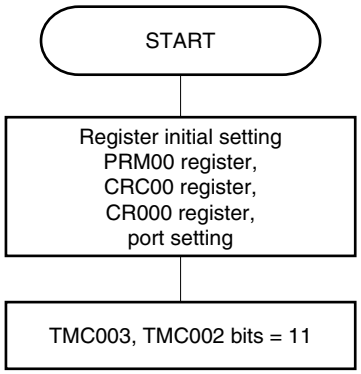
Usually, CR010 is not used for the interval timer function. However, a compare match interrupt (INTTM010) is generated when the set value of CR010 matches the value of TM00.

Therefore, mask the interrupt request by using the interrupt mask flag (TMMK010).

**Figure 6-15. Example of Software Processing for Interval Timer Function**



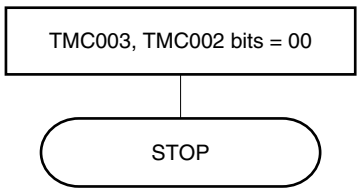

<1> Count operation start flow




Initial setting of these registers is performed before setting the TMC003 and TMC002 bits to 11.

Starts count operation

<2> Count operation stop flow




The counter is initialized and counting is stopped by clearing the TMC003 and TMC002 bits to 00.

### 6.4.2 Square-wave output operation

When 16-bit timer/event counter 00 operates as an interval timer (refer to **6.4.1**), a square wave can be output from the TO00 pin by setting the 16-bit timer output control register 00 (TOC00) to 03H.

When TMC003 and TMC002 are set to 11 (count clear & start mode entered upon a match between TM00 and CR000), the counting operation is started in synchronization with the count clock.

When the value of TM00 later matches the value of CR000, TM00 is cleared to 0000H, an interrupt signal (INTTM000) is generated, and TO00 output is inverted. This TO00 output that is inverted at fixed intervals enables TO0n to output a square wave.

**Remarks 1.** For the setting of I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 6-16. Block Diagram of Square-Wave Output Operation**

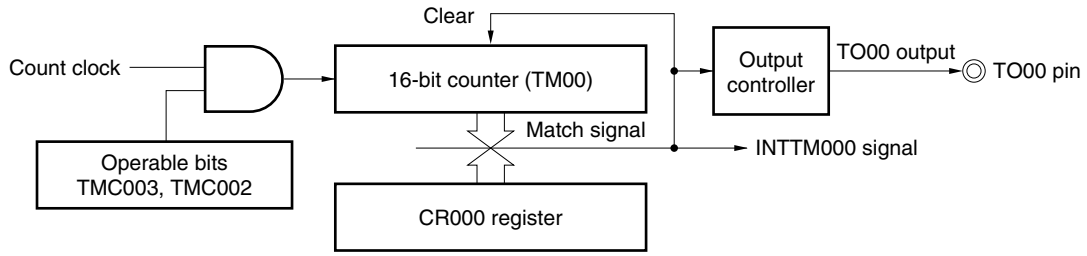



**Figure 6-17. Basic Timing Example of Square-Wave Output Operation**

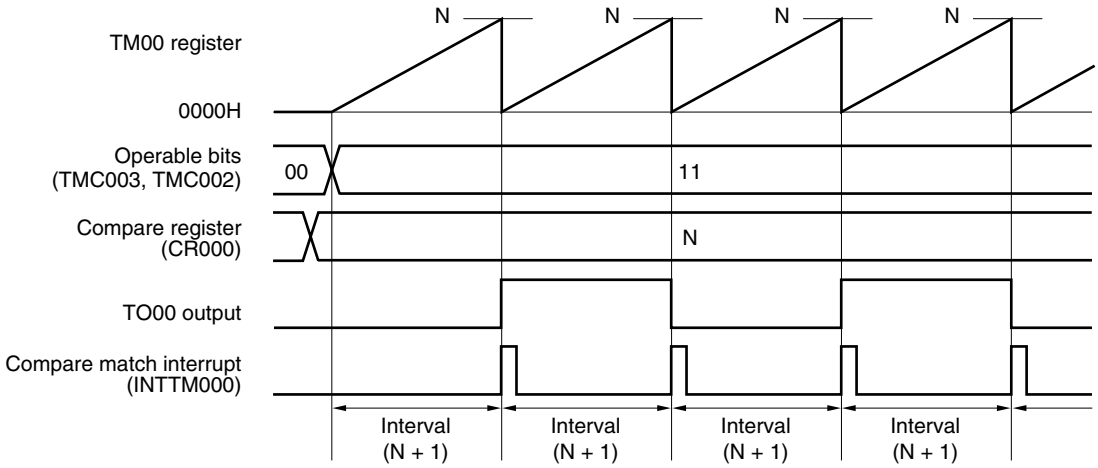
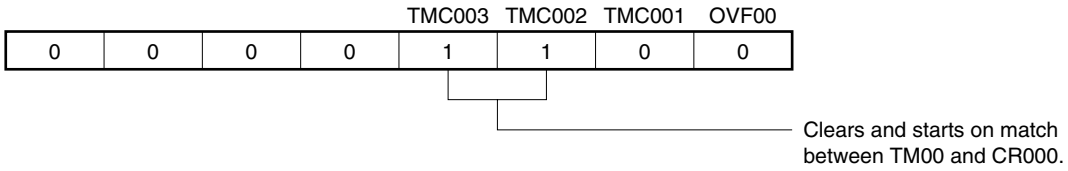
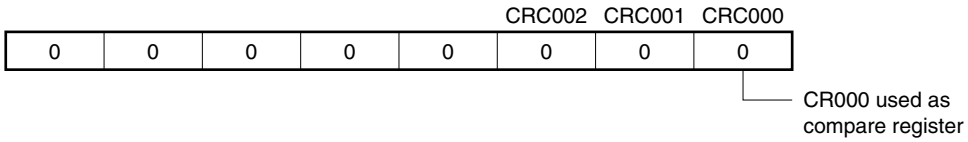


**Figure 6-18. Example of Register Settings for Square-Wave Output Operation**

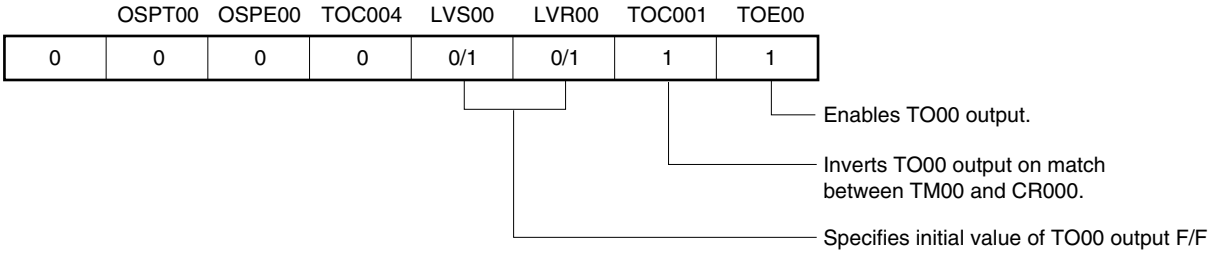
**(a) 16-bit timer mode control register 00 (TMC00)**



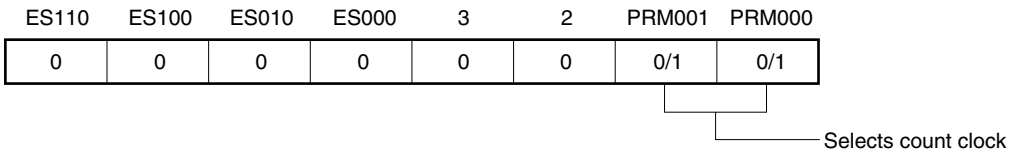

**(b) Capture/compare control register 00 (CRC00)**




**(c) 16-bit timer output control register 00 (TOC00)**




**(d) Prescaler mode register 00 (PRM00)**




**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

If M is set to CR000, the interval time is as follows.

- Square wave frequency = 1 / [2 × (M + 1) × Count clock cycle]

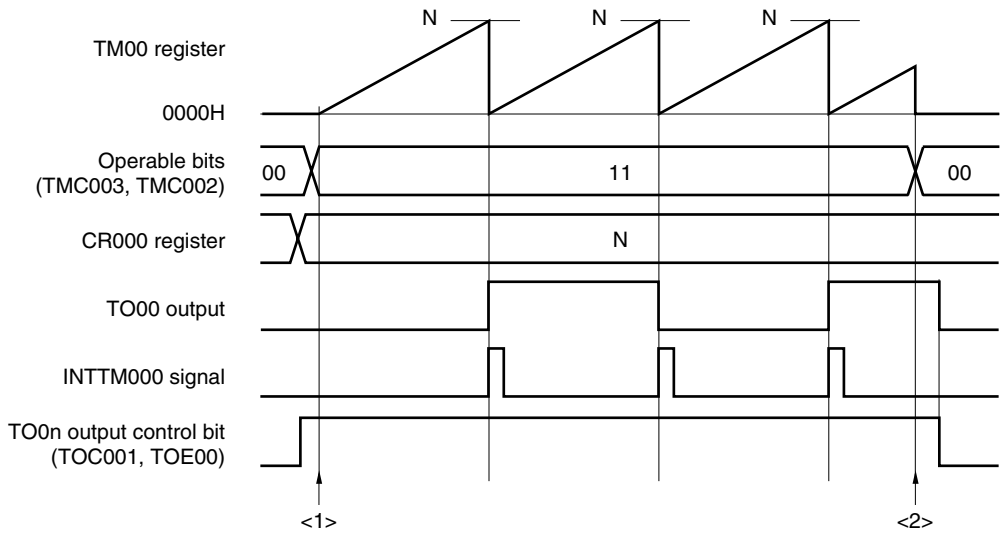
Setting CR000 to 0000H is prohibited.

**(g) 16-bit capture/compare register 010 (CR010)**

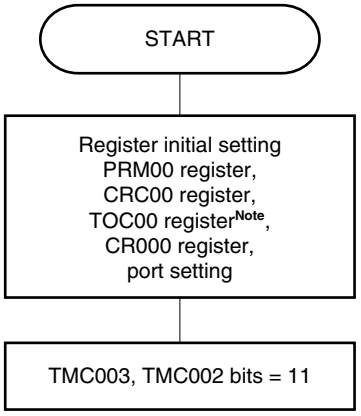
Usually, CR010 is not used for the square-wave output function. However, a compare match interrupt (INTTM010) is generated when the set value of CR010 matches the value of TM00.

Therefore, mask the interrupt request by using the interrupt mask flag (TMMK010).

**Figure 6-19. Example of Software Processing for Square-Wave Output Function**



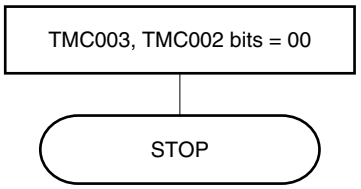

<1> Count operation start flow




Initial setting of these registers is performed before setting the TMC003 and TMC002 bits to 11.

Starts count operation

<2> Count operation stop flow




The counter is initialized and counting is stopped by clearing the TMC003 and TMC002 bits to 00.

**Note** Care must be exercised when setting TOC00. For details, refer to **6.3 (3) 16-bit timer output control register 00 (TOC00)**.

### 6.4.3 External event counter operation

When bits 1 and 0 (PRM001 and PRM000) of the prescaler mode register 00 (PRM00) are set to 11 (for counting up with the valid edge of the TI000 pin) and bits 3 and 2 (TMC003 and TMC002) of 16-bit timer mode control register 00 (TMC00) are set to 11, the valid edge of an external event input is counted, and a match interrupt signal indicating matching between TM00 and CR000 (INTTM000) is generated.

To input the external event, the TI000 pin is used. Therefore, the timer/event counter cannot be used as an external event counter in the clear & start mode entered by the TI000 pin valid edge input (when TMC003 and TMC002 = 10).

The INTTM000 signal is generated with the following timing.

- Timing of generation of INTTM000 signal (second time or later)
  = Number of times of detection of valid edge of external event × (Set value of CR000 + 1)

However, the first match interrupt immediately after the timer/event counter has started operating is generated with the following timing.

- Timing of generation of INTTM000 signal (first time only)
  = Number of times of detection of valid edge of external event input × (Set value of CR000 + 2)

To detect the valid edge, the signal input to the TI000 pin is sampled during the clock cycle of $f_{PRS}$. The valid edge is not detected until it is detected two times in a row. Therefore, a noise with a short pulse width can be eliminated.

**Remarks 1.** For the setting of I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 6-20. Block Diagram of External Event Counter Operation**

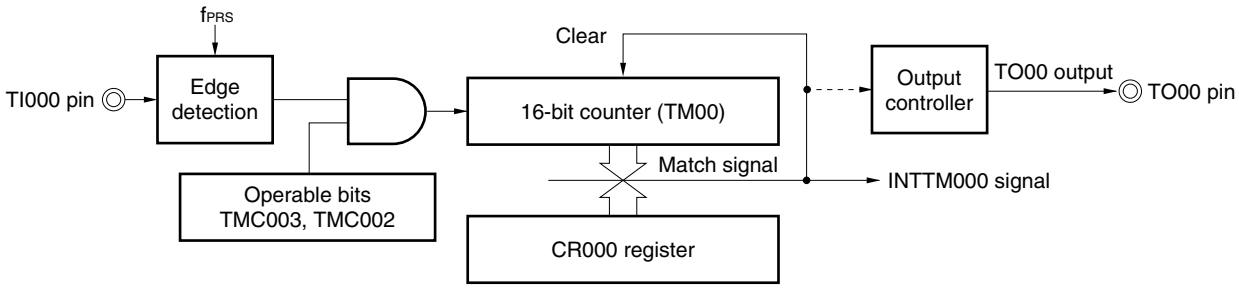
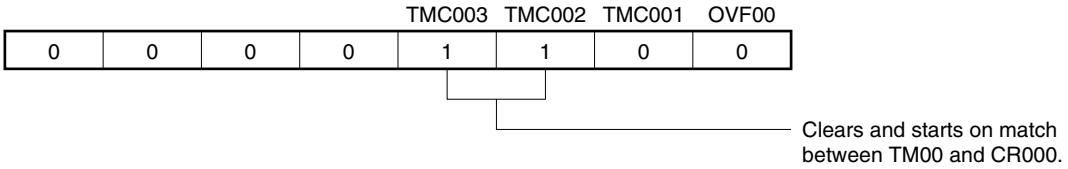
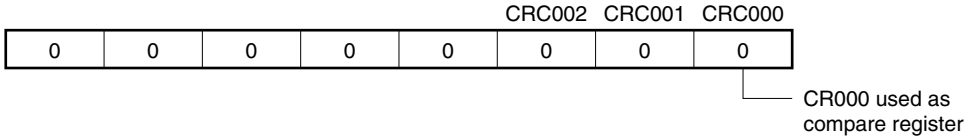


**Figure 6-21. Example of Register Settings in External Event Counter Mode (1/2)**

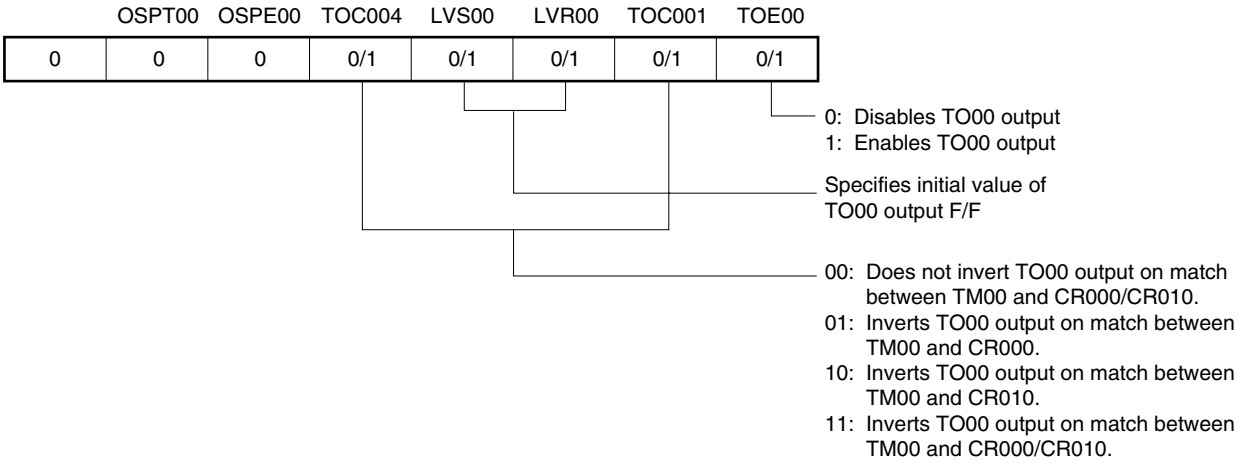
**(a) 16-bit timer mode control register 00 (TMC00)**




**(b) Capture/compare control register 00 (CRC00)**




**(c) 16-bit timer output control register 00 (TOC00)**




**(d) Prescaler mode register 00 (PRM00)**

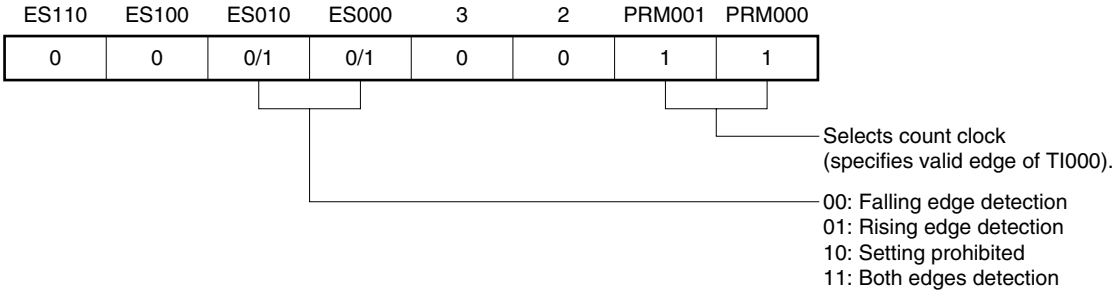


**Figure 6-21. Example of Register Settings in External Event Counter Mode (2/2)**

**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

If M is set to CR000, the interrupt signal (INTTM000) is generated when the number of external events reaches (M + 1).

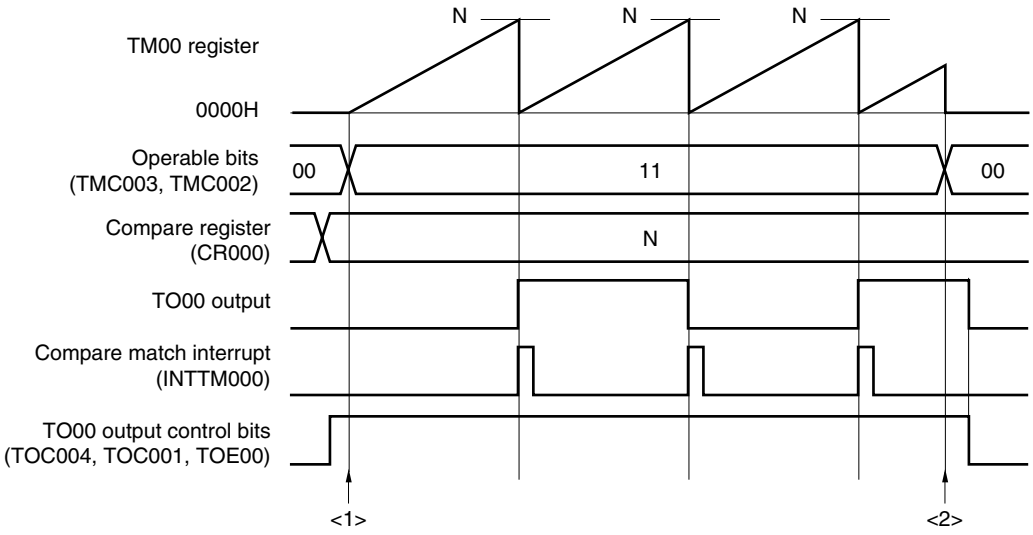
Setting CR000 to 0000H is prohibited.

**(g) 16-bit capture/compare register 010 (CR010)**

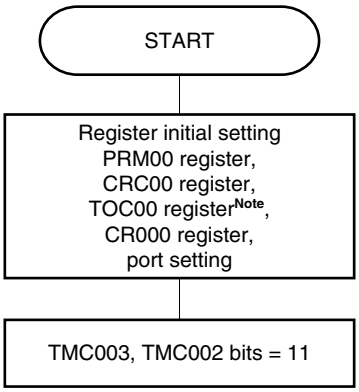
Usually, CR010 is not used in the external event counter mode. However, a compare match interrupt (INTTM010) is generated when the set value of CR010 matches the value of TM00.

Therefore, mask the interrupt request by using the interrupt mask flag (TMMK010).

**Figure 6-22. Example of Software Processing in External Event Counter Mode**



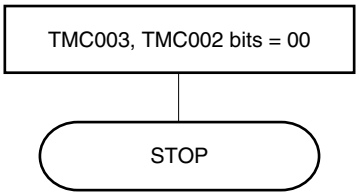

<1> Count operation start flow




Initial setting of these registers is performed before setting the TMC003 and TMC002 bits to 11.

Starts count operation

<2> Count operation stop flow




The counter is initialized and counting is stopped by clearing the TMC003 and TMC002 bits to 00.

**Note** Care must be exercised when setting TOC00. For details, refer to **6.3 (3) 16-bit timer output control register 00 (TOC00)**.

### 6.4.4 Operation in clear & start mode entered by TI000 pin valid edge input

When bits 3 and 2 (TMC003 and TMC002) of 16-bit timer mode control register 00 (TMC00) are set to 10 (clear & start mode entered by the TI000 pin valid edge input) and the count clock (set by PRM00) is supplied to the timer/event counter, TM00 starts counting up. When the valid edge of the TI000 pin is detected during the counting operation, TM00 is cleared to 0000H and starts counting up again. If the valid edge of the TI000 pin is not detected, TM00 overflows and continues counting.

The valid edge of the TI000 pin is a cause to clear TM00. Starting the counter is not controlled immediately after the start of the operation.

CR000 and CR010 are used as compare registers and capture registers.

**(a) When CR000 and CR010 are used as compare registers**

Signals INTTM000 and INTTM010 are generated when the value of TM00 matches the value of CR000 and CR010.

**(b) When CR000 and CR010 are used as capture registers**

The count value of TM00 is captured to CR000 and the INTTM000 signal is generated when the valid edge is input to the TI010 pin (or when the phase reverse to that of the valid edge is input to the TI000 pin).

When the valid edge is input to the TI000 pin, the count value of TM00 is captured to CR010 and the INTTM010 signal is generated. As soon as the count value has been captured, the counter is cleared to 0000H.

**Caution Do not set the count clock as the valid edge of the TI000 pin (PRM001 and PRM000 = 11). When PRM001 and PRM000 = 11, TM00 is cleared.**

**Remarks 1.** For the setting of the I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**(1) Operation in clear & start mode entered by TI000 pin valid edge input (CR000: compare register, CR010: compare register)**

**Figure 6-23. Block Diagram of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Compare Register, CR010: Compare Register)**

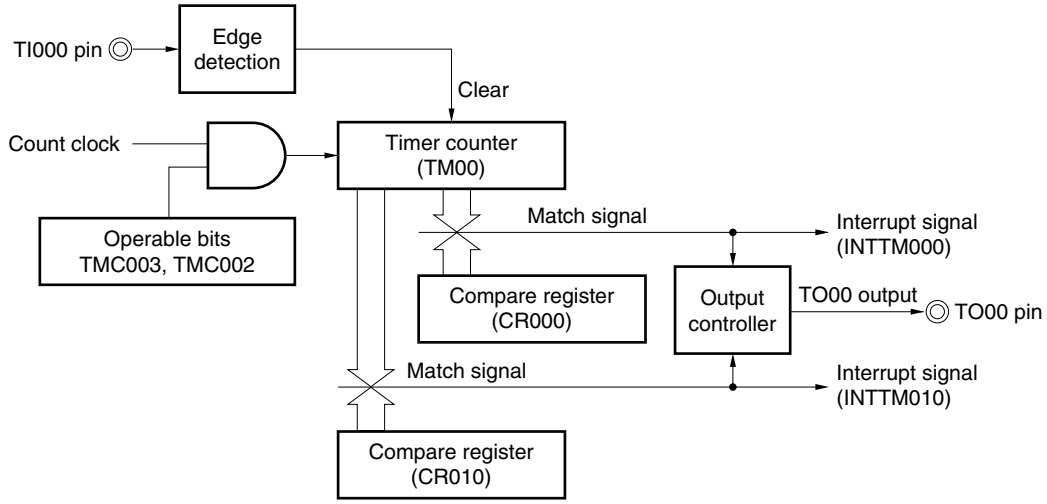
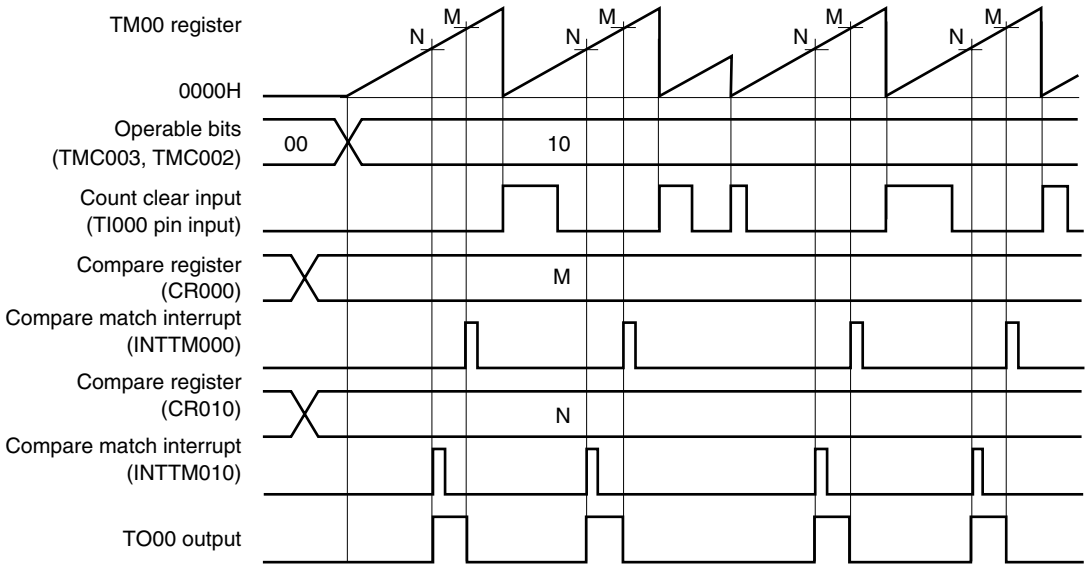
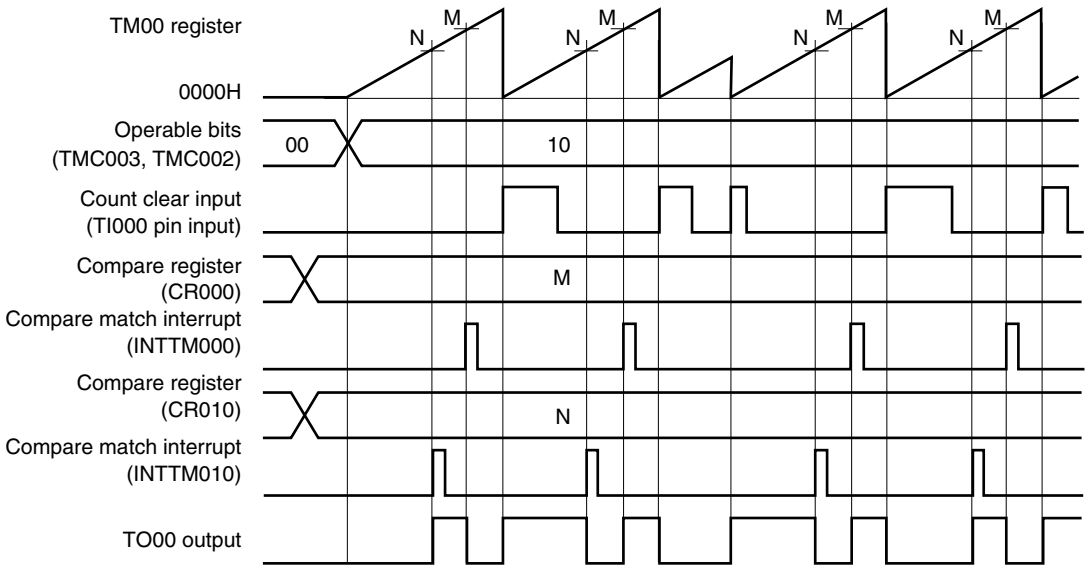


**Figure 6-24. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Compare Register, CR010: Compare Register)**

**(a) TOC00 = 13H, PRM00 = 10H, CRC00 = 00H, TMC00 = 08H**




**(b) TOC00 = 13H, PRM00 = 10H, CRC00 = 00H, TMC00 = 0AH**




(a) and (b) differ as follows depending on the setting of bit 1 (TMC001) of the 16-bit timer mode control register 00 (TMC00).

(a) The TO00 output level is inverted when TM00 matches a compare register.

(b) The TO00 output level is inverted when TM00 matches a compare register or when the valid edge of the TI000 pin is detected.

**(2) Operation in clear & start mode entered by TI000 pin valid edge input**
**(CR000: compare register, CR010: capture register)**

**Figure 6-25. Block Diagram of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Compare Register, CR010: Capture Register)**

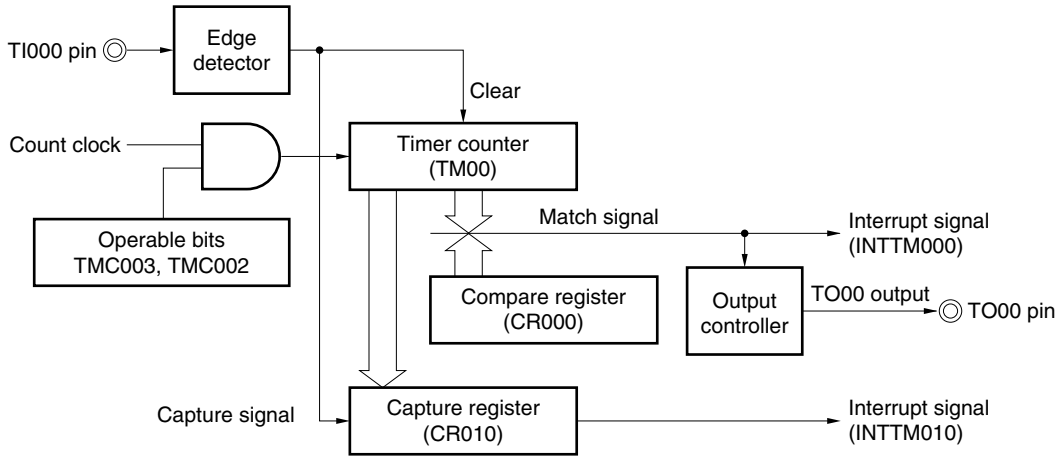
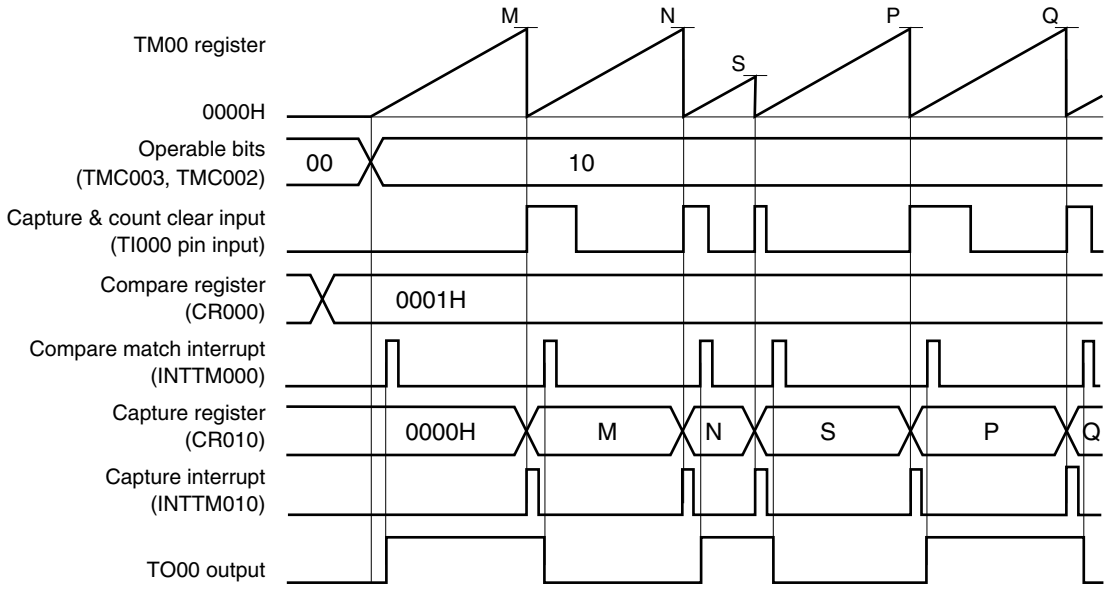



**Figure 6-26. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Compare Register, CR010: Capture Register) (1/2)**

**(a) TOC00 = 13H, PRM00 = 10H, CRC00 = 04H, TMC00 = 08H, CR000 = 0001H**

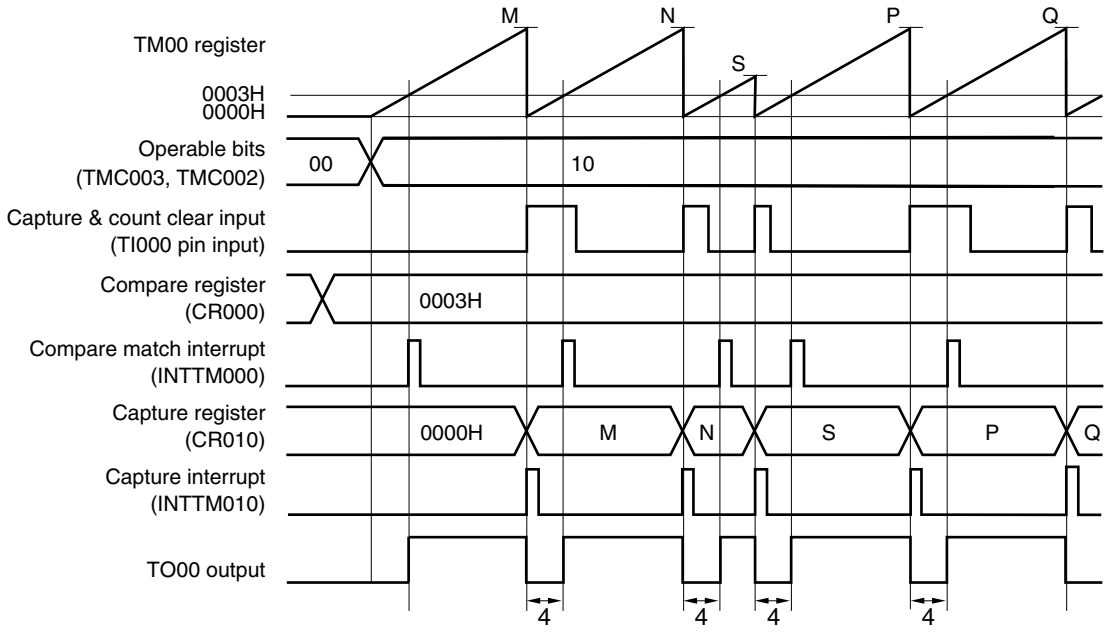



This is an application example where the TO00 output level is inverted when the count value has been captured & cleared.

The count value is captured to CR010 and TM00 is cleared (to 0000H) when the valid edge of the TI000 pin is detected. When the count value of TM00 is 0001H, a compare match interrupt signal (INTTM000) is generated, and the TO00 output level is inverted.

**Figure 6-26. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Compare Register, CR010: Capture Register) (2/2)**

**(b) TOC00 = 13H, PRM00 = 10H, CRC00 = 04H, TMC00 = 0AH, CR000 = 0003H**




This is an application example where the width set to CR000 (4 clocks in this example) is to be output from the TO00 pin when the count value has been captured & cleared.

The count value is captured to CR010, a capture interrupt signal (INTTM010) is generated, TM00 is cleared (to 0000H), and the TO00 output level is inverted when the valid edge of the TI000 pin is detected. When the count value of TM00 is 0003H (four clocks have been counted), a compare match interrupt signal (INTTM000) is generated and the TO00 output level is inverted.

**(3) Operation in clear & start mode by entered TI000 pin valid edge input (CR000: capture register, CR010: compare register)**

**Figure 6-27. Block Diagram of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Compare Register)**

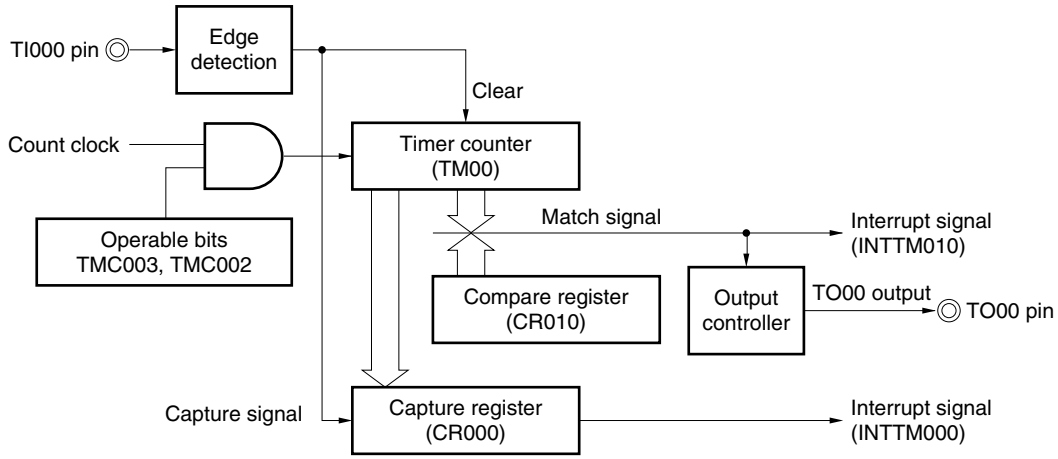
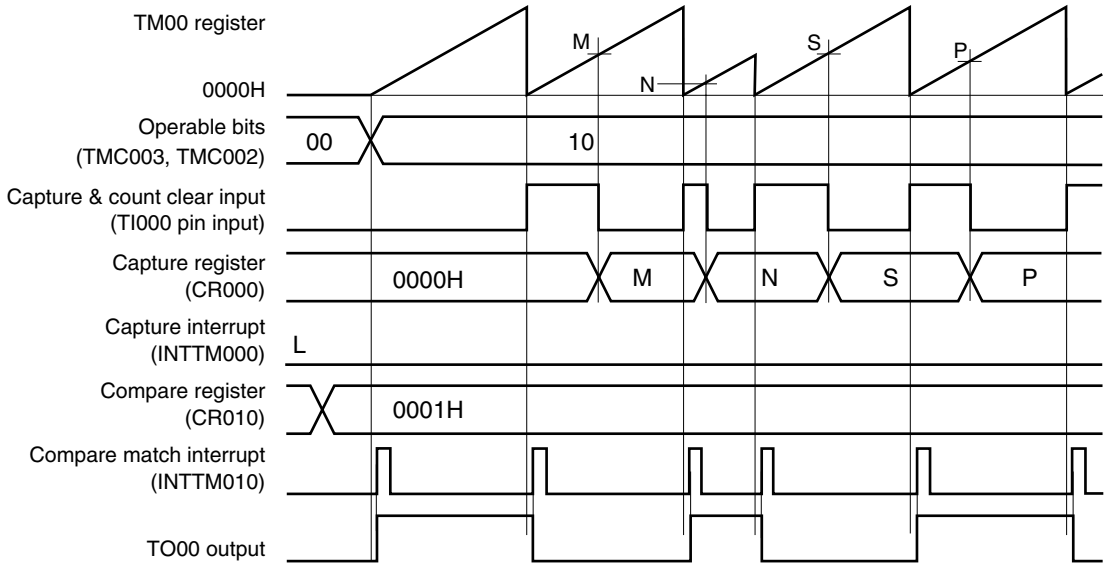


**Figure 6-28. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Compare Register) (1/2)**

**(a) TOC00 = 13H, PRM00 = 10H, CRC00 = 03H, TMC00 = 08H, CR010 = 0001H**



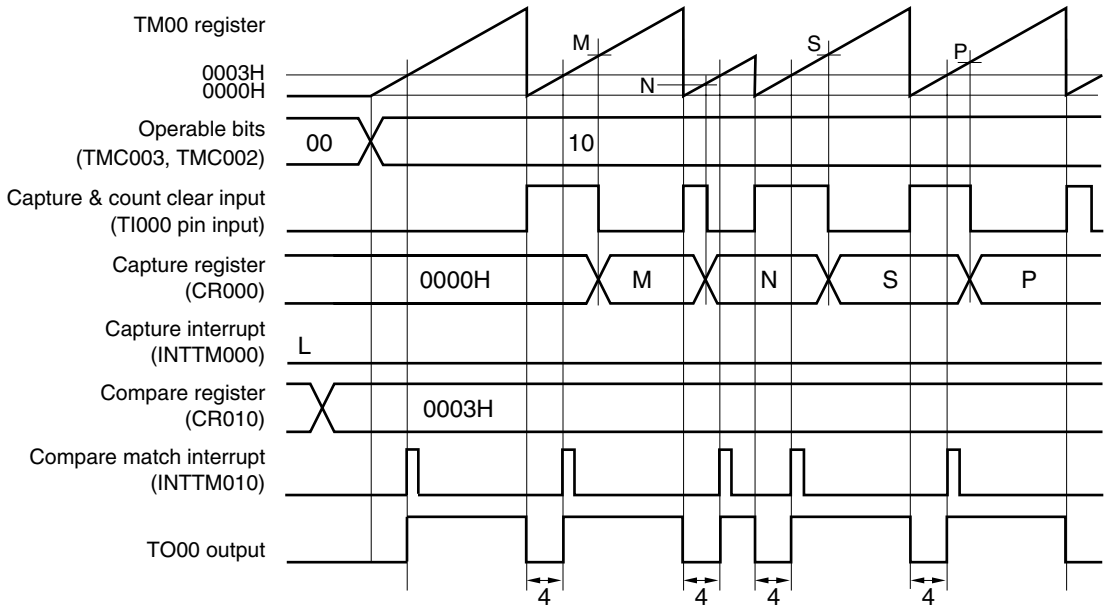

This is an application example where the TO00 output level is to be inverted when the count value has been captured & cleared.

TM00 is cleared at the rising edge detection of the TI000 pin and it is captured to CR000 at the falling edge detection of the TI000 pin.

When bit 1 (CRC001) of capture/compare control register 00 (CRC00) is set to 1, the count value of TM00 is captured to CR000 in the phase reverse to that of the signal input to the TI000 pin, but the capture interrupt signal (INTTM000) is not generated. However, the INTTM000 signal is generated when the valid edge of the TI010 pin is detected. Mask the INTTM000 signal when it is not used.

**Figure 6-28. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Compare Register) (2/2)**

**(b) TOC00 = 13H, PRM00 = 10H, CRC00 = 03H, TMC00 = 0AH, CR010 = 0003H**



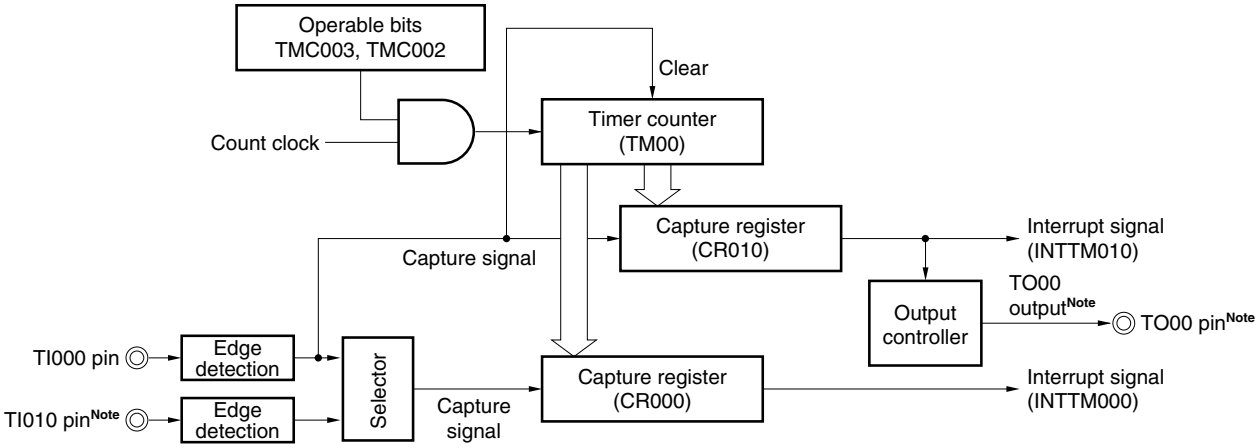

This is an application example where the width set to CR010 (4 clocks in this example) is to be output from the TO00 pin when the count value has been captured & cleared.

TM00 is cleared (to 0000H) at the rising edge detection of the TI000 pin and captured to CR000 at the falling edge detection of the TI000 pin. The TO00 output level is inverted when TM00 is cleared (to 0000H) because the rising edge of the TI000 pin has been detected or when the value of TM00 matches that of a compare register (CR010).

When bit 1 (CRC001) of capture/compare control register 00 (CRC00) is 1, the count value of TM00 is captured to CR000 in the phase reverse to that of the input signal of the TI000 pin, but the capture interrupt signal (INTTM000) is not generated. However, the INTTM000 interrupt is generated when the valid edge of the TI010 pin is detected. Mask the INTTM000 signal when it is not used.

### (4) Operation in clear & start mode entered by TI000 pin valid edge input (CR000: capture register, CR010: capture register)

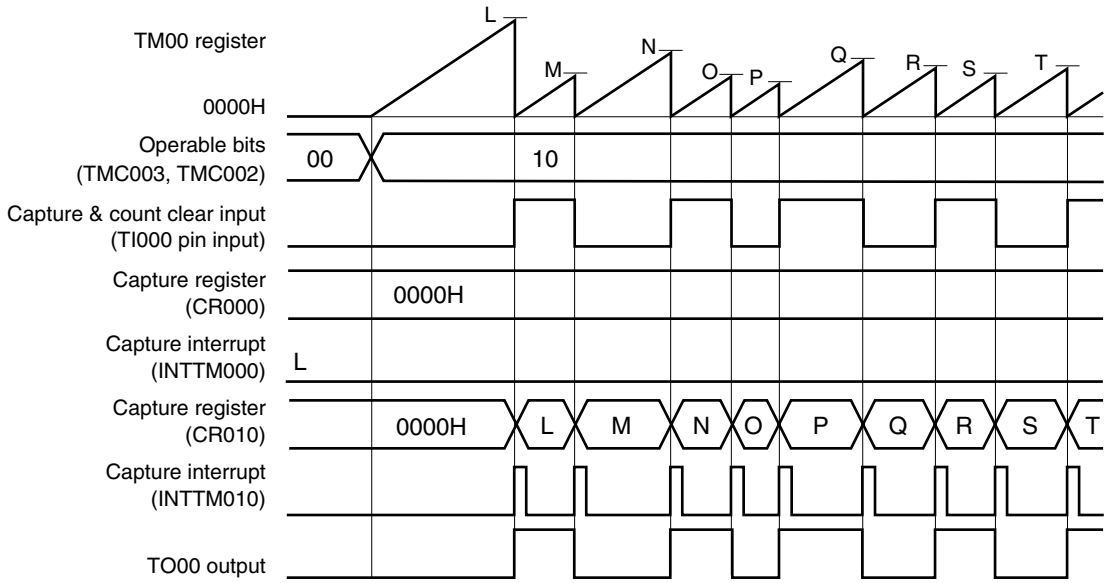
**Figure 6-29. Block Diagram of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Capture Register)**




**Note** The timer output (TO00) cannot be used when detecting the valid edge of the TI010 pin is used.

**Figure 6-30. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Capture Register) (1/3)**

**(a) TOC00 = 13H, PRM00 = 30H, CRC00 = 05H, TMC00 = 0AH**

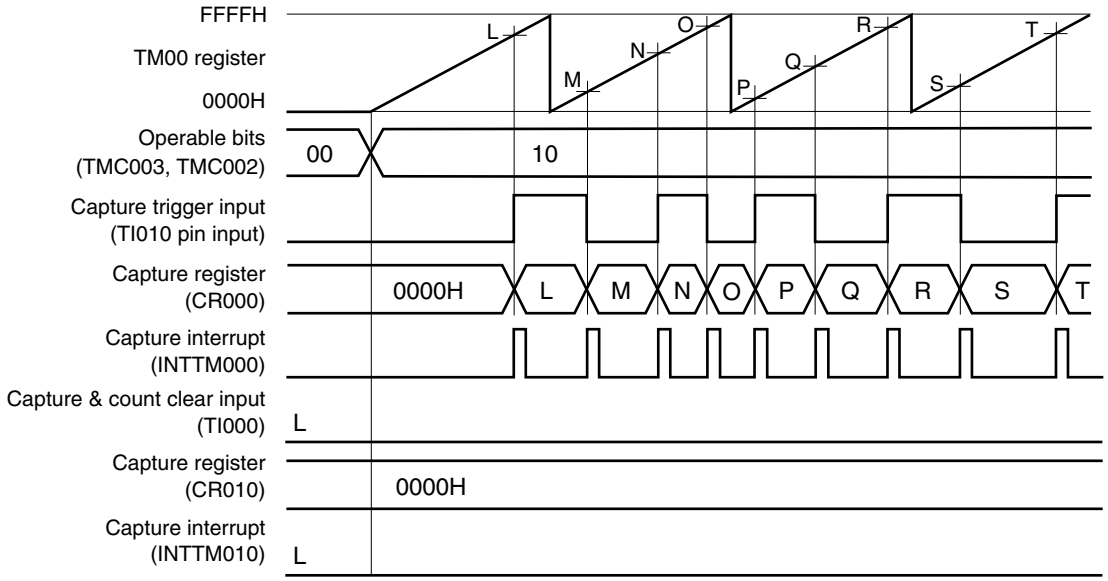



This is an application example where the count value is captured to CR010, TM00 is cleared, and the TO00 output is inverted when the rising or falling edge of the TI000 pin is detected.

When the edge of the TI010 pin is detected, an interrupt signal (INTTM000) is generated. Mask the INTTM000 signal when it is not used.

**Figure 6-30. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Capture Register) (2/3)**

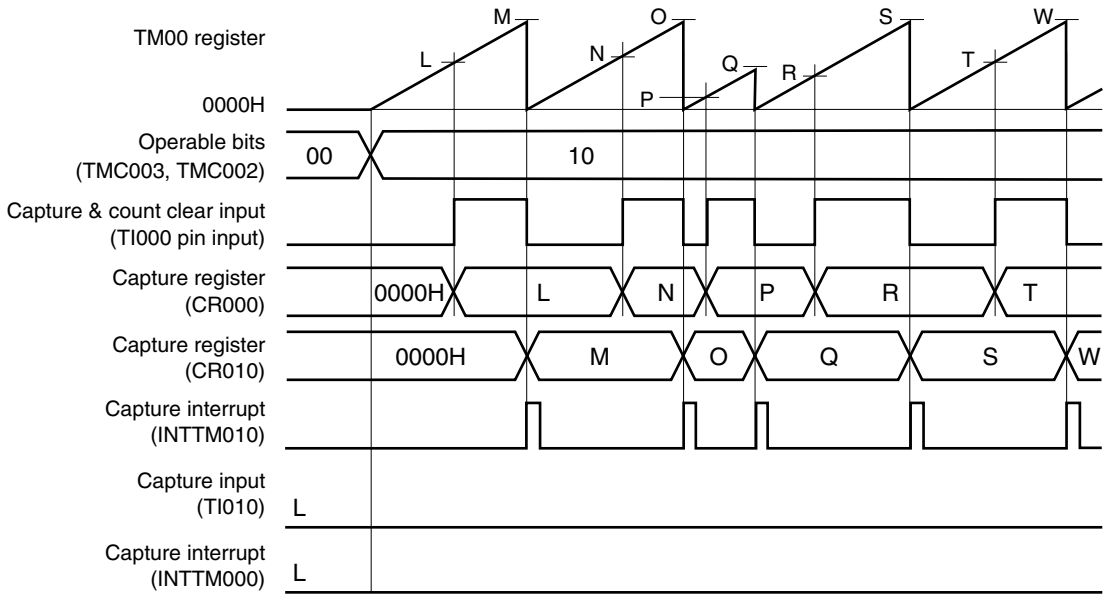
**(b) TOC00 = 13H, PRM00 = C0H, CRC00 = 05H, TMC00 = 0AH**




This is a timing example where an edge is not input to the TI000 pin, in an application where the count value is captured to CR000 when the rising or falling edge of the TI010 pin is detected.

**Figure 6-30. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Capture Register, CR010: Capture Register) (3/3)**

**(c) TOC00 = 13H, PRM00 = 00H, CRC00 = 07H, TMC00 = 0AH**



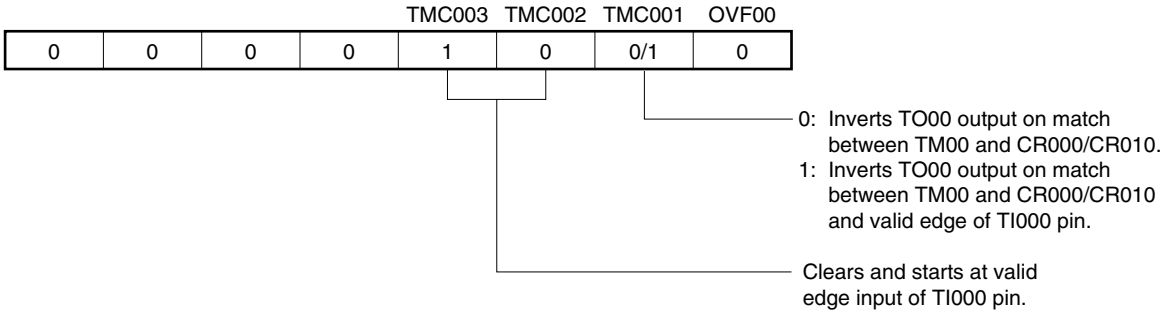

This is an application example where the pulse width of the signal input to the TI000 pin is measured.
By setting CRC00, the count value can be captured to CR000 in the phase reverse to the falling edge of the TI000 pin (i.e., rising edge) and to CR010 at the falling edge of the TI000 pin.
The high- and low-level widths of the input pulse can be calculated by the following expressions.

- High-level width = [CR010 value] – [CR000 value] × [Count clock cycle]
- Low-level width = [CR000 value] × [Count clock cycle]

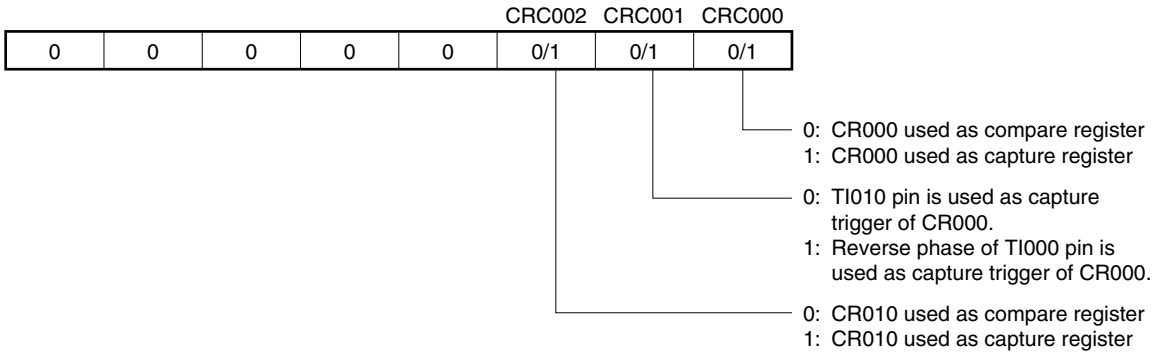
If the reverse phase of the TI000 pin is selected as a trigger to capture the count value to CR000, the INTTM000 signal is not generated. Read the values of CR000 and CR010 to measure the pulse width immediately after the INTTM010 signal is generated.
However, if the valid edge specified by bits 6 and 5 (ES110 and ES100) of prescaler mode register 00 (PRM00) is input to the TI010 pin, the count value is not captured but the INTTM000 signal is generated. To measure the pulse width of the TI000 pin, mask the INTTM000 signal when it is not used.

**Figure 6-31. Example of Register Settings in Clear & Start Mode Entered by TI000 Pin Valid Edge Input (1/2)**

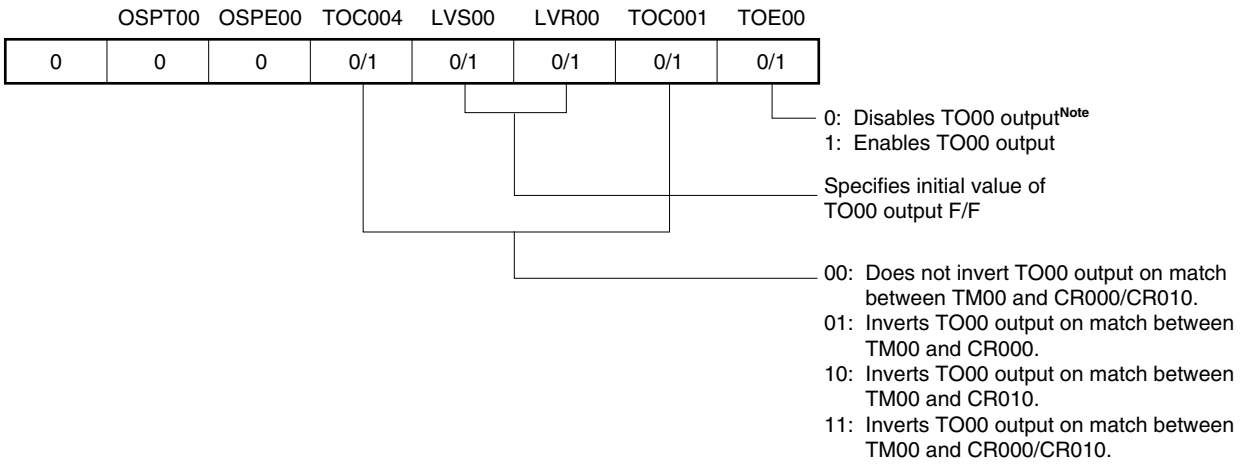
**(a) 16-bit timer mode control register 00 (TMC00)**




**(b) Capture/compare control register 00 (CRC00)**



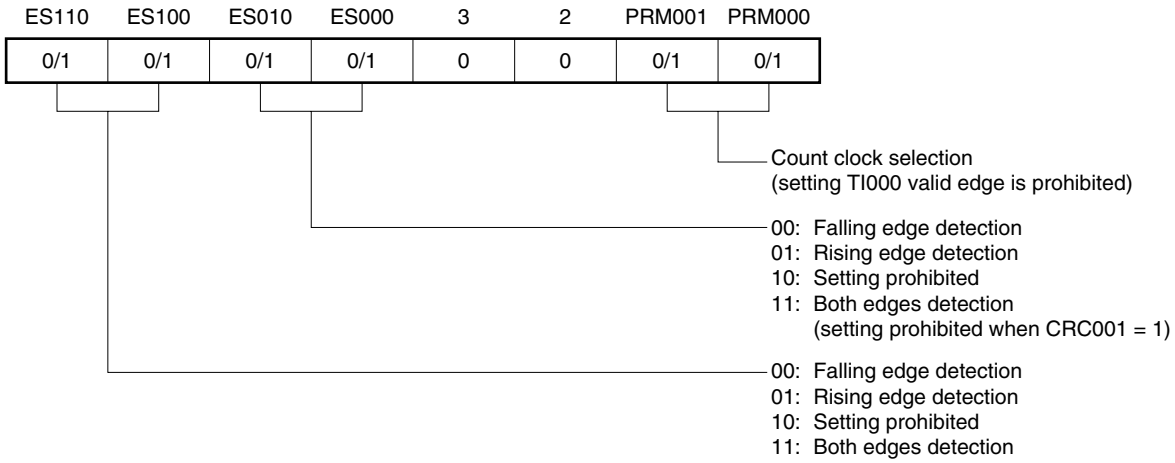

**(c) 16-bit timer output control register 00 (TOC00)**




**Note** The timer output (TO00) cannot be used when detecting the valid edge of the TI010 pin is used.

**Figure 6-31. Example of Register Settings in Clear & Start Mode Entered by TI000 Pin Valid Edge Input (2/2)**

**(d) Prescaler mode register 00 (PRM00)**




**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

When this register is used as a compare register and when its value matches the count value of TM00, an interrupt signal (INTTM000) is generated. The count value of TM00 is not cleared.

To use this register as a capture register, select either the TI000 or TI010 pin[Note] input as a capture trigger. When the valid edge of the capture trigger is detected, the count value of TM00 is stored in CR000.

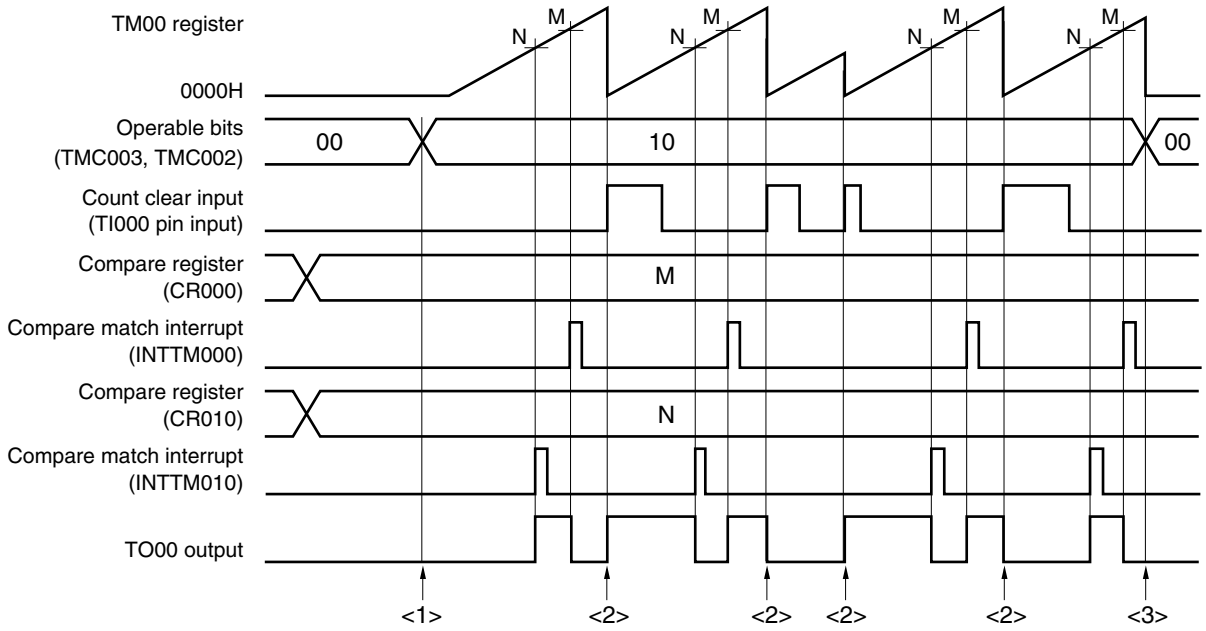
**Note** The timer output (TO00) cannot be used when detection of the valid edge of the TI010 pin is used.

**(g) 16-bit capture/compare register 010 (CR010)**

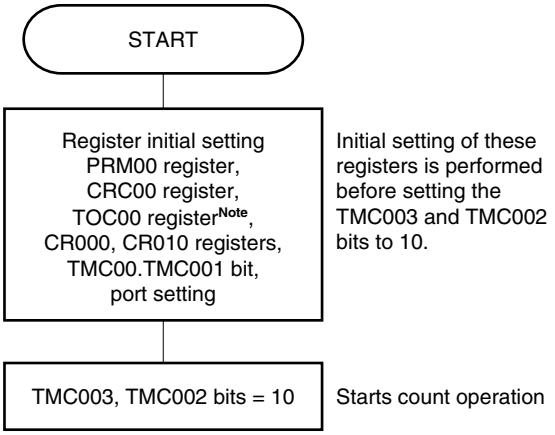
When this register is used as a compare register and when its value matches the count value of TM00, an interrupt signal (INTTM010) is generated. The count value of TM00 is not cleared.

When this register is used as a capture register, the TI000 pin input is used as a capture trigger. When the valid edge of the capture trigger is detected, the count value of TM00 is stored in CR010.

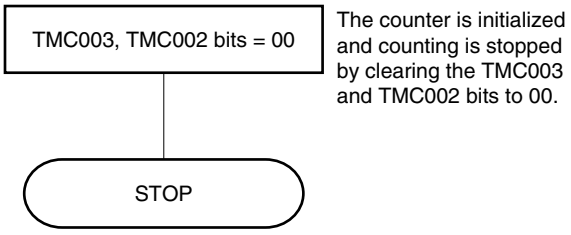
**Figure 6-32. Example of Software Processing in Clear & Start Mode Entered by TI000 Pin Valid Edge Input**



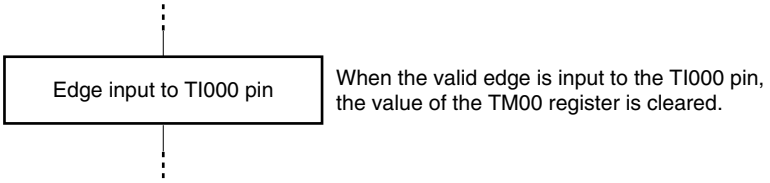

<1> Count operation start flow




<3> Count operation stop flow




<2> TM00 register clear & start flow




**Note** Care must be exercised when setting TOC00. For details, refer to **6.3 (3) 16-bit timer output control register 00 (TOC00)**.

### 6.4.5 Free-running timer operation

When bits 3 and 2 (TMC003 and TMC002) of 16-bit timer mode control register 00 (TMC00) are set to 01 (free-running timer mode), 16-bit timer/event counter 00 continues counting up in synchronization with the count clock. When it has counted up to FFFFH, the overflow flag (OVF00) is set to 1 at the next clock, and TM00 is cleared (to 0000H) and continues counting. Clear OVF00 to 0 by executing the CLR instruction via software.

The following three types of free-running timer operations are available.

- Both CR000 and CR010 are used as compare registers.
- One of CR000 or CR010 is used as a compare register and the other is used as a capture register.
- Both CR000 and CR010 are used as capture registers.

**Remarks 1.** For the setting of the I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

#### (1) Free-running timer mode operation

**(CR000: compare register, CR010: compare register)**

**Figure 6-33. Block Diagram of Free-Running Timer Mode**
**(CR000: Compare Register, CR010: Compare Register)**

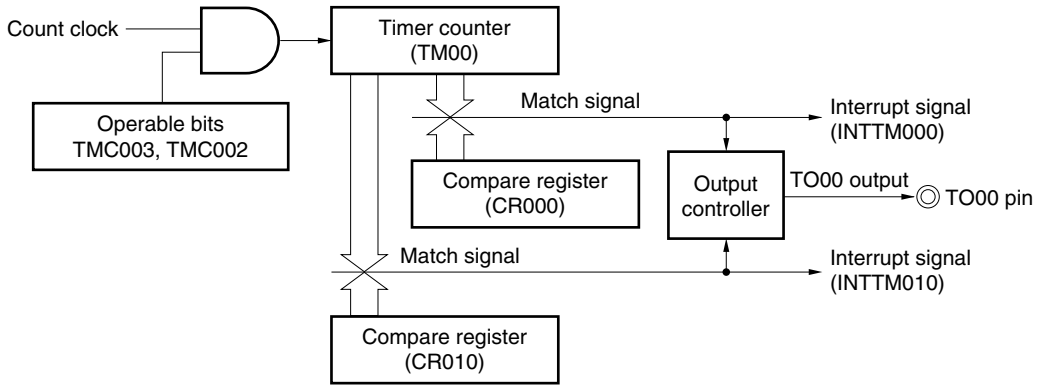
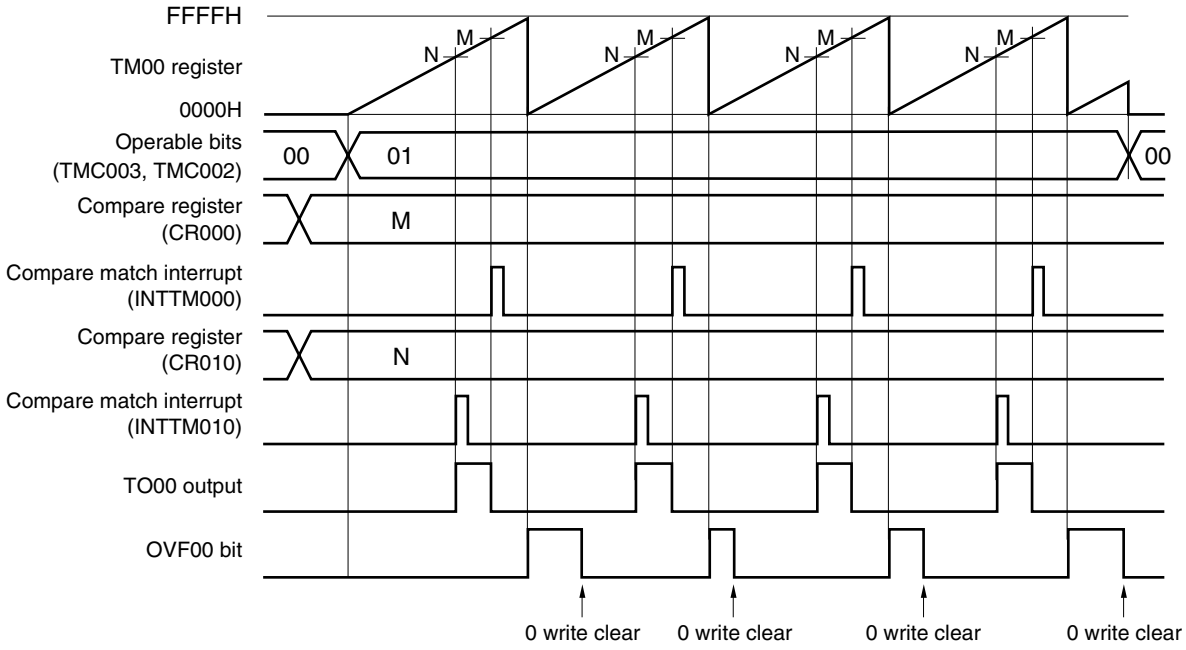


**Figure 6-34. Timing Example of Free-Running Timer Mode**
**(CR000: Compare Register, CR010: Compare Register)**

**• TOC00 = 13H, PRM00 = 00H, CRC00 = 00H, TMC00 = 04H**




This is an application example where two compare registers are used in the free-running timer mode.

The TO00 output level is reversed each time the count value of TM00 matches the set value of CR000 or CR010. When the count value matches the register value, the INTTM000 or INTTM010 signal is generated.

**(2) Free-running timer mode operation**
**(CR000: compare register, CR010: capture register)**

**Figure 6-35. Block Diagram of Free-Running Timer Mode**
**(CR000: Compare Register, CR010: Capture Register)**

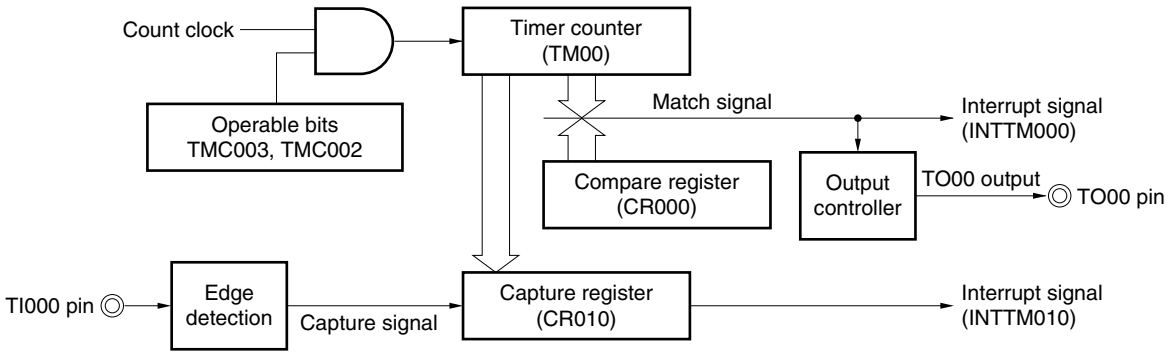
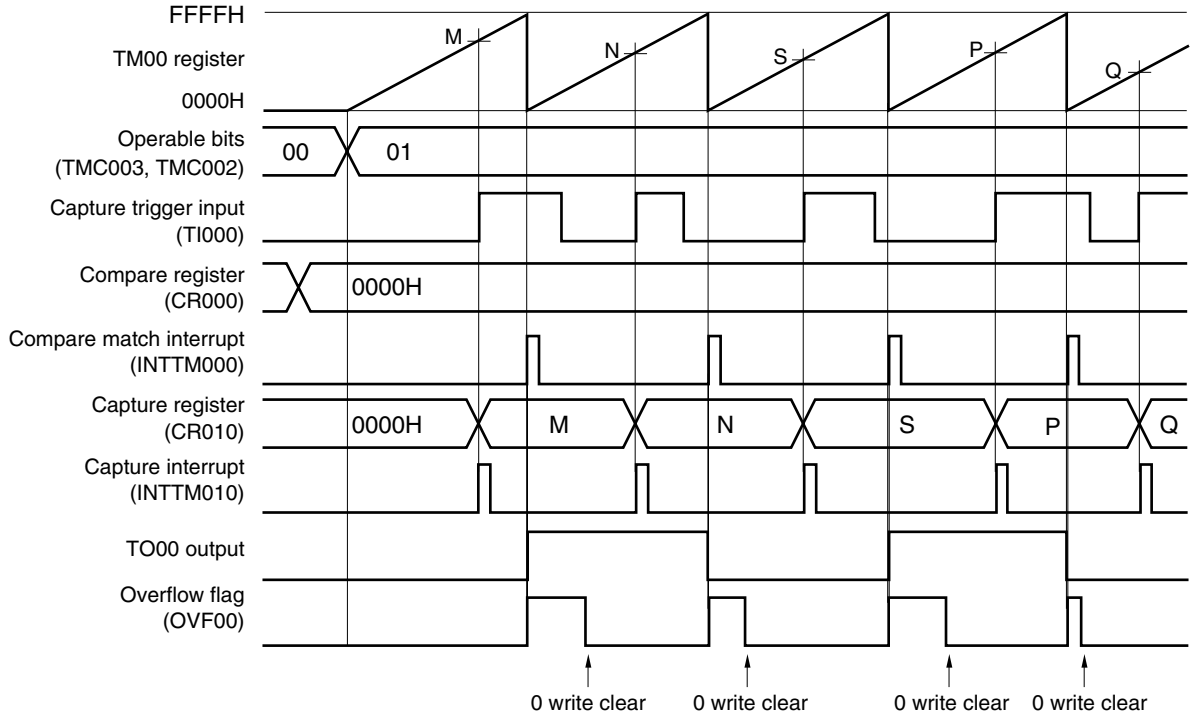


**Figure 6-36. Timing Example of Free-Running Timer Mode**
**(CR000: Compare Register, CR010: Capture Register)**

**• TOC00 = 13H, PRM00 = 10H, CRC00 = 04H, TMC00 = 04H**



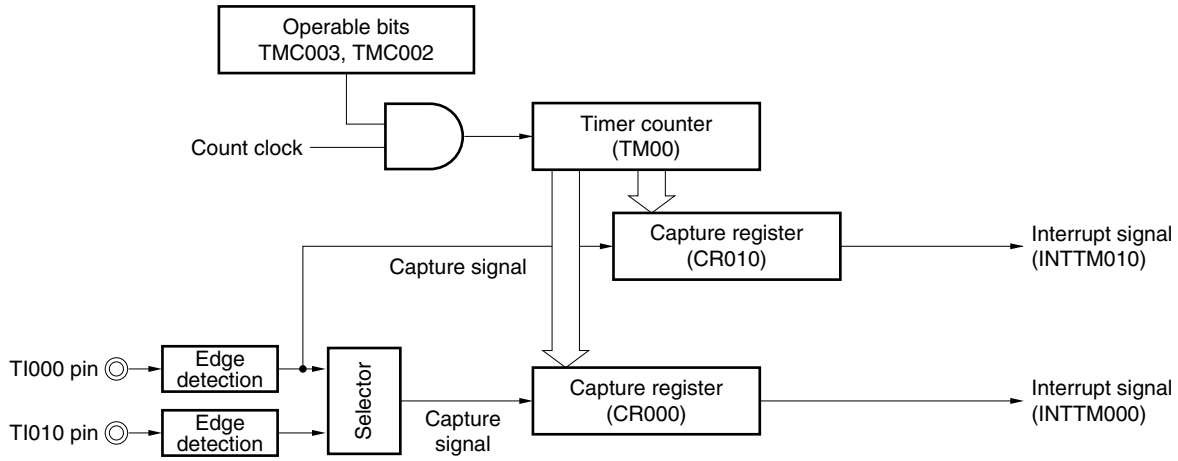

This is an application example where a compare register and a capture register are used at the same time in the free-running timer mode.

In this example, the INTTM000 signal is generated and the TO00 output level is reversed each time the count value of TM00 matches the set value of CR000 (compare register). In addition, the INTTM010 signal is generated and the count value of TM00 is captured to CR010 each time the valid edge of the TI000 pin is detected.

**(3) Free-running timer mode operation**

**(CR000: capture register, CR010: capture register)**

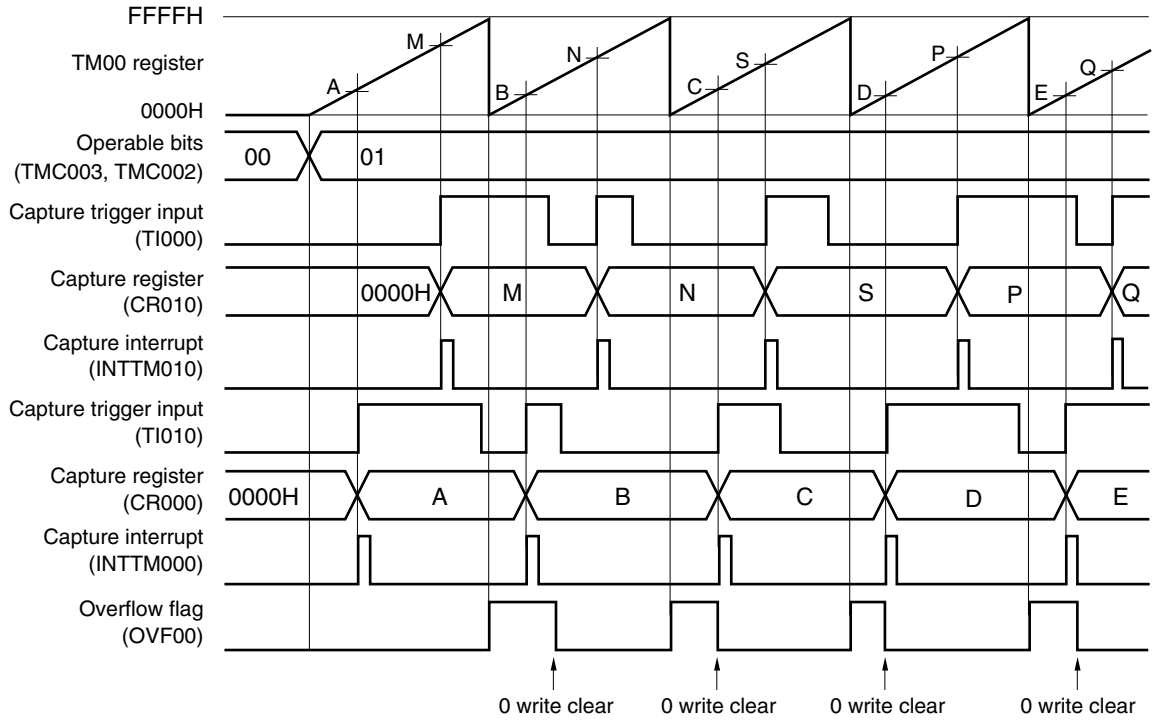
**Figure 6-37. Block Diagram of Free-Running Timer Mode**
**(CR000: Capture Register, CR010: Capture Register)**




**Remark** If both CR000 and CR010 are used as capture registers in the free-running timer mode, the TO00 output level is not inverted.
However, it can be inverted each time the valid edge of the TI000 pin is detected if bit 1 (TMC001) of 16-bit timer mode control register 00 (TMC00) is set to 1.

**Figure 6-38. Timing Example of Free-Running Timer Mode (CR000: Capture Register, CR010: Capture Register) (1/2)**

**(a) TOC00 = 13H, PRM00 = 50H, CRC00 = 05H, TMC00 = 04H**

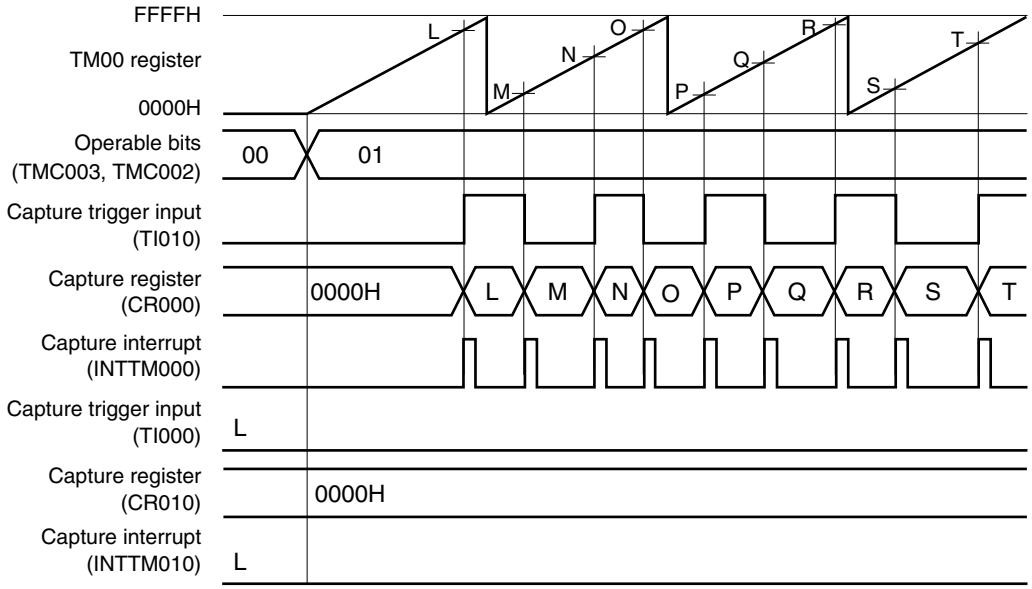



This is an application example where the count values that have been captured at the valid edges of separate capture trigger signals are stored in separate capture registers in the free-running timer mode.

The count value is captured to CR010 when the valid edge of the TI000 pin input is detected and to CR000 when the valid edge of the TI010 pin input is detected.

**Figure 6-38. Timing Example of Free-Running Timer Mode (CR000: Capture Register, CR010: Capture Register) (2/2)**

**(b) TOC00 = 13H, PRM00 = C0H, CRC00 = 05H, TMC00 = 04H**

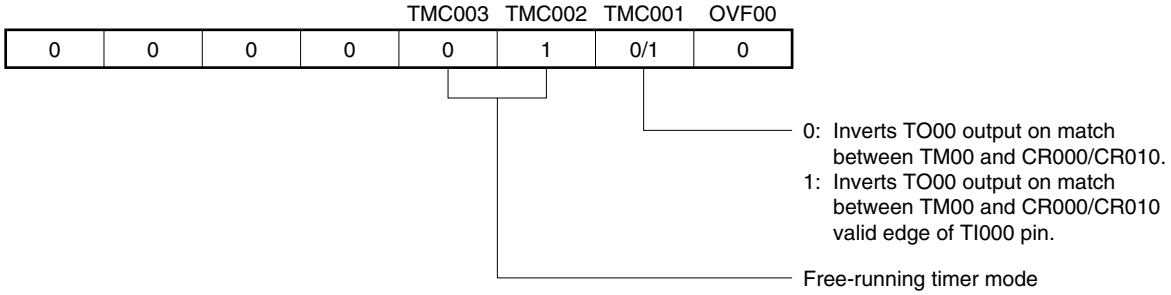



This is an application example where both the edges of the TI010 pin are detected and the count value is captured to CR000 in the free-running timer mode.

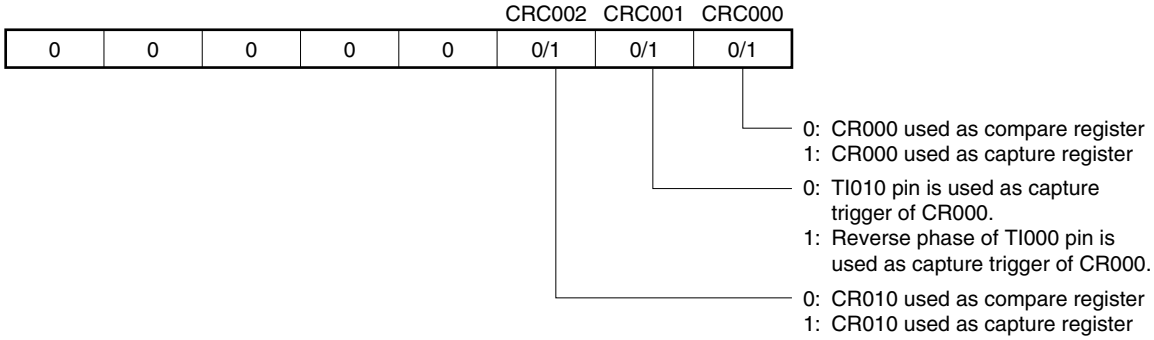
When both CR000 and CR010 are used as capture registers and when the valid edge of only the TI010 pin is to be detected, the count value cannot be captured to CR010.

**Figure 6-39. Example of Register Settings in Free-Running Timer Mode (1/2)**

**(a) 16-bit timer mode control register 00 (TMC00)**




**(b) Capture/compare control register 00 (CRC00)**




**(c) 16-bit timer output control register 00 (TOC00)**

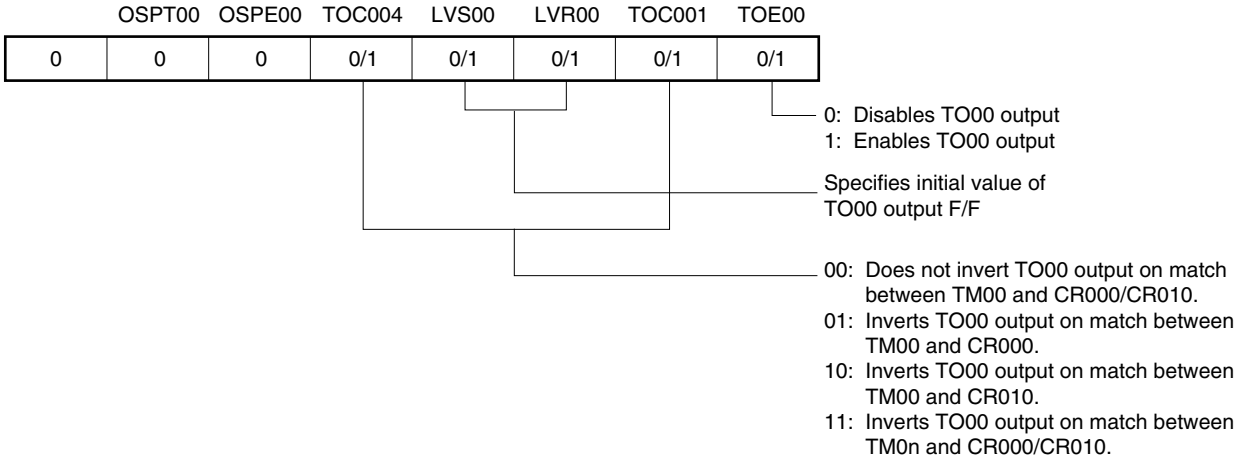
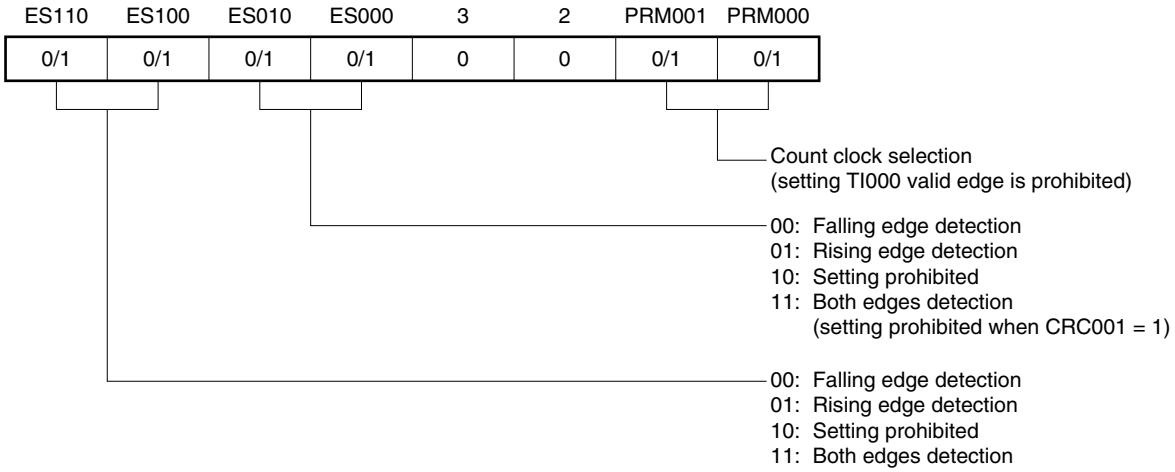


**Figure 6-39. Example of Register Settings in Free-Running Timer Mode (2/2)**

**(d) Prescaler mode register 00 (PRM00)**




**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

When this register is used as a compare register and when its value matches the count value of TM00, an interrupt signal (INTTM000) is generated. The count value of TM00 is not cleared.

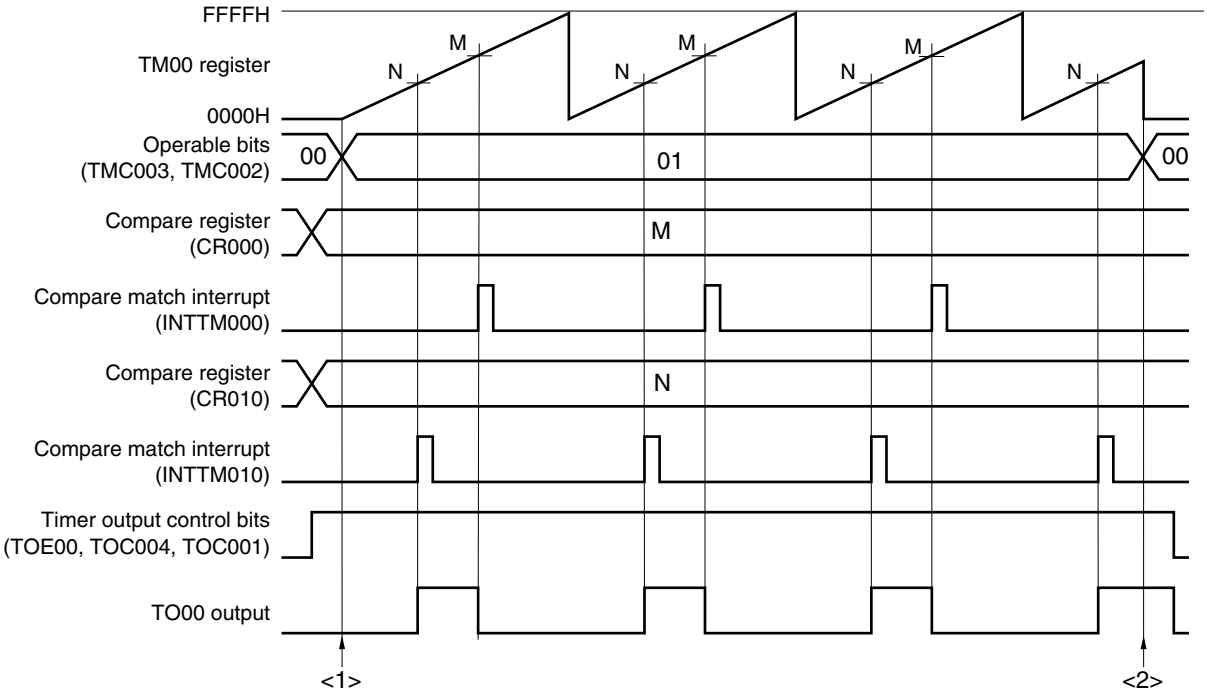
To use this register as a capture register, select either the TI000 or TI010 pin input as a capture trigger. When the valid edge of the capture trigger is detected, the count value of TM00 is stored in CR000.

**(g) 16-bit capture/compare register 010 (CR010)**

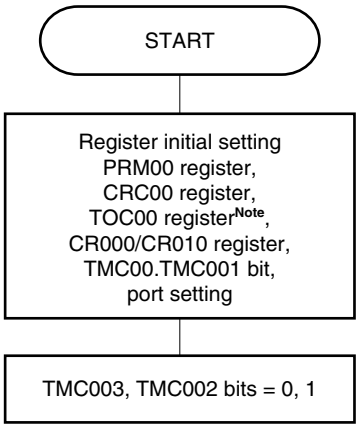
When this register is used as a compare register and when its value matches the count value of TM00, an interrupt signal (INTTM010) is generated. The count value of TM00 is not cleared.

When this register is used as a capture register, the TI000 pin input is used as a capture trigger. When the valid edge of the capture trigger is detected, the count value of TM00 is stored in CR010.

**Figure 6-40. Example of Software Processing in Free-Running Timer Mode**



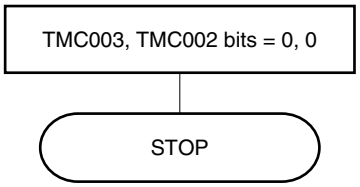

<1> Count operation start flow




Initial setting of these registers is performed before setting the TMC003 and TMC002 bits to 01.

Starts count operation

<2> Count operation stop flow




The counter is initialized and counting is stopped by clearing the TMC003 and TMC002 bits to 00.

**Note** Care must be exercised when setting TOC00. For details, refer to **6.3 (3) 16-bit timer output control register 00 (TOC00)**.

### 6.4.6 PPG output operation

A square wave having a pulse width set in advance by CR010 is output from the TO00 pin as a PPG (Programmable Pulse Generator) signal during a cycle set by CR000 when bits 3 and 2 (TMC003 and TMC002) of 16-bit timer mode control register 00 (TMC00) are set to 11 (clear & start upon a match between TM00 and CR000).

The pulse cycle and duty factor of the pulse generated as the PPG output are as follows.

- Pulse cycle = (Set value of CR000 + 1) × Count clock cycle
- Duty = (Set value of CR010 + 1) / (Set value of CR000 + 1)

**Caution To change the duty factor (value of CR010) during operation, refer to 6.5.1 Rewriting CR010 during TM00 operation.**

**Remarks 1.** For the setting of I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 6-41. Block Diagram of PPG Output Operation**

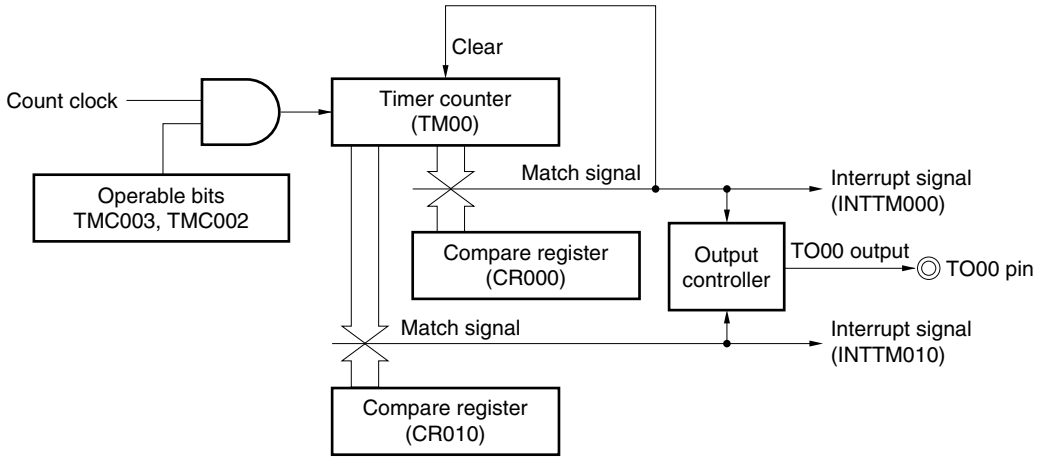
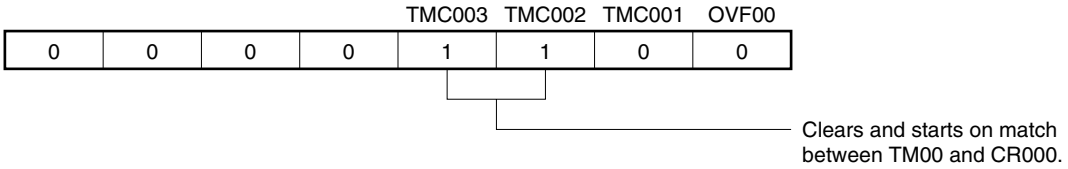
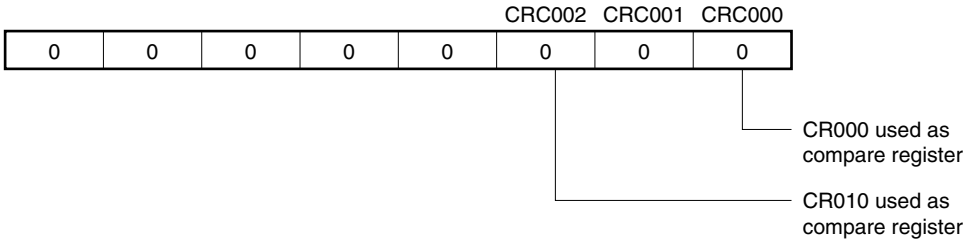


**Figure 6-42. Example of Register Settings for PPG Output Operation (1/2)**

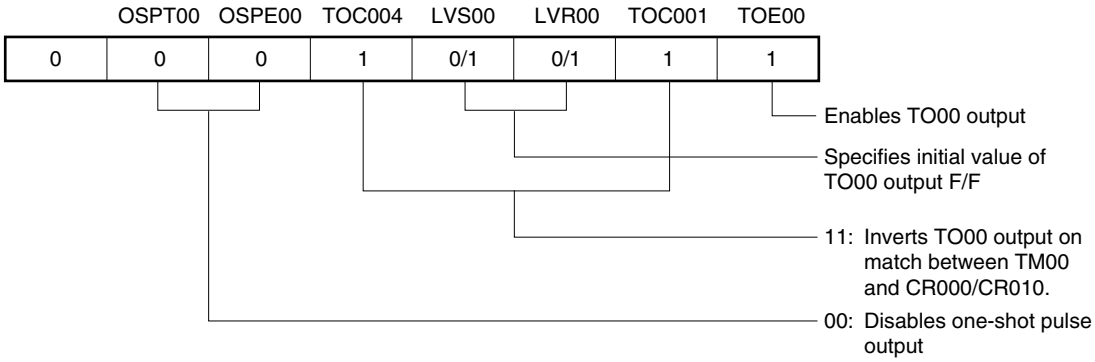
**(a) 16-bit timer mode control register 00 (TMC00)**




**(b) Capture/compare control register 00 (CRC00)**




**(c) 16-bit timer output control register 00 (TOC00)**




**(d) Prescaler mode register 00 (PRM00)**

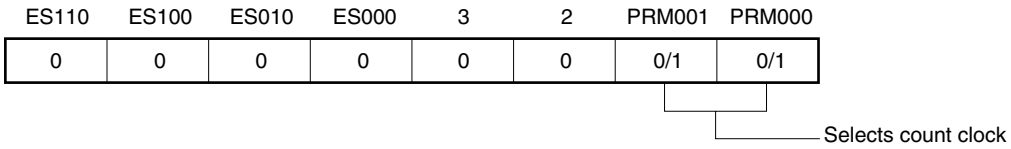


**Figure 6-42. Example of Register Settings for PPG Output Operation (2/2)**

**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

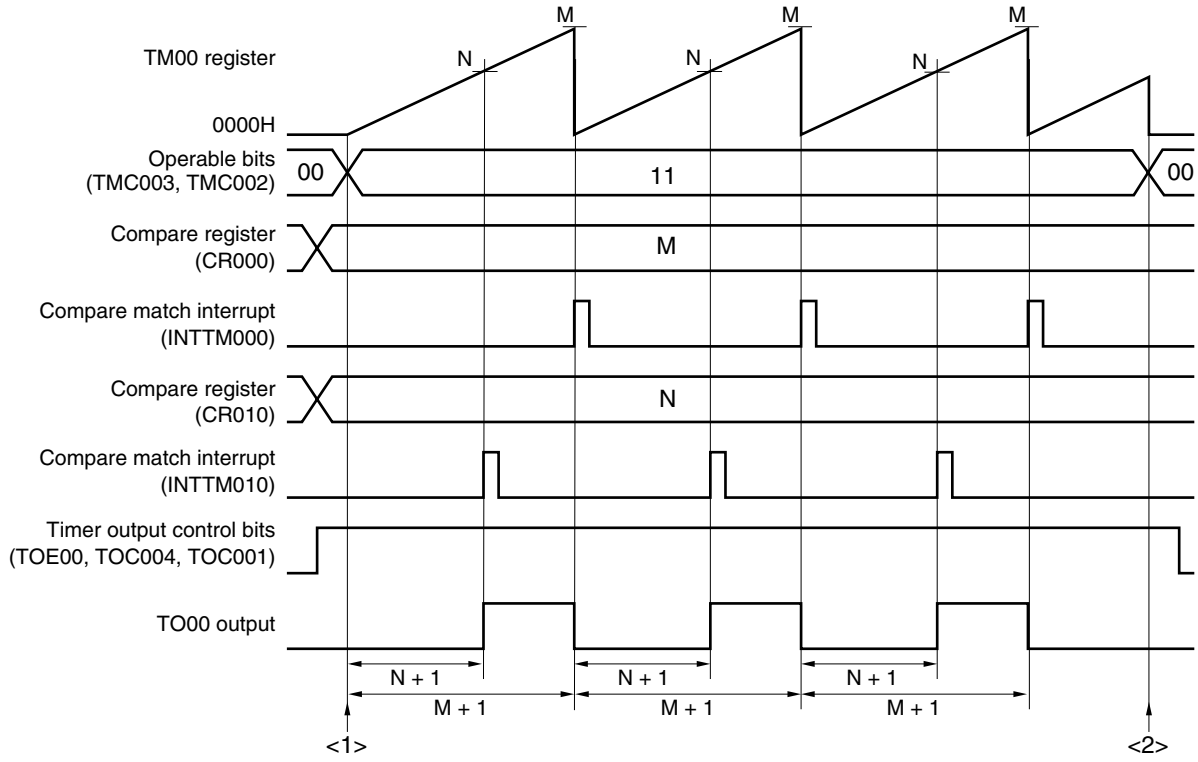
An interrupt signal (INTTM000) is generated when the value of this register matches the count value of TM00. The count value of TM00 is cleared.

**(g) 16-bit capture/compare register 010 (CR010)**

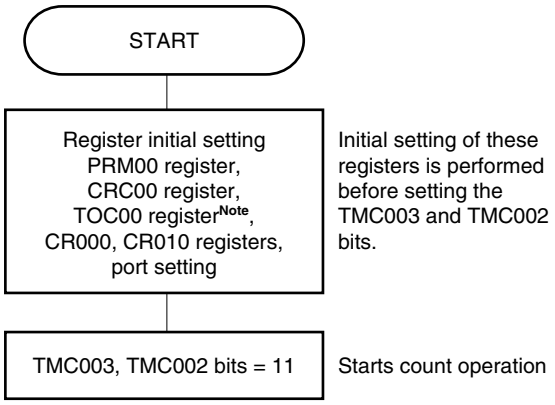
An interrupt signal (INTTM010) is generated when the value of this register matches the count value of TM00. The count value of TM00 is not cleared.

**Caution Set values to CR000 and CR010 such that the condition 0000H ≤ CR010 < CR000 ≤ FFFFH is satisfied.**

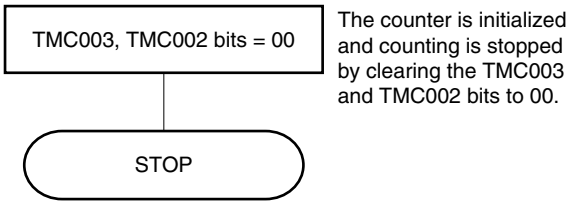
**Figure 6-43. Example of Software Processing for PPG Output Operation**




<1> Count operation start flow




<2> Count operation stop flow




**Note** Care must be exercised when setting TOC00. For details, refer to **6.3 (3) 16-bit timer output control register 00 (TOC00)**.

**Remark** PPG pulse cycle = (M + 1) × Count clock cycle
PPG duty = (N + 1)/(M + 1)

### 6.4.7 One-shot pulse output operation

A one-shot pulse can be output by setting bits 3 and 2 (TMC003 and TMC002) of the 16-bit timer mode control register 00 (TMC00) to 01 (free-running timer mode) or to 10 (clear & start mode entered by the TI000 pin valid edge) and setting bit 5 (OSPE00) of 16-bit timer output control register 00 (TOC00) to 1.

When bit 6 (OSPT00) of TOC00 is set to 1 or when the valid edge is input to the TI000 pin during timer operation, clearing & starting of TM00 is triggered, and a pulse of the difference between the values of CR000 and CR010 is output only once from the TO00 pin.

**Cautions 1. Do not input the trigger again (setting OSPT00 to 1 or detecting the valid edge of the TI000 pin) while the one-shot pulse is output. To output the one-shot pulse again, generate the trigger after the current one-shot pulse output has completed.**

**2. To use only the setting of OSPT00 to 1 as the trigger of one-shot pulse output, do not change the level of the TI000 pin or its alternate function port pin. Otherwise, the pulse will be unexpectedly output.**

**Remarks 1.** For the setting of the I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 6-44. Block Diagram of One-Shot Pulse Output Operation**

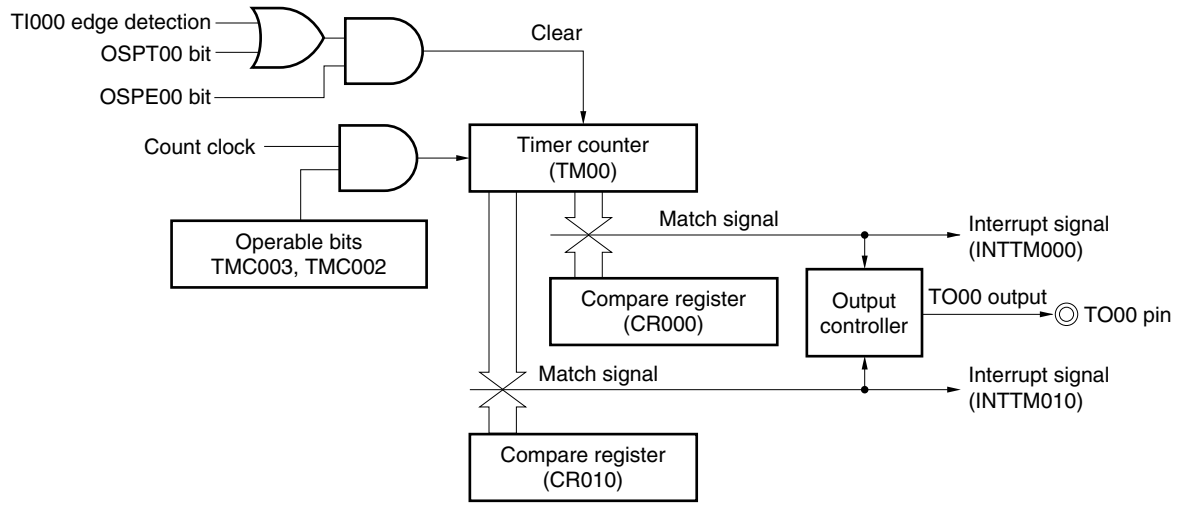
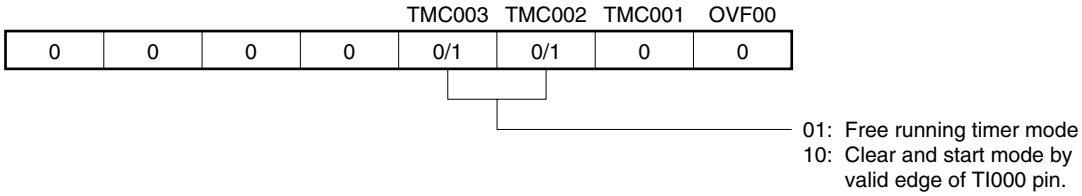
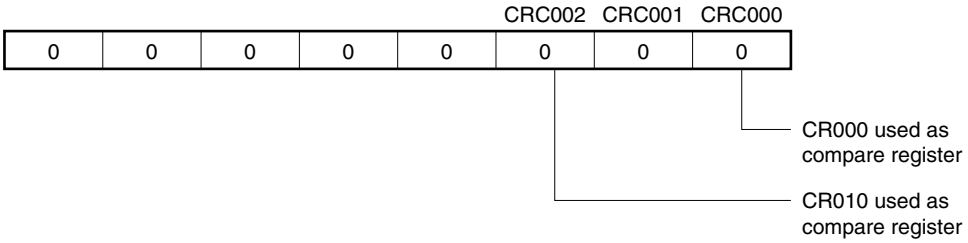


**Figure 6-45. Example of Register Settings for One-Shot Pulse Output Operation (1/2)**

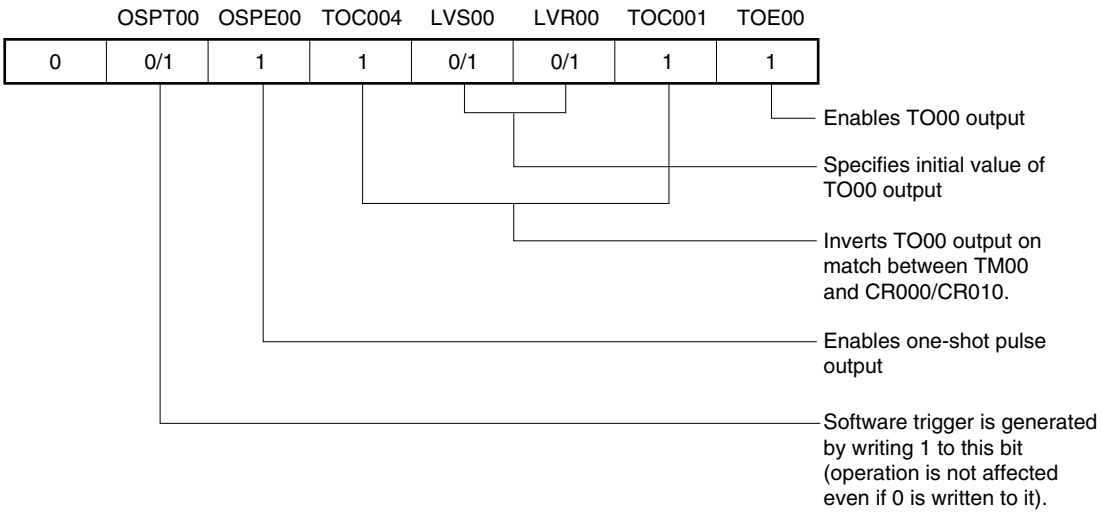
**(a) 16-bit timer mode control register 00 (TMC00)**




**(b) Capture/compare control register 00 (CRC00)**




**(c) 16-bit timer output control register 00 (TOC00)**




**(d) Prescaler mode register 00 (PRM00)**

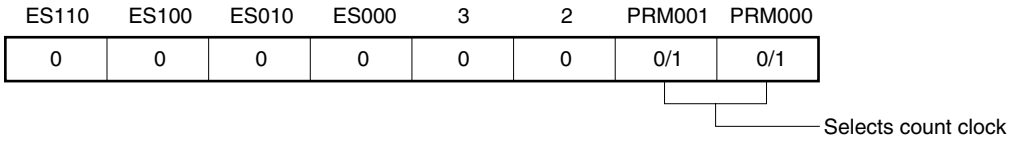


**Figure 6-45. Example of Register Settings for One-Shot Pulse Output Operation (2/2)**

**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

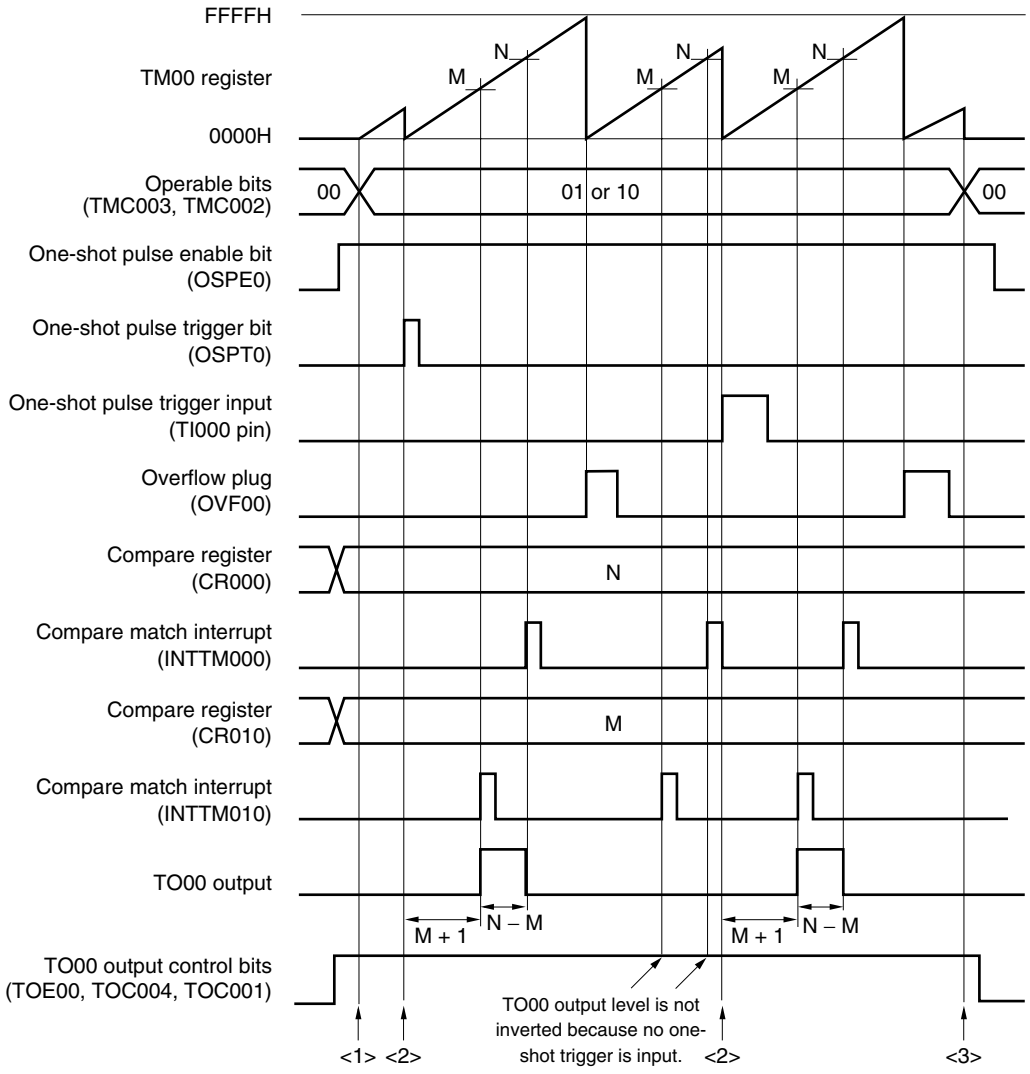
This register is used as a compare register when a one-shot pulse is output. When the value of TM00 matches that of CR000, an interrupt signal (INTTM000) is generated and the TO00 output level is inverted.

**(g) 16-bit capture/compare register 010 (CR010)**

This register is used as a compare register when a one-shot pulse is output. When the value of TM00 matches that of CR010, an interrupt signal (INTTM010) is generated and the TO00 output level is inverted.

**Caution Do not set the same value to CR000 and CR010.**

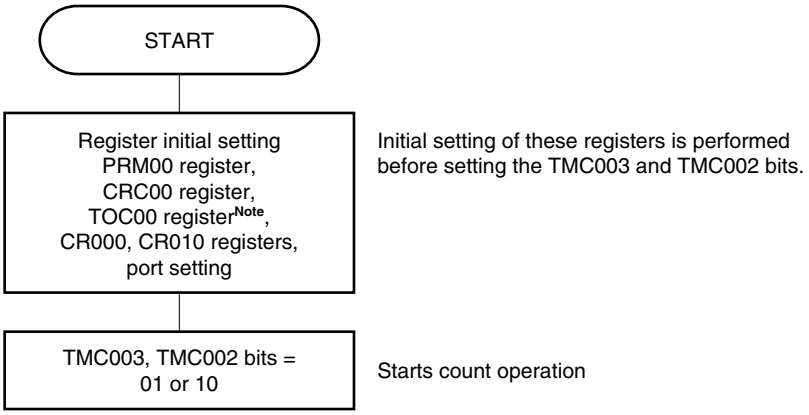
**Figure 6-46. Example of Software Processing for One-Shot Pulse Output Operation (1/2)**



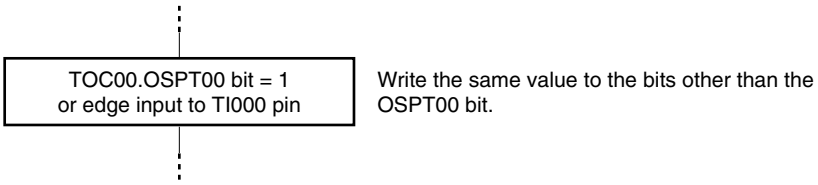

- Time from when the one-shot pulse trigger is input until the one-shot pulse is output
  = (M + 1) × Count clock cycle
- One-shot pulse output active level width
  = (N − M) × Count clock cycle

**Figure 6-46. Example of Software Processing for One-Shot Pulse Output Operation (2/2)**

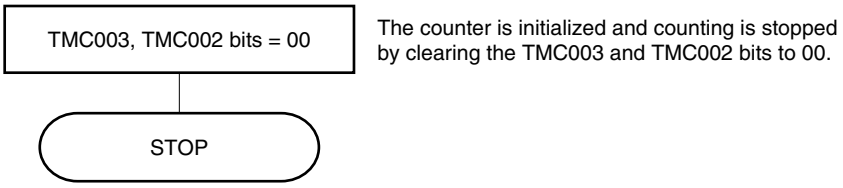
<1> Count operation start flow




<2> One-shot trigger input flow




<3> Count operation stop flow




**Note** Care must be exercised when setting TOC00. For details, refer to **6.3 (3) 16-bit timer output control register 00 (TOC00)**.

### 6.4.8 Pulse width measurement operation

TM00 can be used to measure the pulse width of the signal input to the TI000 and TI010 pins.

Measurement can be accomplished by operating the 16-bit timer/event counter 00 in the free-running timer mode or by restarting the timer in synchronization with the signal input to the TI000 pin.

When an interrupt is generated, read the value of the valid capture register and measure the pulse width. Check bit 0 (OVF00) of 16-bit timer mode control register 00 (TMC00). If it is set (to 1), clear it to 0 by software.

**Figure 6-47. Block Diagram of Pulse Width Measurement (Free-Running Timer Mode)**

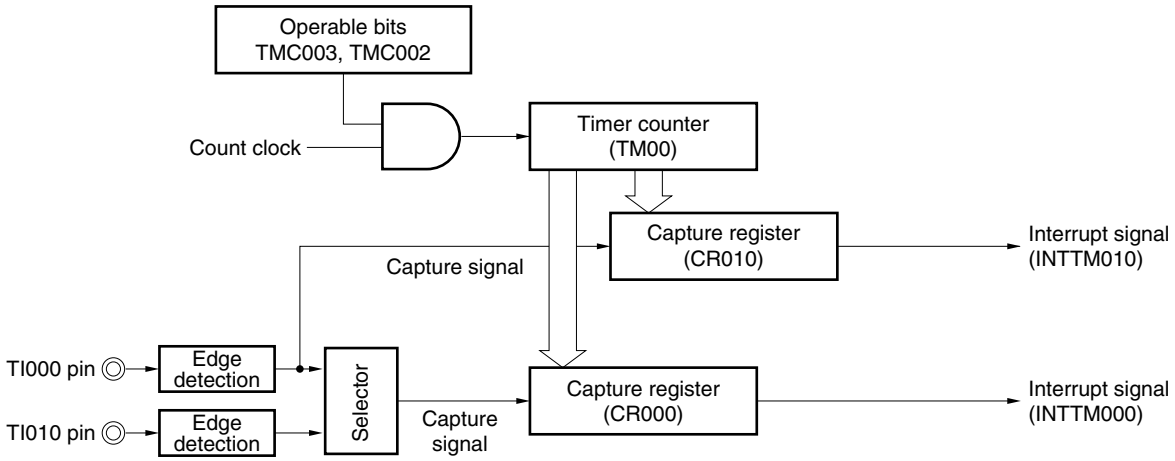
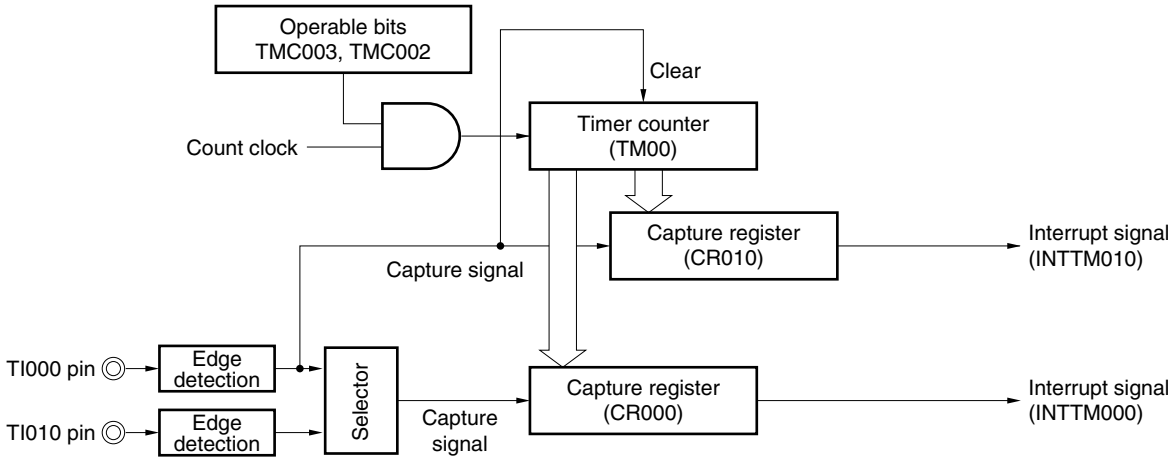



**Figure 6-48. Block Diagram of Pulse Width Measurement**
**(Clear & Start Mode Entered by TI000 Pin Valid Edge Input)**



A pulse width can be measured in the following three ways.

- Measuring the pulse width by using two input signals of the TI000 and TI010 pins (free-running timer mode)
- Measuring the pulse width by using one input signal of the TI000 pin (free-running timer mode)
- Measuring the pulse width by using one input signal of the TI000 pin (clear & start mode entered by the TI000 pin valid edge input)

**Remarks 1.** For the setting of the I/O pins, refer to **6.3 (6) Port mode register 0 (PM0)**.

**2.** For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**(1) Measuring the pulse width by using two input signals of the TI000 and TI010 pins (free-running timer mode)**

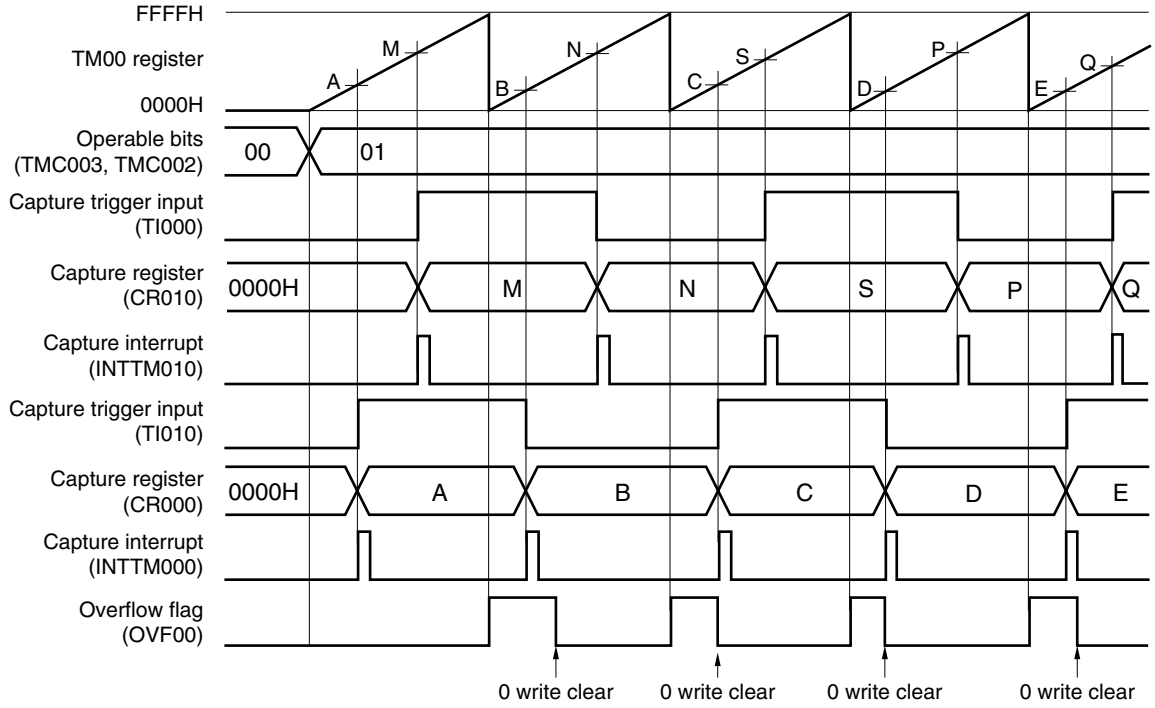
Set the free-running timer mode (TMC003 and TMC002 = 01). When the valid edge of the TI000 pin is detected, the count value of TM00 is captured to CR010. When the valid edge of the TI010 pin is detected, the count value of TM00 is captured to CR000. Specify detection of both the edges of the TI000 and TI010 pins.

By this measurement method, the previous count value is subtracted from the count value captured by the edge of each input signal. Therefore, save the previously captured value to a separate register in advance.

If an overflow occurs, the value becomes negative if the previously captured value is simply subtracted from the current captured value and, therefore, a borrow occurs (bit 0 (CY) of the program status word (PSW) is set to 1). If this happens, ignore CY and take the calculated value as the pulse width. In addition, clear bit 0 (OVF00) of 16-bit timer mode control register 00 (TMC00) to 0.

**Figure 6-49. Timing Example of Pulse Width Measurement (1)**

**• TMC00 = 04H, PRM00 = F0H, CRC00 = 05H**



**(2) Measuring the pulse width by using one input signal of the TI000 pin (free-running timer mode)**

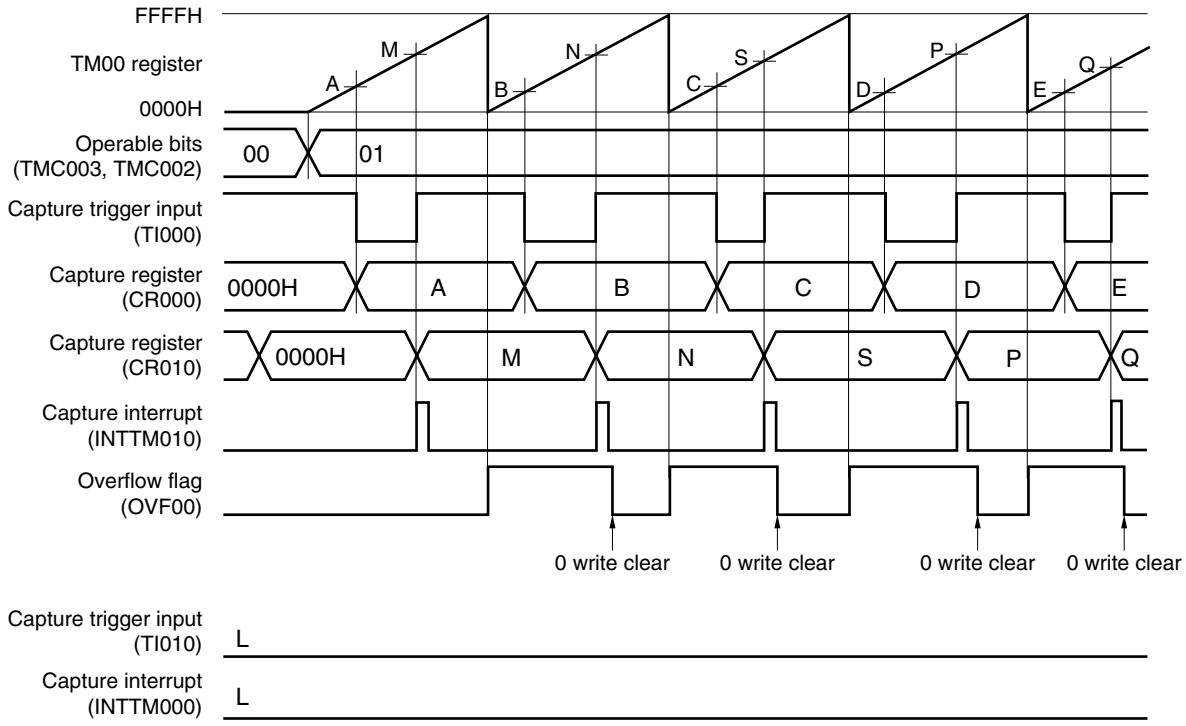
Set the free-running timer mode (TMC003 and TMC002 = 01). The count value of TM00 is captured to CR000 in the phase reverse to the valid edge detected on the TI000 pin. When the valid edge of the TI000 pin is detected, the count value of TM00 is captured to CR010.

By this measurement method, values are stored in separate capture registers when a width from one edge to another is measured. Therefore, the capture values do not have to be saved. By subtracting the value of one capture register from that of another, a high-level width, low-level width, and cycle are calculated.

If an overflow occurs, the value becomes negative if one captured value is simply subtracted from another and, therefore, a borrow occurs (bit 0 (CY) of the program status word (PSW) is set to 1). If this happens, ignore CY and take the calculated value as the pulse width. In addition, clear bit 0 (OVF00) of 16-bit timer mode control register 00 (TMC00) to 0.

**Figure 6-50. Timing Example of Pulse Width Measurement (2)**

**• TMC00 = 04H, PRM00 = 10H, CRC00 = 07H**



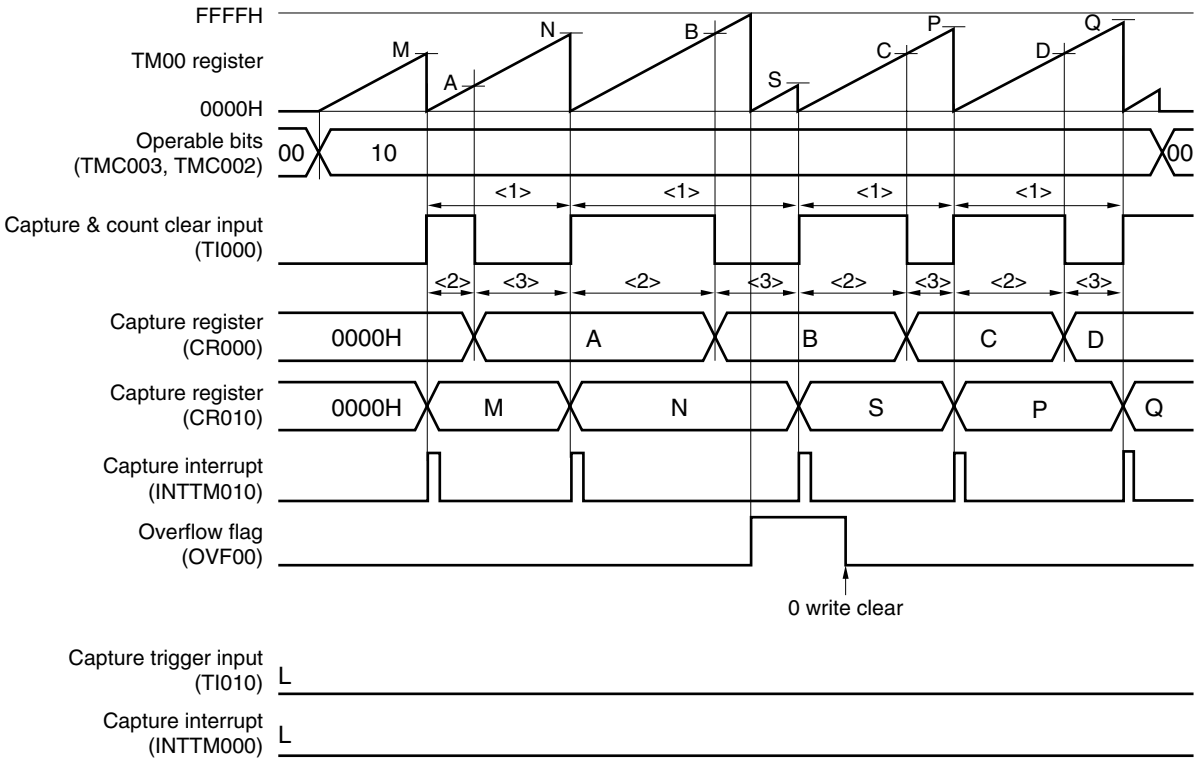
**(3) Measuring the pulse width by using one input signal of the TI000 pin (clear & start mode entered by the TI000 pin valid edge input)**

Set the clear & start mode entered by the TI000 pin valid edge (TMC003 and TMC002 = 10). The count value of TM00 is captured to CR000 in the phase reverse to the valid edge of the TI000 pin, and the count value of TM00 is captured to CR010 and TM00 is cleared (0000H) when the valid edge of the TI000 pin is detected. Therefore, a cycle is stored in CR010 if TM00 does not overflow.

If an overflow occurs, take the value that results from adding 10000H to the value stored in CR010 as a cycle. Clear bit 0 (OVF00) of 16-bit timer mode control register 00 (TMC00) to 0.

**Figure 6-51. Timing Example of Pulse Width Measurement (3)**

**• TMC00 = 08H, PRM00 = 10H, CRC00 = 07H**




<1> Pulse cycle = (10000H × Number of times OVF00 bit is set to 1 + Captured value of CR010) × Count clock cycle

<2> High-level pulse width = (10000H × Number of times OVF00 bit is set to 1 + Captured value of CR000) × Count clock cycle

<3> Low-level pulse width = (Pulse cycle − High-level pulse width)

**Figure 6-52. Example of Register Settings for Pulse Width Measurement (1/2)**

**(a) 16-bit timer mode control register 00 (TMC00)**

| | | | | TMC003 | TMC002 | TMC001 | OVF00 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0/1 | 0/1 | 0 | 0 |

01: Free running timer mode
10: Clear and start mode entered by valid edge of TI000 pin.

**(b) Capture/compare control register 00 (CRC00)**

| | | | | | CRC002 | CRC001 | CRC000 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0/1 | 1 |

1: CR000 used as capture register

0: TI010 pin is used as capture trigger of CR000.
1: Reverse phase of TI000 pin is used as capture trigger of CR000.

1: CR010 used as capture register

**(c) 16-bit timer output control register 00 (TOC00)**

| | OSPT00 | OSPE00 | TOC004 | LVS00 | LVR00 | TOC001 | TOE00 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**(d) Prescaler mode register 00 (PRM00)**

| ES110 | ES100 | ES010 | ES000 | 3 | 2 | PRM001 | PRM000 |
|---|---|---|---|---|---|---|---|
| 0/1 | 0/1 | 0/1 | 0/1 | 0 | 0 | 0/1 | 0/1 |

Selects count clock
(setting valid edge of TI000 is prohibited)

00: Falling edge detection
01: Rising edge detection
10: Setting prohibited
11: Both edges detection
(setting when CRC001 = 1 is prohibited)

00: Falling edge detection
01: Rising edge detection
10: Setting prohibited
11: Both edges detection

**Figure 6-52. Example of Register Settings for Pulse Width Measurement (2/2)**

**(e) 16-bit timer counter 00 (TM00)**

By reading TM00, the count value can be read.

**(f) 16-bit capture/compare register 000 (CR000)**

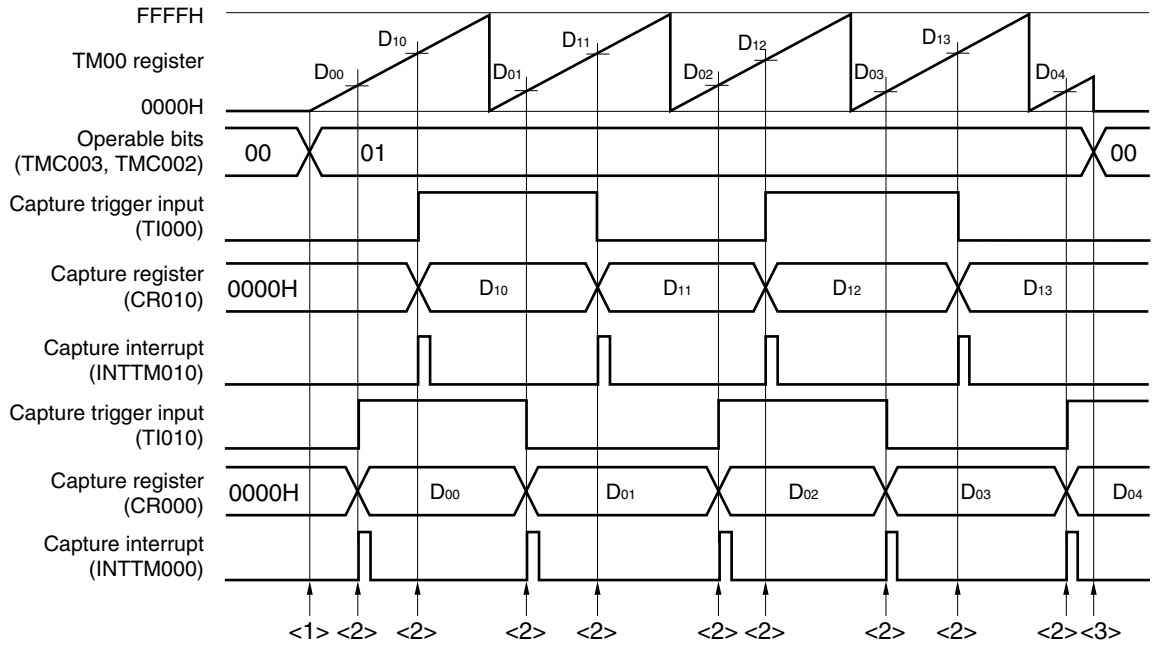
This register is used as a capture register. Either the TI000 or TI010 pin is selected as a capture trigger. When a specified edge of the capture trigger is detected, the count value of TM00 is stored in CR000.

**(g) 16-bit capture/compare register 010 (CR010)**

This register is used as a capture register. The signal input to the TI000 pin is used as a capture trigger. When the capture trigger is detected, the count value of TM00 is stored in CR010.

**Figure 6-53. Example of Software Processing for Pulse Width Measurement (1/2)**

**(a) Example of free-running timer mode**




**(b) Example of clear & start mode entered by TI000 pin valid edge**

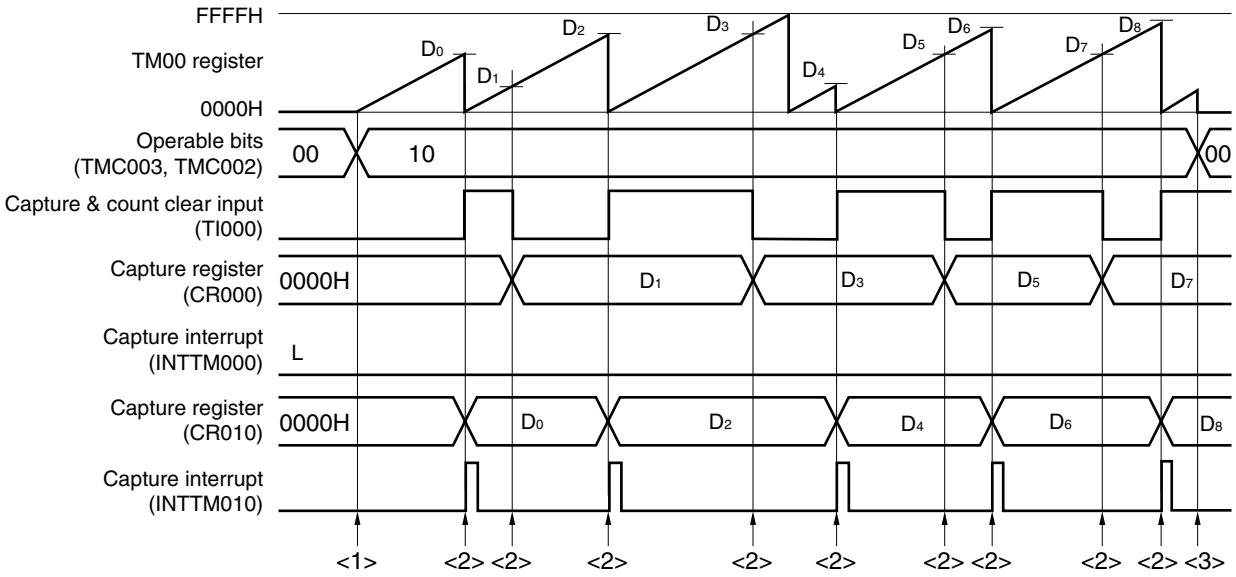
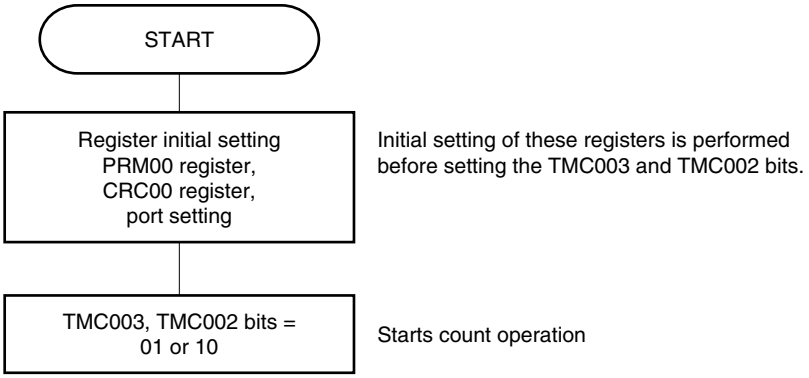
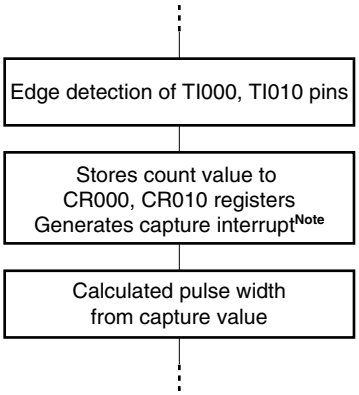


**Figure 6-53. Example of Software Processing for Pulse Width Measurement (2/2)**

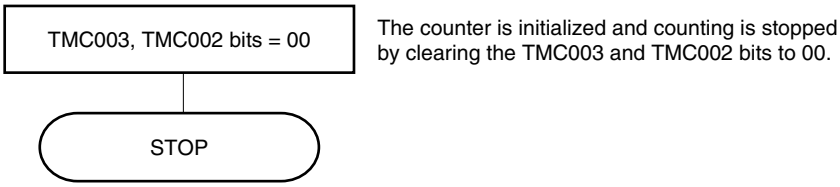
<1> Count operation start flow




<2> Capture trigger input flow




<3> Count operation stop flow




**Note** The capture interrupt signal (INTTM000) is not generated when the reverse-phase edge of the TI000 pin input is selected to the valid edge of CR000.

## 6.5 Special Use of TM00

### 6.5.1 Rewriting CR010 during TM00 operation

In principle, rewriting CR000 and CR010 of the 78K0/Kx2-L microcontrollers when they are used as compare registers is prohibited while TM00 is operating (TMC003 and TMC002 = other than 00).

However, the value of CR010 can be changed, even while TM00 is operating, using the following procedure if CR010 is used for PPG output and the duty factor is changed. (When changing the value of CR010 to a smaller value than the current one, rewrite it immediately after its value matches the value of TM00. When changing the value of CR010 to a larger value than the current one, rewrite it immediately after the values of CR000 and TM00 match. If the value of CR010 is rewritten immediately before a match between CR010 and TM00, or between CR000 and TM00, an unexpected operation may be performed.).

Procedure for changing value of CR010

<1> Disable interrupt INTTM010 (TMMK010 = 1).
<2> Disable reversal of the timer output when the value of TM00 matches that of CR010 (TOC004 = 0).
<3> Change the value of CR010.
<4> Wait for one cycle of the count clock of TM00.
<5> Enable reversal of the timer output when the value of TM00 matches that of CR010 (TOC004 = 1).
<6> Clear the interrupt flag of INTTM010 (TMIF010 = 0) to 0.
<7> Enable interrupt INTTM010 (TMMK010 = 0).

**Remark** For TMIF010 and TMMK010, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

### 6.5.2 Setting LVS00 and LVR00

#### (1) Usage of LVS00 and LVR00

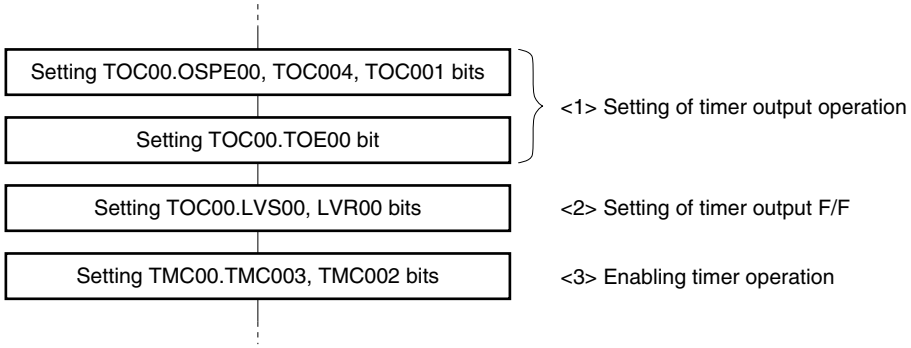
LVS00 and LVR00 are used to set the default value of the TO00 output and to invert the timer output without enabling the timer operation (TMC003 and TMC002 = 00). Clear LVS00 and LVR00 to 00 (default value: low-level output) when software control is unnecessary.

| LVS00 | LVR00 | Timer Output Status |
|---|---|---|
| 0 | 0 | Not changed (low-level output) |
| 0 | 1 | Cleared (low-level output) |
| 1 | 0 | Set (high-level output) |
| 1 | 1 | Setting prohibited |

**(2) Setting LVS00 and LVR00**

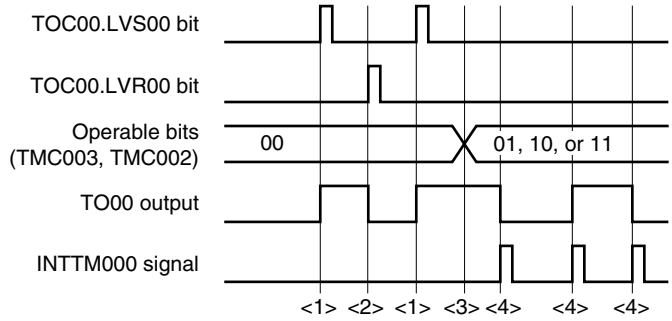
Set LVS00 and LVR00 using the following procedure.

**Figure 6-54. Example of Flow for Setting LVS00 and LVR00 Bits**




**Caution Be sure to set LVS00 and LVR00 following steps <1>, <2>, and <3> above.**
**Step <2> can be performed after <1> and before <3>.**

**Figure 6-55. Timing Example of LVR00 and LVS00**




<1> The TO00 output goes high when LVS00 and LVR00 = 10.

<2> The TO00 output goes low when LVS00 and LVR00 = 01 (the pin output remains unchanged from the high level even if LVS00 and LVR00 are cleared to 00).

<3> The timer starts operating when TMC003 and TMC002 are set to 01, 10, or 11. Because LVS00 and LVR00 were set to 10 before the operation was started, the TO00 output starts from the high level. After the timer starts operating, setting LVS00 and LVR00 is prohibited until TMC003 and TMC002 = 00 (disabling the timer operation).

<4> The TO00 output level is inverted each time an interrupt signal (INTTM000) is generated.

## 6.6 Cautions for 16-Bit Timer/Event Counter 00

### (1) Restrictions for each channel of 16-bit timer/event counter 00

Table 6-3 shows the restrictions for each channel.

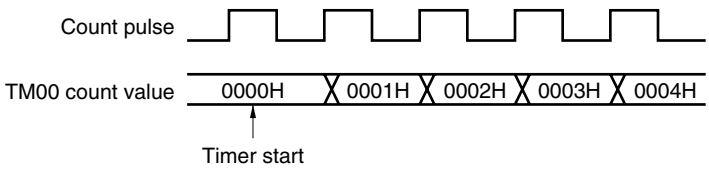
**Table 6-3. Restrictions for Each Channel of 16-Bit Timer/Event Counter 00**

| Operation | Restriction |
|---|---|
| As interval timer | – |
| As square-wave output | |
| As external event counter | |
| As clear & start mode entered by TI000 pin valid edge input | Using timer output (TO00) is prohibited when detection of the valid edge of the TI010 pin is used. (TOC00 = 00H) |
| As free-running timer | – |
| As PPG output | 0000H ≤ CP010 < CR000 ≤ FFFFH |
| As one-shot pulse output | Setting the same value to CR000 and CP010 is prohibited. |
| As pulse width measurement | Using timer output (TO00) is prohibited (TOC00 = 00H) |

### (2) Timer start errors

An error of up to one clock may occur in the time required for a match signal to be generated after timer start. This is because counting TM00 is started asynchronously to the count pulse.

**Figure 6-56. Start Timing of TM00 Count**




### (3) Setting of CR000 and CR010 (clear & start mode entered upon a match between TM00 and CR000)

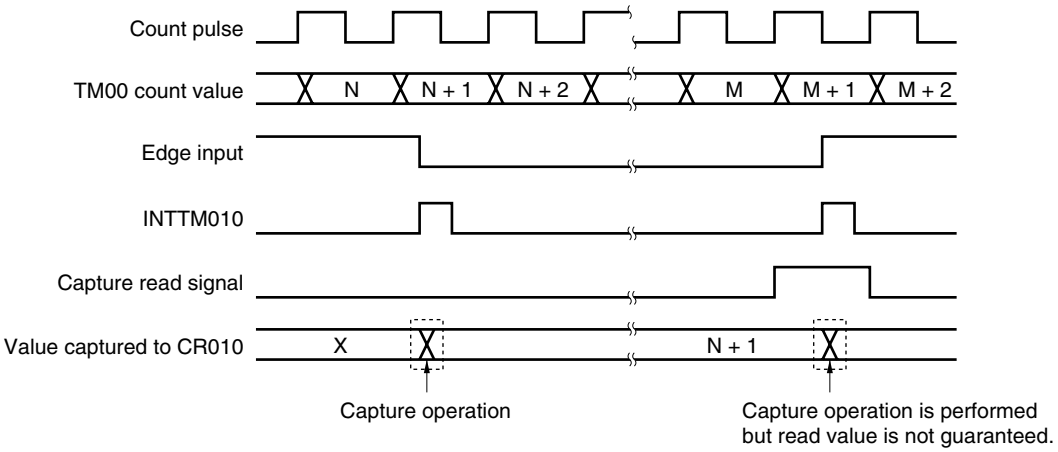
Set a value other than 0000H to CR000 and CR010 (TM00 cannot count one pulse when it is used as an external event counter).

**(4) Timing of holding data by capture register**

(a) When the valid edge is input to the TI000/TI010 pin and the reverse phase of the TI000 pin is detected while CR000/CR010 is read, CR010 performs a capture operation but the read value of CR000/CR010 is not guaranteed. At this time, an interrupt signal (INTTM000/INTTM010) is generated when the valid edge of the TI000/TI010 pin is detected (the interrupt signal is not generated when the reverse-phase edge of the TI000 pin is detected).

When the count value is captured because the valid edge of the TI000/TI010 pin was detected, read the value of CR000/CR010 after INTTM000/INTTM010 is generated.

**Figure 6-57. Timing of Holding Data by Capture Register**




(b) The values of CR000 and CR010 are not guaranteed after 16-bit timer/event counter 00 stops.

**(5) Setting valid edge**

Set the valid edge of the TI000 pin while the timer operation is stopped (TMC003 and TMC002 = 00). Set the valid edge by using ES000 and ES010.

**(6) Re-triggering one-shot pulse**

Make sure that the trigger is not generated while an active level is being output in the one-shot pulse output mode. Be sure to input the next trigger after the current active level is output.

### (7) Operation of OVF00 flag

#### (a) Setting OVF00 flag (1)

The OVF00 flag is set to 1 in the following case, as well as when TM00 overflows.

Select the clear & start mode entered upon a match between TM00 and CR000.

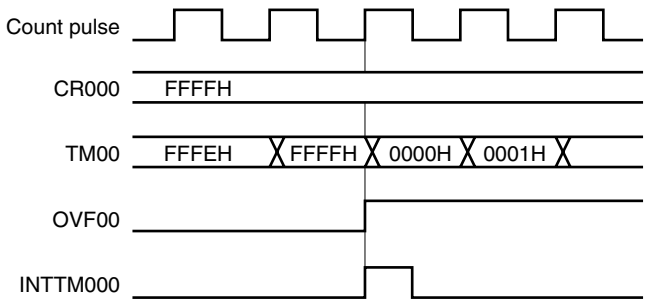
↓

Set CR000 to FFFFH.

↓

When TM00 matches CR000 and TM00 is cleared from FFFFH to 0000H

**Figure 6-58. Operation Timing of OVF00 Flag**




#### (b) Clearing OVF00 flag

Even if the OVF00 flag is cleared to 0 after TM00 overflows and before the next count clock is counted (before the value of TM00 becomes 0001H), it is set to 1 again and clearing is invalid.

### (8) One-shot pulse output

One-shot pulse output operates correctly in the free-running timer mode or the clear & start mode entered by the TI000 pin valid edge. The one-shot pulse cannot be output in the clear & start mode entered upon a match between TM00 and CR000.

**(9) Capture operation**

**(a) When valid edge of TI000 is specified as count clock**

When the valid edge of TI000 is specified as the count clock, the capture register for which TI000 is specified as a trigger does not operate correctly.

**(b) Pulse width to accurately capture value by signals input to TI010 and TI000 pins**

To accurately capture the count value, the pulse input to the TI000 and TI010 pins as a capture trigger must be wider than two count clocks selected by PRM00 (refer to **Figure 6-7**).

**(c) Generation of interrupt signal**

The capture operation is performed at the falling edge of the count clock but the interrupt signals (INTTM000 and INTTM010) are generated at the rising edge of the next count clock (refer to **Figure 6-7**).

**(d) Note when CRC001 (bit 1 of capture/compare control register 00 (CRC00)) is set to 1**

When the count value of the TM00 register is captured to the CR000 register in the phase reverse to the signal input to the TI000 pin, the interrupt signal (INTTM000) is not generated after the count value is captured. If the valid edge is detected on the TI010 pin during this operation, the capture operation is not performed but the INTTM000 signal is generated as an external interrupt signal. Mask the INTTM000 signal when the external interrupt is not used.

**(10) Edge detection**

**(a) Specifying valid edge after reset**

If the operation of the 16-bit timer/event counter 00 is enabled after reset and while the TI000 or TI010 pin is at high level and when the rising edge or both the edges are specified as the valid edge of the TI000 or TI010 pin, then the high level of the TI000 or TI010 pin is detected as the rising edge. Note this when the TI000 or TI010 pin is pulled up. However, the rising edge is not detected when the operation is once stopped and then enabled again.

**(b) Sampling clock for eliminating noise**

The sampling clock for eliminating noise differs depending on whether the valid edge of TI000 is used as the count clock or capture trigger. In the former case, the sampling clock is fixed to $f_{PRS}$. In the latter, the count clock selected by PRM00 is used for sampling.

When the signal input to the TI000 pin is sampled and the valid level is detected two times in a row, the valid edge is detected. Therefore, noise having a short pulse width can be eliminated (refer to **Figure 6-7**).

**(11) Timer operation**

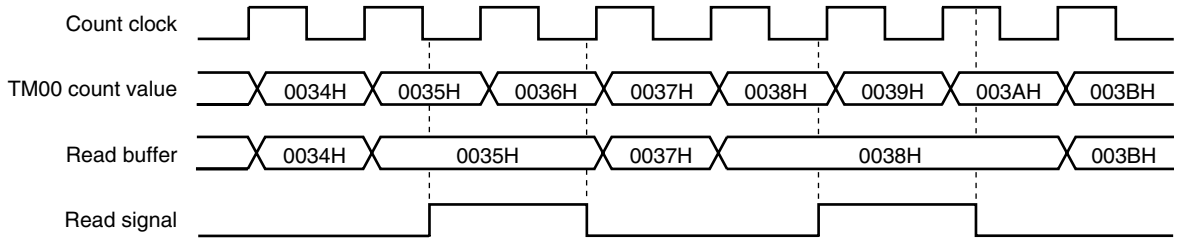
The signal input to the TI000/TI010 pin is not acknowledged while the timer is stopped, regardless of the operation mode of the CPU.

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**(12) Reading of 16-bit timer counter 00 (TM00)**

TM00 can be read without stopping the actual counter, because the count values captured to the buffer are fixed when it is read. The buffer, however, may not be updated when it is read immediately before the counter counts up, because the buffer is updated at the timing the counter counts up.

**Figure 6-59. 16-bit Timer Counter 00 (TM00) Read Timing**



# CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50 AND 51

<R>

| Item | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | | | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|---|---|---|
| | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40, 44, 48 Pins |
| 8-bit timer/event counter 50 | – | – | – | – | √ (PWM output: 1) | |
| 8-bit timer/event counter 51 | √ (No output) | √ (No output) | √ (No output Note) | √ (No I/O) | √ (PWM output: 1) | |

**Remark** √: Mounted, –: Not mounted

**Note** When bits 3 and 2 (TM5SEL1, TM5SEL0) of MUXSEL register = 0, 0: No I/O
When bits 3 and 2 (TM5SEL1, TM5SEL0) of MUXSEL register = 1, 0 or 0, 1: No output (but there is input)

## 7.1 Functions of 8-Bit Timer/Event Counters 50 and 51

8-bit timer/event counters 50 and 51 have the following functions.

<R>
- 78K0/KY2-L, 78K0/KA2-L (20-pin and 25-pin products): 8-bit timer/event counter 51
  (1) Interval timer
  (2) External event counter

<R>
- 78K0/KA2-L (32-pin products): 8-bit timer 51
  (1) Interval timer

- 78K0/KB2-L, 78K0/KC2-L: 8-bit timer/event counters 50 and 51
  (1) Interval timer
  (2) External event counter
  (3) Square-wave output
  (4) PWM output

## 7.2 Configuration of 8-Bit Timer/Event Counters 50 and 51

8-bit timer/event counters 50 and 51 include the following hardware.

**Table 7-1. Configuration of 8-Bit Timer/Event Counters 50 and 51**

(1) 78K0/KY2-L, 78K0/KA2-L (20-pin products)

| Item | Configuration |
|---|---|
| Timer register | 8-bit timer counter 51 (TM51) |
| Timer input | TI51 |
| Register | 8-bit timer compare register 51 (CR51) |
| Control registers | Timer clock selection register 51 (TCL51)<br>8-bit timer mode control register 51 (TMC51)<br>Port mode register 3 (PM3)<br>Port register 3 (P3) |

<R> (2) 78K0/KA2-L (25-pin and 32-pin products)

| Item | Configuration |
|---|---|
| Timer register | 8-bit timer counter 51 (TM51) |
| Timer input | TI51 Note |
| Register | 8-bit timer compare register 51 (CR51) |
| Control registers | Timer clock selection register 51 (TCL51)<br>8-bit timer mode control register 51 (TMC51)<br>Port alternate switch control register (MUXSEL) Note<br>Port mode register 0 (PM0) or port mode register 3 (PM3) Note<br>Port register 0 (P0) or port register 3 (P3) Note |

**Note** 78K0/KA2-L (25-pin products) only

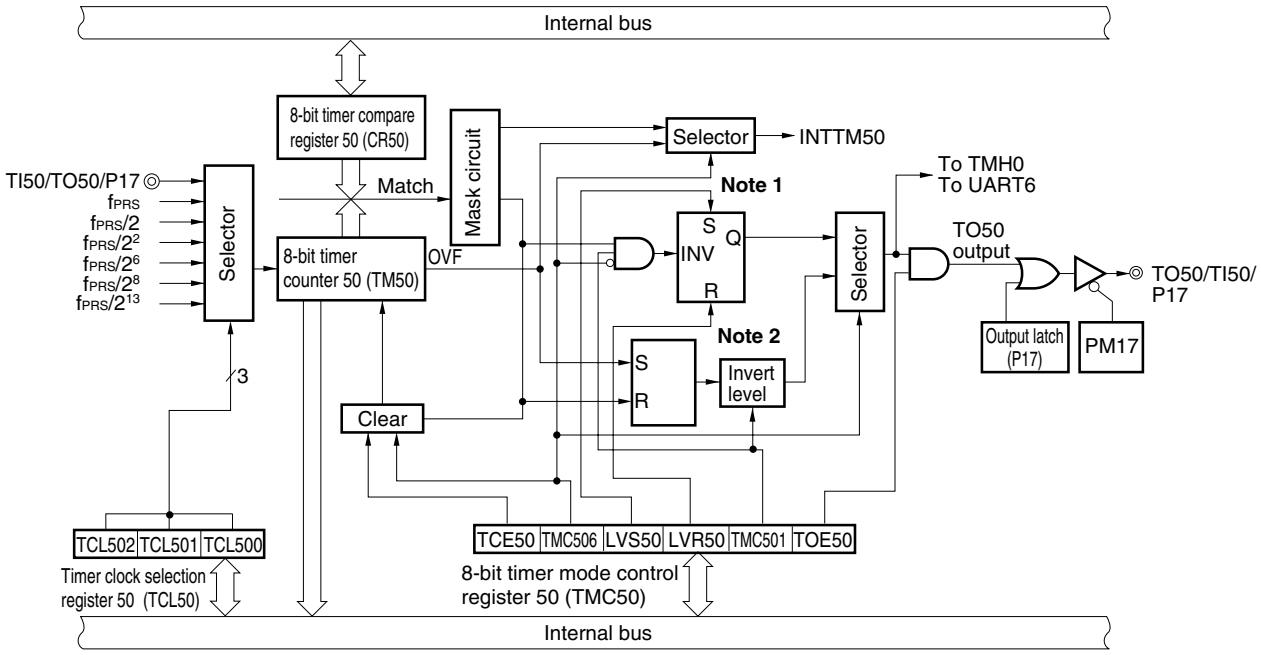
(3) 78K0/KB2-L, 78K0/KC2-L

| Item | Configuration |
|---|---|
| Timer register | 8-bit timer counter 5n (TM5n) |
| Register | 8-bit timer compare register 5n (CR5n) |
| Timer input | TI5n |
| Timer output | TO5n |
| Control registers | Timer clock selection register 5n (TCL5n)<br>8-bit timer mode control register 5n (TMC5n)<br>Port mode register 1 (PM1) or port mode register 3 (PM3)<br>Port register 1 (P1) or port register 3 (P3) |

**Remark** n = 0, 1

Figures 7-1 to 7-4 show the block diagrams of 8-bit timer/event counters 50 and 51.

