

## $\mu$ PD780058, 780058Y Subseries

### 8-Bit Single-Chip Microcontrollers

---

$\mu$ PD780053	$\mu$ PD780053Y
$\mu$ PD780054	$\mu$ PD780054Y
$\mu$ PD780055	$\mu$ PD780055Y
$\mu$ PD780056	$\mu$ PD780056Y
$\mu$ PD780058	$\mu$ PD780058BY
$\mu$ PD780058B	$\mu$ PD78F0058Y
$\mu$ PD78F0058	$\mu$ PD780053Y(A)
$\mu$ PD780053(A)	$\mu$ PD780054Y(A)
$\mu$ PD780054(A)	$\mu$ PD780055Y(A)
$\mu$ PD780055(A)	$\mu$ PD780056Y(A)
$\mu$ PD780056(A)	$\mu$ PD780058BY(A)
$\mu$ PD780058B(A)	

[MEMO]

## NOTES FOR CMOS DEVICES

### ① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note:

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 once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build 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 bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

### ② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note:

No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to  $V_{DD}$  or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

### ③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note:

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

**FIP, EEPROM, and IEBus are trademarks of NEC Electronics Corporation.**

**Windows and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.**

**PC/AT is a trademark of International Business Machines Corporation.**

**HP9000 Series 700 and HP-UX are trademarks of Hewlett-Packard Company.**

**SPARCstation is a trademark of SPARC International, Inc.**

**SunOS is a trademark of Sun Microsystems, Inc.**

**TRON stands for The Realtime Operating system Nucleus.**

**ITRON is an abbreviation of Industrial TRON.**

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

Purchase of NEC Electronics I<sup>2</sup>C components conveys a license under the Philips I<sup>2</sup>C Patent Rights to use these components in an I<sup>2</sup>C system, provided that the system conforms to the I<sup>2</sup>C Standard Specification as defined by Philips.

**• The information in this document is current as of January, 2003. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**

- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".

The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.

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

"Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).

"Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

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

(Note)

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

# Regional Information

Some information contained in this document may vary from country to country. Before using any NEC Electronics product in your application, please contact the NEC Electronics office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

## **NEC Electronics America, Inc. (U.S.)**

Santa Clara, California  
Tel: 408-588-6000  
800-366-9782  
Fax: 408-588-6130  
800-729-9288

## **NEC Electronics (Europe) GmbH**

Duesseldorf, Germany  
Tel: 0211-65 03 01  
Fax: 0211-65 03 327

### **• Sucursal en España**

Madrid, Spain  
Tel: 091-504 27 87  
Fax: 091-504 28 60

### **• Succursale Française**

Vélizy-Villacoublay, France  
Tel: 01-30-67 58 00  
Fax: 01-30-67 58 99

### **• Filiale Italiana**

Milano, Italy  
Tel: 02-66 75 41  
Fax: 02-66 75 42 99

### **• Branch The Netherlands**

Eindhoven, The Netherlands  
Tel: 040-244 58 45  
Fax: 040-244 45 80

### **• Tyskland Filial**

Taeby, Sweden  
Tel: 08-63 80 820  
Fax: 08-63 80 388

### **• United Kingdom Branch**

Milton Keynes, UK  
Tel: 01908-691-133  
Fax: 01908-670-290

## **NEC Electronics Hong Kong Ltd.**

Hong Kong  
Tel: 2886-9318  
Fax: 2886-9022/9044

## **NEC Electronics Hong Kong Ltd.**

Seoul Branch  
Seoul, Korea  
Tel: 02-528-0303  
Fax: 02-528-4411

## **NEC Electronics Shanghai, Ltd.**

Shanghai, P.R. China  
Tel: 021-6841-1138  
Fax: 021-6841-1137

## **NEC Electronics Taiwan Ltd.**

Taipei, Taiwan  
Tel: 02-2719-2377  
Fax: 02-2719-5951

## **NEC Electronics Singapore Pte. Ltd.**

Novena Square, Singapore  
Tel: 6253-8311  
Fax: 6250-3583

## Major Revisions in This Edition (1/2)

Page	Description
Throughout	Deletion of following product <ul style="list-style-type: none"> <li>• <math>\mu</math>PD780058Y</li> </ul>
	Addition of following products <ul style="list-style-type: none"> <li>• <math>\mu</math>PD780058B, 780058BY, 780053(A), 780053Y(A), 780054(A), 780054Y(A), 780055(A), 780055Y(A), 780056(A), 780056Y(A), 780058B(A), 780058BY(A)</li> </ul>
	Deletion of following packages <ul style="list-style-type: none"> <li>• 80-pin plastic QFP (GC-3B9 type)</li> <li>• 80-pin plastic TQFP (GK-BE9 type)</li> </ul>
	Addition of following package <ul style="list-style-type: none"> <li>• 80-pin plastic TQFP (GK-9EU type)</li> </ul>
pp. 31, 32, 38, 39	<b>1.1 Features, 1.7 Outline of Functions</b> <ul style="list-style-type: none"> <li>• Change of operating voltage range of A/D and D/A converters of <math>\mu</math>PD780058 and 78F0058</li> <li>• Change of supply voltage of <math>\mu</math>PD78F0058</li> </ul>
p. 40	Addition of <b>1.9 Differences Between Standard Model and (A) Model</b>
pp. 41, 42, 48, 49	<b>2.1 Features, 2.7 Outline of Functions</b> <ul style="list-style-type: none"> <li>• Change of operating voltage range of A/D and D/A converters of <math>\mu</math>PD78F0058Y</li> <li>• Change of supply voltage of <math>\mu</math>PD78F0058Y</li> </ul>
p. 50	Addition of <b>2.9 Differences Between Standard Model and (A) Model</b>
p. 60	Change of processing when A/D converter is not used in <b>3.2.11 AV<sub>REF0</sub></b>
pp. 62, 63	Change of recommended connection of unused pins and connection of P60 to P63, AV <sub>REF1</sub> , and V <sub>PP</sub> pins in <b>Table 3-1 Pin I/O Circuit Types</b>
p. 75	Change of processing when A/D converter is not used in <b>4.2.11 AV<sub>REF0</sub></b>
pp. 77, 78	Change of recommended connection of unused pins and connection of P60 to P63, AV <sub>REF1</sub> , and V <sub>PP</sub> pins in <b>Table 4-1 Pin I/O Circuit Types</b>
p. 132	Modification of Note 2 in <b>6.2.8 Port 6</b>
p. 149	Addition of note on feedback resistor to <b>Figure 7-3 Processor Clock Control Register Format</b>
p. 167	Addition of <b>Table 8-5 INTP1/TI01 Pin Valid Edge and CR00 Capture Trigger Valid Edge</b>
p. 168	Addition of <b>Table 8-6 INTP0/TI00 Pin Valid Edge and CR01 Capture Trigger Valid Edge</b>
p. 177	Correction of note on valid edge of INTP0/TI00/P00 and INTP1/TI01/P01 pin in <b>Figure 8-8 Format of External Interrupt Mode Register 0</b>
p. 185	Addition of <b>Figure 8-17 Configuration of PPG Output</b> Addition of <b>Figure 8-18 PPG Output Operation Timing</b>
pp. 201 to 204	<b>8.5 16-Bit Timer/Event Counter Operating Cautions</b> Addition of description on TI01/P01/INTP1 to <b>(5) Valid edge setting</b> Addition of <b>(c) One-shot pulse output function to (6) Re-trigger of one-shot pulse</b> Addition of <b>(8) Conflict operation</b> Addition of <b>(9) Timer operation</b> Addition of <b>(10) Capture operation</b> Addition of <b>(11) Compare operation</b> Addition of <b>(12) Edge detection</b>
p. 235	Modification of note on changing count clock in <b>Figure 10-2 Timer Clock Select Register 2 Format</b>
p. 242	Modification of note on changing count clock in <b>Figure 11-2 Timer Clock Select Register 2 Format</b>
p. 252	Addition of note on rewriting TCL2 in <b>Figure 13-2 Format of Timer Clock Select Register 2</b>

## Major Revisions in This Edition (2/2)

Page	Description
p. 263	Modification of <b>Figure 14-5 A/D Converter Basic Operation</b> Addition of <b>Table 14-2 A/D Conversion Sampling Time and A/D Converter Start Delay Time</b>
pp. 267, 268	Addition of <b>14.5 How to Read A/D Converter Characteristics Table</b>
pp. 269, 270, 272, 273	<b>14.6 A/D Converter Cautions</b> Change of description in <b>(1) Power consumption in standby mode</b> Addition of <b>(3) Conflict operations</b> Addition of <b>(6) Input impedance of ANI0 to ANI7 pins</b> Addition of <b>(10) Timing at which A/D conversion result is undefined</b> Addition of <b>(11) Notes on board design</b> Addition of <b>(12) AV<sub>REF0</sub> pin</b> Addition of <b>(13) Internal equivalent circuit of ANI0 to ANI7 pins and permissible signal source impedance</b>
p. 280	Addition of description of processing when D/A converter is not used in <b>15.5 D/A Converter Cautions (3) AV<sub>REF1</sub> pin</b>
p. 379	Addition of <b>17.4.7 Restrictions in I<sup>2</sup>C bus mode 2</b>
p. 468	Addition of <b>19.4.5 Restrictions in UART mode 2</b>
p. 477	Addition of Caution when interrupt is acknowledged to <b>Figure 21-2 Interrupt Request Flag Register Format</b>
p. 483	Addition of description on TI01/P01/INTP1 pin to <b>Figure 21-5 Format of External Interrupt Mode Register 0</b>
p. 525	Addition of Caution to <b>25.1 ROM Correction Function</b>
p. 535	Modification of <b>Table 26-1 Differences Between <math>\mu</math>PD78F0058, 78F0058Y and Mask ROM Versions</b>
pp. 538 to 549	Total revision of description on flash memory programming as <b>26.3 Flash Memory Characteristics</b>
pp. 567 to 596	Addition of <b>CHAPTER 28 ELECTRICAL SPECIFICATIONS (MASK ROM VERSION)</b>
pp. 597 to 626	Addition of <b>CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)</b>
pp. 627 to 657	Addition of <b>CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION (V<sub>DD</sub> = 2.2 V))</b>
pp. 658, 659	Addition of <b>CHAPTER 31 CHARACTERISTICS CURVES (REFERENCE VALUES)</b>
pp. 660, 661	Addition of <b>CHAPTER 32 PACKAGE DRAWINGS</b>
pp. 662 to 665	Addition of <b>CHAPTER 33 RECOMMENDED SOLDERING CONDITIONS</b>
pp. 666, 667	Correction of <b>APPENDIX A DIFFERENCES BETWEEN <math>\mu</math>PD78054, 78058F, AND 780058 SUBSERIES</b>
pp. 668 to 684	Total revision of <b>APPENDIX B DEVELOPMENT TOOLS</b> Transfer of description of embedded software to <b>APPENDIX B DEVELOPMENT TOOLS</b>

The mark ★ shows major revised points.

## PREFACE

### Readers

This manual has been prepared for user engineers who wish to understand the functions of the  $\mu$ PD780058 and 780058Y Subseries and design and develop its application systems and programs.

This manual is intended for the products in the following subseries.

- $\mu$ PD780058 Subseries  
 $\mu$ PD780053, 780054, 780055, 780056, 780058, 780058B, 78F0058, 780053(A), 780054(A), 780055(A), 780056(A), 780058B(A)
- $\mu$ PD780058Y Subseries  
 $\mu$ PD780053Y, 780054Y, 780055Y, 780056Y, 780058BY, 78F0058Y, 780053Y(A), 780054Y(A), 780055Y(A), 780056Y(A), 780058BY(A)

These products are collectively referred to as the “ $\mu$ PD780058, 780058Y Subseries” in this manual.

### Purpose

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

### Organization

The  $\mu$ PD780058, 780058Y Subseries manual is separated into two parts: this manual and the instruction edition (common to the 78K/0 Series).

$\mu$ PD780058, 780058Y  
Subseries  
User's Manual  
(This manual)

- Pin functions
- Internal block functions
- Interrupts
- Other on-chip peripheral functions
- Electrical specifications

78K/0 Series  
User's Manual  
Instructions

- CPU functions
- Instruction set
- Explanation of each instruction



## How to Read This Manual

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

- When using this manual as the manual for the  $\mu$ PD780053(A), 780054(A), 780055(A), 780056(A), 780058B(A), 780053Y(A), 780054Y(A), 780055Y(A), 780056Y(A), and 780058BY(A),

→ The only difference between these products and the  $\mu$ PD780053, 780054, 780055, 780056, 780058B, 780053Y, 780054Y, 780055Y, 780056Y, and 780058BY is the quality grade (see **1.9 Differences Between Standard Model and (A) Model**, and **2.9 Differences Between Standard Model and (A) Model**). The correspondence between the standard model and (A) model is as follows in **CHAPTER 6 PORT FUNCTIONS to CHAPTER 27 INSTRUCTION SET OUTLINE**.

$\mu$ PD780053	→ $\mu$ PD780053(A)	$\mu$ PD780053Y	→ $\mu$ PD780053Y(A)
$\mu$ PD780054	→ $\mu$ PD780054(A)	$\mu$ PD780054Y	→ $\mu$ PD780054Y(A)
$\mu$ PD780055	→ $\mu$ PD780055(A)	$\mu$ PD780055Y	→ $\mu$ PD780055Y(A)
$\mu$ PD780056	→ $\mu$ PD780056(A)	$\mu$ PD780056Y	→ $\mu$ PD780056Y(A)
$\mu$ PD780058B	→ $\mu$ PD780058B(A)	$\mu$ PD780058BY	→ $\mu$ PD780058BY(A)

- To gain a general understanding the functions:
  - Read this manual in the order of the contents.
- To know the  $\mu$ PD780058 and 780058Y Subseries instruction functions in detail:
  - Refer to the **78K/0 Series Instructions User's Manual (U12326E)**
- 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 defined in the header file named sfrbit.h in the CC78K0.
- To learn the function of a register whose register name is known:
  - Refer to **APPENDIX C REGISTER INDEX**.
- To see application examples of each function of the  $\mu$ PD780058, 780058Y Subseries:
  - Refer to **78K/0 Series Basics (III) Application Note (U10182E)** separately available.
- To understand the electrical specifications of the  $\mu$ PD780058, 780058Y Subseries:
  - See **CHAPTER 28 ELECTRICAL SPECIFICATIONS (MASK ROM VERSION)**, **CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)**, **CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION (V<sub>DD</sub> = 2.2 V))**.

**Caution** Examples in this manual employ the “standard” quality grade for general electronics. When using examples in this manual for the “special” quality grade, review the quality grade of each part and/or circuit actually used.

**Chapter Organization:** This manual divides the descriptions for the  $\mu$ PD780058 and 780058Y Subseries into different chapters as shown below. Read only the chapters related to the device being used.

Chapter	$\mu$ PD780058 Subseries	$\mu$ PD780058Y Subseries
CHAPTER 1 Outline ( $\mu$ PD780058 Subseries)	√	—
CHAPTER 2 Outline ( $\mu$ PD780058Y Subseries)	—	√
CHAPTER 3 Pin Functions ( $\mu$ PD780058 Subseries)	√	—
CHAPTER 4 Pin Functions ( $\mu$ PD780058Y Subseries)	—	√
CHAPTER 5 CPU Architecture	√	√
CHAPTER 6 Port Functions	√	√
CHAPTER 7 Clock Generator	√	√
CHAPTER 8 16-Bit Timer/Event Counter	√	√
CHAPTER 9 8-Bit Timer/Event Counter	√	√
CHAPTER 10 Watch Timer	√	√
CHAPTER 11 Watchdog Timer	√	√
CHAPTER 12 Clock Output Controller	√	√
CHAPTER 13 Buzzer Output Controller	√	√
CHAPTER 14 A/D Converter	√	√
CHAPTER 15 D/A Converter	√	√
CHAPTER 16 Serial Interface Channel 0 ( $\mu$ PD780058 Subseries)	√	—
CHAPTER 17 Serial Interface Channel 0 ( $\mu$ PD780058Y Subseries)	—	√
CHAPTER 18 Serial Interface Channel 1	√	√
CHAPTER 19 Serial Interface Channel 2	√	√
CHAPTER 20 Real-Time Output Port	√	√
CHAPTER 21 Interrupt and Test Functions	√	√
CHAPTER 22 External Device Expansion Function	√	√
CHAPTER 23 Standby Function	√	√
CHAPTER 24 Reset Function	√	√
CHAPTER 25 ROM Correction	√	√
CHAPTER 26 $\mu$ PD78F0058, $\mu$ PD78F0058Y	√	√
CHAPTER 27 Outline of Instruction Set	√	√
CHAPTER 28 ELECTRICAL SPECIFICATIONS (MASK ROM VERSION)	√	√
CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)	√	√
CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION ( $V_{DD} = 2.2$ V))	√	√
CHAPTER 31 CHARACTERISTICS CURVES (REFERENCE VALUES)	√	√
CHAPTER 32 PACKAGE DRAWINGS	√	√
CHAPTER 33 RECOMMENDED SOLDERING CONDITIONS	√	√

**Differences Between  $\mu$ PD780058 and  $\mu$ PD780058Y Subseries:**

The  $\mu$ PD780058 and  $\mu$ PD780058Y Subseries differ in the following functions of serial interface channel 0.

Modes of serial interface channel 0	$\mu$ PD780058 Subseries	$\mu$ PD780058Y Subseries
3-wire serial I/O mode	√	√
2-wire serial I/O mode	√	√
SBI (serial bus interface) mode	√	—
I <sup>2</sup> C (inter IC) bus mode	—	√

√: Supported

—: Not supported

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

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

★ **Documents Related to Devices**

Document Name	Document No.
$\mu$ PD780058, 780058Y Subseries User's Manual	This manual
78K/0 Series Instruction User's Manual	U12326E
78K/0 Series Basics (III) Application Note	U10182E

**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 Software Development Tools (User's Manuals)**

Document Name		Document No.
RA78K0 Assembler Package	Operation	U14445E
	Language	U14446E
	Structure Assembly Language	U11789E
CC78K0 C Compiler	Operation	U14297E
	Language	U14298E
SM78K Series System Simulator Ver. 2.30 or Later	Operation (Windows™ Based)	U15373E
	External Part User Open Interface Specification	U15802E
ID78K Series Integrated Debugger Ver. 2.30 or Later	Operation (Windows Based)	U15185E
RX78K0 Real-Time OS	Fundamentals	U11537E
	Installation	U11536E
Project Manager Ver. 3.12 or Later (Windows Based)		U14610E

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

Document Name	Document No.
IE-78K0-NS In-Circuit Emulator	U13731E
IE-78K0-NS-A In-Circuit Emulator	U14889E
IE-780308-NS-EM1 Emulation Board	U13304E
IE-78001-R-A In-Circuit Emulator	U14142E
IE-780308-R-EM Emulation Board	U11362E

**Documents Related to Flash Memory Writing**

Document Name	Document No.
PG-FP3 Flash Memory Programmer User's Manual	U13502E
PG-FP4 Flash Memory Programmer User's Manual	U15260E

**Other Related Documents**

	Document Name	Document No.
★	SEMICONDUCTOR SELECTION GUIDE - Products and Packages -	X13769X
★	Semiconductor Device Mounting Technology Manual	C10535E
	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

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

## CONTENTS

	<b>CHAPTER 1 OUTLINE (<math>\mu</math>PD780058 SUBSERIES) .....</b>	<b>31</b>
	<b>1.1 Features .....</b>	<b>31</b>
	<b>1.2 Applications .....</b>	<b>32</b>
	<b>1.3 Ordering Information .....</b>	<b>32</b>
	<b>1.4 Pin Configuration (Top View) .....</b>	<b>33</b>
	<b>1.5 78K/0 Series Lineup .....</b>	<b>35</b>
	<b>1.6 Block Diagram .....</b>	<b>37</b>
	<b>1.7 Outline of Function .....</b>	<b>38</b>
	<b>1.8 Mask Options .....</b>	<b>40</b>
★	<b>1.9 Differences Between Standard Model and (A) Model .....</b>	<b>40</b>
	<b>CHAPTER 2 OUTLINE (<math>\mu</math>PD780058Y SUBSERIES) .....</b>	<b>41</b>
	<b>2.1 Features .....</b>	<b>41</b>
	<b>2.2 Applications .....</b>	<b>42</b>
	<b>2.3 Ordering Information .....</b>	<b>42</b>
	<b>2.4 Pin Configuration (Top View) .....</b>	<b>43</b>
	<b>2.5 78K/0 Series Lineup .....</b>	<b>45</b>
	<b>2.6 Block Diagram .....</b>	<b>47</b>
	<b>2.7 Outline of Functions .....</b>	<b>48</b>
	<b>2.8 Mask Options .....</b>	<b>50</b>
★	<b>2.9 Differences Between Standard Model and (A) Model .....</b>	<b>50</b>
	<b>CHAPTER 3 PIN FUNCTIONS (<math>\mu</math>PD780058 SUBSERIES) .....</b>	<b>51</b>
	<b>3.1 Pin Function List .....</b>	<b>51</b>
	<b>3.2 Description of Pin Functions .....</b>	<b>55</b>
	3.2.1 P00 to P05, P07 (Port 0) .....	55
	3.2.2 P10 to P17 (Port 1) .....	55
	3.2.3 P20 to P27 (Port 2) .....	56
	3.2.4 P30 to P37 (Port 3) .....	57
	3.2.5 P40 to P47 (Port 4) .....	57
	3.2.6 P50 to P57 (Port 5) .....	58
	3.2.7 P60 to P67 (Port 6) .....	58
	3.2.8 P70 to P72 (Port 7) .....	59
	3.2.9 P120 to P127 (Port 12) .....	59
	3.2.10 P130 and P131 (Port 13) .....	60
	3.2.11 AV <sub>REF0</sub> .....	60
	3.2.12 AV <sub>REF1</sub> .....	60
	3.2.13 AV <sub>SS</sub> .....	60
	3.2.14 $\overline{\text{RESET}}$ .....	60
	3.2.15 X1 and X2 .....	60
	3.2.16 XT1 and XT2 .....	60
	3.2.17 V <sub>DD0</sub> , V <sub>DD1</sub> .....	60
	3.2.18 V <sub>SS0</sub> , V <sub>SS1</sub> .....	61
	3.2.19 V <sub>PP</sub> (Flash memory version only) .....	61
	3.2.20 IC (Mask ROM version only) .....	61
	<b>3.3 I/O Circuits and Recommended Connection of Unused Pins .....</b>	<b>62</b>

<b>CHAPTER 4 PIN FUNCTIONS (<math>\mu</math>PD780058Y SUBSERIES)</b> .....	<b>66</b>
<b>4.1 Pin Function List</b> .....	<b>66</b>
<b>4.2 Description of Pin Functions</b> .....	<b>70</b>
4.2.1 P00 to P05, P07 (Port 0) .....	70
4.2.2 P10 to P17 (Port 1) .....	71
4.2.3 P20 to P27 (Port 2) .....	71
4.2.4 P30 to P37 (Port 3) .....	72
4.2.5 P40 to P47 (Port 4) .....	73
4.2.6 P50 to P57 (Port 5) .....	73
4.2.7 P60 to P67 (Port 6) .....	73
4.2.8 P70 to P72 (Port 7) .....	74
4.2.9 P120 to P127 (Port 12) .....	74
4.2.10 P130 and P131 (Port 13) .....	75
4.2.11 $AV_{REF0}$ .....	75
4.2.12 $AV_{REF1}$ .....	75
4.2.13 $AV_{SS}$ .....	75
4.2.14 $\overline{RESET}$ .....	75
4.2.15 X1 and X2 .....	75
4.2.16 XT1 and XT2 .....	75
4.2.17 $V_{DD0}$ , $V_{DD1}$ .....	75
4.2.18 $V_{SS0}$ , $V_{SS1}$ .....	76
4.2.19 $V_{PP}$ (Flash memory version only) .....	76
4.2.20 IC (Mask ROM version only) .....	76
<b>4.3 I/O Circuits and Recommended Connection of Unused Pins</b> .....	<b>77</b>
<b>CHAPTER 5 CPU ARCHITECTURE</b> .....	<b>81</b>
<b>5.1 Memory Spaces</b> .....	<b>81</b>
5.1.1 Internal program memory space .....	87
5.1.2 Internal data memory space .....	89
5.1.3 Special Function Register (SFR) area .....	89
5.1.4 External memory space .....	89
5.1.5 Data memory addressing .....	89
<b>5.2 Processor Registers</b> .....	<b>96</b>
5.2.1 Control registers .....	96
5.2.2 General registers .....	99
5.2.3 Special-Function Registers (SFRs) .....	100
<b>5.3 Instruction Address Addressing</b> .....	<b>104</b>
5.3.1 Relative addressing .....	104
5.3.2 Immediate addressing .....	105
5.3.3 Table indirect addressing .....	106
5.3.4 Register addressing .....	107
<b>5.4 Operand Address Addressing</b> .....	<b>108</b>
5.4.1 Implied addressing .....	108
5.4.2 Register addressing .....	109
5.4.3 Direct addressing .....	110
5.4.4 Short direct addressing .....	111
5.4.5 Special-Function Register (SFR) addressing .....	113
5.4.6 Register indirect addressing .....	114
5.4.7 Based addressing .....	115

5.4.8 Based indexed addressing .....	116
5.4.9 Stack addressing .....	116
<b>CHAPTER 6 PORT FUNCTIONS .....</b>	<b>117</b>
<b>6.1 Port Functions .....</b>	<b>117</b>
<b>6.2 Port Configuration .....</b>	<b>122</b>
6.2.1 Port 0 .....	122
6.2.2 Port 1 .....	124
6.2.3 Port 2 ( $\mu$ PD780058 Subseries) .....	125
6.2.4 Port 2 ( $\mu$ PD780058Y Subseries) .....	127
6.2.5 Port 3 .....	129
6.2.6 Port 4 .....	130
6.2.7 Port 5 .....	131
6.2.8 Port 6 .....	132
6.2.9 Port 7 .....	134
6.2.10 Port 12 .....	136
6.2.11 Port 13 .....	137
<b>6.3 Port Function Control Registers .....</b>	<b>138</b>
<b>6.4 Port Operations .....</b>	<b>144</b>
6.4.1 Writing to I/O port .....	144
6.4.2 Reading from I/O port .....	144
6.4.3 Operations on I/O port .....	144
<b>6.5 Selection of Mask Option .....</b>	<b>145</b>
<b>CHAPTER 7 CLOCK GENERATOR .....</b>	<b>146</b>
<b>7.1 Clock Generator Functions .....</b>	<b>146</b>
<b>7.2 Clock Generator Configuration .....</b>	<b>146</b>
<b>7.3 Clock Generator Control Registers .....</b>	<b>148</b>
<b>7.4 System Clock Oscillator .....</b>	<b>152</b>
7.4.1 Main system clock oscillator .....	152
7.4.2 Subsystem clock oscillator .....	153
7.4.3 Example of resonator with bad connection .....	154
7.4.4 Divider .....	155
7.4.5 When not using subsystem clock .....	155
<b>7.5 Clock Generator Operations .....</b>	<b>156</b>
7.5.1 Main system clock operations .....	157
7.5.2 Subsystem clock operations .....	158
<b>7.6 Changing System Clock and CPU Clock Settings .....</b>	<b>159</b>
7.6.1 Time required for switchover between system clock and CPU clock .....	159
7.6.2 System clock and CPU clock switching procedure .....	161
<b>CHAPTER 8 16-BIT TIMER/EVENT COUNTER .....</b>	<b>162</b>
<b>8.1 16-Bit Timer/Event Counter Functions .....</b>	<b>162</b>
<b>8.2 16-Bit Timer/Event Counter Configuration .....</b>	<b>164</b>
<b>8.3 16-Bit Timer/Event Counter Control Registers .....</b>	<b>169</b>
<b>8.4 16-Bit Timer/Event Counter Operations .....</b>	<b>179</b>
8.4.1 Interval timer operations .....	179
8.4.2 PWM output operations .....	181
8.4.3 PPG output operations .....	184
8.4.4 Pulse width measurement operations .....	186

8.4.5 External event counter operation .....	193
8.4.6 Square-wave output operation .....	195
8.4.7 One-shot pulse output operation .....	197
<b>8.5 16-Bit Timer/Event Counter Operating Cautions .....</b>	<b>201</b>
<b>CHAPTER 9 8-BIT TIMER/EVENT COUNTER .....</b>	<b>205</b>
<b>9.1 8-Bit Timer/Event Counter Functions .....</b>	<b>205</b>
9.1.1 8-bit timer/event counter mode .....	205
9.1.2 16-bit timer/event counter mode .....	208
<b>9.2 8-Bit Timer/Event Counter Configuration .....</b>	<b>210</b>
<b>9.3 8-Bit Timer/Event Counter Control Registers .....</b>	<b>214</b>
<b>9.4 Operations of 8-Bit Timer/Event Counters 1 and 2 .....</b>	<b>219</b>
9.4.1 8-bit timer/event counter mode .....	219
9.4.2 16-bit timer/event counter mode .....	225
<b>9.5 Cautions on 8-Bit Timer/Event Counters 1 and 2 .....</b>	<b>230</b>
<b>CHAPTER 10 WATCH TIMER .....</b>	<b>232</b>
<b>10.1 Watch Timer Functions .....</b>	<b>232</b>
<b>10.2 Watch Timer Configuration .....</b>	<b>233</b>
<b>10.3 Watch Timer Control Registers .....</b>	<b>233</b>
<b>10.4 Watch Timer Operations .....</b>	<b>237</b>
10.4.1 Watch timer operation .....	237
10.4.2 Interval timer operation .....	237
<b>CHAPTER 11 WATCHDOG TIMER .....</b>	<b>238</b>
<b>11.1 Watchdog Timer Functions .....</b>	<b>238</b>
<b>11.2 Watchdog Timer Configuration .....</b>	<b>240</b>
<b>11.3 Watchdog Timer Control Registers .....</b>	<b>241</b>
<b>11.4 Watchdog Timer Operations .....</b>	<b>244</b>
11.4.1 Watchdog timer operation .....	244
11.4.2 Interval timer operation .....	245
<b>CHAPTER 12 CLOCK OUTPUT CONTROLLER .....</b>	<b>246</b>
<b>12.1 Clock Output Controller Functions .....</b>	<b>246</b>
<b>12.2 Clock Output Controller Configuration .....</b>	<b>247</b>
<b>12.3 Clock Output Function Control Registers .....</b>	<b>247</b>
<b>CHAPTER 13 BUZZER OUTPUT CONTROLLER .....</b>	<b>250</b>
<b>13.1 Buzzer Output Controller Functions .....</b>	<b>250</b>
<b>13.2 Buzzer Output Controller Configuration .....</b>	<b>250</b>
<b>13.3 Buzzer Output Function Control Registers .....</b>	<b>251</b>
<b>CHAPTER 14 A/D CONVERTER .....</b>	<b>254</b>
<b>14.1 A/D Converter Functions .....</b>	<b>254</b>
<b>14.2 A/D Converter Configuration .....</b>	<b>254</b>
<b>14.3 A/D Converter Control Registers .....</b>	<b>258</b>
<b>14.4 A/D Converter Operations .....</b>	<b>262</b>
14.4.1 Basic operations of A/D converter .....	262
14.4.2 Input voltage and conversion results .....	264
14.4.3 A/D converter operating mode .....	265



★	14.5 How to Read the A/D Converter Characteristics Table .....	267
	14.6 A/D Converter Cautions .....	269
	<b>CHAPTER 15 D/A CONVERTER .....</b>	<b>275</b>
	15.1 D/A Converter Functions .....	275
	15.2 D/A Converter Configuration .....	276
	15.3 D/A Converter Control Registers .....	278
	15.4 D/A Converter Operations .....	279
	15.5 D/A Converter Cautions .....	280
	<b>CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD780058 SUBSERIES) .....</b>	<b>281</b>
	16.1 Functions of Serial Interface Channel 0 .....	282
	16.2 Configuration of Serial Interface Channel 0 .....	284
	16.3 Control Registers of Serial Interface Channel 0 .....	288
	16.4 Operations of Serial Interface Channel 0 .....	295
	16.4.1 Operation stop mode .....	295
	16.4.2 3-wire serial I/O mode operation .....	296
	16.4.3 SBI mode operation .....	301
	16.4.4 2-wire serial I/O mode operation .....	327
	16.4.5 $\overline{SCK0}$ /P27 pin output manipulation .....	332
	<b>CHAPTER 17 SERIAL INTERFACE CHANNEL 0 (μPD780058Y SUBSERIES) .....</b>	<b>333</b>
	17.1 Functions of Serial Interface Channel 0 .....	334
	17.2 Configuration of Serial Interface Channel 0 .....	336
	17.3 Control Registers of Serial Interface Channel 0 .....	340
	17.4 Operations of Serial Interface Channel 0 .....	347
	17.4.1 Operation stop mode .....	347
	17.4.2 3-wire serial I/O mode operation .....	348
	17.4.3 2-wire serial I/O mode operation .....	352
	17.4.4 I <sup>2</sup> C bus mode operation .....	357
	17.4.5 Cautions on use of I <sup>2</sup> C bus mode .....	374
	17.4.6 Restrictions in I <sup>2</sup> C bus mode 1 .....	377
	17.4.7 Restrictions in I <sup>2</sup> C bus mode 2 .....	379
	17.4.8 $\overline{SCK0}$ /SCL/P27 pin output manipulation .....	380
	<b>CHAPTER 18 SERIAL INTERFACE CHANNEL 1 .....</b>	<b>382</b>
	18.1 Functions of Serial Interface Channel 1 .....	382
	18.2 Configuration of Serial Interface Channel 1 .....	383
	18.3 Control Registers of Serial Interface Channel 1 .....	386
	18.4 Operations of Serial Interface Channel 1 .....	394
	18.4.1 Operation stop mode .....	394
	18.4.2 3-wire serial I/O mode operation .....	395
	18.4.3 3-wire serial I/O mode operation with automatic transmit/receive function .....	398
	<b>CHAPTER 19 SERIAL INTERFACE CHANNEL 2 .....</b>	<b>427</b>
	19.1 Functions of Serial Interface Channel 2 .....	427
	19.2 Configuration of Serial Interface Channel 2 .....	428
	19.3 Control Registers of Serial Interface Channel 2 .....	432
	19.4 Operation of Serial Interface Channel 2 .....	442
	19.4.1 Operation stop mode .....	442

19.4.2	Asynchronous serial interface (UART) mode (with time-division transfer function) .....	444
19.4.3	3-wire serial I/O mode .....	458
19.4.4	Restrictions in UART mode 1 .....	465
★    19.4.5	Restrictions in UART mode 2 .....	468
<b>CHAPTER 20</b>	<b>REAL-TIME OUTPUT PORT .....</b>	<b>469</b>
<b>20.1</b>	<b>Real-Time Output Port Functions .....</b>	<b>469</b>
<b>20.2</b>	<b>Real-Time Output Port Configuration .....</b>	<b>470</b>
<b>20.3</b>	<b>Real-Time Output Port Control Registers .....</b>	<b>472</b>
<b>CHAPTER 21</b>	<b>INTERRUPT AND TEST FUNCTIONS .....</b>	<b>474</b>
<b>21.1</b>	<b>Interrupt Function Types .....</b>	<b>474</b>
<b>21.2</b>	<b>Interrupt Sources and Configuration .....</b>	<b>475</b>
<b>21.3</b>	<b>Interrupt Function Control Registers .....</b>	<b>479</b>
<b>21.4</b>	<b>Interrupt Servicing Operations .....</b>	<b>488</b>
21.4.1	Non-maskable interrupt request acknowledgment operation .....	488
21.4.2	Maskable interrupt request acknowledgment operation .....	491
21.4.3	Software interrupt request acknowledgment operation .....	494
21.4.4	Multiple interrupt servicing .....	494
21.4.5	Interrupt request pending .....	497
<b>21.5</b>	<b>Test Function .....</b>	<b>498</b>
21.5.1	Registers controlling test function .....	498
21.5.2	Test input signal acknowledgment operation .....	500
<b>CHAPTER 22</b>	<b>EXTERNAL DEVICE EXPANSION FUNCTION .....</b>	<b>501</b>
<b>22.1</b>	<b>External Device Expansion Function .....</b>	<b>501</b>
<b>22.2</b>	<b>External Device Expansion Function Control Register .....</b>	<b>505</b>
<b>22.3</b>	<b>External Device Expansion Function Timing .....</b>	<b>507</b>
<b>22.4</b>	<b>Example of Connection with Memory .....</b>	<b>512</b>
<b>CHAPTER 23</b>	<b>STANDBY FUNCTION .....</b>	<b>513</b>
<b>23.1</b>	<b>Standby Function and Configuration .....</b>	<b>513</b>
23.1.1	Standby function .....	513
23.1.2	Standby function control register .....	514
<b>23.2</b>	<b>Standby Function Operations .....</b>	<b>515</b>
23.2.1	HALT mode .....	515
23.2.2	STOP mode .....	518
<b>CHAPTER 24</b>	<b>RESET FUNCTION .....</b>	<b>521</b>
<b>24.1</b>	<b>Reset Function .....</b>	<b>521</b>
<b>CHAPTER 25</b>	<b>ROM CORRECTION .....</b>	<b>525</b>
<b>25.1</b>	<b>ROM Correction Function .....</b>	<b>525</b>
<b>25.2</b>	<b>ROM Correction Configuration .....</b>	<b>525</b>
<b>25.3</b>	<b>ROM Correction Control Registers .....</b>	<b>527</b>
<b>25.4</b>	<b>ROM Correction Application .....</b>	<b>528</b>
<b>25.5</b>	<b>ROM Correction Usage Example .....</b>	<b>531</b>
<b>25.6</b>	<b>Program Execution Flow .....</b>	<b>532</b>
<b>25.7</b>	<b>ROM Correction Cautions .....</b>	<b>534</b>

	<b>CHAPTER 26 <math>\mu</math>PD78F0058, 78F0058Y .....</b>	<b>535</b>
	<b>26.1 Internal memory Size Switching Register .....</b>	<b>536</b>
	<b>26.2 Internal Expansion RAM Size Switching Register .....</b>	<b>537</b>
	<b>26.3 Flash Memory Characteristics .....</b>	<b>538</b>
	26.3.1 Programming environment .....	538
	26.3.2 Communication mode .....	539
	26.3.3 On-board pin processing .....	543
	26.3.4 Connection of adapter for flash writing .....	546
	<b>CHAPTER 27 INSTRUCTION SET OVERVIEW .....</b>	<b>552</b>
	<b>27.1 Conventions Used in Operation List .....</b>	<b>553</b>
	27.1.1 Operand identifiers and description methods .....	553
	27.1.2 Description of operation column .....	554
	27.1.3 Description of flag operation column .....	554
	<b>27.2 Operation List .....</b>	<b>555</b>
	<b>27.3 Instructions Listed by Addressing Type .....</b>	<b>563</b>
★	<b>CHAPTER 28 ELECTRICAL SPECIFICATIONS (MASK ROM VERSION) .....</b>	<b>567</b>
★	<b>CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION) .....</b>	<b>597</b>
★	<b>CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION (<math>V_{DD} = 2.5 V</math>)) ....</b>	<b>627</b>
★	<b>CHAPTER 31 CHARACTERISTICS CURVES (REFERENCE VALUES) .....</b>	<b>658</b>
★	<b>CHAPTER 32 PACKAGE DRAWINGS .....</b>	<b>660</b>
★	<b>CHAPTER 33 RECOMMENDED SOLDERING CONDITIONS .....</b>	<b>662</b>
	<b>APPENDIX A DIFFERENCES BETWEEN <math>\mu</math>PD78054, 78058F, AND 780058 SUBSERIES .....</b>	<b>666</b>
	<b>APPENDIX B DEVELOPMENT TOOLS .....</b>	<b>668</b>
	<b>B.1 Software Package .....</b>	<b>670</b>
	<b>B.2 Language Processing Software .....</b>	<b>670</b>
	<b>B.3 Control Software .....</b>	<b>671</b>
	<b>B.4 Flash Memory Writing Tools .....</b>	<b>671</b>
	<b>B.5 Debugging Tools (Hardware) .....</b>	<b>672</b>
	B.5.1 When using in-circuit emulator IE-78K0-NS, IE-78K0-NS-A .....	672
	B.5.2 When using in-circuit emulator IE-78001-R-A .....	673
	<b>B.6 Debugging Tools (Software) .....</b>	<b>674</b>
	<b>B.7 Embedded Software .....</b>	<b>675</b>
	<b>B.8 System-Upgrade Method from Former In-Circuit Emulator for 78K/0 Series         to IE-78001-R-A .....</b>	<b>676</b>
	<b>B.9 Drawing and Footprint for Conversion Socket (EV-9200GC-80) .....</b>	<b>677</b>
	<b>B.10 Drawing of Conversion Adapter (TGC-080SDW, TGC-080SBP) .....</b>	<b>679</b>
★	<b>B.11 Cautions on Designing Target System .....</b>	<b>681</b>
	<b>APPENDIX C REGISTER INDEX .....</b>	<b>685</b>
	<b>C.1 Register Index (Register Name) .....</b>	<b>685</b>
	<b>C.2 Register Index (Symbol) .....</b>	<b>688</b>
	<b>APPENDIX D REVISION HISTORY .....</b>	<b>691</b>

## LIST OF FIGURES (1/8)

Figure No.	Title	Page
3-1	Pin I/O Circuit List .....	64
4-1	Pin I/O Circuit List .....	79
5-1	Memory Map ( $\mu$ PD780053, 780053(A), 780053Y, 780053Y(A)) .....	81
5-2	Memory Map ( $\mu$ PD780054, 780054(A), 780054Y, 780054Y(A)) .....	82
5-3	Memory Map ( $\mu$ PD780055, 780055(A), 780055Y, 780055Y(A)) .....	83
5-4	Memory Map ( $\mu$ PD780056, 780056(A), 780056Y, 780056Y(A)) .....	84
5-5	Memory Map ( $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A)) .....	85
5-6	Memory Map ( $\mu$ PD78F0058, 78F0058Y) .....	86
5-7	Data Memory Addressing ( $\mu$ PD780053, 780053(A), 780053Y, 780053Y(A)) .....	90
5-8	Data Memory Addressing ( $\mu$ PD780054, 780054(A), 780054Y, 780054Y(A)) .....	91
5-9	Data Memory Addressing ( $\mu$ PD780055, 780055(A), 780055Y, 780055Y(A)) .....	92
5-10	Data Memory Addressing ( $\mu$ PD780056, 780056(A), 780056Y, 780056Y(A)) .....	93
5-11	Data Memory Addressing ( $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A)) .....	94
5-12	Data Memory Addressing ( $\mu$ PD78F0058, 78F0058Y) .....	95
5-13	Program Counter Format .....	96
5-14	Program Status Word Format .....	96
5-15	Stack Pointer Format .....	98
5-16	Data to Be Saved to Stack Memory .....	98
5-17	Data to Be Reset from Stack Memory .....	98
5-18	General-purpose Register Configuration .....	99
6-1	Port Types .....	117
6-2	Block Diagram of P00 and P07 .....	123
6-3	Block Diagram of P01 to P05 .....	123
6-4	Block Diagram of P10 to P17 .....	124
6-5	Block Diagram of P20, P21, and P23 to P26 .....	125
6-6	Block Diagram of P22 and P27 .....	126
6-7	Block Diagram of P20, P21, and P23 to P26 .....	127
6-8	Block Diagram of P22 and P27 .....	128
6-9	Block Diagram of P30 to P37 .....	129
6-10	Block Diagram of P40 to P47 .....	130
6-11	Block Diagram of Falling Edge Detector .....	130
6-12	Block Diagram of P50 to P57 .....	131
6-13	Block Diagram of P60 to P63 .....	133
6-14	Block Diagram of P64 to P67 .....	133
6-15	Block Diagram of P70 .....	134
6-16	Block Diagram of P71 and P72 .....	135
6-17	Block Diagram of P120 to P127 .....	136
6-18	Block Diagram of P130 and P131 .....	137
6-19	Port Mode Register Format .....	140
6-20	Format of Pull-up Resistor Option Register .....	141
6-21	Format of Memory Expansion Mode Register .....	142

## LIST OF FIGURES (2/8)

Figure No.	Title	Page
6-22	Format of Key Return Mode Register .....	143
7-1	Clock Generator Block Diagram .....	147
7-2	Subsystem Clock Feedback Resistor .....	148
7-3	Format of Processor Clock Control Register .....	149
7-4	Format of Oscillation Mode Selection Register .....	151
7-5	Main System Clock Waveform due to Writing to OSMS .....	151
7-6	External Circuit of Main System Clock Oscillator .....	152
7-7	External Circuit of Subsystem Clock Oscillator .....	153
7-8	Examples of Resonator with Bad Connection .....	154
7-9	Main System Clock Stop Function .....	157
7-10	Switching Between System Clock and CPU Clock .....	161
8-1	Block Diagram of 16-Bit Timer/Event Counter .....	165
8-2	Block Diagram of 16-Bit Timer/Event Counter Output Controller .....	166
8-3	Format of Timer Clock Select Register 0 .....	170
8-4	Format of 16-Bit Timer Mode Control Register .....	172
8-5	Format of Capture/Compare Control Register 0 .....	173
8-6	Format of 16-Bit Timer Output Control Register .....	175
8-7	Format of Port Mode Register 3 .....	176
8-8	Format of External Interrupt Mode Register 0 .....	177
8-9	Format of Sampling Clock Select Register .....	178
8-10	Control Register Settings for Interval Timer Operation .....	179
8-11	Interval Timer Configuration Diagram .....	180
8-12	Interval Timer Operation Timings .....	180
8-13	Control Register Settings for PWM Output Operation .....	182
8-14	Example of D/A Converter Configuration with PWM Output .....	183
8-15	TV Tuner Application Circuit Example .....	183
8-16	Control Register Settings for PPG Output Operation .....	184
8-17	Configuration of PPG Output .....	185
8-18	PPG Output Operation Timing .....	185
8-19	Control Register Settings for Pulse Width Measurement with Free-Running Counter and One Capture Register .....	186
8-20	Configuration Diagram for Pulse Width Measurement by Free-Running Counter .....	187
8-21	Timing of Pulse Width Measurement Operation by Free-Running Counter and One Capture Register (with Both Edges Specified) .....	187
8-22	Control Register Settings for Two Pulse Width Measurements with Free-Running Counter .....	188
8-23	Timing of Pulse Width Measurement Operation with Free-Running Counter (with Both Edges Specified) .....	189
8-24	Control Register Settings for Pulse Width Measurement with Free-Running Counter and Two Capture Registers .....	190
8-25	Timing of Pulse Width Measurement Operation by Free-Running Counter and Two Capture Registers (with Rising Edge Specified) .....	191
8-26	Control Register Settings for Pulse Width Measurement by Means of Restart .....	192

## LIST OF FIGURES (3/8)

Figure No.	Title	Page
8-27	Timing of Pulse Width Measurement Operation by Means of Restart (with Rising Edge Specified) .....	192
8-28	Control Register Settings in External Event Counter Mode .....	193
8-29	External Event Counter Configuration Diagram .....	194
8-30	External Event Counter Operation Timing (with Rising Edge Specified) .....	194
8-31	Control Register Settings in Square-Wave Output Mode .....	195
8-32	Square-Wave Output Operation Timing .....	196
8-33	Control Register Settings for One-Shot Pulse Output Operation Using Software Trigger .....	197
8-34	One-Shot Pulse Output Operation Timing Using Software Trigger .....	198
8-35	Control Register Settings for One-Shot Pulse Output Operation Using External Trigger .....	199
8-36	One-Shot Pulse Output Operation Timing Using External Trigger (with Rising Edge Specified) .....	200
8-37	16-Bit Timer Register Start Timing .....	201
8-38	Timing After Change of Compare Register During Timer Count Operation .....	201
8-39	Capture Register Data Retention Timing .....	202
8-40	Operation Timing of OVFO Flag .....	203
9-1	Block Diagram of 8-Bit Timer/Event Counter .....	211
9-2	Block Diagram of 8-Bit Timer/Event Counter Output Controller 1 .....	212
9-3	Block Diagram of 8-Bit Timer/Event Counter Output Controller 2 .....	212
9-4	Format of Timer Clock Select Register 1 .....	215
9-5	Format of 8-Bit Timer Mode Control Register 1 .....	216
9-6	Format of 8-Bit Timer Output Control Register .....	217
9-7	Format of Port Mode Register 3 .....	218
9-8	Interval Timer Operation Timing .....	219
9-9	External Event Counter Operation Timing (with Rising Edge Specified) .....	222
9-10	Square-Wave Output Operation Timing .....	224
9-11	Interval Timer Operation Timing .....	225
9-12	External Event Counter Operation Timing (with Rising Edge Specified) .....	227
9-13	Square-Wave Output Operation Timing .....	229
9-14	Start Timing of 8-Bit Timer Registers 1 and 2 .....	230
9-15	External Event Counter Operation Timing .....	230
9-16	Timing After Compare Register Change During Timer Count Operation .....	231
10-1	Watch Timer Block Diagram .....	234
10-2	Format of Timer Clock Select Register 2 .....	235
10-3	Format of Watch Timer Mode Control Register .....	236
11-1	Watchdog Timer Block Diagram .....	240
11-2	Format of Timer Clock Select Register 2 .....	242
11-3	Format of Watchdog Timer Mode Register .....	243
12-1	Remote Controlled Output Application Example .....	246
12-2	Clock Output Controller Block Diagram .....	247
12-3	Format of Timer Clock Select Register 0 .....	248

## LIST OF FIGURES (4/8)

Figure No.	Title	Page
12-4	Format of Port Mode Register 3 .....	249
13-1	Buzzer Output Controller Block Diagram .....	250
13-2	Format of Timer Clock Select Register 2 .....	252
13-3	Format of Port Mode Register 3 .....	253
14-1	A/D Converter Block Diagram .....	255
14-2	Format of A/D Converter Mode Register .....	259
14-3	Format of A/D Converter Input Select Register .....	260
14-4	Format of External Interrupt Mode Register 1 .....	261
14-5	A/D Converter Basic Operation .....	263
14-6	Relationship Between Analog Input Voltage and A/D Conversion Result .....	264
14-7	A/D Conversion by Hardware Start .....	265
14-8	A/D Conversion by Software Start .....	266
14-9	Overall Error .....	267
14-10	Quantization Error .....	267
14-11	Example of Method of Reducing Current Consumption in Standby Mode .....	269
14-12	Analog Input Pin Handling .....	270
14-13	A/D Conversion End Interrupt Request Generation Timing .....	271
14-14	Timing of Reading Conversion Result (When Conversion Result is Undefined) .....	272
14-15	Timing of Reading Conversion Result (When Conversion Result is Normal) .....	272
14-16	Example of Connecting Capacitor to AV <sub>REF0</sub> Pin .....	273
14-17	Internal Equivalent Circuit of Pins ANI0 to ANI7 .....	274
14-18	Example of Connection If Signal Source Impedance Is High .....	274
15-1	D/A Converter Block Diagram .....	276
15-2	Format of D/A Converter Mode Register .....	278
15-3	Use Example of Buffer Amplifier .....	280
16-1	Serial Bus Interface (SBI) System Configuration Example .....	283
16-2	Block Diagram of Serial Interface Channel 0 .....	285
16-3	Format of Timer Clock Select Register 3 .....	289
16-4	Format of Serial Operating Mode Register 0 .....	290
16-5	Format of Serial Bus Interface Control Register .....	292
16-6	Format of Interrupt Timing Specification Register .....	294
16-7	3-Wire Serial I/O Mode Timing .....	299
16-8	RELT and CMDT Operations .....	299
16-9	Circuit for Switching Transfer Bit Order .....	300
16-10	Example of Serial Bus Configuration with SBI .....	301
16-11	SBI Transfer Timing .....	303
16-12	Bus Release Signal .....	304
16-13	Command Signal .....	304
16-14	Addresses .....	305
16-15	Slave Selection by Address .....	305

## LIST OF FIGURES (5/8)

Figure No.	Title	Page
16-16	Commands .....	306
16-17	Data .....	306
16-18	Acknowledge Signal .....	307
16-19	BUSY and READY Signals .....	308
16-20	RELT, CMDT, RELD, and CMDD Operations (Master) .....	313
16-21	RELD and CMDD Operations (Slave) .....	313
16-22	ACKT Operation .....	314
16-23	ACKE Operations .....	315
16-24	ACKD Operations .....	316
16-25	BSYE Operation .....	316
16-26	Pin Configuration .....	319
16-27	Address Transmission from Master Device to Slave Device (WUP = 1) .....	321
16-28	Command Transmission from Master Device to Slave Device .....	322
16-29	Data Transmission from Master Device to Slave Device .....	323
16-30	Data Transmission from Slave Device to Master Device .....	324
16-31	Serial Bus Configuration Example Using 2-Wire Serial I/O Mode .....	327
16-32	2-Wire Serial I/O Mode Timing .....	330
16-33	RELT and CMDT Operations .....	331
16-34	$\overline{SCK0}$ /P27 Pin Configuration .....	332
17-1	Serial Bus Configuration Example Using I <sup>2</sup> C Bus .....	335
17-2	Block Diagram of Serial Interface Channel 0 .....	337
17-3	Format of Timer Clock Select Register 3 .....	341
17-4	Format of Serial Operating Mode Register 0 .....	342
17-5	Format of Serial Bus Interface Control Register .....	343
17-6	Format of Interrupt Timing Specification Register .....	345
17-7	3-Wire Serial I/O Mode Timing .....	350
17-8	RELT and CMDT Operations .....	350
17-9	Circuit for Switching Transfer Bit Order .....	351
17-10	Serial Bus Configuration Example Using 2-Wire Serial I/O Mode .....	352
17-11	2-Wire Serial I/O Mode Timing .....	355
17-12	RELT and CMDT Operations .....	356
17-13	Example of Serial Bus Configuration Using I <sup>2</sup> C Bus .....	357
17-14	I <sup>2</sup> C Bus Serial Data Transfer Timing .....	358
17-15	Start Condition .....	359
17-16	Address .....	359
17-17	Transfer Direction Specification .....	359
17-18	Acknowledge Signal .....	360
17-19	Stop Condition .....	360
17-20	Wait Signal .....	361
17-21	Pin Configuration .....	366
17-22	Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) .....	368
17-23	Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) .....	371
17-24	Start Condition Output .....	374



## LIST OF FIGURES (6/8)

Figure No.	Title	Page
17-25	Slave Wait Release (Transmission) .....	375
17-26	Slave Wait Release (Reception) .....	376
17-27	$\overline{\text{SCK0}}$ /SCL/P27 Pin Configuration .....	380
17-28	$\overline{\text{SCK0}}$ /SCL/P27 Pin Configuration .....	380
17-29	Logic Circuit of SCL Signal .....	381
18-1	Block Diagram of Serial Interface Channel 1 .....	384
18-2	Format of Timer Clock Select Register 3 .....	387
18-3	Format of Serial Operation Mode Register 1 .....	388
18-4	Format of Automatic Data Transmit/Receive Control Register .....	389
18-5	Format of Automatic Data Transmit/Receive Interval Specification Register .....	390
18-6	3-Wire Serial I/O Mode Timing .....	396
18-7	Circuit for Switching Transfer Bit Order .....	397
18-8	Basic Transmission/Reception Mode Operation Timing .....	406
18-9	Basic Transmission/Reception Mode Flowchart .....	407
18-10	Internal Buffer RAM Operation in 6-Byte Transmission/Reception (in Basic Transmit/Receive Mode) .....	408
18-11	Basic Transmission Mode Operation Timing .....	410
18-12	Basic Transmission Mode Flowchart .....	411
18-13	Internal Buffer RAM Operation in 6-Byte Transmission (in Basic Transmit Mode) .....	412
18-14	Repeat Transmission Mode Operation Timing .....	414
18-15	Repeat Transmission Mode Flowchart .....	415
18-16	Internal Buffer RAM Operation in 6-Byte Transmission (in Repeat Transmit Mode) .....	416
18-17	Automatic Transmission/Reception Suspension and Restart .....	418
18-18	System Configuration When Busy Control Option Is Used .....	419
18-19	Operation Timing When Busy Control Option Is Used (When BUSY0 = 0) .....	420
18-20	Busy Signal and Wait Release (When BUSY0 = 0) .....	421
18-21	Operation Timing When Busy & Strobe Control Options Are Used (When BUSY0 = 0) .....	422
18-22	Operation Timing of Bit Shift Detection Function by Busy Signal (When BUSY0 = 1) .....	423
18-23	Automatic Data Transmit/Receive Interval Time .....	424
18-24	Operation Timing with Automatic Data Transmit/Receive Function Performed Using Internal Clock .....	425
19-1	Block Diagram of Serial Interface Channel 2 .....	429
19-2	Baud Rate Generator Block Diagram .....	430
19-3	Format of Serial Operating Mode Register 2 .....	432
19-4	Format of Asynchronous Serial Interface Mode Register .....	433
19-5	Format of Asynchronous Serial Interface Status Register .....	436
19-6	Format of Baud Rate Generator Control Register .....	437
19-7	Format of Serial Interface Pin Select Register .....	441
19-8	Format of Asynchronous Serial Interface Transmit/Receive Data .....	452
19-9	Asynchronous Serial Interface Transmission Completion Interrupt Request Generation Timing .....	454
19-10	Asynchronous Serial Interface Reception Completion Interrupt Request Generation Timing .....	455
19-11	Receive Error Timing .....	456

## LIST OF FIGURES (7/8)

Figure No.	Title	Page
19-12	Status of Receive Buffer Register (RXB) and Generation of Interrupt Request (INTSR) When Reception Is Stopped.....	457
19-13	3-Wire Serial I/O Mode Timing .....	463
19-14	Circuit for Switching Transfer Bit Order .....	464
19-15	Reception Completion Interrupt Request Generation Timing (When ISRM = 1) .....	465
19-16	Receive Buffer Register Read Disable Period .....	466
19-17	P23 Output Selector .....	468
20-1	Real-Time Output Port Block Diagram .....	470
20-2	Real-Time Output Buffer Register Configuration .....	471
20-3	Format of Port Mode Register 12 .....	472
20-4	Format of Real-Time Output Port Mode Register .....	472
20-5	Format of Real-Time Output Port Control Register .....	473
21-1	Basic Configuration of Interrupt Function .....	477
21-2	Format of Interrupt Request Flag Register .....	480
21-3	Interrupt Mask Flag Register Format .....	481
21-4	Format of Priority Specification Flag Register .....	482
21-5	Format of External Interrupt Mode Register 0 .....	483
21-6	Format of External Interrupt Mode Register 1 .....	484
21-7	Format of Sampling Clock Select Register .....	485
21-8	Noise Eliminator I/O Timing (During Rising Edge Detection) .....	486
21-9	Format of Program Status Word .....	487
21-10	Non-Maskable Interrupt Request Occurrence and Acknowledgment Flowchart .....	489
21-11	Non-Maskable Interrupt Request Acknowledgment Timing .....	489
21-12	Non-Maskable Interrupt Request Acknowledgment Operation .....	490
21-13	Interrupt Request Acknowledgment Processing Algorithm .....	492
21-14	Interrupt Request Acknowledgment Timing (Minimum Time) .....	493
21-15	Interrupt Request Acknowledgment Timing (Maximum Time) .....	493
21-16	Multiple Interrupt Servicing Example .....	495
21-17	Interrupt Request Pending Timing .....	497
21-18	Basic Configuration of Test Function .....	498
21-19	Format of Interrupt Request Flag Register 1L .....	499
21-20	Format of Interrupt Mask Flag Register 1L .....	499
21-21	Format of Key Return Mode Register .....	500
22-1	Memory Map When Using External Device Expansion Function .....	502
22-2	Format of Memory Expansion Mode Register .....	505
22-3	Format of Internal Memory Size Switching Register .....	506
22-4	Instruction Fetch from External Memory .....	508
22-5	External Memory Read Timing .....	509
22-6	External Memory Write Timing .....	510
22-7	External Memory Read Modify Write Timing .....	511
22-8	Example of Connection Between $\mu$ PD780054 and Memory .....	512

## LIST OF FIGURES (8/8)

Figure No.	Title	Page
23-1	Format of Oscillation Stabilizat Time Selection Register .....	514
23-2	HALT Mode Release by Interrupt Request Generation .....	516
23-3	HALT Mode Release by $\overline{\text{RESET}}$ Input .....	517
23-4	STOP Mode Release by Interrupt Request Generation .....	519
23-5	STOP Mode Release by $\overline{\text{RESET}}$ Input .....	520
24-1	Reset Function Block Diagram .....	521
24-2	Reset Timing by $\overline{\text{RESET}}$ Input .....	522
24-3	Reset Timing due to Watchdog Timer Overflow .....	522
24-4	Reset Timing by $\overline{\text{RESET}}$ Input in STOP Mode .....	522
25-1	ROM Correction Block Diagram .....	525
25-2	Format of Correction Address Registers 0 and 1 .....	526
25-3	Format of Correction Control Register .....	527
25-4	Example of Storing to EEPROM (When One Place Is Corrected) .....	528
25-5	Initialization Routine .....	529
25-6	ROM Correction Operation .....	530
25-7	ROM Correction Usage Example .....	531
25-8	Program Transition Diagram (When One Place Is Corrected) .....	532
25-9	Program Transition Diagram (When Two Places Are Corrected) .....	533
26-1	Format of Memory Size Switching Register .....	536
26-2	Format of Internal Expansion RAM Size Switching Register .....	537
23-3	Environment for Writing Program to Flash Memory .....	538
26-4	Communication Mode Selection Format .....	539
26-5	Example of Connection with Dedicated Flash Programmer .....	540
26-6	V <sub>PP</sub> Pin Connection Example .....	543
26-7	Signal Conflict (Input Pin of Serial Interface) .....	544
26-8	Abnormal Operation of Other Device .....	544
26-9	Signal Conflict ( $\overline{\text{RESET}}$ Pin) .....	545
26-10	Wiring Example for Flash Writing Adapter in 3-Wire Serial I/O Mode (SIO ch-0) .....	546
26-11	Wiring Example for Flash Writing Adapter in 3-Wire Serial I/O Mode (SIO ch-1) .....	547
26-12	Wiring Example for Flash Writing Adapter in 3-Wire Serial I/O Mode (SIO ch-2) .....	548
26-13	Wiring Example for Flash Writing Adapter in UART Mode (UART ch-0) .....	549
26-14	Wiring Example for Flash Writing Adapter in UART Mode (UART ch-1) .....	550
26-15	Wiring Example for Flash Writing Adapter in Pseudo 3-Wire Mode .....	551
B-1	Configuration of Development Tools .....	669
B-2	EV-9200GC-80 Drawing (For Reference Only) .....	677
B-3	EV-9200GC-80 Footprint (For Reference Only) .....	678
B-4	TGK-080SDW Drawing (For Reference Only) (Unit: mm) .....	679
B-5	TGC-080SBP Drawing (For Reference Only) (Unit: mm) .....	680
B-6	Distance Between In-Circuit Emulator and Conversion Socket (80GC) .....	681
B-7	Connection Condition of Target System (NP-80GC-TQ) .....	682
B-8	Distance Between In-Circuit Emulator and Conversion Socket (80GK) .....	683
B-9	Connection Condition of Target System (NP-80GK) .....	684

## LIST OF TABLES (1/3)

Table No.	Title	Page
1-1	Mask Options of Mask ROM Versions .....	40
1-2	Differences Between Standard Model and (A) Model .....	40
2-1	Mask Options of Mask ROM Versions .....	50
2-2	Differences Between Standard Model and (A) Model .....	50
3-1	Pin I/O Circuit Types .....	62
4-1	Pin I/O Circuit Types .....	77
5-1	Vector .....	88
5-2	Special-Function Register List .....	101
6-1	Port Functions ( $\mu$ PD780058 Subseries) .....	118
6-2	Port Functions ( $\mu$ PD780058Y Subseries) .....	120
6-3	Port Configuration .....	122
6-4	Pull-up Resistor of Port 6 .....	132
6-5	Port Mode Register and Output Latch Settings When Using Alternate Functions .....	139
6-6	Comparison Between Mask ROM Version and Flash Memory Version .....	145
7-1	Clock Generator Configuration .....	146
7-2	Relationship Between CPU Clock and Minimum Instruction Execution Time .....	150
7-3	Maximum Time Required for CPU Clock Switchover .....	160
8-1	16-Bit Timer/Event Counter Interval Times .....	162
8-2	16-Bit Timer/Event Counter Square-Wave Output Ranges .....	163
8-3	16-Bit Timer/Event Counter Configuration .....	164
8-4	INTP0/TI00 Pin Valid Edge and CR00 Capture Trigger Valid Edge .....	167
8-5	INTP1/TI01 Pin Valid Edge and CR00 Capture Trigger Valid Edge .....	167
8-6	INTP0/TI00 Pin Valid Edge and CR01 Capture Trigger Valid Edge .....	168
8-7	16-Bit Timer/Event Counter Interval Times .....	181
8-8	16-Bit Timer/Event Counter Square-Wave Output Ranges .....	196
9-1	Interval Times of 8-Bit Timer/Event Counters 1 and 2 .....	206
9-2	Square-Wave Output Ranges of 8-Bit Timer/Event Counters 1 and 2 .....	207
9-3	Interval Times When 8-Bit Timer/Event Counters 1 and 2 Are Used as 16-Bit Timer/Event Counter .....	208
9-4	Square-Wave Output Ranges When 8-Bit Timer/Event Counters 1 and 2 Are Used as 16-Bit Timer/Event Counter .....	209
9-5	8-Bit Timer/Event Counter Configuration .....	210
9-6	Interval Time of 8-Bit Timer/Event Counter 1 .....	220
9-7	Interval Time of 8-Bit Timer/Event Counter 2 .....	221
9-8	Square-Wave Output Ranges of 8-Bit Timer/Event Counters 1 and 2 .....	223

## LIST OF TABLES (2/3)

Table No.	Title	Page
9-9	Interval Times When 2-Channel 8-Bit Timer/Event Counters (TM1 and TM2) Are Used as 16-Bit Timer/Event Counter .....	226
9-10	Square-Wave Output Ranges When 2-Channel 8-Bit Timer/Event Counters (TM1 and TM2) Are Used as 16-Bit Timer/Event Counter .....	228
10-1	Interval Timer Interval Time .....	232
10-2	Watch Timer Configuration .....	233
10-3	Interval Timer Interval Time .....	237
11-1	Watchdog Timer Program Loop Detection Times .....	238
11-2	Interval Times .....	239
11-3	Watchdog Timer Configuration .....	240
11-4	Watchdog Timer Program Loop Detection Time .....	244
11-5	Interval Timer Interval Time .....	245
12-1	Clock Output Controller Configuration .....	247
13-1	Buzzer Output Controller Configuration .....	250
14-1	A/D Converter Configuration .....	254
14-2	A/D Converter Sampling Time and A/D Conversion Start Delay Time .....	263
14-3	Resistances and Capacitances of Equivalent Circuit (Reference Values) .....	274
15-1	D/A Converter Configuration .....	276
16-1	Differences Between Channels 0, 1, and 2 .....	281
16-2	Configuration of Serial Interface Channel 0 .....	284
16-3	Various Signals in SBI Mode .....	317
17-1	Differences Between Channels 0, 1, and 2 .....	333
17-2	Configuration of Serial Interface Channel 0 .....	336
17-3	Interrupt Request Signal Generation of Serial Interface Channel 0 .....	339
17-4	Signals in I <sup>2</sup> C Bus Mode .....	365
18-1	Configuration of Serial Interface Channel 1 .....	383
18-2	Interval Timing According to CPU Processing (When Internal Clock Is Operating) .....	425
18-3	Interval Time According to CPU Processing (with External Clock) .....	426
19-1	Configuration of Serial Interface Channel 2 .....	428
19-2	Operating Mode Settings of Serial Interface Channel 2 .....	434
19-3	Relationship Between Main System Clock and Baud Rate .....	439
19-4	Relationship Between ASCK Pin Input Frequency and Baud Rate (When BRGC Is Set to 00H) .....	440
19-5	Relationship Between Main System Clock and Baud Rate .....	449
19-6	Relationship Between ASCK Pin Input Frequency and Baud Rate (When BRGC Is Set to 00H) .....	450

### LIST OF TABLES (3/3)

Table No.	Title	Page
19-7	Receive Error Causes .....	456
20-1	Real-Time Output Port Configuration .....	470
20-2	Operation in Real-Time Output Buffer Register Manipulation .....	471
20-3	Real-Time Output Port Operating Mode and Output Trigger .....	473
21-1	Interrupt Source List .....	475
21-2	Various Flags Corresponding to Interrupt Request Sources .....	479
21-3	Times from Maskable Interrupt Request Generation to Interrupt Servicing .....	491
21-4	Interrupt Request Enabled for Multiple Interrupt Servicing During Interrupt Servicing .....	494
21-5	Test Input Sources .....	498
21-6	Flags Corresponding to Test Input Signals .....	498
22-1	Pin Functions in External Memory Expansion Mode .....	501
22-2	State of Port 4 to 6 Pins in External Memory Expansion Mode .....	501
22-3	Values After Internal Memory Size Switching Register Is Reset .....	506
23-1	HALT Mode Operating Status .....	515
23-2	Operation After HALT Mode Release .....	517
23-3	STOP Mode Operating Status .....	518
23-4	Operation After STOP Mode Release .....	520
24-1	Hardware Status After Reset .....	523
25-1	ROM Correction Configuration .....	525
26-1	Differences Between $\mu$ PD78F0058, 78F0058Y and Mask ROM Versions .....	535
26-2	Internal Memory Size Switching Register Setting Values .....	536
26-3	Internal Expansion RAM Size Switching Register Setting Values .....	537
26-4	Communication Mode List .....	539
26-5	Pin Connection List .....	542
27-1	Operand Identifiers and Description Methods .....	553
33-1	Surface Mounting Type Soldering Conditions .....	662
A-1	Major Differences Between $\mu$ PD78054, 78058F, and 780058 Subseries .....	666
B-1	System-Upgrade Method from Former In-Circuit Emulator for 78K/0 Series to IE-78001-R-A .....	676

## CHAPTER 1 OUTLINE ( $\mu$ PD780058 SUBSERIES)

### 1.1 Features

- ★ ○ On-chip high-capacity ROM and RAM

Part Number \ Item	Program Memory		Data Memory		
	Mask ROM	Flash Memory	Internal High-Speed RAM	Internal Buffer RAM	Internal Expansion RAM
$\mu$ PD780053, 780053(A)	24 KB	—	1,024 bytes	32 bytes	None
$\mu$ PD780054, 780054(A)	32 KB	—			
$\mu$ PD780055, 780055(A)	40 KB	—			
$\mu$ PD780056, 780056(A)	48 KB	—			
$\mu$ PD780058, 780058B, 780058B(A)	60 KB	—			
$\mu$ PD78F0058	—	60 KB <sup>Note 1</sup>			1,024 bytes <sup>Note 2</sup>

- Notes**
1. The flash memory capacity can be changed by means of the internal memory size switching register (IMS).
  2. The capacity of the internal high-speed RAM can be changed by means of the internal expansion RAM size switching register (IXS).

- External memory expansion space: 64 KB
  - Minimum instruction execution time changeable from high-speed (0.4  $\mu$ s: Main system clock 5.0 MHz operation) to ultra-low speed (122  $\mu$ s: Subsystem clock 32.768 kHz operation)
  - Instruction set suited to system control
    - Bit manipulation possible in all address spaces
    - Multiple and divide instructions
  - I/O ports: 68 (N-ch open-drain: 4)
  - ★ ○ 8-bit resolution A/D converter: 8 channels ( $V_{DD} = 1.8$  to  $5.5$  V<sup>Note</sup>)
  - ★ ○ 8-bit resolution D/A converter: 2 channels ( $V_{DD} = 1.8$  to  $5.5$  V<sup>Note</sup>)
  - Serial interface: 3 channels
    - 3-wire serial I/O/SBI/2-wire serial I/O mode: 1 channel
    - 3-wire serial I/O mode (on-chip automatic transmit/receive function): 1 channel
    - 3-wire serial I/O/UART mode (on-chip time-division transfer function): 1 channel
  - Timer: 5 channels
    - 16-bit timer/event counter: 1 channel
    - 8-bit timer/event counter: 2 channels
    - Watch timer: 1 channel
    - Watchdog timer: 1 channel
- ★ **Note** The operating voltage range of the A/D and D/A converters of the  $\mu$ PD780058 is  $V_{DD} = 2.7$  to  $5.5$  V.

- Vectored interrupt sources: 21
- Test inputs: 2
- Two types of on-chip clock oscillators (main system clock and subsystem clock)
- ★ ○ Supply voltage:  $V_{DD} = 1.8$  to  $5.5$  V (mask ROM version)
- ★  $V_{DD} = 2.7$ <sup>Note</sup> to  $5.5$  V ( $\mu$ PD78F0058)
  
- ★ **Note**  $V_{DD} = 2.2$  V can also be supplied to the  $\mu$ PD78F0058. For details, contact an NEC Electronics sales representative.

## 1.2 Applications

Car audio systems, cellular phones, pagers, printers, AV equipment, cameras, PPCs, vending machines, car electrical components, etc.

## ★ 1.3 Ordering Information

Part Number	Package	Internal ROM
$\mu$ PD780053GC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780053GK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780054GC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780054GK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780055GC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780055GK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780056GC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780056GK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780058GC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780058GK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
★ $\mu$ PD780058BGC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
★ $\mu$ PD780058BGK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780053GC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780054GC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780055GC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780056GC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780058BGC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD78F0058GC-8BT	80-pin plastic QFP (14 × 14)	Flash memory
$\mu$ PD78F0058GK-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Flash memory

**Remark** xxx indicates ROM code suffix.

For details of the quality grades and their applications, see **Quality Grades on NEC Electronics Semiconductor Devices** (Document No.: C11531E).



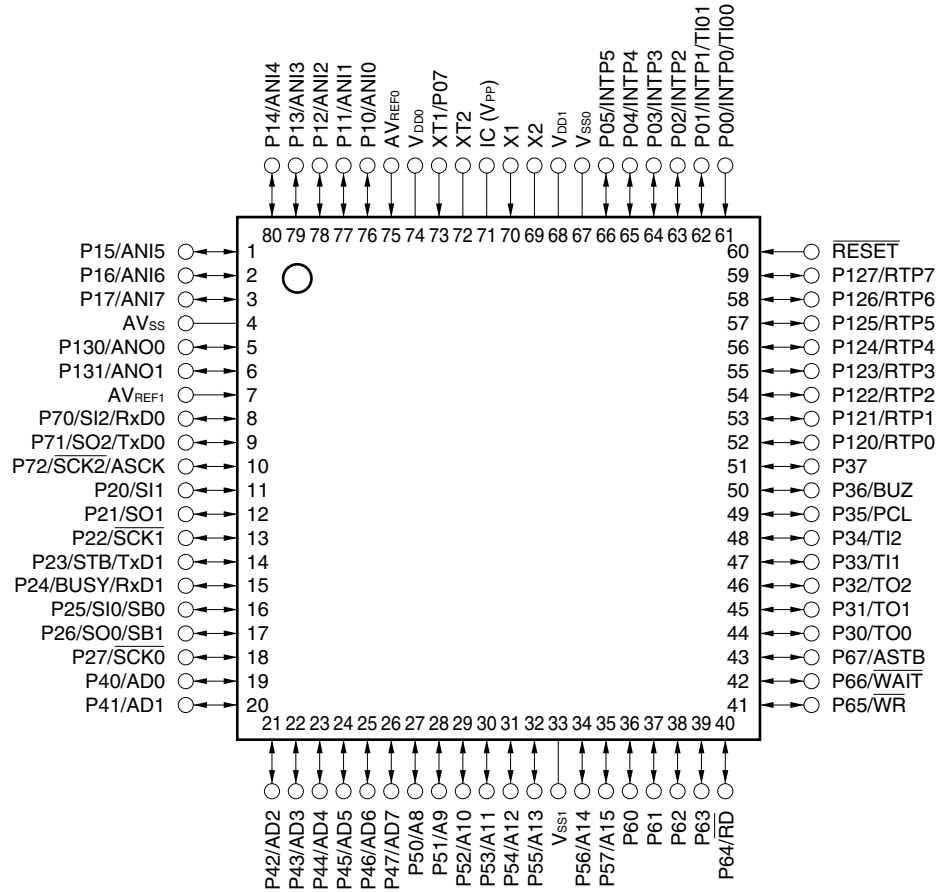
★ 1.4 Pin Configuration (Top View)

• 80-pin plastic QFP (14 × 14)

μPD780053GC-xxx-8BT, 780054GC-xxx-8BT, 780055GC-xxx-8BT,  
 μPD780056GC-xxx-8BT, 780058GC-xxx-8BT, 780058BGC-xxx-8BT,  
 μPD780053GC(A)-xxx-8BT, 780054GC(A)-xxx-8BT, 780055GC(A)-xxx-8BT,  
 μPD780056GC(A)-xxx-8BT, 780058BGC(A)-xxx-8BT, 78F0058GC-8BT

• 80-pin plastic TQFP (fine pitch) (12 × 12)

μPD780053GK-xxx-9EU, 780054GK-xxx-9EU, 780055GK-xxx-9EU,  
 μPD780056GK-xxx-9EU, 780058GK-xxx-9EU, 780058BGK-xxx-9EU, 78F0058GK-9EU



- Cautions**
1. Be sure to connect the IC (Internally Connected) pin to V<sub>SS0</sub> or V<sub>SS1</sub> directly in the normal operating mode.
  2. Connect the AV<sub>SS</sub> pin to V<sub>SS0</sub>.

- Remarks**
1. The pin connection in parentheses is intended for the μPD78F0058.
  2. When the μPD780053, 780054, 780055, 780056, 780058, or 780058B is used in application fields that require reduction of the noise generated from inside the microcontroller, the implementation of noise reduction measures, such as supplying to V<sub>DD0</sub> and V<sub>DD1</sub> individually and connecting V<sub>SS0</sub> and V<sub>SS1</sub> to different ground lines, is recommended.

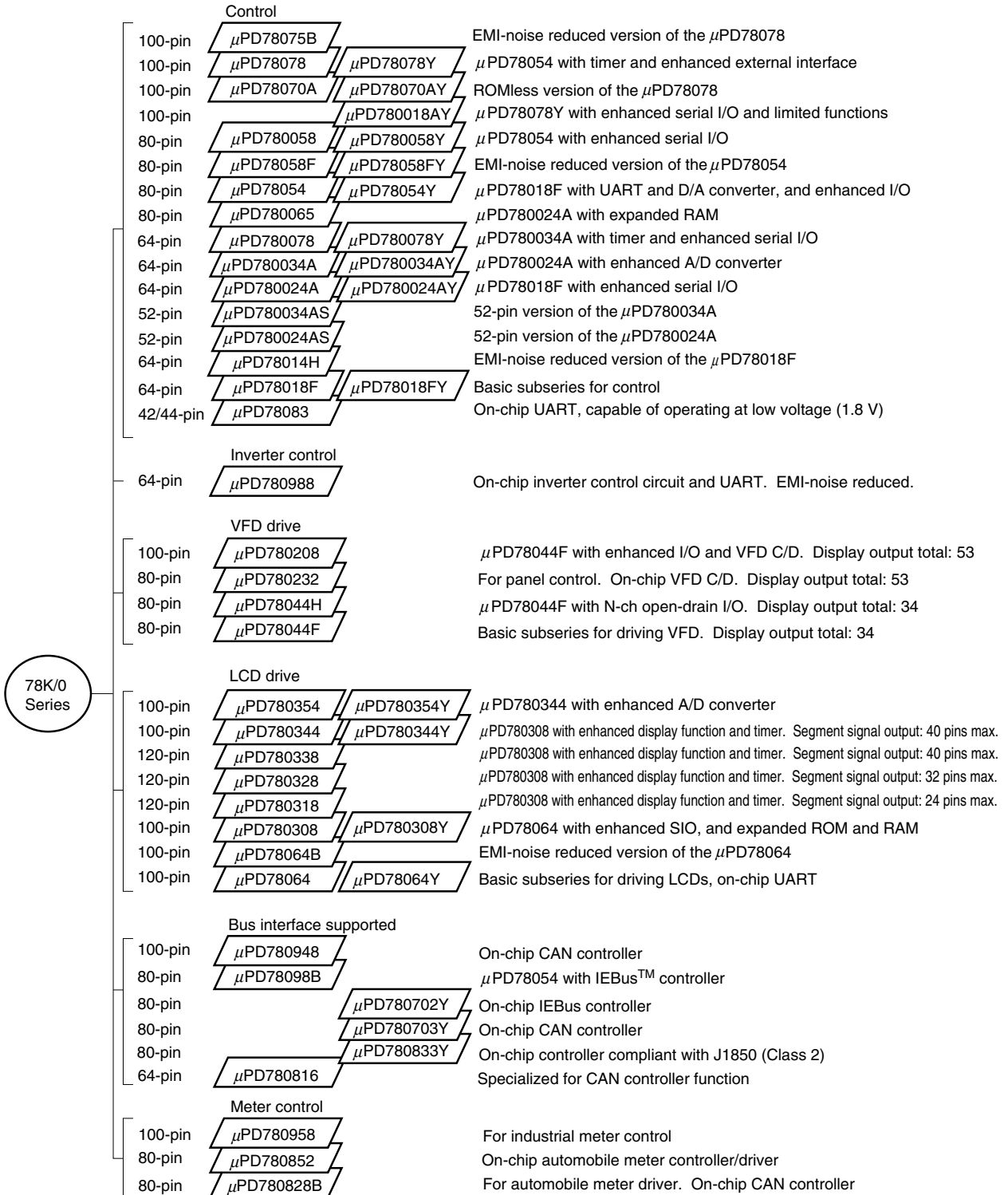
A8 to A15:	Address bus	PCL:	Programmable clock
AD0 to AD7:	Address/data bus	$\overline{RD}$ :	Read strobe
ANI0 to ANI7:	Analog input	$\overline{RESET}$ :	Reset
ANO0, ANO1:	Analog output	RTP0 to RTP7:	Real-time output port
ASCK:	Asynchronous serial clock	RxD0, RxD1:	Receive data
ASTB:	Address strobe	SB0, SB1:	Serial bus
AV <sub>REF0</sub> , AV <sub>REF1</sub> :	Analog reference voltage	$\overline{SCK0}$ to $\overline{SCK2}$ :	Serial clock
AV <sub>SS</sub> :	Analog ground	SI0 to SI2:	Serial input
BUSY:	Busy	SO0 to SO2:	Serial output
BUZ:	Buzzer clock	STB:	Strobe
IC:	Internally connected	TI00, TI01:	Timer input
INTP0 to INTP6:	Interrupt from peripherals	TI1, TI2:	Timer input
P00 to P05, P07:	Port 0	TO0 to TO2:	Timer output
P10 to P17:	Port 1	TxD0, TxD1:	Transmit data
P20 to P27:	Port 2	V <sub>DD0</sub> , V <sub>DD1</sub> :	Power supply
P30 to P37:	Port 3	V <sub>PP</sub> :	Programming power supply
P40 to P47:	Port 4	V <sub>SS0</sub> , V <sub>SS1</sub> :	Ground
P50 to P57:	Port 5	$\overline{WAIT}$ :	Wait
P60 to P67:	Port 6	$\overline{WR}$ :	Write strobe
P70 to P72:	Port 7	X1, X2:	Crystal (main system clock)
P120 to P127:	Port 12	XT1, XT2:	Crystal (subsystem clock)
P130, P131:	Port 13		

★ 1.5 78K/0 Series Lineup

78K/0 Series product lineup is illustrated below. Part numbers in the boxes indicate subseries names.



Y subseries products are compatible with I<sup>2</sup>C bus.



**Remark** VFD (Vacuum Fluorescent Display) is referred to as FIP™ (Fluorescent Indicator Panel) in some documents, but the functions of the two are the same.

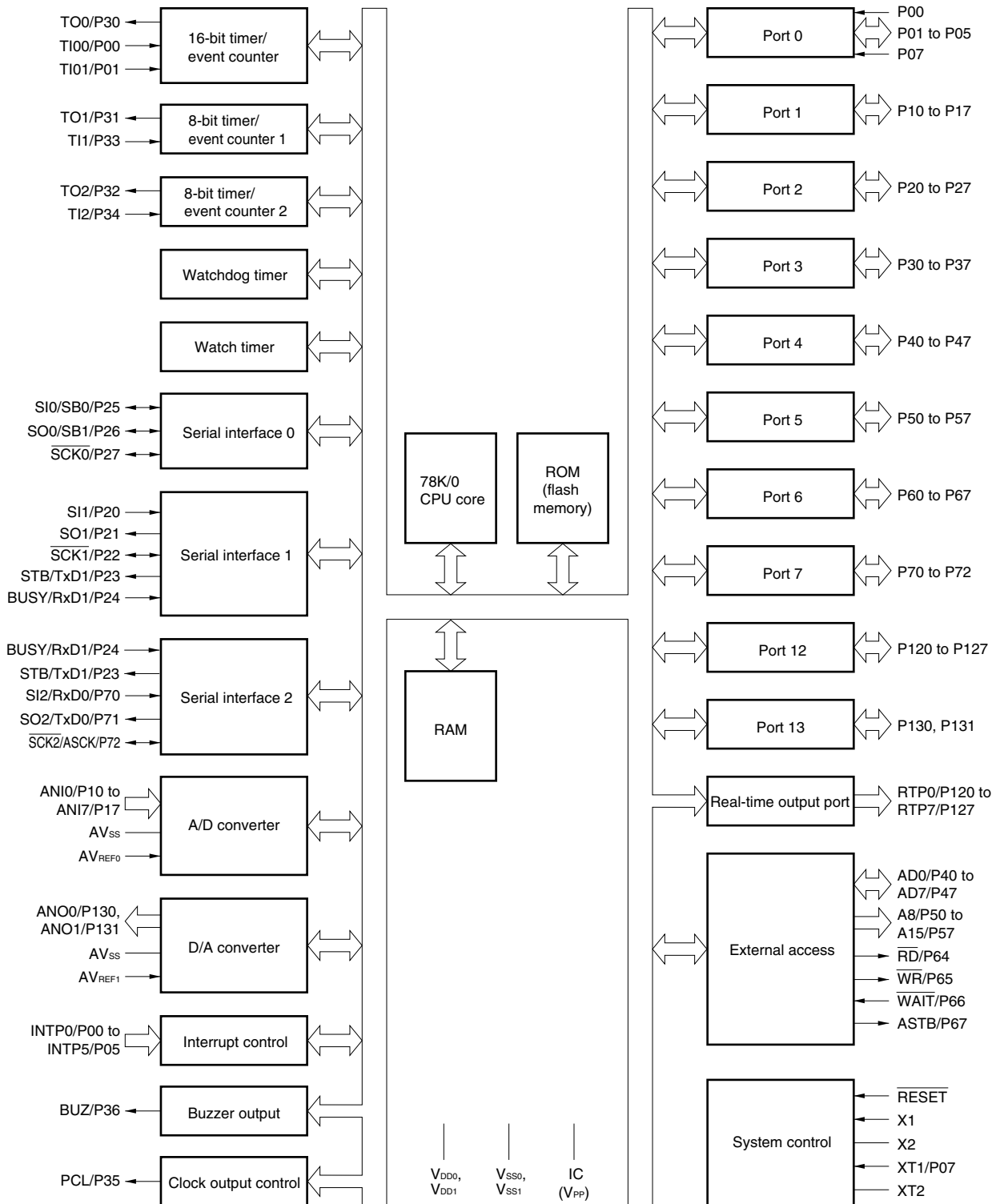
The following lists the main functional differences between subseries products.

- Non-Y subseries

Function Subseries Name	ROM Capacity (Bytes)	Timer				8-Bit	10-Bit	8-Bit	Serial Interface	I/O	V <sub>DD</sub> MIN. Value	External Expansion	
		8-Bit	16-Bit	Watch	WDT	A/D	A/D	D/A					
Control	μPD78075B	32 K to 40 K	4 ch	1 ch	1 ch	1 ch	8 ch	-	2 ch	3 ch (UART: 1 ch)	88	1.8 V	√
	μPD78078	48 K to 60 K									61	2.7 V	
	μPD78070A	-	2 ch	3 ch (time-division UART: 1 ch)	68	1.8 V							
	μPD780058	24 K to 60 K			69	2.7 V							
	μPD78058F	48 K to 60 K	2.0 V										
	μPD78054	16 K to 60 K		-	4 ch (UART: 1 ch)	60	2.7 V						
	μPD780065	40 K to 48 K	2 ch			3 ch (UART: 2 ch)	52	1.8 V					
	μPD780078	48 K to 60 K		1 ch	3 ch (UART: 1 ch)		51	-					
	μPD780034A	8 K to 32 K	8 ch			-	39		-				
	μPD780024A			4 ch	-			53		√			
	μPD780034AS		8 ch			-	53		√				
	μPD780024AS			8 ch	-			53		√			
	μPD78014H		8 ch			-	53		√				
	μPD78018F	8 K to 60 K		1 ch (UART: 1 ch)	33			-		-			
μPD78083	8 K to 16 K	-	-										
Inverter control	μPD780988	16 K to 60 K	3 ch	<b>Note</b>	-	1 ch	-	8 ch	-	3 ch (UART: 2 ch)	47	4.0 V	√
VFD drive	μPD780208	32 K to 60 K	2 ch	1 ch	1 ch	1 ch	8 ch	-	-	2 ch	74	2.7 V	-
	μPD780232	16 K to 24 K	3 ch	-	-		4 ch				40	4.5 V	
	μPD78044H	32 K to 48 K	2 ch	1 ch	1 ch		8 ch			1 ch	68	2.7 V	
	μPD78044F	16 K to 40 K								2 ch			
LCD drive	μPD780354	24 K to 32 K	4 ch	1 ch	1 ch	1 ch	-	8 ch	-	3 ch (UART: 1 ch)	66	1.8 V	-
	μPD780344						8 ch	-					
	μPD780338	48 K to 60 K	3 ch	2 ch			-	10 ch	1 ch	2 ch (UART: 1 ch)	54		
	μPD780328				62								
	μPD780318		70										
	μPD780308	48 K to 60 K	2 ch	1 ch			8 ch	-	-	3 ch (time-division UART: 1 ch)	57	2.0 V	
	μPD78064B	32 K			2 ch (UART: 1 ch)								
μPD78064	16 K to 32 K												
Bus interface supported	μPD780948	60 K	2 ch	2 ch	1 ch	1 ch	8 ch	-	-	3 ch (UART: 1 ch)	79	4.0 V	√
	μPD78098B	40 K to 60 K		1 ch							2 ch	69	2.7 V
	μPD780816	32 K to 60 K	2 ch		12 ch		-	2 ch (UART: 1 ch)	46	4.0 V			
Meter control	μPD780958	48 K to 60 K	4 ch	2 ch	-	1 ch	-	-	-	2 ch (UART: 1 ch)	69	2.2 V	-
Dash-board control	μPD780852	32 K to 40 K	3 ch	1 ch	1 ch	1 ch	5 ch	-	-	3 ch (UART: 1 ch)	56	4.0 V	-
	μPD780828B	32 K to 60 K									59		

**Note** 16-bit timer: 2 channels  
10-bit timer: 1 channel

1.6 Block Diagram



- Remarks**
1. The internal ROM and RAM capacities depend on the product.
  2. The pin connection in parentheses is intended for the μPD78F0058.

1.7 Outline of Function

Item		Part Number						
		μPD780053, 780053(A)	μPD780054, 780054(A)	μPD780055, 780055(A)	μPD780056, 780056(A)	μPD780058B, 780058B(A)	μPD780058	μPD78F0058
Internal memory	ROM	Mask ROM						Flash memory
		24 KB	32 KB	40 KB	48 KB	60 KB	60 KB <sup>Note 1</sup>	
	High-speed RAM	1,024 bytes						
	Buffer RAM	32 bytes						
	Expansion RAM	None				1,024 bytes	1,024 bytes <sup>Note 2</sup>	
Memory space		64 KB						
General-purpose registers		8 bits × 8 × 4 banks						
Minimum instruction execution time		Function to vary minimum instruction execution time incorporated						
	With main system clock selected	0.4 μs/0.8 μs/1.6 μs/3.2 μs/6.4 μs/12.8 μs (@ 5.0 MHz operation)						
	With subsystem clock selected	122 μs (@ 32.768 kHz operation)						
Instruction set		<ul style="list-style-type: none"> <li>• 16-bit operation</li> <li>• Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits)</li> <li>• Bit manipulation (set, reset, test, and boolean operation)</li> <li>• BCD adjust, etc.</li> </ul>						
I/O ports		Total: 68 <ul style="list-style-type: none"> <li>• CMOS input: 2</li> <li>• CMOS I/O: 62</li> <li>• N-ch open-drain I/O: 4</li> </ul>						
A/D converter		8-bit resolution × 8 channels						
	Operating voltage range	V <sub>DD</sub> = 1.8 to 5.5 V				V <sub>DD</sub> = 2.7 to 5.5 V		
D/A converter		8-bit resolution × 2 channels						
	Operating voltage range	V <sub>DD</sub> = 1.8 to 5.5 V				V <sub>DD</sub> = 2.7 to 5.5 V		
Serial interface		<ul style="list-style-type: none"> <li>• 3-wire serial I/O/SBI/2-wire serial I/O mode selection possible: 1 channel</li> <li>• 3-wire serial I/O mode (on-chip max. 32 bytes auto-transmit/receive function): 1 channel</li> <li>• 3-wire serial I/O/UART mode (on-chip time-division transfer function) selectable: 1 channel</li> </ul>						
Timer		<ul style="list-style-type: none"> <li>• 16-bit timer/event counter: 1 channel</li> <li>• 8-bit timer/event counter: 2 channels</li> <li>• Watch timer: 1 channel</li> <li>• Watchdog timer: 1 channel</li> </ul>						
Timer outputs		3: (14-bit PWM output enable: 1)						
Clock output		19.5 kHz, 39.1 kHz, 78.1 kHz, 156 kHz, 313 kHz, 625 kHz, 1.25 MHz, 2.5 MHz, 5.0 MHz (main system clock: @ 5.0 MHz operation) 32.768 kHz (subsystem clock: @ 32.768 kHz operation)						
Buzzer output		1.2 kHz, 2.4 kHz, 4.9 kHz, 9.8 kHz (main system clock: @ 5.0 MHz operation)						

- Notes**
1. The capacity of the flash memory can be changed by using the internal memory switching register (IMS).
  2. The capacity of the internal expansion RAM can be changed by using the internal expansion RAM size switching register (IXS).

Item		Part Number	$\mu$ PD780053, 780053(A)	$\mu$ PD780054, 780054(A)	$\mu$ PD780055, 780055(A)	$\mu$ PD780056, 780056(A)	$\mu$ PD780058B, 780058B(A)	$\mu$ PD780058	$\mu$ PD78F0058	
		Vectored interrupt sources		Maskable	Internal: 13, External: 6					
		Non-maskable	Internal: 1							
		Software	1							
Test input		Internal: 1, External: 1								
Supply voltage		$V_{DD} = 1.8$ to $5.5$ V							$V_{DD} = 2.7$ <sup>Note</sup> to $5.5$ V	
Operating ambient temperature		$T_A = -40$ to $+85^\circ\text{C}$								
Package		<ul style="list-style-type: none"> <li>• 80-pin plastic QFP (<math>14 \times 14</math>)</li> <li>• 80-pin plastic TQFP (fine pitch) (<math>12 \times 12</math>)</li> </ul>								

**Note**  $V_{DD} = 2.2$  V can also be supplied. For details, contact an NEC Electronics sales representative.

The timers are outlined below.

		16-Bit Timer/ Event Counter	8-Bit Timer/Event Counters 1 and 2	Watch Timer	Watchdog Timer
Operating Mode	Interval timer	2 channels <sup>Note 3</sup>	2 channels	1 channel <sup>Note 1</sup>	1 channel <sup>Note 2</sup>
	External event counter	√	√	—	—
Function	Timer output	√	√	—	—
	PWM output	√	—	—	—
	Pulse width measurement	√	—	—	—
	Square-wave output	√	√	—	—
	One-shot pulse output	√	—	—	—
	Interrupt request	√	√	√	√
	Test input	—	—	√	—

**Notes** 1. The watch timer can perform both watch timer and interval timer functions at the same time.

2. The watchdog timer can perform either the watchdog timer function or the interval timer function.

3. When capture/compare registers 00 and 01 (CR00 and CR01) are specified as compare registers.

## 1.8 Mask Options

The mask ROM versions ( $\mu$ PD780053, 780053(A), 780054, 780054(A), 780055, 780055(A), 780056, 780056(A), 780058, 780058B, 780058B(A)) provide pull-up resistor mask options which allow users to specify whether to connect a pull-up resistor to a specific port pin when the user places an order for the device production. Using this mask option when pull-up resistors are required reduces the number of components to add to the device, resulting in board space saving.

The mask options provided in the  $\mu$ PD780058 Subseries are shown in Table 1-1.