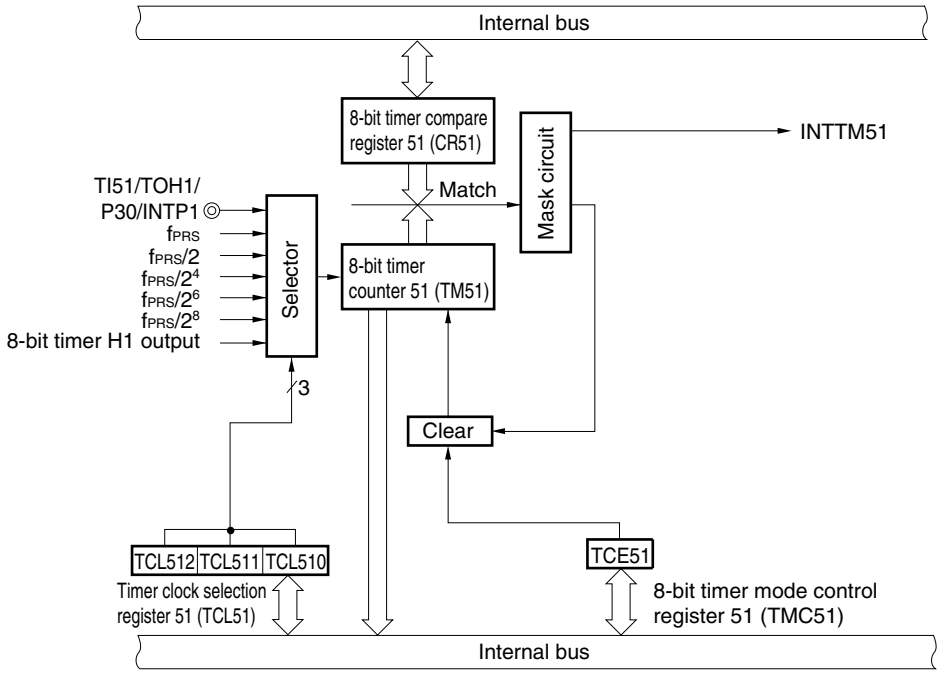
**Figure 7-1. Block Diagram of 8-Bit Timer/Event Counter 50 (78K0/KB2-L, 78K0/KC2-L Only)**



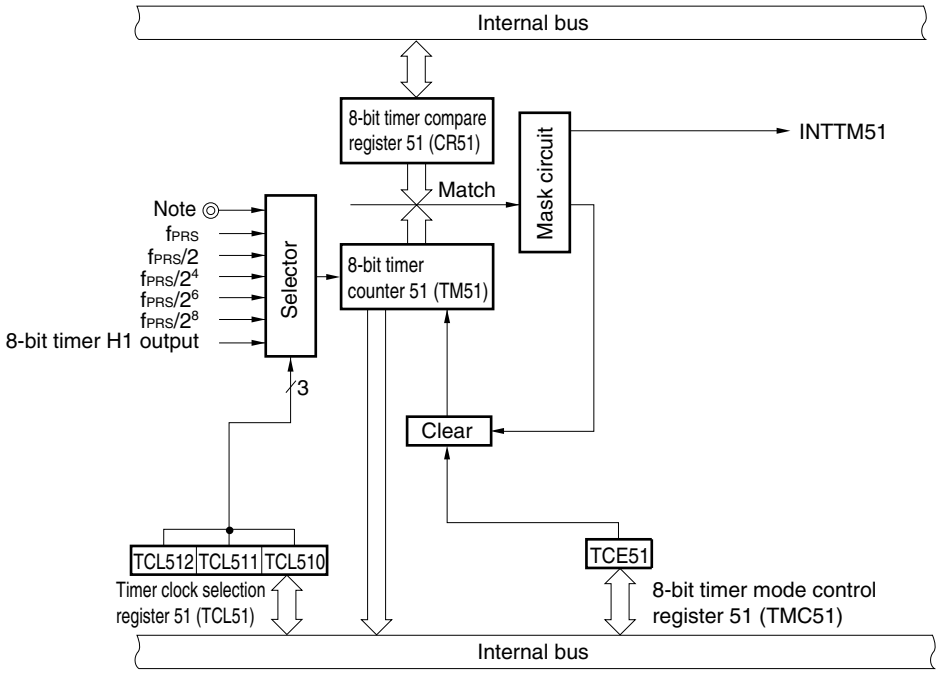

**Notes 1.** Timer output F/F

**2.** PWM output F/F

<R> **Figure 7-2. Block Diagram of 8-Bit Timer 51 (78K0/KY2-L, 78K0/KA2-L (20-pin products))**



<R> **Figure 7-3. Block Diagram of 8-Bit Timer 51 (78K0/KY2-L, 78K0/KA2-L (25-pin and 32-pin products))**




**Note** 78K0/KA2-L (25-pin products)

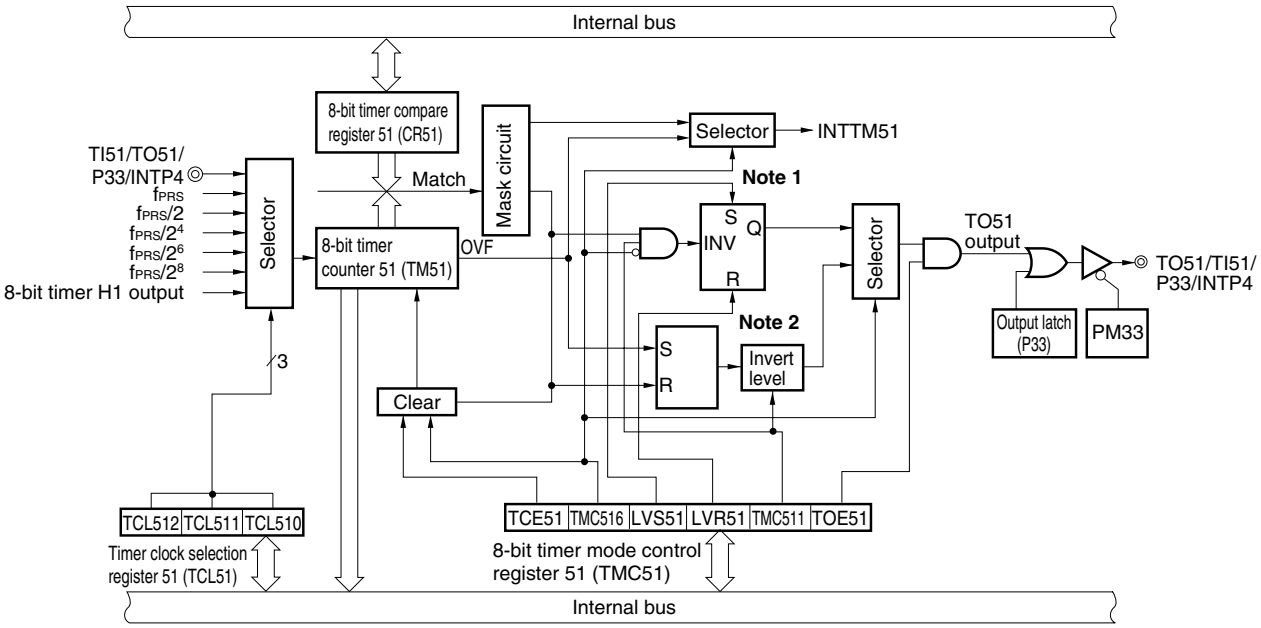
When bits 3 and 2 (TM5SEL1, TM5SEL0) of MUXSEL register = 0, 0: NO timer input pin

When bits 3 and 2 (TM5SEL1, TM5SEL0) of MUXSEL register = 0, 1: (TI51)(/TOH1)/INTP4/P34

When bits 3 and 2 (TM5SEL1, TM5SEL0) of MUXSEL register = 1, 0: (TI51)(/TOH1)/TI000/INTP0/P00

78K0/KA2-L (32-pin products): NO timer input pin

**Figure 7-4. Block Diagram of 8-Bit Timer/Event Counter 51 (78K0/KB2-L, 78K0/KC2-L)**




**Notes 1.** Timer output F/F

**2.** PWM output F/F

**(1) 8-bit timer counter 5n (TM5n)**

TM5n is an 8-bit register that counts the count pulses and is read-only.

The counter is incremented in synchronization with the rising edge of the count clock.

**Figure 7-5. Format of 8-Bit Timer Counter 5n (TM5n)**

Address: FF16H (TM50), FF1FH (TM51) After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| TM5n | | | | | | | | |

In the following situations, the count value is cleared to 00H.

<1> Reset signal generation

<2> When TCE5n is cleared

<3> When TM5n and CR5n match in the mode in which clear & start occurs upon a match of the TM5n and CR5n.

**(2) 8-bit timer compare register 5n (CR5n)**

CR5n can be read and written by an 8-bit memory manipulation instruction.

Except in PWM mode, the value set in CR5n is constantly compared with the 8-bit timer counter 5n (TM5n) count value, and an interrupt request (INTTM5n) is generated if they match.

In the PWM mode, TO5n output becomes inactive when the values of TM5n and CR5n match, but no interrupt is generated.

The value of CR5n can be set within 00H to FFH.

Reset signal generation clears CR5n to 00H.

**Figure 7-6. Format of 8-Bit Timer Compare Register 5n (CR5n)**

Address: FF17H (CR50), FF41H (CR51) After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CR5n | | | | | | | | |

**Cautions 1. In the mode in which clear & start occurs on a match of TM5n and CR5n (TMC5n6 = 0), do not write other values to CR5n during operation.**

**2. In PWM mode, make the CR5n rewrite period 3 count clocks of the count clock (clock selected by TCL5n) or more.**

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1

78K0/KB2-L, 78K0/KC2-L: n = 0, 1

## 7.3 Registers Controlling 8-Bit Timer/Event Counters 50 and 51

The following five registers are used to control 8-bit timer/event counters 50 and 51.

- Timer clock selection register 5n (TCL5n)
- 8-bit timer mode control register 5n (TMC5n)
- <R> Port alternate switch control register (MUXSEL) (78K0/KA2-L (25-pin products) only)
- <R> Port mode register 0 (PM0), port mode register 1 (PM1), or port mode register 3 (PM3)
- <R> Port register 0 (P0), port register 1 (P1), or port register 3 (P3)

### (1) Timer clock selection register 5n (TCL5n)

This register sets the count clock of 8-bit timer/event counter 5n and the valid edge of the TI5n pin input.

TCL5n can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears TCL5n to 00H.

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 7-7. Format of Timer Clock Selection Register 50 (TCL50) (78K0/KB2-L, 78K0/KC2-L Only)**

Address: FF6AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| TCL50 | 0 | 0 | 0 | 0 | 0 | TCL502 | TCL501 | TCL500 |

| TCL502 | TCL501 | TCL500 | Count clock selection[Note 1] | | | |
|---|---|---|---|---|---|---|
| | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | TI50 pin falling edge[Note 2] | | | |
| 0 | 0 | 1 | TI50 pin rising edge[Note 2] | | | |
| 0 | 1 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 1 | 1 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz |
| 1 | 0 | 0 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz |
| 1 | 0 | 1 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz |
| 1 | 1 | 0 | $f_{PRS}/2^8$ | 7.81 kHz | 19.53 kHz | 39.06 kHz |
| 1 | 1 | 1 | $f_{PRS}/2^{13}$ | 0.24 kHz | 0.61 kHz | 1.22 kHz |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.
- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** Do not start timer operation with the external clock from the TI50 pin when the internal high-speed oscillation clock and high-speed system clock are stopped while the CPU operates with the subsystem clock, or when in the STOP mode.

**Cautions 1. When rewriting TCL50 to other data, stop the timer operation beforehand.**
**2. Be sure to clear bits 3 to 7 to "0".**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**Figure 7-8. Format of Timer Clock Selection Register 51 (TCL51)**

Address: FF8CH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| TCL51 | 0 | 0 | 0 | 0 | 0 | TCL512 | TCL511 | TCL510 |

| TCL512 | TCL511 | TCL510 | Count clock selection[Note 1] | | | |
|---|---|---|---|---|---|---|
| | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | TI51 pin falling edge[Note 2] | | | |
| 0 | 0 | 1 | TI51 pin rising edge[Note 2] | | | |
| 0 | 1 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 1 | 1 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz |
| 1 | 0 | 0 | $f_{PRS}/2^4$ | 125 kHz | 312.5 kHz | 625 kHz |
| 1 | 0 | 1 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz |
| 1 | 1 | 0 | $f_{PRS}/2^8$ | 7.81 kHz | 19.53 kHz | 39.06 kHz |
| 1 | 1 | 1 | TMH1 output | | | |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** Do not start timer operation with the external clock from the TI51 pin when the internal high-speed oscillation clock and high-speed system clock are stopped while the CPU operates with the subsystem clock, or when in the STOP mode.

**Cautions 1. When rewriting TCL51 to other data, stop the timer operation beforehand.**

**2. Be sure to clear bits 3 to 7 to "0".**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**(2) 8-bit timer mode control register 5n (TMC5n)**

TMC5n is a register that performs the following five types of settings.

<1> 8-bit timer counter 5n (TM5n) count operation control
<2> 8-bit timer counter 5n (TM5n) operating mode selection
<3> Timer output F/F (flip flop) status setting
<4> Active level selection in timer F/F control or PWM (free-running) mode.
<5> Timer output control

TMC5n can be set by a 1-bit or 8-bit memory manipulation instruction.
Reset signal generation clears this register to 00H.

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 7-9. Format of 8-Bit Timer Mode Control Register 50 (TMC50) (78K0/KB2-L, 78K0/KC2-L Only)**

Address: FF6BH After reset: 00H R/W[Note]

| Symbol | <7> | 6 | 5 | 4 | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| TMC50 | TCE50 | TMC506 | 0 | 0 | LVS50 | LVR50 | TMC501 | TOE50 |

| TCE50 | TM50 count operation control |
|---|---|
| 0 | After clearing to 0, count operation disabled (counter stopped) |
| 1 | Count operation start |

| TMC506 | TM50 operating mode selection |
|---|---|
| 0 | Mode in which clear & start occurs on a match between TM50 and CR50 |
| 1 | PWM (free-running) mode |

| LVS50 | LVR50 | Timer output F/F status setting |
|---|---|---|
| 0 | 0 | No change |
| 0 | 1 | Timer output F/F clear (0) (default value of TO50 output: low level) |
| 1 | 0 | Timer output F/F set (1) (default value of TO50 output: high level) |
| 1 | 1 | Setting prohibited |

| TMC501 | In other modes (TMC506 = 0) | In PWM mode (TMC506 = 1) |
|---|---|---|
| | Timer F/F control | Active level selection |
| 0 | Inversion operation disabled | Active-high |
| 1 | Inversion operation enabled | Active-low |

| TOE50 | Timer output control |
|---|---|
| 0 | Output disabled (TO50 output is low level) |
| 1 | Output enabled |

**Note** Bits 2 and 3 are write-only.

**Cautions 1. The settings of LVS50 and LVR50 are valid in other than PWM mode.**

**2. Perform <1> to <4> below in the following order, not at the same time.**

**<1> Set TMC501, TMC506: Operation mode setting**

**<2> Set TOE50 to enable output: Timer output enable**

**<3> Set LVS50, LVR50 (refer to Caution 1): Timer F/F setting**

**<4> Set TCE50**

**3. When TCE50 = 1, setting the other bits of TMC50 is prohibited.**

**4. The actual TO50/TI50/P17 pin output is determined depending on PM17 and P17 besides TO50 output.**

**Remarks 1.** In PWM mode, PWM output is made inactive by clearing TCE50 to 0.

**2.** If LVS50 and LVR50 are read, the value is 0.

**3.** The values of the TMC506, LVS50, LVR50, TMC501, and TOE50 bits are reflected at the TO50 output regardless of the value of TCE50.

**Figure 7-10. Format of 8-Bit Timer Mode Control Register 51 (TMC51) (1/2)**

**(a) 78K0/KY2-L, 78K0/KA2-L**

Address: FF43H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| TMC51 | TCE51 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| TCE51 | TM51 count operation control |
|---|---|
| 0 | After clearing to 0, count operation disabled (counter stopped) |
| 1 | Count operation start |

**Figure 7-10. Format of 8-Bit Timer Mode Control Register 51 (TMC51) (2/2)**

**(b) 78K0/KB2-L, 78K0/KC2-L**

Address: FF43H After reset: 00H R/W[Note]

| Symbol | <7> | 6 | 5 | 4 | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| TMC51 | TCE51 | TMC516 | 0 | 0 | LVS51 | LVR51 | TMC511 | TOE51 |

| TCE51 | TM51 count operation control |
|---|---|
| 0 | After clearing to 0, count operation disabled (counter stopped) |
| 1 | Count operation start |

| TMC516 | TM51 operating mode selection |
|---|---|
| 0 | Mode in which clear & start occurs on a match between TM51 and CR51 |
| 1 | PWM (free-running) mode |

| LVS51 | LVR51 | Timer output F/F status setting |
|---|---|---|
| 0 | 0 | No change |
| 0 | 1 | Timer output F/F clear (0) (default value of TO51 output: low) |
| 1 | 0 | Timer output F/F set (1) (default value of TO51 output: high) |
| 1 | 1 | Setting prohibited |

| TMC511 | In other modes (TMC516 = 0) | In PWM mode (TMC516 = 1) |
|---|---|---|
| | Timer F/F control | Active level selection |
| 0 | Inversion operation disabled | Active-high |
| 1 | Inversion operation enabled | Active-low |

| TOE51 | Timer output control |
|---|---|
| 0 | Output disabled (TO51 output is low level) |
| 1 | Output enabled |

**Note** Bits 2 and 3 are write-only.

**Cautions 1. The settings of LVS51 and LVR51 are valid in other than PWM mode.**

**2. Perform <1> to <4> below in the following order, not at the same time.**

**<1> Set TMC511, TMC516: Operation mode setting**

**<2> Set TOE51 to enable output: Timer output enable**

**<3> Set LVS51, LVR51 (refer to Caution 1): Timer F/F setting**

**<4> Set TCE51**

**3. When TCE51 = 1, setting the other bits of TMC51 is prohibited.**

**4. The actual TO51/TI51/P33/INTP4 pin output is determined depending on PM33 and P33 besides TO51 output.**

**Remarks 1.** In PWM mode, PWM output is made inactive by clearing TCE51 to 0.

**2.** If LVS51 and LVR51 are read, the value is 0.

**3.** The values of the TMC516, LVS51, LVR51, TMC511, and TOE51 bits are reflected at the TO51 output regardless of the value of TCE51.

<R>**(3) Port alternate switch control register (MUXSEL) (78K0/KA2-L (25-pin) only)**

MUXSEL of 78K0/KA2-L (25-pin products) assigns TOH1, TI51, TI000, and INTP0 pins. By default, INTP0 and TI000 are assigned to P00, while TI51 and TOH1 have no assignment setting.

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

Reset signal generation clears MUXSEL to 00H.

**Figure 7-11. Format of Port Alternate Switch Control Register (MUXSEL)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | 0 | INTP0SEL0 | 0 | TM00SEL0 | TM5SEL1 | TM5SEL0 | TMHSEL1 | TMHSEL0 |

| TM5SEL1 | TM5SEL0 | 8-bit timer 51 input (TI51) pin function assignment |
|---|---|---|
| 0 | 0 | No TI51 function assignment. |
| 0 | 1 | Assign TI51 to the P34 pin as the alternate function. |
| 1 | 0 | Assign TI51 to the P00 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

<R>**(4) Port mode registers 0, 1, 3 (PM0, PM1, PM3)**

These registers set port 0, 1, and 3 input/output in 1-bit units.

PM0, PM1, and PM3 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

• 78K0/KY2-L, 78K0/KA2-L (20-pin products)

When using the P30/TI51/TOH1/INTP1 pin for timer input, set PM30 to 1. The output latches of P30 at this time may be 0 or 1.

• 78K0/KA2-L (25-pin products)

When using the P34/TI51/TOH1/INTP4 pin for timer input, set PM34 to 1. The output latches of P34 at this time may be 0 or 1.

When using the P00/TI51/TOH1/INTP0/TI000 pin for timer input, set PM00 to 1. The output latches of P00 at this time may be 0 or 1.

• 78K0/KB2-L, 78K0/KC2-L

When using the P17/TO50/TI50 and P33/TO51/TI51/INTP4 pins for timer output, clear PM17 and PM33 and the output latches of P17 and P33 to 0.

When using the P17/TO50/TI50 and P33/TO51/TI51/INTP4 pins for timer input, set PM17 and PM33 to 1. The output latches of P17 and P33 at this time may be 0 or 1.

<R> **Figure 7-12. Format of Port Mode Register 0 (PM0)**

Address: FF20H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | PM02 | PM01 | PM00 |

| PM0n | P0n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 0 of the 78K0/KA2-L (25-pin products).

**Figure 7-13. Format of Port Mode Register 1 (PM1)**

Address: FF21H After reset: FFH R/W

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

| PM1n | P1n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 1 of the 78K0/KB2-L and 78K0/KC2-L.

**Figure 7-14. Format of Port Mode Register 3 (PM3)**

Address: FF23H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM3 | 1 | 1 | 1 | 1 | PM33 | PM32 | PM31 | PM30 |

| PM3n | P3n pin I/O mode selection (n = 0 to 3) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 3 of the 78K0/KB2-L and 78K0/KC2-L. For the format of port mode register 3 of other products, refer to **(1) Port mode registers (PMxx)** in **4.3 Registers Controlling Port Function**.

## 7.4 Operations of 8-Bit Timer/Event Counters 50 and 51

### 7.4.1 Operation as interval timer

8-bit timer/event counter 5n operates as an interval timer that generates interrupt requests repeatedly at intervals of the count value preset to 8-bit timer compare register 5n (CR5n).

When the count value of 8-bit timer counter 5n (TM5n) matches the value set to CR5n, counting continues with the TM5n value cleared to 0 and an interrupt request signal (INTTM5n) is generated.

The count clock of TM5n can be selected with bits 0 to 2 (TCL5n0 to TCL5n2) of timer clock selection register 5n (TCL5n).

Setting

<1> Set the registers.
- TCL5n: Select the count clock.
- CR5n: Compare value
- TMC5n: Stop the count operation, select the mode in which clear & start occurs on a match of TM5n and CR5n.
  (TMC5n = 0000×××0B × = Don't care)

<2> After TCE5n = 1 is set, the count operation starts.

<3> If the values of TM5n and CR5n match, INTTM5n is generated (TM5n is cleared to 00H).

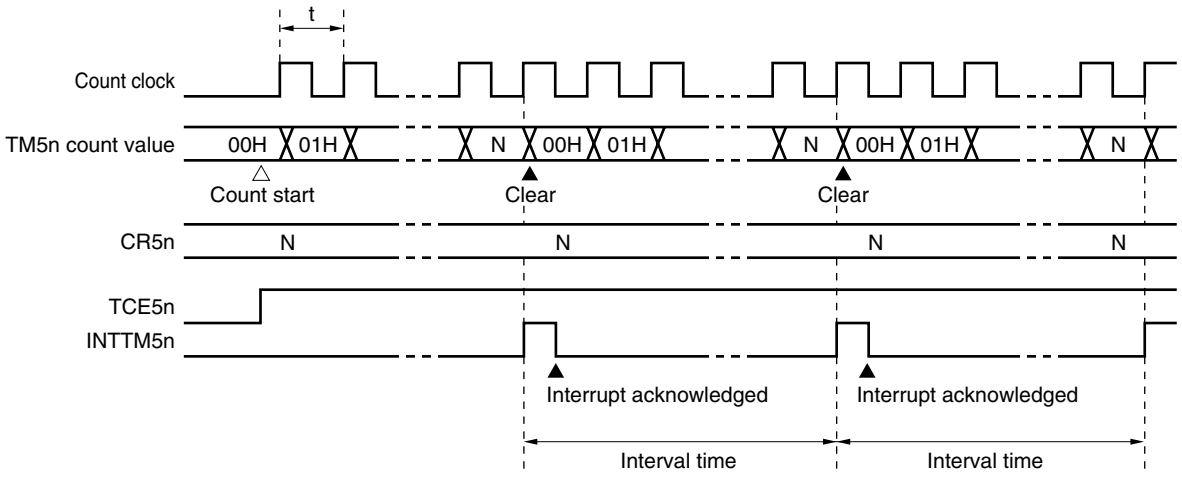
<4> INTTM5n is generated repeatedly at the same interval.
Set TCE5n to 0 to stop the count operation.

**Caution Do not write other values to CR5n during operation.**

**Remark** For how to enable the INTTM5n signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 7-15. Interval Timer Operation Timing (1/2)**

**(a) Basic operation**

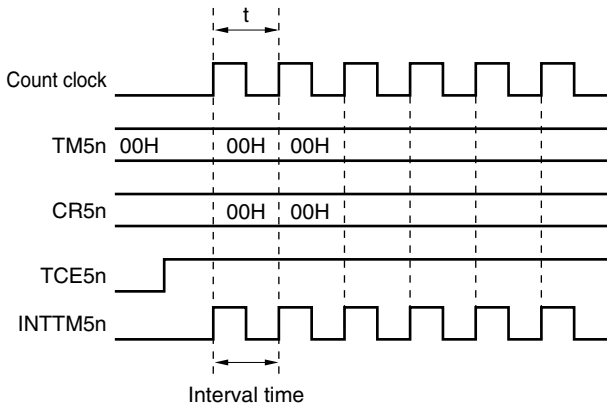



**Remarks 1.** Interval time = (N + 1) × t, N = 01H to FFH

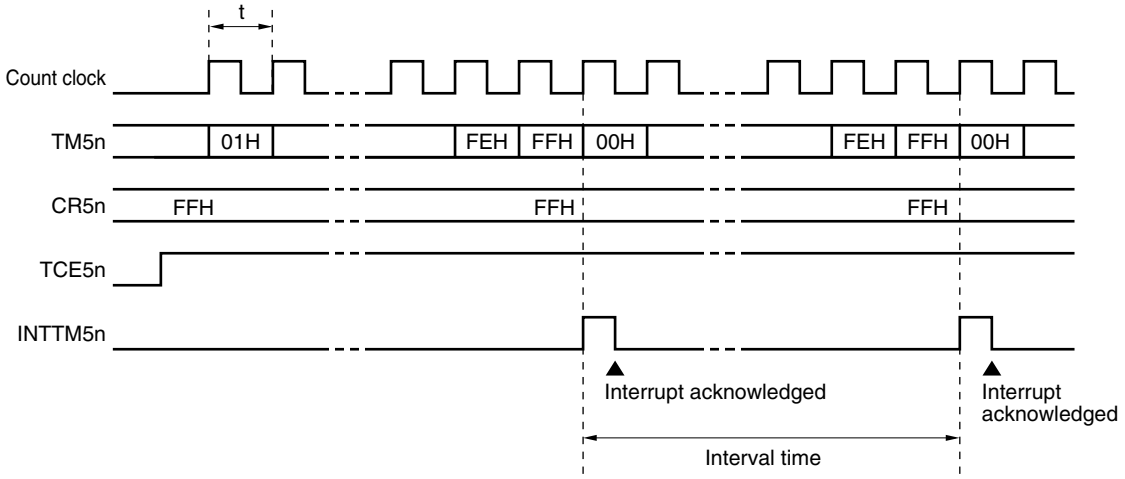
**2.** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 7-15. Interval Timer Operation Timing (2/2)**

**(b) When CR5n = 00H**




**(c) When CR5n = FFH**




**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

### 7.4.2 Operation as external event counter

The external event counter counts the number of external clock pulses to be input to the TI5n pin by 8-bit timer counter 5n (TM5n).

TM5n is incremented each time the valid edge specified by timer clock selection register 5n (TCL5n) is input. Either the rising or falling edge can be selected.

When the TM5n count value matches the value of 8-bit timer compare register 5n (CR5n), TM5n is cleared to 0 and an interrupt request signal (INTTM5n) is generated.

Whenever the TM5n value matches the value of CR5n, INTTM5n is generated.

Setting

<1> Set each register.

- Set the port mode register (PM17, PM30, or PM33)[Note] to 1.
- TCL5n: Select TI5n pin input edge.
  TI5n pin falling edge → TCL5n = 00H
  TI5n pin rising edge → TCL5n = 01H
- CR5n: Compare value
- TMC5n: Stop the count operation, select the mode in which clear & start occurs on match of TM5n and CR5n, disable the timer F/F inversion operation, disable timer output.
  (TMC5n = 00000000B)

<2> When TCE5n = 1 is set, the number of pulses input from the TI5n pin is counted.

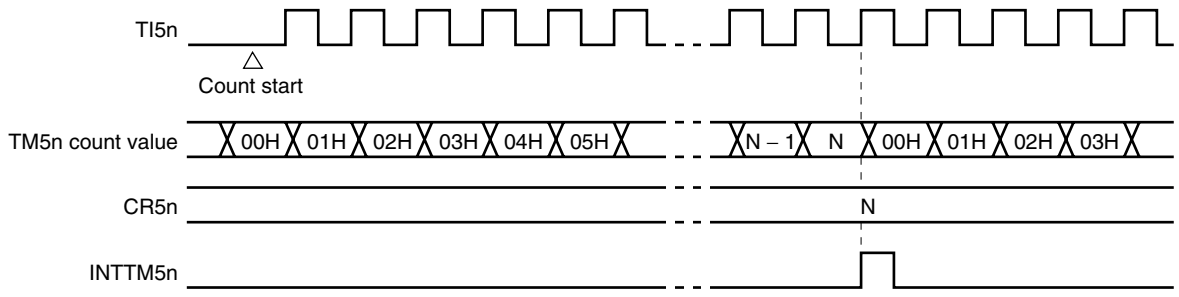
<3> When the values of TM5n and CR5n match, INTTM5n is generated (TM5n is cleared to 00H).

<4> After these settings, INTTM5n is generated each time the values of TM5n and CR5n match.

**Note** 8-bit timer/event counter 50: PM17
8-bit timer/event counter 51: PM30 (78K0/KY2-L, 78K0/KA2-L (20-pin products))
<R> PM34 or PM00 (78K0/KA2-L (25-pin products))
PM33 (78K0/KB2-L, 78K0/KC2-L)

**Remark** For how to enable the INTTM5n signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 7-16. External Event Counter Operation Timing (with Rising Edge Specified)**




**Remarks 1.** N = 00H to FFH

**2.** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

### 7.4.3 Square-wave output operation

A square wave with any selected frequency is output at intervals determined by the value preset to 8-bit timer compare register 5n (CR5n).

The TO5n pin output status is inverted at intervals determined by the count value preset to CR5n by setting bit 0 (TOE5n) of 8-bit timer mode control register 5n (TMC5n) to 1. This enables a square wave with any selected frequency to be output (duty = 50%).

**Remark** Square-wave output is operable only in the 78K0/KB2-L and 78K0/KC2-L.

Setting

<1> Set each register.

- Clear the port output latch (P17 or P33)[Note] and port mode register (PM17 or PM33)[Note] to 0.
- TCL5n: Select the count clock.
- CR5n: Compare value
- TMC5n: Stop the count operation, select the mode in which clear & start occurs on a match of TM5n and CR5n.

| LVS5n | LVR5n | Timer Output F/F Status Setting |
|---|---|---|
| 0 | 1 | Timer output F/F clear (0) (default value of TO5n output: low level) |
| 1 | 0 | Timer output F/F set (1) (default value of TO5n output: high level) |

Timer output enabled
(TMC5n = 00001011B or 00000111B)

<2> After TCE5n = 1 is set, the count operation starts.

<3> The timer output F/F is inverted by a match of TM5n and CR5n. After INTTM5n is generated, TM5n is cleared to 00H.

<4> After these settings, the timer output F/F is inverted at the same interval and a square wave is output from TO5n. The frequency is as follows.

- Frequency = 1/2t (N + 1)
  (N: 00H to FFH)

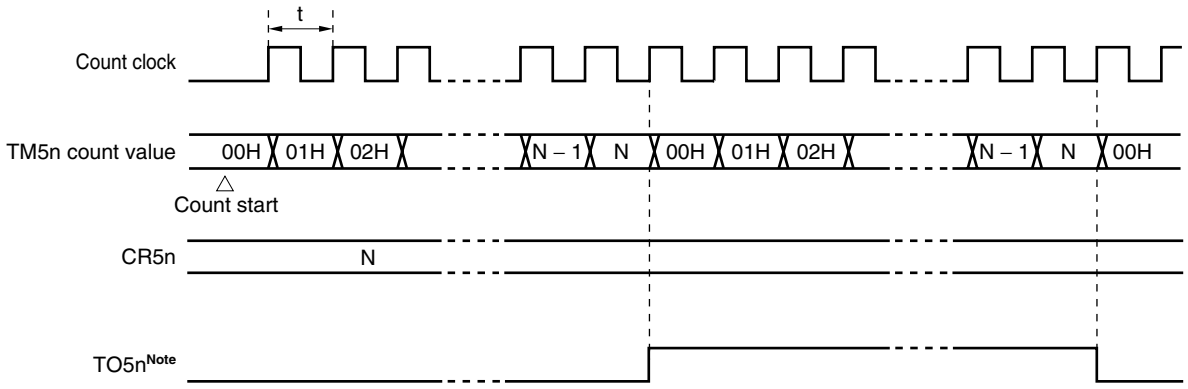
**Note** 8-bit timer/event counter 50: P17, PM17
8-bit timer/event counter 51: P33, PM33

**Caution Do not write other values to CR5n during operation.**

**Remarks 1.** For how to enable the INTTM5n signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**2.** 78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 7-17. Square-Wave Output Operation Timing**




**Note** The initial value of TO5n output can be set by bits 2 and 3 (LVR5n, LVS5n) of 8-bit timer mode control register 5n (TMC5n).

### 7.4.4 PWM output operation

8-bit timer/event counter 5n operates as a PWM output when bit 6 (TMC5n6) of 8-bit timer mode control register 5n (TMC5n) is set to 1.

The duty pulse determined by the value set to 8-bit timer compare register 5n (CR5n) is output from TO5n.

Set the active level width of the PWM pulse to CR5n; the active level can be selected with bit 1 (TMC5n1) of TMC5n.

The count clock can be selected with bits 0 to 2 (TCL5n0 to TCL5n2) of timer clock selection register 5n (TCL5n).

PWM output can be enabled/disabled with bit 0 (TOE5n) of TMC5n.

**Caution In PWM mode, make the CR5n rewrite period 3 count clocks of the count clock (clock selected by TCL5n) or more.**

**Remarks 1.** PWM output is operable only in the 78K0/KB2-L and 78K0/KC2-L.

**2.** 78K0/KB2-L, 78K0/KC2-L: n = 0, 1

### (1) PWM output basic operation

Setting

<1> Set each register.

- Clear the port output latch (P17 or P33)[Note] and port mode register (PM17 or PM33)[Note] to 0.
- TCL5n: Select the count clock.
- CR5n: Compare value
- TMC5n: Stop the count operation, select PWM mode.
  The timer output F/F is not changed.

| TMC5n1 | Active Level Selection |
|---|---|
| 0 | Active-high |
| 1 | Active-low |

Timer output enabled
(TMC5n = 01000001B or 01000011B)

<2> The count operation starts when TCE5n = 1.
Clear TCE5n to 0 to stop the count operation.

**Note** 8-bit timer/event counter 50: P17, PM17
8-bit timer/event counter 51: P33, PM33

PWM output operation

<1> PWM output (TO5n output) outputs an inactive level until an overflow occurs.

<2> When an overflow occurs, the active level is output. The active level is output until CR5n matches the count value of 8-bit timer counter 5n (TM5n).

<3> After the CR5n matches the count value, the inactive level is output until an overflow occurs again.

<4> Operations <2> and <3> are repeated until the count operation stops.

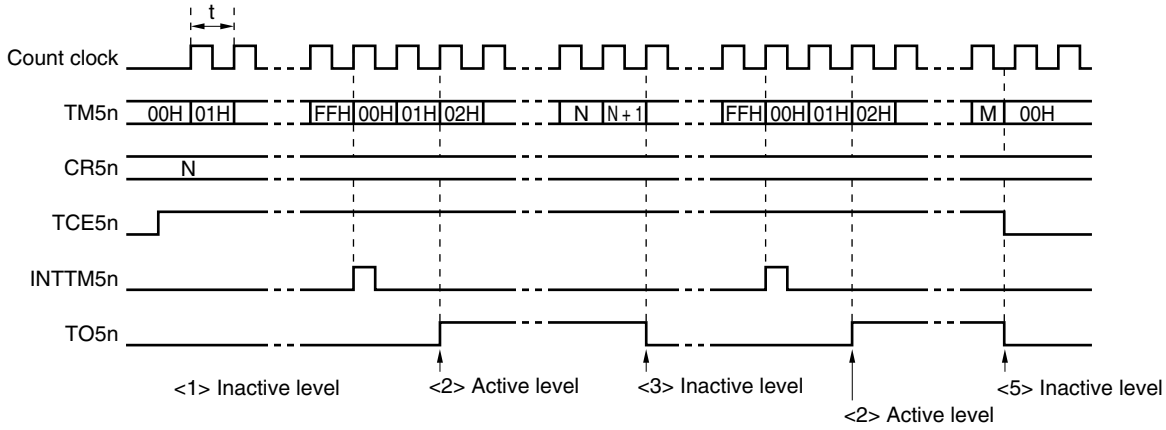
<5> When the count operation is stopped with TCE5n = 0, PWM output becomes inactive.
For details of timing, refer to **Figures 7-18** and **7-19**.
The cycle, active-level width, and duty are as follows.

- Cycle = $2^8$t
- Active-level width = Nt
- Duty = N/$2^8$
  (N = 00H to FFH)

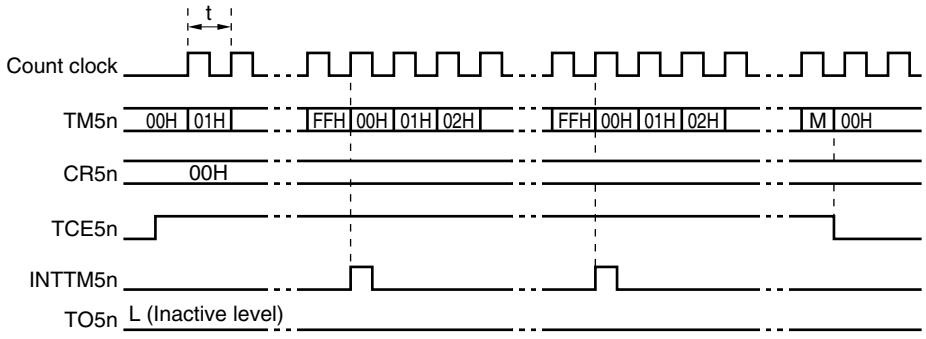
**Remark** 78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 7-18. PWM Output Operation Timing**

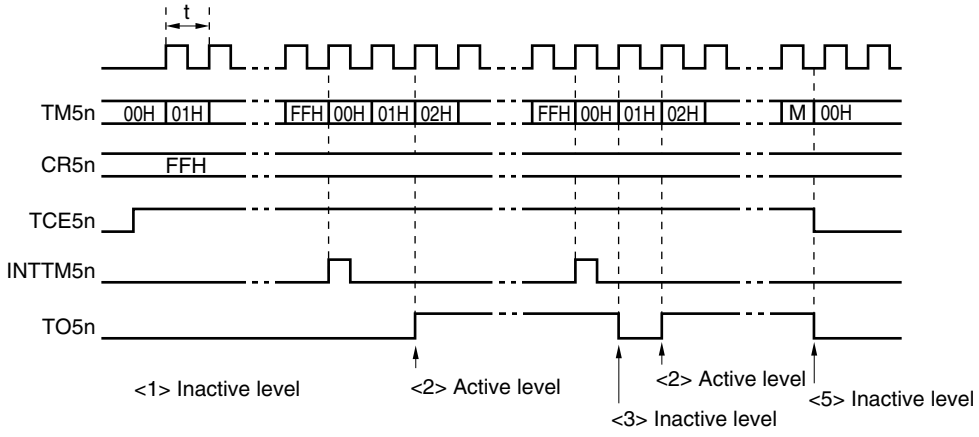
**(a) Basic operation (active level = H)**




**(b) CR5n = 00H**




**(c) CR5n = FFH**



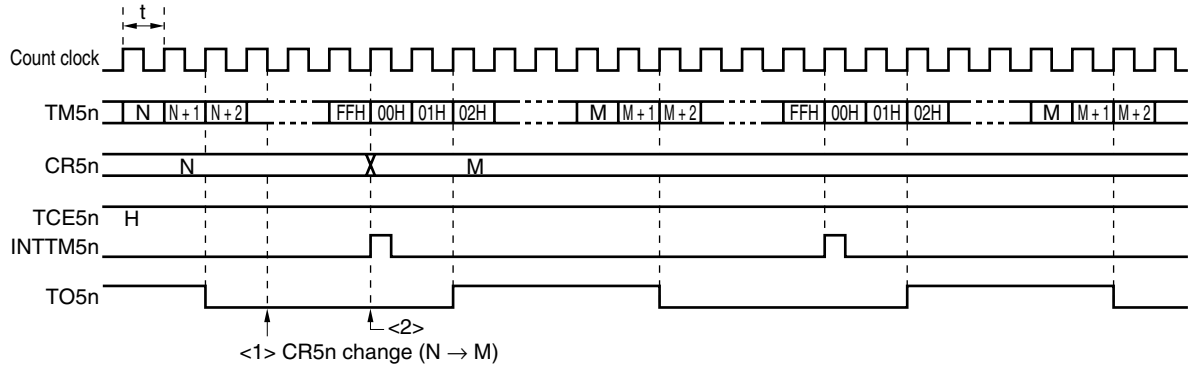

**Remarks 1.** <1> to <3> and <5> in Figure 7-18 (a) and (c) correspond to <1> to <3> and <5> in PWM output operation in **7.4.4 (1) PWM output basic operation**.

**2.** 78K0/KB2-L, 78K0/KC2-L: n = 0, 1

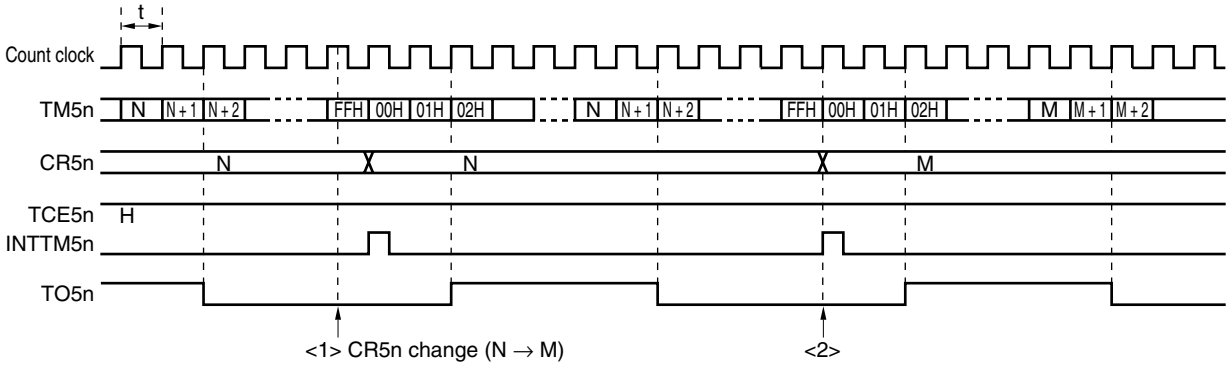
### (2) Operation with CR5n changed

**Figure 7-19. Timing of Operation with CR5n Changed**

**(a) CR5n value is changed from N to M before clock rising edge of FFH**
**→ Value is transferred to CR5n at overflow immediately after change.**




**(b) CR5n value is changed from N to M after clock rising edge of FFH**
**→ Value is transferred to CR5n at second overflow.**




**Caution When reading from CR5n between <1> and <2> in Figure 7-19, the value read differs from the actual value (read value: M, actual value of CR5n: N).**

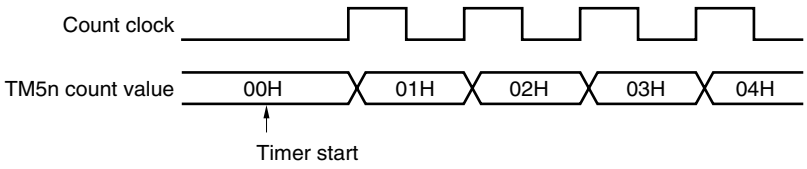
**Remark** 78K0/KB2-L, 78K0/KC2-L: n = 0, 1

## 7.5 Cautions for 8-Bit Timer/Event Counters 50 and 51

**(1) Timer start error**

An error of up to one clock may occur in the time required for a match signal to be generated after timer start. This is because 8-bit timer counters 50 and 51 (TM50, TM51) are started asynchronously to the count clock.

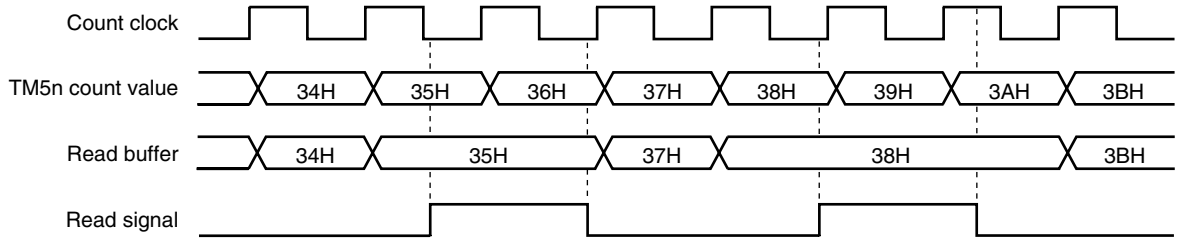
**Figure 7-20. 8-Bit Timer Counter 5n (TM5n) Start Timing**




**(2) Reading of 8-bit timer counter 5n (TM5n)**

TM5n can be read without stopping the actual counter, because the count values captured to the buffer are fixed when it is read. The buffer, however, may not be updated when it is read immediately before the counter counts up, because the buffer is updated at the timing the counter counts up.

**Figure 7-21. 8-bit Timer Counter 5n (TM5n) Read Timing**




**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

# CHAPTER 8 8-BIT TIMERS H0 AND H1

<R>

| Item | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|---|---|
| | 16 Pins | 20 Pins | 25, 32 Pins | 30 Pins | 40, 44, 48 Pins |
| 8-bit timer H0 | – | | | √ (PWM output: 1) | |
| 8-bit timer H1 | √ (PWM output: 1) | | √ (PWM output: 1[Note]) | √ (PWM output: 1) | |

**Remark** √: Mounted, –: Not mounted

**Note** Only when the TOH1 function is assigned by setting the MUXSEL register

## 8.1 Functions of 8-Bit Timers H0 and H1

8-bit timers H0 and H1 have the following functions.

- Interval timer
- Square-wave output
- PWM output
- Carrier generator (8-bit timer H1 only)

## 8.2 Configuration of 8-Bit Timers H0 and H1

8-bit timers H0 and H1 include the following hardware.

**Table 8-1. Configuration of 8-Bit Timers H0 and H1**

| Item | Configuration |
|---|---|
| Timer register | 8-bit timer counter Hn |
| Registers | 8-bit timer H compare register 0n (CMP0n)<br>8-bit timer H compare register 1n (CMP1n) |
| Timer output | TOHn, output controller |
| Control registers | 8-bit timer H mode register n (TMHMDn)<br>8-bit timer H carrier control register 1 (TMCYC1)[Note 1]<br>Port alternate switch control register (MUXSEL)[Note 2]<br>Port mode register 0 (PM0), port mode register 1 (PM1), or port mode register 3 (PM3)<br>Port register 0 (P0), port register 1 (P1), or port register 3 (P3) |

<R>
<R>
<R>

**Notes 1.** 8-bit timer H1 only

**2.** 78K0/KA2-L (25-pin and 32-pin products) only

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-1. Block Diagram of 8-Bit Timer H0 (78K0/KB2-L, 78K0/KC2-L Only)**

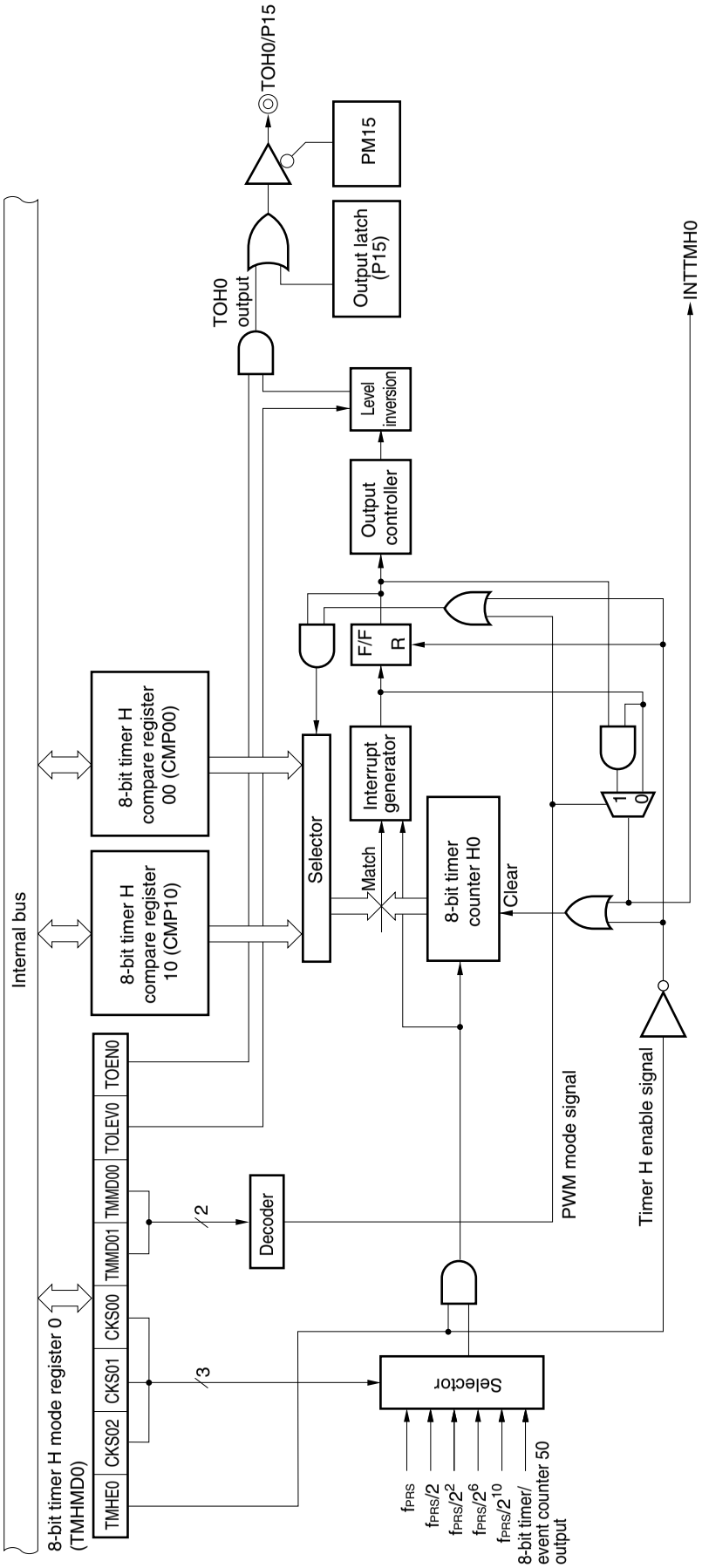
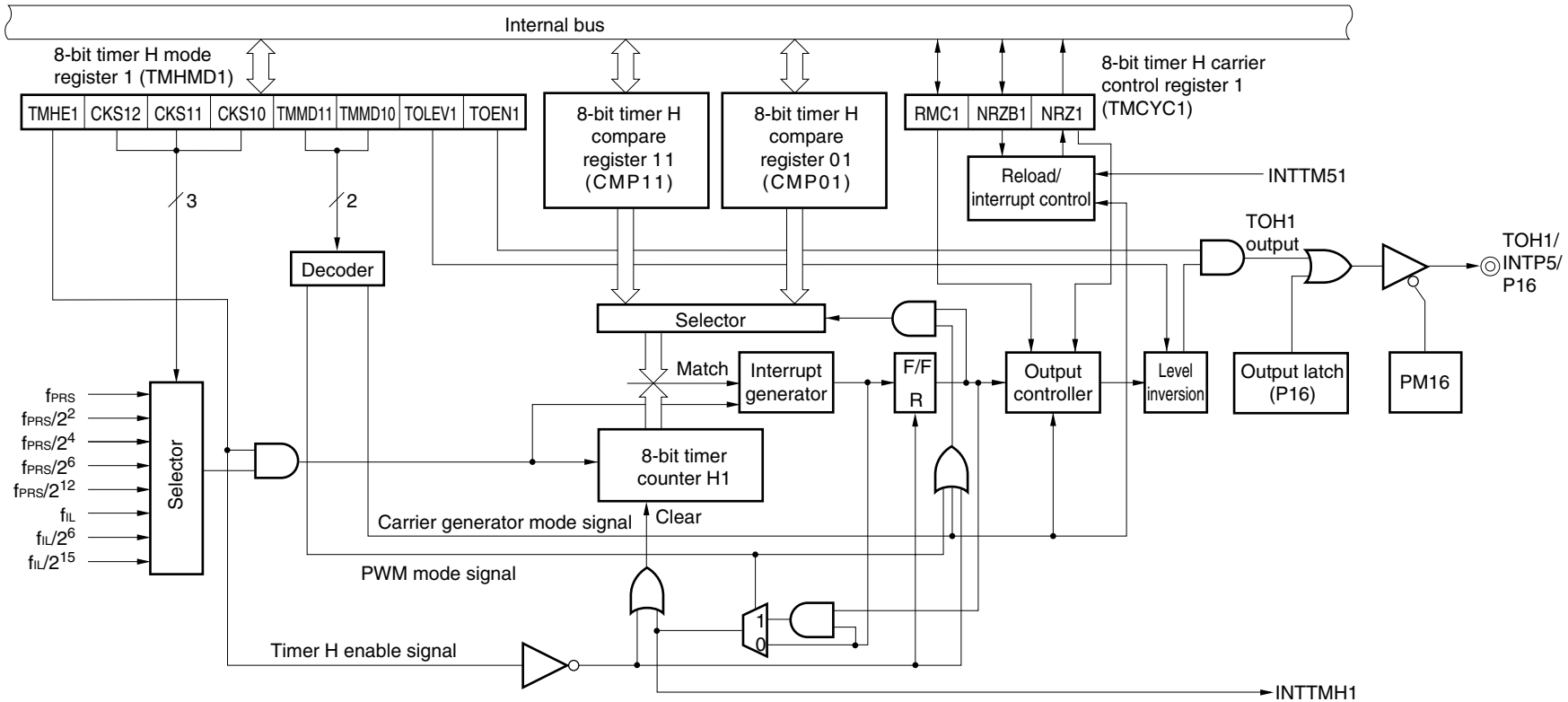


**Figure 8-2. Block Diagram of 8-Bit Timer H1**




<R> **Remark** 78K0/KY2-L, 78K0/KA2-L (20-pin products): TOH1/TI51/INTP1/P30

| | | |
|---|---|---|
| 78K0/KA2-L (25-pin products): | (TOH1)/(TI51)/INTP4/P34 | (When TMHSEL1,TMHSEL0 = 0, 1) |
| | (TOH1)/(TI51)/INTP0/TI00/P00 | (When TMHSEL1,TMHSEL0 = 1, 0) |
| 78K0/KA2-L (32-pin products): | (TOH1)/INTP4/P34 | (When TMHSEL0 = 1) |
| 78K0/KB2-L, 78K0/KC2-L: | TOH1/INTP5/P16 | |

**(1) 8-bit timer H compare register 0n (CMP0n)**

This register can be read or written by an 8-bit memory manipulation instruction. This register is used in all of the timer operation modes.

This register constantly compares the value set to CMP0n with the count value of the 8-bit timer counter Hn and, when the two values match, generates an interrupt request signal (INTTMHn) and inverts the output level of TOHn.

Rewrite the value of CMP0n while the timer is stopped (TMHEn = 0).

A reset signal generation clears this register to 00H.

**Figure 8-3. Format of 8-Bit Timer H Compare Register 0n (CMP0n)**

Address: FF18H (CMP00), FF1AH (CMP01) After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CMP0n | | | | | | | | |

**Caution CMP0n cannot be rewritten during timer count operation. CMP0n can be refreshed (the same value is written) during timer count operation.**

**(2) 8-bit timer H compare register 1n (CMP1n)**

This register can be read or written by an 8-bit memory manipulation instruction. This register is used in the PWM output mode and carrier generator mode.

In the PWM output mode, this register constantly compares the value set to CMP1n with the count value of the 8-bit timer counter Hn and, when the two values match, inverts the output level of TOHn. No interrupt request signal is generated.

In the carrier generator mode, the CMP1n register always compares the value set to CMP1n with the count value of the 8-bit timer counter Hn and, when the two values match, generates an interrupt request signal (INTTMHn). At the same time, the count value is cleared.

CMP1n can be refreshed (the same value is written) and rewritten during timer count operation.

If the value of CMP1n is rewritten while the timer is operating, the new value is latched and transferred to CMP1n when the count value of the timer matches the old value of CMP1n, and then the value of CMP1n is changed to the new value. If matching of the count value and the CMP1n value and writing a value to CMP1n conflict, the value of CMP1n is not changed.

A reset signal generation clears this register to 00H.

**Figure 8-4. Format of 8-Bit Timer H Compare Register 1n (CMP1n)**

Address: FF19H (CMP10), FF1BH (CMP11) After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CMP1n | | | | | | | | |

**Caution In the PWM output mode and carrier generator mode, be sure to set CMP1n when starting the timer count operation (TMHEn = 1) after the timer count operation was stopped (TMHEn = 0) (be sure to set again even if setting the same value to CMP1n).**

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

## 8.3 Registers Controlling 8-Bit Timers H0 and H1

The following five registers are used to control 8-bit timers H0 and H1.

- 8-bit timer H mode register n (TMHMDn)
- 8-bit timer H carrier control register 1 (TMCYC1)[Note]
<R> - Port alternate switch control register (MUXSEL) (78K0/KA2-L (25, 32-pin products) only)
<R> - Port mode register 0 (PM0), port mode register 1 (PM1), or port mode register 3 (PM3)
<R> - Port register 0 (P0), port register 1 (P1), or port register 3 (P3)

**Note** 8-bit timer H1 only

### (1) 8-bit timer H mode register n (TMHMDn)

This register controls the mode of timer H.

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

Reset signal generation clears this register to 00H.

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-5. Format of 8-Bit Timer H Mode Register 0 (TMHMD0)**
**(78K0/KB2-L, 78K0/KC2-L Only)**

Address: FF69H After reset: 00H R/W

| | <7> | 6 | 5 | 4 | 3 | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| TMHMD0 | TMHE0 | CKS02 | CKS01 | CKS00 | TMMD01 | TMMD00 | TOLEV0 | TOEN0 |

| TMHE0 | Timer operation enable |
|---|---|
| 0 | Stops timer count operation (counter is cleared to 0) |
| 1 | Enables timer count operation (count operation started by inputting clock) |

| CKS02 | CKS01 | CKS00 | Count clock selection[Note 1] | | | |
|---|---|---|---|---|---|---|
| | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 0 | 1 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz |
| 0 | 1 | 0 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz |
| 0 | 1 | 1 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz |
| 1 | 0 | 0 | $f_{PRS}/2^{10}$ | 1.95 kHz | 4.88 kHz | 9.77 kHz |
| 1 | 0 | 1 | TM50 output[Note 2] | | | |
| Other than above | | | Setting prohibited | | | |

| TMMD01 | TMMD00 | Timer operation mode |
|---|---|---|
| 0 | 0 | Interval timer mode |
| 1 | 0 | PWM output mode |
| Other than above | | Setting prohibited |

| TOLEV0 | Timer output level control (in default mode) |
|---|---|
| 0 | Low level |
| 1 | High level |

| TOEN0 | Timer output control |
|---|---|
| 0 | Disables output |
| 1 | Enables output |

**Note 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**Note 2.** Note the following points when selecting the TM50 output as the count clock.

- Mode in which the count clock is cleared and started upon a match of TM50 and CR50 (TMC506 = 0)
  Start the operation of 8-bit timer/event counter 50 first and then enable the timer F/F inversion operation (TMC501 = 1).
- PWM mode (TMC506 = 1)
  Start the operation of 8-bit timer/event counter 50 first and then set the count clock to make the duty = 50%.

It is not necessary to enable (TOE50 = 1) TO50 output in any mode.

**Cautions 1. When TMHE0 = 1, setting the other bits of TMHMD0 is prohibited. However, TMHMD0 can be refreshed (the same value is written).**

**2. In the PWM output mode, be sure to set the 8-bit timer H compare register 10 (CMP10) when starting the timer count operation (TMHE0 = 1) after the timer count operation was stopped (TMHE0 = 0) (be sure to set again even if setting the same value to CMP10).**

**3. The actual TOH0/P15 pin output is determined depending on PM15 and P15, besides TOH0 output.**

**Remarks 1.** $f_{PRS}$: Peripheral hardware clock frequency

**2.** TMC506: Bit 6 of 8-bit timer mode control register 50 (TMC50)
TMC501: Bit 1 of TMC50

**Figure 8-6. Format of 8-Bit Timer H Mode Register 1 (TMHMD1)**

Address: FF6CH After reset: 00H R/W

| | <7> | 6 | 5 | 4 | 3 | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| TMHMD1 | TMHE1 | CKS12 | CKS11 | CKS10 | TMMD11 | TMMD10 | TOLEV1 | TOEN1 |

| TMHE1 | Timer operation enable |
|---|---|
| 0 | Stops timer count operation (counter is cleared to 0) |
| 1 | Enables timer count operation (count operation started by inputting clock) |

| CKS12 | CKS11 | CKS10 | Count clock selection[Note] | | | |
|---|---|---|---|---|---|---|
| | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 0 | 1 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz |
| 0 | 1 | 0 | $f_{PRS}/2^4$ | 125 kHz | 312.5 kHz | 625 kHz |
| 0 | 1 | 1 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz |
| 1 | 0 | 0 | $f_{PRS}/2^{12}$ | 0.49 kHz | 1.22 kHz | 2.44 kHz |
| 1 | 0 | 1 | $f_{IL}/2^6$ | 0.47 kHz (TYP.) | | |
| 1 | 1 | 0 | $f_{IL}/2^{15}$ | 0.92 Hz (TYP.) | | |
| 1 | 1 | 1 | $f_{IL}$ | 30 kHz (TYP.) | | |

| TMMD11 | TMMD10 | Timer operation mode |
|---|---|---|
| 0 | 0 | Interval timer mode |
| 0 | 1 | Carrier generator mode |
| 1 | 0 | PWM output mode |
| 1 | 1 | Setting prohibited |

| TOLEV1 | Timer output level control (in default mode) |
|---|---|
| 0 | Low level |
| 1 | High level |

| TOEN1 | Timer output control |
|---|---|
| 0 | Disables output |
| 1 | Enables output |

**Note** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**Cautions 1. When TMHE1 = 1, setting the other bits of TMHMD1 is prohibited. However, TMHMD1 can be refreshed (the same value is written).**

**2. In the PWM output mode and carrier generator mode, be sure to set the 8-bit timer H compare register 11 (CMP11) when starting the timer count operation (TMHE1 = 1) after the timer count operation was stopped (TMHE1 = 0) (be sure to set again even if setting the same value to CMP11).**

**3. When the carrier generator mode is used, set so that the count clock frequency of TMH1 becomes more than 6 times the count clock frequency of TM51.**

**4. In the 78K0/KB2-L and 78K0/KC2-L, the actual TOH1/INTP5/P16 pin output is determined depending on PM16 and P16, besides TOH1 output.**

**5. In the 78K0/KY2-L and 78K0/KA2-L (20-pin products), the actual TOH1/TI51/INTP1/P30 pin output is determined depending on PM30 and P30, besides TOH1 output.**

**Remarks 1.** $f_{PRS}$: Peripheral hardware clock frequency

**2.** $f_{IL}$: Internal low-speed oscillation clock frequency

**(2) 8-bit timer H carrier control register 1 (TMCYC1)**

This register controls the remote control output and carrier pulse output status of 8-bit timer H1.

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

Reset signal generation clears this register to 00H.

**Figure 8-7. Format of 8-Bit Timer H Carrier Control Register 1 (TMCYC1)**

Address: FF6DH After reset: 00H R/W[Note]

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| TMCYC1 | 0 | 0 | 0 | 0 | 0 | RMC1 | NRZB1 | NRZ1 |

| RMC1 | NRZB1 | Remote control output |
|---|---|---|
| 0 | 0 | Low-level output |
| 0 | 1 | High-level output at rising edge of INTTM51 signal input |
| 1 | 0 | Low-level output |
| 1 | 1 | Carrier pulse output at rising edge of INTTM51 signal input |

| NRZ1 | Carrier pulse output status flag |
|---|---|
| 0 | Carrier output disabled status (low-level status) |
| 1 | Carrier output enabled status<br>(RMC1 = 1: Carrier pulse output, RMC1 = 0: High-level status) |

**Note** Bit 0 is read-only.

**Caution Do not rewrite RMC1 when TMHE = 1. However, TMCYC1 can be refreshed (the same value is written).**

<R> **(3) Port alternate switch control register (MUXSEL) (78K0/KA2-L (25, 32-pin products) only)**

MUXSEL of 78K0/KA2-L (25-pin and 32-pin products) assigns TOH1, TI51, TI000, and INTP0 pins. By default, TOH1 has no assignment setting.

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

Reset signal generation clears MUXSEL to 00H.

**Figure 8-8. Format of Port Alternate Switch Control Register (MUXSEL)**

**(1) 78K0/KA2-L (25-pin products)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | 0 | INTP0SEL0 | 0 | TM00SEL0 | TM5SEL1 | TM5SEL0 | TMHSEL1 | TMHSEL0 |

| TMHSEL1 | TMHSEL0 | 8-bit timer H1 output (TOH1) pin function assignment |
|---|---|---|
| 0 | 0 | No TOH1 function assignment. |
| 0 | 1 | Assign TOH1 to the P34 pin as the alternate function. |
| 1 | 0 | Assign TOH1 to the P00 pin as the alternate function. |
| 1 | 1 | Setting prohibited |

**(2) 78K0/KA2-L (32-pin products)**

Address: FF39H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | INTP0SEL1 | INTP0SEL0 | TM00SEL1 | TM00SEL0 | 0 | 0 | 0 | TMHSEL0 |

| TMHSEL0 | 8-bit timer H1 output (TOH1) pin function assignment |
|---|---|
| 0 | No TOH1 function assignment. |
| 1 | Assign TOH1 to the P34 pin as the alternate function. |

<R> **(4) Port mode register 0 (PM0), port mode register 1 (PM1), port mode register 3 (PM3)**

This register sets port 0 input/output, port 1 input/output, and port 3 input/output in 1-bit units.

• 78K0/KY2-L, 78K0/KA2-L (20-pin products)

When using the P30/TOH1/TI51/INTP1 pins for timer output, clear PM30 and the output latches of P30 to 0.

• 78K0/KA2-L (25-pin products)

When using the P34(/TOH1)(/TI51)/INTP4 pins for timer output, clear PM34 and the output latches of P34 to 0.

When using the P00(/TOH1)(/TI51)/INTP0/TI000 pins for timer output, clear PM00 and the output latches of P00 to 0.

• 78K0/KA2-L (32-pin products)

When using the P34(/TOH1)/INTP4 pins for timer output, clear PM34 and the output latches of P34 to 0.

• 78K0/KB2-L, 78K0/KC2-L

When using the P15/TOH0 and P16/TOH1/INTP5 pins for timer output, clear PM15 and PM16 and the output latches of P15 and P16 to 0.

PM0, PM1, and PM3 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

<R> **Figure 8-9. Format of Port Mode Register 0 (PM0)**

Address: FF20H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | PM02 | PM01 | PM00 |

| PM0n | P0n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 0 of the 78K0/KA2-L (25-pin products).

**Figure 8-10. Format of Port Mode Register 1 (PM1)**

Address: FF21H After reset: FFH R/W

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

| PM1n | P1n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 1 of the 78K0/KB2-L and 78K0/KC2-L.

**Figure 8-11. Format of Port Mode Register 3 (PM3)**

Address: FF23H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM3 | 1 | 1 | 1 | 1 | 1 | PM32 | PM31 | PM30 |

| PM3n | P3n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 3 of the 78K0/KA2-L (20-pin products). For the format of port mode register 3 of other products, refer to **(1) Port mode registers (PMxx)** in **4.3 Registers Controlling Port Function**.

## 8.4 Operation of 8-Bit Timers H0 and H1

### 8.4.1 Operation as interval timer/square-wave output

When the 8-bit timer counter Hn and compare register 0n (CMP0n) match, an interrupt request signal (INTTMHn) is generated and the 8-bit timer counter Hn is cleared to 00H.

Compare register 1n (CMP1n) is not used in interval timer mode. Since a match of the 8-bit timer counter Hn and the CMP1n register is not detected even if the CMP1n register is set, timer output is not affected.

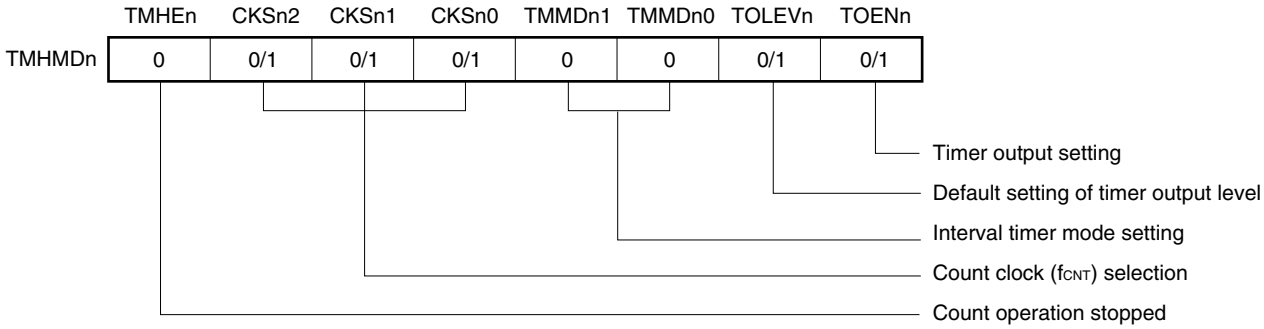
By setting bit 0 (TOENn) of timer H mode register n (TMHMDn) to 1, a square wave of any frequency (duty = 50%) is output from TOHn.

Setting

<1> Set each register.

**Figure 8-12. Register Setting During Interval Timer/Square-Wave Output Operation**

**(i) Setting timer H mode register n (TMHMDn)**




**(ii) CMP0n register setting**

The interval time is as follows if N is set as a comparison value.

- Interval time = (N +1)/$f_{CNT}$

<2> Count operation starts when TMHEn = 1.

<3> When the values of the 8-bit timer counter Hn and the CMP0n register match, the INTTMHn signal is generated and the 8-bit timer counter Hn is cleared to 00H.

<4> Subsequently, the INTTMHn signal is generated at the same interval. To stop the count operation, clear TMHEn to 0.

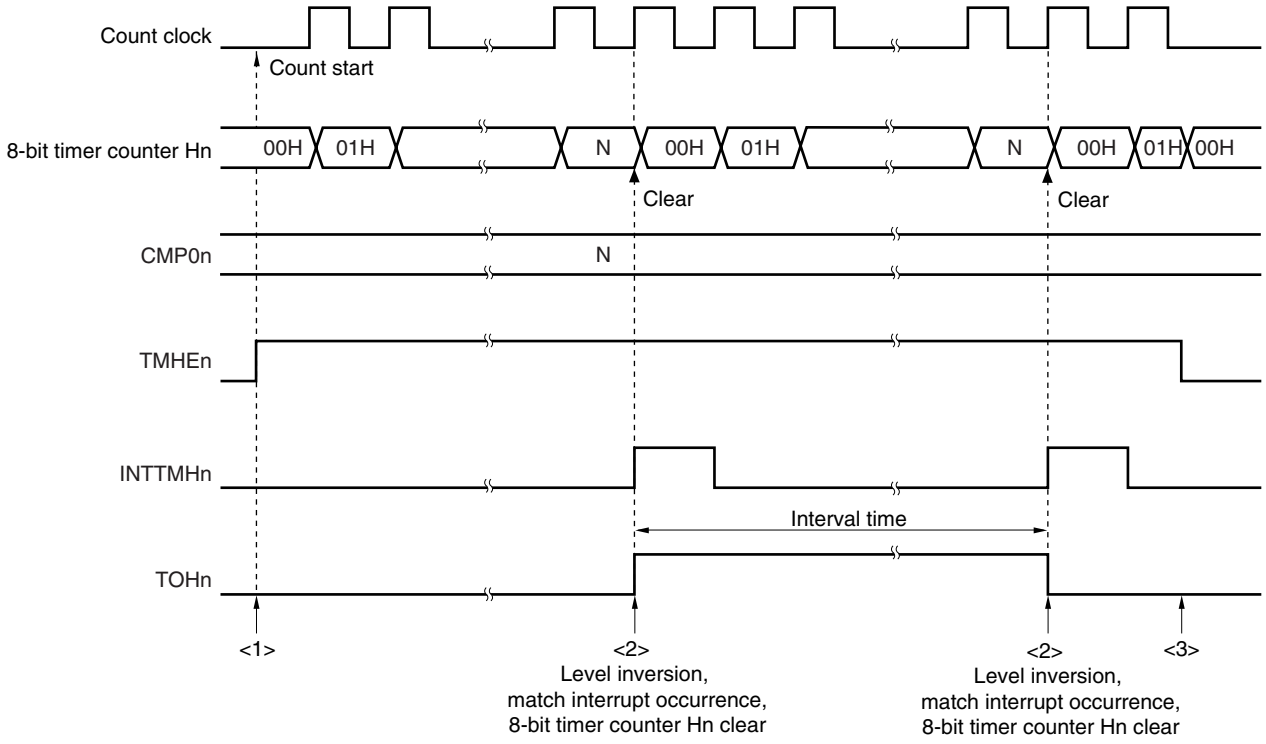
**Remarks 1.** For the setting of the output pin, refer to **8.3 (4) Port mode register 0 (PM0), port mode register 1 (PM1), port mode register 3 (PM3)**.

**2.** For how to enable the INTTMHn signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**3.** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-13. Timing of Interval Timer/Square-Wave Output Operation (1/2)**

**(a) Basic operation (Operation When 01H ≤ CMP0n ≤ FEH)**




<1> The count operation is enabled by setting the TMHEn bit to 1. The count clock starts counting no more than 1 clock after the operation is enabled.

<2> When the value of the 8-bit timer counter Hn matches the value of the CMP0n register, the value of the timer counter is cleared, and the level of the TOHn output is inverted. In addition, the INTTMHn signal is output at the rising edge of the count clock.

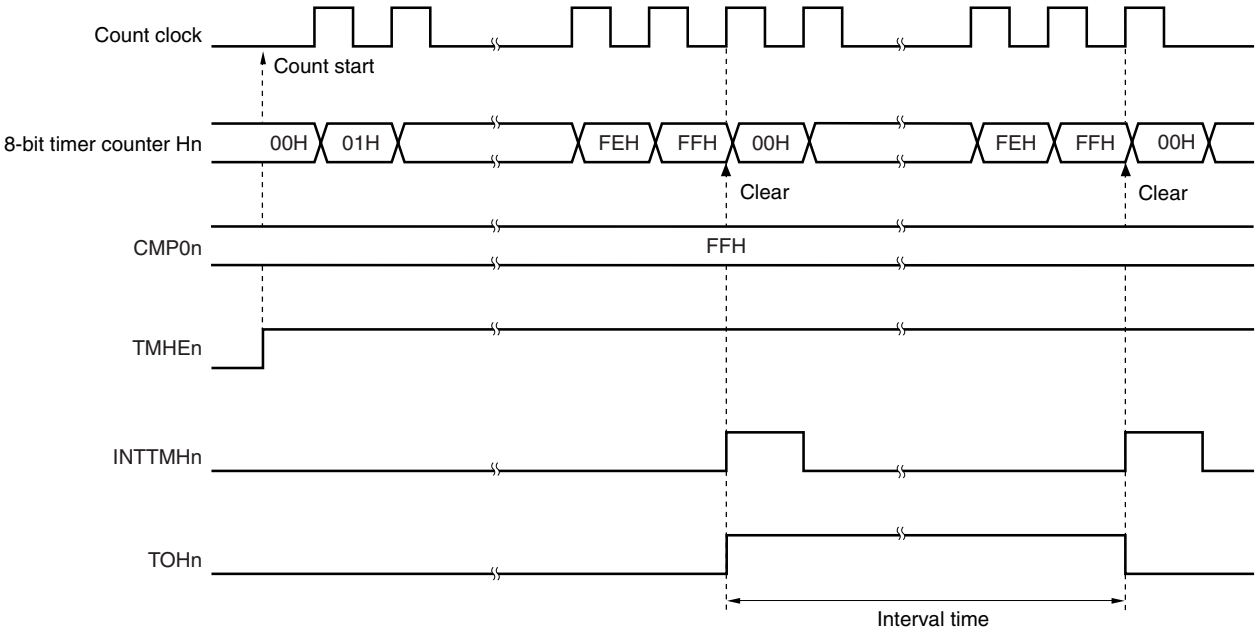
<3> If the TMHEn bit is cleared to 0 while timer H is operating, the INTTMHn signal and TOHn output are set to the default level. If they are already at the default level before the TMHEn bit is cleared to 0, then that level is maintained.

**Remarks 1.** 01H ≤ N ≤ FEH

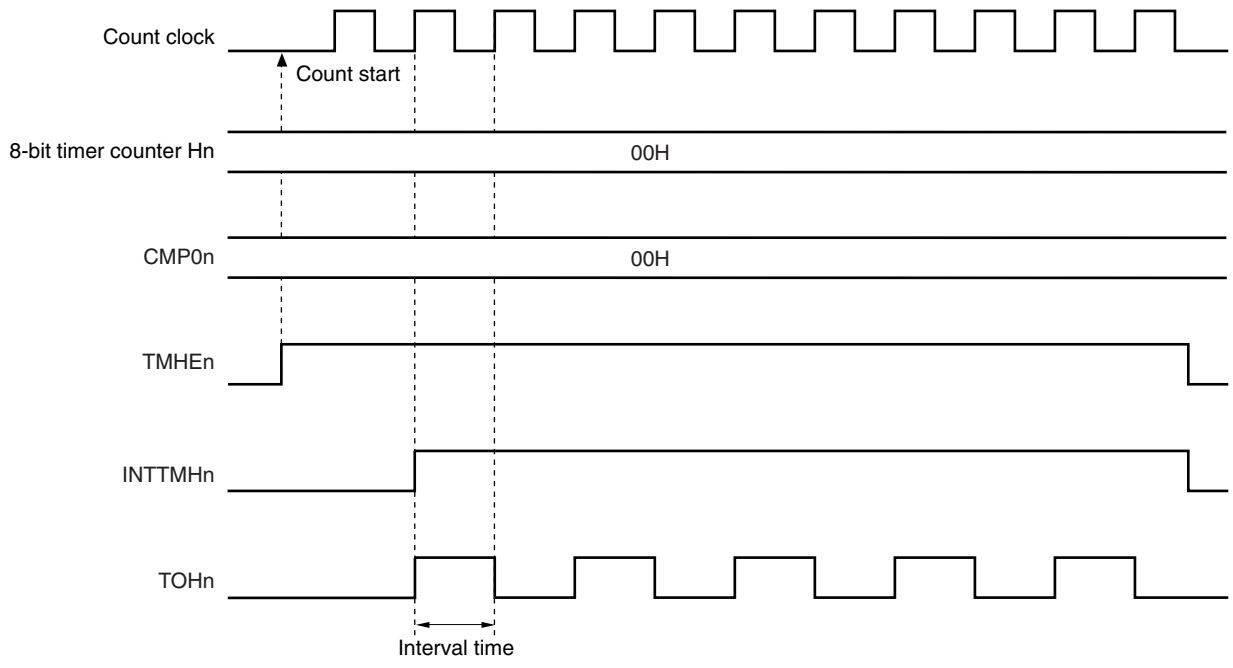
**2.** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-13. Timing of Interval Timer/Square-Wave Output Operation (2/2)**

**(b) Operation when CMP0n = FFH**




**(c) Operation when CMP0n = 00H**




**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

### 8.4.2 Operation as PWM output

In PWM output mode, a pulse with an arbitrary duty and arbitrary cycle can be output.

The 8-bit timer compare register 0n (CMP0n) controls the cycle of timer output (TOHn). Rewriting the CMP0n register during timer operation is prohibited.

The 8-bit timer compare register 1n (CMP1n) controls the duty of timer output (TOHn). Rewriting the CMP1n register during timer operation is possible.

The operation in PWM output mode is as follows.

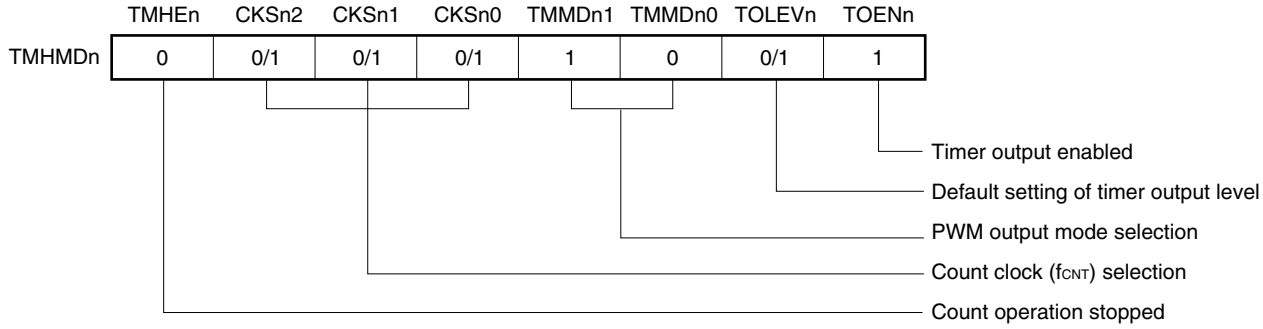
PWM output (TOHn output) outputs an active level and 8-bit timer counter Hn is cleared to 0 when 8-bit timer counter Hn and the CMP0n register match after the timer count is started. PWM output (TOHn output) outputs an inactive level when 8-bit timer counter Hn and the CMP1n register match.

Setting

<1> Set each register.

**Figure 8-14. Register Setting in PWM Output Mode**

**(i) Setting timer H mode register n (TMHMDn)**




**(ii) Setting CMP0n register**

- Compare value (N): Cycle setting

**(iii) Setting CMP1n register**

- Compare value (M): Duty setting

**Remarks 1.** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**2.** 00H ≤ CMP1n (M) < CMP0n (N) ≤ FFH

<2> The count operation starts when TMHEn = 1.

<3> The CMP0n register is the compare register that is to be compared first after counter operation is enabled. When the values of the 8-bit timer counter Hn and the CMP0n register match, the 8-bit timer counter Hn is cleared, an interrupt request signal (INTTMHn) is generated, and an active level is output. At the same time, the compare register to be compared with the 8-bit timer counter Hn is changed from the CMP0n register to the CMP1n register.

<4> When the 8-bit timer counter Hn and the CMP1n register match, an inactive level is output and the compare register to be compared with the 8-bit timer counter Hn is changed from the CMP1n register to the CMP0n register. At this time, the 8-bit timer counter Hn is not cleared and the INTTMHn signal is not generated.

<5> By performing procedures <3> and <4> repeatedly, a pulse with an arbitrary duty can be obtained.

<6> To stop the count operation, set TMHEn = 0.

If the setting value of the CMP0n register is N, the setting value of the CMP1n register is M, and the count clock frequency is $f_{CNT}$, the PWM pulse output cycle and duty are as follows.

- PWM pulse output cycle = (N + 1)/$f_{CNT}$
- Duty = (M + 1)/(N + 1)

**Cautions 1. The set value of the CMP1n register can be changed while the timer counter is operating. However, this takes a duration of three operating clocks (signal selected by the CKSn2 to CKSn0 bits of the TMHMDn register) from when the value of the CMP1n register is changed until the value is transferred to the register.**

**2. Be sure to set the CMP1n register when starting the timer count operation (TMHEn = 1) after the timer count operation was stopped (TMHEn = 0) (be sure to set again even if setting the same value to the CMP1n register).**

**3. Make sure that the CMP1n register setting value (M) and CMP0n register setting value (N) are within the following range.**

**00H ≤ CMP1n (M) < CMP0n (N) ≤ FFH**

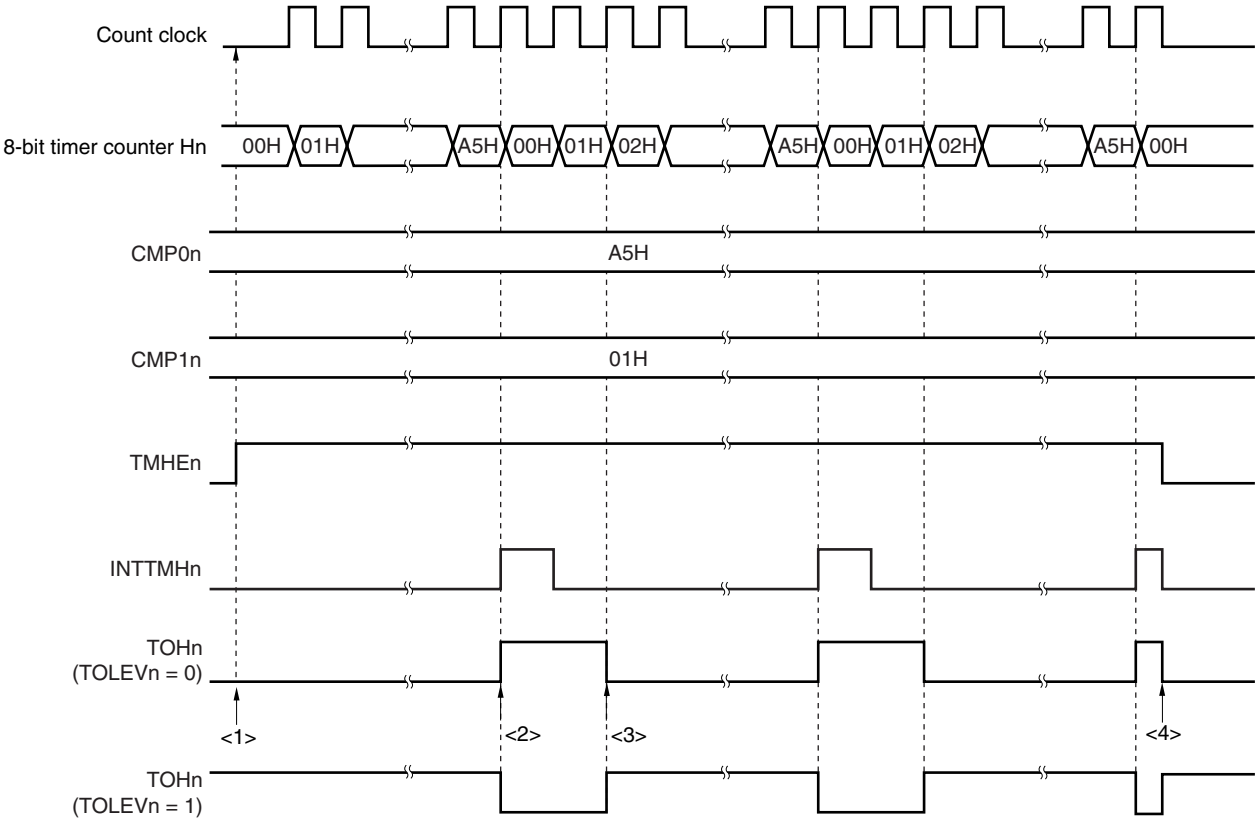
**Remarks 1.** For the setting of the output pin, refer to **8.3 (4) Port mode register 0 (PM0), port mode register 1 (PM1), port mode register 3 (PM3)**.

**2.** For details on how to enable the INTTMHn signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**3.** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-15. Operation Timing in PWM Output Mode (1/4)**

**(a) Basic operation**




<1> The count operation is enabled by setting the TMHEn bit to 1. Start the 8-bit timer counter Hn by masking one count clock to count up. At this time, PWM output outputs an inactive level.

<2> When the values of the 8-bit timer counter Hn and the CMP0n register match, an active level is output. At this time, the value of the 8-bit timer counter Hn is cleared, and the INTTMHn signal is output.

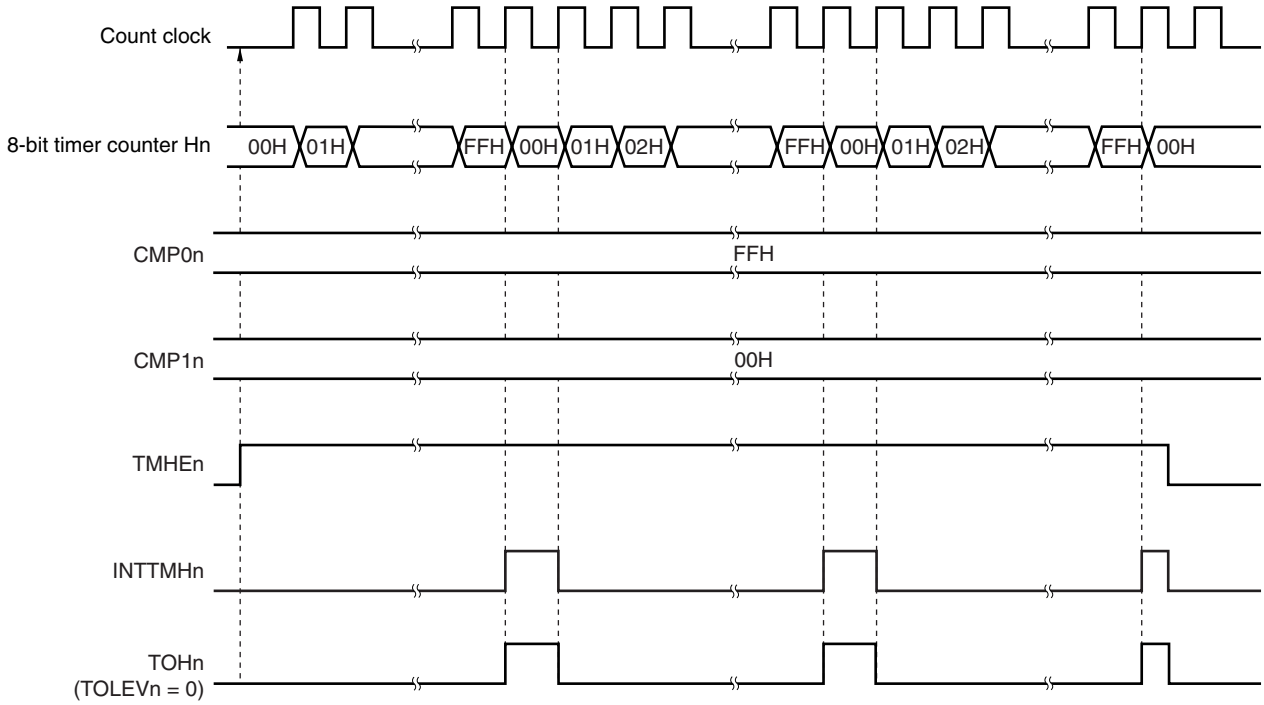
<3> When the values of the 8-bit timer counter Hn and the CMP1n register match, an inactive level is output. At this time, the 8-bit timer counter value is not cleared and the INTTMHn signal is not output.

<4> Clearing the TMHEn bit to 0 during timer Hn operation sets the INTTMHn signal to the default and PWM output to an inactive level.

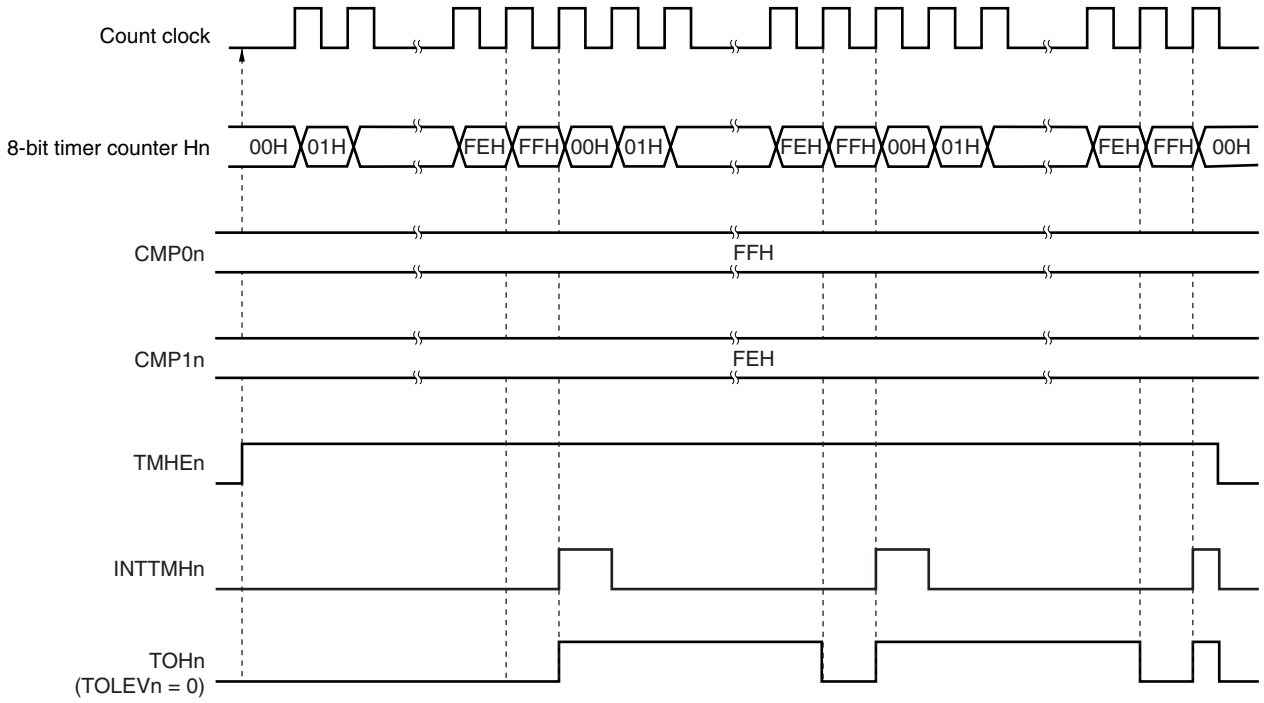
**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-15. Operation Timing in PWM Output Mode (2/4)**

**(b) Operation when CMP0n = FFH, CMP1n = 00H**



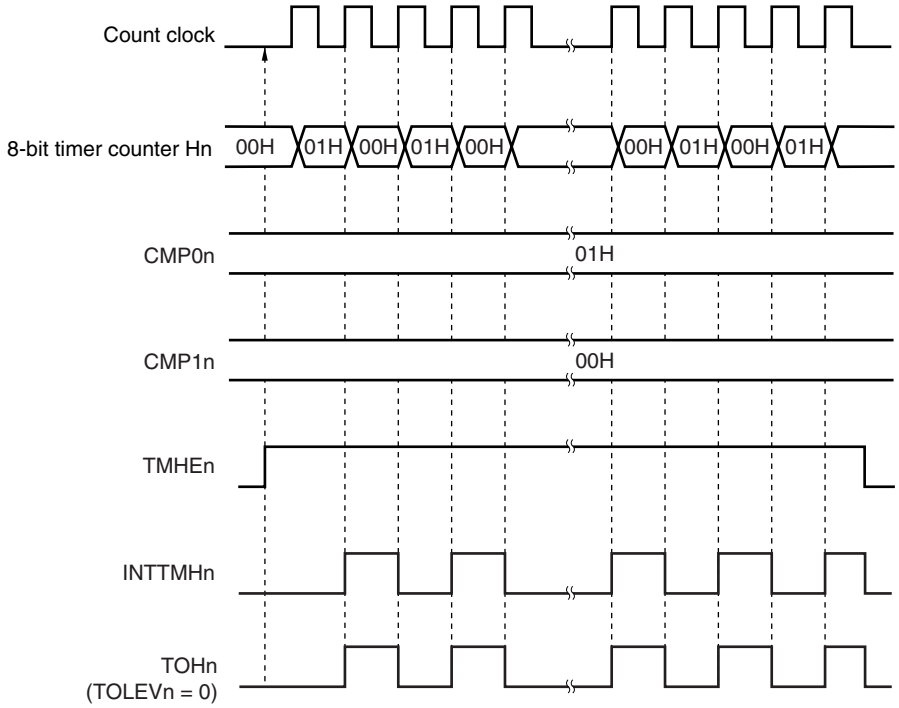

**(c) Operation when CMP0n = FFH, CMP1n = FEH**




**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-15. Operation Timing in PWM Output Mode (3/4)**

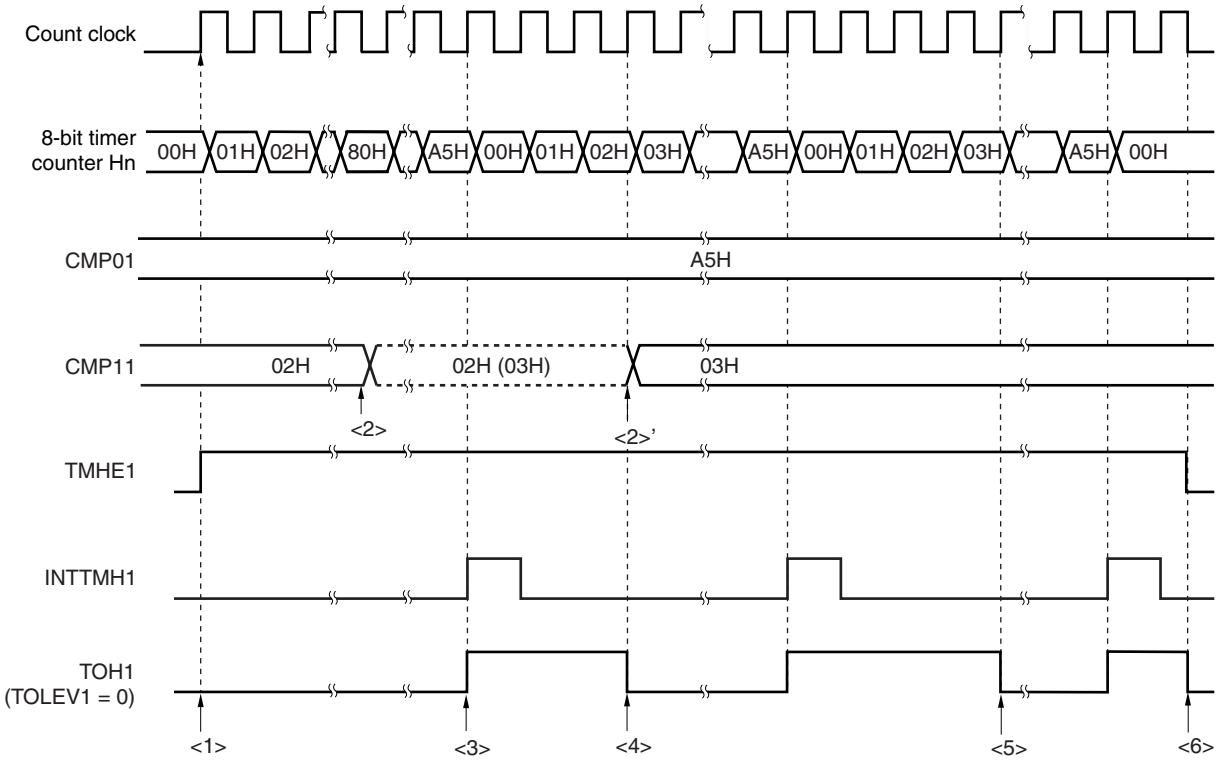
**(d) Operation when CMP0n = 01H, CMP1n = 00H**




**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

**Figure 8-15. Operation Timing in PWM Output Mode (4/4)**

**(e) Operation by changing CMP1n (CMP1n = 02H → 03H, CMP0n = A5H)**




<1> The count operation is enabled by setting TMHEn = 1. Start the 8-bit timer counter Hn by masking one count clock to count up. At this time, PWM output outputs an inactive level.

<2> The CMP1n register value can be changed during timer counter operation. This operation is asynchronous to the count clock.

<3> When the values of the 8-bit timer counter Hn and the CMP0n register match, the value of the 8-bit timer counter Hn is cleared, an active level is output, and the INTTMHn signal is output.

<4> If the CMP1n register value is changed, the value is latched and not transferred to the register. When the values of the 8-bit timer counter Hn and the CMP1n register before the change match, the value is transferred to the CMP1n register and the CMP1n register value is changed (<2>').

However, three count clocks or more are required from when the CMP1n register value is changed to when the value is transferred to the register. If a match signal is generated within three count clocks, the changed value cannot be transferred to the register.

<5> When the values of the 8-bit timer counter Hn and the CMP1n register after the change match, an inactive level is output. The 8-bit timer counter Hn is not cleared and the INTTMHn signal is not generated.

<6> Clearing the TMHEn bit to 0 during timer Hn operation sets the INTTMHn signal to the default and PWM output to an inactive level.

**Remark** 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

### 8.4.3 Carrier generator operation (8-bit timer H1 only)

In the carrier generator mode, the 8-bit timer H1 is used to generate the carrier signal of an infrared remote controller, and the 8-bit timer/event counter 51 is used to generate an infrared remote control signal (time count).

The carrier clock generated by the 8-bit timer H1 is output in the cycle set by the 8-bit timer/event counter 51.

In carrier generator mode, the output of the 8-bit timer H1 carrier pulse is controlled by the 8-bit timer/event counter 51, and the carrier pulse is output from the TOH1 output.

#### (1) Carrier generation

In carrier generator mode, the 8-bit timer H compare register 01 (CMP01) generates a low-level width carrier pulse waveform and the 8-bit timer H compare register 11 (CMP11) generates a high-level width carrier pulse waveform.

Rewriting the CMP11 register during the 8-bit timer H1 operation is possible but rewriting the CMP01 register is prohibited.

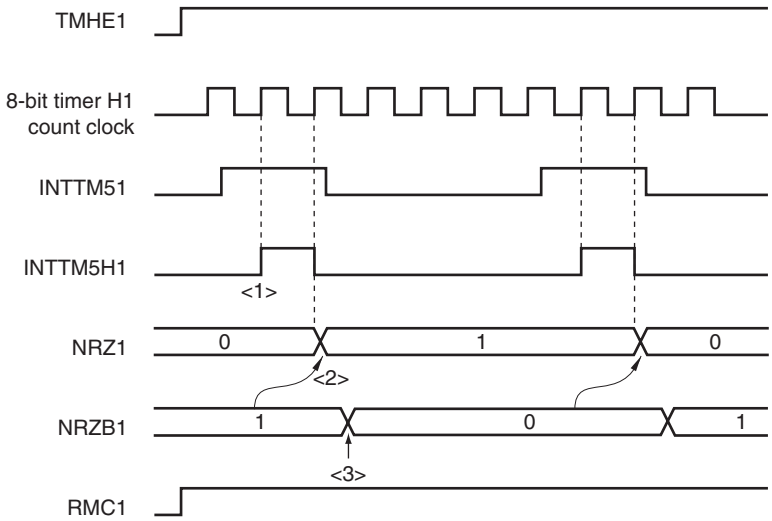
#### (2) Carrier output control

Carrier output is controlled by the interrupt request signal (INTTM51) of the 8-bit timer/event counter 51 and the NRZB1 and RMC1 bits of the 8-bit timer H carrier control register (TMCYC1). The relationship between the outputs is shown below.

| RMC1 Bit | NRZB1 Bit | Output |
|---|---|---|
| 0 | 0 | Low-level output |
| 0 | 1 | High-level output at rising edge of INTTM51 signal input |
| 1 | 0 | Low-level output |
| 1 | 1 | Carrier pulse output at rising edge of INTTM51 signal input |

To control the carrier pulse output during a count operation, the NRZ1 and NRZB1 bits of the TMCYC1 register have a master and slave bit configuration. The NRZ1 bit is read-only but the NRZB1 bit can be read and written. The INTTM51 signal is synchronized with the 8-bit timer H1 count clock and is output as the INTTM5H1 signal. The INTTM5H1 signal becomes the data transfer signal of the NRZ1 bit, and the NRZB1 bit value is transferred to the NRZ1 bit. The timing for transfer from the NRZB1 bit to the NRZ1 bit is as shown below.

**Figure 8-16. Transfer Timing**




<1> The INTTM51 signal is synchronized with the count clock of the 8-bit timer H1 and is output as the INTTM5H1 signal.

<2> The value of the NRZB1 bit is transferred to the NRZ1 bit at the second clock from the rising edge of the INTTM5H1 signal.

<3> Write the next value to the NRZB1 bit in the interrupt servicing program that has been started by the INTTM5H1 interrupt or after timing has been checked by polling the interrupt request flag. Write data to count the next time to the CR51 register.

**Cautions 1. Do not rewrite the NRZB1 bit again until at least the second clock after it has been rewritten, or else the transfer from the NRZB1 bit to the NRZ1 bit is not guaranteed.**

**2. When the 8-bit timer/event counter 51 is used in the carrier generator mode, an interrupt is generated at the timing of <1>. When the 8-bit timer/event counter 51 is used in a mode other than the carrier generator mode, the timing of the interrupt generation differs.**

**Remark** INTTM5H1 is an internal signal and not an interrupt source.

Setting

<1> Set each register.

**Figure 8-17. Register Setting in Carrier Generator Mode**

**(i) Setting 8-bit timer H mode register 1 (TMHMD1)**




**(ii) CMP01 register setting**

- Compare value

**(iii) CMP11 register setting**

- Compare value

**(iv) TMCYC1 register setting**

- RMC1 = 1 ... Remote control output enable bit
- NRZB1 = 0/1 ... carrier output enable bit

**(v) TCL51 and TMC51 register setting**

- Refer to **7.3 Registers Controlling 8-Bit Timer/Event Counters 50 and 51**.

<2> When TMHE1 = 1, the 8-bit timer H1 starts counting.

<3> When TCE51 of the 8-bit timer mode control register 51 (TMC51) is set to 1, the 8-bit timer/event counter 51 starts counting.

<4> After the count operation is enabled, the first compare register to be compared is the CMP01 register. When the count value of the 8-bit timer counter H1 and the CMP01 register value match, the INTTMH1 signal is generated, the 8-bit timer counter H1 is cleared. At the same time, the compare register to be compared with the 8-bit timer counter H1 is switched from the CMP01 register to the CMP11 register.

<5> When the count value of the 8-bit timer counter H1 and the CMP11 register value match, the INTTMH1 signal is generated, the 8-bit timer counter H1 is cleared. At the same time, the compare register to be compared with the 8-bit timer counter H1 is switched from the CMP11 register to the CMP01 register.

<6> By performing procedures <4> and <5> repeatedly, a carrier clock is generated.

<7> The INTTM51 signal is synchronized with count clock of the 8-bit timer H1 and output as the INTTM5H1 signal. The INTTM5H1 signal becomes the data transfer signal for the NRZB1 bit, and the NRZB1 bit value is transferred to the NRZ1 bit.

<8> Write the next value to the NRZB1 bit in the interrupt servicing program that has been started by the INTTM5H1 interrupt or after timing has been checked by polling the interrupt request flag. Write data to count the next time to the CR51 register.

<9> When the NRZ1 bit is high level, a carrier clock is output by TOH1 output.

<10> By performing the procedures above, an arbitrary carrier clock is obtained. To stop the count operation, clear TMHE1 to 0.

If the setting value of the CMP01 register is N, the setting value of the CMP11 register is M, and the count clock frequency is $f_{CNT}$, the carrier clock output cycle and duty are as follows.

- Carrier clock output cycle = (N + M + 2)/$f_{CNT}$
- Duty = High-level width/carrier clock output width = (M + 1)/(N + M + 2)

**Cautions 1. Be sure to set the CMP11 register when starting the timer count operation (TMHE1 = 1) after the timer count operation was stopped (TMHE1 = 0) (be sure to set again even if setting the same value to the CMP11 register).**

**2. Set so that the count clock frequency of TMH1 becomes more than 6 times the count clock frequency of TM51.**

**3. Set the values of the CMP01 and CMP11 registers in a range of 01H to FFH.**

**4. The set value of the CMP11 register can be changed while the timer counter is operating. However, it takes the duration of three operating clocks (signal selected by the CKS12 to CKS10 bits of the TMHMD1 register) since the value of the CMP11 register has been changed until the value is transferred to the register.**

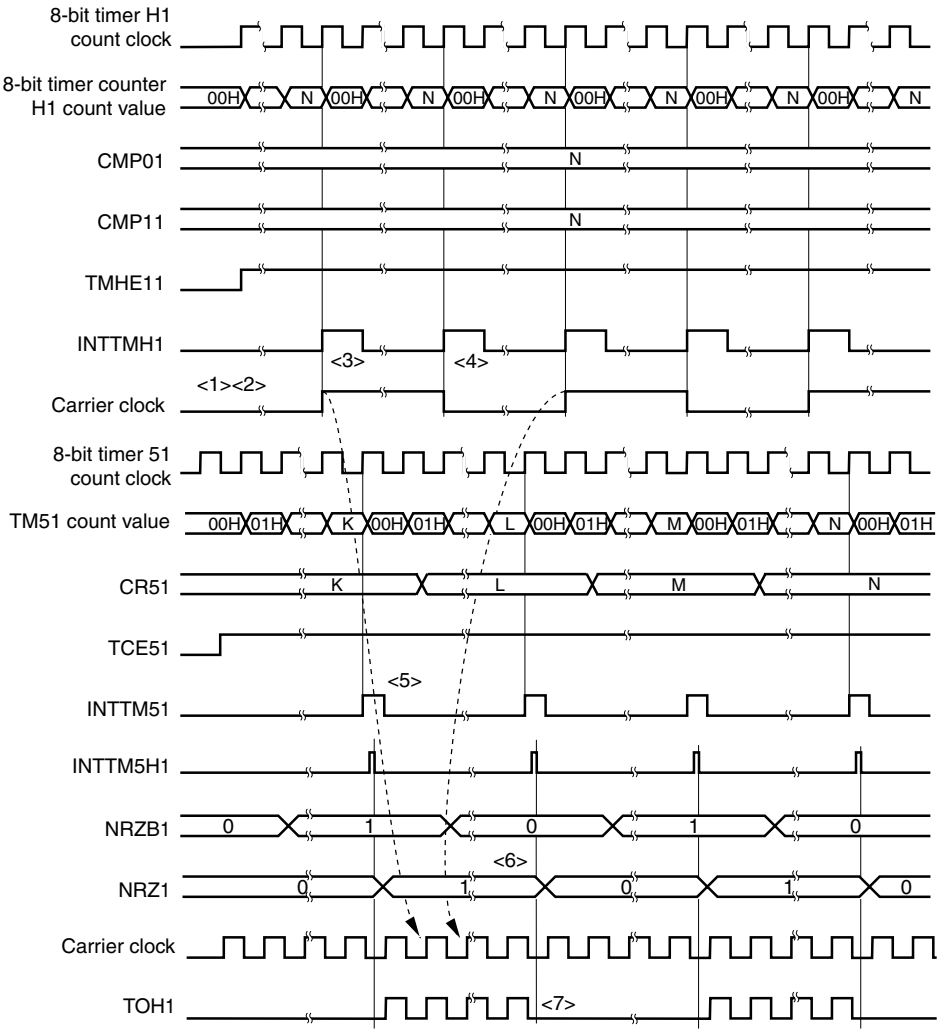
**5. Be sure to set the RMC1 bit before the count operation is started.**

**Remarks 1.** For the setting of the output pin, refer to **8.3 (4) Port mode register 0 (PM0), port mode register 1 (PM1), port mode register 3 (PM3)**.

**2.** For how to enable the INTTMH1 signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

**Figure 8-18. Carrier Generator Mode Operation Timing (1/3)**

**(a) Operation when CMP01 = N, CMP11 = N**




<1> When TMHE1 = 0 and TCE51 = 0, the 8-bit timer counter H1 operation is stopped.

<2> When TMHE1 = 1 is set, the 8-bit timer counter H1 starts a count operation. At that time, the carrier clock remains default.

<3> When the count value of the 8-bit timer counter H1 matches the CMP01 register value, the first INTTMH1 signal is generated, the carrier clock signal is inverted, and the compare register to be compared with the 8-bit timer counter H1 is switched from the CMP01 register to the CMP11 register. The 8-bit timer counter H1 is cleared to 00H.

<4> When the count value of the 8-bit timer counter H1 matches the CMP11 register value, the INTTMH1 signal is generated, the carrier clock signal is inverted, and the compare register to be compared with the 8-bit timer counter H1 is switched from the CMP11 register to the CMP01 register. The 8-bit timer counter H1 is cleared to 00H. By performing procedures <3> and <4> repeatedly, a carrier clock with duty fixed to 50% is generated.

<5> When the INTTM51 signal is generated, it is synchronized with the 8-bit timer H1 count clock and is output as the INTTM5H1 signal.

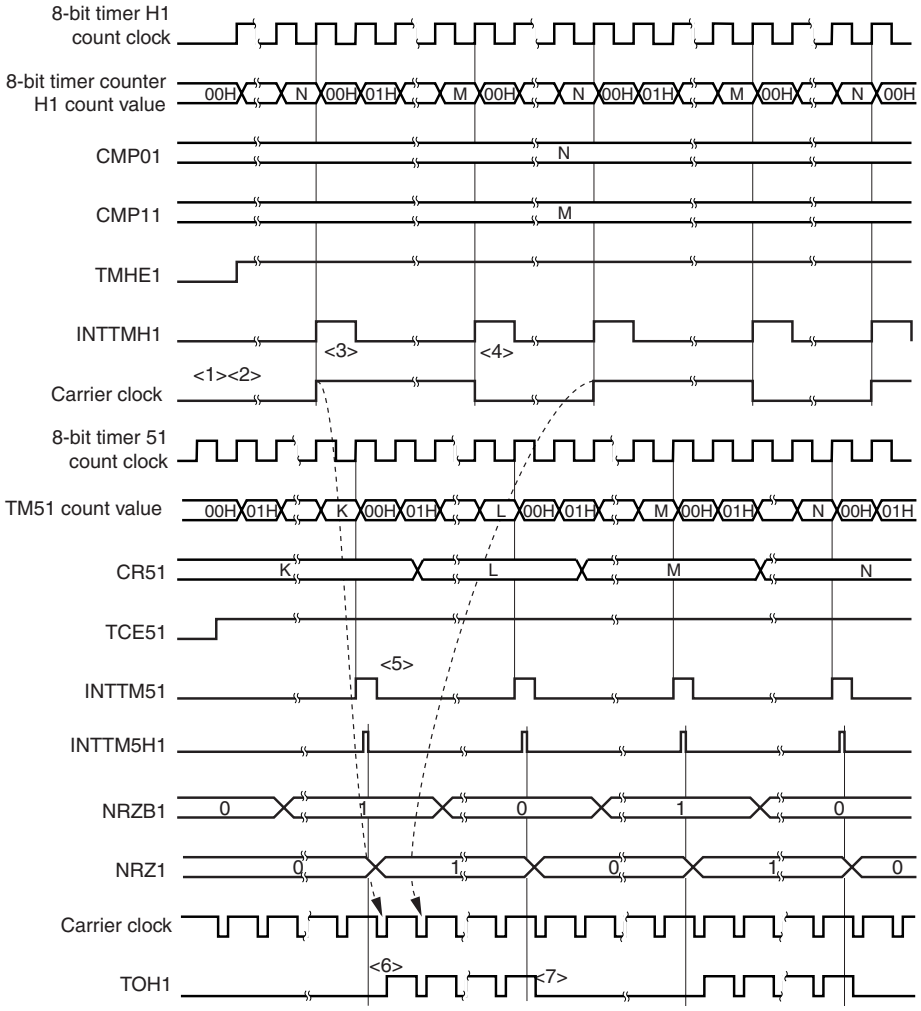
<6> The INTTM5H1 signal becomes the data transfer signal for the NRZB1 bit, and the NRZB1 bit value is transferred to the NRZ1 bit.

<7> When NRZ1 = 0 is set, the TOH1 output becomes low level.

**Remark** INTTM5H1 is an internal signal and not an interrupt source.

**Figure 8-18. Carrier Generator Mode Operation Timing (2/3)**

**(b) Operation when CMP01 = N, CMP11 = M**




<1> When TMHE1 = 0 and TCE51 = 0, the 8-bit timer counter H1 operation is stopped.

<2> When TMHE1 = 1 is set, the 8-bit timer counter H1 starts a count operation. At that time, the carrier clock remains default.

<3> When the count value of the 8-bit timer counter H1 matches the CMP01 register value, the first INTTMH1 signal is generated, the carrier clock signal is inverted, and the compare register to be compared with the 8-bit timer counter H1 is switched from the CMP01 register to the CMP11 register. The 8-bit timer counter H1 is cleared to 00H.

<4> When the count value of the 8-bit timer counter H1 matches the CMP11 register value, the INTTMH1 signal is generated, the carrier clock signal is inverted, and the compare register to be compared with the 8-bit timer counter H1 is switched from the CMP11 register to the CMP01 register. The 8-bit timer counter H1 is cleared to 00H. By performing procedures <3> and <4> repeatedly, a carrier clock with duty fixed to other than 50% is generated.

<5> When the INTTM51 signal is generated, it is synchronized with the 8-bit timer H1 count clock and is output as the INTTM5H1 signal.

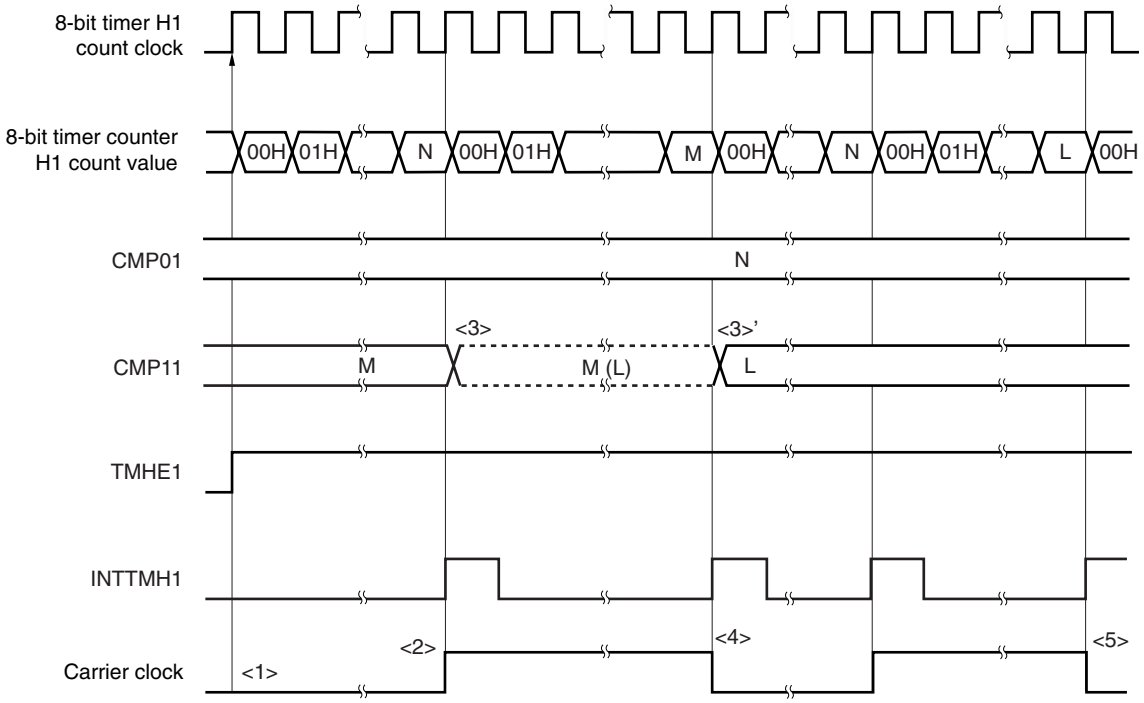
<6> A carrier signal is output at the first rising edge of the carrier clock if NRZ1 is set to 1.

<7> When NRZ1 = 0, the TOH1 output is held at the high level and is not changed to low level while the carrier clock is high level (from <6> and <7>, the high-level width of the carrier clock waveform is guaranteed).

**Remark** INTTM5H1 is an internal signal and not an interrupt source.

**Figure 8-18. Carrier Generator Mode Operation Timing (3/3)**

**(c) Operation when CMP11 is changed**




<1> When TMHE1 = 1 is set, the 8-bit timer H1 starts a count operation. At that time, the carrier clock remains default.

<2> When the count value of the 8-bit timer counter H1 matches the value of the CMP01 register, the INTTMH1 signal is output, the carrier signal is inverted, and the timer counter is cleared to 00H. At the same time, the compare register whose value is to be compared with that of the 8-bit timer counter H1 is changed from the CMP01 register to the CMP11 register.

<3> The CMP11 register is asynchronous to the count clock, and its value can be changed while the 8-bit timer H1 is operating. The new value (L) to which the value of the register is to be changed is latched. When the count value of the 8-bit timer counter H1 matches the value (M) of the CMP11 register before the change, the CMP11 register is changed (<3>').

However, it takes three count clocks or more since the value of the CMP11 register has been changed until the value is transferred to the register. Even if a match signal is generated before the duration of three count clocks elapses, the new value is not transferred to the register.

<4> When the count value of 8-bit timer counter H1 matches the value (M) of the CMP11 register before the change, the INTTMH1 signal is output, the carrier signal is inverted, and the timer counter is cleared to 00H. At the same time, the compare register whose value is to be compared with that of the 8-bit timer counter H1 is changed from the CMP11 register to the CMP01 register.

<5> The timing at which the count value of the 8-bit timer counter H1 and the CMP11 register value match again is indicated by the value after the change (L).

# CHAPTER 9 WATCHDOG TIMER

## 9.1 Functions of Watchdog Timer

The watchdog timer is mounted onto all 78K0/Kx2-L microcontroller products.

The watchdog timer operates on the internal low-speed oscillation clock.

The watchdog timer is used to detect an inadvertent program loop. If a program loop is detected, an internal reset signal is generated.

Program loop is detected in the following cases.

- If the watchdog timer counter overflows
- If a 1-bit manipulation instruction is executed on the watchdog timer enable register (WDTE)
- If data other than “ACH” is written to WDTE
- If data is written to WDTE during a window close period
- If the instruction is fetched from an area not set by the IMS register (detection of an invalid check while the CPU hangs up)
- If the CPU accesses an area that is not set by the IMS register (excluding FB00H to FFFFH) by executing a read/write instruction (detection of an abnormal access during a CPU program loop)

When a reset occurs due to the watchdog timer, bit 4 (WDTRF) of the reset control flag register (RESF) is set to 1. For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

## 9.2 Configuration of Watchdog Timer

The watchdog timer includes the following hardware.

**Table 9-1. Configuration of Watchdog Timer**

| Item | Configuration |
|---|---|
| Control register | Watchdog timer enable register (WDTE) |

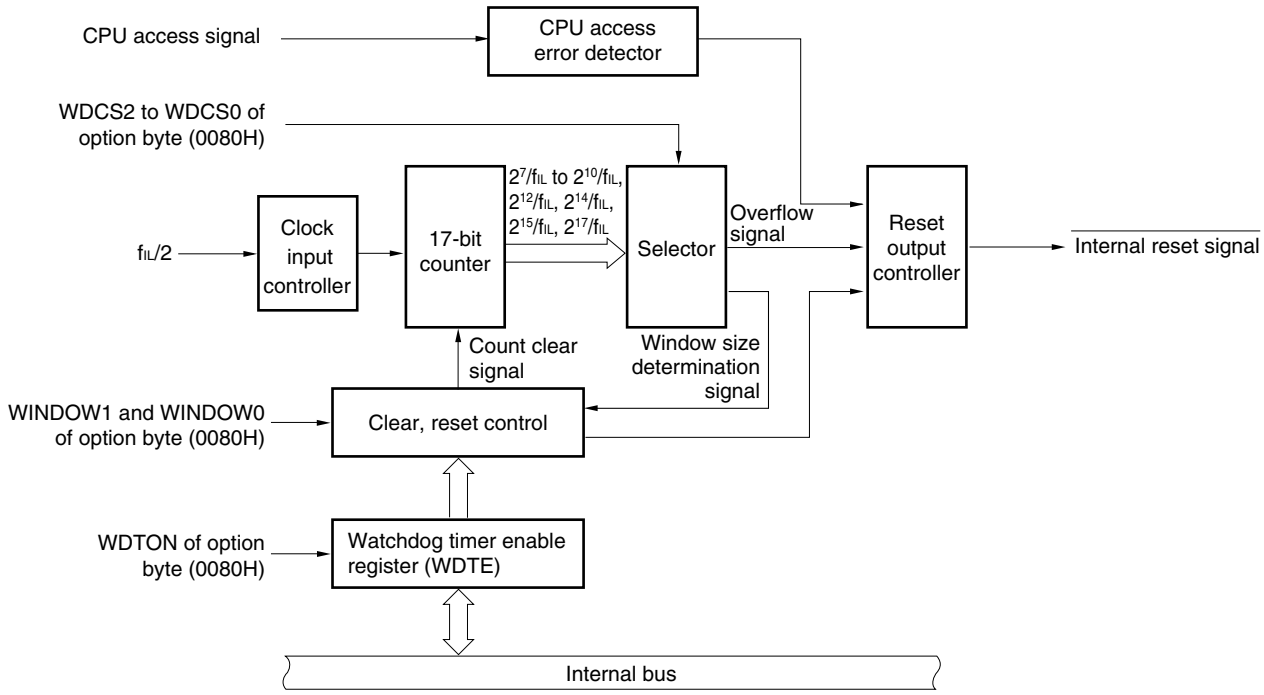
How the counter operation is controlled, overflow time, and window open period are set by the option byte.

**Table 9-2. Setting of Option Bytes and Watchdog Timer**

| Setting of Watchdog Timer | Option Byte (0080H) |
|---|---|
| Window open period | Bits 6 and 5 (WINDOW1, WINDOW0) |
| Controlling counter operation of watchdog timer | Bit 4 (WDTON) |
| Overflow time of watchdog timer | Bits 3 to 1 (WDCS2 to WDCS0) |

**Remark** For the option byte, refer to **CHAPTER 24 OPTION BYTE**.

**Figure 9-1. Block Diagram of Watchdog Timer**



## 9.3 Register Controlling Watchdog Timer

The watchdog timer is controlled by the watchdog timer enable register (WDTE).

**(1) Watchdog timer enable register (WDTE)**

Writing ACH to WDTE clears the watchdog timer counter and starts counting again.

This register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to 9AH or 1AH[Note].

**Figure 9-2. Format of Watchdog Timer Enable Register (WDTE)**

Address: FF99H After reset: 9AH/1AH[Note] R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| WDTE | | | | | | | | |

**Note** The WDTE reset value differs depending on the WDTON setting value of the option byte (0080H). To operate watchdog timer, set WDTON to 1.

| WDTON Setting Value | WDTE Reset Value |
|---|---|
| 0 (watchdog timer count operation disabled) | 1AH |
| 1 (watchdog timer count operation enabled) | 9AH |

**Cautions 1. If a value other than ACH is written to WDTE, an internal reset signal is generated. If the source clock to the watchdog timer is stopped, however, an internal reset signal is generated when the source clock to the watchdog timer resumes operation.**

**2. If a 1-bit memory manipulation instruction is executed for WDTE, an internal reset signal is generated. If the source clock to the watchdog timer is stopped, however, an internal reset signal is generated when the source clock to the watchdog timer resumes operation.**

**3. The value read from WDTE is 9AH/1AH (this differs from the written value (ACH)).**

## 9.4 Operation of Watchdog Timer

### 9.4.1 Controlling operation of watchdog timer

1. When the watchdog timer is used, its operation is specified by the option byte (0080H).

   - Enable counting operation of the watchdog timer by setting bit 4 (WDTON) of the option byte (0080H) to 1 (the counter starts operating after a reset release) (for details, refer to **CHAPTER 24**).

| WDTON | Operation Control of Watchdog Timer Counter/Illegal Access Detection |
|---|---|
| 0 | Counter operation disabled (counting stopped after reset), illegal access detection operation disabled |
| 1 | Counter operation enabled (counting started after reset), illegal access detection operation enabled |

   - Set an overflow time by using bits 3 to 1 (WDCS2 to WDCS0) of the option byte (0080H) (for details, refer to **9.4.2** and **CHAPTER 24**).
   - Set a window open period by using bits 6 and 5 (WINDOW1 and WINDOW0) of the option byte (0080H) (for details, refer to **9.4.3** and **CHAPTER 24**).

2. After a reset release, the watchdog timer starts counting.
3. By writing “ACH” to WDTE after the watchdog timer starts counting and before the overflow time set by the option byte, the watchdog timer is cleared and starts counting again.
4. After that, write WDTE the second time or later after a reset release during the window open period. If WDTE is written during a window close period, an internal reset signal is generated.
5. If the overflow time expires without “ACH” written to WDTE, an internal reset signal is generated.
   A internal reset signal is generated in the following cases.

   - If a 1-bit manipulation instruction is executed on the watchdog timer enable register (WDTE)
   - If data other than “ACH” is written to WDTE
   - If the instruction is fetched from an area not set by the IMS register (detection of an invalid check during a CPU program loop)
   - If the CPU accesses an area not set by the IMS register (excluding FB00H to FFFFH) by executing a read/write instruction (detection of an abnormal access during a CPU program loop)

**Cautions 1. The first writing to WDTE after a reset release clears the watchdog timer, if it is made before the overflow time regardless of the timing of the writing, and the watchdog timer starts counting again.**

**2. If the watchdog timer is cleared by writing “ACH” to WDTE, the actual overflow time may be different from the overflow time set by the option byte by up to 2/$f_{IL}$ seconds.**

**3. The watchdog timer can be cleared immediately before the count value overflows (FFFFH).**

**Cautions 4. The operation of the watchdog timer in the HALT and STOP modes differs as follows depending on the set value of bit 0 (LSROSC) of the option byte.**

| | LSROSC = 0 (Internal Low-Speed Oscillator Can Be Stopped by Software) | LSROSC = 1 (Internal Low-Speed Oscillator Cannot Be Stopped) |
|---|---|---|
| In HALT mode | Watchdog timer operation stops. | Watchdog timer operation continues. |
| In STOP mode | | |

**If LSROSC = 0, the watchdog timer resumes counting after the HALT or STOP mode is released. At this time, the counter is not cleared to 0 but starts counting from the value at which it was stopped.**

**If oscillation of the internal low-speed oscillator is stopped by setting LSRSTOP (bit 1 of the internal oscillation mode register (RCM) = 1) when LSROSC = 0, the watchdog timer stops operating. At this time, the counter is not cleared to 0.**

**5. The watchdog timer continues its operation during self-programming and EEPROM emulation of the flash memory. During processing, the interrupt acknowledge time is delayed. Set the overflow time and window size taking this delay into consideration.**

### 9.4.2 Setting overflow time of watchdog timer

Set the overflow time of the watchdog timer by using bits 3 to 1 (WDCS2 to WDCS0) of the option byte (0080H).

If an overflow occurs, an internal reset signal is generated. The present count is cleared and the watchdog timer starts counting again by writing “ACH” to WDTE during the window open period before the overflow time.

The following overflow time is set.

**Table 9-3. Setting of Overflow Time of Watchdog Timer**

| WDCS2 | WDCS1 | WDCS0 | Overflow Time of Watchdog Timer |
|---|---|---|---|
| 0 | 0 | 0 | $2^7/f_{IL}$ (3.88 ms) |
| 0 | 0 | 1 | $2^8/f_{IL}$ (7.76 ms) |
| 0 | 1 | 0 | $2^9/f_{IL}$ (15.52 ms) |
| 0 | 1 | 1 | $2^{10}/f_{IL}$ (31.03 ms) |
| 1 | 0 | 0 | $2^{12}/f_{IL}$ (124.12 ms) |
| 1 | 0 | 1 | $2^{14}/f_{IL}$ (496.48 ms) |
| 1 | 1 | 0 | $2^{15}/f_{IL}$ (992.97 ms) |
| 1 | 1 | 1 | $2^{17}/f_{IL}$ (3.97 s) |

**Cautions 1. The combination of WDCS2 = WDCS1 = WDCS0 = 0 and WINDOW1 = WINDOW0 = 0 is prohibited.**

**2. The watchdog timer continues its operation during self-programming and EEPROM emulation of the flash memory. During processing, the interrupt acknowledge time is delayed. Set the overflow time and window size taking this delay into consideration.**

**Remarks 1.** $f_{IL}$: Internal low-speed oscillation clock frequency

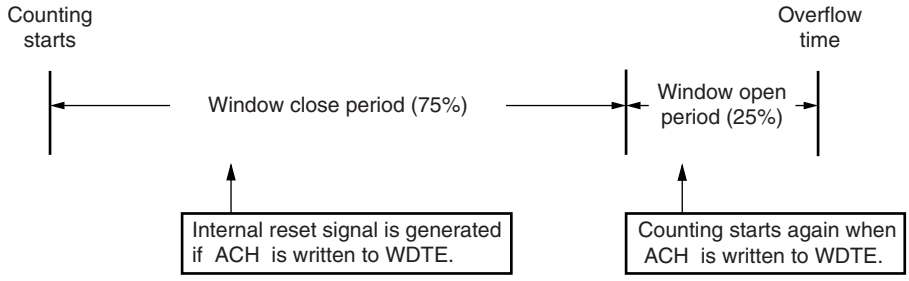
**2.** ( ): $f_{IL}$ = 33 kHz (MAX.)

### 9.4.3 Setting window open period of watchdog timer

Set the window open period of the watchdog timer by using bits 6 and 5 (WINDOW1, WINDOW0) of the option byte (0080H). The outline of the window is as follows.

- If “ACH” is written to WDTE during the window open period, the watchdog timer is cleared and starts counting again.
- Even if “ACH” is written to WDTE during the window close period, an abnormality is detected and an internal reset signal is generated.

**Example**: If the window open period is 25%




**Caution The first writing to WDTE after a reset release clears the watchdog timer, if it is made before the overflow time regardless of the timing of the writing, and the watchdog timer starts counting again.**

The window open period to be set is as follows.

**Table 9-4. Setting Window Open Period of Watchdog Timer**

| WINDOW1 | WINDOW0 | Window Open Period of Watchdog Timer |
|---|---|---|
| 0 | 0 | 25% |
| 0 | 1 | 50% |
| 1 | 0 | 75% |
| 1 | 1 | 100% |

**Cautions 1. The combination of WDCS2 = WDCS1 = WDCS0 = 0 and WINDOW1 = WINDOW0 = 0 is prohibited.**

**2. The watchdog timer continues its operation during self-programming and EEPROM emulation of the flash memory. During processing, the interrupt acknowledge time is delayed. Set the overflow time and window size taking this delay into consideration.**

**Remark** If the overflow time is set to $2^{17}/f_{IL}$, the window close time and open time are as follows.

| | Setting of Window Open Period | | | |
|---|---|---|---|---|
| | 25% | 50% | 75% | 100% |
| Window close time | 0 to 3.64 s | 0 to 2.43 s | 0 to 1.21 s | None |
| Window open time | 3.64 to 3.97 s | 2.43 to 3.97 s | 1.21 to 3.97 s | 0 to 3.97 s |

<When window open period is 25%>

- Overflow time:
  $2^{17}/f_{IL}$ (MAX.) = $2^{17}$/33 kHz (MAX.) = 3.97 s
- Window close time:
  0 to $2^{17}/f_{IL}$ (MIN.) × (1 − 0.25) = 0 to $2^{17}$/27 kHz (MIN.) × 0.75 = 0 to 3.64 s
- Window open time:
  $2^{17}/f_{IL}$ (MIN.) × (1 − 0.25) to $2^{17}/f_{IL}$ (MAX.) = $2^{17}/f_{IL}$ /27 kHz (MIN.) × 0.75 to $2^{17}$/33 kHz (MAX.) = 3.64 to 3.97 s

# CHAPTER 10 REAL-TIME COUNTER

<R>

| Item | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | | |
|---|---|---|---|---|---|---|
| | 16 Pins | 20, 25, 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| Real-time counter | – | | | √ (RTC output: none) | √ (RTC output: 2) | |

**Remark** √: Mounted, –: Not mounted

## 10.1 Functions of Real-Time Counter

The real-time counter has the following features.

- Having counters of year, month, week, day, hour, minute, and second, and can count up to 99 years.
- Constant-period interrupt function (period: 1 month to 0.5 seconds)
- Alarm interrupt function (alarm: week, hour, minute)
- Interval interrupt function
- Pin output function of 1 Hz [Note]
- Pin output function of 512 Hz or 16.384 kHz or 32.768 kHz

**Note** 78K0/KC2-L (44-pin and 48-pin products) only

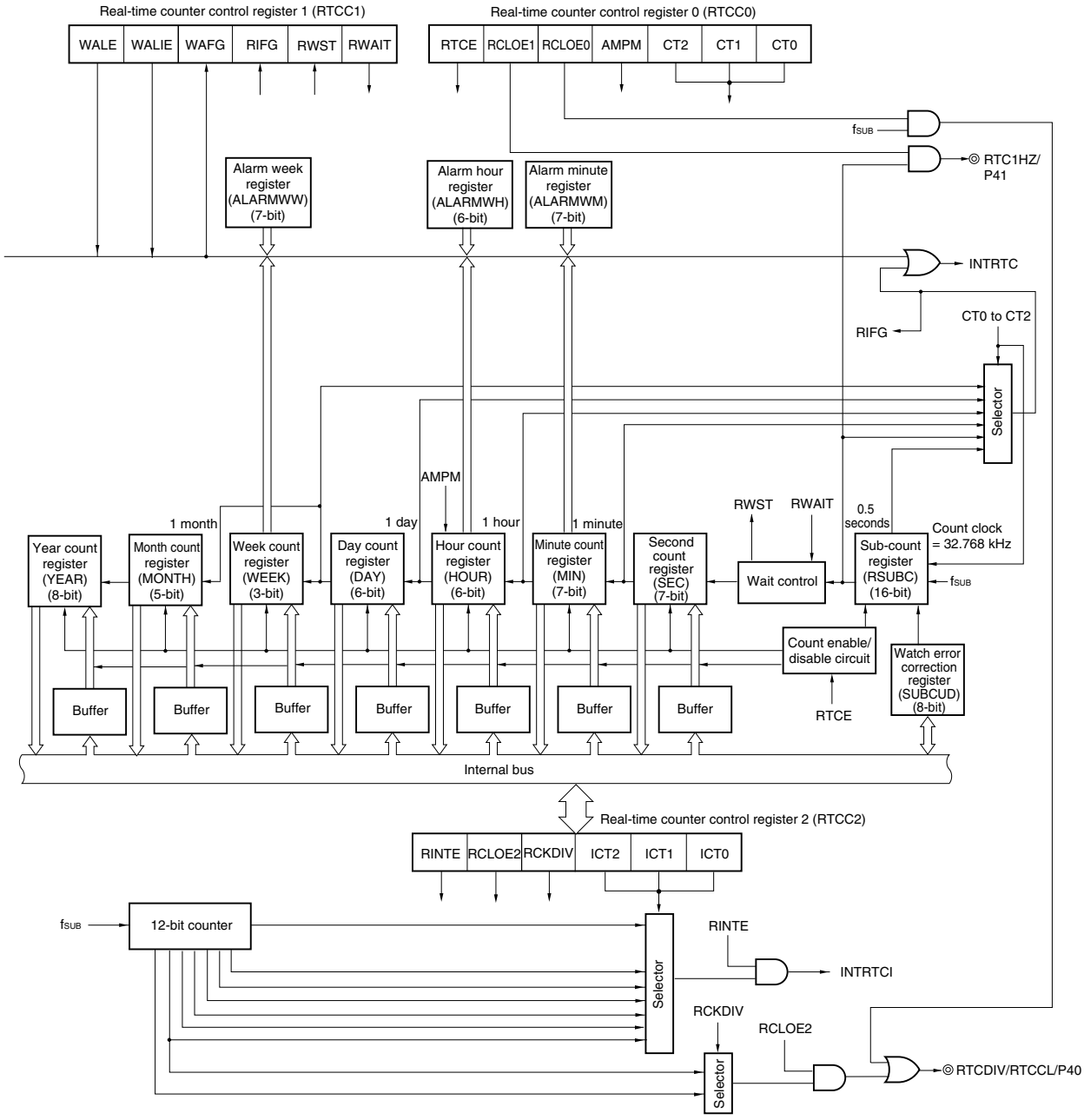
## 10.2 Configuration of Real-Time Counter

The real-time counter includes the following hardware.

**Table 10-1. Configuration of Real-Time Counter**

| Item | Configuration |
|---|---|
| Control registers | Peripheral enable register 0 (PER0)<br>Real-time counter control register 0 (RTCC0)<br>Real-time counter control register 1 (RTCC1)<br>Real-time counter control register 2 (RTCC2)<br>Sub-count register (RSUBC)<br>Second count register (SEC)<br>Minute count register (MIN)<br>Hour count register (HOUR)<br>Day count register (DAY)<br>Week count register (WEEK)<br>Month count register (MONTH)<br>Year count register (YEAR)<br>Watch error correction register (SUBCUD)<br>Alarm minute register (ALARMWM)<br>Alarm hour register (ALARMWH)<br>Alarm week register (ALARMWW)<br>Port mode register 4 (PM4)<br>Port register 4 (P4) |

**Figure 10-1. Block Diagram of Real-Time Counter**



## 10.3 Registers Controlling Real-Time Counter

The real-time counter is controlled by the following 18 registers.

- Peripheral enable register 0 (PER0)
- Real-time counter control register 0 (RTCC0)
- Real-time counter control register 1 (RTCC1)
- Real-time counter control register 2 (RTCC2)
- Sub-count register (RSUBC)
- Second count register (SEC)
- Minute count register (MIN)
- Hour count register (HOUR)
- Day count register (DAY)
- Week count register (WEEK)
- Month count register (MONTH)
- Year count register (YEAR)
- Watch error correction register (SUBCUD)
- Alarm minute register (ALARMWM)
- Alarm hour register (ALARMWH)
- Alarm week register (ALARMWW)
- Port mode register 4 (PM4)
- Port register 4 (P4)

### (1) Peripheral enable register 0 (PER0)

<R> This register controls the clock supplied to peripheral functions other than the real-time counter. By stopping the clock supplied to such peripheral functions, the power consumption can be reduced.

PER0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-2. Format of Peripheral Enable Register 0 (PER0)**

Address: FF25H After reset: 00H R/W

<R>

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PER0 | RTCEN | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| RTCEN | Control of real-time counter (RTC) input clock supply |
|---|---|
| 0 | Sub HALT low power consumption mode |
| 1 | Sub HALT normal mode[Note] |

**Note** To output the subsystem clock by using the PCL function while in the subsystem clock HALT mode, set RTCEN to 1.

**Caution Be sure to clear bits 0 to 6 of PER0 to "0".**

### (2) Real-time counter control register 0 (RTCC0)

The RTCC0 register is an 8-bit register that is used to start or stop the real-time counter operation, control the RTCCL and RTC1HZ pins, and set a 12- or 24-hour system and the constant-period interrupt function.

RTCC0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-3. Format of Real-Time Counter Control Register 0 (RTCC0)**

Address: FF9DH After reset: 00H R/W

| Symbol | <7> | 6 | <5> | <4> | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RTCC0 | RTCE | 0 | RCLOE1 | RCLOE0 | AMPM | CT2 | CT1 | CT0 |

| RTCE | Real-time counter operation control |
|---|---|
| 0 | Stops counter operation. |
| 1 | Starts counter operation. |

| RCLOE1 Note 1 | RTC1HZ pin output control |
|---|---|
| 0 | Disables output of RTC1HZ pin (1 Hz). |
| 1 | Enables output of RTC1HZ pin (1 Hz). |

| RCLOE0 Notes 1, 2 | RTCCL pin output control |
|---|---|
| 0 | Disables output of RTCCL pin (32.768 kHz). |
| 1 | Enables output of RTCCL pin (32.768 kHz). |

| AMPM | Selection of 12-/24-hour system |
|---|---|
| 0 | 12-hour system (a.m. and p.m. are displayed.) |
| 1 | 24-hour system |
| Rewrite the AMPM value after setting RWAIT (bit 0 of RTCC1) to 1. If the AMPM value is changed, the values of the hour count register (HOUR) change according to the specified time system.<br>Table 10-2 shows the displayed time digits. | |

| CT2 | CT1 | CT0 | Constant-period interrupt (INTRTC) selection |
|---|---|---|---|
| 0 | 0 | 0 | Does not use constant-period interrupt function. |
| 0 | 0 | 1 | Once per 0.5 s (synchronized with second count up) |
| 0 | 1 | 0 | Once per 1 s (same time as second count up) |
| 0 | 1 | 1 | Once per 1 m (second 00 of every minute) |
| 1 | 0 | 0 | Once per 1 hour (minute 00 and second 00 of every hour) |
| 1 | 0 | 1 | Once per 1 day (hour 00, minute 00, and second 00 of every day) |
| 1 | 1 | × | Once per 1 month (Day 1, hour 00 a.m., minute 00, and second 00 of every month) |
| When changing the values of CT2 to CT0 while the counter operates (RTCE = 1), rewrite the values of CT2 to CT0 after disabling interrupt servicing INTRTC by using the interrupt mask flag register. Furthermore, after rewriting the values of CT2 to CT0, enable interrupt servicing after clearing the RIFG and RTCIF flags. | | | |

**Notes 1.** 78K0/KC2-L (44-pin and 48-pin products) only

**2.** RCLOE0 and RCLOE2 must not be enabled at the same time.

**Caution If RCLOE0 and RCLOE1 are changed when RTCE = 1, a glitch may be generated on the 32.768 kHz and 1 Hz output signals.**

**Remark** ×: don't care

### (3) Real-time counter control register 1 (RTCC1)

The RTCC1 register is an 8-bit register that is used to control the alarm interrupt function and the wait time of the counter.

RTCC1 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-4. Format of Real-Time Counter Control Register 1 (RTCC1) (1/2)**

Address: FF9EH After reset: 00H R/W

| Symbol | <7> | <6> | 5 | <4> | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| RTCC1 | WALE | WALIE | 0 | WAFG | RIFG | 0 | RWST | RWAIT |

| WALE | Alarm operation control |
|---|---|
| 0 | Match operation is invalid. |
| 1 | Match operation is valid. |
| When setting a value to the WALE bit while the counter operates (RTCE = 1) and WALIE = 1, rewrite the WALE bit after disabling interrupt servicing INTRTC by using the interrupt mask flag register. Furthermore, clear the WAFG and RTCIF flags after rewriting the WALE bit. When setting each alarm register (WALIE flag of RTCC1, the ALARMWM register, the ALARMWH register, and the ALARMWW register), set match operation to be invalid (“0”) for the WALE bit. | |

| WALIE | Control of alarm interrupt (INTRTC) function operation |
|---|---|
| 0 | Does not generate interrupt on matching of alarm. |
| 1 | Generates interrupt on matching of alarm. |

| WAFG | Alarm detection status flag |
|---|---|
| 0 | Alarm mismatch |
| 1 | Detection of matching of alarm |
| This is a status flag that indicates detection of matching with the alarm. It is valid only when WALE = 1 and is set to “1” one clock (32.768 kHz) after matching of the alarm is detected. This flag is cleared when “0” is written to it. Writing “1” to it is invalid. | |

**Figure 10-4. Format of Real-Time Counter Control Register 1 (RTCC1) (2/2)**

| RIFG | Constant-period interrupt status flag |
|---|---|
| 0 | Constant-period interrupt is not generated. |
| 1 | Constant-period interrupt is generated. |
| This flag indicates the status of generation of the constant-period interrupt. When the constant-period interrupt is generated, it is set to “1”.<br>This flag is cleared when “0” is written to it. Writing “1” to it is invalid. | |

| RWST | Wait status flag of real-time counter |
|---|---|
| 0 | Counter is operating. |
| 1 | Mode to read or write counter value |
| This status flag indicates whether the setting of RWAIT is valid.<br>Before reading or writing the counter value, confirm that the value of this flag is 1. | |

| RWAIT | Wait control of real-time counter |
|---|---|
| 0 | Sets counter operation. |
| 1 | Stops SEC to YEAR counters. Mode to read or write counter value |
| This bit controls the operation of the counter.<br>Be sure to write “1” to it to read or write the counter value.<br>Because RSUBC continues operation, complete reading or writing of it in 1 second, and clear this bit back to 0.<br>When RWAIT = 1, it takes up to 1 clock (32.768 kHz) until the counter value can be read or written.<br>If RSUBC overflows when RWAIT = 1, it counts up after RWAIT = 0. If the second count register is written, however, it does not count up because RSUBC is cleared. | |

**Caution If writing is performed to the RTCC1 register with a 1-bit manipulation instruction, the RIFG flag and WAFG flag may be cleared. Therefore, to perform writing to the RTCC1 register, be sure to use an 8-bit manipulation instruction. To prevent the RIFG flag and WAFG flag from being cleared during writing, disable writing by setting 1 to the corresponding bit. If the RIFG flag and WAFG flag are not used and the value may be changed, the RTCC1 register may be written by using a 1-bit manipulation instruction.**

**Remark** Fixed-cycle interrupts and alarm match interrupts use the same interrupt source (INTRTC). When using these two types of interrupts at the same time, which interrupt occurred can be judged by checking the fixed-cycle interrupt status flag (RIFG) and the alarm detection status flag (WAFG) upon INTRTC occurrence.

### (4) Real-time counter control register 2 (RTCC2)

The RTCC2 register is an 8-bit register that is used to control the interval interrupt function and the RTCDIV pin.

RTCC2 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-5. Format of Real-Time Counter Control Register 2 (RTCC2)**

Address: FF6FH After reset: 00H R/W

| Symbol | <7> | <6> | <5> | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RTCC2 | RINTE | RCLOE2 | RCKDIV | 0 | 0 | ICT2 | ICT1 | ICT0 |

| RINTE | ICT2 | ICT1 | ICT0 | Interval interrupt (INTRTCI) selection |
|---|---|---|---|---|
| 0 | × | × | × | Interval interrupt is not generated. |
| 1 | 0 | 0 | 0 | $2^6/f_{SUB}$ (1.953125 ms) |
| 1 | 0 | 0 | 1 | $2^7/f_{SUB}$ (3.90625 ms) |
| 1 | 0 | 1 | 0 | $2^8/f_{SUB}$ (7.8125 ms) |
| 1 | 0 | 1 | 1 | $2^9/f_{SUB}$ (15.625 ms) |
| 1 | 1 | 0 | 0 | $2^{10}/f_{SUB}$ (31.25 ms) |
| 1 | 1 | 0 | 1 | $2^{11}/f_{SUB}$ (62.5 ms) |
| 1 | 1 | 1 | × | $2^{12}/f_{SUB}$ (125 ms) |

| RCLOE2 Notes 1, 2 | RTCDIV pin output control |
|---|---|
| 0 | Output of RTCDIV pin is disabled. |
| 1 | Output of RTCDIV pin is enabled. |

| RCKDIV Note 1 | Selection of RTCDIV pin output frequency |
|---|---|
| 0 | RTCDIV pin outputs 512 Hz (1.95 ms). |
| 1 | RTCDIV pin outputs 16.384 kHz (0.061 ms). |

**Notes 1.** 78K0/KC2-L (44-pin and 48-pin products) only

**2.** RCLOE0 and RCLOE2 must not be enabled at the same time.

**Cautions 1. Change ICT2, ICT1, and ICT0 when RINTE = 0.**

**2. When the output from RTCDIV pin is stopped, the output continues after a maximum of two clocks of $f_{SUB}$ and enters the low level. While 512 Hz is output, and when the output is stopped immediately after entering the high level, a pulse of at least one clock width of $f_{SUB}$ may be generated.**

**Remark** $f_{SUB}$: Subsystem clock oscillation frequency

**(5) Sub-count register (RSUBC)**

The RSUBC register is a 16-bit register that counts the reference time of 1 second of the real-time counter. It takes a value of 0000H to 7FFFH and counts 1 second with a clock of 32.768 kHz.

RSUBC can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Cautions 1. When a correction is made by using the SUBCUD register, the value may become 8000H or more.**

**2. This register is also cleared by reset effected by writing the second count register.**

**3. The value read from this register is not guaranteed if it is read during operation, because a value that is changing is read.**

**Figure 10-6. Format of Sub-Count Register (RSUBC)**

Address: FFB0H, FFB1H After reset: 0000H R

| | FFB1H | | | | | | | | FFB0H | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Symbol | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RSUBC | | | | | | | | | | | | | | | | |

**(6) Second count register (SEC)**

The SEC register is an 8-bit register that takes a value of 0 to 59 (decimal) and indicates the count value of seconds.

It counts up when the sub-counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later.

Set a decimal value of 00 to 59 to this register in BCD code. If a value outside this range is set, the register value returns to the normal value after 1 period.

SEC can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-7. Format of Second Count Register (SEC)**

Address: FFB2H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| SEC | 0 | SEC40 | SEC20 | SEC10 | SEC8 | SEC4 | SEC2 | SEC1 |

**(7) Minute count register (MIN)**

The MIN register is an 8-bit register that takes a value of 0 to 59 (decimal) and indicates the count value of minutes.

It counts up when the second counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later.

Even if the second count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 00 to 59 to this register in BCD code. If a value outside this range is set, the register value returns to the normal value after 1 period.

MIN can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-8. Format of Minute Count Register (MIN)**

Address: FFB3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MIN | 0 | MIN40 | MIN20 | MIN10 | MIN8 | MIN4 | MIN2 | MIN1 |

**(8) Hour count register (HOUR)**

The HOUR register is an 8-bit register that takes a value of 00 to 23 or 01 to 12, 21 to 32 (decimal) and indicates the count value of hours.

It counts up when the minute counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later.

Even if the minute count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Specify a decimal value of 00 to 23, 01 to 12, or 21 to 32 by using BCD code according to the time system specified using bit 3 (AMPM) of real-time counter control register 0 (RTCC0).

If the AMPM bit value is changed, the values of the HOUR register change according to the specified time system.

If a value outside the range is set, the register value returns to the normal value after 1 period.

HOUR can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 12H.

However, the value of this register is 00H if the AMPM bit (bit 3 of the RTCC0 register) is set to 1 after reset.

**Figure 10-9. Format of Hour Count Register (HOUR)**

Address: FFB4H After reset: 12H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| HOUR | 0 | 0 | HOUR20 | HOUR10 | HOUR8 | HOUR4 | HOUR2 | HOUR1 |

**Caution Bit 5 (HOUR20) of HOUR indicates AM(0)/PM(1) if AMPM = 0 (if the 12-hour system is selected).**

**Table 10-2. Displayed Time Digits**

| 24-Hour Display (AMPM bit = 1) | | 12-Hour Display (AMPM bit = 0) | |
|---|---|---|---|
| Time | HOUR Register | Time | HOUR Register |
| 0 | 00H | 0 a.m. | 12H |
| 1 | 01H | 1 a.m. | 01H |
| 2 | 02H | 2 a.m. | 02H |
| 3 | 03H | 3 a.m. | 03H |
| 4 | 04H | 4 a.m. | 04H |
| 5 | 05H | 5 a.m. | 05H |
| 6 | 06H | 6 a.m. | 06H |
| 7 | 07H | 7 a.m. | 07H |
| 8 | 08H | 8 a.m. | 08H |
| 9 | 09H | 9 a.m. | 09H |
| 10 | 10H | 10 a.m. | 10H |
| 11 | 11H | 11 a.m. | 11H |
| 12 | 12H | 0 p.m. | 32H |
| 13 | 13H | 1 p.m. | 21H |
| 14 | 14H | 2 p.m. | 22H |
| 15 | 15H | 3 p.m. | 23H |
| 16 | 16H | 4 p.m. | 24H |
| 17 | 17H | 5 p.m. | 25H |
| 18 | 18H | 6 p.m. | 26H |
| 19 | 19H | 7 p.m. | 27H |
| 20 | 20H | 8 p.m. | 28H |
| 21 | 21H | 9 p.m. | 29H |
| 22 | 22H | 10 p.m. | 30H |
| 23 | 23H | 11 p.m. | 31H |

**(9) Day count register (DAY)**

The DAY register is an 8-bit register that takes a value of 1 to 31 (decimal) and indicates the count value of days. It counts up when the hour counter overflows.

This counter counts as follows.

- 01 to 31 (January, March, May, July, August, October, December)
- 01 to 30 (April, June, September, November)
- 01 to 29 (February, leap year)
- 01 to 28 (February, normal year)

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later. Even if the hour count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 01 to 31 to this register in BCD code. If a value outside the range is set, the register value returns to the normal value after 1 period.

DAY can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 01H.

**Figure 10-10. Format of Day Count Register (DAY)**

Address: FFB6H After reset: 01H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| DAY | 0 | 0 | DAY20 | DAY10 | DAY8 | DAY4 | DAY2 | DAY1 |

**(10) Week count register (WEEK)**

The WEEK register is an 8-bit register that takes a value of 0 to 6 (decimal) and indicates the count value of weekdays.

It counts up in synchronization with the day counter.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later. Set a decimal value of 00 to 06 to this register in BCD code. If a value outside this range is set, the register value returns to the normal value after 1 period.

WEEK can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-11. Format of Week Count Register (WEEK)**

Address: FFB5H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| WEEK | 0 | 0 | 0 | 0 | 0 | WEEK4 | WEEK2 | WEEK1 |

**Caution Values corresponding to the month count register and day count register are not automatically stored to the week count register.**

**Set the week count register as follows, after reset release.**

| Day | WEEK |
|---|---|
| Sunday | 00H |
| Monday | 01H |
| Tuesday | 02H |
| Wednesday | 03H |
| Thursday | 04H |
| Friday | 05H |
| Saturday | 06H |

**(11) Month count register (MONTH)**

The MONTH register is an 8-bit register that takes a value of 1 to 12 (decimal) and indicates the count value of months.

It counts up when the day counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later. Even if the day count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 01 to 12 to this register in BCD code. If a value outside this range is set, the register value returns to the normal value after 1 period.

MONTH can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 01H.

**Figure 10-12. Format of Month Count Register (MONTH)**

Address: FFB7H After reset: 01H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MONTH | 0 | 0 | 0 | MONTH10 | MONTH8 | MONTH4 | MONTH2 | MONTH1 |

**(12) Year count register (YEAR)**

The YEAR register is an 8-bit register that takes a value of 0 to 99 (decimal) and indicates the count value of years.

It counts up when the month counter overflows.

Values 00, 04, 08, …, 92, and 96 indicate a leap year.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later. Even if the month count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 00 to 99 to this register in BCD code. If a value outside this range is set, the register value returns to the normal value after 1 period.

YEAR can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-13. Format of Year Count Register (YEAR)**

Address: FFB8H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| YEAR | YEAR80 | YEAR40 | YEAR20 | YEAR10 | YEAR8 | YEAR4 | YEAR2 | YEAR1 |

### (13) Watch error correction register (SUBCUD)

This register is used to correct the watch with high accuracy when it is slow or fast by changing the value that overflows from the sub-count register (RSUBC) to the second count register (reference value: 7FFFH).

SUBCUD can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-14. Format of Watch Error Correction Register (SUBCUD)**

Address: FFB9H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| SUBCUD | DEV | F6 | F5 | F4 | F3 | F2 | F1 | F0 |

| DEV | Setting of watch error correction timing |
|---|---|
| 0 | Corrects watch error when the second digits are at 00, 20, or 40 (every 20 seconds). |
| 1 | Corrects watch error only when the second digits are at 00 (every 60 seconds). |
| Writing to the SUBCUD register at the following timing is prohibited.<br>• When DEV = 0 is set: For a period of SEC = 00H, 20H, 40H<br>• When DEV = 1 is set: For a period of SEC = 00H | |

| F6 | Setting of watch error correction value |
|---|---|
| 0 | Increases by {(F5, F4, F3, F2, F1, F0) – 1} × 2. |
| 1 | Decreases by {(/F5, /F4, /F3, /F2, /F1, /F0) + 1} × 2. |
| When (F6, F5, F4, F3, F2, F1, F0) = (*, 0, 0, 0, 0, 0, *), the watch error is not corrected. * is 0 or 1.<br>/F5 to /F0 are the inverted values of the corresponding bits (000011 when 111100).<br>Range of correction value: (when F6 = 0) 2, 4, 6, 8, … , 120, 122, 124<br>(when F6 = 1) –2, –4, –6, –8, … , –120, –122, –124 | |

The range of value that can be corrected by using the watch error correction register (SUBCUD) is shown below.

| | DEV = 0 (correction every 20 seconds) | DEV = 1 (correction every 60 seconds) |
|---|---|---|
| Correctable range | –189.2 ppm to 189.2 ppm | –63.1 ppm to 63.1 ppm |
| Maximum excludes quantization error | ± 1.53 ppm | ± 0.51 ppm |
| Minimum resolution | ± 3.05 ppm | ± 1.02 ppm |

**Remark** If a correctable range is –63.1 ppm or lower and 63.1 ppm or higher, set 0 to DEV.

**(14) Alarm minute register (ALARMWM)**

This register is used to set minutes of alarm.

ALARMWM can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Caution Set a decimal value of 00 to 59 to this register in BCD code. If a value outside the range is set, the alarm is not detected.**

**Figure 10-15. Format of Alarm Minute Register (ALARMWM)**

Address: FF9AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ALARMWM | 0 | WM40 | WM20 | WM10 | WM8 | WM4 | WM2 | WM1 |

**(15) Alarm hour register (ALARMWH)**

This register is used to set hours of alarm.

ALARMWH can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 12H.

However, the value of this register is 00H if the AMPM bit (bit 3 of the RTCC0 register) is set to 1 after reset.

**Caution Set a decimal value of 00 to 23, 01 to 12, or 21 to 32 to this register in BCD code. If a value outside the range is set, the alarm is not detected.**

**Figure 10-16. Format of Alarm Hour Register (ALARMWH)**

Address: FF9BH After reset: 12H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ALARMWH | 0 | 0 | WH20 | WH10 | WH8 | WH4 | WH2 | WH1 |

**Caution Bit 5 (WH20) of ALARMWH indicates AM(0)/PM(1) if AMPM = 0 (if the 12-hour system is selected).**

**(16) Alarm week register (ALARMWW)**

This register is used to set date of alarm.