**Table 1-1. Mask Options of Mask ROM Versions**

Pin Names	Mask Options
P60 to P63	Pull-up resistor connection can be specified in 1-bit units.

## 1.9 Differences Between Standard Model and (A) Model

The (A) models of the  $\mu$ PD780058 Subseries ( $\mu$ PD780053(A), 780054(A), 780055(A), 780056(A), and 780058B(A)) have improved reliability by increasing the check items from the standard model ( $\mu$ PD780053, 780054, 780055, 780056, and 780058B). The functions and electrical characteristics of the (A) model are the same as those of the standard model.

**Table 1-2. Differences Between Standard Model and (A) Model**

Product Name Item	Standard Model	(A) Model
Quality grade	Standard (for general-purpose electronic systems)	Special (for high-reliability electronic systems)



## CHAPTER 2 OUTLINE ( $\mu$ PD780058Y SUBSERIES)

### 2.1 Features

- ★ ○ On-chip high-capacity ROM and RAM

Part Number \ Item	Program Memory		Data Memory		
	Mask ROM	Flash Memory	Internal High-Speed RAM	Internal Buffer RAM	Internal Expansion RAM
$\mu$ PD780053Y, 780053Y(A)	24 KB	—	1,024 bytes	32 bytes	None
$\mu$ PD780054Y, 780054Y(A)	32 KB	—			
$\mu$ PD780055Y, 780055Y(A)	40 KB	—			
$\mu$ PD780056Y, 780056Y(A)	48 KB	—			
$\mu$ PD780058BY, 780058BY(A)	60 KB	—			
$\mu$ PD78F0058Y	—	60 KB <sup>Note 1</sup>			1,024 bytes <sup>Note 2</sup>

- Notes**
1. The capacity of flash memory can be changed by means of the internal memory size switching register (IMS).
  2. The capacity of internal high-speed RAM can be changed by means of the internal expansion RAM size switching register (IXS).

- External memory expansion space: 64 KB
- Minimum instruction execution time changeable from high-speed (0.4  $\mu$ s: Main system clock 5.0 MHz operation) to ultra-low speed (122  $\mu$ s: Subsystem clock 32.768 kHz operation)
- Instruction set suited to system control
  - Bit manipulation possible in all address spaces
  - Multiple and divide instructions
- I/O ports: 68 (N-ch open-drain: 4)
- ★ ○ 8-bit resolution A/D converter: 8 channels ( $V_{DD} = 1.8$  to 5.5 V)
- ★ ○ 8-bit resolution D/A converter: 2 channels ( $V_{DD} = 1.8$  to 5.5 V)
- Serial interface: 3 channels
  - 3-wire serial I/O/2-wire serial I/O/I<sup>2</sup>C bus mode: 1 channel
  - 3-wire serial I/O mode (on-chip automatic transmit/receive function): 1 channel
  - 3-wire serial I/O/UART mode (on-chip time-division transfer function): 1 channel
- Timer: 5 channels
  - 16-bit timer/event counter: 1 channel
  - 8-bit timer/event counter: 2 channels
  - Watch timer: 1 channel
  - Watchdog timer: 1 channel
- Vectored interrupt sources: 21
- Test inputs: 2
- Two types of on-chip clock oscillators (main system clock and subsystem clock)
- ★ ○ Supply voltage:  $V_{DD} = 1.8$  to 5.5 V (mask ROM version)
- ★  $V_{DD} = 2.7$ <sup>Note</sup> to 5.5 V ( $\mu$ PD78F0058Y)

**Note**  $V_{DD} = 2.2$  V can also be supplied to the  $\mu$ PD78F0058Y. For details, contact an NEC Electronics sales representative.

## 2.2 Applications

Car audio systems, cellular phones, pagers, printers, AV equipment, cameras, PPCs, vending machines, car electrical components, etc.

## ★ 2.3 Ordering Information

Part Number	Package	Internal ROM
$\mu$ PD780053YGC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780053YGK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780054YGC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780054YGK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780055YGC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780055YGK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780056YGC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
$\mu$ PD780056YGK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
★ $\mu$ PD780058BYGC-xxx-8BT	80-pin plastic QFP (14 × 14)	Mask ROM
★ $\mu$ PD780058BYGK-xxx-9EU	80-pin plastic TQFP (fine pitch) (12 × 12)	Mask ROM
$\mu$ PD780053YGC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780054YGC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780055YGC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780056YGC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD780058BYGC(A)-xxx-8BT	80-pin plastic QFP (14 × 14)	Special
$\mu$ PD78F0058YGC-8BT <b>Note</b>	80-pin plastic QFP (14 × 14)	Flash-memory
$\mu$ PD78F0058YGK-9EU <b>Note</b>	80-pin plastic TQFP (fine pitch) (12 × 12)	Flash-memory

**Remark** xxx indicates ROM code suffix.

For details of the quality grades and their applications, see **Quality Grades on NEC Electronics Semiconductor Devices** (Document No.: C11531E).

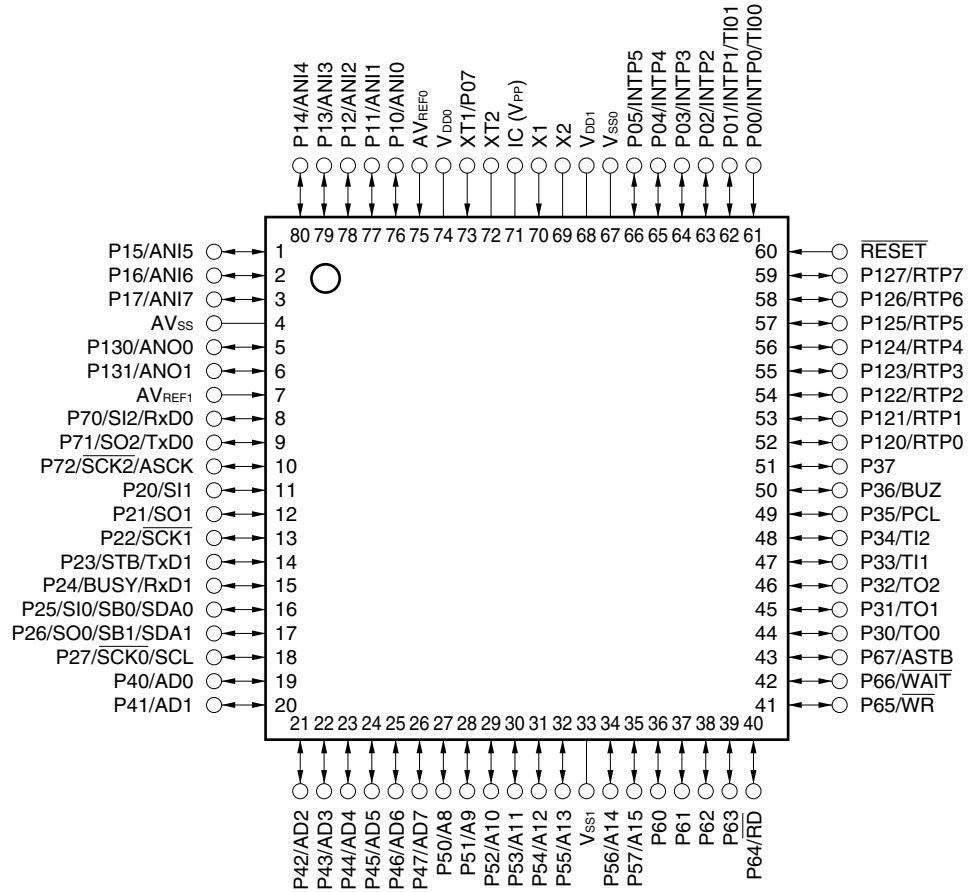
★ 2.4 Pin Configuration (Top View)

• 80-pin plastic QFP (14 × 14)

$\mu$ PD780053YGC-xxx-8BT, 780054YGC-xxx-8BT, 780055YGC-xxx-8BT,  
 $\mu$ PD780056YGC-xxx-8BT, 780058BYGC-xxx-8BT, 780053YGC(A)-xxx-8BT,  
 $\mu$ PD780054YGC(A)-xxx-8BT, 780055YGC(A)-xxx-8BT, 780056YGC(A)-xxx-8BT,  
 $\mu$ PD780058BYGC(A)-xxx-8BT, 78F0058YGC-8BT

• 80-pin plastic TQFP (fine pitch) (12 × 12)

$\mu$ PD780053YGK-xxx-9EU, 780054YGK-xxx-9EU, 780055YGK-xxx-9EU,  
 $\mu$ PD780056YGK-xxx-9EU, 780058BYGK-xxx-9EU, 78F0058YGK-9EU



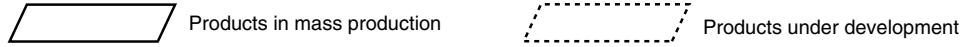
- Cautions**
1. Be sure to connect the IC (Internally Connected) pin to  $V_{SS0}$  directly in the normal operating mode.
  2. Connect the  $AV_{SS}$  pin to  $V_{SS0}$ .

- Remarks**
1. The pin connection in parentheses is intended for the  $\mu$ PD78F0058Y.
  2. When the  $\mu$ PD780053Y, 780054Y, 780055Y, 780056Y, or 780058BY is used in application fields that require reduction of the noise generated from inside the microcontroller, the implementation of noise reduction measures, such as supplying to  $V_{DD0}$  and  $V_{DD1}$  individually and connecting  $V_{SS0}$  and  $V_{SS1}$  to different ground lines, is recommended.

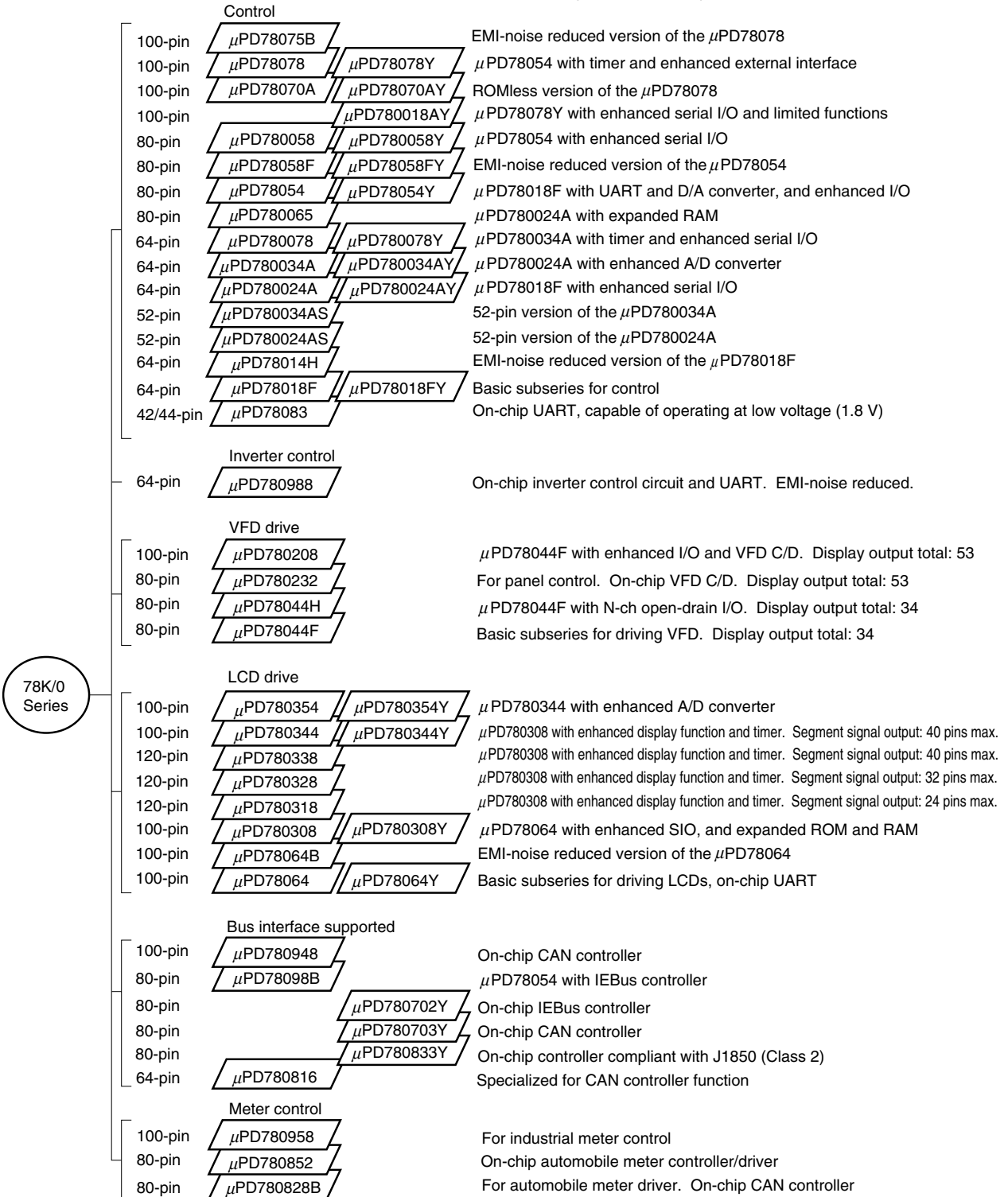
A8 to A15:	Address bus	$\overline{\text{RD}}$ :	Read strobe
AD0 to AD7:	Address/data bus	$\overline{\text{RESET}}$ :	Reset
ANI0 to ANI7:	Analog input	RTP0 to RTP7:	Real-time output port
ANO0, ANO1:	Analog output	RxD0, RxD1:	Receive data
ASCK:	Asynchronous serial clock	SB0, SB1:	Serial bus
ASTB:	Address strobe	$\overline{\text{SCK0}}$ to $\overline{\text{SCK2}}$ :	Serial clock
AVREF0, AVREF1:	Analog reference voltage	SCL:	Serial clock
AVss:	Analog ground	SDA0, SDA1:	Serial data
BUSY:	Busy	SI0 to SI2:	Serial input
BUZ:	Buzzer clock	SO0 to SO2:	Serial output
IC:	Internally connected	STB:	Strobe
INTP0 to INTP6:	Interrupt from peripherals	TI00, TI01:	Timer input
P00 to P05, P07:	Port 0	TI1, TI2:	Timer input
P10 to P17:	Port 1	TO0 to TO2:	Timer output
P20 to P27:	Port 2	TxD0, TxD1:	Transmit data
P30 to P37:	Port 3	VDD0, VDD1:	Power supply
P40 to P47:	Port 4	VPP:	Programming power supply
P50 to P57:	Port 5	VSS0, VSS1:	Ground
P60 to P67:	Port 6	$\overline{\text{WAIT}}$ :	Wait
P70 to P72:	Port 7	$\overline{\text{WR}}$ :	Write strobe
P120 to P127:	Port 12	X1, X2:	Crystal (main system clock)
P130, P131:	Port 13	XT1, XT2:	Crystal (subsystem clock)
PCL:	Programmable clock		

★ 2.5 78K/0 Series Lineup

78K/0 Series product lineup is illustrated below. Part numbers in the boxes indicate subseries names.



Y subseries products are compatible with I<sup>2</sup>C bus.



**Remark** VFD (Vacuum Fluorescent Display) is referred to as FIP (Fluorescent Indicator Panel) in some documents, but the functions of the two are the same.

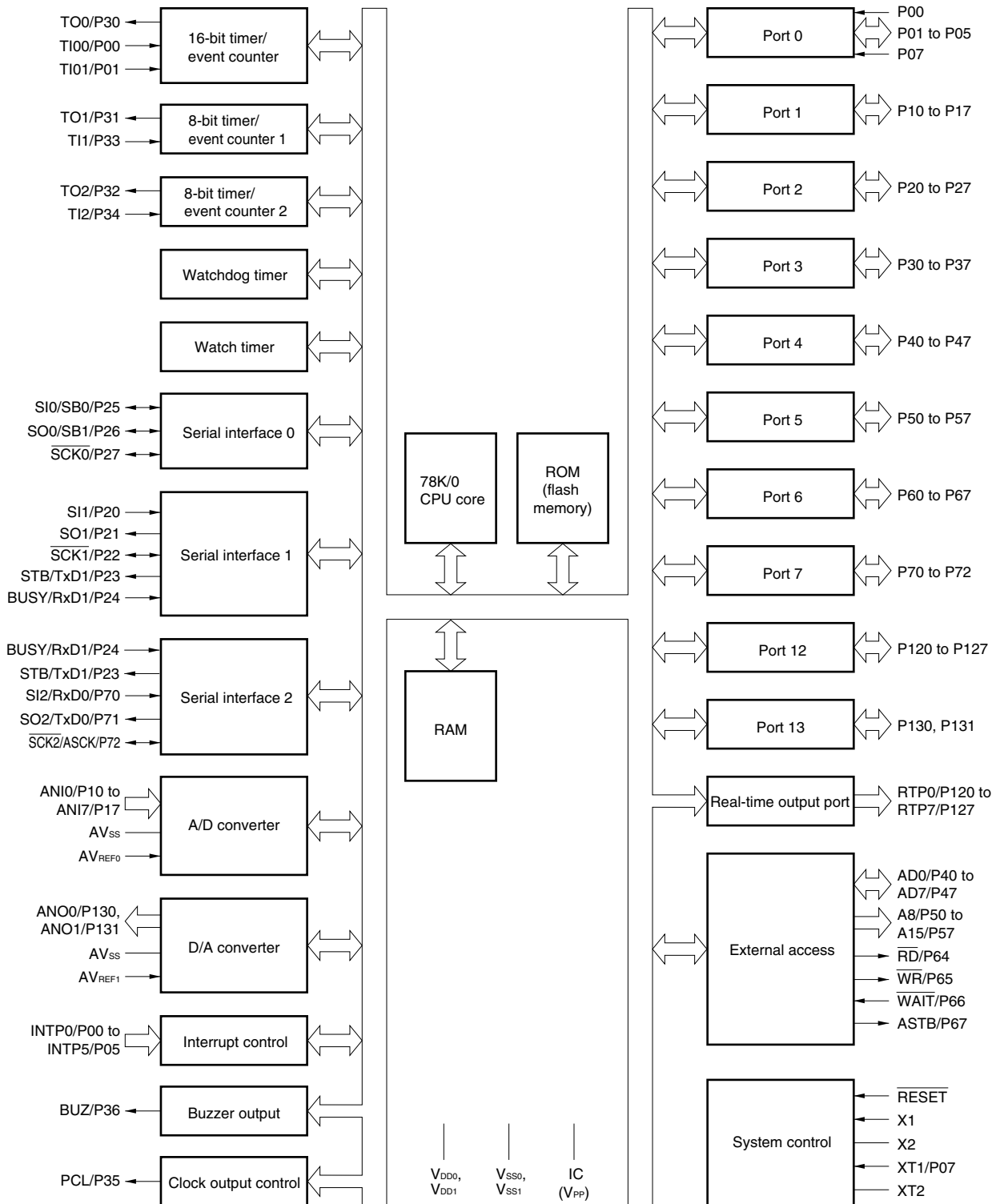
Major functional differences among the Y subseries are shown below.

- Y subseries

Subseries Name	Function	ROM Capacity (Bytes)	Timer				8-Bit	10-Bit	8-Bit	Serial Interface	I/O	V <sub>DD</sub> MIN. Value	External Expansion	
			8-Bit	16-Bit	Watch	WDT	A/D	A/D	D/A					
Control	$\mu$ PD78078Y	48 K to 60 K	4 ch	1 ch	1 ch	1 ch	8 ch	-	2 ch	3 ch (UART: 1 ch, I <sup>2</sup> C: 1 ch)	88	1.8 V	√	
	$\mu$ PD78070AY	-								61	2.7 V			
	$\mu$ PD780018AY	48 K to 60 K								-	3 ch (I <sup>2</sup> C: 1 ch)	88		
	$\mu$ PD780058Y	24 K to 60 K	2 ch						2 ch	3 ch (time-division UART: 1 ch, I <sup>2</sup> C: 1 ch)	68	1.8 V		
	$\mu$ PD78058FY	48 K to 60 K								3 ch (UART: 1 ch, I <sup>2</sup> C: 1 ch)	69	2.7 V		
	$\mu$ PD78054Y	16 K to 60 K									2.0 V			
	$\mu$ PD780078Y	48 K to 60 K		2 ch			-	8 ch	-	4 ch (UART: 2 ch, I <sup>2</sup> C: 1 ch)	52	1.8 V		
	$\mu$ PD780034AY	8 K to 32 K		1 ch						3 ch (UART: 1 ch, I <sup>2</sup> C: 1 ch)	51			
	$\mu$ PD780024AY						8 ch	-						
$\mu$ PD78018FY	8 K to 60 K								2 ch (I <sup>2</sup> C: 1 ch)	53				
LCD drive	$\mu$ PD780354Y	24 K to 32 K	4 ch	1 ch	1 ch	1 ch	-	8 ch	-	4 ch (UART: 1 ch, I <sup>2</sup> C: 1 ch)	66	1.8 V	-	
	$\mu$ PD780344Y						8 ch	-						
	$\mu$ PD780308Y	48 K to 60 K	2 ch							3 ch (time-division UART: 1 ch, I <sup>2</sup> C: 1 ch)	57	2.0 V		
	$\mu$ PD78064Y	16 K to 32 K								2 ch (UART: 1 ch, I <sup>2</sup> C: 1 ch)				
Bus interface supported	$\mu$ PD780701Y	60 K	3 ch	2 ch	1 ch	1 ch	16 ch	-	-	4 ch (UART: 1 ch, I <sup>2</sup> C: 1 ch)	67	3.5 V	-	
	$\mu$ PD780703Y													
	$\mu$ PD780833Y										65	4.5 V		

**Remark** Functions other than the serial interface are common to both the Y and non-Y subseries.

## 2.6 Block Diagram



- Remarks**
1. The internal ROM and RAM capacities depend on the product.
  2. The pin connection in parentheses is intended for the  $\mu$ PD78F0058Y.

2.7 Outline of Functions

★		Part Number	μPD780053Y, 780053Y(A)	μPD780054Y, 780054Y(A)	μPD780055Y, 780055Y(A)	μPD780056Y, 780056Y(A)	μPD780058BY, 780058BY(A)	μPD78F0058Y	
Internal memory	ROM	Mask ROM						Flash memory	
		24 KB	32 KB	40 KB	48 KB	60 KB	60 KB <sup>Note 1</sup>		
	High-speed RAM	1,024 bytes							
	Buffer RAM	32 bytes							
	Expansion RAM	None					1,024 bytes	1,024 bytes <sup>Note 2</sup>	
Memory space		64 KB							
General-purpose registers		8 bits × 8 × 4 banks							
Minimum instruction execution time		Function to vary minimum instruction execution time incorporated							
	With main system clock selected	0.4 μs/0.8 μs/1.6 μs/3.2 μs/6.4 μs/12.8 μs (@ 5.0 MHz operation)							
	With subsystem clock selected	122 μs (@ 32.768 kHz operation)							
Instruction set		<ul style="list-style-type: none"> <li>• 16-bit operation</li> <li>• Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits)</li> <li>• Bit manipulation (set, reset, test, and boolean operation)</li> <li>• BCD adjust, etc.</li> </ul>							
I/O ports		Total: 68 <ul style="list-style-type: none"> <li>• CMOS input: 2</li> <li>• CMOS I/O: 62</li> <li>• N-ch open-drain I/O: 4</li> </ul>							
A/D converter		8-bit resolution × 8 channels							
★	Operating voltage range	V <sub>DD</sub> = 1.8 to 5.5 V					V <sub>DD</sub> = 2.7 to 5.5 V		
D/A converter		8-bit resolution × 2 channels							
★	Operating voltage range	V <sub>DD</sub> = 1.8 to 5.5 V					V <sub>DD</sub> = 2.7 to 5.5 V		
Serial interface		<ul style="list-style-type: none"> <li>• 3-wire serial I/O/SBI/2-wire serial I/O mode selection possible: 1 channel</li> <li>• 3-wire serial I/O mode (on-chip max. 32 bytes auto-transmit/receive function): 1 channel</li> <li>• 3-wire serial I/O/UART mode (on-chip time-division transfer function) selectable: 1 channel</li> </ul>							
Timer		<ul style="list-style-type: none"> <li>• 16-bit timer/event counter: 1 channel</li> <li>• 8-bit timer/event counter: 2 channels</li> <li>• Watch timer: 1 channel</li> <li>• Watchdog timer: 1 channel</li> </ul>							
Timer outputs		3: (14-bit PWM output enable: 1)							
Clock output		19.5 kHz, 39.1 kHz, 78.1 kHz, 156 kHz, 313 kHz, 625 kHz, 1.25 MHz, 2.5 MHz, 5.0 MHz (main system clock: @ 5.0 MHz operation) 32.768 kHz (subsystem clock: @ 32.768 kHz operation)							
Buzzer output		1.2 kHz, 2.4 kHz, 4.9 kHz, 9.8 kHz (main system clock: @ 5.0 MHz operation)							

- Notes**
1. The capacity of the flash memory can be changed by using the internal memory switching register (IMS).
  2. The capacity of the internal expansion RAM can be changed by using the internal expansion RAM size switching register (IXS).



★	Item	Part Number	$\mu$ PD780053Y, 780053Y(A)	$\mu$ PD780054Y, 780054Y(A)	$\mu$ PD780055Y, 780055Y(A)	$\mu$ PD780056Y, 780056Y(A)	$\mu$ PD780058BY, 780058BY(A)	$\mu$ PD78F0058Y
	Vectored interrupt sources	Maskable	Internal: 13, External: 6					
		Non-maskable	Internal: 1					
		Software	1					
	Test input		Internal: 1, External: 1					
★	Supply voltage		$V_{DD} = 1.8$ to $5.5$ V					$V_{DD} = 2.7$ Note to $5.5$ V
	Operating ambient temperature		$T_A = -40$ to $+85^\circ\text{C}$					
★	Package		<ul style="list-style-type: none"> <li>• 80-pin plastic QFP (<math>14 \times 14</math>)</li> <li>• 80-pin plastic TQFP (Fine pitch) (<math>12 \times 12</math>)</li> </ul>					

**Note**  $V_{DD} = 2.2$  V can also be supplied. For details, contact an NEC Electronics sales representative.

The timers are outlined below.

		16-Bit Timer/ Event Counter	8-Bit Timer/Event Counters 1 and 2	Watch Timer	Watchdog Timer
Operating Mode	Interval timer	2 channels <sup>Note 3</sup>	2 channels	1 channel <sup>Note 1</sup>	1 channel <sup>Note 2</sup>
	External event counter	√	√	—	—
Function	Timer output	√	√	—	—
	PWM output	√	—	—	—
	Pulse width measurement	√	—	—	—
	Square-wave output	√	√	—	—
	One-shot pulse output	√	—	—	—
	Interrupt request	√	√	√	√
	Test input	—	—	√	—

- Notes**
1. The watch timer can perform both watch timer and interval timer functions at the same time.
  2. The watchdog timer can perform either the watchdog timer function or the interval timer function.
  3. When capture/compare registers 00 and 01 (CR00 and CR01) are specified as compare registers.

## 2.8 Mask Options

The mask ROM versions ( $\mu$ PD780053Y, 780053Y(A), 780054Y, 780054Y(A), 780055Y, 780055Y(A), 780056Y, 780056Y(A), 780058BY, 780058BY(A)) provide pull-up resistor mask options which allow users to specify whether to connect a pull-up resistor to a specific port pin when the user places an order for the device production. Using this mask option when pull-up resistors are required reduces the number of components to add to the device, resulting in board space saving.

The mask options provided in the  $\mu$ PD780058Y Subseries are shown in Table 2-1.

**Table 2-1. Mask Options of Mask ROM Versions**

Pin Names	Mask Options
P60 to P63	Pull-up resistor connection can be specified in 1-bit units.

## 2.9 Differences Between Standard Model and (A) Model

The (A) models of the  $\mu$ PD780058Y Subseries ( $\mu$ PD780053Y(A), 780054Y(A), 780055Y(A), 780056Y(A), and 780058BY(A)) have improved reliability by increasing the check items from the standard model ( $\mu$ PD780053Y, 780054Y, 780055Y, 780056Y, and 780058BY). The functions and electrical characteristics of the (A) model are the same as those of the standard model.

**Table 2-2. Differences Between Standard Model and (A) Model**

Product Name Item	Standard Model	(A) Model
Quality grade	Standard (for general-purpose electronic systems)	Special (for high-reliability electronic systems)

## CHAPTER 3 PIN FUNCTIONS (μPD780058 SUBSERIES)

### 3.1 Pin Function List

#### (1) Port pins (1/2)

Pin Name	I/O	Function		After Reset	Alternate Function
P00	Input	Port 0 7-bit I/O port	Input only	Input	INTP0/TI00
P01	I/O		Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	INTP1/TI01
P02					INTP2
P03					INTP3
P04					INTP4
P05					INTP5
P07 <sup>Note 1</sup>	Input		Input only	Input	XT1
P10 to P17	I/O	Port 1 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software. <sup>Note 2</sup>		Input	ANI0 to ANI7
P20	I/O	Port 2 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	SI1	
P21				SO1	
P22				SCK1	
P23				STB/TxD1	
P24				BUSY/RxD1	
P25				SI0/SB0	
P26				SO0/SB1	
P27				SCK0	
P30	I/O	Port 3 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	TO0	
P31				TO1	
P32				TO2	
P33				TI1	
P34				TI2	
P35				PCL	
P36				BUZ	
P37				—	

- Notes**
1. When the P07/XT1 pin is used as an input port, set bit 6 (FRC) of the processor clock control register (PCC) to 1 (do not use the feedback resistor incorporated in the subsystem clock oscillator).
  2. When pins P10/ANI0 to P17/ANI7 are used as an analog input of the A/D converter, set port 1 to the input mode. In this case, any connected on-chip pull-up resistors are automatically disabled.

(1) Port pins (2/2)

Pin Name	I/O	Function		After Reset	Alternate Function	
P40 to P47	I/O	Port 4 8-bit I/O port Input/output can be specified in 8-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software. The test input flag (KRIF) is set to 1 by falling edge detection.		Input	AD0 to AD7	
P50 to P57	I/O	Port 5 8-bit I/O port LEDs can be driven directly. Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	A8 to A15	
P60	I/O	Port 6 8-bit I/O port Input/output can be specified in 1-bit units.	N-ch open-drain I/O port On-chip pull-up resistors can be specified by mask option. (Mask ROM version only). LEDs can be driven directly.	Input	—	
P61					If used as an input port, an on-chip pull-up resistor can be connected by setting software.	$\overline{\text{RD}}$
P62						$\overline{\text{WR}}$
P63						$\overline{\text{WAIT}}$
P64			ASTB			
P65						
P66						
P67						
P70	I/O	Port 7 3-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	SI2/RxD0	
P71			SO2/TxD0			
P72			$\overline{\text{SCK2/ASCK}}$			
P120 to P127	I/O	Port 12 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	RTP0 to RTP7	
P130 to P131	I/O	Port 13 2-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	ANO0 to ANO1	

(2) Non-port pins (1/2)

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request inputs with specifiable valid edges (rising edge, falling edge, both rising and falling edges).	Input	P00/TI00
INTP1				P01/TI01
INTP2				P02
INTP3				P03
INTP4				P04
INTP5				P05
SI0	Input	Serial interface serial data input	Input	P25/SB0
SI1				P20
SI2				P70/RxD
SO0	Output	Serial interface serial data output	Input	P26/SB1
SO1				P21
SO2				P71/TxD
SB0	I/O	Serial interface serial data input/output	Input	P25/SI0
SB1				P26/SO0
SCK0	I/O	Serial interface serial clock input/output	Input	P27
SCK1				P22
SCK2				P72/ASCK
STB	Output	Serial interface automatic transmit/receive strobe output	Input	P23/TxD1
BUSY	Input	Serial interface automatic transmit/receive busy input	Input	P24/RxD1
RxD0	Input	Asynchronous serial interface serial data input	Input	P70/SI2
RxD1				P24/BUSY
TxD0	Output	Asynchronous serial interface serial data output	Input	P71/SO2
TxD1				P23/STB
ASCK	Input	Asynchronous serial interface serial clock input	Input	P72/SCK2
TI00	Input	External count clock input to 16-bit timer (TM0)	Input	P00/INTP0
TI01		Capture trigger signal input to capture register (CR00)		P01/INTP1
TI1		External count clock input to 8-bit timer (TM1)		P33
TI2		External count clock input to 8-bit timer (TM2)		P34
TO0	Output	16-bit timer (TM0) output (also used for 14-bit PWM output)	Input	P30
TO1		8-bit timer (TM1) output		P31
TO2		8-bit timer (TM2) output		P32
PCL	Output	Clock output (for main system clock and subsystem clock trimming)	Input	P35
BUZ	Output	Buzzer output	Input	P36
RTP0 to RTP7	Output	Real-time output port outputting data in synchronization with trigger	Input	P120 to P127

## (2) Non-port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
AD0 to AD7	I/O	Lower address/data bus when expanding memory externally	Input	P40 to P47
A8 to A15	Output	Higher address bus when expanding memory externally	Input	P50 to P57
$\overline{\text{RD}}$	Output	Strobe signal output for read operation from external memory	Input	P64
$\overline{\text{WR}}$		Strobe signal output for write operation to external memory		P65
$\overline{\text{WAIT}}$	Input	Wait insertion when accessing external memory	Input	P66
ASTB	Output	Strobe output externally latching address information output to ports 4 and 5 to access external memory	Input	P67
ANI0 to ANI7	Input	A/D converter analog input	Input	P10 to P17
ANO0, ANO1	Output	D/A converter analog output	Input	P130, P131
AV <sub>REF0</sub>	Input	A/D converter reference voltage input (also functions as analog power supply)	—	—
AV <sub>REF1</sub>	Input	D/A converter reference voltage input	—	—
AV <sub>SS</sub>	—	A/D converter, D/A converter ground potential. Use the same potential as V <sub>SS0</sub> .	—	—
$\overline{\text{RESET}}$	Input	System reset input	—	—
X1	Input	Crystal connection for main system clock oscillation	—	—
X2	—		—	—
XT1	Input	Crystal connection for subsystem clock oscillation	Input	P07
XT2	—		—	—
V <sub>DD0</sub>	—	Positive power supply for ports	—	—
V <sub>SS0</sub>	—	Ground potential for ports	—	—
V <sub>DD1</sub>	—	Positive power supply (except ports and analog block)	—	—
V <sub>SS1</sub>	—	Ground potential (except ports and analog block)	—	—
V <sub>PP</sub>	—	High-voltage application for program write/verify.	—	—
IC	—	Internally connected. Connect directly to V <sub>SS0</sub> .	—	—

## 3.2 Description of Pin Functions

### 3.2.1 P00 to P05, P07 (Port 0)

P00 to P05 and P07 function as a 7-bit I/O port. Besides serving as I/O port pins, they also function as an external interrupt request input, an external count clock input to the timer, a capture trigger signal input, and crystal connection for subsystem oscillation.

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

#### (1) Port mode

P00 and P07 function as input-only port pins and P01 to P05 function as I/O port pins.

P01 to P05 can be specified as input or output in 1-bit units using port mode register 0 (PM0). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

#### (2) Control mode

In this mode, P00 to P05 and P07 function as an external interrupt request input, an external count clock input to the timer, and crystal connection for subsystem clock oscillation.

##### (a) INTP0 to INTP5

INTP0 to INTP5 are external interrupt request input pins for which valid edges can be specified (rising edge, falling edge, and both rising and falling edges). INTP0 or INTP1 become 16-bit timer/event counter capture trigger signal input pins with a valid edge input.

##### (b) TI00

This is a pin for inputting the external count clock to the 16-bit timer/event counter.

##### (c) TI01

This is a pin for inputting the capture trigger signal to the capture register (CR00) of the 16-bit timer/event counter.

##### (d) XT1

This is a crystal connection pin for subsystem clock oscillation.

### 3.2.2 P10 to P17 (Port 1)

P10 to P17 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as an A/D converter analog inputs.

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

#### (1) Port mode

P10 to P17 function as an 8-bit I/O port.

They can be specified as input or output in 1-bit units using port mode register 1 (PM1). When they are used as input port pins, on-chip pull-up resistor can be connected to them using pull-up resistor option register L (PUOL).

#### (2) Control mode

P10 to P17 function as A/D converter analog input pins (ANI0 to ANI7). On-chip pull-up resistors are automatically disabled when these pins are specified as analog inputs.

### 3.2.3 P20 to P27 (Port 2)

P20 to P27 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as data input/output to/from the serial interface, clock input/output, automatic transmit/receive busy input, and strobe output.

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

#### (1) Port mode

P20 to P27 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 2 (PM2). When they are used as input ports, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

#### (2) Control mode

P20 to P27 function as serial interface data input/output, clock input/output, automatic transmit/receive busy input, and strobe output.

##### (a) SI0, SI1, SO0, SO1

These are serial data I/O pins of the serial interface.

##### (b) $\overline{\text{SCK0}}$ , $\overline{\text{SCK1}}$

These are serial clock I/O pins of the serial interface.

##### (c) SB0, SB1

These are NEC Electronics standard serial bus interface I/O pins.

##### (d) BUSY

This is an automatic transmit/receive busy input pin of the serial interface.

##### (e) STB

This is an automatic transmit/receive strobe output pin of the serial interface.

##### (f) RxD1, TxD1

These are serial interface serial data I/O pins of the asynchronous serial interface.

**Caution** When P20 to P27 are used as serial interface pins, the I/O and output latches must be set according to the function the user requires. For the setting, see Figure 16-4 Format of Serial Operation Mode Register 0, Figure 18-3 Format of Serial Operation Mode Register 1, and Table 19-2 Serial Interface Channel 2 Operating Mode Settings.



### 3.2.4 P30 to P37 (Port 3)

P30 to P37 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as timer input/output, clock output and buzzer output.

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

#### (1) Port mode

P30 to P37 function as an 8-bit I/O port. They can be specified as an input or output in 1-bit units using port mode register 3 (PM3). When they are used as input port pins, on-chip pull-up resistors can be connected to then using pull-up resistor option register L (PUOL).

#### (2) Control mode

P30 to P37 function as timer input/output, clock output, and buzzer output.

##### (a) TI1 and TI2

These are pins for inputting the external count clock to the 8-bit timer/event counter.

##### (b) TO0 to TO2

These are timer output pins.

##### (c) PCL

This is a clock output pin.

##### (d) BUZ

This is a buzzer output pin.

### 3.2.5 P40 to P47 (Port 4)

P40 to P47 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as an address/data bus.

The test input flag (KRIF) can be set to 1 by detecting a falling edge.

The following operating modes can be specified in 8-bit units.

#### (1) Port mode

P40 to P47 using function as an 8-bit I/O port. They can be specified for as input or output in 8-bit units using the internal memory expansion mode register (MM). When they are used as an input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

#### (2) Control mode

P40 to P47 function as the lower address/data bus pins (AD0 to AD7) in external memory expansion mode. When these pins are used as an address/data bus, on-chip pull-up resistors are automatically disabled.

### 3.2.6 P50 to P57 (Port 5)

P50 to P57 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as an address bus. P50 to P57 can drive LEDs directly.

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

#### (1) Port mode

P50 to P57 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 5 (PM5). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

#### (2) Control mode

P50 to P57 function as the higher address bus pins (A8 to A15) in external memory expansion mode. When these pins are used as an address bus, on-chip pull-up resistors are automatically disabled.

### 3.2.7 P60 to P67 (Port 6)

P60 to P67 function as an 8-bit I/O port. Besides serving as I/O port pins, they are also used for control in external memory expansion mode. P60 to P63 can drive LEDs directly.

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

#### (1) Port mode

P60 to P67 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 6 (PM6).

P60 to P63 are N-ch open-drain outputs. In mask ROM products, on-chip pull-up resistors can be connected to these pins using a mask option.

When P64 to P67 are used as input port pins, on-chip pull-up resistor can be connected using pull-up resistor option register L (PUOL).

#### (2) Control mode

P60 to P67 function as control signal output pins ( $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{WAIT}$ , ASTB) in external memory expansion mode. When pins are used as control signal outputs, on-chip pull-up resistors are automatically disabled.

**Caution** When an external wait is not used in external memory expansion mode, P66 can be used as an I/O port pins.

**3.2.8 P70 to P72 (Port 7)**

P70 to P72 function as a 3-bit I/O port. Besides serving as I/O port pins, they also function as serial interface data I/O and clock I/O.

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

**(1) Port mode**

P70 to P72 function as a 3-bit I/O port. They can be specified as input port or output in 1-bit units using port mode register 7 (PM7). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

**(2) Control mode**

P70 to P72 function as serial interface data I/O and clock I/O.

**(a) SI2, SO2**

These are serial data I/O pins of the serial interface.

**(b)  $\overline{\text{SCK2}}$** 

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

**(c) RxD0, TxD0**

These are serial data I/O pins of the asynchronous serial interface.

**(d) ASCK**

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

**Caution** When P70 to P72 are used as serial interface pins, the I/O and output latches must be set according to the function the user requires.

For the setting, see the operation mode setting list in Table 19-2 Serial Interface Channel 2.

**3.2.9 P120 to P127 (Port 12)**

P120 to P127 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as a real-time output port.

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

**(1) Port mode**

P120 to P127 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 12 (PM12). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register H (PUOH).

**(2) Control mode**

P120 to P127 function as a real-time output port (RTP0 to RTP7) that outputs data in synchronization with a trigger.

**3.2.10 P130 and P131 (Port 13)**

P130 and P131 function as a 2-bit I/O port. Besides serving as I/O port pins, they also function as D/A converter analog output.

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

**(1) Port mode**

P130 and P131 function as a 2-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 13 (PM13). When they are used as an input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register H (PUOH).

**(2) Control mode**

P130 and P131 function as D/A converter analog outputs (ANO0 and ANO1).

**Caution** When only one of the D/A converter channels is used with  $AV_{REF1} < V_{DD0}$ , the other pins that are not used as analog outputs must be set as follows:

- Set the PM13x bit of port mode register 13 (PM13) to 1 (input mode) and connect the pin to  $V_{SS0}$ .
- Clear the PM13x bit of port mode register 13 (PM13) to 0 (output mode) and the output latch to 0, and output a low level from the pin.

**3.2.11  $AV_{REF0}$** 

This is the A/D converter reference voltage input pin. This pin also serves as an analog power supply pin. Supply power to this pin when the A/D converter is used.

- ★ When the A/D converter is not used, use the same voltage that of the  $V_{DD0}$  or  $V_{SS0}$  pin.

**3.2.12  $AV_{REF1}$** 

This is the D/A converter reference voltage input pin.

When the D/A converter is not used, use the same voltage that of the  $V_{DD0}$  pin.

**3.2.13  $AV_{SS}$** 

This is the ground voltage pin of A/D converter and D/A converter. Always use the same voltage as that of the  $V_{SS0}$  pin even when the A/D converter or D/A converter is not used.

**3.2.14  $\overline{RESET}$** 

This is the low-level active system reset input pin.

**3.2.15 X1 and X2**

These are crystal resonator connection pins for main system clock oscillation. For external clock supply, input a signal to X1 and its inverted signal to X2.

**3.2.16 XT1 and XT2**

These are crystal resonator connection pins for subsystem clock oscillation.

For external clock supply, input a signal to XT1 and its inverted signal to XT2.

**3.2.17  $V_{DD0}$ ,  $V_{DD1}$** 

$V_{DD0}$  is the positive power supply pin for ports.

$V_{DD1}$  is the positive power supply pin for blocks other than port and analog blocks.

**3.2.18  $V_{SS0}$ ,  $V_{SS1}$** 

$V_{SS0}$  is the ground potential pin for ports.

$V_{SS1}$  is the ground potential pin for blocks other than port and analog blocks.

**3.2.19  $V_{PP}$  (Flash memory version only)**

This is the high-voltage application pin for flash memory programming mode setting and program write/verify.

★ Connect this pin in either of the following ways.

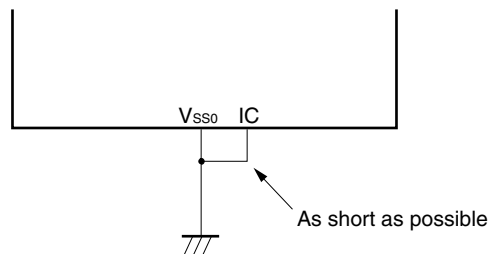
- Connect independently to a 10 k $\Omega$  pull-down resistor.
- By using a jumper on the board, connect directly to the dedicated flash programmer in the programming mode or to  $V_{SS0}$  in the normal operation mode.

**3.2.20 IC (Mask ROM version only)**

The IC (Internally Connected) pin is provided to set the test mode to check the  $\mu$ PD780058 Subseries at delivery. Connect it directly to  $V_{SS0}$  with the shortest possible wire in the normal operating mode.

When a voltage difference is produced between the IC pin and  $V_{SS0}$  pin because the wiring between those two pins is too long or an external noise is input to the IC pin, the user's program may not run normally.

- **Connect the IC pin to  $V_{SS0}$  directly.**



### 3.3 I/O Circuits and Recommended Connection of Unused Pins

Table 3-1 shows the pin I/O circuit types and the recommended connection of unused pins. Refer to Figure 3-1 for the configuration of the I/O circuit of each type.

★

**Table 3-1. Pin I/O Circuit Types (1/2)**

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins		
P00/INTP0/TI00	2	Input	Connect to $V_{SS0}$ .		
P01/INTP1/TI01	8-C	I/O	Input: Independently connect to $V_{SS0}$ via a resistor. Output: Leave open.		
P02/INTP2					
P03/INTP3					
P04/INTP4					
P05/INTP5					
P07/XT1	16	Input	Connect to $V_{DD0}$ .		
P10/ANI0 to P17/ANI7	11-D	I/O	Input: Independently connect to $V_{DD0}$ or $V_{SS0}$ via a resistor. Output: Leave open.		
P20/SI1	8-C				
P21/SO1	5-H				
P22/SCK1	8-C				
P23/STB/TxD1	5-H				
P24/BUSY/RxD1	8-C				
P25/SI0/SB0	10-B				
P26/SO0/SB1					
P27/SCK0					
P30/TO0	5-H				
P31/TO1					
P32/TO2					
P33/TI1	8-C				
P34/TI2					
P35/PCL	5-H				
P36/BUZ					
P37					
P40/AD0 to P47/AD7	5-N			I/O	Input: Independently connect to $V_{DD0}$ via a resistor. Output: Leave open.
P50/A8 to P57/A15	5-H			I/O	Input: Independently connect to $V_{DD0}$ or $V_{SS0}$ via a resistor. Output: Leave open.



Table 3-1. Pin I/O Circuit Types (2/2)

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins
P60 to P63 (mask ROM version)	13-J	I/O	Input: Independently connect to $V_{DD0}$ via a resistor. Output: Set 0 to the port and leave open at low level output.
P60 to P63 (flash memory version)	13-K		
$\overline{P64/RD}$	5-H	I/O	Input: Independently connect to $V_{DD0}$ or $V_{SS0}$ via a resistor. Output: Leave open.
$\overline{P65/WR}$			
$\overline{P66/WAIT}$			
$\overline{P67/ASTB}$			
$\overline{P70/SI2/RxD0}$			
$\overline{P71/SO2/TxD0}$	5-H		
$\overline{P72/SCK2/ASCK}$	8-C		
$\overline{P120/RTP0}$ to $\overline{P127/RTP7}$	5-H		
$\overline{P130/ANO0}$ , $\overline{P131/ANO1}$	12-C	I/O	Input: Independently connect to $V_{SS0}$ via a resistor. Output: Leave open.
$\overline{RESET}$	2	Input	—
XT2	16	—	Leave open.
$AV_{REF0}$	—		Connect to $V_{DD0}$ or $V_{SS0}$ .
$AV_{REF1}$			Connect to $V_{DD0}$ .
$AV_{SS}$			Connect to $V_{SS0}$ .
IC (mask ROM version)			Connect directly to $V_{SS0}$ .
$V_{PP}$ (flash memory version)			Independently connect via a 10 k $\Omega$ pull-down resistor, or connect to $V_{SS0}$ or $V_{SS1}$ directly.