ALARMWW can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10-17. Format of Alarm Week Register (ALARMWW)**

Address: FF9CH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ALARMWW | 0 | WW6 | WW5 | WW4 | WW3 | WW2 | WW1 | WW0 |

Here is an example of setting the alarm.

| Time of Alarm | Day | | | | | | | 12-Hour Display | | | | 24-Hour Display | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Sunday<br>WW0 | Monday<br>WW1 | Tuesday<br>WW2 | Wednesday<br>WW3 | Thursday<br>WW4 | Friday<br>WW5 | Saturday<br>WW6 | Hour 10 | Hour 1 | Minute 10 | Minute 1 | Hour 10 | Hour 1 | Minute 10 | Minute 1 |
| Every day, 0:00 a.m. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| Every day, 1:30 a.m. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 3 | 0 | 0 | 1 | 3 | 0 |
| Every day, 11:59 a.m. | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 5 | 9 | 1 | 1 | 5 | 9 |
| Monday through Friday, 0:00 p.m. | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 3 | 2 | 0 | 0 | 1 | 2 | 0 | 0 |
| Sunday, 1:30 p.m. | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 3 | 0 | 1 | 3 | 3 | 0 |
| Monday, Wednesday, Friday, 11:59 p.m. | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 3 | 1 | 5 | 9 | 2 | 3 | 5 | 9 |

**(17) Port mode register 4 (PM4)**

This register sets port 4 input/output in 1-bit units.

When using the P40/RTCDIV/RTCCL(/SCK11) and P41/RTC1HZ(/SI11) pins for clock output of real-time counter, clear PM40 and PM41 and the output latches of P40 and P41 to 0.

PM4 is set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Figure 10-18. Format of Port Mode Register 4 (PM4)**

Address: FF24H After reset: FFH R/W

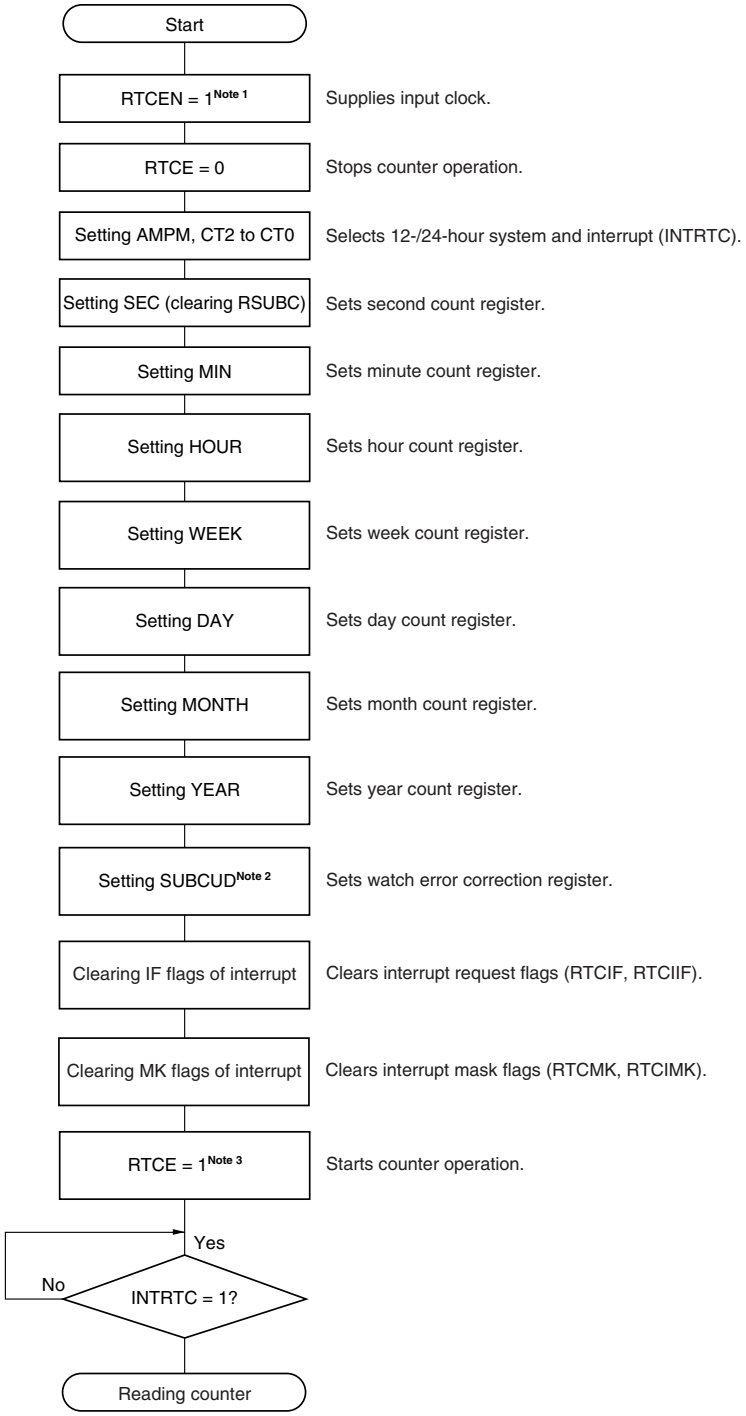
| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM4 | 1 | 1 | 1 | 1 | 1 | PM42 | PM41 | PM40 |

| PM4n | P4n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

## 10.4 Real-Time Counter Operation

### 10.4.1 Starting operation of real-time counter

**Figure 10-19. Procedure for Starting Operation of Real-Time Counter**




**Notes 1.** First set RTCEN to 1, while oscillation of the subsystem clock ($f_{SUB}$) is stable.

**2.** Set up SUBCUD only if the watch error must be corrected. For details about how to calculate the correction value, see **10.4.8 Example of watch error correction of real-time counter**.

**3.** Confirm the procedure described in **10.4.2 Shifting to STOP mode after starting operation** when shifting to STOP mode without waiting for INTRTC = 1 after RTCE = 1.

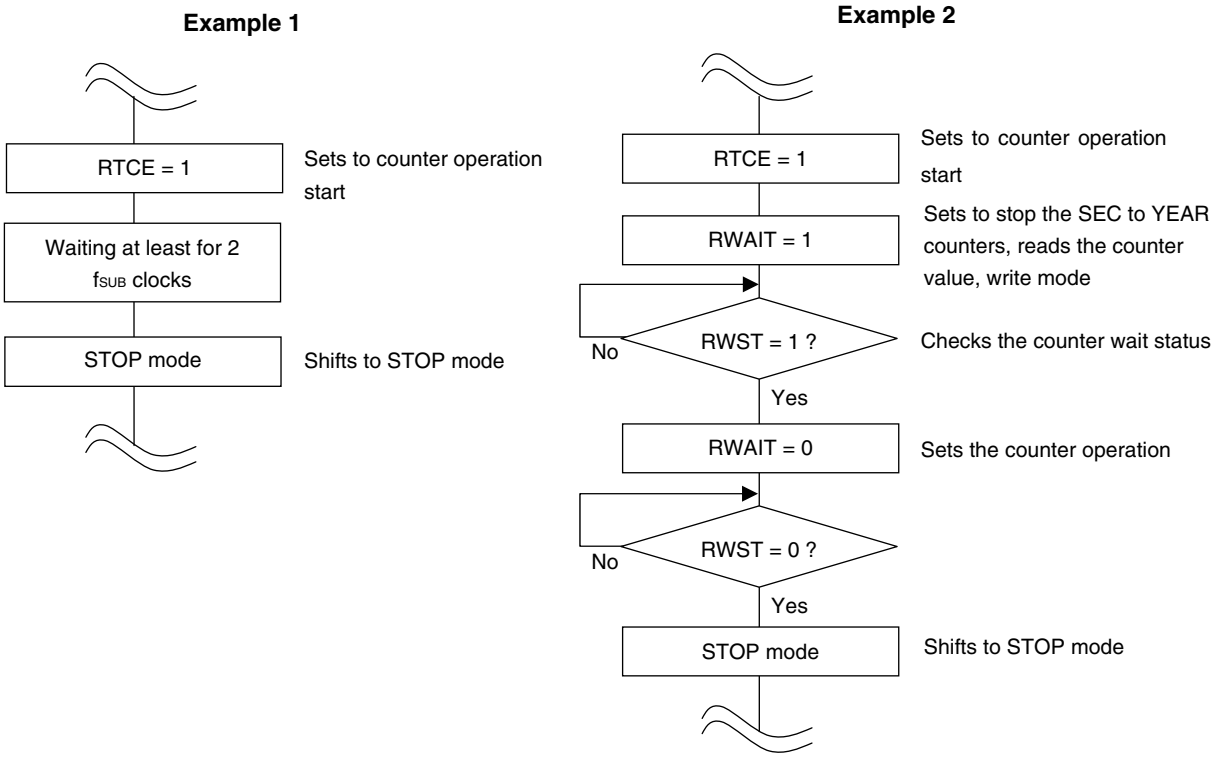
### 10.4.2 Shifting to STOP mode after starting operation

Perform one of the following processing when shifting to STOP mode immediately after setting RTCE to 1.

However, after setting RTCE to 1, this processing is not required when shifting to STOP mode after the first INTRTC interrupt has occurred.

- Shifting to STOP mode when at least two subsystem clocks ($f_{SUB}$) (about 62 $\mu$s) have elapsed after setting RTCE to 1 (see **Figure 10-20**, **Example 1**).
- Checking by polling RWST to become 1, after setting RTCE to 1 and then setting RWAIT to 1. Afterward, setting RWAIT to 0 and shifting to STOP mode after checking again by polling that RWST has become 0 (see **Figure 10-20**, **Example 2**).

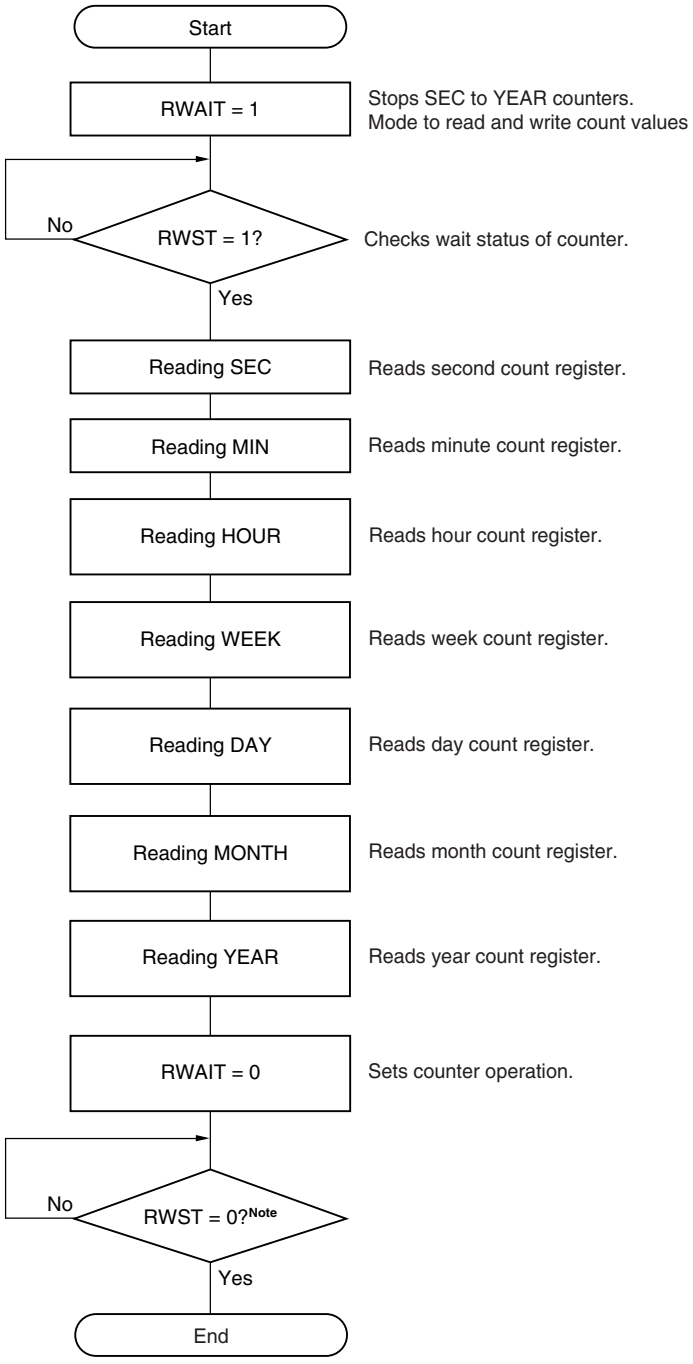
**Figure 10-20. Procedure for Shifting to STOP Mode After Setting RTCE to 1**



### 10.4.3 Reading/writing real-time counter

Read or write the counter after setting 1 to RWAIT first.

**Figure 10-21. Procedure for Reading Real-Time Counter**

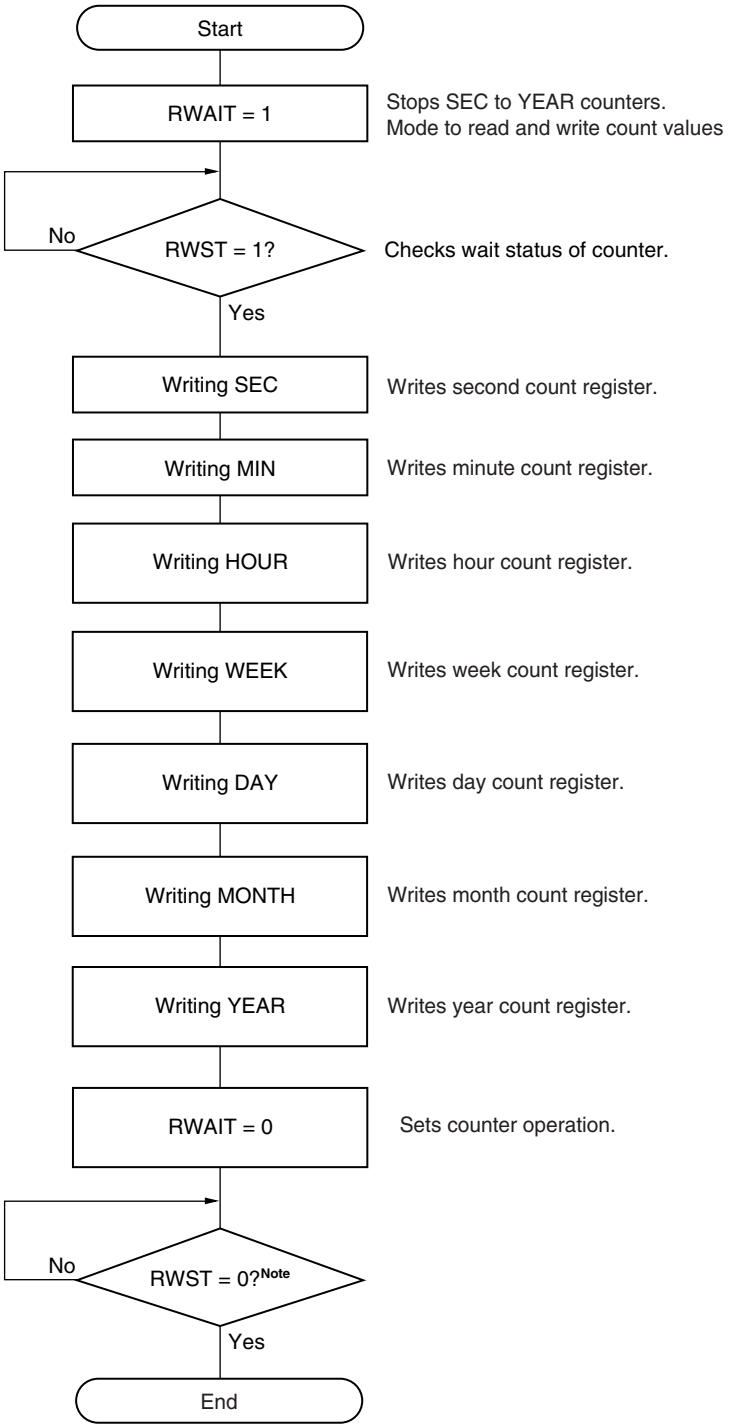



**Note** Be sure to confirm that RWST = 0 before setting STOP mode.

**Caution Complete the series of operations of setting RWAIT to 1 to clearing RWAIT to 0 within 1 second.**

**Remark** SEC, MIN, HOUR, WEEK, DAY, MONTH, and YEAR may be read in any sequence.
All the registers do not have to be set and only some registers may be read.

**Figure 10-22. Procedure for Writing Real-Time Counter**




**Note** Be sure to confirm that RWST = 0 before setting STOP mode.

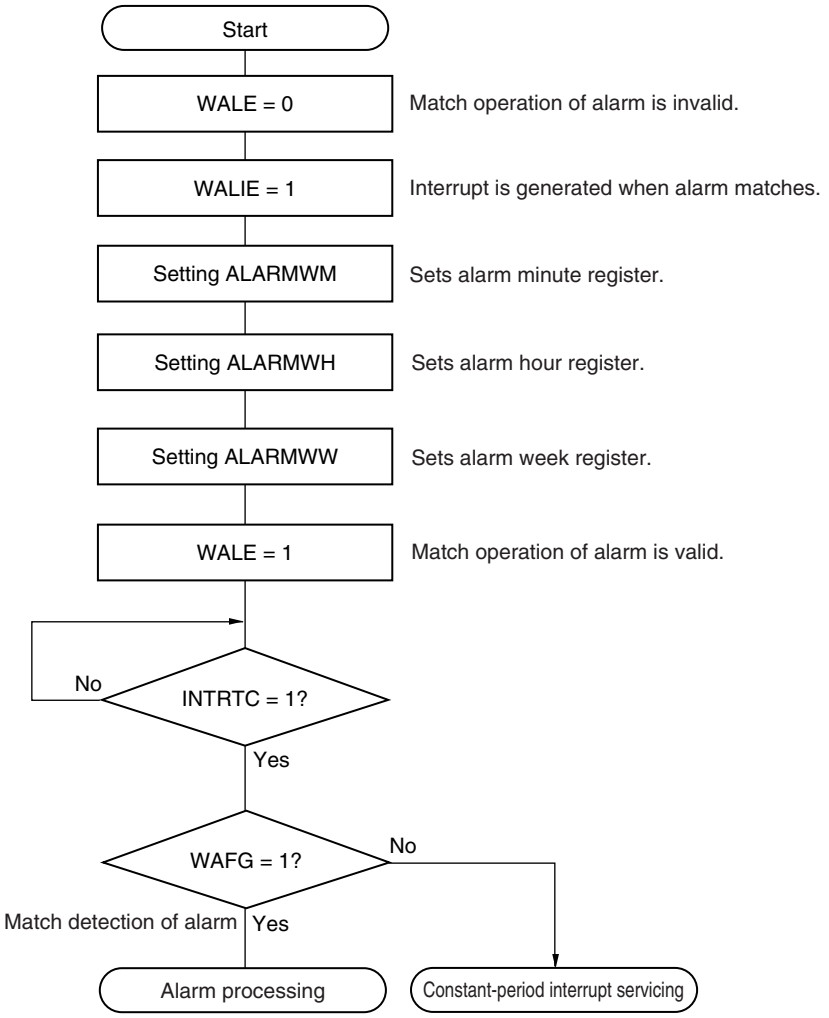
**Caution Complete the series of operations of setting RWAIT to 1 to clearing RWAIT to 0 within 1 second.**

**Remark** SEC, MIN, HOUR, WEEK, DAY, MONTH, and YEAR may be written in any sequence.
All the registers do not have to be set and only some registers may be written.

### 10.4.4 Setting alarm of real-time counter

Set time of alarm after setting 0 to WALE first.

**Figure 10-23. Alarm Setting Procedure**



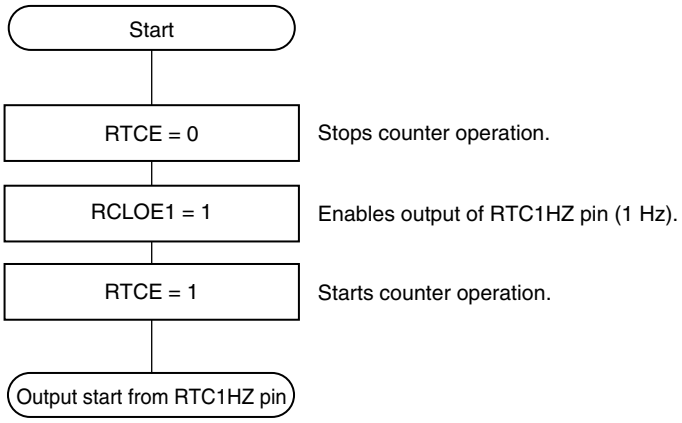

**Remarks 1.** ALARMWM, ALARMWH, and ALARMWW may be written in any sequence.

**2.** Fixed-cycle interrupts and alarm match interrupts use the same interrupt source (INTRTC). When using these two types of interrupts at the same time, which interrupt occurred can be judged by checking the fixed-cycle interrupt status flag (RIFG) and the alarm detection status flag (WAFG) upon INTRTC occurrence.

### 10.4.5 1 Hz output of real-time counter

Set 1 Hz output after setting 0 to RTCE first.

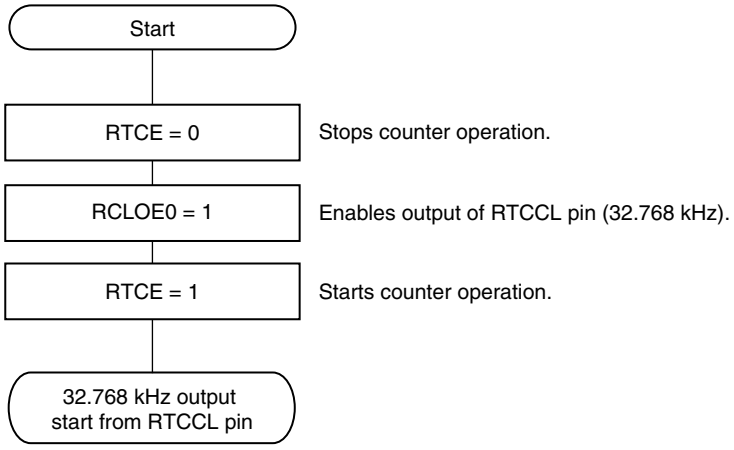
**Figure 10-24. 1 Hz Output Setting Procedure**




### 10.4.6 32.768 kHz output of real-time counter

Set 32.768 kHz output after setting 0 to RTCE first.

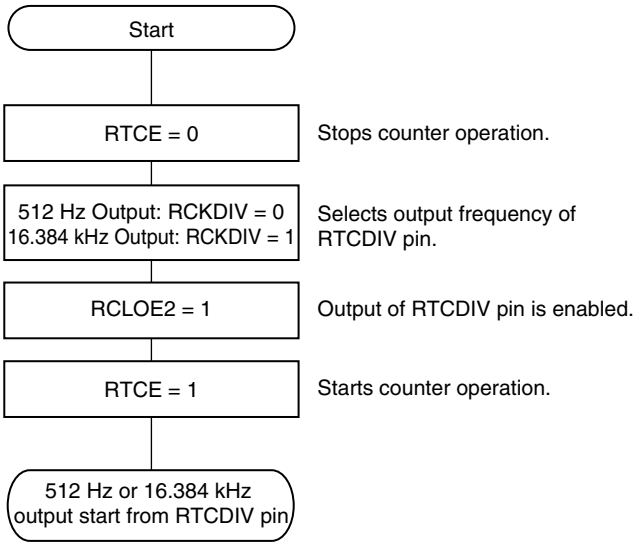
**Figure 10-25. 32.768 kHz Output Setting Procedure**



### 10.4.7 512 Hz, 16.384 kHz output of real-time counter

Set 512 Hz or 16.384 kHz output after setting 0 to RTCE first.

**Figure 10-26. 512 Hz, 16.384 kHz output Setting Procedure**



### 10.4.8 Example of watch error correction of real-time counter

The watch can be corrected with high accuracy when it is slow or fast, by setting a value to the watch error correction register.

Example of calculating the correction value

The correction value used when correcting the count value of the sub-count register (RSUBC) is calculated by using the following expression.

Set DEV to 0 when the correction range is –63.1 ppm or less, or 63.1 ppm or more.

(When DEV = 0)

Correction value[Note] = Number of correction counts in 1 minute ÷ 3 = (Oscillation frequency ÷ Target frequency – 1) × 32768 × 60 ÷ 3

(When DEV = 1)

Correction value[Note] = Number of correction counts in 1 minute = (Oscillation frequency ÷ Target frequency – 1) × 32768 × 60

**Note** The correction value is the watch error correction value calculated by using bits 6 to 0 of the watch error correction register (SUBCUD).

(When F6 = 0) Correction value = {(F5, F4, F3, F2, F1, F0) – 1} × 2

(When F6 = 1) Correction value = – {(/F5, /F4, /F3, /F2, /F1, /F0) + 1} × 2

When (F6, F5, F4, F3, F2, F1, F0) is (*, 0, 0, 0, 0, 0, *), watch error correction is not performed. "*" is 0 or 1. /F5 to /F0 are bit-inverted values (000011 when 111100).

**Remarks 1.** The correction value is 2, 4, 6, 8, … 120, 122, 124 or –2, –4, –6, –8, … –120, –122, –124.

**2.** The oscillation frequency is the subsystem clock ($f_{SUB}$) value.
It can be calculated from the 32.768 kHz output frequency of the RTCCL pin or the output frequency of the RTC1HZ pin × 32768 when the watch error correction register is set to its initial value (00H).

**3.** The target frequency is the frequency resulting after correction performed by using the watch error correction register.

Correction example <1>

Example of correcting from 32772.3 Hz to 32768 Hz (32772.3 Hz − 131.2 ppm)

[Measuring the oscillation frequency]

The oscillation frequency[Note] of each product is measured by outputting about 32 kHz from the RTCCL pin or outputting about 1 Hz from the RTC1HZ pin when the watch error correction register is set to its initial value (00H).

**Note** Refer to **10.4.5 1 Hz output of real-time counter** for the setting procedure of outputting about 1 Hz from the RTC1HZ pin, and **10.4.6 32.768 kHz output of real-time counter** for the setting procedure of outputting about 32 kHz from the RTCCL pin.

[Calculating the correction value]

(When the output frequency from the RTCCL pin is 32772.3 Hz)

If the target frequency is assumed to be 32768 Hz (32772.3 Hz − 131.2 ppm), the correction range for −131.2 ppm is −63.1 ppm or less, so assume DEV to be 0.

The expression for calculating the correction value when DEV is 0 is applied.

Correction value = Number of correction counts in 1 minute ÷ 3
= (Oscillation frequency ÷ Target frequency − 1) × 32768 × 60 ÷ 3
= (32772.3 ÷ 32768 − 1) × 32768 × 60 ÷ 3
= 86

[Calculating the values to be set to (F6 to F0)]

(When the correction value is 86)

If the correction value is 0 or more (when delaying), assume F6 to be 0.

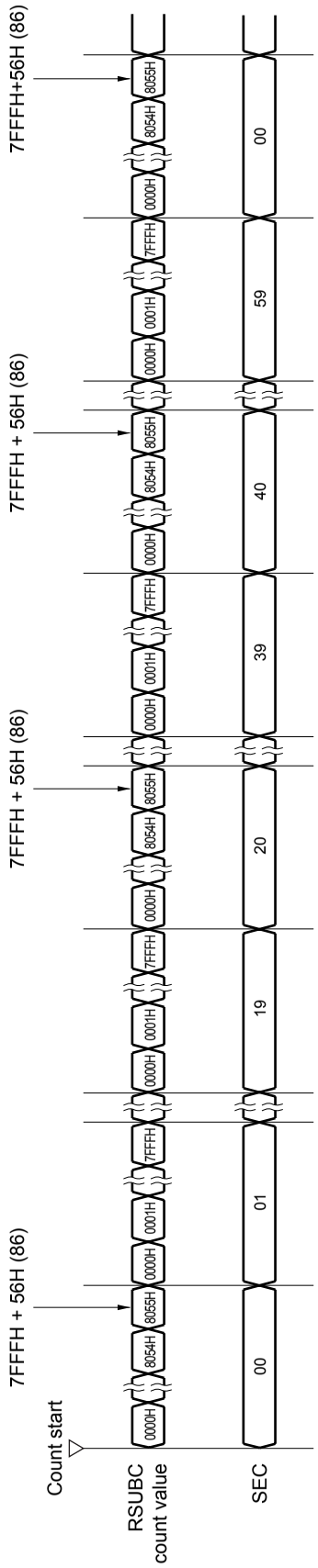
Calculate (F5, F4, F3, F2, F1, F0) from the correction value.

{ (F5, F4, F3, F2, F1, F0) − 1} × 2 = 86
(F5, F4, F3, F2, F1, F0) = 44
(F5, F4, F3, F2, F1, F0) = (1, 0, 1, 1, 0, 0)

Consequently, when correcting from 32772.3 Hz to 32768 Hz (32772.3 Hz − 131.2 ppm), setting the correction register such that DEV is 0 and the correction value is 86 (bits 6 to 0 of SUBCUD: 0101100) results in 32768 Hz (0 ppm).

Figure 10-27 shows the operation when (DEV, F6, F5, F4, F3, F2, F1, F0) is (0, 0, 1, 0, 1, 1, 0, 0).

**Figure 10-27. Operation when (DEV, F6, F5, F4, F3, F2, F1, F0) = (0, 0, 1, 0, 1, 1, 0, 0)**



Correction example <2>

Example of correcting from 32767.4 Hz to 32768 Hz (32767.4 Hz + 18.3 ppm)

[Measuring the oscillation frequency]

The oscillation frequency[Note] of each product is measured by outputting about 32 kHz from the RTCCL pin or outputting about 1 Hz from the RTC1HZ pin when the watch error correction register is set to its initial value (00H).

**Note** Refer to **10.4.5 1 Hz output of real-time counter** for the setting procedure of outputting about 1 Hz from the RTC1HZ pin, and **10.4.6 32.768 kHz output of real-time counter** for the setting procedure of outputting about 32 kHz from the RTCCL pin.

[Calculating the correction value]

(When the output frequency from the RTCCL pin is 0.9999817 Hz)

Oscillation frequency = 32768 × 0.9999817 ≈ 32767.4 Hz

Assume the target frequency to be 32768 Hz (32767.4 Hz + 18.3 ppm) and DEV to be 1.

The expression for calculating the correction value when DEV is 1 is applied.

Correction value = Number of correction counts in 1 minute
= (Oscillation frequency ÷ Target frequency − 1) × 32768 × 60
= (32767.4 ÷ 32768 − 1) × 32768 × 60
= −36

[Calculating the values to be set to (F6 to F0)]

(When the correction value is −36)

If the correction value is 0 or less (when quickening), assume F6 to be 1.

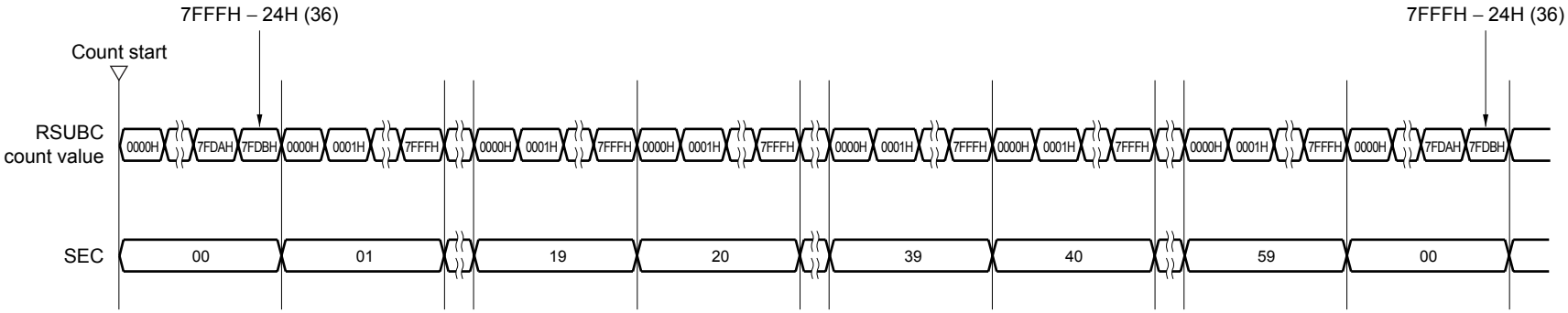
Calculate (F5, F4, F3, F2, F1, F0) from the correction value.

| | |
|---|---|
| − {(/F5, /F4, /F3, /F2, /F1, /F0) − 1} × 2 | = −36 |
| (/F5, /F4, /F3, /F2, /F1, /F0) | = 17 |
| (/F5, /F4, /F3, /F2, /F1, /F0) | = (0, 1, 0, 0, 0, 1) |
| (F5, F4, F3, F2, F1, F0) | = (1, 0, 1, 1, 1, 0) |

Consequently, when correcting from 32767.4 Hz to 32768 Hz (32767.4 Hz + 18.3 ppm), setting the correction register such that DEV is 1 and the correction value is −36 (bits 6 to 0 of SUBCUD: 1101110) results in 32768 Hz (0 ppm).

Figure 10-28 shows the operation when (DEV, F6, F5, F4, F3, F2, F1, F0) is (1, 1, 1, 0, 1, 1, 1, 0).

**Figure 10-28. Operation when (DEV, F6, F5, F4, F3, F2, F1, F0) = (1, 1, 1, 0, 1, 1, 1, 0)**



# CHAPTER 11 CLOCK OUTPUT CONTROLLER

| Item | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | |
|---|---|---|---|---|---|
| | 16 Pins | 20, 25, 32 Pins | 30 Pins | 40, 44 Pins | 48 Pins |
| Clock output controller | – | | | | √ |

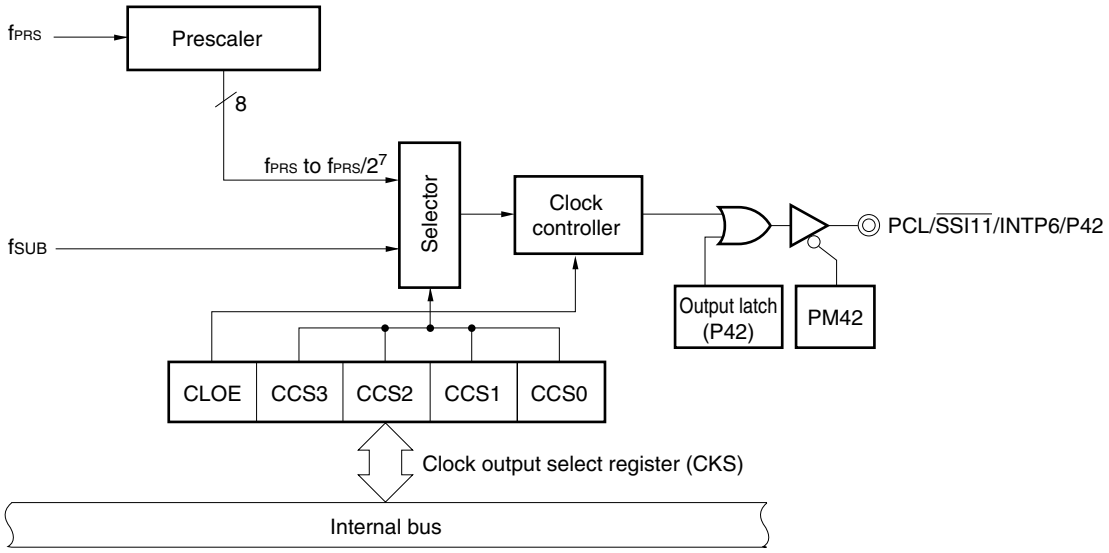
**Remark** √: Mounted, –: Not mounted

## 11.1 Functions of Clock Output Controller

The clock output controller is intended for carrier output during remote controlled transmission and clock output for supply to peripheral ICs. The clock selected with the clock output selection register (CKS) is output.

Figure 11-1 shows the block diagram of clock output controller.

**Figure 11-1. Block Diagram of Clock Output Controller (48-pin products of 78K0/KC2-L)**




## 11.2 Configuration of Clock Output Controller

The clock output controller includes the following hardware.

**Table 11-1. Configuration of Clock Output Controller**

| Item | Configuration |
|---|---|
| Control registers | Clock output selection register (CKS)<br>Port mode register 4 (PM4)<br>Port register 4 (P4) |

## 11.3 Registers Controlling Clock Output Controller

The following two registers are used to control the clock output controller.

- Clock output selection register (CKS)
- Port mode register 4 (PM4)

### (1) Clock output selection register (CKS)

This register sets output enable/disable for clock output (PCL) and sets the output clock.

CKS is set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears CKS to 00H.

**Figure 11-2. Format of Clock Output Selection Register (CKS) (48-pin products of 78K0/KC2-L)**

Address: FF40H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | <4> | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CKS | 0 | 0 | 0 | CLOE | CCS3 | CCS2 | CCS1 | CCS0 |

| CLOE | PCL output enable/disable specification |
|---|---|
| 0 | Clock division circuit operation stopped. PCL fixed to low level. |
| 1 | Clock division circuit operation enabled. PCL output enabled. |

| CCS3 | CCS2 | CCS1 | CCS0 | PCL output clock selection[Note 1] | | | |
|---|---|---|---|---|---|---|---|
| | | | | | $f_{SUB}$ = 32.768 kHz | $f_{PRS}$ = 4 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | 0 | $f_{PRS}$[Note 2] | – | 4 MHz | 10 MHz |
| 0 | 0 | 0 | 1 | $f_{PRS}/2$ | | 2 MHz | 5 MHz |
| 0 | 0 | 1 | 0 | $f_{PRS}/2^2$ | | 1 MHz | 2.5 MHz |
| 0 | 0 | 1 | 1 | $f_{PRS}/2^3$ | | 500 kHz | 1.25 MHz |
| 0 | 1 | 0 | 0 | $f_{PRS}/2^4$ | | 250 kHz | 625 kHz |
| 0 | 1 | 0 | 1 | $f_{PRS}/2^5$ | | 125 kHz | 312.5 kHz |
| 0 | 1 | 1 | 0 | $f_{PRS}/2^6$ | | 62.5 kHz | 156.25 kHz |
| 0 | 1 | 1 | 1 | $f_{PRS}/2^7$ | | 31.25 kHz | 78.125 kHz |
| 1 | 0 | 0 | 0 | $f_{SUB}$ | 32.768 kHz | – | |
| Other than above | | | | Setting prohibited | | | |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** If internal high-speed oscillation clock frequency is set to 8 MHz (R4M8MSEL = 0) by option byte and the peripheral hardware clock ($f_{PRS}$) operates on the internal high-speed oscillation clock ($f_{IH}$) (XSEL = 0) when 1.8 V $\leq V_{DD} <$ 2.7 V, setting CCS3 = CCS2 = CCS1 = CCS0 = 0 (output clock of PCL: $f_{PRS}$) is prohibited.

**Caution Set CCS3 to CCS0 while the clock output operation is stopped (CLOE = 0).**

**Remarks 1.** $f_{PRS}$: Peripheral hardware clock frequency

**2.** $f_{SUB}$: Subsystem clock frequency

**(2) Port mode register 4 (PM4)**

This register sets port 4 input/output in 1-bit units.

When using the P42/PCL/$\overline{SSI11}$/INTP6 pin for clock output, clear PM42 and the output latches of P42 to 0.

PM4 is set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets PM4 to FFH.

**Figure 11-3. Format of Port Mode Register 4 (PM4)**

Address: FF24H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM4 | 1 | 1 | 1 | 1 | 1 | PM42 | PM41 | PM40 |

| PM4n | P4n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 4 of 48-pin products (78K0/KC2-L).

## 11.4 Operations of Clock Output Controller

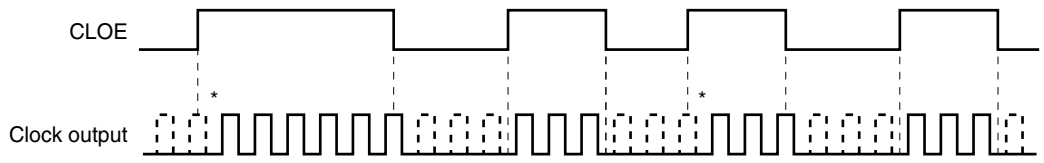
The clock pulse is output as the following procedure.

<1> Select the clock pulse output frequency with bits 0 to 3 (CCS0 to CCS3) of the clock output selection register (CKS) (clock pulse output in disabled status).

<2> Set bit 4 (CLOE) of CKS to 1 to enable clock output.

**Remark** The clock output controller is designed not to output pulses with a small width during output enable/disable switching of the clock output. As shown in Figure 11-4, be sure to start output from the low period of the clock (marked with * in the figure). When stopping output, do so after the high-level period of the clock.

**Figure 11-4. Remote Control Output Application Example**



# CHAPTER 12 A/D CONVERTER

<R>

| Item | 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|---|
| | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| 10-bit A/D converter | 4 ch | 6 ch | 7 ch | 11 ch | 7 ch | 10 ch | 11 ch | 11 ch |

## 12.1 Function of A/D Converter

The A/D converter converts an analog input signal into a digital value, and consists of up to 11 channels (ANI0 to ANI10) with a resolution of 10 bits.

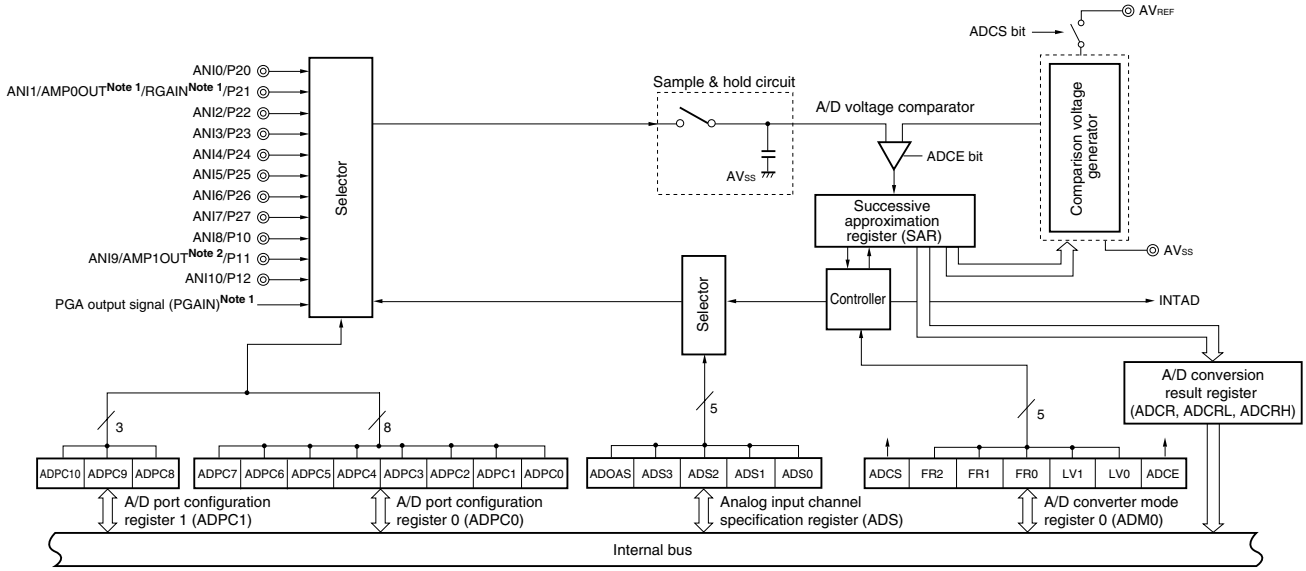
In products with operational amplifier, ANI1 function alternately as operational amplifier 0 output (AMP0OUT) and ANI9 function alternately as operational amplifier 1 output (AMP1OUT). This enables using operational amplifiers 0 and 1 output or PGA output as an analog input source.

The A/D converter has the following function.

- **10-bit resolution A/D conversion**

  10-bit resolution A/D conversion is carried out repeatedly for one analog input channel selected from ANI0 to ANI10, operational amplifiers 0 and 1 output, and PGA output. Each time an A/D conversion operation ends, an interrupt request (INTAD) is generated.

<R> **Figure 12-1. Block Diagram of A/D Converter**




**Notes 1.** Products with operational amplifier only

**2.** Products with operational amplifier of 78K0/KB2-L and 78K0/KC2-L only

**Caution In the 78K0/KY2-L and 78K0/KA2-L, VSS functions alternately as the ground potential of the A/D converter. Be sure to connect VSS to a stabilized GND (= 0 V).**

**Remark** A/D converter analog input pins differ depending on products.

- 78K0/KY2-L: ANI0 to ANI3
- 78K0/KA2-L (20-pin products): ANI0 to ANI5
- 78K0/KA2-L (25-pin products): ANI0 to ANI6
- 78K0/KA2-L (32-pin products): ANI0 to ANI10
- 78K0/KB2-L: ANI0 to ANI3, ANI8 to ANI10
- 78K0/KC2-L (40-pin product): ANI0 to ANI6, ANI8 to ANI10
- 78K0/KC2-L (44-pin and 48-pin products): ANI0 to ANI10

## 12.2 Configuration of A/D Converter

The A/D converter includes the following hardware.

**(1) ANI0 to ANI10 pins**

These are the analog input pins of the 11-channel A/D converter. They input analog signals to be converted into digital signals. Pins other than the one selected as the analog input pin can be used as I/O port pins.

<R> **Remark** A/D converter analog input pins differ depending on products.

- 78K0/KY2-L: ANI0 to ANI3
- 78K0/KA2-L (20-pin products): ANI0 to ANI5
- 78K0/KA2-L (25-pin products): ANI0 to ANI6
- 78K0/KA2-L (32-pin products): ANI0 to ANI10
- 78K0/KB2-L: ANI0 to ANI3, ANI8 to ANI10
- 78K0/KC2-L (40-pin product): ANI0 to ANI6, ANI8 to ANI10
- 78K0/KC2-L (44-pin and 48-pin products): ANI0 to ANI10

**(2) AMP0OUT pin (products with operational amplifier only)**

AMP0OUT is the output pin of operational amplifier 0.

This functions alternately as ANI1. The A/D converter can perform A/D conversion by selecting the output signal of operational amplifier 0 as the analog input source.

**(3) AMP1OUT pin (products with operational amplifier of 78K0/KB2-L and 78K0/KC2-L only)**

AMP1OUT is the output pin of operational amplifier 1.

This functions alternately as ANI9. The A/D converter can perform A/D conversion by selecting the output signal of operational amplifier 1 as the analog input source.

<R> **(4) PGAOUT signal (products with operational amplifier only)**

PGAOUT is the output signal of PGA.

The A/D converter can perform A/D conversion by selecting the output signal of PGA as the analog input source.

**(5) Sample & hold circuit**

The sample & hold circuit samples each of the analog input voltages sequentially sent from the input circuit, and sends them to the A/D voltage comparator. This circuit also holds the sampled analog input voltage during A/D conversion.

**(6) Comparison voltage generator**

The comparison voltage generator is connected between $AV_{REF}$ and $AV_{SS}$, and generates a voltage to be compared with an analog input. The operation of the comparison voltage generator is enabled or disabled by using the ADCS bit (bit 7 of the ADM0 register). The power consumption can be reduced by stopping the operation of the comparison voltage generator when A/D conversion is not performed.

**(7) A/D voltage comparator**

The A/D voltage comparator compares the sampled voltage values with the output voltage of the comparison voltage generator. The operation of the A/D voltage comparator is enabled or disabled by using the ADCE bit (bit 0 of the ADM0 register). The power consumption can be reduced by stopping the operation of the A/D voltage comparator when A/D conversion is not performed.

**(8) Successive approximation register (SAR)**

The SAR register is a 10-bit register that sets a result compared by the A/D voltage comparator, 1 bit at a time starting from the most significant bit (MSB).

If data is set in the SAR register all the way to the least significant bit (LSB) (end of A/D conversion), the contents of the SAR register (conversion results) are held in the A/D conversion result register (ADCR, ADCRH).

**(9) 10-bit A/D conversion result register (ADCR)**

The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCR register holds the A/D conversion result in its lower 10 bits (the higher 6 bits are fixed to 0).

**(10) 8-bit A/D conversion result register L (ADCRL)**

The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCRL register stores the lower 8 bits of the A/D conversion result.

**(11) 8-bit A/D conversion result register H (ADCRH)**

The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCRH register stores the higher 8 bits of the A/D conversion result.

**Caution When data is read from ADCR, ADCRL, and ADCRH, a wait cycle is generated. Do not read data from ADCR, ADCRL, and ADCRH when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(12) Controller**

This circuit controls the conversion time of an input analog signal that is to be converted into a digital signal, as well as starting and stopping of the conversion operation. When all the specified A/D conversion has been completed, this controller generates an A/D conversion end interrupt request signal (INTAD).

**(13) $AV_{REF}$ pin**

This pin inputs an analog power/reference voltage to the A/D converter. Make this pin the same potential as the $V_{DD}$ pin when port 2 is used as a digital port.

The signal input to ANI0 to ANI10 is converted into a digital signal, based on the voltage applied across $AV_{REF}$ and $AV_{SS}$.

**(14) $AV_{SS}$ pin (78K0/KB2-L and 78K0/KC2-L only)**

This is the ground potential pin of the A/D converter. Always use this pin at the same potential as that of the $V_{SS}$ pin even when the A/D converter is not used.

**(15) $V_{SS}$ pin**

This is the ground potential pin. In the 78K0/KY2-L and 78K0/KA2-L, $V_{SS}$ functions alternately as the ground potential of the A/D converter. Be sure to connect $V_{SS}$ to a stabilized GND (= 0 V).

## 12.3 Registers Used in A/D Converter

The A/D converter uses the following seven registers.

- A/D converter mode register 0 (ADM0)
- 10-bit A/D conversion result register (ADCR)
- 8-bit A/D conversion result register L (ADCRL)
- 8-bit A/D conversion result register H (ADCRH)
- Analog input channel specification register (ADS)
- A/D port configuration registers 0, 1 (ADPC0, ADPC1)
- Port mode registers 1, 2, 7 (PM1, PM2, PM7)

### (1) A/D converter mode register 0 (ADM0)

This register sets the conversion time for analog input to be A/D converted, and starts/stops conversion.

ADM0 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 12-2. Format of A/D Converter Mode Register 0 (ADM0)**

Address: FF28H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| ADM0 | ADCS | 0 | FR2Note 1 | FR1Note 1 | FR0Note 1 | LV1Note 1 | LV0Note 1 | ADCE |

| ADCS | A/D conversion operation control |
|---|---|
| 0 | Stops conversion operation |
| 1 | Enables conversion operation |

| ADCE | A/D voltage comparator operation controlNote 2 |
|---|---|
| 0 | Stops A/D voltage comparator operation |
| 1 | Enables A/D voltage comparator operation |

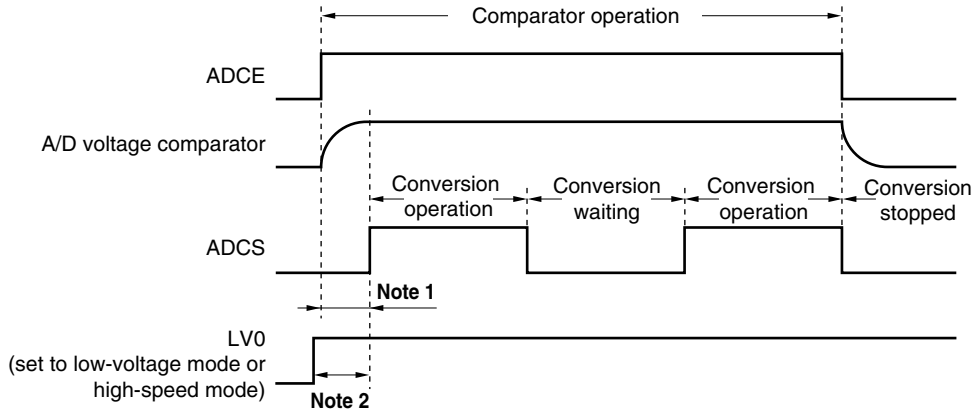
**Notes 1.** For details of FR2 to FR0, LV1, LV0, and A/D conversion, refer to **Table 12-2 A/D Conversion Time Selection**.

**2.** The operation of the A/D voltage comparator is controlled by ADCS and ADCE, and it takes 1 μs from operation start to operation stabilization. Therefore, when ADCS is set to 1 after 1 μs or more has elapsed from the time ADCE is set to 1, the conversion result at that time has priority over the first conversion result. Otherwise, ignore data of the first conversion.

**Table 12-1. Settings of ADCS and ADCE**

| ADCS | ADCE | A/D Conversion Operation |
|---|---|---|
| 0 | 0 | Stop status (DC power consumption path does not exist) |
| 0 | 1 | Conversion waiting mode (only A/D voltage comparator consumes power) |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Conversion mode (A/D voltage comparator operation) |

**Figure 12-3. Timing Chart When Comparator Is Used**




**Notes 1.** To stabilize the internal circuit, the time from setting ADCE to 1 to setting ADCS to 1 must be 1 $\mu$s or longer.

**2.** To stabilize the internal circuit, the time from setting LV0 to 1 (low-voltage mode or high-speed mode 2) to setting ADCS to 1 must be 1 $\mu$s or longer (for operation mode setting, refer to **Table 12-2**).

**Cautions 1. A/D conversion must be stopped before rewriting bits FR0 to FR2, LV1, and LV0 to values other than the identical data.**

**2. If data is written to ADM0, a wait cycle is generated. Do not write data to ADM0 when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**Table 12-2. A/D Conversion Time Selection (1/3)**

**(1) 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V**

| A/D Converter Mode Register 0 (ADM0) | | | | | Mode | Conversion Time Selection | | | | Conversion Clock ($f_{AD}$) |
|---|---|---|---|---|---|---|---|---|---|---|
| FR2 | FR1 | FR0 | LV1 | LV0 | | | $f_{PRS}$ = 4 MHz | $f_{PRS}$ = 8 MHz | $f_{PRS}$ = 10 MHz | |
| 0 | 0 | 0 | 0 | 0 | Standard | 264/$f_{PRS}$ | 66.0 μs | 33.0 μs | 26.4 μs | $f_{PRS}$/12 |
| 0 | 0 | 1 | | | | 176/$f_{PRS}$ | 44.0 μs | 22.0 μs | 17.6 μs | $f_{PRS}$/8 |
| 0 | 1 | 0 | | | | 132/$f_{PRS}$ | 33.0 μs | 16.5 μs | 13.2 μs | $f_{PRS}$/6 |
| 0 | 1 | 1 | | | | 88/$f_{PRS}$ | 22.0 μs | 11.0 μs | 8.8 μs | $f_{PRS}$/4 |
| 1 | 0 | 0 | | | | 66/$f_{PRS}$ | 16.5 μs | 8.25 μs | 6.6 μs | $f_{PRS}$/3 |
| 1 | 0 | 1 | | | | 44/$f_{PRS}$ | 11.0 μs | Setting prohibited | | $f_{PRS}$/2 |
| 1 | 1 | 0 | | | | 33/$f_{PRS}$ | 8.25 μs | Setting prohibited | | $f_{PRS}$/1.5 |
| 1 | 1 | 1 | | | | 22/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$ |
| 1 | 0 | 1 | 1 | 1 | High-speed 2 | 44/$f_{PRS}$ | 11.0 μs | 5.5 μs | 4.4 μs | $f_{PRS}$/2 |
| 1 | 1 | 1 | | | | 22/$f_{PRS}$ | 5.5 μs | Setting prohibited | | $f_{PRS}$ |
| 1 | 0 | 0 | 1 | 0 | High-speed 1 | 66/$f_{PRS}$ | 16.5 μs | 8.25 μs | 6.6 μs | $f_{PRS}$/3 |
| 1 | 1 | 0 | | | | 33/$f_{PRS}$ | 8.25 μs | 4.125 μs | 3.3 μs | $f_{PRS}$/1.5 |
| Other than above | | | | | Setting prohibited | | | | | |

**Cautions 1. When rewriting FR2 to FR0, LV1, and LV0 to other than the same data, stop A/D conversion once (ADCS = 0) beforehand.**

**2. The above conversion time does not include clock frequency errors. Select conversion time, taking clock frequency errors into consideration.**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**Table 12-2. A/D Conversion Time Selection (2/3)**

**(2) 2.7 V ≤ $AV_{REF}$ < 4.0 V**

| A/D Converter Mode Register 0 (ADM0) | | | | | Mode | Conversion Time Selection | | | | Conversion Clock ($f_{AD}$) |
|---|---|---|---|---|---|---|---|---|---|---|
| FR2 | FR1 | FR0 | LV1 | LV0 | | | $f_{PRS}$ = 4 MHz | $f_{PRS}$ = 8 MHz | $f_{PRS}$ = 10 MHz | |
| 0 | 0 | 0 | 0 | 0 | Standard | 264/$f_{PRS}$ | 66.0 μs | 33.0 μs | 26.4 μs | $f_{PRS}$/12 |
| 0 | 0 | 1 | | | | 176/$f_{PRS}$ | 44.0 μs | 22.0 μs | 17.6 μs | $f_{PRS}$/8 |
| 0 | 1 | 0 | | | | 132/$f_{PRS}$ | 33.0 μs | 16.5 μs | 13.2 μs | $f_{PRS}$/6 |
| 0 | 1 | 1 | | | | 88/$f_{PRS}$ | 22.0 μs | Setting prohibited | | $f_{PRS}$/4 |
| 1 | 0 | 0 | | | | 66/$f_{PRS}$ | 16.5 μs | Setting prohibited | | $f_{PRS}$/3 |
| 1 | 0 | 1 | | | | 44/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/2 |
| 1 | 1 | 0 | | | | 33/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/1.5 |
| 1 | 1 | 1 | | | | 22/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$ |
| 0 | 0 | 1 | 1 | 1 | High-speed 2 | 176/$f_{PRS}$ | 44.0 μs | 22.0 μs | 17.6 μs | $f_{PRS}$/8 |
| 0 | 1 | 0 | | | | 132/$f_{PRS}$ | 33.0 μs | 16.5 μs | 13.2 μs | $f_{PRS}$/6 |
| 0 | 1 | 1 | | | | 88/$f_{PRS}$ | 22.0 μs | 11.0 μs | 8.8 μs | $f_{PRS}$/4 |
| 1 | 0 | 0 | | | | 66/$f_{PRS}$ | 16.5 μs | 8.25 μs | 6.6 μs | $f_{PRS}$/3 |
| 1 | 0 | 1 | | | | 44/$f_{PRS}$ | 11.0 μs | 5.5 μs | 4.4 μs | $f_{PRS}$/2 |
| 1 | 1 | 0 | | | | 33/$f_{PRS}$ | 8.25 μs | Setting prohibited | | $f_{PRS}$/1.5 |
| 1 | 1 | 1 | | | | 22/$f_{PRS}$ | 5.5 μs | Setting prohibited | | $f_{PRS}$ |
| 0 | 0 | 0 | 0 | 1 | Low-voltage | 528/$f_{PRS}$ | Setting prohibited | 66.0 μs | 52.8 μs | $f_{PRS}$/12 |
| 0 | 0 | 1 | | | | 352/$f_{PRS}$ | Setting prohibited | 44.0 μs | Setting prohibited | $f_{PRS}$/8 |
| 0 | 1 | 0 | | | | 264/$f_{PRS}$ | 66.0 μs | Setting prohibited | | $f_{PRS}$/6 |
| 0 | 1 | 1 | | | | 176/$f_{PRS}$ | 44.0 μs | Setting prohibited | | $f_{PRS}$/4 |
| 1 | 0 | 0 | | | | 132/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/3 |
| 1 | 0 | 1 | | | | 88/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/2 |
| 1 | 1 | 0 | | | | 66/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/1.5 |
| 1 | 1 | 1 | | | | 44/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$ |
| Other than above | | | | | Setting prohibited | | | | | |

**Cautions 1. When rewriting FR2 to FR0, LV1, and LV0 to other than the same data, stop A/D conversion once (ADCS = 0) beforehand.**

**2. The above conversion time does not include clock frequency errors. Select conversion time, taking clock frequency errors into consideration.**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**Table 12-2. A/D Conversion Time Selection (3/3)**

**(3) 1.8 V ≤ $AV_{REF}$ < 2.7 V**

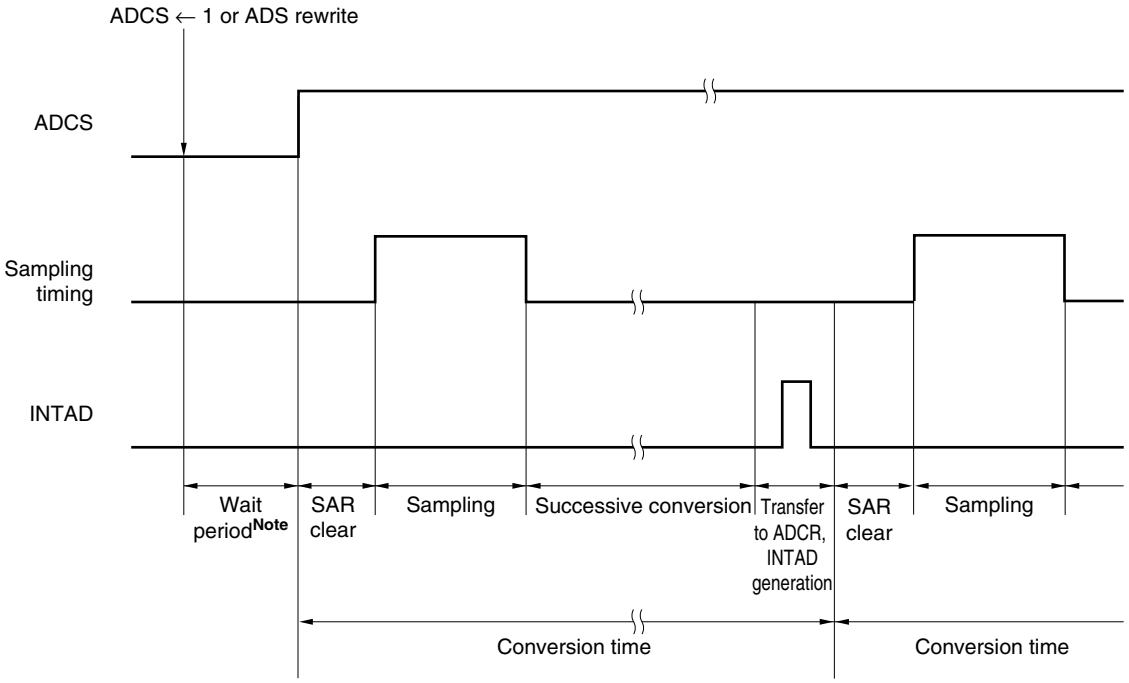
| A/D Converter Mode Register 0 (ADM0) | | | | | Mode | Conversion Time Selection | | | | Conversion Clock ($f_{AD}$) |
|---|---|---|---|---|---|---|---|---|---|---|
| FR2 | FR1 | FR0 | LV1 | LV0 | | | $f_{PRS}$ = 4 MHz | $f_{PRS}$ = 8 MHz | $f_{PRS}$ = 10 MHz | |
| 0 | 0 | 0 | 0 | 1 | Low-voltage | 528/$f_{PRS}$ | Setting prohibited | 66.0 μs | 52.8 μs | $f_{PRS}$/12 |
| 0 | 0 | 1 | | | | 352/$f_{PRS}$ | Setting prohibited | 44.0 μs | Setting prohibited | $f_{PRS}$/8 |
| 0 | 1 | 0 | | | | 264/$f_{PRS}$ | 66.0 μs | Setting prohibited | | $f_{PRS}$/6 |
| 0 | 1 | 1 | | | | 176/$f_{PRS}$ | 44.0 μs | Setting prohibited | | $f_{PRS}$/4 |
| 1 | 0 | 0 | | | | 132/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/3 |
| 1 | 0 | 1 | | | | 88/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/2 |
| 1 | 1 | 0 | | | | 66/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$/1.5 |
| 1 | 1 | 1 | | | | 44/$f_{PRS}$ | Setting prohibited | | | $f_{PRS}$ |
| Other than above | | | | | Setting prohibited | | | | | |

**Cautions 1. When rewriting FR2 to FR0, LV1, and LV0 to other than the same data, stop A/D conversion once (ADCS = 0) beforehand.**

**2. The above conversion time does not include clock frequency errors. Select conversion time, taking clock frequency errors into consideration.**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**Figure 12-4. A/D Converter Sampling and A/D Conversion Timing**




**Note** For details of wait period, refer to **CHAPTER 31 CAUTIONS FOR WAIT**.

**(2) 10-bit A/D conversion result register (ADCR)**

This register is a 16-bit register that stores the A/D conversion result. The higher 6 bits are fixed to 0. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register. The higher 2 bits of the conversion result are stored in FF09H and the lower 8 bits of the conversion result are stored in FF08H.

ADCR can be read by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Figure 12-5. Format of 10-Bit A/D Conversion Result Register (ADCR)**

Address: FF08H, FF09H After reset: 0000H R

| Symbol | FF09H | | | | | | | | FF08H | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADCR | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | |

**Cautions 1. When writing to the A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), and A/D port configuration registers 0, 1 (ADPC0, ADPC1), the contents of ADCR may become undefined. Read the conversion result following conversion completion before writing to ADM0, ADS, ADPC0, and ADPC1. Using timing other than the above may cause an incorrect conversion result to be read.**

**2. If data is read from ADCR, a wait cycle is generated. Do not read data from ADCR when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(3) 8-bit A/D conversion result register L (ADCRL)**

This register is an 8-bit register that stores the A/D conversion result. The lower 8 bits of 10-bit resolution are stored.

ADCRL can be read by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 12-6. Format of 8-Bit A/D Conversion Result Register L (ADCRL)**

Address: FF08H After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADCRL | | | | | | | | |

**Cautions 1. When writing to the A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), and A/D port configuration registers 0, 1 (ADPC0, ADPC1), the contents of ADCRL may become undefined. Read the conversion result following conversion completion before writing to ADM0, ADS, ADPC0, and ADPC1. Using timing other than the above may cause an incorrect conversion result to be read.**

**2. If data is read from ADCRL, a wait cycle is generated. Do not read data from ADCRL when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(4) 8-bit A/D conversion result register H (ADCRH)**

This register is an 8-bit register that stores the A/D conversion result. The higher 8 bits of 10-bit resolution are stored.

ADCRH can be read by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 12-7. Format of 8-Bit A/D Conversion Result Register (ADCRH)**

Address: FF0DH After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADCRH | | | | | | | | |

**Cautions 1. When writing to the A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), and A/D port configuration registers 0, 1 (ADPC0, ADPC1), the contents of ADCRH may become undefined. Read the conversion result following conversion completion before writing to ADM0, ADS, ADPC0, and ADPC1. Using timing other than the above may cause an incorrect conversion result to be read.**

**2. If data is read from ADCRH, a wait cycle is generated. Do not read data from ADCRH when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

### (5) Analog input channel specification register (ADS)

This register specifies the input channel of the analog voltage to be A/D converted.

ADS can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

<R> **Remark** A/D converter analog input pins differ depending on products.

- 78K0/KY2-L: ANI0 to ANI3
- 78K0/KA2-L (20-pin products): ANI0 to ANI5
- 78K0/KA2-L (25-pin products): ANI0 to ANI6
- 78K0/KA2-L (32-pin products): ANI0 to ANI10
- 78K0/KB2-L: ANI0 to ANI3, ANI8 to ANI10
- 78K0/KC2-L (40-pin product): ANI0 to ANI6, ANI8 to ANI10
- 78K0/KC2-L (44-pin and 48-pin products): ANI0 to ANI10

**Figure 12-8. Format of Analog Input Channel Specification Register (ADS)**

Address: FF0EH After reset: 00H R/W

| Symbol | 7 | <6> | 5 | 4 | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| <R> ADS | 0 | ADOAS | 0 | 0 | ADS3 | ADS2 | ADS1 | ADS0 |

| ADOAS | ADS3 | ADS2 | ADS1 | ADS0 | Analog input channel | Input source |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | ANI0 | P20/ANI0 pin |
| 0 | 0 | 0 | 0 | 1 | ANI1 | P21/ANI1 pin or operational amplifier 0 output signal[Note] |
| 0 | 0 | 0 | 1 | 0 | ANI2 | P22/ANI2 pin |
| 0 | 0 | 0 | 1 | 1 | ANI3 | P23/ANI3 pin |
| 0 | 0 | 1 | 0 | 0 | ANI4 | P24/ANI4 pin |
| 0 | 0 | 1 | 0 | 1 | ANI5 | P25/ANI5 pin |
| 0 | 0 | 1 | 1 | 0 | ANI6 | P26/ANI6 pin |
| 0 | 0 | 1 | 1 | 1 | ANI7 | P27/ANI7 pin |
| 0 | 1 | 0 | 0 | 0 | ANI8 | P10/ANI8 pin or P70/ANI8 pin |
| 0 | 1 | 0 | 0 | 1 | ANI9 | P11/ANI9 pin or P71/ANI9 pin or operational amplifier 1 output signal[Note] |
| 0 | 1 | 0 | 1 | 0 | ANI10 | P12/ANI10 pin or P72/ANI10 pin |
| 1 | × | × | × | × | PGAOUT[Note] | PGA output signal[Note] |
| Other than above | | | | | Setting prohibited | |

**Note** Setting permitted in products with operational amplifier

**Cautions 1. Be sure to clear bits 4, 5, and 7 to "0".**

**2. Set a channel to be used for A/D conversion in the input mode by using port mode registers 1, 2, 7 (PM1, PM2, PM7).**

**3. Set ADS after PGA operation setting when selecting the PGA output signal as analog input. Set ADS after single AMP operation setting when selecting the operational amplifier output signal as analog input (refer to CHAPTER 13 OPERATIONAL AMPLIFIERS).**

**4. If data is written to ADS, a wait cycle is generated. Do not write data to ADS when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(6) A/D port configuration registers 0, 1[Note] (ADPC0, ADPC1[Note])**

ADPC0 switches the P20/AMP0-/ANI0 to P27/ANI7 pins to digital I/O or analog I/O of port. Each bit of ADPC0 corresponds to a pin of port 2 and can be specified in 1-bit units.

ADPC1 switches the P10/AMP1-/ANI8 to P12/AMP1+/ANI10 or P70/ANI8 to P72/ANI10 pins to digital I/O or analog I/O of port. Each bit of ADPC1 corresponds to a pin of P10 to P12 in port 1 or P70 to P72 in port7 and can be specified in 1-bit units.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

<R> Reset signal generation clears ADPC0 to 00H, sets ADPC1 of 78K0/KA2-L (32-pin products) to 00H, and sets ADPC1 of 78K0/KB2-L and 78K0/KC2-L to 07H.

**Note** 78K0/KA2-L (32-pin products), 78K0/KB2-L, and 78K0/KC2-L only

<R> **Figure 12-9. Format of A/D Port Configuration Registers 0, 1 (ADPC0, ADPC1) (1/3)**

**(a) 78K0/KY2-L**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(b) 78K0/KA2-L (20-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(c) 78K0/KA2-L (25-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

<R> **Figure 12-9. Format of A/D Port Configuration Registers 0, 1 (ADPC0, ADPC1) (2/3)**

**(d) 78K0/KA2-L (32-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

**(e) 78K0/KB2-L**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

**(f) 78K0/KC2-L (40-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

<R> **Figure 12-9. Format of A/D Port Configuration Registers 0, 1 (ADPC0, ADPC1) (3/3)**

**(g) 78K0/KC2-L (44-pin and 48-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

| ADPCSn | Digital I/O or analog I/O selection (n = 0 to 10) |
|---|---|
| 0 | Analog I/O |
| 1 | Digital I/O |

**Cautions 1. Set the pin set to analog I/O to the input mode by using port mode registers 1, 2, 7 (PM1, PM2, PM7).**

**2. If data is written to ADPC0 and ADPC1, a wait cycle is generated. Do not write data to ADPC0 and ADPC1 when the peripheral hardware clock is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(7) Port mode registers 1, 2, 7 (PM1, PM2, PM7)**

<R> When using the ANI8/AMP1-/P10 to ANI10/AMP1+/P12, ANI0/AMP0-/P20 to ANI7/P27, and ANI8/P70 to ANI10/P72 pins for analog input port, set PM10 to PM12, PM20 to PM27, PM70 to PM72 to 1. The output latches of P10 to P12, P20 to P27, and P70 to P72 at this time may be 0 or 1.

If PM10 to PM12, PM20 to PM27, and P70 to P72 are set to 0, they cannot be used as analog input port pins.

PM1, PM2, and PM7 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Remark** A/D converter analog input pins differ depending on products.

- 78K0/KY2-L: ANI0 to ANI3
- 78K0/KA2-L (20-pin products): ANI0 to ANI5
- 78K0/KA2-L (25-pin products): ANI0 to ANI6
- 78K0/KA2-L (32-pin products): ANI0 to ANI10
- 78K0/KB2-L: ANI0 to ANI3, ANI8 to ANI10
- 78K0/KC2-L (40-pin product): ANI0 to ANI6, ANI8 to ANI10
- 78K0/KC2-L (44-pin and 48-pin products): ANI0 to ANI10

**Figure 12-10. Format of Port Mode Register 1 (PM1) (78K0/KB2-L, 78K0/KC2-L)**

Address: FF21H After reset: FFH R/W

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

| PM1n | P1n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Figure 12-11. Format of Port Mode Register 2 (PM2)**

**(a) 78K0/KY2-L**

Address: FF22H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM2 | 1 | 1 | 1 | 1 | PM23 | PM22 | PM21 | PM20 |

<R> **(b) 78K0/KA2-L**

Address: FF22H After reset: FFH R/W

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

**Notes 1.** 32-pin products only

**2.** 25-pin and 32-pin products only

**(c) 78K0/KB2-L**

Address: FF22H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM2 | 1 | 1 | 1 | 1 | PM23 | PM22 | PM21 | PM20 |

<R> **(d) 78K0/KC2-L**

Address: FF22H After reset: FFH R/W

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

**Note** 44-pin and 48-pin products only

| PM2n | P2n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

<R> **Figure 12-12. Format of Port Mode Register 7 (PM7) (78K0/KA2-L (32-pin products))**

Address: FF27H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM7 | 1 | 1 | 1 | 1 | 1 | PM72 | PM71 | PM70 |

| PM7n | P7n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

When using P10/ANI8/AMP1-, P11/ANI9/AMP1OUT, or P12/ANI10/AMP1+ in the 78K0/KB2-L and 78K0/KC2-L, set the registers according to the pin function to be used (refer to **Tables 12-3** and **12-4**).

**Table 12-3. Setting Functions of P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins**

| ADPC1 Register | PM1 Register | OPAMP1E bit | ADS Register (n = 8, 10) | P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins |
|---|---|---|---|---|
| Analog input selection | Input mode | 0 | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Operational amplifier 1 input |
| | Output mode | – | – | Setting prohibited |
| Digital I/O selection | Input mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital input |
| | Output mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital output |

**Table 12-4. Setting Functions of P11/ANI9/AMP1OUT Pin**

| ADPC1 Register | PM1 Register | OPAMP1E bit | ADS Register | P11/ANI9/AMP1OUT Pin |
|---|---|---|---|---|
| Analog I/O selection | Input mode | 0 | Selects ANI9. | Analog input (to be converted into digital signals) |
| | | | Does not select ANI9. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANI9. | Operational amplifier 1 output (to be converted into digital signals) |
| | | | Does not select ANI9. | Operational amplifier 1 output (not to be converted into digital signals) |
| | Output mode | – | – | Setting prohibited |
| Digital I/O selection | Input mode | 0 | Selects ANI9. | Setting prohibited |
| | | | Does not select ANI9. | Digital input |
| | | 1 | – | Setting prohibited |
| | Output mode | 0 | Selects ANI9. | Setting prohibited |
| | | | Does not select ANI9. | Digital output |
| | | 1 | – | Setting prohibited |

**Remark** ADPC1: A/D port configuration register 1
PM1: Port mode register 1
OPAMP1E: Bit 7 of operational amplifier 1 control register (AMP1M)
ADS: Analog input channel specification register

When using P20/AMP0-/ANI0 to P27/ANI7, set the registers according to the pin function to be used (refer to **Tables 12-5** to **12-7**).

**Table 12-5. Setting Functions of P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins**

| ADPC0 Register | PM2 Register | OPAMP0E bit | ADS Register (n = 0, 2) | P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins |
|---|---|---|---|---|
| Analog input selection | Input mode | 0 | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Operational amplifier 0 input |
| | Output mode | – | – | Setting prohibited |
| Digital I/O selection | Input mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital input |
| | Output mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital output |

**Remark** ADPC0: A/D port configuration register 0
PM2: Port mode register 2
OPAMP0E: Bit 7 of operational amplifier 0 control register (AMP0M)
ADS: Analog input channel specification register

<R> **Table 12-6. Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin**

| ADPC0 Register | PM2 Register | OPAMP0E bit | PGAEN bit | ADS Register | P21/ANI1/AMP0OUT/PGAIN Pin |
|---|---|---|---|---|---|
| Analog I/O selection | Input mode | 0 | 0 | Selects ANI1. | Analog input (to be converted into digital signals) |
| | | | | Does not select ANI1. | Analog input (not to be converted into digital signals) |
| | | 0 | 1 | Selects PGAOUT. | PGA input (PGA output is converted into digital signals) |
| | | | | Selects ANI1. | PGA input (to be converted into digital signals) |
| | | | | Does not select PGAOUT and ANI1. | PGA input (not to be converted into digital signals) |
| | | 1 | 0 | Selects ANI1. | Operational amplifier 0 output (to be converted into digital signals) |
| | | | | Does not select ANI1. | Operational amplifier 0 output (not to be converted into digital signals) |
| | | 1 | 1 | Selects PGAOUT. | Operational amplifier 0 output and PGA input (PGA output is converted into digital signals) |
| | | | | Selects ANI1. | Operational amplifier 0 output (to be converted into digital signals) |
| | | | | Does not select PGAOUT and ANI1. | Operational amplifier 0 output (not to be converted into digital signals) |
| | Output mode | – | | – | Setting prohibited |
| Digital I/O selection | Input mode | 0 | – | Selects ANI1. | Setting prohibited |
| | | | | Does not select ANI1. | Digital input |
| | | 1 | – | – | Setting prohibited |
| | Output mode | 0 | – | Selects ANI1. | Setting prohibited |
| | | | | Does not select ANI1. | Digital output |
| | | 1 | – | – | Setting prohibited |

**Table 12-7. Setting Functions of P23/ANI3 to P27/ANI7 Pins**

| ADPC0 Register | PM2 Register | ADS Register(n = 3 to 7) | P23/ANI3 to P27/ANI7 Pins |
|---|---|---|---|
| Analog input selection | Input mode | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | Output mode | – | Setting prohibited |
| Digital I/O selection | Input mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital input |
| | Output mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital output |

**Remark** ADPC0: A/D port configuration register 0
PM2: Port mode register 2
OPAMP0E: Bit 7 of operational amplifier 0 control register (AMP0M)
PGAEN: Bit 6 of AMP0M
ADS: Analog input channel specification register

When using P70/ANI8 to P72/ANI10 of 78K0/KA2-L (32-pin products), set the registers according to the pin function to be used (refer to **Table 12-8**).

<R> **Table 12-8. Setting Functions of P70/ANI8 to P72/ANI10 Pins**

| ADPC1 Register | PM7 Register | ADS Register(n = 8 to 10) | P70/ANI8 to P72/ANI10 Pins |
|---|---|---|---|
| Analog input selection | Input mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital input |
| | Output mode | Selects ANIn. | Setting prohibited |
| | | Does not select ANIn. | Digital output |
| Digital I/O selection | Input mode | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | Output mode | – | Setting prohibited |

**Remark** ADPC1: A/D port configuration register 1
PM7: Port mode register 7
ADS: Analog input channel specification register

## 12.4 A/D Converter Operations

### 12.4.1 Basic operations of A/D converter

<1> Set the A/D conversion time and the operation mode by using bits 5 to 1 (FR2 to FR0, LV1, and LV0) of the A/D converter mode register 0 (ADM0).

<2> Set bit 0 (ADCE) of ADM0 to 1 to start the operation of the A/D voltage comparator.

<3> Set channels for A/D conversion to analog I/O by using the A/D port configuration registers 0, 1 (ADPC0, ADPC1) and set to input mode by using port mode registers 1, 2 (PM1, PM2).

<4> Set the PGA operation to set the PGA output or the single AMP operation to set the operational amplifier output for analog input. (refer to **CHAPTER 13 OPERATIONAL AMPLIFIERS**).

<5> Select one channel for A/D conversion by using the analog input channel specification register (ADS).

<6> Start the conversion operation by setting bit 7 (ADCS) of ADM0 to 1.
(<7> to <14> are operations performed by hardware.)

<7> The voltage input to the selected analog input channel is sampled by the sample & hold circuit.

<8> When sampling has been done for a certain time, the sample & hold circuit is placed in the hold state and the sampled voltage is held until the A/D conversion operation has ended.

<9> Bit 9 of the successive approximation register (SAR) is set. The comparison voltage generator outputs (1/2) $AV_{REF}$ voltage.

<10> The voltage difference between the output voltage of the comparison voltage generator and sampled voltage is compared by the voltage comparator. If the analog input is greater than (1/2) $AV_{REF}$, the MSB of SAR remains set to 1. If the analog input is smaller than (1/2) $AV_{REF}$, the MSB is reset to 0.

<11> Next, bit 8 of SAR is automatically set to 1, and the operation proceeds to the next comparison. The output voltage of the comparison voltage generator is selected according to the preset value of bit 9, as described below.

- Bit 9 = 1: (3/4) $AV_{REF}$
- Bit 9 = 0: (1/4) $AV_{REF}$

The output voltage of the comparison voltage generator and sampled voltage are compared and bit 8 of SAR is manipulated as follows.

- Analog input voltage ≥ Output voltage of comparison voltage generator: Bit 8 = 1
- Analog input voltage < Output voltage of comparison voltage generator: Bit 8 = 0

<12> Comparison is continued in this way up to bit 0 of SAR.

<13> Upon completion of the comparison of 10 bits, an effective digital result value remains in SAR, and the result value is transferred to the A/D conversion result register (ADCR, ADCRH, ADCRL) and then latched.
At the same time, the A/D conversion end interrupt request (INTAD) can also be generated.

<14> Repeat steps <7> to <13>, until ADCS is cleared to 0.
To stop the A/D converter, clear ADCS to 0.
To restart A/D conversion from the status of ADCE = 1, start from <6>. To start A/D conversion again when ADCE = 0, set ADCE to 1, wait for 1 $\mu$s or longer, and start <6>. To change a channel of A/D conversion, start from <5>.

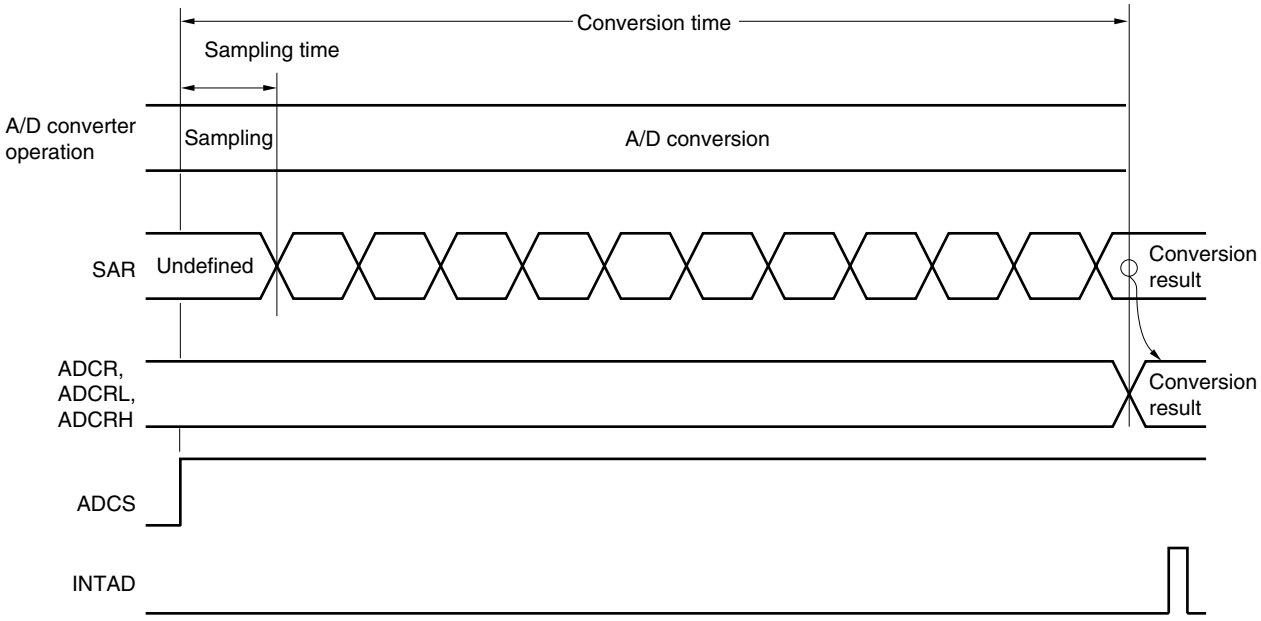
**Cautions 1. Make sure the period of <2> to <6> is 1 μs or more.**

**2. If the timing of <2> is earlier than that of <4>, <2> may be performed any time.**

**Remark** Three types of A/D conversion result registers are available.

- ADCR (16 bits): Store 10-bit A/D conversion value
- ADCRH (8 bits): Store the higher 8-bit A/D conversion value
- ADCRL (8 bits): Store the lower 8-bit A/D conversion value

**Figure 12-13. Basic Operation of A/D Converter**




A/D conversion operations are performed continuously until bit 7 (ADCS) of the A/D converter mode register 0 (ADM0) is reset (0) by software.

If a write operation is performed to the analog input channel specification register (ADS) during an A/D conversion operation, the conversion operation is initialized, and if the ADCS bit is set (1), conversion starts again from the beginning.

Reset signal generation clears the A/D conversion result register (ADCR, ADCRH, ADCRL) to 0000H or 00H.

### 12.4.2 Input voltage and conversion results

The relationship between the analog input voltage input to the analog input pins (ANI0 to ANI10) and the theoretical A/D conversion result (stored in the 10-bit A/D conversion result register (ADCR)) is shown by the following expression.

$$ADCR = INT\left(\frac{V_{AIN}}{AV_{REF}} \times 1024 + 0.5\right)$$

or

$$(ADCR - 0.5) \times \frac{AV_{REF}}{1024} \leq V_{AIN} < (ADCR + 0.5) \times \frac{AV_{REF}}{1024}$$

where, INT( ): Function which returns integer part of value in parentheses

$V_{AIN}$: Analog input voltage

$AV_{REF}$: $AV_{REF}$ pin voltage

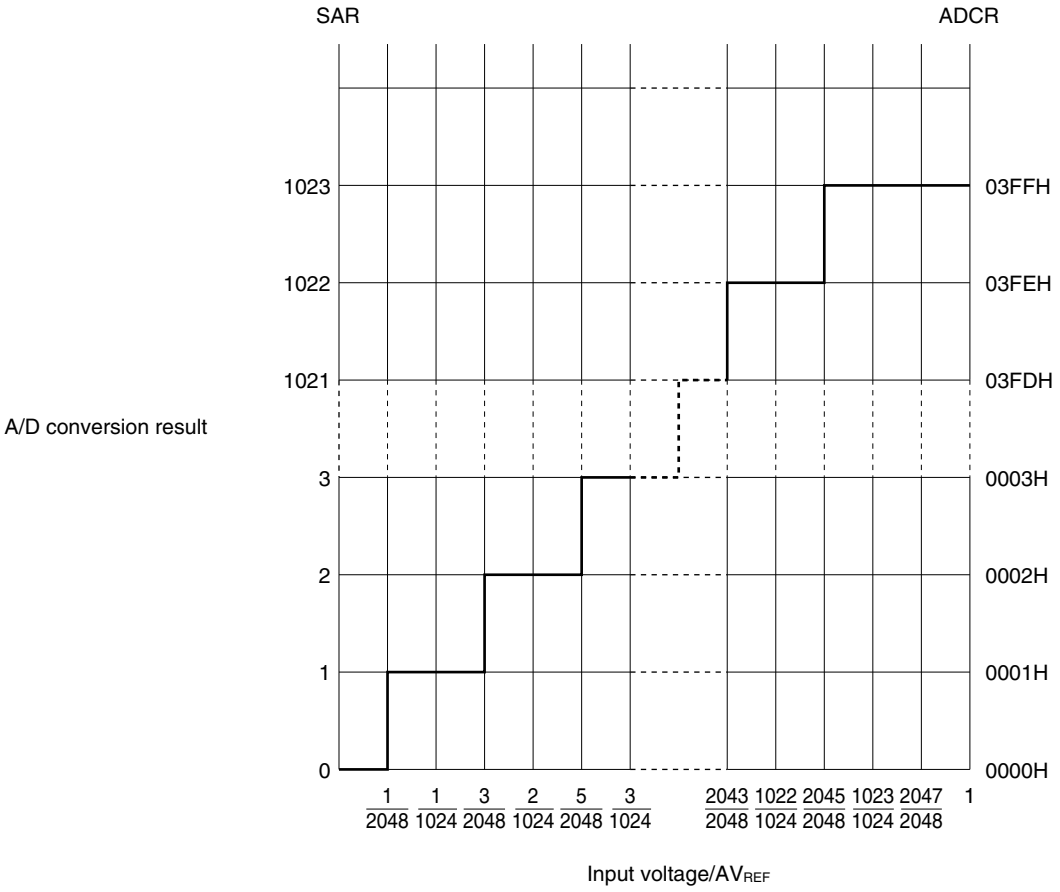
ADCR: 10-bit A/D conversion result register (ADCR) value

<R> **Remark** A/D converter analog input pins differ depending on products.

- 78K0/KY2-L: ANI0 to ANI3
- 78K0/KA2-L (20-pin products): ANI0 to ANI5
- 78K0/KA2-L (25-pin products): ANI0 to ANI6
- 78K0/KA2-L (32-pin products): ANI0 to ANI10
- 78K0/KB2-L: ANI0 to ANI3, ANI8 to ANI10
- 78K0/KC2-L (40-pin product): ANI0 to ANI6, ANI8 to ANI10
- 78K0/KC2-L (44-pin and 48-pin products): ANI0 to ANI10

Figure 12-14 shows the relationship between the analog input voltage and the A/D conversion result.

**Figure 12-14. Relationship between Analog Input Voltage and A/D Conversion Result**



### 12.4.3 A/D converter operation mode

One channel of analog input is selected from ANI0 to ANI10 and PGA output by the analog input channel specification register (ADS) and A/D conversion is executed.

<R> **Remark** A/D converter analog input pins differ depending on products.

- 78K0/KY2-L: ANI0 to ANI3
- 78K0/KA2-L (20-pin products): ANI0 to ANI5
- 78K0/KA2-L (25-pin products): ANI0 to ANI6
- 78K0/KA2-L (32-pin products): ANI0 to ANI10
- 78K0/KB2-L: ANI0 to ANI3, ANI8 to ANI10
- 78K0/KC2-L (40-pin product): ANI0 to ANI6, ANI8 to ANI10
- 78K0/KC2-L (44-pin and 48-pin products): ANI0 to ANI10

#### (1) A/D conversion operation

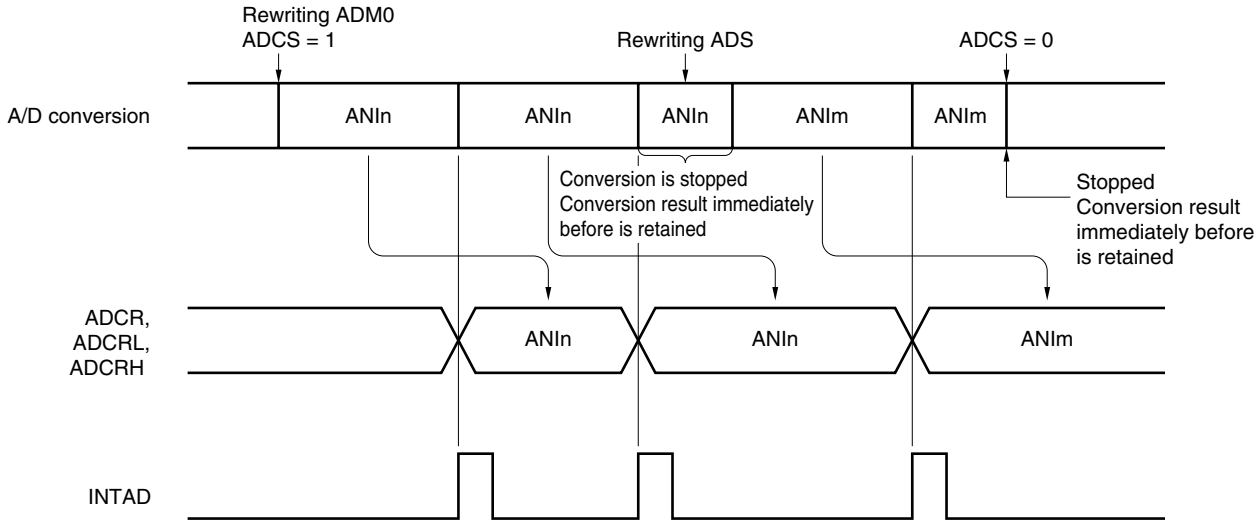
By setting bit 7 (ADCS) of the A/D converter mode register 0 (ADM0) to 1, the A/D conversion operation of the voltage, which is applied to the analog input pin specified by the analog input channel specification register (ADS), is started.

When A/D conversion has been completed, the result of the A/D conversion is stored in the A/D conversion result register (ADCR, ADCRH, ADCRL), and an interrupt request signal (INTAD) is generated. When one A/D conversion has been completed, the next A/D conversion operation is immediately started.

If ADS is rewritten during A/D conversion, the A/D conversion operation under execution is stopped and restarted from the beginning.

If 0 is written to ADCS during A/D conversion, A/D conversion is immediately stopped. At this time, the conversion result immediately before is retained.

**Figure 12-15. A/D Conversion Operation**




**Remarks 1.** n = 0 to 10 (it depends on products)

**2.** m = 0 to 10 (it depends on products)

The setting methods are described below.

<1> Set the A/D conversion time and the operation mode by using bits 5 to 1 (FR2 to FR0, LV1, and LV0) of the A/D converter mode register 0 (ADM0).
<2> Set bit 0 (ADCE) of ADM0 to 1.
<3> Set the channel to be used to analog input by using the A/D port configuration registers 0, 1 (ADPC0, ADPC1) and port mode registers 1, 2 (PM1, PM2).
<4> Set the PGA operation to set the PGA output or the single AMP operation to set the operational amplifier output for analog input. (refer to **CHAPTER 13 OPERATIONAL AMPLIFIERS**).
<5> Select a channel to be used by using the analog input channel specification register (ADS).
<6> Set bit 7 (ADCS) of ADM0 to 1 to start A/D conversion.
<7> When one A/D conversion has been completed, an interrupt request signal (INTAD) is generated.
<8> Transfer the A/D conversion data to the A/D conversion result register (ADCR, ADCRH, ADCRL).

<Change the channel>

<9> Set bit 0 (ADMK) of the interrupt mask flag register 1L (MK1L) to 1[Note].
<10> Change the channel by using ADS to start A/D conversion.
<11> Clear bit 0 (ADIF) of the interrupt request flag register 1L (IF1L) to 0.
<12> Clear ADMK to 0[Note].
<13> When one A/D conversion has been completed, an interrupt request signal (INTAD) is generated.
<14> Transfer the A/D conversion data to the A/D conversion result register (ADCR, ADCRH, ADCRL).

<Complete A/D conversion>

<15> Clear ADCS to 0.
<16> Clear ADCE to 0.

**Note** Execute this only if interrupt servicing is used for A/D conversion.

**Cautions 1. Make sure the period of <2> to <6> is 1 μs or more.**
**2. If the timing of <2> is earlier than that of <4>, <2> may be performed any time.**
**3. <2> can be omitted. However, ignore data of the first conversion after <6> in this case.**
**4. The period from <7> to <13> differs from the conversion time set using bits 5 to 1 (FR2 to FR0, LV1, LV0) of ADM0. The period from <10> to <13> is the conversion time set using FR2 to FR0, LV1, and LV0.**

## 12.5 How to Read A/D Converter Characteristics Table

Here, special terms unique to the A/D converter are explained.

**(1) Resolution**

This is the minimum analog input voltage that can be identified. That is, the percentage of the analog input voltage per bit of digital output is called 1LSB (Least Significant Bit). The percentage of 1LSB with respect to the full scale is expressed by %FSR (Full Scale Range).

1LSB is as follows when the resolution is 10 bits.

$1LSB = 1/2^{10} = 1/1024$
$= 0.098\%FSR$

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, integral linearity error, and differential linearity errors that are combinations of these express the overall error.
Note that the quantization error is not included in the overall error in the characteristics table.

**(3) Quantization error**

When analog values are converted to digital values, a ±1/2LSB error naturally occurs. In an A/D converter, an analog input voltage in a range of ±1/2LSB 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 12-16. Overall Error**

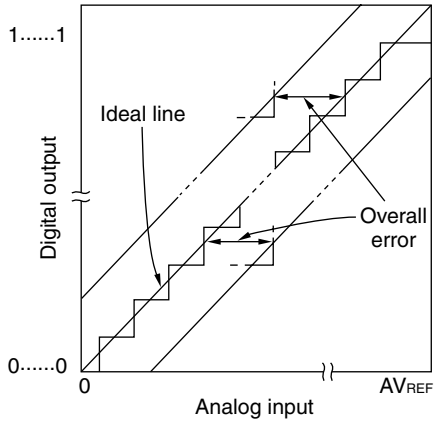
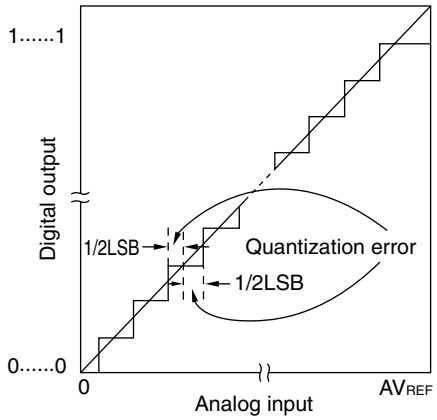



**Figure 12-17. Quantization Error**




**(4) Zero-scale error**

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (1/2LSB) when the digital output changes from 0.....000 to 0.....001.
If the actual measurement value is greater than the theoretical value, it shows the difference between the actual measurement value of the analog input voltage and the theoretical value (3/2LSB) when the digital output changes from 0……001 to 0……010.

**(5) Full-scale error**

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (Full-scale – 3/2LSB) when the digital output changes from 1.....110 to 1.....111.

**(6) Integral linearity error**

This shows the degree to which the conversion characteristics deviate from the ideal linear relationship. It expresses the maximum value of the difference between the actual measurement value and the ideal straight line when the zero-scale error and full-scale error are 0.

**(7) Differential linearity error**

While the ideal width of code output is 1LSB, this indicates the difference between the actual measurement value and the ideal value.

**Figure 12-18. Zero-Scale Error**

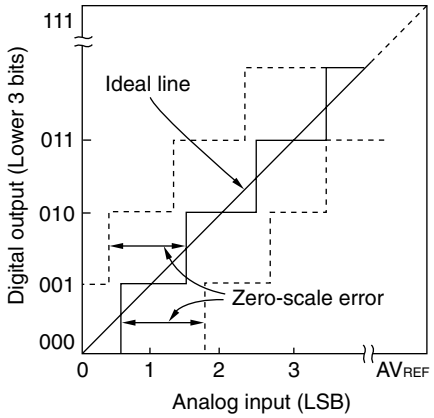



**Figure 12-19. Full-Scale Error**

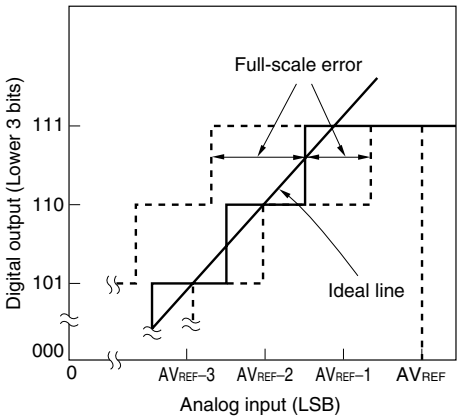



**Figure 12-20. Integral Linearity Error**

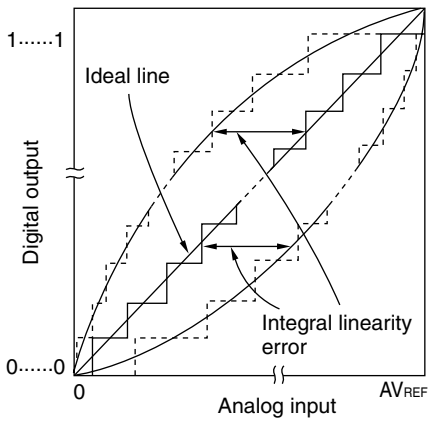
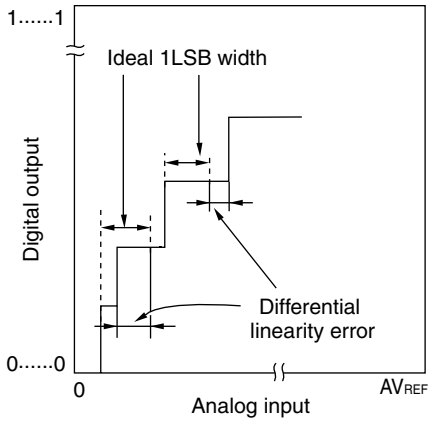



**Figure 12-21. Differential Linearity Error**



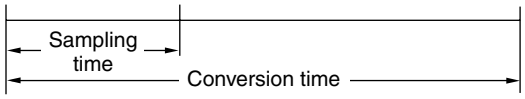

**(8) Conversion time**

This expresses the time from the start of sampling to when the digital output is obtained.

The sampling time is included in the conversion time in the characteristics table.

**(9) Sampling time**

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.



## 12.6 Cautions for A/D Converter

**(1) Operating current in STOP mode**

To satisfy the DC characteristics of the power supply current in STOP mode, clear bits 7 (ADCS) and 0 (ADCE) of A/D converter mode register 0 (ADM0) to 0 before executing a STOP instruction.

To restart from the standby status, clear bit 0 (ADIF) of interrupt request flag register 1L (IF1L) to 0 and start operation.

**(2) Input range of ANI0 to ANI10**

Observe the rated range of the ANI0 to ANI10 input voltage. If a voltage of $AV_{REF}$ or higher and $AV_{SS}$ or lower (even in the range of absolute maximum ratings) is input to an analog input channel, the converted value of that channel becomes undefined. In addition, the converted values of the other channels may also be affected.

**(3) Conflicting operations**

<1> Conflict between A/D conversion result register (ADCR, ADCRL, ADCRH) write and ADCR, ADCRL, or ADCRH read by instruction upon the end of conversion

ADCR, ADCRL, or ADCRH read has priority. After the read operation, the new conversion result is written to ADCR, ADCRL, or ADCRH.

<2> Conflict between ADCR, ADCRL, or ADCRH write and A/D converter mode register 0 (ADM0) write, analog input channel specification register (ADS), or A/D port configuration registers 0, 1 (ADPC0, ADPC1) write upon the end of conversion

ADM0, ADS, ADPC0, or ADPC1 write has priority. ADCR, ADCRL, or ADCRH write is not performed, nor is the conversion end interrupt signal (INTAD) generated.

**(4) Noise countermeasures**

To maintain the 10-bit resolution, attention must be paid to noise input to the $AV_{REF}$ pin and pins ANI0 to ANI10.

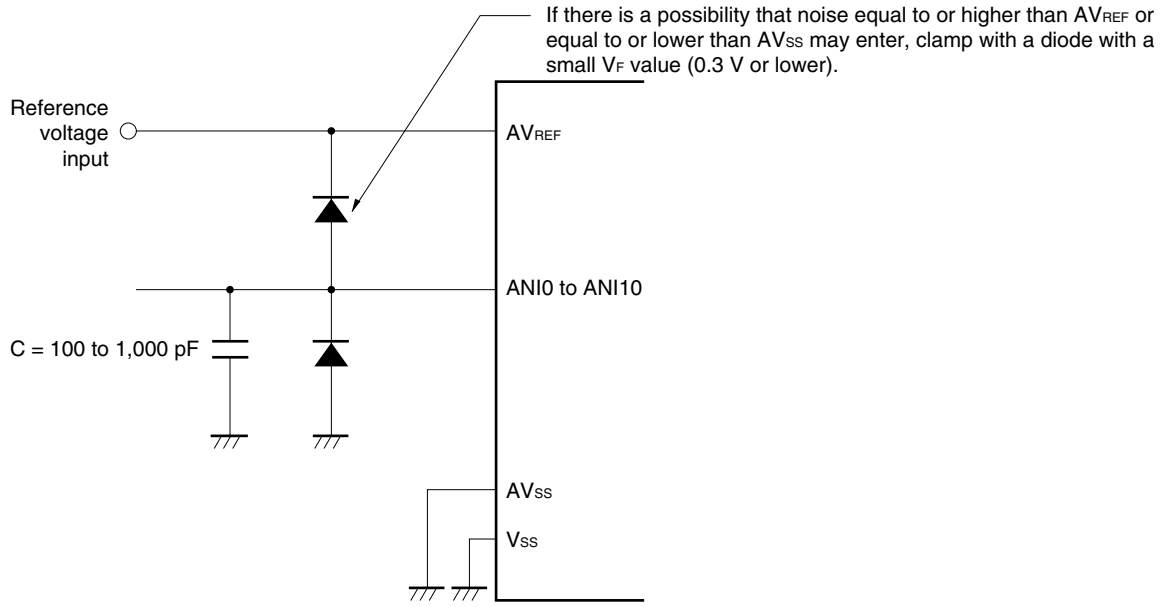
<1> Connect a capacitor with a low equivalent resistance and a good frequency response to the power supply.

<2> The higher the output impedance of the analog input source, the greater the influence. To reduce the noise, connecting external C as shown in Figure 12-22 is recommended.

<3> Do not switch these pins with other pins during conversion.

<4> The accuracy is improved if the HALT mode is set immediately after the start of conversion.

**Figure 12-22. Analog Input Pin Connection**




**(5) ANI0/P20 to ANI7/P27, ANI8/P10 to ANI10/P12, and ANI8/P70 to ANI10/P72**

<1> The analog input pins (ANI0 to ANI7 and ANI8 to ANI10) are also used as digital I/O port pins (P20 to P27 and P10 to P12). When A/D conversion is performed with any of ANI0 to ANI7 and ANI8 to ANI10 selected, do not access P20 to P27 while conversion is in progress; otherwise the conversion resolution may be degraded.

<2> To use the ANI0/P20 to ANI7/P27 and ANI8/P10 to ANI10/P12 pins for digital I/O port, it is recommended to use starting with the furthest pin from AVREF (for example, the ANI0/P20 pin in the 78K0/KC2-L). To use these pins as analog input, it is recommended to use starting with the closest pin to AVSS (for example, the ANI7/P27 pin in the 78K0/KC2-L (44-pin and 48-pin products)).

<3> If a digital pulse is applied to the pins adjacent to the pins currently used for A/D conversion, the expected value of the A/D conversion may not be obtained due to coupling noise. Therefore, do not apply a pulse to the pins adjacent to the pin undergoing A/D conversion.

**(6) Input impedance of ANI0 to ANI10 pins**

This A/D converter charges a sampling capacitor for sampling during sampling time.

Therefore, only a leakage current flow when sampling is not in progress, and a current that charges the capacitor flows during sampling. Consequently, the input impedance fluctuates depending on whether sampling is in progress, and on the other states.

To make sure that sampling is effective, however, it is recommended to keep the output impedance of the analog input source to within 10 kΩ, and to connect a capacitor of about 100 pF to the ANI0 to ANI10 pins (refer to **Figure 12-22**).

**(7) AVREF pin input impedance**

A series resistor string of several tens of kΩ is connected between the AVREF and AVSS pins.

Therefore, if the output impedance of the reference voltage source is high, this will result in a series connection to the series resistor string between the AVREF and AVSS pins, resulting in a large reference voltage error.

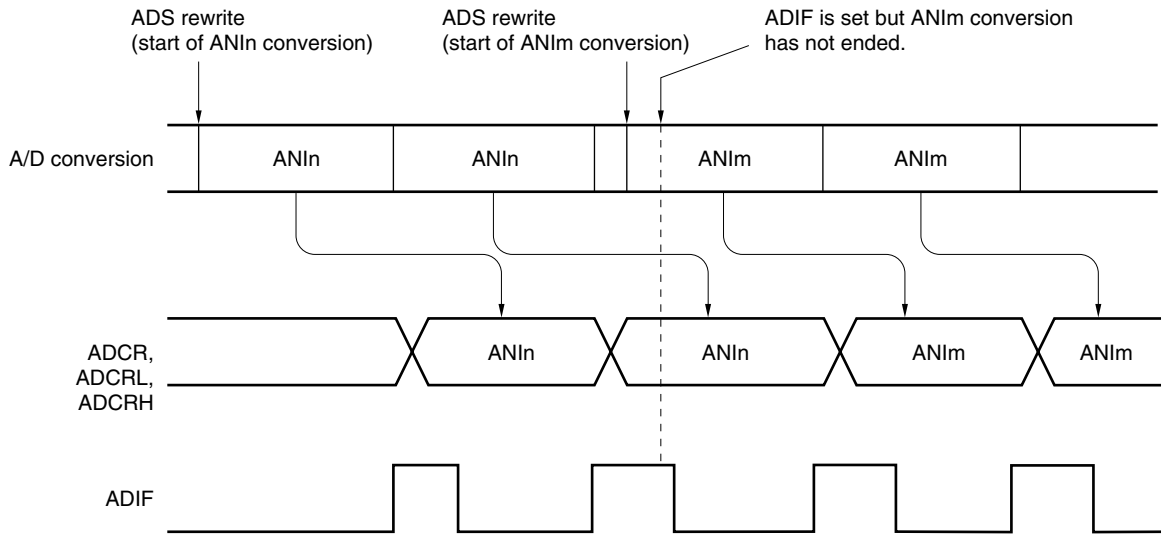
**(8) Interrupt request flag (ADIF)**

The interrupt request flag (ADIF) is not cleared even if the analog input channel specification register (ADS) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and ADIF for the pre-change analog input may be set just before the ADS rewrite. Caution is therefore required since, at this time, when ADIF is read immediately after the ADS rewrite, ADIF is set despite the fact A/D conversion for the post-change analog input has not ended.

When A/D conversion is stopped and then resumed, clear ADIF before the A/D conversion operation is resumed.

**Figure 12-23. Timing of A/D Conversion End Interrupt Request Generation**




**Remarks 1.** n = 0 to 10 (it depends on products)

**2.** m = 0 to 10 (it depends on products)

**(9) Conversion results just after A/D conversion start**

The first A/D conversion value immediately after A/D conversion starts may not fall within the rating range if the ADCS bit is set to 1 within 1 $\mu$s after the ADCE bit was set to 1, or if the ADCS bit is set to 1 with the ADCE bit = 0. Take measures such as polling the A/D conversion end interrupt request (INTAD) and removing the first conversion result.

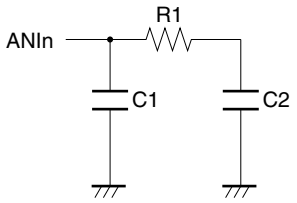
**(10) A/D conversion result register (ADCR, ADCRL, ADCRH) read operation**

When a write operation is performed to the A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), and A/D port configuration registers 0, 1 (ADPC0, ADPC1), the contents of ADCR, ADCRL, and ADCRH may become undefined. Read the conversion result following conversion completion before writing to ADM0, ADS, ADPC0, and ADPC1. Using a timing other than the above may cause an incorrect conversion result to be read.

**(11) Internal equivalent circuit**

The equivalent circuit of the analog input block is shown below.

**Figure 12-24. Internal Equivalent Circuit of ANIn Pin**




<R> **Table 12-9. Resistance and Capacitance Values of Equivalent Circuit (Reference Values)**

| $AV_{REF}$ | Mode | R1 | C1 | C2 |
|---|---|---|---|---|
| 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | Standard | 5.2 kΩ | 8 pF | 6.3 pF |
| | High-speed 2 | 7.8 kΩ | | |
| | High-speed 1 | 5.2 kΩ | | |
| 2.7 V ≤ $AV_{REF}$ < 4.0 V | Standard | 18.6 kΩ | | |
| | High-speed 2 | 7.8 kΩ | | |
| 1.8 V ≤ $AV_{REF}$ < 4.0 V | Low-voltage | 169.8 kΩ | | |

**Remarks 1.** The resistance and capacitance values shown in Table 12-9 are not guaranteed values.

**2.** n = 0 to 10 (it depends on products)

# CHAPTER 13 OPERATIONAL AMPLIFIERS

| Item | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) |
|---|---|---|---|---|
| | 16 Pins | 20, 25, 32 Pins | 30 Pins | 40, 44, 48 Pins |
| Operational amplifier (products with operational amplifier only) | 1 ch (operational amplifier 0) | | 2 ch (operational amplifiers 0 and 1) | |

## 13.1 Function of Operational Amplifier

Operational amplifiers 0 and 1 are mounted onto products with operational amplifier of the 78K0/Kx2-L microcontrollers. The operational amplifiers 0 and 1 have the following modes.

- **Single AMP mode (operational amplifiers 0 and 1)**
  Operational amplifiers 0 and 1 both have two input pins (the AMPn- pin and the AMPn+ pin) and one output pin (the AMPnOUT pin), and can be used as single-power supply amplifiers that can be externally connected.
  The amplified voltage can be used as an analog input of the A/D converter, because the AMPnOUT pin is alternatively used with analog input pin of the A/D converter.

- **PGA (Programmable gain amplifier) mode (operational amplifier 0 only)**
  In this mode, the analog voltage input from the PGAIN pin is amplified within the microcontroller. The gain can be selected from four types (×4, ×8, ×16, ×32).
  The amplified voltage can be used as an analog input of the A/D converter.

**Remark** Products with operational amplifier of the 78K0/KY2-L and 78K0/KA2-L: n = 0
Products with operational amplifier of the 78K0/KB2-L and 78K0/KC2-L: n = 0, 1

## 13.2 Configuration of Operational Amplifier

The operational amplifiers consist of the following hardware.

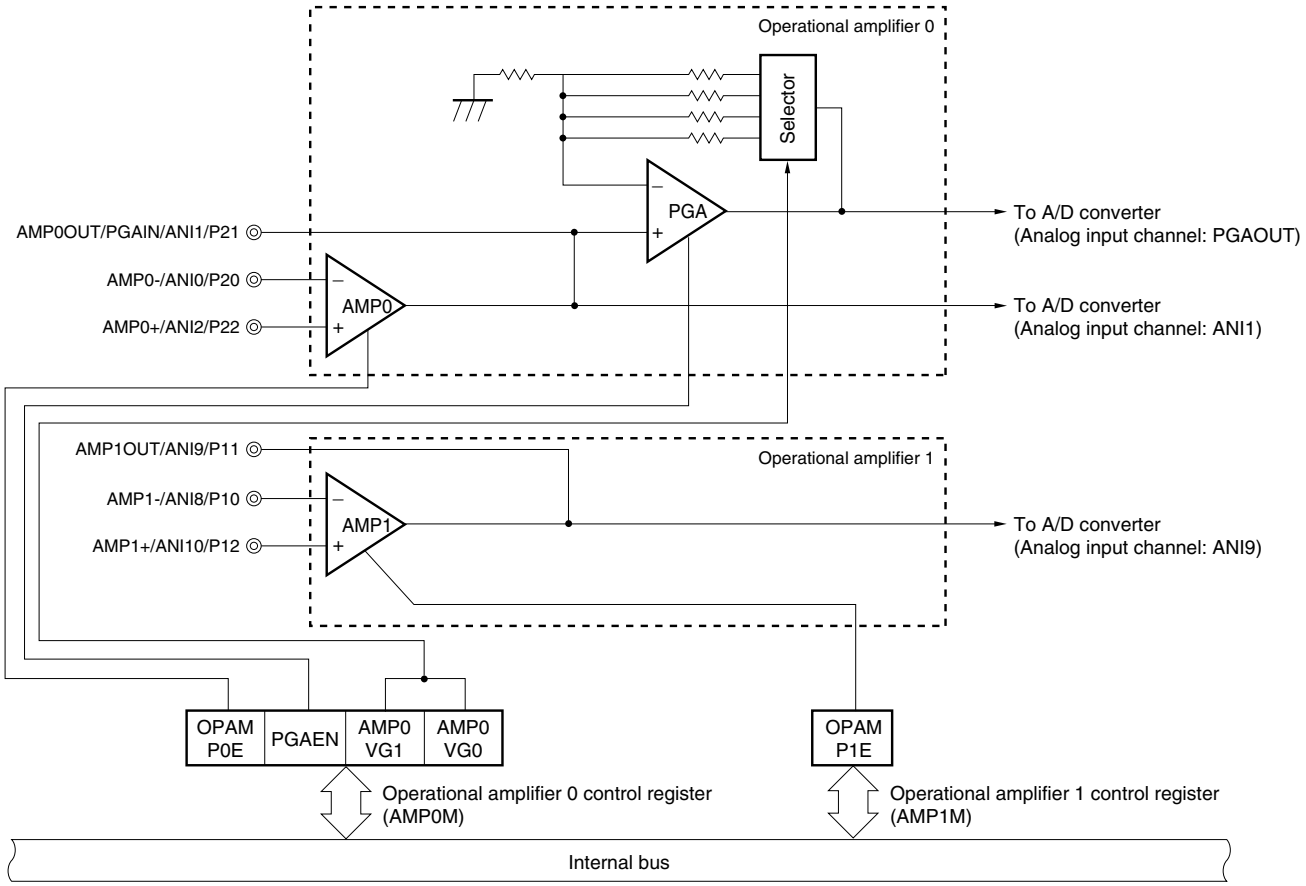
**Table 13-1. Configuration of Operational Amplifier**

| Item | Configuration |
|---|---|
| Operational amplifier input | PGAIN pin, AMPn- pin, AMPn+ pin |
| Operational amplifier output | AMPnOUT pin |
| Control registers | Operational amplifier n control register (AMPnM)<br>A/D configuration register n (ADPCn)<br>Analog input channel specification register (ADS)<br>Port mode registers 1, 2 (PM1, PM2) |

**Remark** Products with operational amplifier of the 78K0/KY2-L and 78K0/KA2-L: n = 0
Products with operational amplifier of the 78K0/KB2-L and 78K0/KC2-L: n = 0, 1

<R>

**Figure 13-1. Block Diagram of Operational Amplifier**



## 13.3 Registers Used in Operational Amplifier

The operational amplifiers use the following four registers.

- Operational amplifier 0 control register (AMP0M), operational amplifier 1 control register (AMP1M)
- A/D port configuration registers 0 and 1 (ADPC0, ADPC1)
- Analog input channel specification register (ADS)
- Port mode registers 1, 2 (PM1, PM2)

### (1) Operational amplifier 0 control register (AMP0M), operational amplifier 1 control register (AMP1M)

These registers control the operations of operational amplifiers 0 and 1.

AMP0M and AMP1M can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Remark** Products with operational amplifier of the 78K0/KY2-L and 78K0/KA2-L:
Operational amplifier 0 is mounted.
Products with operational amplifier of the 78K0/KB2-L and 78K0/KC2-L:
Operational amplifiers 0 and 1 are mounted.

**Figure 13-2. Format of Operational Amplifier 0 Control Register (AMP0M)**
**(Products with Operational Amplifier Only)**

Address: FF60H After reset: 00H R/W

| Symbol | <7> | <6> | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| AMP0M | OPAMP0E | PGAEN | 0 | 0 | 0 | 0 | AMP0VG1 | AMP0VG0 |

| OPAMP0E | PGAEN | Operational amplifier 0 operation control |
|---|---|---|
| 0 | 0 | Stops operational amplifier 0 operation |
| 0 | 1 | Enables operational amplifier 0 (PGA mode only) operation |
| 1 | 0 | Enables operational amplifier 0 (single AMP mode only) operation |
| 1 | 1 | Enables operational amplifier 0 (simultaneous operation in the PGA and single AMP modes) operation |

| AMP0VG1 | AMP0VG0 | PGA mode of operational amplifier 0 gain selection |
|---|---|---|
| 0 | 0 | ×4 |
| 0 | 1 | ×8 |
| 1 | 0 | ×16 |
| 1 | 1 | ×32 |

**Cautions 1. When using the PGA mode, use the ADPC0 register to select the PGAIN/AMP0OUT/ANI1/P21 pin as an analog I/O.**

**2. When using the single AMP mode, use the ADPC0 register to select the AMP0OUT/PGAIN/ANI1/P21, AMP0-/ANI0/P20, and AMP0+/ANI2/P22 pins as analog I/O.**

**3. When using as digital inputs the pins of port 2, which are not used with the operational amplifier 0, when the operational amplifier 0 is used, make sure that the input levels of digital input ports are fixed to prevent degradation of the A/D conversion resolution.**

**Remark** The output of operational amplifier 0 is amplified by the PGA by setting OPAMP0E = PGAEN = 1.

**Figure 13-3. Format of Operational Amplifier 1 Control Register (AMP1M)**
**(Products with Operational Amplifier of the 78K0/KB2-L and 78K0/KC2-L Only)**

Address: FF61H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| AMP1M | OPAMP1E | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| OPAMP1E | Operational amplifier 1 operation control |
|---|---|
| 0 | Stops operational amplifier 1 operation |
| 1 | Enables operational amplifier 1 (single AMP mode) operation |

**Cautions 1. When using the single AMP mode, use the ADPC1 register to select the AMP1OUT/ANI9/P11, AMP1-/ANI8/P10, and AMP1+/ANI10/P12 pins as analog I/O.**

**2. When using as digital inputs the pins of port 1, which are not used with the operational amplifier 1, when the operational amplifier 1 is used, make sure that the input levels of digital input ports are fixed to prevent degradation of the A/D conversion resolution.**

**(2) A/D port configuration registers 0 and 1 (ADPC0, ADPC1)**

ADPC0 switches the P20/AMP0-/ANI0 to P27/ANI7 pins to digital I/O or analog I/O of port. Each bit of ADPC0 corresponds to a pin of port 2 and can be specified in 1-bit units.

ADPC1 switches the P10/AMP1-/ANI8 to P12/AMP1+/ANI10 pins to digital I/O or analog I/O of port. Each bit of ADPC1 corresponds to a pin of P10 to P12 in port 1 and can be specified in 1-bit units.

Use the ADPC0 and ADPC1 registers to select a pin used in the PGA mode or single AMP mode as an analog I/O.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears ADPC0 to 00H, and sets ADPC1 of 78K0/KB2-L and 78K0/KC2-L to 07H.

**Figure 13-4. Format of A/D Port Configuration Register 0 (ADPC0)**

**(a) 78K0/KY2-L**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(b) 78K0/KA2-L (20-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(c) 78K0/KA2-L (25-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(d) 78K0/KA2-L (32-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(e) 78K0/KB2-L**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | 0 | 0 | 0 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(f) 78K0/KC2-L (40-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | 0 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

**(g) 78K0/KC2-L (44-pin and 48-pin products)**

Address: FF2EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC0 | ADPCS7 | ADPCS6 | ADPCS5 | ADPCS4 | ADPCS3 | ADPCS2 | ADPCS1 | ADPCS0 |

| ADPCSn | Digital I/O or analog I/O selection (n = 0 to 7) |
|---|---|
| 0 | Analog I/O |
| 1 | Digital I/O |

**Cautions 1. Set the pin set to analog I/O to the input mode by using port mode register 2 (PM2).**

**2. If data is written to ADPC0, a wait cycle is generated. Do not write data to ADPC0 when the peripheral hardware clock is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**Figure 13-5. Format of A/D Port Configuration Register 1 (ADPC1)**
**(78K0/KB2-L and 78K0/KC2-L)**

Address: FF2FH After reset: 07H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ADPC1 | 0 | 0 | 0 | 0 | 0 | ADPCS10 | ADPCS9 | ADPCS8 |

| ADPCSn | Digital I/O or analog I/O selection (n = 8 to 10) |
|---|---|
| 0 | Analog I/O |
| 1 | Digital I/O |

**Cautions 1. Set the pin set to analog I/O to the input mode by using port mode register 1 (PM1).**

**2. If data is written to ADPC1, a wait cycle is generated. Do not write data to ADPC1 when the peripheral hardware clock is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(3) Analog input channel specification register (ADS)**

This register specifies the input channel of the analog voltage to be A/D converted.

ADS can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 13-6. Format of Analog Input Channel Specification Register (ADS)**

Address: FF0EH After reset: 00H R/W

| Symbol | 7 | <6> | 5 | 4 | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| ADS | 0 | ADOAS | 0 | 0 | ADS3 | ADS2 | ADS1 | ADS0 |

<R>

| ADOAS0 | ADS3 | ADS2 | ADS1 | ADS0 | Analog input channel | Input source |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | ANI0 | P20/ANI0 pin |
| 0 | 0 | 0 | 0 | 1 | ANI1 | P21/ANI1 pin or operational amplifier 0 output signal |
| 0 | 0 | 0 | 1 | 0 | ANI2 | P22/ANI2 pin |
| 0 | 0 | 0 | 1 | 1 | ANI3 | P23/ANI3 pin |
| 0 | 0 | 1 | 0 | 0 | ANI4 | P24/ANI4 pin |
| 0 | 0 | 1 | 0 | 1 | ANI5 | P25/ANI5 pin |
| 0 | 0 | 1 | 1 | 0 | ANI6 | P26/ANI6 pin |
| 0 | 0 | 1 | 1 | 1 | ANI7 | P27/ANI7 pin |
| 0 | 1 | 0 | 0 | 0 | ANI8 | P10/ANI8 pin |
| 0 | 1 | 0 | 0 | 1 | ANI9 | P11/ANI9 pin or operational amplifier 1 output signal |
| 0 | 1 | 0 | 1 | 0 | ANI10 | P12/ANI10 pin |
| 1 | × | × | × | × | PGAOUT | PGA output signal |
| Other than above | | | | | Setting prohibited | |

**Cautions 1. Be sure to clear bits 4, 5, and 7 to "0".**

**2. Set a channel to be used for A/D conversion in the input mode by using port mode registers 1, 2 (PM1, PM2).**

**Caution 3. Set ADS after PGA operation setting when selecting the PGA output signal as analog input. Set ADS after single AMP operation setting when selecting the operational amplifier output signal as analog input.**

**(4) Port mode registers 1, 2 (PM1, PM2)**

When using AMP0-/ANI0/P20, AMP0OUT/PGAIN/ANI1/P21, and AMP0+/ANI2/P22 pins for the operational amplifier 0, set PM20 to PM22 to 1.

When using AMP1-/ANI8/P10, AMP1OUT/ANI9/P11, and AMP1+/ANI10/P12 pins for the operational amplifier 1, set PM10 to PM12 to 1.

The output latches of P20 to P22 and P10 to P12 at this time may be 0 or 1.

If PM20 to PM22 and PM10 to PM12 are set to 0, they cannot be used as the operational amplifier 0 and 1 pins.

PM1 and PM2 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

**Figure 13-7. Format of Port Mode Register 1 (PM1)**

Address: FF21H After reset: FFH R/W

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

| PM1n | P1n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 1 of the 78K0/KB2-L and 78K0/KC2-L.

**Figure 13-8. Format of Port Mode Register 2 (PM2)**

**(a) 78K0/KY2-L**

Address: FF22H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM2 | 1 | 1 | 1 | 1 | PM23 | PM22 | PM21 | PM20 |

**Caution Be sure to set bits 4 to 7 of PM2 to 1.**

<R>**(b) 78K0/KA2-L**

Address: FF22H After reset: FFH R/W

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

**Notes 1.** 32-pin products only

**2.** 25-pin and 32-pin products only

**(c) 78K0/KB2-L**

Address: FF22H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM2 | 1 | 1 | 1 | 1 | PM23 | PM22 | PM21 | PM20 |

<R>**(d) 78K0/KC2-L**

Address: FF22H After reset: FFH R/W

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

**Note** 44-pin and 48-pin products only

| PM2n | P2n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

When using P10/ANI8/AMP1-, P11/ANI9/AMP1OUT, or P12/ANI10/AMP1+ in the 78K0/KB2-L and 78K0/KC2-L, set the registers according to the pin function to be used (refer to **Tables 13-2** and **13-3**).

**Table 13-2. Setting Functions of P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins**

| ADPC1 Register | PM1 Register | OPAMP1E bit | ADS Register (n = 8, 10) | P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins |
|---|---|---|---|---|
| Analog input selection | Input mode | 0 | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Operational amplifier 1 input |
| | Output mode | – | – | Setting prohibited |
| Digital I/O selection | Input mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital input |
| | Output mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital output |

**Table 13-3. Setting Functions of P11/ANI9/AMP1OUT Pin**

| ADPC1 Register | PM1 Register | OPAMP1E bit | ADS Register | P11/ANI9/AMP1OUT Pin |
|---|---|---|---|---|
| Analog I/O selection | Input mode | 0 | Selects ANI9. | Analog input (to be converted into digital signals) |
| | | | Does not select ANI9. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANI9. | Operational amplifier 1 output (to be converted into digital signals) |
| | | | Does not select ANI9. | Operational amplifier 1 output (not to be converted into digital signals) |
| | Output mode | – | – | Setting prohibited |
| Digital I/O selection | Input mode | 0 | Selects ANI9. | Setting prohibited |
| | | | Does not select ANI9. | Digital input |
| | | 1 | – | Setting prohibited |
| | Output mode | 0 | Selects ANI9. | Setting prohibited |
| | | | Does not select ANI9. | Digital output |
| | | 1 | – | Setting prohibited |

**Remark** ADPC1: A/D port configuration register 1
PM1: Port mode register 1
OPAMP1E: Bit 7 of operational amplifier 1 control register (AMP1M)
ADS: Analog input channel specification register

When using P20/ANI0/AMP0-, P21/ANI1/AMP0OUT/PGAIN, and P22/ANI2/AMP0+, set the registers according to the pin function to be used (refer to **Tables 13-4** and **13-5**).

**Table 13-4. Setting Functions of P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins**

| ADPC0 Register | PM2 Register | OPAMP0E bit | ADS Register (n = 0, 2) | P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins |
|---|---|---|---|---|
| Analog input selection | Input mode | 0 | Selects ANIn. | Analog input (to be converted into digital signals) |
| | | | Does not select ANIn. | Analog input (not to be converted into digital signals) |
| | | 1 | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Operational amplifier input |
| | Output mode | – | – | Setting prohibited |
| Digital I/O selection | Input mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital input |
| | Output mode | – | Selects ANIn. | Setting prohibited |
| | | | Does not select ANIn. | Digital output |

**Remark** ADPC0: A/D port configuration register 0
PM2: Port mode register 2
OPAMP0E: Bit 7 of operational amplifier 0 control register (AMP0M)
ADS: Analog input channel specification register

<R> **Table 13-5. Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin**

| ADPC0 Register | PM2 Register | OPAMP0E bit | PGAEN bit | ADS Register | P21/ANI1/AMP0OUT/PGAIN Pin |
|---|---|---|---|---|---|
| Analog I/O selection | Input mode | 0 | 0 | Selects ANI1. | Analog input (to be converted into digital signals) |
| | | | | Does not select ANI1. | Analog input (not to be converted into digital signals) |
| | | 0 | 1 | Selects PGAOUT. | PGA input (PGA output is converted into digital signals) |
| | | | | Selects ANI1. | PGA input (to be converted into digital signals) |
| | | | | Does not select PGAOUT. | PGA input (not to be converted into digital signals) |
| | | 1 | 0 | Selects ANI1. | Operational amplifier output (to be converted into digital signals) |
| | | | | Does not select ANI1. | Operational amplifier 0 output (not to be converted into digital signals) |
| | | 1 | 1 | Selects PGAOUT. | Operational amplifier 0 output and PGA input (PGA output is converted into digital signals) |
| | | | | Selects ANI1. | Operational amplifier 0 output (to be converted into digital signals) |
| | | | | Does not select PGAOUT and ANI1. | Operational amplifier 0 output (not to be converted into digital signals) |
| | Output mode | – | | – | Setting prohibited |
| Digital I/O selection | Input mode | 0 | – | Selects ANI1. | Setting prohibited |
| | | | | Does not select ANI1. | Digital input |
| | | 1 | – | – | Setting prohibited |
| | Output mode | 0 | – | Selects ANI1. | Setting prohibited |
| | | | | Does not select ANI1. | Digital output |
| | | 1 | – | – | Setting prohibited |

**Remark** ADPC0: A/D port configuration register 0
PM2: Port mode register 2
OPAMP0E: Bit 7 of operational amplifier 0 control register (AMP0M)
PGAEN: Bit 6 of AMP0M
ADS: Analog input channel specification register

## 13.4 Operational Amplifier Operations

The operational amplifiers 0 and 1 have the following mode.

- Single AMP mode (operational amplifiers 0 and 1)
- PGA (Programmable gain amplifier) mode (operational amplifier 0 only)

### 13.4.1 Single AMP mode (operational amplifiers 0 and 1)

Operational amplifiers 0 and 1 both have two input pins (the AMPn- pin and the AMPn+ pin) and one output pin (the AMPnOUT pin), and can be used as single-power supply amplifiers that can be externally connected.

The amplified voltage can be used as an analog input of the A/D converter, because the AMPnOUT pin is alternatively used with analog input pin of the A/D converter.

The procedure for starting operation in single amplifier mode is described below.

<1> Use the ADPCn register to set the pins (AMPn−, AMPn+, AMPnOUT) to be used in single amplifier mode as analog I/O.
<2> Use the PMx register to set the pins (AMPn−, AMPn+, AMPnOUT) to be used in single amplifier mode to input mode.
<3> Set (1) the OPAMPnE bit and enable operation in single amplifier mode.

**Caution To use as an input of the A/D converter a voltage that has been amplified in single amplifier mode, enable operation in single amplifier mode before selecting an analog input channel by using the ADS register.**

**Remark** Products with operational amplifier of the 78K0/KY2-L and 78K0/KA2-L: n = 0, x = 2
Products with operational amplifier of the 78K0/KB2-L and 78K0/KC2-L: n = 0, 1, x = 2, 1

### 13.4.2 PGA (Programmable gain amplifier) mode (operational amplifier 0 only)

In this mode, the analog voltage input from the PGAIN pin is amplified within the microcontroller. The gain can be selected from four types (×4, ×8, ×16, ×32).

The amplified voltage can be used as an analog input of the A/D converter.

The procedure for starting operation in PGA mode is described below.

<1> Use the ADPC0 register to set the pins (PGAIN) to be used in PGA mode as analog I/O.
<2> Use the PM2 register to set the pins (PGAIN) to be used in PGA mode to input mode.
<3> Use the AMP0VG0 and AMP0VG1 bits to select the gain (×4, ×8, ×16, ×32).
<4> Set (1) the PGAEN bit and enable operation in PGA mode.

**Caution To use as an input of the A/D converter a voltage that has been amplified in PGA mode, enable operation in PGA mode before selecting an analog input channel by using the ADS register.**

# CHAPTER 14 SERIAL INTERFACE UART6

## 14.1 Functions of Serial Interface UART6

Serial interface UART6 are mounted onto all 78K0/Kx2-L microcontroller products.
Serial interface UART6 has the following two modes.

**(1) Operation stop mode**

This mode is used when serial communication is not executed and can enable a reduction in the power consumption.
For details, refer to **14.4.1 Operation stop mode**.

**(2) Asynchronous serial interface (UART) mode**

This mode supports the LIN (Local Interconnect Network)-bus. The functions of this mode are outlined below.
For details, refer to **14.4.2 Asynchronous serial interface (UART) mode** and **14.4.3 Dedicated baud rate generator**.

- Maximum transfer rate: 625 kbps
- Two-pin configuration TxD6: Transmit data output pin
  RxD6: Receive data input pin
- Data length of communication data can be selected from 7 or 8 bits.
- Dedicated internal 8-bit baud rate generator allowing any baud rate to be set
- Transmission and reception can be performed independently (full duplex operation).
- MSB- or LSB-first communication selectable
- Inverted transmission operation
- Sync break field transmission from 13 to 20 bits
- More than 11 bits can be identified for sync break field reception (SBF reception flag provided).

**Cautions 1. The TxD6 output inversion function inverts only the transmission side and not the reception side. To use this function, the reception side must be ready for reception of inverted data.**

**2. If clock supply to serial interface UART6 is not stopped (e.g., in the HALT mode), normal operation continues. If clock supply to serial interface UART6 is stopped (e.g., in the STOP mode), each register stops operating, and holds the value immediately before clock supply was stopped. The TxD6 pin also holds the value immediately before clock supply was stopped and outputs it. However, the operation is not guaranteed after clock supply is resumed. Therefore, reset the circuit so that POWER6 = 0, RXE6 = 0, and TXE6 = 0.**

**3. Set POWER6 = 1 and then set TXE6 = 1 (transmission) or RXE6 = 1 (reception) to start communication.**

**4. TXE6 and RXE6 are synchronized by the base clock ($f_{XCLK6}$) set by CKSR6. To enable transmission or reception again, set TXE6 or RXE6 to 1 at least two clocks of the base clock after TXE6 or RXE6 has been cleared to 0. If TXE6 or RXE6 is set within two clocks of the base clock, the transmission circuit or reception circuit may not be initialized.**

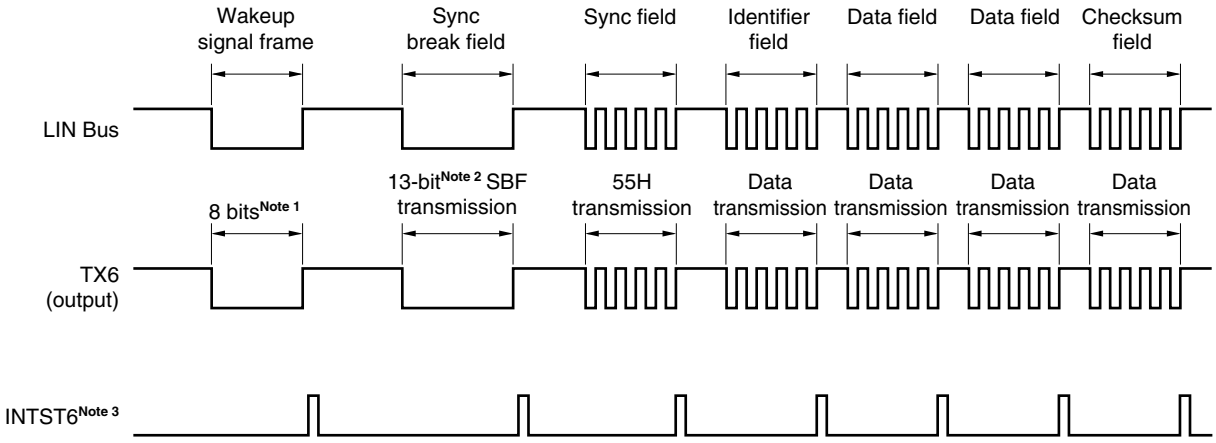
**5. Set transmit data to TXB6 at least one base clock ($f_{XCLK6}$) after setting TXE6 = 1.**

**6. If data is continuously transmitted, the communication timing from the stop bit to the next start bit is extended two operating clocks of the macro. However, this does not affect the result of communication because the reception side initializes the timing when it has detected a start bit. Do not use the continuous transmission function if the interface is used in LIN communication operation.**

**Remark** LIN stands for Local Interconnect Network and is a low-speed (1 to 20 kbps) serial communication protocol intended to aid the cost reduction of an automotive network.
LIN communication is single-master communication, and up to 15 slaves can be connected to one master.
The LIN slaves are used to control the switches, actuators, and sensors, and these are connected to the LIN master via the LIN network.
Normally, the LIN master is connected to a network such as CAN (Controller Area Network).
In addition, the LIN bus uses a single-wire method and is connected to the nodes via a transceiver that complies with ISO9141.
In the LIN protocol, the master transmits a frame with baud rate information and the slave receives it and corrects the baud rate error. Therefore, communication is possible when the baud rate error in the slave is ±15% or less.

Figures 14-1 and 14-2 outline the transmission and reception operations of LIN.

**Figure 14-1. LIN Transmission Operation**



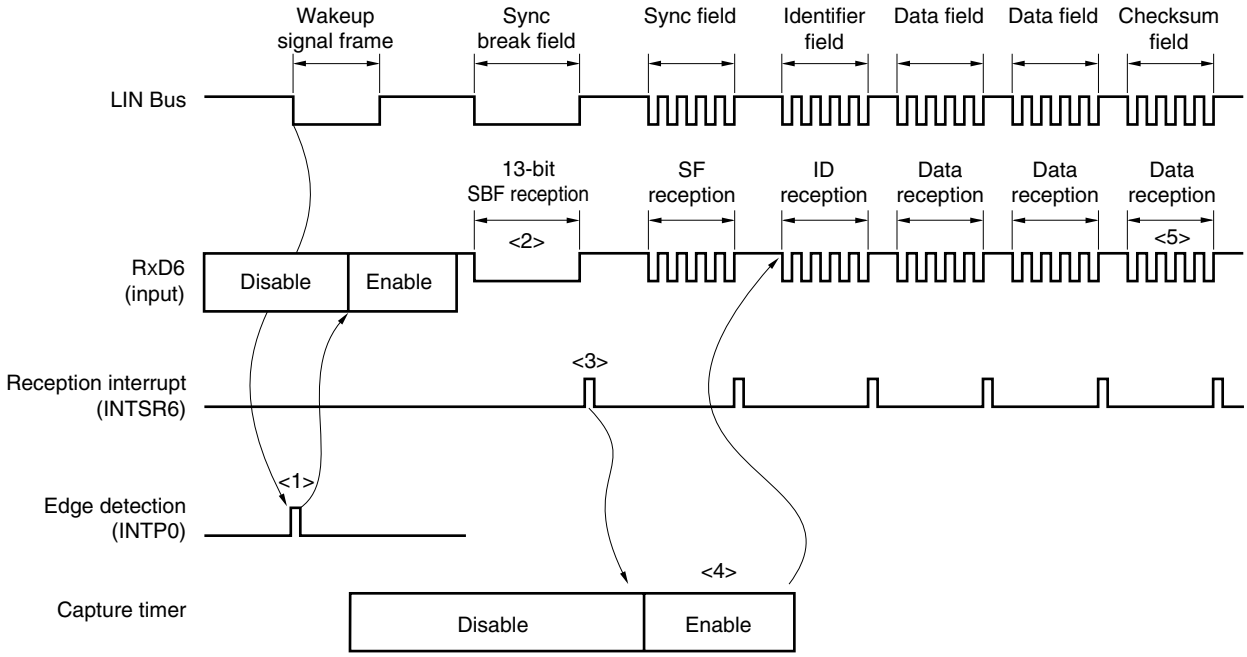

**Notes 1.** The wakeup signal frame is substituted by 80H transmission in the 8-bit mode.

**2.** The sync break field is output by hardware. The output width is the bit length set by bits 4 to 2 (SBL62 to SBL60) of asynchronous serial interface control register 6 (ASICL6) (refer to **14.4.2 (2) (h) SBF transmission**).

**3.** INTST6 is output on completion of each transmission. It is also output when SBF is transmitted.

**Remark** The interval between each field is controlled by software.

**Figure 14-2. LIN Reception Operation**




Reception processing is as follows.

<1> The wakeup signal is detected at the edge of the pin, and enables UART6 and sets the SBF reception mode.

<2> Reception continues until the STOP bit is detected. When an SBF with low-level data of 11 bits or more has been detected, it is assumed that SBF reception has been completed correctly, and an interrupt signal is output. If an SBF with low-level data of less than 11 bits has been detected, it is assumed that an SBF reception error has occurred. The interrupt signal is not output and the SBF reception mode is restored.

<3> If SBF reception has been completed correctly, an interrupt signal is output. Start 16-bit timer/event counter 00 by the SBF reception end interrupt servicing and measure the bit interval (pulse width) of the sync field (refer to **6.4.8 Pulse width measurement operation**). Detection of errors OVE6, PE6, and FE6 is suppressed, and error detection processing of UART communication and data transfer of the shift register and RXB6 is not performed. The shift register holds the reset value FFH.

<4> Calculate the baud rate error from the bit interval of the sync field, disable UART6 after SF reception, and then re-set baud rate generator control register 6 (BRGC6).

<5> Distinguish the checksum field by software. Also perform processing by software to initialize UART6 after reception of the checksum field and to set the SBF reception mode again.

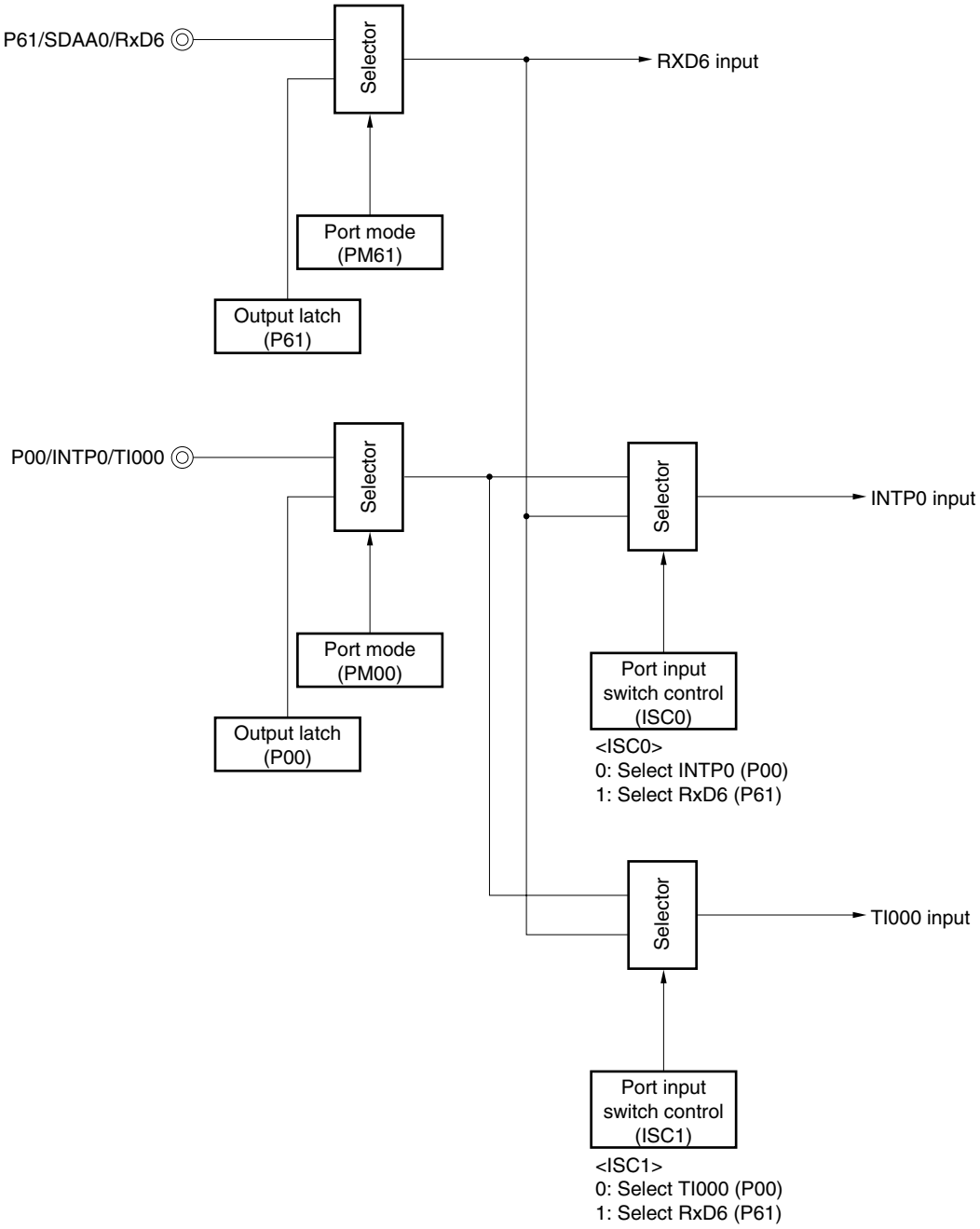
Figure 14-3 shows the port configuration for LIN reception operation.

The wakeup signal transmitted from the LIN master is received by detecting the edge of the external interrupt (INTP0). The length of the sync field transmitted from the LIN master can be measured using the external event capture operation of 16-bit timer/event counter 00, and the baud rate error can be calculated.

The input source of the reception port input (RxD6) can be input to the external interrupt (INTP0) and 16-bit timer/event counter 00 by port input switch control (ISC0/ISC1), without connecting RxD6 and INTP0/TI000 externally.

**Figure 14-3. Port Configuration for LIN Reception Operation (1/2)**

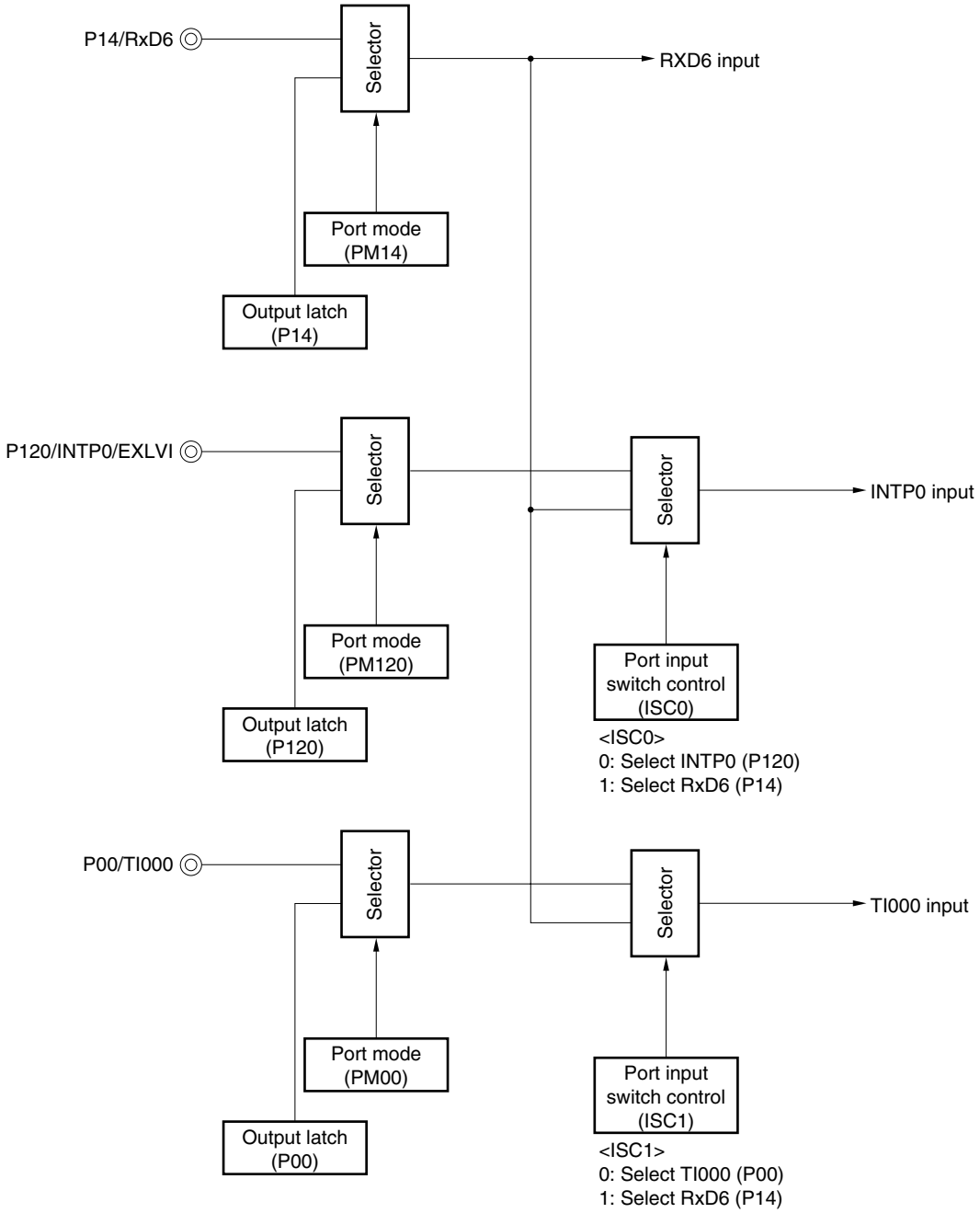
**(1) 78K0/KY2-L and 78K0/KA2-L**




**Remark** ISC0, ISC1: Bits 0 and 1 of the input switch control register (ISC) (refer to **Figure 14-11**)

**Figure 14-3. Port Configuration for LIN Reception Operation (2/2)**

**(2) 78K0/KB2-L and 78K0/KC2-L**




**Remark** ISC0, ISC1: Bits 0 and 1 of the input switch control register (ISC) (refer to **Figure 14-11**)

The peripheral functions used in the LIN communication operation are shown below.

<Peripheral functions used>

- External interrupt (INTP0); wakeup signal detection
  Use: Detects the wakeup signal edges and detects start of communication.
- 16-bit timer/event counter 00 (TI000); baud rate error detection
  Use: Detects the baud rate error (measures the TI000 input edge interval in the capture mode) by detecting the sync field (SF) length and divides it by the number of bits.
- Serial interface UART6

## 14.2 Configuration of Serial Interface UART6

Serial interface UART6 includes the following hardware.

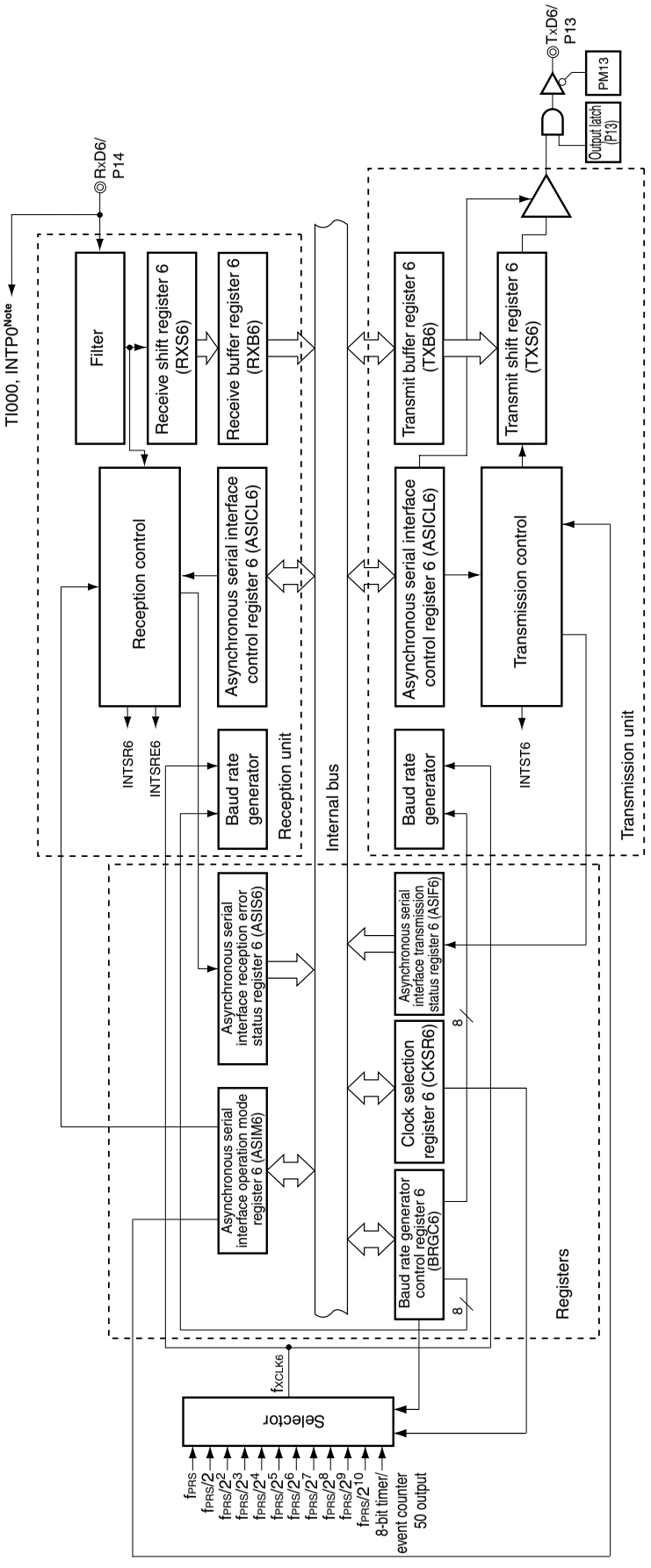
**Table 14-1. Configuration of Serial Interface UART6**

| Item | Configuration |
|---|---|
| Registers | Receive buffer register 6 (RXB6)<br>Receive shift register 6 (RXS6)<br>Transmit buffer register 6 (TXB6)<br>Transmit shift register 6 (TXS6) |
| Control registers | Asynchronous serial interface operation mode register 6 (ASIM6)<br>Asynchronous serial interface reception error status register 6 (ASIS6)<br>Asynchronous serial interface transmission status register 6 (ASIF6)<br>Clock selection register 6 (CKSR6)<br>Baud rate generator control register 6 (BRGC6)<br>Asynchronous serial interface control register 6 (ASICL6)<br>Input switch control register (ISC)<br>Port mode register 1 (PM1), port mode register 6 (PM6)Note 1<br>Port register 1 (P1), port register 6 (P6)Note 1<br>Port output mode register 6 (POM6) Note 2 |

**Notes 1.** 78K0/KY2-L, 78K0/KA2-L: Port mode register 6 (PM6), port register 6 (P6)
78K0/KB2-L, 78K0/KC2-L: Port mode register 1 (PM1), port register 1 (P1)

**2.** In the 78K0/KY2-L and 78K0/KA2-L, this register is used when using serial interface UART6.

**Figure 14-4. Block Diagram of Serial Interface UART6**




**Note** Selectable with input switch control register (ISC).

**Remark** 78K0/KY2-L, 78K0/KA2-L: RxD6/SDAA0/P61, TxD6/SCLA0/P60
78K0/KB2-L, 78K0/KC2-L: RxD6/P14, TxD6/P13

### (1) Receive buffer register 6 (RXB6)

This 8-bit register stores parallel data converted by receive shift register 6 (RXS6).

Each time 1 byte of data has been received, new receive data is transferred to this register from RXS6. If the data length is set to 7 bits, data is transferred as follows.

- In LSB-first reception, the receive data is transferred to bits 0 to 6 of RXB6 and the MSB of RXB6 is always 0.
- In MSB-first reception, the receive data is transferred to bits 1 to 7 of RXB6 and the LSB of RXB6 is always 0.

If an overrun error (OVE6) occurs, the receive data is not transferred to RXB6.

RXB6 can be read by an 8-bit memory manipulation instruction. No data can be written to this register.

Reset signal generation sets this register to FFH.

**(2) Receive shift register 6 (RXS6)**

This register converts the serial data input to the RxD6 pin into parallel data.

RXS6 cannot be directly manipulated by a program.

**(3) Transmit buffer register 6 (TXB6)**

This buffer register is used to set transmit data. Transmission is started when data is written to TXB6.

This register can be read or written by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Cautions 1. Do not write data to TXB6 when bit 1 (TXBF6) of asynchronous serial interface transmission status register 6 (ASIF6) is 1.**

**2. Do not refresh (write the same value to) TXB6 by software during a communication operation (when bits 7 and 6 (POWER6, TXE6) of asynchronous serial interface operation mode register 6 (ASIM6) are 1 or when bits 7 and 5 (POWER6, RXE6) of ASIM6 are 1).**

**3. Set transmit data to TXB6 at least one base clock ($f_{XCLK6}$) after setting TXE6 = 1.**

**(4) Transmit shift register 6 (TXS6)**

This register transmits the data transferred from TXB6 from the TxD6 pin as serial data. Data is transferred from TXB6 immediately after TXB6 is written for the first transmission, or immediately before INTST6 occurs after one frame was transmitted for continuous transmission. Data is transferred from TXB6 and transmitted from the TxD6 pin at the falling edge of the base clock.

TXS6 cannot be directly manipulated by a program.

## 14.3 Registers Controlling Serial Interface UART6

Serial interface UART6 is controlled by the following ten registers.

- Asynchronous serial interface operation mode register 6 (ASIM6)
- Asynchronous serial interface reception error status register 6 (ASIS6)
- Asynchronous serial interface transmission status register 6 (ASIF6)
- Clock selection register 6 (CKSR6)
- Baud rate generator control register 6 (BRGC6)
- Asynchronous serial interface control register 6 (ASICL6)
- Input switch control register (ISC)
- Port mode register 1 (PM1), port mode register 6 (PM6)[Note 1]
- Port register 1 (P1), port register 6 (P6)[Note 1]
- Port output mode register 6 (POM6)[Note 2]

**Notes 1.** 78K0/KY2-L, 78K0/KA2-L: Port mode register 6 (PM6), port register 6 (P6)
78K0/KB2-L, 78K0/KC2-L: Port mode register 1 (PM1), port register 1 (P1)

**2.** In the 78K0/KY2-L and 78K0/KA2-L, this register is used when using serial interface UART6.

### (1) Asynchronous serial interface operation mode register 6 (ASIM6)

This 8-bit register controls the serial communication operations of serial interface UART6.
This register can be set by a 1-bit or 8-bit memory manipulation instruction.
Reset signal generation sets this register to 01H.

**Remark** ASIM6 can be refreshed (the same value is written) by software during a communication operation (when bits 7 and 6 (POWER6, TXE6) of ASIM6 = 1 or bits 7 and 5 (POWER6, RXE6) of ASIM6 = 1).

**Figure 14-5. Format of Asynchronous Serial Interface Operation Mode Register 6 (ASIM6) (1/2)**

Address: FF50H After reset: 01H R/W

| Symbol | <7> | <6> | <5> | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ASIM6 | POWER6 | TXE6 | RXE6 | PS61 | PS60 | CL6 | SL6 | ISRM6 |

| POWER6 | Enables/disables operation of internal operation clock |
|---|---|
| 0[Note 1] | Disables operation of the internal operation clock (fixes the clock to low level) and asynchronously resets the internal circuit[Note 2]. |
| 1 | Enables operation of the internal operation clock |

| TXE6 | Enables/disables transmission |
|---|---|
| 0 | Disables transmission (synchronously resets the transmission circuit). |
| 1 | Enables transmission |

| RXE6 | Enables/disables reception |
|---|---|
| 0 | Disables reception (synchronously resets the reception circuit). |
| 1 | Enables reception |

**Notes 1.** The output of the TxD6 pin is fixed to the high level (when TXDLV6 = 0) and the input from the RxD6 pin is fixed to the high level when POWER6 = 0 during transmission.

**2.** Asynchronous serial interface reception error status register 6 (ASIS6), asynchronous serial interface transmission status register 6 (ASIF6), bit 7 (SBRF6) and bit 6 (SBRT6) of asynchronous serial interface control register 6 (ASICL6), and receive buffer register 6 (RXB6) are reset.

**Figure 14-5. Format of Asynchronous Serial Interface Operation Mode Register 6 (ASIM6) (2/2)**

| PS61 | PS60 | Transmission operation | Reception operation |
|---|---|---|---|
| 0 | 0 | Does not output parity bit. | Reception without parity |
| 0 | 1 | Outputs 0 parity. | Reception as 0 parity[Note] |
| 1 | 0 | Outputs odd parity. | Judges as odd parity. |
| 1 | 1 | Outputs even parity. | Judges as even parity. |

| CL6 | Specifies character length of transmit/receive data |
|---|---|
| 0 | Character length of data = 7 bits |
| 1 | Character length of data = 8 bits |

| SL6 | Specifies number of stop bits of transmit data |
|---|---|
| 0 | Number of stop bits = 1 |
| 1 | Number of stop bits = 2 |

| ISRM6 | Enables/disables occurrence of reception completion interrupt in case of error |
|---|---|
| 0 | “INTSRE6” occurs in case of error (at this time, INTSR6 does not occur). |
| 1 | “INTSR6” occurs in case of error (at this time, INTSRE6 does not occur). |

**Note** If “reception as 0 parity” is selected, the parity is not judged. Therefore, bit 2 (PE6) of asynchronous serial interface reception error status register 6 (ASIS6) is not set and the error interrupt does not occur.

**Cautions**
1. **To start the transmission, set POWER6 to 1 and then set TXE6 to 1. To stop the transmission, clear TXE6 to 0, and then clear POWER6 to 0.**
2. **To start the reception, set POWER6 to 1 and then set RXE6 to 1. To stop the reception, clear RXE6 to 0, and then clear POWER6 to 0.**
3. **Set POWER6 to 1 and then set RXE6 to 1 while a high level is input to the RxD6 pin. If POWER6 is set to 1 and RXE6 is set to 1 while a low level is input, reception is started.**
4. **TXE6 and RXE6 are synchronized by the base clock ($f_{XCLK6}$) set by CKSR6. To enable transmission or reception again, set TXE6 or RXE6 to 1 at least two clocks of the base clock after TXE6 or RXE6 has been cleared to 0. If TXE6 or RXE6 is set within two clocks of the base clock, the transmission circuit or reception circuit may not be initialized.**
5. **Set transmit data to TXB6 at least one base clock ($f_{XCLK6}$) after setting TXE6 = 1.**
6. **Clear the TXE6 and RXE6 bits to 0 before rewriting the PS61, PS60, and CL6 bits.**
7. **Fix the PS61 and PS60 bits to 0 when used in LIN communication operation.**
8. **Clear TXE6 to 0 before rewriting the SL6 bit. Reception is always performed with “the number of stop bits = 1”, and therefore, is not affected by the set value of the SL6 bit.**
9. **Make sure that RXE6 = 0 when rewriting the ISRM6 bit.**

**(2) Asynchronous serial interface reception error status register 6 (ASIS6)**

This register indicates an error status on completion of reception by serial interface UART6. It includes three error flag bits (PE6, FE6, OVE6).

This register is read-only by an 8-bit memory manipulation instruction.

Reset signal generation, or clearing bit 7 (POWER6) or bit 5 (RXE6) of ASIM6 to 0 clears this register to 00H. 00H is read when this register is read. If a reception error occurs, read ASIS6 and then read receive buffer register 6 (RXB6) to clear the error flag.

**Figure 14-6. Format of Asynchronous Serial Interface Reception Error Status Register 6 (ASIS6)**

Address: FF53H After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ASIS6 | 0 | 0 | 0 | 0 | 0 | PE6 | FE6 | OVE6 |

| PE6 | Status flag indicating parity error |
|---|---|
| 0 | If POWER6 = 0 or RXE6 = 0, or if ASIS6 register is read |
| 1 | If the parity of transmit data does not match the parity bit on completion of reception |

| FE6 | Status flag indicating framing error |
|---|---|
| 0 | If POWER6 = 0 or RXE6 = 0, or if ASIS6 register is read |
| 1 | If the stop bit is not detected on completion of reception |

| OVE6 | Status flag indicating overrun error |
|---|---|
| 0 | If POWER6 = 0 or RXE6 = 0, or if ASIS6 register is read |
| 1 | If receive data is set to the RXB6 register and the next reception operation is completed before the data is read. |

**Cautions 1. The operation of the PE6 bit differs depending on the set values of the PS61 and PS60 bits of asynchronous serial interface operation mode register 6 (ASIM6).**

**2. For the stop bit of the receive data, only the first stop bit is checked regardless of the number of stop bits.**

**3. If an overrun error occurs, the next receive data is not written to receive buffer register 6 (RXB6) but discarded.**

**4. If data is read from ASIS6, a wait cycle is generated. Do not read data from ASIS6 when the peripheral hardware clock ($f_{PRS}$) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.**

**(3) Asynchronous serial interface transmission status register 6 (ASIF6)**

This register indicates the status of transmission by serial interface UART6. It includes two status flag bits (TXBF6 and TXSF6).

Transmission can be continued without disruption even during an interrupt period, by writing the next data to the TXB6 register after data has been transferred from the TXB6 register to the TXS6 register.

This register is read-only by an 8-bit memory manipulation instruction.

Reset signal generation, or clearing bit 7 (POWER6) or bit 6 (TXE6) of ASIM6 to 0 clears this register to 00H.

**Figure 14-7. Format of Asynchronous Serial Interface Transmission Status Register 6 (ASIF6)**

Address: FF55H After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ASIF6 | 0 | 0 | 0 | 0 | 0 | 0 | TXBF6 | TXSF6 |

| TXBF6 | Transmit buffer data flag |
|---|---|
| 0 | If POWER6 = 0 or TXE6 = 0, or if data is transferred to transmit shift register 6 (TXS6) |
| 1 | If data is written to transmit buffer register 6 (TXB6) (if data exists in TXB6) |

| TXSF6 | Transmit shift register data flag |
|---|---|
| 0 | If POWER6 = 0 or TXE6 = 0, or if the next data is not transferred from transmit buffer register 6 (TXB6) after completion of transfer |
| 1 | If data is transferred from transmit buffer register 6 (TXB6) (if data transmission is in progress) |

**Cautions 1. To transmit data continuously, write the first transmit data (first byte) to the TXB6 register. Be sure to check that the TXBF6 flag is "0". If so, write the next transmit data (second byte) to the TXB6 register. If data is written to the TXB6 register while the TXBF6 flag is "1", the transmit data cannot be guaranteed.**

**2. To initialize the transmission unit upon completion of continuous transmission, be sure to check that the TXSF6 flag is "0" after generation of the transmission completion interrupt, and then execute initialization. If initialization is executed while the TXSF6 flag is "1", the transmit data cannot be guaranteed.**

**(4) Clock selection register 6 (CKSR6)**

This register selects the base clock of serial interface UART6.

CKSR6 can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Remark** CKSR6 can be refreshed (the same value is written) by software during a communication operation (when bits 7 and 6 (POWER6, TXE6) of ASIM6 = 1 or bits 7 and 5 (POWER6, RXE6) of ASIM6 = 1).

**Figure 14-8. Format of Clock Selection Register 6 (CKSR6)**

Address: FF56H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CKSR6 | 0 | 0 | 0 | 0 | TPS63 | TPS62 | TPS61 | TPS60 |

| TPS63 | TPS62 | TPS61 | TPS60 | Base clock ($f_{XCLK6}$) selection[Note 1] | | | |
|---|---|---|---|---|---|---|---|
| | | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 0 | 0 | 1 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz |
| 0 | 0 | 1 | 0 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz |
| 0 | 0 | 1 | 1 | $f_{PRS}/2^3$ | 250 kHz | 625 kHz | 1.25 MHz |
| 0 | 1 | 0 | 0 | $f_{PRS}/2^4$ | 125 kHz | 312.5 kHz | 625 kHz |
| 0 | 1 | 0 | 1 | $f_{PRS}/2^5$ | 62.5 kHz | 156.25 kHz | 312.5 kHz |
| 0 | 1 | 1 | 0 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz |
| 0 | 1 | 1 | 1 | $f_{PRS}/2^7$ | 15.625 kHz | 39.06 kHz | 78.13 kHz |
| 1 | 0 | 0 | 0 | $f_{PRS}/2^8$ | 7.813 kHz | 19.53 kHz | 39.06 kHz |
| 1 | 0 | 0 | 1 | $f_{PRS}/2^9$ | 3.906 kHz | 9.77 kHz | 19.53 kHz |
| 1 | 0 | 1 | 0 | $f_{PRS}/2^{10}$ | 1.953 kHz | 4.88 kHz | 9.77 kHz |
| 1 | 0 | 1 | 1 | TM50 output[Notes 2, 3] | | | |
| Other than above | | | | Setting prohibited | | | |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** Note the following points when selecting the TM50 output as the base clock.

- Mode in which the count clock is cleared and started upon a match of TM50 and CR50 (TMC506 = 0)
  Start the operation of 8-bit timer/event counter 50 first and then enable the timer F/F inversion operation (TMC501 = 1).
- PWM mode (TMC506 = 1)
  Start the operation of 8-bit timer/event counter 50 first and then set the count clock to make the duty = 50%.

It is not necessary to enable (TOE50 = 1) TO50 output in any mode.

**3.** 78K0/KB2-L and 78K0/KC2-L only

**Caution Make sure POWER6 = 0 when rewriting TPS63 to TPS60.**

**Remarks 1.** $f_{PRS}$: Peripheral hardware clock frequency

**2.** TMC506: Bit 6 of 8-bit timer mode control register 50 (TMC50)
TMC501: Bit 1 of TMC50

**(5) Baud rate generator control register 6 (BRGC6)**

This register sets the division value of the 8-bit counter of serial interface UART6.

BRGC6 can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Remark** BRGC6 can be refreshed (the same value is written) by software during a communication operation (when bits 7 and 6 (POWER6, TXE6) of ASIM6 = 1 or bits 7 and 5 (POWER6, RXE6) of ASIM6 = 1).

**Figure 14-9. Format of Baud Rate Generator Control Register 6 (BRGC6)**

Address: FF57H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| BRGC6 | MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 |

| MDL67 | MDL66 | MDL65 | MDL64 | MDL63 | MDL62 | MDL61 | MDL60 | k | Output clock selection of 8-bit counter |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | × | × | × | Setting prohibited |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 4 | $f_{XCLK6}/4$ |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 5 | $f_{XCLK6}/5$ |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 6 | $f_{XCLK6}/6$ |
| • | • | • | • | • | • | • | • | • | • |
| • | • | • | • | • | • | • | • | • | • |
| • | • | • | • | • | • | • | • | • | • |
| • | • | • | • | • | • | • | • | • | • |
| • | • | • | • | • | • | • | • | • | • |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 252 | $f_{XCLK6}/252$ |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 253 | $f_{XCLK6}/253$ |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 254 | $f_{XCLK6}/254$ |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 255 | $f_{XCLK6}/255$ |

**Cautions 1. Make sure that bit 6 (TXE6) and bit 5 (RXE6) of the ASIM6 register = 0 when rewriting the MDL67 to MDL60 bits.**

**2. The baud rate is the output clock of the 8-bit counter divided by 2.**

**Remarks 1.** $f_{XCLK6}$: Frequency of base clock selected by the TPS63 to TPS60 bits of CKSR6 register

**2.** k: Value set by MDL67 to MDL60 bits (k = 4, 5, 6, ..., 255)

**3.** ×: Don't care

**(6) Asynchronous serial interface control register 6 (ASICL6)**

This register controls the serial communication operations of serial interface UART6.

ASICL6 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 16H.

**Caution ASICL6 can be refreshed (the same value is written) by software during a communication operation (when bits 7 and 6 (POWER6, TXE6) of ASIM6 = 1 or bits 7 and 5 (POWER6, RXE6) of ASIM6 = 1). However, do not set both SBRT6 and SBTT6 to 1 by a refresh operation during SBF reception (SBRT6 = 1) or SBF transmission (until INTST6 occurs since SBTT6 has been set (1)), because it may re-trigger SBF reception or SBF transmission.**

**Figure 14-10. Format of Asynchronous Serial Interface Control Register 6 (ASICL6) (1/2)**

Address: FF58H After reset: 16H R/W[Note]

| Symbol | <7> | <6> | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ASICL6 | SBRF6 | SBRT6 | SBTT6 | SBL62 | SBL61 | SBL60 | DIR6 | TXDLV6 |

| SBRF6 | SBF reception status flag |
|---|---|
| 0 | If POWER6 = 0 and RXE6 = 0 or if SBF reception has been completed correctly |
| 1 | SBF reception in progress |

| SBRT6 | SBF reception trigger |
|---|---|
| 0 | – |
| 1 | SBF reception trigger |

| SBTT6 | SBF transmission trigger |
|---|---|
| 0 | – |
| 1 | SBF transmission trigger |

**Note** Bit 7 is read-only.

**Figure 14-10. Format of Asynchronous Serial Interface Control Register 6 (ASICL6) (2/2)**

| SBL62 | SBL61 | SBL60 | SBF transmission output width control |
|---|---|---|---|
| 1 | 0 | 1 | SBF is output with 13-bit length. |
| 1 | 1 | 0 | SBF is output with 14-bit length. |
| 1 | 1 | 1 | SBF is output with 15-bit length. |
| 0 | 0 | 0 | SBF is output with 16-bit length. |
| 0 | 0 | 1 | SBF is output with 17-bit length. |
| 0 | 1 | 0 | SBF is output with 18-bit length. |
| 0 | 1 | 1 | SBF is output with 19-bit length. |
| 1 | 0 | 0 | SBF is output with 20-bit length. |

| DIR6 | First-bit specification |
|---|---|
| 0 | MSB |
| 1 | LSB |

| TXDLV6 | Enables/disables inverting TxD6 output |
|---|---|
| 0 | Normal output of TxD6 |
| 1 | Inverted output of TxD6 |

**Cautions**
1. **In the case of an SBF reception error, the mode returns to the SBF reception mode. The status of the SBRF6 flag is held (1).**
2. **Before setting the SBRT6 bit, make sure that bit 7 (POWER6) and bit 5 (RXE6) of ASIM6 = 1. After setting the SBRT6 bit to 1, do not clear it to 0 before SBF reception is completed (before an interrupt request signal is generated).**
3. **The read value of the SBRT6 bit is always 0. SBRT6 is automatically cleared to 0 after SBF reception has been correctly completed.**
4. **Before setting the SBTT6 bit to 1, make sure that bit 7 (POWER6) and bit 6 (TXE6) of ASIM6 = 1. After setting the SBTT6 bit to 1, do not clear it to 0 before SBF transmission is completed (before an interrupt request signal is generated).**
5. **The read value of the SBTT6 bit is always 0. SBTT6 is automatically cleared to 0 at the end of SBF transmission.**
6. **Do not set the SBRT6 bit to 1 during reception, and do not set the SBTT6 bit to 1 during transmission.**
7. **Before rewriting the DIR6 and TXDLV6 bits, clear the TXE6 and RXE6 bits to 0.**
8. **When the TXDLV6 bit is set to 1 (inverted TxD6 output), the TxD6/SCLA0/P60 pin (78K0/KY2-L, 78K0/KA2-L) or TxD6/P13 pin (78K0/KB2-L, 78K0/KC2-L) cannot be used as a general-purpose port, regardless of the settings of POWER6 and TXE6. When using the TxD6/SCLA0/P60 or TxD6/P13 pin as a general-purpose port, clear the TXDLV6 bit to 0 (normal TxD6 output).**

**(7) Input switch control register (ISC)**

The input switch control register (ISC) is used to receive a status signal transmitted from the master during LIN (Local Interconnect Network) reception.

The signal input from the RxD6 pin is selected as the input source of INTP0 and TI000 when ISC0 and ISC1 are set to 1 (refer to **Figure 14-3 Port Configuration for LIN Reception Operation**).

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

Reset signal generation sets this register to 00H.

**Figure 14-11. Format of Input Switch Control Register (ISC)**

Address: FF4FH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ISC | 0 | 0 | 0 | 0 | 0 | 0 | ISC1 | ISC0 |

| ISC1 | TI000 input source selection |
|---|---|
| 0 | TI000 |
| 1 | RxD6 |

| ISC0 | INTP0 input source selection |
|---|---|
| 0 | INTP0 |
| 1 | RxD6 |

**Remark** 78K0/KY2-L, 78K0/KA2-L: TI000/INTP0/P00, RxD6/SDAA0/P61
78K0/KB2-L, 78K0/KC2-L: TI000/P00, INTP0/EXLVI/P120, RxD6/P14

**(8) Port mode register 1 (PM1), port mode register 6 (PM6)**

These registers set port 1 input/output or port 6 input/output in 1-bit units.

PM1 and PM6 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

• 78K0/KY2-L, 78K0/KA2-L

When using the P60/TxD6/SCLA0 pin for serial interface data output, clear PM60 to 0 and set the output latch of P60 to 1.

When using the P61/RxD6/SDAA0 pin for serial interface data input, set PM61 to 1. The output latch of P61 at this time may be 0 or 1.

• 78K0/KB2-L, 78K0/KC2-L

When using the P13/TxD6 pin for serial interface data output, clear PM13 to 0 and set the output latch of P13 to 1.

When using the P14/RxD6 pin for serial interface data input, set PM14 to 1. The output latch of P14 at this time may be 0 or 1.

**Figure 14-12. Format of Port Mode Register 1 (PM1)**

Address: FF21H After reset: FFH R/W

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

| PM1n | P1n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 1 (PM1) of the 78K0/KB2-L and 78K0/KC2-L.

**Figure 14-13. Format of Port Mode Register 6 (PM6)**

Address: FF26H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM6 | 1 | 1 | 1 | 1 | 1 | 1 | PM61 | PM60 |

| PM6n | P6n pin I/O mode selection (n = 0, 1) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 6 (PM6) of the 78K0/KY2-L and 78K0/KA2-L.

**(9) Port output mode register 6 (POM6)**

This register sets the output mode of P60 and P61 in 1-bit units.

In the 78K0/KY2-L and 78K0/KA2-L, clear POM60 to 0 when using the P60/TxD6/SCLA0 pin as the data output of serial interface UART6.

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

Reset signal generation clears this register to 00H.

**Figure 14-14. Format of Port Output Mode Register 6 (POM6)**

Address: FF2AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| POM6 | 0 | 0 | 0 | 0 | 0 | 0 | POM61 | POM60 |

| POM6n | P6n pin output mode selection (n = 0 and 1) |
|---|---|
| 0 | Normal output (CMOS output) mode |
| 1 | N-ch open drain output ($V_{DD}$ tolerance) mode |

**Remark** The figure shown above presents the format of port output mode register 6 (POM6) of the 78K0/KY2-L and 78K0/KA2-L.

## 14.4 Operation of Serial Interface UART6

Serial interface UART6 has the following two modes.

- Operation stop mode
- Asynchronous serial interface (UART) mode

### 14.4.1 Operation stop mode

In this mode, serial communication cannot be executed; therefore, the power consumption can be reduced. In addition, the pins can be used as ordinary port pins in this mode. To set the operation stop mode, clear bits 7, 6, and 5 (POWER6, TXE6, and RXE6) of ASIM6 to 0.

**(1) Register used**

The operation stop mode is set by asynchronous serial interface operation mode register 6 (ASIM6).

ASIM6 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 01H.

Address: FF50H After reset: 01H R/W

| Symbol | <7> | <6> | <5> | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| ASIM6 | POWER6 | TXE6 | RXE6 | PS61 | PS60 | CL6 | SL6 | ISRM6 |

| POWER6 | Enables/disables operation of internal operation clock |
|---|---|
| 0[Note 1] | Disables operation of the internal operation clock (fixes the clock to low level) and asynchronously resets the internal circuit[Note 2]. |

| TXE6 | Enables/disables transmission |
|---|---|
| 0 | Disables transmission operation (synchronously resets the transmission circuit). |

| RXE6 | Enables/disables reception |
|---|---|
| 0 | Disables reception (synchronously resets the reception circuit). |

**Notes 1.** The output of the TxD6 pin is fixed to high level (when TXDLV6 = 0) and the input from the RxD6 pin is fixed to high level when POWER6 = 0 during transmission.

**2.** Asynchronous serial interface reception error status register 6 (ASIS6), asynchronous serial interface transmission status register 6 (ASIF6), bit 7 (SBRF6) and bit 6 (SBRT6) of asynchronous serial interface control register 6 (ASICL6), and receive buffer register 6 (RXB6) are reset.

**Caution Clear POWER6 to 0 after clearing TXE6 and RXE6 to 0 to stop the operation.**
**To start the communication, set POWER6 to 1, and then set TXE6 or RXE6 to 1.**

**Remark** To use the RxD6/SDAA0/P61 and TxD6/SCLA0/P60 pins of the 78K0/KY2-L, 78K0/KA2-L and the RxD6/P14 and TxD6/P13 pins of the 78K0/KB2-L, 78K0/KC2-L as general-purpose port pins, refer to **CHAPTER 4 PORT FUNCTIONS**.

### 14.4.2 Asynchronous serial interface (UART) mode

In this mode, data of 1 byte is transmitted/received following a start bit, and a full-duplex operation can be performed.

A dedicated UART baud rate generator is incorporated, so that communication can be executed at a wide range of baud rates.

#### (1) Registers used

- Asynchronous serial interface operation mode register 6 (ASIM6)
- Asynchronous serial interface reception error status register 6 (ASIS6)
- Asynchronous serial interface transmission status register 6 (ASIF6)
- Clock selection register 6 (CKSR6)
- Baud rate generator control register 6 (BRGC6)
- Asynchronous serial interface control register 6 (ASICL6)
- Input switch control register (ISC)
- Port mode register 1 (PM1), port mode register 6 (PM6)[Note 1]
- Port register 1 (P1), port register 6 (P6)[Note 1]
- Port output mode register 6 (POM6)[Note 2]

**Notes 1.** 78K0/KY2-L, 78K0/KA2-L: Port mode register 6 (PM6), port register 6 (P6)
78K0/KB2-L, 78K0/KC2-L: Port mode register 1 (PM1), port register 1 (P1)

**2.** In the 78K0/KY2-L and 78K0/KA2-L, this register is used when using serial interface UART6.

The basic procedure of setting an operation in the UART mode is as follows.

<1> Set the CKSR6 register (refer to **Figure 14-8**).
<2> Set the BRGC6 register (refer to **Figure 14-9**).
<3> Set bits 0 to 4 (ISRM6, SL6, CL6, PS60, PS61) of the ASIM6 register (refer to **Figure 14-5**).
<4> Set bits 0 and 1 (TXDLV6, DIR6) of the ASICL6 register (refer to **Figure 14-10**).
<5> Set bit 7 (POWER6) of the ASIM6 register to 1.
<6> Set bit 6 (TXE6) of the ASIM6 register to 1. → Transmission is enabled.
Set bit 5 (RXE6) of the ASIM6 register to 1. → Reception is enabled.
<7> Write data to transmit buffer register 6 (TXB6). → Data transmission is started.

**Caution Take relationship with the other party of communication when setting the port mode register and port register.**

The relationship between the register settings and pins is shown below.

**Table 14-2. Relationship Between Register Settings and Pins**

**(1) 78K0/KY2-L and 78K0/KA2-L**

| POWER6 | TXE6 | RXE6 | PM60 | P60 | PM61 | P61 | POM60 | POM61 | UART6 Operation | Pin Function | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | TxD6/ SCLA0/ P60 | RxD6/ SDAA0/ P61 |
| 0 | 0 | 0 | ×Note | ×Note | ×Note | ×Note | ×Note | ×Note | Stop | P60 | P61 |
| | | | 0 | 1 | 0 | 1 | 1 | 1 | | SCLA0 | SDAA0 |
| 1 | 0 | 1 | ×Note | ×Note | 1 | × | ×Note | × | Reception | P60 | RxD6 |
| | 1 | 0 | 0 | 1 | ×Note | ×Note | 0 | ×Note | Transmission | TxD6 | P61 |
| | 1 | 1 | 0 | 1 | 1 | × | 0 | × | Transmission/ reception | TxD6 | RxD6 |

**Note** Can be set as port function.

**(2) 78K0/KB2-L and 78K0/KC2-L**

| POWER6 | TXE6 | RXE6 | PM13 | P13 | PM14 | P14 | UART6 Operation | Pin Function | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | TxD6/P13 | RxD6/P14 |
| 0 | 0 | 0 | ×Note | ×Note | ×Note | ×Note | Stop | P13 | P14 |
| 1 | 0 | 1 | ×Note | ×Note | 1 | × | Reception | P13 | RxD6 |
| | 1 | 0 | 0 | 1 | ×Note | ×Note | Transmission | TxD6 | P14 |
| | 1 | 1 | 0 | 1 | 1 | × | Transmission/ reception | TxD6 | RxD6 |

**Note** Can be set as port function.

**Remark** ×: don't care
POWER6: Bit 7 of asynchronous serial interface operation mode register 6 (ASIM6)
TXE6: Bit 6 of ASIM6
RXE6: Bit 5 of ASIM6
PM6×, PM1×: Port mode register
P6×, P1×: Port output latch
POM60, POM61: Bits 0 and 1 of port output mode register 6 (POM6)

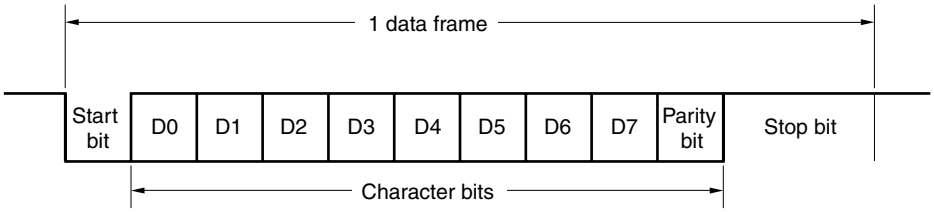
### (2) Communication operation

#### (a) Format and waveform example of normal transmit/receive data

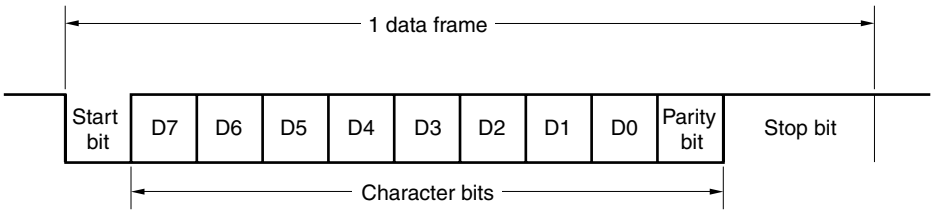
Figures 14-15 and 14-16 show the format and waveform example of the normal transmit/receive data.

**Figure 14-15. Format of Normal UART Transmit/Receive Data**

**1. LSB-first transmission/reception**




**2. MSB-first transmission/reception**




One data frame consists of the following bits.

- Start bit ... 1 bit
- Character bits ... 7 or 8 bits
- Parity bit ... Even parity, odd parity, 0 parity, or no parity
- Stop bit ... 1 or 2 bits

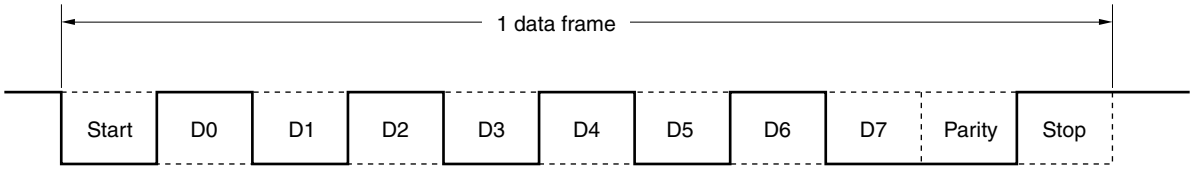
The character bit length, parity, and stop bit length in one data frame are specified by asynchronous serial interface operation mode register 6 (ASIM6).

Whether data is communicated with the LSB or MSB first is specified by bit 1 (DIR6) of asynchronous serial interface control register 6 (ASICL6).

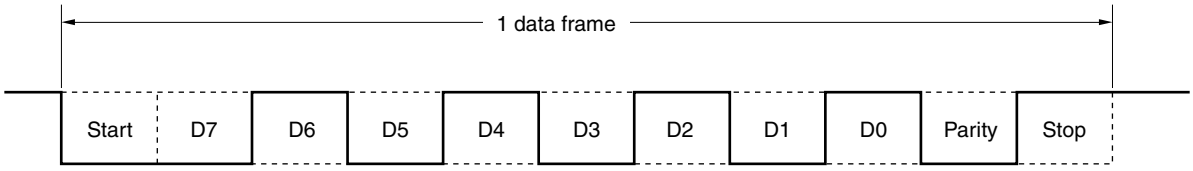
Whether the TxD6 pin outputs normal or inverted data is specified by bit 0 (TXDLV6) of ASICL6.

**Figure 14-16. Example of Normal UART Transmit/Receive Data Waveform**

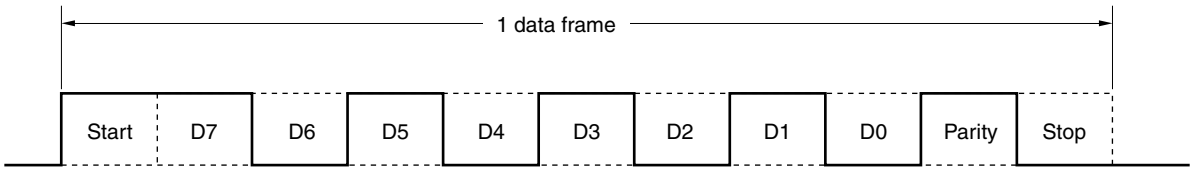
1. **Data length: 8 bits, LSB first, Parity: Even parity, Stop bit: 1 bit, Communication data: 55H**



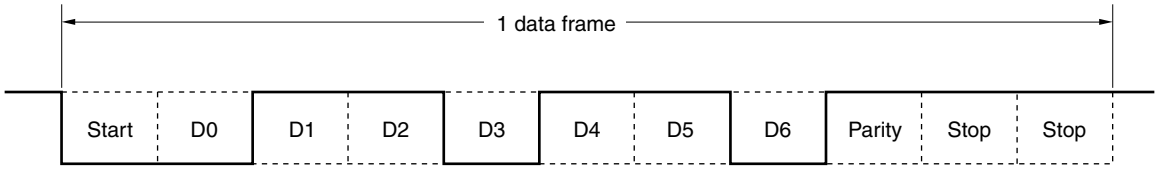

2. **Data length: 8 bits, MSB first, Parity: Even parity, Stop bit: 1 bit, Communication data: 55H**



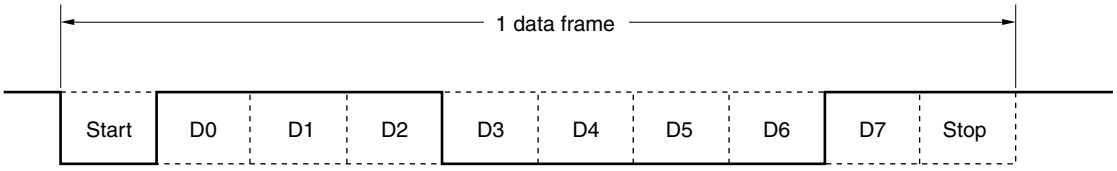

3. **Data length: 8 bits, MSB first, Parity: Even parity, Stop bit: 1 bit, Communication data: 55H, TxD6 pin inverted output**




4. **Data length: 7 bits, LSB first, Parity: Odd parity, Stop bit: 2 bits, Communication data: 36H**




5. **Data length: 8 bits, LSB first, Parity: None, Stop bit: 1 bit, Communication data: 87H**



**(b) Parity types and operation**

The parity bit is used to detect a bit error in communication data. Usually, the same type of parity bit is used on both the transmission and reception sides. With even parity and odd parity, a 1-bit (odd number) error can be detected. With zero parity and no parity, an error cannot be detected.

**Caution Fix the PS61 and PS60 bits to 0 when the device is used in LIN communication operation.**

**(i) Even parity**

- Transmission

  Transmit data, including the parity bit, is controlled so that the number of bits that are "1" is even.
  The value of the parity bit is as follows.

  If transmit data has an odd number of bits that are "1": 1
  If transmit data has an even number of bits that are "1": 0

- Reception

  The number of bits that are "1" in the receive data, including the parity bit, is counted. If it is odd, a parity error occurs.

**(ii) Odd parity**

- Transmission

  Unlike even parity, transmit data, including the parity bit, is controlled so that the number of bits that are "1" is odd.

  If transmit data has an odd number of bits that are "1": 0
  If transmit data has an even number of bits that are "1": 1

- Reception

  The number of bits that are "1" in the receive data, including the parity bit, is counted. If it is even, a parity error occurs.

**(iii) 0 parity**

The parity bit is cleared to 0 when data is transmitted, regardless of the transmit data.
The parity bit is not detected when the data is received. Therefore, a parity error does not occur regardless of whether the parity bit is "0" or "1".

**(iv) No parity**

No parity bit is appended to the transmit data.
Reception is performed assuming that there is no parity bit when data is received. Because there is no parity bit, a parity error does not occur.

**(c) Normal transmission**

When bit 7 (POWER6) of asynchronous serial interface operation mode register 6 (ASIM6) is set to 1 and bit 6 (TXE6) of ASIM6 is then set to 1, transmission is enabled. Transmission can be started by writing transmit data to transmit buffer register 6 (TXB6). The start bit, parity bit, and stop bit are automatically appended to the data.

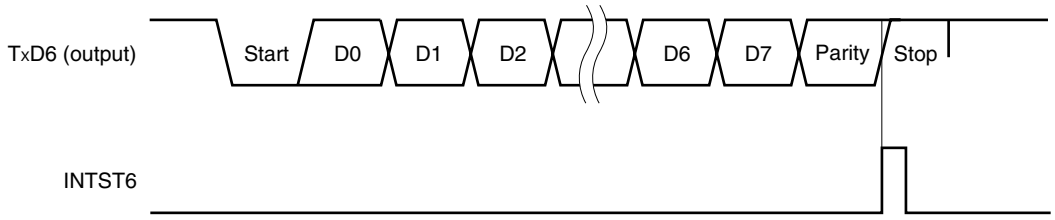
When transmission is started, the data in TXB6 is transferred to transmit shift register 6 (TXS6). After that, the transmit data is sequentially output from TXS6 to the TxD6 pin. When transmission is completed, the parity and stop bits set by ASIM6 are appended and a transmission completion interrupt request (INTST6) is generated.

Transmission is stopped until the data to be transmitted next is written to TXB6.

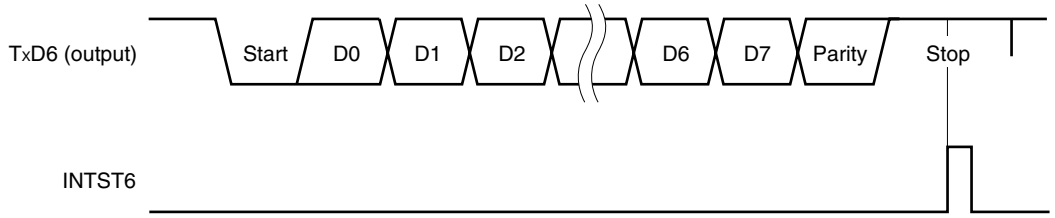
Figure 14-17 shows the timing of the transmission completion interrupt request (INTST6). This interrupt occurs as soon as the last stop bit has been output.

**Figure 14-17. Normal Transmission Completion Interrupt Request Timing**

**1. Stop bit length: 1**




**2. Stop bit length: 2**



**(d) Continuous transmission**

The next transmit data can be written to transmit buffer register 6 (TXB6) as soon as transmit shift register 6 (TXS6) has started its shift operation. Consequently, even while the INTST6 interrupt is being serviced after transmission of one data frame, data can be continuously transmitted and an efficient communication rate can be realized. In addition, the TXB6 register can be efficiently written twice (2 bytes) without having to wait for the transmission time of one data frame, by reading bit 0 (TXSF6) of asynchronous serial interface transmission status register 6 (ASIF6) when the transmission completion interrupt has occurred.

To transmit data continuously, be sure to reference the ASIF6 register to check the transmission status and whether the TXB6 register can be written, and then write the data.

**Cautions 1. The TXBF6 and TXSF6 flags of the ASIF6 register change from "10" to "11", and to "01" during continuous transmission. To check the status, therefore, do not use a combination of the TXBF6 and TXSF6 flags for judgment. Read only the TXBF6 flag when executing continuous transmission.**

**2. When the device is use in LIN communication operation, the continuous transmission function cannot be used. Make sure that asynchronous serial interface transmission status register 6 (ASIF6) is 00H before writing transmit data to transmit buffer register 6 (TXB6).**

| TXBF6 | Writing to TXB6 Register |
|---|---|
| 0 | Writing enabled |
| 1 | Writing disabled |

**Caution To transmit data continuously, write the first transmit data (first byte) to the TXB6 register. Be sure to check that the TXBF6 flag is "0". If so, write the next transmit data (second byte) to the TXB6 register. If data is written to the TXB6 register while the TXBF6 flag is "1", the transmit data cannot be guaranteed.**

The communication status can be checked using the TXSF6 flag.

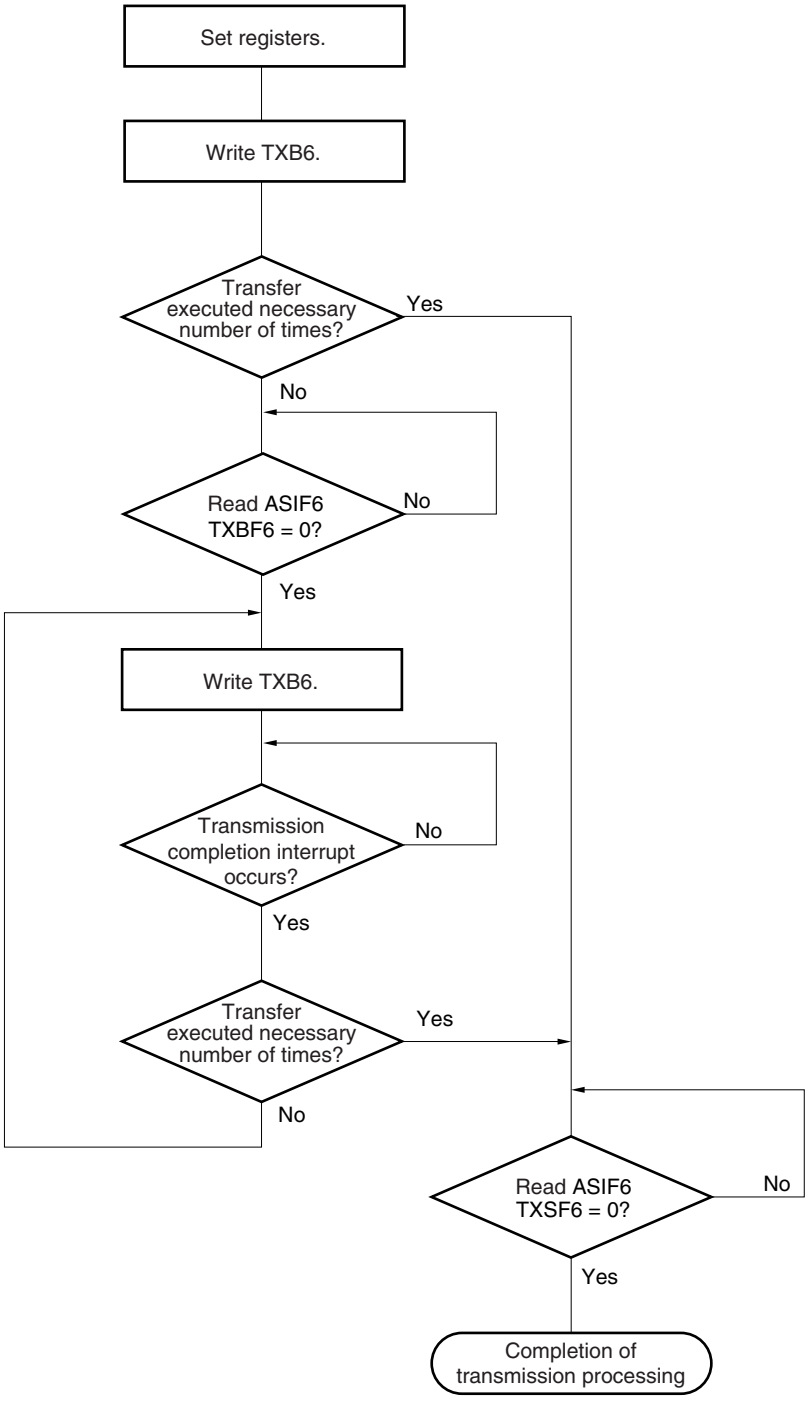
| TXSF6 | Transmission Status |
|---|---|
| 0 | Transmission is completed. |
| 1 | Transmission is in progress. |

**Cautions 1. To initialize the transmission unit upon completion of continuous transmission, be sure to check that the TXSF6 flag is "0" after generation of the transmission completion interrupt, and then execute initialization. If initialization is executed while the TXSF6 flag is "1", the transmit data cannot be guaranteed.**

**2. During continuous transmission, the next transmission may complete before execution of INTST6 interrupt servicing after transmission of one data frame. As a countermeasure, detection can be performed by developing a program that can count the number of transmit data and by referencing the TXSF6 flag.**

Figure 14-18 shows an example of the continuous transmission processing flow.

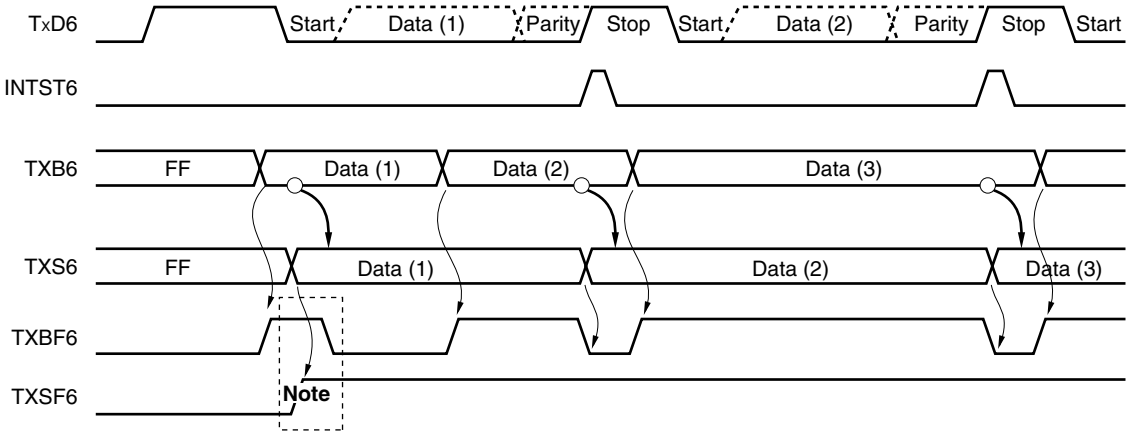
**Figure 14-18. Example of Continuous Transmission Processing Flow**




**Remark** TXB6: Transmit buffer register 6
ASIF6: Asynchronous serial interface transmission status register 6
TXBF6: Bit 1 of ASIF6 (transmit buffer data flag)
TXSF6: Bit 0 of ASIF6 (transmit shift register data flag)

Figure 14-19 shows the timing of starting continuous transmission, and Figure 14-20 shows the timing of ending continuous transmission.

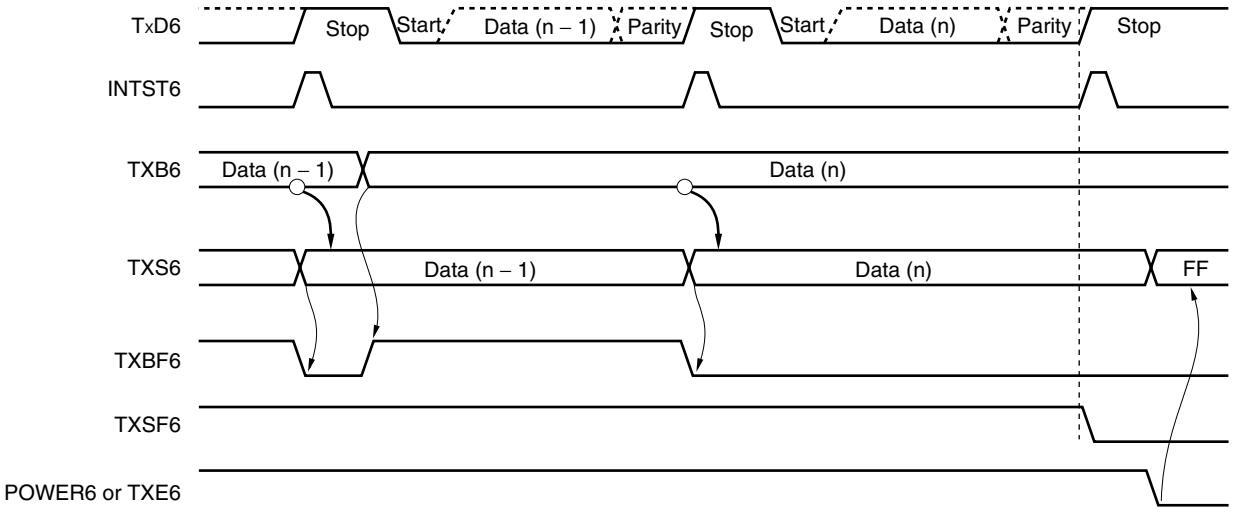
**Figure 14-19. Timing of Starting Continuous Transmission**




**Note** When ASIF6 is read, there is a period in which TXBF6 and TXSF6 = 1, 1. Therefore, judge whether writing is enabled using only the TXBF6 bit.

**Remark** TxD6: TxD6 pin (output)
INTST6: Interrupt request signal
TXB6: Transmit buffer register 6
TXS6: Transmit shift register 6
ASIF6: Asynchronous serial interface transmission status register 6
TXBF6: Bit 1 of ASIF6
TXSF6: Bit 0 of ASIF6

**Figure 14-20. Timing of Ending Continuous Transmission**




**Remark** TxD6: TxD6 pin (output)
INTST6: Interrupt request signal
TXB6: Transmit buffer register 6
TXS6: Transmit shift register 6
ASIF6: Asynchronous serial interface transmission status register 6
TXBF6: Bit 1 of ASIF6
TXSF6: Bit 0 of ASIF6
POWER6: Bit 7 of asynchronous serial interface operation mode register (ASIM6)
TXE6: Bit 6 of asynchronous serial interface operation mode register (ASIM6)

**(e) Normal reception**

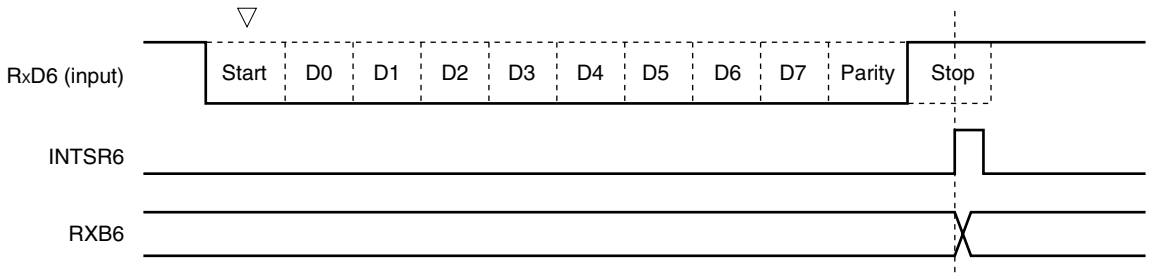
Reception is enabled and the RxD6 pin input is sampled when bit 7 (POWER6) of asynchronous serial interface operation mode register 6 (ASIM6) is set to 1 and then bit 5 (RXE6) of ASIM6 is set to 1.

The 8-bit counter of the baud rate generator starts counting when the falling edge of the RxD6 pin input is detected. When the set value of baud rate generator control register 6 (BRGC6) has been counted, the RxD6 pin input is sampled again ( in Figur▽14-21). If the RxD6 pin is low level at this time, it is recognized as a start bit.

When the start bit is detected, reception is started, and serial data is sequentially stored in the receive shift register 6 (RXS6) at the set baud rate. When the stop bit has been received, the reception completion interrupt (INTSR6) is generated and the data of RXS6 is written to receive buffer register 6 (RXB6). If an overrun error (OVE6) occurs, however, the receive data is not written to RXB6.

Even if a parity error (PE6) occurs while reception is in progress, reception continues to the reception position of the stop bit, and a reception error interrupt (INTSR6/INTSRE6) is generated on completion of reception.

**Figure 14-21. Reception Completion Interrupt Request Timing**




**Cautions 1. If a reception error occurs, read ASIS6 and then RXB6 to clear the error flag. Otherwise, an overrun error will occur when the next data is received, and the reception error status will persist.**

**2. Reception is always performed with the "number of stop bits = 1". The second stop bit is ignored.**

**3. Be sure to read asynchronous serial interface reception error status register 6 (ASIS6) before reading RXB6.**

**(f) Reception error**

Three types of errors may occur during reception: a parity error, framing error, or overrun error. If the error flag of asynchronous serial interface reception error status register 6 (ASIS6) is set as a result of data reception, a reception error interrupt request (INTSR6/INTSRE6) is generated.

Which error has occurred during reception can be identified by reading the contents of ASIS6 in the reception error interrupt (INTSR6/INTSRE6) servicing (refer to **Figure 14-6**).

The contents of ASIS6 are cleared to 0 when ASIS6 is read.

**Table 14-3. Cause of Reception Error**

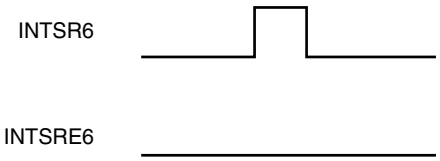
| Reception Error | Cause |
|---|---|
| Parity error | The parity specified for transmission does not match the parity of the receive data. |
| Framing error | Stop bit is not detected. |
| Overrun error | Reception of the next data is completed before data is read from receive buffer register 6 (RXB6). |

The reception error interrupt can be separated into reception completion interrupt (INTSR6) and error interrupt (INTSRE6) by clearing bit 0 (ISRM6) of asynchronous serial interface operation mode register 6 (ASIM6) to 0.

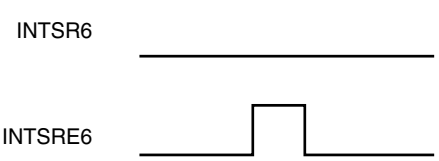
**Figure 14-22. Reception Error Interrupt**

**1. If ISRM6 is cleared to 0 (reception completion interrupt (INTSR6) and error interrupt (INTSRE6) are separated)**

**(a) No error during reception**

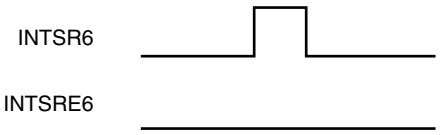



**(b) Error during reception**

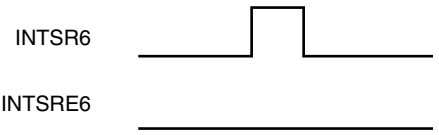



**2. If ISRM6 is set to 1 (error interrupt is included in INTSR6)**

**(a) No error during reception**




**(b) Error during reception**



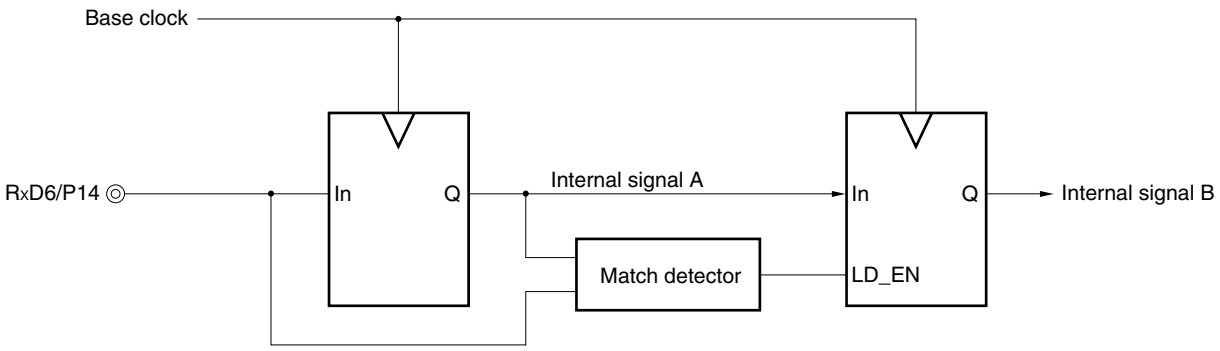
**(g) Noise filter of receive data**

The RxD6 signal is sampled with the base clock output by the prescaler block.

If two sampled values are the same, the output of the match detector changes, and the data is sampled as input data.

Because the circuit is configured as shown in Figure 14-23, the internal processing of the reception operation is delayed by two clocks from the external signal status.

**Figure 14-23. Noise Filter Circuit**




**(h) SBF transmission**

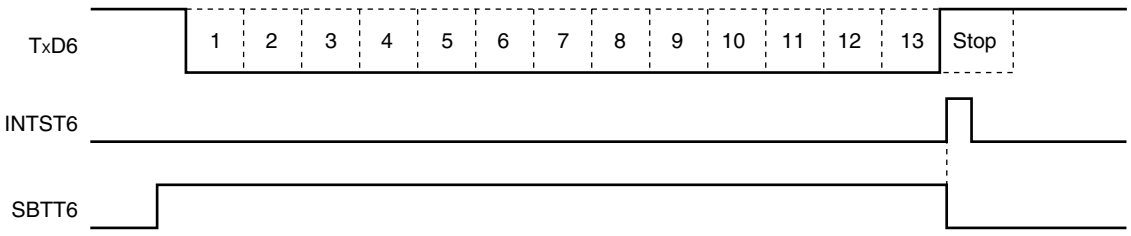
When the device is use in LIN communication operation, the SBF (Synchronous Break Field) transmission control function is used for transmission. For the transmission operation of LIN, refer to **Figure 14-1 LIN Transmission Operation**.

When bit 7 (POWER6) of asynchronous serial interface mode register 6 (ASIM6) is set to 1, the TxD6 pin outputs high level. Next, when bit 6 (TXE6) of ASIM6 is set to 1, the transmission enabled status is entered, and SBF transmission is started by setting bit 5 (SBTT6) of asynchronous serial interface control register 6 (ASICL6) to 1.

Thereafter, a low level of bits 13 to 20 (set by bits 4 to 2 (SBL62 to SBL60) of ASICL6) is output. Following the end of SBF transmission, the transmission completion interrupt request (INTST6) is generated and SBTT6 is automatically cleared. Thereafter, the normal transmission mode is restored.

Transmission is suspended until the data to be transmitted next is written to transmit buffer register 6 (TXB6), or until SBTT6 is set to 1.

**Figure 14-24. SBF Transmission**




**Remark** TxD6: TxD6 pin (output)
INTST6: Transmission completion interrupt request
SBTT6: Bit 5 of asynchronous serial interface control register 6 (ASICL6)

**(i) SBF reception**

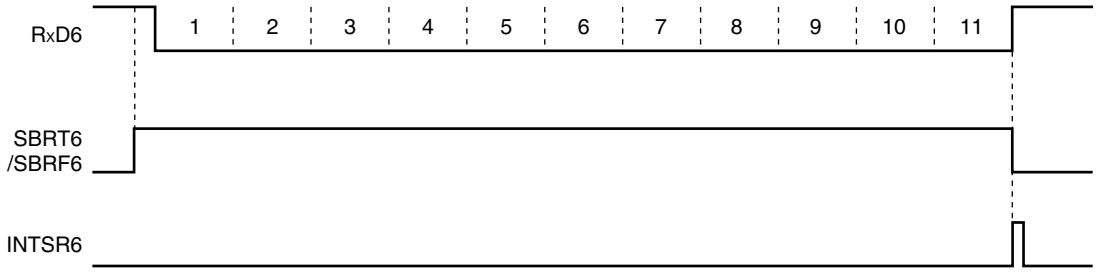
When the device is used in LIN communication operation, the SBF (Synchronous Break Field) reception control function is used for reception. For the reception operation of LIN, refer to **Figure 14-2 LIN Reception Operation**.

Reception is enabled when bit 7 (POWER6) of asynchronous serial interface operation mode register 6 (ASIM6) is set to 1 and then bit 5 (RXE6) of ASIM6 is set to 1. SBF reception is enabled when bit 6 (SBRT6) of asynchronous serial interface control register 6 (ASICL6) is set to 1. In the SBF reception enabled status, the RxD6 pin is sampled and the start bit is detected in the same manner as the normal reception enable status.

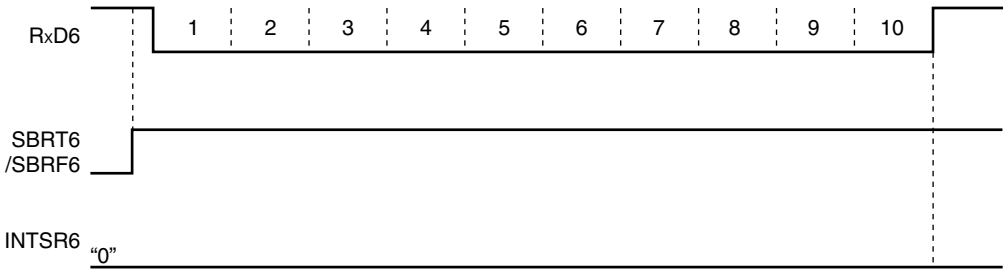
When the start bit has been detected, reception is started, and serial data is sequentially stored in the receive shift register 6 (RXS6) at the set baud rate. When the stop bit is received and if the width of SBF is 11 bits or more, a reception completion interrupt request (INTSR6) is generated as normal processing. At this time, the SBRF6 and SBRT6 bits are automatically cleared, and SBF reception ends. Detection of errors, such as OVE6, PE6, and FE6 (bits 0 to 2 of asynchronous serial interface reception error status register 6 (ASIS6)) is suppressed, and error detection processing of UART communication is not performed. In addition, data transfer between receive shift register 6 (RXS6) and receive buffer register 6 (RXB6) is not performed, and the reset value of FFH is retained. If the width of SBF is 10 bits or less, an interrupt does not occur as error processing after the stop bit has been received, and the SBF reception mode is restored. In this case, the SBRF6 and SBRT6 bits are not cleared.

**Figure 14-25. SBF Reception**

**1. Normal SBF reception (stop bit is detected with a width of more than 10.5 bits)**




**2. SBF reception error (stop bit is detected with a width of 10.5 bits or less)**




**Remark** RxD6: RxD6 pin (input)
SBRT6: Bit 6 of asynchronous serial interface control register 6 (ASICL6)
SBRF6: Bit 7 of ASICL6
INTSR6: Reception completion interrupt request

### 14.4.3 Dedicated baud rate generator

The dedicated baud rate generator consists of a source clock selector and an 8-bit programmable counter, and generates a serial clock for transmission/reception of UART6.

Separate 8-bit counters are provided for transmission and reception.

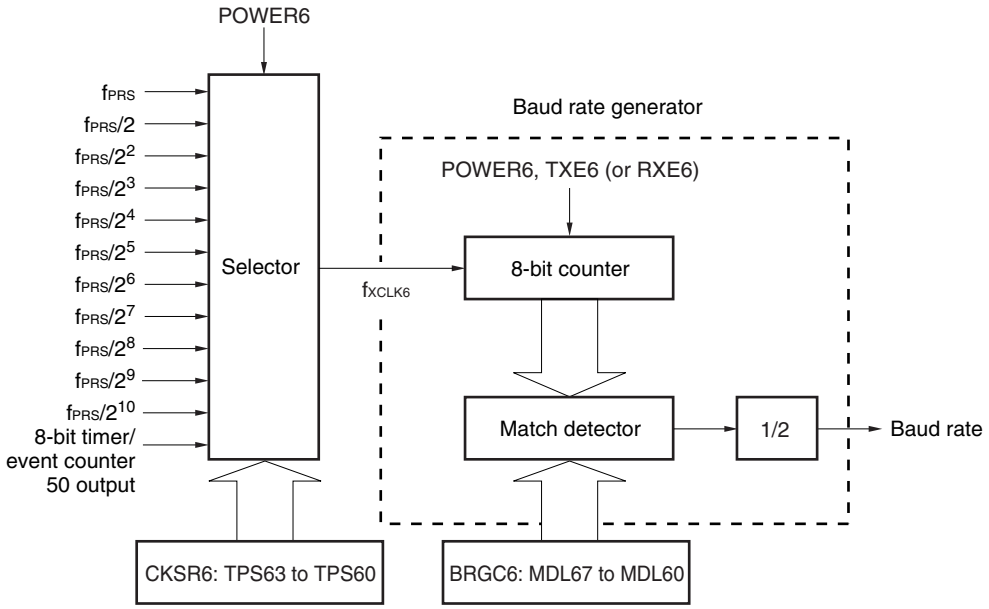
#### (1) Configuration of baud rate generator

- Base clock
  The clock selected by bits 3 to 0 (TPS63 to TPS60) of clock selection register 6 (CKSR6) is supplied to each module when bit 7 (POWER6) of asynchronous serial interface operation mode register 6 (ASIM6) is 1. This clock is called the base clock and its frequency is called $f_{XCLK6}$. The base clock is fixed to low level when POWER6 = 0.

- Transmission counter
  This counter stops operation, cleared to 0, when bit 7 (POWER6) or bit 6 (TXE6) of asynchronous serial interface operation mode register 6 (ASIM6) is 0.
  It starts counting when POWER6 = 1 and TXE6 = 1.
  The counter is cleared to 0 when the first data transmitted is written to transmit buffer register 6 (TXB6).
  If data are continuously transmitted, the counter is cleared to 0 again when one frame of data has been completely transmitted. If there is no data to be transmitted next, the counter is not cleared to 0 and continues counting until POWER6 or TXE6 is cleared to 0.

- Reception counter
  This counter stops operation, cleared to 0, when bit 7 (POWER6) or bit 5 (RXE6) of asynchronous serial interface operation mode register 6 (ASIM6) is 0.
  It starts counting when the start bit has been detected.
  The counter stops operation after one frame has been received, until the next start bit is detected.

**Figure 14-26. Configuration of Baud Rate Generator**




**Remark** POWER6: Bit 7 of asynchronous serial interface operation mode register 6 (ASIM6)
TXE6: Bit 6 of ASIM6
RXE6: Bit 5 of ASIM6
CKSR6: Clock selection register 6
BRGC6: Baud rate generator control register 6

**(2) Generation of serial clock**

A serial clock to be generated can be specified by using clock selection register 6 (CKSR6) and baud rate generator control register 6 (BRGC6).

The clock to be input to the 8-bit counter can be set by bits 3 to 0 (TPS63 to TPS60) of CKSR6 and the division value ($f_{XCLK6}/4$ to $f_{XCLK6}/255$) of the 8-bit counter can be set by bits 7 to 0 (MDL67 to MDL60) of BRGC6.

### 14.4.4 Calculation of baud rate

#### (1) Baud rate calculation expression

The baud rate can be calculated by the following expression.

- Baud rate = $\frac{f_{XCLK6}}{2 \times k}$ [bps]

$f_{XCLK6}$: Frequency of base clock selected by TPS63 to TPS60 bits of CKSR6 register

k: Value set by MDL67 to MDL60 bits of BRGC6 register (k = 4, 5, 6, ..., 255)

**Table 14-4. Set Value of TPS63 to TPS60**

| TPS63 | TPS62 | TPS61 | TPS60 | Base Clock ($f_{XCLK6}$) Selection[Note 1] | | | |
|---|---|---|---|---|---|---|---|
| | | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz |
| 0 | 0 | 0 | 0 | $f_{PRS}$ | 2 MHz | 5 MHz | 10 MHz |
| 0 | 0 | 0 | 1 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz |
| 0 | 0 | 1 | 0 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz |
| 0 | 0 | 1 | 1 | $f_{PRS}/2^3$ | 250 kHz | 625 kHz | 1.25 MHz |
| 0 | 1 | 0 | 0 | $f_{PRS}/2^4$ | 125 kHz | 312.5 kHz | 625 kHz |
| 0 | 1 | 0 | 1 | $f_{PRS}/2^5$ | 62.5 kHz | 156.25 kHz | 312.5 kHz |
| 0 | 1 | 1 | 0 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz |
| 0 | 1 | 1 | 1 | $f_{PRS}/2^7$ | 15.625 kHz | 39.06 kHz | 78.13 kHz |
| 1 | 0 | 0 | 0 | $f_{PRS}/2^8$ | 7.813 kHz | 19.53 kHz | 39.06 kHz |
| 1 | 0 | 0 | 1 | $f_{PRS}/2^9$ | 3.906 kHz | 9.77 kHz | 19.53 kHz |
| 1 | 0 | 1 | 0 | $f_{PRS}/2^{10}$ | 1.953 kHz | 4.88 kHz | 9.77 kHz |
| 1 | 0 | 1 | 1 | TM50 output[Note 2] | | | |
| Other than above | | | | Setting prohibited | | | |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** Note the following points when selecting the TM50 output as the base clock.

- Mode in which the count clock is cleared and started upon a match of TM50 and CR50 (TMC506 = 0)

  Start the operation of 8-bit timer/event counter 50 first and then enable the timer F/F inversion operation (TMC501 = 1).
- PWM mode (TMC506 = 1)

  Start the operation of 8-bit timer/event counter 50 first and then set the count clock to make the duty = 50%.

It is not necessary to enable (TOE50 = 1) TO50 output in any mode.

### (2) Error of baud rate

The baud rate error can be calculated by the following expression.

- Error (%) = $\left(\frac{\text{Actual baud rate (baud rate with error)}}{\text{Desired baud rate (correct baud rate)}} - 1\right) \times 100$ [%]

**Cautions 1. Keep the baud rate error during transmission to within the permissible error range at the reception destination.**

**2. Make sure that the baud rate error during reception satisfies the range shown in (4) Permissible baud rate range during reception.**

**Example:** Frequency of base clock = 10 MHz = 10,000,000 Hz
Set value of MDL67 to MDL60 bits of BRGC6 register = 00100001B (k = 33)
Target baud rate = 153600 bps

Baud rate = 10 M / (2 × 33)
= 10000000 / (2 × 33) = 151,515 [bps]

Error = (151515/153600 − 1) × 100
= −1.357 [%]

### (3) Example of setting baud rate

**Table 14-5. Set Data of Baud Rate Generator**

| Baud Rate [bps] | $f_{PRS}$ = 2.0 MHz | | | | $f_{PRS}$ = 5.0 MHz | | | | $f_{PRS}$ = 10.0 MHz | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | TPS63-TPS60 | k | Calculated Value | ERR [%] | TPS63-TPS60 | k | Calculated Value | ERR [%] | TPS63-TPS60 | k | Calculated Value | ERR [%] |
| 300 | 8H | 13 | 301 | 0.16 | 7H | 65 | 301 | 0.16 | 8H | 65 | 301 | 0.16 |
| 600 | 7H | 13 | 601 | 0.16 | 6H | 65 | 601 | 0.16 | 7H | 65 | 601 | 0.16 |
| 1200 | 6H | 13 | 1202 | 0.16 | 5H | 65 | 1202 | 0.16 | 6H | 65 | 1202 | 0.16 |
| 2400 | 5H | 13 | 2404 | 0.16 | 4H | 65 | 2404 | 0.16 | 5H | 65 | 2404 | 0.16 |
| 4800 | 4H | 13 | 4808 | 0.16 | 3H | 65 | 4808 | 0.16 | 4H | 65 | 4808 | 0.16 |
| 9600 | 3H | 13 | 9615 | 0.16 | 2H | 65 | 9615 | 0.16 | 3H | 65 | 9615 | 0.16 |
| 19200 | 2H | 13 | 19231 | 0.16 | 1H | 65 | 19231 | 0.16 | 2H | 65 | 19231 | 0.16 |
| 24000 | 1H | 21 | 23810 | −0.79 | 3H | 13 | 24038 | 0.16 | 4H | 13 | 24038 | 0.16 |
| 31250 | 1H | 16 | 31250 | 0 | 4H | 5 | 31250 | 0 | 5H | 5 | 31250 | 0 |
| 38400 | 1H | 13 | 38462 | 0.16 | 0H | 65 | 38462 | 0.16 | 1H | 65 | 38462 | 0.16 |
| 48000 | 0H | 21 | 47619 | −0.79 | 2H | 13 | 48077 | 0.16 | 3H | 13 | 48077 | 0.16 |
| 76800 | 0H | 13 | 76923 | 0.16 | 0H | 33 | 75758 | −1.36 | 0H | 65 | 76923 | 0.16 |
| 115200 | 0H | 9 | 111111 | −3.55 | 1H | 11 | 113636 | −1.36 | 0H | 43 | 116279 | 0.94 |
| 153600 | − | − | − | − | 1H | 8 | 156250 | 1.73 | 0H | 33 | 151515 | −1.36 |
| 312500 | − | − | − | − | 0H | 8 | 312500 | 0 | 1H | 8 | 312500 | 0 |
| 625000 | − | − | − | − | 0H | 4 | 625000 | 0 | 1H | 4 | 625000 | 0 |

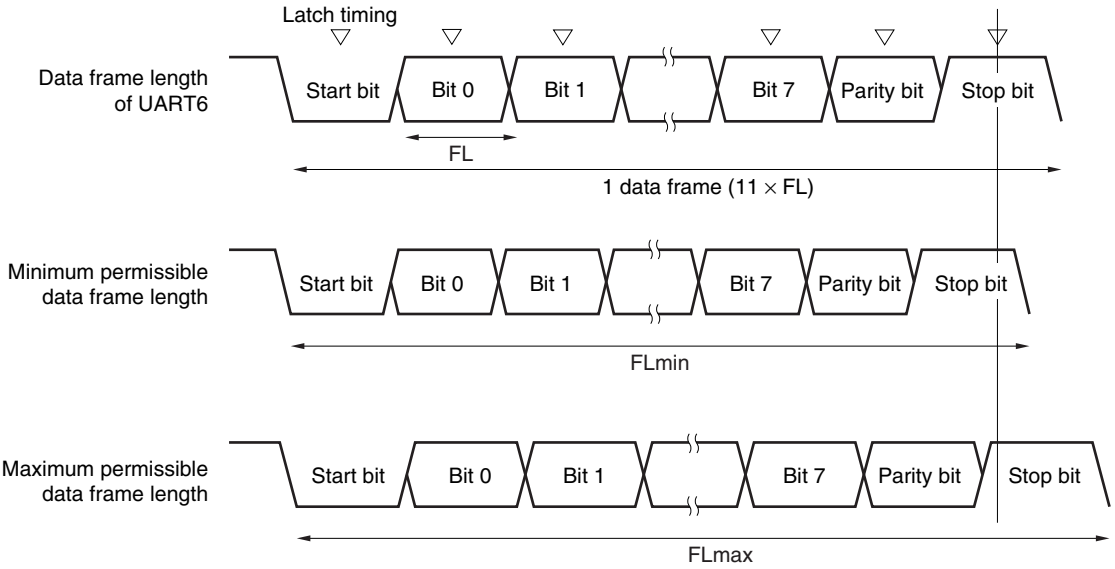
**Remark** TPS63 to TPS60: Bits 3 to 0 of clock selection register 6 (CKSR6) (setting of base clock ($f_{XCLK6}$))
k: Value set by MDL67 to MDL60 bits of baud rate generator control register 6 (BRGC6) (k = 4, 5, 6, ..., 255)
$f_{PRS}$: Peripheral hardware clock frequency
ERR: Baud rate error

**(4) Permissible baud rate range during reception**

The permissible error from the baud rate at the transmission destination during reception is shown below.

**Caution Make sure that the baud rate error during reception is within the permissible error range, by using the calculation expression shown below.**

**Figure 14-27. Permissible Baud Rate Range During Reception**




As shown in Figure 14-27, the latch timing of the receive data is determined by the counter set by baud rate generator control register 6 (BRGC6) after the start bit has been detected. If the last data (stop bit) meets this latch timing, the data can be correctly received.

Assuming that 11-bit data is received, the theoretical values can be calculated as follows.

$FL = (Brate)^{-1}$

Brate: Baud rate of UART6
k: Set value of BRGC6
FL: 1-bit data length
Margin of latch timing: 2 clocks

$$\text{Minimum permissible data frame length: FLmin} = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$$

Therefore, the maximum receivable baud rate at the transmission destination is as follows.

$$BRmax = (FLmin/11)^{-1} = \frac{22k}{21k+2} \text{ Brate}$$

Similarly, the maximum permissible data frame length can be calculated as follows.

$$\frac{10}{11} \times FLmax = 11 \times FL - \frac{k+2}{2 \times k} \times FL = \frac{21k-2}{2 \times k} FL$$

$$FLmax = \frac{21k-2}{20k} FL \times 11$$

Therefore, the minimum receivable baud rate at the transmission destination is as follows.

$$BRmin = (FLmax/11)^{-1} = \frac{20k}{21k-2} \text{ Brate}$$

The permissible baud rate error between UART6 and the transmission destination can be calculated from the above minimum and maximum baud rate expressions, as follows.

**Table 14-6. Maximum/Minimum Permissible Baud Rate Error**

| Division Ratio (k) | Maximum Permissible Baud Rate Error | Minimum Permissible Baud Rate Error |
|---|---|---|
| 4 | +2.33% | −2.44% |
| 8 | +3.53% | −3.61% |
| 20 | +4.26% | −4.31% |
| 50 | +4.56% | −4.58% |
| 100 | +4.66% | −4.67% |
| 255 | +4.72% | −4.73% |

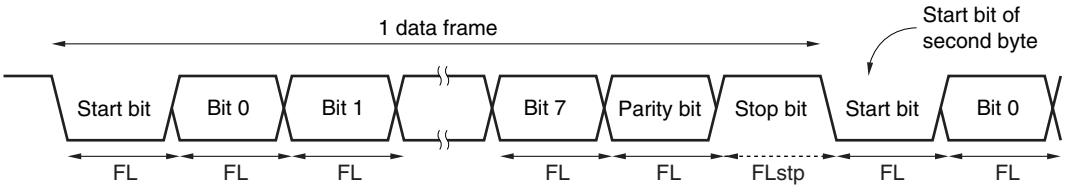
**Remarks 1.** The permissible error of reception depends on the number of bits in one frame, input clock frequency, and division ratio (k). The higher the input clock frequency and the higher the division ratio (k), the higher the permissible error.

**2.** k: Set value of BRGC6

**(5) Data frame length during continuous transmission**

When data is continuously transmitted, the data frame length from a stop bit to the next start bit is extended by two clocks of base clock from the normal value. However, the result of communication is not affected because the timing is initialized on the reception side when the start bit is detected.

**Figure 14-28. Data Frame Length During Continuous Transmission**




Where the 1-bit data length is FL, the stop bit length is FLstp, and base clock frequency is $f_{XCLK6}$, the following expression is satisfied.

$$FLstp = FL + 2/f_{XCLK6}$$

Therefore, the data frame length during continuous transmission is:

$$\text{Data frame length} = 11 \times FL + 2/f_{XCLK6}$$

# CHAPTER 15 SERIAL INTERFACE IICA

## 15.1 Functions of Serial Interface IICA

Serial interface IICA is mounted onto all 78K0/Kx2-L microcontroller products.
Serial interface IICA has the following three modes.

**(1) Operation stop mode**

This mode is used when serial transfers are not performed. It can therefore be used to reduce power consumption.

**(2) $I^2C$ bus mode (multimaster supported)**

This mode is used for 8-bit data transfers with several devices via two lines: a serial clock (SCLA0) line and a serial data bus (SDAA0) line.

This mode complies with the $I^2C$ bus format and the master device can generated “start condition”, “address”, “transfer direction specification”, “data”, and “stop condition” data to the slave device, via the serial data bus. The slave device automatically detects these received status and data by hardware. This function can simplify the part of application program that controls the $I^2C$ bus.

Since the SCLA0 and SDAA0 pins are used for open drain outputs, serial interface IICA requires pull-up resistors for the serial clock line and the serial data bus line.

**(3) Wakeup mode**

The STOP mode can be released by generating an interrupt request signal (INTIICA0) when an extension code from the master device or a local address has been received while in STOP mode. This can be set by using the WUP bit of the IICA control register 1 (IICACTL1).

Figure 15-1 shows a block diagram of serial interface IICA.

**Figure 15-1. Block Diagram of Serial Interface IICA**

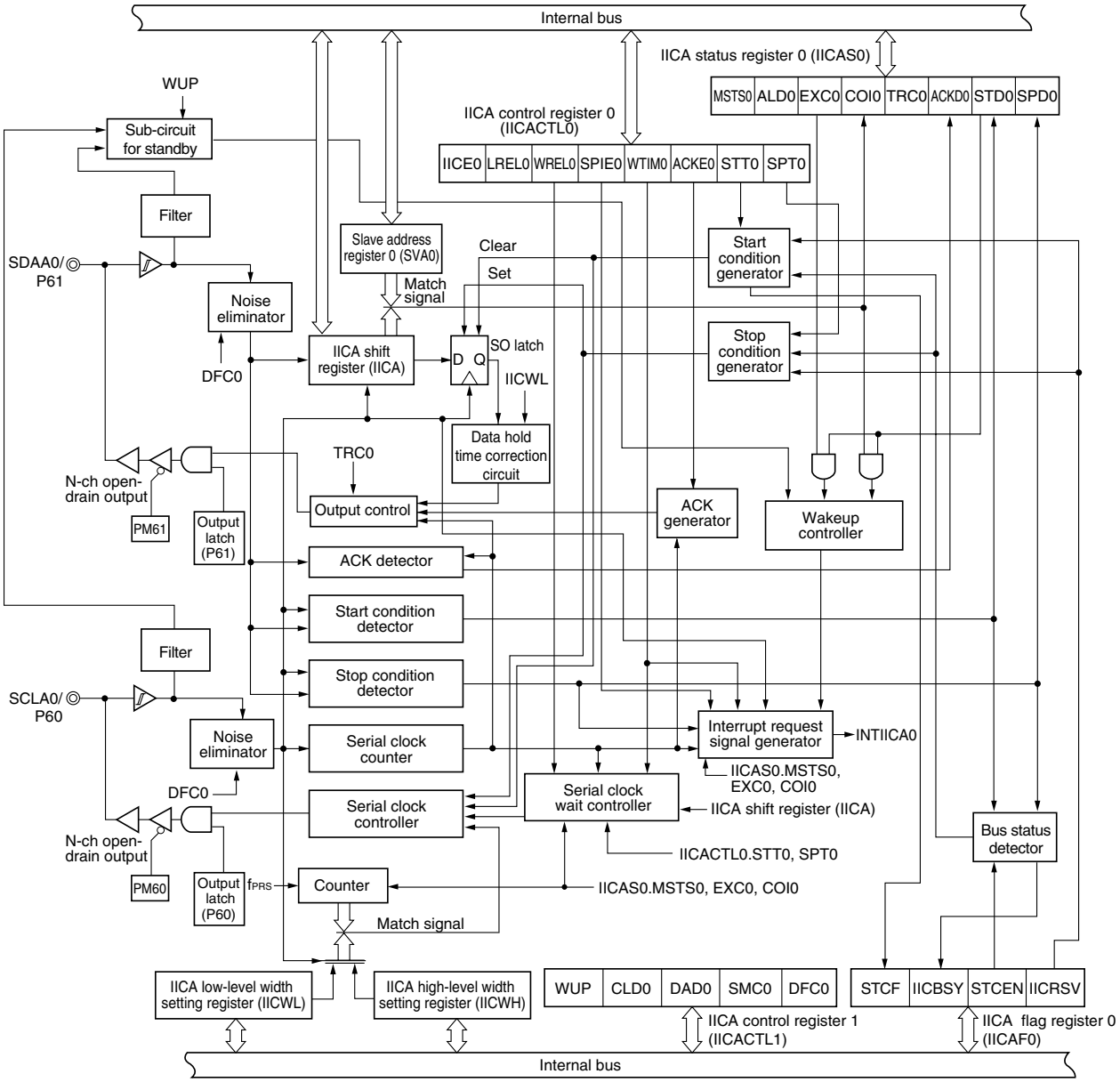
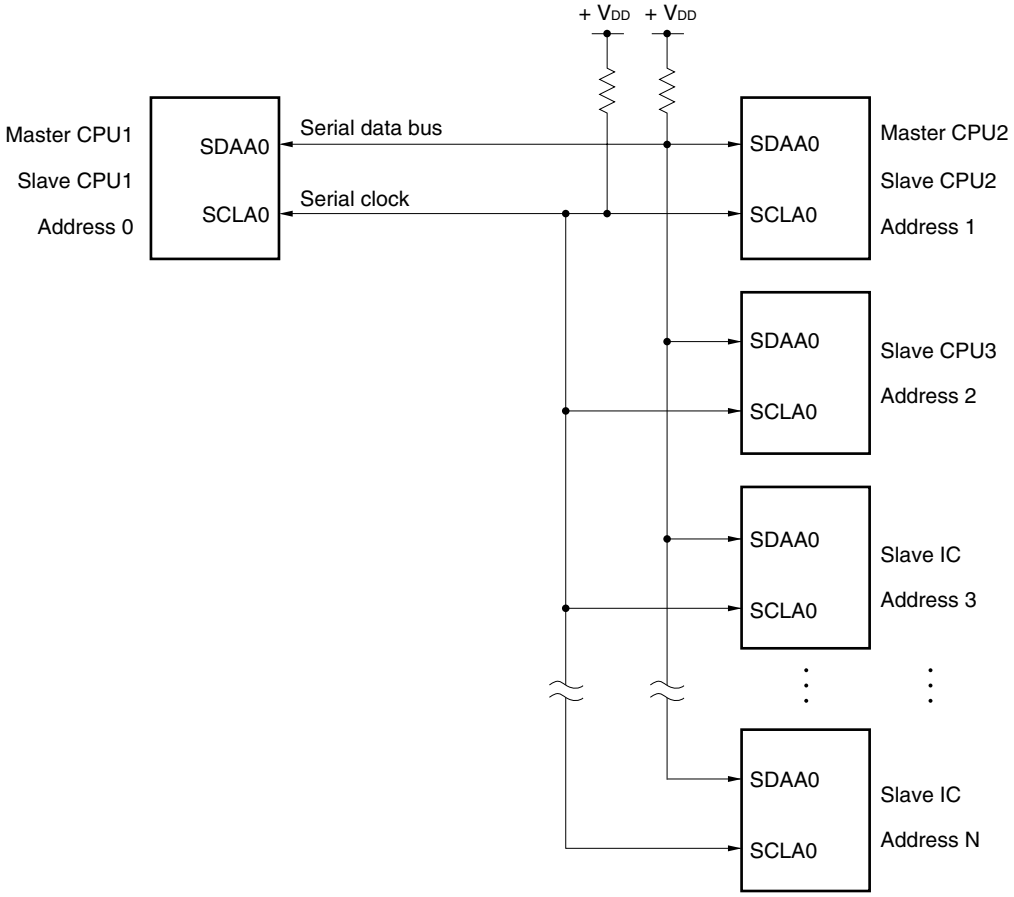


Figure 15-2 shows a serial bus configuration example.

**Figure 15-2. Serial Bus Configuration Example Using I²C Bus**



## 15.2 Configuration of Serial Interface IICA

Serial interface IICA includes the following hardware.

**Table 15-1. Configuration of Serial Interface IICA**

| Item | Configuration |
|---|---|
| Registers | IICA shift register (IICA)<br>Slave address register 0 (SVA0) |
| Control registers | IICA control register 0 (IICACTL0)<br>IICA status register 0 (IICAS0)<br>IICA flag register 0 (IICAF0)<br>IICA control register 1 (IICACTL1)<br>IICA low-level width setting register (IICWL)<br>IICA high-level width setting register (IICWH)<br>Port input mode register 6 (PIM6)<br>Port output mode register 6 (POM6)<br>Port mode register 6 (PM6)<br>Port register 6 (P6) |

### (1) IICA shift register (IICA)

This register is used to convert 8-bit serial data to 8-bit parallel data and vice versa in synchronization with the serial clock. This register can be used for both transmission and reception.

The actual transmit and receive operations can be controlled by writing and reading operations to this register.

Cancel the wait state and start data transfer by writing data to this register during the wait period.

This register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears IICA to 00H.

**Figure 15-3. Format of IICA Shift Register (IICA)**

Address: FFA5H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IICA | | | | | | | | |

**Cautions 1. Do not write data to the IICA register during data transfer.**

**2. Write or read the IICA register only during the wait period. Accessing the IICA register in a communication state other than during the wait period is prohibited. When the device serves as the master, however, the IICA register can be written only once after the communication trigger bit (STT0) is set to 1.**

**3. When communication is reserved, write data to the IICA register after the interrupt triggered by a stop condition is detected.**

### (2) Slave address register 0 (SVA0)

This register stores seven bits of local addresses {A6, A5, A4, A3, A2, A1, A0} when in slave mode.

This register can be set by an 8-bit memory manipulation instruction.

However, rewriting to this register is prohibited while STD0 = 1 (while the start condition is detected).

Reset signal generation clears SVA0 to 00H.

**Figure 15-4. Format of Slave Address Register 0 (SVA0)**

Address: FFA6H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| SVA0 | | | | | | | | 0[Note] |

**Note** Bit 0 is fixed to 0.

**(3) SO latch**

The SO latch is used to retain the SDAA0 pin's output level.

**(4) Wakeup controller**

This circuit generates an interrupt request (INTIICA0) when the address received by this register matches the address value set to the slave address register 0 (SVA0) or when an extension code is received.

**(5) Serial clock counter**

This counter counts the serial clocks that are output or input during transmit/receive operations and is used to verify that 8-bit data was transmitted or received.

**(6) Interrupt request signal generator**

This circuit controls the generation of interrupt request signals (INTIICA0).

An $I^2C$ interrupt request is generated by the following two triggers.

- Falling edge of eighth or ninth clock of the serial clock (set by WTIM0 bit)
- Interrupt request generated when a stop condition is detected (set by SPIE0 bit)

**Remark** WTIM0 bit: Bit 3 of IICA control register 0 (IICACTL0)
SPIE0 bit: Bit 4 of IICA control register 0 (IICACTL0)

**(7) Serial clock controller**

In master mode, this circuit generates the clock output via the SCLA0 pin from a sampling clock.

**(8) Serial clock wait controller**

This circuit controls the wait timing.

**(9) ACK generator, stop condition detector, start condition detector, and ACK detector**

These circuits generate and detect each status.

**(10) Data hold time correction circuit**

This circuit generates the hold time for data corresponding to the falling edge of the serial clock.

**(11) Start condition generator**

This circuit generates a start condition when the STT0 bit is set to 1.

However, in the communication reservation disabled status (IICRSV bit = 1), when the bus is not released (IICBSY bit = 1), start condition requests are ignored and the STCF bit is set to 1.

**(12) Stop condition generator**

This circuit generates a stop condition when the SPT0 bit is set to 1.

**(13) Bus status detector**

This circuit detects whether or not the bus is released by detecting start conditions and stop conditions.

However, as the bus status cannot be detected immediately following operation, the initial status is set by the STCEN bit.

**Remark** STT0 bit: Bit 1 of IICA control register 0 (IICACTL0)
SPT0 bit: Bit 0 of IICA control register 0 (IICACTL0)
IICRSV bit: Bit 0 of IICA flag register 0 (IICAF0)
IICBSY bit: Bit 6 of IICA flag register 0 (IICAF0)
STCF bit: Bit 7 of IICA flag register 0 (IICAF0)
STCEN bit: Bit 1 of IICA flag register 0 (IICAF0)

## 15.3 Registers Controlling Serial Interface IICA

Serial interface IICA is controlled by the following ten registers.

- IICA control register 0 (IICACTL0)
- IICA status register 0 (IICAS0)
- IICA flag register (IICAF0)
- IICA control register 1 (IICACTL1)
- IICA low-level width setting register (IICWL)
- IICA high-level width setting register (IICWH)
- Port input mode register 6 (PIM6)
- Port output mode register 6 (POM6)
- Port mode register 6 (PM6)
- Port register 6 (P6)

**(1) IICA control register 0 (IICACTL0)**

This register is used to enable/stop $I^2C$ operations, set wait timing, and set other $I^2C$ operations.

This register can be set by a 1-bit or 8-bit memory manipulation instruction. However, set the SPIE0, WTIM0, and ACKE0 bits while IICE0 = 0 or during the wait period. These bits can be set at the same time when the IICE0 bit is set from “0” to “1”.

Reset signal generation clears this register to 00H.

**Figure 15-5. Format of IICA Control Register 0 (IICACTL0) (1/4)**

Address: FFA7H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IICACTL0 | IICE0 | LREL0 | WREL0 | SPIE0 | WTIM0 | ACKE0 | STT0 | SPT0 |

| IICE0 | $I^2C$ operation enable |
|---|---|
| 0 | Stop operation. Reset the IICA status register 0 (IICAS0)[Note 1]. Stop internal operation. |
| 1 | Enable operation. |
| Be sure to set this bit (1) while the SCLA0 and SDLA0 lines are at high level. | |
| Condition for clearing (IICE0 = 0) | Condition for setting (IICE0 = 1) |
| • Cleared by instruction<br>• Reset | • Set by instruction |

| LREL0[Notes 2,3] | Exit from communications |
|---|---|
| 0 | Normal operation |
| 1 | This exits from the current communications and sets standby mode. This setting is automatically cleared to 0 after being executed.<br>Its uses include cases in which a locally irrelevant extension code has been received.<br>The SCLA0 and SDAA0 lines are set to high impedance.<br>The following flags of IICA control register 0 (IICACTL0) and IICA status register 0 (IICAS0) are cleared to 0.<br>• STT0 • SPT0 • MSTS0 • EXC0 • COI0 • TRC0 • ACKD0 • STD0 |
| The standby mode following exit from communications remains in effect until the following communications entry conditions are met.<br>• After a stop condition is detected, restart is in master mode.<br>• An address match or extension code reception occurs after the start condition. | |
| Condition for clearing (LREL0 = 0) | Condition for setting (LREL0 = 1) |
| • Automatically cleared after execution<br>• Reset | • Set by instruction |

| WREL0[Notes 2,3] | Wait cancellation |
|---|---|
| 0 | Do not cancel wait |
| 1 | Cancel wait. This setting is automatically cleared after wait is canceled. |
| When WREL0 is set (wait canceled) during the wait period at the ninth clock pulse in the transmission status (TRC0 = 1), the SDAA0 line goes into the high impedance state (TRC0 = 0). | |
| Condition for clearing (WREL0 = 0) | Condition for setting (WREL0 = 1) |
| • Automatically cleared after execution<br>• Reset | • Set by instruction |

**Notes 1.** The IICAS0 register, the STCF and IICBSY bits of the IICAF0 register, and the CLD0 and DAD0 bits of the IICACTL1 register are reset.

**2.** The signals of these bits are invalid while the IICE0 bit is 0.

**3.** When the LREL0 and WREL0 bits are read, 0 is always read.

**Caution If the operation of $I^2C$ is enabled (IICE0 = 1) when the SCLA0 line is high level, the SDAA0 line is low level, and the digital filter is turned on (DFC0 of the IICACTL1 register = 1), a start condition will be inadvertently detected immediately. In this case, set (1) the LREL0 bit by using a 1-bit memory manipulation instruction immediately after enabling operation of $I^2C$ (IICE0 = 1).**

**Figure 15-5. Format of IICA Control Register 0 (IICACTL0) (2/4)**

| SPIE0[Note 1] | Enable/disable generation of interrupt request when stop condition is detected |
|---|---|
| 0 | Disable |
| 1 | Enable |
| If the WUP bit of the IICA control register 1 (IICACTL1) is 1, no stop condition interrupt will be generated even if SPIE0 = 1. | |
| Condition for clearing (SPIE0 = 0) | Condition for setting (SPIE0 = 1) |
| • Cleared by instruction<br>• Reset | • Set by instruction |

| WTIM0[Note 1] | Control of wait and interrupt request generation |
|---|---|
| 0 | Interrupt request is generated at the eighth clock's falling edge.<br>Master mode: After output of eight clocks, clock output is set to low level and wait is set.<br>Slave mode: After input of eight clocks, the clock is set to low level and wait is set for master device. |
| 1 | Interrupt request is generated at the ninth clock's falling edge.<br>Master mode: After output of nine clocks, clock output is set to low level and wait is set.<br>Slave mode: After input of nine clocks, the clock is set to low level and wait is set for master device. |
| An interrupt is generated at the falling edge of the ninth clock during address transfer independently of the setting of this bit. The setting of this bit is valid when the address transfer is completed. When in master mode, a wait is inserted at the falling edge of the ninth clock during address transfers. For a slave device that has received a local address, a wait is inserted at the falling edge of the ninth clock after an acknowledge ($\overline{ACK}$) is issued. However, when the slave device has received an extension code, a wait is inserted at the falling edge of the eighth clock. | |
| Condition for clearing (WTIM0 = 0) | Condition for setting (WTIM0 = 1) |
| • Cleared by instruction<br>• Reset | • Set by instruction |

| ACKE0 Notes 1, 2 | Acknowledgment control |
|---|---|
| 0 | Disable acknowledgment. |
| 1 | Enable acknowledgment. During the ninth clock period, the SDAA0 line is set to low level. |
| Condition for clearing (ACKE0 = 0) | Condition for setting (ACKE0 = 1) |
| • Cleared by instruction<br>• Reset | • Set by instruction |

**Notes 1.** The signal of this bit is invalid while the IICE0 bit is 0. Set this bit during that period.

**2.** The set value is invalid during address transfer and if the code is not an extension code.
When the device serves as a slave and the addresses match, an acknowledgment is generated regardless of the set value.

**Figure 15-5. Format of IICA Control Register 0 (IICACTL0) (3/4)**

| STT0[Note] | Start condition trigger |
|---|---|
| 0 | Do not generate a start condition. |
| 1 | When bus is released (in standby state, when IICBSY = 0):<br>If this bit is set (1), a start condition is generated (startup as the master).<br>When a third party is communicating:<br>• When communication reservation function is enabled (IICRSV = 0)<br>Functions as the start condition reservation flag. When set to 1, automatically generates a start condition after the bus is released.<br>• When communication reservation function is disabled (IICRSV = 1)<br>Even if this bit is set (1), the STT0 bit is cleared and the STT0 clear flag (STCF) is set (1). No start condition is generated.<br>In the wait state (when master device):<br>Generates a restart condition after releasing the wait. |

Cautions concerning set timing
- For master reception: Cannot be set to 1 during transfer. Can be set to 1 only in the waiting period when ACKE0 has been cleared to 0 and slave has been notified of final reception.
- For master transmission: A start condition cannot be generated normally during the acknowledge period. Set to 1 during the wait period that follows output of the ninth clock.
- Cannot be set to 1 at the same time as stop condition trigger (SPT0).
- Setting the STT0 bit to 1 and then setting it again before it is cleared to 0 is prohibited.

| Condition for clearing (STT0 = 0) | Condition for setting (STT0 = 1) |
|---|---|
| • Cleared by setting the STT0 bit to 1 while communication reservation is prohibited.<br>• Cleared by loss in arbitration<br>• Cleared after start condition is generated by master device<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When IICE0 = 0 (operation stop)<br>• Reset | • Set by instruction |

**Note** The signal of this bit is invalid while IICE0 is 0.

**Remarks 1.** Bit 1 (STT0) becomes 0 when it is read after data setting.

**2.** IICRSV: Bit 0 of IICA flag register 0 (IICAF0)
STCF: Bit 7 of IICA flag register 0 (IICAF0)

**Figure 15-5. Format of IICA Control Register 0 (IICACTL0) (4/4)**

| SPT0 | Stop condition trigger |
|---|---|
| 0 | Stop condition is not generated. |
| 1 | Stop condition is generated (termination of master device's transfer). |

Cautions concerning set timing
- For master reception: Cannot be set to 1 during transfer. Can be set to 1 only in the waiting period when ACKE0 has been cleared to 0 and slave has been notified of final reception.
- For master transmission: A stop condition cannot be generated normally during the acknowledge period. Therefore, set it during the wait period that follows output of the ninth clock.
- Cannot be set to 1 at the same time as start condition trigger (STT0).
- The SPT0 bit can be set to 1 only when in master mode.
- When the WTIM0 bit has been cleared to 0, if the SPT0 bit is set to 1 during the wait period that follows output of eight clocks, note that a stop condition will be generated during the high-level period of the ninth clock. The WTIM0 bit should be changed from 0 to 1 during the wait period following the output of eight clocks, and the SPT0 bit should be set to 1 during the wait period that follows the output of the ninth clock.
- Setting the SPT0 bit to 1 and then setting it again before it is cleared to 0 is prohibited.

| Condition for clearing (SPT0 = 0) | Condition for setting (SPT0 = 1) |
|---|---|
| • Cleared by loss in arbitration<br>• Automatically cleared after stop condition is detected<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When IICE0 = 0 (operation stop)<br>• Reset | • Set by instruction |

**Caution When bit 3 (TRC0) of the IICA status register 0 (IICAS0) is set to 1 (transmission status), bit 5 (WREL0) of the IICACTL0 register is set to 1 during the ninth clock and wait is canceled, after which the TRC0 bit is cleared (reception status) and the SDAA0 line is set to high impedance. Release the wait performed while the TRC0 bit is 1 (transmission status) by writing to the IICA shift register.**

**Remark** Bit 0 (SPT0) becomes 0 when it is read after data setting.

**(2) IICA status register 0 (IICAS0)**

This register indicates the status of $I^2C$.

This register is read by a 1-bit or 8-bit memory manipulation instruction only when STT0 = 1 and during the wait period.

Reset signal generation clears this register to 00H.

**Caution Reading the IICAS0 register while the address match wakeup function is enabled (WUP = 1) in STOP mode is prohibited. When the WUP bit is changed from 1 to 0 (wakeup operation is stopped), regardless of the INTIICA0 interrupt request, the change in status is not reflected until the next start condition or stop condition is detected. To use the wakeup function, therefore, enable (SPIE0 = 1) the interrupt generated by detecting a stop condition and read the IICAS0 register after the interrupt has been detected.**

**Remark** STT0: Bit 1 of IICA control register 0 (IICACTL0)
WUP: Bit 7 of IICA control register 1 (IICACTL1)

**Figure 15-6. Format of IICA Status Register 0 (IICAS0) (1/3)**

Address: FFAAH After reset: 00H R

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IICAS0 | MSTS0 | ALD0 | EXC0 | COI0 | TRC0 | ACKD0 | STD0 | SPD0 |

| MSTS0 | Master status |
|---|---|
| 0 | Slave device status or communication standby status |
| 1 | Master device communication status |

| Condition for clearing (MSTS0 = 0) | Condition for setting (MSTS0 = 1) |
|---|---|
| • When a stop condition is detected<br>• When ALD0 = 1 (arbitration loss)<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • When a start condition is generated |

| ALD0 | Detection of arbitration loss |
|---|---|
| 0 | This status means either that there was no arbitration or that the arbitration result was a “win”. |
| 1 | This status indicates the arbitration result was a “loss”. The MSTS0 bit is cleared. |

| Condition for clearing (ALD0 = 0) | Condition for setting (ALD0 = 1) |
|---|---|
| • Automatically cleared after the IICAS0 register is read[Note]<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • When the arbitration result is a “loss”. |

**Note** This register is also cleared when a 1-bit memory manipulation instruction is executed for bits other than the ALD0 bit of the IICAS0 register. Therefore, when using the ALD0 bit, read the data of this bit before the data of the other bits.

**Remark** LREL0: Bit 6 of IICA control register 0 (IICACTL0)
IICE0: Bit 7 of IICA control register 0 (IICACTL0)

**Figure 15-6. Format of IICA Status Register 0 (IICAS0) (2/3)**

| EXC0 | Detection of extension code reception |
|---|---|
| 0 | Extension code was not received. |
| 1 | Extension code was received. |

| Condition for clearing (EXC0 = 0) | Condition for setting (EXC0 = 1) |
|---|---|
| • When a start condition is detected<br>• When a stop condition is detected<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • When the higher four bits of the received address data is either “0000” or “1111” (set at the rising edge of the eighth clock). |

| COI0 | Detection of matching addresses |
|---|---|
| 0 | Addresses do not match. |
| 1 | Addresses match. |

| Condition for clearing (COI0 = 0) | Condition for setting (COI0 = 1) |
|---|---|
| • When a start condition is detected<br>• When a stop condition is detected<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • When the received address matches the local address (slave address register 0 (SVA0)) (set at the rising edge of the eighth clock). |

| TRC0 | Detection of transmit/receive status |
|---|---|
| 0 | Receive status (other than transmit status). The SDAA0 line is set for high impedance. |
| 1 | Transmit status. The value in the SO0 latch is enabled for output to the SDAA0 line (valid starting at the falling edge of the first byte’s ninth clock). |

| Condition for clearing (TRC0 = 0) | Condition for setting (TRC0 = 1) |
|---|---|
| <Both master and slave><br>• When a stop condition is detected<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Cleared by WREL0 = 1[Note] (wait cancel)<br>• When the ALD0 bit changes from 0 to 1 (arbitration loss)<br>• Reset<br>• When not used for communication (MSTS0, EXC0, COI0 = 0)<br><Master><br>• When “1” is output to the first byte’s LSB (transfer direction specification bit)<br><Slave><br>• When a start condition is detected<br>• When “0” is input to the first byte’s LSB (transfer direction specification bit) | <Master><br>• When a start condition is generated<br>• When 0 (master transmission) is output to the LSB (transfer direction specification bit) of the first byte (during address transfer)<br><Slave><br>• When 1 (slave transmission) is input to the LSB (transfer direction specification bit) of the first byte from the master (during address transfer) |

**Note** When bit 3 (TRC0) of the IICA status register 0 (IICAS0) is set to 1 (transmission status), bit 5 (WREL0) of the IICA control register 0 (IICACTL0) is set to 1 during the ninth clock and wait is canceled, after which the TRC0 bit is cleared (reception status) and the SDAA0 line is set to high impedance. Release the wait performed while TRC0 bit is 1 (transmission status) by writing to the IICA shift register.

**Remark** LREL0: Bit 6 of IICA control register 0 (IICACTL0)
IICE0: Bit 7 of IICA control register 0 (IICACTL0)

**Figure 15-6. Format of IICA Status Register 0 (IICAS0) (3/3)**

| ACKD0 | Detection of acknowledge ($\overline{ACK}$) |
|---|---|
| 0 | Acknowledge was not detected. |
| 1 | Acknowledge was detected. |

| Condition for clearing (ACKD0 = 0) | Condition for setting (ACKD0 = 1) |
|---|---|
| • When a stop condition is detected<br>• At the rising edge of the next byte's first clock<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • After the SDAA0 line is set to low level at the rising edge of SCLA0's ninth clock |

| STD0 | Detection of start condition |
|---|---|
| 0 | Start condition was not detected. |
| 1 | Start condition was detected. This indicates that the address transfer period is in effect. |

| Condition for clearing (STD0 = 0) | Condition for setting (STD0 = 1) |
|---|---|
| • When a stop condition is detected<br>• At the rising edge of the next byte's first clock following address transfer<br>• Cleared by LREL0 = 1 (exit from communications)<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • When a start condition is detected |

| SPD0 | Detection of stop condition |
|---|---|
| 0 | Stop condition was not detected. |
| 1 | Stop condition was detected. The master device's communication is terminated and the bus is released. |

| Condition for clearing (SPD0 = 0) | Condition for setting (SPD0 = 1) |
|---|---|
| • At the rising edge of the address transfer byte's first clock following setting of this bit and detection of a start condition<br>• When the IICE0 bit changes from 1 to 0 (operation stop)<br>• Reset | • When a stop condition is detected |

**Remark** LREL0: Bit 6 of IICA control register 0 (IICACTL0)
IICE0: Bit 7 of IICA control register 0 (IICACTL0)

### (3) IICA flag register 0 (IICAF0)

This register sets the operation mode of $I^2C$ and indicates the status of the $I^2C$ bus.

This register can be set by a 1-bit or 8-bit memory manipulation instruction. However, the STT0 clear flag (STCF) and $I^2C$ bus status flag (IICBSY) are read-only.

The IICRSV bit can be used to enable/disable the communication reservation function.

The STCEN bit can be used to set the initial value of the IICBSY bit.

The IICRSV and STCEN bits can be written only when the operation of $I^2C$ is disabled (bit 7 (IICE0) of the IICA control register 0 (IICACTL0) = 0). When operation is enabled, the IICAF0 register can be read.

Reset signal generation clears this register to 00H.

**Figure 15-7. Format of IICA Flag Register 0 (IICAF0)**

Address: FFA9H After reset: 00H R/W[Note]

| Symbol | <7> | <6> | 5 | 4 | 3 | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IICAF0 | STCF | IICBSY | 0 | 0 | 0 | 0 | STCEN | IICRSV |

| STCF | STT0 clear flag |
|---|---|
| 0 | Generate start condition |
| 1 | Start condition generation unsuccessful: clear STT0 flag |

| Condition for clearing (STCF = 0) | Condition for setting (STCF = 1) |
|---|---|
| • Cleared by STT0 = 1<br>• When IICE0 = 0 (operation stop)<br>• Reset | • Generating start condition unsuccessful and the STT0 bit cleared to 0 when communication reservation is disabled (IICRSV = 1). |

| IICBSY | $I^2C$ bus status flag |
|---|---|
| 0 | Bus release status (communication initial status when STCEN = 1) |
| 1 | Bus communication status (communication initial status when STCEN = 0) |

| Condition for clearing (IICBSY = 0) | Condition for setting (IICBSY = 1) |
|---|---|
| • Detection of stop condition<br>• When IICE0 = 0 (operation stop)<br>• Reset | • Detection of start condition<br>• Setting of the IICE0 bit when STCEN = 0 |

| STCEN | Initial start enable trigger |
|---|---|
| 0 | After operation is enabled (IICE0 = 1), enable generation of a start condition upon detection of a stop condition. |
| 1 | After operation is enabled (IICE0 = 1), enable generation of a start condition without detecting a stop condition. |

| Condition for clearing (STCEN = 0) | Condition for setting (STCEN = 1) |
|---|---|
| • Cleared by instruction<br>• Detection of start condition<br>• Reset | • Set by instruction |

| IICRSV | Communication reservation function disable bit |
|---|---|
| 0 | Enable communication reservation |
| 1 | Disable communication reservation |

| Condition for clearing (IICRSV = 0) | Condition for setting (IICRSV = 1) |
|---|---|
| • Cleared by instruction<br>• Reset | • Set by instruction |

**Note** Bits 6 and 7 are read-only.

**Cautions 1. Write to the STCEN bit only when the operation is stopped (IICE0 = 0).**

**2. As the bus release status (IICBSY = 0) is recognized regardless of the actual bus status when STCEN = 1, when generating the first start condition (STT0 = 1), it is necessary to verify that no third party communications are in progress in order to prevent such communications from being destroyed.**

**3. Write to the IICRSV bit only when the operation is stopped (IICE0 = 0).**

**Remark** STT0: Bit 1 of IICA control register 0 (IICACTL0)
IICE0: Bit 7 of IICA control register 0 (IICACTL0)

### (4) IICA control register 1 (IICACTL1)

This register is used to set the operation mode of $I^2C$ and detect the statuses of the SCLA0 and SDAA0 pins.

This register can be set by a 1-bit or 8-bit memory manipulation instruction. However, the CLD0 and DAD0 bits are read-only.

Set the IICACTL1 register, except the WUP bit, while operation of $I^2C$ is disabled (bit 7 (IICE0) of IICA control register 0 (IICACTL0) is 0).

Reset signal generation clears this register to 00H.

**Figure 15-8. Format of IICA Control Register 1 (IICACTL1) (1/2)**

Address: FFA8H After reset: 00H R/W[Note 1]

| Symbol | 7 | 6 | <5> | <4> | <3> | <2> | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IICACTL1 | WUP | 0 | CLD0 | DAD0 | SMC0 | DFC0 | 0 | 0 |

| WUP | Control of address match wakeup |
|---|---|
| 0 | Stops operation of address match wakeup function in STOP mode. |
| 1 | Enables operation of address match wakeup function in STOP mode. |

To shift to STOP mode when WUP = 1, execute the STOP instruction at least three clocks after setting (1) WUP bit (see **Figure 15-23 Flow When Setting WUP = 1**).
Clear (0) the WUP bit after the address has matched or an extension code has been received. The subsequent communication can be entered by clearing (0) the WUP bit (The wait must be released and transmit data must be written after the WUP bit has been cleared (0).).
The interrupt timing when the address has matched or when an extension code has been received, while WUP = 1, is identical to the interrupt timing when WUP = 0. (A delay of the difference of sampling by the clock will occur.) Furthermore, when WUP = 1, a stop condition interrupt is not generated even if the SPIE0 bit is set to 1.

| Condition for clearing (WUP = 0) | Condition for setting (WUP = 1) |
|---|---|
| • Cleared by instruction (after address match or extension code reception) | • Set by instruction (when MSTS0, EXC0, and COI0 are “0”, and STD0 also “0” (communication not entered))[Note 2] |

**Notes 1.** Bits 4 and 5 are read-only.

**2.** The status of IICAS0 must be checked and WUP must be set during the period shown below.

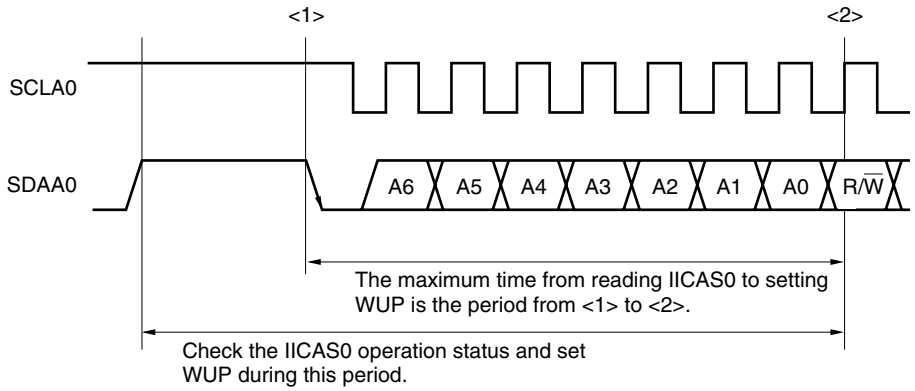


**Figure 15-8. Format of IICA Control Register 1 (IICACTL1) (2/2)**

| CLD0 | Detection of SCLA0 pin level (valid only when IICE0 = 1) |
|---|---|
| 0 | The SCLA0 pin was detected at low level. |
| 1 | The SCLA0 pin was detected at high level. |
| Condition for clearing (CLD0 = 0) | Condition for setting (CLD0 = 1) |
| • When the SCLA0 pin is at low level<br>• When IICE0 = 0 (operation stop)<br>• Reset | • When the SCLA0 pin is at high level |

| DAD0 | Detection of SDAA0 pin level (valid only when IICE0 = 1) |
|---|---|
| 0 | The SDAA0 pin was detected at low level. |
| 1 | The SDAA0 pin was detected at high level. |
| Condition for clearing (DAD0 = 0) | Condition for setting (DAD0 = 1) |
| • When the SDAA0 pin is at low level<br>• When IICE0 = 0 (operation stop)<br>• Reset | • When the SDAA0 pin is at high level |

| SMC0 | Operation mode switching |
|---|---|
| 0 | Operates in standard mode. |
| 1 | Operates in fast mode. |

| DFC0 | Digital filter operation control |
|---|---|
| 0 | Digital filter off. |
| 1 | Digital filter on. |
| Digital filter can be used only in fast mode.<br>In fast mode, the transfer clock does not vary, regardless of the DFC0 bit being set (1) or cleared (0).<br>The digital filter is used for noise elimination in fast mode. | |

**Remark** IICE0: Bit 7 of IICA control register 0 (IICACTL0)

**(5) IICA low-level width setting register (IICWL)**

This register is used to set the low-level width of the SCLA0 pin signal that is output by serial interface IICA being in master mode.

This register can be set by an 8-bit memory manipulation instruction.

Set this register while operation of $I^2C$ is disabled (bit 7 (IICE0) of the IICA control register 0 (IICACTL0) is 0).

Reset signal generation sets this register to FFH.

**Figure 15-9. Format of IICA Low-Level Width Setting Register (IICWL)**

Address: FFADH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IICWL | | | | | | | | |

**(6) IICA high-level width setting register (IICWH)**

This register is used to set the high-level width of the SCLA0 pin signal that is output by serial interface IICA being in master mode.

This register can be set by an 8-bit memory manipulation instruction.

Set this register while operation of $I^2C$ is disabled (bit 7 (IICE0) of the IICA control register 0 (IICACTL0) is 0).

Reset signal generation sets this register to FFH.

**Figure 15-10. Format of IICA High-Level Width Setting Register (IICWH)**

Address: FFAEH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IICWH | | | | | | | | |

**Remark** For how to set the transfer clock by using the IICWL and IICWH registers, see **15.4.2 Setting transfer clock by using IICWL and IICWH registers**.

**(7) Port input mode register 6 (PIM6)**

This register sets the input buffer of P60 and P61 in 1-bit units. When using an input compliant with the SMBus specifications in $I^2C$ communication, set PIM60 and PIM61 to 1.

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

Reset signal generation clears this register to 00H.

**Figure 15-11. Format of Port Input Mode Register 6 (PIM6)**

Address: FF3EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PIM6 | 0 | 0 | 0 | 0 | 0 | 0 | PIM61 | PIM60 |

| PIM6n | P6n pin input buffer selection (n = 0, 1) |
|---|---|
| 0 | Normal input (Schmitt) buffer |
| 1 | SMBus input buffer |

**(8) Port output mode register 6 (POM6)**

This register sets the output mode of P60 to P63 in 1-bit units. During $I^2C$ communication, set POM60 and POM61 to 1.

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

Reset signal generation clears this register to 00H.

**Figure 15-12. Format of Port Output Mode Register 6 (POM6)**

Address: FF2AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| POM6 | 0 | 0 | 0 | 0 | POM63Note | POM62Note | POM61 | POM60 |

| POM6n | P6n pin output mode selection (n = 0 to 3) |
|---|---|
| 0 | Normal output (CMOS output) mode |
| 1 | N-ch open drain output ($V_{DD}$ tolerance) mode |

**Note** 78K0/KC2-L only

**(9) Port mode register 6 (PM6)**

This register sets the input/output of port 6 in 1-bit units.

When using the P60/SCLA0 pin as clock I/O and the P61/SDAA0 pin as serial data I/O, clear PM60 and PM61 to 0, and set the output latches of P60 and P61 to 1.

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

Reset signal generation sets this register to FFH.

**Figure 15-13. Format of Port Mode Register 6 (PM6)**

Address: FF26H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM6 | 1 | 1 | 1 | 1 | PM63 | PM62 | PM61 | PM60 |

| PM6n | P6n pin I/O mode selection (n = 0 to 3) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 6 of the 78K0/KC2-L. For the format of port mode register 6 of other products, refer to **(1) Port mode registers (PMxx)** in **4.3 Registers Controlling Port Function**.

## 15.4 $I^2C$ Bus Mode Functions

### 15.4.1 Pin configuration

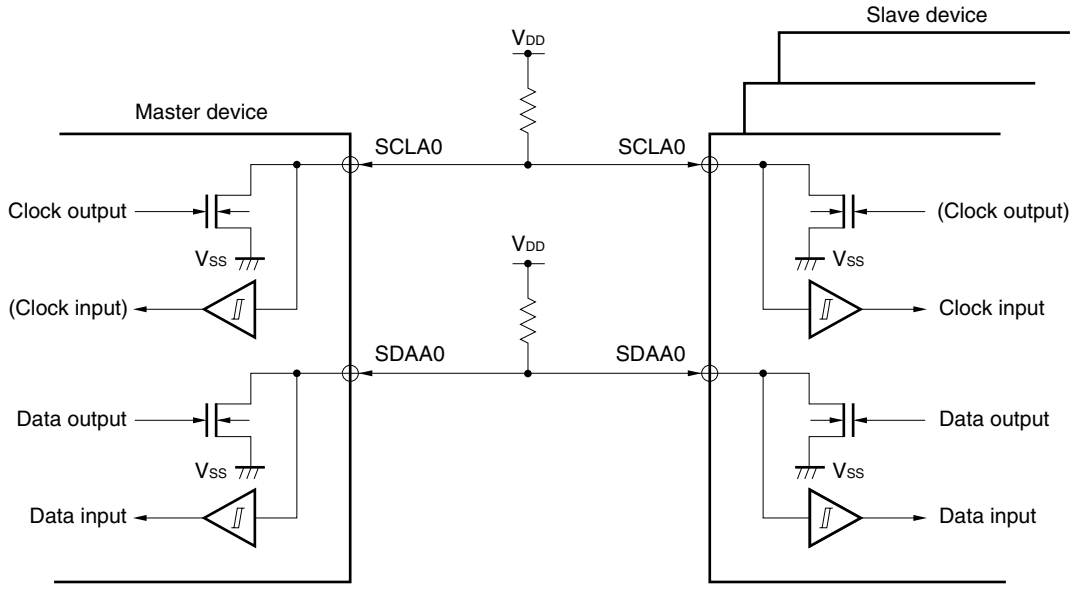
The serial clock pin (SCLA0) and serial data bus pin (SDAA0) are configured as follows.

(1) SCLA0 .... This pin is used for serial clock input and output.
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

(2) SDAA0 .... This pin is used for serial data input and output.
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

Since outputs from the serial clock line and the serial data bus line are N-ch open-drain outputs, an external pull-up resistor is required.

**Figure 15-14. Pin Configuration Diagram**



### 15.4.2 Setting transfer clock by using IICWL and IICWH registers

#### (1) Setting transfer clock on master side

$$\text{Transfer clock} = \frac{f_{PRS}}{IICWL + IICWH + f_{PRS}(t_R + t_F)}$$

At this time, the optimal setting values of the IICWL and IICWH registers are as follows.
(The fractional parts of all setting values are rounded up.)

- When the fast mode

$$IICWL = \frac{0.52}{\text{Transfer clock}} \times f_{PRS}$$
$$IICWH = (\frac{0.48}{\text{Transfer clock}} - t_R - t_F) \times f_{PRS}$$

- When the normal mode

$$IICWL = \frac{0.47}{\text{Transfer clock}} \times f_{PRS}$$
$$IICWH = (\frac{0.53}{\text{Transfer clock}} - t_R - t_F) \times f_{PRS}$$

#### (2) Setting IICWL and IICWH on slave side

(The fractional parts of all setting values are truncated.)

- When the fast mode

$$IICWL = 1.3\ \mu s \times f_{PRS}$$
$$IICWH = (1.2\ \mu s - t_R - t_F) \times f_{PRS}$$

- When the normal mode

$$IICWL = 4.7\ \mu s \times f_{PRS}$$
$$IICWH = (5.3\ \mu s - t_R - t_F) \times f_{PRS}$$

**Caution Note the minimum $f_{PRS}$ operation frequency when setting the transfer clock. The minimum $f_{PRS}$ operation frequency for serial interface IICA is determined according to the mode.**

**Fast mode: $f_{PRS}$ = 3.5 MHz (min.)**

**Normal mode: $f_{PRS}$ = 1 MHz (min.)**

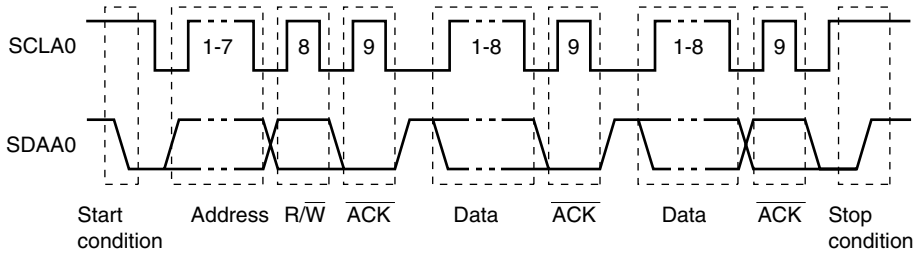
**Remarks 1.** Calculate the rise time ($t_R$) and fall time ($t_F$) of the SDA0 and SCLA0 signals separately, because they differ depending on the pull-up resistance and wire load.

**2.** IICWL: IICA low-level width setting register
IICWH: IICA high-level width setting register
$t_F$: SDAA0 and SCLA0 signal falling times (refer to **CHAPTER 28 ELECTRICAL SPECIFICATIONS**)
$t_R$: SDAA0 and SCLA0 signal rising times (refer to **CHAPTER 28 ELECTRICAL SPECIFICATIONS**)
$f_{PRS}$: Peripheral hardware clock frequency

## 15.5 $I^2C$ Bus Definitions and Control Methods

The following section describes the $I^2C$ bus's serial data communication format and the signals used by the $I^2C$ bus. Figure 15-15 shows the transfer timing for the "start condition", "address", "data", and "stop condition" output via the $I^2C$ bus's serial data bus.

**Figure 15-15. $I^2C$ Bus Serial Data Transfer Timing**



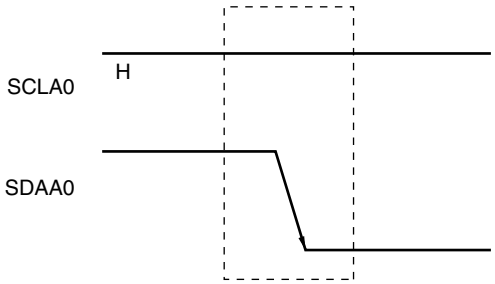

The master device generates the start condition, slave address, and stop condition.
The acknowledge ($\overline{ACK}$) can be generated by either the master or slave device (normally, it is output by the device that receives 8-bit data).

The serial clock (SCLA0) is continuously output by the master device. However, in the slave device, the SCLA0's low level period can be extended and a wait can be inserted.

### 15.5.1 Start conditions

A start condition is met when the SCLA0 pin is at high level and the SDAA0 pin changes from high level to low level. The start conditions for the SCLA0 pin and SDAA0 pin are signals that the master device generates to the slave device when starting a serial transfer. When the device is used as a slave, start conditions can be detected.

**Figure 15-16. Start Conditions**




A start condition is output when bit 1 (STT0) of IICA control register 0 (IICACTL0) is set (1) after a stop condition has been detected (SPD0: Bit 0 of the IICA status register 0 (IICAS0) = 1). When a start condition is detected, bit 1 (STD0) of the IICAS0 register is set (1).

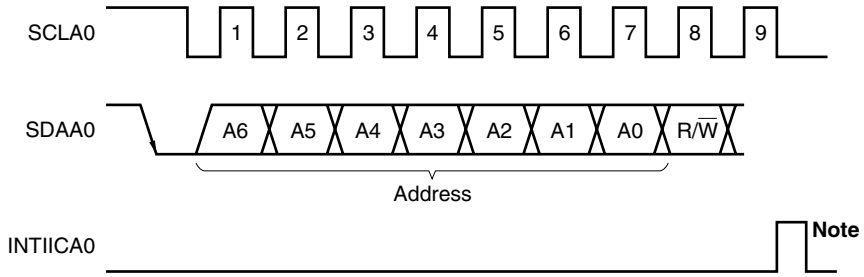
### 15.5.2 Addresses

The address is defined by the 7 bits of data that follow the start condition.

An address is a 7-bit data segment that is output in order to select one of the slave devices that are connected to the master device via the bus lines. Therefore, each slave device connected via the bus lines must have a unique address.

The slave devices include hardware that detects the start condition and checks whether or not the 7-bit address data matches the data values stored in the slave address register 0 (SVA0). If the address data matches the SVA0 register values, the slave device is selected and communicates with the master device until the master device generates a start condition or stop condition.

**Figure 15-17. Address**




**Note** INTIICA0 is not issued if data other than a local address or extension code is received during slave device operation.

Addresses are output when a total of 8 bits consisting of the slave address and the transfer direction described in **15.5.3 Transfer direction specification** are written to the IICA shift register (IICA). The received addresses are written to the IICA register.

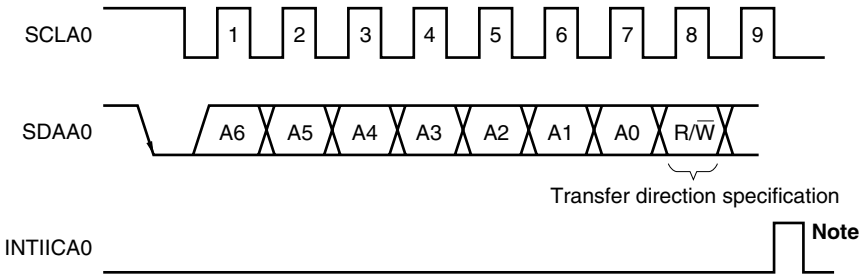
The slave address is assigned to the higher 7 bits of the IICA register.

### 15.5.3 Transfer direction specification

In addition to the 7-bit address data, the master device sends 1 bit that specifies the transfer direction.

When this transfer direction specification bit has a value of “0”, it indicates that the master device is transmitting data to a slave device. When the transfer direction specification bit has a value of “1”, it indicates that the master device is receiving data from a slave device.

**Figure 15-18. Transfer Direction Specification**




**Note** INTIICA0 is not issued if data other than a local address or extension code is received during slave device operation.

### 15.5.4 Acknowledge ($\overline{ACK}$)

$\overline{ACK}$ is used to check the status of serial data at the transmission and reception sides.

The reception side returns $\overline{ACK}$ each time it has received 8-bit data.

The transmission side usually receives $\overline{ACK}$ after transmitting 8-bit data. When $\overline{ACK}$ is returned from the reception side, it is assumed that reception has been correctly performed and processing is continued. Whether $\overline{ACK}$ has been detected can be checked by using bit 2 (ACKD0) of the IICA status register 0 (IICAS0).

When the master receives the last data item, it does not return $\overline{ACK}$ and instead generates a stop condition. If a slave does not return $\overline{ACK}$ after receiving data, the master outputs a stop condition or restart condition and stops transmission. If $\overline{ACK}$ is not returned, the possible causes are as follows.

<1> Reception was not performed normally.
<2> The final data item was received.
<3> The reception side specified by the address does not exist.

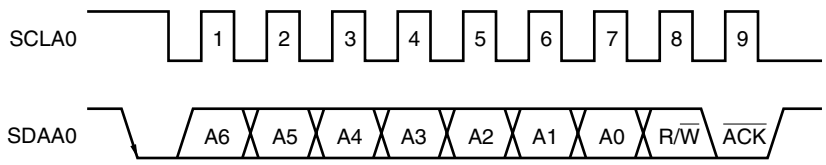
To generate $\overline{ACK}$, the reception side makes the SDAA0 line low at the ninth clock (indicating normal reception).

Automatic generation of $\overline{ACK}$ is enabled by setting bit 2 (ACKE0) of IICA control register 0 (IICACTL0) to 1. Bit 3 (TRC0) of the IICAS0 register is set by the data of the eighth bit that follows 7-bit address information. Usually, set ACKE0 to 1 for reception (TRC0 = 0).

If a slave can receive no more data during reception (TRC0 = 0) or does not require the next data item, then the slave must inform the master, by clearing ACKE0 to 0, that it will not receive any more data.

When the master does not require the next data item during reception (TRC0 = 0), it must clear ACKE0 to 0 so that $\overline{ACK}$ is not generated. In this way, the master informs a slave at the transmission side that it does not require any more data (transmission will be stopped).

**Figure 15-19. $\overline{ACK}$**




When the local address is received, $\overline{ACK}$ is automatically generated, regardless of the value of the ACKE0 bit. When an address other than that of the local address is received, $\overline{ACK}$ is not generated (NACK).

When an extension code is received, $\overline{ACK}$ is generated if the ACKE0 bit is set to 1 in advance.

How $\overline{ACK}$ is generated when data is received differs as follows depending on the setting of the wait timing.

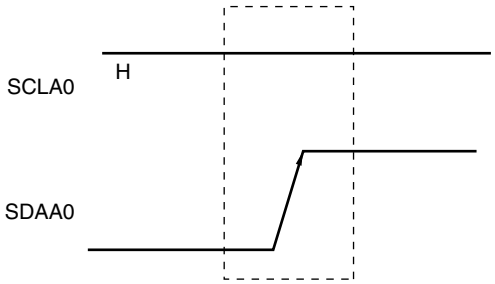
- When 8-clock wait state is selected (bit 3 (WTIM0) of IICACTL0 register = 0):
  By setting the ACKE0 bit to 1 before releasing the wait state, $\overline{ACK}$ is generated at the falling edge of the eighth clock of the SCLA0 pin.
- When 9-clock wait state is selected (bit 3 (WTIM0) of IICACTL0 register = 1):
  $\overline{ACK}$ is generated by setting the ACKE0 bit to 1 in advance.

### 15.5.5 Stop condition

When the SCLA0 pin is at high level, changing the SDAA0 pin from low level to high level generates a stop condition.

A stop condition is a signal that the master device generates to the slave device when serial transfer has been completed. When the device is used as a slave, stop conditions can be detected.

**Figure 15-20. Stop Condition**




A stop condition is generated when bit 0 (SPT0) of the IICA control register 0 (IICACTL0) is set to 1. When the stop condition is detected, bit 0 (SPD0) of the IICA status register 0 (IICAS0) is set to 1 and INTIICA0 is generated when bit 4 (SPIE0) of the IICACTL0 register is set to 1.

### 15.5.6 Wait

The wait is used to notify the communication partner that a device (master or slave) is preparing to transmit or receive data (i.e., is in a wait state).

Setting the SCLA0 pin to low level notifies the communication partner of the wait state. When wait state has been canceled for both the master and slave devices, the next data transfer can begin.

**Figure 15-21. Wait (1/2)**

**(1) When master device has a nine-clock wait and slave device has an eight-clock wait (master transmits, slave receives, and ACKE0 = 1)**

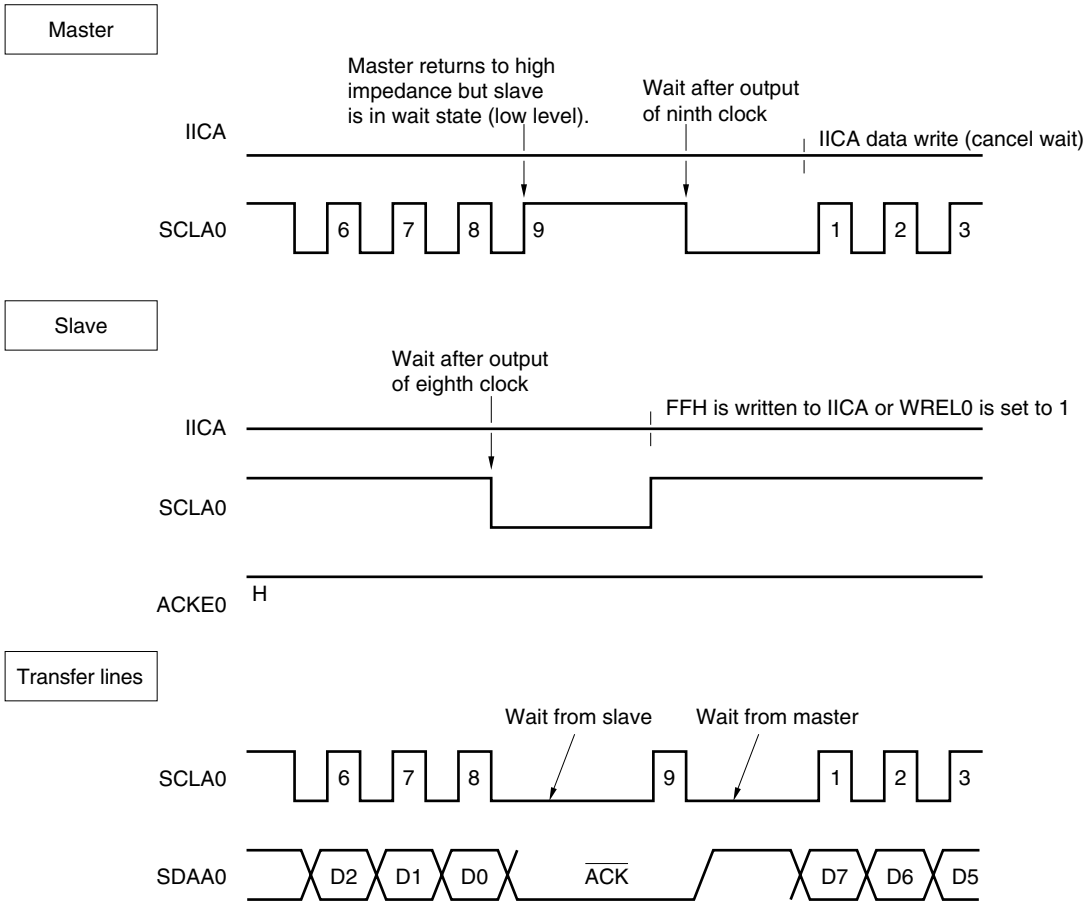
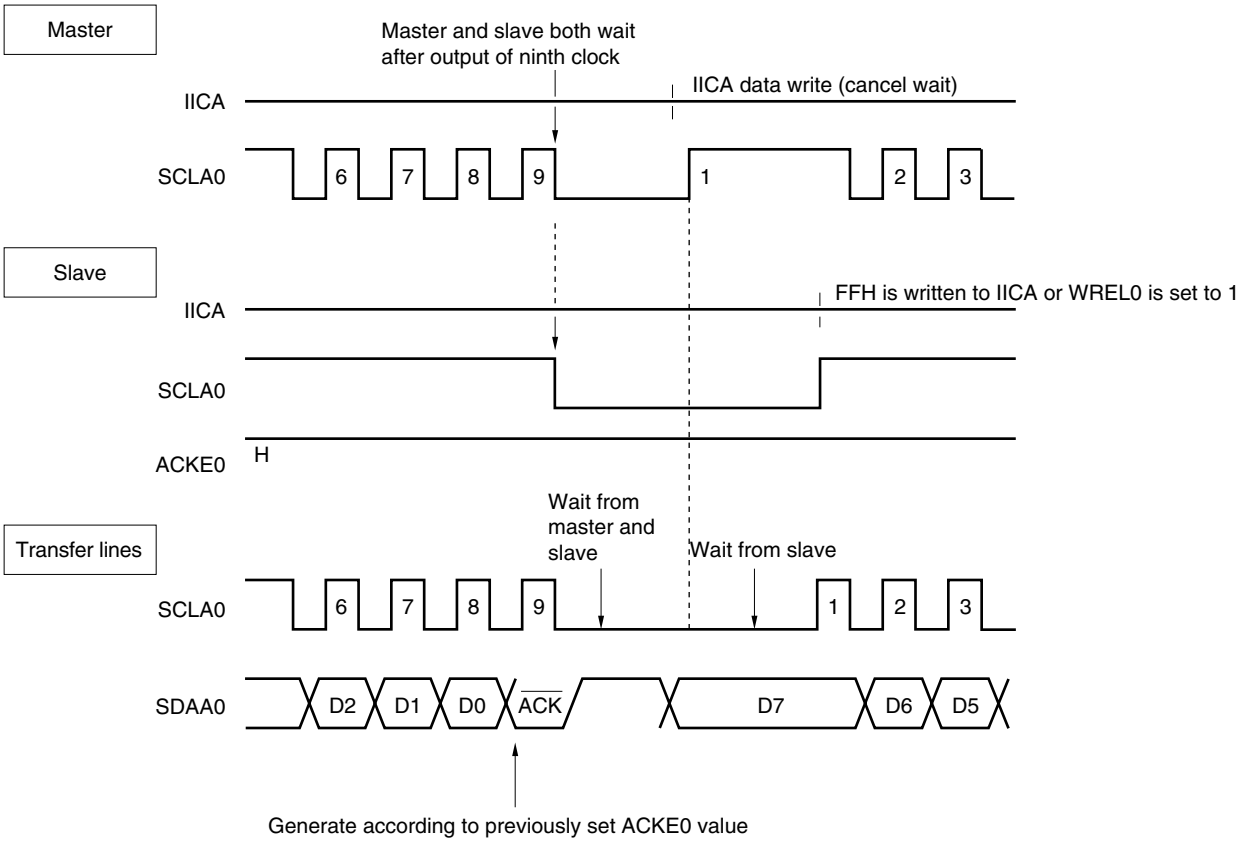


**Figure 15-21. Wait (2/2)**

**(2) When master and slave devices both have a nine-clock wait (master transmits, slave receives, and ACKE0 = 1)**




**Remark** ACKE0: Bit 2 of IICA control register 0 (IICACTL0)
WREL0: Bit 5 of IICA control register 0 (IICACTL0)

A wait may be automatically generated depending on the setting of bit 3 (WTIM0) of the IICA control register 0 (IICACTL0).

Normally, the receiving side cancels the wait state when bit 5 (WREL0) of the IICACTL0 register is set to 1 or when FFH is written to the IICA shift register (IICA), and the transmitting side cancels the wait state when data is written to the IICA register.

The master device can also cancel the wait state via either of the following methods.

- By setting bit 1 (STT0) of IICACTL0 register to 1
- By setting bit 0 (SPT0) of IICACTL0 register to 1

### 15.5.7 Canceling wait

The $I^2C$ usually cancels a wait state by the following processing.

- Writing data to IICA shift register (IICA)
- Setting bit 5 (WREL0) of IICA control register 0 (IICACTL0) (canceling wait)
- Setting bit 1 (STT0) of IICACTL0 register (generating start condition)[Note]
- Setting bit 0 (SPT0) of IICACTL0 register (generating stop condition)[Note]

**Note** Master only

When the above wait canceling processing is executed, the $I^2C$ cancels the wait state and communication is resumed.

To cancel a wait state and transmit data (including addresses), write the data to the IICA register.

To receive data after canceling a wait state, or to complete data transmission, set bit 5 (WREL0) of the IICA control register 0 (IICACTL0) to 1.

To generate a restart condition after canceling a wait state, set bit 1 (STT0) of the IICACTL0 register to 1.

To generate a stop condition after canceling a wait state, set bit 0 (SPT0) of the IICACTL0 register to 1.

Execute the canceling processing only once for one wait state.

If, for example, data is written to the IICA register after canceling a wait state by setting the WREL0 bit to 1, an incorrect value may be output to SDAA0 line because the timing for changing the SDAA0 line conflicts with the timing for writing the IICA register.

In addition to the above, communication is stopped if the IICE0 bit is cleared to 0 when communication has been aborted, so that the wait state can be canceled.

If the $I^2C$ bus has deadlocked due to noise, processing is saved from communication by setting bit 6 (LREL0) of the IICACTL0 register, so that the wait state can be canceled.

**Caution If a processing to cancel a wait state executed when WUP (bit 7 of the IICA control register 1 (IICACTL1)) = 1, the wait state will not be canceled.**

### 15.5.8 Interrupt request (INTIICA0) generation timing and wait control

The setting of bit 3 (WTIM0) of IICA control register 0 (IICACTL0) determines the timing by which INTIICA0 is generated and the corresponding wait control, as shown in Table 15-2.

**Table 15-2. INTIICA0 Generation Timing and Wait Control**

| WTIM0 | During Slave Device Operation | | | During Master Device Operation | | |
|---|---|---|---|---|---|---|
| | Address | Data Reception | Data Transmission | Address | Data Reception | Data Transmission |
| 0 | 9 Notes 1, 2 | 8 Note 2 | 8 Note 2 | 9 | 8 | 8 |
| 1 | 9 Notes 1, 2 | 9 Note 2 | 9 Note 2 | 9 | 9 | 9 |

**Notes 1.** The slave device's INTIICA0 signal and wait period occurs at the falling edge of the ninth clock only when there is a match with the address set to the slave address register 0 (SVA0).
At this point, $\overline{ACK}$ is generated regardless of the value set to bit 2 (ACKE0) of the IICACTL0 register. For a slave device that has received an extension code, INTIICA0 occurs at the falling edge of the eighth clock.
However, if the address does not match after restart, INTIICA0 is generated at the falling edge of the 9th clock, but wait does not occur.

**2.** If the received address does not match the contents of the slave address register 0 (SVA0) and extension code is not received, neither INTIICA0 nor a wait occurs.

**Remark** The numbers in the table indicate the number of the serial clock's clock signals. Interrupt requests and wait control are both synchronized with the falling edge of these clock signals.

**(1) During address transmission/reception**

- Slave device operation: Interrupt and wait timing are determined depending on the conditions described in Notes 1 and 2 above, regardless of the WTIM0 bit.
- Master device operation: Interrupt and wait timing occur at the falling edge of the ninth clock regardless of the WTIM0 bit.

**(2) During data reception**

- Master/slave device operation: Interrupt and wait timing are determined according to the WTIM0 bit.

**(3) During data transmission**

- Master/slave device operation: Interrupt and wait timing are determined according to the WTIM0 bit.

**(4) Wait cancellation method**

The four wait cancellation methods are as follows.

- Writing data to IICA shift register (IICA)
- Setting bit 5 (WREL0) of IICA control register 0 (IICACTL0) (canceling wait)
- Setting bit 1 (STT0) of IICACTL0 register (generating start condition) Note
- Setting bit 0 (SPT0) of IICACTL0 register (generating stop condition) Note

**Note** Master only.

When an 8-clock wait has been selected (WTIM0 = 0), the presence/absence of $\overline{ACK}$ generation must be determined prior to wait cancellation.

**(5) Stop condition detection**

INTIICA0 is generated when a stop condition is detected (only when SPIE0 = 1).

### 15.5.9 Address match detection method

In $I^2C$ bus mode, the master device can select a particular slave device by transmitting the corresponding slave address.

Address match can be detected automatically by hardware. An interrupt request (INTIICA0) occurs when a local address has been set to the slave address register 0 (SVA0) and when the address set to the SVA0 register matches the slave address sent by the master device, or when an extension code has been received.

### 15.5.10 Error detection

In $I^2C$ bus mode, the status of the serial data bus (SDAA0) during data transmission is captured by the IICA shift register (IICA) of the transmitting device, so the IICA data prior to transmission can be compared with the transmitted IICA data to enable detection of transmission errors. A transmission error is judged as having occurred when the compared data values do not match.

### 15.5.11 Extension code

(1) When the higher 4 bits of the receive address are either “0000” or “1111”, the extension code reception flag (EXC0) is set to 1 for extension code reception and an interrupt request (INTIICA0) is issued at the falling edge of the eighth clock. The local address stored in the slave address register 0 (SVA0) is not affected.

(2) If “11110××0” is set to the SVA0 register by a 10-bit address transfer and “11110××0” is transferred from the master device, the results are as follows. Note that INTIICA0 occurs at the falling edge of the eighth clock.

- Higher four bits of data match: EXC0 = 1
- Seven bits of data match: COI0 = 1

**Remark** EXC0: Bit 5 of IICA status register 0 (IICAS0)
COI0: Bit 4 of IICA status register 0 (IICAS0)

(3) Since the processing after the interrupt request occurs differs according to the data that follows the extension code, such processing is performed by software.
If the extension code is received while a slave device is operating, then the slave device is participating in communication even if its address does not match.
For example, after the extension code is received, if you do not wish to operate the target device as a slave device, set bit 6 (LREL0) of the IICA control register 0 (IICACTL0) to 1 to set the standby mode for the next communication operation.

**Table 15-3. Bit Definitions of Main Extension Code**

| Slave Address | R/W Bit | Description |
|---|---|---|
| 0 0 0 0 0 0 0 | 0 | General call address |
| 1 1 1 1 0 x x | 0 | 10-bit slave address specification (for address authentication) |
| 1 1 1 1 0 x x | 1 | 10-bit slave address specification (for read command issuance after address match) |

**Remark** For extension codes other than the above, refer to THE $I^2C$-BUS SPECIFICATION published by NXP.

### 15.5.12 Arbitration

When several master devices simultaneously generate a start condition (when the STT0 bit is set to 1 before the STD0 bit is set to 1), communication among the master devices is performed as the number of clocks are adjusted until the data differs. This kind of operation is called arbitration.

When one of the master devices loses in arbitration, an arbitration loss flag (ALD0) in the IICA status register 0 (IICAS0) is set (1) via the timing by which the arbitration loss occurred, and the SCLA0 and SDAA0 lines are both set to high impedance, which releases the bus.

The arbitration loss is detected based on the timing of the next interrupt request (the eighth or ninth clock, when a stop condition is detected, etc.) and the ALD0 = 1 setting that has been made by software.

For details of interrupt request timing, refer to **15.5.8 Interrupt request (INTIICA0) generation timing and wait control**.

**Remark** STD0: Bit 1 of IICA status register 0 (IICAS0)
STT0: Bit 1 of IICA control register 0 (IICACTL0)

**Figure 15-22. Arbitration Timing Example**

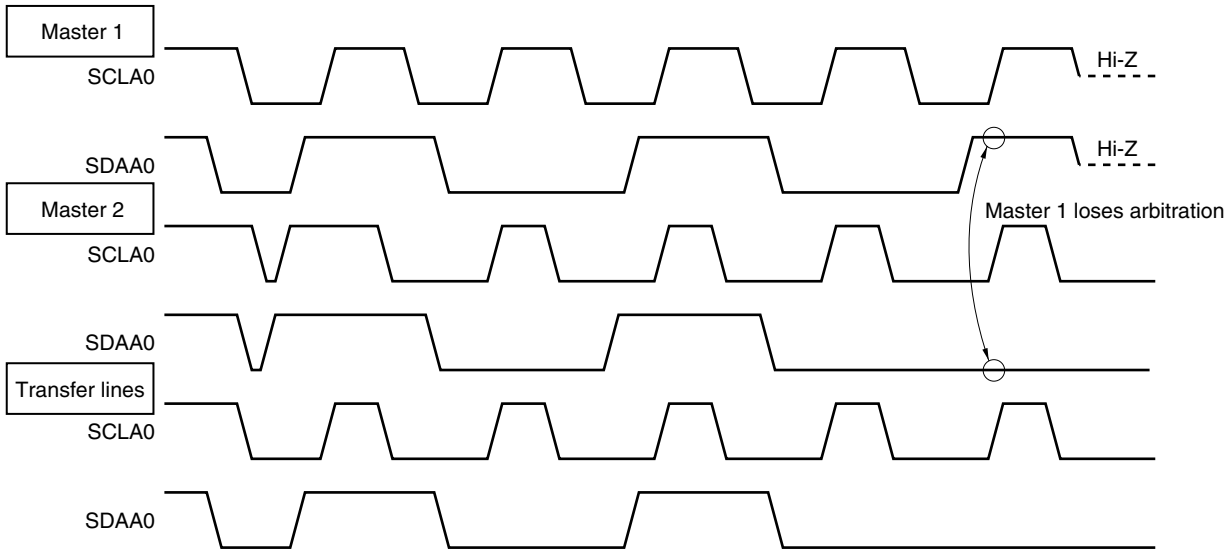


**Table 15-4. Status During Arbitration and Interrupt Request Generation Timing**

| Status During Arbitration | Interrupt Request Generation Timing |
|---|---|
| During address transmission | At falling edge of eighth or ninth clock following byte transfer[Note 1] |
| Read/write data after address transmission | |
| During extension code transmission | |
| Read/write data after extension code transmission | |
| During data transmission | |
| During $\overline{\text{ACK}}$ transfer period after data transmission | |
| When restart condition is detected during data transfer | |
| When stop condition is detected during data transfer | When stop condition is generated (when SPIE0 = 1)[Note 2] |
| When data is at low level while attempting to generate a restart condition | At falling edge of eighth or ninth clock following byte transfer[Note 1] |
| When stop condition is detected while attempting to generate a restart condition | When stop condition is generated (when SPIE0 = 1)[Note 2] |
| When data is at low level while attempting to generate a stop condition | At falling edge of eighth or ninth clock following byte transfer[Note 1] |
| When SCLA0 is at low level while attempting to generate a restart condition | |

**Notes 1.** When the WTIM0 bit (bit 3 of the IICA control register 0 (IICACTL0)) = 1, an interrupt request occurs at the falling edge of the ninth clock. When WTIM0 = 0 and the extension code's slave address is received, an interrupt request occurs at the falling edge of the eighth clock.

**2.** When there is a chance that arbitration will occur, set SPIE0 = 1 for master device operation.

**Remark** SPIE0: Bit 4 of IICA control register 0 (IICACTL0)

### 15.5.13 Wakeup function

The $I^2C$ bus slave function is a function that generates an interrupt request signal (INTIICA0) when a local address and extension code have been received.

This function makes processing more efficient by preventing unnecessary INTIICA0 signal from occurring when addresses do not match.

When a start condition is detected, wakeup standby mode is set. This wakeup standby mode is in effect while addresses are transmitted due to the possibility that an arbitration loss may change the master device (which has generated a start condition) to a slave device.

However, when a stop condition is detected, bit 4 (SPIE0) of the IICA control register 0 (IICACTL0) is set regardless of the wakeup function, and this determines whether interrupt requests are enabled or disabled.

To use the wakeup function in the STOP mode, set WUP to 1. Addresses can be received regardless of the operation clock. An interrupt request signal (INTIICA0) is also generated when a local address and extension code have been received. Operation returns to normal operation by using an instruction to clear (0) the WUP bit after this interrupt has been generated.

Figure 15-23 shows the flow for setting WUP = 1 and Figure 15-24 shows the flow for setting WUP = 0 upon an address match.

**Figure 15-23. Flow When Setting WUP = 1**

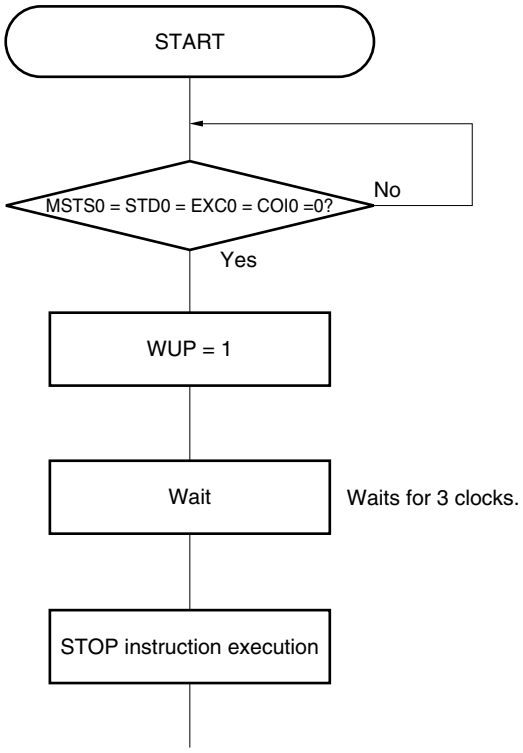
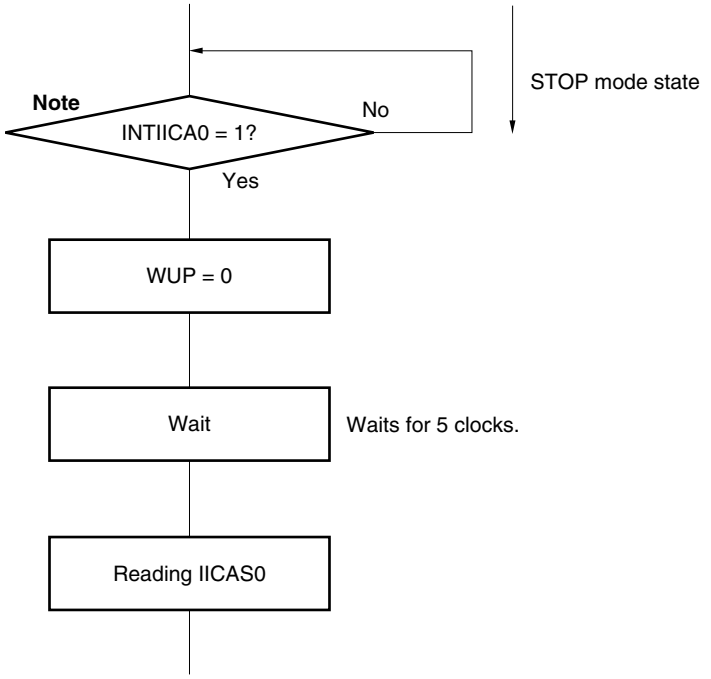


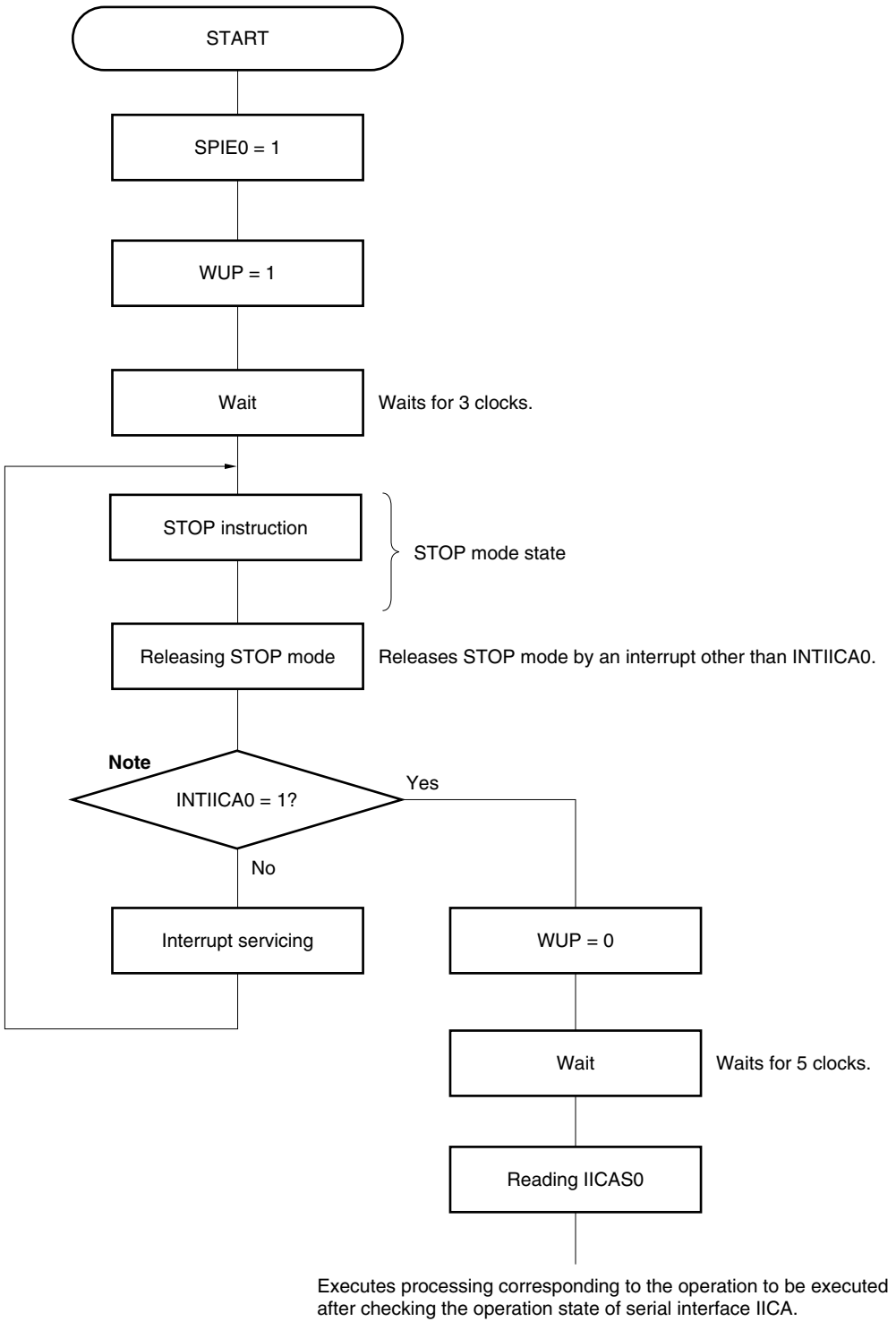
**Figure 15-24. Flow When Setting WUP = 0 upon Address Match (Including Extension Code Reception)**




**Note** Perform the processing after “INTIICA0 = 1?” also when an INTIICA0 vector interrupt occurs.

Use the following flows to perform the processing to release the STOP mode other than by an interrupt request (INTIICA0) generated from serial interface IICA.

**Figure 15-25. When Releasing STOP Mode other than by INTIICA0**




**Note** INTIICA0 also becomes 1 when a STOP condition is issued.

### 15.5.14 Communication reservation

**(1) When communication reservation function is enabled (bit 0 (IICRSV) of IICA flag register 0 (IICAF0) = 0)**

To start master device communications when not currently using a bus, a communication reservation can be made to enable transmission of a start condition when the bus is released. There are two modes under which the bus is not used.

- When arbitration results in neither master nor slave operation
- When an extension code is received and slave operation is disabled ($\overline{ACK}$ is not returned and the bus was released by setting bit 6 (LREL0) of IICA control register 0 (IICACTL0) to 1 and saving communication).

If bit 1 (STT0) of the IICACTL0 register is set to 1 while the bus is not used (after a stop condition is detected), a start condition is automatically generated and wait state is set.
If an address is written to the IICA shift register (IICA) after bit 4 (SPIE0) of the IICACTL0 register was set to 1, and it was detected by generation of an interrupt request signal (INTIICA0) that the bus was released (detection of the stop condition), then the device automatically starts communication as the master. Data written to the IICA register before the stop condition is detected is invalid.
When the STT0 bit has been set to 1, the operation mode (as start condition or as communication reservation) is determined according to the bus status.

- If the bus has been released ..... a start condition is generated
- If the bus has not been released (standby mode)..... communication reservation

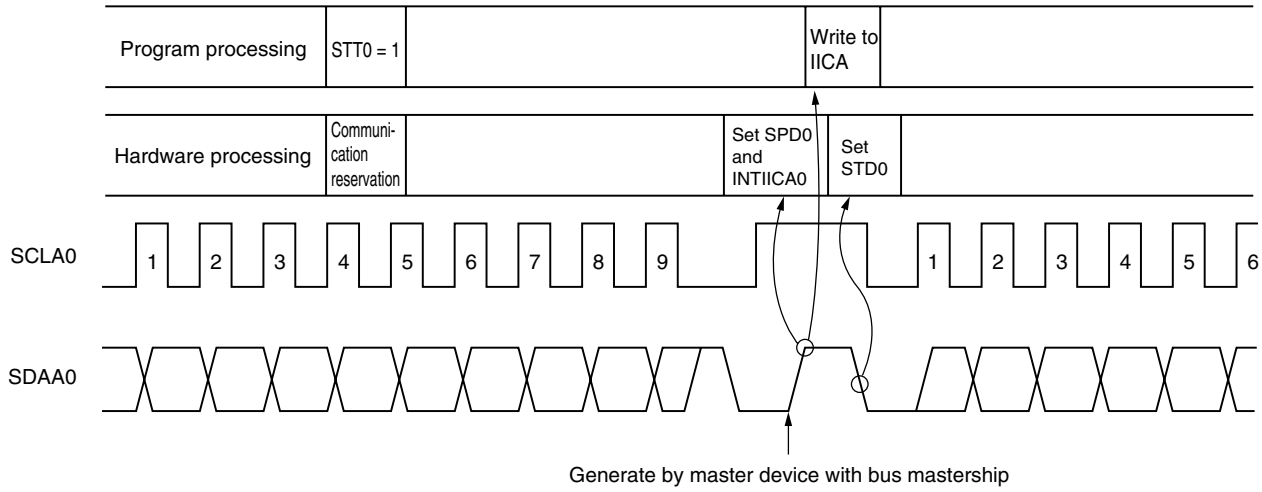
Check whether the communication reservation operates or not by using the MSTS0 bit (bit 7 of the IICA status register 0 (IICAS0)) after the STT0 bit is set to 1 and the wait time elapses.
Use software to secure the wait time calculated by the following expression.

| Wait time from setting STT0 = 1 to checking the MSTS0 flag: |
|---|
| (IICWL setting value + IICWH setting value + 4) + $t_F \times 2 \times f_{PRS}$ [clocks] |

**Remark** IICWL: IICA low-level width setting register
IICWH: IICA high-level width setting register
$t_F$: SDAA0 and SCLA0 signal falling times
(refer to **CHAPTER 28 ELECTRICAL SPECIFICATIONS**)
$f_{PRS}$: Peripheral hardware clock frequency

Figure 15-26 shows the communication reservation timing.

**Figure 15-26. Communication Reservation Timing**




**Remark** IICA: IICA shift register
STT0: Bit 1 of IICA control register 0 (IICACTL0)
STD0: Bit 1 of IICA status register 0 (IICAS0)
SPD0: Bit 0 of IICA status register 0 (IICAS0)

Communication reservations are accepted via the timing shown in Figure 15-27. After bit 1 (STD0) of the IICA status register 0 (IICAS0) is set to 1, a communication reservation can be made by setting bit 1 (STT0) of the IICA control register 0 (IICACTL0) to 1 before a stop condition is detected.

**Figure 15-27. Timing for Accepting Communication Reservations**

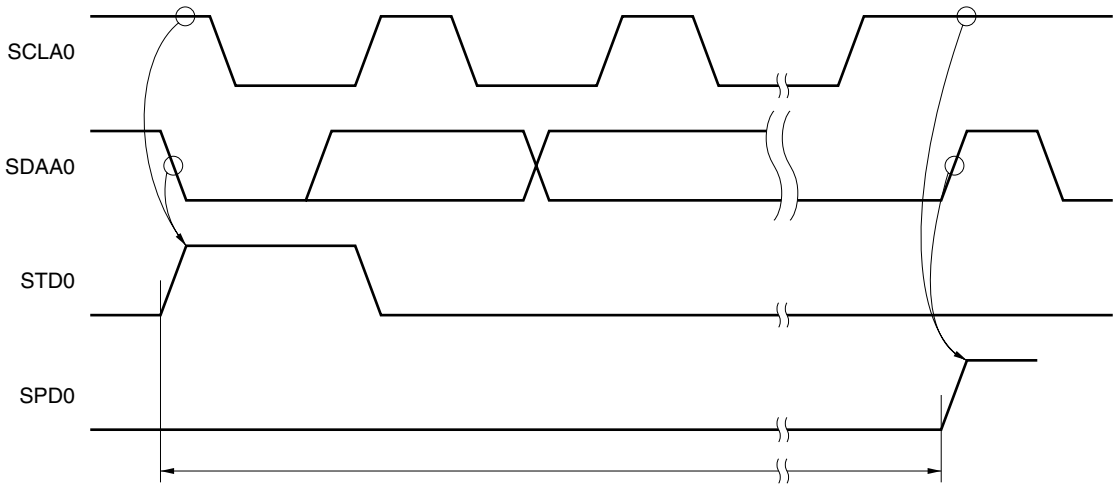
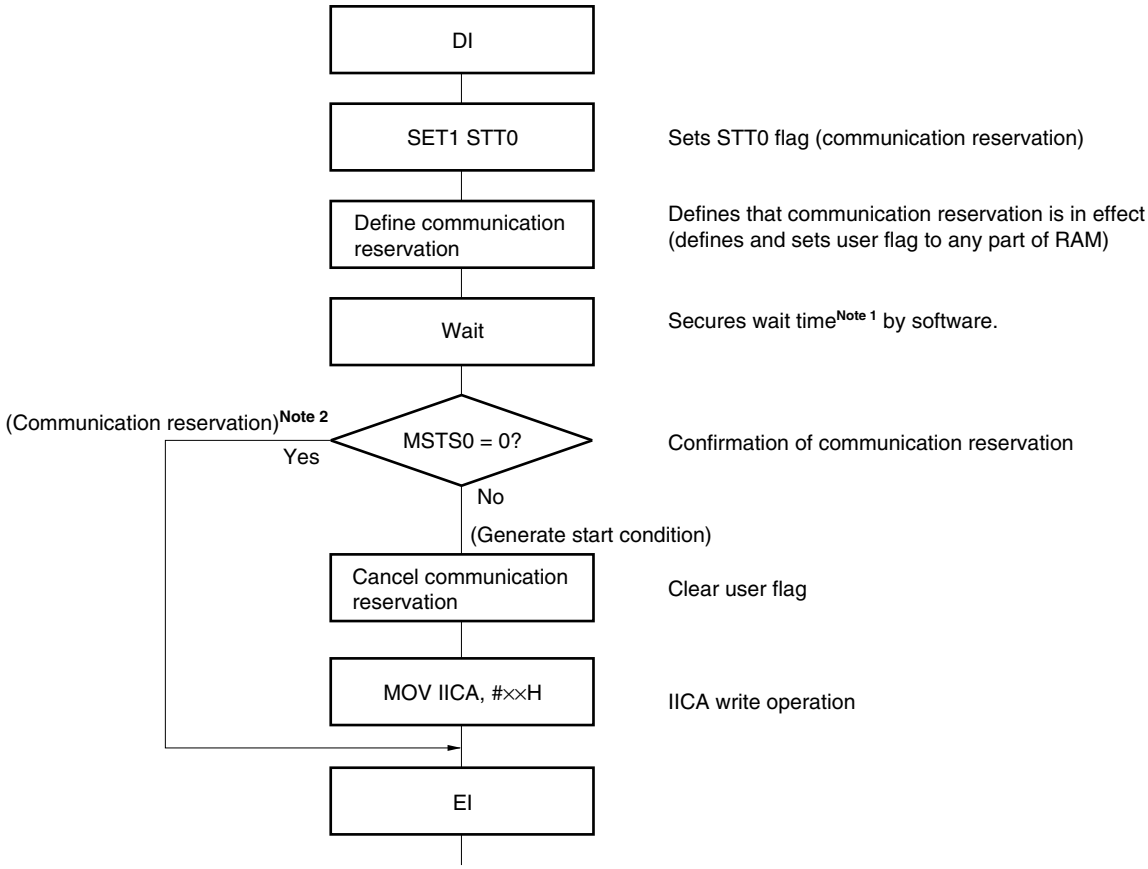



Figure 15-28 shows the communication reservation protocol.