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

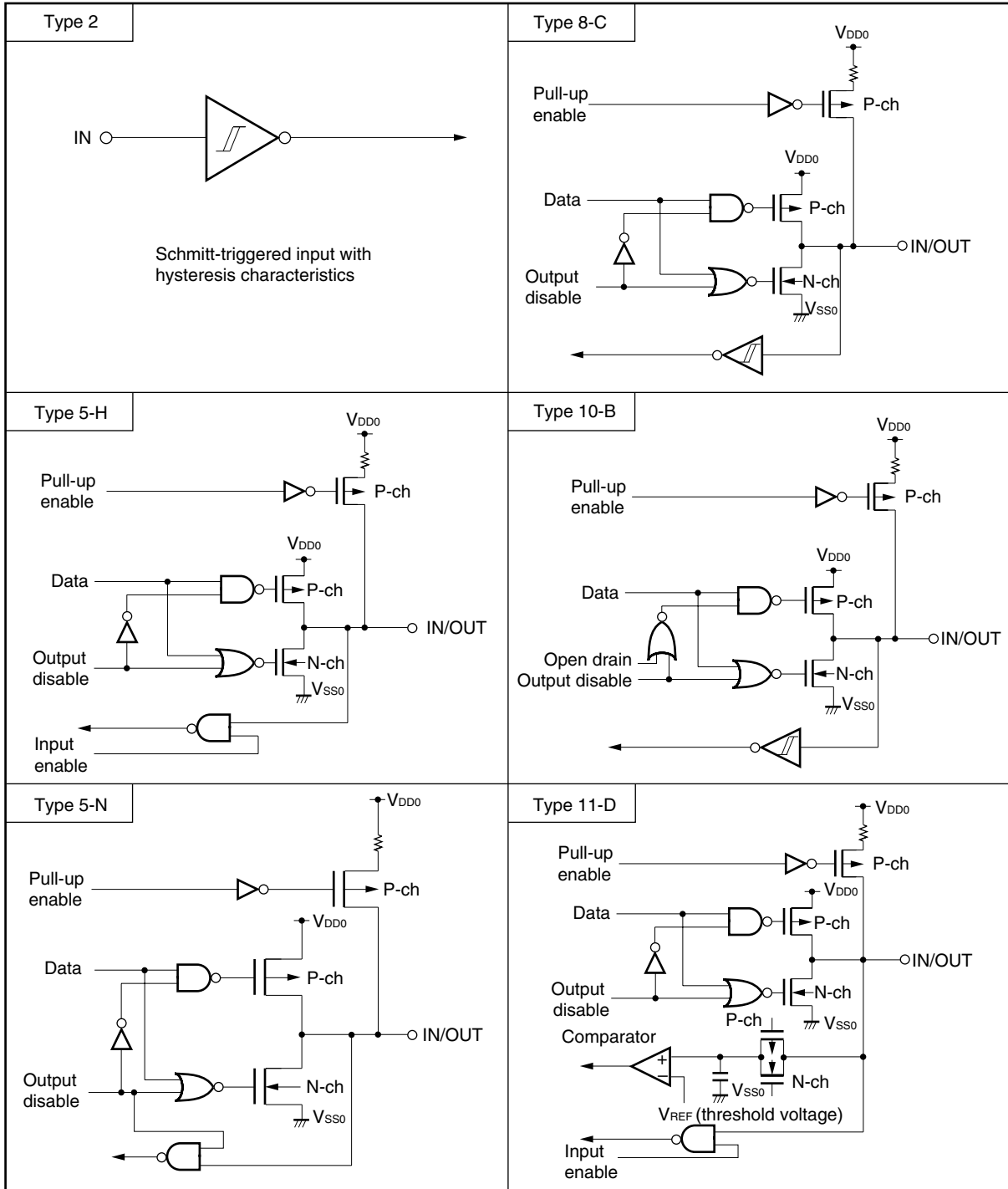
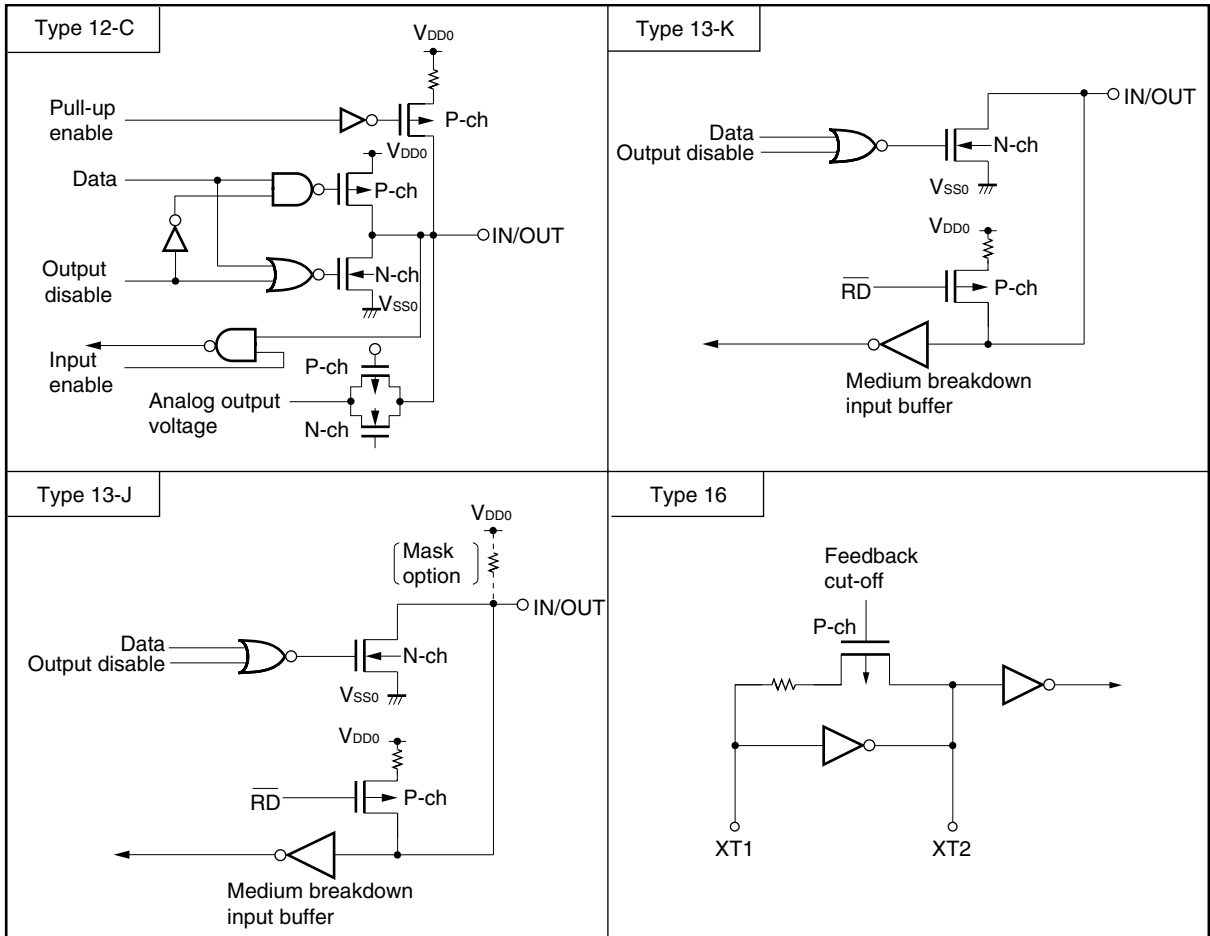




Figure 3-1. Pin I/O Circuit List (2/2)

★



## CHAPTER 4 PIN FUNCTIONS ( $\mu$ PD780058Y SUBSERIES)

### 4.1 Pin Function List

#### (1) Port pins (1/2)

Pin Name	I/O	Function		After Reset	Alternate Function
P00	Input	Port 0 7-bit I/O port	Input only	Input	INTP0/TI00
P01	I/O		Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	INTP1/TI01
P02					INTP2
P03					INTP3
P04					INTP4
P05					INTP5
P07 <sup>Note 1</sup>	Input		Input only	Input	XT1
P10 to P17	I/O	Port 1 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software <sup>Note 2</sup> .		Input	ANI0 to ANI7
P20	I/O	Port 2 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	SI1	
P21				SO1	
P22				SCK1	
P23				STB/TxD1	
P24				BUSY/RxD1	
P25				SI0/SB0/SDA0	
P26				SO0/SB1/SDA1	
P27				SCK0/SCL	
P30	I/O	Port 3 8-bit I/O port Input/output mode can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	TO0	
P31				TO1	
P32				TO2	
P33				TI1	
P34				TI2	
P35				PCL	
P36				BUZ	
P37				—	

- Notes**
1. When the P07/XT1 pin is used as an input port, set bit 6 (FRC) of the processor clock control register (PCC) to 1 (do not use the feedback resistor incorporated in the subsystem clock oscillator).
  2. When pins P10/ANI0 to P17/ANI7 are used as an analog input of the A/D converter, set port 1 to the input mode. In this case, any connected on-chip pull-up resistors are automatically disabled.

(1) Port pins (2/2)

Pin Name	I/O	Function		After Reset	Alternate Function
P40 to P47	I/O	Port 4 8-bit I/O port Input/output can be specified in 8-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software. The test input flag (KRIF) is set to 1 by falling edge detection.		Input	AD0 to AD7
P50 to P57	I/O	Port 5 8-bit I/O port LEDs can be driven directly. Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	A8 to A15
P60	I/O	Port 6 8-bit I/O port Input/output can be specified in 1-bit units.	N-ch open-drain I/O port On-chip pull-up resistors can be specified by mask option. (Mask ROM version only). LEDs can be driven directly.	Input	—
P61					
P62					
P63					
P64			If used as an input port, an on-chip pull-up resistor can be connected by setting software.		$\overline{\text{RD}}$
P65					$\overline{\text{WR}}$
P66					$\overline{\text{WAIT}}$
P67					ASTB
P70	I/O	Port 7 3-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input	SI2/RxD0	
P71				$\overline{\text{SO2/TxD0}}$	
P72				$\overline{\text{SCK2/ASCK}}$	
P120 to P127	I/O	Port 12 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	RTP0 to RTP7
P130 to P131	I/O	Port 13 2-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		Input	ANO0 to ANO1

## (2) Non-port pins (1/2)

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request inputs with specifiable valid edges (rising edge, falling edge, both rising and falling edges).	Input	P00/TI00
INTP1				P01/TI01
INTP2				P02
INTP3				P03
INTP4				P04
INTP5				P05
SI0	Input	Serial interface serial data input	Input	P25/SB0/SDA0
SI1				P20
SI2				P70/RxD
SO0	Output	Serial interface serial data output	Input	P26/SB1/SDA1
SO1				P21
SO2				P71/TxD
SB0	I/O	Serial interface serial data input/output	Input	P25/SI0/SDA0
SB1				P26/SO0/SDA1
SDA0				P25/SI0/SB0
SDA1				P26/SO0/SB1
$\overline{\text{SCK0}}$	I/O	Serial interface serial clock input/output	Input	P27/SCL
$\overline{\text{SCK1}}$				P22
$\overline{\text{SCK2}}$				P72/ASCK
SCL				$\overline{\text{P27/SCK0}}$
STB	Output	Serial interface automatic transmit/receive strobe output	Input	P23/TxD1
BUSY	Input	Serial interface automatic transmit/receive busy input	Input	P24/RxD1
RxD0	Input	Asynchronous serial interface serial data input	Input	P70/SI2
RxD1				P24/BUSY
TxD	Output	Asynchronous serial interface serial data output	Input	P71/ $\overline{\text{SO2}}$
TxD1				P23/ $\overline{\text{STB}}$
ASCK	Input	Asynchronous serial interface serial clock input	Input	P72/ $\overline{\text{SCK2}}$
TI00	Input	External count clock input to 16-bit timer (TM0)	Input	P00/INTP0
TI01		Capture trigger signal input to capture register (CR00)		P01/INTP1
TI1		External count clock input to 8-bit timer (TM1)		P33
TI2		External count clock input to 8-bit timer (TM2)		P34
TO0	Output	16-bit timer (TM0) output (also used for 14-bit PWM output)	Input	P30
TO1		8-bit timer (TM1) output		P31
TO2		8-bit timer (TM2) output		P32
PCL	Output	Clock output (for main system clock and subsystem clock trimming)	Input	P35
BUZ	Output	Buzzer output	Input	P36
RTP0 to RTP7	Output	Real-time output port outputting data in synchronization with trigger	Input	P120 to P127

(2) Non-port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
AD0 to AD7	I/O	Lower address/data bus when expanding memory externally	Input	P40 to P47
A8 to A15	Output	Higher address bus when expanding memory externally	Input	P50 to P57
$\overline{\text{RD}}$	Output	Strobe signal output for read operation from external memory	Input	P64
$\overline{\text{WR}}$		Strobe signal output for write operation to external memory		P65
$\overline{\text{WAIT}}$	Input	Wait insertion when accessing external memory	Input	P66
ASTB	Output	Strobe output externally latching address information output to ports 4 and 5 to access external memory	Input	P67
ANI0 to ANI7	Input	A/D converter analog input	Input	P10 to P17
ANO0, ANO1	Output	D/A converter analog output	Input	P130, P131
$\text{AV}_{\text{REF}0}$	Input	A/D converter reference voltage input (also functions as analog power supply)	—	—
$\text{AV}_{\text{REF}1}$	Input	D/A converter reference voltage input	—	—
$\text{AV}_{\text{SS}}$	—	A/D converter, D/A converter ground potential. Use the same potential as $\text{V}_{\text{SS}0}$ .	—	—
$\overline{\text{RESET}}$	Input	System reset input	—	—
X1	Input	Crystal connection for main system clock oscillation	—	—
X2	—		—	—
XT1	Input	Crystal connection for subsystem clock oscillation	Input	P07
XT2	—		—	—
$\text{V}_{\text{DD}0}$	—	Positive power supply for ports	—	—
$\text{V}_{\text{SS}0}$	—	Ground potential for ports	—	—
$\text{V}_{\text{DD}1}$	—	Positive power supply (except ports and analog block)	—	—
$\text{V}_{\text{SS}1}$	—	Ground potential (except ports and analog block)	—	—
$\text{V}_{\text{PP}}$	—	High-voltage application for program write/verify.	—	—
$\text{V}_{\text{SS}}$	—	Ground potential	—	—
IC	—	Internally connected. Connect directly to $\text{V}_{\text{SS}0}$ .	—	—

## 4.2 Description of Pin Functions

### 4.2.1 P00 to P05, P07 (Port 0)

P00 to P05 and P07 function as a 7-bit I/O port. Besides serving as I/O port pins, they function as an external interrupt request input, an external count clock input to the timer, a capture trigger signal input, and crystal connection for subsystem oscillation.

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

#### (1) Port mode

P00 and P07 function as input-only port pins and P01 to P05 function as I/O port pins.

P01 to P05 can be specified as input or output in 1-bit units using port mode register 0 (PM0). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

#### (2) Control mode

In this mode, P00 to P05 and P07 function as an external interrupt request input, an external count clock input to the timer, and crystal connection for subsystem clock oscillation.

##### (a) INTP0 to INTP5

INTP0 to INTP5 are external interrupt request input pins for which valid edges can be specified (rising edge, falling edge, and both rising and falling edges). INTP0 and INTP1 become a 16-bit timer/event counter capture trigger signal input pins with a valid edge input.

##### (b) TI00

This is a pin for inputting the external count clock to the 16-bit timer/event counter.

##### (c) TI01

This is a pin for inputting the capture trigger signal to the capture register (CR00) of the 16-bit timer/event counter.

##### (d) XT1

This is a crystal connection pin for subsystem clock oscillation.

**4.2.2 P10 to P17 (Port 1)**

P10 to P17 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as an A/D converter analog inputs.

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

**(1) Port mode**

P10 to P17 function as an 8-bit I/O port.

They can be specified as input or output in 1-bit units using port mode register 1 (PM1). When they are used as input port pins, on-chip pull-up resistor can be connected to them using pull-up resistor option register L (PUOL).

**(2) Control mode**

P10 to P17 function as A/D converter analog input pins (ANI0 to ANI7). On-chip pull-up resistors are automatically disabled when these pins are specified as analog inputs.

**4.2.3 P20 to P27 (Port 2)**

P20 to P27 an 8-bit I/O port. Besides serving as I/O port pins, they also function as data input/output to/from the serial interface, clock input/output, automatic transmit/receive busy input, and strobe output.

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

**(1) Port mode**

P20 to P27 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 2 (PM2). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

**(2) Control mode**

P20 to P27 function as serial interface data input/output, clock input/output, automatic transmit/receive busy input, and strobe output.

**(a) SI0, SI1, SO0, SO1, SB0, SB1, SDA0, SDA1**

These are serial data I/O pins of the serial interface.

**(b)  $\overline{\text{SCK0}}$ ,  $\overline{\text{SCK1}}$ , SCL**

These are serial clock I/O pins of the serial interface.

**(c) BUSY**

This is an automatic transmit/receive busy input pin of the serial interface.

**(d) STB**

This is an automatic transmit/receive strobe output pin of the serial interface.

**(e) RxD1, TxD1**

These are serial interface serial data I/O pins of the asynchronous serial interface.

**Caution** When P20 to P27 are used as a serial interface pins, the I/O and output latches must be set according to the function the user requires. For the setting, see Figure 17-4 Format of Serial Operation Mode Register 0, Figure 18-3 Format of Serial Operation Mode Register 1, and Table 19-2 Serial Interface Channel 2 Operating Mode Settings.

#### 4.2.4 P30 to P37 (Port 3)

P30 to P37 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as timer input/output, clock output, and buzzer output.

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

##### (1) Port mode

P30 to P37 function as an 8-bit I/O port. They can be specified as an input or output using in 1-bit units port mode register 3 (PM3). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

##### (2) Control mode

P30 to P37 function as timer input/output, clock output, and buzzer output.

##### (a) TI1 and TI2

These are pins for inputting the external count clock to the 8-bit timer/event counter.

##### (b) TO0 to TO2

These are timer output pins.

##### (c) PCL

This is a clock output pin.

##### (d) BUZ

This is a buzzer output pin.



#### 4.2.5 P40 to P47 (Port 4)

P40 to P47 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as an address/data bus.

The test input flag (KRIF) can be set to 1 by detecting a falling edge.

The following operating modes can be specified in 8-bit units.

##### (1) Port mode

P40 to P47 function as an 8-bit I/O port. They can be specified as input or output in 8-bit units using the internal memory expansion mode register (MM). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

##### (2) Control mode

P40 to P47 function as the lower address/data bus pins (AD0 to AD7) in external memory expansion mode. When these pins are used as an address/data bus, on-chip pull-up resistors are automatically disabled.

#### 4.2.6 P50 to P57 (Port 5)

P50 to P57 function as an 8-bit I/O port. Besides serving as I/O port pins, they also function as an address bus. P50 to P57 can drive LEDs directly.

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

##### (1) Port mode

P50 to P57 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 5 (PM5). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

##### (2) Control mode

P50 to P57 function as the higher address bus pins (A8 to A15) in external memory expansion mode. When these pins are used as an address bus, on-chip pull-up resistors are automatically disabled.

#### 4.2.7 P60 to P67 (Port 6)

P60 to P67 function as an 8-bit I/O port. Besides serving as I/O port pins, they are also used for control in external memory expansion mode. P60 to P63 can drive LEDs directly.

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

##### (1) Port mode

P60 to P67 function as an 8-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 6 (PM6).

P60 to P63 are N-ch open-drain outputs. In mask ROM products, on-chip pull-up resistors can be connected to these pins using a mask option.

When P64 to P67 are used as input port pins, on-chip pull-up resistor can be connected using pull-up resistor option register L (PUOL).

##### (2) Control mode

P60 to P67 functions as control signal output pins ( $\overline{RD}$ ,  $\overline{WR}$ ,  $\overline{WAIT}$ , ASTB) in external memory expansion mode. When pins are used as control signal outputs, the on-chip pull-up resistors are automatically disabled.

**Caution** When an external wait is not used in external memory expansion mode, P66 can be used as an I/O port pin.

**4.2.8 P70 to P72 (Port 7)**

P70 to P72 function as a 3-bit I/O port. Besides serving as I/O port pins, they also function as serial interface data I/O and clock I/O.

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

**(1) Port mode**

P70 to P72 function as a 3-bit I/O port. They can be specified as input port or output in 1-bit units using port mode register 7 (PM7). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register L (PUOL).

**(2) Control mode**

P70 to P72 function as serial interface data I/O and clock I/O.

**(a) SI2, SO2**

These are serial data I/O pins of the serial interface.

**(b)  $\overline{\text{SCK2}}$** 

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

**(c) RxD0, TxD0**

These are serial interface serial data I/O pins of the asynchronous serial interface.

**(d) ASCK**

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

**Caution** When P70 to P72 are used as serial interface pins, the I/O and output latches must be set according to the function the user requires.

For the setting, see to the operation mode setting list in Table 19-2 Serial Interface Channel 2.

**4.2.9 P120 to P127 (Port 12)**

P120 to P127 function as an 8-bit I/O port. Besides serving as an I/O port pins, they also function as a real-time output port.

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

**(1) Port mode**

P120 to P127 function as an 8-bit I/O port. They can be specified as input or output port in 1-bit units using port mode register 12 (PM12). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register H (PUOH).

**(2) Control mode**

P120 to P127 function as a real-time output port (RTP0 to RTP7) that outputs data in synchronization with a trigger.

**4.2.10 P130 and P131 (Port 13)**

P130 and P131 function as a 2-bit I/O port. Besides serving as I/O port pins, they also function as D/A converter analog output.

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

**(1) Port mode**

P130 and P131 function as a 2-bit I/O port. They can be specified as input or output in 1-bit units using port mode register 13 (PM13). When they are used as input port pins, on-chip pull-up resistors can be connected to them using pull-up resistor option register H (PUOH).

**(2) Control mode**

P130 and P131 function as D/A converter analog outputs (ANO0 and ANO1).

**Caution** When only one of the D/A converter channels is used with  $AV_{REF1} < V_{DD0}$ , the other pins that are not used as analog outputs must be set as follows:

- Set the PM13x bit of port mode register 13 (PM13) to 1 (input mode) and connect the pin to  $V_{SS0}$ .
- Clear the PM13x bit of port mode register 13 (PM13) to 0 (output mode) and the output latch to 0, and output a low level from the pin.

**4.2.11  $AV_{REF0}$** 

This is the A/D converter reference voltage input pin. This pin also serves as an analog power supply pin. Supply power to this pin when the A/D converter is used.

★

When the A/D converter is not used, use the same voltage that of the  $V_{DD0}$  or  $V_{SS0}$  pin.

**4.2.12  $AV_{REF1}$** 

This is the D/A converter reference voltage input pin.

When the D/A converter is not used, use the same voltage that of the  $V_{DD0}$  pin.

**4.2.13  $AV_{SS}$** 

This is the ground voltage pin of A/D converter and D/A converter. Always use the same voltage as that of the  $V_{SS0}$  pin even when the A/D converter or D/A converter is not used.

**4.2.14  $\overline{RESET}$** 

This is the low-level active system reset input pin.

**4.2.15 X1 and X2**

These are crystal resonator connection pins for main system clock oscillation. For external clock supply, input a signal to X1 and its inverted signal to X2.

**4.2.16 XT1 and XT2**

These are crystal resonator connection pins for subsystem clock oscillation.

For external clock supply, input a signal to XT1 and its inverted signal to XT2.

**4.2.17  $V_{DD0}$ ,  $V_{DD1}$** 

$V_{DD0}$  is the positive power supply pin for ports.

$V_{DD1}$  is the positive power supply pin for blocks other than port and analog blocks.

**4.2.18  $V_{SS0}$ ,  $V_{SS1}$** 

$V_{SS0}$  is the ground potential pin for ports.

$V_{SS1}$  is the ground potential pin for blocks other than port and analog blocks.

**4.2.19  $V_{PP}$  (Flash memory version only)**

This is the high-voltage apply pin for flash memory programming mode setting and program write/verify.

★ Connect this pin in either of the following ways.

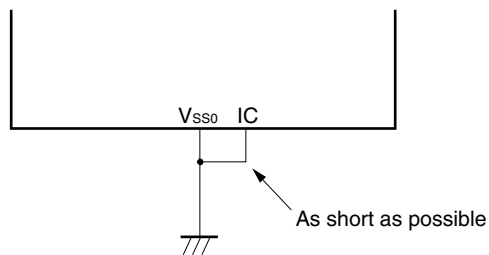
- Connect independently to a 10 k $\Omega$  pull-down resistor.
- By using a jumper on the board, connect directly to the dedicated flash programmer in the programming mode or to  $V_{SS0}$  in the normal operation mode.

**4.2.20 IC (Mask ROM version only)**

The IC (Internally Connected) pin is provided to set the test mode to check the  $\mu$ PD780058Y Subseries at delivery. Connect it directly to  $V_{SS0}$  with the shortest possible wire in the normal operating mode.

When a voltage difference is produced between the IC pin and  $V_{SS0}$  pin because the wiring between those two pins is too long or an external noise is input to the IC pin, the user's program may not run normally.

- **Connect the IC pin to  $V_{SS0}$  directly.**



### 4.3 I/O Circuits and Recommended Connection of Unused Pins

Table 4-1 shows the pin I/O circuit types and the recommended connection of unused pins. Refer to Figure 4-1 for the configuration of the I/O circuit of each type.

★

**Table 4-1. Pin I/O Circuit Types (1/2)**

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins		
P00/INTP0/TI00	2	Input	Connect to $V_{SS0}$ .		
P01/INTP1/TI01	8-C	I/O	Input: Independently connect to $V_{SS0}$ via a resistor. Output: Leave open.		
P02/INTP2					
P03/INTP3					
P04/INTP4					
P05/INTP5					
P07/XT1	16	Input	Connect to $V_{DD0}$		
P10/ANI0 to P17/ANI7	11-D	I/O	Input: Independently connect to $V_{DD0}$ or $V_{SS0}$ via a resistor. Output: Leave open.		
P20/SI1	8-C				
P21/SO1	5-H				
P22/ $\overline{SCK1}$	8-C				
P23/STB/TxD1	5-H				
P24/BUSY/RxD1	8-C				
P25/SI0/SB0/SDA0	10-B				
P26/SO0/SB1/SDA1					
P27/ $\overline{SCK0}$ /SCL					
P30/TO0	5-H				
P31/TO1					
P32/TO2					
P33/TI1	8-C				
P34/TI2					
P35/PCL	5-H				
P36/BUZ					
P37					
P40/AD0 to P47/AD7	5-N			I/O	Input: Independently connect to $V_{DD0}$ via a resistor. Output: Leave open.
P50/A8 to P57/A15	5-H			I/O	Input: Independently connect to $V_{DD0}$ or $V_{SS0}$ via a resistor. Output: Leave open.

★

Table 4-1. Pin I/O Circuit Types (2/2)

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins
P60 to P63 (mask ROM version)	13-J	I/O	Input: Independently connect to $V_{DD0}$ via a resistor. Output: Set 0 to the port and leave open at low level output.
P60 to P63 (flash memory version)	13-K	I/O	Input: Independently connect to $V_{DD0}$ or $V_{SS0}$ via a resistor. Output: Leave open.
$\overline{P64/RD}$	5-H		
$\overline{P65/WR}$			
$\overline{P66/WAIT}$			
$\overline{P67/ASTB}$			
$\overline{P70/SI2/RxD0}$			
$\overline{P71/SO2/TxD0}$	5-H		
$\overline{P72/SCK2/ASCK}$	8-C		
$\overline{P120/RTP0}$ to $\overline{P127/RTP7}$	5-H		
$\overline{P130/ANO0}$ , $\overline{P131/ANO1}$	12-C		
$\overline{RESET}$	2	Input	—
XT2	16	—	Leave open.
$AV_{REF0}$	—		Connect to $V_{DD0}$ or $V_{SS0}$ .
$AV_{REF1}$			Connect to $V_{DD0}$ .
$AV_{SS}$			Connect to $V_{SS0}$ .
IC (mask ROM version)			Connect directly to $V_{SS0}$ .
$V_{PP}$ (flash memory version)			Independently connect 10 k $\Omega$ pull-down resistor, or connect to $V_{SS0}$ or $V_{SS1}$ directly.

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

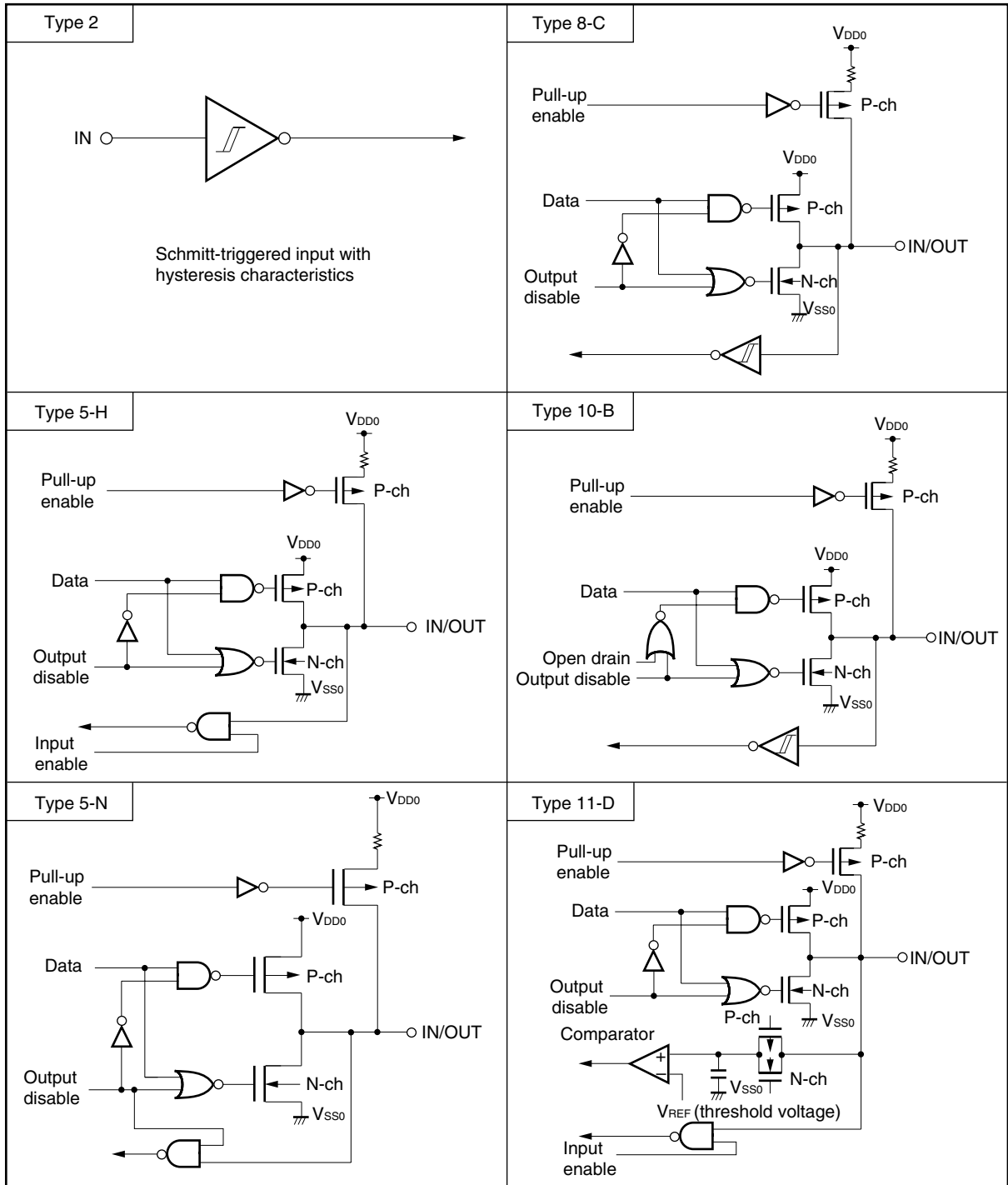
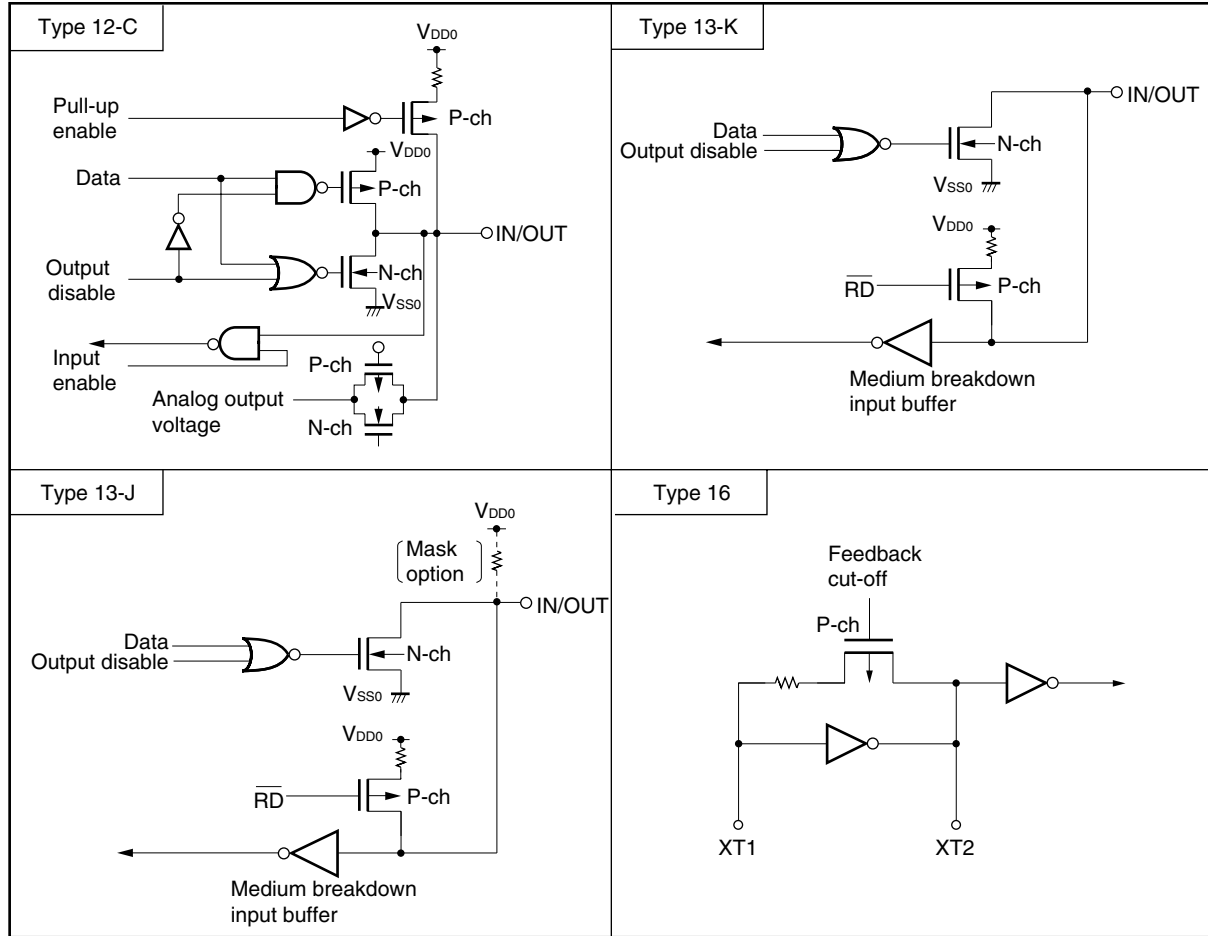


Figure 4-1. Pin I/O Circuit List (2/2)

★





## CHAPTER 5 CPU ARCHITECTURE

### 5.1 Memory Spaces

Figures 5-1 to 5-6 show the memory maps.

**Figure 5-1. Memory Map ( $\mu$ PD780053, 780053(A), 780053Y, 780053Y(A))**

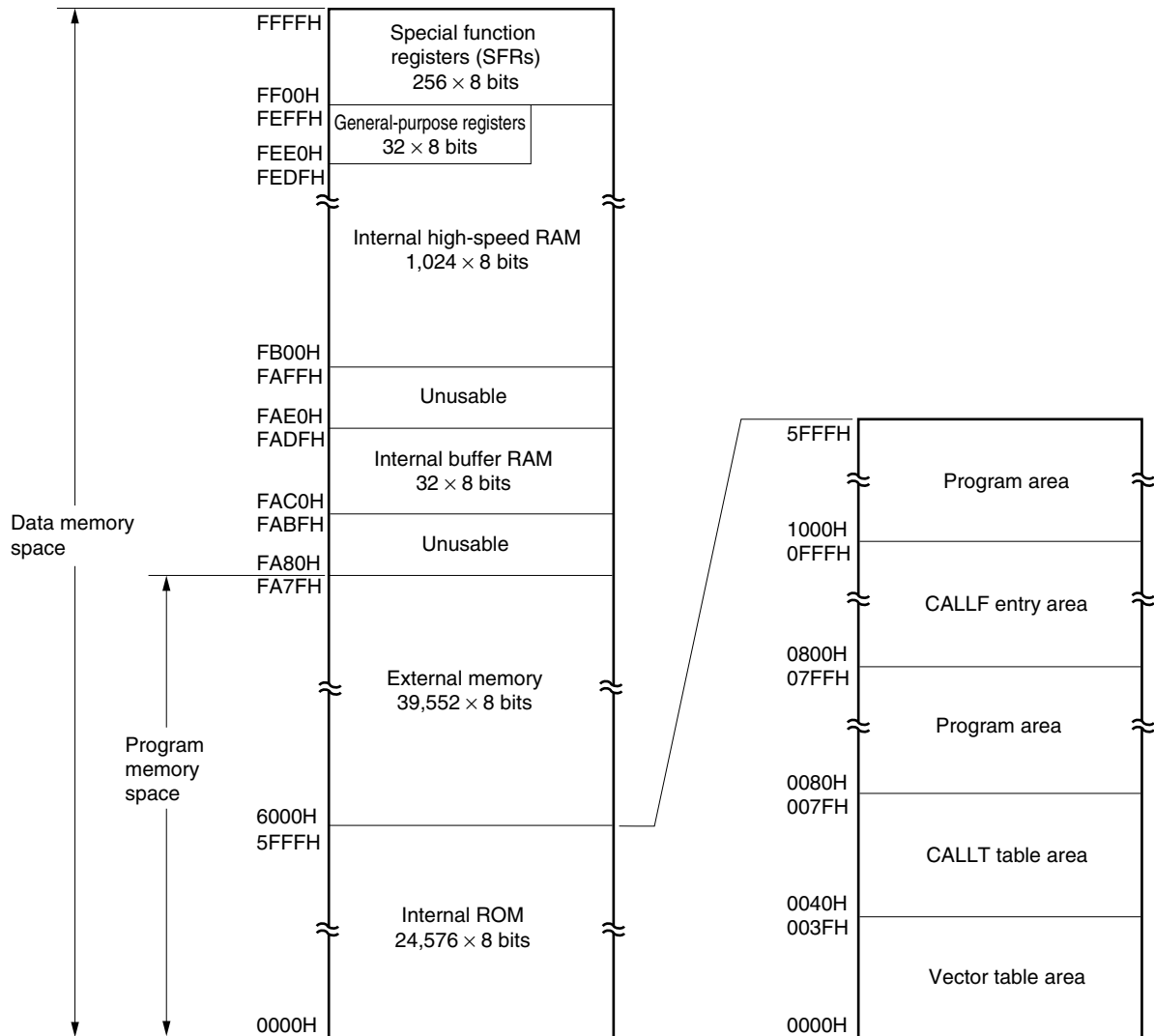


Figure 5-2. Memory Map ( $\mu$ PD780054, 780054(A), 780054Y, 780054Y(A))

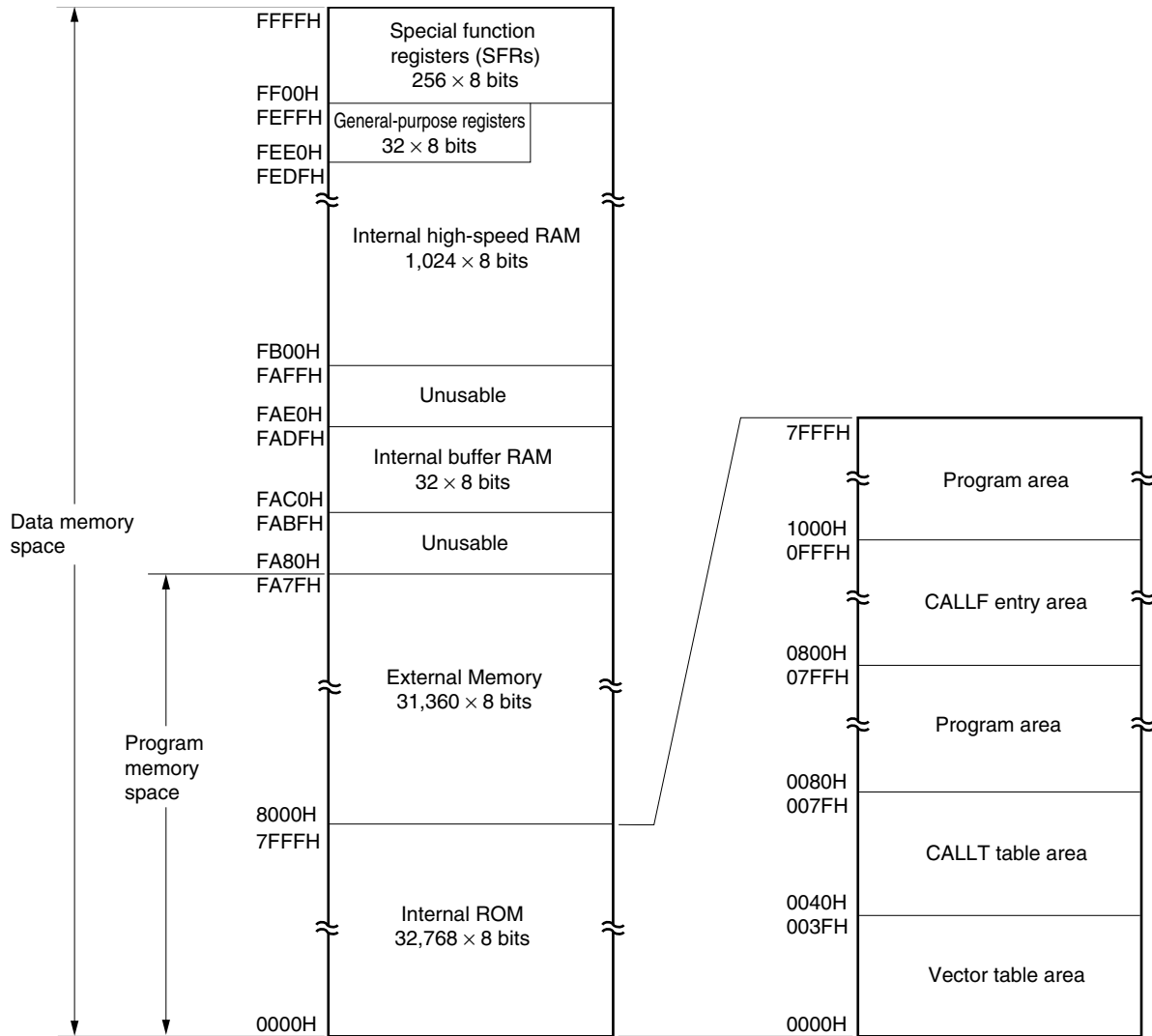


Figure 5-3. Memory Map ( $\mu$ PD780055, 780055(A), 780055Y, 780055Y(A))

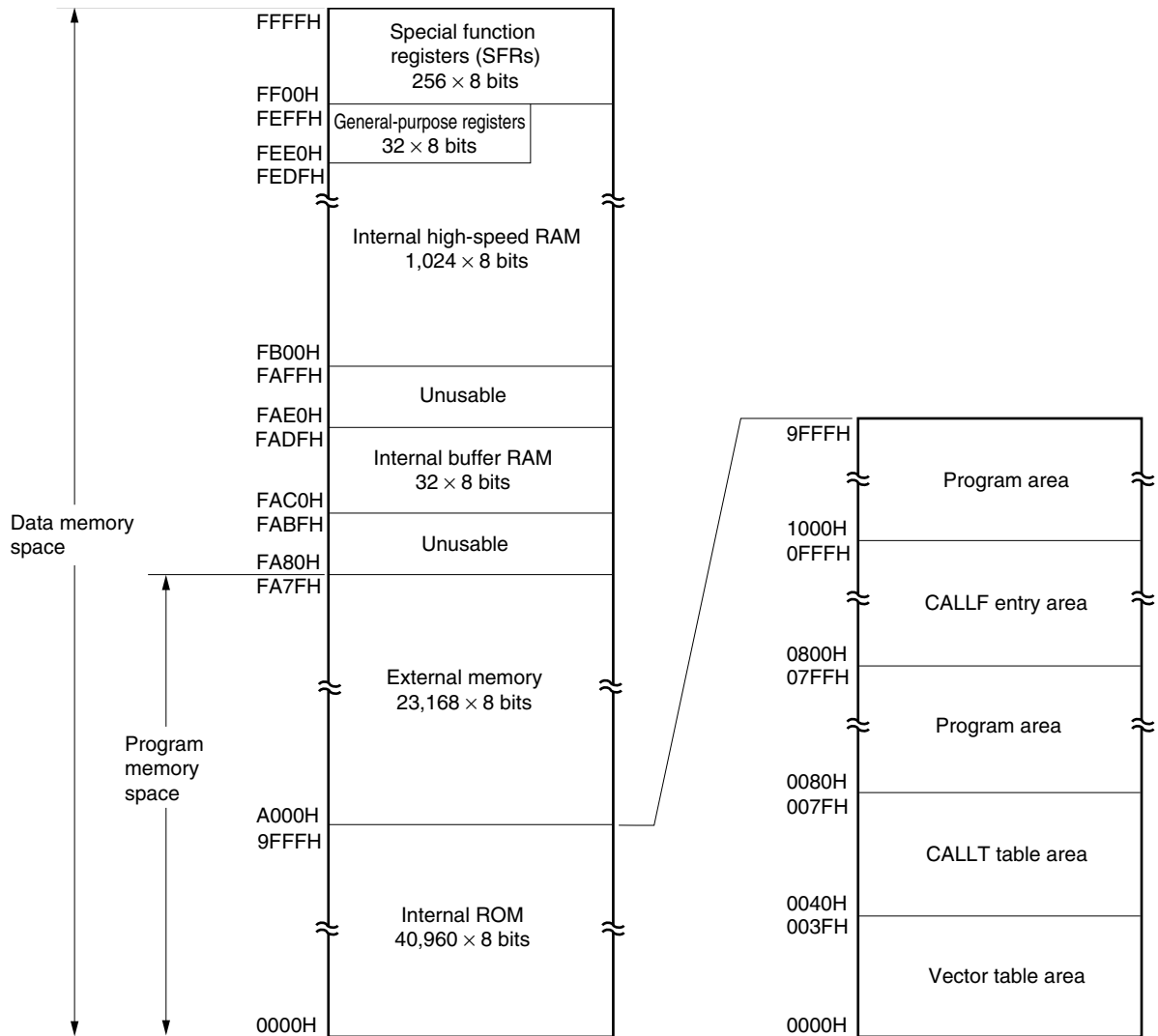


Figure 5-4. Memory Map ( $\mu$ PD780056, 780056(A), 780056Y, 780056Y(A))