**Figure 15-28. Communication Reservation Protocol**




**Notes 1.** The wait time is calculated as follows.
(IICWL setting value + IICWH setting value + 4) + $t_F \times 2 \times f_{PRS}$ [clocks]

**2.** The communication reservation operation executes a write to the IICA shift register (IICA) when a stop condition interrupt request occurs.

**Remark** STT0: Bit 1 of IICA control register 0 (IICACTL0)
MSTS0: Bit 7 of IICA status register 0 (IICAS0)
IICA: IICA shift register
IICWL: IICA low-level width setting register
IICWH: IICA high-level width setting register
$t_F$: SDAA0 and SCLA0 signal falling times
(refer to **CHAPTER 28 ELECTRICAL SPECIFICATIONS**)
$f_{PRS}$: Peripheral hardware clock frequency

**(2) When communication reservation function is disabled (bit 0 (IICRSV) of IICA flag register 0 (IICAF0) = 1)**

When bit 1 (STT0) of the IICA control register 0 (IICACTL0) is set to 1 when the bus is not used in a communication during bus communication, this request is rejected and a start condition is not generated. The following two statuses are included in the status where bus is not used.

- When arbitration results in neither master nor slave operation
- When an extension code is received and slave operation is disabled ($\overline{\text{ACK}}$ is not returned and the bus was released by setting bit 6 (LREL0) of the IICACTL0 register to 1 and saving communication)

To confirm whether the start condition was generated or request was rejected, check the STCF flag (bit 7 of IICF0 register). It takes up to 5 clocks until the STCF flag is set to 1 after setting STT0 = 1. Therefore, secure the time by software.

### 15.5.15 Cautions

(1) When STCEN (bit 1 of IICA flag register 0 (IICAF0)) = 0

Immediately after $I^2C$ operation is enabled (IICE0 = 1), the bus communication status (the IICBSY flag (bit 6 of the IICAF0 register) = 1) is recognized regardless of the actual bus status. When changing from a mode in which no stop condition has been detected to a master device communication mode, first generate a stop condition to release the bus, then perform master device communication.

When using multiple masters, it is not possible to perform master device communication when the bus has not been released (when a stop condition has not been detected).

Use the following sequence for generating a stop condition.

<1> Set IICA control register 1 (IICACTL1).
<2> Set bit 7 (IICE0) of IICA control register 0 (IICACTL0) to 1.
<3> Set bit 0 (SPT0) of IICACTL0 to 1.

(2) When STCEN = 1

Immediately after $I^2C$ operation is enabled (IICE0 = 1), the bus released status (IICBSY = 0) is recognized regardless of the actual bus status. To generate the first start condition (STT0 (bit 1 of the IICA control register 0 (IICACTL0)) = 1), it is necessary to confirm that the bus has been released, so as to not disturb other communications.

(3) If other $I^2C$ communications are already in progress

If $I^2C$ operation is enabled and the device participates in communication already in progress when the SDAA0 pin is low and the SCLA0 pin is high, the macro of $I^2C$ recognizes that the SDAA0 pin has gone low (detects a start condition). If the value on the bus at this time can be recognized as an extension code, $\overline{ACK}$ is returned, but this interferes with other $I^2C$ communications. To avoid this, start $I^2C$ in the following sequence.

<1> Clear bit 4 (SPIE0) of the IICACTL0 register to 0 to disable generation of an interrupt request signal (INTIICA0) when the stop condition is detected.
<2> Set bit 7 (IICE0) of the IICACTL0 register to 1 to enable the operation of $I^2C$.
<3> Wait for detection of the start condition.
<4> Set bit 6 (LREL0) of the IICACTL0 register to 1 before $\overline{ACK}$ is returned (4 to 80 clocks after setting the IICE0 bit to 1), to forcibly disable detection.

(4) Setting the STT0 and SPT0 bits (bits 1 and 0 of the IICACTL0 register) again after they are set and before they are cleared to 0 is prohibited.

(5) When transmission is reserved, set SPIE0 (bit 4 of the IICACTL0 register) to 1 so that an interrupt request is generated when the stop condition is detected. Transfer is started when communication data is written to the IICA shift register (IICA) after the interrupt request is generated. Unless the interrupt is generated when the stop condition is detected, the device stops in the wait state because the interrupt request is not generated when communication is started. However, it is not necessary to set the SPIE0 bit to 1 when the MSTS0 bit (bit 7 of the IICA status register (IICAS0)) is detected by software.

### 15.5.16 Communication operations

The following shows three operation procedures with the flowchart.

#### (1) Master operation in single master system

The flowchart when using the 78K0/Kx2-L microcontrollers as the master in a single master system is shown below. This flowchart is broadly divided into the initial settings and communication processing. Execute the initial settings at startup. If communication with the slave is required, prepare the communication and then execute communication processing.

#### (2) Master operation in multimaster system

In the $I^2C$ bus multimaster system, whether the bus is released or used cannot be judged by the $I^2C$ bus specifications when the bus takes part in a communication. Here, when data and clock are at a high level for a certain period (1 frame), the 78K0/Kx2-L microcontrollers take part in a communication with bus released state.

This flowchart is broadly divided into the initial settings, communication waiting, and communication processing. The processing when the 78K0/Kx2-L microcontrollers lose in arbitration and is specified as the slave is omitted here, and only the processing as the master is shown. Execute the initial settings at startup to take part in a communication. Then, wait for the communication request as the master or wait for the specification as the slave. The actual communication is performed in the communication processing, and it supports the transmission/reception with the slave and the arbitration with other masters.

#### (3) Slave operation

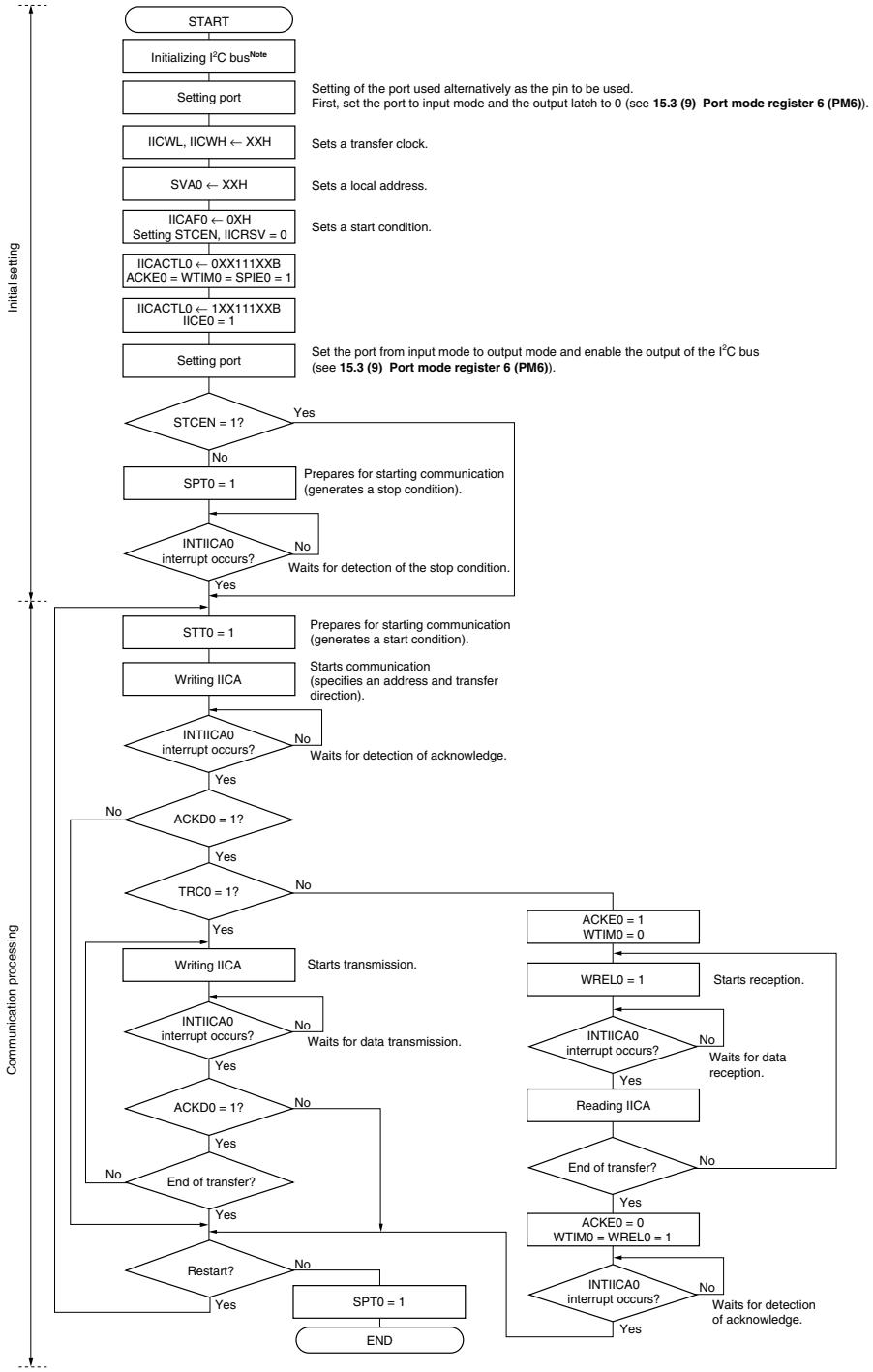
An example of when the 78K0/Kx2-L microcontrollers are used as the $I^2C$ bus slave is shown below.

When used as the slave, operation is started by an interrupt. Execute the initial settings at startup, then wait for the INTIICA0 interrupt occurrence (communication waiting). When an INTIICA0 interrupt occurs, the communication status is judged and its result is passed as a flag over to the main processing.

By checking the flags, necessary communication processing is performed.

### (1) Master operation in single-master system

**Figure 15-29. Master Operation in Single-Master System**

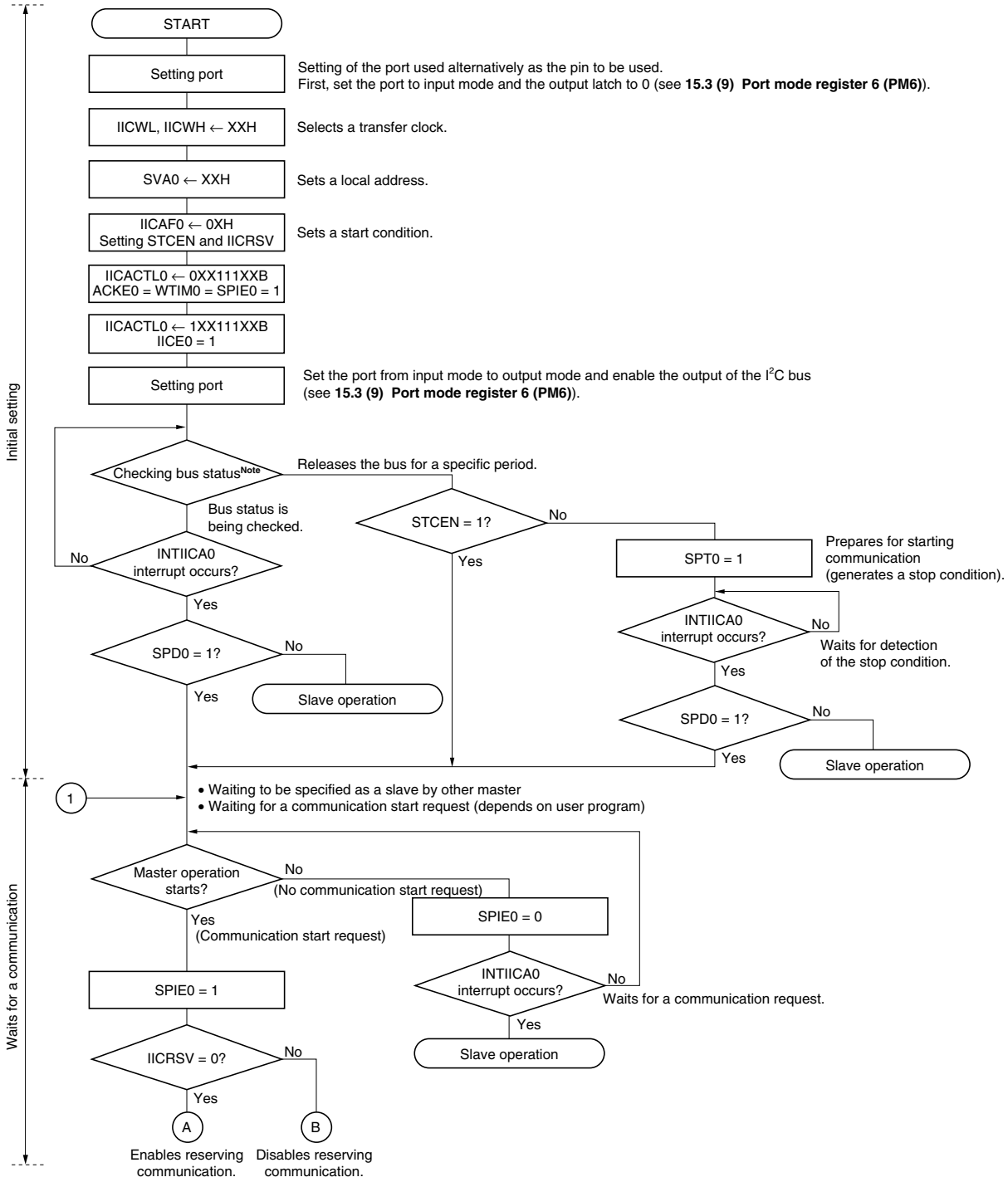



**Note** Release (SCLA0 and SDAA0 pins = high level) the $I^2C$ bus in conformance with the specifications of the product that is communicating. If EEPROM is outputting a low level to the SDAA0 pin, for example, set the SCLA0 pin in the output port mode, and output a clock pulse from the output port until the SDAA0 pin is constantly at high level.

**Remark** Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

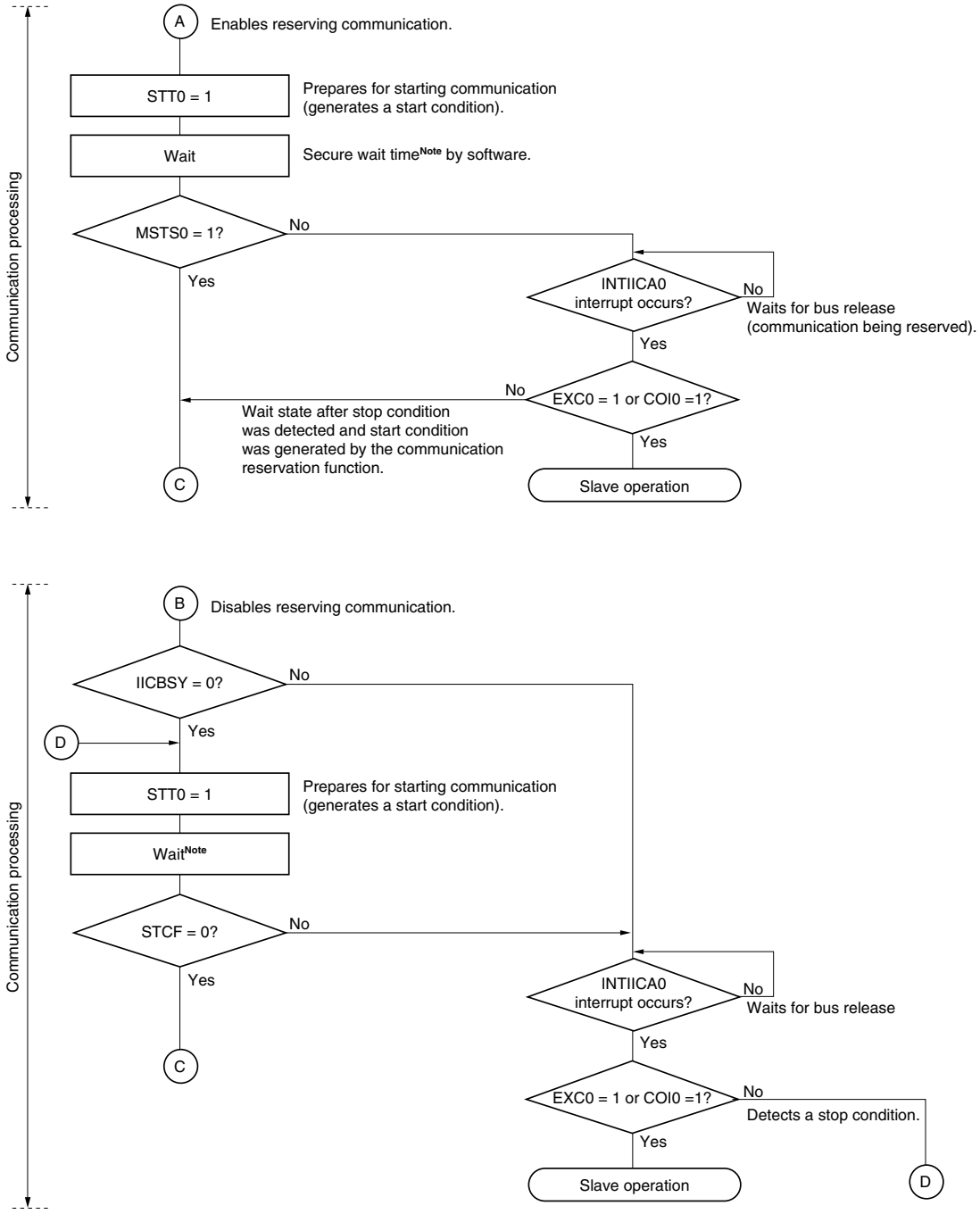
### (2) Master operation in multi-master system

**Figure 15-30. Master Operation in Multi-Master System (1/3)**




**Note** Confirm that the bus is released (CLD0 bit = 1, DAD0 bit = 1) for a specific period (for example, for a period of one frame). If the SDAA0 pin is constantly at low level, decide whether to release the $I^2C$ bus (SCLA0 and SDAA0 pins = high level) in conformance with the specifications of the product that is communicating.

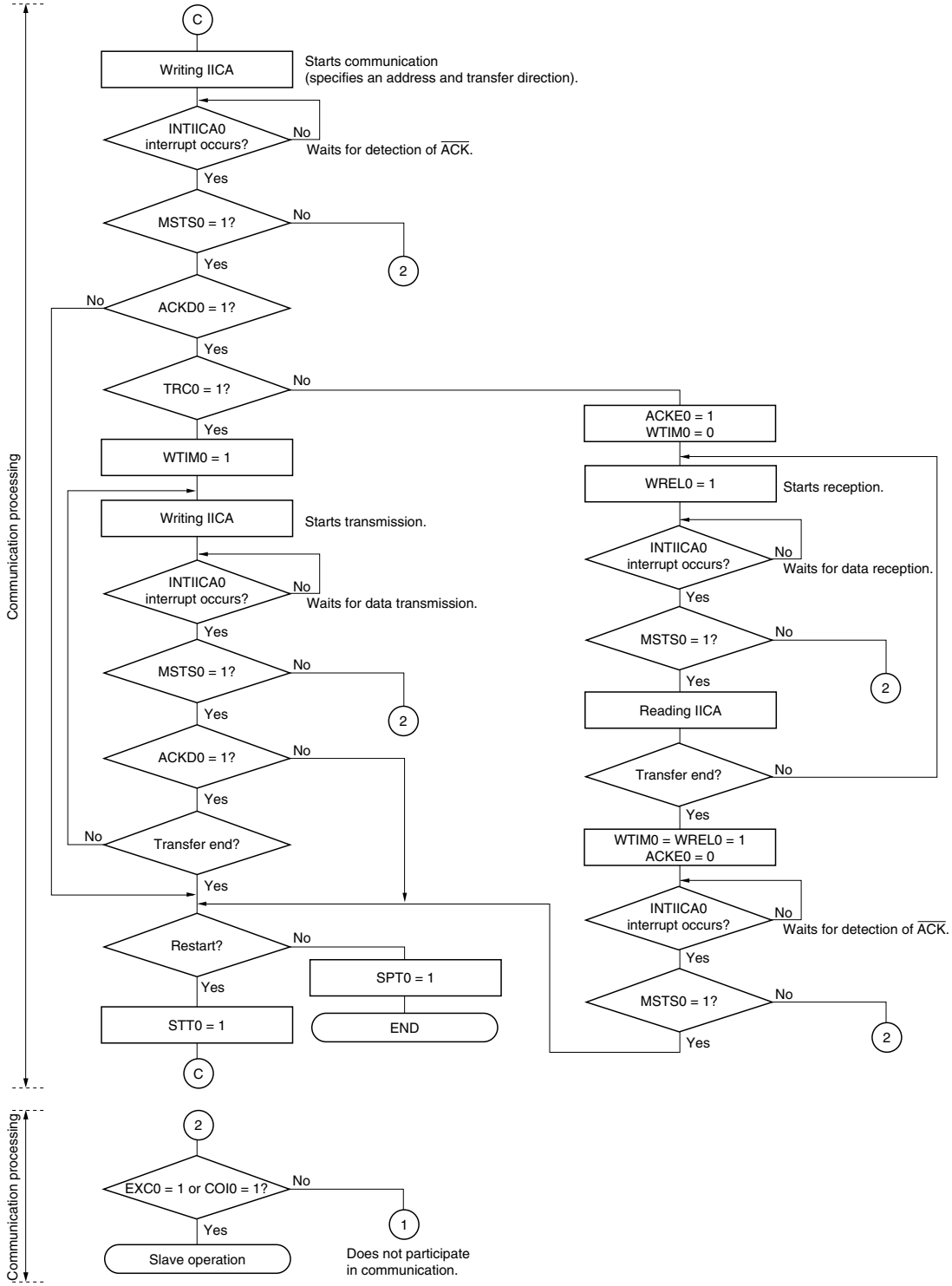
**Figure 15-30. Master Operation in Multi-Master System (2/3)**




**Note** The wait time is calculated as follows.
(IICWL setting value + IICWH setting value + 4) + $t_F \times 2 \times f_{PRS}$ (clocks)

**Remark** IICWL: IICA low-level width setting register
IICWH: IICA high-level width setting register
$t_F$: SDAA0 and SCLA0 signal falling times (refer to **CHAPTER 28 ELECTRICAL SPECIFICATIONS**)
$f_{PRS}$: Peripheral hardware clock frequency

**Figure 15-30. Master Operation in Multi-Master System (3/3)**




**Remarks 1.** Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

**2.** To use the device as a master in a multi-master system, read the MSTS0 bit each time interrupt INTIICA0 has occurred to check the arbitration result.

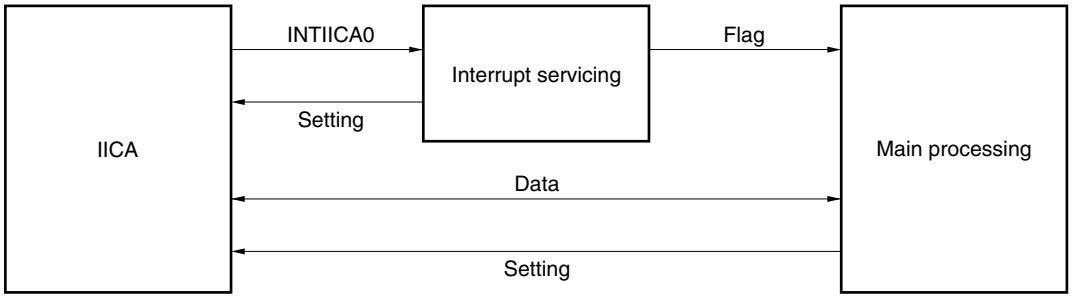
**3.** To use the device as a slave in a multi-master system, check the status by using the IICAS0 and IICAF0 registers each time interrupt INTIICA0 has occurred, and determine the processing to be performed next.

### (3) Slave operation

The processing procedure of the slave operation is as follows.

Basically, the slave operation is event-driven. Therefore, processing by the INTIICA0 interrupt (processing that must substantially change the operation status such as detection of a stop condition during communication) is necessary.

In the following explanation, it is assumed that the extension code is not supported for data communication. It is also assumed that the INTIICA0 interrupt servicing only performs status transition processing, and that actual data communication is performed by the main processing.




Therefore, data communication processing is performed by preparing the following three flags and passing them to the main processing instead of INTIICA0.

**<1> Communication mode flag**

This flag indicates the following two communication statuses.

- Clear mode: Status in which data communication is not performed
- Communication mode: Status in which data communication is performed (from valid address detection to stop condition detection, no detection of $\overline{\text{ACK}}$ from master, address mismatch)

**<2> Ready flag**

This flag indicates that data communication is enabled. Its function is the same as the INTIICA0 interrupt for ordinary data communication. This flag is set by interrupt servicing and cleared by the main processing. Clear this flag by interrupt servicing when communication is started. However, the ready flag is not set by interrupt servicing when the first data is transmitted. Therefore, the first data is transmitted without the flag being cleared (an address match is interpreted as a request for the next data).

**<3> Communication direction flag**

This flag indicates the direction of communication. Its value is the same as the TRC0 bit.

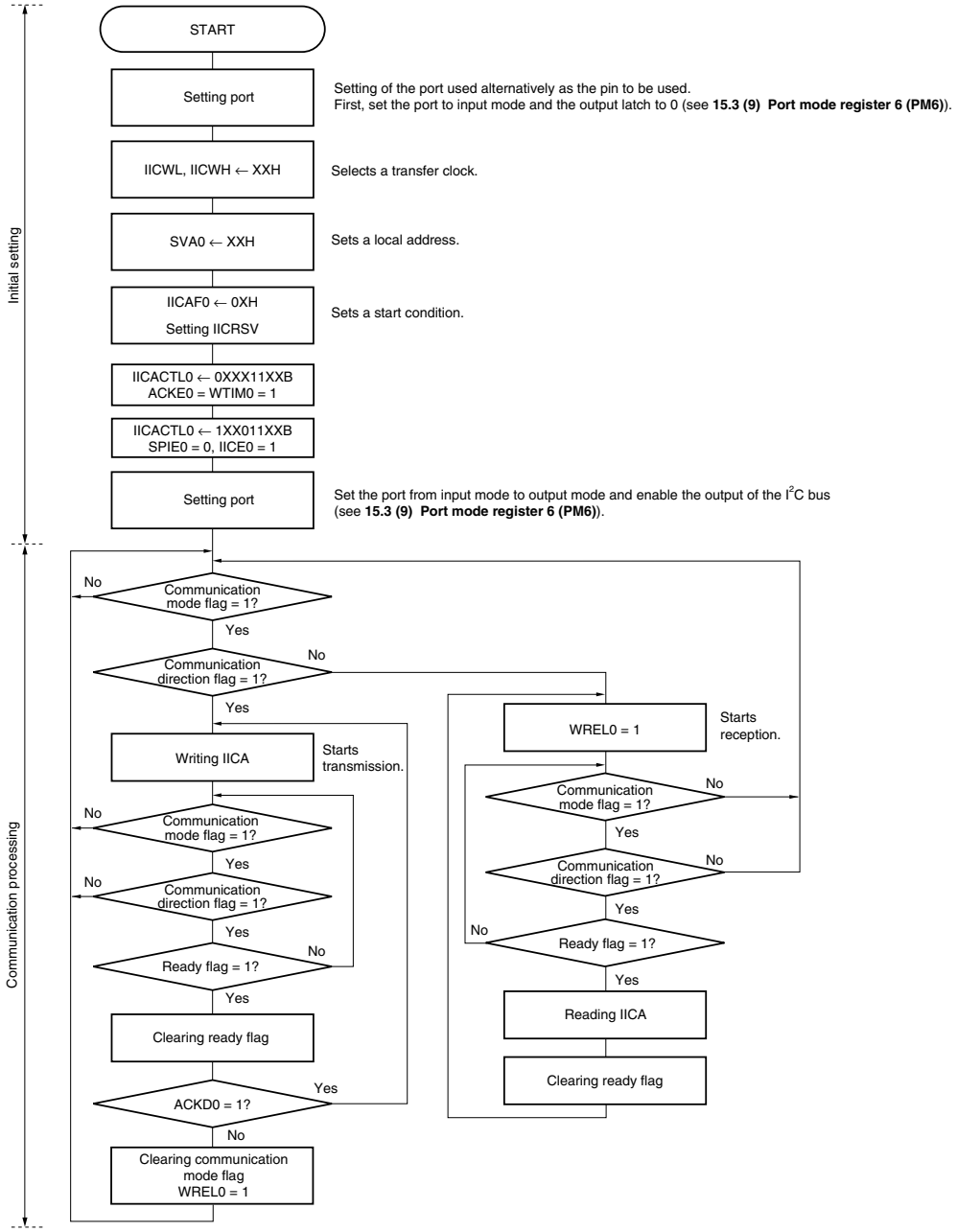
The main processing of the slave operation is explained next.

Start serial interface IICA and wait until communication is enabled. When communication is enabled, execute communication by using the communication mode flag and ready flag (processing of the stop condition and start condition is performed by an interrupt. Here, check the status by using the flags).

The transmission operation is repeated until the master no longer returns $\overline{ACK}$. If $\overline{ACK}$ is not returned from the master, communication is completed.

For reception, the necessary amount of data is received. When communication is completed, $\overline{ACK}$ is not returned as the next data. After that, the master generates a stop condition or restart condition. Exit from the communication status occurs in this way.

**Figure 15-31. Slave Operation Flowchart (1)**



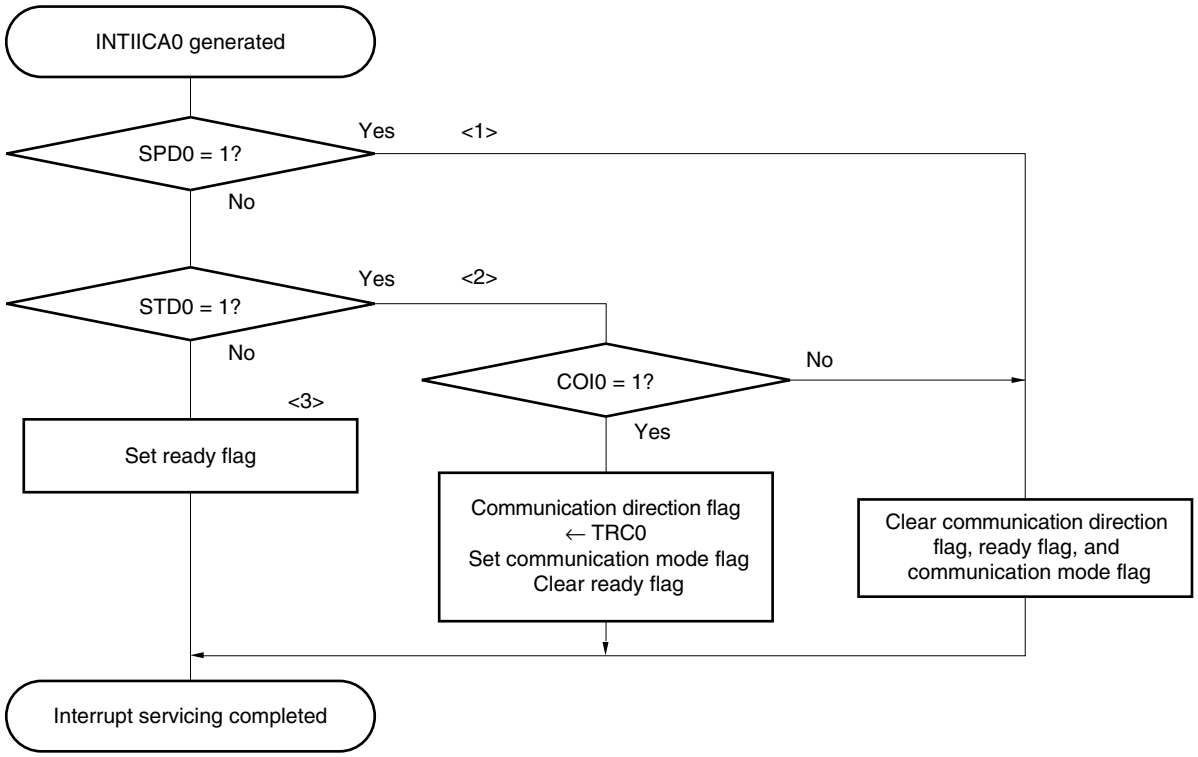

**Remark** Conform to the specifications of the product that is in communication, regarding the transmission and reception formats.

An example of the processing procedure of the slave with the INTIICA0 interrupt is explained below (processing is performed assuming that no extension code is used). The INTIICA0 interrupt checks the status, and the following operations are performed.

<1> Communication is stopped if the stop condition is issued.
<2> If the start condition is issued, the address is checked and communication is completed if the address does not match. If the address matches, the communication mode is set, wait is cancelled, and processing returns from the interrupt (the ready flag is cleared).
<3> For data transmit/receive, only the ready flag is set. Processing returns from the interrupt with the $I^2C$ bus remaining in the wait state.

**Remark** <1> to <3> above correspond to <1> to <3> in Figure 15-32 Slave Operation Flowchart (2).

**Figure 15-32. Slave Operation Flowchart (2)**



### 15.5.17 Timing of I²C interrupt request (INTIICA0) occurrence

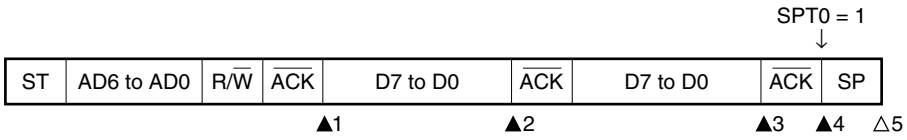
The timing of transmitting or receiving data and generation of interrupt request signal INTIICA0, and the value of the IICAS0 register when the INTIICA0 signal is generated are shown below.

**Remark** ST: Start condition
AD6 to AD0: Address
R/$\overline{W}$: Transfer direction specification
$\overline{ACK}$: Acknowledge
D7 to D0: Data
SP: Stop condition

### (1) Master device operation

#### (a) Start ~ Address ~ Data ~ Data ~ Stop (transmission/reception)

##### (i) When WTIM0 = 0

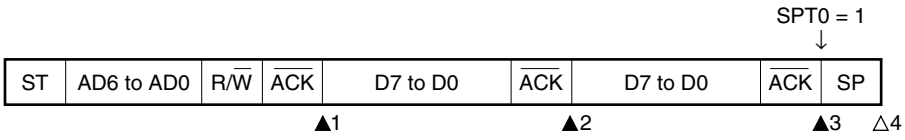



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×000B
▲3: IICAS0 = 1000×000B (Sets WTIM0 to 1)[Note]
▲4: IICAS0 = 1000××00B (Sets SPT0 to 1)[Note]
△5: IICAS0 = 00000001B

**Note** To generate a stop condition, set WTIM0 to 1 and change the timing for generating the INTIICA0 interrupt request signal.

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

##### (ii) When WTIM0 = 1

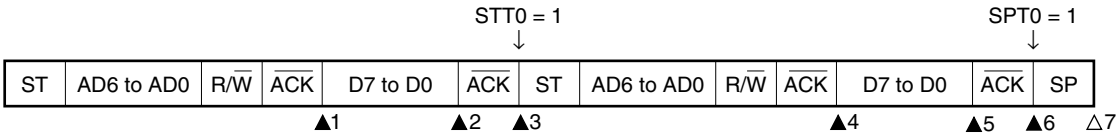



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×100B
▲3: IICAS0 = 1000××00B (Sets SPT0 to 1)
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop (restart)**

**(i) When WTIM0 = 0**

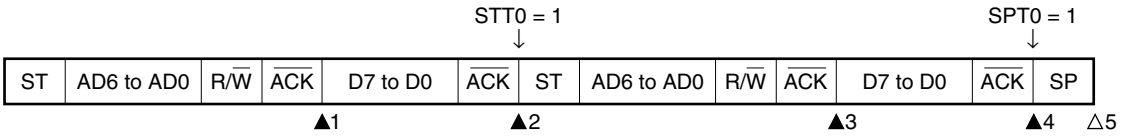



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×000B (Sets WTIM0 to 1)[Note 1]
▲3: IICAS0 = 1000××00B (Clears WTIM0 to 0[Note 2], sets STT0 to 1)
▲4: IICAS0 = 1000×110B
▲5: IICAS0 = 1000×000B (Sets WTIM0 to 1)[Note 3]
▲6: IICAS0 = 1000××00B (Sets SPT0 to 1)
△7: IICAS0 = 00000001B

**Notes 1.** To generate a start condition, set WTIM0 to 1 and change the timing for generating the INTIICA0 interrupt request signal.
**2.** Clear WTIM0 to 0 to restore the original setting.
**3.** To generate a stop condition, set WTIM0 to 1 and change the timing for generating the INTIICA0 interrupt request signal.

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**

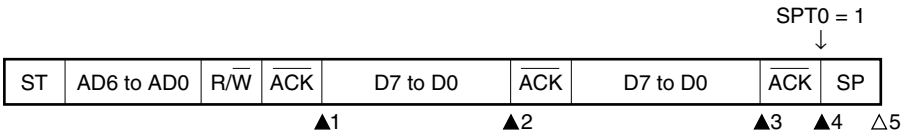



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000××00B (Sets STT0 to 1)
▲3: IICAS0 = 1000×110B
▲4: IICAS0 = 1000××00B (Sets SPT0 to 1)
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(c) Start ~ Code ~ Data ~ Data ~ Stop (extension code transmission)**

**(i) When WTIM0 = 0**

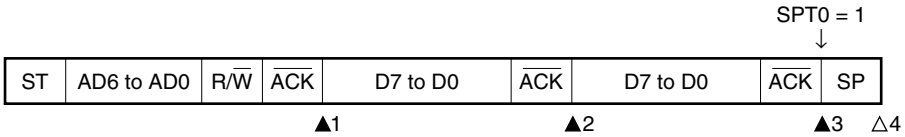



▲1: IICAS0 = 1010×110B
▲2: IICAS0 = 1010×000B
▲3: IICAS0 = 1010×000B (Sets WTIM0 to 1)[Note]
▲4: IICAS0 = 1010××00B (Sets SPT0 to 1)
△5: IICAS0 = 00000001B

**Note** To generate a stop condition, set WTIM0 to 1 and change the timing for generating the INTIICA0 interrupt request signal.

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**



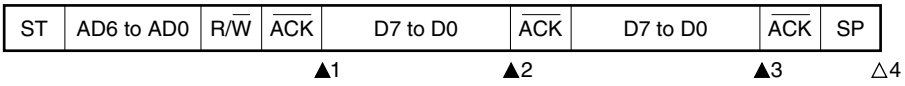

▲1: IICAS0 = 1010×110B
▲2: IICAS0 = 1010×100B
▲3: IICAS0 = 1010××00B (Sets SPT0 to 1)
△4: IICAS0 = 00001001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(2) Slave device operation (slave address data reception)**

**(a) Start ~ Address ~ Data ~ Data ~ Stop**

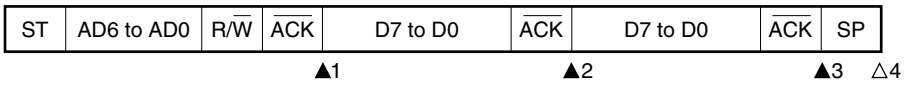
**(i) When WTIM0 = 0**




▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001×000B
▲3: IICAS0 = 0001×000B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**

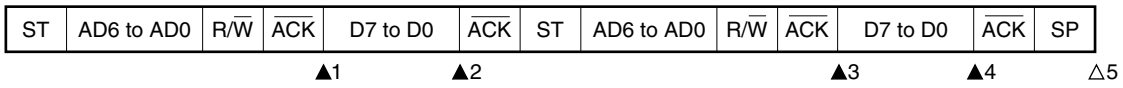



▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001×100B
▲3: IICAS0 = 0001××00B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop**

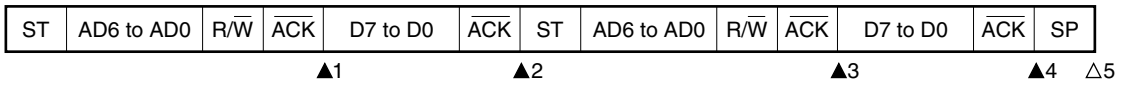
**(i) When WTIM0 = 0 (after restart, matches with SVA0)**




▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001×000B
▲3: IICAS0 = 0001×110B
▲4: IICAS0 = 0001×000B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1 (after restart, matches with SVA0)**

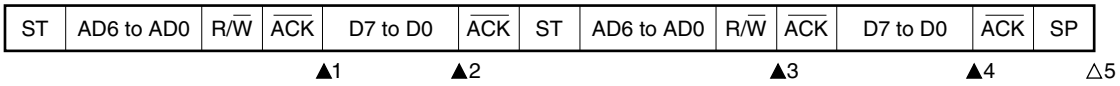



▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001××00B
▲3: IICAS0 = 0001×110B
▲4: IICAS0 = 0001××00B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(c) Start ~ Address ~ Data ~ Start ~ Code ~ Data ~ Stop**

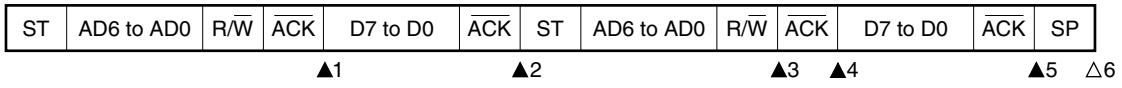
**(i) When WTIM0 = 0 (after restart, does not match address (= extension code))**




▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001×000B
▲3: IICAS0 = 0010×010B
▲4: IICAS0 = 0010×000B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1 (after restart, does not match address (= extension code))**

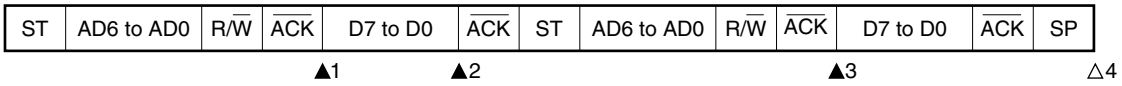



▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001××00B
▲3: IICAS0 = 0010×010B
▲4: IICAS0 = 0010×110B
▲5: IICAS0 = 0010××00B
△6: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(d) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop**

**(i) When WTIM0 = 0 (after restart, does not match address (= not extension code))**

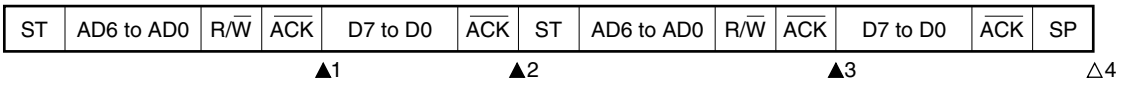



▲1 ▲2 ▲3 △4

▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001×000B
▲3: IICAS0 = 00000110B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1 (after restart, does not match address (= not extension code))**




▲1 ▲2 ▲3 △4

▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001××00B
▲3: IICAS0 = 00000110B
△4: IICAS0 = 00000001B

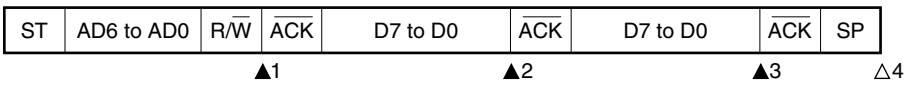
**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(3) Slave device operation (when receiving extension code)**

The device is always participating in communication when it receives an extension code.

**(a) Start ~ Code ~ Data ~ Data ~ Stop**

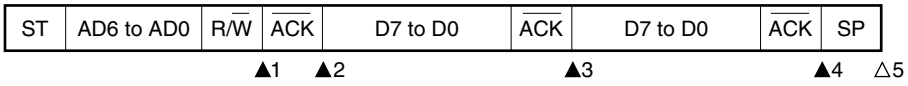
**(i) When WTIM0 = 0**




▲1: IICAS0 = 0010×010B
▲2: IICAS0 = 0010×000B
▲3: IICAS0 = 0010×000B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**

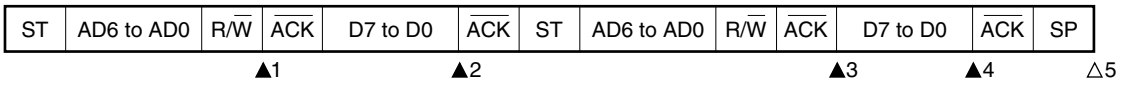



▲1: IICAS0 = 0010×010B
▲2: IICAS0 = 0010×110B
▲3: IICAS0 = 0010×100B
▲4: IICAS0 = 0010××00B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(b) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop**

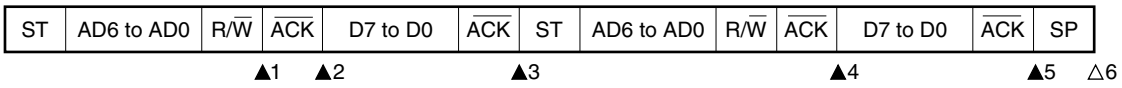
**(i) When WTIM0 = 0 (after restart, matches SVA0)**




▲1: IICAS0 = 0010×010B
▲2: IICAS0 = 0010×000B
▲3: IICAS0 = 0001×110B
▲4: IICAS0 = 0001×000B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1 (after restart, matches SVA0)**

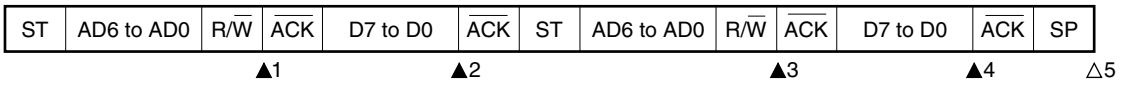



▲1: IICAS0 = 0010×010B
▲2: IICAS0 = 0010×110B
▲3: IICAS0 = 0010××00B
▲4: IICAS0 = 0001×110B
▲5: IICAS0 = 0001××00B
△6: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(c) Start ~ Code ~ Data ~ Start ~ Code ~ Data ~ Stop**

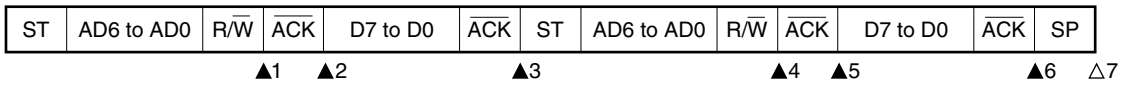
**(i) When WTIM0 = 0 (after restart, extension code reception)**




▲1: IICAS0 = 0010×010B
▲2: IICAS0 = 0010×000B
▲3: IICAS0 = 0010×010B
▲4: IICAS0 = 0010×000B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1 (after restart, extension code reception)**




▲1: IICAS0 = 0010×010B
▲2: IICAS0 = 0010×110B
▲3: IICAS0 = 0010××00B
▲4: IICAS0 = 0010×010B
▲5: IICAS0 = 0010×110B
▲6: IICAS0 = 0010××00B
△7: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(d) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop**

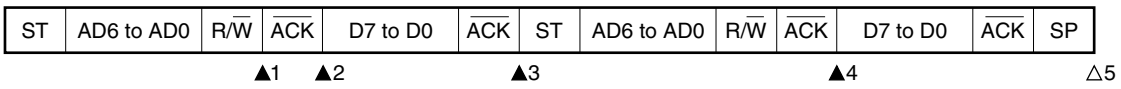
**(i) When WTIM0 = 0 (after restart, does not match address (= not extension code))**




▲1: IICAS0 = 00100010B
▲2: IICAS0 = 00100000B
▲3: IICAS0 = 00000110B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1 (after restart, does not match address (= not extension code))**

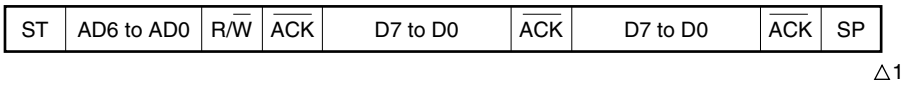



▲1: IICAS0 = 00100010B
▲2: IICAS0 = 00100110B
▲3: IICAS0 = 00100×00B
▲4: IICAS0 = 00000110B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

#### (4) Operation without communication

##### (a) Start ~ Code ~ Data ~ Data ~ Stop




△1: IICAS0 = 00000001B

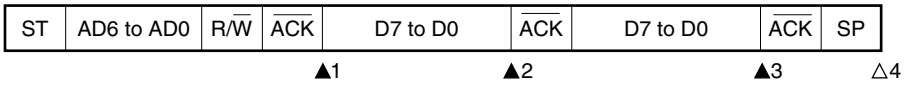
**Remark** △: Generated only when SPIE0 = 1

#### (5) Arbitration loss operation (operation as slave after arbitration loss)

When the device is used as a master in a multi-master system, read the MSTS0 bit each time interrupt request signal INTIICA0 has occurred to check the arbitration result.

##### (a) When arbitration loss occurs during transmission of slave address data

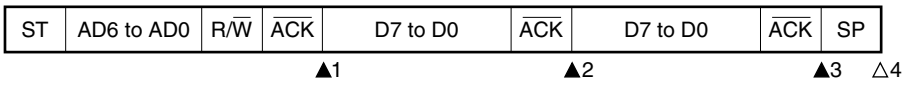
###### (i) When WTIM0 = 0




▲1: IICAS0 = 0101×110B
▲2: IICAS0 = 0001×000B
▲3: IICAS0 = 0001×000B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**



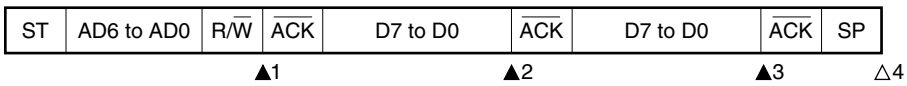

▲1 ▲2 ▲3 △4

▲1: IICAS0 = 0101×110B
▲2: IICAS0 = 0001×100B
▲3: IICAS0 = 0001××00B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(b) When arbitration loss occurs during transmission of extension code**

**(i) When WTIM0 = 0**

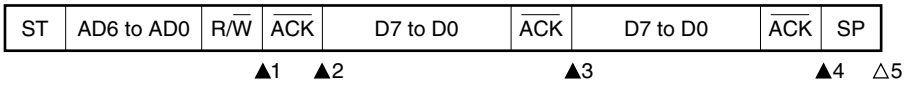



▲1 ▲2 ▲3 △4

▲1: IICAS0 = 0110×010B
▲2: IICAS0 = 0010×000B
▲3: IICAS0 = 0010×000B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**



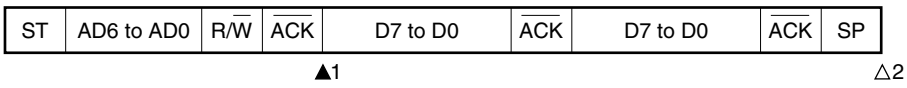

▲1: IICAS0 = 0110×010B
▲2: IICAS0 = 0010×110B
▲3: IICAS0 = 0010×100B
▲4: IICAS0 = 0010××00B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(6) Operation when arbitration loss occurs (no communication after arbitration loss)**

When the device is used as a master in a multi-master system, read the MSTS0 bit each time interrupt request signal INTIICA0 has occurred to check the arbitration result.

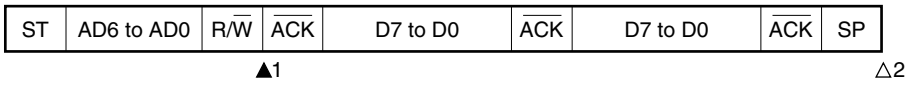
**(a) When arbitration loss occurs during transmission of slave address data (when WTIM0 = 1)**




▲1: IICAS0 = 01000110B
△2: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1

**(b) When arbitration loss occurs during transmission of extension code**

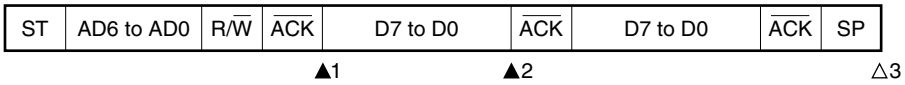



▲1: IICAS0 = 0110×010B
Sets LREL0 = 1 by software
△2: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(c) When arbitration loss occurs during transmission of data**

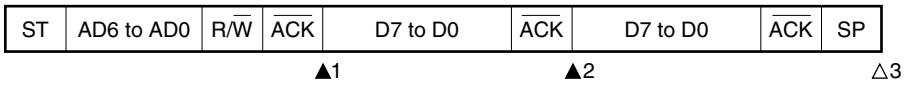
**(i) When WTIM0 = 0**




▲1: IICAS0 = 10001110B
▲2: IICAS0 = 01000000B
△3: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1

**(ii) When WTIM0 = 1**

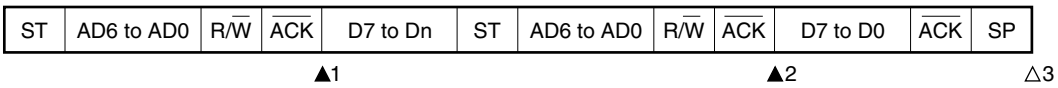



▲1: IICAS0 = 10001110B
▲2: IICAS0 = 01000100B
△3: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1

**(d) When loss occurs due to restart condition during data transfer**

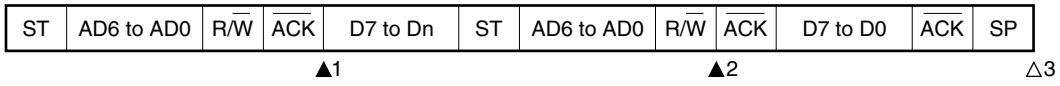
**(i) Not extension code (Example: unmatches with SVA0)**




▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 01000110B
△3: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care
n = 6 to 0

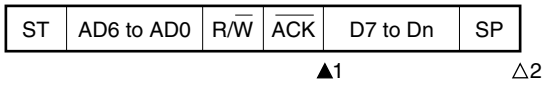
**(ii) Extension code**




▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 01100010B
Sets LREL0 = 1 by software
△3: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care
n = 6 to 0

**(e) When loss occurs due to stop condition during data transfer**

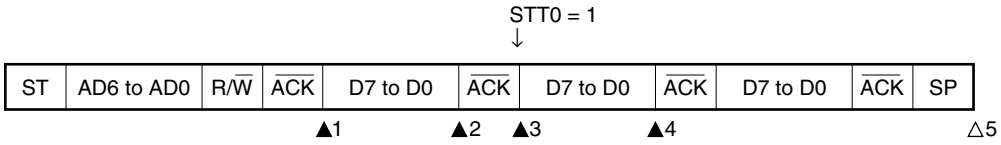



▲1: IICAS0 = 10000110B
△2: IICAS0 = 01000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care
n = 6 to 0

**(f) When arbitration loss occurs due to low-level data when attempting to generate a restart condition**

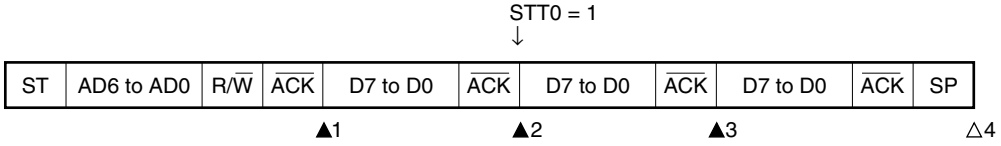
**(i) When WTIM0 = 0**




▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×000B (Sets WTIM0 to 1)
▲3: IICAS0 = 1000×100B (Clears WTIM0 to 0)
▲4: IICAS0 = 01000000B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**

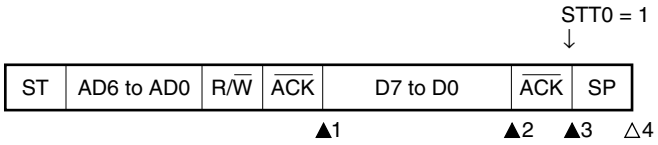



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×100B (Sets STT0 to 1)
▲3: IICAS0 = 01000100B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(g) When arbitration loss occurs due to a stop condition when attempting to generate a restart condition**

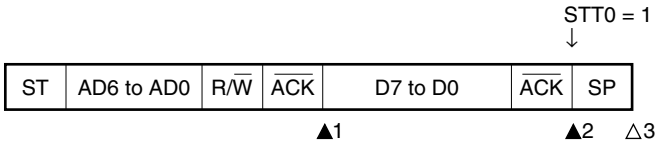
**(i) When WTIM0 = 0**




▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×000B (Sets WTIM0 to 1)
▲3: IICAS0 = 1000××00B (Sets STT0 to 1)
△4: IICAS0 = 01000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**

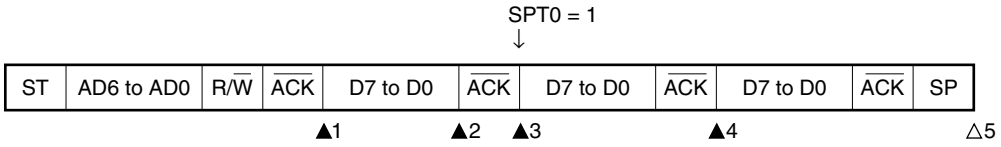



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000××00B (Sets STT0 to 1)
△3: IICAS0 = 01000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(h) When arbitration loss occurs due to low-level data when attempting to generate a stop condition**

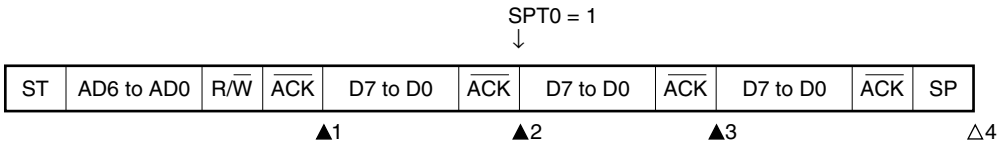
**(i) When WTIM0 = 0**




▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×000B (Sets WTIM0 to 1)
▲3: IICAS0 = 1000×100B (Clears WTIM0 to 0)
▲4: IICAS0 = 01000100B
△5: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

**(ii) When WTIM0 = 1**




▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×100B (Sets SPT0 to 1)
▲3: IICAS0 = 01000100B
△4: IICAS0 = 00000001B

**Remark** ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

## 15.6 Timing Charts

When using the $I^2C$ bus mode, the master device outputs an address via the serial bus to select one of several slave devices as its communication partner.

After outputting the slave address, the master device transmits the TRC0 bit (bit 3 of the IICA status register 0 (IICAS0)), which specifies the data transfer direction, and then starts serial communication with the slave device.

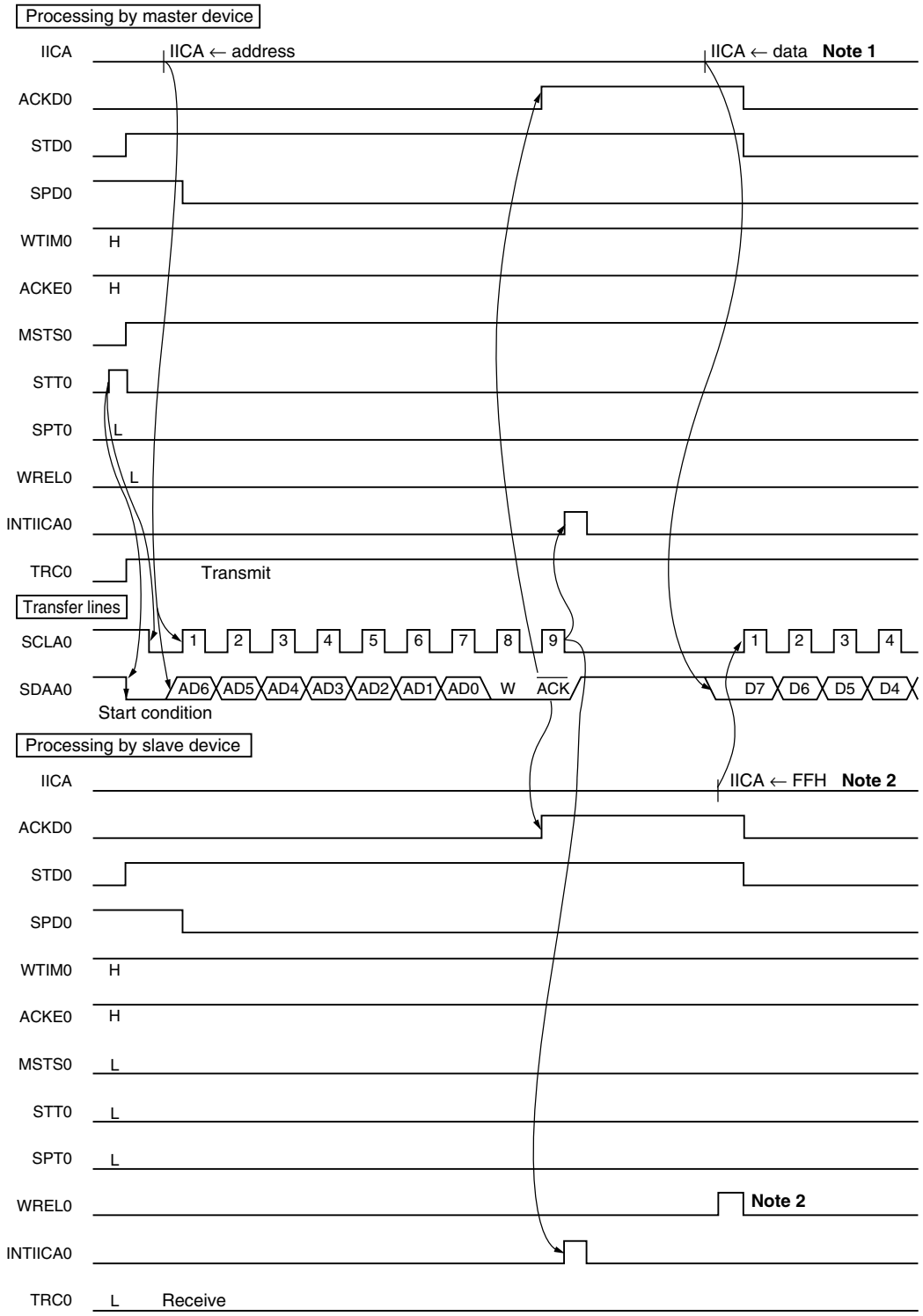
Figures 15-33 and 15-34 show timing charts of the data communication.

The IICA shift register (IICA)'s shift operation is synchronized with the falling edge of the serial clock (SCLA0). The transmit data is transferred to the SO latch and is output (MSB first) via the SDAA0 pin.

Data input via the SDAA0 pin is captured into IICA at the rising edge of SCLA0.

**Figure 15-33. Example of Master to Slave Communication**
**(When 9-Clock Wait Is Selected for Both Master and Slave) (1/3)**

**(1) Start condition ~ address**

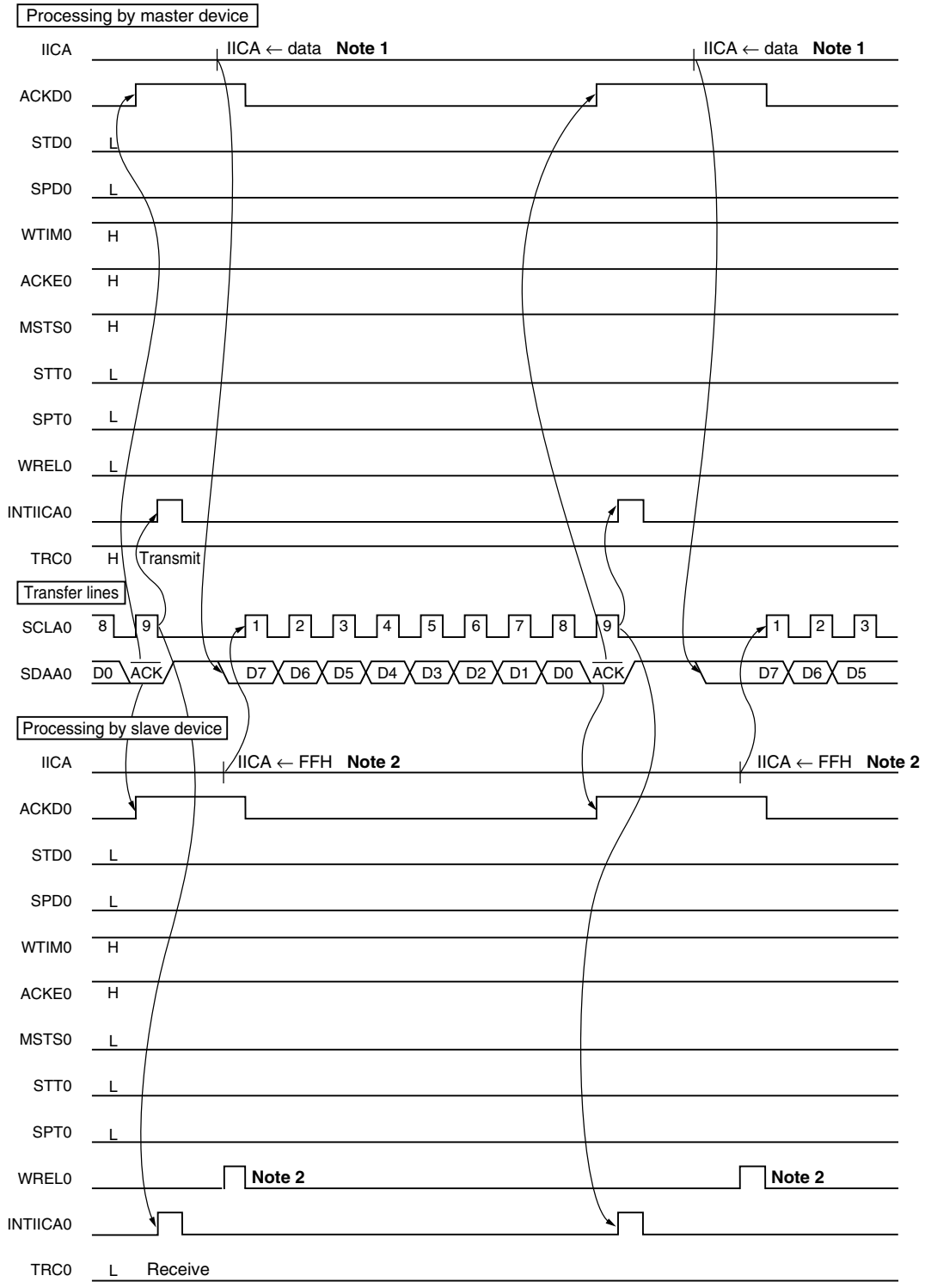



**Notes 1.** Write data to IICA, not setting WREL0, in order to cancel a wait state during master transmission.

**2.** To cancel slave wait, write “FFH” to IICA or set WREL0.

**Figure 15-33. Example of Master to Slave Communication**
**(When 9-Clock Wait Is Selected for Both Master and Slave) (2/3)**

**(2) Data**

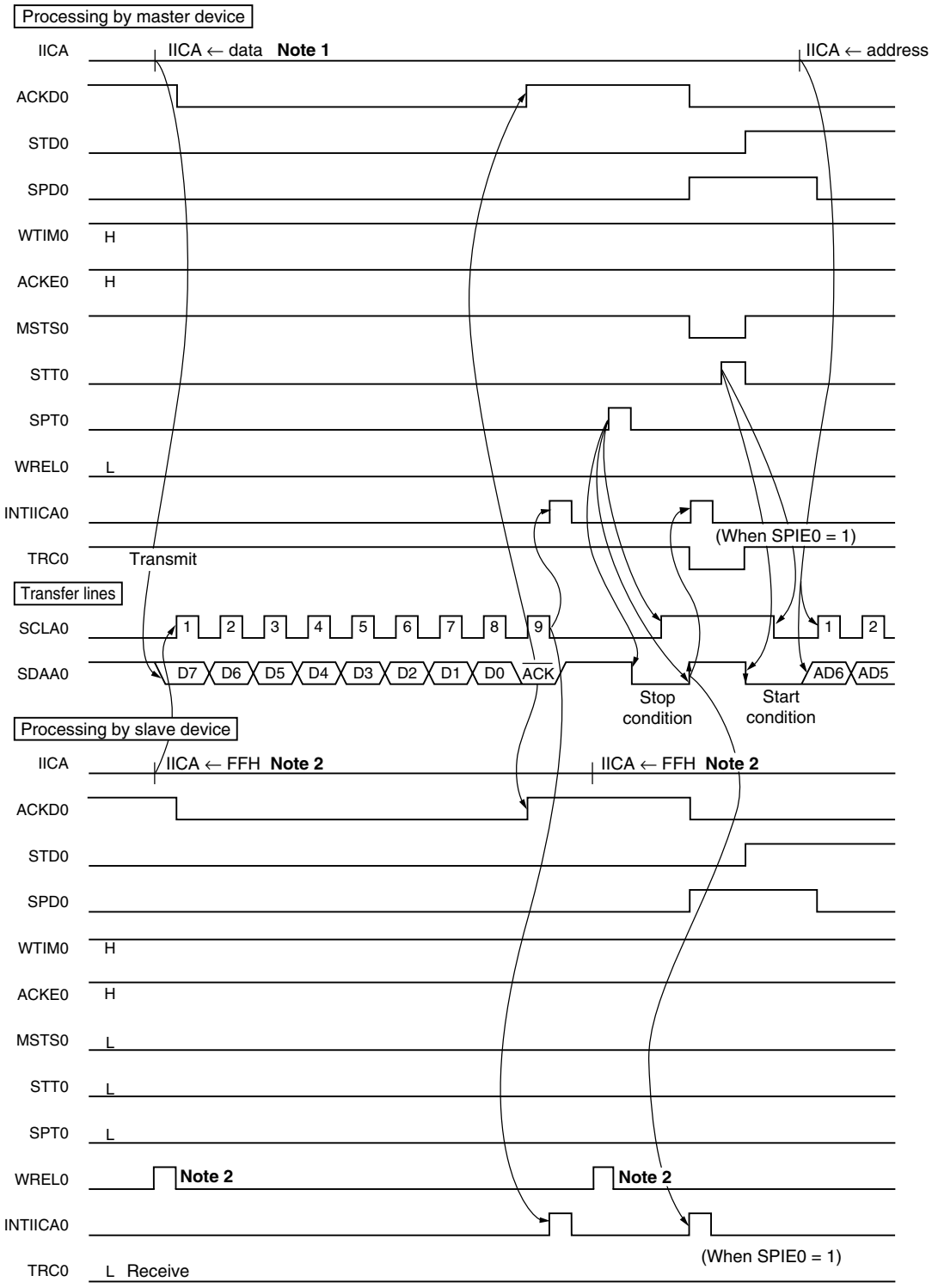



**Notes 1.** Write data to IICA, not setting WREL0, in order to cancel a wait state during master transmission.

**2.** To cancel slave wait, write “FFH” to IICA or set WREL0.

**Figure 15-33. Example of Master to Slave Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (3/3)**

**(3) Stop condition**

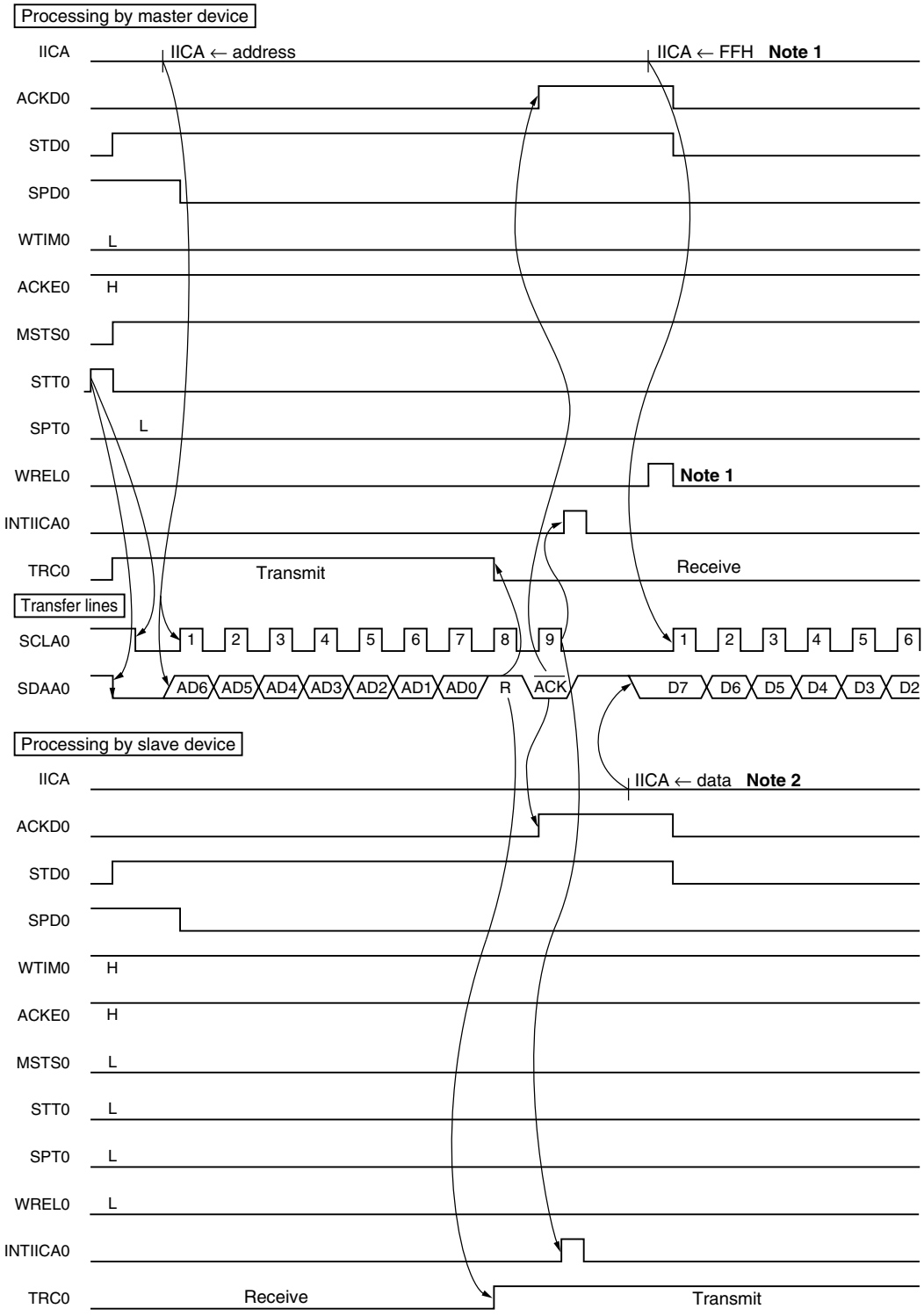



**Notes 1.** Write data to IICA, not setting WREL0, in order to cancel a wait state during master transmission.

**2.** To cancel slave wait, write “FFH” to IICA or set WREL0.

**Figure 15-34. Example of Slave to Master Communication**
**(When 8-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (1/3)**

**(1) Start condition ~ address**

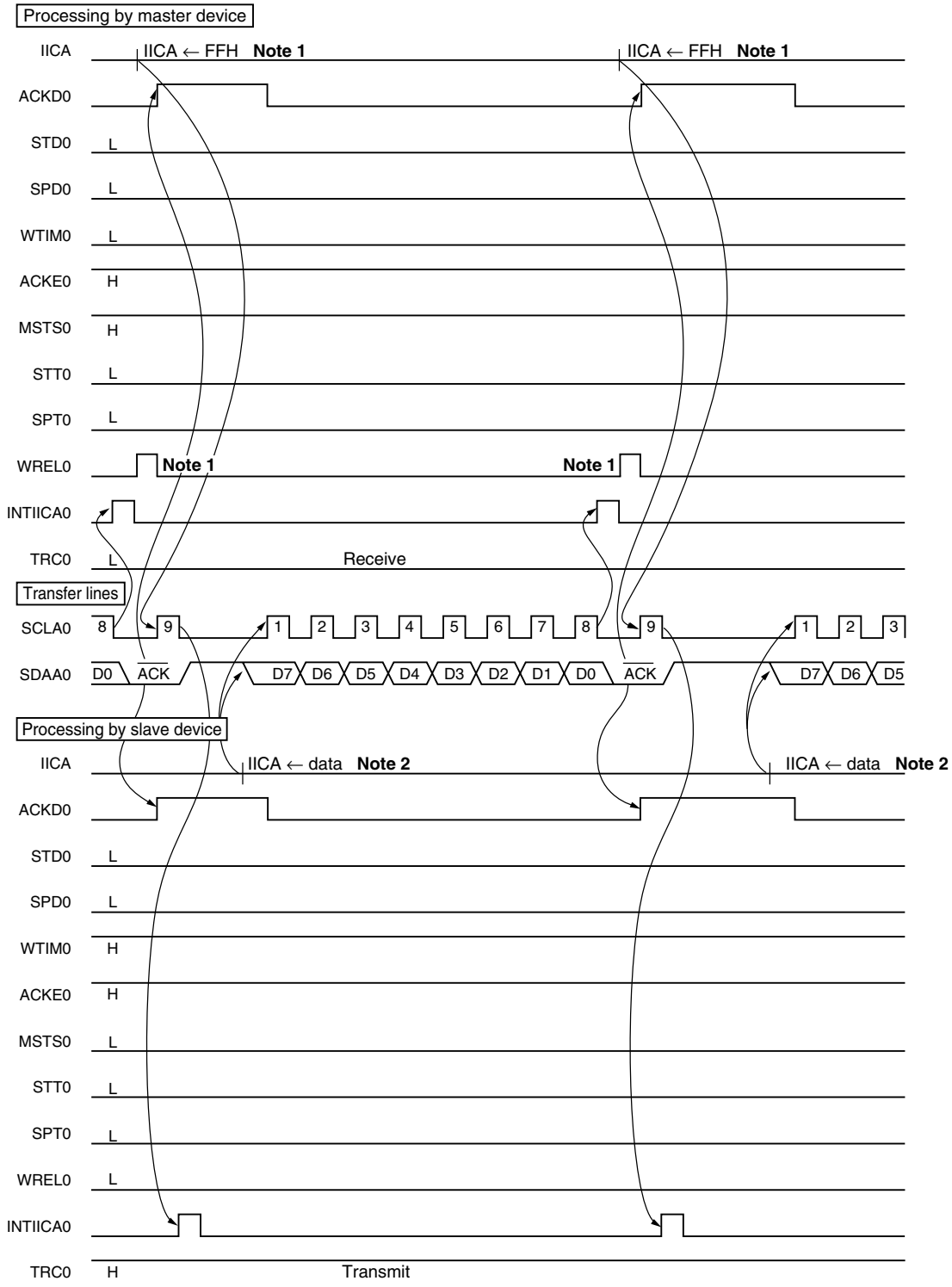



**Notes 1.** To cancel master wait, write “FFH” to IICA or set WREL0.

**2.** Write data to IICA, not setting WREL0, in order to cancel a wait state during slave transmission.

**Figure 15-34. Example of Slave to Master Communication**
**(When 8-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (2/3)**

**(2) Data**

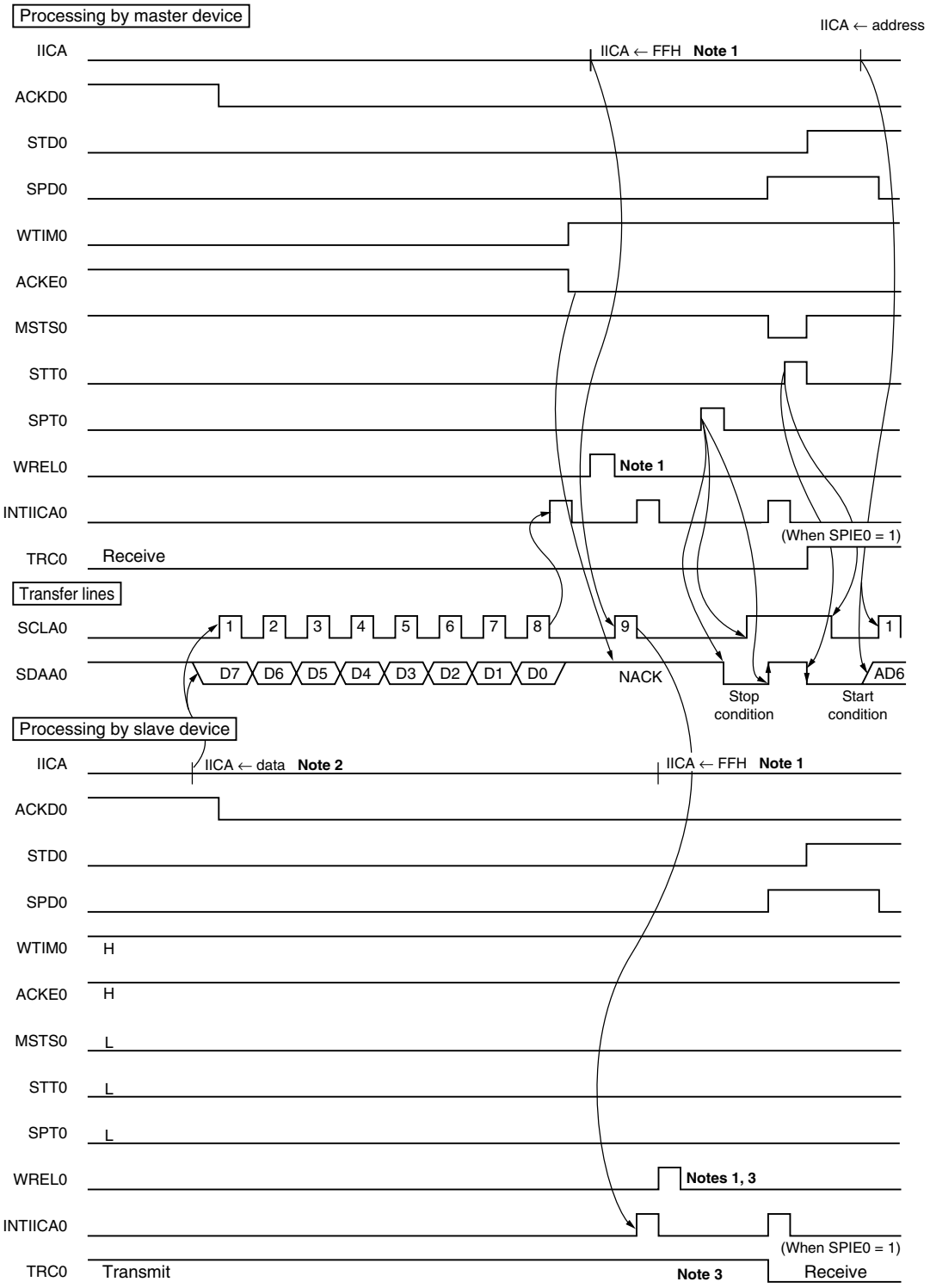



**Notes 1.** To cancel master wait, write “FFH” to IICA or set WREL0.

**2.** Write data to IICA, not setting WREL0, in order to cancel a wait state during slave transmission.

**Figure 15-34. Example of Slave to Master Communication**
**(When 8-Clock and 9-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (3/3)**

### (3) Stop condition




**Notes 1.** To cancel wait, write “FFH” to IICA or set WREL0.

**2.** Write data to IICA, not setting WREL0, in order to cancel a wait state during slave transmission.

**3.** If a wait state during slave transmission is canceled by setting WREL0, TRC0 will be cleared.

# CHAPTER 16 SERIAL INTERFACES CSI10 AND CSI11

<R>

| Item | 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) |
|---|---|---|---|---|---|
| | 16 pins | 20 pins | 25, 32 pins | 30 pins | 40, 44, 48 pins |
| Serial interface CSI10 | – | – | – | √ | √ |
| Serial interface CSI11 | – | – | √ | – | √ |

**Remark** √: Mounted, –: Not mounted

## 16.1 Functions of Serial Interfaces CSI10 and CSI11

Serial interfaces CSI10 and CSI11 have the following two modes.

**(1) Operation stop mode**

This mode is used when serial communication is not performed and can enable a reduction in the power consumption.

For details, refer to **16.4.1 Operation stop mode**.

**(2) 3-wire serial I/O mode (MSB/LSB-first selectable)**

This mode is used to communicate 8-bit data using three lines: a serial clock line ($\overline{\text{SCK1n}}$) and two serial data lines (SI1n and SO1n).

The processing time of data communication can be shortened in the 3-wire serial I/O mode because transmission and reception can be simultaneously executed.

In addition, whether 8-bit data is communicated with the MSB or LSB first can be specified, so this interface can be connected to any device.

The 3-wire serial I/O mode is used for connecting peripheral ICs and display controllers with a clocked serial interface.

For details, refer to **16.4.2 3-wire serial I/O mode**.

**Remark** 78K0/KA2-L (25 pins, 32 pins): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

## 16.2 Configuration of Serial Interfaces CSI10 and CSI11

Serial interfaces CSI10 and CSI11 include the following hardware.

**Table 16-1. Configuration of Serial Interfaces CSI10 and CSI11**

| Item | Configuration |
|---|---|
| Controller | Transmit controller<br>Clock start/stop controller & clock phase controller |
| Registers | Transmit buffer register 1n (SOTB1n)<br>Serial I/O shift register 1n (SIO1n) |
| Control registers | Serial operation mode register 1n (CSIM1n)<br>Serial clock selection register 1n (CSIC1n)<br>Port alternate switch control register (MUXSEL)<br>Port mode register x (PMx)<br>Port register x (Px) |

**Remark** 78K0/KA2-L (25 pins, 32 pins): n = 1, x = 0, 3
78K0/KB2-L: n = 0, x = 1
78K0/KC2-L: n = 0, 1, x = 1, 4, 6, 12

**Figure 16-1. Block Diagram of Serial Interface CSI10 (78K0/KB2-L and 78K0/KC2-L)**

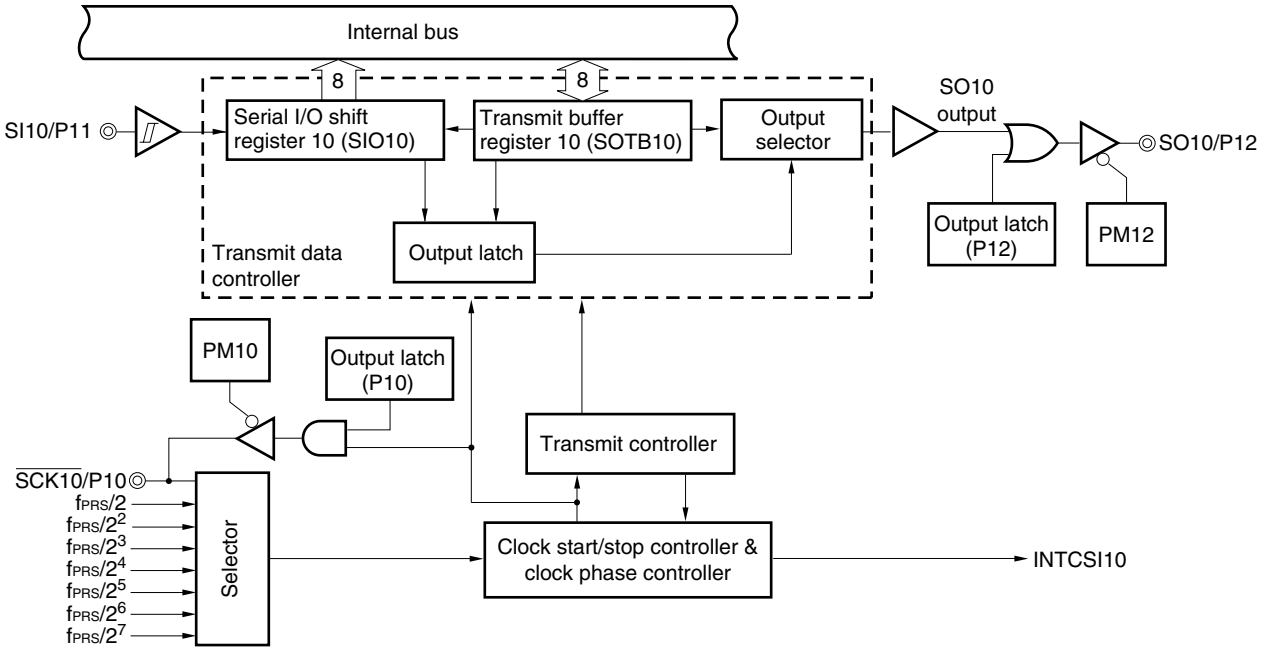
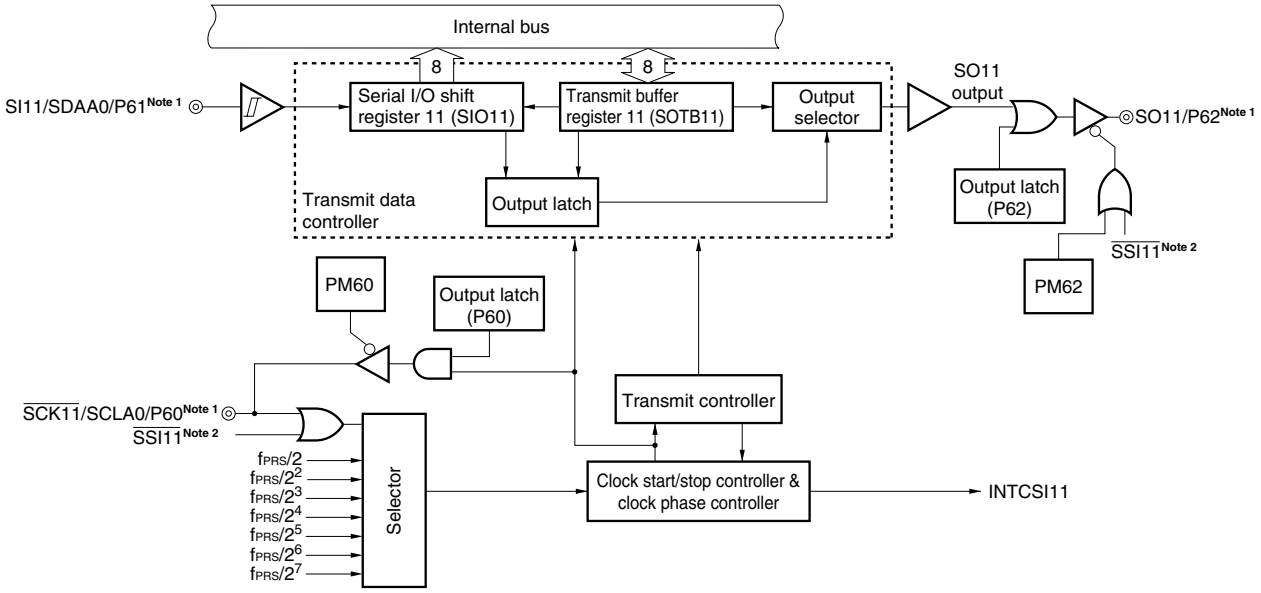


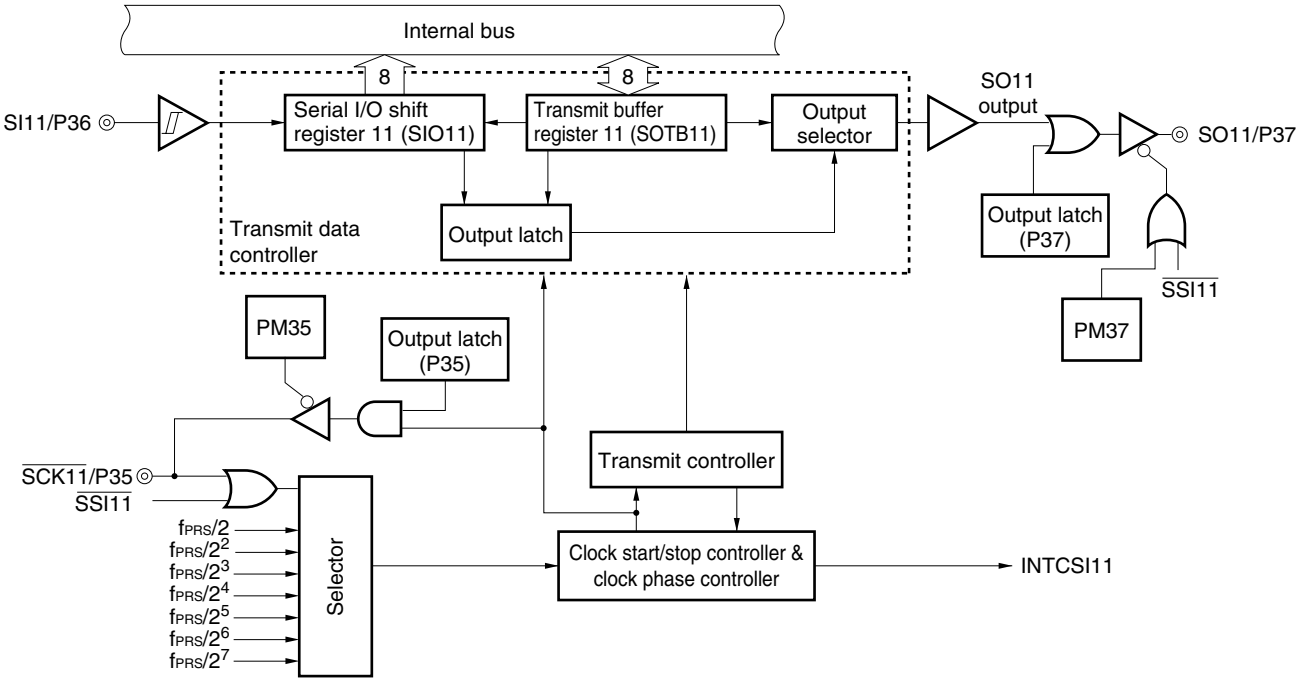
**Figure 16-2. Block Diagram of Serial Interface CSI11 (78K0/KC2-L)**




**Notes 1.** For 78K0/KC2-L (44, 48-pin products), by MUXSEL register setting, $\overline{\text{SCK11}}$, SI11, and SO11 can be assigned as P40, P41, and P120, respectively.

**2.** 48-pin products of 78K0/KC2-L only

**Figure 16-3. Block Diagram of Serial Interface CSI11 (78K0/KA2-L (25, 32-pin products))**



**(1) Transmit buffer register 1n (SOTB1n)**

This register sets the transmit data.

Transmission/reception is started by writing data to SOTB1n when bit 7 (CSIE1n) and bit 6 (TRMD1n) of serial operation mode register 1n (CSIM1n) is 1.

The data written to SOTB1n is converted from parallel data into serial data by serial I/O shift register 1n, and output to the serial output pin (SO1n).

SOTB1n can be written or read by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Cautions 1. Do not access SOTB1n when CSOT1n = 1 (during serial communication).**

**2. In the slave mode, transmission/reception is started when data is written to SOTB11 with a low level input to the $\overline{\text{SSI11}}$ pin. For details on the transmission/reception operation, refer to 16.4.2 (2) Communication operation.**

**Remarks 1.** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**2.** The $\overline{\text{SSI11}}$ pin is available only in 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L (48-pin products).

**(2) Serial I/O shift register 1n (SIO1n)**

This is an 8-bit register that converts data from parallel data into serial data and vice versa.

This register can be read by an 8-bit memory manipulation instruction.

Reception is started by reading data from SIO1n if bit 6 (TRMD1n) of serial operation mode register 1n (CSIM1n) is 0.

During reception, the data is read from the serial input pin (SI1n) to SIO1n.

Reset signal generation clears this register to 00H.

**Cautions 1. Do not access SIO1n when CSOT1n = 1 (during serial communication).**

**2. In the slave mode, reception is started when data is read from SIO11 with a low level input to the $\overline{\text{SSI11}}$ pin. For details on the reception operation, refer to 16.4.2 (2) Communication operation.**

**Remarks 1.** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**2.** The $\overline{\text{SSI11}}$ pin is available only in 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L (48-pin products).

## 16.3 Registers Controlling Serial Interfaces CSI10 and CSI11

Serial interfaces CSI10 and CSI11 are controlled by the following five registers.

- Serial operation mode register 1n (CSIM1n)
- Serial clock selection register 1n (CSIC1n)
- Port alternate switch control register (MUXSEL)
- Port mode register x (PMx)
- Port register x (Px)

**Remark** 78K0/KA2-L (25, 32-pin products): n = 1, x = 0, 3
78K0/KB2-L: n = 0, x = 1
78K0/KC2-L: n = 0, 1, x = 1, 4, 6, 12

**(1) Serial operation mode register 1n (CSIM1n)**

CSIM1n is used to select the operation mode and enable or disable operation.

CSIM1n can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**Figure 16-4. Format of Serial Operation Mode Register 10 (CSIM10) (78K0/KB2-L and 78K0/KC2-L)**

Address: FF80H After reset: 00H R/W[Note 1]

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CSIM10 | CSIE10 | TRMD10 | 0 | DIR10 | 0 | 0 | 0 | CSOT10 |

| CSIE10 | Operation control in 3-wire serial I/O mode |
|---|---|
| 0 | Disables operation[Note 2] and asynchronously resets the internal circuit[Note 3]. |
| 1 | Enables operation |

| TRMD10[Note 4] | Transmit/receive mode control |
|---|---|
| 0[Note 5] | Receive mode (transmission disabled). |
| 1 | Transmit/receive mode |

| DIR10[Note 6] | First bit specification |
|---|---|
| 0 | MSB |
| 1 | LSB |

| CSOT10 | Communication status flag |
|---|---|
| 0 | Communication is stopped. |
| 1 | Communication is in progress. |

**Notes 1.** Bit 0 is a read-only bit.

**2.** To use P10/$\overline{SCK10}$ and P12/SO10 as general-purpose ports, set CSIM10 in the default status (00H).

**3.** Bit 0 (CSOT10) of CSIM10 and serial I/O shift register 10 (SIO10) are reset.

**4.** Do not rewrite TRMD10 when CSOT10 = 1 (during serial communication).

**5.** The SO10 output (refer to **Figure 16-1**) is fixed to the low level when TRMD10 is 0. Reception is started when data is read from SIO10.

**6.** Do not rewrite DIR10 when CSOT10 = 1 (during serial communication).

**Caution Be sure to clear bit 5 to 0.**

**Figure 16-5. Format of Serial Operation Mode Register 11 (CSIM11)**
**(78K0/KA2-L (25, 32-pin products), 78K0/KC2-L)**

Address: FF88H After reset: 00H R/W[Note 1]

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CSIM11 | CSIE11 | TRMD11 | SSE11[Note 2] | DIR11 | 0 | 0 | 0 | CSOT11 |

| CSIE11 | Operation control in 3-wire serial I/O mode |
|---|---|
| 0 | Disables operation[Note 3] and asynchronously resets the internal circuit[Note 4]. |
| 1 | Enables operation |

| TRMD11[Note 5] | Transmit/receive mode control |
|---|---|
| 0[Note 6] | Receive mode (transmission disabled). |
| 1 | Transmit/receive mode |

| SSE11[Notes 7, 8] | $\overline{\text{SSI11}}$ pin use selection |
|---|---|
| 0 | $\overline{\text{SSI11}}$ pin is not used |
| 1 | $\overline{\text{SSI11}}$ pin is used |

| DIR11[Note 9] | First bit specification |
|---|---|
| 0 | MSB |
| 1 | LSB |

| CSOT11 | Communication status flag |
|---|---|
| 0 | Communication is stopped. |
| 1 | Communication is in progress. |

**Notes 1.** Bit 0 is a read-only bit.

**2.** 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L(48-pin products) only. For the products of the other than 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L(48-pin products), be sure to clear to 0.

**3.** To use P62/SO11, P60/$\overline{\text{SCK11}}$/SCLA0, and P42/$\overline{\text{SSI11}}$/PCL/INTP6 as general-purpose ports when CSISEL = 0, set CSIM11 in the default status (00H).
To use P120/SO11/INTP0/EXLVI, P40/$\overline{\text{SCK11}}$/RTCCL/RTCDIV, and P42/$\overline{\text{SSI11}}$/PCL/INTP6 as general-purpose ports when CSISEL = 1, set CSIM11 in the default status (00H).
To use P37/SO11, P35/$\overline{\text{SCK11}}$, and P02/$\overline{\text{SSI11}}$/INTP5 as general-purpose ports when CSISEL = 1, set CSIM11 in the default status (00H).

**4.** Bit 0 (CSOT11) of CSIM11 and serial I/O shift register 11 (SIO11) are reset.

**5.** Do not rewrite TRMD11 when CSOT11 = 1 (during serial communication).

**6.** The SO11 output (refer to **Figure 16-2**) is fixed to the low level when TRMD11 is 0. Reception is started when data is read from SIO11.

**7.** Do not rewrite SSE11 when CSOT11 = 1 (during serial communication).

**8.** Before setting this bit to 1, fix the $\overline{\text{SSI11}}$ pin input level to 0 or 1.

**9.** Do not rewrite DIR11 when CSOT11 = 1 (during serial communication).

**Remark** The $\overline{\text{SSI11}}$ pin is available only in 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L(48-pin products).

### (2) Serial clock selection register 1n (CSIC1n)

This register specifies the timing of the data transmission/reception and sets the serial clock.

CSIC1n can be set by a 1-bit or 8-bit memory manipulation instruction.

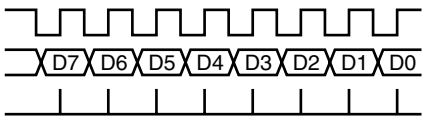
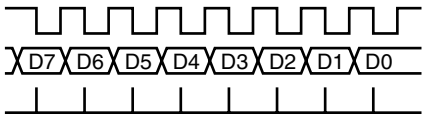
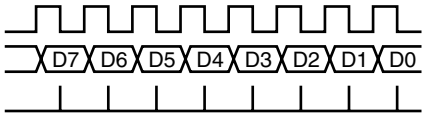
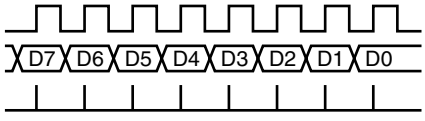
Reset signal generation clears this register to 00H.

**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**Figure 16-6. Format of Serial Clock Selection Register 10 (CSIC10) (78K0/KB2-L and 78K0/KC2-L)**

Address: FF81H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CSIC10 | 0 | 0 | 0 | CKP10 | DAP10 | CKS102 | CKS101 | CKS100 |

| CKP10 | DAP10 | Specification of data transmission/reception timing | Type |
|---|---|---|---|
| 0 | 0 |  | 1 |
| 0 | 1 |  | 2 |
| 1 | 0 |  | 3 |
| 1 | 1 |  | 4 |

| CKS102 | CKS101 | CKS100 | CSI10 serial clock selection[Note 1] | | | | Mode |
|---|---|---|---|---|---|---|---|
| | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz | |
| 0 | 0 | 0 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz | Master mode |
| 0 | 0 | 1 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz | |
| 0 | 1 | 0 | $f_{PRS}/2^3$ | 250 kHz | 625 kHz | 1.25 MHz | |
| 0 | 1 | 1 | $f_{PRS}/2^4$ | 125 kHz | 312.5 kHz | 625 kHz | |
| 1 | 0 | 0 | $f_{PRS}/2^5$ | 62.5 kHz | 156.25 kHz | 312.5 kHz | |
| 1 | 0 | 1 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz | |
| 1 | 1 | 0 | $f_{PRS}/2^7$ | 15.63 kHz | 39.06 kHz | 78.13 kHz | |
| 1 | 1 | 1 | External clock input from $\overline{SCK10}$[Note 2] | | | | Slave mode |

**Note 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.

- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**Note 2.** Do not start communication with the external clock from the $\overline{\text{SCK10}}$ pin when the internal high-speed oscillation clock and high-speed system clock are stopped while the CPU operates with the subsystem clock, or when in the STOP mode.

**Cautions 1. Do not write to CSIC10 while CSIE10 = 1 (operation enabled).**

**2. To use P10/$\overline{\text{SCK10}}$ and P12/SO10 as general-purpose ports, set CSIC10 in the default status (00H).**

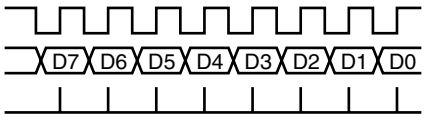
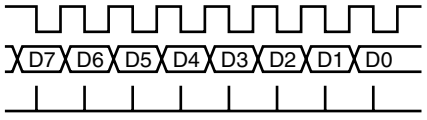
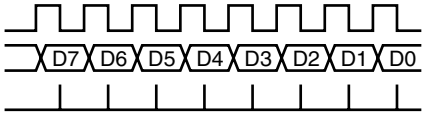
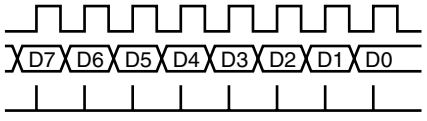
**3. The phase type of the data clock is type 1 after reset.**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**Figure 16-7. Format of Serial Clock Selection Register 11 (CSIC11)**
**(78K0/KA2-L (25, 32-pin products), 78K0/KC2-L)**

Address: FF89H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CSIC11 | 0 | 0 | 0 | CKP11 | DAP11 | CKS112 | CKS111 | CKS110 |

| CKP11 | DAP11 | Specification of data transmission/reception timing | Type |
|---|---|---|---|
| 0 | 0 |  | 1 |
| 0 | 1 |  | 2 |
| 1 | 0 |  | 3 |
| 1 | 1 |  | 4 |

| CKS112 | CKS111 | CKS110 | CSI11 serial clock selection[Note 1] | | | | Mode |
|---|---|---|---|---|---|---|---|
| | | | | $f_{PRS}$ = 2 MHz | $f_{PRS}$ = 5 MHz | $f_{PRS}$ = 10 MHz | |
| 0 | 0 | 0 | $f_{PRS}/2$ | 1 MHz | 2.5 MHz | 5 MHz | Master mode |
| 0 | 0 | 1 | $f_{PRS}/2^2$ | 500 kHz | 1.25 MHz | 2.5 MHz | |
| 0 | 1 | 0 | $f_{PRS}/2^3$ | 250 kHz | 625 kHz | 1.25 MHz | |
| 0 | 1 | 1 | $f_{PRS}/2^4$ | 125 kHz | 312.5 kHz | 625 kHz | |
| 1 | 0 | 0 | $f_{PRS}/2^5$ | 62.5 kHz | 156.25 kHz | 312.5 kHz | |
| 1 | 0 | 1 | $f_{PRS}/2^6$ | 31.25 kHz | 78.13 kHz | 156.25 kHz | |
| 1 | 1 | 0 | $f_{PRS}/2^7$ | 15.63 kHz | 39.06 kHz | 78.13 kHz | |
| 1 | 1 | 1 | External clock input from $\overline{SCK11}$[Note 2] | | | | Slave mode |

**Notes 1.** If the peripheral hardware clock ($f_{PRS}$) operates on the high-speed system clock ($f_{XH}$) (XSEL = 1), the $f_{PRS}$ operating frequency varies depending on the supply voltage.
- $V_{DD}$ = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- $V_{DD}$ = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

**2.** Do not start communication with the external clock from the $\overline{SCK11}$ pin when the internal high-speed oscillation clock and high-speed system clock are stopped while the CPU operates with the subsystem clock, or when in the STOP mode.

**Cautions 1. Do not write to CSIC11 while CSIE11 = 1 (operation enabled).**

**2. To use P62/SO11 and P60/$\overline{\text{SCK11}}$/SCLA0 as general-purpose ports when CSISEL = 0, set CSIC11 in the default status (00H).**

**To use P120/SO11/INTP0/EXLVI and P40/$\overline{\text{SCK11}}$/RTCCL/RTCDIV as general-purpose ports when CSISEL = 1, set CSIC11 in the default status (00H).**

**To use P37/SO11, P35/$\overline{\text{SCK11}}$, and P02/$\overline{\text{SSI11}}$/INTP5 as general-purpose ports, set CSIM11 in the default status (00H).**

**3. The phase type of the data clock is type 1 after reset.**

**Remark** $f_{PRS}$: Peripheral hardware clock frequency

**(3) Port alternate switch control register (MUXSEL) (78K0/KC2-L (44, 48-pin products) only)**

This register assigns the pin function to be used with serial interface CSI11. $\overline{\text{SCK11}}$ is assigned to P60, SI11 to P61, and SO11 to P62 by default.

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

Reset signal generation clears MUXSEL to 00H.

**Figure 16-8. Format of Port Alternate Switch Control Register (MUXSEL) (78K0/KC2-L (44, 48-pin products))**

Address: FF3FH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| MUXSEL | 0 | 0 | 0 | 0 | 0 | CSISEL | 0 | 0 |

| CSISEL | Pin function assignment to be used with serial interface CSI11 |
|---|---|
| 0 | $\overline{\text{SCK11}}$/P60, SI11/P61, SO11/P62 |
| 1 | $\overline{\text{SCK11}}$/P40, SI11/P41, SO11/P120 |

**(4) Port mode registers 0, 1, 3, 4, 6, 12 (PM0, PM1, PM3, PM4, PM6, PM12)**

These registers set input/output of ports 0, 1, 3, 4, 6, and 12 in 1-bit units.

<R> • 78K0/KA2-L (25, 32-pin products)

When using P35/SCK11 as the clock output pin of the serial interface, clear PM35 to 0, and set the output latches of P35 to 1.

When using P37/SO11 as the data output pin of the serial interface, clear PM37 and the output latches of P37 to 0.

When using P35/SCK10 as the clock input pins of the serial interface, P36/SI11 as the data input pins of the serial interface, and P02/SSI11/INTP5 as the chip select input pin of the serial interface, set PM35, PM36, and PM02 to 1. At this time, the output latches of P35, P36, and P02 may be 0 or 1.

• 78K0/KB2-L

When using P10/SCK10 as the clock output pin of the serial interface, clear PM10 to 0, and set the output latches of P10 to 1.

When using P12/SO10 as the data output pin of the serial interface, clear PM12 and the output latches of P12 to 0.

When using P10/SCK10 as the clock input pin of the serial interface and P11/SI10 as the data input pin of the serial interface, set PM10 and PM11 to 1. At this time, the output latches of P10 and P11 may be 0 or 1.

• 78K0/KC2-L (when CSISEL = 0)

When using P10/SCK10 and P60/SCK11/SCLA0 as the clock output pins of the serial interface, clear PM10 and PM60 to 0, and set the output latches of P10 and P60 to 1.

When using P12/SO10 and P62/SO11 as the data output pins of the serial interface, clear PM12 and PM62 and the output latches of P12 and P62 to 0.

When using P10/SCK10 and P60/SCK11/SCLA0 as the clock input pins of the serial interface, P11/SI10 and P61/SI11/SDAA0 as the data input pins of the serial interface, and P42/SSI11/PCL/INTP6 as the chip select input pin of the serial interface, set PM10, PM60, PM11, PM61, and PM42 to 1. At this time, the output latches of P10, P60, P11, P61, and P42 may be 0 or 1.

• 78K0/KC2-L (when CSISEL = 1)

When using P10/SCK10 and P40/SCK11/RTCCL/RTCDIV as the clock output pins of the serial interface, clear PM10 and PM40 to 0, and set the output latches of P10 and P40 to 1.

When using P12/SO10 and P120/SO11/INTP0/EXLVI as the data output pins of the serial interface, clear PM12 and PM120 and the output latches of P12 and P120 to 0.

When using P10/SCK10 and P40/SCK11/RTCCL/RTCDIV as the clock input pins of the serial interface, P11/SI10 and P41/SI11/RTC1HZ as the data input pins of the serial interface, and P42/SSI11/PCL/INTP6 as the chip select input pin of the serial interface, set PM10, PM40, PM11, PM41, and PM42 to 1. At this time, the output latches of P10, P40, P11, P41, and P42 may be 0 or 1.

**Remark** The SSI11 pin is available only in 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L(48-pin products).

PM0, PM1, PM3, PM4, PM6, and PM12 can be set by a 1-bit or 8-bit memory manipulation instruction.
Reset signal generation sets these registers to FFH.

<R> **Figure 16-9. Format of Port Mode Register 0 (PM0)**

Address: FF20H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM0 | 1 | 1 | 1 | 1 | 1 | PM02 | PM01 Note 1 | PM00 Note 2 |

| PM0n | P1n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Notes 1.** 32-pin products only
**2.** 25-pin products only

**Remark** The figure shown above presents the format of port mode register 1 of the 78K0/KB2-L and 78K0/KA2-L (25, 32-pin products).

**Figure 16-10. Format of Port Mode Register 1 (PM1)**

Address: FF21H After reset: FFH R/W

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

| PM1n | P1n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 1 of the 78K0/KB2-L and 78K0/KC2-L.

<R> **Figure 16-11. Format of Port Mode Register 3 (PM3)**

Address: FF23H After reset: FFH R/W

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

| PM3n | P3n pin I/O mode selection (n = 0 to 7) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 1 of the 78K0/KB2-L and 78K0/KA2-L (25, 32-pin products).

**Figure 16-12. Format of Port Mode Register 4 (PM4)**

Address: FF24H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM4 | 1 | 1 | 1 | 1 | 1 | PM42 Note 1 | PM41 Note 2 | PM40 Note 2 |

| PM4n | P4n pin I/O mode selection (n = 0 to 2) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Notes 1.** 48-pin products only

**2.** 44-pin and 48-pin products only

**Remark** The figure shown above presents the format of port mode register 4 of the 78K0/KC2-L.

**Figure 16-13. Format of Port Mode Register 6 (PM6)**

Address: FF26H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM6 | 1 | 1 | 1 | 1 | PM63 | PM62 | PM61 | PM60 |

| PM6n | P6n pin I/O mode selection (n = 0 to 3) |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 6 of the 78K0/KC2-L. For the format of port mode register 6 of other products, refer to **(1) Port mode registers (PMxx)** in **4.3 Registers Controlling Port Function**.

**Figure 16-14. Format of Port Mode Register 12 (PM12)**

Address: FF2CH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM120 |

| PM120 | P120 pin I/O mode selection |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

**Remark** The figure shown above presents the format of port mode register 12 of the 78K0/KB2-L and 78K0/KC2-L.

## 16.4 Operation of Serial Interfaces CSI10 and CSI11

Serial interfaces CSI10 and CSI11 can be used in the following two modes.

- Operation stop mode
- 3-wire serial I/O mode

### 16.4.1 Operation stop mode

Serial communication is not executed in this mode. Therefore, the power consumption can be reduced. In addition, the $\overline{SCK1n}$, SI1n, SO1n, and $\overline{SSI11}$ pins can be used as ordinary I/O port pins in this mode.

**(1) Register used**

The operation stop mode is set by serial operation mode register 1n (CSIM1n).

To set the operation stop mode, clear bit 7 (CSIE1n) of CSIM1n to 0.

**(a) Serial operation mode register 1n (CSIM1n)**

CSIM1n can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears CSIM1n to 00H.

**Remarks 1.** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**2.** The $\overline{SSI11}$ pin is available only in 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L (48-pin products).

- Serial operation mode register 10 (CSIM10)

Address: FF80H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CSIM10 | CSIE10 | TRMD10 | 0 | DIR10 | 0 | 0 | 0 | CSOT10 |

| CSIE10 | Operation control in 3-wire serial I/O mode |
|---|---|
| 0 | Disables operation[Note 1] and asynchronously resets the internal circuit[Note 2]. |

**Notes 1.** To use P10/$\overline{SCK10}$ and P12/SO10 as general-purpose ports, set CSIM10 in the default status (00H).

**2.** Bit 0 (CSOT10) of CSIM10 and serial I/O shift register 10 (SIO10) are reset.

- Serial operation mode register 11 (CSIM11)

Address: FF88H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| CSIM11 | CSIE11 | TRMD11 | SSE11 | DIR11 | 0 | 0 | 0 | CSOT11 |

| CSIE11 | Operation control in 3-wire serial I/O mode |
|---|---|
| 0 | Disables operation[Note 1] and asynchronously resets the internal circuit[Note 2]. |

**Notes 1.** To use P62/SO11, P60/$\overline{\text{SCK11}}$/SCLA0, and P42/$\overline{\text{SSI11}}$/PCL/INTP6 as general-purpose ports when CSISEL = 0, set CSIM11 in the default status (00H).
To use P120/SO11/INTP0/EXLVI, P40/$\overline{\text{SCK11}}$/RTCCL/RTCDIV, and P42/$\overline{\text{SSI11}}$/PCL/INTP6 as general-purpose ports when CSISEL = 1, set CSIM11 in the default status (00H).
To use P37/SO11, P35/$\overline{\text{SCK11}}$, and P02/$\overline{\text{SSI11}}$/INTP5 as general-purpose ports when CSISEL = 1, set CSIM11 in the default status (00H).

**2.** Bit 0 (CSOT11) of CSIM11 and serial I/O shift register 11 (SIO11) are reset.

### 16.4.2 3-wire serial I/O mode

The 3-wire serial I/O mode is used for connecting peripheral ICs and display controllers with a clocked serial interface.

In this mode, communication is executed by using three lines: the serial clock ($\overline{\text{SCK1n}}$), serial output (SO1n), and serial input (SI1n) lines.

#### (1) Registers used

- Serial operation mode register 1n (CSIM1n)
- Serial clock selection register 1n (CSIC1n)
- Port mode register x (PMx)
- Port register x (Px)

The basic procedure of setting an operation in the 3-wire serial I/O mode is as follows.

<1> Set the CSIC1n register (refer to **Figures 16-6** and **16-7**).
<2> Set bits 4 to 6 (DIR1n, SSE11 (serial interface CSI11 only), and TRMD1n) of the CSIM1n register (refer to **Figures 16-4** and **16-5**).
<3> Set bit 7 (CSIE1n) of the CSIM1n register to 1. → Transmission/reception is enabled.
<4> Write data to transmit buffer register 1n (SOTB1n). → Data transmission/reception is started.
Read data from serial I/O shift register 1n (SIO1n). → Data reception is started.

**Caution Take relationship with the other party of communication when setting the port mode register and port register.**

**Remark** 78K0/KA2-L (25, 32-pin products): n = 0, x = 0, 3
78K0/KB2-L: n = 0, x = 1
78K0/KC2-L: n = 0, 1, x = 1, 4, 6, 12

The relationship between the register settings and pins is shown below.

**Table 16-2. Relationship Between Register Settings and Pins (1/4)**

**(a) Serial interface CSI10**

| CSIE10 | TRMD10 | PM11 | P11 | PM12 | P12 | PM10 | P10 | CSI10 Operation | Pin Function | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | SI10/P11 | SO10/P12 | $\overline{SCK10}$/P10 |
| 0 | 0 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | Stop | P11 | P12 | P10Note 2 |
| 1 | 0 | 1 | × | ×Note 1 | ×Note 1 | 1 | × | Slave receptionNote 3 | SI10 | P12 | $\overline{SCK10}$ (input)Note 3 |
| 1 | 1 | ×Note 1 | ×Note 1 | 0 | 0 | 1 | × | Slave transmissionNote 3 | P11 | SO10 | $\overline{SCK10}$ (input)Note 3 |
| 1 | 1 | 1 | × | 0 | 0 | 1 | × | Slave transmission/ receptionNote 3 | SI10 | SO10 | $\overline{SCK10}$ (input)Note 3 |
| 1 | 0 | 1 | × | ×Note 1 | ×Note 1 | 0 | 1 | Master reception | SI10 | P12 | $\overline{SCK10}$ (output) |
| 1 | 1 | ×Note 1 | ×Note 1 | 0 | 0 | 0 | 1 | Master transmission | P11 | SO10 | $\overline{SCK10}$ (output) |
| 1 | 1 | 1 | × | 0 | 0 | 0 | 1 | Master transmission/ reception | SI10 | SO10 | $\overline{SCK10}$ (output) |

**Notes 1.** Can be set as port function.

**2.** To use P10/$\overline{SCK10}$ as port pins, clear CKP10 to 0.

**3.** To use the slave mode, set CKS102, CKS101, and CKS100 to 1, 1, 1.

**Remark** ×: don't care

CSIE10: Bit 7 of serial operation mode register 10 (CSIM10)

TRMD10: Bit 6 of CSIM10

CKP10: Bit 4 of serial clock selection register 10 (CSIC10)

CKS102, CKS101, CKS100: Bits 2 to 0 of CSIC10

PM×: Port mode register

P×: Port output latch

**Table 16-2. Relationship Between Register Settings and Pins (2/4)**

**(b) Serial interface CSI11 (CSISEL = 0) (78K0/KC2-L)**

| CSIE11 | TRMD11 | SSE11 | PM61 | P61 | PM62 | P62 | PM60 | P60 | PM42 | P42 | CSI11 Operation | Pin Function | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | SI11/P61/ SDAA0/ INTP10 | SO11/ P62/ INTP9 | $\overline{\text{SCK11}}$/ P60/ SCLA0/ INTP11 | $\overline{\text{SSI11}}$/ P42/PCL/ INTP6 |
| 0 | 0 | × | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | Stop | P61/ SDAA0/ INTP10 | P62/ INTP9 | P60/ SCLA0/ INTP11 Note 2 | P42/ PCL/ INTP6 |
| 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | 1 | × | ×Note 1 | ×Note 1 | Slave reception Note 3 | SI11 | P62/ INTP9 | $\overline{\text{SCK11}}$ (input) Note 3 | P42/ PCL/ INTP6 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 1 | 0 | ×Note 1 | ×Note 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | Slave transmission Note 3 | P61/ INTP10 | SO11 | $\overline{\text{SCK11}}$ (input) Note 3 | P42/ PCL/ INTP6 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 1 | 0 | 1 | × | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | Slave transmission/ reception Note 3 | SI11 | SO11 | $\overline{\text{SCK11}}$ (input) Note 3 | P42/ PCL/ INTP6 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | 0 | 1 | ×Note 1 | ×Note 1 | Master reception | SI11 | P62/ INTP9 | $\overline{\text{SCK11}}$ (output) | P42/ PCL/ INTP6 |
| 1 | 1 | 0 | ×Note 1 | ×Note 1 | 0 | 0 | 0 | 1 | ×Note 1 | ×Note 1 | Master transmission | P61/ INTP10 | SO11 | $\overline{\text{SCK11}}$ (output) | P42/ PCL/ INTP6 |
| 1 | 1 | 0 | 1 | × | 0 | 0 | 0 | 1 | ×Note 1 | ×Note 1 | Master transmission/ reception | SI11 | SO11 | $\overline{\text{SCK11}}$ (output) | P42/ PCL/ INTP6 |

**Notes 1.** Can be set as port function.

**2.** To use P60/$\overline{\text{SCK11}}$/SCLA0/INTP11 as port pins, clear CKP11 to 0.

**3.** To use the slave mode, set CKS112, CKS111, and CKS110 to 1, 1, 1.

**Remarks 1.** ×: don't care
CSIE11: Bit 7 of serial operation mode register 11 (CSIM11)
TRMD11: Bit 6 of CSIM11
CKP11: Bit 4 of serial clock selection register 11 (CSIC11)
CKS112, CKS111, CKS110: Bits 2 to 0 of CSIC11
PM×: Port mode register
P×: Port output latch

**2.** The $\overline{\text{SSI11}}$ pin is available only in 48-pin products of 78K0/KC2-L.

**Table 16-2. Relationship Between Register Settings and Pins (3/4)**

**(c) Serial interface CSI11 (CSISEL = 1) (78K0/KC2-L)**

| CSIE11 | TRMD11 | SSE11 | PM41 | P41 | PM120 | P120 | PM40 | P40 | PM42 | P42 | CSI11 Operation | Pin Function | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | SI11/P41/RTC1HZ | SO11/P120/EXLVI/INTP0 | $\overline{\text{SCK11}}$/P40/RTCCL/RTCDIV | $\overline{\text{SSI11}}$/P42/PCL/INTP6 |
| 0 | 0 | × | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | Stop | P41/RTC1HZ | P120/EXLVI/INTP0 | P40/RTCCL/RTCDIV Note 2 | P42/PCL/INTP6 |
| 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | 1 | × | ×Note 1 | ×Note 1 | Slave reception Note 3 | SI11 | P120/EXLVI/INTP0 | $\overline{\text{SCK11}}$ (input) Note 3 | P42/PCL/INTP6 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 1 | 0 | ×Note 1 | ×Note 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | Slave transmission Note 3 | P41/RTC1HZ | SO11 | $\overline{\text{SCK11}}$ (input) Note 3 | P42/PCL/INTP6 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 1 | 0 | 1 | × | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | Slave transmission/reception Note 3 | SI11 | SO11 | $\overline{\text{SCK11}}$ (input) Note 3 | P42/PCL/INTP6 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | 0 | 1 | ×Note 1 | ×Note 1 | Master reception | SI11 | P120/EXLVI/INTP0 | $\overline{\text{SCK11}}$ (output) | P42/PCL/INTP6 |
| 1 | 1 | 0 | ×Note 1 | ×Note 1 | 0 | 0 | 0 | 1 | ×Note 1 | ×Note 1 | Master transmission | P41/RTC1HZ | SO11 | $\overline{\text{SCK11}}$ (output) | P42/PCL/INTP6 |
| 1 | 1 | 0 | 1 | × | 0 | 0 | 0 | 1 | ×Note 1 | ×Note 1 | Master transmission/reception | SI11 | SO11 | $\overline{\text{SCK11}}$ (output) | P42/PCL/INTP6 |

**Notes 1.** Can be set as port function.

**2.** To use P40/$\overline{\text{SCK11}}$/RTCCL/RTCDIV as port pins, clear CKP11 to 0.

**3.** To use the slave mode, set CKS112, CKS111, and CKS110 to 1, 1, 1.

**Remarks 1.** ×: don't care
CSIE11: Bit 7 of serial operation mode register 11 (CSIM11)
TRMD11: Bit 6 of CSIM11
CKP11: Bit 4 of serial clock selection register 11 (CSIC11)
CKS112, CKS111, CKS110: Bits 2 to 0 of CSIC11
PM×: Port mode register
P×: Port output latch

**2.** The $\overline{\text{SSI11}}$ pin is available only in 48-pin products of 78K0/KC2-L.

**Table 16-2. Relationship Between Register Settings and Pins (4/4)**

**(d) Serial interface CSI11 (78K0/KA2-L (25-pin and 32-pin products))**

| CSIE11 | TRMD11 | SSE11 | PM36 | P36 | PM37 | P37 | PM35 | P35 | PM02 | P02 | CSI11 Operation | Pin Function | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | SI11/P36 | SO11/P37 | $\overline{\text{SCK11}}$/P35 | $\overline{\text{SSI11}}$/P02/INTP5 |
| 0 | 0 | × | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | ×Note 1 | Stop | P36 | P37 | P35Note 2 | P02/INTP5 |
| 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | 1 | × | ×Note 1 | ×Note 1 | Slave reception Note 3 | SI11 | P37 | $\overline{\text{SCK11}}$ (input) Note 3 | P02/INTP5 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 1 | 0 | ×Note 1 | ×Note 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | Slave transmission Note 3 | P36 | SO11 | $\overline{\text{SCK11}}$ (input) Note 3 | P02/INTP5 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 1 | 0 | 1 | × | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | Slave transmission/ reception Note 3 | SI11 | SO11 | $\overline{\text{SCK11}}$ (input) Note 3 | P02/INTP5 |
| | | 1 | | | | | | | 1 | × | | | | | $\overline{\text{SSI11}}$ |
| 1 | 0 | 0 | 1 | × | ×Note 1 | ×Note 1 | 0 | 1 | ×Note 1 | ×Note 1 | Master reception | SI11 | P37 | $\overline{\text{SCK11}}$ (output) | P02/INTP5 |
| 1 | 1 | 0 | ×Note 1 | ×Note 1 | 0 | 0 | 0 | 1 | ×Note 1 | ×Note 1 | Master transmission | P36 | SO11 | $\overline{\text{SCK11}}$ (output) | P02/INTP5 |
| 1 | 1 | 0 | 1 | × | 0 | 0 | 0 | 1 | ×Note 1 | ×Note 1 | Master transmission/ reception | SI11 | SO11 | $\overline{\text{SCK11}}$ (output) | P02/INTP5 |

**Notes 1.** Can be set as port function.

**2.** To use P37/$\overline{\text{SCK11}}$ as port pins, clear CKP11 to 0.

**3.** To use the slave mode, set CKS112, CKS111, and CKS110 to 1, 1, 1.

**Remark** ×: don't care

CSIE11: Bit 7 of serial operation mode register 11 (CSIM11)

TRMD11: Bit 6 of CSIM11

CKP11: Bit 4 of serial clock selection register 11 (CSIC11)

CKS112, CKS111, CKS110: Bits 2 to 0 of CSIC11

PM×: Port mode register

P×: Port output latch

**(2) Communication operation**

In the 3-wire serial I/O mode, data is transmitted or received in 8-bit units. Each bit of the data is transmitted or received in synchronization with the serial clock.

Data can be transmitted or received if bit 6 (TRMD1n) of serial operation mode register 1n (CSIM1n) is 1. Transmission/reception is started when a value is written to transmit buffer register 1n (SOTB1n). In addition, data can be received when bit 6 (TRMD1n) of serial operation mode register 1n (CSIM1n) is 0.

Reception is started when data is read from serial I/O shift register 1n (SIO1n).

However, communication is performed as follows if bit 5 (SSE11) of CSIM11 is 1 when serial interface CSI11 is in the slave mode.

<1> Low level input to the $\overline{\text{SSI11}}$ pin
→ Transmission/reception is started when SOTB11 is written, or reception is started when SIO11 is read.

<2> High level input to the $\overline{\text{SSI11}}$ pin
→ Transmission/reception or reception is held, therefore, even if SOTB11 is written or SIO11 is read, transmission/reception or reception will not be started.

<3> Data is written to SOTB11 or data is read from SIO11 while a high level is input to the $\overline{\text{SSI11}}$ pin, then a low level is input to the $\overline{\text{SSI11}}$ pin
→ Transmission/reception or reception is started.

<4> A high level is input to the $\overline{\text{SSI11}}$ pin during transmission/reception or reception
→ Transmission/reception or reception is suspended.

After communication has been started, bit 0 (CSOT1n) of CSIM1n is set to 1. When communication of 8-bit data has been completed, a communication completion interrupt request flag (CSIIF1n) is set, and CSOT1n is cleared to 0. Then the next communication is enabled.

**Cautions 1. Do not access the control register and data register when CSOT1n = 1 (during serial communication).**

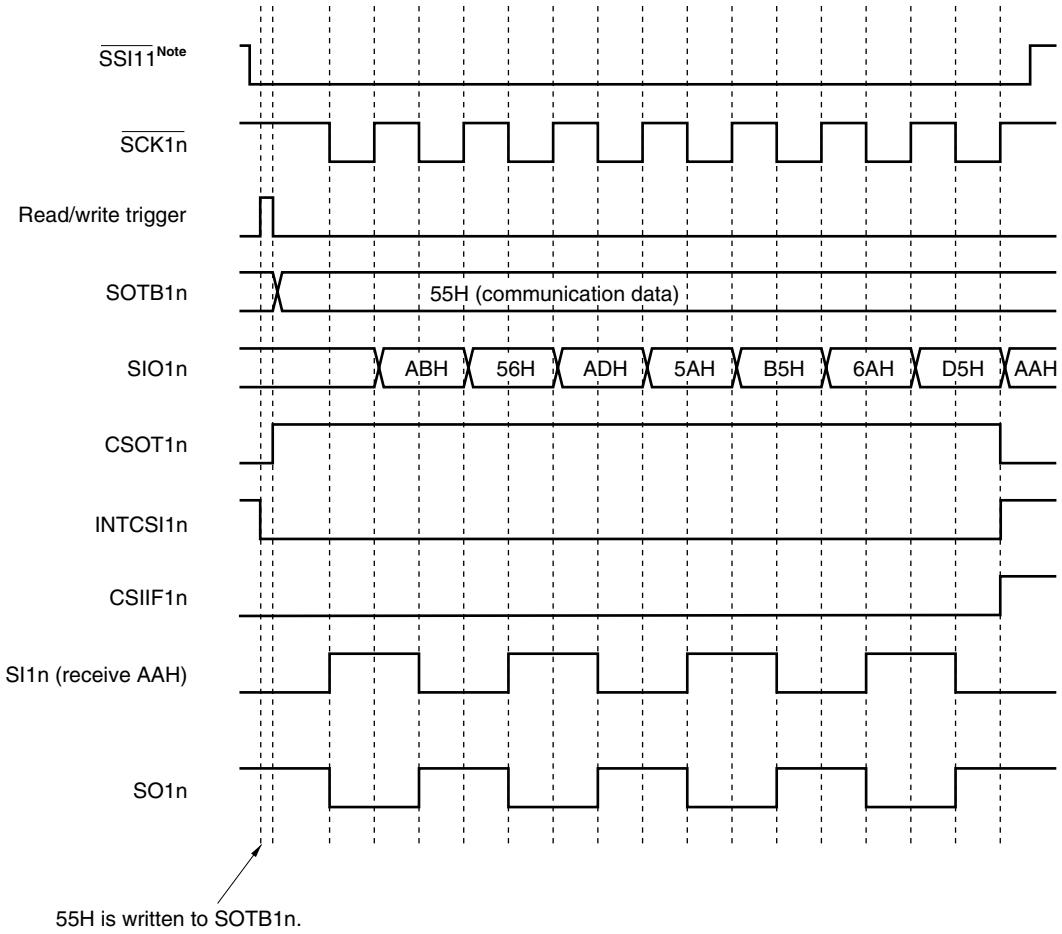
**2. When using serial interface CSI11, wait for the duration of at least one clock before the clock operation is started to change the level of the $\overline{\text{SSI11}}$ pin in the slave mode; otherwise, malfunctioning may occur.**

**Remarks 1.** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**2.** The $\overline{\text{SSI11}}$ pin is available only in 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L (48-pin products).

**Figure 16-15. Timing in 3-Wire Serial I/O Mode (1/2)**

**(a) Transmission/reception timing (Type 1: TRMD1n = 1, DIR1n = 0, CKP1n = 0, DAP1n = 0, SSE11 = 1[Note])**

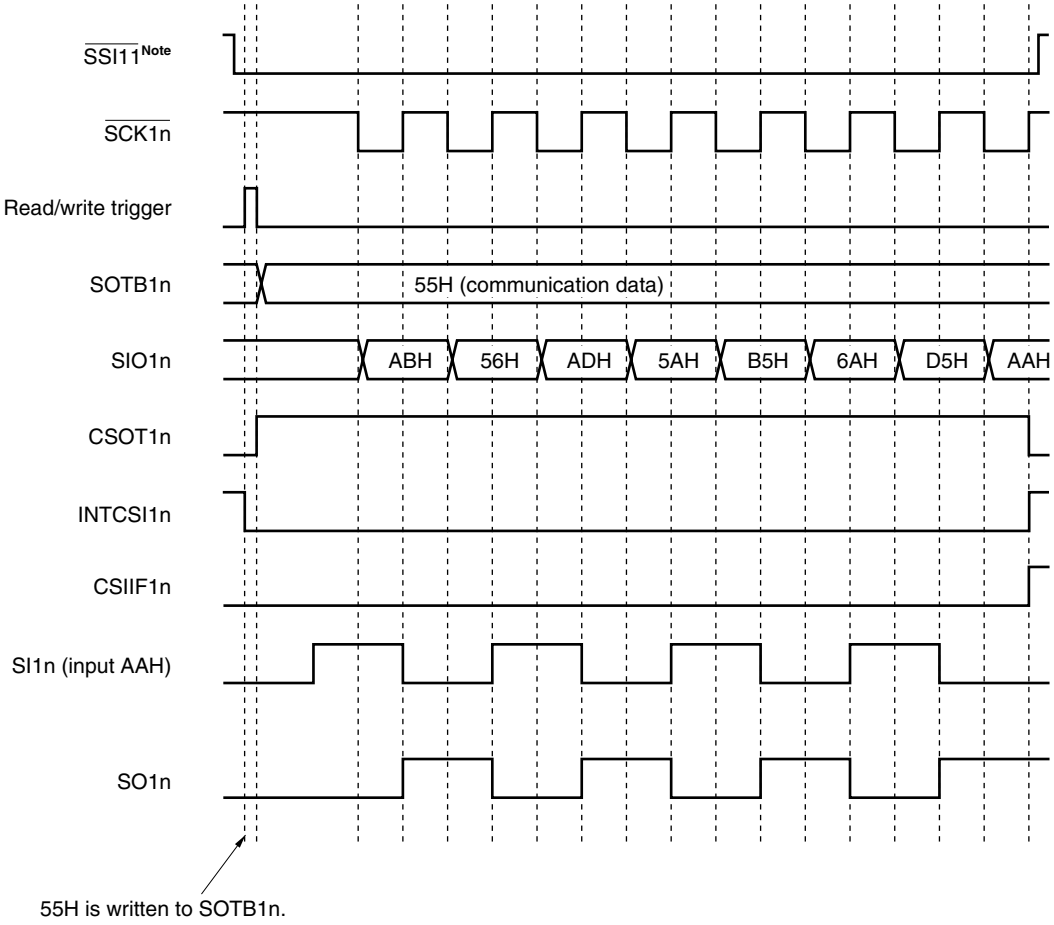



**Note** The SSE11 flag and $\overline{SSI11}$ pin are available only for serial interface CSI11 of 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L (48-pin products), and are used in the slave mode.

**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**Figure 16-15. Timing in 3-Wire Serial I/O Mode (2/2)**

**(b) Transmission/reception timing (Type 2: TRMD1n = 1, DIR1n = 0, CKP1n = 0, DAP1n = 1, SSE11 = 1[Note])**

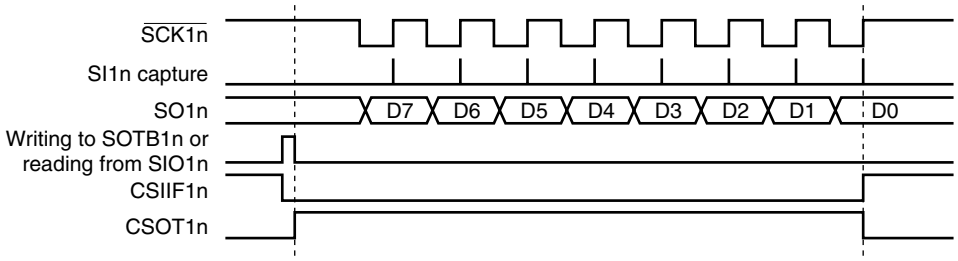



**Note** The SSE11 flag and $\overline{\text{SSI11}}$ pin are available only for serial interface CSI11 of 78K0/KA2-L (25, 32-pin products) and 78K0/KC2-L (48-pin products), and are used in the slave mode.

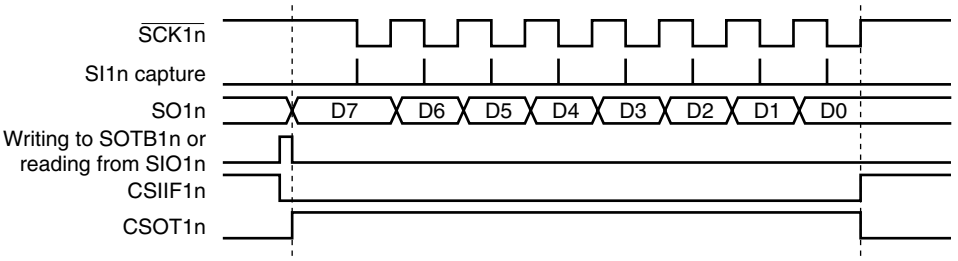
**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**Figure 16-16. Timing of Clock/Data Phase**

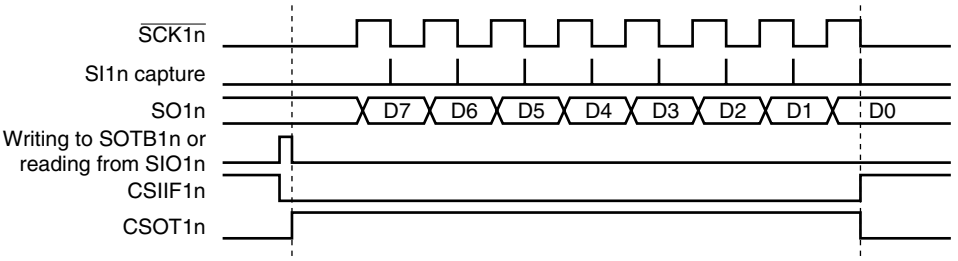
**(a) Type 1: CKP1n = 0, DAP1n = 0, DIR1n = 0**



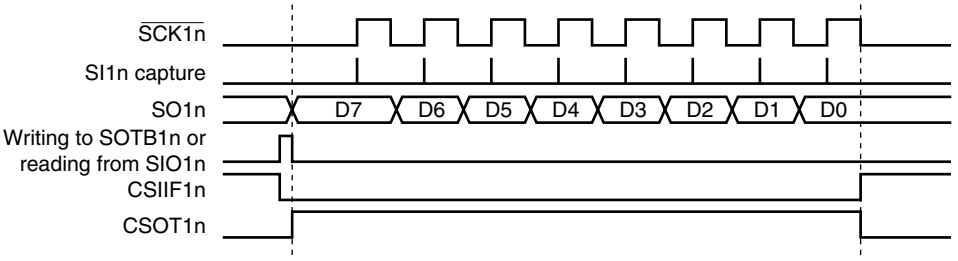

**(b) Type 2: CKP1n = 0, DAP1n = 1, DIR1n = 0**




**(c) Type 3: CKP1n = 1, DAP1n = 0, DIR1n = 0**




**(d) Type 4: CKP1n = 1, DAP1n = 1, DIR1n = 0**




**Remarks 1.** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

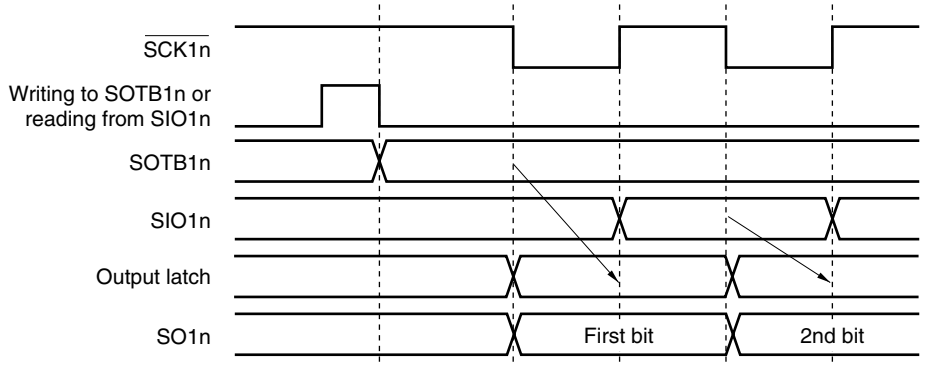
**2.** The above figure illustrates a communication operation where data is transmitted with the MSB first.

**(3) Timing of output to SO1n pin (first bit)**

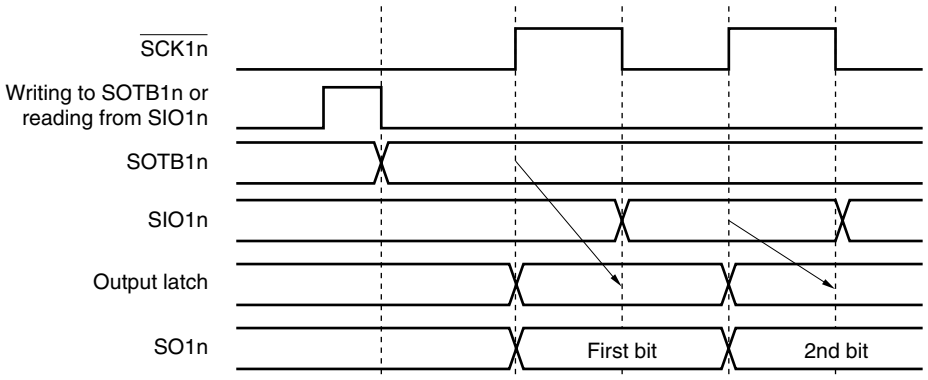
When communication is started, the value of transmit buffer register 1n (SOTB1n) is output from the SO1n pin. The output operation of the first bit at this time is described below.

**Figure 16-17. Output Operation of First Bit (1/2)**

**(a) Type 1: CKP1n = 0, DAP1n = 0**




**(b) Type 3: CKP1n = 1, DAP1n = 0**



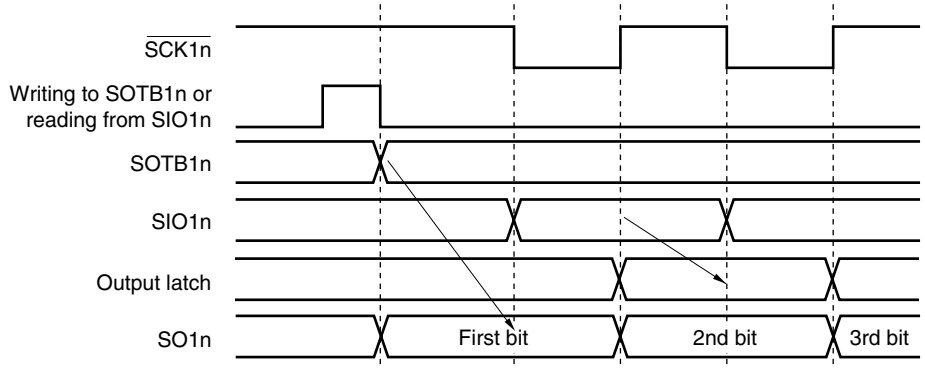

The first bit is directly latched by the SOTB1n register to the output latch at the falling (or rising) edge of $\overline{\text{SCK1n}}$, and output from the SO1n pin via an output selector. Then, the value of the SOTB1n register is transferred to the SIO1n register at the next rising (or falling) edge of $\overline{\text{SCK1n}}$, and shifted one bit. At the same time, the first bit of the receive data is stored in the SIO1n register via the SI1n pin.

The second and subsequent bits are latched by the SIO1n register to the output latch at the next falling (or rising) edge of $\overline{\text{SCK1n}}$, and the data is output from the SO1n pin.

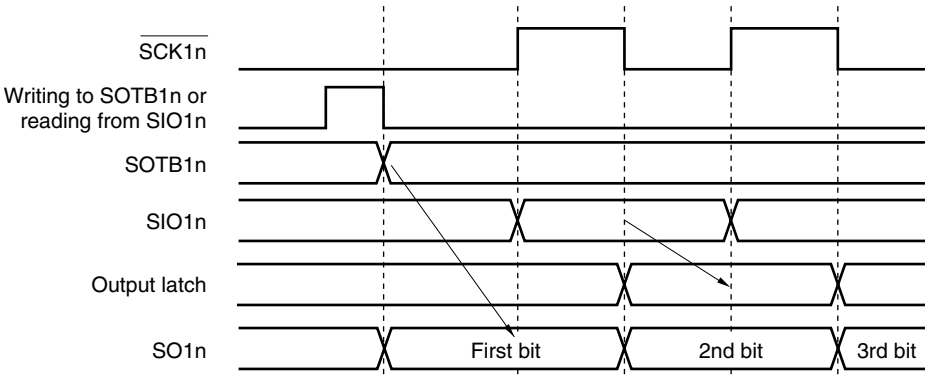
**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**Figure 16-17. Output Operation of First Bit (2/2)**

**(c) Type 2: CKP1n = 0, DAP1n = 1**




**(d) Type 4: CKP1n = 1, DAP1n = 1**




The first bit is directly latched by the SOTB1n register at the falling edge of the write signal of the SOTB1n register or the read signal of the SIO1n register, and output from the SO1n pin via an output selector. Then, the value of the SOTB1n register is transferred to the SIO1n register at the next falling (or rising) edge of $\overline{\text{SCK1n}}$, and shifted one bit. At the same time, the first bit of the receive data is stored in the SIO1n register via the SI1n pin.

The second and subsequent bits are latched by the SIO1n register to the output latch at the next rising (or falling) edge of $\overline{\text{SCK1n}}$, and the data is output from the SO1n pin.

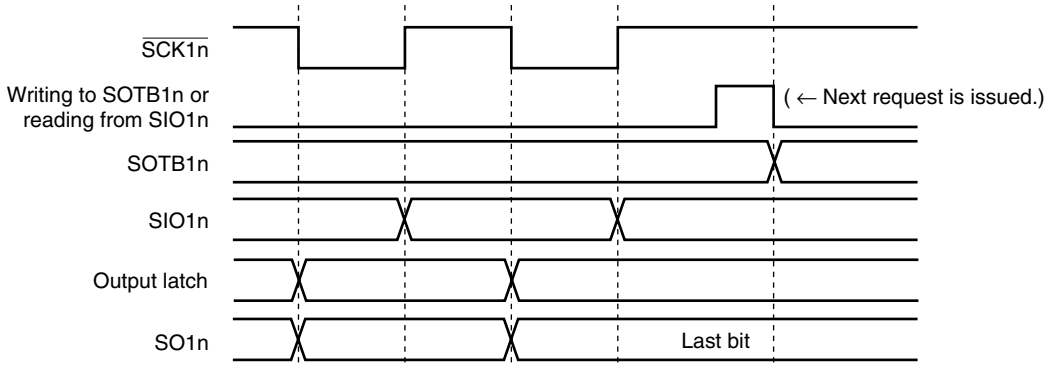
**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**(4) Output value of SO1n pin (last bit)**

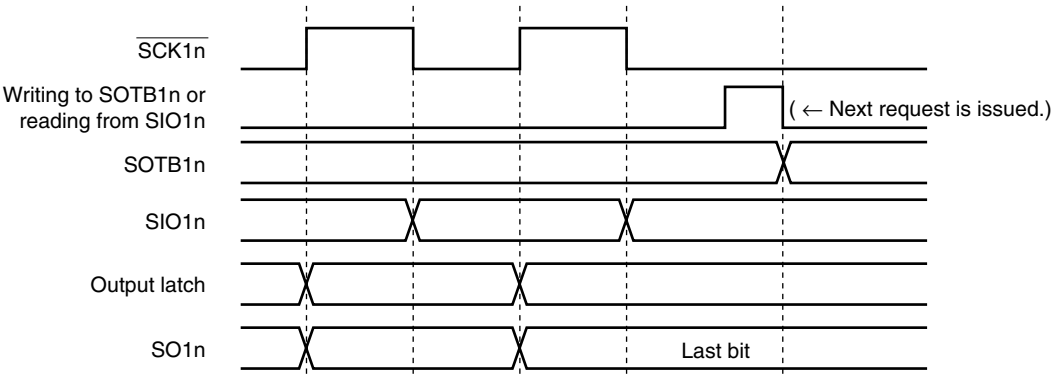
After communication has been completed, the SO1n pin holds the output value of the last bit.

**Figure 16-18. Output Value of SO1n Pin (Last Bit) (1/2)**

**(a) Type 1: CKP1n = 0, DAP1n = 0**



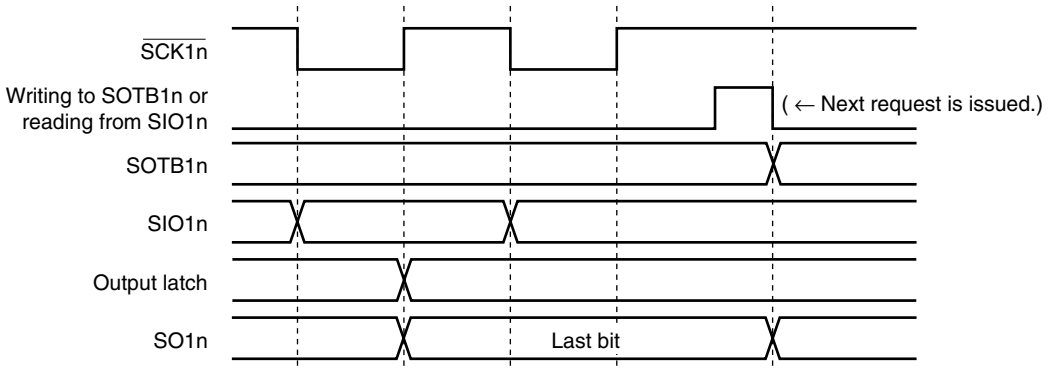

**(b) Type 3: CKP1n = 1, DAP1n = 0**



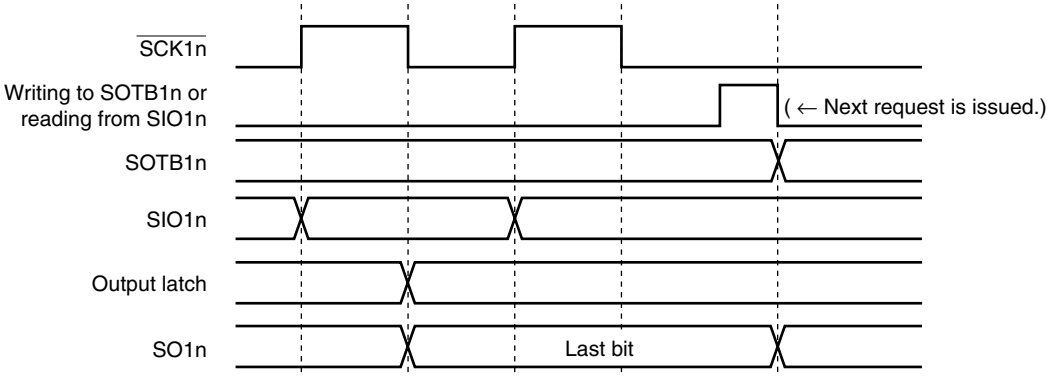

**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**Figure 16-18. Output Value of SO1n Pin (Last Bit) (2/2)**

**(c) Type 2: CKP1n = 0, DAP1n = 1**




**(d) Type 4: CKP1n = 1, DAP1n = 1**




**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

**(5) SO1n output (refer to Figures 16-1 to 16-3)**

The status of the SO1n output is as follows depending on the setting of CSIE1n, TRMD1n, DAP1n, and DIR1n.

**Table 16-3. SO1n Output Status**

| CSIE1n | TRMD1n | DAP1n | DIR1n | SO1n Output[Note 1] |
|---|---|---|---|---|
| CSIE1n = 0[Note 2] | TRMD1n = 0[Note 2] | – | – | Low level output[Note 2] |
| | TRMD1n = 1[Note 3] | DAP1n = 0 | – | Low level output |
| | | DAP1n = 1 | DIR1n = 0 | Value of bit 7 of SOTB1n |
| | | | DIR1n = 1 | Value of bit 0 of SOTB1n |
| CSIE1n = 1 | TRMD1n = 0 | – | – | Low level output |
| | TRMD1n = 1 | – | – | Transmission data[Note 4] |

**Notes 1.** The actual output of the SO10 or SO11 pin is determined according to the port mode register and the port register corresponding to SO10 or SO11, as well as the SO1n output.

**2.** This is a status after reset.

**3.** To use the SO11 pin as general-purpose port, set CSIC11 in the default status (00H).

**4.** After transmission has been completed, the SO1n pin holds the output value of the last bit of transmission data.

**Caution If a value is written to CSIE1n, TRMD1n, DAP1n, and DIR1n, the output value of SO1n changes.**

**Remark** 78K0/KA2-L (25, 32-pin products): n = 1
78K0/KB2-L: n = 0
78K0/KC2-L: n = 0, 1

# CHAPTER 17 INTERRUPT FUNCTIONS

<R>

| Item | | 78K0/KY2-L (μPD78F055x) | 78K0/KA2-L (μPD78F056x) | | 78K0/KB2-L (μPD78F057x) | 78K0/KC2-L (μPD78F058x) | | |
|---|---|---|---|---|---|---|---|---|
| | | 16 pins | 20 pins | 25, 32 pins | 30 pins | 40 pins | 44 pins | 48 pins |
| Maskable interrupts | External | 2 | 4 | 5 | 8 | 10 | 11 | 13 |
| | Internal | 10 | 10 | 11 | 13 | 16 | 16 | 16 |

## 17.1 Interrupt Function Types

The following two types of interrupt functions are used.

**(1) Maskable interrupts**

These interrupts undergo mask control. Maskable interrupts can be divided into a high interrupt priority group and a low interrupt priority group by setting the priority specification flag registers (PR0L, PR0H, PR1L, PR1H).

Multiple interrupt servicing can be applied to low-priority interrupts when high-priority interrupts are generated. If two or more interrupt requests, each having the same priority, are simultaneously generated, then they are processed according to the priority of vectored interrupt servicing. For the priority order, refer to **Table 17-1**.

A standby release signal is generated and STOP and HALT modes are released.

External interrupt requests and internal interrupt requests are provided as maskable interrupts.

**(2) Software interrupt**

This is a vectored interrupt generated by executing the BRK instruction. It is acknowledged even when interrupts are disabled. The software interrupt does not undergo interrupt priority control.

## 17.2 Interrupt Sources and Configuration

The interrupt sources consist of maskable interrupts and software interrupts. In addition, they also have up to four reset sources (refer to **Table 17-1**).

<R>

**Table 17-1. Interrupt Source List (1/2)**

| Interrupt Type | Internal/ External | Basic Configuration Type Note 1 | Default Priority Note 2 | Interrupt Source | | Vector Table Address | KY 2-L | KA2-L | | KB 2-L | KC2-L | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Name | Trigger | | 16 pins | 20 pins | 25, 32 pins | 30 pins | 40pins | 44 pins | 48 pins |
| Maskable | Internal | (A) | 0 | INTLVI | Low-voltage detection Note 3 | 0004H | √ | √ | √ | √ | √ | √ | √ |
| | External | (B) | 1 | INTP0 | Pin input edge detection | 0006H | √ | √ | √ | √ | √ | √ | √ |
| | | | 2 | INTP1 | | 0008H | √ | √ | – | √ | √ | √ | √ |
| | | | 3 | INTP2 | | 000AH | – | √ | √ | √ | √ | √ | √ |
| | | | 4 | INTP3 | | 000CH | – | √ | √ | √ | √ | √ | √ |
| | | | 5 | INTP4 | | 000EH | – | – | √ | √ | √ | √ | √ |
| | | | 6 | INTP5 | | 0010H | – | – | √ | √ | √ | √ | √ |
| | Internal | (A) | 7 | INTSRE6 | UART6 reception error generation | 0012H | √ | √ | √ | √ | √ | √ | √ |
| | | | 8 | INTSR6 | End of UART6 reception | 0014H | √ | √ | √ | √ | √ | √ | √ |
| | | | 9 | INTST6 | End of UART6 transmission | 0016H | √ | √ | √ | √ | √ | √ | √ |
| | | | 10 | INTCSI10 | End of CSI10 communication | 0018H | – | – | – | √ | √ | √ | √ |
| | | | | INTCSI11 | End of CSI11 communication | 0018H | – | – | √ | – | – | – | – |
| | | | 11 | INTTMH1 | Match between TMH1 and CMP01 (when compare register is specified) | 001AH | √ | √ | √ | √ | √ | √ | √ |
| | | | 12 | INTTMH0 | Match between TMH0 and CMP00 (when compare register is specified) | 001CH | – | – | – | √ | √ | √ | √ |
| | | | 13 | INTTM50 | Match between TM50 and CR50 (when compare register is specified) | 001EH | – | – | – | √ | √ | √ | √ |
| | | | 14 | INTTM000 | Match between TM00 and CR000 (when compare register is specified), TI010 pin valid edge detection (when capture register is specified) | 0020H | √ | √ | √ | √ | √ | √ | √ |
| | | | 15 | INTTM010 | Match between TM00 and CR010 (when compare register is specified), TI000 pin valid edge detection (when capture register is specified) | 0022H | √ | √ | √ | √ | √ | √ | √ |
| | | | 16 | INTAD | End of A/D conversion | 0024H | √ | √ | √ | √ | √ | √ | √ |
| | External | (B) | 17 | INTP6 | Pin input edge detection | 0026H | – | – | – | – | – | – | √ |
| | Internal | (A) | 18 | INTRTCI | Interval signal detection of real-time counter | 0028H | – | – | – | – | √ | √ | √ |
| | | | 19 | INTTM51 Note 4 | Match between TM51 and CR51 (when compare register is specified) | 002AH | √ | √ | √ | √ | √ | √ | √ |

**Notes 1.** Basic configuration types (A) to (D) correspond to (A) to (D) in **Figure 17-1**.

**2.** The default priority determines the sequence of processing vectored interrupts if two or more maskable interrupts occur simultaneously. Zero indicates the highest priority and 28 indicates the lowest priority.

**3.** When bit 1 (LVIMD) of the low-voltage detection register (LVIM) is cleared to 0.

**4.** When 8-bit timer/event counter 51 is used in the carrier generator mode, an interrupt is generated upon the timing when the INTTM5H1 signal is generated (refer to **Figure 8-14 Transfer Timing**).

<R>

**Table 17-1. Interrupt Source List (2/2)**

| Interrupt Type | Internal/ External | Basic Configuration Type Note 1 | Default Priority Note 2 | Interrupt Source | | Vector Table Address | KY2-L | KA2-L | | KB2-L | KC2-L | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Name | Trigger | | 16 pins | 20 pins | 25, 32 pins | 30 pins | 40 pins | 44 pins | 48 pins |
| Maskable | External | (C) | 20 | INTKR | Key interrupt detection | 002CH | – | – | – | – | √ | √ | √ |
| | Internal | (A) | 21 | INTRTC | Fixed-cycle signal of real-time counter/alarm match detection | 002EH | – | – | – | – | √ | √ | √ |
| | External | (B) | 22 | INTP7 | Pin input edge detection | 0030H | – | – | – | – | – | – | √ |
| | | | 23 | INTP8 | | 0032H | – | – | – | – | – | √ | √ |
| | Internal | (A) | 24 | INTIICA0 | End of IICA communication | 0034H | √ | √ | √ | √ | √ | √ | √ |
| | | | 25 | INTCSI11 | End of CSI11 communication | 0036H | – | – | – | – | √ | √ | √ |
| | External | (B) | 26 | INTP9 | Pin input edge detection | 0038H | – | – | – | – | √ | √ | √ |
| | | | 27 | INTP10 | | 003AH | – | – | – | √ | √ | √ | √ |
| | | | 28 | INTP11 | | 003CH | – | – | – | √ | √ | √ | √ |
| Software | – | (D) | – | BRK | BRK instruction execution | 003EH | √ | √ | √ | √ | √ | √ | √ |
| Reset | – | – | – | RESET | Reset input | 0000H | √ | √ | √ | √ | √ | √ | √ |
| | | | | POC | Power-on clear | | | | | | | | |
| | | | | LVI | Low-voltage detection Note 3 | | | | | | | | |
| | | | | WDT | WDT overflow | | | | | | | | |

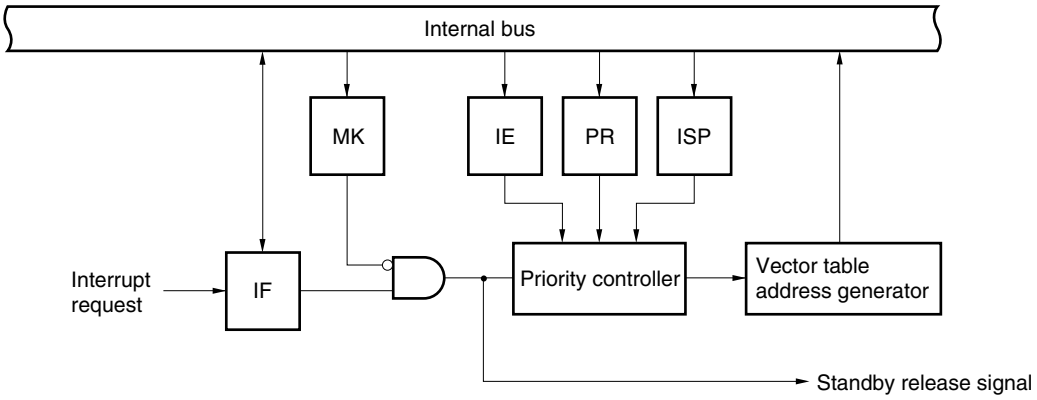
**Notes 1.** Basic configuration types (A) to (D) correspond to (A) to (D) in **Figure 17-1**.

**2.** The default priority determines the sequence of processing vectored interrupts if two or more maskable interrupts occur simultaneously. Zero indicates the highest priority and 28 indicates the lowest priority.

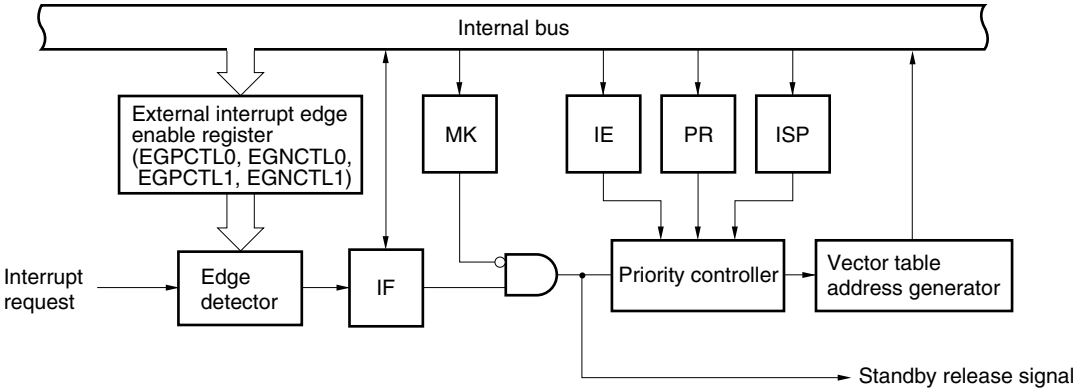
**3.** When bit 1 (LVIMD) of the low-voltage detection register (LVIM) is set to 1.

**Figure 17-1. Basic Configuration of Interrupt Function (1/2)**

**(A) Internal maskable interrupt**




**(B) External maskable interrupt (INTPn)**

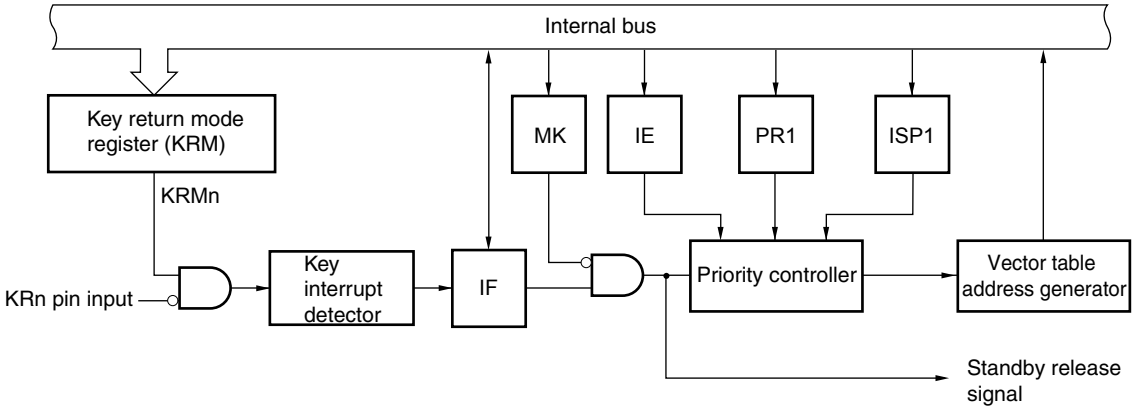



**Remark** n = 0, 1: 78K0/KY2-L
n = 0 to 3: 20-pin products of 78K0/KA2-L
<R> n = 0, 2 to 5: 25-pin products of 78K0/KA2-L
n = 0 to 5, 10, 11: 78K0/KB2-L
<R> n = 0 to 5, 9 to 11: 40-pin products of 78K0/KC2-L
n = 0 to 5, 8 to 11: 44-pin products of 78K0/KC2-L
n = 0 to 11: 48-pin products of 78K0/KC2-L

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specification flag

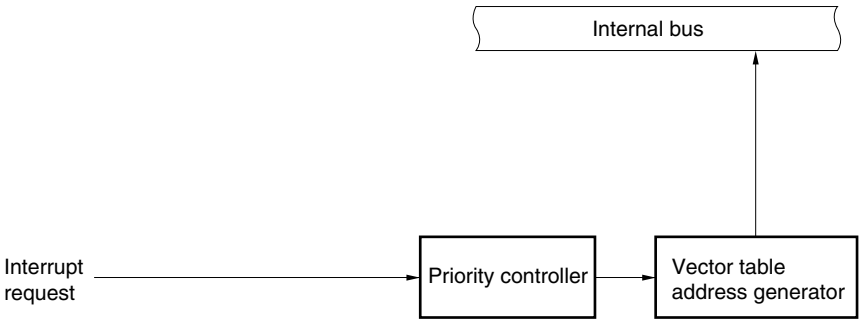
**Figure 17-1. Basic Configuration of Interrupt Function (2/2)**

**(C) External maskable interrupt (INTKR)**




<R> **Remark** n = 0 to 3: 40-pin and 44-pin products of 78K0/KC2-L
n = 0 to 5: 48-pin products of 78K0/KC2-L

**(D) Software interrupt**




IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specification flag
KRM: Key return mode register

## 17.3 Registers Controlling Interrupt Functions

The following 6 types of registers are used to control the interrupt functions.

- Interrupt request flag registers (IF0L, IF0H, IF1L, IF1H)
- Interrupt mask flag registers (MK0L, MK0H, MK1L, MK1H)
- Priority specification flag registers (PR0L, PR0H, PR1L, PR1H)
- External interrupt rising edge enable registers (EGPCTL0, EGPCTL1)
- External interrupt falling edge enable registers (EGNCTL0, EGNCTL1)
- Program status word (PSW)

Table 17-2 shows a list of interrupt request flags, interrupt mask flags, and priority specification flags corresponding to interrupt request sources.

<R> **Table 17-2. Flags Corresponding to Interrupt Request Sources (1/2)**

| KY 2-L | KA 2-L | | KB 2-L | KC2-L | | | Interrupt Source | Interrupt Request Flag | | Interrupt Mask Flag | | Priority Specification Flag | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | Register | | Register | | Register |
| 16 pins | 20 pins | 25, 32 pins | 30 pins | 40 pins | 44 pins | 48 pins | | | | | | | |
| √ | √ | √ | √ | √ | √ | √ | INTLVI | LVIIF | IF0L | LVIMK | MK0L | LVIPR | PR0L |
| √ | √ | √ | √ | √ | √ | √ | INTP0 | PIF0 | | PMK0 | | PPR0 | |
| √ | √ | – | √ | √ | √ | √ | INTP1 | PIF1 | | PMK1 | | PPR1 | |
| – | √ | √ | √ | √ | √ | √ | INTP2 | PIF2 | | PMK2 | | PPR2 | |
| – | √ | √ | √ | √ | √ | √ | INTP3 | PIF3 | | PMK3 | | PPR3 | |
| – | – | √ | √ | √ | √ | √ | INTP4 | PIF4 | | PMK4 | | PPR4 | |
| – | – | √ | √ | √ | √ | √ | INTP5 | PIF5 | | PMK5 | | PPR5 | |
| √ | √ | √ | √ | √ | √ | √ | INTSRE6 | SREIF6 | | SREMK6 | | SREPR6 | |
| √ | √ | √ | √ | √ | √ | √ | INTSR6 | SRIF6 | IF0H | SRMK6 | MK0H | SRPR6 | PR0H |
| √ | √ | √ | √ | √ | √ | √ | INTST6 | STIF6 | | STMK6 | | STPR6 | |
| – | – | – | √ | √ | √ | √ | INTCSI10 | CSIIF10 | | CSIMK10 | | CSIPR10 | |
| – | – | √ | – | – | – | – | INTCSI11 | CSIIF11 | | CSIMK11 | | CSIPR11 | |
| √ | √ | √ | √ | √ | √ | √ | INTTMH1 | TMIFH1 | | TMMKH1 | | TMPRH1 | |
| – | – | – | √ | √ | √ | √ | INTTMH0 | TMIFH0 | | TMMKH0 | | TMPRH0 | |
| – | – | – | √ | √ | √ | √ | INTTM50 | TMIF50 | | TMMK50 | | TMPR50 | |
| √ | √ | √ | √ | √ | √ | √ | INTTM000 | TMIF000 | | TMMK000 | | TMPR000 | |
| √ | √ | √ | √ | √ | √ | √ | INTTM010 | TMIF010 | | TMMK010 | | TMPR010 | |

<R> **Table 17-2. Flags Corresponding to Interrupt Request Sources (2/2)**

| KY 2-L | KA 2-L | | KB 2-L | KC2-L | | | Interrupt Source | Interrupt Request Flag | | Interrupt Mask Flag | | Priority Specification Flag | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | Register | | Register | | Register |
| 16 pins | 20 pins | 25, 32 pins | 30 pins | 40 pins | 44 pins | 48 pins | | | | | | | |
| √ | √ | √ | √ | √ | √ | √ | INTAD | ADIF | IF1L | ADMK | MK1L | ADPR | PR1L |
| – | – | – | – | – | – | √ | INTP6 | PIF6 | | PMK6 | | PPR6 | |
| – | – | – | – | √ | √ | √ | INTRTCI | RTCIIF | | RTCIMK | | RTCIPR | |
| √ | √ | √ | √ | √ | √ | √ | INTTM51[Note] | TMIF51 | | TMMK51 | | TMPR51 | |
| – | – | – | – | √ | √ | √ | INTKR | KRIF | | KRMK | | KRPR | |
| – | – | – | – | √ | √ | √ | INTRTC | RTCIF | | RTCMK | | RTCPR | |
| – | – | – | – | – | – | √ | INTP7 | PIF7 | | PMK7 | | PPR7 | |
| – | – | – | – | – | √ | √ | INTP8 | PIF8 | | PMK8 | | PPR8 | |
| √ | √ | √ | √ | √ | √ | √ | INTIICA0 | IICAIF0 | IF1H | IICAMK0 | MK1H | IICAPR0 | PR1H |
| – | – | – | – | – | √ | √ | INTCSI11 | CSIIF11 | | CSIMK11 | | CSIPR11 | |
| – | – | – | – | √ | √ | √ | INTP9 | PIF9 | | PMK9 | | PPR9 | |
| – | – | – | √ | √ | √ | √ | INTP10 | PIF10 | | PMK10 | | PPR10 | |
| – | – | – | √ | √ | √ | √ | INTP11 | PIF11 | | PMK11 | | PPR11 | |

**Note** When 8-bit timer/event counter 51 is used in the carrier generator mode, an interrupt is generated upon the timing when the INTTM5H1 signal is generated (refer to **Figure 8-16 Transfer Timing**).

**(1) Interrupt request flag registers (IF0L, IF0H, IF1L, IF1H)**

The interrupt request flags are set to 1 when the corresponding interrupt request is generated or an instruction is executed. They are cleared to 0 when an instruction is executed upon acknowledgment of an interrupt request or upon reset signal generation.

When an interrupt is acknowledged, the interrupt request flag is automatically cleared and then the interrupt routine is entered.

IF0L, IF0H, IF1L, and IF1H are set by a 1-bit or 8-bit memory manipulation instruction. When IF0L and IF0H, and IF1L and IF1H are combined to form 16-bit registers IF0 and IF1, they are set by a 16-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Cautions 1. When operating a timer, serial interface, or A/D converter after standby release, operate it once after clearing the interrupt request flag. An interrupt request flag may be set by noise.**

**2. When manipulating a flag of the interrupt request flag register, use a 1-bit memory manipulation instruction (CLR1). When describing in C language, use a bit manipulation instruction such as "IF0L.0 = 0;" or "_asm("clr1 IF0L, 0");" because the compiled assembler must be a 1-bit memory manipulation instruction (CLR1).**

**If a program is described in C language using an 8-bit memory manipulation instruction such as "IF0L &= 0xfe;" and compiled, it becomes the assembler of three instructions.**

```
mov a, IF0L
and a, #0FEH
mov IF0L, a
```

**In this case, even if the request flag of another bit of the same interrupt request flag register (IF0L) is set to 1 at the timing between "mov a, IF0L" and "mov IF0L, a", the flag is cleared to 0 at "mov IF0L, a". Therefore, care must be exercised when using an 8-bit memory manipulation instruction in C language.**

**Figure 17-2. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (78K0/KY2-L)**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | 0 | 0 | 0 | 0 | PIF1 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | 0 | 0 | TMIFH1 | 0 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | 0 | 0 | 0 | 0 | TMIF51 | 0 | 0 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bits 3 to 6 of IF0L, bits 2, 4 and 5 of IF0H, bits 1, 2, 4 to 7 of IF1L, and bits 1 to 7 of IF1H to 0.**

**Figure 17-3. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (78K0/KA2-L (20-pin products))**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | 0 | 0 | PIF3 | PIF2 | PIF1 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | 0 | 0 | TMIFH1 | 0 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | 0 | 0 | 0 | 0 | TMIF51 | 0 | 0 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bits 5 and 6 of IF0L, bits 2, 4 and 5 of IF0H, bits 1, 2, 4 to 7 of IF1L, and bits 1 to 7 of IF1H to 0.**

<R> **Figure 17-4. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (78K0/KA2-L (25-pin and 32-pin products))**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | PIF5 | PIF4 | PIF3 | PIF2 | 0 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | 0 | 0 | TMIFH1 | CSIIF11 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | 0 | 0 | 0 | 0 | TMIF51 | 0 | 0 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bit 2 of IF0L, bits 4 and 5 of IF0H, bits 1, 2, 4 to 7 of IF1L, and bits 1 to 7 of IF1H to 0.**

**Figure 17-5. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (78K0/KB2-L)**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | PIF5 | PIF4 | PIF3 | PIF2 | PIF1 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | TMIF50 | TMIFH0 | TMIFH1 | CSIIF10 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | 0 | 0 | 0 | 0 | TMIF51 | 0 | 0 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | PIF11 | PIF10 | 0 | 0 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bits 1, 2, 4 to 7 of IF1L and bits 1, 2, 5 to 7 of IF1H to 0.**

<R> **Figure 17-6. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (40-pin products of 78K0/KC2-L)**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | PIF5 | PIF4 | PIF3 | PIF2 | PIF1 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | TMIF50 | TMIFH0 | TMIFH1 | CSIIF10 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | 7 | 6 | <5> | <4> | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | 0 | 0 | RTCIF | KRIF | TMIF51 | RTCIIF | 0 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | PIF11 | PIF10 | PIF9 | CSIIF11 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bits 1, 6, and 7 of IF1L, and bits 5 to 7 of IF1H to 0.**

**Figure 17-7. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H)**
**(44-pin products of 78K0/KC2-L)**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | PIF5 | PIF4 | PIF3 | PIF2 | PIF1 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | TMIF50 | TMIFH0 | TMIFH1 | CSIIF10 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | <7> | 6 | <5> | <4> | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | PIF8 | 0 | RTCIF | KRIF | TMIF51 | RTCIIF | 0 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | PIF11 | PIF10 | PIF9 | CSIIF11 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bits 1 and 6 of IF1L, and bits 5 to 7 of IF1H to 0.**

**Figure 17-8. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H)**
**(48-pin products of 78K0/KC2-L)**

Address: FFE0H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0L | SREIF6 | PIF5 | PIF4 | PIF3 | PIF2 | PIF1 | PIF0 | LVIIF |

Address: FFE1H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF0H | TMIF010 | TMIF000 | TMIF50 | TMIFH0 | TMIFH1 | CSIIF10 | STIF6 | SRIF6 |

Address: FFE2H After reset: 00H R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1L | PIF8 | PIF7 | RTCIF | KRIF | TMIF51 | RTCIIF | PIF6 | ADIF |

Address: FFE3H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| IF1H | 0 | 0 | 0 | PIF11 | PIF10 | PIF9 | CSIIF11 | IICAIF0 |

| XXIFX | Interrupt request flag |
|---|---|
| 0 | No interrupt request signal is generated |
| 1 | Interrupt request is generated, interrupt request status |

**Caution Be sure to clear bits 5 to 7 of IF1H to 0.**

**(2) Interrupt mask flag registers (MK0L, MK0H, MK1L, MK1H)**

The interrupt mask flags are used to enable/disable the corresponding maskable interrupt servicing.

MK0L, MK0H, MK1L, and MK1H are set by a 1-bit or 8-bit memory manipulation instruction. When MK0L and MK0H, and MK1L and MK1H are combined to form 16-bit registers MK0 and MK1, they are set by a 16-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

**Figure 17-9. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (78K0/KY2-L)**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | 1 | 1 | 1 | 1 | PMK1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | 1 | 1 | TMMKH1 | 1 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | 1 | 1 | 1 | 1 | TMMK51 | 1 | 1 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bits 3 to 6 of MK0L, bits 2, 4 and 5 of MK0H, bits 1, 2, 4 to 7 of MK1L, and bits 1 to 7 of MK1H to 1.**

**Figure 17-10. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (78K0/KA2-L (20-pin products))**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | 1 | 1 | PMK3 | PMK2 | PMK1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | 1 | 1 | TMMKH1 | 1 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | 1 | 1 | 1 | 1 | TMMK51 | 1 | 1 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bits 5 and 6 of MK0L, bits 2, 4 and 5 of MK0H, bits 1, 2, 4 to 7 of MK1L, and bits 1 to 7 of MK1H to 1.**

<R>

**Figure 17-11. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (78K0/KA2-L (25-pin and 32-pin products))**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | PMK5 | PMK4 | PMK3 | PMK2 | 1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | 1 | 1 | TMMKH1 | CSIMK11 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | 1 | 1 | 1 | 1 | TMMK51 | 1 | 1 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bit 2 of MK0L, bits 4 and 5 of MK0H, bits 1, 2, 4 to 7 of MK1L, and bits 1 to 7 of MK1H to 1.**

**Figure 17-12. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (78K0/KB2-L)**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | PMK5 | PMK4 | PMK3 | PMK2 | PMK1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | TMMK50 | TMMKH0 | TMMKH1 | CSIMK10 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | 1 | 1 | 1 | 1 | TMMK51 | 1 | 1 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | PMK11 | PMK10 | 1 | 1 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bits 1, 2, 4 to 7 of MK1L, and bits 1, 2, 5 to 7 of MK1H to 1.**

<R> **Figure 17-13. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (40-pin products of 78K0/KC2-L)**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | PMK5 | PMK4 | PMK3 | PMK2 | PMK1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | TMMK50 | TMMKH0 | TMMKH1 | CSIMK10 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | 7 | 6 | <5> | <4> | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | 1 | 1 | RTCMK | KRMK | TMMK51 | RTCIMK | 1 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | PMK11 | PMK10 | PMK9 | CSIMK11 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bits 1, 6, and 7 of MK1L, and bits 5 to 7 of MK1H to 1.**

**Figure 17-14. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H)**
**(44-pin products of 78K0/KC2-L)**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | PMK5 | PMK4 | PMK3 | PMK2 | PMK1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | TMMK50 | TMMKH0 | TMMKH1 | CSIMK10 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | <7> | 6 | <5> | <4> | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | PMK8 | 1 | RTCMK | KRMK | TMMK51 | RTCIMK | 1 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | PMK11 | PMK10 | PMK9 | CSIMK11 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bits 1 and 6 of MK1L, and bits 5 to 7 of MK1H to 1.**

**Figure 17-15. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H)**
**(48-pin products of 78K0/KC2-L)**

Address: FFE4H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0L | SREMK6 | PMK5 | PMK4 | PMK3 | PMK2 | PMK1 | PMK0 | LVIMK |

Address: FFE5H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK0H | TMMK010 | TMMK000 | TMMK50 | TMMKH0 | TMMKH1 | CSIMK10 | STMK6 | SRMK6 |

Address: FFE6H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1L | PMK8 | PMK7 | RTCMK | KRMK | TMMK51 | RTCIMK | PMK6 | ADMK |

Address: FFE7H After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| MK1H | 1 | 1 | 1 | PMK11 | PMK10 | PMK9 | CSIMK11 | IICAMK0 |

| XXMKX | Interrupt servicing control |
|---|---|
| 0 | Interrupt servicing enabled |
| 1 | Interrupt servicing disabled |

**Caution Be sure to set bits 5 to 7 of MK1H to 1.**

**(3) Priority specification flag registers (PR0L, PR0H, PR1L, PR1H)**

The priority specification flag registers are used to set the corresponding maskable interrupt priority order.

PR0L, PR0H, PR1L, and PR1H are set by a 1-bit or 8-bit memory manipulation instruction. If PR0L and PR0H, and PR1L and PR1H are combined to form 16-bit registers PR0 and PR1, they are set by a 16-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

**Figure 17-16. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (78K0/KY2-L)**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | 6 | 5 | 4 | 3 | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | 1 | 1 | 1 | 1 | PPR1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | 1 | 1 | TMPRH1 | 1 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | 1 | 1 | 1 | 1 | TMPR51 | 1 | 1 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bits 3 to 6 of PR0L, bits 2, 4 and 5 of PR0H, bits 1, 2, 4 to 7 of PR1L, and bits 1 to 7 of PR1H to 1.**

**Figure 17-17. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (78K0/KA2-L (20-pin products))**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | 1 | 1 | PPR3 | PPR2 | PPR1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | 1 | 1 | TMPRH1 | 1 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | 1 | 1 | 1 | 1 | TMPR51 | 1 | 1 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bits 5 and 6 of PR0L, bits 2, 4 and 5 of PR0H, bits 1, 2, 4 to 7 of PR1L, and bits 1 to 7 of PR1H to 1.**

<R> **Figure 17-18. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (78K0/KA2-L (25-pin and 32-pin products))**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | 2 | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | PPR5 | PPR4 | PPR3 | PPR2 | 1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | 5 | 4 | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | 1 | 1 | TMPRH1 | CSIPR11 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | 1 | 1 | 1 | 1 | TMPR51 | 1 | 1 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | 1 | 1 | 1 | 1 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bit 2 of PR0L, bits 4 and 5 of PR0H, bits 1, 2, 4 to 7 of PR1L, and bits 1 to 7 of PR1H to 1.**

**Figure 17-19. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (78K0/KB2-L)**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | PPR5 | PPR4 | PPR3 | PPR2 | PPR1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | TMPR50 | TMPRH0 | TMPRH1 | CSIPR10 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | 1 | 1 | 1 | 1 | TMPR51 | 1 | 1 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | PPR11 | PPR10 | 1 | 1 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bits 1, 2, 4 to 7 of PR1L, and bits 1, 2, 5 to 7 of PR1H to 1.**

<R> **Figure 17-20. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (40-pin products of 78K0/KC2-L)**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | PPR5 | PPR4 | PPR3 | PPR2 | PPR1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | TMPR50 | TMPRH0 | TMPRH1 | CSIPR10 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | 7 | 6 | <5> | <4> | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | 1 | 1 | RTCPR | KRPR | TMPR51 | RTCIPR | 1 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | PPR11 | PPR10 | PPR9 | CSIPR11 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bits 1, 6, and 7 of PR1L, and bits 5 to 7 of PR1H to 1.**

**Figure 17-21. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H)**
**(44-pin products of 78K0/KC2-L)**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | PPR5 | PPR4 | PPR3 | PPR2 | PPR1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | TMPR50 | TMPRH0 | TMPRH1 | CSIPR10 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | <7> | 6 | <5> | <4> | <3> | <2> | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | PPR8 | 1 | RTCPR | KRPR | TMPR51 | RTCIPR | 1 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | PPR11 | PPR10 | PPR9 | CSIPR11 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bits 1 and 6 of PR1L, and bits 5 to 7 of PR1H to 1.**

**Figure 17-22. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (48-pin products of 78K0/KC2-L)**

Address: FFE8H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0L | SREPR6 | PPR5 | PPR4 | PPR3 | PPR2 | PPR1 | PPR0 | LVIPR |

Address: FFE9H After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR0H | TMPR010 | TMPR000 | TMPR50 | TMPRH0 | TMPRH1 | CSIPR10 | STPR6 | SRPR6 |

Address: FFEAH After reset: FFH R/W

| Symbol | <7> | <6> | <5> | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1L | PPR8 | PPR7 | RTCPR | KRPR | TMPR51 | RTCIPR | PPR6 | ADPR |

Address: FFEBH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | <4> | <3> | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| PR1H | 1 | 1 | 1 | PPR11 | PPR10 | PPR9 | CSIPR11 | IICAPR0 |

| XXPRX | Priority level selection |
|---|---|
| 0 | High priority level |
| 1 | Low priority level |

**Caution Be sure to set bits 5 to 7 of PR1H to 1.**

**(4) External interrupt rising edge enable registers (EGPCTL0, EGPCTL1), external interrupt falling edge enable registers (EGNCTL0, EGNCTL1)**

These registers specify the valid edge for INTPn.

EGPCTL0, EGPCTL1, EGNCTL0, and EGNCTL1 are set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Remark** n = 0, 1: 78K0/KY2-L
n = 0 to 3: 20-pin products of 78K0/KA2-L
<R> n = 0, 2 to 5: 25, 32-pin products of 78K0/KA2-L
n = 0 to 5, 10, 11: 78K0/KB2-L
<R> n = 0 to 5, 9 to 11: 40-pin products of 78K0/KC2-L
n = 0 to 5, 8 to 11: 44-pin products of 78K0/KC2-L
n = 0 to 11: 48-pin products of 78K0/KC2-L

**Figure 17-23. Format of External Interrupt Rising Edge Enable Registers (EGPCTL0, EGPCTL1) and External Interrupt Falling Edge Enable Registers (EGNCTL0, EGNCTL1) (1/5)**

**(a) 78K0/KY2-L**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | 0 | 0 | 0 | 0 | 0 | 0 | EGP1 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | 0 | 0 | 0 | 0 | 0 | 0 | EGN1 | EGN0 |

**(b) 78K0/KA2-L (20-pin products)**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | 0 | 0 | 0 | 0 | EGP3 | EGP2 | EGP1 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | 0 | 0 | 0 | 0 | EGN3 | EGN2 | EGN1 | EGN0 |

<R>**(c) 78K0/KA2-L (25-pin and 32-pin products)**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | 0 | 0 | EGP5 | EGP4 | EGP3 | EGP2 | 0 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | 0 | 0 | EGN5 | EGN4 | EGN3 | EGN2 | 0 | EGN0 |

| EGPn | EGNn | INTPn pin valid edge selection |
|---|---|---|
| 0 | 0 | Edge detection disabled |
| 0 | 1 | Falling edge |
| 1 | 0 | Rising edge |
| 1 | 1 | Both rising and falling edges |

**Caution Be sure to clear bits 2 to 7 of EGPCTL0 and EGNCTL0 to 0 in 78K0/KY2-L.**
**Be sure to clear bits 4 to 7 of EGPCTL0 and EGNCTL0 to 0 in 78K0/KA2-L (20-pin products).**
**Be sure to clear bits 1, 6, and 7 of EGPCTL0 and EGNCTL0 to 0 in 78K0/KA2-L (25-pin and 32-pin product).**

**Remark** n = 0, 1: 78K0/KY2-L
n = 0 to 3: 78K0/KA2-L

**Figure 17-23. Format of External Interrupt Rising Edge Enable Registers (EGPCTL0, EGPCTL1) and External Interrupt Falling Edge Enable Registers (EGNCTL0, EGNCTL1) (2/5)**

**(d) 78K0/KB2-L**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | 0 | 0 | EGP5 | EGP4 | EGP3 | EGP2 | EGP1 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | 0 | 0 | EGN5 | EGN4 | EGN3 | EGN2 | EGN1 | EGN0 |

Address: FF4AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL1 | 0 | 0 | 0 | 0 | EGP11 | EGP10 | 0 | 0 |

Address: FF4BH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL1 | 0 | 0 | 0 | 0 | EGN11 | EGN10 | 0 | 0 |

| EGPn | EGNn | INTPn pin valid edge selection |
|---|---|---|
| 0 | 0 | Edge detection disabled |
| 0 | 1 | Falling edge |
| 1 | 0 | Rising edge |
| 1 | 1 | Both rising and falling edges |

**Caution Be sure to clear bits 6 and 7 of EGPCTL0 and EGNCTL0, and bits 0, 1, 4 to 7 of EGPCTL1 and EGNCTL1 to 0 in 78K0/KB2-L.**

**Remark** n = 0 to 5, 10, 11: 78K0/KB2-L

**Figure 17-23. Format of External Interrupt Rising Edge Enable Registers (EGPCTL0, EGPCTL1) and External Interrupt Falling Edge Enable Registers (EGNCTL0, EGNCTL1) (3/5)**

<R>**(e) 40-pin products of 78K0/KC2-L**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | 0 | 0 | EGP5 | EGP4 | EGP3 | EGP2 | EGP1 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | 0 | 0 | EGN5 | EGN4 | EGN3 | EGN2 | EGN1 | EGN0 |

Address: FF4AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL1 | 0 | 0 | 0 | 0 | EGP11 | EGP10 | EGP9 | 0 |

Address: FF4BH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL1 | 0 | 0 | 0 | 0 | EGN11 | EGN10 | EGP9 | 0 |

| EGPn | EGNn | INTPn pin valid edge selection |
|---|---|---|
| 0 | 0 | Edge detection disabled |
| 0 | 1 | Falling edge |
| 1 | 0 | Rising edge |
| 1 | 1 | Both rising and falling edges |

**Caution Be sure to clear bits 6 and 7 of EGPCTL0 and EGNCTL0, and bits 0 and 4 to 7 of EGPCTL1 and EGNCTL1 to 0 in the 40-pin products of 78K0/KC2-L.**

**Remark** n = 0 to 5, 9 to 11: 40-pin products of 78K0/KC2-L

**Figure 17-23. Format of External Interrupt Rising Edge Enable Registers (EGPCTL0, EGPCTL1) and External Interrupt Falling Edge Enable Registers (EGNCTL0, EGNCTL1) (4/5)**

**(f) 44-pin products of 78K0/KC2-L**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | 0 | 0 | EGP5 | EGP4 | EGP3 | EGP2 | EGP1 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | 0 | 0 | EGN5 | EGN4 | EGN3 | EGN2 | EGN1 | EGN0 |

Address: FF4AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL1 | 0 | 0 | 0 | 0 | EGP11 | EGP10 | EGP9 | EGP8 |

Address: FF4BH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL1 | 0 | 0 | 0 | 0 | EGN11 | EGN10 | EGP9 | EGP8 |

| EGPn | EGNn | INTPn pin valid edge selection |
|---|---|---|
| 0 | 0 | Edge detection disabled |
| 0 | 1 | Falling edge |
| 1 | 0 | Rising edge |
| 1 | 1 | Both rising and falling edges |

**Caution Be sure to clear bits 6 and 7 of EGPCTL0 and EGNCTL0, and bits 4 to 7 of EGPCTL1 and EGNCTL1 to 0 in the 44-pin products of 78K0/KC2-L.**

**Remark** n = 0 to 5, 8 to 11: 44-pin products of 78K0/KC2-L

**Figure 17-23. Format of External Interrupt Rising Edge Enable Registers (EGPCTL0, EGPCTL1) and External Interrupt Falling Edge Enable Registers (EGNCTL0, EGNCTL1) (5/5)**

**(g) 48-pin products of 78K0/KC2-L**

Address: FF48H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL0 | EGP7 | EGP6 | EGP5 | EGP4 | EGP3 | EGP2 | EGP1 | EGP0 |

Address: FF49H After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL0 | EGN7 | EGN6 | EGN5 | EGN4 | EGN3 | EGN2 | EGN1 | EGN0 |

Address: FF4AH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGPCTL1 | 0 | 0 | 0 | 0 | EGP11 | EGP10 | EGP9 | EGP8 |

Address: FF4BH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| EGNCTL1 | 0 | 0 | 0 | 0 | EGN11 | EGN10 | EGP9 | EGP8 |

| EGPn | EGNn | INTPn pin valid edge selection |
|---|---|---|
| 0 | 0 | Edge detection disabled |
| 0 | 1 | Falling edge |
| 1 | 0 | Rising edge |
| 1 | 1 | Both rising and falling edges |

**Caution Be sure to clear bits 4 to 7 of EGPCTL1 and EGNCTL1 to 0 in the 48-pin products of 78K0/KC2-L.**

**Remark** n = 0 to 11: 48-pin products of 78K0/KC2-L

Table 17-3 shows the ports corresponding to EGPn and EGNn.

**Table 17-3. Ports Corresponding to EGPn and EGNn (1/3)**

**(a) 78K0/KY2-L**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P00 | INTP0 |
| EGP1 | EGN1 | P30 | INTP1 |

**(b) 78K0/KA2-L (20-pin products)**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P00 | INTP0 |
| EGP1 | EGN1 | P30 | INTP1 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |

**(c) 78K0/KA2-L (25-pin products)**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P00 or P121 | INTP0 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |
| EGP4 | EGN4 | P34 | INTP4 |
| EGP5 | EGN5 | P02 | INTP5 |

**(d) 78K0/KA2-L (32-pin products)**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P121 or P125 | INTP0 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |
| EGP4 | EGN4 | P34 | INTP4 |
| EGP5 | EGN5 | P02 | INTP5 |

**Caution Select the port mode by clearing EGPn and EGNn to 0 because an edge may be detected when the external interrupt function is switched to the port function.**

**Remark** n = 0, 1: 78K0/KY2-L
n = 0 to 3: 78K0/KA2-L (20-pin products)
n = 0, 2 to 5: 78K0/KA2-L (25-pin and 32-pin products)

**Table 17-3. Ports Corresponding to EGPn and EGNn (2/3)**

**(e) 78K0/KB2-L**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P120 | INTP0 |
| EGP1 | EGN1 | P30 | INTP1 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |
| EGP4 | EGN4 | P33 | INTP4 |
| EGP5 | EGN5 | P16 | INTP5 |
| EGP10 | EGN10 | P61 | INTP10 |
| EGP11 | EGN11 | P60 | INTP11 |

<R> **(f) 78K0/KC2-L (40-pin products)**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P120 | INTP0 |
| EGP1 | EGN1 | P30 | INTP1 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |
| EGP4 | EGN4 | P33 | INTP4 |
| EGP5 | EGN5 | P16 | INTP5 |
| EGP9 | EGN9 | P62 | INTP9 |
| EGP10 | EGN10 | P61 | INTP10 |
| EGP11 | EGN11 | P60 | INTP11 |

**Caution Select the port mode by clearing EGPn and EGNn to 0 because an edge may be detected when the external interrupt function is switched to the port function.**

**Remark** n = 0 to 5, 10, 11: 78K0/KB2-L
n = 0 to 5, 9 to 11: 78K0/KC2-L (40-pin products)

**Table 17-3. Ports Corresponding to EGPn and EGNn (3/3)**

**(g) 78K0/KC2-L (44-pin products)**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P120 | INTP0 |
| EGP1 | EGN1 | P30 | INTP1 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |
| EGP4 | EGN4 | P33 | INTP4 |
| EGP5 | EGN5 | P16 | INTP5 |
| EGP8 | EGN8 | P63 | INTP8 |
| EGP9 | EGN9 | P62 | INTP9 |
| EGP10 | EGN10 | P61 | INTP10 |
| EGP11 | EGN11 | P60 | INTP11 |

**(h) 78K0/KC2-L (48-pin products)**

| Detection Enable Register | | Edge Detection Port | Interrupt Request Signal |
|---|---|---|---|
| EGP0 | EGN0 | P120 | INTP0 |
| EGP1 | EGN1 | P30 | INTP1 |
| EGP2 | EGN2 | P31 | INTP2 |
| EGP3 | EGN3 | P32 | INTP3 |
| EGP4 | EGN4 | P33 | INTP4 |
| EGP5 | EGN5 | P16 | INTP5 |
| EGP6 | EGN6 | P42 | INTP6 |
| EGP7 | EGN7 | P02 | INTP7 |
| EGP8 | EGN8 | P63 | INTP8 |
| EGP9 | EGN9 | P62 | INTP9 |
| EGP10 | EGN10 | P61 | INTP10 |
| EGP11 | EGN11 | P60 | INTP11 |

**Caution Select the port mode by clearing EGPn and EGNn to 0 because an edge may be detected when the external interrupt function is switched to the port function.**

**Remark** n = 0 to 5, 8 to 11: 78K0/KC2-L (44-pin products)
n = 0 to 11: 78K0/KC2-L (48-pin products)

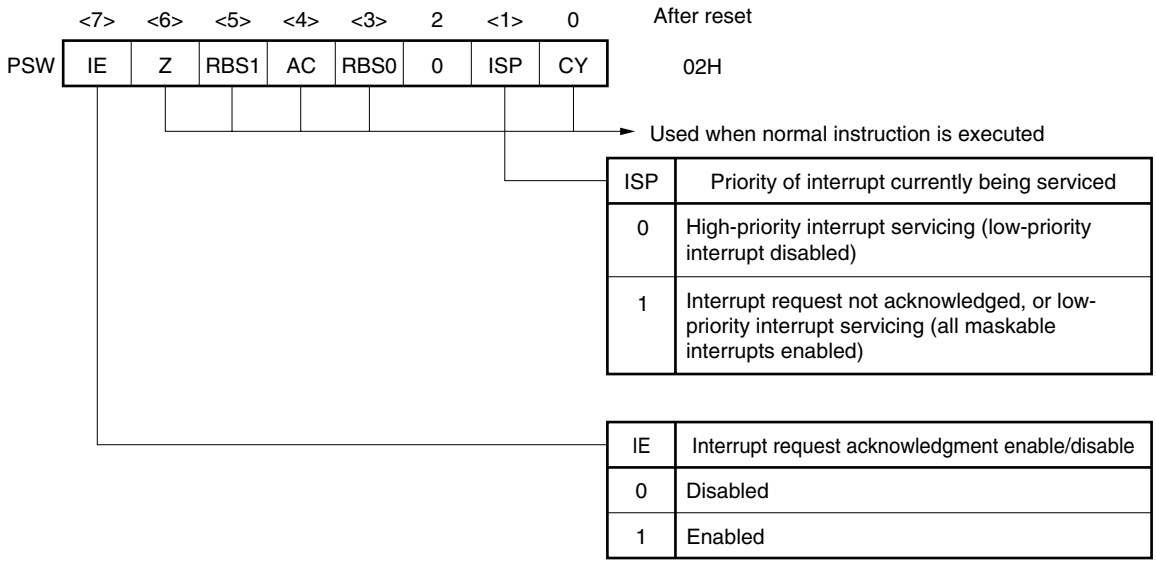
**(5) Program status word (PSW)**

The program status word is a register used to hold the instruction execution result and the current status for an interrupt request. The IE flag that sets maskable interrupt enable/disable and the ISP flag that controls multiple interrupt servicing are mapped to the PSW.

Besides 8-bit read/write, this register can carry out operations using bit manipulation instructions and dedicated instructions (EI and DI). When a vectored interrupt request is acknowledged, if the BRK instruction is executed, the contents of the PSW are automatically saved into a stack and the IE flag is reset to 0. If a maskable interrupt request is acknowledged, the contents of the priority specification flag of the acknowledged interrupt are transferred to the ISP flag. The PSW contents are also saved into the stack with the PUSH PSW instruction. They are restored from the stack with the RETI, RETB, and POP PSW instructions.

Reset signal generation sets PSW to 02H.

**Figure 17-24. Format of Program Status Word**




| ISP | Priority of interrupt currently being serviced |
|---|---|
| 0 | High-priority interrupt servicing (low-priority interrupt disabled) |
| 1 | Interrupt request not acknowledged, or low-priority interrupt servicing (all maskable interrupts enabled) |

| IE | Interrupt request acknowledgment enable/disable |
|---|---|
| 0 | Disabled |
| 1 | Enabled |

## 17.4 Interrupt Servicing Operations

### 17.4.1 Maskable interrupt acknowledgment

A maskable interrupt becomes acknowledgeable when the interrupt request flag is set to 1 and the mask (MK) flag corresponding to that interrupt request is cleared to 0. A vectored interrupt request is acknowledged if interrupts are in the interrupt enabled state (when the IE flag is set to 1). However, a low-priority interrupt request is not acknowledged during servicing of a higher priority interrupt request (when the ISP flag is reset to 0).

The times from generation of a maskable interrupt request until vectored interrupt servicing is performed are listed in Table 17-4 below.

For the interrupt request acknowledgment timing, refer to **Figures 17-26** and **17-27**.

**Table 17-4. Time from Generation of Maskable Interrupt Until Servicing**

| | Minimum Time | Maximum Time[Note] |
|---|---|---|
| When ××PR = 0 | 7 clocks | 32 clocks |
| When ××PR = 1 | 8 clocks | 33 clocks |

**Note** If an interrupt request is generated just before a divide instruction, the wait time becomes longer.

**Remark** 1 clock: $1/f_{CPU}$ ($f_{CPU}$: CPU clock)

If two or more maskable interrupt requests are generated simultaneously, the request with a higher priority level specified in the priority specification flag is acknowledged first. If two or more interrupts requests have the same priority level, the request with the highest default priority is acknowledged first.

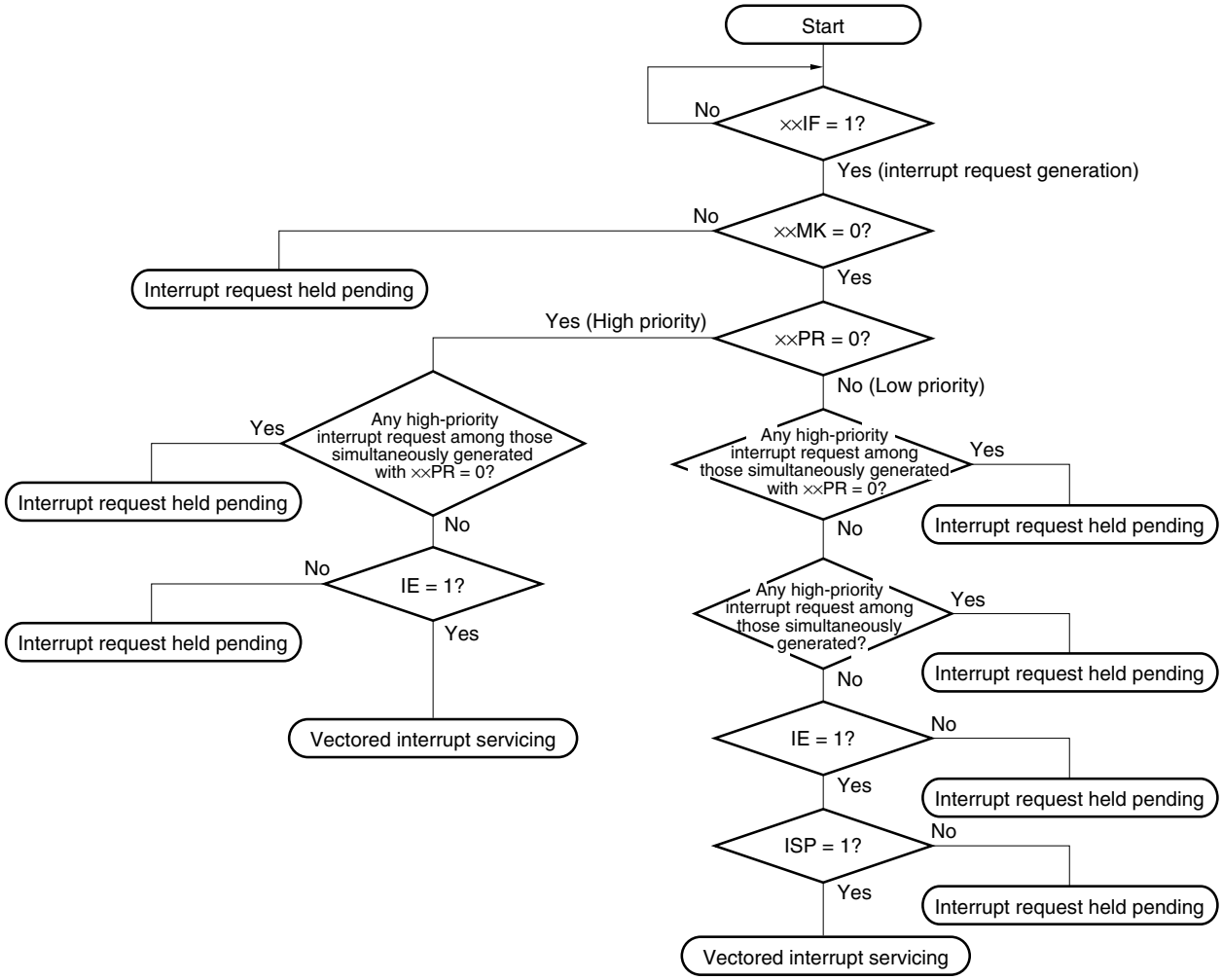
An interrupt request that is held pending is acknowledged when it becomes acknowledgeable.

Figure 17-25 shows the interrupt request acknowledgment algorithm.

If a maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag is reset (0), and the contents of the priority specification flag corresponding to the acknowledged interrupt are transferred to the ISP flag. The vector table data determined for each interrupt request is the loaded into the PC and branched.

Restoring from an interrupt is possible by using the RETI instruction.

**Figure 17-25. Interrupt Request Acknowledgment Processing Algorithm**




××IF: Interrupt request flag

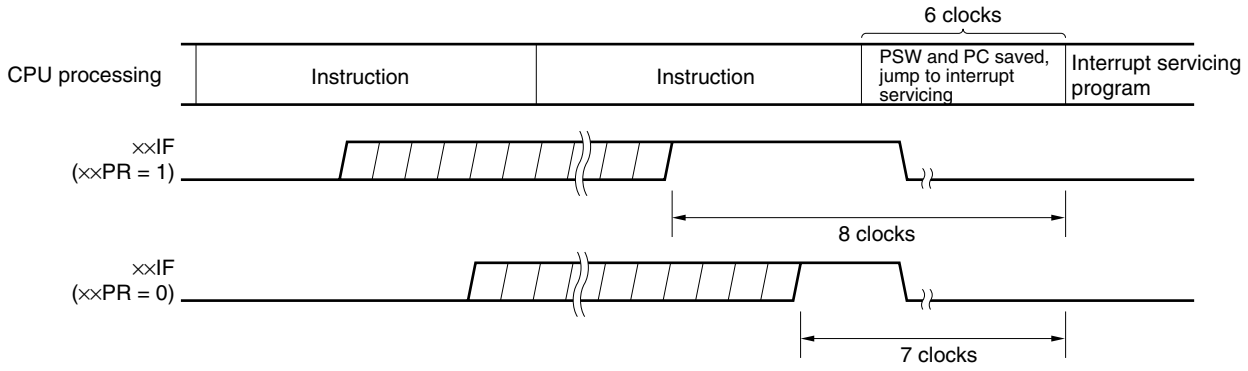
××MK: Interrupt mask flag

××PR: Priority specification flag

IE: Flag that controls acknowledgment of maskable interrupt request (1 = Enable, 0 = Disable)

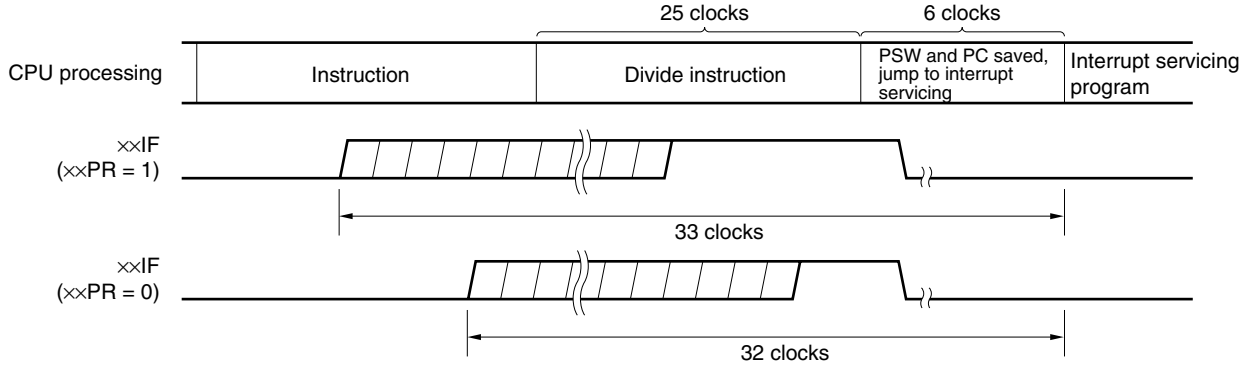
ISP: Flag that indicates the priority level of the interrupt currently being serviced (0 = high-priority interrupt servicing, 1 = No interrupt request acknowledged, or low-priority interrupt servicing)

**Figure 17-26. Interrupt Request Acknowledgment Timing (Minimum Time)**




**Remark** 1 clock: 1/$f_{CPU}$ ($f_{CPU}$: CPU clock)

**Figure 17-27. Interrupt Request Acknowledgment Timing (Maximum Time)**




**Remark** 1 clock: 1/$f_{CPU}$ ($f_{CPU}$: CPU clock)

### 17.4.2 Software interrupt request acknowledgment

A software interrupt acknowledge is acknowledged by BRK instruction execution. Software interrupts cannot be disabled.

If a software interrupt request is acknowledged, the contents are saved into the stacks in the order of the program status word (PSW), then program counter (PC), the IE flag is reset (0), and the contents of the vector table (003EH, 003FH) are loaded into the PC and branched.

Restoring from a software interrupt is possible by using the RETB instruction.

**Caution Do not use the RETI instruction for restoring from the software interrupt.**

### 17.4.3 Multiple interrupt servicing

Multiple interrupt servicing occurs when another interrupt request is acknowledged during execution of an interrupt.

Multiple interrupt servicing does not occur unless the interrupt request acknowledgment enabled state is selected (IE = 1). When an interrupt request is acknowledged, interrupt request acknowledgment becomes disabled (IE = 0). Therefore, to enable multiple interrupt servicing, it is necessary to set (1) the IE flag with the EI instruction during interrupt servicing to enable interrupt acknowledgment.

Moreover, even if interrupts are enabled, multiple interrupt servicing may not be enabled, this being subject to interrupt priority control. Two types of priority control are available: default priority control and programmable priority control. Programmable priority control is used for multiple interrupt servicing.

In the interrupt enabled state, if an interrupt request with a priority equal to or higher than that of the interrupt currently being serviced is generated, it is acknowledged for multiple interrupt servicing. If an interrupt with a priority lower than that of the interrupt currently being serviced is generated during interrupt servicing, it is not acknowledged for multiple interrupt servicing. Interrupt requests that are not enabled because interrupts are in the interrupt disabled state or because they have a lower priority are held pending. When servicing of the current interrupt ends, the pending interrupt request is acknowledged following execution of at least one main processing instruction execution.

Table 17-5 shows relationship between interrupt requests enabled for multiple interrupt servicing and Figure 17-28 shows multiple interrupt servicing examples.

**Table 17-5. Relationship Between Interrupt Requests Enabled for Multiple Interrupt Servicing During Interrupt Servicing**

| Multiple Interrupt Request / Interrupt Being Serviced | | Maskable Interrupt Request | | | | Software Interrupt Request |
|---|---|---|---|---|---|---|
| | | PR = 0 | | PR = 1 | | |
| | | IE = 1 | IE = 0 | IE = 1 | IE = 0 | |
| Maskable interrupt | ISP = 0 | ○ | × | × | × | ○ |
| | ISP = 1 | ○ | × | ○ | × | ○ |
| Software interrupt | | ○ | × | ○ | × | ○ |

**Remarks 1.** ○: Multiple interrupt servicing enabled

**2.** ×: Multiple interrupt servicing disabled

**3.** ISP and IE are flags contained in the PSW.

ISP = 0: An interrupt with higher priority is being serviced.

ISP = 1: No interrupt request has been acknowledged, or an interrupt with a lower priority is being serviced.

IE = 0: Interrupt request acknowledgment is disabled.

IE = 1: Interrupt request acknowledgment is enabled.

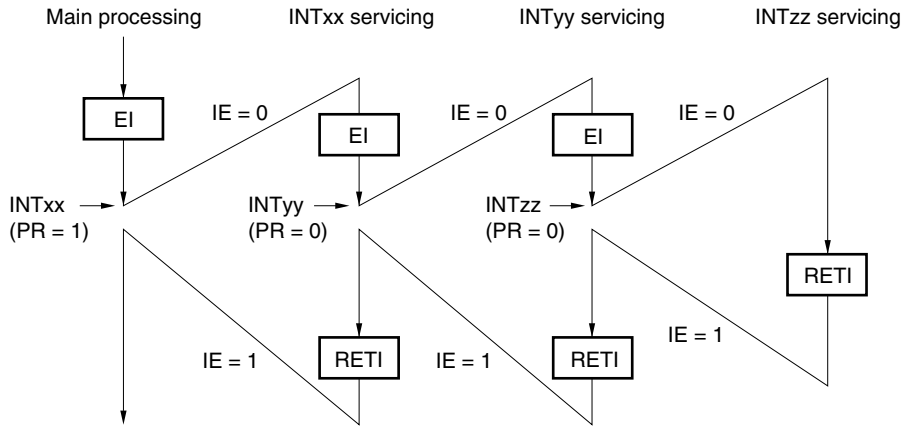
**4.** PR is a flag contained in PR0L, PR0H, PR1L, and PR1H.

PR = 0: Higher priority level

PR = 1: Lower priority level

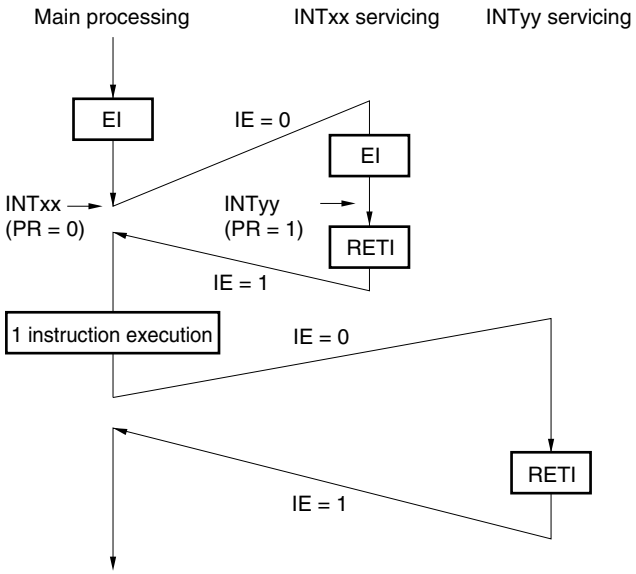
**Figure 17-28. Examples of Multiple Interrupt Servicing (1/2)**

**Example 1. Multiple interrupt servicing occurs twice**




During servicing of interrupt INTxx, two interrupt requests, INTyy and INTzz, are acknowledged, and multiple interrupt servicing takes place. Before each interrupt request is acknowledged, the EI instruction must always be issued to enable interrupt request acknowledgment.

**Example 2. Multiple interrupt servicing does not occur due to priority control**

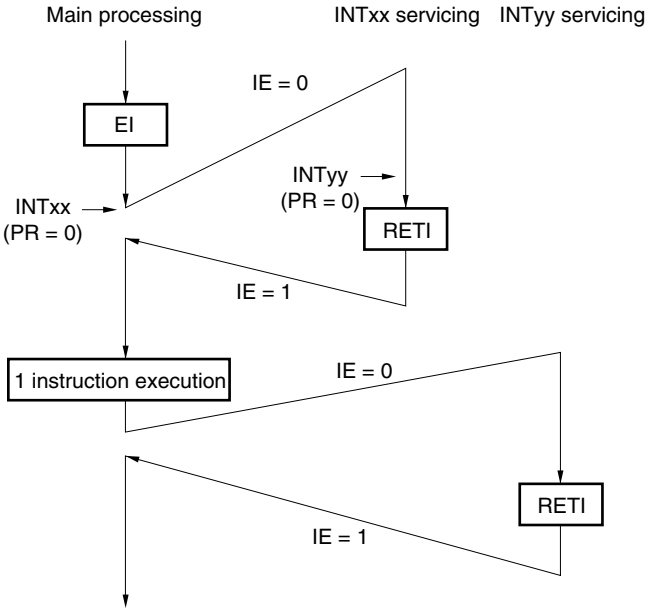



Interrupt request INTyy issued during servicing of interrupt INTxx is not acknowledged because its priority is lower than that of INTxx, and multiple interrupt servicing does not take place. The INTyy interrupt request is held pending, and is acknowledged following execution of one main processing instruction.

PR = 0: Higher priority level
PR = 1: Lower priority level
IE = 0: Interrupt request acknowledgment disabled

**Figure 17-28. Examples of Multiple Interrupt Servicing (2/2)**

**Example 3. Multiple interrupt servicing does not occur because interrupts are not enabled**




Interrupts are not enabled during servicing of interrupt INTxx (EI instruction is not issued), therefore, interrupt request INTyy is not acknowledged and multiple interrupt servicing does not take place. The INTyy interrupt request is held pending, and is acknowledged following execution of one main processing instruction.

PR = 0: Higher priority level
IE = 0: Interrupt request acknowledgment disabled

### 17.4.4 Interrupt request hold

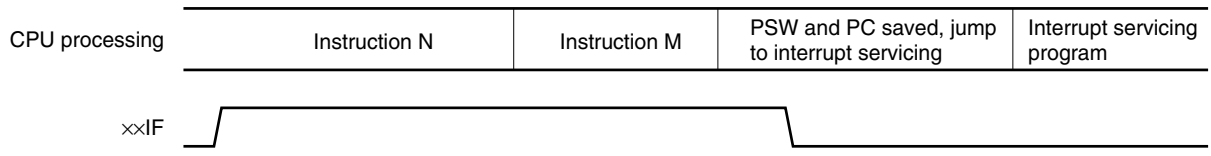
There are instructions where, even if an interrupt request is issued for them while another instruction is being executed, request acknowledgment is held pending until the end of execution of the next instruction. These instructions (interrupt request hold instructions) are listed below.

- MOV PSW, #byte
- MOV A, PSW
- MOV PSW, A
- MOV1 PSW. bit, CY
- MOV1 CY, PSW. bit
- AND1 CY, PSW. bit
- OR1 CY, PSW. bit
- XOR1 CY, PSW. bit
- SET1 PSW. bit
- CLR1 PSW. bit
- RETB
- RETI
- PUSH PSW
- POP PSW
- BT PSW. bit, $addr16
- BF PSW. bit, $addr16
- BTCLR PSW. bit, $addr16
- EI
- DI
- Manipulation instructions for the IF0L, IF0H, IF1L, IF1H, MK0L, MK0H, MK1L, MK1H, PR0L, PR0H, PR1L, and PR1H registers.

**Caution The BRK instruction is not one of the above-listed interrupt request hold instructions. However, the software interrupt activated by executing the BRK instruction causes the IE flag to be cleared. Therefore, even if a maskable interrupt request is generated during execution of the BRK instruction, the interrupt request is not acknowledged.**

Figure 17-29 shows the timing at which interrupt requests are held pending.

**Figure 17-29. Interrupt Request Hold**




**Remarks 1.** Instruction N: Interrupt request hold instruction

**2.** Instruction M: Instruction other than interrupt request hold instruction

**3.** The ××PR (priority level) values do not affect the operation of ××IF (interrupt request).

# CHAPTER 18 KEY INTERRUPT FUNCTION

<R>

| Item | 78K0/KY2-L ($\mu$PD78F055x) | 78K0/KA2-L ($\mu$PD78F056x) | 78K0/KB2-L ($\mu$PD78F057x) | 78K0/KC2-L ($\mu$PD78F058x) | |
|---|---|---|---|---|---|
| | 16 pins | 20, 25 32 pins | 30 pins | 40, 44 pins | 48 pins |
| Key interrupt | – | | | 4 ch | 6 ch |

## 18.1 Functions of Key Interrupt

A key interrupt (INTKR) can be generated by setting the key return mode register (KRM) and inputting a falling edge to the key interrupt input pins (KRn).

**Table 18-1. Assignment of Key Interrupt Detection Pins**

| Flag | Description |
|---|---|
| KRMn | Controls KRn signal in 1-bit units. |

**Remark** n = 0 to 3: 40, 44-pin products of 78K0/KC2-L
n = 0 to 5: 48-pin products of 78K0/KC2-L

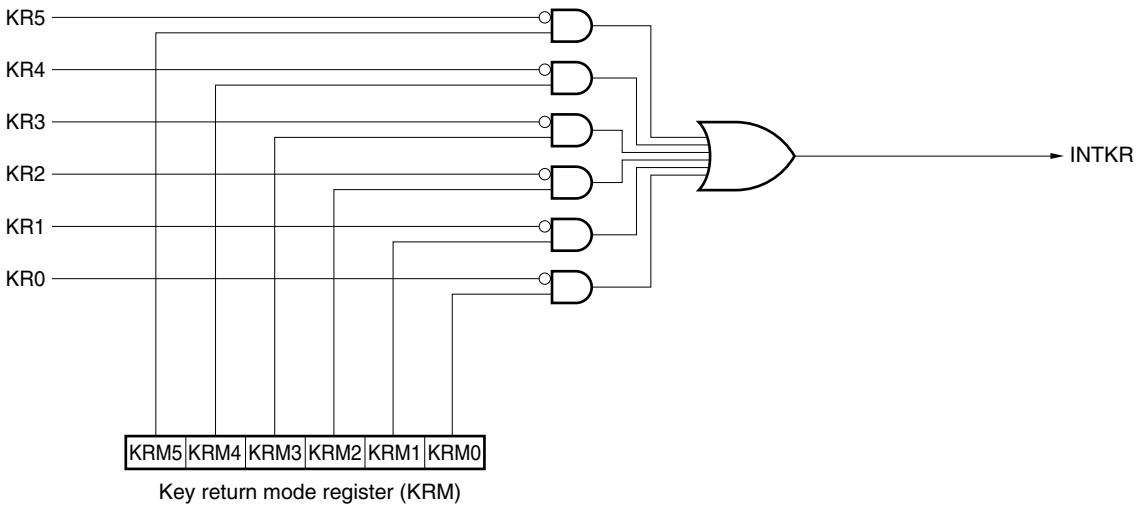
## 18.2 Configuration of Key Interrupt

The key interrupt includes the following hardware.

**Table 18-2. Configuration of Key Interrupt**

| Item | Configuration |
|---|---|
| Control register | Key return mode register (KRM) |

**Figure 18-1. Block Diagram of Key Interrupt**




**Remark** KR0 to KR3, KRM0 to KRM3: 40, 44-pin products of 78K0/KC2-L
KR0 to KR5, KRM0 to KRM5: 48-pin products of 78K0/KC2-L

## 18.3 Register Controlling Key Interrupt

### (1) Key return mode register (KRM)

This register controls the KRMn bit using the KRn signal.

KRM is set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears KRM to 00H.

**Figure 18-2. Format of Key Return Mode Register (KRM) (78K0/KC2-L)**

<R> (a) 40-pin and 44-pin products

Address: FF6EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| KRM | 0 | 0 | 0 | 0 | KRM3 | KRM2 | KRM1 | KRM0 |

(b) 48-pin products

Address: FF6EH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| KRM | 0 | 0 | KRM5 | KRM4 | KRM3 | KRM2 | KRM1 | KRM0 |

| KRMn | Key interrupt mode control |
|---|---|
| 0 | Does not detect key interrupt signal |
| 1 | Detects key interrupt signal |

**Cautions 1. If any of the KRMn bits used is set to 1, set bit n (PU7n) of the corresponding pull-up resistor register 7 (PU7) to 1.**

**2. If KRM is changed, the interrupt request flag may be set. Therefore, disable interrupts and then change the KRM register. Clear the interrupt request flag and enable interrupts.**

**3. The bits not used in the key interrupt mode can be used as normal ports.**

**4. For the 40, 44-pin products of 78K0/KC2-L, be sure to set bits 4 to 7 of KRM to "0". For the 48-pin products of 78K0/KC2-L, be sure to set bits 6 and 7 of KRM to "0".**

**Remark** n = 0 to 3: 40-pin and 44-pin products of 78K0/KC2-L

n = 0 to 5: 48-pin products of 78K0/KC2-L

# CHAPTER 19 STANDBY FUNCTION

## 19.1 Standby Function and Configuration

### 19.1.1 Standby function

The standby function is mounted onto all 78K0/Kx2-L microcontroller products.

The standby function is designed to reduce the operating current of the system. The following two modes are available.

**(1) HALT mode**

HALT instruction execution sets the HALT mode. In the HALT mode, the CPU operation clock is stopped. If the high-speed system clock oscillator, internal high-speed oscillator, internal low-speed oscillator, or subsystem clock oscillator[Note] is operating before the HALT mode is set, oscillation of each clock continues. In this mode, the operating current is not decreased as much as in the STOP mode, but the HALT mode is effective for restarting operation immediately upon interrupt request generation and carrying out intermittent operations frequently.

**Note** 78K0/KC2-L only

**(2) STOP mode**

STOP instruction execution sets the STOP mode. In the STOP mode, the high-speed system clock oscillator and internal high-speed oscillator stop, stopping the whole system, thereby considerably reducing the CPU operating current.

Because this mode can be cleared by an interrupt request, it enables intermittent operations to be carried out. However, because a wait time is required to secure the oscillation stabilization time after the STOP mode is released when the X1 clock is selected, select the HALT mode if it is necessary to start processing immediately upon interrupt request generation.

In either of these two modes, all the contents of registers, flags and data memory just before the standby mode is set are held. The I/O port output latches and output buffer statuses are also held.

**Cautions 1. The STOP mode can be used only when the CPU is operating on the main system clock. The subsystem clock oscillation cannot be stopped. The HALT mode can be used when the CPU is operating on either the main system clock or the subsystem clock.**

**2. When shifting to the STOP mode, be sure to stop the peripheral hardware operation operating with main system clock before executing STOP instruction.**

**3. The following sequence is recommended for operating current reduction of the A/D converter when the standby function is used: First clear bit 7 (ADCS) and bit 0 (ADCE) of the A/D converter mode register 0 (ADM0) to 0 to stop the A/D conversion operation, and then execute the STOP instruction.**

**4. Stop the operational amplifier before executing the STOP instruction.**

### 19.1.2 Registers controlling standby function

The standby function is controlled by the following two registers.

- Oscillation stabilization time counter status register (OSTC)
- Oscillation stabilization time select register (OSTS)

**Remark** For the registers that start, stop, or select the clock, refer to **CHAPTER 5 CLOCK GENERATOR**.

#### (1) Oscillation stabilization time counter status register (OSTC)

This is the register that indicates the count status of the X1 clock oscillation stabilization time counter. When X1 clock oscillation starts with the internal high-speed oscillation clock or subsystem clock used as the CPU clock, the X1 clock oscillation stabilization time can be checked.

OSTC can be read by a 1-bit or 8-bit memory manipulation instruction.

When reset is released (reset by $\overline{\text{RESET}}$ input, POC, LVI, and WDT), the STOP instruction and MSTOP (bit 7 of MOC register) = 1 clear OSTC to 00H.

**Figure 19-1. Format of Oscillation Stabilization Time Counter Status Register (OSTC)**

Address: FFA3H After reset: 00H R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| OSTC | 0 | 0 | 0 | MOST11 | MOST13 | MOST14 | MOST15 | MOST16 |

| MOST11 | MOST13 | MOST14 | MOST15 | MOST16 | Oscillation stabilization time status | |
|---|---|---|---|---|---|---|
| | | | | | | fx = 10 MHz |
| 1 | 0 | 0 | 0 | 0 | $2^{11}$/fx min. | 204.8 μs min. |
| 1 | 1 | 0 | 0 | 0 | $2^{13}$/fx min. | 819.2 μs min. |
| 1 | 1 | 1 | 0 | 0 | $2^{14}$/fx min. | 1.64 ms min. |
| 1 | 1 | 1 | 1 | 0 | $2^{15}$/fx min. | 3.27 ms min. |
| 1 | 1 | 1 | 1 | 1 | $2^{16}$/fx min. | 6.55 ms min. |

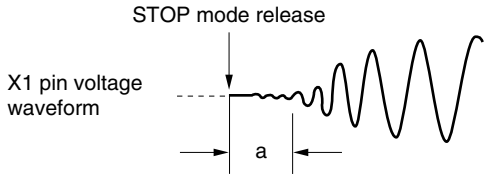
**Cautions 1. After the above time has elapsed, the bits are set to 1 in order from MOST11 and remain 1.**

**2. The oscillation stabilization time counter counts up to the oscillation stabilization time set by OSTS. If the STOP mode is entered and then released while the internal high-speed oscillation clock is being used as the CPU clock, set the oscillation stabilization time as follows.**

- **Desired OSTC oscillation stabilization time ≤ Oscillation stabilization time set by OSTS**

**Note, therefore, that only the status up to the oscillation stabilization time set by OSTS is set to OSTC after STOP mode is released.**

**3. The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).**




**Remark** fx: X1 clock oscillation frequency

**(2) Oscillation stabilization time select register (OSTS)**

This register is used to select the X1 clock oscillation stabilization wait time when the STOP mode is released.

When the X1 clock is selected as the CPU clock, the operation waits for the time set using OSTS after the STOP mode is released.

When the internal high-speed oscillation clock is selected as the CPU clock, confirm with OSTC that the desired oscillation stabilization time has elapsed after the STOP mode is released. The oscillation stabilization time can be checked up to the time set using OSTC.

OSTS can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets OSTS to 05H.

**Figure 19-2. Format of Oscillation Stabilization Time Select Register (OSTS)**

Address: FFA4H After reset: 05H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| OSTS | 0 | 0 | 0 | 0 | 0 | OSTS2 | OSTS1 | OSTS0 |

| OSTS2 | OSTS1 | OSTS0 | Oscillation stabilization time selection | |
|---|---|---|---|---|
| | | | | fx = 10 MHz |
| 0 | 0 | 1 | $2^{11}$/fx | 204.8 μs |
| 0 | 1 | 0 | $2^{13}$/fx | 819.2 μs |
| 0 | 1 | 1 | $2^{14}$/fx | 1.64 ms |
| 1 | 0 | 0 | $2^{15}$/fx | 3.27 ms |
| 1 | 0 | 1 | $2^{16}$/fx | 6.55 ms |
| Other than above | | | Setting prohibited | |

**Cautions 1. To set the STOP mode when the X1 clock is used as the CPU clock, set OSTS before executing the STOP instruction.**

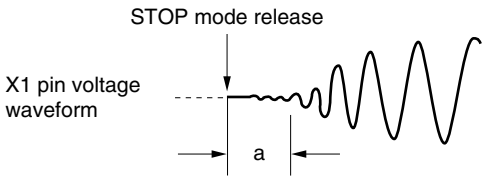
**2. Do not change the value of the OSTS register during the X1 clock oscillation stabilization time.**

**3. The oscillation stabilization time counter counts up to the oscillation stabilization time set by OSTS. If the STOP mode is entered and then released while the internal high-speed oscillation clock is being used as the CPU clock, set the oscillation stabilization time as follows.**

- **Desired OSTC oscillation stabilization time ≤ Oscillation stabilization time set by OSTS**

**Note, therefore, that only the status up to the oscillation stabilization time set by OSTS is set to OSTC after STOP mode is released.**

**4. The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).**




**Remark** fx: X1 clock oscillation frequency

## 19.2 Standby Function Operation

### 19.2.1 HALT mode

**(1) HALT mode**

The HALT mode is set by executing the HALT instruction. HALT mode can be set regardless of whether the CPU clock before the setting was the high-speed system clock, internal high-speed oscillation clock, or subsystem clock[Note]. The operating statuses in the HALT mode are shown below.

**Note** 78K0/KC2-L only

**Table 19-1. Operating Statuses in HALT Mode (1/2)**

| HALT Mode Setting / Item | | | When HALT Instruction Is Executed While CPU Is Operating on Main System Clock | | |
|---|---|---|---|---|---|
| | | | When CPU Is Operating on Internal High-Speed Oscillation Clock ($f_{IH}$) | When CPU Is Operating on X1 Clock ($f_X$) | When CPU Is Operating on External Main System Clock ($f_{EXCLK}$) |
| System clock | | | Clock supply to the CPU is stopped | | |
| | Main system clock | $f_{IH}$ | Operation continues (cannot be stopped) | Status before HALT mode was set is retained | |
| | | $f_X$ | Status before HALT mode was set is retained | Operation continues (cannot be stopped) | Status before HALT mode was set is retained |
| | | $f_{EXCLK}$ | Operates or stops by external clock input | | Operation continues (cannot be stopped) |
| | Subsystem clock | $f_{XT}$ | Status before HALT mode was set is retained | | |
| | | $f_{EXCLKS}$ | Operates or stops by external clock input | | |
| | $f_{IL}$ | | Status before HALT mode was set is retained | | |
| CPU | | | Operation stopped | | |
| Flash memory | | | | | |
| RAM | | | Status before HALT mode was set is retained | | |
| Port (latch) | | | | | |
| 16-bit timer/event counter 00 | | | Operable | | |
| 8-bit timer/event counter | | 50 | | | |
| | | 51 | | | |
| 8-bit timer | | H0 | | | |
| | | H1 | | | |
| Real-time counter (RTC) | | | | | |
| Watchdog timer | | | Operable. Clock supply to watchdog timer stops when “internal low-speed oscillator can be stopped by software” is set by option byte. | | |
| Clock output | | | Operable | | |
| A/D converter | | | | | |
| Operational amplifiers 0, 1 | | | | | |
| Serial interface | | UART6 | | | |
| | | CSI10 | | | |
| | | CSI11 | | | |
| | | IICA | | | |
| Key interrupt | | | | | |
| Power-on-clear function | | | | | |
| Low-voltage detection function | | | | | |
| External interrupt | | | | | |

**Remarks 1.** $f_{IH}$: Internal high-speed oscillation clock, $f_X$: X1 clock
$f_{EXCLK}$: External main system clock, $f_{XT}$: XT1 clock
$f_{EXCLKS}$: External subsystem clock, $f_{IL}$: Internal low-speed oscillation clock

**2.** The functions mounted depend on the product. Refer to **1.4 Block Diagram** and **1.5 Outline of Functions**.

**Table 19-1. Operating Statuses in HALT Mode (2/2)**

| HALT Mode Setting / Item | | | When HALT Instruction Is Executed While CPU Is Operating on Subsystem Clock | |
|---|---|---|---|---|
| | | | When CPU Is Operating on XT1 Clock ($f_{XT}$) | When CPU Is Operating on External Subsystem Clock ($f_{EXCLKS}$) |
| System clock | | | Clock supply to the CPU is stopped | |
| | Main system clock | $f_{IH}$ | Status before HALT mode was set is retained | |
| | | $f_X$ | | |
| | | $f_{EXCLK}$ | Operates or stops by external clock input | |
| | Subsystem clock | $f_{XT}$ | Operation continues (cannot be stopped) | Status before HALT mode was set is retained |
| | | $f_{EXCLKS}$ | Operates or stops by external clock input | Operation continues (cannot be stopped) |
| | $f_{IL}$ | | Status before HALT mode was set is retained | |
| CPU | | | Operation stopped | |
| Flash memory | | | | |
| RAM | | | Status before HALT mode was set is retained | |
| Port (latch) | | | | |
| 16-bit timer/event counter 00 | | | Operable | |
| 8-bit timer/event counter | | 50 | | |
| | | 51 | | |
| 8-bit timer | | H0 | | |
| | | H1 | | |
| Real-time counter (RTC) | | | | |
| Watchdog timer | | | Operable. Clock supply to watchdog timer stops when “internal low-speed oscillator can be stopped by software” is set by option byte. | |
| Clock output | | | Operable | |
| A/D converter | | | Operable. However, operation disabled when peripheral hardware clock ($f_{PRS}$) is stopped. | |
| Operational amplifiers 0, 1 | | | Operable | |
| Serial interface | | UART6 | | |
| | | CSI10 | | |
| | | CSI11 | | |
| | | IICA | | |
| Key interrupt | | | | |
| Power-on-clear function | | | | |
| Low-voltage detection function | | | | |
| External interrupt | | | | |

**Remarks 1.** $f_{IH}$: Internal high-speed oscillation clock, $f_X$: X1 clock
$f_{EXCLK}$: External main system clock, $f_{XT}$: XT1 clock
$f_{EXCLKS}$: External subsystem clock, $f_{IL}$: Internal low-speed oscillation clock

**2.** The functions mounted depend on the product. Refer to **1.4 Block Diagram** and **1.5 Outline of Functions**.

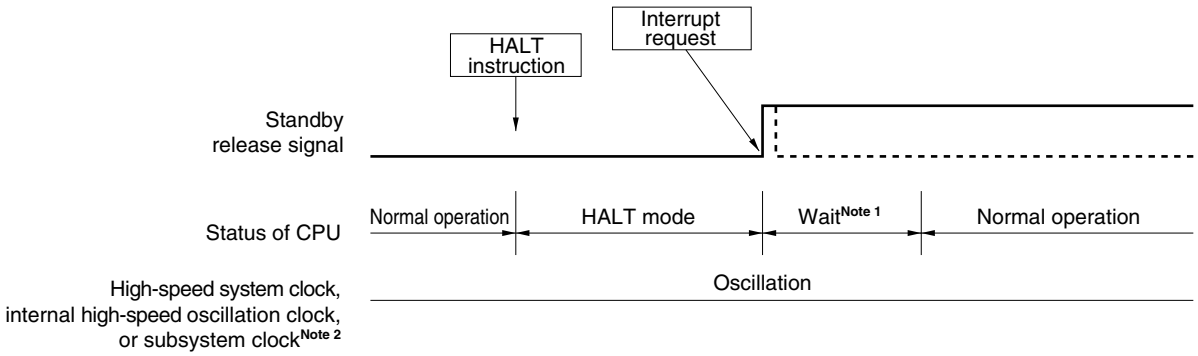
**(2) HALT mode release**

The HALT mode can be released by the following two sources.

**(a) Release by unmasked interrupt request**

When an unmasked interrupt request is generated, the HALT mode is released. If interrupt acknowledgment is enabled, vectored interrupt servicing is carried out. If interrupt acknowledgment is disabled, the next address instruction is executed.

**Figure 19-3. HALT Mode Release by Interrupt Request Generation**




**Notes 1.** The wait time is as follows:

- When vectored interrupt servicing is carried out: 11 or 12 clocks
- When vectored interrupt servicing is not carried out: 4 or 5 clocks

**2.** 78K0/KC2-L only

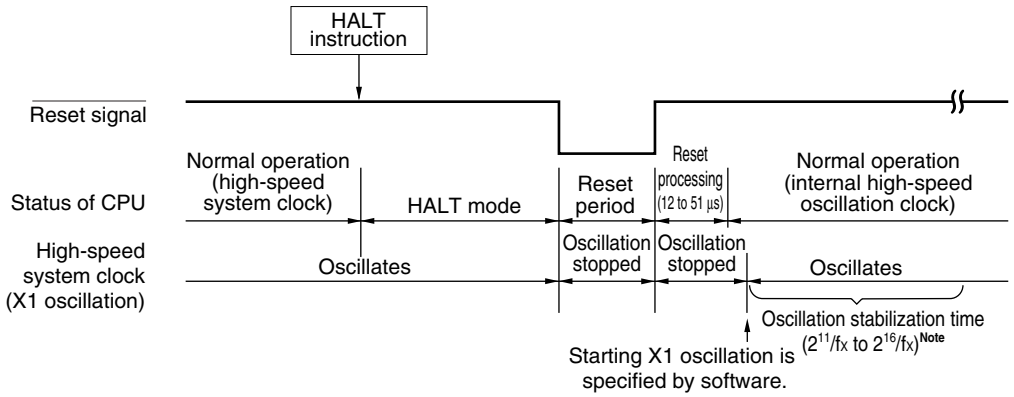
**Remark** The broken lines indicate the case when the interrupt request which has released the standby mode is acknowledged.

**(b) Release by reset signal generation**

When the reset signal is generated, HALT mode is released, and then, as in the case with a normal reset operation, the program is executed after branching to the reset vector address.

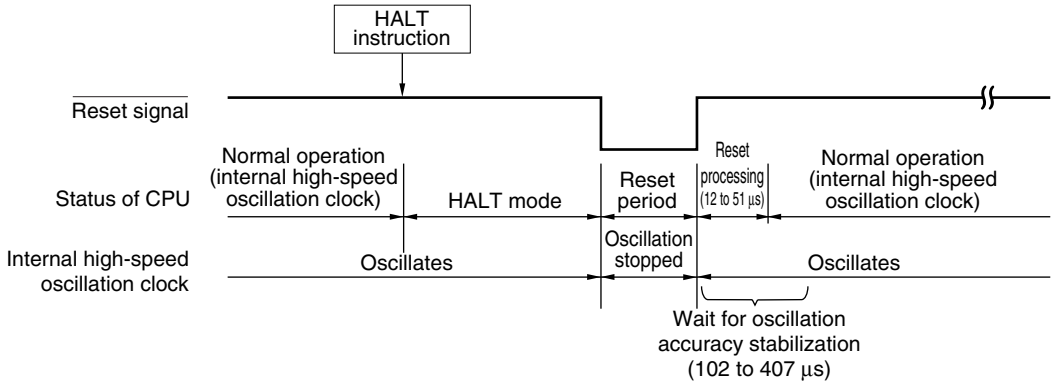
**Figure 19-4. HALT Mode Release by Reset (1/2)**

**(1) When high-speed system clock is used as CPU clock**




**Note** Oscillation stabilization time is not required when using the external main system clock ($f_{EXCLK}$) as the high-speed system clock.

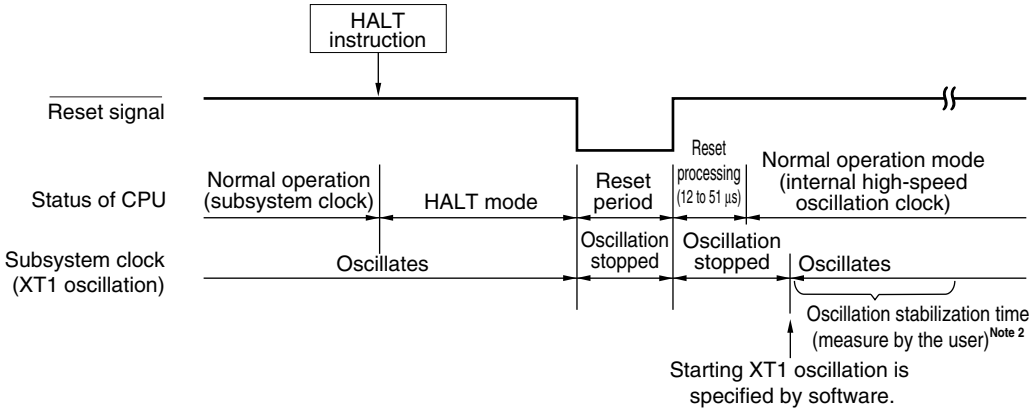
**(2) When internal high-speed oscillation clock is used as CPU clock**




**Remark** fx: X1 clock oscillation frequency

**Figure 19-4. HALT Mode Release by Reset (2/2)**

**(3) When subsystem clock is used as CPU clock[Note1]**




**Notes 1.** 78K0/KC2-L only

**2.** Oscillation stabilization time is not required when using the external subsystem clock ($f_{EXCLKS}$) as the subsystem clock.

**Table 19-2. Operation in Response to Interrupt Request in HALT Mode**

| Release Source | MK×× | PR×× | IE | ISP | Operation |
|---|---|---|---|---|---|
| Maskable interrupt request | 0 | 0 | 0 | × | Next address instruction execution |
| | 0 | 0 | 1 | × | Interrupt servicing execution |
| | 0 | 1 | 0 | 1 | Next address instruction execution |
| | 0 | 1 | × | 0 | |
| | 0 | 1 | 1 | 1 | Interrupt servicing execution |
| | 1 | × | × | × | HALT mode held |
| Reset | – | – | × | × | Reset processing |

×: don't care

### 19.2.2 STOP mode

**(1) STOP mode setting and operating statuses**

The STOP mode is set by executing the STOP instruction, and it can be set only when the CPU clock before the setting was the main system clock.

**Caution Because the interrupt request signal is used to clear the standby mode, if there is an interrupt source with the interrupt request flag set and the interrupt mask flag reset, the standby mode is immediately cleared if set. Thus, the STOP mode is reset to the HALT mode immediately after execution of the STOP instruction and the system returns to the operating mode as soon as the wait time set using the oscillation stabilization time select register (OSTS) has elapsed.**

The operating statuses in the STOP mode are shown below.

**Table 19-3. Operating Statuses in STOP Mode**

| STOP Mode Setting / Item | | | When STOP Instruction Is Executed While CPU Is Operating on Main System Clock | | |
|---|---|---|---|---|---|
| | | | When CPU Is Operating on Internal High-Speed Oscillation Clock ($f_{IH}$) | When CPU Is Operating on X1 Clock ($f_X$) | When CPU Is Operating on External Main System Clock ($f_{EXCLK}$) |
| System clock | | | Clock supply to the CPU is stopped | | |
| | Main system clock | $f_{IH}$ | Stopped | | |
| | | $f_X$ | | | |
| | | $f_{EXCLK}$ | Input invalid | | |
| | Subsystem clock | $f_{XT}$ | Status before STOP mode was set is retained | | |
| | | $f_{EXCLKS}$ | Operates or stops by external clock input | | |
| | $f_{IL}$ | | Status before STOP mode was set is retained | | |
| CPU | | | Operation stopped | | |
| Flash memory | | | | | |
| RAM | | | Status before STOP mode was set is retained | | |
| Port (latch) | | | | | |
| 16-bit timer/event counter 00 | | | Operation stopped | | |
| 8-bit timer/event counter | | 50 | Operable only when TI50 is selected as the count clock | | |
| | | 51 | Operable only when TI51 is selected as the count clock | | |
| 8-bit timer | | H0 | Operable only when TM50 output is selected as the count clock during 8-bit timer/event counter 50 operation | | |
| | | H1 | Operable only when $f_{IL}$, $f_{IL}/2^6$, $f_{IL}/2^{15}$ is selected as the count clock | | |
| Real-time counter (RTC) | | | Operable | | |
| Watchdog timer | | | Operable. Clock supply to watchdog timer stops when “internal low-speed oscillator can be stopped by software” is set by option byte. | | |
| Clock output | | | Operable only when subsystem clock is selected as the count clock | | |
| A/D converter | | | Operation stopped | | |
| Operational amplifiers 0, 1 | | | Operable | | |
| Serial interface | UART6 | | Operable only when TM50 output is selected as the serial clock during 8-bit timer/event counter 50 operation | | |
| | CSI10 | | Operable only when external clock is selected as the serial clock | | |
| | CSI11 | | | | |
| | IICA | | Wakeup by address match operable | | |
| Key interrupt | | | Operable | | |
| Power-on-clear function | | | | | |
| Low-voltage detection function | | | | | |
| External interrupt | | | | | |

**Remarks 1.** $f_{IH}$: Internal high-speed oscillation clock, $f_X$: X1 clock
$f_{EXCLK}$: External main system clock, $f_{XT}$: XT1 clock
$f_{EXCLKS}$: External subsystem clock, $f_{IL}$: Internal low-speed oscillation clock

**2.** The functions mounted depend on the product. Refer to **1.4 Block Diagram** and **1.5 Outline of Functions**.

**Cautions 1. To use the peripheral hardware that stops operation in the STOP mode, and the peripheral hardware for which the clock that stops oscillating in the STOP mode after the STOP mode is released, restart the peripheral hardware.**

<R> **2. When transitioning to the STOP mode, it is possible to achieve low power consumption by setting RMC = 56H.**

**3. Even if "internal low-speed oscillator can be stopped by software" is selected by the option byte, the internal low-speed oscillation clock continues in the STOP mode in the status before the STOP mode is set. To stop the internal low-speed oscillator's oscillation in the STOP mode, stop it by software and then execute the STOP instruction.**

**4. To shorten oscillation stabilization time after the STOP mode is released when the CPU operates with the high-speed system clock (X1 oscillation), switch the CPU clock to the internal high-speed oscillation clock before the execution of the STOP instruction using the following procedure.**

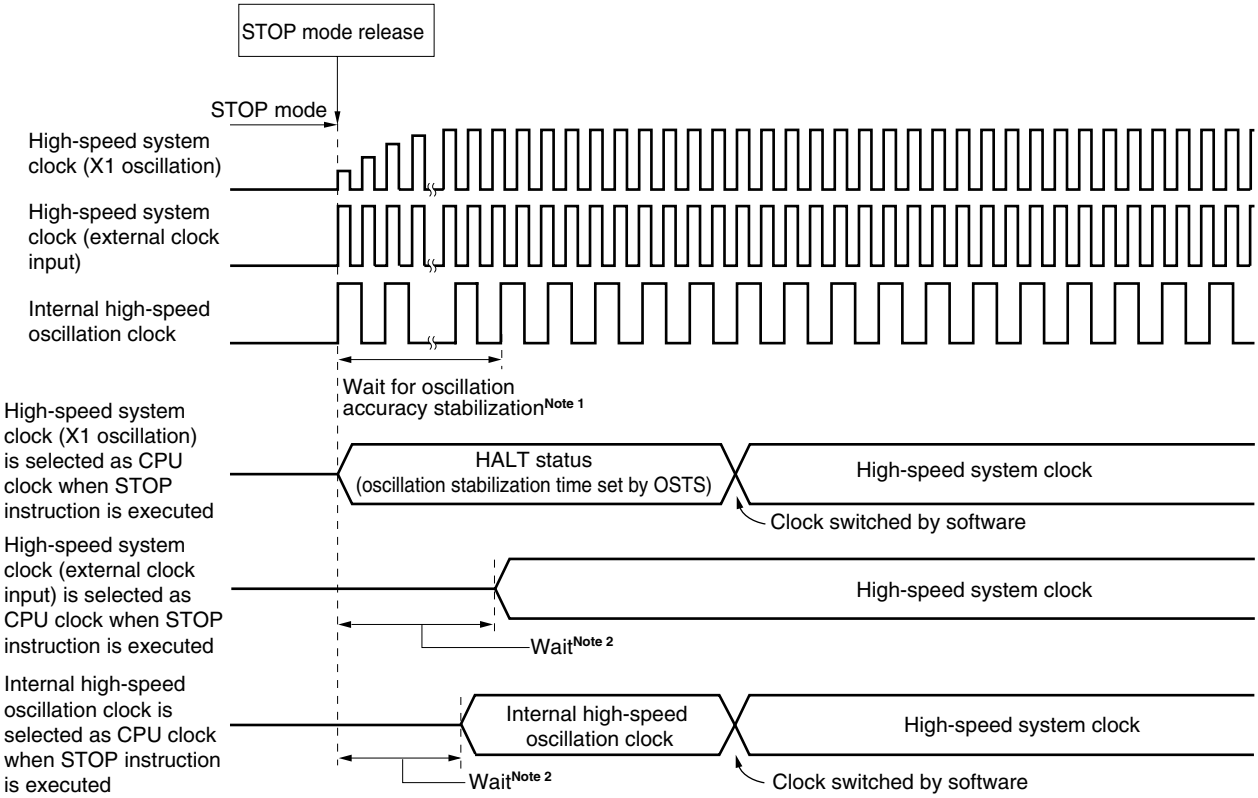
**<1> Set RSTOP to 0 (starting oscillation of the internal high-speed oscillator) → <2> Set MCM0 to 0 (switching the CPU from X1 oscillation to internal high-speed oscillation) → <3> Check that MCS is 0 (checking the CPU clock) → <4> Check that RSTS is 1 (checking internal high-speed oscillation operation) → <5> Execute the STOP instruction**

**Before changing the CPU clock from the internal high-speed oscillation clock to the high-speed system clock (X1 oscillation) after the STOP mode is released, check the oscillation stabilization time with the oscillation stabilization time counter status register (OSTC).**

**5. Execute the STOP instruction after having confirmed that the internal high-speed oscillator is operating stably (RSTS = 1).**

### (2) STOP mode release

**Figure 19-5. Operation Timing When STOP Mode Is Released (When Unmasked Interrupt Request Is Generated)**




**Notes 1.** The wait time for oscillation accuracy stabilization is as follows:

- RMC register = 00H: 102 to 407 $\mu$s
- RMC register = 56H: 120 to 481 $\mu$s

**2.** The wait time is as follows:

- When vectored interrupt servicing is carried out: 17 or 18 clocks
- When vectored interrupt servicing is not carried out: 11 or 12 clocks

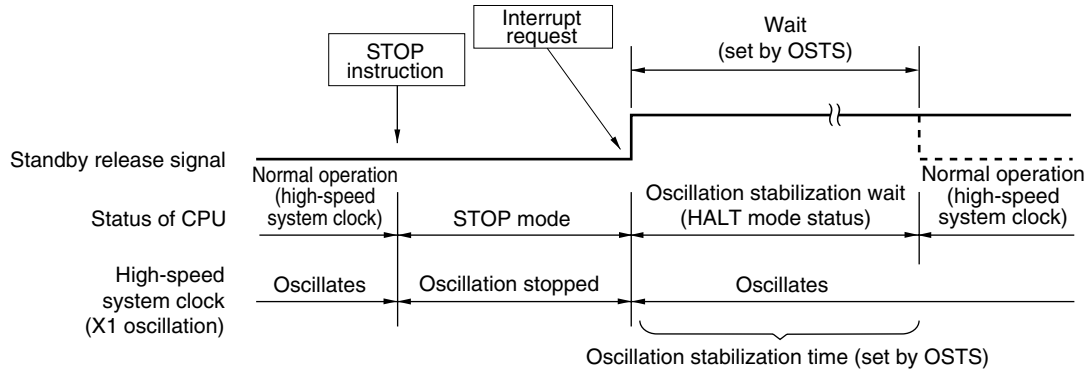
The STOP mode can be released by the following two sources.

#### (a) Release by unmasked interrupt request

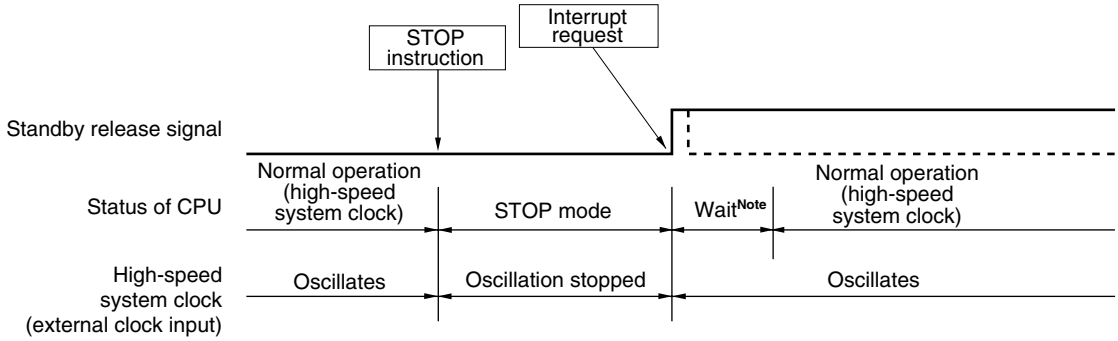
When an unmasked interrupt request is generated, the STOP mode is released. After the oscillation stabilization time has elapsed, if interrupt acknowledgment is enabled, vectored interrupt servicing is carried out. If interrupt acknowledgment is disabled, the next address instruction is executed.

**Figure 19-6. STOP Mode Release by Interrupt Request Generation (1/2)**

**(1) When high-speed system clock (X1 oscillation) is used as CPU clock**




**(2) When high-speed system clock (external clock input) is used as CPU clock**



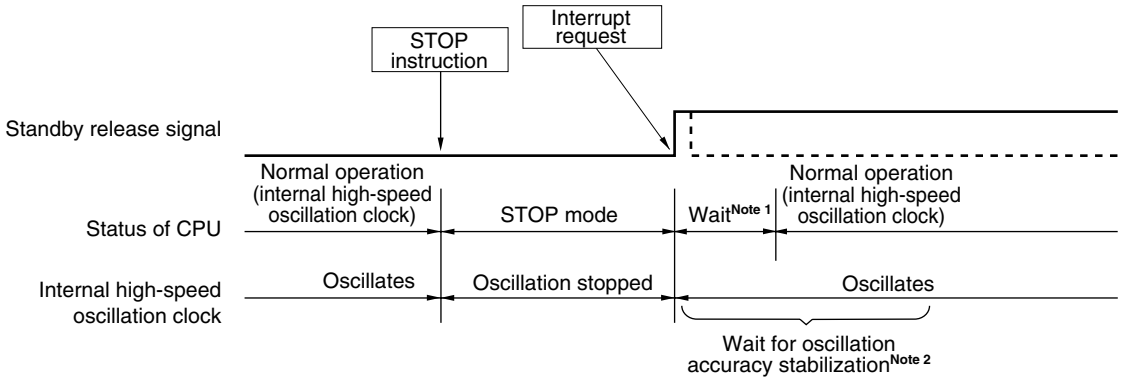

**Note** The wait time is as follows:

- When vectored interrupt servicing is carried out: 17 or 18 clocks
- When vectored interrupt servicing is not carried out: 11 or 12 clocks

**Remark** The broken lines indicate the case when the interrupt request that has released the standby mode is acknowledged.

**Figure 19-6. STOP Mode Release by Interrupt Request Generation (2/2)**

**(3) When internal high-speed oscillation clock is used as CPU clock**




**Notes 1.** The wait time is as follows:

- When vectored interrupt servicing is carried out: 17 or 18 clocks
- When vectored interrupt servicing is not carried out: 11 or 12 clocks

**2.** The wait time for oscillation accuracy stabilization is as follows:

- RMC register = 00H: 102 to 407 $\mu$s
- RMC register = 56H: 120 to 481 $\mu$s

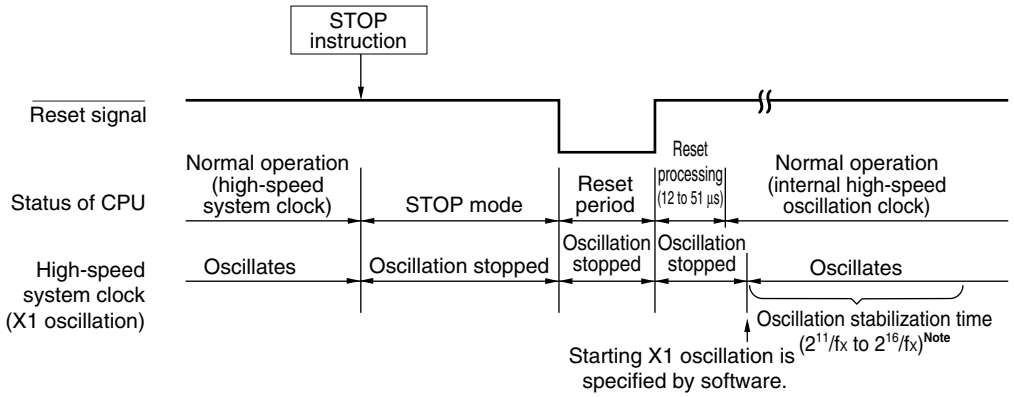
**Remark** The broken lines indicate the case when the interrupt request that has released the standby mode is acknowledged.

**(b) Release by reset signal generation**

When the reset signal is generated, STOP mode is released, and then, as in the case with a normal reset operation, the program is executed after branching to the reset vector address.

**Figure 19-7. STOP Mode Release by Reset**

**(1) When high-speed system clock is used as CPU clock**




**Note** Oscillation stabilization time is not required when using the external main system clock ($f_{EXCLK}$) as the high-speed system clock.

**Remark** $f_X$: X1 clock oscillation frequency

**(2) When internal high-speed oscillation clock is used as CPU clock**

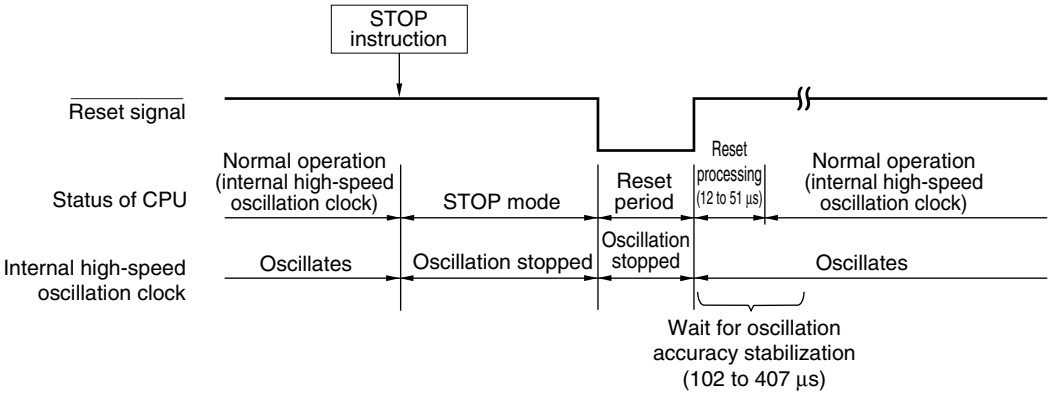


**Table 19-4. Operation in Response to Interrupt Request in STOP Mode**

| Release Source | MK×× | PR×× | IE | ISP | Operation |
|---|---|---|---|---|---|
| Maskable interrupt request | 0 | 0 | 0 | × | Next address instruction execution |
| | 0 | 0 | 1 | × | Interrupt servicing execution |
| | 0 | 1 | 0 | 1 | Next address instruction execution |
| | 0 | 1 | × | 0 | |
| | 0 | 1 | 1 | 1 | Interrupt servicing execution |
| | 1 | × | × | × | STOP mode held |
| Reset | – | – | × | × | Reset processing |

×: don't care

# CHAPTER 20 RESET FUNCTION

The reset function is mounted onto all 78K0/Kx2-L microcontroller products.

The following four operations are available to generate a reset signal.

(1) External reset input via $\overline{\text{RESET}}$ pin
(2) Internal reset by watchdog timer program loop detection
(3) Internal reset by comparison of supply voltage and detection voltage of power-on-clear (POC) circuit
(4) Internal reset by comparison of supply voltage of the low-voltage detector (LVI) or input voltage from external input pin (EXLVI pin), and detection voltage

External and internal resets start program execution from the address at 0000H and 0001H when the reset signal is generated.

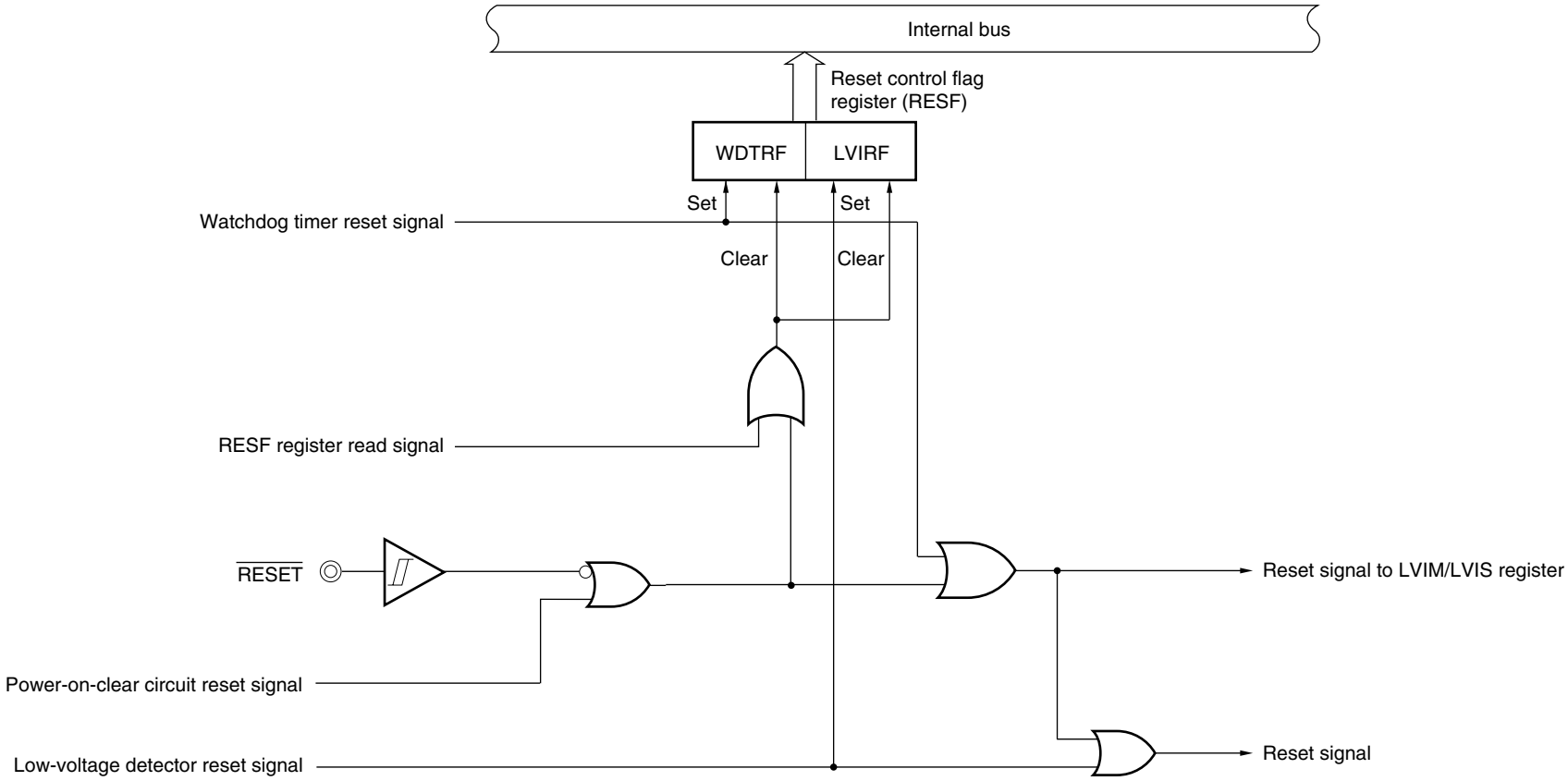
A reset is applied when a low level is input to the $\overline{\text{RESET}}$ pin, the watchdog timer overflows, or by POC and LVI circuit voltage detection, and each item of hardware is set to the status shown in Tables 20-1 and 20-2. Each pin is high impedance during reset signal generation or during the oscillation stabilization time just after a reset release.

When a low level is input to the $\overline{\text{RESET}}$ pin, the device is reset. It is released from the reset status when a high level is input to the $\overline{\text{RESET}}$ pin and program execution is started with the internal high-speed oscillation clock after reset processing. A reset by the watchdog timer is automatically released, and program execution starts using the internal high-speed oscillation clock (refer to **Figures 20-2** to **20-4**) after reset processing. Reset by POC and LVI circuit power supply detection is automatically released when $V_{DD} \geq V_{POR}$ or $V_{DD} \geq V_{LVI}$ after the reset, and program execution starts using the internal high-speed oscillation clock (refer to **CHAPTER 21 POWER-ON-CLEAR CIRCUIT** and **CHAPTER 22 LOW-VOLTAGE DETECTOR**) after reset processing.

**Cautions 1. For an external reset, input a low level for 10 μs or more to the $\overline{\text{RESET}}$ pin.**
**(If an external reset is effected upon power application, the period during which the supply voltage is outside the operating range ($V_{DD}$ < 1.8 V) is not counted in the 10 μs. However, the low-level input may be continued before POC is released.)**
**2. During reset signal generation, the X1 clock, XT1 clock[Note], internal high-speed oscillation clock, and internal low-speed oscillation clock stop oscillating. External main system clock input and external subsystem clock[Note] input become invalid.**
**3. When the STOP mode is released by a reset, the RAM contents in the STOP mode are held during reset input. However, because SFR is initialized, the port pins become high-impedance.**

**Note** 78K0/KC2-L only

**Figure 20-1. Block Diagram of Reset Function**




**Caution An LVI circuit internal reset does not reset the LVI circuit.**

**Remarks 1.** LVIM: Low-voltage detection register

**2.** LVIS: Low-voltage detection level selection register

**Figure 20-2. Timing of Reset by $\overline{\text{RESET}}$ Input**

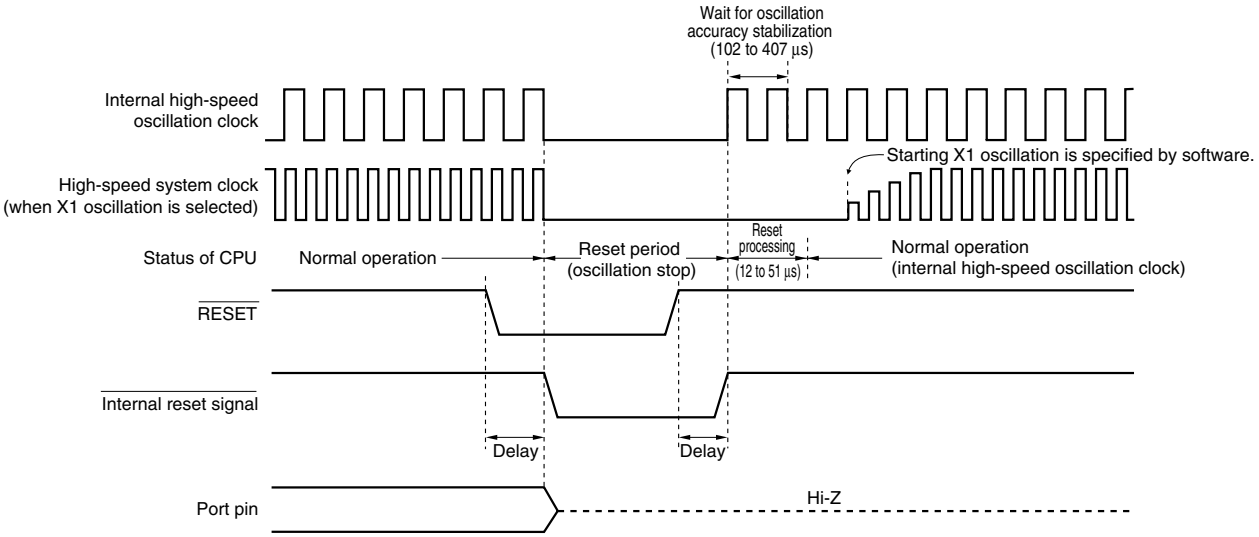
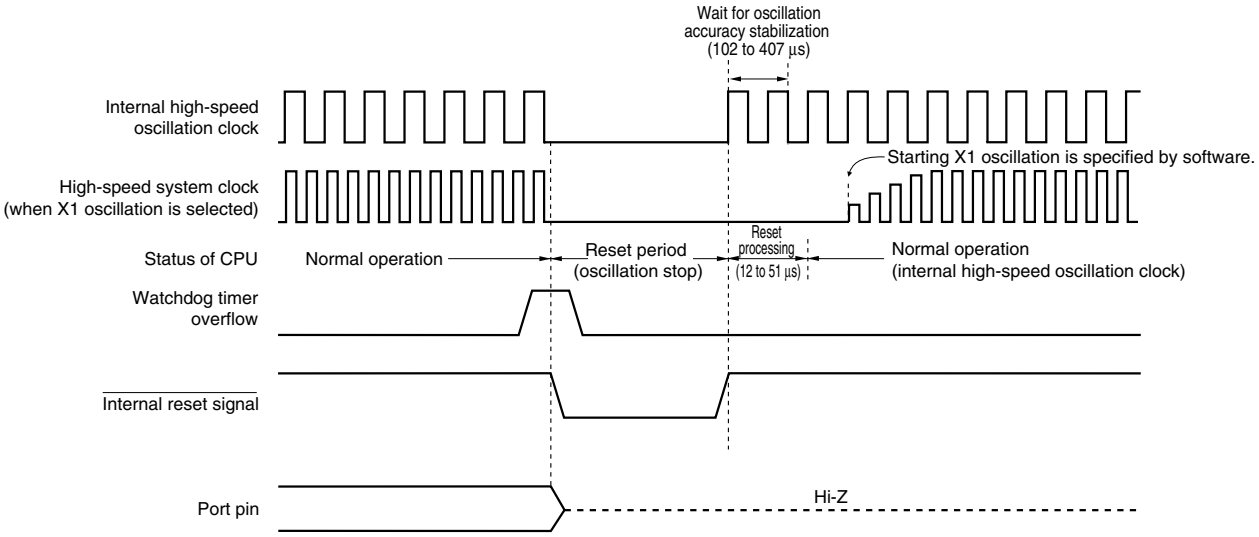
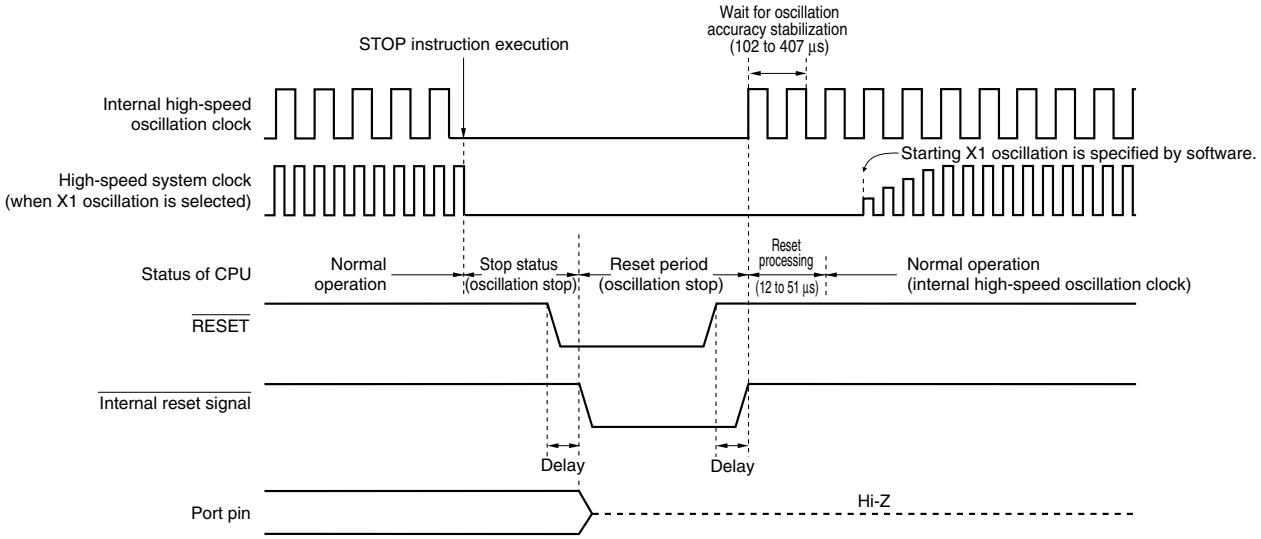



**Figure 20-3. Timing of Reset Due to Watchdog Timer Overflow**




**Caution A watchdog timer internal reset resets the watchdog timer.**

**Figure 20-4. Timing of Reset in STOP Mode by $\overline{\text{RESET}}$ Input**




**Remark** For the reset timing of the power-on-clear circuit and low-voltage detector, refer to **CHAPTER 21 POWER-ON-CLEAR CIRCUIT** and **CHAPTER 22 LOW-VOLTAGE DETECTOR**.

**Table 20-1. Operation Statuses During Reset Period**

| Item | | | During Reset Period |
|---|---|---|---|
| System clock | | | Clock supply to the CPU is stopped. |
| | Main system clock | $f_{IH}$ | Operation stopped |
| | | $f_X$ | Operation stopped (X1 and X2 pins are input port mode) |
| | | $f_{EXCLK}$ | Clock input invalid (EXCLK pin is input port mode) |
| | Subsystem clock | $f_{XT}$ | Operation stopped (XT1 and XT2 pins are input port mode) |
| | | $f_{EXCLKS}$ | Clock input invalid (EXCLKS pin is input port mode) |
| | $f_{IL}$ | | Operation stopped |
| CPU | | | |
| Flash memory | | | |
| RAM | | | Operation stopped (The value, however, is retained when the voltage is at least the power-onclear detection voltage.) |
| Port (latch) | | | Operation stopped |
| 16-bit timer/event counter 00 | | | |
| 8-bit timer/event counter | | 50 | |
| | | 51 | |
| 8-bit timer | | H0 | |
| | | H1 | |
| Real-time counter (RTC) | | | |
| Watchdog timer | | | |
| Clock output | | | |
| A/D converter | | | |
| Operational amplifier 0 (AMP0, PGA) | | | |
| Operational amplifier 1 (AMP1) | | | |
| Serial interface | | UART6 | |
| | | CSI10 | |
| | | CSI11 | |
| | | IICA | |
| External interrupt | | | |
| Key interrupt | | | |
| Power-on-clear function | | | Operable |
| Low-voltage detection function | | | Operation stopped (however, operation continues at LVI reset) |
| On-chip debug function | | | Operation stopped |

**Remarks 1.** $f_{IH}$: Internal high-speed oscillation clock, $f_X$: X1 clock
$f_{EXCLK}$: External main system clock, $f_{XT}$: XT1 clock
$f_{EXCLKS}$: External subsystem clock, $f_{IL}$: Internal low-speed oscillation clock

**2.** The functions mounted depend on the product. Refer to **1.4 Block Diagram** and **1.5 Outline of Functions**.

**Table 20-2. Hardware Statuses After Reset Acknowledgment (1/4)**

| Hardware | | After Reset Acknowledgment[Note 1] |
|---|---|---|
| Program counter (PC) | | The contents of the reset vector table (0000H, 0001H) are set. |
| Stack pointer (SP) | | Undefined |
| Program status word (PSW) | | 02H |
| RAM | Data memory | Undefined[Note 2] |
| | General-purpose registers | Undefined[Note 2] |
| Port registers 0 to 4, 6, 7, 12 (P0 to P4, P6, P7, P12) (output latches) | | 00H |
| Port mode registers 0 to 4, 6, 7, 12 (PM0 to PM4, PM6, PM7, PM12) | | FFH |
| Pull-up resistor option registers 0, 1, 3, 4, 6, 7 (PU0, PU1, PU3, PU4, PU6, PU7) | | 00H |
| Pull-up resistor option register 12 (PU12) | | 20H |
| Port input mode register 6 (PIM6) | | 00H |
| Port output mode register 6 (POM6) | | 00H |
| Reset pin mode register (RSTMASK) | | 00H |
| Port alternate switch control register (MUXSEL) | | 00H |
| Internal memory size switching register (IMS) | | CFH[Note 3] |

**Notes 1.** During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

**2.** When a reset is executed in the standby mode, the pre-reset status is held even after reset.

**3.** Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated below after release of reset.

| | Products | | | | IMS | ROM Capacity | Internal High-Speed RAM Capacity |
|---|---|---|---|---|---|---|---|
| | 78K0/KY2-L | 78K0/KA2-L | 78K0/KB2-L | 78K0/KC2-L | | | |
| | μPD78F0550, 78F0555 | μPD78F0560, 78F0565 | – | – | 61H | 4 KB | 384 bytes |
| | μPD78F0551, 78F0556 | μPD78F0561, 78F0566 | μPD78F0571, 78F0576 | μPD78F0581, 78F0586 | 42H | 8 KB | 512 bytes |
| | μPD78F0552, 78F0557 | μPD78F0562, 78F0567 | μPD78F0572, 78F0577 | μPD78F0582, 78F0587 | 04H | 16 KB | 768 bytes |
| <R> | – | – | – | μPD78F0584, 78F0589 | C6H | 24 KB | 1 KB |
| | – | – | μPD78F0573, 78F0578 | μPD78F0583, 78F0588 | C8H | 32 KB | 1 KB |

**Remark** The special function registers (SFRs) mounted depend on the product. Refer to **3.2.3 Special function registers (SFRs)**.

**Table 20-2. Hardware Statuses After Reset Acknowledgment (2/4)**

| Hardware | | Status After Reset Acknowledgment[Note 1] |
|---|---|---|
| Clock operation mode select register (OSCCTL) | | 00H |
| Processor clock control register (PCC) | | 01H |
| Internal oscillation mode register (RCM) | | 80H |
| Main OSC control register (MOC) | | 80H |
| Main clock mode register (MCM) | | 00H |
| Oscillation stabilization time counter status register (OSTC) | | 00H |
| Oscillation stabilization time select register (OSTS) | | 05H |
| Peripheral enable register 0 (PER0) | | 00H |
| 16-bit timer/event counter 00 | Timer counter 00 (TM00) | 0000H |
| | Capture/compare registers 000, 010 (CR000, CR010) | 0000H |
| | Mode control register 00 (TMC00) | 00H |
| | Prescaler mode register 00 (PRM00) | 00H |
| | Capture/compare control register 00 (CRC00) | 00H |
| | Timer output control register 00 (TOC00) | 00H |
| 8-bit timer/event counters 50, 51 | Timer counters 50, 51 (TM50, TM51) | 00H |
| | Compare registers 50, 51 (CR50, CR51) | 00H |
| | Timer clock selection registers 50, 51 (TCL50, TCL51) | 00H |
| | Mode control registers 50, 51 (TMC50, TMC51) | 00H |
| 8-bit timers H0, H1 | Compare registers 00, 10, 01, 11 (CMP00, CMP10, CMP01, CMP11) | 00H |
| | Mode registers (TMHMD0, TMHMD1) | 00H |
| | Carrier control register 1 (TMCYC1)[Note 2] | 00H |
| Real-time counter | Sub-count register (RSUBC) | 0000H |
| | Second count register (SEC) | 00H |
| | Minute count register (MIN) | 00H |
| | Hour count register (HOUR) | 12H |
| | Week count register (WEEK) | 00H |
| | Day count register (DAY) | 01H |
| | Month count register (MONTH) | 01H |
| | Year count register (YEAR) | 00H |
| | Watch error correction register (SUBCUD) | 00H |
| | Alarm minute register (ALARMWM) | 00H |
| | Alarm hour register (ALARMWH) | 12H |
| | Alarm week register (ALARMWW) | 00H |
| | Control register 0 (RTCC0) | 00H |
| | Control register 1 (RTCC1) | 00H |
| | Control register 2 (RTCC2) | 00H |

**Notes 1.** During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

**2.** 8-bit timer H1 only.

**Remark** The special function registers (SFRs) mounted depend on the product. Refer to **3.2.3 Special function registers (SFRs)**.

**Table 20-2. Hardware Statuses After Reset Acknowledgment (3/4)**

| Hardware | | Status After Reset Acknowledgment Note 1 |
|---|---|---|
| Clock output controller | Clock output selection register (CKS) | 00H |
| Watchdog timer | Enable register (WDTE) | 1AH/9AH Note 2 |
| A/D converter | 10-bit A/D conversion result register (ADCR) | 0000H |
| | 8-bit A/D conversion result register L (ADCRL) | 00H |
| | 8-bit A/D conversion result register H (ADCRH) | 00H |
| | Mode register 0 (ADM0) | 00H |
| | Analog input channel specification register (ADS) | 00H |
| | A/D port configuration register 0 (ADPC0) | 00H |
| | A/D port configuration register 1 (ADPC1) | 07H Note 3 |
| Operational amplifier 0 (AMP0, PGA) | Operational amplifier 0 control register (AMP0M) | 00H |
| Operational amplifier 1 (AMP1) | Operational amplifier 1 control register (AMP1M) | 00H |
| Serial interface UART6 | Receive buffer register 6 (RXB6) | FFH |
| | Transmit buffer register 6 (TXB6) | FFH |
| | Asynchronous serial interface operation mode register 6 (ASIM6) | 01H |
| | Asynchronous serial interface reception error status register 6 (ASIS6) | 00H |
| | Asynchronous serial interface transmission status register 6 (ASIF6) | 00H |
| | Clock selection register 6 (CKSR6) | 00H |
| | Baud rate generator control register 6 (BRGC6) | FFH |
| | Asynchronous serial interface control register 6 (ASICL6) | 16H |
| | Input switch control register (ISC) | 00H |
| Serial interfaces CSI10, CSI11 | Transmit buffer registers 10, 11 (SOTB10, SOTB11) | 00H |
| | Serial I/O shift registers 10, 11 (SIO10, SIO11) | 00H |
| | Serial operation mode registers 10, 11 (CSIM10, CSIM11) | 00H |
| | Serial clock selection registers 10, 11 (CSIC10, CSIC11) | 00H |
| Serial interface IICA | Shift register (IICA) | 00H |
| | Status register 0 (IICS0) | 00H |
| | Flag register 0 (IICF0) | 00H |
| | Control register 0 (IICCTL0) | 00H |
| | Control register 1 (IICCTL1) | 00H |
| | Low-level width setting register (IICWL) | FFH |
| | High-level width setting register (IICWH) | FFH |
| | Slave address register 0 (SVA0) | 00H |
| Key interrupt | Key return mode register (KRM) | 00H |

**Notes 1.** During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

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

<R> **3.** For the 78K0/KA2-L (32-pin products), cleared to 00H.

**Remark** The special function registers (SFRs) mounted depend on the product. Refer to **3.2.3 Special function registers (SFRs)**.

**Table 20-2. Hardware Statuses After Reset Acknowledgment (4/4)**

| Hardware | | Status After Reset Acknowledgment[Note 1] |
|---|---|---|
| Reset function | Reset control flag register (RESF) | 00H[Note 2] |
| Low-voltage detector | Low-voltage detection register (LVIM) | 00H[Note 2] |
| | Low-voltage detection level selection register (LVIS) | 00H[Note 2] |
| Interrupt | Request flag registers 0L, 0H, 1L, 1H (IF0L, IF0H, IF1L, IF1H) | 00H |
| | Mask flag registers 0L, 0H, 1L, 1H (MK0L, MK0H, MK1L, MK1H) | FFH |
| | Priority specification flag registers 0L, 0H, 1L, 1H (PR0L, PR0H, PR1L, PR1H) | FFH |
| | External interrupt rising edge enable registers 0, 1 (EGPCTL0, EGPCTL1) | 00H |
| | External interrupt falling edge enable registers 0, 1 (EGNCTL0, EGNCTL1) | 00H |
| Regulator | Regulator mode control register (RMC) | 00H |

**Notes 1.** During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

**2.** These values vary depending on the reset source.

<R>

| Register \ Reset Source | | $\overline{\text{RESET}}$ Input | Reset by POC | Reset by WDT | Reset by LVI (Except Reset by LVI Default Start Function) | Reset by LVI Default Start Function |
|---|---|---|---|---|---|---|
| RESF | WDTRF flag | Cleared (0) | Cleared (0) | Set (1) | Held | Cleared (0) |
| | LVIRF flag | | | Held | Set (1) | |
| LVIM | | Cleared (00H) | Cleared (00H) | Cleared (00H) | Held | Cleared (00H) |
| LVIS | | | | | | |

**Remark** The special function registers (SFRs) mounted depend on the product. Refer to **3.2.3 Special function registers (SFRs)**.

## 20.1 Register for Confirming Reset Source

Many internal reset generation sources exist in the 78K0/Kx2-L microcontrollers. The reset control flag register (RESF) is used to store which source has generated the reset request.

RESF can be read by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input, reset by power-on-clear (POC) circuit, and reading RESF set RESF to 00H.

**Figure 20-5. Format of Reset Control Flag Register (RESF)**

Address: FFACH After reset: 00H[Note] R

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RESF | 0 | 0 | 0 | WDTRF | 0 | 0 | 0 | LVIRF |

| WDTRF | Internal reset request by watchdog timer (WDT) |
|---|---|
| 0 | Internal reset request is not generated, or RESF is cleared. |
| 1 | Internal reset request is generated. |

| LVIRF | Internal reset request by low-voltage detector (LVI) |
|---|---|
| 0 | Internal reset request is not generated, or RESF is cleared. |
| 1 | Internal reset request is generated. |

**Note** The value after reset varies depending on the reset source.

**Caution Do not read data by a 1-bit memory manipulation instruction.**

The status of RESF when a reset request is generated is shown in Table 20-3.

**Table 20-3. RESF Status When Reset Request Is Generated**

<R>

| Flag \ Reset Source | $\overline{\text{RESET}}$ Input | Reset by POC | Reset by WDT | Reset by LVI (Except Reset by LVI Default Start Function) | Reset by LVI Default Start Function |
|---|---|---|---|---|---|
| WDTRF | Cleared (0) | Cleared (0) | Set (1) | Held | Cleared (0) |
| LVIRF | | | Held | Set (1) | |

# CHAPTER 21 POWER-ON-CLEAR CIRCUIT

## 21.1 Functions of Power-on-Clear Circuit

The power-on-clear circuit (POC) is mounted onto all 78K0/Kx2-L microcontroller products.
The power-on-clear circuit has the following functions.

- Generates internal reset signal at power on.
- The reset signal is released when the supply voltage ($V_{DD}$) exceeds POC detection voltage ($V_{POR}$ = 1.61 V ±0.09 V).

**Caution If the LVI default function enabled is set by using an option byte, the reset signal is not released until the supply voltage ($V_{DD}$) exceeds 1.91 V ±0.1 V.**

- Compares supply voltage ($V_{DD}$) and POC detection voltage ($V_{PDR}$ = 1.59 V ±0.09 V), generates internal reset signal when $V_{DD} < V_{PDR}$.

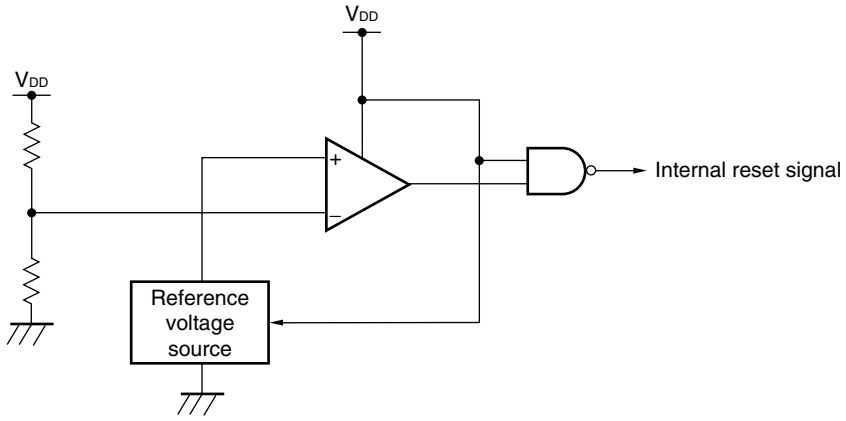
**Caution If an internal reset signal is generated in the POC circuit, the reset control flag register (RESF) is cleared to 00H.**

**Remark** The 78K0/Kx2-L microcontrollers incorporate multiple hardware functions that generate an internal reset signal. A flag that indicates the reset source is located in the reset control flag register (RESF) for when an internal reset signal is generated by the watchdog timer (WDT) and low-voltage-detector (LVI). RESF is not cleared to 00H and the flag is set to 1 when an internal reset signal is generated by WDT or LVI. For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

## 21.2 Configuration of Power-on-Clear Circuit

The block diagram of the power-on-clear circuit is shown in Figure 21-1.

**Figure 21-1. Block Diagram of Power-on-Clear Circuit**




## 21.3 Operation of Power-on-Clear Circuit

- An internal reset signal is generated on power application. When the supply voltage ($V_{DD}$) exceeds POC detection voltage ($V_{POR}$ = 1.61 V ±0.09 V), the reset status is released.

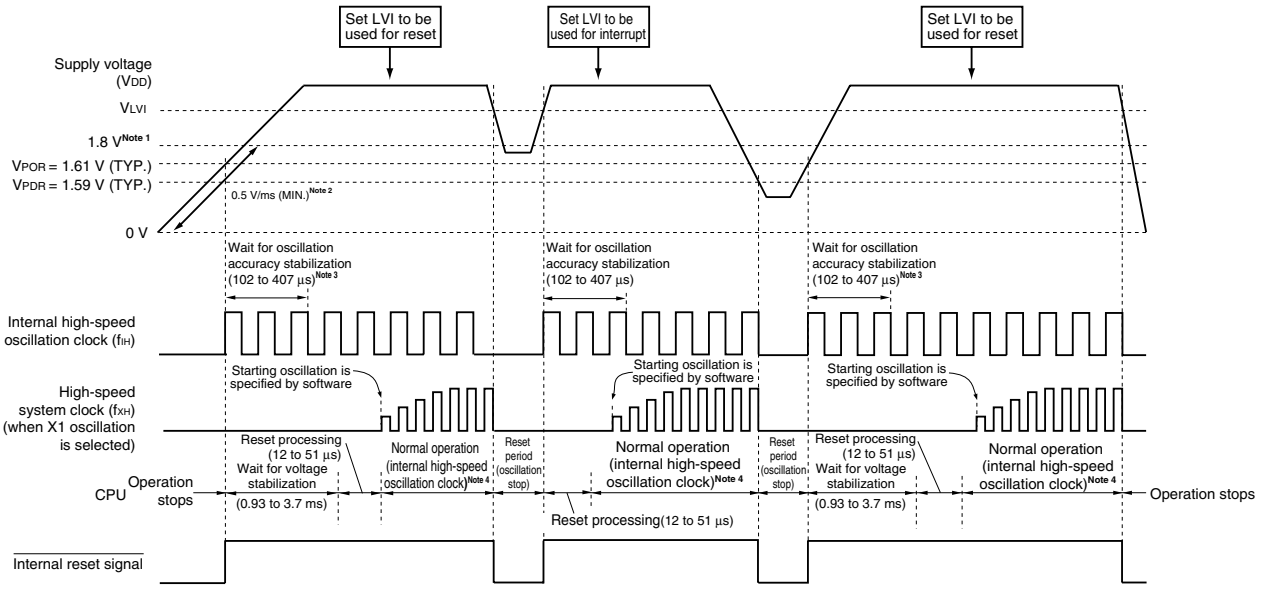
**Caution If the LVI default function enabled is set by using an option byte, the reset signal is not released until the supply voltage ($V_{DD}$) exceeds 1.91 V ±0.1 V.**

- The supply voltage ($V_{DD}$) and POC detection voltage ($V_{PDR}$ = 1.59 V ±0.09 V) are compared. When $V_{DD} < V_{PDR}$, the internal reset signal is generated.

The timing of generation of the internal reset signal by the power-on-clear circuit and low-voltage detector is shown below.

**Figure 21-2. Timing of Generation of Internal Reset Signal by Power-on-Clear Circuit and Low-Voltage Detector (1/2)**

**(1) When LVI is OFF upon power application (option byte: LVISTART = 0)**




**Notes 1.** The operation guaranteed range is 1.8 V ≤ $V_{DD}$ ≤ 5.5 V. To make the state at lower than 1.8 V reset state when the supply voltage falls, use the reset function of the low-voltage detector, or input the low level to the $\overline{\text{RESET}}$ pin.

**2.** If the rate at which the voltage rises to 1.8 V after power application is slower than 0.5 V/ms (MIN.), input a low level to the $\overline{\text{RESET}}$ pin before the voltage reaches to 1.8 V.

**3.** The internal voltage stabilization wait time includes the oscillation accuracy stabilization time of the internal high-speed oscillation clock.

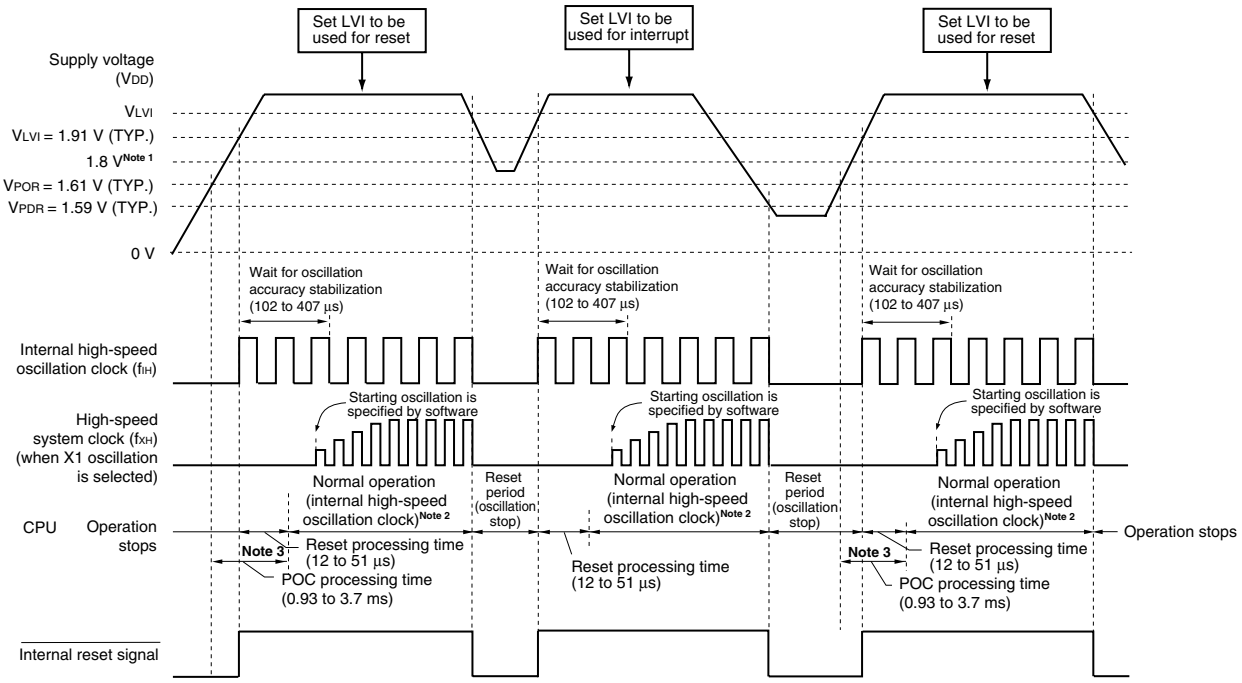
**4.** The internal high-speed oscillation clock, high-speed system clock or subsystem clock can be selected as the CPU clock. To use the X1 clock, use the OSTC register to confirm the lapse of the oscillation stabilization time. To use the XT1 clock, use the timer function for confirmation of the lapse of the stabilization time.

**Caution Set the low-voltage detector by software after the reset status is released (refer to CHAPTER 22 LOW-VOLTAGE DETECTOR).**

**Remark** $V_{LVI}$: LVI detection voltage
$V_{POR}$: POC power supply rise detection voltage
$V_{PDR}$: POC power supply fall detection voltage

<R> **Figure 21-2. Timing of Generation of Internal Reset Signal by Power-on-Clear Circuit and Low-Voltage Detector (2/2)**

**(2) When LVI is ON upon power application (option byte: LVISTART = 1)**




**Notes 1.** The operation guaranteed range is 1.8 V ≤ $V_{DD}$ ≤ 5.5 V. To make the state at lower than 1.8 V reset state when the supply voltage falls, use the reset function of the low-voltage detector, or input the low level to the $\overline{\text{RESET}}$ pin.

**2.** The internal high-speed oscillation clock, high-speed system clock or subsystem clock can be selected as the CPU clock. To use the X1 clock, use the OSTC register to confirm the lapse of the oscillation stabilization time. To use the XT1 clock, use the timer function for confirmation of the lapse of the stabilization time.

**3.** The following times are required between reaching the POC detection voltage (1.59 V (TYP.)) and starting normal operation.

- When the time to reach 1.91 V (TYP.) from 1.59 V (TYP.) is less than 3.7 ms:
  A POC processing time of about 1.0 to 3.8 ms is required between reaching 1.59 V (TYP.) and starting normal operation.
- When the time to reach 1.91 V (TYP.) from 1.59 V (TYP.) is greater than 3.7 ms:
  A reset processing time of about 12 to 51 μs is required between reaching 1.91 V (TYP.) and starting normal operation.

**Caution Set the low-voltage detector by software after the reset status is released (refer to CHAPTER 22 LOW-VOLTAGE DETECTOR).**

**Remark** $V_{LVI}$: LVI detection voltage
$V_{POR}$: POC power supply rise detection voltage
$V_{PDR}$: POC power supply fall detection voltage

## 21.4 Cautions for Power-on-Clear Circuit

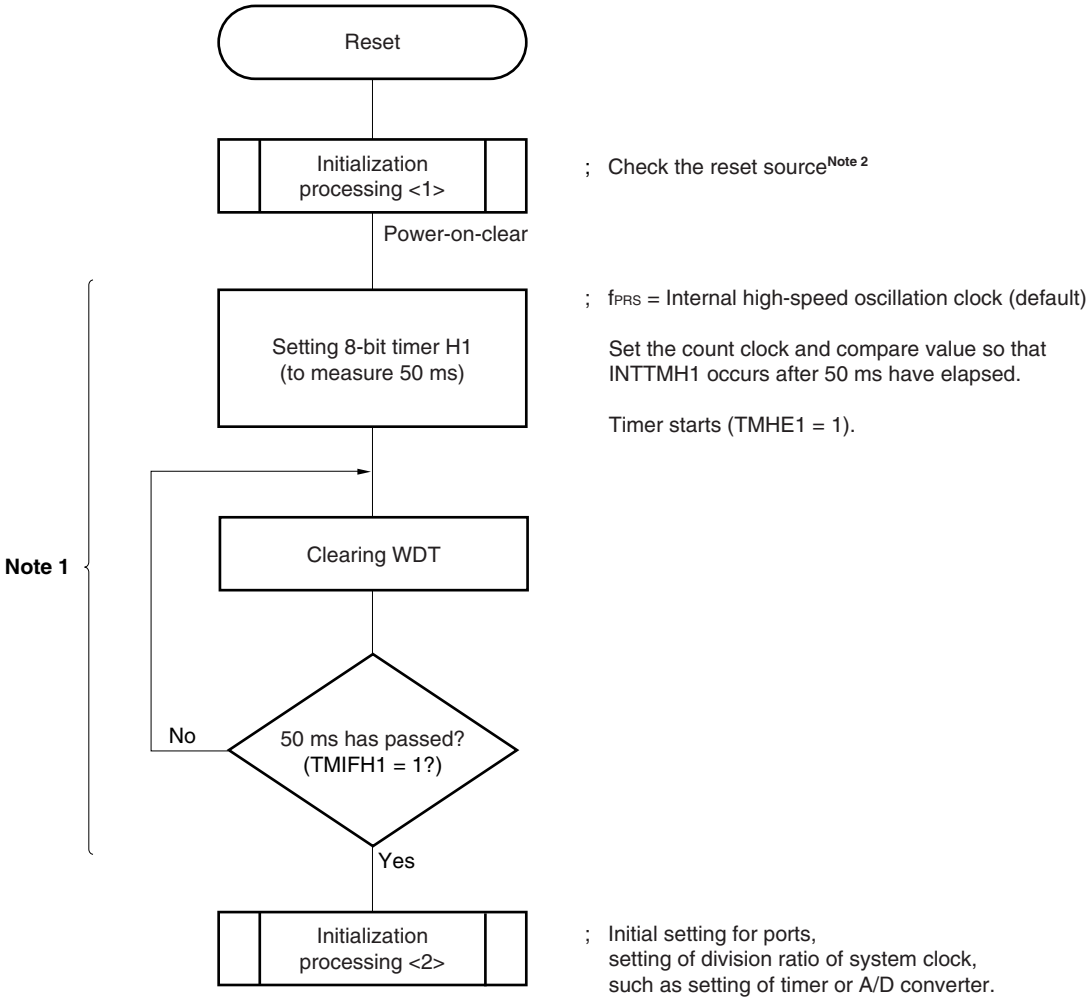
In a system where the supply voltage ($V_{DD}$) fluctuates for a certain period in the vicinity of the POC detection voltage ($V_{POR}$, $V_{PDR}$), the system may be repeatedly reset and released from the reset status. In this case, the time from release of reset to the start of the operation of the microcontroller can be arbitrarily set by taking the following action.

<Action>

After releasing the reset signal, wait for the supply voltage fluctuation period of each system by means of a software counter that uses a timer, and then initialize the ports.

**Figure 21-3. Example of Software Processing After Reset Release (1/2)**

• If supply voltage fluctuation is 50 ms or less in vicinity of POC detection voltage

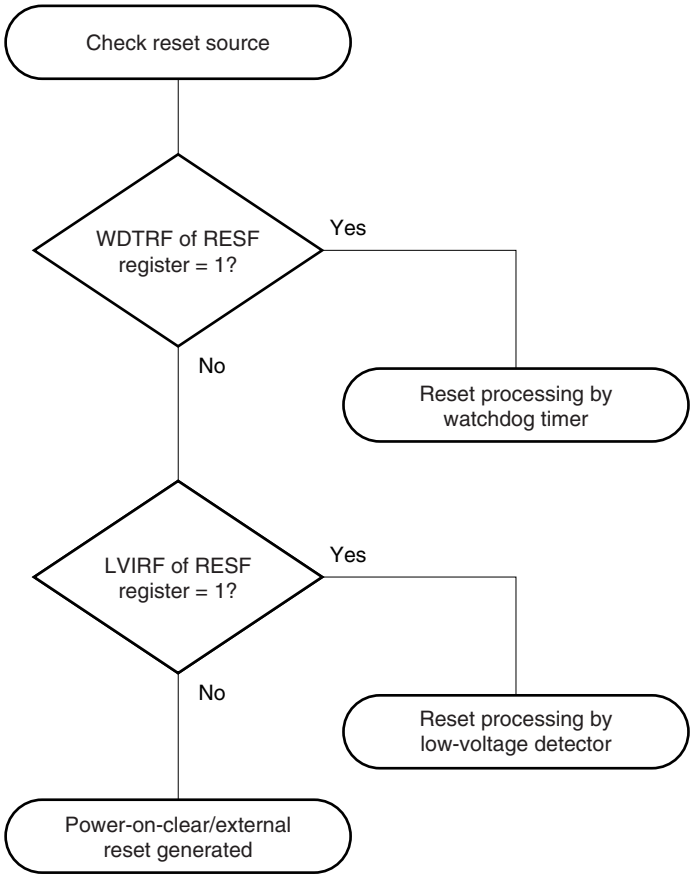



**Notes 1.** If reset is generated again during this period, initialization processing <2> is not started.

**2.** A flowchart is shown on the next page.

**Figure 21-3. Example of Software Processing After Reset Release (2/2)**

• Checking reset source



# CHAPTER 22 LOW-VOLTAGE DETECTOR

## 22.1 Functions of Low-Voltage Detector

The low-voltage detector (LVI) is mounted onto all 78K0/Kx2-L microcontroller products.
The low-voltage detector has the following functions.

- The LVI circuit compares the supply voltage ($V_{DD}$) with the LVI detection voltage ($V_{LVI}$) or the input voltage from an external input pin (EXLVI) with the LVI detection voltage ($V_{EXLVI}$ = 1.21 V ±0.1 V), and generates an internal reset or internal interrupt signal.
- The low-voltage detector (LVI) can be set to ON by an option byte by default. If it is set to ON to raise the power supply from the POC detection voltage ($V_{POR}$ = 1.61 V (TYP.)) or lower, the internal reset signal is generated when the supply voltage ($V_{DD}$) < the LVI detection voltage ($V_{LVI}$ = 1.91 V ±0.1 V). After that, the internal reset signal is generated when the supply voltage ($V_{DD}$) < the LVI detection voltage ($V_{LVI}$ = 1.91 V ±0.1 V).
- The supply voltage ($V_{DD}$) or the input voltage from the external input pin (EXLVI) can be selected to be detected by software.
- A reset or an interrupt can be selected to be generated after detection by software.
- Detection levels ($V_{LVI}$,16 levels) of supply voltage can be changed by software.
- Operable in STOP mode.

**Remark** Level detection of input voltage from external input pin (EXLVI) is available only in 78K0/KB2-L and 78K0/KC2-L.

The reset and interrupt signals are generated as follows depending on selection by software.

| Selection of Level Detection of Supply Voltage ($V_{DD}$) (LVISEL = 0) | | Selection Level Detection of Input Voltage from External Input Pin (EXLVI) (LVISEL = 1) | |
|---|---|---|---|
| Selects reset (LVIMD = 1). | Selects interrupt (LVIMD = 0). | Selects reset (LVIMD = 1). | Selects interrupt (LVIMD = 0). |
| Generates an internal reset signal when $V_{DD} < V_{LVI}$ and releases the reset signal when $V_{DD} \geq V_{LVI}$. | Generates an internal interrupt signal when $V_{DD}$ drops lower than $V_{LVI}$ ($V_{DD} < V_{LVI}$) or when $V_{DD}$ becomes $V_{LVI}$ or higher ($V_{DD} \geq V_{LVI}$). | Generates an internal reset signal when EXLVI < $V_{EXLVI}$ and releases the reset signal when EXLVI ≥ $V_{EXLVI}$. | Generates an internal interrupt signal when EXLVI drops lower than $V_{EXLVI}$ (EXLVI < $V_{EXLVI}$) or when EXLVI becomes $V_{EXLVI}$ or higher (EXLVI ≥ $V_{EXLVI}$). |

**Remark** LVISEL: Bit 2 of low-voltage detection register (LVIM)
LVIMD: Bit 1 of LVIM

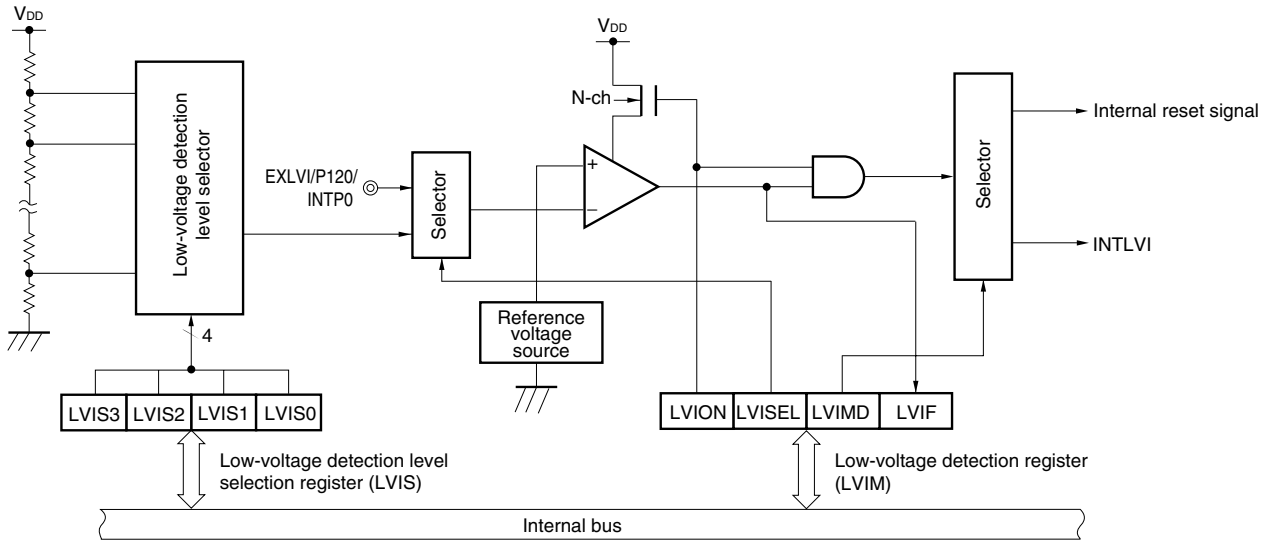
While the low-voltage detector is operating, whether the supply voltage or the input voltage from an external input pin is more than or less than the detection level can be checked by reading the low-voltage detection flag (LVIF: bit 0 of LVIM).

When the low-voltage detector is used to reset, bit 0 (LVIRF) of the reset control flag register (RESF) is set to 1 if reset occurs. For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

## 22.2 Configuration of Low-Voltage Detector

The block diagram of the low-voltage detector is shown in Figure 22-1.

**Figure 22-1. Block Diagram of Low-Voltage Detector**




**Remark** EXLVI/P120/INTP0 is mounted only on 78K0/KB2-L and 78K0/KC2-L.

## 22.3 Registers Controlling Low-Voltage Detector

The low-voltage detector is controlled by the following registers.

- Low-voltage detection register (LVIM)
- Low-voltage detection level select register (LVIS)
- Port mode register 12 (PM12)

### (1) Low-voltage detection register (LVIM)

This register sets low-voltage detection and the operation mode.

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

The generation of a reset signal other than an LVI reset clears this register to 00H.

**Figure 22-2. Format of Low-Voltage Detection Register (LVIM)**

Address: FFBEH After reset: 00H[Note 1] R/W[Note 2]

| Symbol | <7> | 6 | 5 | 4 | 3 | <2> | <1> | <0> |
|---|---|---|---|---|---|---|---|---|
| LVIM | LVION | 0 | 0 | 0 | 0 | LVISEL[Note 5] | LVIMD | LVIF |

| LVION[Notes 3, 4] | Enables low-voltage detection operation |
|---|---|
| 0 | Disables operation |
| 1 | Enables operation |

| LVISEL[Notes 3, 5] | Voltage detection selection |
|---|---|
| 0 | Detects level of supply voltage ($V_{DD}$) |
| 1 | Detects level of input voltage from external input pin (EXLVI) |

| LVIMD[Note 3] | Low-voltage detection operation mode (interrupt/reset) selection |
|---|---|
| 0 | • LVISEL = 0: Generates an internal interrupt signal when the supply voltage ($V_{DD}$) drops lower than the LVI detection voltage ($V_{LVI}$) ($V_{DD} < V_{LVI}$) or when $V_{DD}$ becomes $V_{LVI}$ or higher ($V_{DD} \geq V_{LVI}$).<br>• LVISEL = 1: Generates an interrupt signal when the input voltage from an external input pin (EXLVI) drops lower than the LVI detection voltage ($V_{EXLVI}$) (EXLVI < $V_{EXLVI}$) or when EXLVI becomes $V_{EXLVI}$ or higher (EXLVI ≥ $V_{EXLVI}$). |
| 1 | • LVISEL = 0: Generates an internal reset signal when the supply voltage ($V_{DD}$) < the LVI detection voltage ($V_{LVI}$) and releases the reset signal when $V_{DD} \geq V_{LVI}$.<br>• LVISEL = 1: Generates an internal reset signal when the input voltage from an external input pin (EXLVI) < the LVI detection voltage ($V_{EXLVI}$) and releases the reset signal when EXLVI ≥ $V_{EXLVI}$. |

| LVIF | Low-voltage detection flag |
|---|---|
| 0 | • LVISEL = 0: Supply voltage ($V_{DD}$) ≥ LVI detection voltage ($V_{LVI}$), or when LVI operation is disabled<br>• LVISEL = 1: Input voltage from external input pin (EXLVI) ≥ LVI detection voltage ($V_{EXLVI}$), or when LVI operation is disabled |
| 1 | • LVISEL = 0: Supply voltage ($V_{DD}$) < LVI detection voltage ($V_{LVI}$)<br>• LVISEL = 1: Input voltage from external input pin (EXLVI) < LVI detection voltage ($V_{EXLVI}$) |

**Notes 1.** The reset value changes depending on the reset source and the setting of the option byte.

<R> This register is not cleared (00H) by LVI resets (except resets by the LVI default start function). The value of this register is reset to “00H” by other resets.

**2.** Bit 0 is read-only.

**3.** LVION, LVIMD, and LVISEL are cleared to 0 in the case of a reset other than an LVI reset. These are not cleared to 0 in the case of an LVI reset.

**4.** When LVION is set to 1, operation of the comparator in the LVI circuit is started. Use software to wait for an operation stabilization time (10 $\mu$s (MAX.)) from when LVION is set to 1 until operation is stabilized. After the operation stabilizes, an external input (minimum pulse width: 200 $\mu$s) of 200 $\mu$s or more is required until LVIF is set (1) after the voltage drops to the LVI detection voltage or less.

**5.** 78K0/KB2-L and 78K0/KC2-L only.

**Cautions 1. To stop LVI, follow either of the procedures below.**

- **When using 8-bit memory manipulation instruction: Write 00H to LVIM.**
- **When using 1-bit memory manipulation instruction: Clear LVION to 0.**

**2. Input voltage from external input pin (EXLVI) must be EXLVI < $V_{DD}$.**

**3. If LVI operation is disabled (clears LVION) when LVI is used in interrupt mode (LVIMD = 0), LVISEL is set to 0, and the supply voltage ($V_{DD}$) is less than or equal to the detection voltage ($V_{LVI}$), or when LVI is used in interrupt mode (LVIMD = 0), LVISEL is set to 1, and input voltage of external input pin (EXLVI) is less than or equal to the detection voltage ($V_{EXLVI}$)), an interrupt request signal (INTLVI) is generated and LVIIF may be set to 1.**

**4. For 78K0/KY2-L and 78K0/KA2-L, be sure to clear bit 2 to 0.**

**(2) Low-voltage detection level select register (LVIS)**

This register selects the low-voltage detection level.

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

Reset signal generation input sets this register to 00H.

**Figure 22-3. Format of Low-Voltage Detection Level Select Register (LVIS)**

Address: FFBFH After reset: 00H[Note] R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| LVIS | 0 | 0 | 0 | 0 | LVIS3 | LVIS2 | LVIS1 | LVIS0 |

| LVIS3 | LVIS2 | LVIS1 | LVIS0 | Detection level |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $V_{LVI0}$ (4.22 ±0.1 V) |
| 0 | 0 | 0 | 1 | $V_{LVI1}$ (4.07 ±0.1 V) |
| 0 | 0 | 1 | 0 | $V_{LVI2}$ (3.92 ±0.1 V) |
| 0 | 0 | 1 | 1 | $V_{LVI3}$ (3.76 ±0.1 V) |
| 0 | 1 | 0 | 0 | $V_{LVI4}$ (3.61 ±0.1 V) |
| 0 | 1 | 0 | 1 | $V_{LVI5}$ (3.45 ±0.1 V) |
| 0 | 1 | 1 | 0 | $V_{LVI6}$ (3.30 ±0.1 V) |
| 0 | 1 | 1 | 1 | $V_{LVI7}$ (3.15 ±0.1 V) |
| 1 | 0 | 0 | 0 | $V_{LVI8}$ (2.99 ±0.1 V) |
| 1 | 0 | 0 | 1 | $V_{LVI9}$ (2.84 ±0.1 V) |
| 1 | 0 | 1 | 0 | $V_{LVI10}$ (2.68 ±0.1 V) |
| 1 | 0 | 1 | 1 | $V_{LVI11}$ (2.53 ±0.1 V) |
| 1 | 1 | 0 | 0 | $V_{LVI12}$ (2.38 ±0.1 V) |
| 1 | 1 | 0 | 1 | $V_{LVI13}$ (2.22 ±0.1 V) |
| 1 | 1 | 1 | 0 | $V_{LVI14}$ (2.07 ±0.07 V) |
| 1 | 1 | 1 | 1 | $V_{LVI15}$ (1.91 ±0.1 V) |

**Note** The reset value changes depending on the reset source.

<R> If the LVIS register is reset by LVI resets (except resets by the LVI default start function), it is not reset but holds the current value. The value of this register is reset to “00H” by other resets.

**Cautions 1. Be sure to clear bits 4 to 7 to 0.**

**2. Do not change the value of LVIS during LVI operation.**

**3. When an input voltage from the external input pin (EXLVI) is detected, the detection voltage ($V_{EXLVI}$ = 1.21 V (TYP.)) is fixed. Therefore, setting of LVIS is not necessary.**

**(3) Port mode register 12 (PM12) (78K0/KB2-L and 78K0/KC2-L only)**

When using the P120/EXLVI/INTP0 pin for external low-voltage detection potential input, set PM120 to 1. At this time, the output latch of P120 may be 0 or 1.

PM12 can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Figure 22-4. Format of Port Mode Register 12 (PM12) (78K0/KB2-L and 78K0/KC2-L only)**

Address: FF2CH After reset: FFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| PM12 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | PM120 |

| PM120 | P120 pin I/O mode selection |
|---|---|
| 0 | Output mode (output buffer on) |
| 1 | Input mode (output buffer off) |

## 22.4 Operation of Low-Voltage Detector

The low-voltage detector can be used in the following two modes.

**(1) Used as reset (LVIMD = 1)**

- If LVISEL = 0, compares the supply voltage ($V_{DD}$) and LVI detection voltage ($V_{LVI}$), generates an internal reset signal when $V_{DD} < V_{LVI}$, and releases internal reset when $V_{DD} \geq V_{LVI}$.
- If LVISEL = 1, compares the input voltage from external input pin (EXLVI) and LVI detection voltage ($V_{EXLVI}$), generates an internal reset signal when EXLVI < $V_{EXLVI}$, and releases internal reset when EXLVI ≥ $V_{EXLVI}$.

<R> **Remarks 1.** The low-voltage detector (LVI) can be set to ON by an option byte by default. If it is set to ON to raise the power supply from the POC detection voltage ($V_{POR}$ = 1.61 V (TYP.)) or lower, the internal reset signal is generated when the supply voltage ($V_{DD}$) < detection voltage ($V_{LVI}$ = 1.91 V ±0.1 V).

**2.** Level detection of input voltage from external input pin (EXLVI) is available only in 78K0/KB2-L and 78K0/KC2-L.

**(2) Used as interrupt (LVIMD = 0)**

- If LVISEL = 0, compares the supply voltage ($V_{DD}$) and LVI detection voltage ($V_{LVI}$). When $V_{DD}$ drops lower than $V_{LVI}$ ($V_{DD} < V_{LVI}$) or when $V_{DD}$ becomes $V_{LVI}$ or higher ($V_{DD} \geq V_{LVI}$), generates an interrupt signal (INTLVI).
- If LVISEL = 1, compares the input voltage from external input pin (EXLVI) and LVI detection voltage ($V_{EXLVI}$ = 1.21 V ±0.1 V). When EXLVI drops lower than $V_{EXLVI}$ (EXLVI < $V_{EXLVI}$) or when EXLVI becomes $V_{EXLVI}$ or higher (EXLVI ≥ $V_{EXLVI}$), generates an interrupt signal (INTLVI).

While the low-voltage detector is operating, whether the supply voltage or the input voltage from an external input pin is more than or less than the detection level can be checked by reading the low-voltage detection flag (LVIF: bit 0 of LVIM).