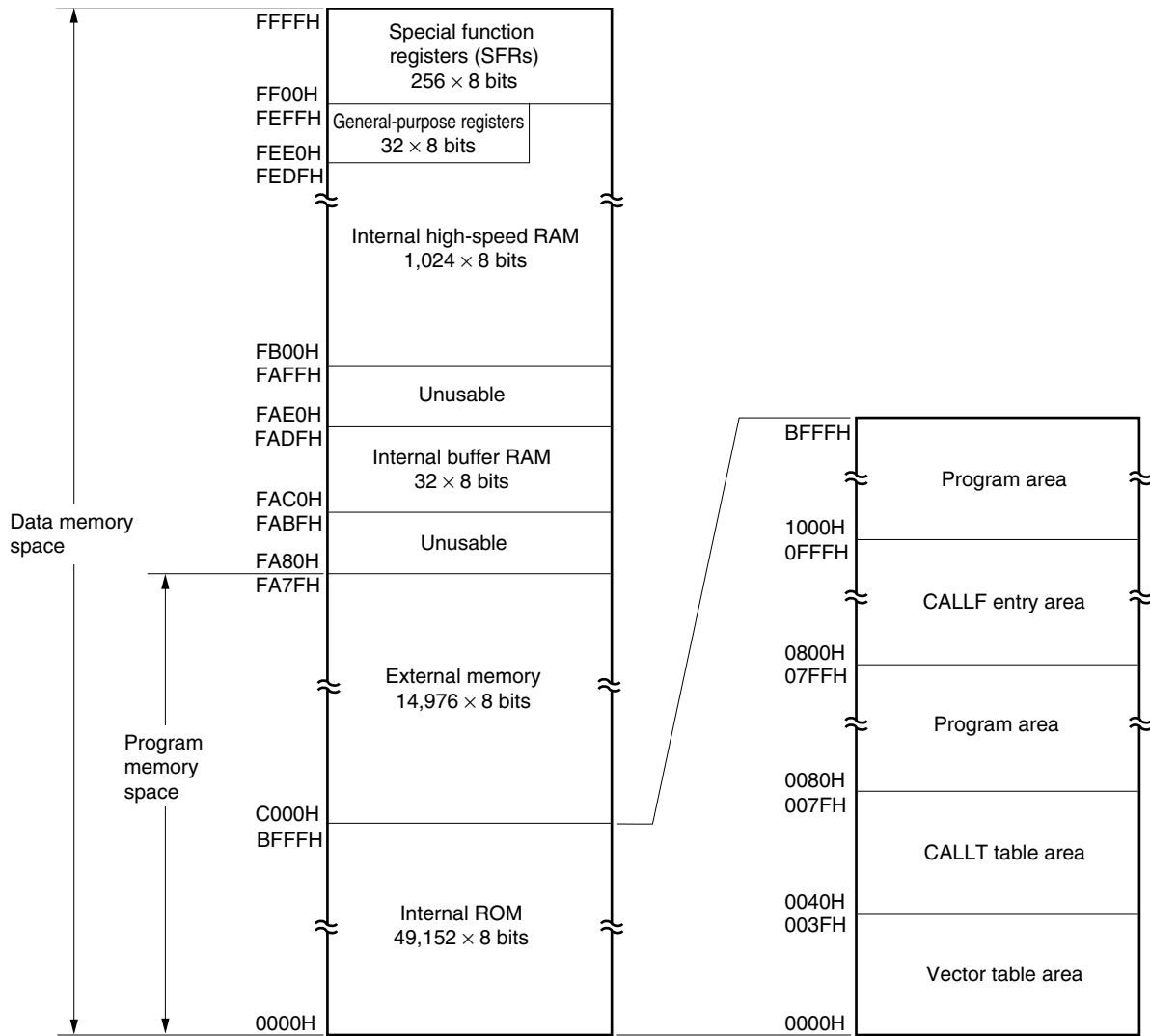
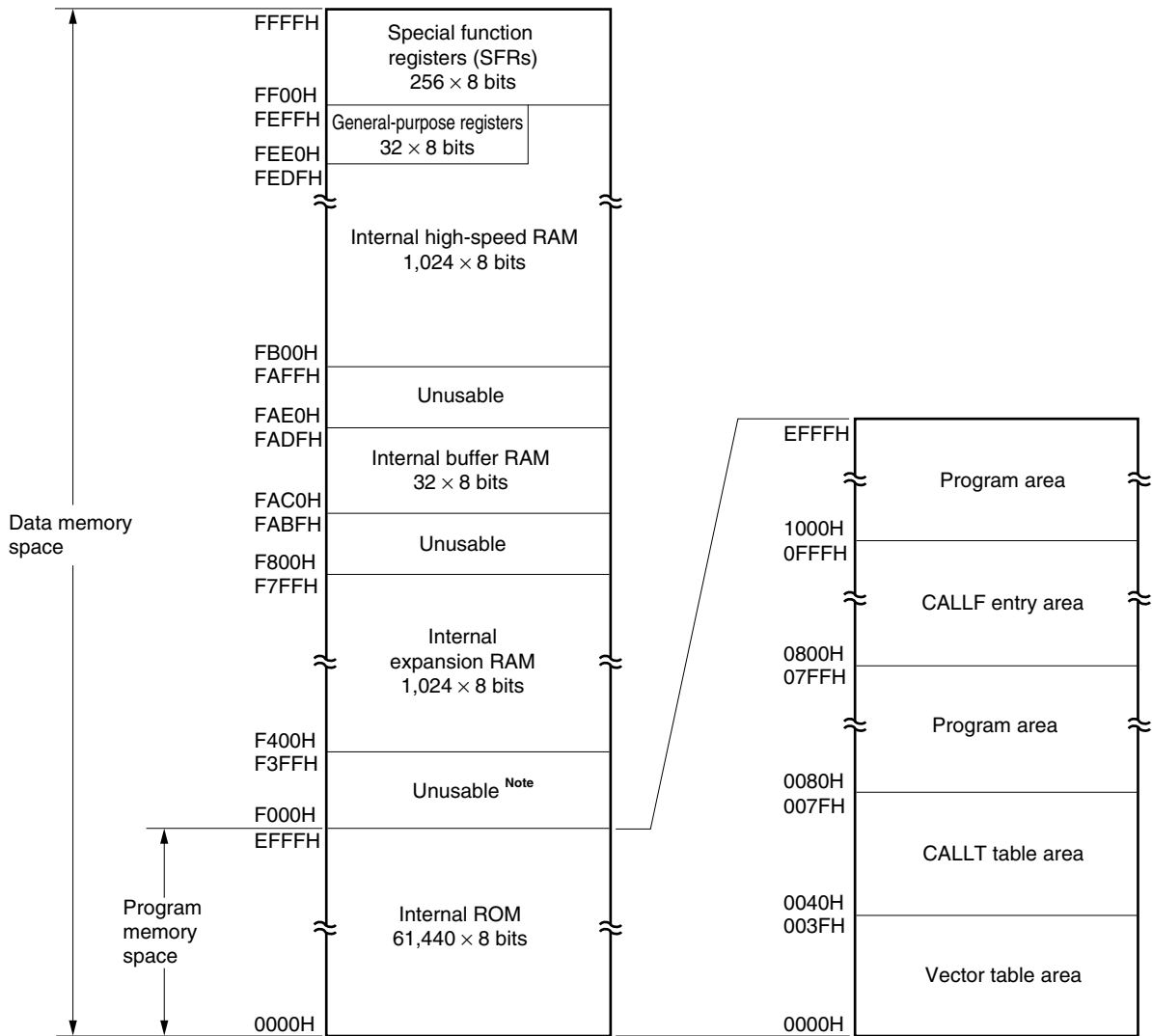
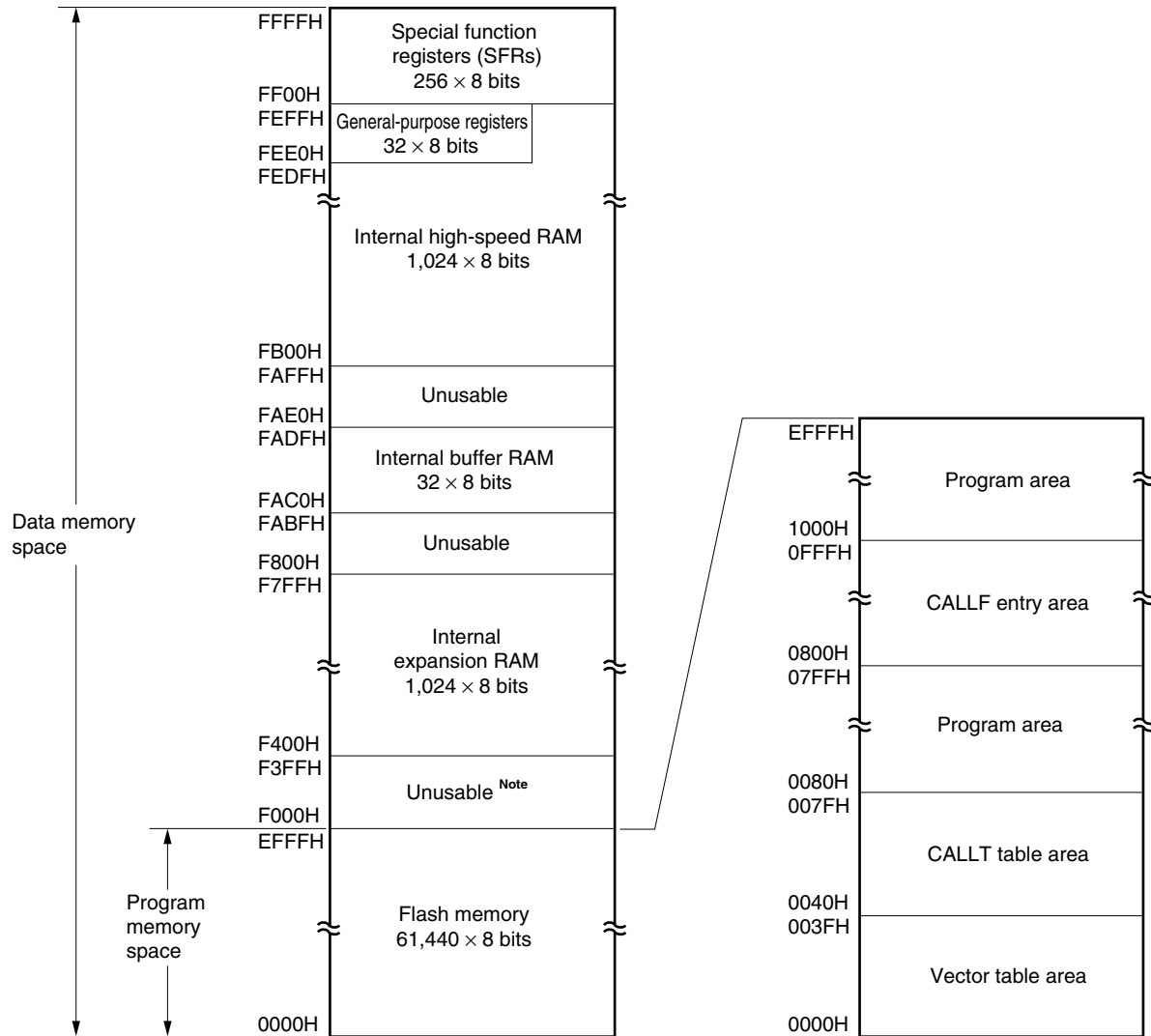


Figure 5-5. Memory Map ( $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A))



**Note** When the internal ROM size is 60 KB, the area F000H to F3FFH cannot be used. F000H to F3FFH can be used as external memory by setting the internal ROM size to 56 KB or less using the internal memory size switching register (IMS).

Figure 5-6. Memory Map ( $\mu$ PD78F0058, 78F0058Y)



**Note** When the flash memory size is 60 KB, the area F000H to F3FFH cannot be used. F000H to F3FFH can be used as external memory by setting the flash memory size to 56 KB or less using the internal memory size switching register (IMS).

### 5.1.1 Internal program memory space

The  $\mu$ PD780058 and 780058Y Subseries have various sizes of internal ROM or flash memory as shown below.

The internal program memory space stores programs and table data. Normally, they are addressed with a program counter (PC).

Part Number	Internal ROM	
	Type	Capacity
$\mu$ PD780053, 780053(A), 780053Y, 780053Y(A)	Mask ROM	24,576 $\times$ 8 bits
$\mu$ PD780054, 780054(A), 780054Y, 780054Y(A)		32,768 $\times$ 8 bits
$\mu$ PD780055, 780055(A), 780055Y, 780055Y(A)		40,960 $\times$ 8 bits
$\mu$ PD780056, 780056(A), 780056Y, 780056Y(A)		49,152 $\times$ 8 bits
$\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A)		61,440 $\times$ 8 bits
$\mu$ PD78F0058, 78F0058Y	Flash memory	61,440 $\times$ 8 bits

The internal program memory is divided into the following three 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  $\overline{\text{RESET}}$  input interrupt request or generation 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.

**Table 5-1. Vector Table**

Vector Table Address	Interrupt Source
0000H	$\overline{\text{RESET}}$ input
0004H	INTWDT
0006H	INTP0
0008H	INTP1
000AH	INTP2
000CH	INTP3
000EH	INTP4
0010H	INTP5
0014H	INTCSI0
0016H	INTCSI1
0018H	INTSER
001AH	INTSR/INTCSI2
001CH	INTST
001EH	INTTM3
0020H	INTTM00
0022H	INTTM01
0024H	INTTM1
0026H	INTTM2
0028H	INTAD
003EH	BRK

**(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) CALLF instruction entry area**

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



### 5.1.2 Internal data memory space

The  $\mu$ PD780058 and 780058Y Subseries incorporate the following RAMs.

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

High-speed memory of the following configuration is incorporated:

1,024  $\times$  8 bits (FB00H to FEFFH)

In this area, four banks of general-purpose registers, each bank consisting of eight 8-bit registers, are allocated to the 32-byte area FEE0H to FEFFH.

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

#### (2) Internal buffer RAM

Buffer RAM is allocated to the 32-byte area from FAC0H to FADFH. The internal buffer RAM is used to store transmit/receive data of serial interface channel 1 (in 3-wire serial I/O mode with automatic transmit/receive function). If the 3-wire serial I/O mode with automatic transmit/receive function is not used, the internal buffer RAM can also be used as normal RAM.

#### (3) Internal expansion RAM ( $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A), 78F0058, 78F0058Y only)

Internal expansion RAM is allocated to the 1,024-byte area from F400H to F7FFH.

### 5.1.3 Special Function Register (SFR) area

On-chip peripheral hardware special-function registers (SFRs) are allocated to the area FF00H to FFFFH. (See **Table 5-2 Special-Function Register List** in **5.2.3 Special Function Registers (SFRs)**).

**Caution Do not access addresses where SFRs are not assigned.**

### 5.1.4 External memory space

The external memory space is accessible by setting the internal memory expansion mode register (MM). External memory space can store program, table data, etc. and allocate peripheral devices.

### 5.1.5 Data memory addressing

The method to specify the address of the instruction to be executed next, or the address of a register or memory to be manipulated when an instruction is executed is called addressing.

The address of the instruction to be executed next is addressed by the program counter PC (for details, see **5.3 Instruction Address Addressing**).

To address the memory that is manipulated when an instruction is executed, the  $\mu$ PD780058, 780058Y Subseries is provided with many addressing modes with a high operability. Especially at addresses corresponding to data memory area, particular addressing modes can be used in accordance with the functions of the special function registers (SFRs) and general-purpose registers. This area is between FB00H and FFFFH. The data memory space is the entire 64 KB space (0000H to FFFFH). Figures 5-7 to 5-12 show the data memory addressing modes. For details of each addressing, see **5.4 Operand Address Addressing**.

Figure 5-7. Data Memory Addressing ( $\mu$ PD780053, 780053(A), 780053Y, 780053Y(A))

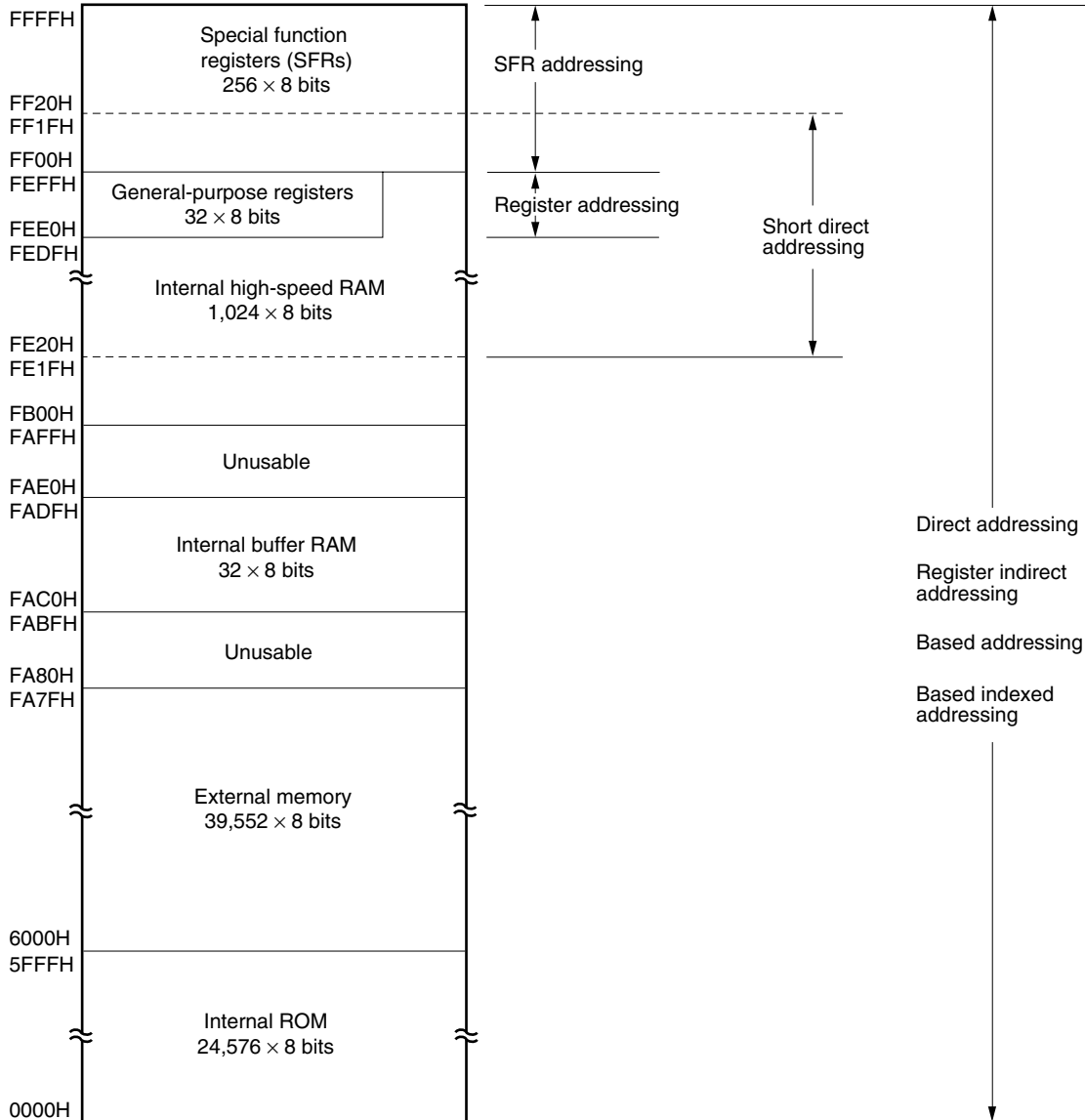


Figure 5-8. Data Memory Addressing ( $\mu$ PD780054, 780054(A), 780054Y, 780054Y(A))

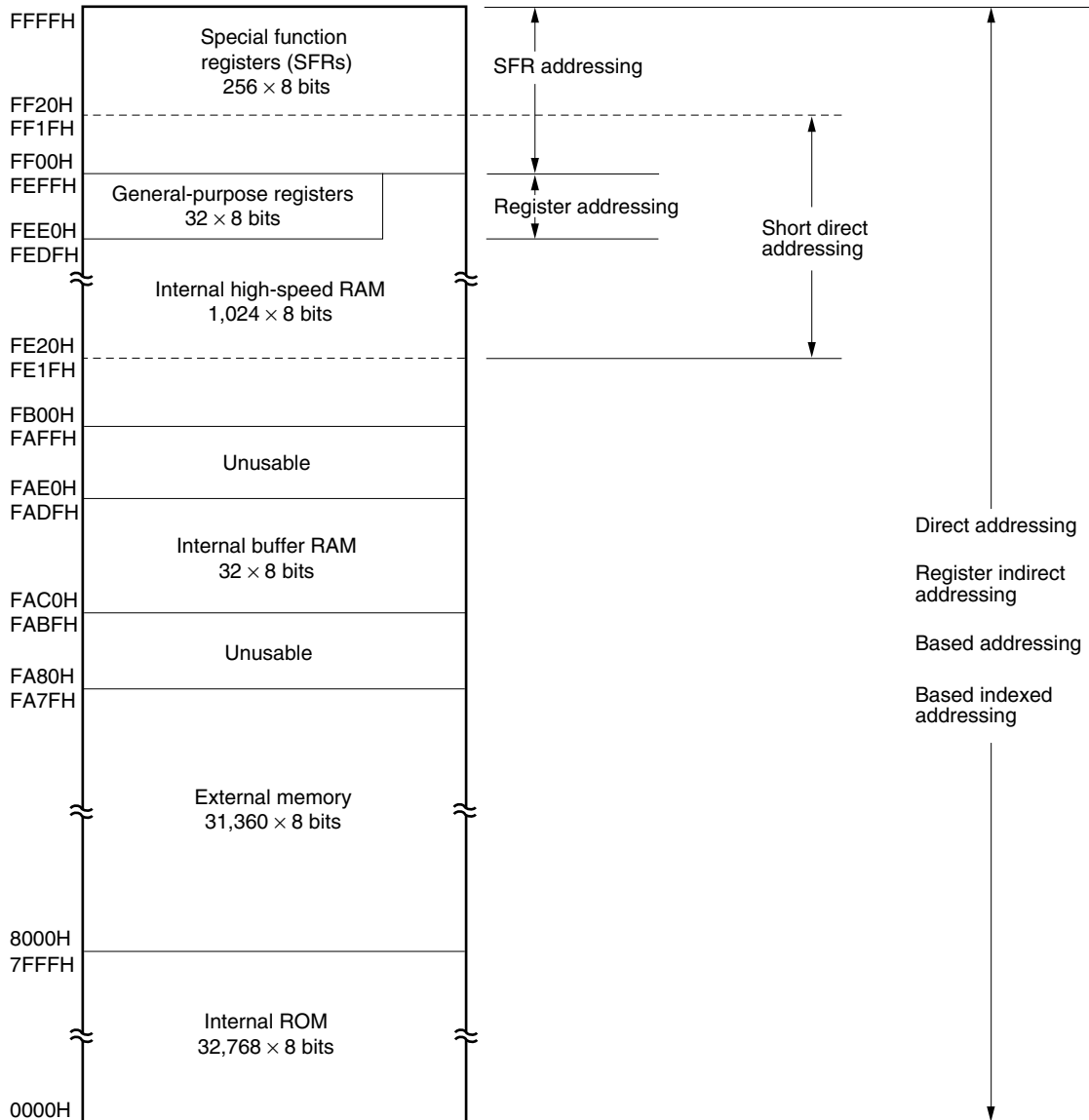


Figure 5-9. Data Memory Addressing ( $\mu$ PD780055, 780055(A), 780055Y, 780055Y(A))

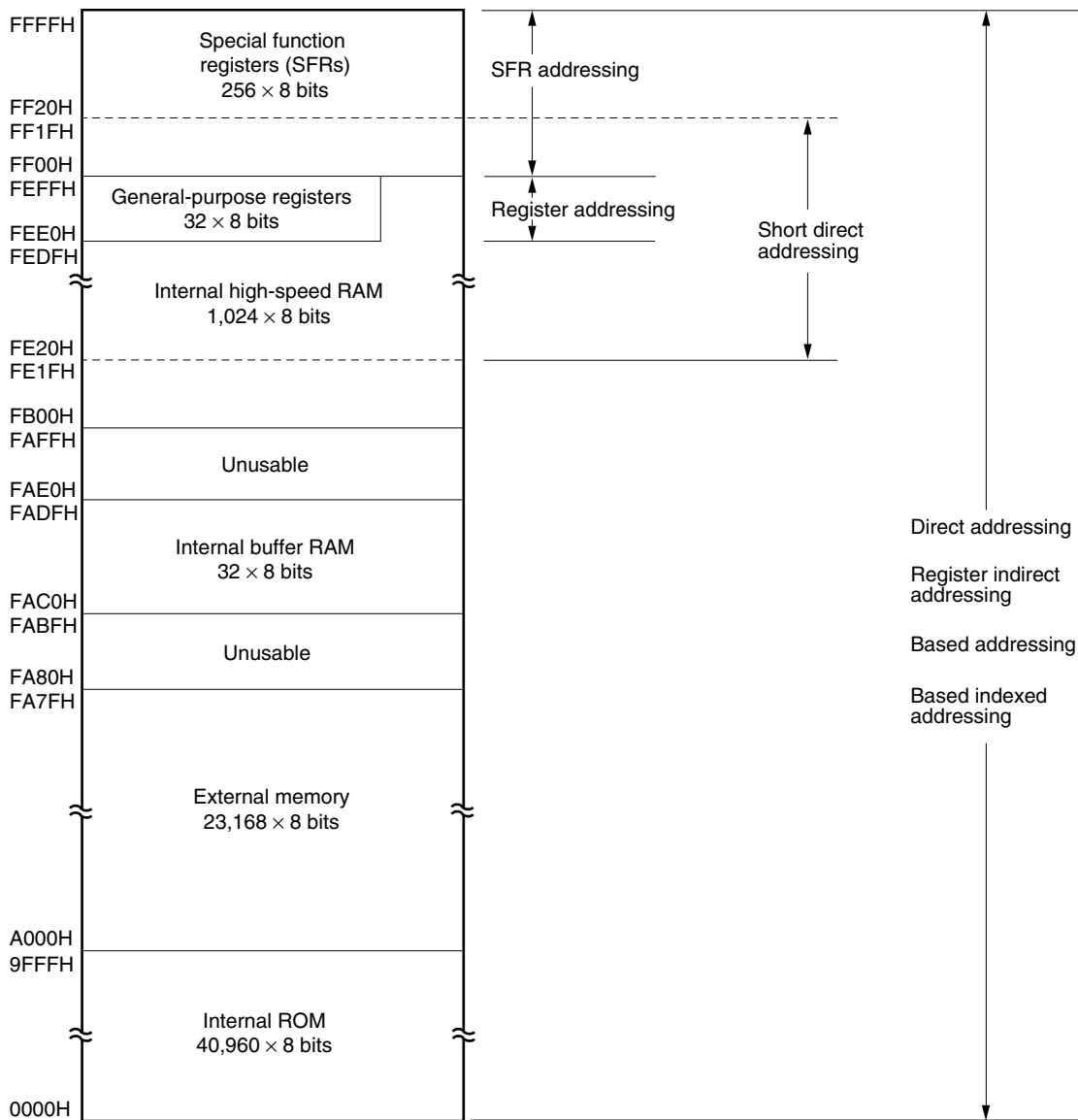


Figure 5-10. Data Memory Addressing ( $\mu$ PD780056, 780056(A), 780056Y, 780056Y(A))

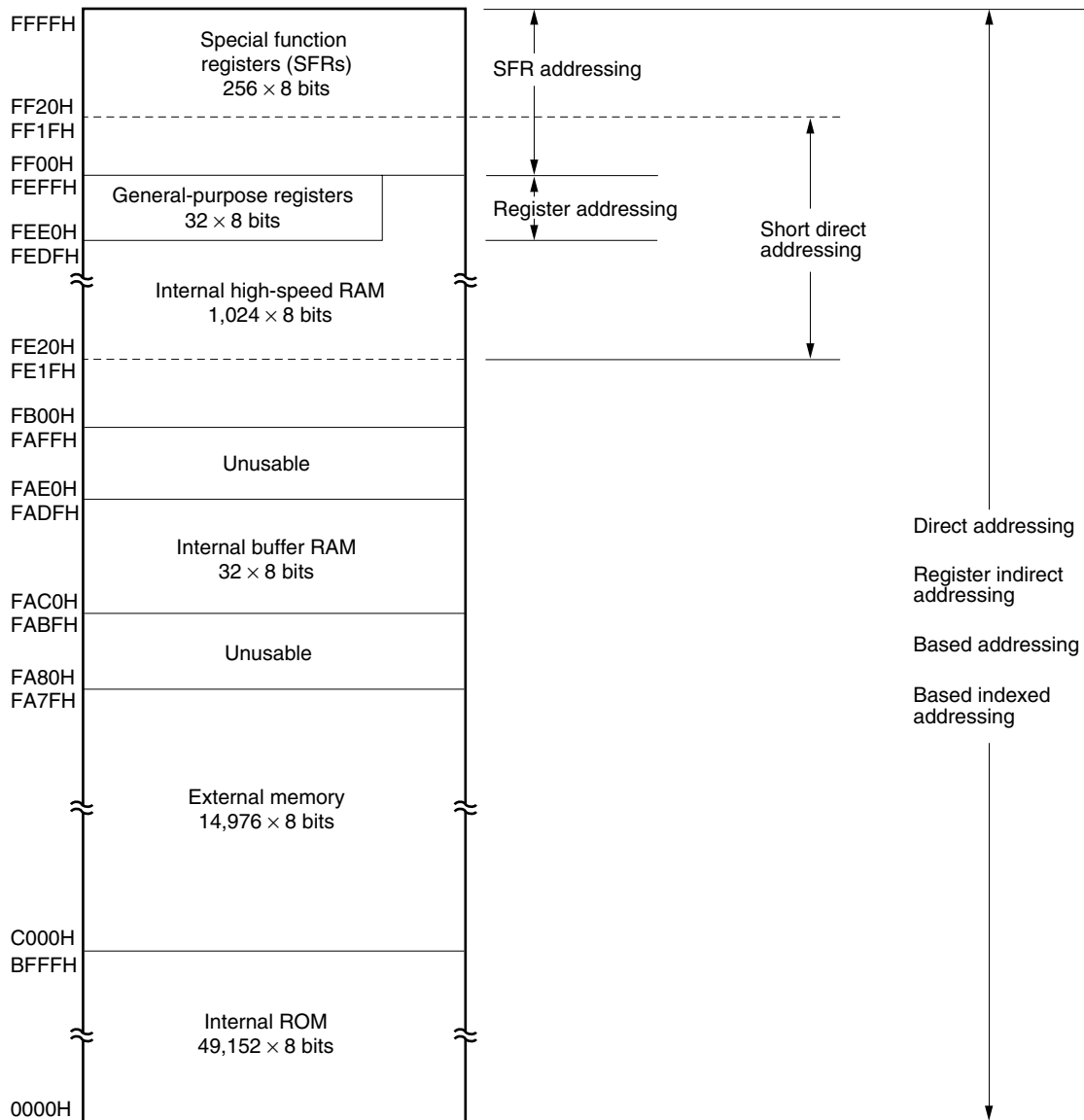
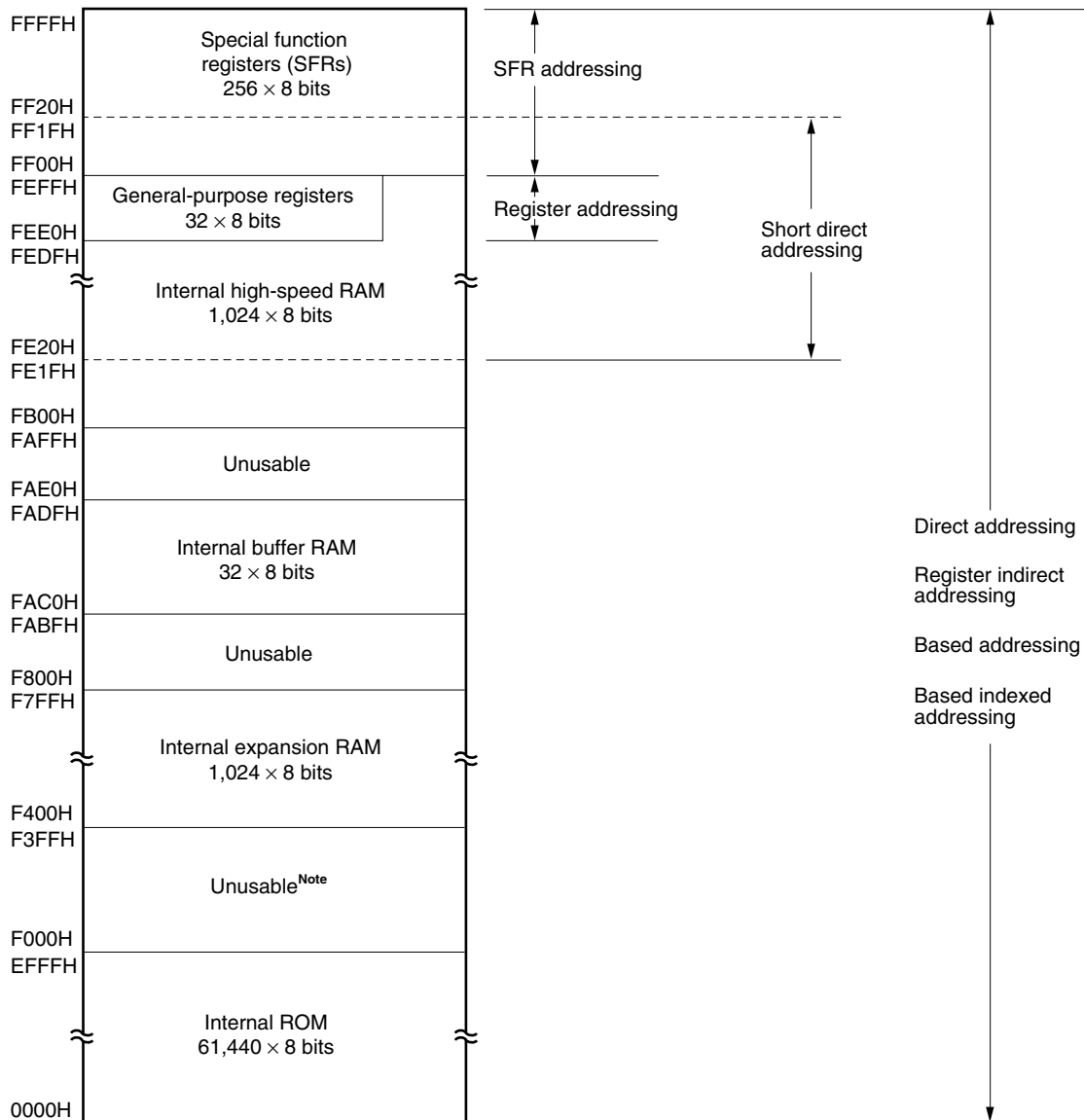
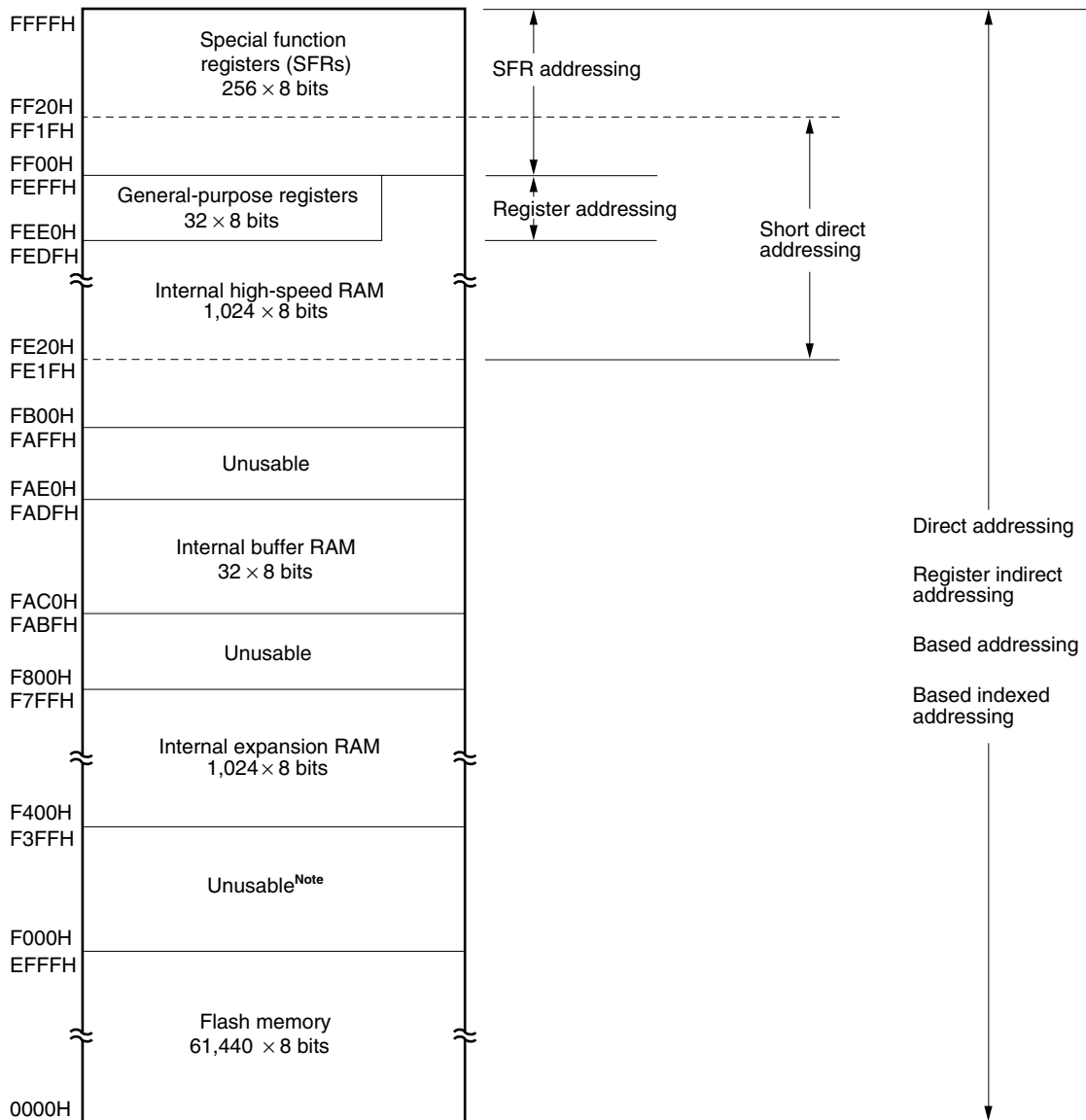


Figure 5-11. Data Memory Addressing ( $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A))



**Note** When the internal ROM size is 60 KB, the area F000H to F3FFH cannot be used. F000H to F3FFH can be used as external memory by setting the internal ROM size to 56 KB or less using the internal memory size switching register (IMS).

Figure 5-12. Data Memory Addressing ( $\mu$ PD78F0058, 78F0058Y)



**Note** When the flash memory size is 60 KB, the area F000H to F3FFH cannot be used. F000H to F3FFH can be used as external memory by setting the flash memory size to 56 KB or less using the internal memory size switching register (IMS).

## 5.2 Processor Registers

The  $\mu$ PD780058 and 780058Y Subseries incorporate the following processor registers.

### 5.2.1 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).

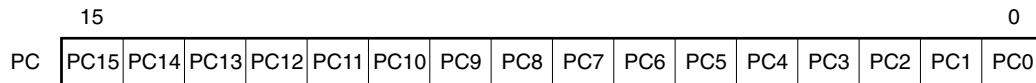
#### (1) Program counter (PC)

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

In normal operation, the 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.

$\overline{\text{RESET}}$  input sets the reset vector table values at addresses 0000H and 0001H to the program counter.

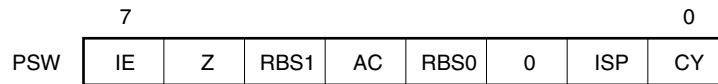
**Figure 5-13. Program Counter Format**



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

The program status word is an 8-bit register consisting of various flags to be set/reset by instruction execution. Program status word contents are automatically stacked upon interrupt request generation or PUSH PSW instruction execution and are automatically reset upon execution of the RETB, RETI, and POP PSW instructions.  $\overline{\text{RESET}}$  input sets the PSW to 02H.

**Figure 5-14. Program Status Word Format**





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

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

When IE = 0, all interrupt requests except the non-maskable interrupt are disabled (DI status).

When IE = 1, interrupts are enabled (EI status). At this time, acknowledgment of interrupts is controlled with an in-service priority flag (ISP), an interrupt mask flag for various interrupt sources, and a priority specification flag.

The interrupt enable flag is reset to 0 when the DI instruction is executed or when an interrupt request is acknowledged, and set to 1 when the EI instruction is executed.

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

When the operation result is zero, this flag is set to 1. It is reset to 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.

The 2-bit information which indicates the register bank selected by SEL R<sub>n</sub> instruction execution is stored in these flags.

**(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 to 1. It is reset to 0 in all other cases.

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

This flag manages the priority of acknowledgeable maskable vectored interrupts. When ISP = 0, the vectored interrupt whose priority is specified by the priority specification flag registers (PR0L, PR0H, and PR1L) (see **21.3 (3) Priority specification flag registers (PR0L, PR0H, and PR1L)**) to be low is disabled. Whether the interrupt is actually acknowledged is controlled by the status of the interrupt enable flag (IE).

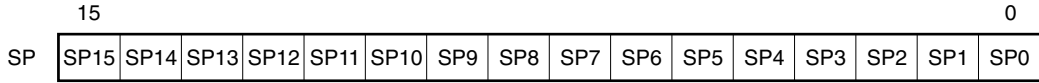
**(f) Carry flag (CY)**

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 manipulation 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 (FB00H to FEFFH) can be set as the stack area.

**Figure 5-15. Stack Pointer Format**

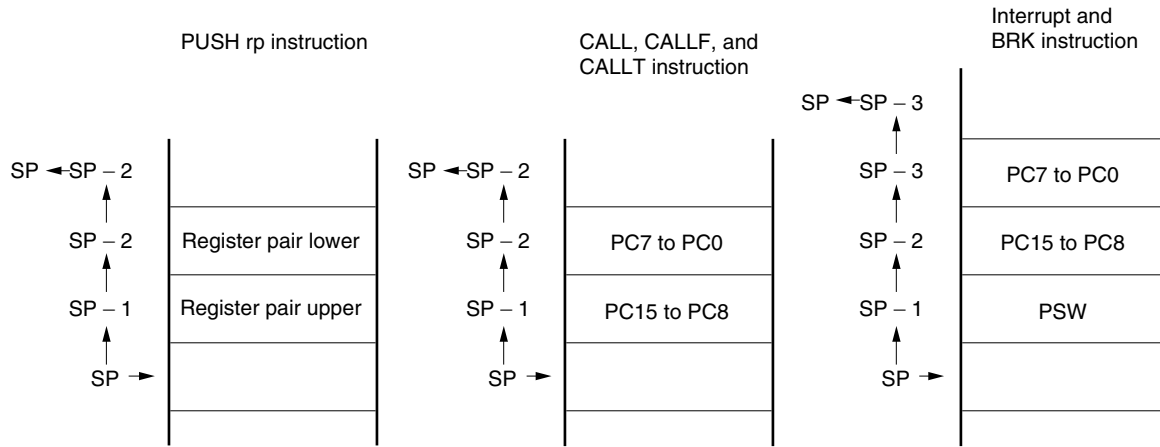


The SP is decremented ahead of write (save) to the stack memory and is incremented after read (reset) from the stack memory.

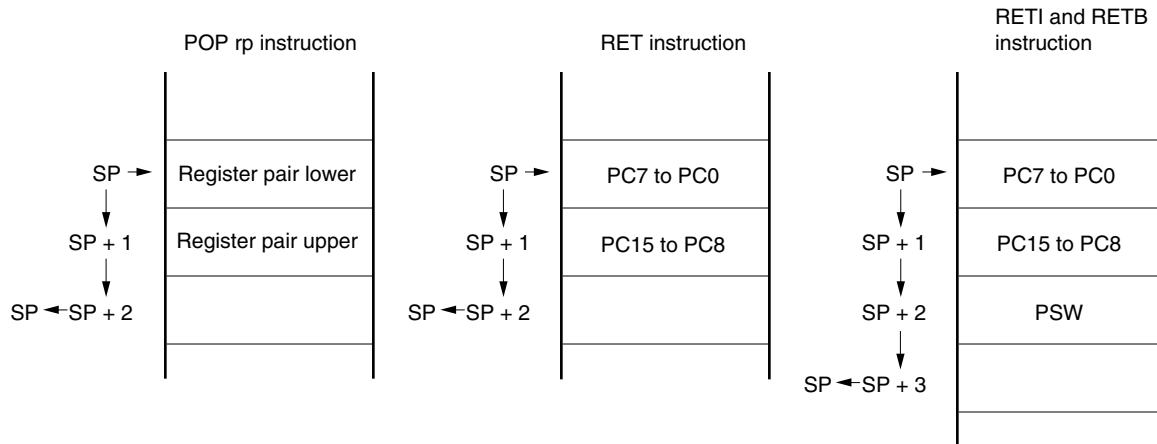
Each stack operation saves/resets data as shown in Figures 5-16 and 5-17.

**Caution** Because **RESET** input makes **SP** contents indeterminate, be sure to initialize the **SP** before instruction execution.

**Figure 5-16. Data to Be Saved to Stack Memory**



**Figure 5-17. Data to Be Reset from Stack Memory**



**5.2.2 General registers**

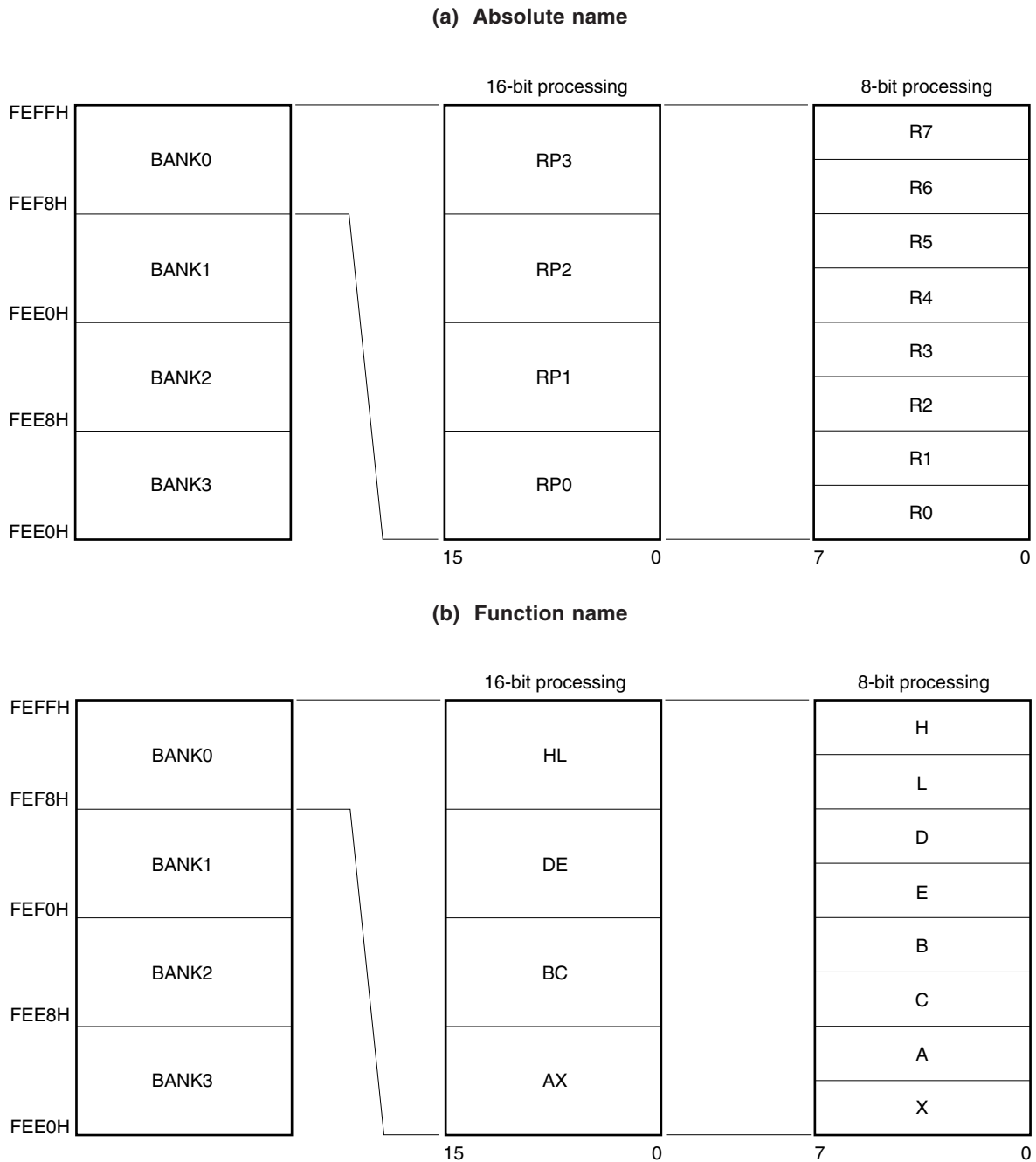
General-purpose registers are mapped at particular addresses (FEE0H to FEFFH) of the data memory. They consist of 4 banks, each bank containing eight 8-bit registers (X, A, C, B, E, D, L, and H).

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

They 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 with the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interrupt requests for each bank.

**Figure 5-18. General-Purpose Register Configuration**



### 5.2.3 Special-Function Registers (SFRs)

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

These registers are allocated in the FF00H to FFFFH area.

Special-function registers can be manipulated like general-purpose registers, with 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 assembler for the 1-bit manipulation instruction operand (sfr.bit).  
This manipulation can also be specified using an address.
- 8-bit manipulation  
Describe the symbol reserved by assembler for the 8-bit manipulation instruction operand (sfr).  
This manipulation can also be specified using an address.
- 16-bit manipulation  
Describe the symbol reserved by assembler for the 16-bit manipulation instruction operand (sfrp).  
When addressing an address, describe an even address.