**Remarks 1.** Level detection of input voltage from external input pin (EXLVI) is available only in 78K0/KB2-L and 78K0/KC2-L.

**2.** LVIMD: Bit 1 of low-voltage detection register (LVIM)
LVISEL: Bit 2 of LVIM

### 22.4.1 When used as reset

**(1) When detecting level of supply voltage ($V_{DD}$)**

(a) When LVI default start function stopped is set (LVISTART = 0)

- When starting operation

<1> Mask the LVI interrupt (LVIMK = 1).

<2> Clear bit 2 (LVISEL) of the low-voltage detection register (LVIM) to 0 (detects level of supply voltage ($V_{DD}$)) (default value).

<3> Set the LVI detection voltage using bits 3 to 0 (LVIS3 to LVIS0) of the low-voltage detection level selection register (LVIS).

<4> Set bit 7 (LVION) of LVIM to 1 (enables LVI operation).

<5> Use software to wait for an operation stabilization time (10 $\mu$s (MAX.)).

<6> Wait until it is checked that (supply voltage ($V_{DD}$) ≥ LVI detection voltage ($V_{LVI}$)) by bit 0 (LVIF) of LVIM.

<7> Set bit 1 (LVIMD) of LVIM to 1 (generates reset when the level is detected).

Figure 22-5 shows the timing of the internal reset signal generated by the low-voltage detector. The numbers in this timing chart correspond to <1> to <7> above.

**Cautions 1. Be sure to execute <1>. When LVIMK = 0, an interrupt may occur immediately after the processing in <4>.**

**2. If supply voltage ($V_{DD}$) ≥ LVI detection voltage ($V_{LVI}$) when LVIMD is set to 1, an internal reset signal is not generated.**

- When stopping operation

Either of the following procedures must be executed.

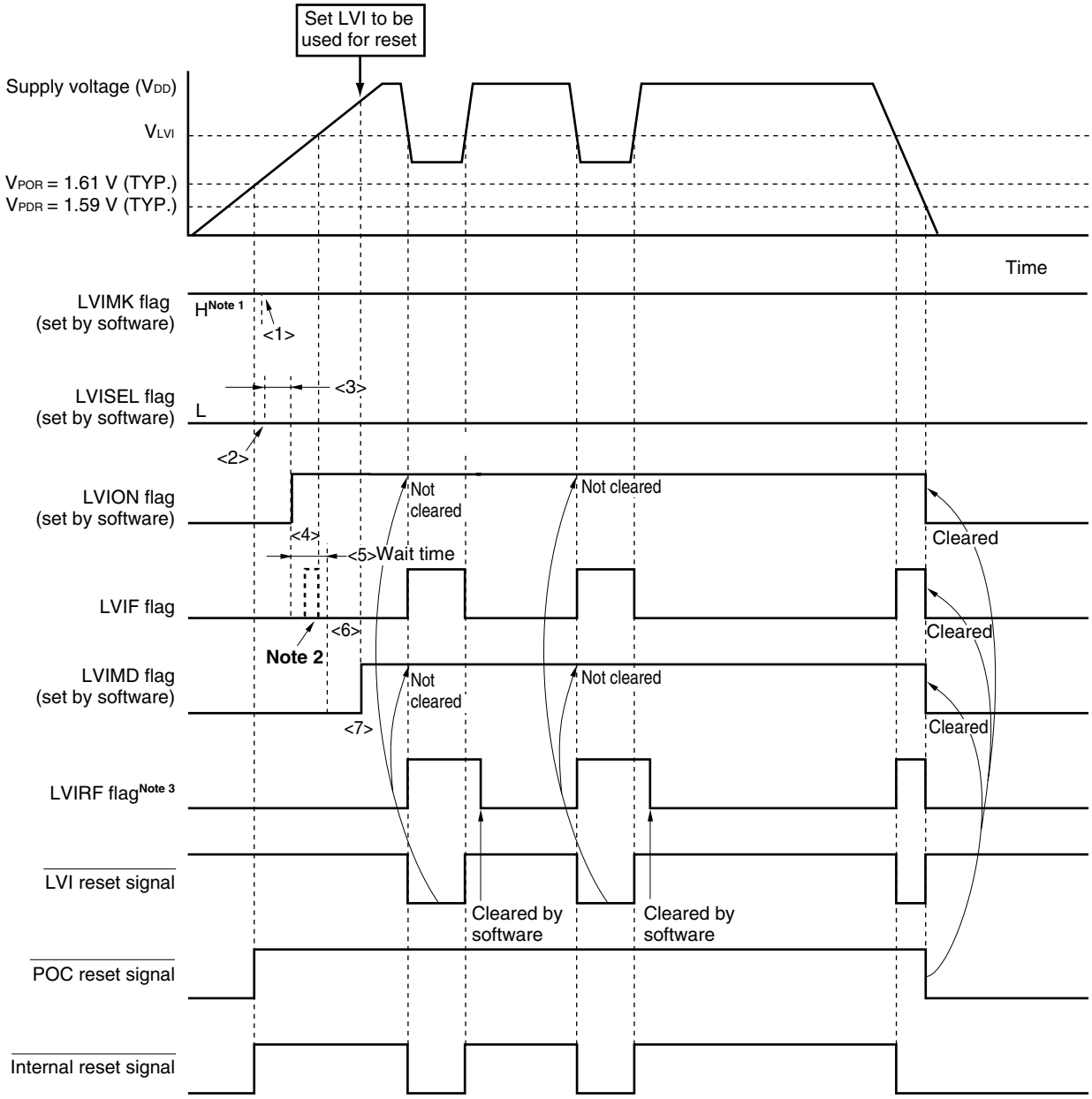
• When using 8-bit memory manipulation instruction:

Write 00H to LVIM.

• When using 1-bit memory manipulation instruction:

Clear LVIMD to 0 and then LVION to 0.

**Figure 22-5. Timing of Low-Voltage Detector Internal Reset Signal Generation (Bit: LVISEL = 0, Option Byte: LVISTART = 0)**




**Notes 1.** The LVIMK flag is set to "1" by reset signal generation.

**2.** The LVIIF flag of the interrupt request flag registers and the LVIF flag may be set (1).

**3.** LVIRF is bit 0 of the reset control flag register (RESF). For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

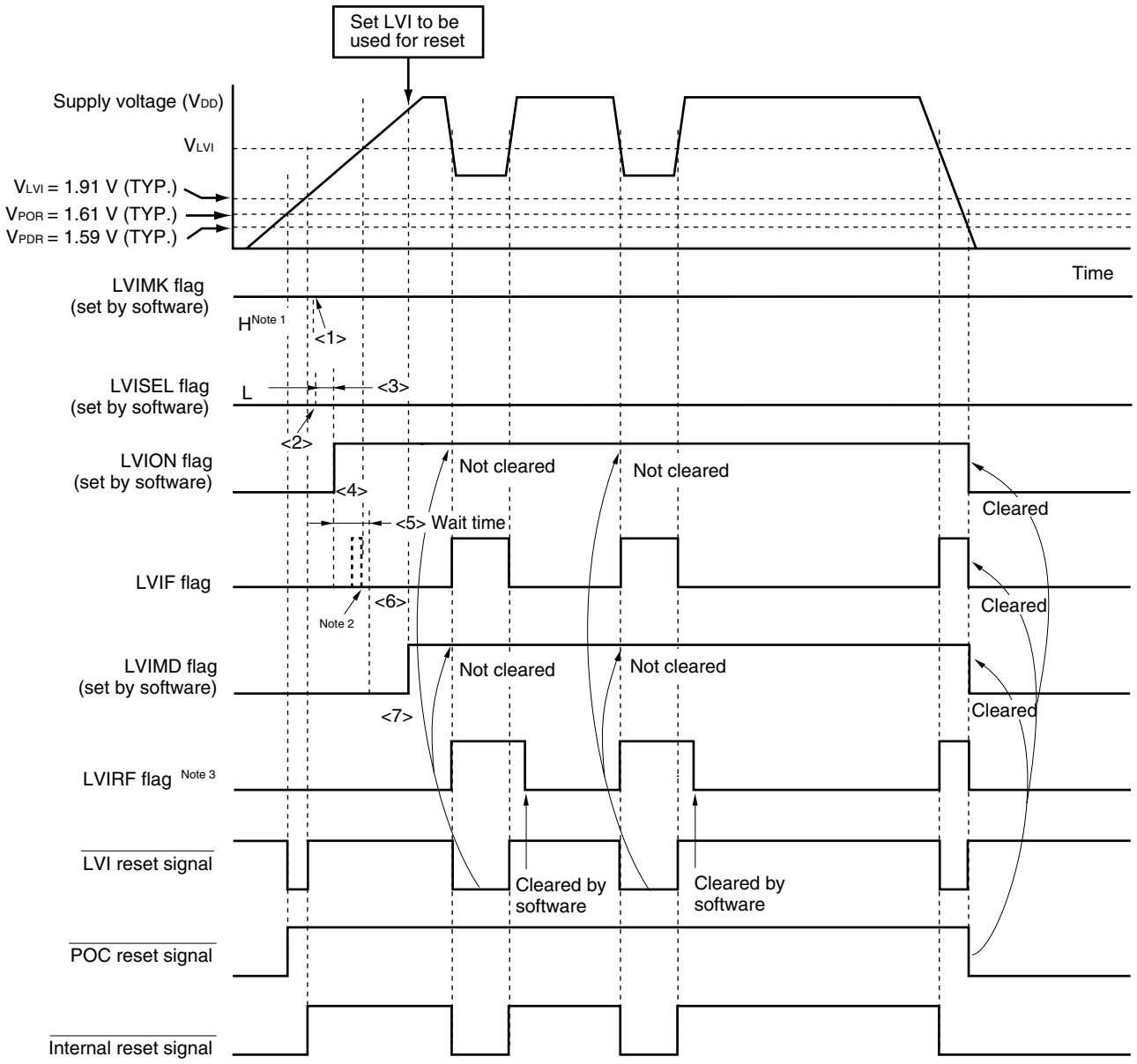
**Remarks 1.** <1> to <7> in Figure 22-5 above correspond to <1> to <7> in the description of "When starting operation" in **22.4.1 (1) (a) When LVI default start function stopped is set (LVISTART = 0).**

**2.** $V_{POR}$: POC power supply rise detection voltage
$V_{PDR}$: POC power supply fall detection voltage

<R> (b) When LVI default start function enabled is set (LVISTART = 1)

The setting when operation starts and when operation stops is the same as that described in **22.4.1 (1) When LVI default start function stopped is set (LVISTART = 0)**.

<R> **Figure 22-6. Timing of Low-Voltage Detector Internal Reset Signal Generation (Bit: LVISEL = 0, Option Byte: LVISTART = 1)**




**Notes 1.** The LVIMK flag is set to "1" by reset signal generation.

**2.** The LVIIF flag of the interrupt request flag registers and the LVIF flag may be set (1).

**3.** LVIRF is bit 0 of the reset control flag register (RESF).
For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

**Remarks 1.** <1> to <7> in Figure 22-6 above correspond to <1> to <7> in the description of "When starting operation" in **22.4.1 (1) (a) When LVI default start function stopped is set (LVISTART = 0)**.

**2.** VPOR: POC power supply rise detection voltage
VPDR: POC power supply fall detection voltage

**(2) When detecting level of input voltage from external input pin (EXLVI) (78K0/KB2-L and 78K0/KC2-L only)**

- When starting operation
  - <1> Mask the LVI interrupt (LVIMK = 1).
  - <2> Set bit 2 (LVISEL) of the low-voltage detection register (LVIM) to 1 (detects level of input voltage from external input pin (EXLVI)).
  - <3> Set bit 7 (LVION) of LVIM to 1 (enables LVI operation).
  - <4> Use software to wait for an operation stabilization time (10 $\mu$s (MAX.)).
  - <5> Wait until it is checked that (input voltage from external input pin (EXLVI) ≥ LVI detection voltage ($V_{EXLVI}$ = 1.21 V (TYP.))) by bit 0 (LVIF) of LVIM.
  - <6> Set bit 1 (LVIMD) of LVIM to 1 (generates reset signal when the level is detected).

  Figure 22-7 shows the timing of the internal reset signal generated by the low-voltage detector. The numbers in this timing chart correspond to <1> to <6> above.

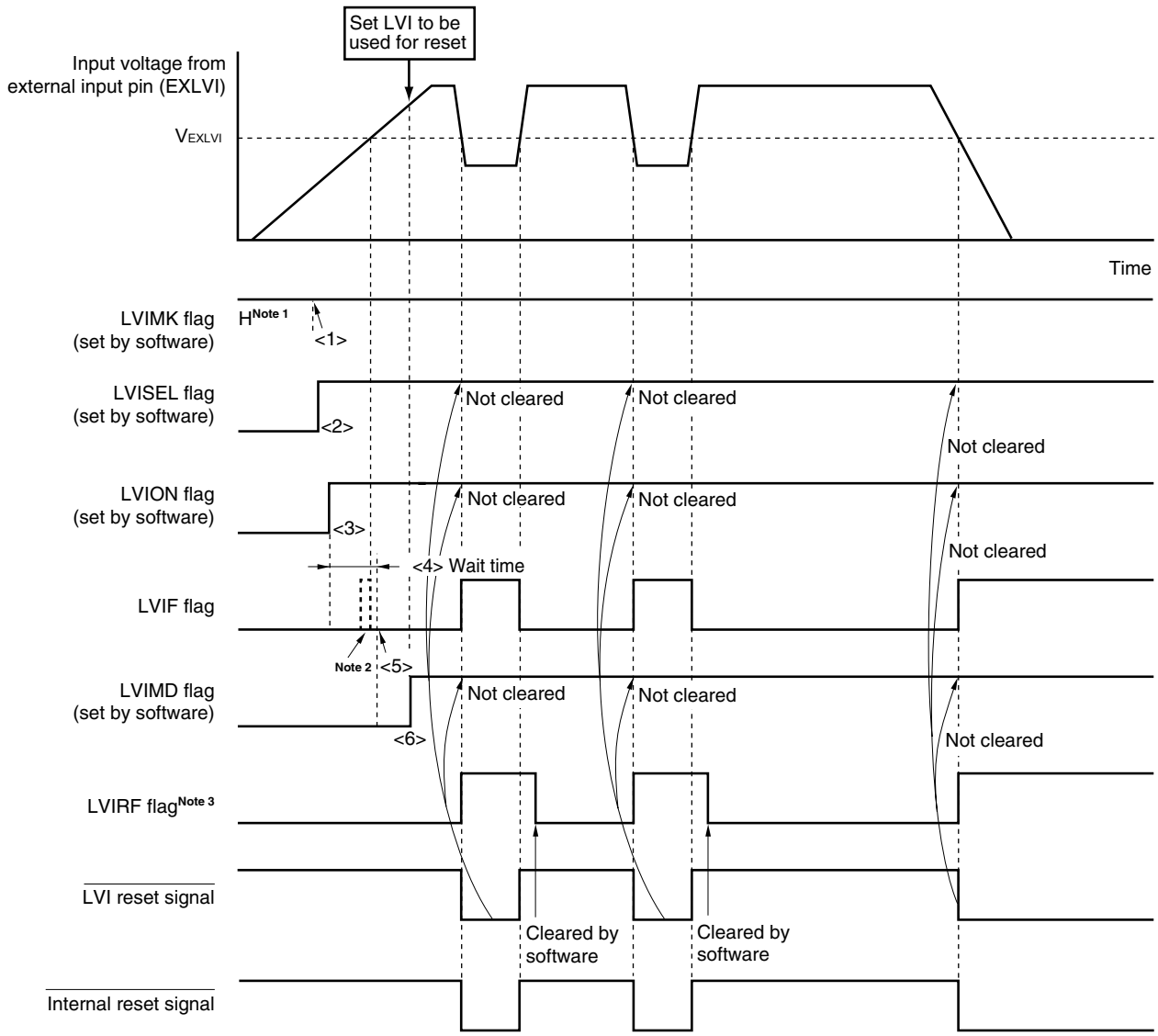
  **Cautions 1. Be sure to execute <1>. When LVIMK = 0, an interrupt may occur immediately after the processing in <3>.**
  **2. If input voltage from external input pin (EXLVI) ≥ LVI detection voltage ($V_{EXLVI}$ = 1.21 V (TYP.)) when LVIMD is set to 1, an internal reset signal is not generated.**
  **3. Input voltage from external input pin (EXLVI) must be EXLVI < $V_{DD}$.**

- When stopping operation
  Either of the following procedures must be executed.

  - When using 8-bit memory manipulation instruction:
    Write 00H to LVIM.

  - When using 1-bit memory manipulation instruction:
    Clear LVIMD to 0 and then LVION to 0.

**Figure 22-7. Timing of Low-Voltage Detector Internal Reset Signal Generation (Bit: LVISEL = 1)**




**Notes 1.** The LVIMK flag is set to "1" by reset signal generation.

**2.** The LVIIF flag of the interrupt request flag registers and the LVIF flag may be set (1).

**3.** LVIRF is bit 0 of the reset control flag register (RESF). For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

**Remark** <1> to <6> in Figure 22-7 above correspond to <1> to <6> in the description of "When starting operation" in **22.4.1 (2) When detecting level of input voltage from external input pin (EXLVI)**.

### 22.4.2 When used as interrupt

**(1) When detecting level of supply voltage ($V_{DD}$)**

(a) When LVI default start function stopped is set (LVISTART = 0)

- When starting operation

<1> Mask the LVI interrupt (LVIMK = 1).

<2> Clear bit 2 (LVISEL) of the low-voltage detection register (LVIM) to 0 (detects level of supply voltage ($V_{DD}$)) (default value).

<3> Set the LVI detection voltage using bits 3 to 0 (LVIS3 to LVIS0) of the low-voltage detection level selection register (LVIS).

<4> Clear bit 1 (LVIMD) of LVIM to 0 (generates interrupt signal when the level is detected) (default value).

<5> Set bit 7 (LVION) of LVIM to 1 (enables LVI operation).

<6> Use software to wait for an operation stabilization time (10 $\mu$s (MAX.)).

<7> Confirm that "supply voltage ($V_{DD}$) ≥ LVI detection voltage ($V_{LVI}$)" when detecting the falling edge of $V_{DD}$, or "supply voltage ($V_{DD}$) < LVI detection voltage ($V_{LVI}$)" when detecting the rising edge of $V_{DD}$, at bit 0 (LVIF) of LVIM.

<8> Clear the interrupt request flag of LVI (LVIIF) to 0.

<9> Release the interrupt mask flag of LVI (LVIMK).

<10> Execute the EI instruction (when vector interrupts are used).

Figure 22-8 shows the timing of the interrupt signal generated by the low-voltage detector. The numbers in this timing chart correspond to <1> to <9> above.

- When stopping operation

Either of the following procedures must be executed.

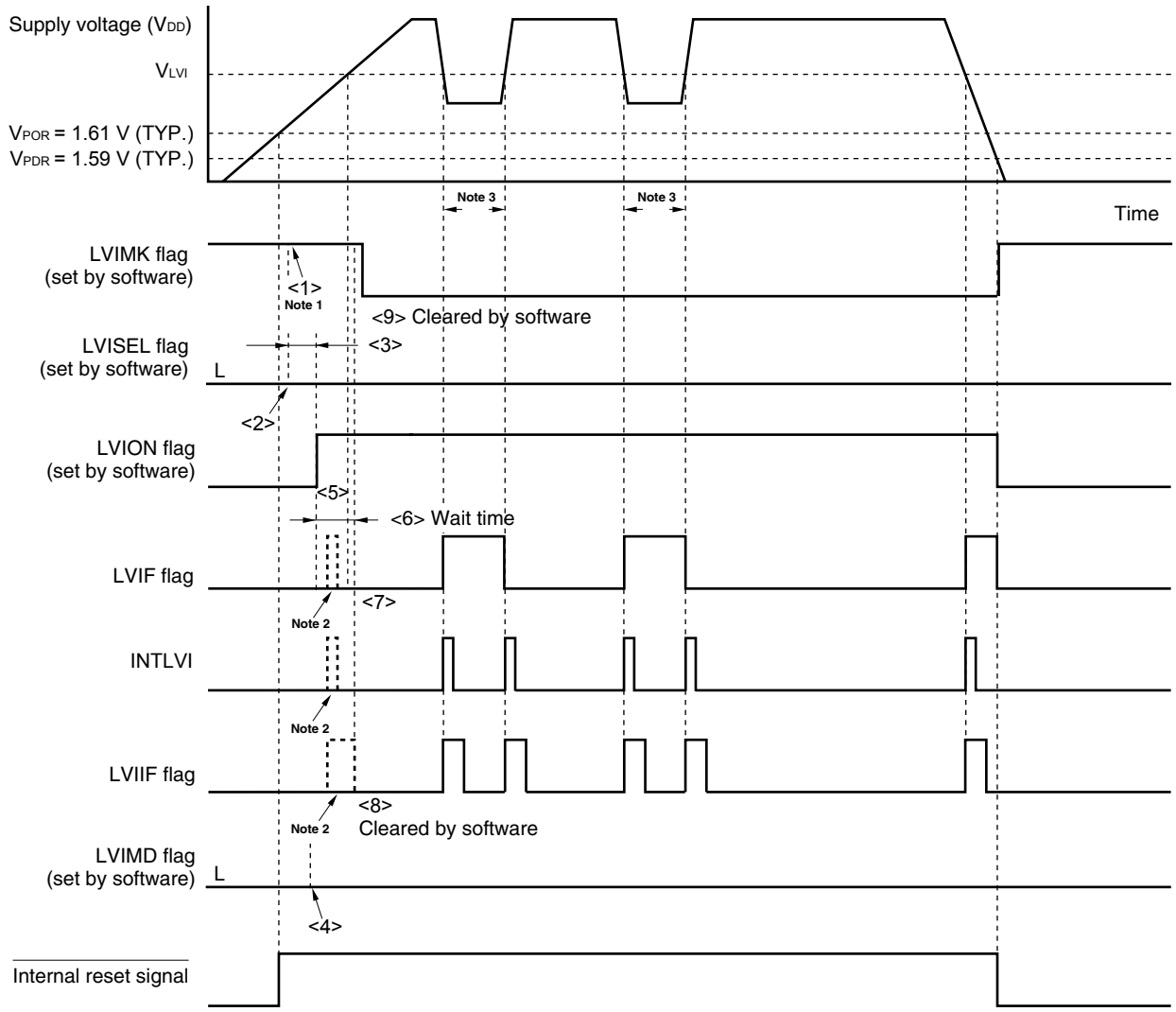
- When using 8-bit memory manipulation instruction:

Write 00H to LVIM.

- When using 1-bit memory manipulation instruction:

Clear LVION to 0.

**Figure 22-8. Timing of Low-Voltage Detector Interrupt Signal Generation (Bit: LVISEL = 0, Option Byte: LVISTART = 0)**




**Notes 1.** The LVIMK flag is set to "1" by reset signal generation.

**2.** The interrupt request signal (INTLVI) is generated and the LVIF and LVIIF flags may be set (1).

**3.** If LVI operation is disabled (clears LVION) when the supply voltage ($V_{DD}$) is less than or equal to the detection voltage ($V_{LVI}$), an interrupt request signal (INTLVI) is generated and LVIIF may be set to 1.

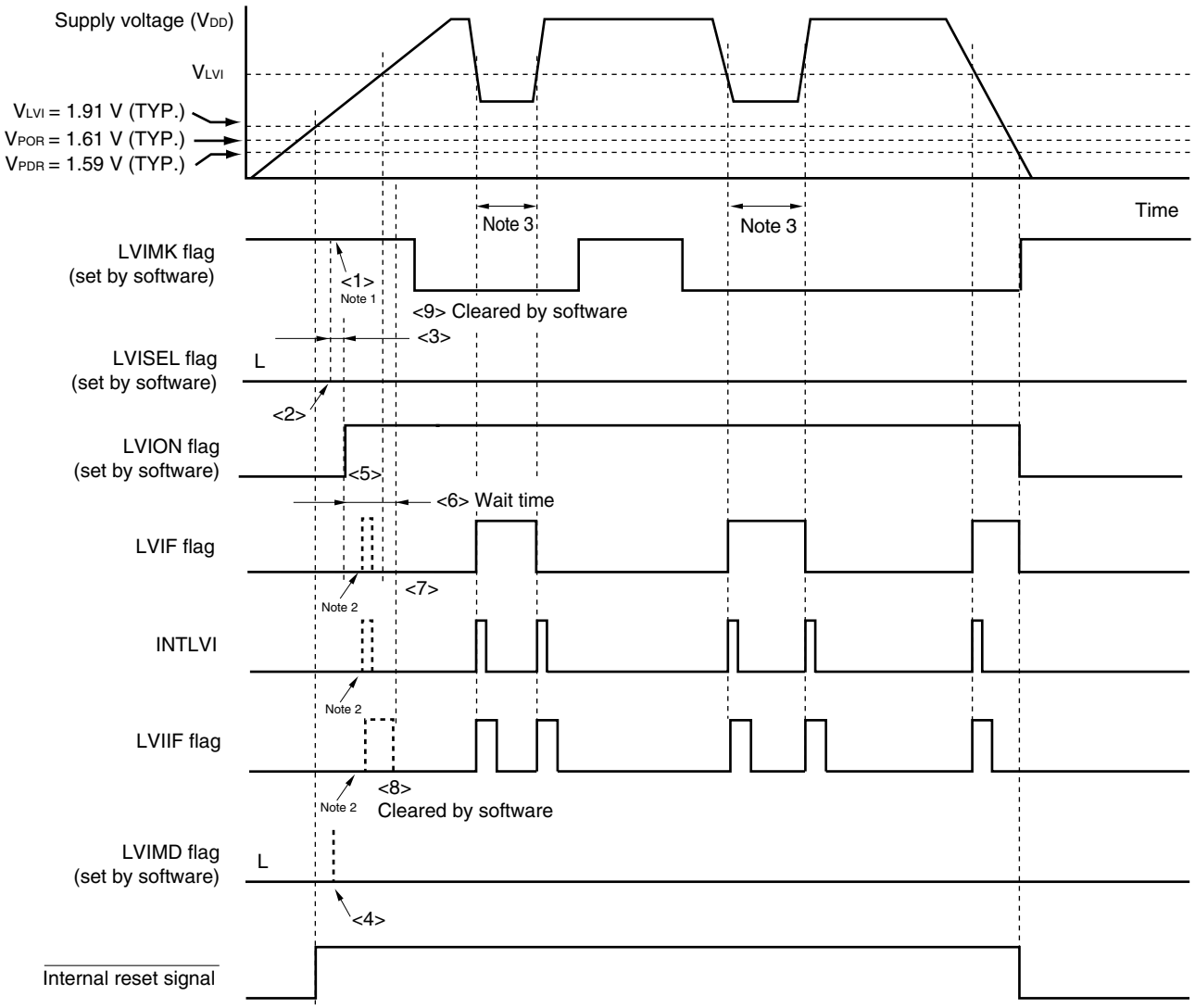
**Remarks 1.** <1> to <9> in Figure 22-8 above correspond to <1> to <9> in the description of "When starting operation" in **22.4.2 (1) (a) When LVI default start function stopped is set (LVISTART = 0).**

**2.** $V_{POR}$: POC power supply rise detection voltage
$V_{PDR}$: POC power supply fall detection voltage

<R> (b) When LVI default start function enabled is set (LVISTART = 1)

The setting when operation starts and when operation stops is the same as that described in **22.4.2 (1) When LVI default start function stopped is set (LVISTART = 0)**.

<R> **Figure 22-9. Timing of Low-Voltage Detector Interrupt Signal Generation (Bit: LVISEL = 0, Option Byte: LVISTART = 1)**




**Notes 1.** The LVIMK flag is set to "1" by reset signal generation.

**2.** The LVIIF flag of the interrupt request flag registers and the LVIF flag may be set (1).

**3.** LVIRF is bit 0 of the reset control flag register (RESF).
For details of RESF, refer to **CHAPTER 20 RESET FUNCTION**.

**Remarks 1.** <1> to <9> in Figure 22-9 above correspond to <1> to <9> in the description of "When starting operation" in **22.4.2 (1) (b) When LVI default start function enabled is set (LVISTART = 1).**

**2.** VPOR: POC power supply rise detection voltage
VPDR: POC power supply fall detection voltage

**(2) When detecting level of input voltage from external input pin (EXLVI) (78K0/KB2-L and 78K0/KC2-L only)**

- When starting operation
  - <1> Mask the LVI interrupt (LVIMK = 1).
  - <2> Set bit 2 (LVISEL) of the low-voltage detection register (LVIM) to 1 (detects level of input voltage from external input pin (EXLVI)).
  - <3> Clear bit 1 (LVIMD) of LVIM to 0 (generates interrupt signal when the level is detected) (default value).
  - <4> Set bit 7 (LVION) of LVIM to 1 (enables LVI operation).
  - <5> Use software to wait for an operation stabilization time (10 $\mu$s (MAX.)).
  - <6> Confirm that "input voltage from external input pin (EXLVI) ≥ detection voltage ($V_{EXLVI}$ = 1.21 V (TYP.))" when detecting the falling edge of EXLVI, or "input voltage from external input pin (EXLVI) < detection voltage ($V_{EXLVI}$ = 1.21 V (TYP.))" when detecting the rising edge of EXLVI, at bit 0 (LVIF) of LVIM.
  - <7> Clear the interrupt request flag of LVI (LVIIF) to 0.
  - <8> Release the interrupt mask flag of LVI (LVIMK).
  - <9> Execute the EI instruction (when vector interrupts are used).

  Figure 22-10 shows the timing of the interrupt signal generated by the low-voltage detector. The numbers in this timing chart correspond to <1> to <8> above.

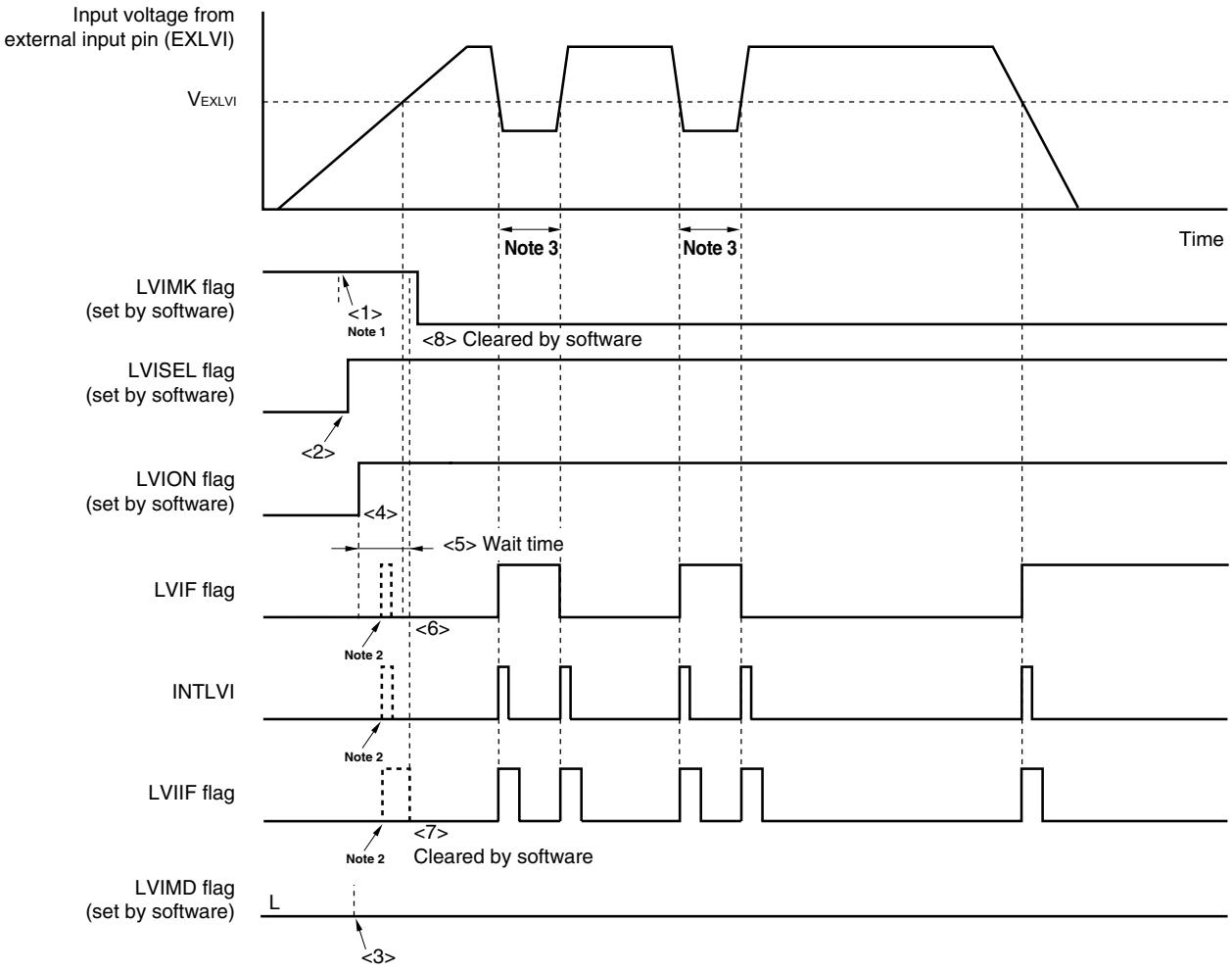
  **Caution Input voltage from external input pin (EXLVI) must be EXLVI < $V_{DD}$.**

- When stopping operation
  Either of the following procedures must be executed.

  - When using 8-bit memory manipulation instruction:
    Write 00H to LVIM.

  - When using 1-bit memory manipulation instruction:
    Clear LVION to 0.

**Figure 22-10. Timing of Low-Voltage Detector Interrupt Signal Generation (Bit: LVISEL = 1)**




**Notes 1.** The LVIMK flag is set to "1" by reset signal generation.

**2.** The interrupt request signal (INTLVI) is generated and the LVIF and LVIIF flags may be set (1).

**3.** If LVI operation is disabled when the input voltage of external input pin (EXLVI) is less than or equal to the detection voltage ($V_{EXLVI}$), an interrupt request signal (INTLVI) is generated and LVIIF may be set to 1.

**Remark** <1> to <8> in Figure 22-10 above correspond to <1> to <8> in the description of "When starting operation" in **22.4.2 (2) When detecting level of input voltage from external input pin (EXLVI)**.

## 22.5 Cautions for Low-Voltage Detector

In a system where the supply voltage ($V_{DD}$) fluctuates for a certain period in the vicinity of the LVI detection voltage ($V_{LVI}$), the operation is as follows depending on how the low-voltage detector is used.

**Operation example 1: When used as reset**

The system may be repeatedly reset and released from the reset status.

The time from reset release through microcontroller operation start can be set arbitrarily by the following action.

**<Action>**

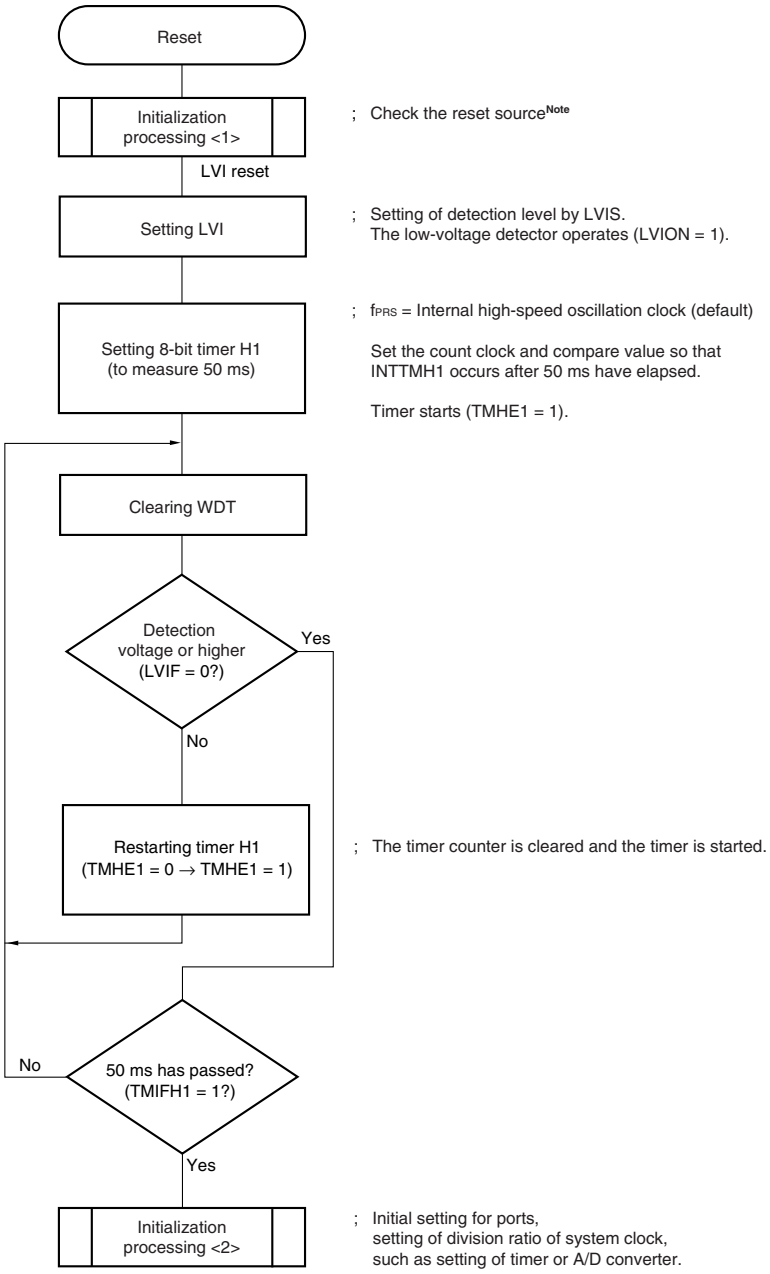
After releasing the reset signal, wait for the supply voltage fluctuation period of each system by means of a software counter that uses a timer, and then initialize the ports (refer to **Figure 22-11**).

**Remark** If bit 2 (LVISEL) of the low voltage detection register (LVIM) is set to "1", the meanings of the above words change as follows.

- Supply voltage ($V_{DD}$) → Input voltage from external input pin (EXLVI)
- Detection voltage ($V_{LVI}$) → Detection voltage ($V_{EXLVI}$ = 1.21 V)

**Figure 22-11. Example of Software Processing After Reset Release (1/2)**

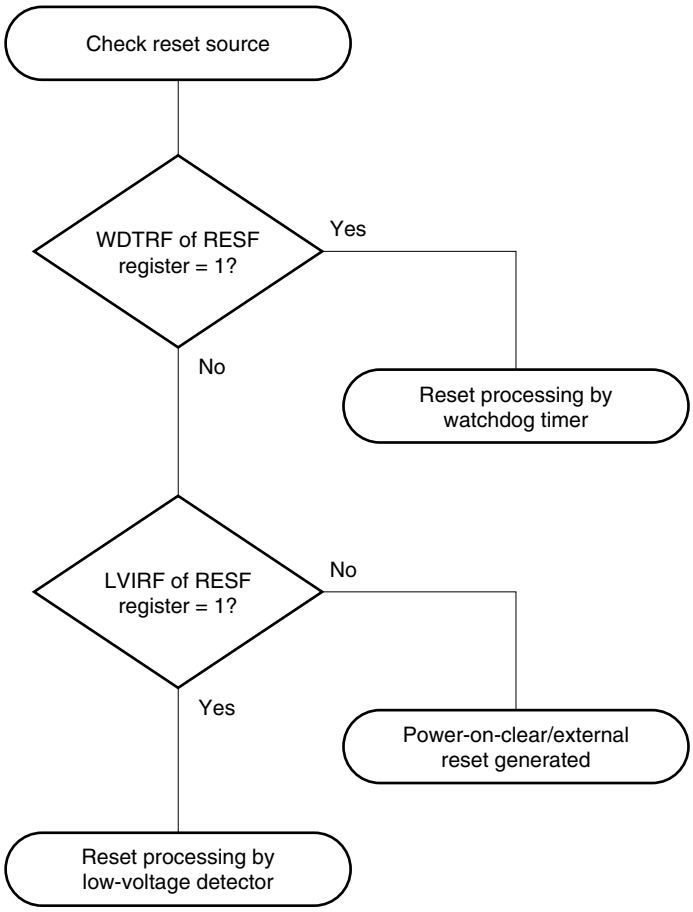
• If supply voltage fluctuation is 50 ms or less in vicinity of LVI detection voltage




**Note** A flowchart is shown on the next page.

**Figure 22-11. Example of Software Processing After Reset Release (2/2)**

• Checking reset source




**Operation example 2: When used as interrupt**

Interrupt requests may be generated frequently.

Take the following action.

**<Action>**

Confirm that "supply voltage ($V_{DD}$) ≥ LVI detection voltage ($V_{LVI}$)" when detecting the falling edge of $V_{DD}$, or "supply voltage ($V_{DD}$) < LVI detection voltage ($V_{LVI}$)" when detecting the rising edge of $V_{DD}$, in the servicing routine of the LVI interrupt by using bit 0 (LVIF) of the low-voltage detection register (LVIM). Clear bit 1 (LVIIF) of interrupt request flag register 0L (IF0L) to 0.

For a system with a long supply voltage fluctuation period near the LVI detection voltage, take the above action after waiting for the supply voltage fluctuation time.

**Remark** If bit 2 (LVISEL) of the low voltage detection register (LVIM) is set to "1", the meanings of the above words change as follows.

- Supply voltage ($V_{DD}$) → Input voltage from external input pin (EXLVI)
- Detection voltage ($V_{LVI}$) → Detection voltage ($V_{EXLVI}$ = 1.21 V)

# CHAPTER 23 REGULATOR

## 23.1 Regulator Overview

The 78K0/Kx2-L microcontrollers contain a circuit for operating the device with a constant voltage. At this time, in order to stabilize the regulator output voltage, connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 μF). However, when using the STOP mode that has been entered since operation of the internal high-speed oscillation clock and external main system clock, 0.47 μF is recommended. Also, use a capacitor with good characteristics, since it is used to stabilize internal voltage.

The regulator output voltage is normally 2.4 V (TYP.), and in the low power consumption mode, 2.0 V (TYP.).

## 23.2 Register Controlling Regulator

<R> **(1) Regulator mode control register (RMC)**

This register sets the output voltage of the regulator.

RMC is set with an 8-bit memory manipulation instruction.

Reset input sets this register to 00H.

**Figure 23-1. Format of Regulator Mode Control Register (RMC)**

Address: FF3DH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| RMC | | | | | | | | |

| RMC[7:0] | Control of output voltage of regulator |
|---|---|
| 56H | Low power consumption mode (fixed to 2.0 V) |
| 00H | Normal power mode (fixed to 2.4 V) |
| Other than above | Setting prohibited |

**Cautions 1. To change the RMC register setting value from 56H to 00H and use a CPU operating frequency of 5 MHz or more, change the PCC and RCM registers when 10 μs or more has elapsed after the RMC register was set.**

**2. When transitioning to the STOP mode, sub-system clock operation mode, and sub-system clock HALT mode, it is possible to achieve low power consumption by setting RMC = 56H.**

**3. When using the setting fixed to the low power consumption mode, the RMC register can be used in the following cases.**

**<When X1 clock is selected as the CPU clock> $f_X \leq 5$ MHz and $f_{CPU} \leq 5$ MHz**

**<When the high-speed internal oscillation clock, external input clock, or subsystem clock are selected for the CPU clock> $f_{CPU} \leq 5$ MHz**

## 23.3 Cautions for Self Programming

1. Make sure that the regulator output voltage mode is fixed when executing self programming or EEPROM emulation.
2. The power supply voltage range in which the flash memory can be rewritten in normal power mode is $V_{DD} \geq 2.5$ V. Note that program area can be rewritten by using the self programming library in normal power mode.
3. Observe the following points when rewriting the flash memory in low power consumption mode:
   - Data area can be rewritten in low power consumption mode, but program area cannot.
     Data area: Flash memory area handled as data
     Program area: Flash memory area handled as the program
   - The flash memory cannot be rewritten in low power consumption mode if the power supply voltage is 2.0 V or lower.
   - Flash memory that is erased and written in low power consumption mode cannot be accessed in normal power mode. To use this data in normal power mode, switch to low power consumption mode and transfer the flash memory contents to RAM.
   - A wait time of 2 ms is required before executing self programming after switching from normal power mode to low power consumption mode.

**Remark** For details of the self-programming function and the self programming library, refer to **"78K0 Microcontrollers User's Manual Self Programming Library Type 01 (U18274E)"** and **"78K0 Microcontrollers Self Programming Library Type 01 Ver. 3.10 Operating Precautions (notification document) (ZUD-CD-09-0122)"**.

For details of the EEPROM emulation library, refer to **"78K0 Microcontrollers User's Manual EEPROM Emulation Library Type 01 (U18275E)"** and **"78K0 Microcontrollers EEPROM Emulation Library Type 01 Ver.2.10 Operating Precautions (notification document) (ZUD-CD-09-0165)"**.

# CHAPTER 24 OPTION BYTE

## 24.1 Functions of Option Bytes

The flash memory at 0080H to 0084H of the 78K0/Kx2-L microcontrollers is an option byte area. When power is turned on or when the device is restarted from the reset status, the device automatically references the option bytes and sets specified functions. When using the product, be sure to set the following functions by using the option bytes.

When the boot swap operation is used during self-programming, 0080H to 0084H are switched to 1080H to 1084H. Therefore, set values that are the same as those of 0080H to 0084H to 1080H to 1084H in advance.

**(1) 0080H/1080H**

- Internal low-speed oscillator operation
  - Can be stopped by software
  - Cannot be stopped
- Watchdog timer interval time setting
- Watchdog timer counter operation
  - Enabled counter operation
  - Disabled counter operation
- Watchdog timer window open period setting

**Caution Set a value that is the same as that of 0080H to 1080H because 0080H and 1080H are switched during the boot swap operation.**

**(2) 0081H/1081H**

- LVI default start operation control
  - During LVI default start function enabled (LVISTART = 1)
    The device is in the reset state after reset release or upon power application and until the supply voltage reaches 1.91 V (TYP.). It is released from the reset state when the voltage exceeds 1.91 V (TYP.).
    If the supply voltage rises to 1.8 V after reset release or power application at a rate slower than 0.5 V/ms (MIN.), LVI default start function operation is recommended.
  - During LVI default start function stopped (LVISTART = 0)
    The device is in the reset state after reset release or upon power application and until the supply voltage reaches 1.61 V (TYP.). It is released from the reset state when the voltage exceeds 1.61 V (TYP.).

**Caution LVISTART can only be written by using a dedicated flash memory programmer. It cannot be set or change during self-programming or boot swap operation during self-programming. However, because 0080H and 1080H are switched during the boot swap operation, set a value that is the same as that of 0080H to 1080H.**

**(3) 0082H/1082H**

○ Internal high-speed oscillation clock frequency selection
- 4 MHz (TYP.)
- 8 MHz (TYP.)

**Caution Set a value that is the same as that of 0082H to 1082H because 0082H and 1082H are switched during the boot swap operation.**

<R> **(4) 0083H/1083H**

○ Pin selection used during on-chip debugging
- TOOLC0/X1, TOOLD0/X2
- TOOLC1/P31, TOOLD1/P32

**Caution Set a value that is the same as that of 0083H to 1083H because 0083H and 1083H are switched during the boot swap operation.**

**(5) 0084H/1084H**

○ On-chip debug operation control
- Disabling on-chip debug operation
- Enabling on-chip debug operation and erasing data of the flash memory in case authentication of the on-chip debug security ID fails
- Enabling on-chip debug operation and not erasing data of the flash memory even in case authentication of the on-chip debug security ID fails

**Caution Set a value that is the same as that of 0084H to 1084H because 0084H and 1084H are switched during the boot swap operation.**

## 24.2 Format of Option Byte

The format of the option byte is shown below.

**Figure 24-1. Format of Option Byte (1/3)**

Address: 0080H/1080H[Note]

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | WINDOW1 | WINDOW0 | WDTON | WDCS2 | WDCS1 | WDCS0 | LSROSC |

| WINDOW1 | WINDOW0 | Watchdog timer window open period |
|---|---|---|
| 0 | 0 | 25% |
| 0 | 1 | 50% |
| 1 | 0 | 75% |
| 1 | 1 | 100% |

| WDTON | Operation control of watchdog timer counter/illegal access detection |
|---|---|
| 0 | Counter operation disabled (counting stopped after reset), illegal access detection operation disabled |
| 1 | Counter operation enabled (counting started after reset), illegal access detection operation enabled |

| WDCS2 | WDCS1 | WDCS0 | Watchdog timer overflow time |
|---|---|---|---|
| 0 | 0 | 0 | $2^7/f_{IL}$ (3.88 ms) |
| 0 | 0 | 1 | $2^8/f_{IL}$ (7.76 ms) |
| 0 | 1 | 0 | $2^9/f_{IL}$ (15.52 ms) |
| 0 | 1 | 1 | $2^{10}/f_{IL}$ (31.03 ms) |
| 1 | 0 | 0 | $2^{12}/f_{IL}$ (124.12 ms) |
| 1 | 0 | 1 | $2^{14}/f_{IL}$ (496.48 ms) |
| 1 | 1 | 0 | $2^{15}/f_{IL}$ (992.97 ms) |
| 1 | 1 | 1 | $2^{17}/f_{IL}$ (3.97 s) |

| LSROSC | Internal low-speed oscillator operation |
|---|---|
| 0 | Can be stopped by software (stopped when 1 is written to bit 1 (LSRSTOP) of RCM register) |
| 1 | Cannot be stopped (not stopped even if 1 is written to LSRSTOP bit) |

**Note** Set a value that is the same as that of 0080H to 1080H because 0080H and 1080H are switched during the boot swap operation.

**Cautions 1. The combination of WDCS2 = WDCS1 = WDCS0 = 0 and WINDOW1 = WINDOW0 = 0 is prohibited.**

**2. The watchdog timer continues its operation during self-programming and EEPROM emulation of the flash memory. During processing, the interrupt acknowledge time is delayed. Set the overflow time and window size taking this delay into consideration.**

**3. If LSROSC = 0 (oscillation can be stopped by software), the count clock is not supplied to the watchdog timer in the HALT and STOP modes, regardless of the setting of bit 0 (LSRSTOP) of the internal oscillation mode register (RCM).**

**When 8-bit timer H1 operates with the internal low-speed oscillation clock, the count clock is supplied to 8-bit timer H1 even in the HALT/STOP mode.**

**4. Be sure to clear bit 7 to 0.**

**Remarks 1.** $f_{IL}$: Internal low-speed oscillation clock frequency

**2.** ( ): $f_{IL}$ = 33 kHz (MAX.)

**Figure 24-1. Format of Option Byte (2/3)**

Address: 0081H/1081H[Notes 1, 2]

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | LVISTART |

<R>

| LVISTART | LVI default start operation control |
|---|---|
| 0 | LVI is OFF by default upon power application (LVI default start function stopped) |
| 1 | LVI is ON by default upon power application (LVI default start function enabled) |

**Notes 1.** LVISTART can only be written by using a dedicated flash memory programmer. It cannot be set during self-programming or boot swap operation during self-programming. However, because 0080H and 1080H are switched during the boot swap operation, set a value that is the same as that of 0080H to 1080H.

**2.** To change the setting for the LVI default start, set the value to 0081H again after batch erasure (chip erasure) of the flash memory. The setting cannot be changed after the memory of the specified block is erased.

**Caution Be sure to clear bits 7 to 1 to "0".**

Address: 0082H/1082H[Note]

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | R4M8MSEL |

| R4M8MSEL | Internal high-speed oscillation clock frequency selection |
|---|---|
| 0 | 8 MHz (TYP.) |
| 1 | 4 MHz (TYP.) |

**Note** Set a value that is the same as that of 0082H to 1082H because 0082H and 1082H are switched during the boot swap operation.

**Caution Be sure to clear bits 7 to 1 to "0".**

**Figure 24-1. Format of Option Byte (3/3)**

Address: 0083H/1083H[Note]

<R>

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | OCDPSEL | 0 |

| OCDPSEL | Pin selection used during on-chip debugging |
|---|---|
| 0 | TOOLC1/P31, TOOLD1/P32 |
| 1 | TOOLC0/X1, TOOLD0/X2 |

**Note** Set a value that is the same as that of 0083H to 1083H because 0083H and 1083H are switched during the boot swap operation.

**Cautions 1. Be sure to clear bits 7 to 5 and 0 to "0" and set bits 4 to 2 to "1".**
**2. The setting of OCDPSEL bit is valid while OCDONB = 1.**

Address: 0084H/1084H[Note]

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | OCDEN1 | OCDEN0 |

| OCDEN1 | OCDEN0 | On-chip debug operation control |
|---|---|---|
| 0 | 0 | Operation disabled |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Operation enabled. Does not erase data of the flash memory in case authentication of the on-chip debug security ID fails. |
| 1 | 1 | Operation enabled. Erases data of the flash memory in case authentication of the on-chip debug security ID fails. |

**Note** Set a value that is the same as that of 0084H to 1084H because 0084H and 1084H are switched during the boot swap operation.

**Caution Be sure to clear bits 7 to 2 to "0".**

**Remark** For the on-chip debug security ID, refer to **CHAPTER 26 ON-CHIP DEBUG FUNCTION**.

Here is an example of description of the software for setting the option bytes.

```
OPT     CSEG   AT 0080H
OPTION: DB     30H      ; Enables watchdog timer operation (illegal access detection operation),
                        ; Window open period of watchdog timer: 50%,
                        ; Overflow time of watchdog timer: 2^7/fIL,
                        ; Internal low-speed oscillator can be stopped by software.
        DB     00H      ; LVI default start function stopped
        DB     00H      ; Internal high-speed oscillation clock frequency 8 MHz (TYP.)
<R>     DB     1EH      ; Use the TOOLC0/X1, TOOLD0/X2 pins
        DB     02H      ; Operation enabled.  Does not erase data of the flash memory in case
                        ; authentication of the on-chip debug security ID fails.
```

**Remark** Referencing of the option byte is performed during reset processing. For the reset processing timing, refer to **CHAPTER 20 RESET FUNCTION**.

# CHAPTER 25 FLASH MEMORY

The 78K0/Kx2-L microcontrollers incorporates the flash memory to which a program can be written, erased, and overwritten while mounted on the board.

## 25.1 Internal Memory Size Switching Register

Select the internal memory capacity using the internal memory size switching register (IMS).
IMS is set by an 8-bit memory manipulation instruction.
Reset signal generation sets IMS to CFH.

**Caution Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated Table 25-1 after release of reset.**

**Figure 25-1. Format of Internal Memory Size Switching Register (IMS)**

Address: FFF0H After reset: CFH R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| IMS | RAM2 | RAM1 | RAM0 | 0 | ROM3 | ROM2 | ROM1 | ROM0 |

| RAM2 | RAM1 | RAM0 | Internal high-speed RAM capacity selection |
|---|---|---|---|
| 0 | 0 | 0 | 768 bytes |
| 0 | 1 | 0 | 512 bytes |
| 0 | 1 | 1 | 384 bytes |
| 1 | 1 | 0 | 1024 bytes |
| Other than above | | | Setting prohibited |

| ROM3 | ROM2 | ROM1 | ROM0 | Internal ROM capacity selection |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 4 KB |
| 0 | 0 | 1 | 0 | 8 KB |
| 0 | 1 | 0 | 0 | 16 KB |
| 1 | 0 | 0 | 0 | 32 KB |
| 1 | 1 | 1 | 1 | (Default value) |
| Other than above | | | | Setting prohibited |

**Table 25-1. Set Values of Internal Memory Size Switching Register**

| Products | | | | IMS Setting |
|---|---|---|---|---|
| 78K0/KY2-L | 78K0/KA2-L | 78K0/KB2-L | 78K0/KC2-L | |
| μPD78F0550, 78F0555 | μPD78F0560, 78F0565 | – | – | 61H |
| μPD78F0551, 78F0556 | μPD78F0561, 78F0566 | μPD78F0571, 78F0576 | μPD78F0581, 78F0586 | 42H |
| μPD78F0552, 78F0557 | μPD78F0562, 78F0567 | μPD78F0572, 78F0577 | μPD78F0582, 78F0587 | 04H |
| – | – | μPD78F0573, 78F0578 | μPD78F0583, 78F0588 | C8H |

## 25.2 Writing with Flash Memory Programmer

Data can be written to the flash memory on-board or off-board, by using a dedicated flash memory programmer.

**(1) On-board programming**

The contents of the flash memory can be rewritten after the 78K0/Kx2-L microcontrollers have been mounted on the target system. The connectors that connect the dedicated flash memory programmer must be mounted on the target system.

**(2) Off-board programming**

Data can be written to the flash memory with a dedicated program adapter (FA series) before the 78K0/Kx2-L microcontrollers are mounted on the target system.

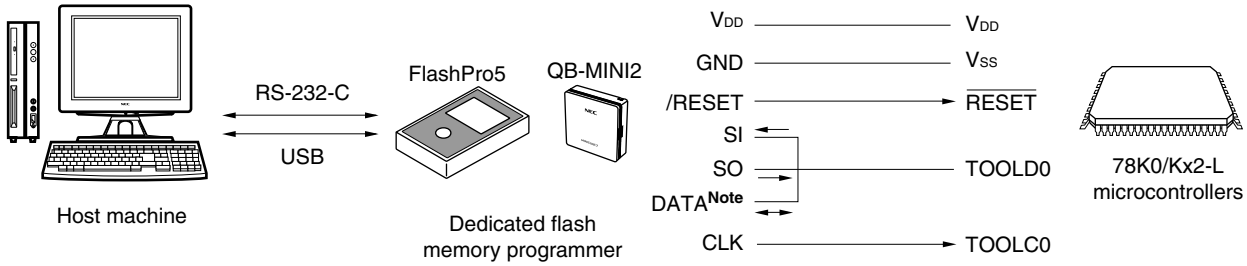
**Remark** The FA series is a product of Naito Densei Machida Mfg. Co., Ltd.

## 25.3 Programming Environment

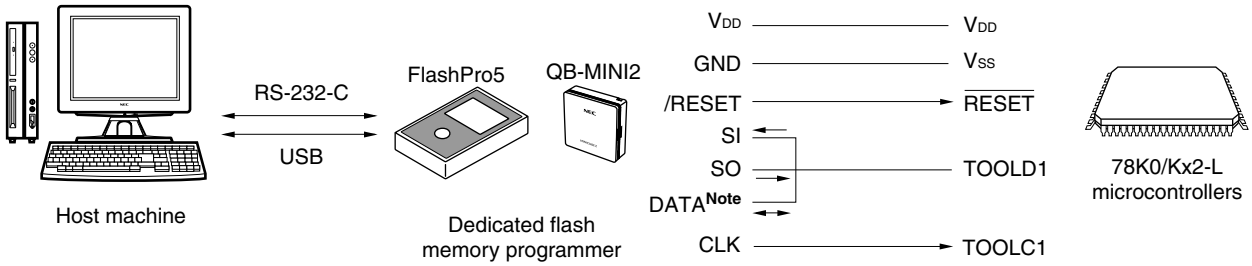
The environment required for writing a program to the flash memory of the 78K0/Kx2-L microcontrollers are illustrated below.

**Figure 25-2. Environment for Writing Program to Flash Memory**

**(1) When using the TOOLC0 and TOOLD0 pins**




**(2) When using the TOOLC1 and TOOLD1 pins**




**Note** QB-MINI2 only

A host machine that controls the dedicated flash memory programmer is necessary.

To interface between the dedicated flash memory programmer and the 78K0/Kx2-L microcontrollers, the TOOLD0 or TOOLD1 pins is used for manipulation such as writing and erasing via a dedicated single-line UART. To write the flash memory off-board, a dedicated program adapter (FA series) is necessary.

**Table 25-2. Pin Connection**

| Dedicated Flash memory programmer | | | 78K0/Kx2-L microcontrollers |
|---|---|---|---|
| Signal Name | I/O | Pin Function | Pin Name |
| CLK | Output | Clock output to 78K0/Kx2-L microcontrollers | TOOLC0/TOOLC1 |
| SI | Input | Receive signal | TOOLD0/TOOLD1 |
| SO | Output | Transmit signal | |
| DATA[Note] | I/O | Input/output signal for data communication during debugging | |
| /RESET | Output | Reset signal | $\overline{\text{RESET}}$ |
| $V_{DD}$ | I/O | $V_{DD}$ voltage generation/power monitoring | $V_{DD}$ |
| GND | – | Ground | $V_{SS}$ |

**Note** QB-MINI2 only

## 25.4 Connection of Pins on Board

To write the flash memory on-board, connectors that connect the dedicated flash memory programmer must be provided on the target system. First provide a function that selects the normal operation mode or flash memory programming mode on the board.

When the flash memory programming mode is set, all the pins not used for programming the flash memory are in the same status as immediately after reset. Therefore, if the external device does not recognize the state immediately after reset, the pins must be handled as described below.

### 25.4.1 TOOL pins

The pins used for communication in flash memory programming mode are shown in the table below.

**Table 25-3. Pins Used for Communication in Flash Memory Programming Mode**

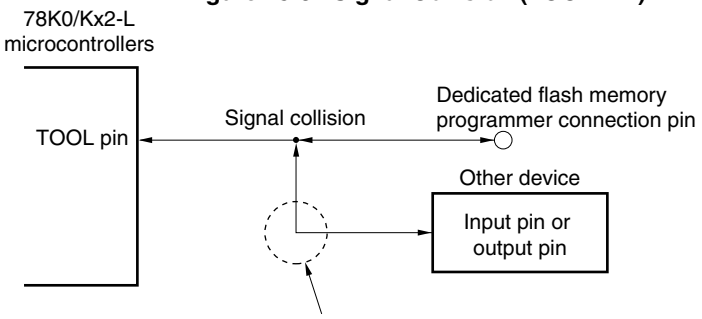
| Pin Name | Connection of Pins |
|---|---|
| TOOLC0, TOOLC1 | Connect this pin directly to the dedicated flash memory programmer or pull it down by connecting it to $V_{SS}$ via a resistor (10 kΩ) |
| TOOLD0, TOOLD1 | Connect this pin directly to the dedicated flash memory programmer or pull it up by connecting it to $V_{DD}$ via a resistor (3 k to 10 kΩ) |

To connect the dedicated flash memory programmer to the pins of a serial interface that is connected to another device on the board, care must be exercised so that signals do not collide or that the other device does not malfunction.

**(1) Signal collision**

If the dedicated flash memory programmer is connected to the TOOL pin that is connected to another device, signal collision takes place. To avoid this collision, either isolate the connection with the other device, or make the other device go into a high-impedance state.

**Figure 25-3. Signal Collision (TOOL Pin)**



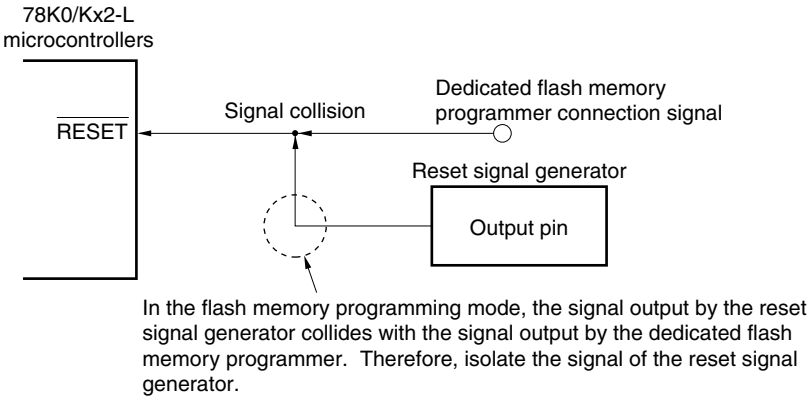

In the flash memory programming mode, the signal of the other device collides with the signal of the dedicated flash programmer. Therefore, isolate the signal of the other device.

### 25.4.2 $\overline{\text{RESET}}$ pin

If the reset signal of the dedicated flash memory programmer is connected to the $\overline{\text{RESET}}$ pin that is connected to the reset signal generator on the board, signal collision takes place. To prevent this collision, isolate the connection with the reset signal generator.

If the reset signal is input from the user system while the flash memory programming mode is set, the flash memory will not be correctly programmed. Do not input any signal other than the reset signal of the dedicated flash memory programmer.

**Figure 25-4. Signal Collision ($\overline{\text{RESET}}$ Pin)**




### 25.4.3 Port pins

When the flash memory programming mode is set, all the pins not used for flash memory programming enter the same status as that immediately after reset. If external devices connected to the ports do not recognize the port status immediately after reset, the port pin must be connected to $V_{DD}$ or $V_{SS}$ via a resistor.

### 25.4.4 REGC pin

Connect the REGC pin to GND via a capacitor (0.47 to 1 $\mu$F) in the same manner as during normal operation. However, when using the STOP mode that has been entered since operation of the internal high-speed oscillation clock and external main system clock, 0.47 $\mu$F is recommended. Also, use a capacitor with good characteristics, since it is used to stabilize internal voltage.

### 25.4.5 Other signal pins

Connect X1, X2, XT1, and XT2 in the same status as in the normal operation mode.

**Remark** In the flash memory programming mode, the internal high-speed oscillation clock ($f_{IH}$) is used.

### 25.4.6 Power supply

To use the supply voltage output of the flash memory programmer, connect the $V_{DD}$ pin to $V_{DD}$ of the flash memory programmer, and the $V_{SS}$ pin to GND of the flash memory programmer.

To use the on-board supply voltage, connect in compliance with the normal operation mode.

However, be sure to connect the $V_{DD}$ and $V_{SS}$ pins to $V_{DD}$ and GND of the flash memory programmer to use the power monitor function with the flash memory programmer, even when using the on-board supply voltage.

Supply the same other power supplies ($AV_{REF}$ and $AV_{SS}$) as those in the normal operation mode.

### 25.4.7 On-board writing when connecting crystal/ceramic resonator

To write the flash memory on-board, connectors that connect the dedicated flash memory programmer must be provided on the target system. First provide a function that selects the normal operation mode or flash memory programming mode on the board.

When the flash memory programming mode is set, all the pins not used for programming the flash memory are in the same status as immediately after reset. Therefore, if the external device does not recognize the state immediately after reset, the pins must be processed as described below.

The state of the pins in the self programming mode is the same as that in the HALT mode.

When using the X1 (TOOLC0) and X2 (TOOLD0) pins as the serial interface for flash memory programming, signals will collide if an external device is connected. To prevent the conflict of signals, isolate the connection with the external device.

Similarly, when a capacitor is connected to the X1 and X2 pins, the waveform during communication is changed, and thus communication may be disabled depending on the capacitor capacitance. Make sure to isolate the connection with the capacitor during flash programming.

In cases when a crystal or ceramic resonator has been selected to generate the system clock, and the decision has been made to execute on-board flash programming with the resonator mounted on the device because it is difficult to isolate the resonator, be sure to thoroughly evaluate the flash memory programming with the resonator mounted on the device before executing the processing described next.

- Mount the minimum-possible test pads between the device and the resonator, and connect the programmer via the test pad. Keep the wiring as short as possible (refer to **Figure 25-5 and Table 25-4**).

**Figure 25-5. Example of Mounting Test Pads**

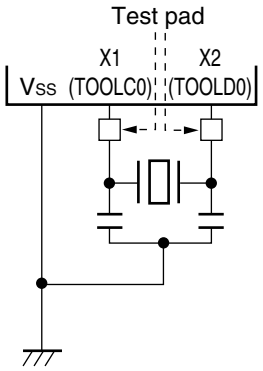



**Table 25-4. Clock to Be Used and Mounting of Test Pads**

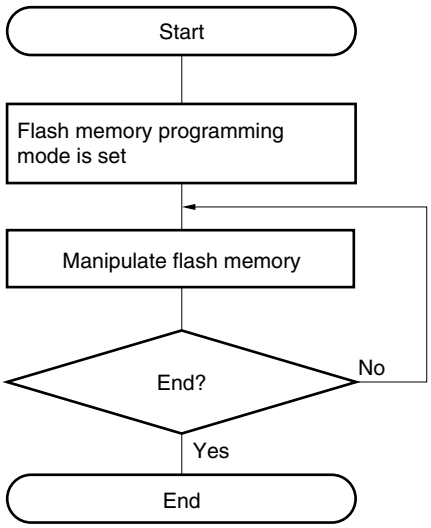
| Clock to Be Used | | Mounting of Test Pads |
|---|---|---|
| High-speed internal oscillation clock | | Not required |
| External clock | | |
| Crystal/ceramic oscillation clock | Before resonator is mounted | |
| | After resonator is mounted | Required |

## 25.5 Programming Method

### 25.5.1 Controlling flash memory

The following figure illustrates the procedure to manipulate the flash memory.

**Figure 25-6. Flash Memory Manipulation Procedure**




### 25.5.2 Flash memory programming mode

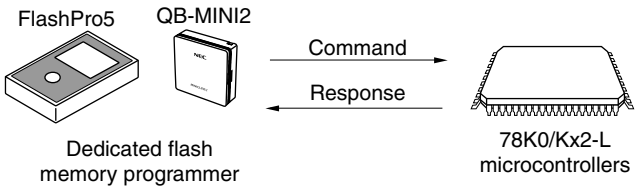
To rewrite the contents of the flash memory by using the dedicated flash memory programmer, set the 78K0/Kx2-L microcontrollers in the flash memory programming mode. The system switches to the flash memory programming mode once the dedicated flash memory programmer is connected and communication starts.

Change the mode by using a jumper when writing the flash memory on-board.

### 25.5.3 Communication commands

The 78K0/Kx2-L microcontrollers communicate with the dedicated flash memory programmer by using commands. The signals sent from the flash memory programmer to the 78K0/Kx2-L microcontrollers are called commands, and the signals sent from the 78K0/Kx2-L microcontrollers to the dedicated flash memory programmer are called response.

**Figure 25-7. Communication Commands**




The flash memory control commands of the 78K0/Kx2-L microcontrollers are listed in the table below. All these commands are issued from the programmer and the 78K0/Kx2-L microcontrollers perform processing corresponding to the respective commands.

**Table 25-5. Flash Memory Control Commands**

| Classification | Command Name | Function |
|---|---|---|
| Verify | Verify | Compares the contents of a specified area of the flash memory with data transmitted from the programmer. |
| Erase | Chip Erase | Erases the entire flash memory. |
| | Block Erase | Erases a specified area in the flash memory. |
| Blank check | Block Blank Check | Checks if a specified block in the flash memory has been correctly erased. |
| Write | Programming | Writes data to a specified area in the flash memory. |
| Getting information | Silicon Signature | Gets 78K0/Kx2-L information (such as the part number and flash memory configuration). |
| | Version Get | Gets the 78K0/Kx2-L version and firmware version. |
| | Checksum | Gets the checksum data for a specified area. |
| Security | Security Set | Sets security information. |
| Others | Reset | Used to detect synchronization status of communication. |
| | Baud Rate Set | Sets baud rate when UART communication mode is selected. |

The 78K0/Kx2-L microcontrollers return a response for the command issued by the dedicated flash memory programmer. The response names sent from the 78K0/Kx2-L microcontrollers are listed below.

**Table 25-6. Response Names**

| Response Name | Function |
|---|---|
| ACK | Acknowledges command/data. |
| NAK | Acknowledges illegal command/data. |

## 25.6 Security Settings

The 78K0/Kx2-L microcontrollers support a security function that prohibits rewriting the user program written to the internal flash memory, so that the program cannot be changed by an unauthorized person.

The operations shown below can be performed using the Security Set command. The security setting is valid when the programming mode is set next.

- Disabling batch erase (chip erase)
  Execution of the block erase and batch erase (chip erase) commands for entire blocks in the flash memory is prohibited by this setting during on-board/off-board programming. Once execution of the batch erase (chip erase) command is prohibited, all of the prohibition settings (including prohibition of batch erase (chip erase)) can no longer be cancelled.

  **Caution After the security setting for the batch erase is set, erasure cannot be performed for the device. In addition, even if a write command is executed, data different from that which has already been written to the flash memory cannot be written, because the erase command is disabled.**

- Disabling block erase
  Execution of the block erase command for a specific block in the flash memory is prohibited during on-board/off-board programming. However, blocks can be erased by means of self programming.

- Disabling write
  Execution of the write and block erase commands for entire blocks in the flash memory is prohibited during on-board/off-board programming. However, blocks can be written by means of self programming.

- Disabling rewriting boot cluster 0
  Execution of the block erase command and write command on boot cluster 0 (0000H to 0FFFH) in the flash memory is prohibited by this setting. Execution of the batch erase (chip erase) command is also prohibited by this setting.

The batch erase (chip erase), block erase, write commands, and rewriting boot cluster 0 are enabled by the default setting when the flash memory is shipped. Security can be set by on-board/off-board programming and self programming. Each security setting can be used in combination.

All the security settings are cleared by executing the batch erase (chip erase) command.

Table 25-7 shows the relationship between the erase and write commands when the 78K0/Kx2-L microcontroller security function is enabled.

**Table 25-7. Relationship Between Enabling Security Function and Command**

**(1) During on-board/off-board programming**

| Valid Security | Executed Command | | |
|---|---|---|---|
| | Batch Erase (Chip Erase) | Block Erase | Write |
| Prohibition of batch erase (chip erase) | Cannot be erased in batch | Blocks cannot be erased. | Can be performed[Note]. |
| Prohibition of block erase | Can be erased in batch. | | Can be performed. |
| Prohibition of writing | | | Cannot be performed. |
| Prohibition of rewriting boot cluster 0 | Cannot be erased in batch | Boot cluster 0 cannot be erased. | Boot cluster 0 cannot be written. |

**Note** Confirm that no data has been written to the write area. Because data cannot be erased after batch erase (chip erase) is prohibited, do not write data if the data has not been erased.

**(2) During self programming**

| Valid Security | Executed Command | |
|---|---|---|
| | Block Erase | Write |
| Prohibition of batch erase (chip erase) | Blocks can be erased. | Can be performed. |
| Prohibition of block erase | | |
| Prohibition of writing | | |
| Prohibition of rewriting boot cluster 0 | Boot cluster 0 cannot be erased. | Boot cluster 0 cannot be written. |

Table 25-8 shows how to perform security settings in each programming mode.

**Table 25-8. Setting Security in Each Programming Mode**

**(1) On-board/off-board programming**

| Security | Security Setting | How to Disable Security Setting |
|---|---|---|
| Prohibition of batch erase (chip erase) | Set via GUI of dedicated flash memory programmer, etc. | Cannot be disabled after set. |
| Prohibition of block erase | | Execute batch erase (chip erase) command |
| Prohibition of writing | | |
| Prohibition of rewriting boot cluster 0 | | Cannot be disabled after set. |

**(2) Self programming**

| Security | Security Setting | How to Disable Security Setting |
|---|---|---|
| Prohibition of batch erase (chip erase) | Set by using information library. | Cannot be disabled after set. |
| Prohibition of block erase | | Execute batch erase (chip erase) command during on-board/off-board programming (cannot be disabled during self programming) |
| Prohibition of writing | | |
| Prohibition of rewriting boot cluster 0 | | |

## 25.7 Processing Time for Each Command When PG-FP5 Is Used (Reference)

The following table shows the processing time for each command (reference) when the PG-FP5 is used as a dedicated flash memory programmer.

**Table 25-9. Processing Time for Each Command When PG-FP5 Is Used (Reference) (1/3)**

(1) 78K0/KY2-L, 78K0/KA2-L (1/2)

(a) Products with internal ROMs of the 4 KB: μPD78F0550, 78F0555, 78F0560, 78F0565

| Command of PG-FP5 | Port: UART-Internal-OSC (Internal high-speed oscillation clock ($f_{IH}$: 8 MHz (typ.)), Speed: 500,000 bps |
|---|---|
| Signature | 0.5 s (typ.) |
| Blankcheck | 0.5 s (typ.) |
| Erase | 0.5 s (typ.) |
| Program | 1 s (typ.) |
| Verify | 1 s (typ.) |
| E.P.V | 1 s (typ.) |
| Checksum | 0.5 s (typ.) |
| Security | 0.5 s (typ.) |

(b) Products with internal ROMs of the 8 KB: μPD78F0551, 78F0556, 78F0561, 78F0566

| Command of PG-FP5 | Port: UART-Internal-OSC (Internal high-speed oscillation clock ($f_{IH}$: 8 MHz (typ.)), Speed: 500,000 bps |
|---|---|
| Signature | 0.5 s (typ.) |
| Blankcheck | 0.5 s (typ.) |
| Erase | 1 s (typ.) |
| Program | 1.5 s (typ.) |
| Verify | 1 s (typ.) |
| E.P.V | 1.5 s (typ.) |
| Checksum | 0.5 s (typ.) |
| Security | 0.5 s (typ.) |

**Caution When executing boot swapping, do not use the E.P.V. command with the dedicated flash memory programmer.**

**Table 25-9. Processing Time for Each Command When PG-FP5 Is Used (Reference) (2/3)**

(1) 78K0/KY2-L, 78K0/KA2-L (2/2)

(c) Products with internal ROMs of the 16 KB: μPD78F0552, 78F0557, 78F0562, 78F0567

| Command of PG-FP5 | Port: UART-Internal-OSC (Internal high-speed oscillation clock ($f_{IH}$: 8 MHz (typ.)), Speed: 500,000 bps |
|---|---|
| Signature | 0.5 s (typ.) |
| Blankcheck | 0.5 s (typ.) |
| Erase | 1 s (typ.) |
| Program | 2.5 s (typ.) |
| Verify | 1.5 s (typ.) |
| E.P.V | 2.5 s (typ.) |
| Checksum | 1 s (typ.) |
| Security | 0.5 s (typ.) |

(2) 78K0/KB2-L, 78K0/KC2-L (1/2)

(a) Products with internal ROMs of the 8 KB: μPD78F0571, 78F0576, 78F0581, 78F0586

| Command of PG-FP5 | Port: UART-Internal-OSC (Internal high-speed oscillation clock ($f_{IH}$: 8 MHz (typ.)), Speed: 500,000 bps |
|---|---|
| Signature | 0.5 s (typ.) |
| Blankcheck | 1 s (typ.) |
| Erase | 1 s (typ.) |
| Program | 1.5 s (typ.) |
| Verify | 1 s (typ.) |
| E.P.V | 1.5 s (typ.) |
| Checksum | 1 s (typ.) |
| Security | 1 s (typ.) |

**Caution When executing boot swapping, do not use the E.P.V. command with the dedicated flash memory programmer.**

**Table 25-9. Processing Time for Each Command When PG-FP5 Is Used (Reference) (3/3)**

(2) 78K0/KB2-L, 78K0/KC2-L (2/2)

(b) Products with internal ROMs of the 16 KB: μPD78F0572, 78F0577, 78F0582, 78F0587

| Command of PG-FP5 | Port: UART-Internal-OSC (Internal high-speed oscillation clock ($f_{IH}$: 8 MHz (typ.)), Speed: 500,000 bps |
|---|---|
| Signature | 0.5 s (typ.) |
| Blankcheck | 1 s (typ.) |
| Erase | 1 s (typ.) |
| Program | 2.5 s (typ.) |
| Verify | 1.5 s (typ.) |
| E.P.V | 2.5 s (typ.) |
| Checksum | 1 s (typ.) |
| Security | 1 s (typ.) |

(c) Products with internal ROMs of the 32 KB: μPD78F0573, 78F0578, 78F0583, 78F0588

| Command of PG-FP5 | Port: UART-Internal-OSC (Internal high-speed oscillation clock ($f_{IH}$: 8 MHz (typ.)), Speed: 500,000 bps |
|---|---|
| Signature | 0.5 s (typ.) |
| Blankcheck | 1 s (typ.) |
| Erase | 1 s (typ.) |
| Program | 4.5 s (typ.) |
| Verify | 3 s (typ.) |
| E.P.V | 4.5 s (typ.) |
| Checksum | 1 s (typ.) |
| Security | 1 s (typ.) |

**Caution When executing boot swapping, do not use the E.P.V. command with the dedicated flash memory programmer.**

## 25.8 Flash Memory Programming by Self Programming

The 78K0/Kx2-L microcontrollers support a self programming function that can be used to rewrite the flash memory via a user program. Because this function allows a user application to rewrite the flash memory by using the 78K0/Kx2-L microcontroller self programming library, it can be used to upgrade the program in the field.

If an interrupt occurs during self programming, self programming can be temporarily stopped and interrupt servicing can be executed. If an unmasked interrupt request is generated in the EI state, the request branches directly from the self programming library to the interrupt routine. After the self programming mode is later restored, self programming can be resumed. However, the interrupt response time is different from that of the normal operation mode.

**Cautions 1. The self programming function cannot be used when the CPU operates with the subsystem clock.**

**2. To prohibit an interrupt during self programming, in the same way as in the normal operation mode, execute the self programming library in the state where the IE flag is cleared (0) by the DI instruction. To enable an interrupt, clear (0) the interrupt mask flag to accept in the state where the IE flag is set (1) by the EI instruction, and then execute the self programming library.**

**3. Make sure that the regulator output voltage mode is fixed when executing self programming or EEPROM emulation.**

**4. The power supply voltage range in which the flash memory can be rewritten in normal power mode is $V_{DD} \geq 2.5$ V. Note that program area can be rewritten by using the self programming library in normal power mode.**

**5. Observe the following points when rewriting the flash memory in low power consumption mode:**

- **Data area can be rewritten in low power consumption mode, but program area cannot.**
  **Data area: Flash memory area handled as data**
  **Program area: Flash memory area handled as the program**
- **The flash memory cannot be rewritten in low power consumption mode if the power supply voltage is 2.0 V or lower.**
- **Flash memory that is erased and written in low power consumption mode cannot be accessed in normal power mode. To use this data in normal power mode, switch to low power consumption mode and transfer the flash memory contents to RAM.**
- **Blocks cannot be overwritten by using the self programming library. Be sure to erase a block first before rewriting data to it.**
- **A wait time of 2 ms is required before executing self programming after switching from normal power mode to low power consumption mode.**

<R> **Remark** For details of the self programming function and the self programming library, refer to **"78K0 Microcontrollers User's Manual Self Programming Library Type 01 (U18274E)"** and **"78K0 Microcontrollers Self Programming Library Type 01 Ver. 3.10 Operating Precautions (notification document) (ZUD-CD-09-0122)"**.

For details of the EEPROM emulation library, refer to **"78K0 Microcontrollers User's Manual EEPROM Emulation Library Type 01 (U18275E)"** and **"78K0 Microcontrollers EEPROM Emulation Library Type 01 Ver.2.10 Operating Precautions (notification document) (ZUD-CD-09-0165)"**.

### 25.8.1 Register controlling self programming mode

The self programming mode is controlled by the self programming mode control register (FPCTL).

FPCTL can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears FPCTL to 00H.

**Figure 25-8. Format of Self Programming Mode Control Register (FPCTL)**

Address: FF2BH After reset: 00H R/W

| Symbol | 7 | 6 | 5 | 4 | 3 | 2 | 1 | <0> |
|---|---|---|---|---|---|---|---|---|
| FPCTL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | FLMDPUP Note |

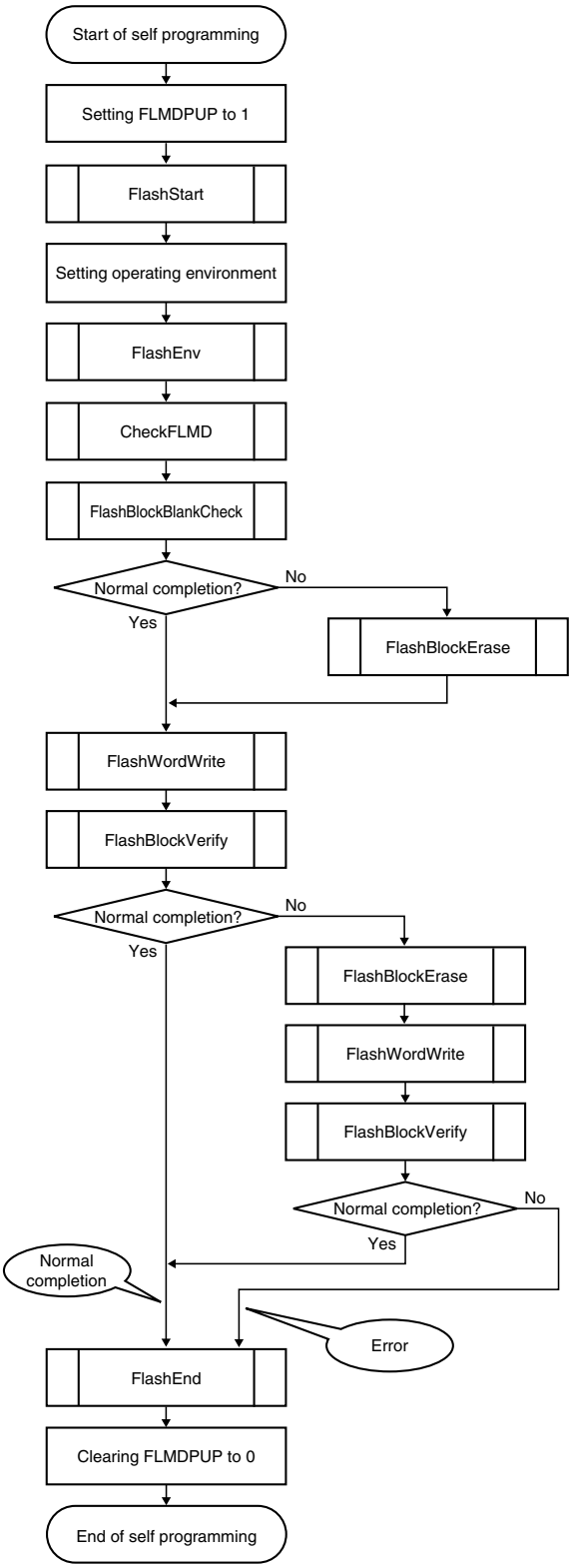
| FLMDPUP Note | Self programming mode control |
|---|---|
| 0 | Normal operation mode |
| 1 | Self programming mode |

**Note** The FLMDPUP bit must be set to 0 (normal operation mode) while the regular user program is being executed, and set to 1 (self programming mode) while self programming is being executed. The flash memory rewrite circuit does not operate in normal operation mode, so even though the firmware and software for rewriting will work, no actual rewriting will take place.

### 25.8.2 Flow of self programming (Rewriting Flash Memory)

The following figure illustrates a flow of rewriting the flash memory by using a self programming library.

**Figure 25-9. Flow of Self Programming (Rewriting Flash Memory)**




**Remark** For details of the self programming function and the self programming library, refer to **"78K0 Microcontrollers User's Manual Self Programming Library Type 01 (U18274E)"** and **"78K0 Microcontrollers Self Programming Library Type 01 Ver. 3.10 Operating Precautions (notification document) (ZUD-CD-09-0122)"**.

### 25.8.3 Boot swap function

If rewriting the boot area failed by temporary power failure or other reasons, restarting a program by resetting or overwriting is disabled due to data destruction in the boot area.

The boot swap function is used to avoid this problem.

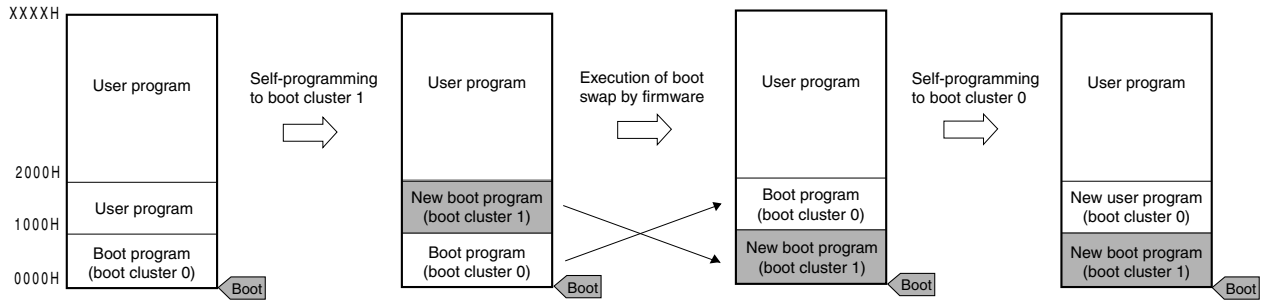
Before erasing boot cluster 0[Note], which is a boot program area, by self programming, write a new boot program to boot cluster 1 in advance. When the program has been correctly written to boot cluster 1, swap this boot cluster 1 and boot cluster 0 by using the set information function of the firmware of the 78K0/Kx2-L microcontrollers, so that boot cluster 1 is used as a boot area. After that, erase or write the original boot program area, boot cluster 0.

As a result, even if a power failure occurs while the boot programming area is being rewritten, the program is executed correctly because it is booted from boot cluster 1 to be swapped when the program is reset and started next.

**Note** A boot cluster is a 4 KB area and boot clusters 0 and 1 are swapped by the boot swap function.

**Caution The products whose ROM size is 4 KB can not use the boot swap function.**

**Figure 25-10. Boot Swap Function**

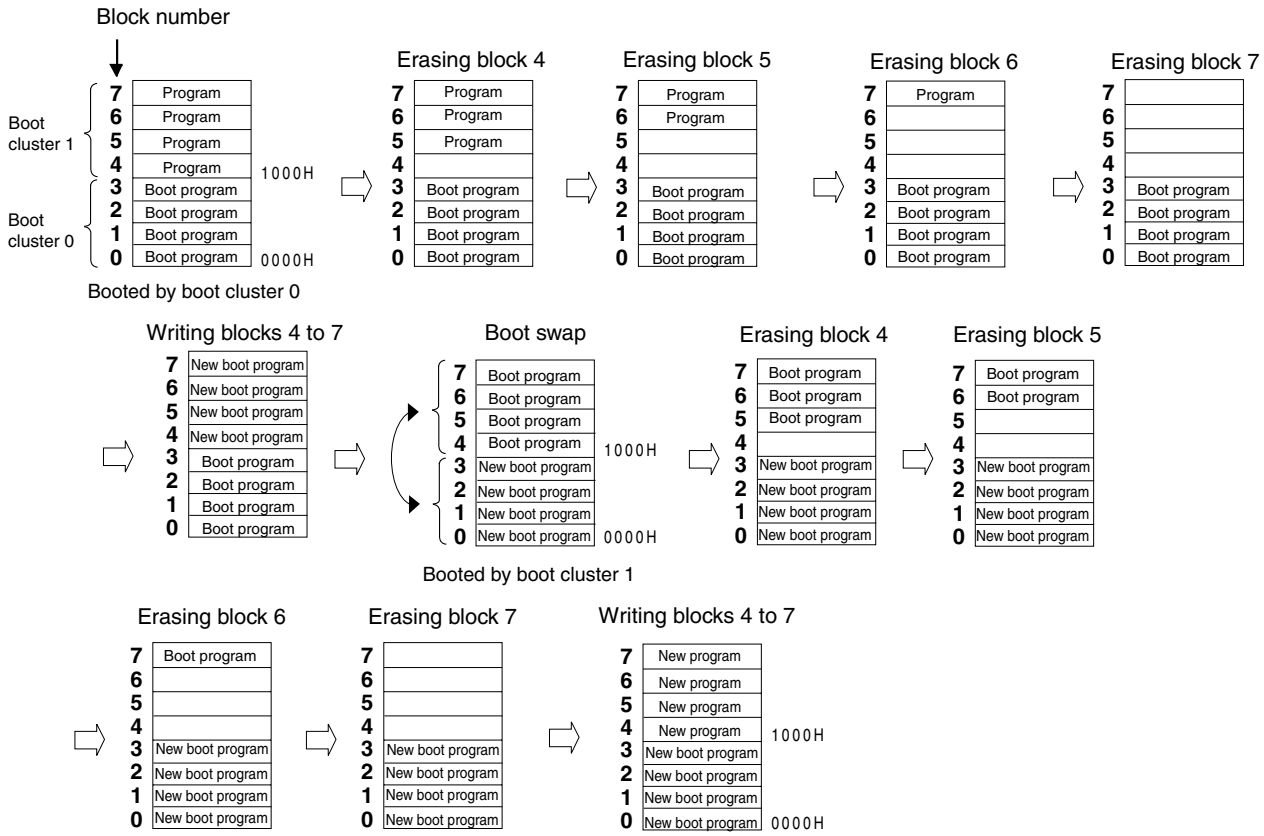



In an example of above figure, it is as follows.

Boot cluster 0: Boot program area before boot swap

Boot cluster 1: Boot program area after boot swap

**Figure 25-11. Example of Executing Boot Swapping**



## 25.9 Creating ROM Code to Place Order for Previously Written Product

Before placing an order with Renesas Electronics for a previously written product, the ROM code for the order must be created.

To create the ROM code, use the Hex Consolidation Utility (hereafter abbreviated to HCU) on the finished programs (hex files) and optional data (such as security settings for flash memory programs).

The HCU is a software tool that includes functions required for creating ROM code.

The HCU can be downloaded at the Renesas Electronics website.

**(1) Website**

<R> http://www2.renesas.com/micro/en/ods/ → Click Version-up Service.

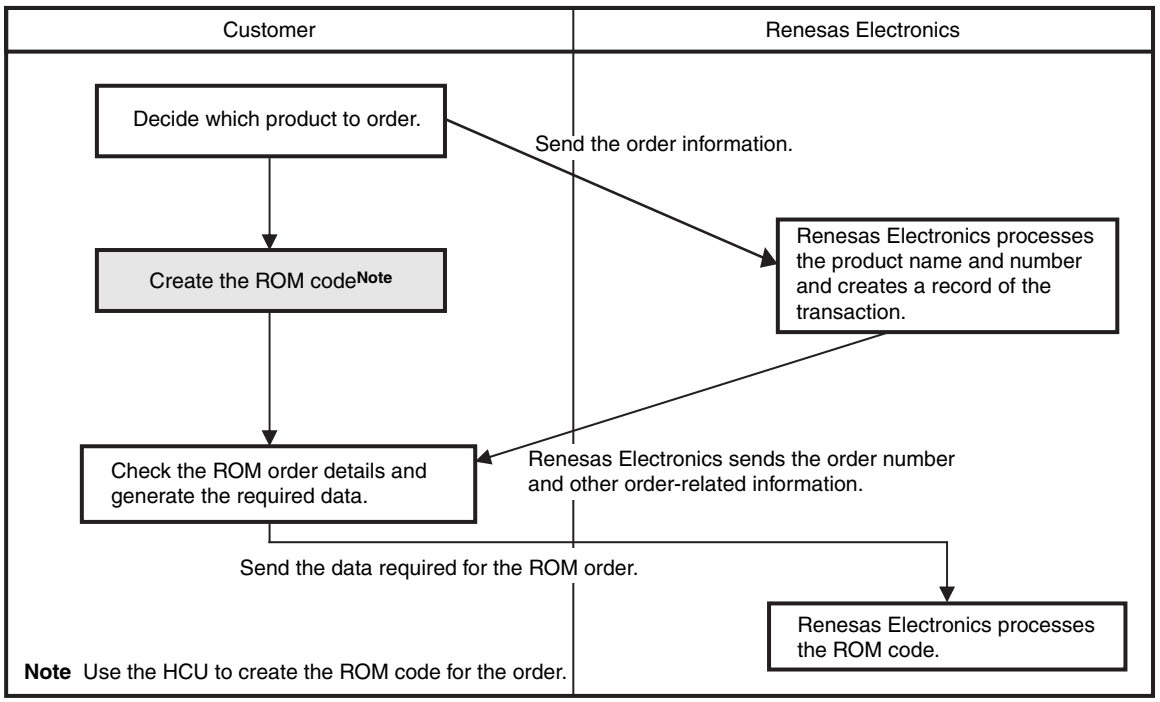
**(2) Downloading the HCU**

To download the HCU, click Software for previously written flash products and then HCU_GUI.

**Remark** For details about how to install and use the HCU, see the materials (the user's manual) that comes with the HCU at the above website.

### 25.9.1 Procedure for using ROM code to place an order

Use the HCU to create the ROM code by following the procedure below, and then place your order with Renesas Electronics. For details, see the ROM Code Ordering Method Information (C10302J).



# CHAPTER 26 ON-CHIP DEBUG FUNCTION

## 26.1 Connecting QB-MINI2 to 78K0/Kx2-L Microcontrollers

The 78K0/Kx2-L microcontrollers use the $V_{DD}$, $\overline{RESET}$, TOOLC0/X1 (or TOOLC1/P31), TOOLD0/X2 (or TOOLD1/P32), and $V_{SS}$ pins to communicate with the host machine via an on-chip debug emulator (QB-MINI2). Whether TOOLC0/X1 and TOOLC1/P31, or TOOLD0/X2 and TOOLD1/P32 are used can be selected.

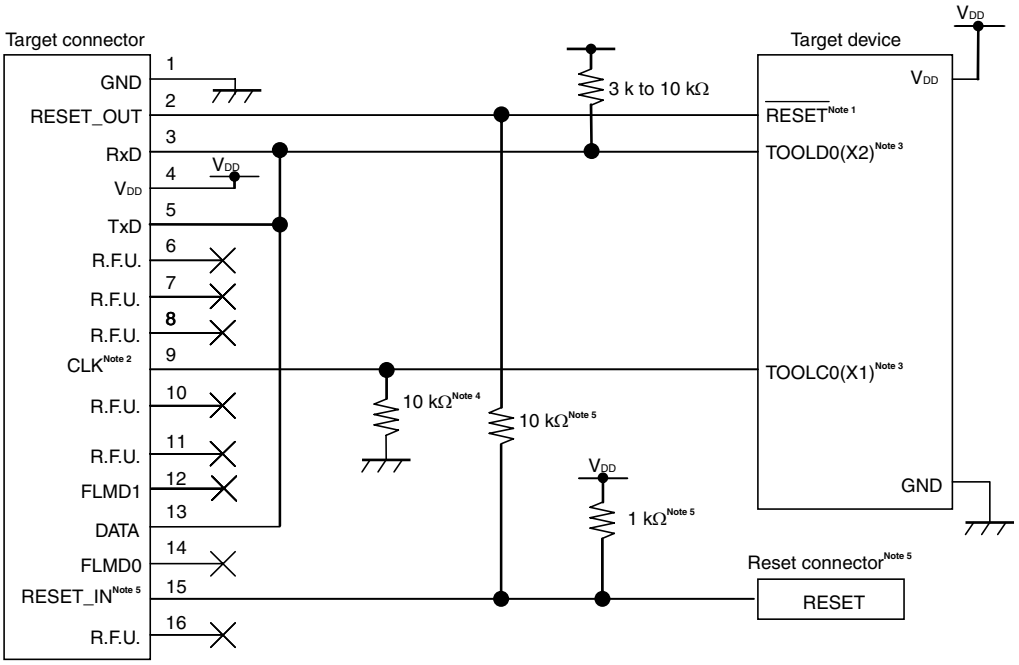
**Cautions 1. The 78K0/Kx2-L microcontrollers have an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.**

<R> **2. When transitioning to STOP mode during on-chip debugging, oscillation of the internal high-speed oscillator continues, but the on-chip debug operation is not affected.**

**Remark** The 78K0/KY2-L is not provided with the TOOLC1/P31 and TOOLD1/P32 pins.

**Figure 26-1. Connection Example of QB-MINI2 and 78K0/Kx2-L Microcontrollers (1/3)**

**(1) When using the TOOLC0 and TOOLD0 pins (X1 oscillator or EXCLK input clock is not used, both debugging and programming are performed)**




**Notes 1.** If there are capacitance elements such as capacitors, on-chip debugging might not operate normally.

**2.** A clock signal provided on the 78K0-OCD board, a 4, 8, or 16 MHz clock signal generated in QB-MINI2, or the clock signal generated by the internal high-speed oscillator of the device can be used for the clock signal of the target device during on-chip debugging.
Only the internal high-speed oscillator of the device can be used during flash programming.

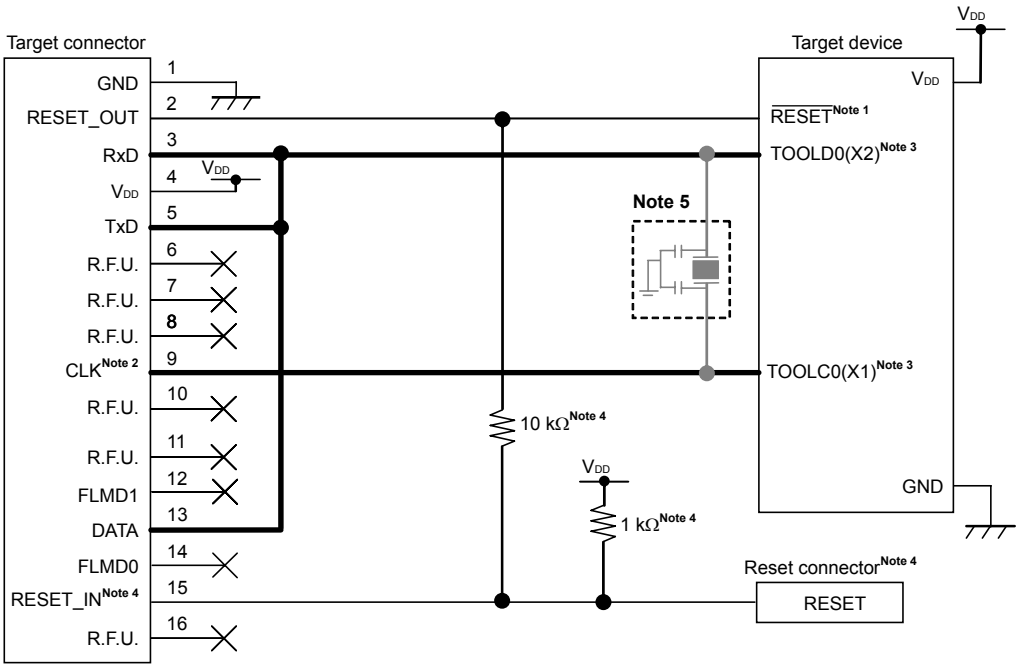
**3.** During on-chip debugging, the settings specified by the user program are ignored, because these pins are used as pins dedicated to on-chip debugging. However, if the pins are specified as input pins, the pins must be processed (because they are left open when QB-MINI2 is not connected.)

**4.** This is the processing for the pin that is unused (the input is left open) when the target device operates (when QB-MINI2 is not connected). (This processing is not required if an oscillator circuit is used.)

**5.** This connection is designed assuming that the reset signal is output from the N-ch open-drain buffer (output resistance: 100 Ω or less).

<R> **Figure 26-1. Connection Example of QB-MINI2 and 78K0/Ix2 Microcontrollers (2/3)**

**(2) When using the TOOLC0 and TOOLD0 pins (with X1/X2 oscillator is used, both debugging and programming are performed)**




**Notes 1.** If there are capacitance elements such as capacitors, on-chip debugging might not operate normally.

**2.** A clock signal provided on the 78K0-OCD board, a 4, 8, or 16 MHz clock signal generated in QB-MINI2, or the clock signal generated by the internal high-speed oscillator of the device can be used for the clock signal of the target device during on-chip debugging.
Only the internal high-speed oscillator of the device can be used during flash programming.

**3.** During on-chip debugging, the settings specified by the user program are ignored, because these pins are used as pins dedicated to on-chip debugging. However, if the pins are specified as input pins, the pins must be processed (because they are left open when QB-MINI2 is not connected.)

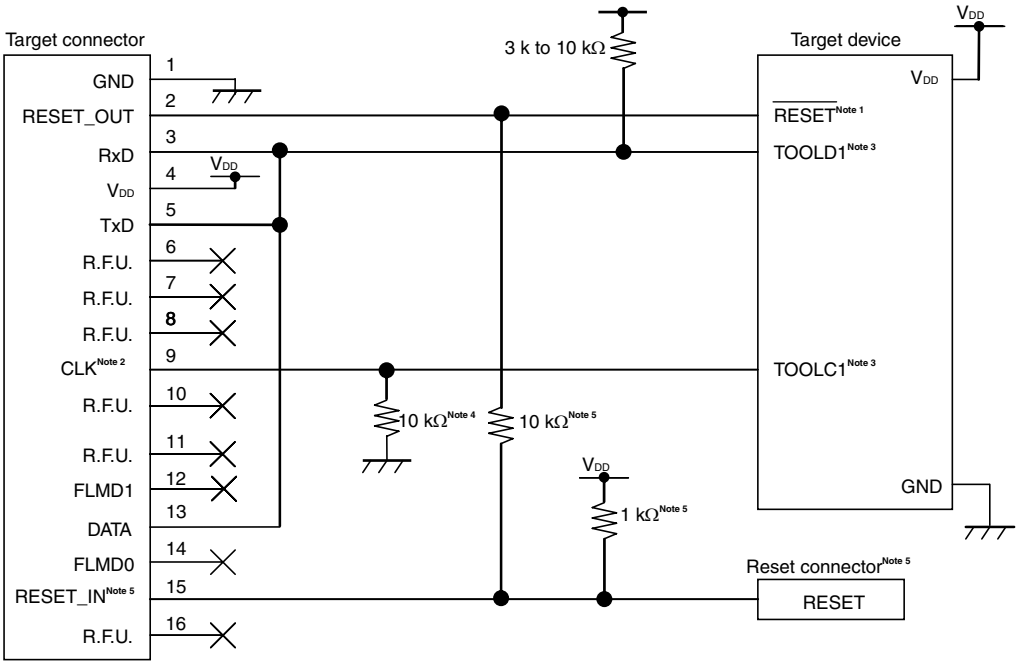
**4.** This connection is designed assuming that the reset signal is output from the N-ch open-drain buffer (output resistance: 100 Ω or less). For details, refer to **4.1.3 Connection of reset pin** of QB-MINI2 On-Chip Debug Emulator with Programming Function (18371E).

**5.** Never connect an oscillation circuit to the 78K0-OCD board during on-chip debugging and flash programming.
To prevent an oscillation circuit from not oscillating due to wiring capacitance when the target device operates (when QB-MINI2 is not connected), also consider countermeasures such as disconnecting the oscillation circuit from the target connectors by setting the jumpers.
A program that was downloaded using the debugger does not operate when QB-MINI2 is not connected.

**Caution** The bold lines in the figure (TOOLD0 and TOOLC0) must be designed so that the device pins are less than 30 mm from the QB-MINI2 connectors or the paths must be shielded by connecting them to GND.

**Figure 26-1. Connection Example of QB-MINI2 and 78K0/Kx2-L Microcontrollers (3/3)**

**(3) When using the TOOLC1 and TOOLD1 pins (both debugging and programming are performed)**




**Notes 1.** If there are capacitance elements such as capacitors, on-chip debugging might not operate normally.

**2.** The clock signal generated by the clock circuit on the target system or by the internal high-speed oscillator of the device can be used for the clock signal of the target device during on-chip debugging.
Only the internal high-speed oscillator of the device can be used during flash programming.

**3.** During on-chip debugging, the settings specified by the user program are ignored, because these pins are used as pins dedicated to on-chip debugging. However, if the pins are specified as input pins, the pins must be processed (because they are left open when QB-MINI2 is not connected.)

**4.** This is the processing for the pin that is unused (the input is left open) when the target device operates (when QB-MINI2 is not connected). (This processing is not required if the pin is set to output.)

**5.** This connection is designed assuming that the reset signal is output from the N-ch open-drain buffer (output resistance: 100 Ω or less).

## 26.2 On-Chip Debug Security ID

The 78K0/Kx2-L microcontrollers have an on-chip debug operation control bit in the flash memory at 0084H (refer to **CHAPTER 24 OPTION BYTE**) and an on-chip debug security ID setting area at 0085H to 008EH, to prevent third parties from reading memory content.

When the boot swap function is used, also set a value that is the same as that of 1084H and 1085H to 108EH in advance, because 0084H, 0085H to 008EH and 1083H, and 1085H to 108EH are switched.

For details on the on-chip debug security ID, refer to the **QB-MINI2 On-Chip Debug Emulator with Programming Function User's Manual (U18371E)**.

**Table 26-1. On-Chip Debug Security ID**

| Address | On-Chip Debug Security ID |
|---|---|
| 0085H to 008EH | Any ID code of 10 bytes |
| 1085H to 108EH | |

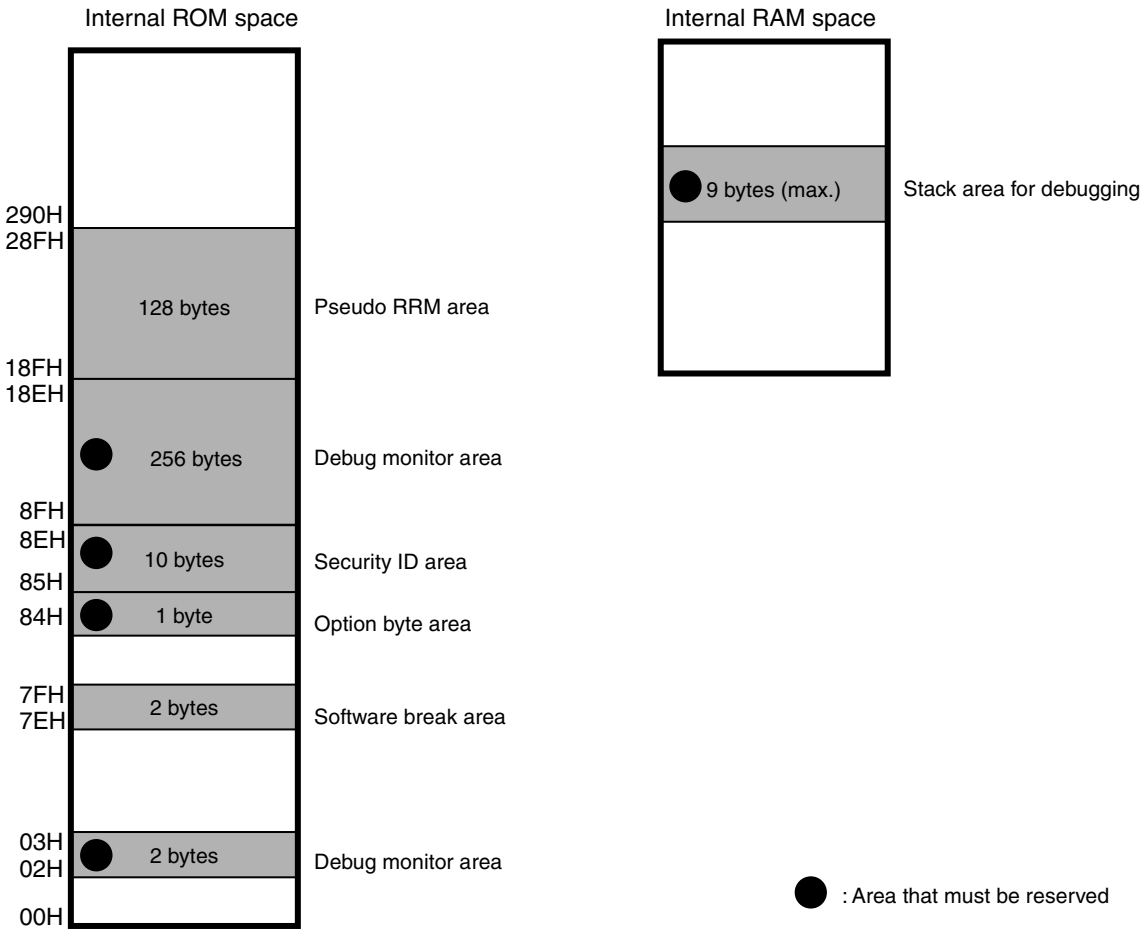
## 26.3 Securing of User Resources

QB-MINI2 uses the user memory spaces (shaded portions in Figure 26-2) to implement communication with the target device, or each debug functions. The areas marked with a dot (•) are always used for debugging, and other areas are used for each debug function used.

These areas can be secured by using user programs or the linker option.

For details on the securing of these areas, refer to the **QB-MINI2 On-Chip Debug Emulator with Programming Function User's Manual (U18371E)**.

**Figure 26-2. Reserved Area Used by QB-MINI2**



# CHAPTER 27 INSTRUCTION SET

This chapter lists each instruction set of the 78K0/Kx2-L microcontrollers in table form. For details of each operation and operation code, refer to the separate document **78K/0 Series Instructions User's Manual (U12326E)**.

## 27.1 Conventions Used in Operation List

### 27.1.1 Operand identifiers and specification methods

Operands are written in the "Operand" column of each instruction in accordance with the specification method of the instruction operand identifier (refer to the assembler specifications for details). When there are two or more methods, select one of them. Uppercase letters and the symbols #, !, $ and [ ] are keywords and must be written as they are. Each symbol has the following meaning.

- #: Immediate data specification
- !: Absolute address specification
- $: Relative address specification
- [ ]: Indirect address specification

In the case of immediate data, describe an appropriate numeric value or a label. When using a label, be sure to write the #, !, $, and [ ] symbols.

For operand register identifiers r and rp, either function names (X, A, C, etc.) or absolute names (names in parentheses in the table below, R0, R1, R2, etc.) can be used for specification.

**Table 27-1. Operand Identifiers and Specification Methods**

| Identifier | Specification Method |
|---|---|
| r | X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7) |
| rp | AX (RP0), BC (RP1), DE (RP2), HL (RP3) |
| sfr | Special function register symbol[Note] |
| sfrp | Special function register symbol (16-bit manipulatable register even addresses only)[Note] |
| saddr | FE20H to FF1FH Immediate data or labels |
| saddrp | FE20H to FF1FH Immediate data or labels (even address only) |
| addr16 | 0000H to FFFFH Immediate data or labels<br>(Only even addresses for 16-bit data transfer instructions) |
| addr11 | 0800H to 0FFFH Immediate data or labels |
| addr5 | 0040H to 007FH Immediate data or labels (even address only) |
| word | 16-bit immediate data or label |
| byte | 8-bit immediate data or label |
| bit | 3-bit immediate data or label |
| RBn | RB0 to RB3 |

**Note** Addresses from FFD0H to FFDFH cannot be accessed with these operands.

**Remark** For special function register symbols, refer to **Tables 3-6 to 3-9 Special Function Register List**.

### 27.1.2 Description of operation column

A: A register; 8-bit accumulator
X: X register
B: B register
C: C register
D: D register
E: E register
H: H register
L: L register
AX: AX register pair; 16-bit accumulator
BC: BC register pair
DE: DE register pair
HL: HL register pair
PC: Program counter
SP: Stack pointer
PSW: Program status word
CY: Carry flag
AC: Auxiliary carry flag
Z: Zero flag
RBS: Register bank select flag
IE: Interrupt request enable flag
( ): Memory contents indicated by address or register contents in parentheses
$X_H$, $X_L$: Higher 8 bits and lower 8 bits of 16-bit register
$\wedge$: Logical product (AND)
$\vee$: Logical sum (OR)
$\veebar$: Exclusive logical sum (exclusive OR)
$\overline{\ \ }$: Inverted data
addr16: 16-bit immediate data or label
jdisp8: Signed 8-bit data (displacement value)

### 27.1.3 Description of flag operation column

(Blank): Not affected
0: Cleared to 0
1: Set to 1
×: Set/cleared according to the result
R: Previously saved value is restored

## 27.2 Operation List

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| 8-bit data transfer | **MOV** | r, #byte | 2 | 4 | – | r ← byte | |
| | | saddr, #byte | 3 | 6 | 7 | (saddr) ← byte | |
| | | sfr, #byte | 3 | – | 7 | sfr ← byte | |
| | | A, r Note 3 | 1 | 2 | – | A ← r | |
| | | r, A Note 3 | 1 | 2 | – | r ← A | |
| | | A, saddr | 2 | 4 | 5 | A ← (saddr) | |
| | | saddr, A | 2 | 4 | 5 | (saddr) ← A | |
| | | A, sfr | 2 | – | 5 | A ← sfr | |
| | | sfr, A | 2 | – | 5 | sfr ← A | |
| | | A, !addr16 | 3 | 8 | 9 | A ← (addr16) | |
| | | !addr16, A | 3 | 8 | 9 | (addr16) ← A | |
| | | PSW, #byte | 3 | – | 7 | PSW ← byte | × × × |
| | | A, PSW | 2 | – | 5 | A ← PSW | |
| | | PSW, A | 2 | – | 5 | PSW ← A | × × × |
| | | A, [DE] | 1 | 4 | 5 | A ← (DE) | |
| | | [DE], A | 1 | 4 | 5 | (DE) ← A | |
| | | A, [HL] | 1 | 4 | 5 | A ← (HL) | |
| | | [HL], A | 1 | 4 | 5 | (HL) ← A | |
| | | A, [HL + byte] | 2 | 8 | 9 | A ← (HL + byte) | |
| | | [HL + byte], A | 2 | 8 | 9 | (HL + byte) ← A | |
| | | A, [HL + B] | 1 | 6 | 7 | A ← (HL + B) | |
| | | [HL + B], A | 1 | 6 | 7 | (HL + B) ← A | |
| | | A, [HL + C] | 1 | 6 | 7 | A ← (HL + C) | |
| | | [HL + C], A | 1 | 6 | 7 | (HL + C) ← A | |
| | **XCH** | A, r Note 3 | 1 | 2 | – | A ↔ r | |
| | | A, saddr | 2 | 4 | 6 | A ↔ (saddr) | |
| | | A, sfr | 2 | – | 6 | A ↔ (sfr) | |
| | | A, !addr16 | 3 | 8 | 10 | A ↔ (addr16) | |
| | | A, [DE] | 1 | 4 | 6 | A ↔ (DE) | |
| | | A, [HL] | 1 | 4 | 6 | A ↔ (HL) | |
| | | A, [HL + byte] | 2 | 8 | 10 | A ↔ (HL + byte) | |
| | | A, [HL + B] | 2 | 8 | 10 | A ↔ (HL + B) | |
| | | A, [HL + C] | 2 | 8 | 10 | A ↔ (HL + C) | |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**3.** Except “r = A”

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| 16-bit data transfer | **MOVW** | rp, #word | 3 | 6 | – | rp ← word | |
| | | saddrp, #word | 4 | 8 | 10 | (saddrp) ← word | |
| | | sfrp, #word | 4 | – | 10 | sfrp ← word | |
| | | AX, saddrp | 2 | 6 | 8 | AX ← (saddrp) | |
| | | saddrp, AX | 2 | 6 | 8 | (saddrp) ← AX | |
| | | AX, sfrp | 2 | – | 8 | AX ← sfrp | |
| | | sfrp, AX | 2 | – | 8 | sfrp ← AX | |
| | | AX, rp Note 3 | 1 | 4 | – | AX ← rp | |
| | | rp, AX Note 3 | 1 | 4 | – | rp ← AX | |
| | | AX, !addr16 | 3 | 10 | 12 | AX ← (addr16) | |
| | | !addr16, AX | 3 | 10 | 12 | (addr16) ← AX | |
| | **XCHW** | AX, rp Note 3 | 1 | 4 | – | AX ↔ rp | |
| 8-bit operation | **ADD** | A, #byte | 2 | 4 | – | A, CY ← A + byte | × × × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr), CY ← (saddr) + byte | × × × |
| | | A, r Note 4 | 2 | 4 | – | A, CY ← A + r | × × × |
| | | r, A | 2 | 4 | – | r, CY ← r + A | × × × |
| | | A, saddr | 2 | 4 | 5 | A, CY ← A + (saddr) | × × × |
| | | A, !addr16 | 3 | 8 | 9 | A, CY ← A + (addr16) | × × × |
| | | A, [HL] | 1 | 4 | 5 | A, CY ← A + (HL) | × × × |
| | | A, [HL + byte] | 2 | 8 | 9 | A, CY ← A + (HL + byte) | × × × |
| | | A, [HL + B] | 2 | 8 | 9 | A, CY ← A + (HL + B) | × × × |
| | | A, [HL + C] | 2 | 8 | 9 | A, CY ← A + (HL + C) | × × × |
| | **ADDC** | A, #byte | 2 | 4 | – | A, CY ← A + byte + CY | × × × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr), CY ← (saddr) + byte + CY | × × × |
| | | A, r Note 4 | 2 | 4 | – | A, CY ← A + r + CY | × × × |
| | | r, A | 2 | 4 | – | r, CY ← r + A + CY | × × × |
| | | A, saddr | 2 | 4 | 5 | A, CY ← A + (saddr) + CY | × × × |
| | | A, !addr16 | 3 | 8 | 9 | A, CY ← A + (addr16) + C | × × × |
| | | A, [HL] | 1 | 4 | 5 | A, CY ← A + (HL) + CY | × × × |
| | | A, [HL + byte] | 2 | 8 | 9 | A, CY ← A + (HL + byte) + CY | × × × |
| | | A, [HL + B] | 2 | 8 | 9 | A, CY ← A + (HL + B) + CY | × × × |
| | | A, [HL + C] | 2 | 8 | 9 | A, CY ← A + (HL + C) + CY | × × × |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**3.** Only when rp = BC, DE or HL

**4.** Except “r = A”

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| 8-bit operation | **SUB** | A, #byte | 2 | 4 | – | A, CY ← A – byte | × × × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr), CY ← (saddr) – byte | × × × |
| | | A, r Note 3 | 2 | 4 | – | A, CY ← A – r | × × × |
| | | r, A | 2 | 4 | – | r, CY ← r – A | × × × |
| | | A, saddr | 2 | 4 | 5 | A, CY ← A – (saddr) | × × × |
| | | A, !addr16 | 3 | 8 | 9 | A, CY ← A – (addr16) | × × × |
| | | A, [HL] | 1 | 4 | 5 | A, CY ← A – (HL) | × × × |
| | | A, [HL + byte] | 2 | 8 | 9 | A, CY ← A – (HL + byte) | × × × |
| | | A, [HL + B] | 2 | 8 | 9 | A, CY ← A – (HL + B) | × × × |
| | | A, [HL + C] | 2 | 8 | 9 | A, CY ← A – (HL + C) | × × × |
| | **SUBC** | A, #byte | 2 | 4 | – | A, CY ← A – byte – CY | × × × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr), CY ← (saddr) – byte – CY | × × × |
| | | A, r Note 3 | 2 | 4 | – | A, CY ← A – r – CY | × × × |
| | | r, A | 2 | 4 | – | r, CY ← r – A – CY | × × × |
| | | A, saddr | 2 | 4 | 5 | A, CY ← A – (saddr) – CY | × × × |
| | | A, !addr16 | 3 | 8 | 9 | A, CY ← A – (addr16) – CY | × × × |
| | | A, [HL] | 1 | 4 | 5 | A, CY ← A – (HL) – CY | × × × |
| | | A, [HL + byte] | 2 | 8 | 9 | A, CY ← A – (HL + byte) – CY | × × × |
| | | A, [HL + B] | 2 | 8 | 9 | A, CY ← A – (HL + B) – CY | × × × |
| | | A, [HL + C] | 2 | 8 | 9 | A, CY ← A – (HL + C) – CY | × × × |
| | **AND** | A, #byte | 2 | 4 | – | A ← A ∧ byte | × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr) ← (saddr) ∧ byte | × |
| | | A, r Note 3 | 2 | 4 | – | A ← A ∧ r | × |
| | | r, A | 2 | 4 | – | r ← r ∧ A | × |
| | | A, saddr | 2 | 4 | 5 | A ← A ∧ (saddr) | × |
| | | A, !addr16 | 3 | 8 | 9 | A ← A ∧ (addr16) | × |
| | | A, [HL] | 1 | 4 | 5 | A ← A ∧ (HL) | × |
| | | A, [HL + byte] | 2 | 8 | 9 | A ← A ∧ (HL + byte) | × |
| | | A, [HL + B] | 2 | 8 | 9 | A ← A ∧ (HL + B) | × |
| | | A, [HL + C] | 2 | 8 | 9 | A ← A ∧ (HL + C) | × |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**3.** Except “r = A”

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| 8-bit operation | **OR** | A, #byte | 2 | 4 | – | A ← A ∨ byte | × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr) ← (saddr) ∨ byte | × |
| | | A, r Note 3 | 2 | 4 | – | A ← A ∨ r | × |
| | | r, A | 2 | 4 | – | r ← r ∨ A | × |
| | | A, saddr | 2 | 4 | 5 | A ← A ∨ (saddr) | × |
| | | A, !addr16 | 3 | 8 | 9 | A ← A ∨ (addr16) | × |
| | | A, [HL] | 1 | 4 | 5 | A ← A ∨ (HL) | × |
| | | A, [HL + byte] | 2 | 8 | 9 | A ← A ∨ (HL + byte) | × |
| | | A, [HL + B] | 2 | 8 | 9 | A ← A ∨ (HL + B) | × |
| | | A, [HL + C] | 2 | 8 | 9 | A ← A ∨ (HL + C) | × |
| | **XOR** | A, #byte | 2 | 4 | – | A ← A ⊻ byte | × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr) ← (saddr) ⊻ byte | × |
| | | A, r Note 3 | 2 | 4 | – | A ← A ⊻ r | × |
| | | r, A | 2 | 4 | – | r ← r ⊻ A | × |
| | | A, saddr | 2 | 4 | 5 | A ← A ⊻ (saddr) | × |
| | | A, !addr16 | 3 | 8 | 9 | A ← A ⊻ (addr16) | × |
| | | A, [HL] | 1 | 4 | 5 | A ← A ⊻ (HL) | × |
| | | A, [HL + byte] | 2 | 8 | 9 | A ← A ⊻ (HL + byte) | × |
| | | A, [HL + B] | 2 | 8 | 9 | A ← A ⊻ (HL + B) | × |
| | | A, [HL + C] | 2 | 8 | 9 | A ← A ⊻ (HL + C) | × |
| | **CMP** | A, #byte | 2 | 4 | – | A – byte | × × × |
| | | saddr, #byte | 3 | 6 | 8 | (saddr) – byte | × × × |
| | | A, r Note 3 | 2 | 4 | – | A – r | × × × |
| | | r, A | 2 | 4 | – | r – A | × × × |
| | | A, saddr | 2 | 4 | 5 | A – (saddr) | × × × |
| | | A, !addr16 | 3 | 8 | 9 | A – (addr16) | × × × |
| | | A, [HL] | 1 | 4 | 5 | A – (HL) | × × × |
| | | A, [HL + byte] | 2 | 8 | 9 | A – (HL + byte) | × × × |
| | | A, [HL + B] | 2 | 8 | 9 | A – (HL + B) | × × × |
| | | A, [HL + C] | 2 | 8 | 9 | A – (HL + C) | × × × |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**3.** Except “r = A”

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| 16-bit operation | **ADDW** | AX, #word | 3 | 6 | – | AX, CY ← AX + word | × × × |
| | **SUBW** | AX, #word | 3 | 6 | – | AX, CY ← AX – word | × × × |
| | **CMPW** | AX, #word | 3 | 6 | – | AX – word | × × × |
| Multiply/ divide | **MULU** | X | 2 | 16 | – | AX ← A × X | |
| | **DIVUW** | C | 2 | 25 | – | AX (Quotient), C (Remainder) ← AX ÷ C | |
| Increment/ decrement | **INC** | r | 1 | 2 | – | r ← r + 1 | × × |
| | | saddr | 2 | 4 | 6 | (saddr) ← (saddr) + 1 | × × |
| | **DEC** | r | 1 | 2 | – | r ← r – 1 | × × |
| | | saddr | 2 | 4 | 6 | (saddr) ← (saddr) – 1 | × × |
| | **INCW** | rp | 1 | 4 | – | rp ← rp + 1 | |
| | **DECW** | rp | 1 | 4 | – | rp ← rp – 1 | |
| Rotate | **ROR** | A, 1 | 1 | 2 | – | (CY, $A_7$ ← $A_0$, $A_{m-1}$ ← $A_m$) × 1 time | × |
| | **ROL** | A, 1 | 1 | 2 | – | (CY, $A_0$ ← $A_7$, $A_{m+1}$ ← $A_m$) × 1 time | × |
| | **RORC** | A, 1 | 1 | 2 | – | (CY ← $A_0$, $A_7$ ← CY, $A_{m-1}$ ← $A_m$) × 1 time | × |
| | **ROLC** | A, 1 | 1 | 2 | – | (CY ← $A_7$, $A_0$ ← CY, $A_{m+1}$ ← $A_m$) × 1 time | × |
| | **ROR4** | [HL] | 2 | 10 | 12 | $A_{3-0}$ ← $(HL)_{3-0}$, $(HL)_{7-4}$ ← $A_{3-0}$, $(HL)_{3-0}$ ← $(HL)_{7-4}$ | |
| | **ROL4** | [HL] | 2 | 10 | 12 | $A_{3-0}$ ← $(HL)_{7-4}$, $(HL)_{3-0}$ ← $A_{3-0}$, $(HL)_{7-4}$ ← $(HL)_{3-0}$ | |
| BCD adjustment | **ADJBA** | | 2 | 4 | – | Decimal Adjust Accumulator after Addition | × × × |
| | **ADJBS** | | 2 | 4 | – | Decimal Adjust Accumulator after Subtract | × × × |
| Bit manipulate | **MOV1** | CY, saddr.bit | 3 | 6 | 7 | CY ← (saddr.bit) | × |
| | | CY, sfr.bit | 3 | – | 7 | CY ← sfr.bit | × |
| | | CY, A.bit | 2 | 4 | – | CY ← A.bit | × |
| | | CY, PSW.bit | 3 | – | 7 | CY ← PSW.bit | × |
| | | CY, [HL].bit | 2 | 6 | 7 | CY ← (HL).bit | × |
| | | saddr.bit, CY | 3 | 6 | 8 | (saddr.bit) ← CY | |
| | | sfr.bit, CY | 3 | – | 8 | sfr.bit ← CY | |
| | | A.bit, CY | 2 | 4 | – | A.bit ← CY | |
| | | PSW.bit, CY | 3 | – | 8 | PSW.bit ← CY | × × |
| | | [HL].bit, CY | 2 | 6 | 8 | (HL).bit ← CY | |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| Bit manipulate | **AND1** | CY, saddr.bit | 3 | 6 | 7 | CY ← CY ∧ (saddr.bit) | × |
| | | CY, sfr.bit | 3 | – | 7 | CY ← CY ∧ sfr.bit | × |
| | | CY, A.bit | 2 | 4 | – | CY ← CY ∧ A.bit | × |
| | | CY, PSW.bit | 3 | – | 7 | CY ← CY ∧ PSW.bit | × |
| | | CY, [HL].bit | 2 | 6 | 7 | CY ← CY ∧ (HL).bit | × |
| | **OR1** | CY, saddr.bit | 3 | 6 | 7 | CY ← CY ∨ (saddr.bit) | × |
| | | CY, sfr.bit | 3 | – | 7 | CY ← CY ∨ sfr.bit | × |
| | | CY, A.bit | 2 | 4 | – | CY ← CY ∨ A.bit | × |
| | | CY, PSW.bit | 3 | – | 7 | CY ← CY ∨ PSW.bit | × |
| | | CY, [HL].bit | 2 | 6 | 7 | CY ← CY ∨ (HL).bit | × |
| | **XOR1** | CY, saddr.bit | 3 | 6 | 7 | CY ← CY ⊻ (saddr.bit) | × |
| | | CY, sfr.bit | 3 | – | 7 | CY ← CY ⊻ sfr.bit | × |
| | | CY, A.bit | 2 | 4 | – | CY ← CY ⊻ A.bit | × |
| | | CY, PSW. bit | 3 | – | 7 | CY ← CY ⊻ PSW.bit | × |
| | | CY, [HL].bit | 2 | 6 | 7 | CY ← CY ⊻ (HL).bit | × |
| | **SET1** | saddr.bit | 2 | 4 | 6 | (saddr.bit) ← 1 | |
| | | sfr.bit | 3 | – | 8 | sfr.bit ← 1 | |
| | | A.bit | 2 | 4 | – | A.bit ← 1 | |
| | | PSW.bit | 2 | – | 6 | PSW.bit ← 1 | × × × |
| | | [HL].bit | 2 | 6 | 8 | (HL).bit ← 1 | |
| | **CLR1** | saddr.bit | 2 | 4 | 6 | (saddr.bit) ← 0 | |
| | | sfr.bit | 3 | – | 8 | sfr.bit ← 0 | |
| | | A.bit | 2 | 4 | – | A.bit ← 0 | |
| | | PSW.bit | 2 | – | 6 | PSW.bit ← 0 | × × × |
| | | [HL].bit | 2 | 6 | 8 | (HL).bit ← 0 | |
| | **SET1** | CY | 1 | 2 | – | CY ← 1 | 1 |
| | **CLR1** | CY | 1 | 2 | – | CY ← 0 | 0 |
| | **NOT1** | CY | 1 | 2 | – | CY ← $\overline{CY}$ | × |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| Call/return | **CALL** | !addr16 | 3 | 7 | – | (SP – 1) ← (PC + 3)H, (SP – 2) ← (PC + 3)L, PC ← addr16, SP ← SP – 2 | |
| | **CALLF** | !addr11 | 2 | 5 | – | (SP – 1) ← (PC + 2)H, (SP – 2) ← (PC + 2)L, PC15 – 11 ← 00001, PC10 – 0 ← addr11, SP ← SP – 2 | |
| | **CALLT** | [addr5] | 1 | 6 | – | (SP – 1) ← (PC + 1)H, (SP – 2) ← (PC + 1)L, PCH ← (addr5 + 1), PCL ← (addr5), SP ← SP – 2 | |
| | **BRK** | | 1 | 6 | – | (SP – 1) ← PSW, (SP – 2) ← (PC + 1)H, (SP – 3) ← (PC + 1)L, PCH ← (003FH), PCL ← (003EH), SP ← SP – 3, IE ← 0 | |
| | **RET** | | 1 | 6 | – | PCH ← (SP + 1), PCL ← (SP), SP ← SP + 2 | |
| | **RETI** | | 1 | 6 | – | PCH ← (SP + 1), PCL ← (SP), PSW ← (SP + 2), SP ← SP + 3 | R R R |
| | **RETB** | | 1 | 6 | – | PCH ← (SP + 1), PCL ← (SP), PSW ← (SP + 2), SP ← SP + 3 | R R R |
| Stack manipulate | **PUSH** | PSW | 1 | 2 | – | (SP – 1) ← PSW, SP ← SP – 1 | |
| | | rp | 1 | 4 | – | (SP – 1) ← rpH, (SP – 2) ← rpL, SP ← SP – 2 | |
| | **POP** | PSW | 1 | 2 | – | PSW ← (SP), SP ← SP + 1 | R R R |
| | | rp | 1 | 4 | – | rpH ← (SP + 1), rpL ← (SP), SP ← SP + 2 | |
| | **MOVW** | SP, #word | 4 | – | 10 | SP ← word | |
| | | SP, AX | 2 | – | 8 | SP ← AX | |
| | | AX, SP | 2 | – | 8 | AX ← SP | |
| Unconditional branch | **BR** | !addr16 | 3 | 6 | – | PC ← addr16 | |
| | | $addr16 | 2 | 6 | – | PC ← PC + 2 + jdisp8 | |
| | | AX | 2 | 8 | – | PCH ← A, PCL ← X | |
| Conditional branch | **BC** | $addr16 | 2 | 6 | – | PC ← PC + 2 + jdisp8 if CY = 1 | |
| | **BNC** | $addr16 | 2 | 6 | – | PC ← PC + 2 + jdisp8 if CY = 0 | |
| | **BZ** | $addr16 | 2 | 6 | – | PC ← PC + 2 + jdisp8 if Z = 1 | |
| | **BNZ** | $addr16 | 2 | 6 | – | PC ← PC + 2 + jdisp8 if Z = 0 | |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock (fCPU) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

| Instruction Group | Mnemonic | Operands | Bytes | Clocks | | Operation | Flag |
|---|---|---|---|---|---|---|---|
| | | | | Note 1 | Note 2 | | Z AC CY |
| Conditional branch | **BT** | saddr.bit, $addr16 | 3 | 8 | 9 | PC ← PC + 3 + jdisp8 if (saddr.bit) = 1 | |
| | | sfr.bit, $addr16 | 4 | – | 11 | PC ← PC + 4 + jdisp8 if sfr.bit = 1 | |
| | | A.bit, $addr16 | 3 | 8 | – | PC ← PC + 3 + jdisp8 if A.bit = 1 | |
| | | PSW.bit, $addr16 | 3 | – | 9 | PC ← PC + 3 + jdisp8 if PSW.bit = 1 | |
| | | [HL].bit, $addr16 | 3 | 10 | 11 | PC ← PC + 3 + jdisp8 if (HL).bit = 1 | |
| | **BF** | saddr.bit, $addr16 | 4 | 10 | 11 | PC ← PC + 4 + jdisp8 if (saddr.bit) = 0 | |
| | | sfr.bit, $addr16 | 4 | – | 11 | PC ← PC + 4 + jdisp8 if sfr.bit = 0 | |
| | | A.bit, $addr16 | 3 | 8 | – | PC ← PC + 3 + jdisp8 if A.bit = 0 | |
| | | PSW.bit, $addr16 | 4 | – | 11 | PC ← PC + 4 + jdisp8 if PSW. bit = 0 | |
| | | [HL].bit, $addr16 | 3 | 10 | 11 | PC ← PC + 3 + jdisp8 if (HL).bit = 0 | |
| | **BTCLR** | saddr.bit, $addr16 | 4 | 10 | 12 | PC ← PC + 4 + jdisp8 if (saddr.bit) = 1<br>then reset (saddr.bit) | |
| | | sfr.bit, $addr16 | 4 | – | 12 | PC ← PC + 4 + jdisp8 if sfr.bit = 1<br>then reset sfr.bit | |
| | | A.bit, $addr16 | 3 | 8 | – | PC ← PC + 3 + jdisp8 if A.bit = 1<br>then reset A.bit | |
| | | PSW.bit, $addr16 | 4 | – | 12 | PC ← PC + 4 + jdisp8 if PSW.bit = 1<br>then reset PSW.bit | × × × |
| | | [HL].bit, $addr16 | 3 | 10 | 12 | PC ← PC + 3 + jdisp8 if (HL).bit = 1<br>then reset (HL).bit | |
| | **DBNZ** | B, $addr16 | 2 | 6 | – | B ← B – 1, then<br>PC ← PC + 2 + jdisp8 if B ≠ 0 | |
| | | C, $addr16 | 2 | 6 | – | C ← C –1, then<br>PC ← PC + 2 + jdisp8 if C ≠ 0 | |
| | | saddr, $addr16 | 3 | 8 | 10 | (saddr) ← (saddr) – 1, then<br>PC ← PC + 3 + jdisp8 if (saddr) ≠ 0 | |
| CPU control | **SEL** | RBn | 2 | 4 | – | RBS1, 0 ← n | |
| | **NOP** | | 1 | 2 | – | No Operation | |
| | **EI** | | 2 | – | 6 | IE ← 1 (Enable Interrupt) | |
| | **DI** | | 2 | – | 6 | IE ← 0 (Disable Interrupt) | |
| | **HALT** | | 2 | 6 | – | Set HALT Mode | |
| | **STOP** | | 2 | 6 | – | Set STOP Mode | |

**Notes 1.** When the internal high-speed RAM area is accessed or for an instruction with no data access

**2.** When an area except the internal high-speed RAM area is accessed

**Remarks 1.** One instruction clock cycle is one cycle of the CPU clock ($f_{CPU}$) selected by the processor clock control register (PCC).

**2.** This clock cycle applies to the internal ROM program.

## 27.3 Instructions Listed by Addressing Type

### (1) 8-bit instructions

MOV, XCH, ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP, MULU, DIVUW, INC, DEC, ROR, ROL, RORC, ROLC, ROR4, ROL4, PUSH, POP, DBNZ

| First Operand \ Second Operand | #byte | A | r[Note] | sfr | saddr | !addr16 | PSW | [DE] | [HL] | [HL + byte]<br>[HL + B]<br>[HL + C] | $addr16 | 1 | None |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | | MOV<br>XCH<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | MOV<br>XCH | MOV<br>XCH<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | MOV<br>XCH<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | MOV | MOV<br>XCH | MOV<br>XCH<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | MOV<br>XCH<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | | ROR<br>ROL<br>RORC<br>ROLC | |
| r | MOV | MOV<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | | | | | | | | | | | INC<br>DEC |
| B, C | | | | | | | | | | | DBNZ | | |
| sfr | MOV | MOV | | | | | | | | | | | |
| saddr | MOV<br>ADD<br>ADDC<br>SUB<br>SUBC<br>AND<br>OR<br>XOR<br>CMP | MOV | | | | | | | | | DBNZ | | INC<br>DEC |
| !addr16 | | MOV | | | | | | | | | | | |
| PSW | MOV | MOV | | | | | | | | | | | PUSH<br>POP |
| [DE] | | MOV | | | | | | | | | | | |
| [HL] | | MOV | | | | | | | | | | | ROR4<br>ROL4 |
| [HL + byte]<br>[HL + B]<br>[HL + C] | | MOV | | | | | | | | | | | |
| X | | | | | | | | | | | | | MULU |
| C | | | | | | | | | | | | | DIVUW |

**Note** Except “r = A”

### (2) 16-bit instructions

MOVW, XCHW, ADDW, SUBW, CMPW, PUSH, POP, INCW, DECW

| Second Operand / First Operand | #word | AX | rp[Note] | sfrp | saddrp | !addr16 | SP | None |
|---|---|---|---|---|---|---|---|---|
| AX | ADDW<br>SUBW<br>CMPW | | MOVW<br>XCHW | MOVW | MOVW | MOVW | MOVW | |
| rp | MOVW | MOVW[Note] | | | | | | INCW<br>DECW<br>PUSH<br>POP |
| sfrp | MOVW | MOVW | | | | | | |
| saddrp | MOVW | MOVW | | | | | | |
| !addr16 | | MOVW | | | | | | |
| SP | MOVW | MOVW | | | | | | |

**Note** Only when rp = BC, DE, HL

### (3) Bit manipulation instructions

MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1, BT, BF, BTCLR

| Second Operand / First Operand | A.bit | sfr.bit | saddr.bit | PSW.bit | [HL].bit | CY | $addr16 | None |
|---|---|---|---|---|---|---|---|---|
| A.bit | | | | | | MOV1 | BT<br>BF<br>BTCLR | SET1<br>CLR1 |
| sfr.bit | | | | | | MOV1 | BT<br>BF<br>BTCLR | SET1<br>CLR1 |
| saddr.bit | | | | | | MOV1 | BT<br>BF<br>BTCLR | SET1<br>CLR1 |
| PSW.bit | | | | | | MOV1 | BT<br>BF<br>BTCLR | SET1<br>CLR1 |
| [HL].bit | | | | | | MOV1 | BT<br>BF<br>BTCLR | SET1<br>CLR1 |
| CY | MOV1<br>AND1<br>OR1<br>XOR1 | MOV1<br>AND1<br>OR1<br>XOR1 | MOV1<br>AND1<br>OR1<br>XOR1 | MOV1<br>AND1<br>OR1<br>XOR1 | MOV1<br>AND1<br>OR1<br>XOR1 | | | SET1<br>CLR1<br>NOT1 |

**(4) Call instructions/branch instructions**

CALL, CALLF, CALLT, BR, BC, BNC, BZ, BNZ, BT, BF, BTCLR, DBNZ

| First Operand \ Second Operand | AX | !addr16 | !addr11 | [addr5] | $addr16 |
|---|---|---|---|---|---|
| Basic instruction | BR | CALL<br>BR | CALLF | CALLT | BR<br>BC<br>BNC<br>BZ<br>BNZ |
| Compound instruction | | | | | BT<br>BF<br>BTCLR<br>DBNZ |

**(5) Other instructions**

ADJBA, ADJBS, BRK, RET, RETI, RETB, SEL, NOP, EI, DI, HALT, STOP

# CHAPTER 28 ELECTRICAL SPECIFICATIONS

**Target products: 78K0/KY2-L: μPD78F0550, 78F0551, 78F0552, 78F0555, 78F0556, 78F0557**
**78K0/KA2-L: μPD78F0560, 78F0561, 78F0562, 78F0565, 78F0566, 78F0567**
**78K0/KB2-L: μPD78F0571, 78F0572, 78F0573, 78F0576, 78F0577, 78F0578**
**78K0/KC2-L: μPD78F0581, 78F0582, 78F0583, 78F0586, 78F0587, 78F0588**

**Cautions 1. The 78K0/Kx2-L microcontrollers have an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.**

**2. The pins mounted depend on the product as follows.**

<R> **(1) Port functions**

| Port | 78K0/KY2-L | 78K0/KA2-L | | | 78K0/KB2-L | 78K0/KC2-L | | |
|---|---|---|---|---|---|---|---|---|
| | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| Port 0 | P00, P01 | | P00, P02 | P01, P02 | P00, P01 | | | P00 to P02 |
| Port 1 | – | | | | P10 to P17 | | | |
| Port 2 | P20 to P23 | P20 to P25 | P20 to P26 | P20 to P27 | P20 to P23 | P20 to P26 | P20 to P27 | |
| Port 3 | P30 | P30 to P32 | P31 to P37 | | P30 to P33 | | | |
| Port 4 | – | | | | | | P40, P41 | P40 to P42 |
| Port 6 | P60, P61 | | | | | P60 to P62 | P60 to P63 | |
| Port 7 | – | | | P70 to P73 | – | P70 to P73 | | P70 to P75 |
| Port 12 | P121, P122, P125 | | | | P120 to P122, P125 | P120 to P125 | | |

(The remaining table is on the next page.)

<R> **(2) Non-port functions**

| Port | | 78K0/KY2-L | 78K0/KA2-L | | | 78K0/KB2-L | 78K0/KC2-L | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 16 Pins | 20 Pins | 25 Pins | 32 Pins | 30 Pins | 40 Pins | 44 Pins | 48 Pins |
| Power supply, ground | | $V_{DD}$, $V_{SS}$, $AV_{REF}$ | | | | $V_{DD}$, $V_{SS}$, $AV_{REF}$, $AV_{SS}$ | | | |
| Regulator | | REGC | | | | | | | |
| Reset | | $\overline{\text{RESET}}$ | | | | | | | |
| Clock oscillation | | X1, X2, EXCLK | | | | | X1, X2, EXCLK, XT1, XT2, EXCLKS | | |
| Interrupt | | INTP0, INTP1 | INTP0 to INTP3 | INTP0, INTP2 to INTP5 | | INTP0 to INTP5, INTP10, INTP11 | INTP0 to INTP5, INTP9 to INTP11 | INTP0 to INTP5, INTP8 to INTP11 | INTP0 to INTP11 |
| Key interrupt | | – | | | | | KR0 to KR3 | | KR0 to KR5 |
| Timer | TM00 | TI000, TI010, TO00 | | TI000 | (TI000) TI010, TO00 | TI000, TI010, TO00 | | | |
| | TM5x | TI51 | | (TI51) | – | TI50, TO50, TI51, TO51 | | | |
| | TMHx | TOH1 | | (TOH1) | | TOH0, TOH1 | | | |
| | RTC | – | | | | | | RTC1HZ, RTCCL, RTCDIV | |
| Serial interface | UART6 | RxD6, TxD6 | | | | | | | |
| | IICA | SCLA0, SDAA0 | | | | | | | |
| | CSI10 | – | | | | $\overline{\text{SCK10}}$, SI10, SO10 | | | |
| | CSI11 | – | | $\overline{\text{SCK11}}$, SI11, SO11, $\overline{\text{SSI11}}$ | | – | $\overline{\text{SCK11}}$, SI11, SO11 | | $\overline{\text{SCK11}}$, SI11, SO11, $\overline{\text{SSI11}}$ |
| A/D converter | | ANI0 to ANI3 | ANI0 to ANI5 | ANI0 to ANI6 | ANI0 to ANI10 | ANI0 to ANI3, ANI8 to ANI10 | ANI0 to ANI6, ANI8 to ANI10 | ANI0 to ANI10 | |
| Operational amplifier[Note] | | AMP0+, AMP0-, AMP0OUT, PGAIN | | | | AMP0+, AMP0-, AMP0OUT, PGAIN, AMP1+, AMP1-, AMP1OUT | | | |
| Clock output | | – | | | | | | | PCL |
| Low-voltage detector (LVI) | | – | | | | EXLVI | | | |
| On-chip debug function | | TOOLC0, TOOLD0 | TOOLC0, TOOLC1, TOOLD0, TOOLD1 | | | | | | |

**Note** Products with operational amplifier only.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**Absolute Maximum Ratings ($T_A$ = 25°C) (78K0/KY2-L, 78K0/KA2-L (20 pins), 78K0/KB2-L, 78K0/KC2-L) (1/2)**

| Parameter | Symbol | Conditions | Ratings | Unit |
|---|---|---|---|---|
| Supply voltage | $V_{DD}$ | | −0.5 to +6.5 | V |
| | $V_{SS}$ | | −0.5 to +0.3 | V |
| | $AV_{REF}$ | | −0.5 to $V_{DD}$ + 0.3 Note 1 | V |
| | $AV_{SS}$ | | −0.5 to +0.3 | V |
| REGC pin input voltage Note 2 | $V_{IREGC}$ | | −0.5 to +3.6 and −0.5 to $V_{DD}$ +0.3 | V |
| Input voltage | $V_{I1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 to P125, X1, X2, XT1, XT2, $\overline{\text{RESET}}$ | −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| | $V_{I2}$ | P20 to P27 | −0.3 to $AV_{REF}$ + 0.3 Note 1 and −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| Output voltage | $V_{O1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 | −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| | $V_{O2}$ | P20 to P27 | −0.3 to $AV_{REF}$ + 0.3 Note 1 | V |
| Analog input voltage | $V_{AN1}$ | ANI0 to ANI7, AMP0+, AMP0- | −0.3 to $AV_{REF}$ + 0.3 Note 1 and −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| | $V_{AN2}$ | ANI8 to ANI10, AMP1+, AMP1- | −0.3 to $V_{DD}$ + 0.3 Note 1 | V |

**Notes 1.** Must be 6.5 V or lower.

**2.** Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 μF). This value regulates the absolute maximum rating of the REGC pin. Do not use this pin with voltage applied to it.

**Caution Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.**

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**Absolute Maximum Ratings ($T_A$ = 25°C) (78K0/KY2-L, 78K0/KA2-L (20 pins), 78K0/KB2-L, 78K0/KC2-L) (2/2)**

| Parameter | Symbol | Conditions | | Ratings | Unit |
|---|---|---|---|---|---|
| Output current, high | $I_{OH1}$ | Per pin | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 | −10 | mA |
| | | Total of all pins −80 mA | P00 to P02, P40 to P42, P120 | −25 | mA |
| | | | P10 to P17, P30 to P33, P60 to P63, P70 to P75 | −55 | mA |
| | $I_{OH2}$ | Per pin | P20 to P27 | −0.5 | mA |
| | | Total of all pins | | −2 | mA |
| Output current, low | $I_{OL1}$ | Per pin | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 | 30 | mA |
| | | Total of all pins 200 mA | P00 to P02, P40 to P42, P120 | 60 | mA |
| | | | P10 to P17, P30 to P33, P60 to P63, P70 to P75 | 140 | mA |
| | $I_{OL2}$ | Per pin | P20 to P27 | 1 | mA |
| | | Total of all pins | | 5 | mA |
| Operating ambient temperature | $T_A$ | | | −40 to +85 | °C |
| Storage temperature | $T_{stg}$ | | | −65 to +150 | °C |

**Cautions 1. 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.**

**2. The value of the current that can be run per pin must satisfy the value of the current per pin and the total value of the currents of all pins.**

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

<R> **Absolute Maximum Ratings ($T_A$ = 25°C) (78K0/KA2-L (25 pins, 32 pins))**

| Parameter | Symbol | Conditions | | Ratings | Unit |
|---|---|---|---|---|---|
| Supply voltage | $V_{DD}$ | | | −0.5 to +6.5 | V |
| | $V_{SS}$ | | | −0.5 to +0.3 | V |
| | $AV_{REF}$ | | | −0.5 to $V_{DD}$ + 0.3 Note 1 | V |
| | $AV_{SS}$ | | | −0.5 to +0.3 | V |
| REGC pin input voltage Note 2 | $V_{IREGC}$ | | | −0.5 to +3.6 and −0.5 to $V_{DD}$ +0.3 | V |
| Input voltage | $V_{I1}$ | P00 to P02, P31 to P37, P60, P61, P121, P122, P125, X1, X2, $\overline{\text{RESET}}$ | | −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| | $V_{I2}$ | P20 to P27, P70 to P72 | | −0.3 to $AV_{REF}$ + 0.3 Note 1 and −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| Output voltage | $V_{O1}$ | P00 to P02, P31 to P37, P60, P61 | | −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| | $V_{O2}$ | P20 to P27, P70 to P72 | | −0.3 to $AV_{REF}$ + 0.3 Note 1 | V |
| Analog input voltage | $V_{AN1}$ | ANI0 to ANI10, AMP0+, AMP0- | | −0.3 to $AV_{REF}$ + 0.3 Note 1 and −0.3 to $V_{DD}$ + 0.3 Note 1 | V |
| Output current, high | $I_{OH1}$ | Per pin | P00 to P02, P31 to P37, P60, P61 | −10 | mA |
| | | Total of all pins −80 mA | P02, P60, P61 | −30 | mA |
| | | | P00, P01, P31 to P37 | −55 | mA |
| | $I_{OH2}$ | Per pin | P20 to P27, P70 to P72 | −0.5 | mA |
| | | Total of all pins | | −2 | mA |
| Output current, low | $I_{OL1}$ | Per pin | P00 to P02, P31 to P37, P60, P61 | 30 | mA |
| | | Total of all pins 200 mA | P02, P60, P61 | 85 | mA |
| | | | P00, P01, P31 to P37 | 140 | mA |
| | $I_{OL2}$ | Per pin | P20 to P27, P70 to P72 | 1 | mA |
| | | Total of all pins | | 5 | mA |
| Operating ambient temperature | $T_A$ | | | −40 to +85 | °C |
| Storage temperature | $T_{stg}$ | | | −65 to +150 | °C |

**Notes 1.** Must be 6.5 V or lower.

**2.** Connect the REGC pin to $V_{SS}$ via a capacitor (0.47 to 1 μF). This value regulates the absolute maximum rating of the REGC pin. Do not use this pin with voltage applied to it.

**Cautions 1. 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.**

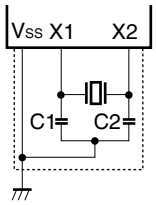
**2. The value of the current that can be run per pin must satisfy the value of the current per pin and the total value of the currents of all pins.**

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**X1 Oscillator Characteristics**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Resonator | Recommended Circuit | Parameter | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Ceramic resonator, crystal resonator |  | X1 clock oscillation frequency ($f_X$)Note | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | 1.0 | | 10.0 | MHz |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | 1.0 | | 5.0 | |

**Note** Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

**Cautions 1. When using the X1 oscillator, wire as follows in the area enclosed by the broken lines in the above figures to avoid an adverse effect from wiring capacitance.**

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines.**
- **Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as $V_{SS}$.**
- **Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

**2. Since the CPU is started by the internal high-speed oscillation clock after a reset release, check the X1 clock oscillation stabilization time using the oscillation stabilization time counter status register (OSTC) by the user. Determine the oscillation stabilization time of the OSTC register and oscillation stabilization time select register (OSTS) after sufficiently evaluating the oscillation stabilization time with the resonator to be used.**

**Remark** For the resonator selection and oscillator constant, customers are requested to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**Internal High-speed Oscillator Characteristics**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Resonator | Parameter | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Internal high-speed oscillator | Oscillation frequency ($f_{IH}$ = 4 MHz) deviation[Notes 1, 2] | RSTS = 1 | $T_A$ = –20 to +70°C | | | ±2 | % |
| | | | $T_A$ = –40 to +85°C | | | ±3 | % |
| | Oscillation frequency ($f_{IH}$ = 8 MHz) deviation[Notes 1, 2] | | $T_A$ = –40 to +85°C | | | ±3 | % |
| | Oscillation frequency ($f_{IH}$)[Note 1] | RSTS = 0 | In low power consumption mode (RMC = 56H) | 1.86 | 4.2 | 7.42 | MHz |
| | | | In normal power mode (RMC = 00H) | 1.86 | 5.0 | 8.7 | MHz |

**Notes 1.** Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

**2.** Internal high-speed oscillation frequency (4 MHz or 8 MHz) is set by the option byte. Refer to **CHAPTER 24 OPTION BYTE**.

**Internal Low-speed Oscillator Characteristics**

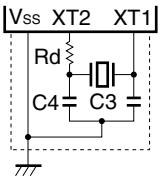
**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Resonator | Parameter | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Internal low-speed oscillator | Oscillation frequency ($f_{IL}$) | In low power consumption mode (RMC = 56H) | | 25.5 | 30 | 34.5 | kHz |
| | | In normal power mode (RMC = 00H) | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | 27 | 30 | 33 | kHz |
| | | | 1.8 V ≤ $V_{DD}$ < 5.5 V | 25.5 | 30 | 34.5 | kHz |

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**XT1 Oscillator Characteristics**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Resonator | Recommended Circuit | Parameter | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Crystal resonator |  | XT1 clock oscillation frequency ($f_{XT}$)Note | AMPHXT = 0 | 32 | 32.768 | 35 | kHz |

**Note** Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

**Cautions 1. When using the XT1 oscillator, wire as follows in the area enclosed by the broken lines in the above figure to avoid an adverse effect from wiring capacitance.**

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines.**
- **Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as $V_{SS}$.**
- **Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

**2. The XT1 oscillator is designed as a low-amplitude circuit for reducing power consumption, and is more prone to malfunction due to noise than the X1 oscillator. Particular care is therefore required with the wiring method when the XT1 clock is used.**

**Remark** For the resonator selection and oscillator constant, customers are requested to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**DC Characteristics (1/8) (78K0/KY2-L, 78K0/KA2-L (20 pins), 78K0/KB2-L, 78K0/KC2-L)**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Output current, high[Note 1] | $I_{OH1}$ | Per pin for P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | –3.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | –2.5 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | –1.0 | mA |
| | | Total of P00 to P02, P40 to P42, P120 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | –20.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | –10.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | –5.0 | mA |
| | | Total of P10 to P17, P30 to P33, P60 to P63, P70 to P75 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | –30.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | –19.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | –10.0 | mA |
| | | Total of P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120[Note 3] | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | –50.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | –29.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | –15.0 | mA |
| | $I_{OH2}$ | Per pin for P20 to P27 | $AV_{REF}$ = $V_{DD}$ | | | –0.1 | mA |
| Output current, low[Note 2] | $I_{OL1}$ | Per pin for P00 to P02, P10 to P17, P30 to P33, P40 to P42, P70 to P75, P120 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 8.5 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 5.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 2.0 | mA |
| | | Per pin for P60 to P63 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 15.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 5.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 2.0 | mA |
| | | Total of P00 to P02, P40 to P42, P120 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 20.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 15.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 9.0 | mA |
| | | Total of P10 to P17, P30 to P33, P60 to P63, P70 to P75 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 45.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 35.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 20.0 | mA |
| | | Total of P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120[Note 3] | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 65.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 50.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 29.0 | mA |
| | $I_{OL2}$ | Per pin for P20 to P27 | $AV_{REF}$ = $V_{DD}$ | | | 0.4 | mA |

**Notes 1.** Value of current at which the device operation is guaranteed even if the current flows from $V_{DD}$ to an output pin.

**2.** Value of current at which the device operation is guaranteed even if the current flows from an output pin to GND.

**3.** Specification under conditions where the duty factor is 70% (time for which current is output is 0.7 × t and time for which current is not output is 0.3 × t, where t is a specific time). The total output current of the pins at a duty factor of other than 70% can be calculated by the following expression.

- Where the duty factor of $I_{OH}$ is n%: Total output current of pins = ($I_{OH}$ × 0.7)/(n × 0.01)

<Example> Where the duty factor is 50%, $I_{OH}$ = –20.0 mA

Total output current of pins = (–20.0 × 0.7)/(50 × 0.01) = –28.0 mA

However, the current that is allowed to flow into one pin does not vary depending on the duty factor. A current higher than the absolute maximum rating must not flow into one pin.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**DC Characteristics (2/8) (78K0/KY2-L, 78K0/KA2-L (20 pins), 78K0/KB2-L, 78K0/KC2-L)**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Input voltage, high | $V_{IH1}$ | P12, P13, P15, P121 to P125 | | 0.7$V_{DD}$ | | $V_{DD}$ | V |
| | $V_{IH2}$ | P20 to P27 | $AV_{REF}$ = $V_{DD}$ | 0.7$AV_{REF}$ | | $AV_{REF}$ | V |
| | $V_{IH3}$ | P60 to P62 (I/O port mode) | | 0.7$V_{DD}$ | | $V_{DD}$ | V |
| | $V_{IH4}$ | P00 to P02, P10, P11, P14, P16, P17, P30 to P33, P40 to P42, P63, P70 to P75, P120, $\overline{\text{RESET}}$, EXCLK | | 0.8$V_{DD}$ | | $V_{DD}$ | V |
| | $V_{IH5}$ | P60, P61 (SMBus input mode) | 2.4 V ≤ $V_{DD}$ ≤ 3.4 V | 2.1 | | | V |
| Input voltage, low | $V_{IL1}$ | P12, P13, P15, P121 to P125 | | 0 | | 0.3$V_{DD}$ | V |
| | $V_{IL2}$ | P20 to P27 | $AV_{REF}$ = $V_{DD}$ | 0 | | 0.3$AV_{REF}$ | V |
| | $V_{IL3}$ | P60 to P62 (I/O port mode) | | 0 | | 0.3$V_{DD}$ | V |
| | $V_{IL4}$ | P00 to P02, P10, P11, P14, P16, P17, P30 to P33, P40 to P42, P63, P70 to P75, P120, $\overline{\text{RESET}}$, EXCLK | | 0 | | 0.2$V_{DD}$ | V |
| | $V_{IL5}$ | P60, P61 (SMBus input mode) | 2.4 V ≤ $V_{DD}$ ≤ 3.4 V | 0 | | 0.8 | V |
| Output voltage, high | $V_{OH1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OH1}$ = –3.0 mA | $V_{DD}$ – 0.7 | | | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OH1}$ = –2.5 mA | $V_{DD}$ – 0.5 | | | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OH1}$ = –1.0 mA | $V_{DD}$ – 0.5 | | | V |
| | $V_{OH2}$ | P20 to P27 | $AV_{REF}$ = $V_{DD}$, $I_{OH2}$ = –100 μA | $V_{DD}$ – 0.5 | | | V |

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**DC Characteristics (3/8) (78K0/KY2-L, 78K0/KA2-L (20 pins), 78K0/KB2-L, 78K0/KC2-L)**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|---|
| Output voltage, low | $V_{OL1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P70 to P75, P120 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OL1}$ = 8.5 mA | | | | 0.7 | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OL1}$ = 5.0 mA | | | | 0.7 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 2.0 mA | | | | 0.5 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 0.5 mA | | | | 0.4 | V |
| | $V_{OL2}$ | P20 to P27 | $AV_{REF}$ = $V_{DD}$, $I_{OL2}$ = 0.4 mA | | | | 0.4 | V |
| | $V_{OL3}$ | P60 to P63 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OL1}$ = 15.0 mA | | | | 2.0 | V |
| | | | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OL1}$ = 5.0 mA | | | | 0.4 | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OL1}$ = 5.0 mA | | | | 0.6 | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OL1}$ = 3.0 mA | | | | 0.4 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 2.0 mA | | | | 0.4 | V |
| Input leakage current, high | $I_{LIH1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120, P125/$\overline{\text{RESET}}$ | $V_I$ = $V_{DD}$ | | | | 1 | μA |
| | $I_{LIH2}$ | P20 to P27 | $V_I$ = $AV_{REF}$ = $V_{DD}$ | | | | 1 | μA |
| | $I_{LIH3}$ | P121 to 124 | $V_I$ = $V_{DD}$ | I/O port mode | | | 1 | μA |
| | | X1, X2 | | OSC mode | | | 20 | μA |
| | | XT1, XT2 | | | | | 10 | μA |
| Input leakage current, low | $I_{LIL1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120, P125/$\overline{\text{RESET}}$ | $V_I$ = $V_{SS}$ | | | | –1 | μA |
| | $I_{LIL2}$ | P20 to P27 | $V_I$ = $V_{SS}$, $AV_{REF}$ = $V_{DD}$ | | | | –1 | μA |
| | $I_{LIL3}$ | P121 to 124 | $V_I$ = $V_{SS}$ | I/O port mode | | | –1 | μA |
| | | X1, X2 | | OSC mode | | | –20 | μA |
| | | XT1, XT2 | | | | | –10 | μA |
| Pull-up resistor | $R_{PLU1}$ | P00 to P02, P10 to P17, P30 to P33, P40 to P42, P60 to P63, P70 to P75, P120 | $V_I$ = $V_{SS}$ | | 10 | 20 | 100 | kΩ |
| | $R_{PLU2}$ | P125/$\overline{\text{RESET}}$ | | | 75 | 150 | 300 | kΩ |

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

<R> **DC Characteristics (4/8) (78K0/KA2-L (25 pins, 32 pins))**

**($T_A$ = −40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Output current, high[Note 1] | $I_{OH1}$ | Per pin for P00 to P02, P31 to P37, P60, P61 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | −3.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | −2.5 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | −1.0 | mA |
| | | Total of P02, P60, P61 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | −9.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | −7.5 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | −3.0 | mA |
| | | Total of P00, P01, P31 to P37 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | −24.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | −19.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | −8.0 | mA |
| | | Total of P00 to P02, P31 to P37, P60, P61[Note 3] | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | −50.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | −29.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | −15.0 | mA |
| | $I_{OH2}$ | Per pin for P20 to P27, P70 to P72 | $AV_{REF}$ = $V_{DD}$ | | | −0.1 | mA |
| Output current, low[Note 2] | $I_{OL1}$ | Per pin for P00 to P02, P31 to P37 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 8.5 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 5.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 2.0 | mA |
| | | Total of P60, P61 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 15.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 5.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 2.0 | mA |
| | | Total of P02, P60, P61 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 38.5 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 15.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 6.0 | mA |
| | | Total of P00, P01, P31 to P37 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 45.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 35.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 16.0 | mA |
| | | Per pin for P00 to P02, P31 to P37, P60, P61[Note 3] | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 65.0 | mA |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 50.0 | mA |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 29.0 | mA |
| | $I_{OL2}$ | Per pin for P20 to P27, P70 to P72 | $AV_{REF}$ = $V_{DD}$ | | | 0.4 | mA |

**Notes 1.** Value of current at which the device operation is guaranteed even if the current flows from $V_{DD}$ to an output pin.

**2.** Value of current at which the device operation is guaranteed even if the current flows from an output pin to GND.

**3.** Specification under conditions where the duty factor is 70% (time for which current is output is 0.7 × t and time for which current is not output is 0.3 × t, where t is a specific time). The total output current of the pins at a duty factor of other than 70% can be calculated by the following expression.

- Where the duty factor of $I_{OH}$ is n%: Total output current of pins = ($I_{OH}$ × 0.7)/(n × 0.01)

<Example> Where the duty factor is 50%, $I_{OH}$ = −20.0 mA

Total output current of pins = (−20.0 × 0.7)/(50 × 0.01) = −28.0 mA

However, the current that is allowed to flow into one pin does not vary depending on the duty factor. A current higher than the absolute maximum rating must not flow into one pin.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

<R>**DC Characteristics (5/8) (78K0/KA2-L (25 pins, 32 pins))**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Input voltage, high | $V_{IH1}$ | P122, P37 | | 0.7$V_{DD}$ | | $V_{DD}$ | V |
| | $V_{IH2}$ | P20 to P27, P70 to P72 | $AV_{REF}$ = $V_{DD}$ | 0.7$AV_{REF}$ | | $AV_{REF}$ | V |
| | $V_{IH3}$ | P60, P61, P121, P125 (I/O port mode) | | 0.7$V_{DD}$ | | $V_{DD}$ | V |
| | $V_{IH4}$ | P00 to P02, P31 to P36, $\overline{\text{RESET}}$, EXCLK | | 0.8$V_{DD}$ | | $V_{DD}$ | V |
| | $V_{IH5}$ | P60, P61 (SMBus input mode) | 2.4 V ≤ $V_{DD}$ ≤ 3.4 V | 2.1 | | | V |
| Input voltage, low | $V_{IL1}$ | P122, P37 | | 0 | | 0.3$V_{DD}$ | V |
| | $V_{IL2}$ | P20 to P27, P70 to P72 | $AV_{REF}$ = $V_{DD}$ | 0 | | 0.3$AV_{REF}$ | V |
| | $V_{IL3}$ | P60, P61, P121, P125 (I/O port mode) | | 0 | | 0.3$V_{DD}$ | V |
| | $V_{IL4}$ | P00 to P02, P31 to P36, $\overline{\text{RESET}}$, EXCLK | | 0 | | 0.2$V_{DD}$ | V |
| | $V_{IL5}$ | P60, P61 (SMBus input mode) | 2.4 V ≤ $V_{DD}$ ≤ 3.4 V | 0 | | 0.8 | V |
| Output voltage, high | $V_{OH1}$ | P00 to P02, P31 to P37, P60, P61 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OH1}$ = –3.0 mA | $V_{DD}$ – 0.7 | | | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OH1}$ = –2.5 mA | $V_{DD}$ – 0.5 | | | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OH1}$ = –1.0 mA | $V_{DD}$ – 0.5 | | | V |
| | $V_{OH2}$ | P20 to P27, P70 to P72 | $AV_{REF}$ = $V_{DD}$, $I_{OH2}$ = –100 μA | $V_{DD}$ – 0.5 | | | V |

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

<R>**DC Characteristics (6/8) (78K0/KA2-L (25 pins, 32 pins))**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Output voltage, low | $V_{OL1}$ | P00 to P02, P31 to P37 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OL1}$ = 8.5 mA | | | 0.7 | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OL1}$ = 5.0 mA | | | 0.7 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 2.0 mA | | | 0.5 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 1.0 mA | | | 0.5 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 0.5 mA | | | 0.4 | V |
| | $V_{OL2}$ | P20 to P27, P70 to P72 | $AV_{REF}$ = $V_{DD}$, $I_{OL2}$ = 0.4 mA | | | 0.4 | V |
| | $V_{OL3}$ | P60, P61 | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OL1}$ = 15.0 mA | | | 2.0 | V |
| | | | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V, $I_{OL1}$ = 5.0 mA | | | 0.4 | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OL1}$ = 5.0 mA | | | 0.6 | V |
| | | | 2.7 V ≤ $V_{DD}$ < 4.0 V, $I_{OL1}$ = 3.0 mA | | | 0.4 | V |
| | | | 1.8 V ≤ $V_{DD}$ < 2.7 V, $I_{OL1}$ = 2.0 mA | | | 0.4 | V |
| Input leakage current, high | $I_{LIH1}$ | P00 to P02, P31 to P37, P60, P61, P125/RESET | $V_I$ = $V_{DD}$ | | | 1 | μA |
| | $I_{LIH2}$ | P20 to P27, P70 to P72 | $V_I$ = $AV_{REF}$ = $V_{DD}$ | | | 1 | μA |
| | $I_{LIH3}$ | P121, P122 | $V_I$ = $V_{DD}$ I/O port mode | | | 1 | μA |
| | | X1, X2 | $V_I$ = $V_{DD}$ OSC mode | | | 20 | μA |
| Input leakage current, low | $I_{LIL1}$ | P00 to P02, P31 to P37, P60, P61, P125/RESET | $V_I$ = $V_{SS}$ | | | –1 | μA |
| | $I_{LIL2}$ | P20 to P27, P70 to P72 | $V_I$ = $V_{SS}$, $AV_{REF}$ = $V_{DD}$ | | | –1 | μA |
| | $I_{LIL3}$ | P121, P122 | $V_I$ = $V_{SS}$ I/O port mode | | | –1 | μA |
| | | X1, X2 | $V_I$ = $V_{SS}$ OSC mode | | | –20 | μA |
| Pull-up resistor | $R_{PLU1}$ | P00 to P02, P31 to P37, P60, P61 | $V_I$ = $V_{SS}$ | 10 | 20 | 100 | kΩ |
| | $R_{PLU2}$ | P125/RESET | | 75 | 150 | 300 | kΩ |

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**DC Characteristics (7/8)**

**($T_A$ = −40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|---|
| Supply current[Note 1] | $I_{DD1}$[Note 2] | Operating mode | $f_{XH}$ = 10 MHz, $V_{DD}$ = 5.0 V, RMC = 00H | Square wave input | | 1.6 | 2.8 | mA |
| | | | | Resonator connection | | 2.3 | 3.9 | mA |
| | | | $f_{XH}$ = 10 MHz, $V_{DD}$ = 3.0 V, RMC = 00H | Square wave input | | 1.5 | 2.7 | mA |
| | | | | Resonator connection | | 2.2 | 3.2 | mA |
| | | | $f_{XH}$ = 5 MHz, $V_{DD}$ = 3.0 V, RMC = 00H | Square wave input | | 0.9 | 1.6 | mA |
| | | | | Resonator connection | | 1.3 | 2.0 | mA |
| | | | $f_{XH}$ = 5 MHz, $V_{DD}$ = 2.0 V, RMC = 00H | Square wave input | | 0.7 | 1.4 | mA |
| | | | | Resonator connection | | 1.0 | 1.6 | mA |
| | | | $f_{IH}$ = 4 MHz[Note 4], $V_{DD}$ = 3.0 V, RMC = 56H | | | 0.5 | 1.4 | mA |
| | | | $f_{IH}$ = 8 MHz[Note 4], $V_{DD}$ = 5.0 V, RMC = 00H | | | 1.3 | 2.5 | mA |
| | | | $f_{IH}$ = 4 MHz[Note 4], $f_{CPU}$ = 1 MHz,[Note 5] $V_{DD}$ = 3.0 V, RMC = 56H | | | 0.22 | 0.65 | mA |
| | | | $f_{SUB}$ = 32.768 kHz, $V_{DD}$ = 3.0 V, RMC = 56H[Note 6] | $T_A$ = −40 to +50°C | | 3.0 | 6.3 | μA |
| | | | | $T_A$ = −40 to +70°C | | 3.0 | 7.5 | μA |
| | | | | $T_A$ = −40 to +85°C | | 3.0 | 9.7 | μA |
| | $I_{DD2}$[Note 2] | HALT mode | $f_{XH}$ = 10 MHz, $V_{DD}$ = 5.0 V, RMC = 00H | Square wave input | | 0.4 | 1.3 | mA |
| | | | | Resonator connection | | 1.0 | 2.4 | mA |
| | | | $f_{XH}$ = 5 MHz, $V_{DD}$ = 3.0 V, RMC = 00H | Square wave input | | 0.2 | 0.65 | mA |
| | | | | Resonator connection | | 0.5 | 1.1 | mA |
| | | | $f_{IH}$ = 4 MHz[Note 4], $V_{DD}$ = 3.0 V, RMC = 56H | | | 0.2 | 0.5 | mA |
| | | | $f_{IH}$ = 8 MHz[Note 4], $V_{DD}$ = 5.0 V, RMC = 00H | | | 0.3 | 1.2 | mA |
| | | | $f_{SUB}$ = 32.768 kHz, $V_{DD}$ = 3.0 V, RMC = 56H[Note 6] | $T_A$ = −40 to +50°C | | 0.98 | 3.2 | μA |
| | | | | $T_A$ = −40 to +70°C | | 0.98 | 4.8 | μA |
| | | | | $T_A$ = −40 to +85°C | | 0.98 | 6.7 | μA |
| | $I_{DD3}$[Note 3] | STOP mode | $V_{DD}$ = 3.0 V, RMC = 56H | $T_A$ = −40 to +50°C | | 0.3 | 2.7 | μA |
| | | | | $T_A$ = −40 to +70°C | | 0.3 | 3.7 | μA |
| | | | | $T_A$ = −40 to +85°C | | 0.3 | 5.5 | μA |

**Notes 1.** Total current flowing into the internal power supply ($V_{DD}$, $AV_{REF}$), including the input leakage current flowing when the level of the input pin is fixed to $V_{DD}$ or $V_{SS}$. However, the current flowing into the pull-up resistors, the pull-down resistors and the output current of the port are not included.

**2.** Not including the current flowing into the oscillation circuit other than the circuit which generates the clock supplied to CPU. Not including the current flowing into the LVI circuit, A/D converter, operational amplifier, watchdog timer, real-time counter and 8-bits timer H1 (When using the 30 kHz internal low-speed oscillation clock as the count clock).

**3.** Not including the current flowing into the LVI circuit, watchdog timer, real-time counter and 8-bits timer H1 (When using the 30 kHz internal low-speed oscillation clock as the count clock).

**4.** The internal high-speed oscillation clock frequency is set by option byte (R4M8MSEL = 0: 8 MHz, R4M8MSEL = 1: 4 MHz).

**5.** This is the value when PCC2 = 0, PCC1 = 1, PCC0 = 0.

**6.** This is the value when a resonator is connected.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**DC Characteristics (8/8)**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $AV_{REF}$ ≤ $V_{DD}$, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|---|
| Real-time counter operating current[Note 1] | $I_{RTC}$ | $V_{DD}$ = 3.0 V | | | | 0.15 | 1 | μA |
| Watchdog timer operating current[Note 2] | $I_{WDT}$ | $V_{DD}$ = 3.0 V | In 30 kHz internal low-speed oscillation clock operation | | | 0.28 | 0.35 | μA |
| TMH1 operating current[Note 3] | $I_{TMH}$ | $V_{DD}$ = 3.0 V | When using the 30 kHz internal low-speed oscillation clock as the count clock | | | 0.35 | 1.5 | μA |
| LVI operating current[Note 4] | $I_{LVI}$ | | | | | 9 | 18 | μA |
| A/D converter operating current[Note 5] | $I_{ADC}$ | During conversion at maximum speed | High-speed mode 1 | $AV_{REF}$ = $V_{DD}$ = 5.0 V | | 1.72 | 3.2 | mA |
| | | | High-speed mode 2 | $AV_{REF}$ = $V_{DD}$ = 3.0 V | | 0.72 | 1.6 | mA |
| | | | Normal mode | $AV_{REF}$ = $V_{DD}$ = 5.0 V | | 0.86 | 1.9 | mA |
| | | | Low-voltage mode | $AV_{REF}$ = $V_{DD}$ = 3.0 V | | 0.37 | 0.8 | mA |
| Operational amplifier operating current[Note 6] | $I_{AMP}$ | PGA operating | | | | | 1.2 | mA |
| | | Operational amplifier 0 operating | | $AV_{REF}$ = $V_{DD}$ = 5.0 V | | 263 | 380 | μA |
| | | | | $AV_{REF}$ = $V_{DD}$ = 3.0 V | | 232 | 321 | μA |
| | | Operational amplifier 1 operating | | $V_{DD}$ = 5.0 V | | 263 | 380 | μA |
| | | | | $V_{DD}$ = 3.0 V | | 232 | 321 | μA |
| Reset current | $I_{DDrst}$ | After reset ($\overline{\text{RESET}}$ pull-up resistor current + leakage current) | | $AV_{REF}$ = $V_{DD}$ = 5.0 V | | 35 | 100 | μA |

**Notes 1.** Current flowing only to the real-time counter. The current value of the 78K0/Kx2-L microcontrollers is the sum of $I_{DD1}$, $I_{DD2}$ or $I_{DD3}$ and $I_{RTC}$ when the real-time counter operates.

**2.** Current flowing only to the watchdog timer (including the operating current of the 30 kHz internal oscillator). The current value of the 78K0/Kx2-L microcontrollers is the sum of $I_{DD1}$, $I_{DD2}$ or $I_{DD3}$ and $I_{WDT}$ when the watchdog timer operates.

**3.** Current flowing only to the 8-bits timer H1. The current value of the 78K0/Kx2-L microcontrollers is the sum of $I_{DD1}$, $I_{DD2}$ or $I_{DD3}$ and $I_{TWH}$ when the 8-bits timer H1 operates (When using the 30 kHz internal low-speed oscillation clock as the count clock).

**4.** Current flowing only to the LVI circuit. The current value of the 78K0/Kx2-L microcontrollers is the sum of $I_{DD1}$, $I_{DD2}$ or $I_{DD3}$ and $I_{LVI}$ when the LVI circuit operates.

**5.** Current flowing only to the A/D converter ($AV_{REF}$). The current value of the 78K0/Kx2-L microcontrollers is the sum of $I_{DD1}$ or $I_{DD2}$ and $I_{ADC}$ when the A/D converter operates in an operation mode or the HALT mode.

**6.** Current flowing only to the operational amplifier ($AV_{REF}$ or $V_{DD}$). The current value of the 78K0/Kx2-L microcontrollers is the sum of $I_{DD1}$ or $I_{DD2}$ and $I_{AMP}$ when the operational amplifier operates in an operation mode or the HALT mode.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

## AC Characteristics

### (1) Basic operation

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

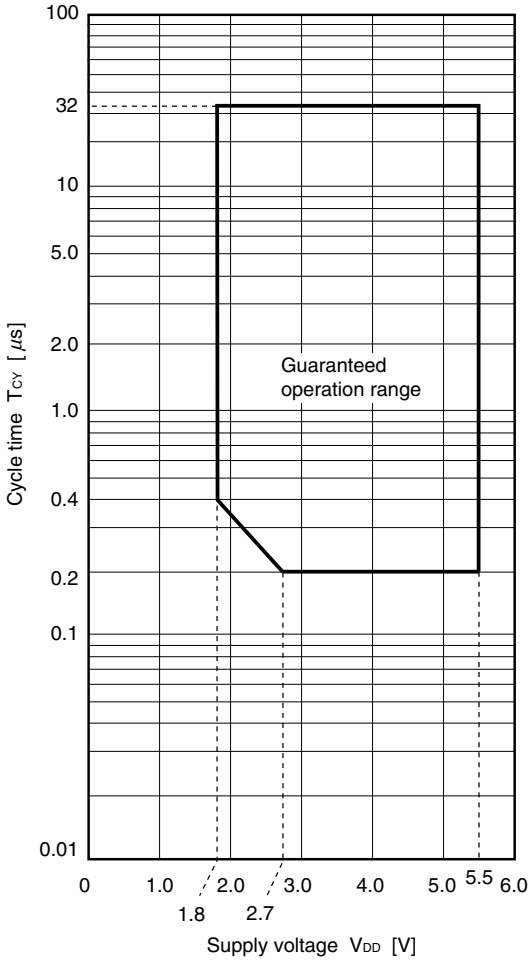
| Items | Symbol | Conditions | | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|---|
| Instruction cycle (minimum instruction execution time) | $T_{CY}$ | Main system clock ($f_{XP}$) operation | In normal power mode (RMC = 00H) | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | 0.2 | | 32 | μs |
| | | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | 0.4 Note 1 | | 32 | μs |
| | | | In low power consumption mode (RMC = 56H) | | 0.4 Note 1 | | 32 | μs |
| | | Subsystem clock ($f_{SUB}$) operation Note 1 | | | 114 | 122 | 125 | μs |
| | | Self programming mode operation | In normal power mode (RMC = 00H) | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | 0.2 | | 32 | μs |
| | | | In low power consumption mode (RMC = 56H) | | 0.4 Note 1 | | 32 | μs |
| Peripheral hardware clock frequency | $f_{PRS}$ | $f_{PRS}$ = $f_{XP}$ | | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | | | 10 | MHz |
| | | | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 5 | MHz |
| | | $f_{PRS}$ = $f_{IH}$ | | R4M8MSEL = 0 | 7.6 | | 8.4 | MHz |
| | | | | R4M8MSEL = 1 | 3.88 | | 4.12 | MHz |
| External main system clock frequency | $f_{EXCLK}$ | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | | | 1.0 | | 10.0 | MHz |
| | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 1.0 | | 5.0 | MHz |
| External main system clock input high-level width, low-level width | $t_{EXCLKH}$, $t_{EXCLKL}$ | | | | (1/$f_{EXCLK}$ ×1/2) –1 | | | ns |
| External subsystem clock frequency | $f_{EXCLKS}$ | | | | 32 | 32.768 | 35 | kHz |
| External subsystem clock input high-level width, low-level width | $t_{EXCLKSH}$, $t_{EXCLKSL}$ | | | | (1/$f_{EXCLKS}$ ×1/2) –5 | | | ns |
| TI000, TI010 input high-level width, low-level width | $t_{TIH0}$, $t_{TIL0}$ | 4.0 V ≤ $V_{DD}$ ≤ 5.5 V | | | 2/$f_{sam}$+0.1 Note 2 | | | μs |
| | | 2.7 V ≤ $V_{DD}$ < 4.0 V | | | 2/$f_{sam}$+0.2 Note 2 | | | μs |
| | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 2/$f_{sam}$+0.5 Note 2 | | | μs |
| TI50, TI51 input frequency | $f_{TI5}$ | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | | | | | 10.0 | MHz |
| | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | | | 5.0 | MHz |
| TI50, TI51 input high-level width, low-level width | $t_{TIH5}$ | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | | | 50 | | | ns |
| | | 1.8 V ≤ $V_{DD}$ < 2.7 V | | | 100 | | | ns |
| Interrupt input high-level width, low-level width | $t_{INTH}$, $t_{INTL}$ | | | | 1 | | | μs |
| Key interrupt input low-level width | $t_{KR}$ | | | | 250 | | | ns |
| $\overline{\text{RESET}}$ low-level width | $t_{RSL}$ | | | | 10 | | | μs |

**Notes 1.** 0.38 μs when operating with the internal high-speed oscillation clock.

**2.** Selection of $f_{sam}$ = $f_{PRS}$, $f_{PRS}$/4, $f_{PRS}$/256 is possible using bits 0 and 1 (PRM000, PRM001) of prescaler mode register 00 (PRM00). Note that when selecting the TI000 valid edge as the count clock, $f_{sam}$ = $f_{PRS}$.

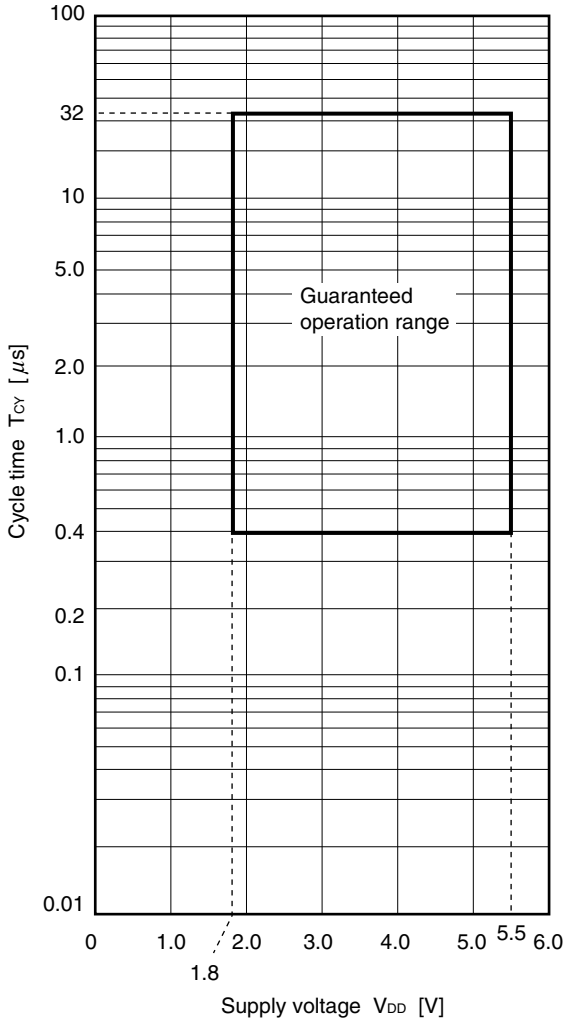
**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**$T_{CY}$ vs. $V_{DD}$ (Main System Clock Operation, RMC = 00H (Normal Power Mode))**

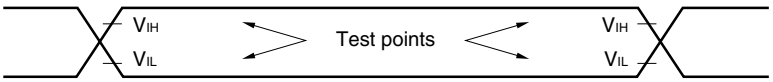


**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**$T_{CY}$ vs. $V_{DD}$ (Main System Clock Operation, RMC = 56H (Low Power Consumption Mode))**

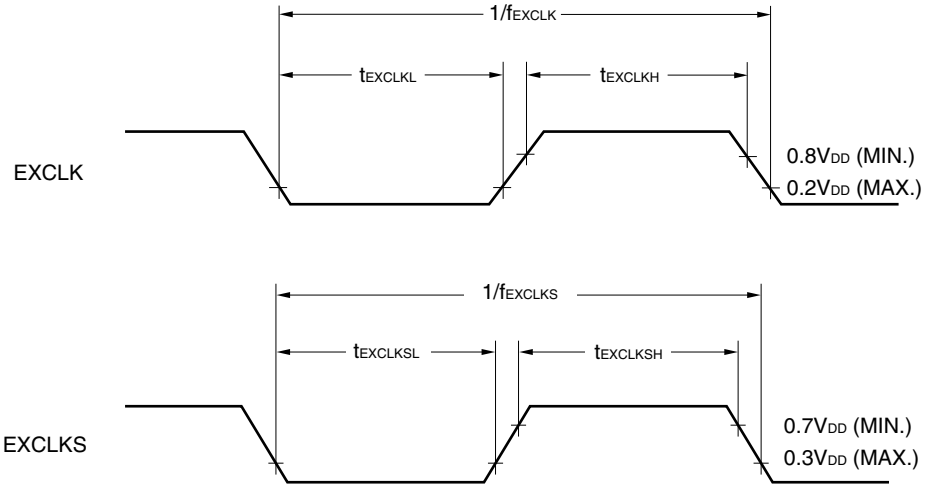



**AC Timing Test Points**

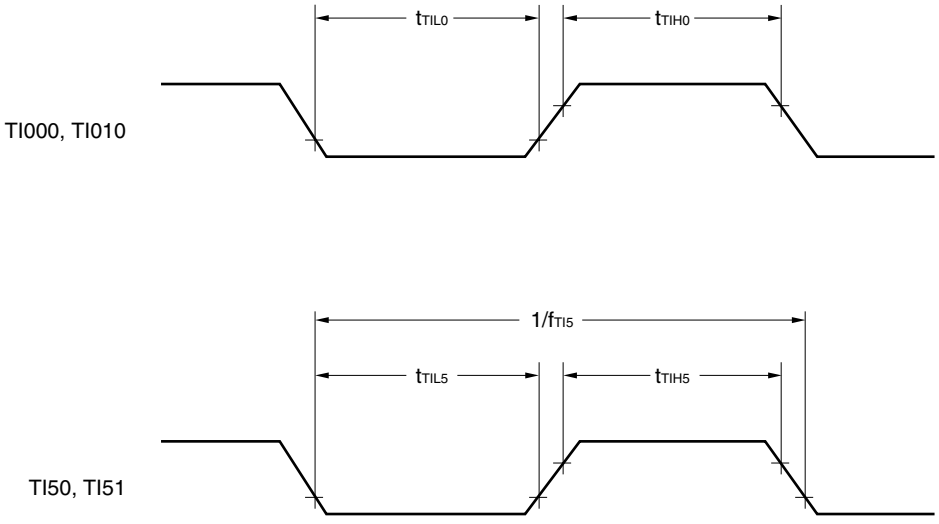


**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

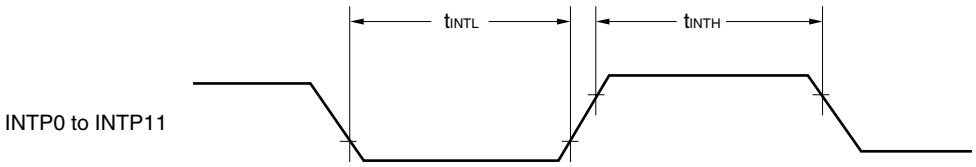
**External Main System Clock Timing, External Subsystem Clock Timing**




**TI Timing**

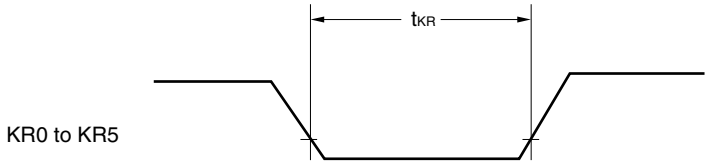



**Interrupt Request Input Timing**

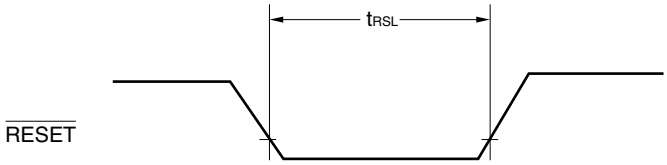


**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**Key Interrupt Input Timing**




**$\overline{\text{RESET}}$ Input Timing**



**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**(2) Serial interface**

**($T_A$ = –40 to +85°C, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

**(a) UART6 (dedicated baud rate generator output)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Transfer rate | | | | | 625 | kbps |

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**(b) IICA**

| Parameter | Symbol | Conditions | Standard Mode | | High-Speed Mode | | Unit |
|---|---|---|---|---|---|---|---|
| | | | MIN. | MAX. | MIN. | MAX. | |
| SCLA0 clock frequency | $f_{SCL}$ | Fast mode: $f_{PRS} \geq 3.5$ MHz, Normal mode: $f_{PRS} \geq 1$ MHz | 0 | 100 | 0 | 400 | kHz |
| Setup time of start condition and stop condition | $t_{SU:STA}$ | | 4.7 | – | 0.6 | – | μs |
| Hold time[Note 1] | $t_{HD:STA}$ | | 4.0 | – | 0.6 | – | μs |
| Hold time when SCLA0 = "L" | $t_{LOW}$ | | 4.7 | – | 1.3 | – | μs |
| Hold time when SCLA0 = "H" | $t_{HIGH}$ | | 4.0 | – | 0.6 | – | μs |
| Data setup time (reception) | $t_{SU:DAT}$ | | 250 | – | 100 | – | ns |
| Data hold time (transmission)[Notes 2,3] | $t_{HD:DAT}$ | | 0 | 3.45 | 0 | 0.9 | μs |
| Setup time of stop condition | $t_{SU:STO}$ | | 4.0 | – | 0.6 | – | μs |
| Bus free time between stop condition and start condition | $t_{BUF}$ | | 4.7 | – | 1.3 | – | μs |
| Rise time of SDAA0 and SCLA0 signals | $t_R$ | | | 1000 | 2.0+ 0.1$C_b$ | 300 | ns |
| Fall time of SDAA0 and SCLA0 signals | $t_F$ | | | 300 | 2.0+ 0.1$C_b$ | 300 | ns |
| Total load capacitance value of each communication line (SCLA0, SDAA0) | $C_b$ | | | 400 | | 400 | pF |

**Notes 1.** The first clock pulse is generated after this period when the start/restart condition is detected.

**2.** The maximum value (MAX.) of $t_{HD:DAT}$ is during normal transfer and a wait state is inserted in the $\overline{ACK}$ (acknowledge) timing.

**3.** The data hold time differs depending on the setting of the IICA low-level width setting register (IICWL).

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**(c) CSI1n (master mode, $\overline{SCK1n}$... internal clock output)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| $\overline{SCK1n}$ cycle time | $t_{KCY1}$ | 2.7 V ≤ $V_{DD}$ ≤ 5.5 V | 200 | | | ns |
| | | 1.8 V ≤ $V_{DD}$ < 2.7 V | 400 | | | ns |
| $\overline{SCK1n}$ high-/low-level width | $t_{KH1}$, $t_{KL1}$ | | $t_{KCY1}/2 - 10$ Note 1 | | | ns |
| SI1n setup time (to $\overline{SCK1n}$↑) | $t_{SIK1}$ | | 30 | | | ns |
| SI1n hold time (from $\overline{SCK1n}$↑) | $t_{KSI1}$ | | 30 | | | ns |
| Delay time from $\overline{SCK1n}$↓ →to SO1n output | $t_{KSO1}$ | C = 50 pF Note 2 | | | 40 | ns |

**Notes 1.** This value is when high-speed system clock ($f_{XH}$) is used.

**2.** C is the load capacitance of the $\overline{SCK1n}$ and SO1n output lines.

**(d) CSI1n (slave mode, $\overline{SCK1n}$... external clock input)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| $\overline{SCK1n}$ cycle time | $t_{KCY2}$ | | 400 | | | ns |
| $\overline{SCK1n}$ high-/low-level width | $t_{KH2}$, $t_{KL2}$ | | $t_{KCY2}/2$ | | | ns |
| SI1n setup time (to $\overline{SCK1n}$↑) | $t_{SIK2}$ | | 80 | | | ns |
| SI1n hold time (from $\overline{SCK1n}$↑) | $t_{KSI2}$ | | 50 | | | ns |
| Delay time from $\overline{SCK1n}$↓ →to SO1n output | $t_{KSO2}$ | C = 50 pF Note | | | 120 | ns |

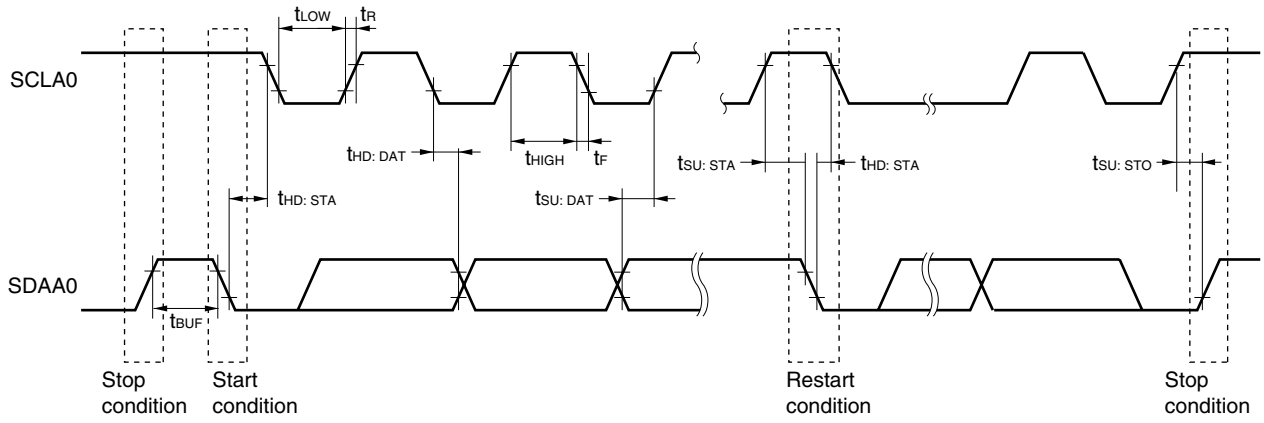
**Note** C is the load capacitance of the SO1n output line.

**Remark** n = 0, 1

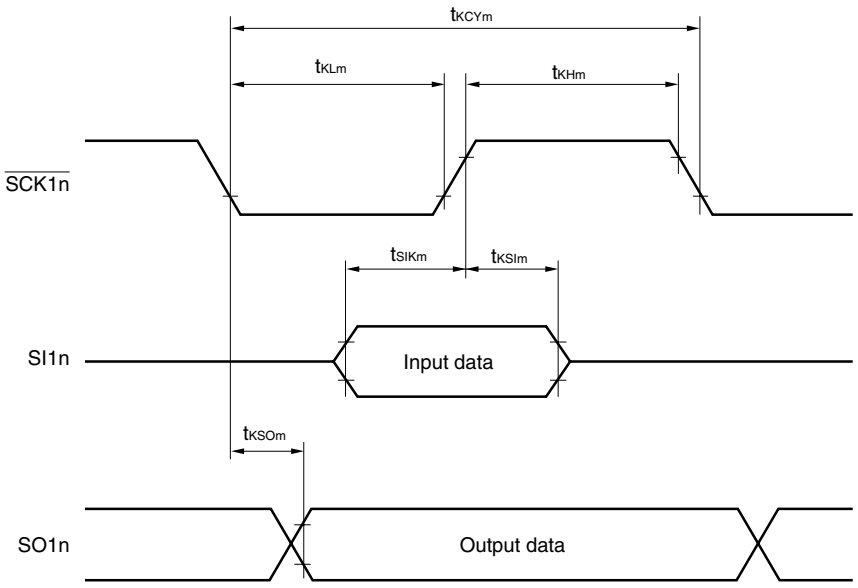
**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**Serial Transfer Timing**

**IICA:**




**CSI1n:**




**Remark** m = 1, 2
n = 0, 1

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

## Analog Characteristics

### (1) A/D Converter (1/2)

**($T_A$ = –40 to +85°C, 1.8 V ≤ $AV_{REF}$ ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Resolution | RES | | | | | 10 | bit |
| Overall error[Notes 1, 2] | AINL | High-speed mode 1 | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.4 | %FSR |
| | | High-speed mode 2 | 2.7 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.6 | %FSR |
| | | Normal mode | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.4 | %FSR |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | | | ±0.6 | %FSR |
| | | Low-voltage mode | 1.8 V ≤ $AV_{REF}$ < 4.0 V | | | ±1.2 | %FSR |
| Conversion time | tCONV | High-speed mode 1 | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | 3.3 | | 66 | μs |
| | | High-speed mode 2 | 2.7 V ≤ $AV_{REF}$ ≤ 5.5 V | 4.4 | | 66 | μs |
| | | Normal mode | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | 6.6 | | 66 | μs |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | 13.2 | | 66 | μs |
| | | Low-voltage mode | 1.8 V ≤ $AV_{REF}$ < 4.0 V | 44 | | 66 | μs |
| Zero-scale error[Notes 1, 2] | EZS | High-speed mode 1 | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.4 | %FSR |
| | | High-speed mode 2 | 2.7 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.6 | %FSR |
| | | Normal mode | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.4 | %FSR |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | | | ±0.6 | %FSR |
| | | Low-voltage mode | 1.8 V ≤ $AV_{REF}$ < 4.0 V | | | ±0.6 | %FSR |
| Full-scale error[Notes 1, 2] | EFS | High-speed mode 1 | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.4 | %FSR |
| | | High-speed mode 2 | 2.7 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.6 | %FSR |
| | | Normal mode | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±0.4 | %FSR |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | | | ±0.6 | %FSR |
| | | Low-voltage mode | 1.8 V ≤ $AV_{REF}$ < 4.0 V | | | ±0.6 | %FSR |
| Integral non-linearity error[Note 1] | ILE | High-speed mode 1 | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±2.5 | LSB |
| | | High-speed mode 2 | 2.7 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±4.5 | LSB |
| | | Normal mode | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±2.5 | LSB |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | | | ±4.5 | LSB |
| | | Low-voltage mode | 1.8 V ≤ $AV_{REF}$ < 4.0 V | | | ±6.5 | LSB |
| Differential non-linearity error[Note 1] | DLE | High-speed mode 1 | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±1.5 | LSB |
| | | High-speed mode 2 | 2.7 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±2.0 | LSB |
| | | Normal mode | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | | | ±1.5 | LSB |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | | | ±2.0 | LSB |
| | | Low-voltage mode | 1.8 V ≤ $AV_{REF}$ < 4.0 V | | | ±2.0 | LSB |
| Analog input voltage | VAIN | 1.8 V ≤ $AV_{REF}$ ≤ 5.5 V | | $AV_{SS}$ | | $AV_{REF}$ | V |

**Notes 1.** Excludes quantization error (±1/2 LSB).

**2.** This value is indicated as a ratio (%FSR) to the full-scale value.

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

### (2) PGA

**($T_A$ = –40 to +85°C, 2.7 V ≤ $AV_{REF}$ ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|---|
| Input offset voltage | $V_{IOPGA}$ | | | | | ±5 | ±10 | mV |
| Input voltage range | $V_{IPGA}$ | | | | 0.1$AV_{REF}$/gain | | 0.9$AV_{REF}$/gain | V |
| Maximum output voltage | $V_{OPGA}$ | | | | 0.1$AV_{REF}$ | | 0.9$AV_{REF}$ | V |
| Gain error | | 4, 8 times | | | | | ±1 | % |
| | | 16 times | | | | | ±1.5 | % |
| | | 32 times | | | | | ±2 | % |
| Slew rate | $SR_{RPGA}$ | Rising edge | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | 4, 8 times | 4 | | | V/μs |
| | | | | 16, 32 times | 1.5 | | | V/μs |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | 4, 8 times | 1.8 | | | V/μs |
| | | | | 16, 32 times | 0.5 | | | V/μs |
| | $SR_{FPGA}$ | Falling edge | 4.0 V ≤ $AV_{REF}$ ≤ 5.5 V | 4, 8 times | 3.2 | | | V/μs |
| | | | | 16, 32 times | 1.5 | | | V/μs |
| | | | 2.7 V ≤ $AV_{REF}$ < 4.0 V | 4, 8 times | 1.2 | | | V/μs |
| | | | | 16, 32 times | 0.5 | | | V/μs |
| Operation stabilization wait time[Note] | $t_{PGA}$ | 4, 8 times | | | | | 5 | μs |
| | | 16, 32 times | | | | | 10 | μs |

**Note** Time required until a state is entered where the DC and AC specifications of the PGA are satisfied after the PGA operation has been enabled (PGAEN = 1).

### (3) Operational amplifier 0

**($T_A$ = –40 to +85°C, 2.2 V ≤ $V_{DD}$ ≤ 5.5 V, 2.2 V ≤ $AV_{REF}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V,**
**Output load: $R_L$ = 47 kΩ, $C_L$ = 50 pF)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Input offset voltage | $V_{IOP0}$ | $V_{BIAS}$ = 1/2 $V_{DD}$, $AV_{REF}$ = 3.0 V | | | ±10 | mV |
| Power supply voltage rejection ratio | $PSRR_{OP0}$ | $AV_{REF}$ = 3.0 V | | 70 | | dB |
| Output voltage, high | $V_{OHOP0}$ | $AV_{REF}$ = 3.0 V/2.2 V, $I_{OH}$ = –500 μA | $AV_{REF}$–0.2 | | | V |
| Output voltage, low | $V_{OLOP0}$ | $AV_{REF}$ = 3.0 V/2.2 V, $I_{OL}$ = 500 μA | | | 0.1 | V |
| Common-mode input voltage | $V_{ICMOP0}$ | $AV_{REF}$ = 3.0 V/2.2 V | 0 | | $AV_{REF}$–0.6 | V |
| Slew rate | $SR_{OP0}$ | $AV_{REF}$ = 3.0 V | | 1.8 | | V/μs |
| | | $AV_{REF}$ = 5.0 V | | 2.0 | | V/μs |
| Input noise spectral density (Inoise) | | $AV_{REF}$ = 3.0 V, $V_{IN}$ = 0.1 V, f = 1 kHz | | 73 | | nV/√Hz |
| | | $AV_{REF}$ = 3.0 V, $V_{IN}$ = $AV_{REF}$/2 V, f = 1 kHz | | 60 | | |
| | | $AV_{REF}$ = 3.0 V, $V_{IN}$ = $AV_{REF}$ –0.6 V, f = 1 kHz | | 55 | | |
| Phase margin | | $AV_{REF}$ = 3.0 V | | 40 | | deg |
| Large-amplitude voltage gain | $AV_{OP0}$ | $AV_{REF}$ = 3.0 V | | 100 | | dB |
| Gain-bandwidth product | $GBW_{OP0}$ | $AV_{REF}$ = 5.0 V/3.0 V/2.2 V | | 3.0 | | MHz |
| Operation stabilization wait time[Note] | $t_{OP0}$ | $AV_{REF}$ = 3.0 V | | 10 | | μs |

**Note** Time required until a state is entered where the DC and AC specifications of the operational amplifier 0 are satisfied after the operational amplifier 0 operation has been enabled (OPAMP0E = 1).

**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

### (4) Operational amplifier 1

**($T_A$ = –40 to +85°C, 2.2 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0 V, Output load: $R_L$ = 47 kΩ, $C_L$ = 50 pF)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Input offset voltage | $V_{IOP1}$ | $V_{DD}$ = 3.0 V | | | ±10 | mV |
| Power supply voltage rejection ratio | $PSRR_{OP1}$ | $V_{DD}$ = 3.0 V | | 70 | | dB |
| Output voltage, high | $V_{OHOP1}$ | $V_{DD}$ = 3.0 V/2.2 V, $I_{OH}$ = –500 μA | $V_{DD}$ – 0.2 | | | V |
| Output voltage, low | $V_{OLOP1}$ | $V_{DD}$ = 3.0 V/2.2 V, $I_{OL}$ = 500 μA | | | 0.1 | V |
| Common-mode input voltage | $V_{ICMOP1}$ | $V_{DD}$ = 3.0 V/2.2 V | 0 | | $V_{DD}$ – 0.6 | V |
| Slew rate | $SR_{OP1}$ | $V_{DD}$ = 3.0 V | | 1.8 | | V/μs |
| | | $V_{DD}$ = 5.0 V | | 2.0 | | V/μs |
| Input noise spectral density (Inoise) | | $V_{DD}$ = 3.0 V, $V_{IN}$ = 0.1 V, f = 1 kHz | | 73 | | nV / √Hz |
| | | $V_{DD}$ = 3.0 V, $V_{IN}$ = $V_{DD}$/2 V, f = 1 kHz | | 60 | | |
| | | $V_{DD}$ = 3.0 V, $V_{IN}$ = $V_{DD}$–0.6 V, f = 1 kHz | | 55 | | |
| Phase margin | | $V_{DD}$ = 3.0 V | | 40 | | deg |
| Large-amplitude voltage gain | $AV_{OP1}$ | $V_{DD}$ = 3.0 V | | 100 | | dB |
| Gain-bandwidth product | $GBW_{OP1}$ | $V_{DD}$ = 5.0 V/3.0 V/2.2 V | | 3.0 | | MHz |
| Operation stabilization wait time[Note] | $t_{OP1}$ | $V_{DD}$ = 3.0 V | | 10 | | μs |

**Note** Time required until a state is entered where the DC and AC specifications of the operational amplifier 1 are satisfied after the operational amplifier 1 operation has been enabled (OPAMP1E = 1).

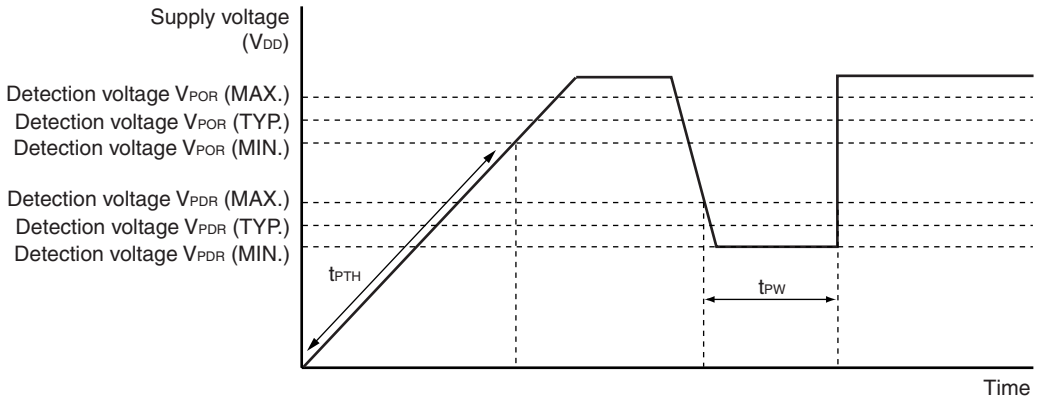
**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

### (5) POC

**($T_A$ = –40 to +85°C, $V_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Detection voltage | $V_{POR}$ | | 1.52 | 1.61 | 1.70 | V |
| | $V_{PDR}$ | | 1.50 | 1.59 | 1.68 | V |
| Power supply voltage rise inclination | $t_{PTH}$ | Change inclination of $V_{DD}$: 0 V → $V_{POR}$ | 0.5 | | | V/ms |
| Minimum pulse width | $t_{PW}$ | When the voltage drops | 200 | | | μs |
| Detection delay time | | | | | 200 | μs |

**POC Circuit Timing**



**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

### (6) Supply Voltage Rise Time

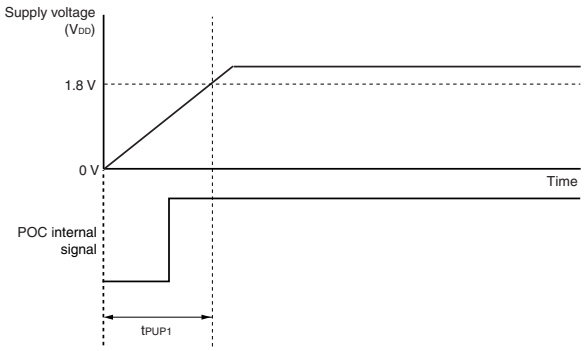
**($T_A$ = –40 to +85°C, $V_{SS}$ = 0 V)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Maximum time to rise to 1.8 V ($V_{DD}$ (MIN.))[Note] ($V_{DD}$: 0 V → 1.8 V) | $t_{PUP1}$ | LVI default start function stopped is set (LVISTART (Option Byte) = 0), when $\overline{\text{RESET}}$ input is not used | | | 3.6 | ms |
| Maximum time to rise to 1.8 V ($V_{DD}$ (MIN.))[Note] (releasing $\overline{\text{RESET}}$ input → $V_{DD}$: 1.8 V) | $t_{PUP2}$ | LVI default start function stopped is set (LVISTART (Option Byte) = 0), when $\overline{\text{RESET}}$ input is used | | | 1.9 | ms |

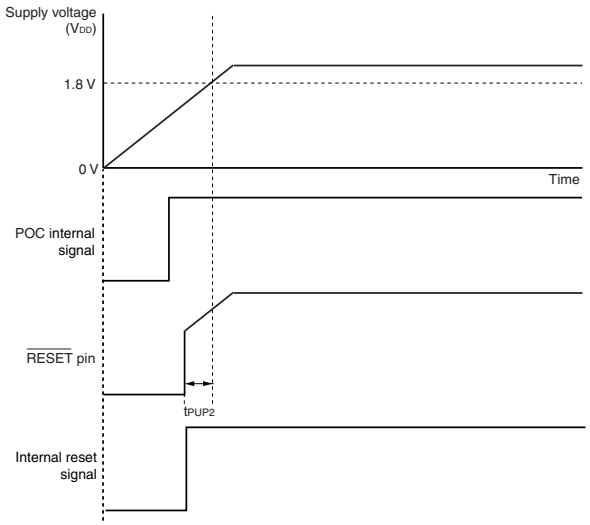
**Note** Make sure to raise the power supply in a shorter time than this.

**Supply Voltage Rise Time Timing**

- When $\overline{\text{RESET}}$ pin input is not used




- When $\overline{\text{RESET}}$ pin input is used (when external reset is released by the $\overline{\text{RESET}}$ pin, after POC has been released)



**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

### (7) LVI

**($T_A$ = –40 to +85°C, $V_{PDR} \le V_{DD} \le 5.5$ V, $V_{SS}$ =0 V)**

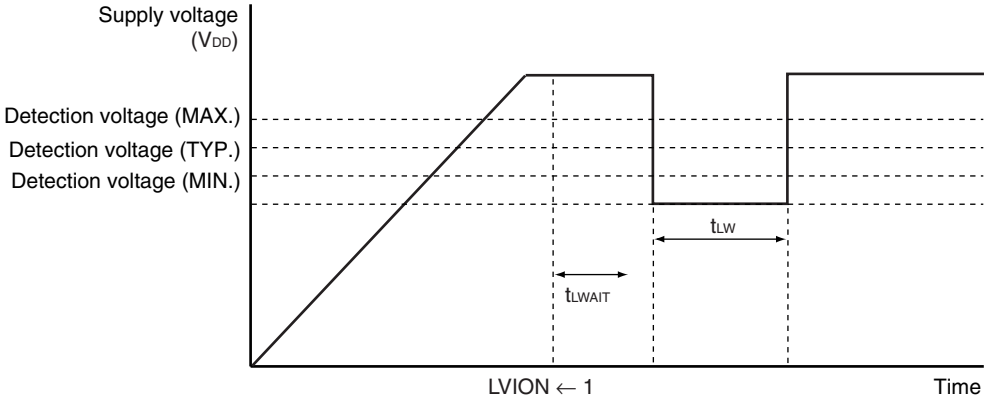
| Parameter | | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|
| Detection voltage | Supply voltage level | $V_{LVI0}$ | | 4.12 | 4.22 | 4.32 | V |
| | | $V_{LVI1}$ | | 3.97 | 4.07 | 4.17 | V |
| | | $V_{LVI2}$ | | 3.82 | 3.92 | 4.02 | V |
| | | $V_{LVI3}$ | | 3.66 | 3.76 | 3.86 | V |
| | | $V_{LVI4}$ | | 3.51 | 3.61 | 3.71 | V |
| | | $V_{LVI5}$ | | 3.35 | 3.45 | 3.55 | V |
| | | $V_{LVI6}$ | | 3.20 | 3.30 | 3.40 | V |
| | | $V_{LVI7}$ | | 3.05 | 3.15 | 3.25 | V |
| | | $V_{LVI8}$ | | 2.89 | 2.99 | 3.09 | V |
| | | $V_{LVI9}$ | | 2.74 | 2.84 | 2.94 | V |
| | | $V_{LVI10}$ | | 2.58 | 2.68 | 2.78 | V |
| | | $V_{LVI11}$ | | 2.43 | 2.53 | 2.63 | V |
| | | $V_{LVI12}$ | | 2.28 | 2.38 | 2.48 | V |
| | | $V_{LVI13}$ | | 2.12 | 2.22 | 2.32 | V |
| | | $V_{LVI14}$ | | 2.00 | 2.07 | 2.14 | V |
| | | $V_{LVI15}$ | | 1.81 | 1.91 | 2.01 | V |
| | External input pin[Note 1] | EXLVI | EXLVI < $V_{DD}$, 1.8 V ≤ $V_{DD}$ ≤ 5.5 V | 1.11 | 1.21 | 1.31 | V |
| | Supply voltage when power supply voltage is turned on | $V_{DDLVI}$ | When LVI default start function enabled is set (LVISTART = 1) | 1.81 | 1.91 | 2.01 | V |
| Minimum pulse width | | $t_{LW}$ | | 200 | | | μs |
| Detection delay time | | | | | | 200 | μs |
| Operation stabilization wait time[Note 2] | | $t_{LWAIT}$ | | | | 10 | μs |

**Notes 1.** The EXLVI/P120/INTP0 pin is used.

**2.** Time required from setting bit 7 (LVION) of the low-voltage detection register (LVIM) to 1 to operation stabilization

**Remark** $V_{LVI(n-1)} > V_{LVIn}$: n = 1 to 15

**LVI Circuit Timing**

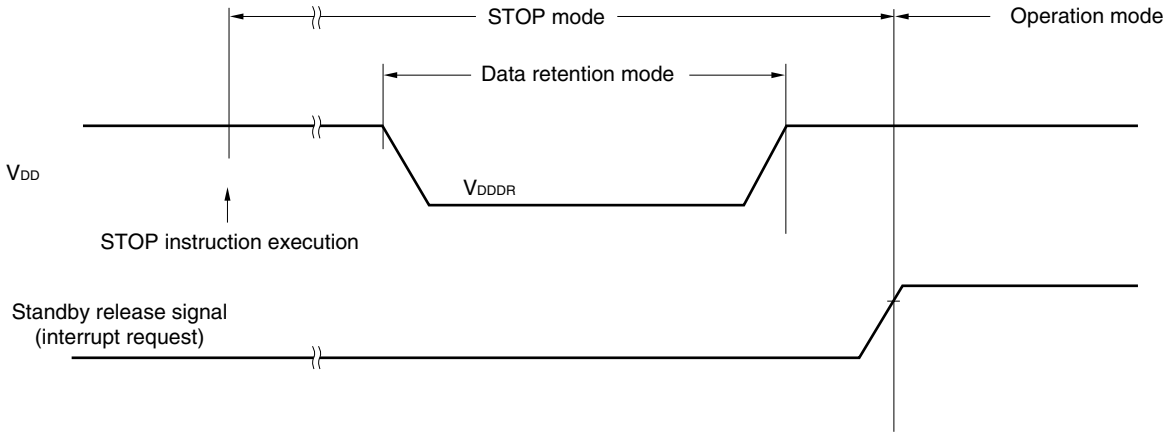


**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

**Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics ($T_A$ = –40 to +85°C)**

| Parameter | Symbol | Conditions | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|
| Data retention supply voltage | $V_{DDDR}$ | | 1.5[Note] | | 5.5 | V |

**Note** The value depends on the POC detection voltage. When the voltage drops, the data is retained until a POC reset is effected, but data is not retained when a POC reset is effected.



**Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.**

## Flash Memory Programming Characteristics

**($T_A$ = –40 to +85°C, 2.0 V ≤ $V_{DD}$ ≤ 5.5 V, $V_{SS}$ = 0 V)**

**• Basic characteristics**

| Parameter | Symbol | Conditions | | | | MIN. | TYP. | MAX. | Unit |
|---|---|---|---|---|---|---|---|---|---|
| $V_{DD}$ supply current | $I_{DD}$ | | | | | | 4.5 | 10.0 | mA |
| Number of rewrites per chip | $C_{erwr}$ | 1 erase + 1 write after erase = 1 rewrite Note 1 | In normal power mode (RMC = 00H) | When a flash memory programmer is used, and the self-programming libraries provided by Renesas Electronics are used | Retention: 15 years | 1000 | | | Times |
| | | | | When the EEPROM emulation libraries (the rewritable ROM size is 4 KB) provided by Renesas Electronics are used | Retention: 5 years | 10000 | | | Times |
| | | | In low power consumption mode (RMC = 56H) | When the self-programming libraries Note 2, and the EEPROM emulation libraries (the rewritable ROM size is 4 KB) provided by Renesas Electronics are used | Retention: 5 years | 1000 | | | Times |
| Operating temperature | | When a flash memory programmer is used: 10 to 40 °C, during self-programming: –40 to +85 °C | | | | | | | |
| Operating voltage range | | In normal power mode (RMC = 00H) | | When a flash memory programmer is used | | 2.5 to 5.5 V@8 MHz (MAX.) | | | |
| | | | | During self-programming | | 2.5 to 5.5 V@10 MHz (MAX.) | | | |
| | | In low power consumption mode (RMC = 56H) | | During self-programming | | 2.0 to 5.5 V@5 MHz (MAX.) | | | |

**Notes 1.** When a product is first written after shipment, “erase → write” and “write only” are both taken as one rewrite.

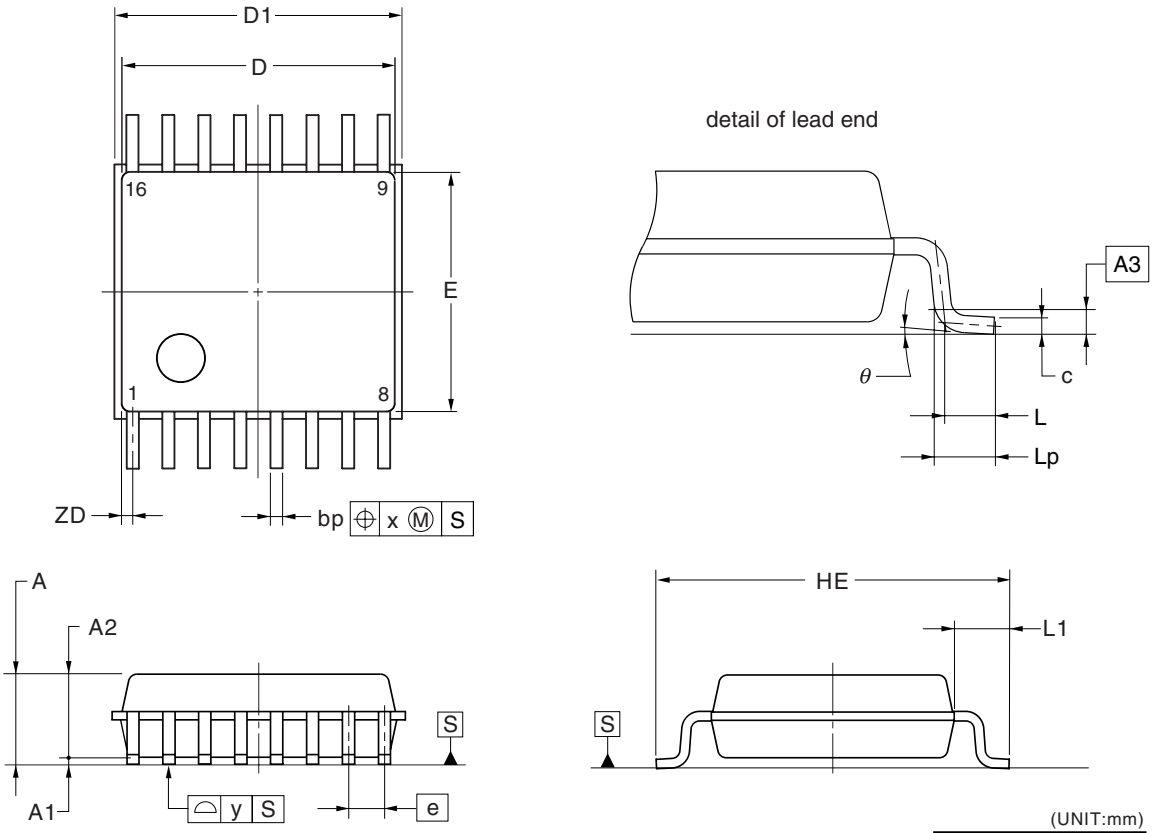
**2.** Only the data area can be rewritten.

# CHAPTER 29 PACKAGE DRAWINGS

## 29.1 78K0/KY2-L

- μPD78F0550MA-FAA-AX, 78F0551MA-FAA-AX, 78F0552MA-FAA-AX, 78F0555MA-FAA-AX, 78F0556MA-FAA-AX, 78F0557MA-FAA-AX

### 16-PIN PLASTIC SSOP (4.4x5.0)




(UNIT:mm)

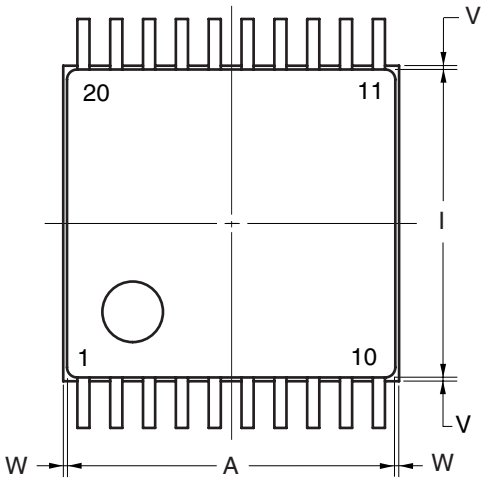
| ITEM | DIMENSIONS |
|---|---|
| D | 5.00±0.15 |
| D1 | 5.20±0.15 |
| E | 4.40±0.20 |
| HE | 6.40±0.20 |
| A | 1.725 MAX. |
| A1 | 0.125±0.05 |
| A2 | 1.50 |
| A3 | 0.25 |
| e | 0.65 |
| bp | $0.22^{+0.08}_{-0.07}$ |
| c | $0.15^{+0.03}_{-0.04}$ |
| L | 0.50 |
| Lp | 0.60±0.10 |
| L1 | 1.00±0.20 |
| x | 0.13 |
| y | 0.10 |
| θ | $3^{\circ}{}^{+5^{\circ}}_{-3^{\circ}}$ |
| ZD | 0.325 |

P16MA-65-FAA

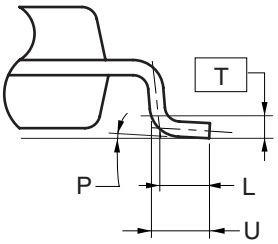
### 29.2 78K0/KA2-L

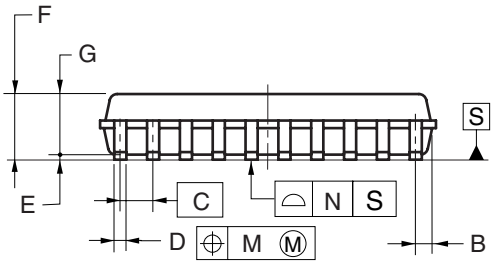
- μPD78F0560MC-CAA-AX, 78F0561MC-CAA-AX, 78F0562MC-CAA-AX, 78F0565MC-CAA-AX, 78F0566MC-CAA-AX, 78F0567MC-CAA-AX

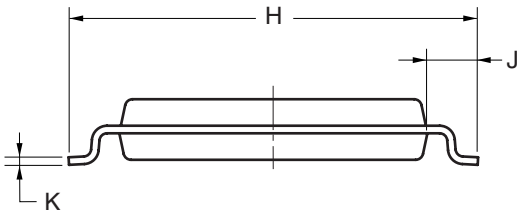
## 20-PIN PLASTIC SSOP (7.62 mm (300))




detail of lead end




(UNIT:mm)

| ITEM | DIMENSIONS |
|---|---|
| A | 6.50±0.10 |
| B | 0.325 |
| C | 0.65 (T.P.) |
| D | $0.22^{+0.10}_{-0.05}$ |
| E | 0.10±0.05 |
| F | 1.30±0.10 |
| G | 1.20 |
| H | 8.10±0.20 |
| I | 6.10±0.10 |
| J | 1.00±0.20 |
| K | $0.15^{+0.05}_{-0.01}$ |
| L | 0.50 |
| M | 0.13 |
| N | 0.10 |
| P | $3^{\circ}{}^{+5^{\circ}}_{-3^{\circ}}$ |
| T | 0.25(T.P) |
| U | 0.60±0.15 |
| V | 0.25 MAX. |
| W | 0.15 MAX. |

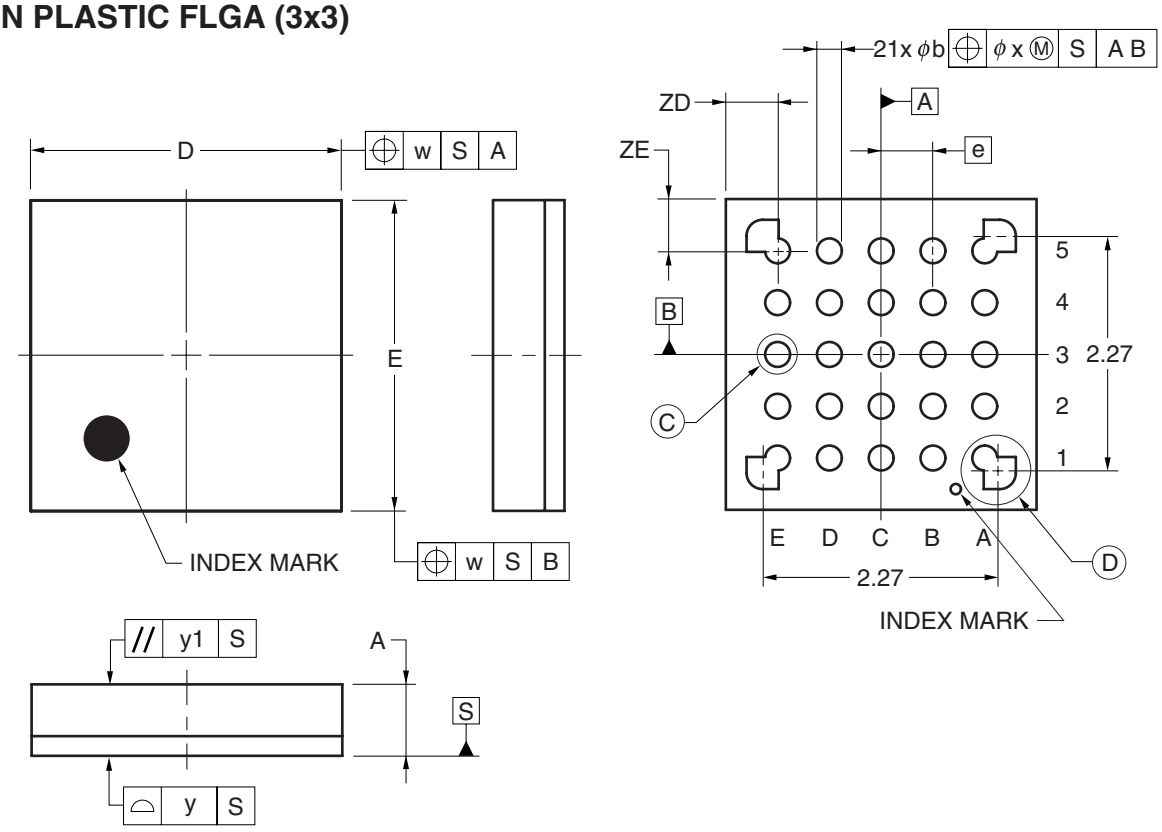
P20MC-65-CAA

**NOTE**

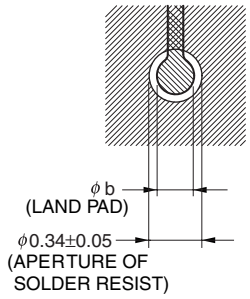
Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

<R> • μPD78F0560FC-2N2-A, 78F0561FC-2N2-A, 78F0562FC-2N2-A, 78F0565FC-2N2-A, 78F0566FC-2N2-A, 78F0567FC-2N2-A

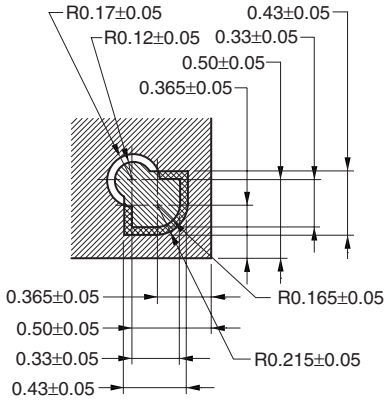
## 25-PIN PLASTIC FLGA (3x3)




DETAIL OF Ⓒ PART




DETAIL OF Ⓓ PART



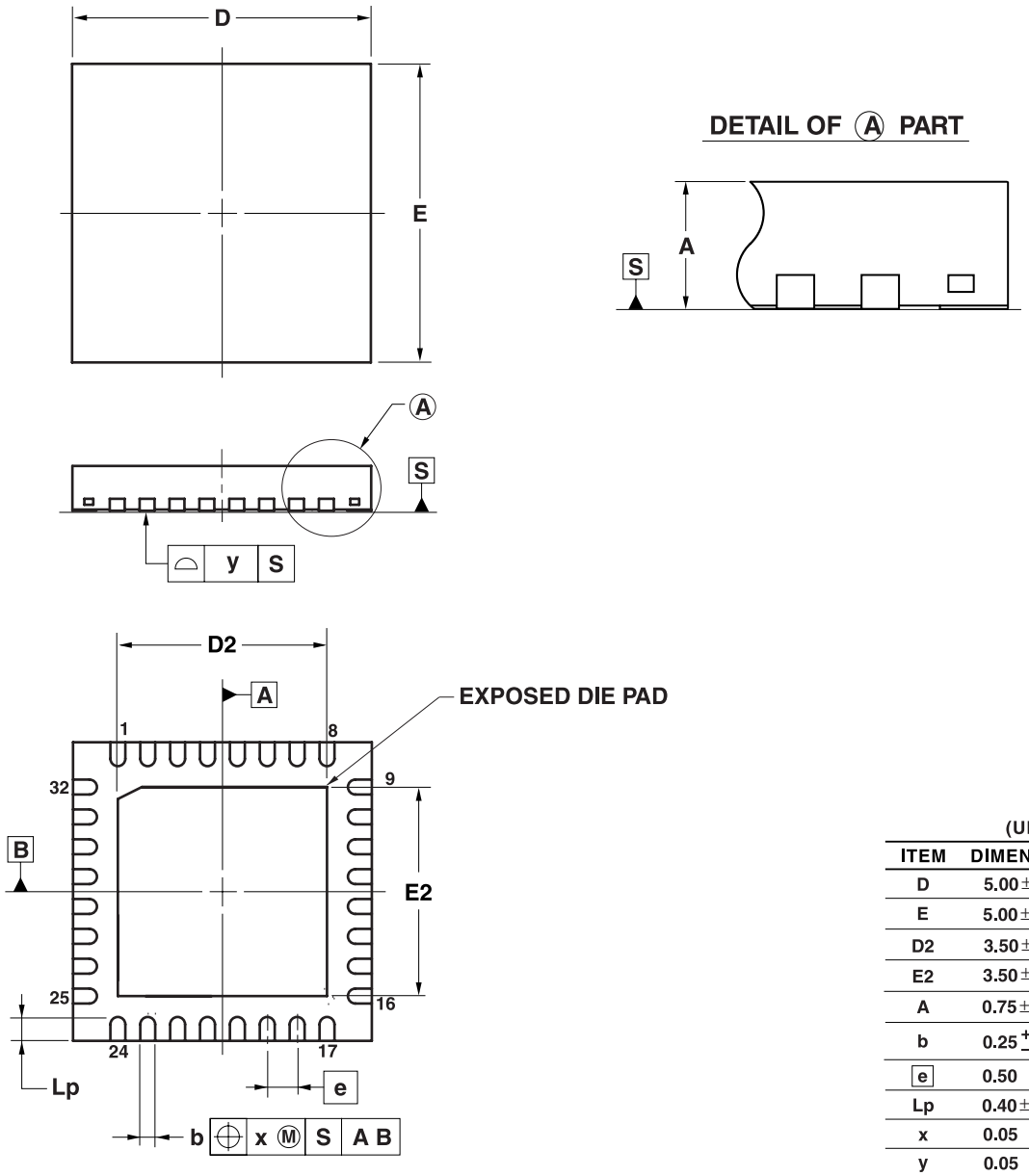

(UNIT:mm)

| ITEM | DIMENSIONS |
|---|---|
| D | 3.00±0.10 |
| E | 3.00±0.10 |
| w | 0.20 |
| e | 0.50 |
| A | 0.69±0.07 |
| b | 0.24±0.05 |
| x | 0.05 |
| y | 0.08 |
| y1 | 0.20 |
| ZD | 0.50 |
| ZE | 0.50 |

P25FC-50-2N2

<R> • μPD78F0560K8-3B4-AX, 78F0561K8-3B4-AX, 78F0562K8-3B4-AX, 78F0565K8-3B4-AX, 78F0566K8-3B4-AX, 78F0567K8-3B4-AX

## 32-PIN PLASTIC WQFN(5x5)




(UNIT:mm)

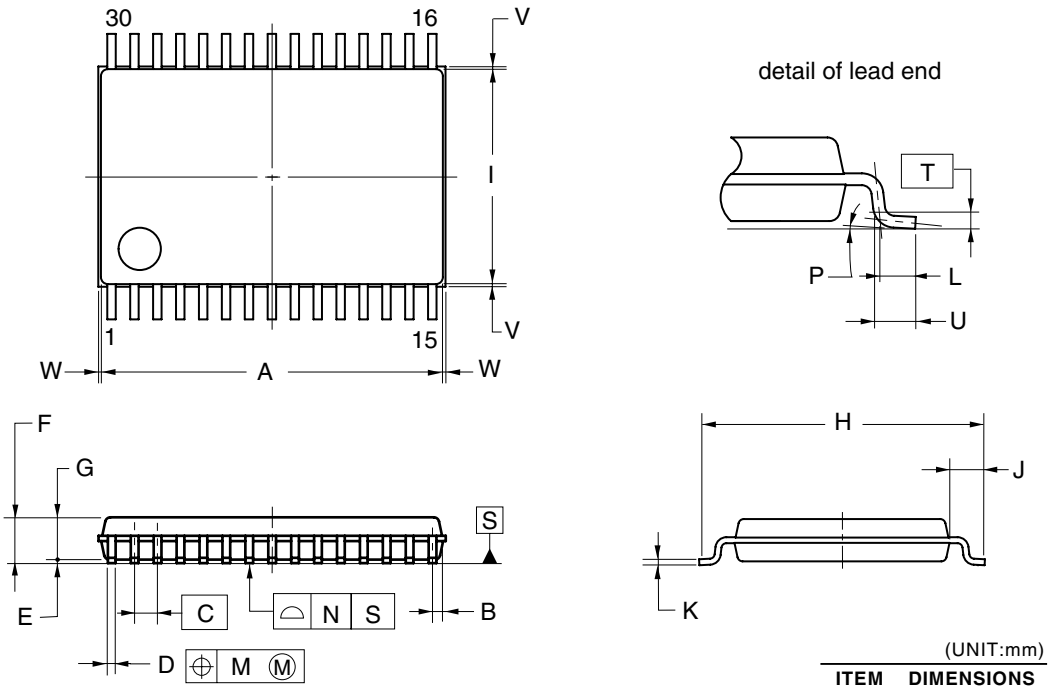
| ITEM | DIMENSIONS |
|---|---|
| D | 5.00±0.05 |
| E | 5.00±0.05 |
| D2 | 3.50±0.05 |
| E2 | 3.50±0.05 |
| A | 0.75±0.05 |
| b | $0.25^{+0.05}_{-0.07}$ |
| e | 0.50 |
| Lp | 0.40±0.10 |
| x | 0.05 |
| y | 0.05 |

P32K8-50-3B4-1

### 29.3 78K0/KB2-L

- μPD78F0571MC-CAB-AX, 78F0572MC-CAB-AX, 78F0573MC-CAB-AX, 78F0576MC-CAB-AX, 78F0577MC-CAB-AX, 78F0578MC-CAB-AX

## 30-PIN PLASTIC SSOP (7.62mm (300))




(UNIT:mm)

| ITEM | DIMENSIONS |
|---|---|
| A | 9.70±0.10 |
| B | 0.30 |
| C | 0.65 (T.P.) |
| D | $0.22^{+0.10}_{-0.05}$ |
| E | 0.10±0.05 |
| F | 1.30±0.10 |
| G | 1.20 |
| H | 8.10±0.20 |
| I | 6.10±0.10 |
| J | 1.00±0.20 |
| K | $0.15^{+0.05}_{-0.01}$ |
| L | 0.50 |
| M | 0.13 |
| N | 0.10 |
| P | $3^{\circ}{}^{+5^{\circ}}_{-3^{\circ}}$ |
| T | 0.25(T.P.) |
| U | 0.60±0.15 |
| V | 0.25 MAX. |
| W | 0.15 MAX. |

P30MC-65-CAB

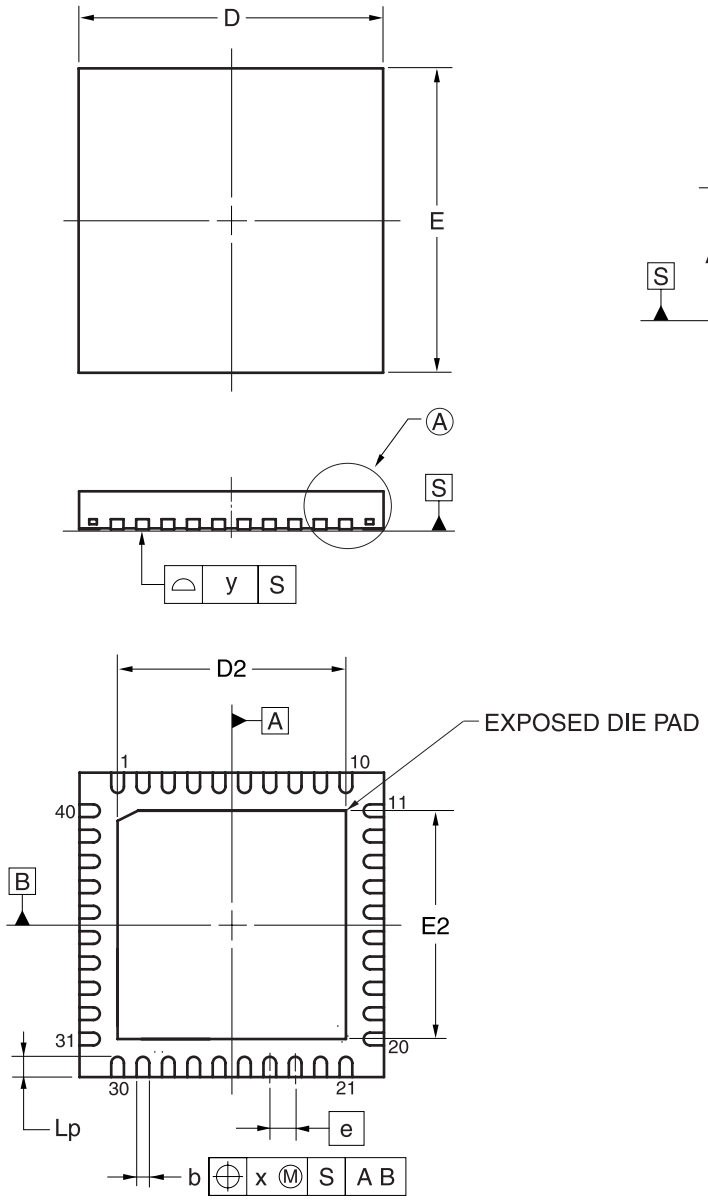
**NOTE**

Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

### 29.4 78K0/KC2-L

<R>• μPD78F0581K8-4B4-AX, 78F0582K8-4B4-AX, 78F0583K8-4B4-AX, 78F0586K8-4B4-AX, 78F0587K8-4B4-AX, 78F0588K8-4B4-AX

## 40-PIN PLASTIC WQFN(6x6)



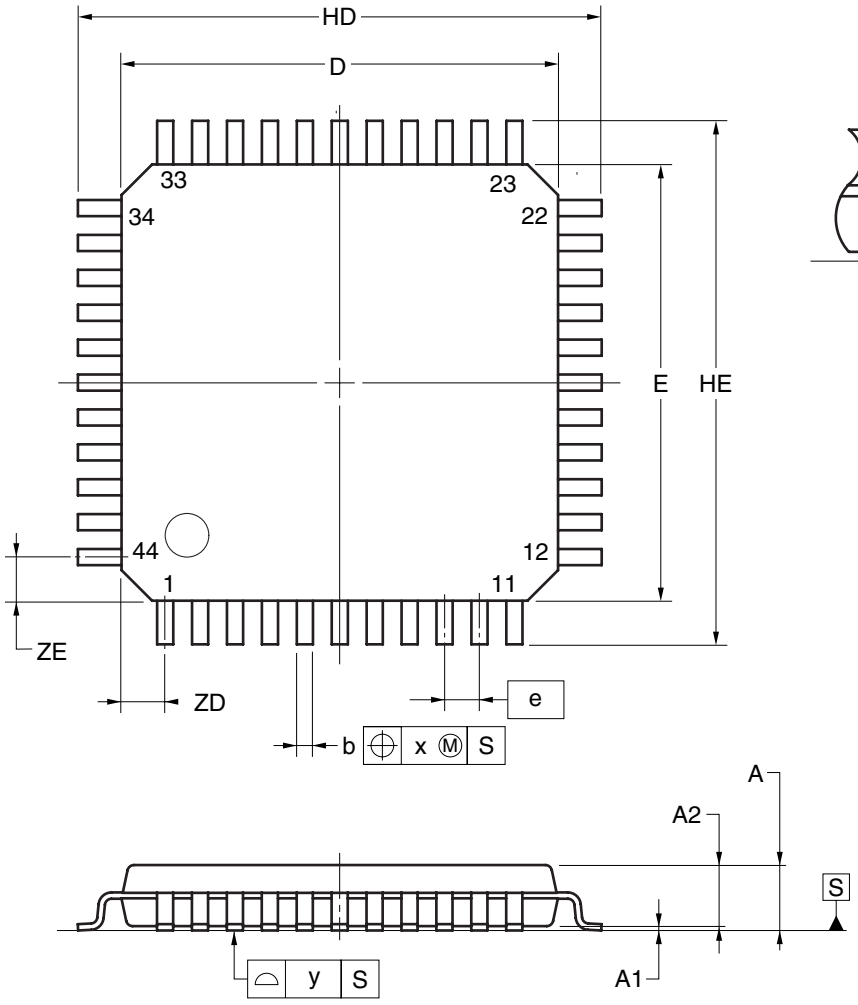

(UNIT:mm)

| ITEM | DIMENSIONS |
|---|---|
| D | 6.00±0.05 |
| E | 6.00±0.05 |
| D2 | 4.50±0.05 |
| E2 | 4.50±0.05 |
| A | 0.75±0.05 |
| b | $0.25^{+0.05}_{-0.07}$ |
| e | 0.50 |
| Lp | 0.40±0.10 |
| x | 0.05 |
| y | 0.05 |

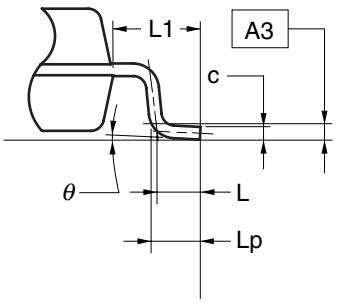
P40K8-50-4B4-1

- $\mu$PD78F0581GB-GAF-AX, 78F0582GB-GAF-AX, 78F0583GB-GAF-AX, 78F0586GB-GAF-AX, 78F0587GB-GAF-AX, 78F0588GB-GAF-AX

## 44-PIN PLASTIC LQFP (10x10)




detail of lead end




(UNIT:mm)

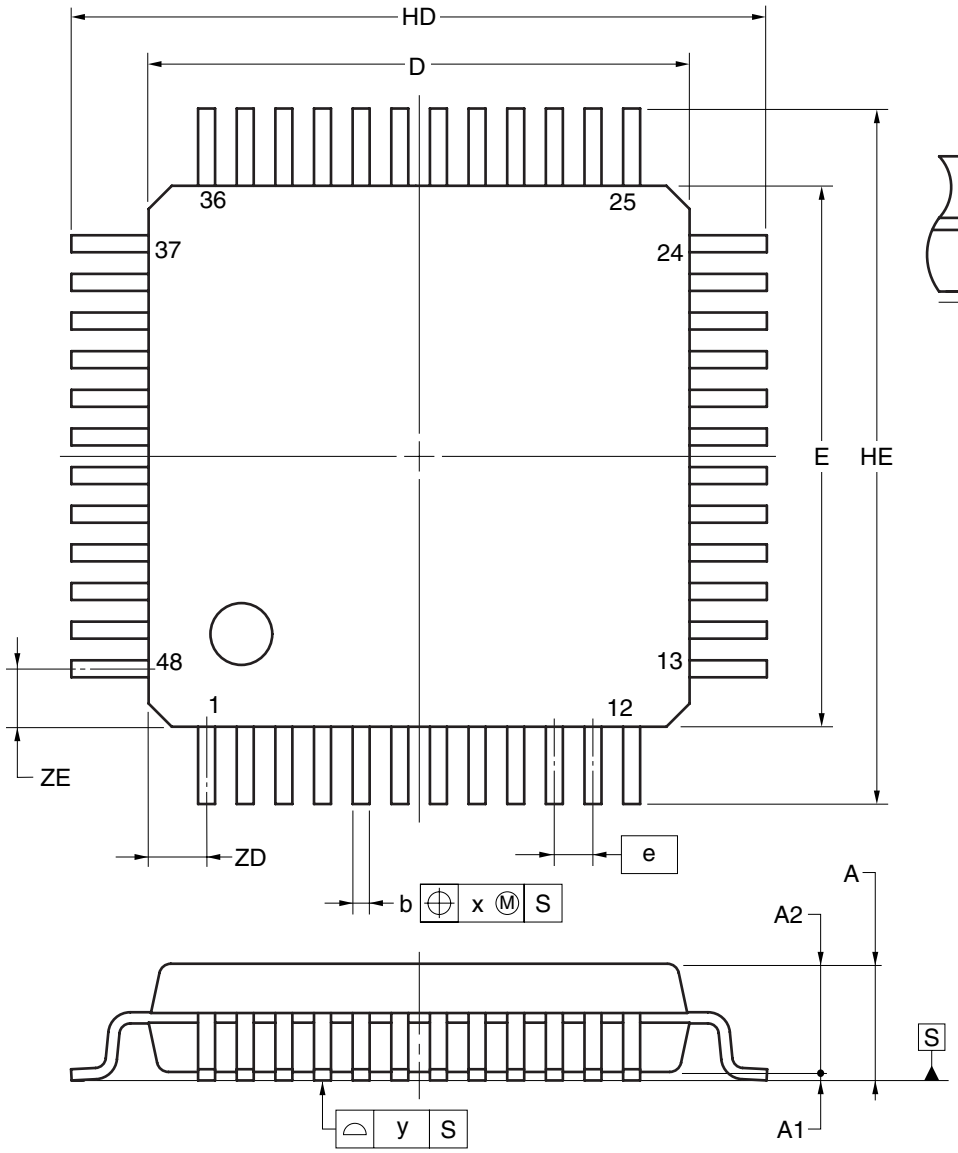
| ITEM | DIMENSIONS |
|---|---|
| D | 10.00±0.20 |
| E | 10.00±0.20 |
| HD | 12.00±0.20 |
| HE | 12.00±0.20 |
| A | 1.60 MAX. |
| A1 | 0.10±0.05 |
| A2 | 1.40±0.05 |
| A3 | 0.25 |
| b | $0.35^{+0.08}_{-0.04}$ |
| c | $0.125^{+0.075}_{-0.025}$ |
| L | 0.50 |
| Lp | 0.60±0.15 |
| L1 | 1.00±0.20 |
| θ | $3^{\circ}{}^{+5^{\circ}}_{-3^{\circ}}$ |
| e | 0.80 |
| x | 0.20 |
| y | 0.10 |
| ZD | 1.00 |
| ZE | 1.00 |

P44GB-80-GAF

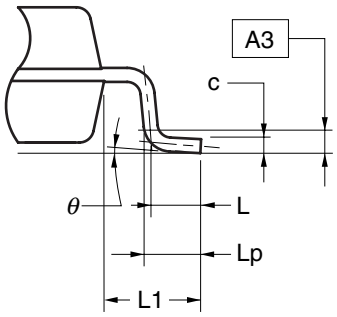
**NOTE**
Each lead centerline is located within 0.20 mm of its true position at maximum material condition.

- μPD78F0581GA-GAM-AX, 78F0582GA-GAM-AX, 78F0583GA-GAM-AX, 78F0586GA-GAM-AX, 78F0587GA-GAM-AX, 78F0588GA-GAM-AX

## 48-PIN PLASTIC LQFP (FINE PITCH) (7x7)




detail of lead end




(UNIT:mm)

| ITEM | DIMENSIONS |
|---|---|
| D | 7.00±0.20 |
| E | 7.00±0.20 |
| HD | 9.00±0.20 |
| HE | 9.00±0.20 |
| A | 1.60 MAX. |
| A1 | 0.10±0.05 |
| A2 | 1.40±0.05 |
| A3 | 0.25 |
| b | $0.20^{+0.07}_{-0.03}$ |
| c | $0.125^{+0.075}_{-0.025}$ |
| L | 0.50 |
| Lp | 0.60±0.15 |
| L1 | 1.00±0.20 |
| θ | $3^{\circ}{}^{+5^{\circ}}_{-3^{\circ}}$ |
| e | 0.50 |
| x | 0.08 |
| y | 0.08 |
| ZD | 0.75 |
| ZE | 0.75 |

P48GA-50-GAM

**NOTE**
Each lead centerline is located within 0.08 mm of its true position at maximum material condition.

# CHAPTER 30 RECOMMENDED SOLDERING CONDITIONS

These products should be soldered and mounted under the following recommended conditions.

For soldering methods and conditions other than those recommended below, please contact a Renesas Electronics sales representative.

For technical information, see the following website.

<R> Semiconductor Device Mount Manual (http://www2.renesas.com/pkg/en/mount/index.html)

**Table 30-1. Surface Mounting Type Soldering Conditions**

<R> **(1) 78K0/KY2-L, 78K0/KA2-L (20-pin products), 78K0/KB2-L, 78K0/KC2-L (44-pin and 48-pin products)**

| 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: 7 days[Note] (after that, prebake at 125°C for 10 to 72 hours) | IR60-107-3 |
| Partial heating | Pin temperature: 350°C max., Time: 3 seconds max. (per pin row) | – |

<R> **(2) 78K0/KA2-L (25-pin and 32-pin products), 78K0/KC2-L (40-pin products)**

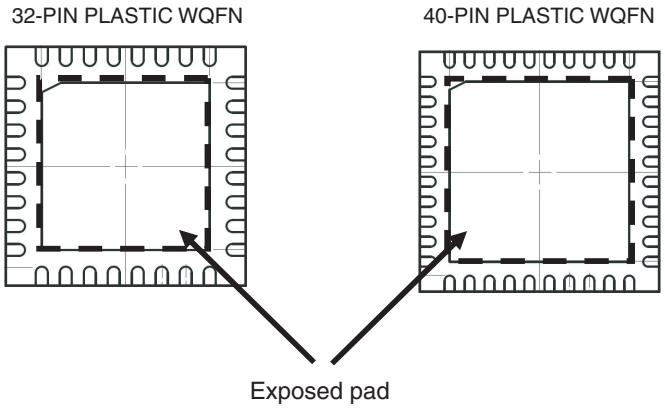
| Soldering Method | Soldering Conditions | Recommended Condition Symbol |
|---|---|---|
| Infrared reflow | Package peak temperature: 260°C, Time: 60 seconds max. (at 220°C or higher), Count: 3 times or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 to 72 hours) | IR60-107-3 |

**Note** After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

**Cautions 1. Do not use different soldering methods together (except for partial heating).**

**2. The 78K0/Kx2-L microcontrollers have an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.**

**3. Solder the exposed pad of a 32 or 40-pin plastic WQFN package, and set the potential to the same value as $V_{SS}$.**



# CHAPTER 31 CAUTIONS FOR WAIT

## 31.1 Cautions for Wait

This product has two internal system buses.

One is a CPU bus and the other is a peripheral bus that interfaces with the low-speed peripheral hardware.

Because the clock of the CPU bus and the clock of the peripheral bus are asynchronous, unexpected illegal data may be passed if an access to the CPU conflicts with an access to the peripheral hardware.

When accessing the peripheral hardware that may cause a conflict, therefore, the CPU repeatedly executes processing, until the correct data is passed.

As a result, the CPU does not start the next instruction processing but waits. If this happens, the number of execution clocks of an instruction increases by the number of wait clocks (for the number of wait clocks, refer to **Table 31-1**). This must be noted when real-time processing is performed.

## 31.2 Peripheral Hardware That Generates Wait

Table 31-1 lists the registers that issue a wait request when accessed by the CPU, and the number of CPU wait clocks.

**Table 31-1. Registers That Generate Wait and Number of CPU Wait Clocks**

| Peripheral Hardware | Register | Access | Number of Wait Clocks |
|---|---|---|---|
| Serial interface UART6 | ASIS6 | Read | 1 clock (fixed) |
| Serial interface IICA | IICAS0 | Read | 1 clock (fixed) |
| A/D converter | ADM0 | Write | 1 to 5 clocks (when $f_{AD} = f_{PRS}/2$ is selected)<br>1 to 7 clocks (when $f_{AD} = f_{PRS}/3$ is selected)<br>1 to 9 clocks (when $f_{AD} = f_{PRS}/4$ is selected)<br>2 to 13 clocks (when $f_{AD} = f_{PRS}/6$ is selected)<br>2 to 17 clocks (when $f_{AD} = f_{PRS}/8$ is selected)<br>2 to 25 clocks (when $f_{AD} = f_{PRS}/12$ is selected) |
| | ADS | Write | |
| | ADPC0, ADPC1 | Write | |
| | ADCR, ADCRH | Read | |
| | The above number of clocks is when the same source clock is selected for $f_{CPU}$ and $f_{PRS}$. The number of wait clocks can be calculated by the following expression and under the following conditions.<br><Calculating number of wait clocks><br>• Number of wait clocks = $\frac{2 f_{CPU}}{f_{AD}} + 1$<br>* Fraction is truncated if the number of wait clocks ≤ 0.5 and rounded up if the number of wait clocks > 0.5.<br>$f_{AD}$: A/D conversion clock frequency ($f_{PRS}$ to $f_{PRS}/12$)<br>$f_{CPU}$: CPU clock frequency<br>$f_{PRS}$: Peripheral hardware clock frequency<br>$f_{XP}$: Main system clock frequency<br><Conditions for maximum/minimum number of wait clocks><br>• Maximum number of times: Maximum speed of CPU ($f_{XP}$), lowest speed of A/D conversion clock ($f_{PRS}/12$)<br>• Minimum number of times: Minimum speed of CPU ($f_{SUB}$), highest speed of A/D conversion clock ($f_{PRS}$) | | |

**Caution When the peripheral hardware clock ($f_{PRS}$) is stopped, do not access the registers listed above using an access method in which a wait request is issued.**

**Remark** The clock is the CPU clock ($f_{CPU}$).

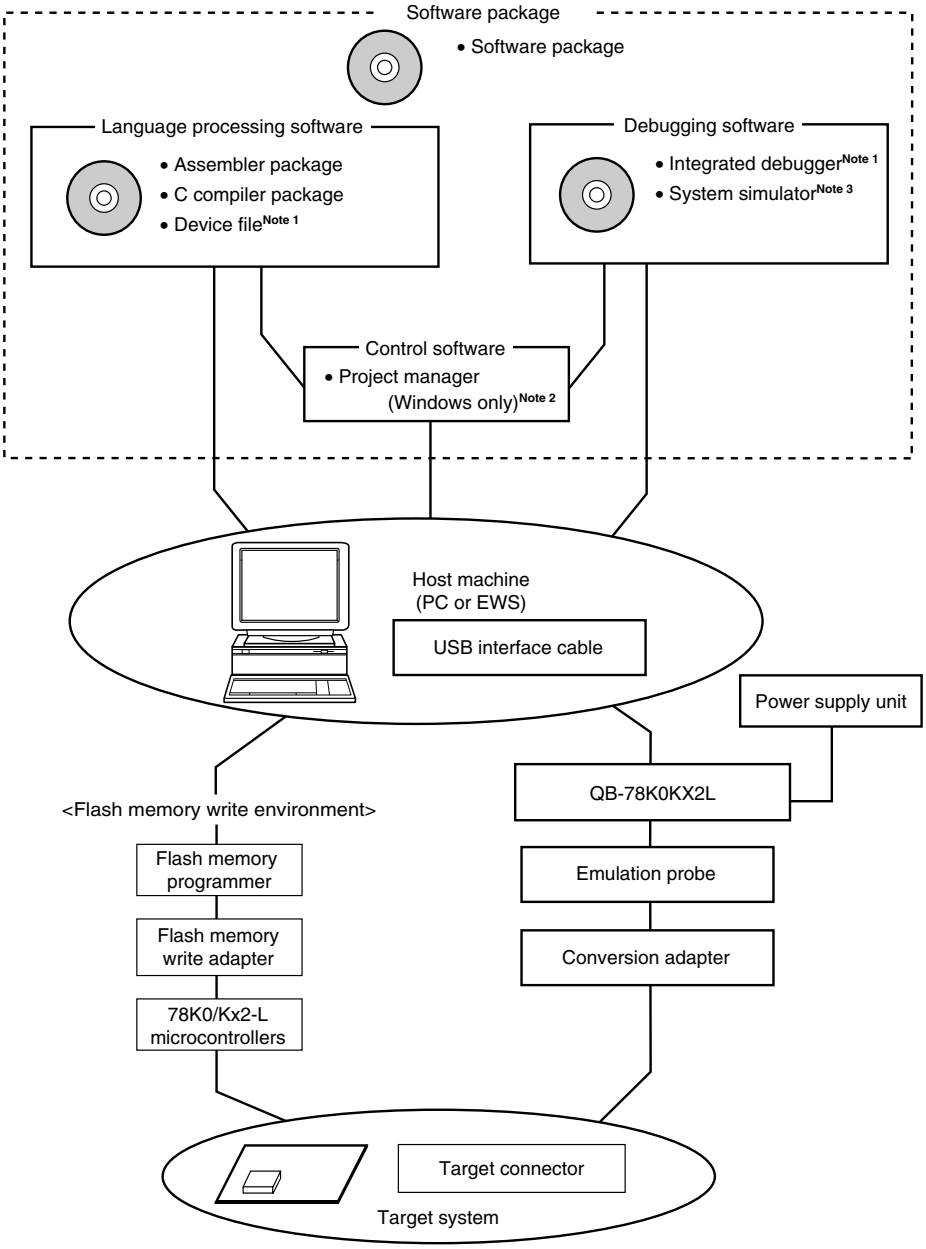
# APPENDIX A DEVELOPMENT TOOLS

The following development tools are available for the development of systems that employ the 78K0/Kx2-L microcontrollers.

Figure A-1 shows the development tool configuration.

**Figure A-1. Development Tool Configuration (1/2)**

**(1) When using the in-circuit emulator QB-78K0KX2L**



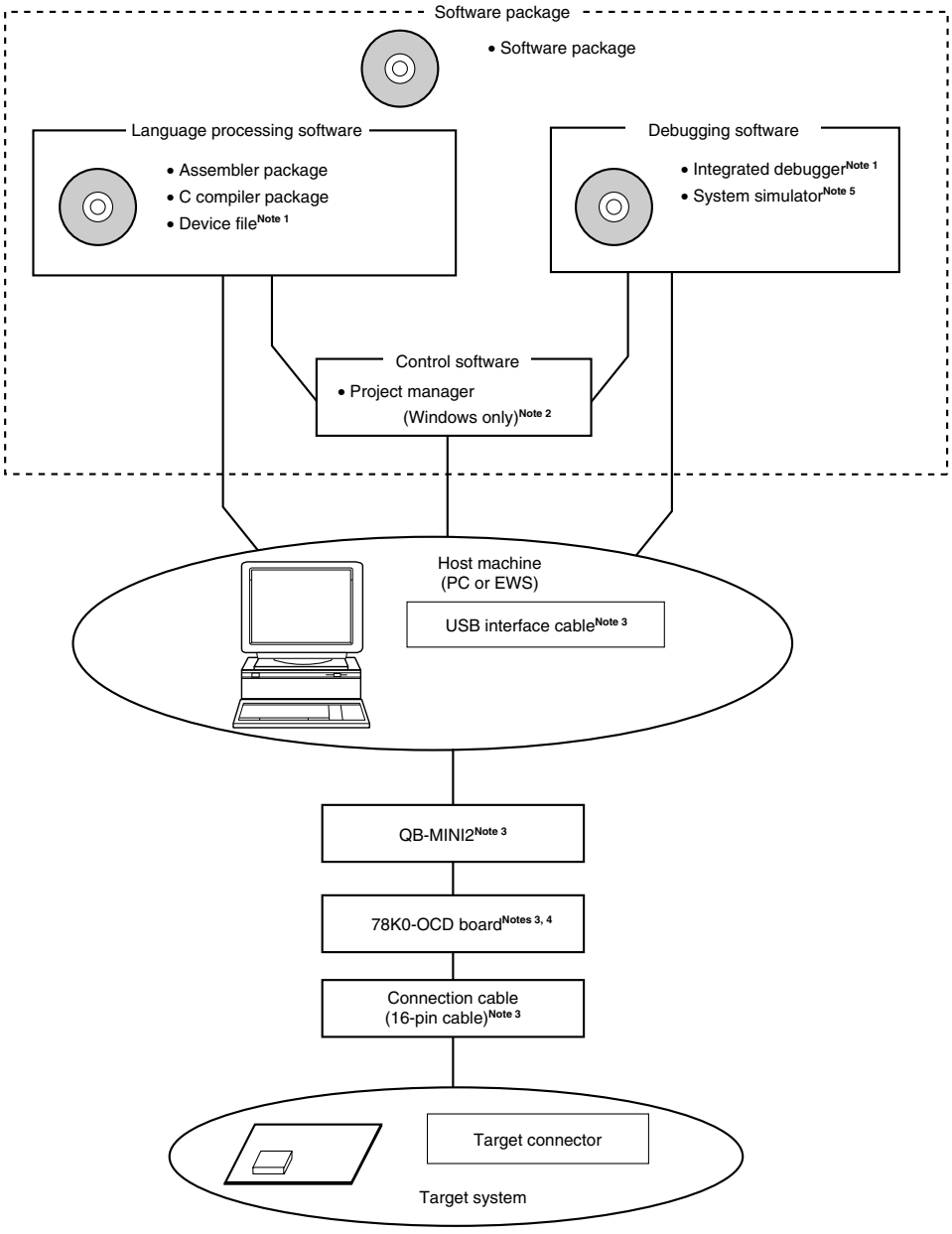

**Notes 1.** Download the device file for 78K0/Kx2-L microcontrollers (DF780588) and the integrated debugger ID78K0-QB from the download site for development tools
<R> (http://www2.renesas.com/micro/en/ods/index.html).

**2.** The project manager PM+ is included in the assembler package.
The PM+ is only used for Windows™.

**3.** This is an instruction simulation version included in the software package.

**Figure A-1. Development Tool Configuration (2/2)**

**(2) When using the on-chip debug emulator with programming function QB-MINI2**




**Notes 1.** Download the device file for 78K0/Kx2-L microcontrollers (DF780588) and the integrated debugger ID78K0-QB from the download site for development tools (http://www2.renesas.com/micro/en/ods/index.html). <R>

**2.** The project manager PM+ is included in the assembler package.
The PM+ is only used for Windows.

**3.** On-chip debug emulator QB-MINI2 is supplied with USB interface cable, connection cables (10-pin cable and 16-pin cable), and 78K0-OCD board. In addition, download the software for operating the QB-MINI2 from the download site for development tools (http://www2.renesas.com/micro/en/ods/index.html). <R>

**4.** This is used only when using QB-MINI2 as an on-chip debug emulator.

**5.** This is an instruction simulation version included in the software package.

## A.1 Software Package

| | |
|---|---|
| SP78K0<br>78K0 microcontroller software package | Development tools (software) common to the 78K0 microcontrollers are combined in this package. |

## A.2 Language Processing Software

| | |
|---|---|
| RA78K0[Note 1]<br>Assembler package | This assembler converts programs written in mnemonics into object codes executable with a microcontroller.<br>This assembler is also provided with functions capable of automatically creating symbol tables and branch instruction optimization.<br>This assembler should be used in combination with a device file (DF780588).<br>**<Precaution when using RA78K0 in PC environment>**<br>This assembler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (PM+) on Windows. PM+ is included in assembler package. |
| CC78K0[Note 1]<br>C compiler package | This compiler converts programs written in C language into object codes executable with a microcontroller.<br>This compiler should be used in combination with an assembler package and device file.<br>**<Precaution when using CC78K0 in PC environment>**<br>This C compiler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (PM+) on Windows. PM+ is included in assembler package. |
| DF780588[Note 2]<br>Device file | This file contains information peculiar to the device.<br>This device file should be used in combination with a tool (RA78K0, CC78K0, ID78K0-QB, and SM+ for 78K0).<br>The corresponding OS and host machine differ depending on the tool to be used. |

**Notes 1.** If the versions of RA78K0 and CC78K0 are Ver.4.00 or later, different versions of RA78K0 and CC78K0 can be installed on the same machine.

**2.** The DF780588 can be used in common with the RA78K0, CC78K0, ID78K0-QB, and SM+ for 78K0. Download the DF780588 from the download site for development tools <R> (http://www2.renesas.com/micro/en/ods/index.html).

## A.3 Flash Memory Programming Tools

### A.3.1 When using flash memory programmer PG-FP5 and FL-PR5

| | |
|---|---|
| PG-FP5, FL-PR5<br>Flash memory programmer | Flash memory programmer dedicated to microcontrollers with on-chip flash memory. |
| FA-xxxx[Note]<br>Flash memory programming adapter | Flash memory programming adapter used connected to the flash memory programmer for use. |

**Note** The part numbers of the flash memory programming adapter and the packages of the target device are described below.

<R>

| Packages of Target Device | | Flash Memory Programming Adapter |
|---|---|---|
| 78K0/KY2-L | 16-pin plastic SSOP (MA-FAA type) | FA-78F0557MA-FAA-RX |
| 78K0/KA2-L | 20-pin plastic SSOP (MC-CAA type) | FA-78F0567MC-CAA-RX |
| | 25-pin plastic FLGA (FC-2N2 type) | FA-78F0567FC-2N2-RX |
| | 32-pin plastic WQFN (K8-3B4 type) | FA-78F0567K8-3B4-RX |
| 78K0/KB2-L | 30-pin plastic SSOP (MC-CAB type) | FA-78F0578MC-CAB-RX |
| 78K0/KC2-L | 40-pin plastic WQFN (K8-4B4 types) | FA-78F0588K8-4B4-RX |
| | 44-pin plastic LQFP (GB-GAF types) | FA-78F0588GB-GAF-RX |
| | 48-pin plastic LQFP (GA-GAM types) | FA-78F0588GA-GAM-RX |

**Remarks 1.** FL-PR5 and FA-xxxx are products of Naito Densei Machida Mfg. Co., Ltd.
TEL: +81-42-750-4172 Naito Densei Machida Mfg. Co., Ltd.

**2.** Use the latest version of the flash memory programming adapter.

### A.3.2 When using on-chip debug emulator with programming function QB-MINI2

| | |
|---|---|
| QB-MINI2<br>On-chip debug emulator with programming function | This is a flash memory programmer dedicated to microcontrollers with on-chip flash memory. It is available also as on-chip debug emulator which serves to debug hardware and software when developing application systems using the 78K0/Kx2-L microcontrollers. When using this as flash memory programmer, it should be used in combination with a connection cable (16-pin cable) and a USB interface cable that is used to connect the host machine. |
| Target connector specifications | 16-pin general-purpose connector (2.54 mm pitch) |

**Remark** Download the software for operating the QB-MINI2 from the download site for development tools (http://www2.renesas.com/micro/en/ods/index.html).

<R>

## A.4 Debugging Tools (Hardware)

### A.4.1 When using in-circuit emulator

| | |
|---|---|
| QB-78K0KX2L<br>In-circuit emulator | This in-circuit emulator serves to debug hardware and software when developing application systems using the 78K0/Kx2-L microcontrollers. It supports to the integrated debugger (ID78K0-QB). This emulator should be used in combination with a power supply unit and emulation probe, and the USB is used to connect this emulator to the host machine. |

### A.4.2 When using on-chip debug emulator with programming function QB-MINI2

| | |
|---|---|
| QB-MINI2<br>On-chip debug emulator with programming function | This on-chip debug emulator serves to debug hardware and software when developing application systems using the 78K0/Kx2-L. It is available also as flash memory programmer dedicated to microcontrollers with on-chip flash memory. When using this as on-chip debug emulator, it should be used in combination with a connection cable (16-pin cable), a USB interface cable that is used to connect the host machine, and the 78K0-OCD board. |
| Target connector specifications | 16-pin general-purpose connector (2.54 mm pitch) |

<R> **Remark** Download the software for operating the QB-MINI2 from the download site for development tools (http://www2.renesas.com/micro/en/ods/index.html).

## A.5 Debugging Tools (Software)

| | |
|---|---|
| ID78K0-QB<br>Integrated debugger | This debugger supports the in-circuit emulators for the 78K0 microcontrollers. The ID78K0-QB is Windows-based software.<br>It has improved C-compatible debugging functions and can display the results of tracing with the source program using an integrating window function that associates the source program, disassemble display, and memory display with the trace result. It should be used in combination with the device file (DF780588). |
| SM+ for 78K0<br>System simulator | System simulator is Windows-based software.<br>It is used to perform debugging at the C source level or assembler level while simulating the operation of the target system on a host machine.<br>Use of system simulator allows the execution of application logical testing and performance testing on an independent basis from hardware development, thereby providing higher development efficiency and software quality.<br>System simulator should be used in combination with the device file (DF780588). |

# APPENDIX B REGISTER INDEX

## B.1 Register Index (In Alphabetical Order with Respect to Register Names)


**[A]**

A/D converter mode register 0 (ADM0) ..... 405
A/D port configuration register 0 (ADPC0) ..... 181, 413, 437
A/D port configuration register 1 (ADPC1) ..... 181, 413, 437
Alarm hour register (ALARMWH) ..... 384
Alarm minute register (ALARMWM) ..... 384
Alarm week register (ALARMWW) ..... 384
Analog input channel specification register (ADS) ..... 412, 439
Asynchronous serial interface control register 6 (ASICL6) ..... 461
Asynchronous serial interface operation mode register 6 (ASIM6) ..... 454
Asynchronous serial interface reception error status register 6 (ASIS6) ..... 457
Asynchronous serial interface transmission status register 6 (ASIF6) ..... 458

**[B]**

Baud rate generator control register 6 (BRGC6) ..... 460

**[C]**

Capture/compare control register 00 (CRC00) ..... 249
Clock operation mode select register (OSCCTL) ..... 202
Clock output selection register (CKS) ..... 399
Clock selection register 6 (CKSR6) ..... 458

**[D]**

Day count register (DAY) ..... 380

**[E]**

8-bit A/D conversion result register H (ADCRH) ..... 411
8-bit A/D conversion result register L (ADCRL) ..... 411
8-bit timer compare register 50 (CR50) ..... 317
8-bit timer compare register 51 (CR51) ..... 317
8-bit timer counter 50 (TM50) ..... 317
8-bit timer counter 51 (TM51) ..... 317
8-bit timer H carrier control register 1 (TMCYC1) ..... 343
8-bit timer H compare register 00 (CMP00) ..... 338
8-bit timer H compare register 01 (CMP01) ..... 338
8-bit timer H compare register 10 (CMP10) ..... 338
8-bit timer H compare register 11 (CMP11) ..... 338
8-bit timer H mode register 0 (TMHMD0) ..... 339
8-bit timer H mode register 1 (TMHMD1) ..... 339
8-bit timer mode control register 50 (TMC50) ..... 320
8-bit timer mode control register 51 (TMC51) ..... 320
External interrupt falling edge enable register 0 (EGNCTL0) ..... 619
External interrupt falling edge enable register 1 (EGNCTL1) ..... 619
External interrupt rising edge enable register 0 (EGPCTL0) ..... 619

External interrupt rising edge enable register 1 (EGPCTL1) ..... 619

**[H]**

Hour count register (HOUR) ..... 379

**[I]**

IICA control register 0 (IICACTL0) ..... 492
IICA control register 1 (IICACTL1) ..... 501
IICA flag register 0 (IICAF0) ..... 499
IICA high-level width setting register (IICWH) ..... 503
IICA low-level width setting register (IICWL) ..... 503
IICA shift register (IICA) ..... 490
IICA status register 0 (IICAS0) ..... 497
Input switch control register (ISC) ..... 463
Internal memory size switching register (IMS) ..... 699
Internal oscillation mode register (RCM) ..... 207
Interrupt mask flag register 0H (MK0H) ..... 606
Interrupt mask flag register 0L (MK0L) ..... 606
Interrupt mask flag register 1H (MK1H) ..... 606
Interrupt mask flag register 1L (MK1L) ..... 606
Interrupt request flag register 0H (IF0H) ..... 598
Interrupt request flag register 0L (IF0L) ..... 598
Interrupt request flag register 1H (IF1H) ..... 598
Interrupt request flag register 1L (IF1L) ..... 598

**[K]**

Key return mode register (KRM) ..... 638

**[L]**

Low-voltage detection level select register (LVIS) ..... 675
Low-voltage detection register (LVIM) ..... 672

**[M]**

Main clock mode register (MCM) ..... 209
Main OSC control register (MOC) ..... 208
Minute count register (MIN) ..... 379
Month count register (MONTH) ..... 382

**[O]**

Operational amplifier 0 control register (AMP0M) ..... 436
Operational amplifier 1 control register (AMP1M) ..... 436
Oscillation stabilization time counter status register (OSTC) ..... 210, 640
Oscillation stabilization time select register (OSTS) ..... 211, 641

**[P]**

Peripheral enable register 0 (PER0) ..... 212, 373
Port alternate switch control register (MUXSEL) ..... 183, 572
Port input mode register 6 (PIM6) ..... 179, 503
Port mode register 0 (PM0) ..... 167, 256
Port mode register 1 (PM1) ..... 167, 324, 345, 415, 440, 463, 573
Port mode register 2 (PM2) ..... 167, 415, 440

Port mode register 3 (PM3) ..... 167, 324, 345
Port mode register 4 (PM4) ..... 167, 385, 400, 573
Port mode register 6 (PM6) ..... 167, 463, 504, 573
Port mode register 7 (PM7) ..... 167
Port mode register 12 (PM12) ..... 167, 573, 676
Port output mode register 6 (POM6) ..... 180, 464, 504
Port register 0 (P0) ..... 172
Port register 1 (P1) ..... 172
Port register 2 (P2) ..... 172
Port register 3 (P3) ..... 172
Port register 4 (P4) ..... 172
Port register 6 (P6) ..... 172
Port register 7 (P7) ..... 172
Port register 12 (P12) ..... 172
Prescaler mode register 00 (PRM00) ..... 253
Priority specification flag register 0H (PR0H) ..... 613
Priority specification flag register 0L (PR0L) ..... 613
Priority specification flag register 1H (PR1H) ..... 613
Priority specification flag register 1L (PR1L) ..... 613
Processor clock control register (PCC) ..... 204
Pull-up resistor option register 0 (PU0) ..... 177
Pull-up resistor option register 1 (PU1) ..... 177
Pull-up resistor option register 3 (PU3) ..... 177
Pull-up resistor option register 4 (PU4) ..... 177
Pull-up resistor option register 6 (PU6) ..... 177
Pull-up resistor option register 7 (PU7) ..... 177
Pull-up resistor option register 12 (PU12) ..... 177

**[R]**

Real-time counter control register 0 (RTCC0) ..... 373
Real-time counter control register 1 (RTCC1) ..... 375
Real-time counter control register 2 (RTCC2) ..... 377
Receive buffer register 6 (RXB6) ..... 452
Receive shift register 6 (RXS6) ..... 453
Regulator mode control register (RMC) ..... 691
Reset control flag register (RESF) ..... 664
Reset pin mode register (RSTMASK) ..... 180

**[S]**

Second count register (SEC) ..... 378
Self programming mode control register (FPCTL) ..... 713
Serial clock selection register 10 (CSIC10) ..... 569
Serial clock selection register 11 (CSIC11) ..... 569
Serial I/O shift register 10 (SIO10) ..... 566
Serial I/O shift register 11 (SIO11) ..... 566
Serial operation mode register 10 (CSIM10) ..... 566
Serial operation mode register 11 (CSIM11) ..... 566
16-bit timer capture/compare register 000 (CR000) ..... 244

16-bit timer capture/compare register 010 (CR010) ..... 244
16-bit timer counter 00 (TM00) ..... 243
16-bit timer mode control register 00 (TMC00) ..... 248
16-bit timer output control register 00 (TOC00) ..... 251
Slave address register 0 (SVA0) ..... 490
Sub-count register (RSUBC) ..... 378

**[T]**

10-bit A/D conversion result register (ADCR) ..... 410
Timer clock selection register 50 (TCL50) ..... 318
Timer clock selection register 51 (TCL51) ..... 318
Transmit buffer register 10 (SOTB10) ..... 565
Transmit buffer register 11 (SOTB11) ..... 565
Transmit buffer register 6 (TXB6) ..... 453
Transmit shift register 6 (TXS6) ..... 453

**[W]**

Watch error correction register (SUBCUD) ..... 383
Watchdog timer enable register (WDTE) ..... 365
Week count register (WEEK) ..... 381

**[Y]**

Year count register (YEAR) ..... 382

## B.2 Register Index (In Alphabetical Order with Respect to Register Symbol)


**[A]**

ADCR: 10-bit A/D conversion result register ... 410
ADCRH: 8-bit A/D conversion result register H ... 411
ADCRL: 8-bit A/D conversion result register L ... 411
ADM0: A/D converter mode register 0 ... 405
ADPC0: A/D port configuration register 0 ... 181, 413, 437
ADPC1: A/D port configuration register 1 ... 181, 413, 437
ADS: Analog input channel specification register ... 412, 439
ALARMWH: Alarm hour register ... 384
ALARMWM: Alarm minute register ... 384
ALARMWW: Alarm week register ... 384
AMP0M: Operational amplifier 0 control register ... 436
AMP1M: Operational amplifier 1 control register ... 436
ASICL6: Asynchronous serial interface control register 6 ... 461
ASIF6: Asynchronous serial interface transmission status register 6 ... 458
ASIM6: Asynchronous serial interface operation mode register 6 ... 454
ASIS6: Asynchronous serial interface reception error status register 6 ... 457

**[B]**

BRGC6: Baud rate generator control register 6 ... 460

**[C]**

CKS: Clock output selection register ... 399
CKSR6: Clock selection register 6 ... 458
CMP00: 8-bit timer H compare register 00 ... 338
CMP01: 8-bit timer H compare register 01 ... 338
CMP10: 8-bit timer H compare register 10 ... 338
CMP11: 8-bit timer H compare register 11 ... 338
CR000: 16-bit timer capture/compare register 000 ... 244
CR010: 16-bit timer capture/compare register 010 ... 244
CR50: 8-bit timer compare register 50 ... 317
CR51: 8-bit timer compare register 51 ... 317
CRC00: Capture/compare control register 00 ... 249
CSIC10: Serial clock selection register 10 ... 569
CSIC11: Serial clock selection register 11 ... 569
CSIM10: Serial operation mode register 10 ... 566
CSIM11: Serial operation mode register 11 ... 566

**[D]**

DAY: Day count register ... 380

**[E]**

EGNCTL0: External interrupt falling edge enable register 0 ... 619
EGNCTL1: External interrupt falling edge enable register 1 ... 619
EGPCTL0: External interrupt rising edge enable register 0 ... 619
EGPCTL1: External interrupt rising edge enable register 1 ... 619

**[F]**

FPCTL: Self programming mode control register ..... 713

**[H]**

HOUR: Hour count register ..... 379

**[I]**

IF0H: Interrupt request flag register 0H ..... 598
IF0L: Interrupt request flag register 0L ..... 598
IF1H: Interrupt request flag register 1H ..... 598
IF1L: Interrupt request flag register 1L ..... 598
IICA: IICA shift register ..... 490
IICACTL0: IICA control register 0 ..... 492
IICACTL1: IICA control register 1 ..... 501
IICAF0: IICA flag register 0 ..... 499
IICAS0: IICA status register 0 ..... 497
IICWH: IICA high-level width setting register ..... 503
IICWL: IICA low-level width setting register ..... 503
IMS: Internal memory size switching register ..... 699
ISC: Input switch control register ..... 463

**[K]**

KRM: Key return mode register ..... 638

**[L]**

LVIM: Low-voltage detection register ..... 672
LVIS: Low-voltage detection level select register ..... 675

**[M]**

MCM: Main clock mode register ..... 209
MIN: Minute count register ..... 379
MK0H: Interrupt mask flag register 0H ..... 606
MK0L: Interrupt mask flag register 0L ..... 606
MK1H: Interrupt mask flag register 1H ..... 606
MK1L: Interrupt mask flag register 1L ..... 606
MOC: Main OSC control register ..... 208
MONTH: Month count register ..... 382
MUXSEL: Port alternate switch control register ..... 183, 572

**[O]**

OSCCTL: Clock operation mode select register ..... 202
OSTC: Oscillation stabilization time counter status register ..... 210, 640
OSTS: Oscillation stabilization time select register ..... 211, 641

**[P]**

P0: Port register 0 ..... 172
P1: Port register 1 ..... 172
P2: Port register 2 ..... 172
P3: Port register 3 ..... 172
P4: Port register 4 ..... 172
P6: Port register 6 ..... 172

P7: Port register 7 ..... 172
P12: Port register 12 ..... 172
PCC: Processor clock control register ..... 204
PER0: Peripheral enable register 0 ..... 212, 373
PIM6: Port input mode register 6 ..... 179, 503
PM0: Port mode register 0 ..... 167, 256
PM1: Port mode register 1 ..... 167, 324, 345, 415, 440, 463, 573
PM2: Port mode register 2 ..... 167, 415, 440
PM3: Port mode register 3 ..... 167, 324, 345
PM4: Port mode register 4 ..... 167, 385, 400, 573
PM6: Port mode register 6 ..... 167, 463, 504, 573
PM7: Port mode register 7 ..... 167
PM12: Port mode register 12 ..... 167, 573, 676
POM6: Port output mode register 6 ..... 180, 464, 504
PR0H: Priority specification flag register 0H ..... 613
PR0L: Priority specification flag register 0L ..... 613
PR1H: Priority specification flag register 1H ..... 613
PR1L: Priority specification flag register 1L ..... 613
PRM00: Prescaler mode register 00 ..... 253
PU0: Pull-up resistor option register 0 ..... 177
PU1: Pull-up resistor option register 1 ..... 177
PU3: Pull-up resistor option register 3 ..... 177
PU4: Pull-up resistor option register 4 ..... 177
PU6: Pull-up resistor option register 6 ..... 177
PU7: Pull-up resistor option register 7 ..... 177
PU12: Pull-up resistor option register 12 ..... 177

**[R]**

RCM: Internal oscillation mode register ..... 207
RESF: Reset control flag register ..... 664
RMC: Regulator mode control register ..... 691
RSTMASK: Reset pin mode register ..... 180
RSUBC: Sub-count register ..... 378
RTCC0: Real-time counter control register 0 ..... 373
RTCC1: Real-time counter control register 1 ..... 375
RTCC2: Real-time counter control register 2 ..... 377
RXB6: Receive buffer register 6 ..... 452
RXS6: Receive shift register 6 ..... 453

**[S]**

SEC: Second count register ..... 378
SIO10: Serial I/O shift register 10 ..... 566
SIO11: Serial I/O shift register 11 ..... 566
SOTB10: Transmit buffer register 10 ..... 565
SOTB11: Transmit buffer register 11 ..... 565
SUBCUD: Watch error correction register ..... 383
SVA0: Slave address register 0 ..... 490

**[T]**

TCL50: Timer clock selection register 50 ..... 318
TCL51: Timer clock selection register 51 ..... 318
TM00: 16-bit timer counter 00 ..... 243
TM50: 8-bit timer counter 50 ..... 317
TM51: 8-bit timer counter 51 ..... 317
TMC00: 16-bit timer mode control register 00 ..... 248
TMC50: 8-bit timer mode control register 50 ..... 320
TMC51: 8-bit timer mode control register 51 ..... 320
TMCYC1: 8-bit timer H carrier control register 1 ..... 343
TMHMD0: 8-bit timer H mode register 0 ..... 339
TMHMD1: 8-bit timer H mode register 1 ..... 339
TOC00: 16-bit timer output control register 00 ..... 251
TXB6: Transmit buffer register 6 ..... 453
TXS6: Transmit shift register 6 ..... 453

**[W]**

WDTE: Watchdog timer enable register ..... 365
WEEK: Week count register ..... 381

**[Y]**

YEAR: Year count register ..... 382

# APPENDIX C REVISION HISTORY

## C.1 Major Revisions in This Edition

(1/4)

| Page | Description | Classification |
|---|---|---|
| **Throughout** | | |
| – | Addition of 78K0/KA2-L (25, 32-pin products), 78K0/KC2-L (40-pin products) | (d) |
| – | Change URL of Renesas Electronics website | – |
| **CHAPTER 1 OUTLINE** | | |
| p.4 | Change of **1.2 Ordering Information** | (d) |
| pp.24, 25 | Change of **1.5 Outline of Functions** | (d) |
| **CHAPTER 2 PIN FUNCTIONS** | | |
| pp.27 to 30, 34, 35 | Change of the state of $\overline{\text{RESET}}$/P125 pin after reset in **2.1.1 78K0/KY2-L** to **2.1.3 78K0/KB2-L** | (c) |
| p.43 | Addition of **Caution** to **(a) ANI8 to ANI10** of **(2) Control mode** in **2.2.2 P10 to P17 (port 1)** | (c) |
| p.50 | Change of description of **2.2.8 P120 to P125 (port 12)** | (c) |
| p.51 | Change of **Caution** of **(2) Control mode** in **2.2.8 P120 to P125 (port 12)** | (c) |
| pp.54 to 59 | Change of **Table 2-2. Pin I/O Circuit Types (78K0/KY2-L)** to **Table 2-65. Pin I/O Circuit Types (78K0/KC2-L)** | (c) |
| p.63 | Change of Type 42-A in **Figure 2-1. Pin I/O Circuit List** | (c) |
| **CHAPTER 3 CPU ARCHITECTURE** | | |
| p.70 | Change of **Table 3-4. Vector Table** | (d) |
| pp.90 to 94 | Addition of **Table 3-8. Special Function Register List: 78K0/KA2-L (25-pin and 32-pin products)** | (d) |
| pp.100 to 105 | Change of **Table 3-10. Special Function Register List: 78K0/KC2-L** | (d) |
| **CHAPTER 4 PORT FUNCTIONS** | | |
| p.125 | Change of **Table 4-7. Port Configuration** | (c) |
| p.128 | Change of **Figure 4-3. Block Diagram of P02** | (c) |
| pp.131 to 136 | Change of **Figure 4-4. Block Diagram of P10**, **Figure 4-5. Block Diagram of P11**, and **Figure 4-6. Block Diagram of P12** | (c) |
| p.142 | Change of **Table 4-11. Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin** | (c) |
| p.144 | Change of **Figure 4-12. Block Diagram of P21** | (c) |
| p.160 | Change of description in **4.2.7 Port 7** | (d) |
| p.160 | Addition of **Table 4-13. Setting Functions of P70/ANI8 to P72/ANI10 Pins** | (d) |
| p.161 | Addition of **(2) 78K0/KA2-L (32-pin products)** to **Figure 4-26. Block Diagram of P70 to P75** | (d) |
| p.162 | Addition of **Caution** to **4.2.8 Port 12** | (c) |
| p.163 | Addition of **Caution** to **Figure 4-29. Block Diagram of P125** | (c) |
| p.166 | Addition of **Figure 4-32. Format of Port Mode Register (78K0/KA2-L (25-pin and 32-pin products)** | (d) |
| p.171 | Change of **Figure 4-34. Format of Port Mode Register (78K0/KC2-L)** | (d) |
| p.174 | Addition of **Figure 4-37. Format of Port Register (78K0/KA2-L (25-pin and 32-pin products))** | (d) |
| p.176 | Change of **Figure 4-39. Format of Port Register (78K0/KC2-L)** | (d) |
| p.179 | Change of **Figure 4-44. Format of Pull-up Resistor Option Register (78K0/KC2-L)** | (d) |
| pp.194 to 196 | Addition of description to **Table 4-18. Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KC2-L)** | (c) |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(2/4)

| Page | Description | Classification |
|---|---|---|
| **CHAPTER 5 CLOCK GENERATOR** | | |
| p.212 | Change of description of **(9) Peripheral enable register 0 (PER0)** in **5.3 Registers Controlling Clock Generator** | (c) |
| p.212 | Change of **Figure 5-12. Format of Peripheral Enable Register 0 (PER0)** | (c) |
| p.230 | Addition of **Note** to **Figure 5-18. CPU Clock Status Transition Diagram (When LVI Default Start Mode Function Stopped Is Set (Option Byte: LVISTART = 0), 78K0/KY2-L, 78K0/KA2-L, and 78K0/KB2-L)** | (c) |
| p.231 | Addition of **Note** to **Figure 5-19. CPU Clock Status Transition Diagram (When LVI Default Start Mode Function Stopped Is Set (Option Byte: LVISTART = 0), 78K0/KC2-L)** | (c) |
| p.235 | Addition of **Note** to **(11) • STOP mode (H) set while CPU is operating with internal high-speed oscillation clock (B) • STOP mode (I) set while CPU is operating with high-speed system clock (C)** in **Table 5-6. CPU Clock Transition and SFR Register Setting Examples (4/4)** | (c) |
| **CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00** | | |
| p.255 | Addition of **(5) Port alternate switch control register (MUXSEL) (78K0/KA2-L (25-pin and 32-pin products) only)** to **6.3 Registers Controlling 16-Bit Timer/Event Counter 00** | (d) |
| **CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50 AND 51** | | |
| pp.318, 324, 325 | Addition of MUXSEL and PM0 to **7.3 Registers Controlling 8-Bit Timer/Event Counters 50 and 51** | (d) |
| p.324 | Change of **(3) Port alternate switch control register (MUXSEL) (78K0/KA2-L (25-pin) only)** and **(4) Port mode registers 0, 1, 3 (PM0, PM1, PM3)** of **7.3 Registers Controlling 8-Bit Timer/Event Counters 50 and 51** | (d) |
| p.325 | Addition of **Figure 7-12. Format of Port Mode Register 0 (PM0)** | (d) |
| **CHAPTER 8 8-BIT TIMERS H0 AND H1** | | |
| p.339 | Addition of name of registers to **8.3 Registers Controlling 8-Bit Timers H0 and H1** | (d) |
| pp.344, 345 | Change of **(3) Port alternate switch control register (MUXSEL) (78K0/KA2-L (25, 32-pin products) only)** and **(4) Port mode register 0 (PM0), port mode register 1 (PM1), port mode register 3 (PM3)** of **8.3 Registers Controlling 8-Bit Timers H0 and H1** | (d) |
| p.345 | Addition of **Figure 8-9. Format of Port Mode Register 0 (PM0)** | (d) |
| **CHAPTER 10 REAL-TIME COUNTER** | | |
| p.373 | Change of **(1) Peripheral enable register 0 (PER0) of 10.3 Registers Controlling Real-Time Counter** | (c) |
| **CHAPTER 12 A/D CONVERTER** | | |
| p.402 | Change of **Figure 12-1. Block Diagram of A/D Converter** | (c) |
| p.403 | Change of **(4) PGAOUT signal (products with operational amplifier only)** in **12.2 Configuration of A/D Converter** | (c) |
| p.412 | Change of **Figure 12-8. Format of Analog Input Channel Specification Register (ADS)** | (c) |
| p.413 | Change of **(6) A/D port configuration registers 0, 1 (ADPC0, ADPC1)** in **12.2 Configuration of A/D Converter** | (d) |
| pp.413 to 415 | Change of **Figure 12-9. Format of A/D Port Configuration Registers 0, 1 (ADPC0, ADPC1)** | (d) |
| p.415 | Change of **(7) Port mode registers 1, 2, 7 (PM1, PM2, PM7)** in **12.2 Configuration of A/D Converter** | (d) |
| p.416 | Change of **Figure 12-11. Format of Port Mode Register 2 (PM2)** | (d) |
| p.417 | Addition of **Figure 12-12. Format of Port Mode Register 7 (PM7) (78K0/KA2-L (32-pin products))** | (d) |
| p.420 | Change of **Table 12-6. Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin** | (c) |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(3/4)

| Page | Description | Classification |
|---|---|---|
| **CHAPTER 12 A/D CONVERTER** | | |
| p.421 | Change of **Table 12-8. Setting Functions of P70/ANI8 to P72/ANI10 Pins** | (d) |
| p.433 | Change of mode name in **Table 12-9. Resistance and Capacitance Values of Equivalent Circuit (Reference Values)** | (c) |
| **CHAPTER 13 OPERATIONAL AMPLIFIERS** | | |
| p.435 | Change of **Figure 13-1. Block Diagram of Operational Amplifier** | (c) |
| p.439 | Change of **Figure 13-6. Format of Analog Input Channel Specification Register (ADS)** | (c) |
| p.444 | Change of **Table 13-5. Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin** | (c) |
| **CHAPTER 16 SERIAL INTERFACES CSI10 AND CSI11** | | |
| p.573 | Change of **(4) Port mode registers 0, 1, 3, 4, 6, 12 (PM0, PM1, PM3, PM4, PM6, PM12)** in **16.3 Registers Controlling Serial Interfaces CSI10 and CSI11** | (d) |
| p.574 | Addition of **Figure 16-9. Format of Port Mode Register 0 (PM0)** | (d) |
| p.574 | Addition of **Figure 16-11. Format of Port Mode Register 3 (PM3)** | (d) |
| **CHAPTER 17 INTERRUPT FUNCTIONS** | | |
| pp.592, 593 | Change of **Table 17-1. Interrupt Source List** | (d) |
| **CHAPTER 19 STANDBY FUNCTION** | | |
| p.649 | Addition of **Caution** in **Table 19-3. Operating Statuses in STOP Mode** | (c) |
| **CHAPTER 20 RESET FUNCTION** | | |
| pp.662, 663 | Change of **Note** in **Table 20-2. Hardware Statuses After Reset Acknowledgment** | (b) |
| p.664 | Change of **Table 20-3. RESF Status When Reset Request Is Generated** | (b) |
| **CHAPTER 21 POWER-ON-CLEAR CIRCUIT** | | |
| p.668 | Change of **Figure 21-2. Timing of Generation of Internal Reset Signal by Power-on-Clear Circuit and Low-Voltage Detector (2/2)** | (b) |
| **CHAPTER 22 LOW-VOLTAGE DETECTOR** | | |
| p.673 | Change of **Note 1** in **Figure 22-2. Format of Low-Voltage Detection Register (LVIM)** | (b) |
| p.675 | Change of **Note** in **Figure 22-3. Format of Low-Voltage Detection Level Select Register (LVIS)** | (b) |
| p.676 | Change of **Remark 1** in **22.4 (1) Used as reset (LVIMD = 1)** | (b) |
| p.680 | Change of description in **22.4.1 (1) (b) When LVI default start function enabled is set (LVISTART = 1)** | (b) |
| p.680 | Change of **Figure 22-6. Timing of Low-Voltage Detector Internal Reset Signal Generation (Bit: LVISEL = 0, Option Byte: LVISTART = 1)** | (b) |
| p.685 | Change of description in **22.4.2 (1) (b) When LVI default start function enabled is set (LVISTART = 1)** | (b) |
| p.685 | Change of **Figure 22-9. Timing of Low-Voltage Detector Interrupt Signal Generation (Bit: LVISEL = 0, Option Byte: LVISTART = 1)** | (b) |
| **CHAPTER 23 REGULATOR** | | |
| p.691 | Change of **(1) Regulator mode control register (RMC)** in **23.2 Register Controlling Regulator** | (b) |
| **CHAPTER 24 OPTION BYTE** | | |
| p.694 | Change of **(4) 0083H/1083H** in **24.1 Functions of Option Bytes** | (b) |
| p.696 | Change of description of LVISTART bit in **Figure 24-1. Format of Option Byte (2/3)** | (b) |
| p.697 | Change of **Figure 24-1. Format of Option Byte (3/3)** | (b) |
| p.698 | Change of description example of software in **24.2 Format of Option Byte** | (b) |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(4/4)

| Page | Description | Classification |
|---|---|---|
| **CHAPTER 25 FLASH MEMORY** | | |
| p.712 | Change of **Remark** in **25.8 Flash Memory Programming by Self Programming** | (e) |
| **CHAPTER 26 ON-CHIP DEBUG FUNCTION** | | |
| p.718 | Addition of **Caution 2** in **26.1 Connecting QB-MINI2 to 78K0/Kx2-L Microcontrollers** | (c) |
| p.720 | Addition of **Figure 26-1. Connection Example of QB-MINI2 and 78K0/Ix2 Microcontrollers (2/3)** | (c) |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

## C.2 Revision History of Preceding Editions

Here is the revision history of the preceding editions. Chapter indicates the chapter of each edition.

(1/8)

| Edition | Description | Chapter |
|---|---|---|
| 2nd Edition | Modification of P60 and P61 pins alternate function in the 78K0/KY2-L and 78K0/KA2-L | **Throughout** |
| | **1.1 Features**<br>• Modification of description of low power consumption<br>• Modification of description and table of serial interface<br>• Modification of 10-bit resolution A/D conversion | **CHAPTER 1 OUTLINE** |
| | Modification of **1.2 Ordering Information** | |
| | Modification of pin configurations in **1.3.1 78K0/KY2-L** and **1.3.2 78K0/KA2-L** | |
| | Modification of **1.3.3 78K0/KB2-L** and **1.3.4 78K0/KC2-L** | |
| | Modification of **1.4.1 78K0/KY2-L** to **1.4.4 78K0/KC2-L** | |
| | Modification of **1.5 Outline of Functions** | |
| | Modification of **2.1.3 78K0/KB2-L** and **2.1.4 78K0/KC2-L** | **CHAPTER 2 PIN FUNCTIONS** |
| | Modification of table of pins in **2.2.2 P10 to P17 (port 1)** | |
| | Modification of **2.2.10 (b) IC** | |
| | Modification of **Table 2-4 Pin I/O Circuit Types (78K0/KB2-L)** | |
| | Modification of **Table 2-5 Pin I/O Circuit Types (78K0/KC2-L) (1/2)** | |
| | Modification of **Table 3-4 Vector Table** | **CHAPTER 3 CPU ARCHITECTURE** |
| | Addition of 8-bit A/D conversion result register L and modification of serial I/O shift register 10, serial operation mode register 10, serial clock selection register 10, and transmit buffer register 10 in **Table 3-6 Special Function Register List** | |
| | Modification of **Table 4-4 Port Functions (78K0/KB2-L)** | **CHAPTER 4 PORT FUNCTIONS** |
| | Modification of **Table 4-5 Port Functions (78K0/KC2-L) (1/2)** | |
| | Modification of **Note** in **Table 4-6 Port Configuration** | |
| | Modification of table of pins and description in **4.2.2 Port 1** | |
| | Modification of **Table 4-7 Setting Functions of P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins** | |
| | Modification of **Figure 4-4 Block Diagram of P10** | |
| | Modification of **Figure 4-5 Block Diagram of P11** | |
| | Modification of **Figure 4-6 Block Diagram of P12** | |
| | Modification of **Table 4-9 Setting Functions of P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins** | |
| | Modification of **Figure 4-19 Block Diagram of P40** | |
| | Addition of **Figure 4-20 Block Diagram of P41** and modification of **Figure 4-21 Block Diagram of P42** | |
| | Modification of **Figure 4-27 Block Diagram of P120 (2/2)** | |
| | Modification of **Note** in **4.3 (7) A/D port configuration registers 0, 1 (ADPC0, ADPC1)** | |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(2/8)

| Edition | Description | Chapter |
|---|---|---|
| 2nd Edition | Modification of **Figure 4-45 Format of A/D Port Configuration Register 0 (ADPC0)** | **CHAPTER 4 PORT FUNCTIONS** |
| | Modification of **Figure 4-46 Format of A/D Port Configuration Register 1 (ADPC1) (78K0/KB2-L and 78K0/KC2-L Only)** | |
| | Modification of PM×× and P×× value of P125 pin in **Table 4-12 Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KY2-L)** to **Table 4-15 Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KC2-L)** | |
| | Modification of **Table 4-14 Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KB2-L) (1/2)** and **Table 4-15 Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KC2-L) (1/3)** | |
| | Modification of **Figure 5-2 Block Diagram of Clock Generator (78K0/KC2-L)** | **CHAPTER 5 CLOCK GENERATOR** |
| | Modification of and addition of **Caution 3** to **Figure 5-4 Format of Clock Operation Mode Select Register (OSCCTL) (78K0/KC2-L)** | |
| | Modification of **Caution 1** in **5.4 System Clock Oscillator** | |
| | Modification of **Figure 6-9 Format of Prescaler Mode Register 00 (PRM00)** | **CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00** |
| | Modification of **Figure 6-13 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 6-17 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 6-20 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 6-30 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 6-38 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 6-41 (d) Prescaler mode register 00 (PRM00)** and **(f) 16-bit capture/compare register 000 (CR000)** | |
| | Modification of **Figure 6-44 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 6-51 (d) Prescaler mode register 00 (PRM00)** | |
| | Modification of **Figure 7-2 Block Diagram of 8-bit Timer 51 (78K0/KY2-L, 78K0/KA2-L)** and **Figure 7-3 Block Diagram of 8-bit Timer 51 (78K0/KB2-L, 78K0/KC2-L)** | **CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50 AND 51** |
| | Modification of **Figure 7-7 Format of Timer Clock Selection Register 51 (TCL51)** | |
| | Deletion of **Caution 2** from and modification of **Remark** in **Table 9-4 Setting Window Open Period of Watchdog Timer** | **CHAPTER 9 WATCHDOG TIMER** |
| | • Modification of the number of channels in the 78K0/KB2-L and 78K0/KC2-L<br>• Addition of 8-bit A/D conversion result register L (ADCRL) | **CHAPTER 12 A/D CONVERTER** |
| | Modification of **Table 12-2 A/D Conversion Time Selection** | |
| | Partial deletion of description in **12.3 (2) 10-bit A/D conversion result register (ADCR)** | |
| | Modification of **Figure 12-9 Format of A/D Port Configuration Register 0 (ADPC0)** | |
| | Modification of **Figure 12-10 Format of A/D Port Configuration Register 1 (ADPC1) (78K0/KB2-L and 78K0/KC2-L Only)** | |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(3/8)

| Edition | Description | Chapter |
|---|---|---|
| 2nd Edition | Modification of **Table 12-3 Setting Functions of P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins** | **CHAPTER 12 A/D CONVERTER** |
| | Modification of **Table 12-5 Setting Functions of P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins** | |
| | Deletion of **Caution 2** in **12.4.1 Basic operations of A/D converter** | |
| | Modification of description of setting methods and deletion of **Caution 2** in **12.4.3 (1) A/D conversion operation** | |
| | Modification of **Figure 13-1 Block Diagram of Operational Amplifier** | **CHAPTER 13 OPERATIONAL AMPLIFIERS** |
| | Addition of **Remark** to **Figure 13-2 Format of Operational Amplifier 0 Control Register (AMP0M) (Products with Operational Amplifier Only)** | |
| | Modification of **Figure 13-4 Format of A/D Port Configuration Register 0 (ADPC0)** | |
| | Modification of **Figure 13-5 Format of A/D Port Configuration Register 1 (ADPC1) (78K0/KB2-L and 78K0/KC2-L Only)** | |
| | Modification of **Table 13-2 Setting Functions of P10/ANI8/AMP1-, P12/ANI10/AMP1+ Pins** | |
| | Modification of **Table 13-4 Setting Functions of P20/ANI0/AMP0-, P22/ANI2/AMP0+ Pins** | |
| | Modification of **Remark** in **Figure 14-4 Block Diagram of Serial Interface UART6** | **CHAPTER 14 SERIAL INTERFACE UART6** |
| | Addition of **Note 3** to **Figure 14-8 Format of Clock Selection Register 6 (CKSR6)** | |
| | Modification of description in **14.3 (8) Port mode register 1 (PM1), port mode register 6 (PM6)** | |
| | Modification of **(1) 78K0/KY2-L and 78K0/KA2-L** in **Table 14-2 Relationship Between Register Settings and Pins** | |
| | Addition of **15.4.2 Setting transfer clock by using IICWL and IICWH registers** | **CHAPTER 15 SERIAL INTERFACE IICA** |
| | Modification of the mounted situation in the 78K0/KB2-L and 78K0/KC2-L | **CHAPTER 16 SERIAL INTERFACES CSI10 AND CSI11** |
| | Modification of description in **16.3 (4) Port mode registers 1, 4, 6, 12 (PM1, PM4, PM6, PM12)** | |
| | Modification of and addition of **Notes 3** and **4** to **Table 16-3 SO1n Output Status** | |
| | Modification of maskable interrupts (internal) in the 78K0/KB2-L and 78K0/KC2-L | **CHAPTER 17 INTERRUPT FUNCTIONS** |
| | Modification of **Table 17-1 Interrupt Source List (1/2)** | |
| | Modification of **Table 17-2 Flags Corresponding to Interrupt Request Sources (1/2)** | |
| | Modification of **Caution** in **Figure 17-4 Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (78K0/KB2-L)** to **Figure 17-6 Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H) (48-pin products of 78K0/KC2-L)** | |
| | Modification of **Caution** in **Figure 17-9 Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (78K0/KB2-L)** to **Figure 17-11 Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H) (48-pin products of 78K0/KC2-L)** | |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(4/8)

| Edition | Description | Chapter |
|---|---|---|
| 2nd Edition | Modification of **Caution** in **Figure 17-14 Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (78K0/KB2-L)** to **Figure 17-16 Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (48-pin products of 78K0/KC2-L)** | **CHAPTER 17 INTERRUPT FUNCTIONS** |
| | Addition of **Caution 4** to **19.1.1 (2) STOP mode** | **CHAPTER 19 STANDBY FUNCTION** |
| | Addition of 8-bit A/D conversion result register L (ADCRL) to **Table 20-2 Hardware Statuses After Reset Acknowledgment (3/4)** | **CHAPTER 20 RESET FUNCTION** |
| | Addition of **Note 5** to **Figure 21-2 Timing of Generation of Internal Reset Signal by Power-on-Clear Circuit and Low-Voltage Detector (2/2)** | **CHAPTER 21 POWER-ON-CLEAR CIRCUIT** |
| | Modification of **Note 4** in **Figure 22-2 Format of Low-Voltage Detection Register (LVIM)** | **CHAPTER 22 LOW-VOLTAGE DETECTOR** |
| | Modification of **Figure 22-3 Format of Low-Voltage Detection Level Select Register (LVIS)**, modification of **Note 1** | |
| | Modification of description in **22.4.1 (1) (b) When LVI default start function enabled is set (LVISTART = 1) • When starting operation** | |
| | Modification of description in **22.4.2 (1) (b) When LVI default start function enabled is set (LVISTART = 1) • When starting operation** | |
| | Modification of **Cautions 1** and **3** in **Figure 23-1 Format of Regulator Mode Control Register (RMC)** | **CHAPTER 23 REGULATOR** |
| | Modification of **Caution** in **24.1 (2) 0081H/1081H** | **CHAPTER 24 OPTION BYTE** |
| | Modification of **Note 1** in **Figure 24-1 Format of Option Byte (2/3)** | |
| | Modification of **Caution 2** in **Figure 24-1 Format of Option Byte (3/3)** | |
| | Addition of **Note** to **Figure 25-2 Environment for Writing Program to Flash Memory** | **CHAPTER 25 FLASH MEMORY** |
| | Addition of **Note** to **Table 25-2 Pin Connection** | |
| | Modification of **25.4.2 TOOLD0 and TOOLD1 pins** | |
| | Modification of **Caution 3** and **Remark** in **25.7 Flash Memory Programming by Self Programming** | |
| | Revision of **Figure 26-1 Connection Example of QB-MINI2 and 78K0/Kx2-L Microcontrollers** | **CHAPTER 26 ON-CHIP DEBUG FUNCTION** |
| | Modification of A/D converter pins in **(2) Non-port functions** | **CHAPTER 28 ELECTRICAL SPECIFICATIONS (TARGET VALUES)** |
| | Modification of oscillation frequency ($f_{IH}$) in **Internal High-speed Oscillator Characteristics** | |
| | Modification of oscillation clock frequency ($f_{IL}$ = 30 kHz) in **Internal Low-speed Oscillator Characteristics** | |
| | **DC Characteristics**<br>• Deletion of pull-down resistor ($R_{PLD1}$)<br>• Modification of supply current ($I_{DD1}$, $I_{DD2}$, $I_{DD3}$)<br>• Modification of real-time counter operating current ($I_{RTC}$), TMH1 operating current ($I_{TMH}$), A/D converter operating current ($I_{ADC}$), and operational amplifier operating current ($I_{AMP}$) | |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(5/8)

| Edition | Description | Chapter |
|---|---|---|
| 2nd Edition | Modification of **Caution** in **Figure 17-14 Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (78K0/KB2-L)** to **Figure 17-16 Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L, PR1H) (48-pin products of 78K0/KC2-L)** | **CHAPTER 28 ELECTRICAL SPECIFICATIONS (TARGET VALUES)** |
| | • Addition of reset current ($I_{DDrst}$) | |
| | **(1) A/D Converter** in **Analog Characteristics**<br>• Modification of conversion time ($t_{CONV}$) in **<1> ANI0 to ANI7**<br>• Addition of **<2> ANI8 to ANI10 (78K0/KB2-L and 78K0/KC2-L only)** | |
| | **(3) Operational amplifier 0** in **Analog Characteristics**<br>• Modification of $V_{DD}$ range<br>• Addition of phase margin and large-amplitude voltage gain ($AV_{OP0}$)<br>• Modification of gain-bandwidth product ($GBW_{OP0}$) | |
| | **(4) Operational amplifier 1** in **Analog Characteristics**<br>• Addition of phase margin and large-amplitude voltage gain ($AV_{OP1}$)<br>• Modification of gain-bandwidth product ($GBW_{OP1}$) | |
| | **(7) LVI** in **Analog Characteristics**<br>• Addition of supply voltage level ($V_{LVI14}$) and supply voltage when power supply voltage is turned on ($V_{DDLVI}$) | |
| | **Flash Memory Programming Characteristics**<br>• Modification of $V_{DD}$ range<br>• Modification of system clock frequency ($f_{CLK}$)<br>• Modification of Number of rewrites per chip ($C_{erwr}$)<br>• Addition of **Note 1** | |
| | Modification of **29.1 78K0/KY2-L** | **CHAPTER 29 PACKAGE DRAWINGS** |
| | Addition of preliminary | **CHAPTER 30 RECOMMENDED SOLDERING CONDITIONS (PRELIMINARY)** |
| | Modification of URL of download site for development tools | **APPENDIX A DEVELOPMENT TOOLS** |
| | Addition of chapter | **APPENDIX B REGISTER INDEX** |
| | Addition of chapter | **APPENDIX C REVISION HISTORY** |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(6/8)

| Edition | Description | Chapter |
|---|---|---|
| 3rd Edition | Modification of **Related Documents** | **INTRODUCTION** |
| | Modification of description in **1.1 Features** | **CHAPTER 1 OUTLINE** |
| | Modification of **Caution 1** in **1.3.3 78K0/KB2-L** and **1.3.4 78K0/KC2-L** | |
| | Modification of **Caution 1** in **1.4.3 78K0/KB2-L** and **1.4.4 78K0/KC2-L** | |
| | Modification of description in **1.5 Outline of Functions** | |
| | Modification of **Table 3-6 Special Function Register List: 78K0/KY2-L** to **Table 3-9 Special Function Register List: 78K0/KC2-L** | **CHAPTER 3 CPU ARCHITECTURE** |
| | Modification of **Table 4-10 Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin** | **CHAPTER 4 PORT FUNCTIONS** |
| | Addition of description to **4.3 (5) Port output mode register 6 (POM6)** | |
| | Addition of **Caution 5** to **Table 4-12 Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KY2-L) (1/2)** | |
| | Addition of **Caution 5** to **Table 4-13 Settings of Port Mode Register and Output Latch When Using Alternate Function (78K0/KA2-L) (1/2)** | |
| | Addition of **Caution 1** to **Figure 5-8 Format of Main OSC Control Register (MOC)** | **CHAPTER 5 CLOCK GENERATOR** |
| | Addition of **Caution 1** to **Figure 5-12 Format of Peripheral Enable Register 0 (PER0)** | |
| | Modification of **Figure 5-16 Clock Generator Operation When Power Supply Voltage Is Turned On, (When LVI Default Start Function Stopped Is Set (Option Byte: LVISTART = 0))** and **Figure 5-17 Clock Generator Operation When Power Supply Voltage Is Turned On (When LVI Default Start Function Enabled Is Set (Option Byte: LVISTART = 1))** | |
| | Modification of **Figure 7-2 Block Diagram of 8-Bit Timer 51 (78K0/KY2-L, 78K0/KA2-L)** | **CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50 AND 51** |
| | Addition of the port mode register 4 (PM4) and the port register 4 (P4) to **Table 10-1 Configuration of Real-Time Counter** | **CHAPTER 10 REAL-TIME COUNTER** |
| | Addition of **Caution 1** to **Figure 10-2 Format of Peripheral Enable Register 0 (PER0)** | |
| | Addition of the port mode register 4 (PM4) and the port register 4 (P4) to **10.3 Registers Controlling Real-Time Counter** | |
| | Modification of **Figure 10-3 Format of Real-Time Counter Control Register 0 (RTCC0)** | |
| | Modification of **Figure 10-4 Format of Real-Time Counter Control Register 0 (RTCC1)** | |
| | Modification of description in **(7) Minute count register (MIN), (8) Hour count register (HOUR), (9) Day count register (DAY), (11) Month count register (MONTH), (12) Year count register (YEAR)** | |
| | Modification of **Figure 10-14 Format of Watch Error Correction Register (SUBCUD)** | |
| | Modification of **Figure 10-19 Procedure for Starting Operation of Real-Time Counter** | |
| | Addition of **10.4.2 Shifting to STOP mode after starting operation** | |
| | Modification of **Figure 10-26 512 Hz, 16.384 kHz output Setting Procedure** | |
| | Modification of **(2) 2.7 V ≤ $AV_{REF}$ < 4.0 V** in **Table 12-2 A/D Conversion Time Selection** | **CHAPTER 12 A/D CONVERTER** |
| | Modification of **Caution 3** in **Figure 12-8 Format of Analog Input Channel Specification Register (ADS)** | |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(7/8)

| Edition | Description | Chapter |
|---|---|---|
| 3rd Edition | Modification of **Table 12-6 Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin** | **CHAPTER 12 A/D CONVERTER** |
| | Modification of <4> in **12.4.1 Basic operations of A/D converter** | |
| | Modification of **Table 12-8 Resistance and Capacitance Values of Equivalent Circuit (Reference Values)** | |
| | Addition of the analog input channel specification register (ADS) to **Table 13-1 Configuration of Operational Amplifier** | **CHAPTER 13 OPERATIONAL AMPLIFIERS** |
| | Addition of **(3) Analog input channel specification register (ADS)** to **13.3 Registers Used in Operational Amplifier** | |
| | Modification of **Table 13-5 Setting Functions of P21/ANI1/AMP0OUT/PGAIN Pin** | |
| | Addition of the port output mode register 6 (POM6) to **Table 14-1 Configuration of Serial Interface UART6** | **CHAPTER 14 SERIAL INTERFACE UART6** |
| | Addition of **(9) Port output mode register 6 (POM6)** to **14.3 Registers Controlling Serial Interface UART6** | |
| | Addition of the port output mode register 6 (POM6) to **14.4.2 (1) Registers used** | |
| | Modification of **(1) 78K0/KY2-L and 78K0/KA2-L** in **Table 14-2 Relationship Between Register Settings and Pins** | |
| | Addition of **Caution 3** to **Figure 15-3 Format of IICA Shift Register (IICA)** | **CHAPTER 15 SERIAL INTERFACE IICA** |
| | Addition of **Note 3** to, and modification of **Caution** in **Figure 15-5 Format of IICA Control Register 0 (IICACTL0) (1/4)** | |
| | Addition of description of the SPIE0 bit to **Figure 15-5 Format of IICA Control Register 0 (IICACTL0) (2/4)** | |
| | Modification of description of the STT0 bit in **Figure 15-5 Format of IICA Control Register 0 (IICACTL0) (3/4)** | |
| | Modification of **Caution** in **Figure 15-5 Format of IICA Control Register 0 (IICACTL0) (4/4)** | |
| | Modification of **Figure 15-6 Format of IICA Status Register 0 (IICAS0) (2/3)** | |
| | Partial deletion of description in **15.3 (9) Port mode register 6 (PM6)** | |
| | Modification of **15.4.2 Setting transfer clock by using IICWL and IICWH registers** | |
| | Modification of **(C) External maskable interrupt (INTKR)** in **Figure 17-1 Basic Configuration of Interrupt Function** | **CHAPTER 17 INTERRUPT FUNCTIONS** |
| | Addition of **Note** to **19.2.1 (2) (b) Release by reset signal generation** | **CHAPTER 19 STANDBY FUNCTION** |
| | Modification of **Figure 19-4 HALT Mode Release by Reset** | |
| | Addition of **Note 1** to **Figure 19-5 Operation Timing When STOP Mode Is Released (When Unmasked Interrupt Request Is Generated)** | |
| | Addition of **Note 2** to **(3) When internal high-speed oscillation clock is used as CPU clock** in **Figure 19-6 STOP Mode Release by Interrupt Request Generation** | |
| | Addition of **Note** to **19.2.2 (2) (b) Release by reset signal generation** | |
| | Modification of **Figure 19-7 STOP Mode Release by Reset** | |
| | Modification of **Figure 20-1 Block Diagram of Reset Function** to **Figure 20-4 Timing of Reset in STOP Mode by $\overline{\text{RESET}}$ Input** | **CHAPTER 20 RESET FUNCTION** |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

(8/8)

| Edition | Description | Chapter |
|---|---|---|
| 3rd Edition | Modification of **Figure 21-2 Timing of Generation of Internal Reset Signal by Power-on-Clear Circuit and Low-Voltage Detector** | **CHAPTER 21 POWER-ON-CLEAR CIRCUIT** |
| | Modification of **23.1 Regulator Overview** | **CHAPTER 23 REGULATOR** |
| | Addition of **23.3 Cautions for Self Programming** | |
| | Modification of **Figure 25-2 Environment for Writing Program to Flash Memory** | **CHAPTER 25 FLASH MEMORY** |
| | Modification of **Table 25-2 Pin Connection** | |
| | Modification of **25.4.1 TOOL pins** | |
| | Addition of **25.4.7 On-board writing when connecting crystal/ceramic resonator** | |
| | Modification of **25.5.2 Flash memory programming mode** | |
| | Addition of **25.7 Processing Time for Each Command When PG-FP5 Is Used (Reference)** | |
| | Modification of **Cautions 3** to **5** and **Remark** in **25.8 Flash Memory Programming by Self Programming** | |
| | Addition of **25.8.1 Register controlling self programming mode** and **25.8.2 Flow of self programming (Rewriting Flash Memory)** | |
| | Modification of **Caution** in **25.8.3 Boot swap function** | |
| | Addition of **25.9 Creating ROM Code to Place Order for Previously Written Product** | |
| | Modification of **Figure 26-1 Connection Example of QB-MINI2 and 78K0/Kx2-L Microcontrollers** | **CHAPTER 26 ON-CHIP DEBUG FUNCTION** |
| | Modification of **26.2 On-Chip Debug Security ID** | |
| | Addition of **26.3 Securing of User Resources** | |
| | Revision of chapter | **CHAPTER 28 ELECTRICAL SPECIFICATIONS** |
| | Modification of **Table 30-1 Surface Mounting Type Soldering Conditions** | **CHAPTER 30 RECOMMENDED SOLDERING CONDITIONS** |
| | Under development → Under mass production<br>• QB-78K0KX2-L | **APPENDIX A DEVELOPMENT TOOLS** |
| | Modification of **Notes 3** and **4** in **Figure A-1 Development Tool Configuration (1/2)** | |
| | Modification of **Note 5** in **Figure A-1 Development Tool Configuration (2/2)** | |
| | Addition of **Note 1** to **A.3.1 When using flash memory programmer PG-FP5 and FL-PR5** | |
| | Addition of the product name of system simulator to **A.5 Debugging Tools (Software)** | |
| | Addition of **C.2 Revision History of Preceding Editions** | **APPENDIX C REVISION HISTORY** |

**Remark** "Classification" in the above table classifies revisions as follows.
(a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note, (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

78K0/Kx2-L User's Manual: Hardware

Publication Date: Rev.0.01 Jul 9, 2008
Rev.4.00 Sep 27, 2010

Published by: Renesas Electronics Corporation

**SALES OFFICES** Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

**Renesas Electronics America Inc.**
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

**Renesas Electronics Canada Limited**
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

**Renesas Electronics Europe Limited**
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

**Renesas Electronics Europe GmbH**
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

**Renesas Electronics (Shanghai) Co., Ltd.**
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

**Renesas Electronics Hong Kong Limited**
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

**Renesas Electronics Taiwan Co., Ltd.**
7F, No. 363 Fu Shing North Road Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

**Renesas Electronics Singapore Pte. Ltd.**
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

**Renesas Electronics Malaysia Sdn.Bhd.**
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

**Renesas Electronics Korea Co., Ltd.**
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141



© 2010 Renesas Electronics Corporation. All rights reserved.


Colophon 1.0

78K0/Kx2-L

Renesas Electronics Corporation

R01UH0028EJ0400
(Previous Number: U19111EJ3V0UD00)