Table 5-2 gives a list of special-function registers. The meanings of items in the table are as follows.

- Symbol  
Symbol indicating the addresses of the special function register. These symbols are reserved words in the RA78K0 and defined by header file sfrbit.h in the CC78K0, and can be used as the operands of instructions when the RA78K0, ID78K0, ID78K0-NS, and SM78K0 are used.
- R/W  
Indicates whether the corresponding special-function register can be read or written.  
R/W: Read/write enabled  
R: Read only  
W: Write only
- Manipulatable bit units  
√ indicates the bit units (1, 8, or 16 bits) in which the register can be manipulated. — indicates that the register cannot be manipulated in the indicated bit units.
- After reset  
Indicates each register status upon  $\overline{\text{RESET}}$  input.

Table 5-2. Special-Function Register List (1/3)

Address	Special-Function Register (SFR) Name	Symbol		R/W	Manipulatable Bit Unit			After Reset		
					1 Bit	8 Bits	16 Bits			
FF00H	Port 0	P0		R/W	√	√	—	00H		
FF01H	Port 1	P1			√	√	—			
FF02H	Port 2	P2			√	√	—			
FF03H	Port 3	P3			√	√	—			
FF04H	Port 4	P4			√	√	—	Undefined		
FF05H	Port 5	P5			√	√	—			
FF06H	Port 6	P6			√	√	—			
FF07H	Port 7	P7			√	√	—	00H		
FF0CH	Port 12	P12			√	√	—			
FF0DH	Port 13	P13			√	√	—			
FF10H FF11H	Capture/compare register 00	CR00			R	—	—	√	Undefined	
FF12H FF13H	Capture/compare register 01	CR01				—	—	√		
FF14H FF15H	16-bit timer register	TM0			R	—	—	√	0000H	
FF16H	Compare register 10	CR10		R/W	—	√	—	Undefined		
FF17H	Compare register 20	CR20			—	√	—			
FF18H	8-bit timer register 1	TMS	TM1	R	—	√	√	00H		
FF19H	8-bit timer register 2		TM2		—	√				
FF1AH	Serial I/O shift register 0	SIO0		R/W	—	√	—	Undefined		
FF1BH	Serial I/O shift register 1	SIO1			—	√	—			
FF1FH	A/D conversion result register	ADCR		R	—	√	—			
FF20H	Port mode register 0	PM0		R/W	√	√	—	FFH		
FF21H	Port mode register 1	PM1			√	√	—			
FF22H	Port mode register 2	PM2			√	√	—			
FF23H	Port mode register 3	PM3			√	√	—			
FF25H	Port mode register 5	PM5			√	√	—			
FF26H	Port mode register 6	PM6			√	√	—			
FF27H	Port mode register 7	PM7			√	√	—			
FF2CH	Port mode register 12	PM12			√	√	—			
FF2DH	Port mode register 13	PM13			√	√	—			
FF30H	Real-time output buffer register L	RTBL			R	—	√		—	00H
FF31H	Real-time output buffer register H	RTBH				—	√		—	
FF34H	Real-time output port mode register	RTPM				√	√		—	
FF36H	Real-time output port control register	RTPC				√	√		—	

Table 5-2. Special-Function Register List (2/3)

Address	Special-Function Register (SFR) Name	Symbol		R/W	Manipulatable Bit Unit			After Reset
					1 Bit	8 Bits	16 Bits	
FF38H FF39H	Correction address register 0 <sup>Note</sup>	CORAD0		R/W	—	—	√	0000H
FF3AH FF3BH	Correction address register 1 <sup>Note</sup>	CORAD1			—	—	√	
FF40H	Timer clock select register 0	TCL0			√	√	—	00H
FF41H	Timer clock select register 1	TCL1			—	√	—	
FF42H	Timer clock select register 2	TCL2			—	√	—	
FF43H	Timer clock select register 3	TCL3			—	√	—	88H
FF47H	Sampling clock select register	SCS			—	√	—	00H
FF48H	16-bit timer mode control register	TMC0			√	√	—	
FF49H	8-bit timer mode control register 1	TMC1			√	√	—	
FF4AH	Watch timer mode control register	TMC2			√	√	—	
FF4CH	Capture/compare control register 0	CRC0			√	√	—	04H
FF4EH	16-bit timer output control register	TOC0			√	√	—	00H
FF4FH	8-bit timer output control register	TOC1			√	√	—	
FF60H	Serial operating mode register 0	CSIM0			√	√	—	
FF61H	Serial bus interface control register	SBIC			√	√	—	
FF62H	Slave address register	SVA			—	√	—	Undefined
FF63H	Interrupt timing specify register	SINT			√	√	—	00H
FF68H	Serial operating mode register 1	CSIM1			√	√	—	
FF69H	Automatic data transmit/receive control register	ADTC			√	√	—	
FF6AH	Automatic data transmit/receive address pointer	ADTP			—	√	—	
FF6BH	Automatic data transmit/receive interval specify register	ADTI			√	√	—	
FF70H	Asynchronous serial interface mode register	ASIM			√	√	—	
FF71H	Asynchronous serial interface status register	ASIS		R	√	√	—	
FF72H	Serial operating mode register 2	CSIM2		RW	√	√	—	
FF73H	Baud rate generator control register	BRGC			—	√	—	
FF74H	Transmit shift register	TXS	SIO2	W	—	√	—	FFH
	Receive buffer register	RXB		R				
FF75H	Serial interface pin select register	SIPS		R/W	√	√	—	00H
FF80H	A/D converter mode register	ADM			√	√	—	01H
FF84H	A/D converter input select register	ADIS			—	√	—	00H
FF8AH	Correction control register <sup>Note</sup>	CORCN			√	√	—	
FF90H	D/A conversion value setting register 0	DACS0			—	√	—	
FF91H	D/A conversion value setting register 1	DACS1			—	√	—	
FF98H	D/A converter mode register	DAM			√	√	—	

**Note** This register is provided only in the  $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A), 78F0058, and 78F0058Y.

Table 5-2. Special-Function Register List (3/3)

Address	Special-Function Register (SFR) Name	Symbol		R/W	Manipulatable Bit Unit			After Reset
					1 Bit	8 Bits	16 Bits	
FFD0H to FFD0H to FFD0H	External access area <sup>Note 1</sup>			R/W	√	√	—	Undefined
FFE0H	Interrupt request flag register 0L	IF0	IF0L	R/W	√	√	√	00H
FFE1H	Interrupt request flag register 0H		IF0H		√	√	—	
FFE2H	Interrupt request flag register 1L	IF1L	√		√	—		
FFE4H	Interrupt mask flag register 0L	MK0	MK0L		√	√	√	FFH
FFE5H	Interrupt mask flag register 0H		MK0H		√	√	—	
FFE6H	Interrupt mask flag register 1L	MK1L	√		√	—		
FFE8H	Priority order specification flag register 0L	PR0	PR0L		√	√	√	00H
FFE9H	Priority order specification flag register 0H		PR0H		√	√	—	
FFEAH	Priority order specification flag register 1L	PR1L	√		√	—		
FFECH	External interrupt mode register 0	INTM0			—	√	—	00H
FFEDH	External interrupt mode register 1	INTM1		—	√	—		
FFF0H	Internal memory size switching register	IMS		—	√	—	Note 2	
FFF2H	Oscillation mode select register	OSMS		W	—	√	—	00H
FFF3H	Pull-up resistor option register H	PUOH		R/W	√	√	—	
FFF4H	Internal expansion RAM size switching register <sup>Note 3</sup>	IXS		W	—	√	—	0AH
FFF6H	Key return mode register	KRM		R/W	√	√	—	02H
FFF7H	Pull-up resistor option register L	PUOL			√	√	—	00H
FFF8H	Memory expansion mode register	MM			√	√	—	10H
FFF9H	Watchdog timer mode register	WDTM			√	√	—	00H
FFFAH	Oscillation stabilization time select register	OSTS			—	√	—	04H
FFFBH	Processor clock control register	PCC			√	√	—	

- Notes**
- The external access area cannot be accessed using SFR addressing. Access the area using direct addressing.
  - The value after reset depends on the product.  
 $\mu$ PD780053, 780053(A), 780053Y, 780053Y(A): C6H  
 $\mu$ PD780054, 780054(A), 780054Y, 780054Y(A): C8H  
 $\mu$ PD780055, 780055(A), 780055Y, 780055Y(A): CAH  
 $\mu$ PD780056, 780056(A), 780056Y, 780056Y(A): CCH  
 $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A): CFH  
 $\mu$ PD78F0058, 78F0058Y: CFH
  - This register is provided only in the  $\mu$ PD780058, 780058B, 780058B(A), 780058BY, 780058BY(A), 78F0058, and 78F0058Y.

### 5.3 Instruction Address Addressing

The instruction address is determined by the program counter (PC) contents. The contents of the PC are 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 the PC and branched by the following addressing. (For details of instructions, refer to **78K/0 Instructions User's Manual (U12326E)**).

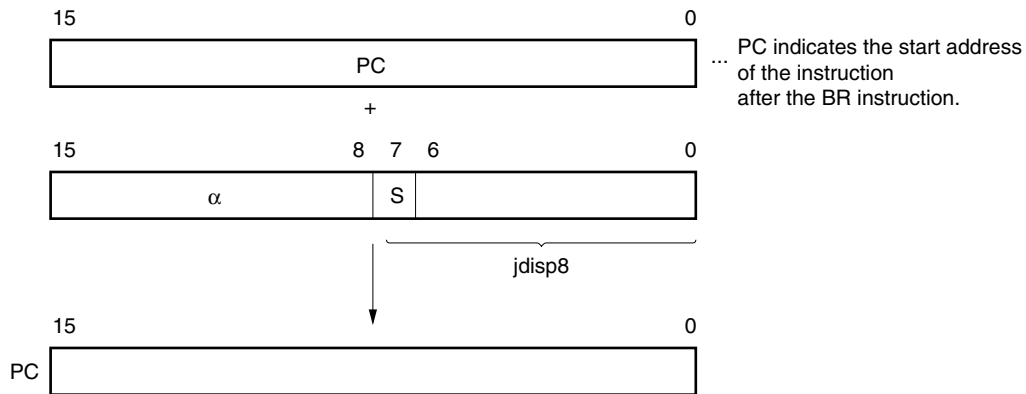
#### 5.3.1 Relative addressing

**[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 the relative addressing modes, execution branches in a relative range of −128 to +127 from the first address of the next instruction.

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

**[Illustration]**



When S = 0, all bits of α are 0.  
 When S = 1, all bits of α are 1.



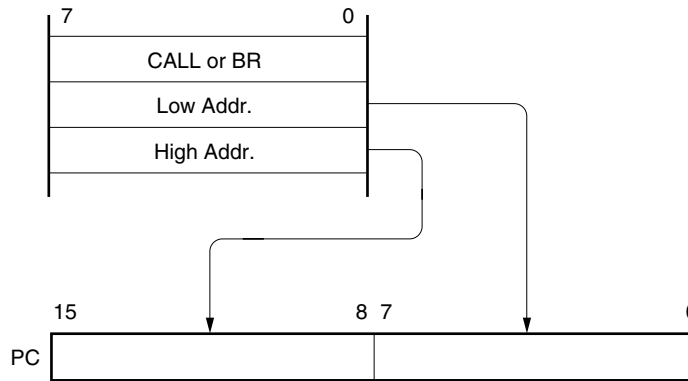
5.3.2 Immediate addressing

[Function]

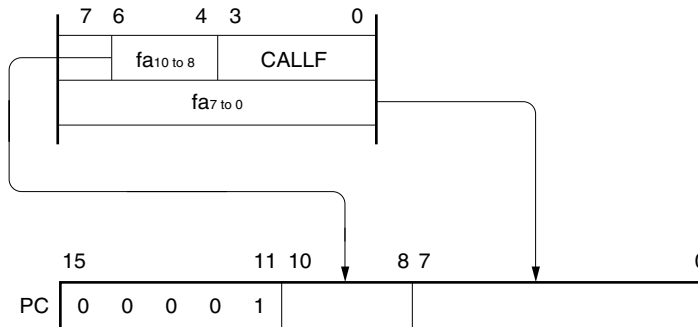
Immediate data in the instruction word is transferred to the program counter (PC) and branched. This function is carried out when the CALL !addr16 or BR !addr16 or CALLF !addr11 instruction is executed. The CALL !addr16 and BR !addr16 instructions can branch in the entire memory space. The CALLF !addr11 instruction branches to an area of addresses 0800H to 0FFFH.

[Illustration]

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



In the case of CALLF !addr11 instruction



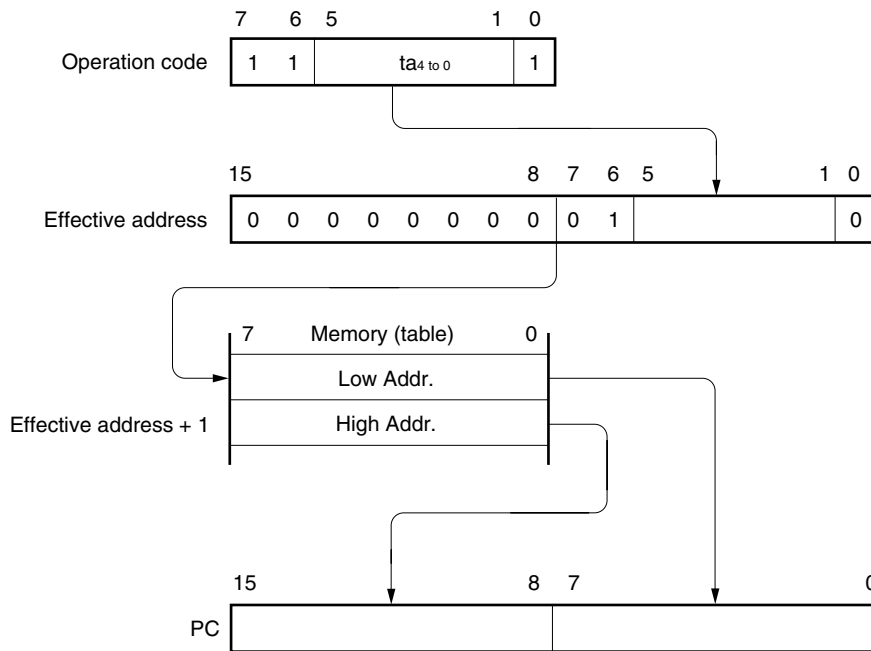
5.3.3 Table indirect addressing

[Function]

The 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.

Before the CALLT [addr5] instruction is executed, table indirect addressing is performed. This instruction references an address stored in the memory table at addresses 40H to 7FH, and can branch in the entire memory space.

[Illustration]

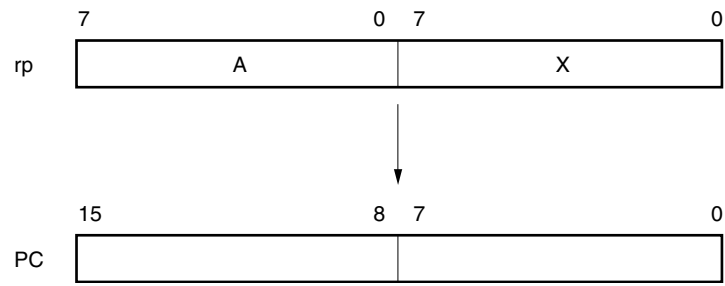


### 5.3.4 Register addressing

**[Function]**

The register pair (AX) contents to be specified by 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]**

## 5.4 Operand Address Addressing

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

### 5.4.1 Implied addressing

#### [Function]

The register which functions as an accumulator (A and AX) in the general-purpose register area is automatically (illicitly) addressed.

In the  $\mu$ PD780058 and 780058Y Subseries instruction words, the following instructions employ implied addressing.

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

#### [Operand format]

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

#### [Description example]

In the case of MULU X

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

5.4.2 Register addressing

**[Function]**

This addressing accesses a general-purpose register as an operand. The general-purpose register accessed is specified by the register bank select flags (RBS0 and RBS1) and register specification code (Rn or RPN) in an instruction 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 by 3 bits in the operation code.

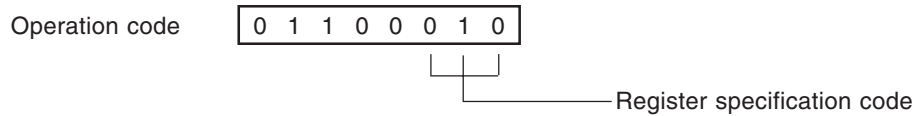
**[Operand format]**

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

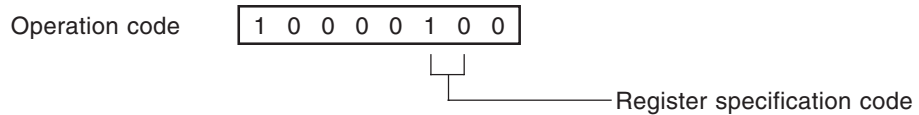
'r' and 'rp' can be described with function names (X, A, C, B, E, D, L, H, AX, BC, DE, and HL) as well as absolute names (R0 to R7 and RP0 to RP3).

**[Description example]**

MOV A, C; when selecting C register as r



INCW DE; when selecting DE register pair as rp



### 5.4.3 Direct addressing

**[Function]**

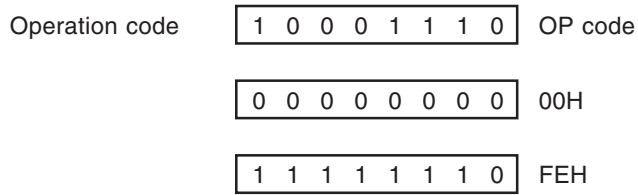
This addressing directly addresses the memory indicated by the immediate data in an instruction word.

**[Operand format]**

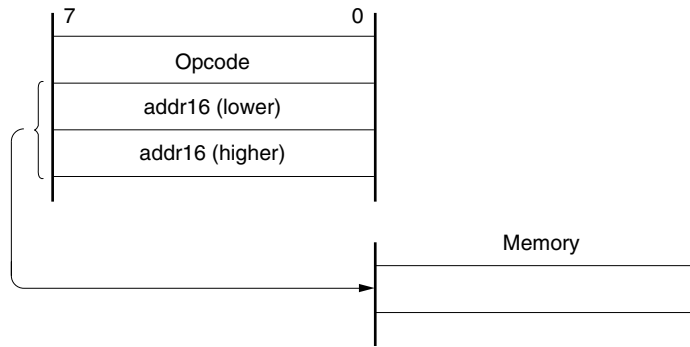
Identifier	Description
addr16	Label or 16-bit immediate data

**[Description example]**

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



**[Illustration]**



#### 5.4.4 Short direct addressing

**[Function]**

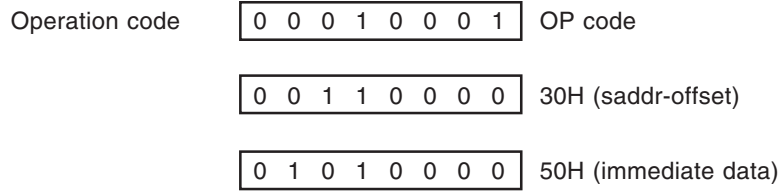
The memory to be manipulated in the fixed space is directly addressed with 8-bit data in an instruction word. The fixed space to which this address is applied is a 256-byte space of addresses FE20H to FF1FH. An internal RAM and special-function registers (SFRs) are mapped at FE20H to FEFFH and FF00H to FF1FH, respectively. The SFR area (FF00H to FF1FH) to which short direct addressing is applied is a part of the entire SFR area. Ports frequently accessed by the program, and the compare registers and capture registers of timer/event counters are mapped to this area. These SFRs can be manipulated with a short byte length and few 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. See **[Illustration]** on next page.

**[Operand format]**

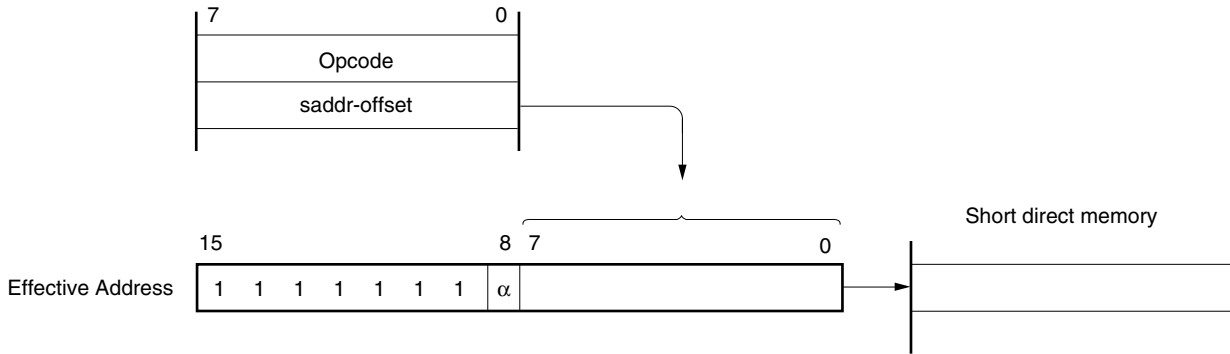
Identifier	Description
saddr	Label or immediate data of FE20H to FF1FH
saddrp	Label or immediate data of FE20H to FF1FH (even address only)

[Description example]

MOV 0FE30H, #50H; when setting saddr to FE30H and immediate data to 50H



[Illustration]



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

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



## 5.4.5 Special-Function Register (SFR) addressing

**[Function]**

The memory-mapped special-function registers (SFRs) are 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]**

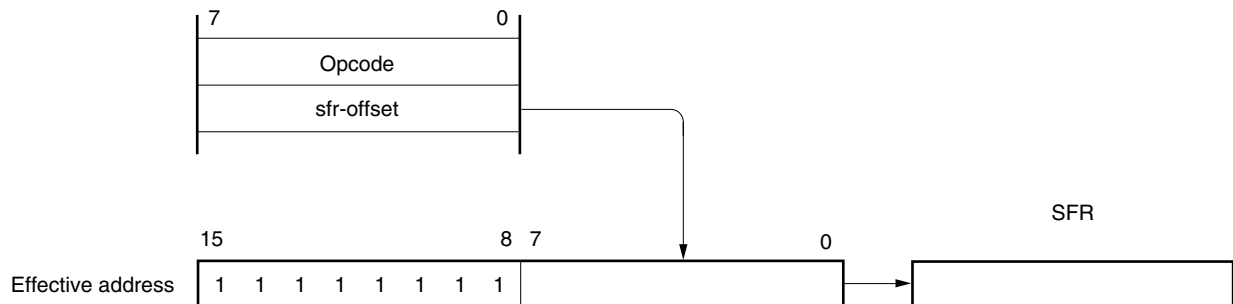
Identifier	Description
sfr	Special-function register name
sfrp	16-bit manipulatable special-function register name (even address only)

**[Description example]**

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

Operation code    1 1 1 1 0 1 1 0    OP code

0 0 1 0 0 0 0 0    20H (sfr-offset)

**[Illustration]**

5.4.6 Register indirect addressing

[Function]

This addressing addresses the memory with the contents of a register pair specified as an operand. The register pair to be accessed is specified by the register bank select flags (RBS0 and RBS1) and register pair specification code in an instruction code. This addressing can be carried out for all the memory spaces.

[Operand format]

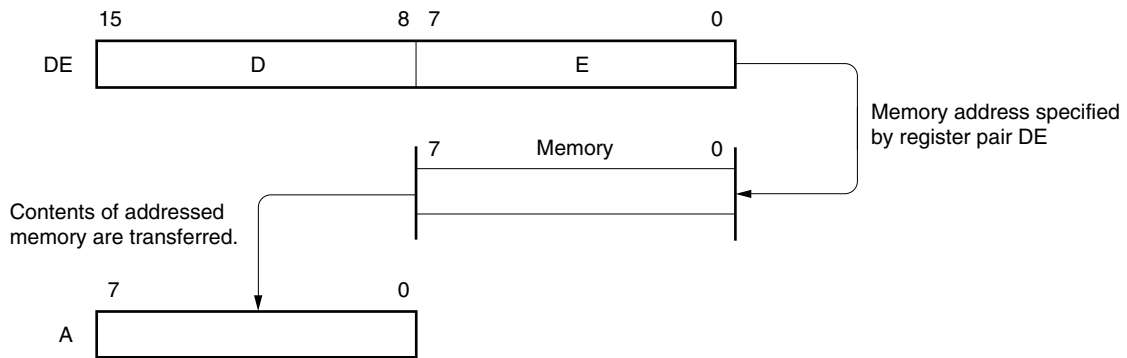
Identifier	Description
—	[DE], [HL]

[Description example]

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

Operation code 1 0 0 0 0 1 0 1

[Illustration]



### 5.4.7 Based addressing

#### [Function]

This addressing addresses the memory by adding 8-bit immediate data to the contents of the HL register pair which is used as a base register and by using the result of the addition. The HL register pair to be accessed is in the register bank specified by the register bank select flags (RBS0 and RBS1). The 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 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
---	---	---	---	---	---	---	---

### 5.4.8 Based indexed addressing

#### [Function]

This addressing addresses the memory by adding the contents of the HL register, which is used as a base register, to the contents of the B or C register specified in the instruction word, and by using the result of the addition. The HL, B, and C registers to be accessed are registers in the register bank specified by the register bank select flags (RBS0 and RBS1). The addition is performed by extending the contents of the B or C register to 16 bits as a positive number. A carry from the 16th bit is ignored. This addressing can be carried out for all the memory spaces.

#### [Operand format]

Identifier	Description
—	[HL + B], [HL + C]

#### [Description example]

In the case of MOV A, [HL + B]

Operation code 

1	0	1	0	1	0	1	1
---	---	---	---	---	---	---	---

### 5.4.9 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. Stack addressing can be used to address the internal high-speed RAM area only.

#### [Description example]

In the case of PUSH DE

Operation code 

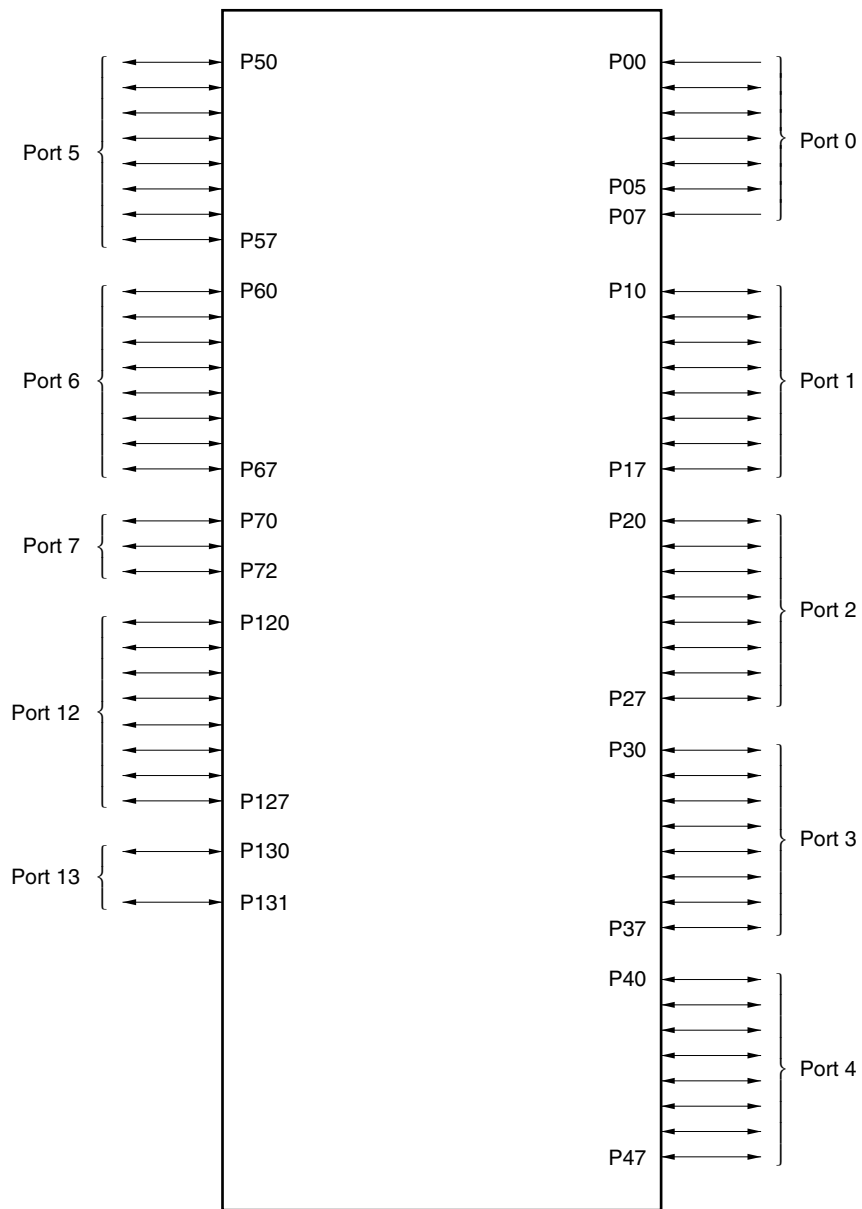
1	0	1	1	0	1	0	1
---	---	---	---	---	---	---	---

## CHAPTER 6 PORT FUNCTIONS

### 6.1 Port Functions

The  $\mu$ PD780058 and 780058Y Subseries incorporate two input ports and sixty-six I/O ports. Figure 6-1 shows the port types. Every port can be manipulated in 1-bit and 8-bit units and can carry out considerably varied control operations. Besides port functions, the ports can also serve as on-chip hardware I/O pins.

Figure 6-1. Port Types



**Table 6-1. Port Functions (μPD780058 Subseries) (1/2)**

Pin Name	Function	Alternate Function
P00	Port 0 7-bit I/O port  Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	Input only
P01		INTP0/TI00
P02		INTP1/TI01
P03		INTP2
P04		INTP3
P05		INTP4
P07		INTP5
P10 to P17	Port 1 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	XT1
P20	Port 2 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	ANI0 to ANI7
P21		SI1
P22		SO1
P23		$\overline{\text{SCK1}}$
P24		STB/TxD1
P25		BUSY/RxD1
P26		SI0/SB0
P27		SO0/SB1
P30	Port 3 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	$\overline{\text{SCK0}}$
P31		TO0
P32		TO1
P33		TO2
P34		TI1
P35		TI2
P36		PCL
P37		BUZ
P40 to P47	Port 4 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software. The test input flag (KRIF) is set to 1 by falling edge detection.	—
P50 to P57	Port 5 8-bit I/O port LED can be driven directly. Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	AD0 to AD7
		A8 to A15

Table 6-1. Port Functions ( $\mu$ PD780058 Subseries) (2/2)

Pin Name	Function		Alternate Function	
P60	Port 6 8-bit I/O port Input/output can be specified in 1-bit units.	N-ch open-drain I/O port On-chip pull-up resistors can be specified by mask option. (Mask ROM version only). LEDs can be driven directly.	—	
P61				
P62				
P63				
P64			If used as an input port, an on-chip pull-up resistor can be connected by setting software.	$\overline{\text{RD}}$
P65				$\overline{\text{WR}}$
P66				$\overline{\text{WAIT}}$
P67				ASTB
P70	Port 7 3-bit I/O port Input/output can be specified in 1-bit units.	If used as an input port, an on-chip pull-up resistor can be connected by setting software.	SI2/RxD0	
P71			SO2/TxD0	
P72			$\overline{\text{SCK2/ASCK}}$	
P120 to P127	Port 12 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, on-chip pull-up resistor can be connected by setting software.		RTP0 to RTP7	
P130 and P131	Port 13 2-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, on-chip pull-up resistor can be connected by setting software.		ANO0, ANO1	

**Table 6-2. Port Functions ( $\mu$ PD780058Y Subseries) (1/2)**

Pin Name	Function		Alternate Function
P00	Port 0 7-bit I/O port	Input only	INTP0/TI00
P01		Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.	INTP1/TI01
P02			INTP2
P03			INTP3
P04			INTP4
P05			INTP5
P07		Input only	XT1
P10 to P17	Port 1 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		ANI0 to ANI7
P20	Port 2 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		SI1
P21			SO1
P22			SCK1
P23			STB/TxD1
P24			BUSY/RxD1
P25			SI0/SB0/SDA0
P26			SO0/SB1/SDA1
P27			SCK0/SCL
P30	Port 3 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		TO0
P31			TO1
P32			TO2
P33			TI1
P34			TI2
P35			PCL
P36			BUZ
P37			—
P40 to P47	Port 4 8-bit I/O port Input/output can be specified in 8-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software. The test input flag (KRIF) is set to 1 by falling edge detection.		AD0 to AD7
P50 to P57	Port 5 8-bit I/O port LEDs can be driven directly. Input/output can be specified in 1-bit units. If used as an input port, an on-chip pull-up resistor can be connected by setting software.		A8 to A15



Table 6-2. Port Functions ( $\mu$ PD780058Y Subseries) (2/2)

Pin Name	Function		Alternate Function	
P60	Port 6 8-bit I/O port Input/output can be specified in 1-bit units.	N-ch open-drain I/O port On-chip pull-up resistors can be specified by mask option. (Mask ROM version only). LEDs can be driven directly.	—	
P61				
P62				
P63				
P64			If used as an input port, an on-chip pull-up resistor can be connected by setting software.	$\overline{\text{RD}}$
P65				$\overline{\text{WR}}$
P66				$\overline{\text{WAIT}}$
P67				$\overline{\text{ASTB}}$
P70	Port 7 3-bit I/O port Input/output can be specified in 1-bit units.	If used as an input port, an on-chip pull-up resistor can be connected by setting software.	$\overline{\text{SI2/RxD0}}$	
P71			$\overline{\text{SO2/TxD0}}$	
P72			$\overline{\text{SCK2/ASCK}}$	
P120 to P127	Port 12 8-bit I/O port Input/output can be specified in 1-bit units. If used as an input port, on-chip pull-up resistor can be connected by setting software.		RTP0 to RTP7	
P130 and P131	Port 13 2-bit I/O port Input/output mode can be specified in 1-bit units. If used as an input port, on-chip pull-up resistor can be connected by setting software.		ANO0, ANO1	

## 6.2 Port Configuration

A port consists of the following hardware.

**Table 6-3. Port Configuration**

Item	Configuration
Control register	Port mode register (PMm: m = 0 to 3, 5 to 10, 12, 13) Pull-up resistor option register (PUOH, PUOL) Memory expansion mode register (MM) <sup>Note</sup> Key return mode register (KRM)
Port	Total: 68 (Input: 2, I/O: 66)
Pull-up resistor	<ul style="list-style-type: none"> <li>• Mask ROM version Total: 66 (software specifiable: 62, mask option: 4)</li> <li>• Flash memory version Total: 62</li> </ul>

**Note** MM specifies the input/output mode of port 4.

### 6.2.1 Port 0

Port 0 is a 7-bit I/O port with an output latch. Pins P01 to P05 can be set to input or output mode in 1-bit units using port mode register 0 (PM0). Pins P00 and P07 are input-only ports. When pins P01 to P05 are used as input ports, an on-chip pull-up resistor can be connected to them in 6-bit units using pull-up resistor option register L (PUOL).

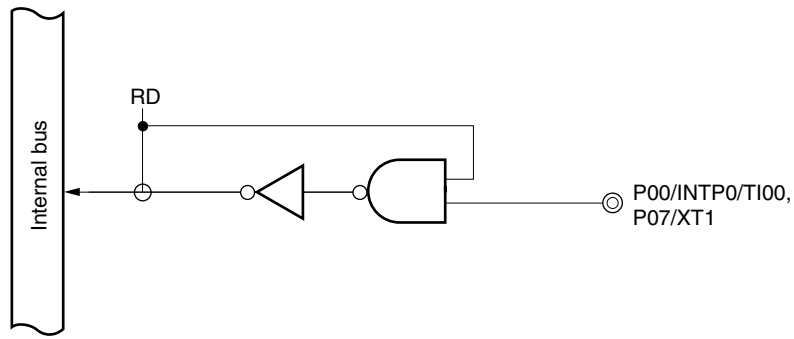
Alternate functions include external interrupt request input, external count clock input to the timer and crystal connection for subsystem clock oscillation.

$\overline{\text{RESET}}$  input sets port 0 to input mode.

Figures 6-2 and 6-3 show block diagrams of port 0.

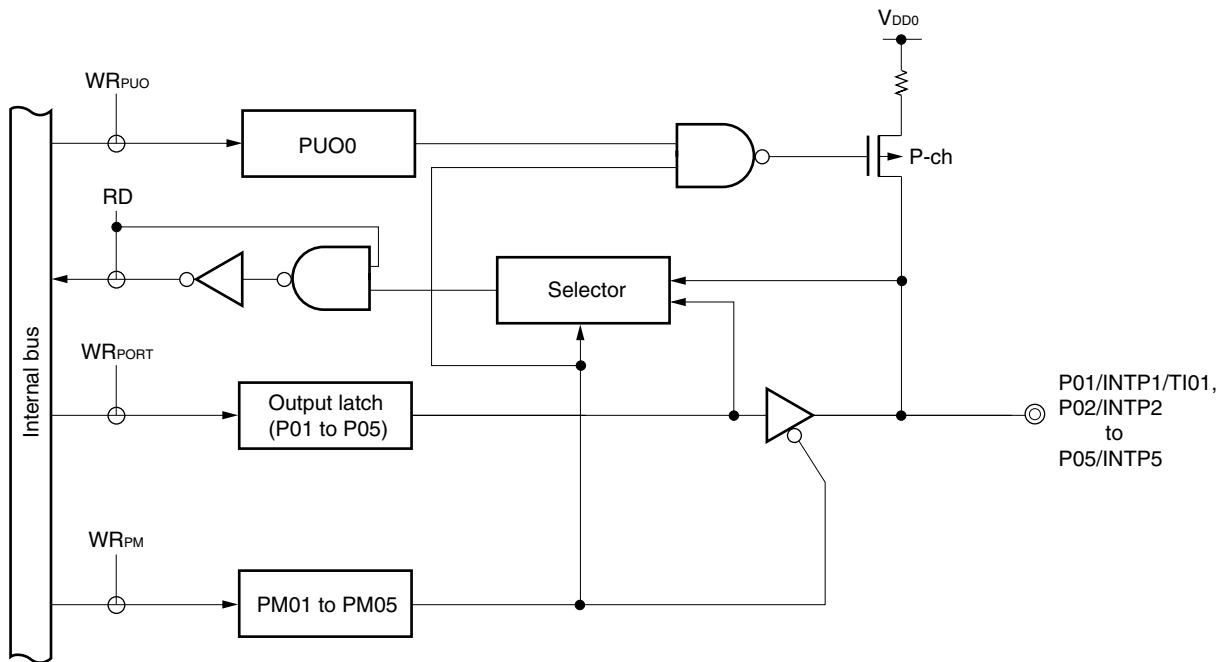
**Caution** Because port 0 also serves as external interrupt request input, when the port function output mode is specified and the output level is changed, the interrupt request flag is set. Thus, when the output mode is used, set the interrupt mask flag to 1.

Figure 6-2. Block Diagram of P00 and P07



RD: Port 0 read signal

Figure 6-3. Block Diagram of P01 to P05



PUO: Pull-up resistor option register  
 PM: Port mode register  
 RD: Port 0 read signal  
 WR: Port 0 write signal

6.2.2 Port 1

Port 1 is an 8-bit I/O port with an output latch. Port 1 can be set to input or output mode in 1-bit units using port mode register 1 (PM1). When pins P10 to P17 are used as an input port, an on-chip pull-up resistor can be connected to them in 8-bit units using pull-up resistor option register L (PUOL).

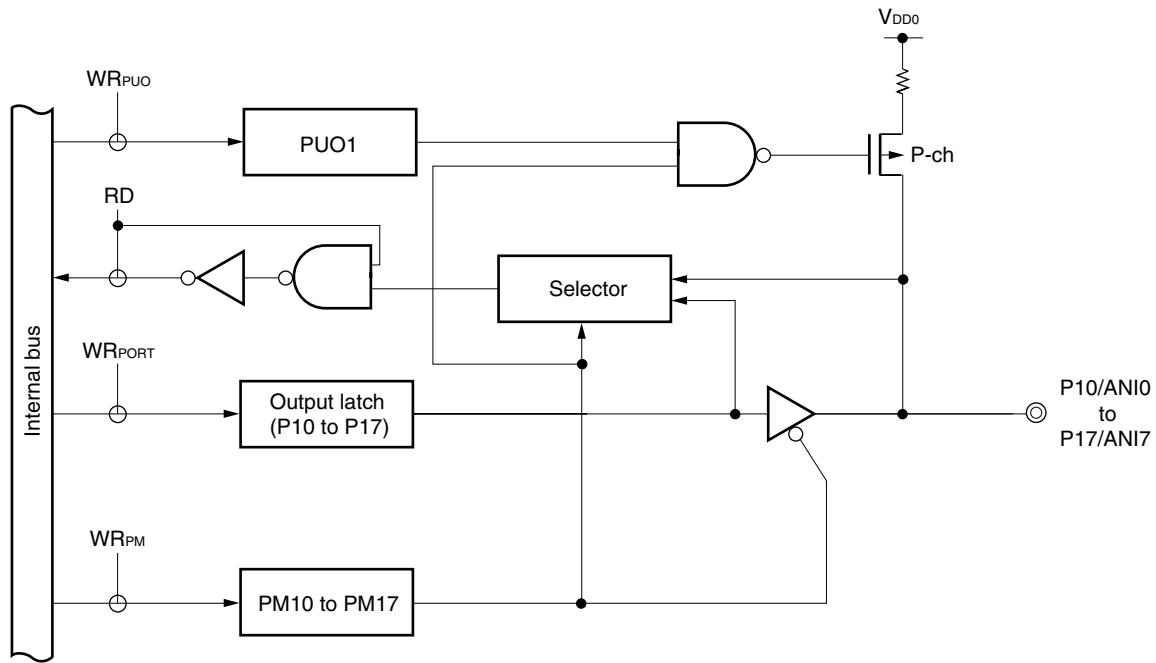
Alternate functions include A/D converter analog input.

$\overline{\text{RESET}}$  input sets port 1 to input mode.

Figure 6-4 shows a block diagram of port 1.

**Caution** A pull-up resistor cannot be used for pins used as A/D converter analog inputs.

Figure 6-4. Block Diagram of P10 to P17



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 1 read signal
- WR: Port 1 write signal

**6.2.3 Port 2 ( $\mu$ PD780058 Subseries)**

Port 2 is an 8-bit I/O port with an output latch. Pins P20 to P27 can be set to input or output mode in 1-bit units using port mode register 2 (PM2). When pins P20 to P27 are used as an input port, an on-chip pull-up resistor can be connected to them in 8-bit units using pull-up resistor option register L (PUOL).

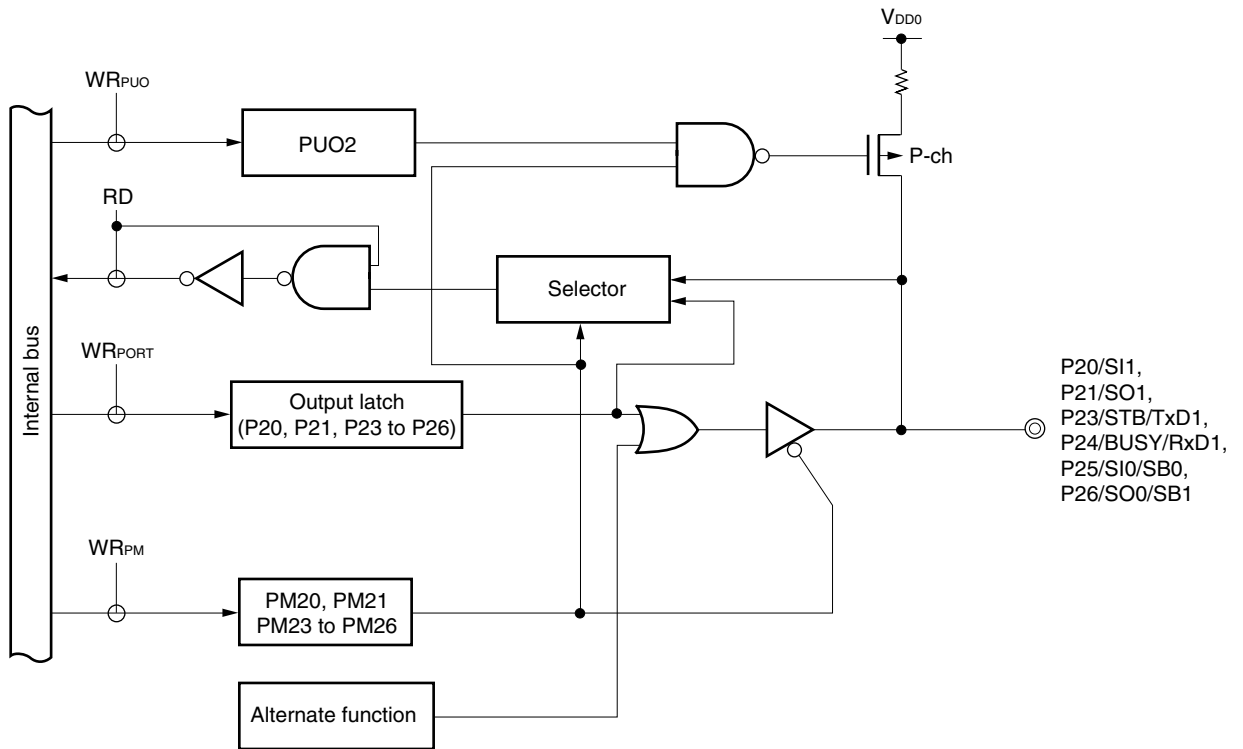
Alternate functions include serial interface data I/O, clock I/O, automatic transmit/receive busy input, and strobe output.

$\overline{\text{RESET}}$  input sets port 2 to input mode.

Figures 6-5 and 6-6 show a block diagram of port 2.

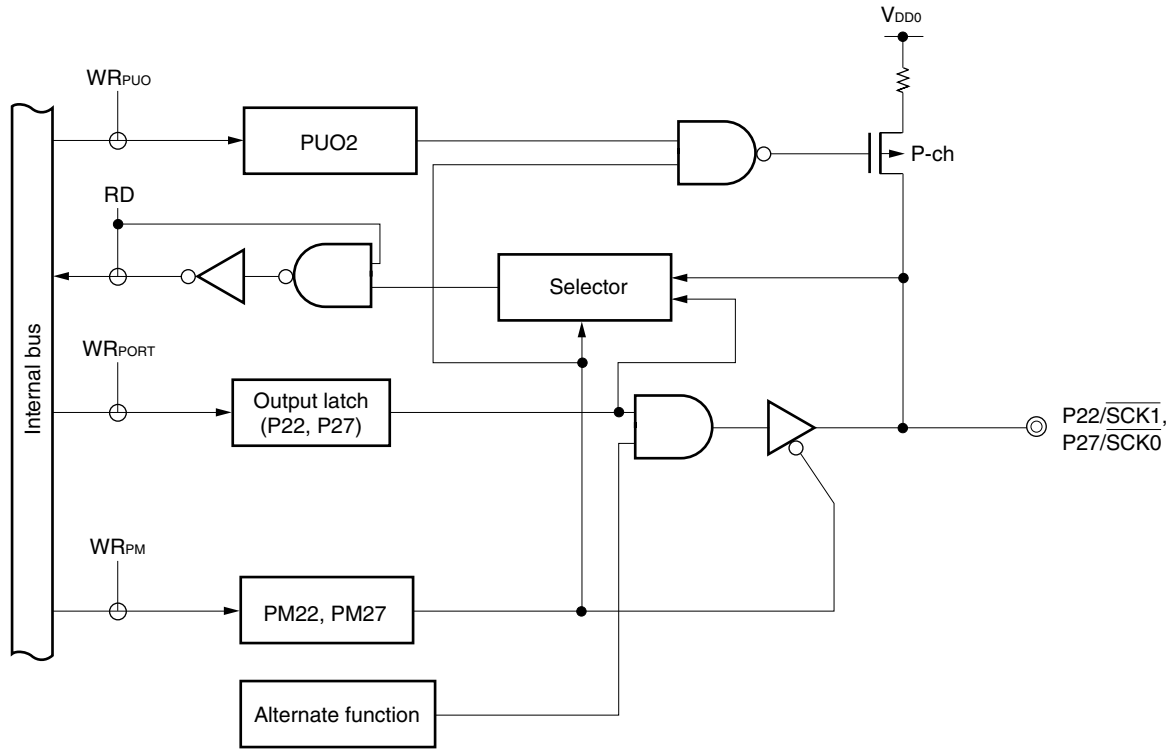
- Cautions 1.** When used as serial interface pins, set input/output and the output latch according to the function. For the setting method, see Figure 16-4 Format of Serial Operating Mode Register 0, Figure 18-3 Format of Serial Operating Mode Register 1, and Table 19-2 Serial Interface Channel 2 Operating Mode Settings.
- 2.** When reading the pin state in SBI mode, set the PM2n bit of PM2 to 1 (n = 5, 6) (See the description of (10) Judging busy state of slave in section 16.4.3 SBI mode operation).

Figure 6-5. Block Diagram of P20, P21, and P23 to P26



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 2 read signal
- WR: Port 2 write signal

Figure 6-6. Block Diagram of P22 and P27



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 2 read signal
- WR: Port 2 write signal

**6.2.4 Port 2 ( $\mu$ PD780058Y Subseries)**

Port 2 is an 8-bit I/O port with an output latch. Pins P20 to P27 can be set to input or output mode in 1-bit units using port mode register 2 (PM2). When pins P20 to P27 are used as an input port, an on-chip pull-up resistor can be connected to them in 8-bit units using pull-up resistor option register L (PUOL).

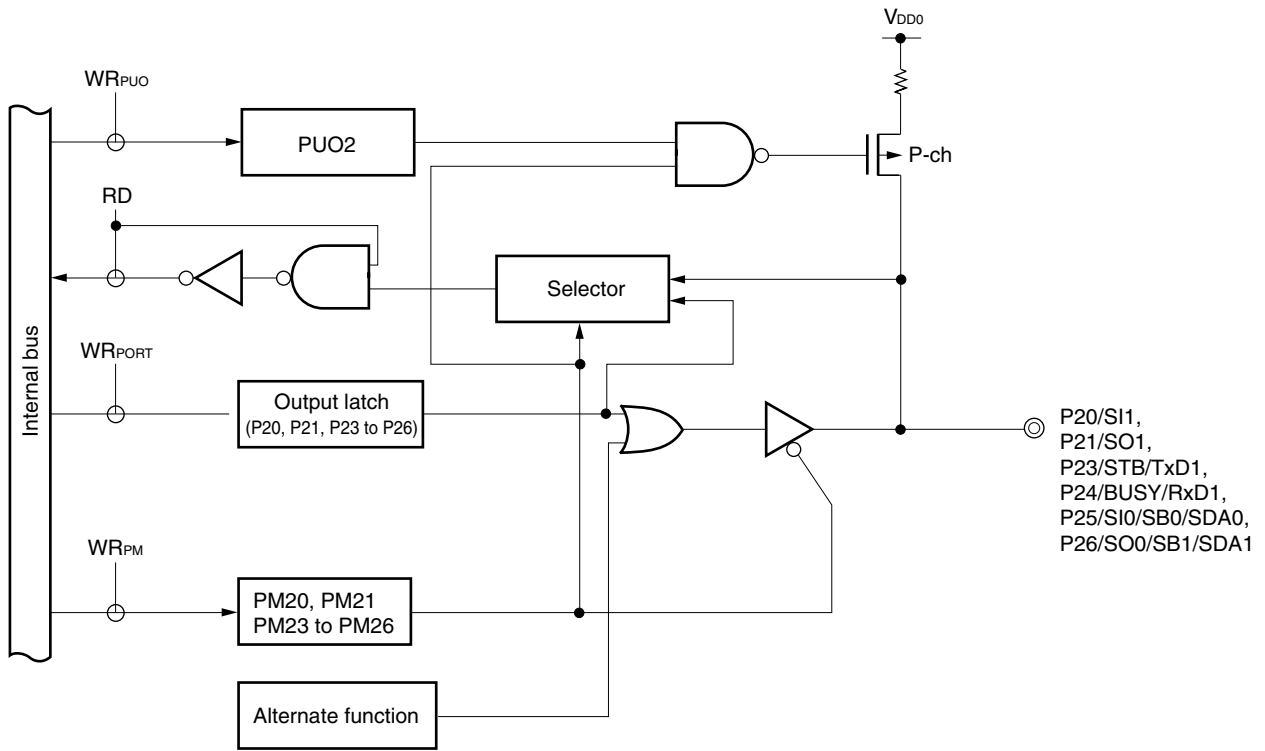
Alternate functions include serial interface data I/O, clock I/O, automatic transmit/receive busy input, and strobe output.

$\overline{\text{RESET}}$  input sets port 2 to input mode.

Figures 6-7 and 6-8 show a block diagram of port 2.

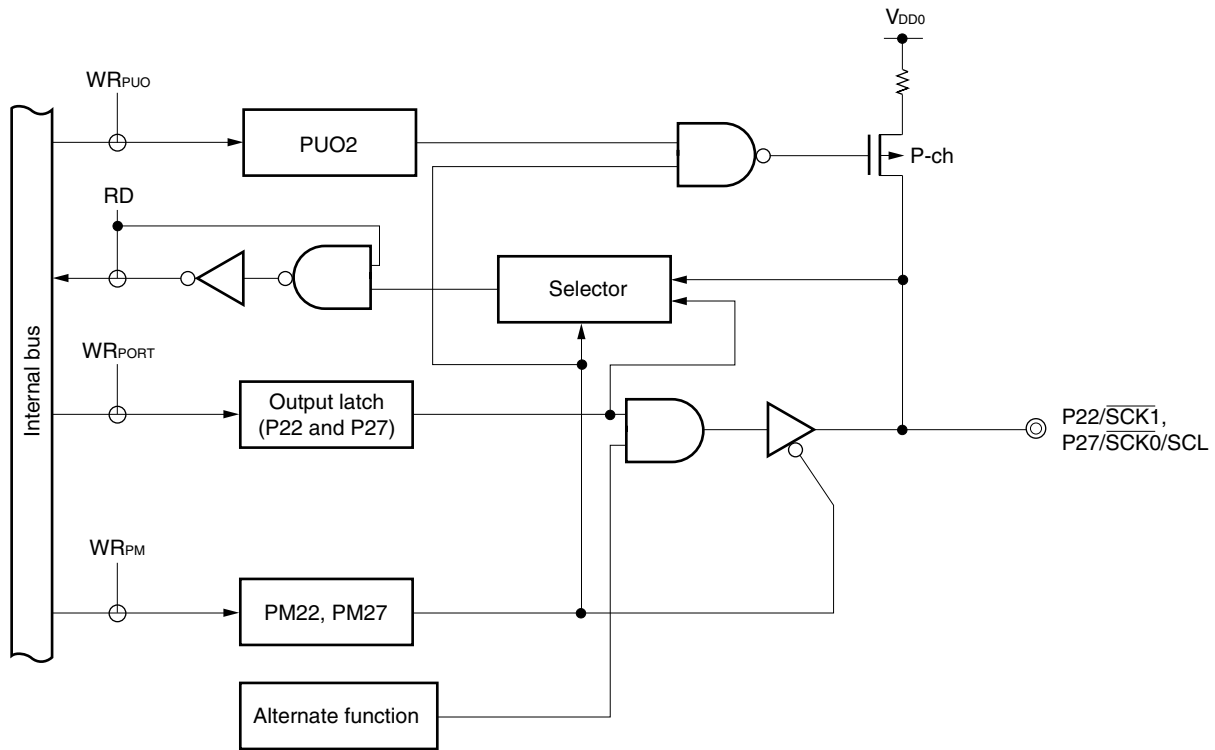
**Caution** When used as serial interface pins, set input/output and the output latch according to the function. For the setting method, see Figure 17-4 Format of Serial Operating Mode Register 0, Figure 18-3 Format of Serial Operating Mode Register 1, and Table 19-2 Serial Interface Channel 2 Operating Mode Settings.

Figure 6-7. Block Diagram of P20, P21, and P23 to P26



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 2 read signal
- WR: Port 2 write signal

Figure 6-8. Block Diagram of P22 and P27



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 2 read signal
- WR: Port 2 write signal



6.2.5 Port 3

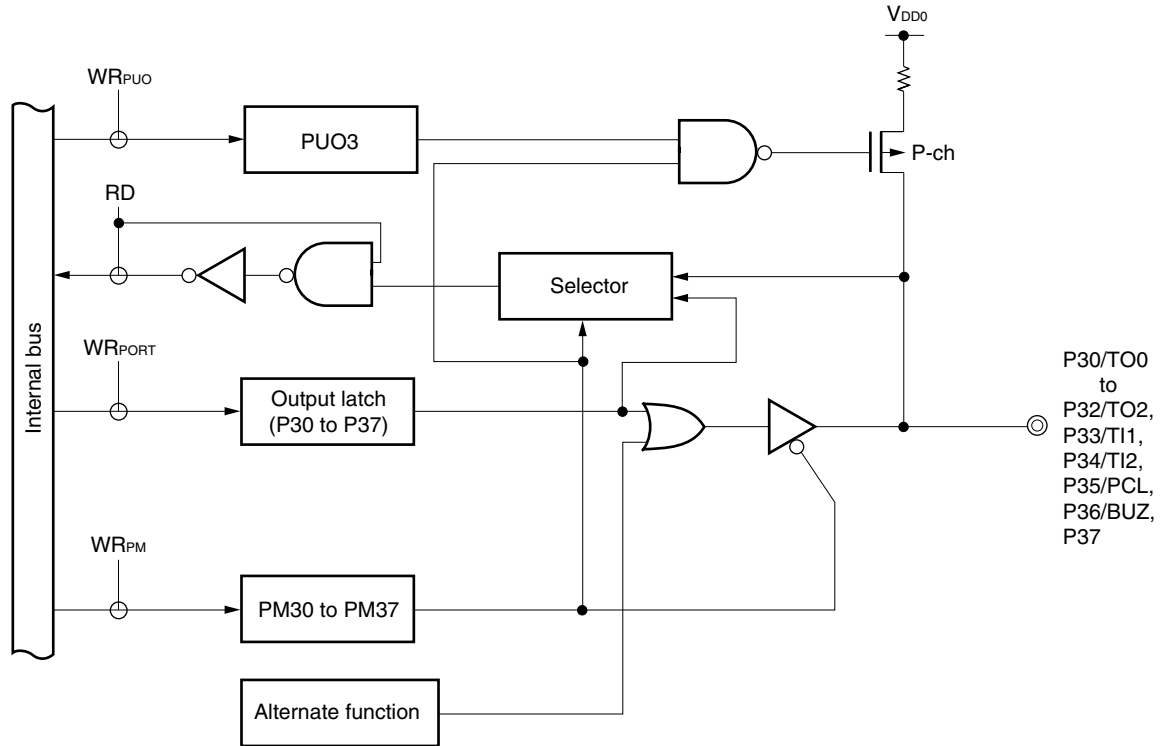
Port 3 is an 8-bit I/O port with an output latch. Pins P30 to P37 can be set to input or output mode in 1-bit units using port mode register 3 (PM3). When pins P30 to P37 are used as an input port, an on-chip pull-up resistor can be connected to them in 8-bit units using pull-up resistor option register L (PUOL).

Alternate functions include timer I/O, clock output and buzzer output.

$\overline{\text{RESET}}$  input sets port 3 to input mode.

Figure 6-9 shows a block diagram of port 3.

Figure 6-9. Block Diagram of P30 to P37



PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 3 read signal

WR: Port 3 write signal

6.2.6 Port 4

Port 4 is an 8-bit I/O port with an output latch. Pins P40 to P47 can be set to input or output mode in 8-bit units using the memory expansion mode register (MM). When pins P40 to P47 are used as an input port, an on-chip pull-up resistor can be connected to them in 8-bit units using pull-up resistor option register L (PUOL).

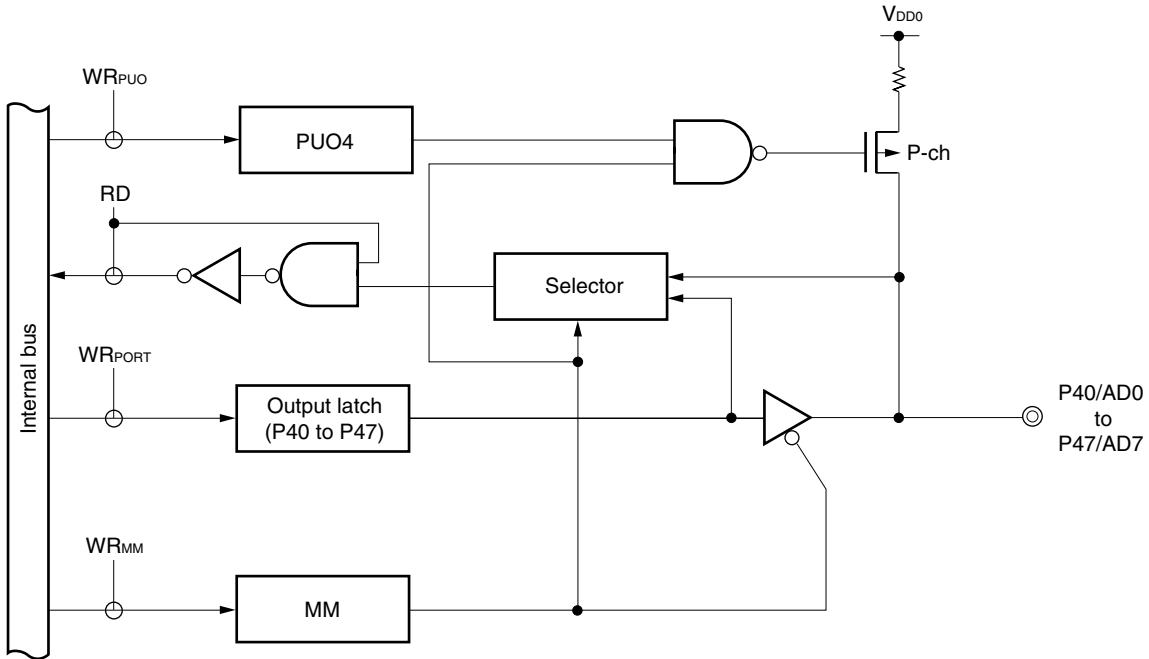
The test input flag (KRIF) can be set to 1 by detecting a falling edge.

Alternate functions include an address/data bus function in external memory expansion mode.

$\overline{\text{RESET}}$  input sets port 4 to input mode.

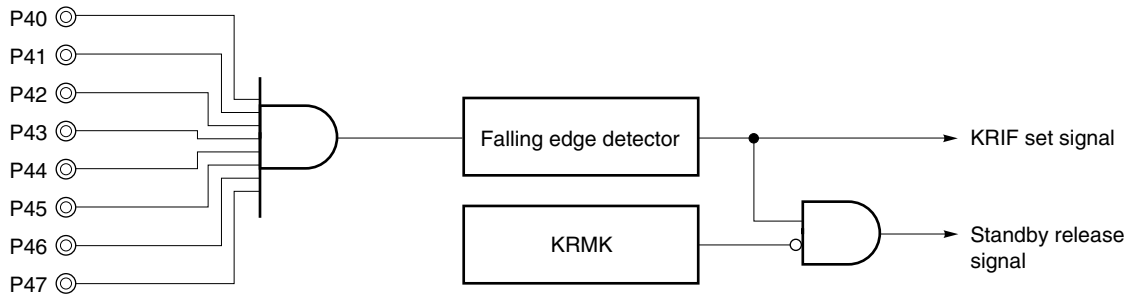
Figures 6-10 and 6-11 show a block diagram of port 4 and of the falling edge detector, respectively.

Figure 6-10. Block Diagram of P40 to P47



- PUO: Pull-up resistor option register
- MM: Memory expansion mode register
- RD: Port 4 read signal
- WR: Port 4 write signal

Figure 6-11. Block Diagram of Falling Edge Detector



**6.2.7 Port 5**

Port 5 is an 8-bit I/O port with an output latch. Pins P50 to P57 can be set to input or output mode in 1-bit units using the port mode register 5 (PM5). When pins P50 to P57 are used as an input port, an on-chip pull-up resistor can be connected to them in 8-bit units using pull-up resistor option register L (PUOL).

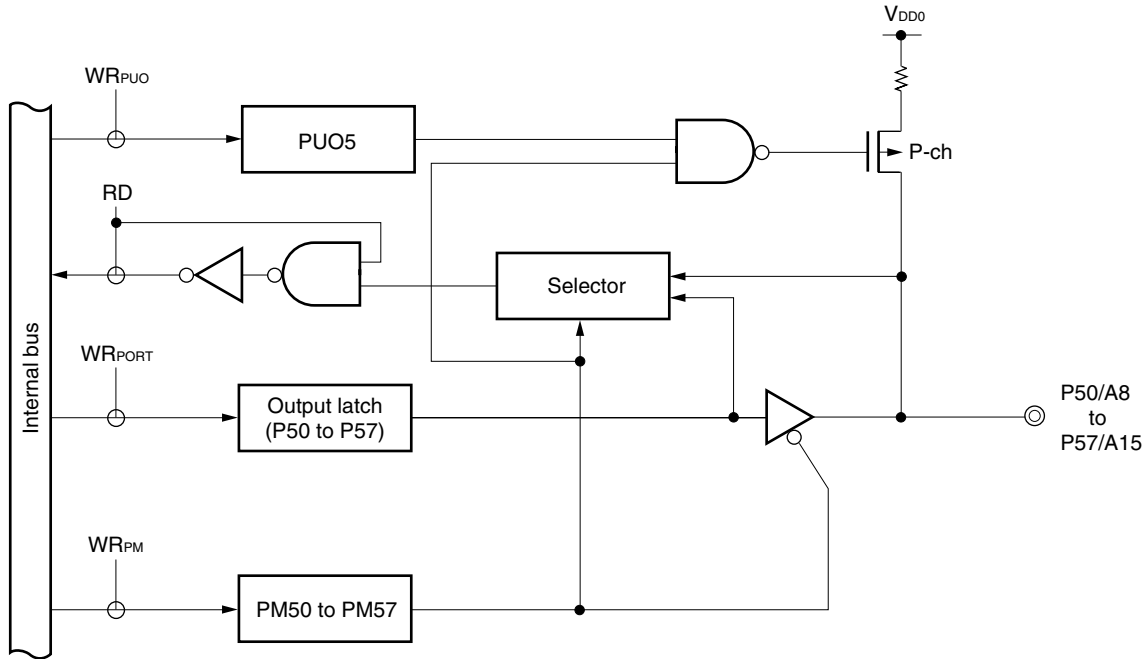
Port 5 can drive LEDs directly.

Alternate functions include an address bus function in external memory expansion mode.

$\overline{\text{RESET}}$  input sets port 5 to input mode.

Figure 6-12 shows a block diagram of port 5.

**Figure 6-12. Block Diagram of P50 to P57**



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 5 read signal
- WR: Port 5 write signal

**6.2.8 Port 6**

Port 6 is an 8-bit I/O port with an output latch. Pins P60 to P67 can be set to input or output mode in 1-bit units using port mode register 6 (PM6).

This port has functions related to pull-up resistors as shown below. These functions differ depending on whether the higher 4 bits or lower 4 bits of a port are used, and whether the mask ROM model or flash memory model is used.

**Table 6-4. Pull-up Resistor of Port 6**

	Higher 4 Bits (P64 to P67 Pins)	Lower 4 Bits (P60 to P63 Pins)
Mask ROM version	On-chip pull-up resistor can be connected in 4-bit units by PUO6	Pull-up resistor can be connected in 1-bit units by mask option
Flash memory version		Pull-up resistor is not connected

PUO6: Bit 6 of pull-up resistor option register L (PUOL)

Pins P60 to P63 can drive LEDs directly.

Alternate functions include a control signal output function in external memory expansion mode.

$\overline{\text{RESET}}$  input sets port 6 to input mode.

Figures 6-13 and 6-14 show block diagrams of port 6.

- Cautions**
1. When an external wait is not used in external memory expansion mode, P66 can be used as an I/O port.
  2. The value of the low-level input leakage current flowing to the P60 to P63 pins differ depending on the following conditions:

**[Mask ROM version]**

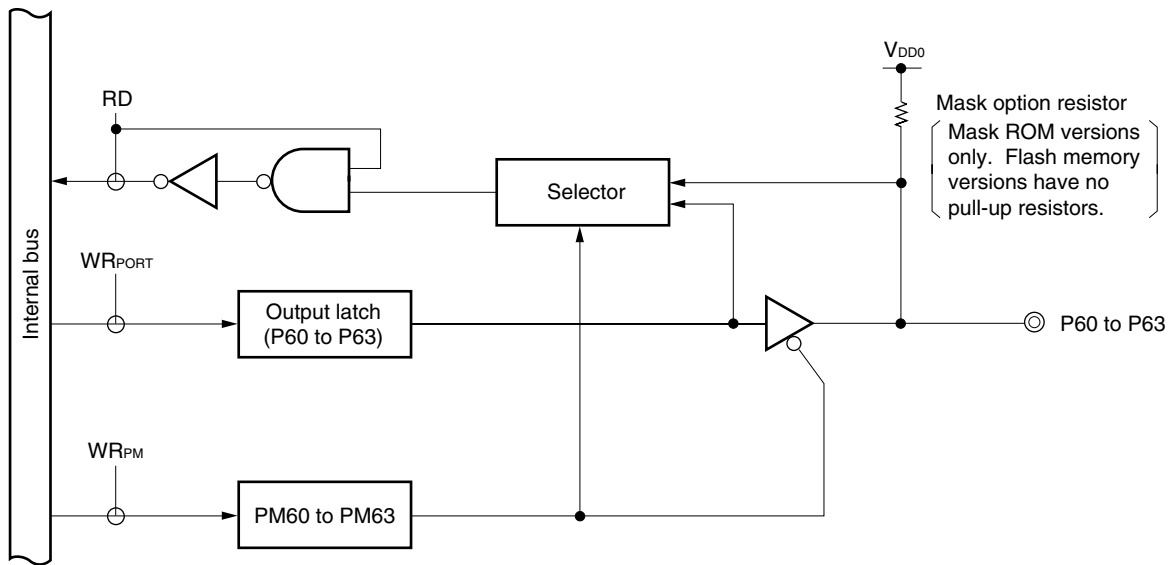
- When pull-up resistor is connected: Always  $-3 \mu\text{A}$  (MAX.)
- When pull-up resistor is not connected
  - For duration of 1.5 clocks (no wait)<sup>Note</sup> when instruction such as MOV instruction to read port 6 (P6) and port mode register 6 (PM6) is executed:  $-200 \mu\text{A}$  (MAX.)
  - Other than above:  $-3 \mu\text{A}$  (MAX.)

**[Flash memory version]**

- For duration of 1.5 clocks (no wait)<sup>Note</sup> when instruction such as MOV instruction to read port 6 (P6) and port mode register 6 (PM6) is executed:  $-200 \mu\text{A}$  (MAX.)
- Other than above:  $-3 \mu\text{A}$  (MAX.)

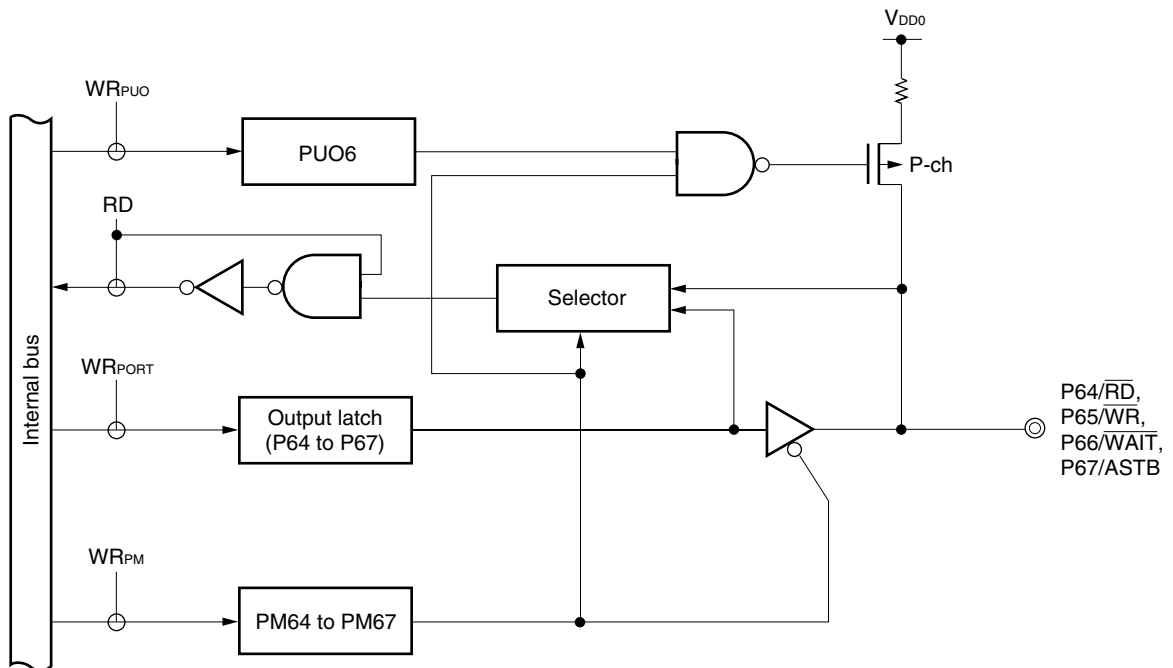
★ **Note** At this time, on-chip pull-up resistors are enabled.

Figure 6-13. Block Diagram of P60 to P63



PM: Port mode register  
 RD: Port 6 read signal  
 WR: Port 6 write signal

Figure 6-14. Block Diagram of P64 to P67



PUO: Pull-up resistor option register  
 PM: Port mode register  
 RD: Port 6 read signal  
 WR: Port 6 write signal

### 6.2.9 Port 7

This is a 3-bit I/O port with an output latch. Pins P70 to P72 can be set to input or output mode in 1-bit units using port mode register 7 (PM7). When pins P70 to P72 are used as an input port, an on-chip pull-up resistor can be connected in 3-bit units using pull-up resistor option register L (PUOL).

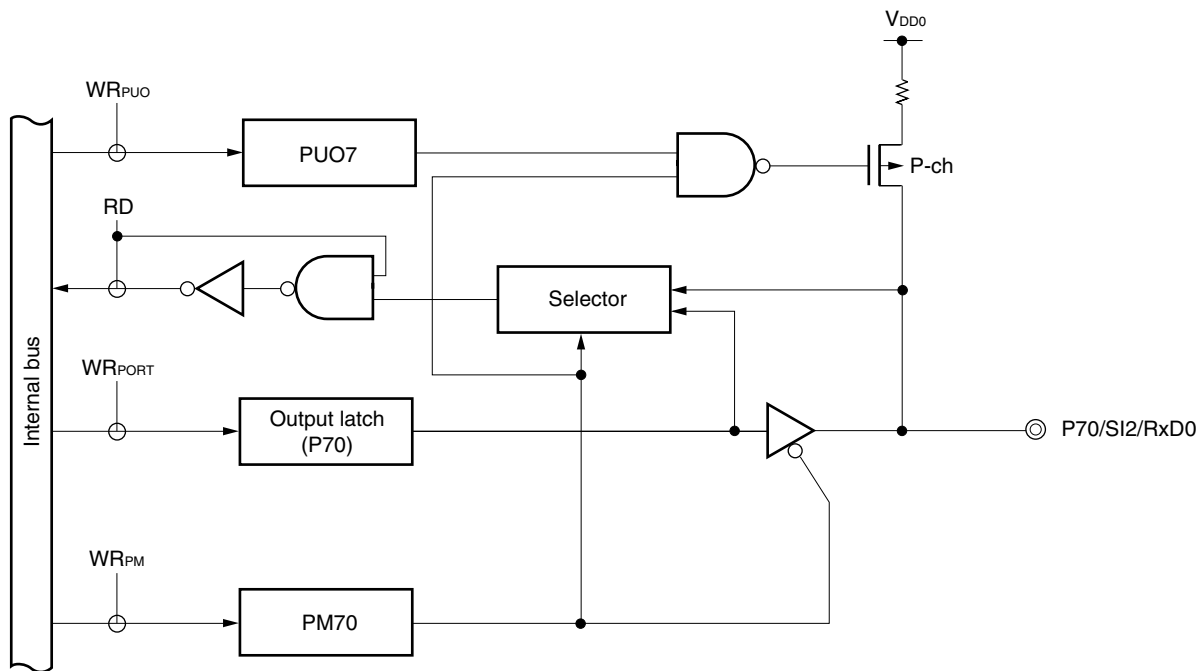
Alternate functions include serial interface channel 2 data I/O and clock I/O.

$\overline{\text{RESET}}$  input sets port 7 to input mode.

Figures 6-15 and 6-16 show a block diagram of port 7.

**Caution** When used as serial interface pins, set input/output and the output latch according to the function. For the setting method, see Table 19-2 Serial Interface Channel 2 Operating Mode Setting.

Figure 6-15. Block Diagram of P70



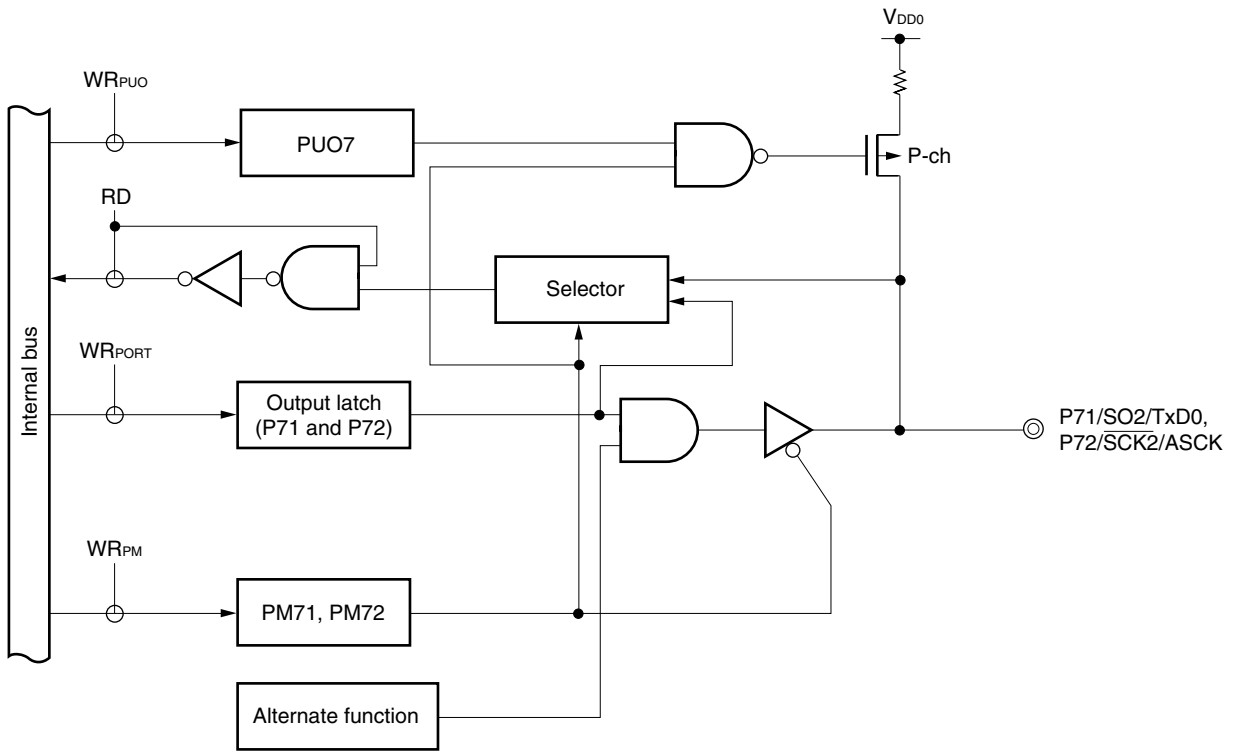
PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 7 read signal

WR: Port 7 write signal

Figure 6-16. Block Diagram of P71 and P72



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 7 read signal
- WR: Port 7 write signal

6.2.10 Port 12

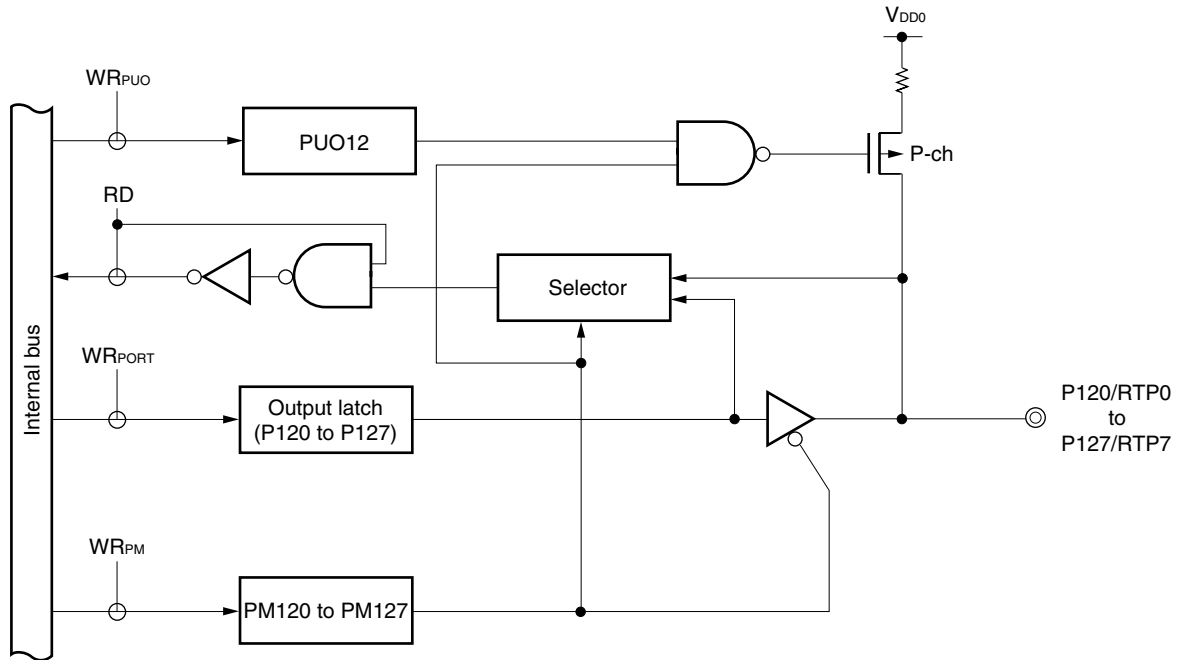
This is an 8-bit I/O port with an output latch. Pins P120 to P127 can be set to input or output mode in 1-bit units using port mode register 12 (PM12). When pins P120 to P127 are used as an input port, an on-chip pull-up resistor can be connected in 8-bit units using pull-up resistor option register H (PUOH).

These pins have an alternate function, serving as real-time outputs.

$\overline{\text{RESET}}$  input sets port 12 to input mode.

Figure 6-17 shows a block diagram of port 12.

Figure 6-17. Block Diagram of P120 to P127



- PUO: Pull-up resistor option register
- PM: Port mode register
- RD: Port 12 read signal
- WR: Port 12 write signal



### 6.2.11 Port 13

This is a 2-bit I/O port with an output latch. Pins P130 and P131 can be set to input mode/output mode in 1-bit units using port mode register 13 (PM13). When pins P130 and P131 are used as an input port, an on-chip pull-up resistor can be connected in 2-bits using pull-up resistor option register H (PUOH).

These pins have an alternate function, serving as D/A converter analog outputs.

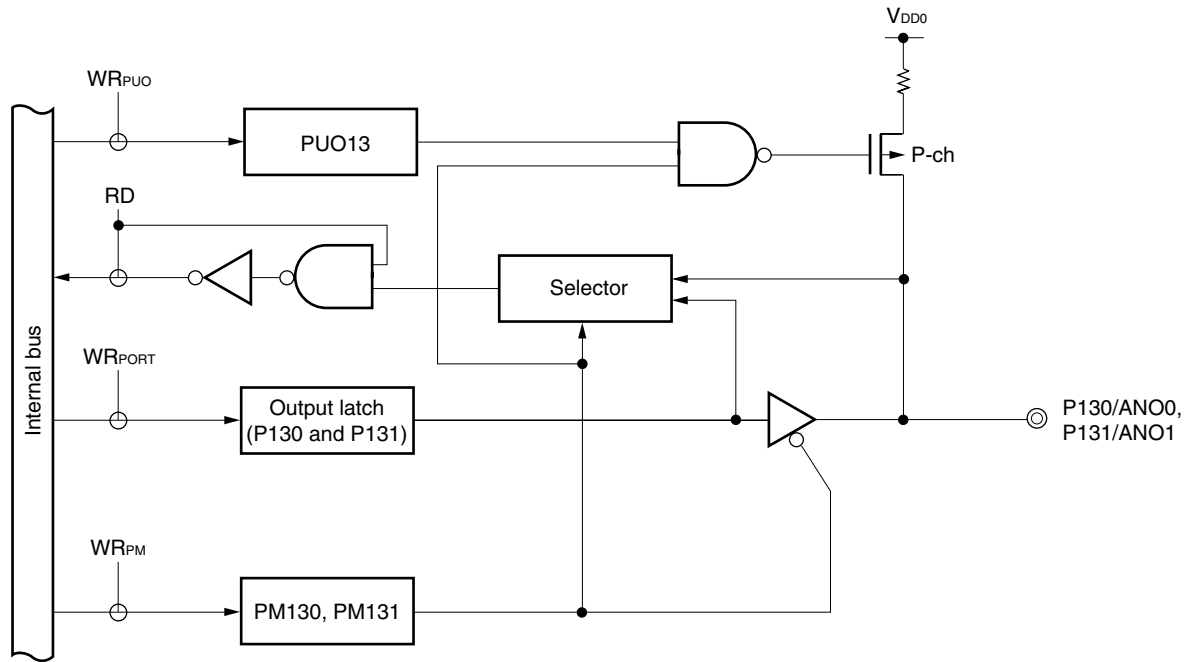
$\overline{\text{RESET}}$  input sets port 13 to input mode.

Figure 6-18 shows a block diagram of port 13.

**Caution** When only one of the D/A converter channels is used with  $\text{AV}_{\text{REF1}} < \text{V}_{\text{DD0}}$ , the other pins that are not used as analog outputs must be set as follows:

- Set the PM13 bit of port mode register 13 (PM13) to 1 (input mode) and connect the pin to  $\text{V}_{\text{SS0}}$ .
- Clear the PM13x bit of port mode register 13 (PM13) to 0 (output mode) and the output latch to 0, and output a low level from the pin.

Figure 6-18. Block Diagram of P130 and P131



PUO: Pull-up resistor option register  
 PM: Port mode register  
 RD: Port 13 read signal  
 WR: Port 13 write signal

### 6.3 Port Function Control Registers

The following four types of registers control the ports.

- Port mode registers (PM0 to PM3, PM5 to PM7, PM12, PM13)
- Pull-up resistor option registers (PUOH, PUOL)
- Memory expansion mode register (MM)
- Key return mode register (KRM)

#### (1) Port mode registers (PM0 to PM3, PM5 to PM7, PM12, PM13)

These registers are used to set port input/output in 1-bit units.

PM0 to PM3, PM5 to PM7, PM12, and PM13 are independently set with a 1-bit or 8-bit memory manipulation instruction

$\overline{\text{RESET}}$  input sets these registers to FFH.

When port pins are used as the alternate-function pins, set the port mode register and output latch according to Table 6-5.

**Cautions** 1. Pins P00 and P07 are input-only pins.

2. As port 0 has an alternate function as external interrupt request input, when the port function output mode is specified and the output level is changed, the interrupt request flag is set. When the output mode is used, therefore, the interrupt mask flag should be set to 1 beforehand.
3. The memory expansion mode register (MM) specifies the input/output mode of pins P40 to P47.

**Table 6-5. Port Mode Register and Output Latch Settings When Using Alternate Functions**

Pin Name	Alternate Function		PM <sub>xx</sub>	P <sub>xx</sub>
	Name	I/O		
P00	INTP0	Input	1 (fixed)	None
	TI00	Input	1 (fixed)	None
P01	INTP1	Input	1	×
	TI01	Input	1	×
P02 to P05	INTP2 to INTP5	Input	1	×
P07 <sup>Note 1</sup>	XT1	Input	1 (fixed)	None
P10 to P17 <sup>Note 1</sup>	ANI0 to ANI7	Input	1	×
P30 to P32	TO0 to TO2	Output	0	0
P33, P34	TI1, TI2	Input	1	×
P35	PCL	Output	0	0
P36	BUZ	Output	0	0
P40 to P47	AD0 to AD7	I/O	× <sup>Note 2</sup>	
P50 to P57	A8 to A15	Output	× <sup>Note 2</sup>	
P64	$\overline{\text{RD}}$	Output	× <sup>Note 2</sup>	
P65	$\overline{\text{WR}}$	Output	× <sup>Note 2</sup>	
P66	$\overline{\text{WAIT}}$	Input	× <sup>Note 2</sup>	
P67	ASTB	Output	× <sup>Note 2</sup>	
P120 to P127	RTP0 to RTP7	Output	0	Desired value
P130, P131 <sup>Note 1</sup>	ANO0, ANO1	Output	1	×

- Notes**
1. If these ports are read out when these pins are used in the alternate-function mode, undefined values are read.
  2. When the P40 to P47 pins, P50 to P57 pins, and P64 to P67 pins are used for alternate functions, set the function by the memory extension mode register (MM).

- Cautions**
1. When not using an external wait in the external memory extension mode, the P66 pin can be used as an I/O port.
  2. When port 2 and port 7 are used for the serial interface, input/output and the output latch must be set according to the function. For the setting methods, see Figure 16-4 Format of Serial Operation Mode Register 0, Figure 17-4 Format of Serial Operation Mode Register 0, Figure 18-3 Format of Serial Operation Mode Register 1, and Table 19-2 Serial Interface Channel 2 Operating Mode Settings.

**Remark**

- ×: don't care
- PM<sub>xx</sub>: Port mode register
- P<sub>xx</sub>: Port output latch

Figure 6-19. Port Mode Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PM0	1	1	PM05	PM04	PM03	PM02	PM01	1	FF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50	FF25H	FFH	R/W
PM6	PM67	PM66	PM65	PM64	PM63	PM62	PM61	PM60	FF26H	FFH	R/W
PM7	1	1	1	1	1	PM72	PM71	PM70	FF27H	FFH	R/W
PM12	PM127	PM126	PM125	PM124	PM123	PM122	PM121	PM120	FF2CH	FFH	R/W
PM13	1	1	1	1	1	1	PM131	PM130	FF2DH	FFH	R/W

PMmn	Pmn pin I/O mode selection (m = 0 to 3, 5 to 7, 12, 13 : n = 0 to 7)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

**(2) Pull-up resistor option registers (PUOH, PUOL)**

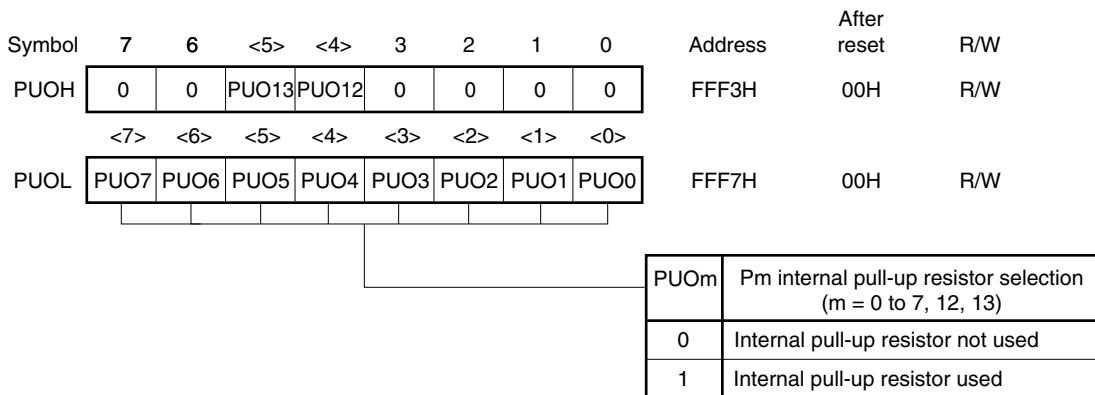
These registers are used to set whether to use an on-chip pull-up resistor at each port or not. A pull-up resistor is internally used at bits set to the input mode in a port where on-chip pull-up resistor use has been specified with PUOH, PUOL. No on-chip pull-up resistors can be used for bits set to the output mode or bits used as an analog input pin, irrespective of the PUOH or PUOL setting.

PUOH and PUOL are set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears these registers to 00H.

- Cautions**
1. Pins P00 and P07 do not incorporate a pull-up resistor.
  2. When ports 1, 4, 5, and pins P64 to P67 are used as alternate-function pins, an on-chip pull-up resistor cannot be used even if the PUOm bit of PUOH, PUOL (m = 1, 4 to 6) is set to 1.
  3. Pins P60 to P63 can be connected to pull-up resistors by a mask option only for mask ROM versions.

**Figure 6-20. Format of Pull-up Resistor Option Register**



**Caution** Be sure to clear bits 0 to 3, 6, and 7 of PUOH to 0.

**(3) Memory expansion mode register (MM)**

This register is used to set the input/output mode of port 4.

MM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets MM to 10H.

**Figure 6-21. Format of Memory Expansion Mode Register**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
MM	0	0	PW1	PW0	0	MM2	MM1	MM0	FFF8H	10H	R/W

MM2	MM1	MM0	Single-chip/memory expansion mode selection		P40 to P47, P50 to P57, P64 to P67 pin state					
					P40 to P47	P50 to P53	P54, P55	P56, P57	P64 to P67	
0	0	0	Single-chip mode		Port mode	Input	Port mode			
0	0	1								
0	1	1	Memory expansion mode	256-byte mode	AD0 to AD7	Port mode			P64 = $\overline{\text{RD}}$	
1	0	0		4 KB mode		A8 to A11	Port mode		P65 = $\overline{\text{WR}}$	
1	0	1		16 KB mode			A12, A13	Port mode	P66 = $\overline{\text{WAIT}}$	
1	1	1		Full <sup>Note</sup> address mode		A14, A15				
Other than above			Setting prohibited							

PW1	PW0	Wait control
0	0	No wait
0	1	Wait (one wait state inserted)
1	0	Setting prohibited
1	1	Wait control by external wait pin

**Note** The full address mode allows external expansion for all areas of the 64 KB address space, except the internal ROM, RAM, SFR, and use-prohibited areas.

**Remarks**

1. Pins P60 to P63 enter the port mode in both the single-chip and memory expansion mode.
2. Besides setting port 4 input/output mode, MM also sets the wait count and external expansion area.

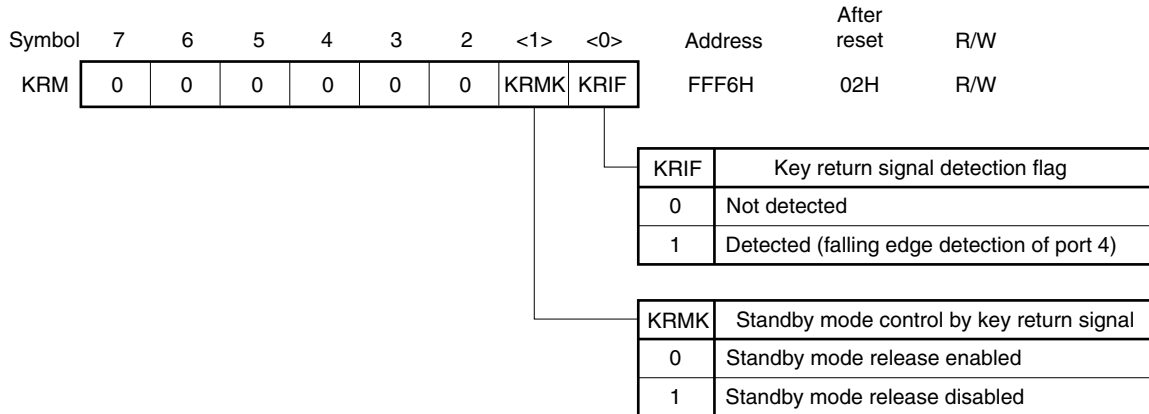
**(4) Key return mode register (KRM)**

This register sets enabling/disabling of standby function release by a key return signal (falling edge detection of port 4).

KRM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets KRM to 02H.

**Figure 6-22. Format of Key Return Mode Register**



**Caution** When falling edge detection of port 4 is used, KRIF should be cleared to 0 (it is not cleared to 0 automatically).

## 6.4 Port Operations

Port operations differ depending on whether the input or output mode is set, as shown below.

### 6.4.1 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.

#### (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.

**Caution** In the case of a 1-bit memory manipulation instruction, although a single bit is manipulated, the port is accessed as an 8-bit unit. Therefore, on a port with a mixture of input and output pins, the output latch contents for pins specified as input are undefined, even for bits other than the manipulated bit.

### 6.4.2 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.

### 6.4.3 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.

#### (2) Input mode

The output latch contents are undefined, but since the output buffer is off, the pin status does not change.

**Caution** In the case of a 1-bit memory manipulation instruction, although a single bit is manipulated, the port is accessed as an 8-bit unit. Therefore, on a port with a mixture of input and output pins, the output latch contents for pins specified as input are undefined, even for bits other than the manipulated bit.



## 6.5 Selection of Mask Option

The following mask option is provided in mask ROM versions. The flash memory versions have no mask options.

**Table 6-6. Comparison Between Mask ROM Version and Flash Memory Version**

Pin Name	Mask ROM Version	Flash Memory Version
Mask option for pins P60 to P63	On-chip pull-up resistors can be selected in 1-bit units.	No on-chip pull-up resistor

## CHAPTER 7 CLOCK GENERATOR

### 7.1 Clock Generator Functions

The clock generator generates the clock to be supplied to the CPU and peripheral hardware. The following two types of system clock oscillators are available.

**(1) Main system clock oscillator**

This circuit oscillates at frequencies of 1 to 5.0 MHz. Oscillation can be stopped by executing the STOP instruction or setting the processor clock control register (PCC).

**(2) Subsystem clock oscillator**

The circuit oscillates at a frequency of 32.768 kHz. Oscillation cannot be stopped. If the subsystem clock oscillator is not used, not using the internal feedback resistor can be set by the processor clock control register (PCC). This enables a decrease in the power consumption in STOP mode.

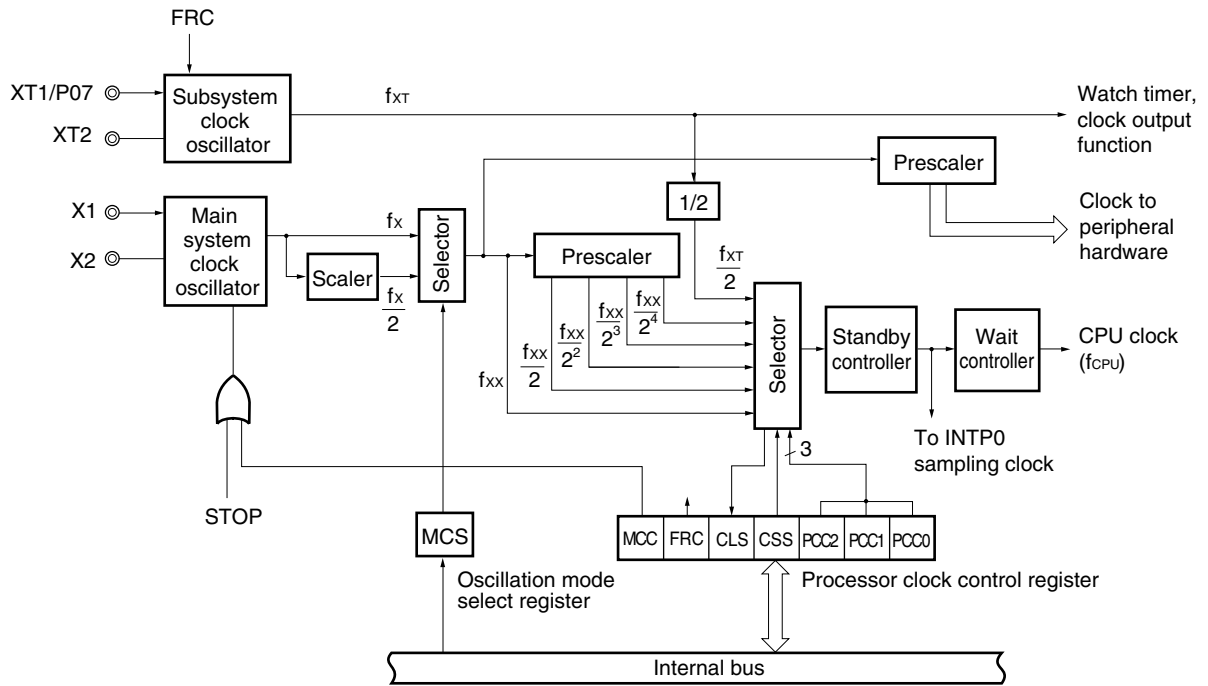
### 7.2 Clock Generator Configuration

The clock generator consists of the following hardware.

**Table 7-1. Clock Generator Configuration**

Item	Configuration
Control registers	Processor clock control register (PCC) Oscillation mode select register (OSMS)
Oscillator	Main system clock oscillator Subsystem clock oscillator

Figure 7-1. Clock Generator Block Diagram



### 7.3 Clock Generator Control Registers

The clock generator is controlled by the following two registers.

- Processor clock control register (PCC)
- Oscillation mode select register (OSMS)

#### (1) Processor clock control register (PCC)

PCC sets the CPU clock selection, division ratio, main system clock oscillator operation/stop and whether to use the subsystem clock oscillator internal feedback resistor<sup>Note</sup>.

PCC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets PCC to 04H.

- ★ **Note** The feedback resistor is necessary for adjusting the bias point of an oscillated waveform to the middle level of the supply voltage. Only when the subsystem clock is not used, the current consumption in the STOP mode can be further reduced by setting bit 6 (FRC) of PCC to 1.

**Figure 7-2. Subsystem Clock Feedback Resistor**

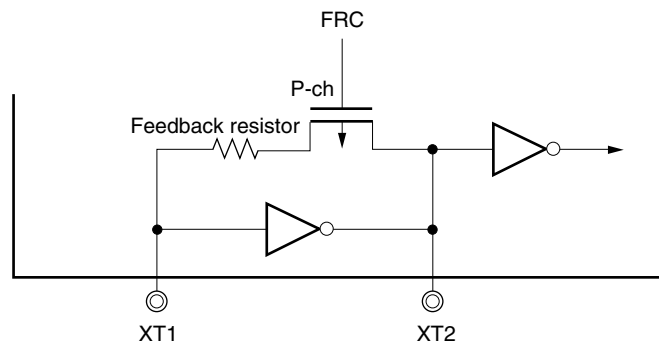


Figure 7-3. Format of Processor Clock Control Register

Symbol	<7>	<6>	<5>	<4>	3	2	1	0	Address	After reset	R/W
PCC	MCC	FRC	CLS	CSS	0	PCC2	PCC1	PCC0	FFFBH	04H	R/W <sup>Note 1</sup>

R/W	CSS	PCC2	PCC1	PCC0	CPU clock selection ( $f_{CPU}$ )		
					MCS = 1		MCS = 0
0	0	0	0	0	$f_{xx}$	$f_x$	$f_x/2$
	0	0	1	0	$f_{xx}/2$	$f_x/2$	$f_x/2^2$
	0	1	0	0	$f_{xx}/2^2$	$f_x/2^2$	$f_x/2^3$
	0	1	1	0	$f_{xx}/2^3$	$f_x/2^3$	$f_x/2^4$
	1	0	0	0	$f_{xx}/2^4$	$f_x/2^4$	$f_x/2^5$
1	0	0	0	0	$f_{xt}/2$		
	0	0	1	0			
	0	1	0	0			
	0	1	1	0			
	1	0	0	0			
Other than above				Setting prohibited			

R	CLS	CPU clock status
	0	Main system clock
	1	Subsystem clock

R/W	FRC	Subsystem clock feedback resistor selection
	0	Internal feedback resistor used
	1	Internal feedback resistor not used <sup>Note 2</sup>

R/W	MCC	Main system clock oscillation control <sup>Note 3</sup>
	0	Oscillation possible
	1	Oscillation stopped

**Notes** 1. Bit 5 is a read-only bit.

2. This bit can be set to 1 only when the subsystem clock is not used.

3. When the CPU is operating on the subsystem clock, MCC should be used to stop the main system clock oscillation. A STOP instruction should not be used.

**Caution** Be sure to clear bit 3 to 0.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3.  $f_{xt}$ : Subsystem clock oscillation frequency
  4. MCS: Bit 0 of oscillation mode select register (OSMS)

★

The fastest instruction of the  $\mu$ PD780058, 780058Y Subseries is executed in 2 CPU clocks. Therefore, the relationship between the CPU clock ( $f_{CPU}$ ) and minimum instruction execution time is as shown in Table 7-2.

**Table 7-2. Relationship Between CPU Clock and Minimum Instruction Execution Time**

CPU Clock ( $f_{CPU}$ )	Minimum Instruction Execution Time: $2/f_{CPU}$
$f_x$	0.4 $\mu s$
$f_x/2$	0.8 $\mu s$
$f_x/2^2$	1.6 $\mu s$
$f_x/2^3$	3.2 $\mu s$
$f_x/2^4$	6.4 $\mu s$
$f_x/2^5$	12.8 $\mu s$
$f_{XT}/2$	122 $\mu s$

$f_x = 5.0$  MHz,  $f_{XT} = 32.768$  kHz

$f_x$ : Main system clock oscillation frequency

$f_{XT}$ : Subsystem clock oscillation frequency

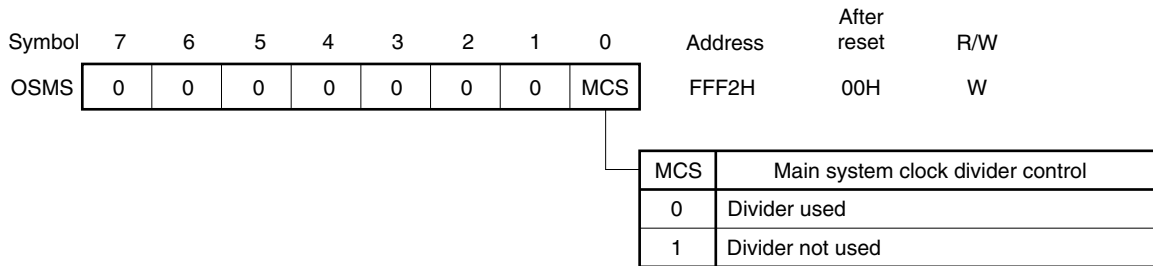
**(2) Oscillation mode select register (OSMS)**

This register specifies whether the clock output from the main system clock oscillator without passing through the divider is used as the main system clock, or the clock output via the divider is used as the main system clock.

OSMS is set with an 8-bit memory manipulation instruction.

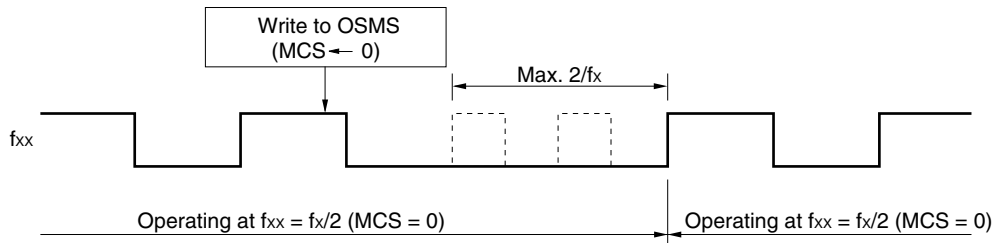
$\overline{\text{RESET}}$  input clears OSMS to 00H.

**Figure 7-4. Format of Oscillation Mode Selection Register**



**Cautions** 1. Writing to OSMS should be performed only immediately after reset signal release and before peripheral hardware operation starts. As shown in Figure 7-5 below, writing data (including the same data as previously) to OSMS causes a main system clock cycle delay of up to  $2/f_x$  during the write operation. Therefore, if this register is written during the operation, in peripheral hardware which operates on the main system clock, a temporary error occurs in the count clock cycle of timer, etc. In addition, because the oscillation mode is changed by this register, the clock for peripheral hardware as well as that for the CPU is switched.

**Figure 7-5. Main System Clock Waveform due to Writing to OSMS**



2. When writing 1 to MCS,  $V_{DD}$  must be 2.7 V or higher before the write operation.

**Remark**  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )  
 $f_x$ : Main system clock oscillation frequency

## 7.4 System Clock Oscillator

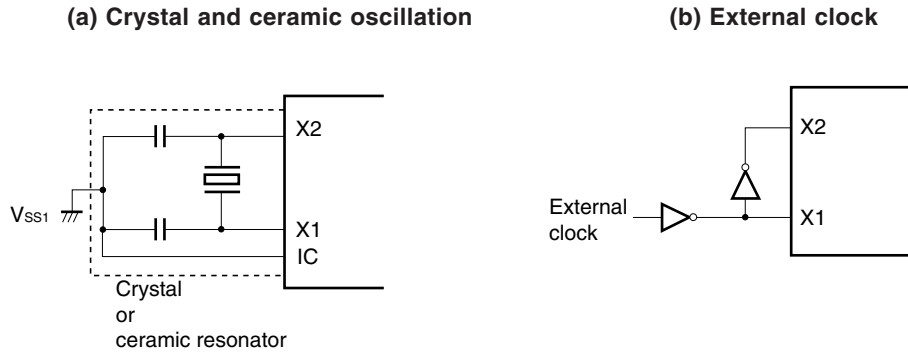
### 7.4.1 Main system clock oscillator

The main system clock oscillator oscillates with a crystal resonator or a ceramic resonator (standard: 5.0 MHz) connected to the X1 and X2 pins.

External clocks can be input to the main system clock oscillator. In this case, input a clock signal to the X1 pin and the inverse signal to the X2 pin.

Figure 7-6 shows an external circuit of the main system clock oscillator.

Figure 7-6. External Circuit of Main System Clock Oscillator



- Cautions**
1. Do not execute the STOP instruction or set MCC (bit 7 of the processor clock control register (PCC)) to 1 if an external clock is used. Otherwise, the operation of the main system clock will be stopped and the X2 pin will be pulled up to VDD1.
  2. When using a main system clock oscillator and a subsystem clock oscillator, carry out wiring in the broken line area in Figures 7-6 and 7-7 to prevent any effects from wiring capacities.
    - Minimize the wiring length.
    - Do not allow wiring to intersect with other signal conductors. Do not allow wiring to come near changing high current.
    - Set the potential of the grounding position of the oscillator capacitor to that of VSS1. Do not ground to any ground pattern where high current is present.
    - Do not fetch signals from the oscillator.



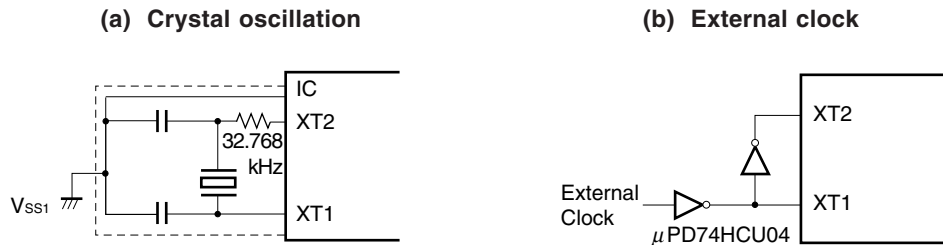
### 7.4.2 Subsystem clock oscillator

The subsystem clock oscillator oscillates with a crystal resonator (standard: 32.768 kHz) connected to the XT1 and XT2 pins.

External clocks can be input to the main system clock oscillator. In this case, input a clock signal to the XT1 pin and an antiphase clock signal to the XT2 pin.

Figure 7-7 shows an external circuit of the subsystem clock oscillator.

**Figure 7-7. External Circuit of Subsystem Clock Oscillator**



**Caution** When using a main system clock oscillator and a subsystem clock oscillator, carry out wiring in the broken line area in Figures 7-6 and 7-7 to prevent any effects from wiring capacities.

- Minimize the wiring length.
- Do not allow wiring to intersect with other signal conductors. Do not allow wiring to come near changing high current.
- Set the potential of the grounding position of the oscillator capacitor to that of VSS1. Do not ground to any ground pattern where high current is present.
- Do not fetch signals from the oscillator.

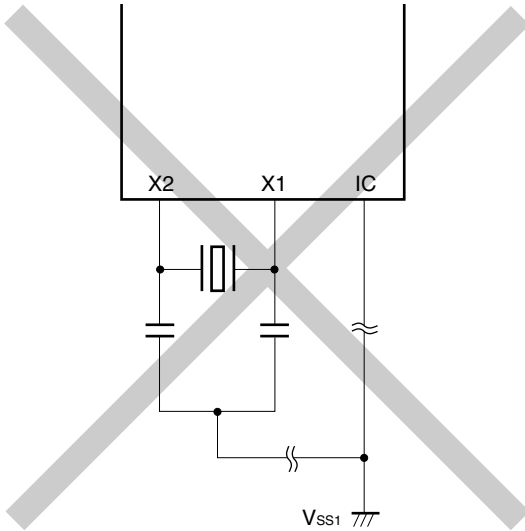
Take special note of the fact that the subsystem clock oscillator is designed as a low-amplitude circuit for reducing current consumption.

7.4.3 Example of resonator with bad connection

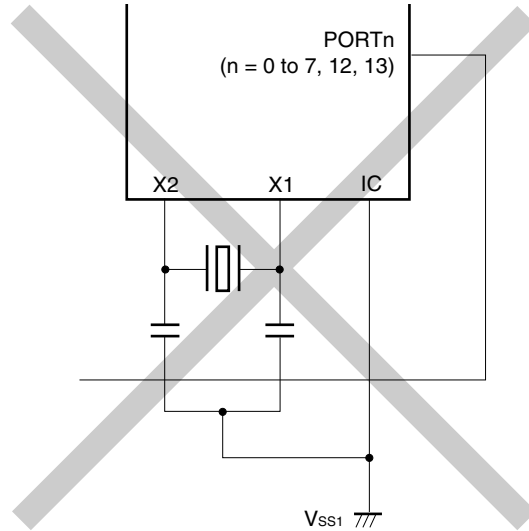
Figure 7-8 shows examples of resonators with bad connections.

Figure 7-8. Examples of Resonator with Bad Connection (1/2)

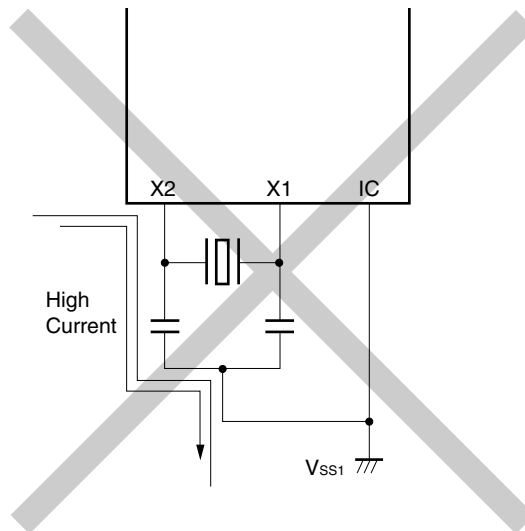
(a) Too long wiring



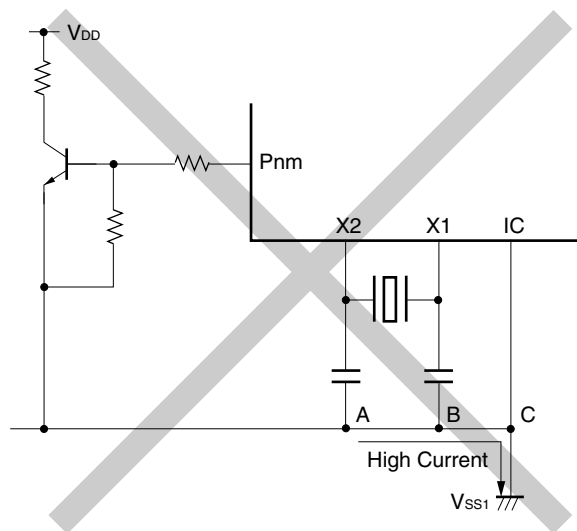
(b) Crossed signal lines



(c) Wiring near high fluctuating current



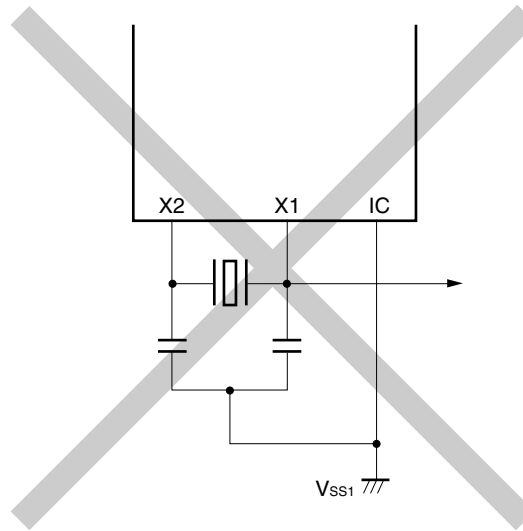
(d) Current flowing through ground line of oscillator (potential at points A, B, and C fluctuates)



**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 7-8. Examples of Resonator with Bad 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** If XT2 and XT1 are wired in parallel, the cross-talk noise of X1 may increase with XT2, resulting in malfunction. To prevent this, it is recommended to wire XT2 and X1 so that they are not in parallel, and to connect the IC pin between XT2 and X1 directly to VSS1.

#### 7.4.4 Divider

The divider divides the main system clock oscillator output ( $f_{xx}$ ) and generates various clocks.

#### 7.4.5 When not using subsystem clock

If it is not necessary to use the subsystem clock for low power consumption operations and clock operations, connect the XT1 and XT2 pins as follows.

XT1: Connect to VDD0

XT2: Leave open

In this state, however, some current may leak via the internal feedback resistor of the subsystem clock oscillator when the main system clock stops. To suppress the leakage current, disconnect the above internal feedback resistor by setting bit 6 (FRC) of the processor clock control register (PCC) to 1. In this case also, connect the XT1 and XT2 pins as described above.

## 7.5 Clock Generator Operations

The clock generator generates the following clocks and controls the CPU operating mode including the standby mode.

- Main system clock  $f_{XX}$
- Subsystem clock  $f_{XT}$
- CPU clock  $f_{CPU}$
- Clock to peripheral hardware

The following clock generator functions and operations are determined by the processor clock control register (PCC) and the oscillation mode selection register (OSMS).

- Upon generation of the  $\overline{\text{RESET}}$  signal, the lowest speed mode of the main system clock (12.8  $\mu\text{s}$  when operated at 5.0 MHz) is selected (PCC = 04H, OSMS = 00H). Main system clock oscillation stops while a low level is applied to the  $\overline{\text{RESET}}$  pin.
- With the main system clock selected, one of the six types of minimum instruction execution times (0.4  $\mu\text{s}$ , 0.8  $\mu\text{s}$ , 1.6  $\mu\text{s}$ , 3.2  $\mu\text{s}$ , 6.4  $\mu\text{s}$ , 12.8  $\mu\text{s}$  @ 5.0 MHz) can be selected by setting the PCC and OSMS registers.
- With the main system clock selected, two standby modes, the STOP and HALT modes, are available. In a system where the subsystem clock is not used, the current consumption in the STOP mode can be further reduced by specifying with not to use the feedback resistor using bit 6 (FRC) of the PCC register.
- The PCC register can be used to select the subsystem clock and to operate the system on a low current consumption (122  $\mu\text{s}$  when operated at 32.768 kHz).
- With the subsystem clock selected, main system clock oscillation can be stopped by the PCC register. The HALT mode can be used, but not the STOP mode. (Subsystem clock oscillation cannot be stopped.)
- The main system clock is divided and supplied to the peripheral hardware. The subsystem clock is supplied to the 16-bit timer/event counter, watch timer, and clock output functions only. Thus, the 16-bit timer/event counter (when selecting watch timer output as the count clock when operating on the subsystem clock), the watch function, and the clock output function can also be continued in the standby state. However, since all other peripheral hardware operate on the main system clock, the peripheral hardware also stops if the main system clock is stopped (except external input clock operation).

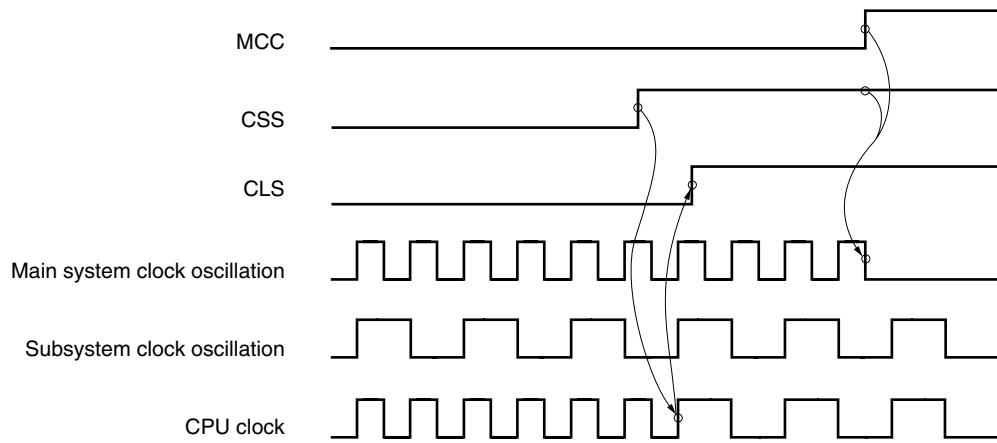
**7.5.1 Main system clock operations**

When operating on the main system clock (with bit 5 (CLS) of the processor clock control register (PCC) cleared to 0), the following operations are carried out by PCC settings.

- (a) Because the operation guaranteed instruction execution speed depends on the power supply voltage, the minimum instruction execution time can be changed by bits 0 to 2 (PCC0 to PCC2) of the PCC register.
- (b) If bit 7 (MCC) of the PCC register is set to 1 when operating on the main system clock, the main system clock oscillation does not stop. When bit 4 (CSS) of PCC is set to 1 and the operation is subsequently switched to the subsystem clock (CLS = 1), the main system clock oscillation stops (see **Figure 7-9**).

**Figure 7-9. Main System Clock Stop Function (1/2)**

**(a) Operation when MCC is set after setting CSS in case of main system clock operation**



**(b) Operation when MCC is set in case of main system clock operation**

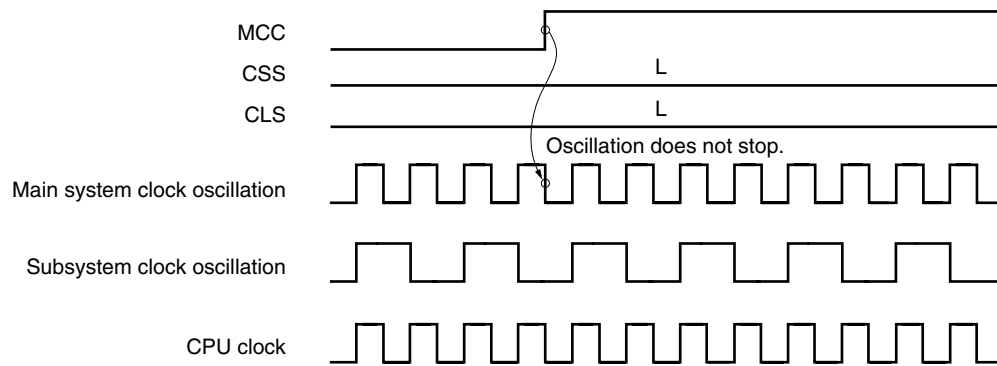
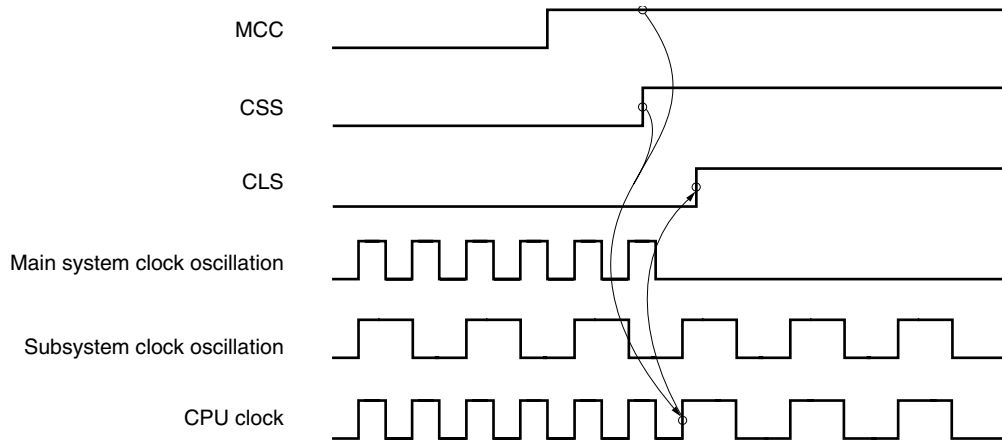


Figure 7-9. Main System Clock Stop Function (2/2)

## (c) Operation when CSS is set after setting MCC in case of main system clock operation



## 7.5.2 Subsystem clock operations

When operating on the subsystem clock (with bit 5 (CLS) of the processor clock control register (PCC) set to 1), the following operations are carried out.

- The minimum instruction execution time remains constant ( $122 \mu\text{s}$  when operating at 32.768 kHz) irrespective of bits 0 to 2 (PCC0 to PCC2) of the PCC register.
- The watchdog timer stops counting.

**Caution** Do not execute the STOP instruction while the subsystem clock is in operation.

## 7.6 Changing System Clock and CPU Clock Settings

### 7.6.1 Time required for switchover between system clock and CPU clock

The system clock and CPU clock can be switched over by bits 0 to 2 (PCC0 to PCC2) and bit 4 (CSS) of the processor clock control register (PCC).

The actual switchover operation is not performed directly after writing to the PCC; operation continues on the pre-switchover clock for several instructions (see **Table 7-3**).

Determination as to whether the system is operating on the main system clock or the subsystem clock is performed using bit 5 (CLS) of the PCC register.

**Table 7-3. Maximum Time Required for CPU Clock Switchover**

Set Values Before Switchover				Set Values After Switchover																				MCS = 1				MCS = 0			
																								CSS	PCC2	PCC1	PCC0	CSS	PCC2	PCC1	PCC0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	1	×	×	×	1	×	×	×
0	0	0	0	/				16 instructions				16 instructions				16 instructions				16 instructions				f <sub>x</sub> /2f <sub>xT</sub> instruction (77 instructions)				f <sub>x</sub> /4f <sub>xT</sub> instruction (39 instructions)			
	0	0	1					8 instructions				8 instructions				8 instructions				8 instructions				f <sub>x</sub> /4f <sub>xT</sub> instruction (39 instructions)				f <sub>x</sub> /8f <sub>xT</sub> instruction (20 instructions)			
	0	1	0					4 instructions				4 instructions				4 instructions				4 instructions				f <sub>x</sub> /8f <sub>xT</sub> instruction (20 instructions)				f <sub>x</sub> /16f <sub>xT</sub> instruction (10 instructions)			
	0	1	1					2 instructions				2 instructions				2 instructions				2 instructions				f <sub>x</sub> /16f <sub>xT</sub> instruction (10 instructions)				f <sub>x</sub> /32f <sub>xT</sub> instruction (5 instructions)			
	1	0	0					1 instruction				1 instruction				1 instruction				1 instruction				f <sub>x</sub> /32f <sub>xT</sub> instruction (5 instructions)				f <sub>x</sub> /64f <sub>xT</sub> instruction (3 instructions)			
1	×	×	×	1 instruction				1 instruction				1 instruction				1 instruction				1 instruction				/							

- Remarks**
- One instruction is executed in the minimum instruction execution time with the pre-switchover CPU clock.
  - MCS: Bit 0 of the oscillation mode selection register (OSMS)
  - Values in parentheses apply to operation with f<sub>x</sub> = 5.0 MHz or f<sub>xT</sub> = 32.768 kHz.

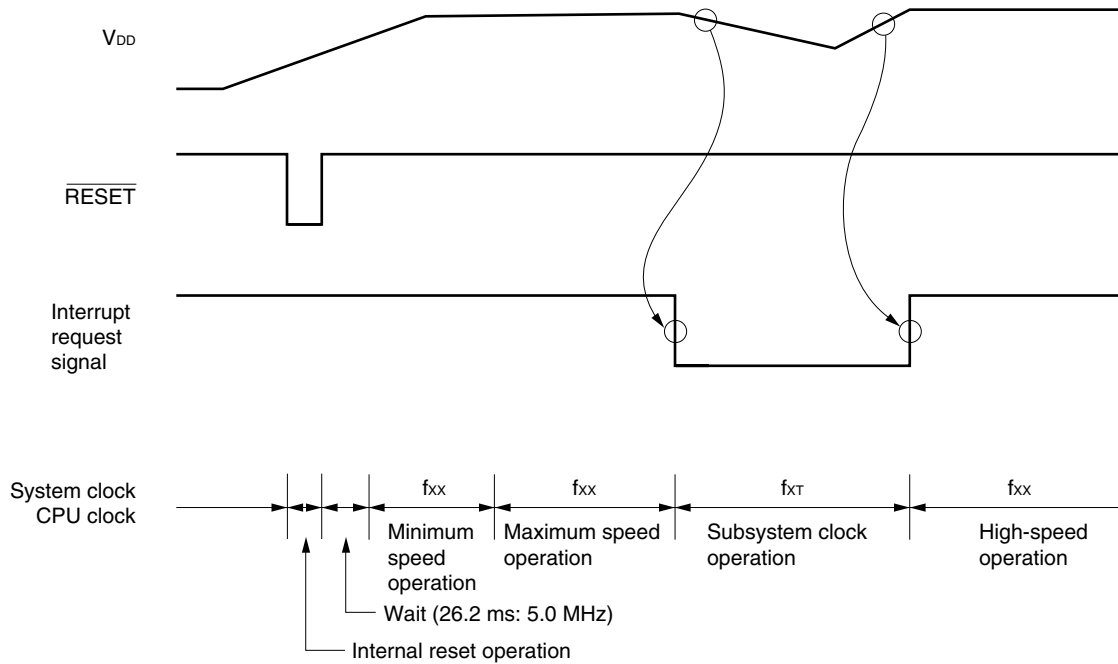
**Caution** Selection of the CPU clock cycle division ratio (PCC0 to PCC2) and switchover from the main system clock to the subsystem clock (changing CSS from 0 to 1) should not be performed simultaneously. Simultaneous setting is possible, however, for selection of the CPU clock cycle division ratio (PCC0 to PCC2) and switchover from the subsystem clock to the main system clock (changing CSS from 1 to 0).



### 7.6.2 System clock and CPU clock switching procedure

This section describes the procedure for switching between the system clock and the CPU clock.

Figure 7-10. Switching Between System Clock and CPU Clock



- (1) The CPU is reset by setting the  $\overline{\text{RESET}}$  signal to low level after power-on. After that, when reset is released by setting the  $\overline{\text{RESET}}$  signal to high level, the main system clock starts oscillation. At this time, the oscillation stabilization time ( $2^{17}/f_x$ ) is secured automatically. After that, the CPU starts executing the instruction at the minimum speed of the main system clock ( $12.8 \mu\text{s}$  when operated at 5.0 MHz).
- (2) After the lapse of a sufficient time for the  $V_{DD}$  voltage to increase to enable operation at maximum speeds, the processor clock control register (PCC) and oscillation mode selection register (OSMS) are rewritten and the maximum-speed operation is carried out.
- (3) Upon detection of a decrease of the  $V_{DD}$  voltage due to an interrupt request signal, the main system clock is switched to the subsystem clock (which must be in an oscillation stable state).
- (4) Upon detection of  $V_{DD}$  voltage reset due to an interrupt request signal, bit 7 (MCC) of PCC is cleared to 0 and oscillation of the main system clock is started. After the lapse of time required for stabilization of oscillation, the PCC and OSMS registers are rewritten and the maximum-speed operation is resumed.

**Caution** When the main system clock is stopped and the device is operating on the subsystem clock, wait until the oscillation stabilization time has been secured by the program before switching back to the main system clock.

## CHAPTER 8 16-BIT TIMER/EVENT COUNTER

### 8.1 16-Bit Timer/Event Counter Functions

The 16-bit timer/event counter (TM0) has the following functions.

- Interval timer
- PWM output
- Pulse width measurement
- External event counter
- Square-wave output
- One-shot pulse output

PWM output and pulse width measurement can be used at the same time.

#### (1) Interval timer

TM0 generates interrupt requests at the preset time interval.

**Table 8-1. 16-Bit Timer/Event Counter Interval Times**

Minimum Interval Time		Maximum Interval Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × TI00 input cycle		2 <sup>16</sup> × TI00 input cycle		TI00 input edge cycle	
—	2 × 1/fx (400 ns)	—	2 <sup>16</sup> × 1/fx (13.1 ms)	—	1/fx (200 ns)
2 × 1/fx (400 ns)	2 <sup>2</sup> × 1/fx (800 ns)	2 <sup>16</sup> × 1/fx (13.1 ms)	2 <sup>17</sup> × 1/fx (26.2 ms)	1/fx (200 ns)	2 × 1/fx (400 ns)
2 <sup>2</sup> × 1/fx (800 ns)	2 <sup>3</sup> × 1/fx (1.6 μs)	2 <sup>17</sup> × 1/fx (26.2 ms)	2 <sup>18</sup> × 1/fx (52.4 ms)	2 × 1/fx (400 ns)	2 <sup>2</sup> × 1/fx (800 ns)
2 <sup>3</sup> × 1/fx (1.6 μs)	2 <sup>4</sup> × 1/fx (3.2 μs)	2 <sup>18</sup> × 1/fx (52.4 ms)	2 <sup>19</sup> × 1/fx (104.9 ms)	2 <sup>2</sup> × 1/fx (800 ns)	2 <sup>3</sup> × 1/fx (1.6 μs)
2 × watch timer output cycle		2 <sup>16</sup> × watch timer output cycle		Watch timer output edge cycle	

- Remarks**
1. fx: Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with fx = 5.0 MHz

#### (2) PWM output

TM0 can generate 14-bit resolution PWM output.

#### (3) Pulse width measurement

TM0 can measure the pulse width of an externally input signal.

#### (4) External event counter

TM0 can measure the number of pulses of an externally input signal.

**(5) Square-wave output**

TM0 can output a square wave with any selected frequency.

**Table 8-2. 16-Bit Timer/Event Counter Square-Wave Output Ranges**

Minimum Pulse Time		Maximum Pulse Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × TI00 input cycle		2 <sup>16</sup> × TI00 input cycle		TI00 input edge cycle	
—	2 × 1/f <sub>x</sub> (400 ns)	—	2 <sup>16</sup> × 1/f <sub>x</sub> (13.1 ms)	—	1/f <sub>x</sub> (200 ns)
2 × 1/f <sub>x</sub> (400 ns)	2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)	2 <sup>16</sup> × 1/f <sub>x</sub> (13.1 ms)	2 <sup>17</sup> × 1/f <sub>x</sub> (26.2 ms)	1/f <sub>x</sub> (200 ns)	2 × 1/f <sub>x</sub> (400 ns)
2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)	2 <sup>3</sup> × 1/f <sub>x</sub> (1.6 μs)	2 <sup>17</sup> × 1/f <sub>x</sub> (26.2 ms)	2 <sup>18</sup> × 1/f <sub>x</sub> (52.4 ms)	2 × 1/f <sub>x</sub> (400 ns)	2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)
2 <sup>3</sup> × 1/f <sub>x</sub> (1.6 μs)	2 <sup>4</sup> × 1/f <sub>x</sub> (3.2 μs)	2 <sup>18</sup> × 1/f <sub>x</sub> (52.4 ms)	2 <sup>19</sup> × 1/f <sub>x</sub> (104.9 ms)	2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)	2 <sup>3</sup> × 1/f <sub>x</sub> (1.6 μs)
2 × watch timer output cycle		2 <sup>16</sup> × watch timer output cycle		Watch timer output edge cycle	

- Remarks**
1. f<sub>x</sub>: Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with f<sub>x</sub> = 5.0 MHz

**(6) One-shot pulse output**

TM0 is able to output a one-shot pulse with any output pulse width.

## 8.2 16-Bit Timer/Event Counter Configuration

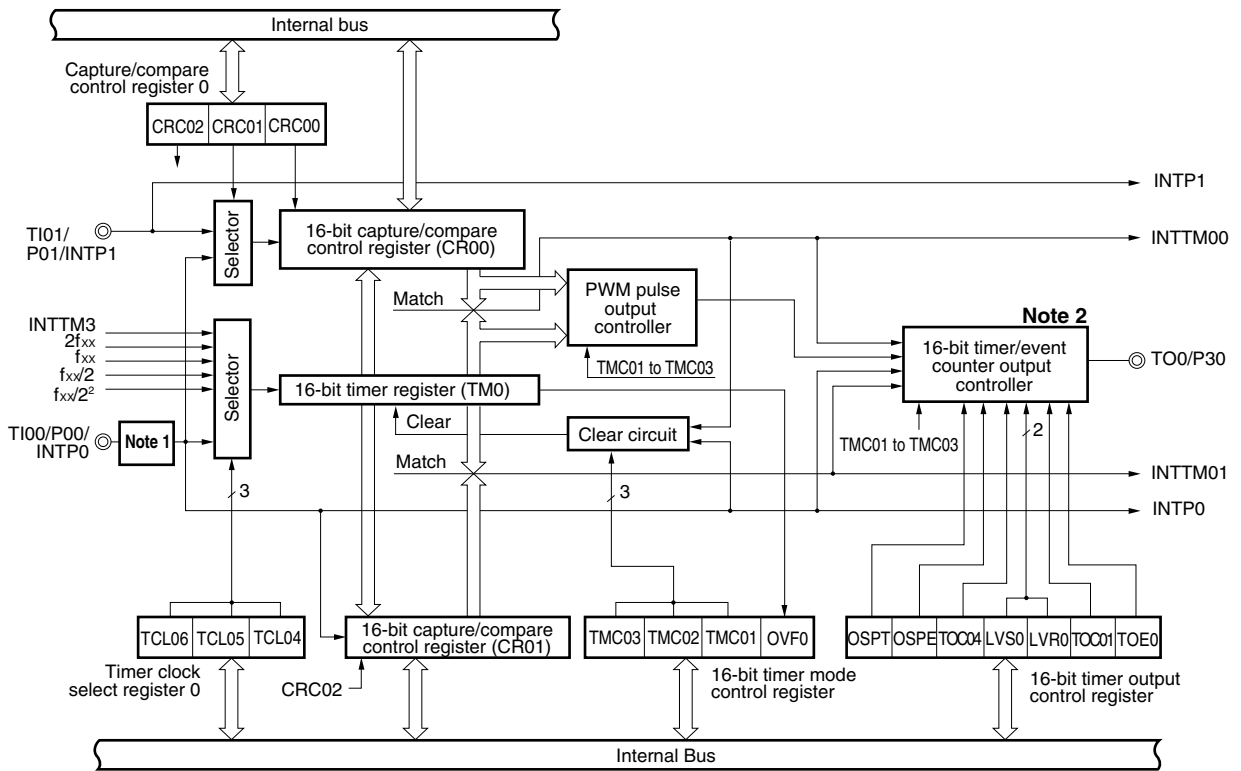
The 16-bit timer/event counter consists of the following hardware.

**Table 8-3. 16-Bit Timer/Event Counter Configuration**

Item	Configuration
Timer register	16 bits × 1 (TM0)
Register	Capture/compare register: 16 bits × 2 (CR00, CR01)
Timer outputs	1 (TO0)
Control registers	Timer clock select register 0 (TCL0) 16-bit timer mode control register (TMC0) Capture/compare control register 0 (CRC0) 16-bit timer output control register (TOC0) Port mode register 3 (PM3) External interrupt mode register 0 (INTM0) Sampling clock select register (SCS) <sup>Note</sup>

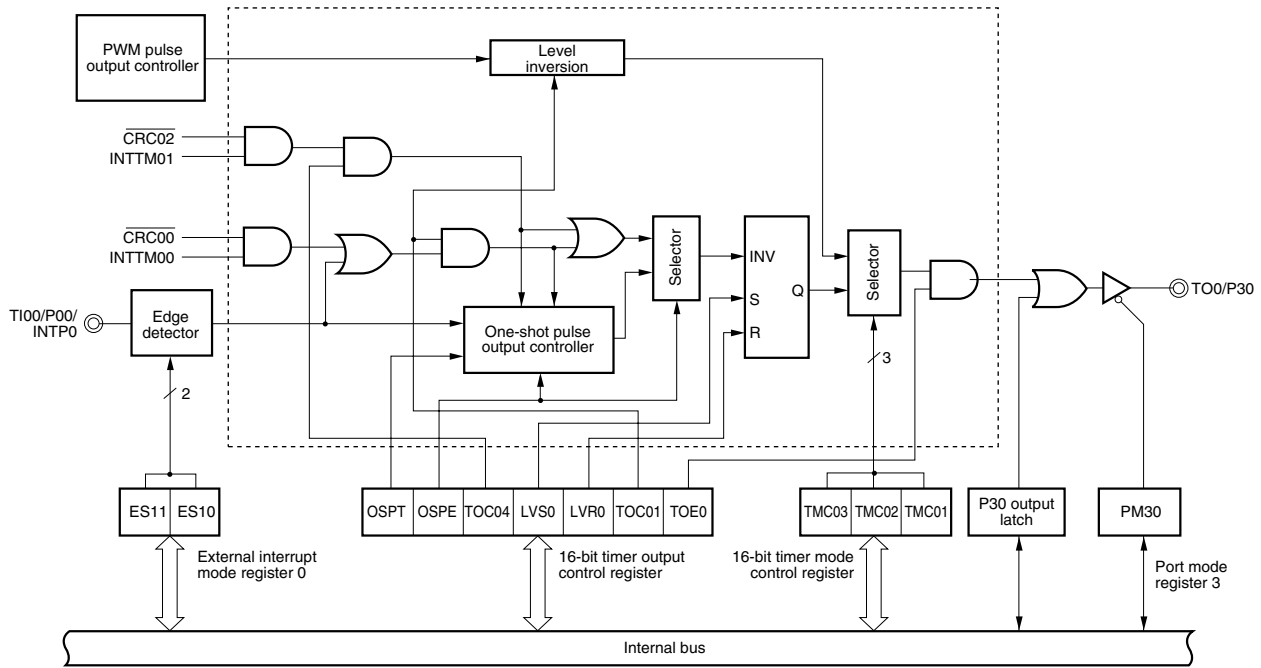
**Note** See Figure 21-1 Basic Configuration of Interrupt Function.

Figure 8-1. Block Diagram of 16-Bit Timer/Event Counter



- Notes**
1. Edge detector
  2. The configuration of the 16-bit timer/event counter output controller is shown in Figure 8-2.

Figure 8-2. Block Diagram of 16-Bit Timer/Event Counter Output Controller



**Remark** The circuitry enclosed by the broken line is the output controller.

★ (1) **Capture/compare register 00 (CR00)**

CR00 is a 16-bit register which has the functions of both a capture register and a compare register. Whether it is used as a capture register or as a compare register is set by bit 0 (CRC00) of capture/compare control register 0.

(a) **When CR00 is used as a compare register**

The value set in CR00 is constantly compared with the 16-bit timer register (TM0) count value, and an interrupt request (INTTM00) is generated if they match. It can also be used as the register that holds the interval time when TM0 is set to interval timer operation, and as the register that sets the pulse width when TM0 is set to PWM output operation.

(b) **When CR00 is used as a capture register**

It is possible to select the valid edge of the INTP0/TI00 pin or the INTP1/TI01 pin as the capture trigger. The INTP0/TI00 or INTP1/TI01 valid edge is set by external interrupt mode register 0 (INTM0).

If CR00 is specified as a capture register and the capture trigger is specified to be the valid edge of the INTP0/TI00 pin, the situation is as shown in Table 8-4. On the other hand, when the capture trigger is specified to be the valid edge of the INTP1/TI01 pin, the situation is as shown in Table 8-5.

**Table 8-4. INTP0/TI00 Pin Valid Edge and CR00 Capture Trigger Valid Edge**

ES11	ES10	INTP0/TI00 Pin Valid Edge	CR00 Capture Trigger Valid Edge
0	0	Falling edge	Rising edge
0	1	Rising edge	Falling edge
1	0	Setting prohibited	
1	1	Both rising and falling edges	No capture operation

★ **Table 8-5. INTP1/TI01 Pin Valid Edge and CR00 Capture Trigger Valid Edge**

ES21	ES20	INTP1/TI01 Pin Valid Edge	CR00 Capture Trigger Valid Edge
0	0	Falling edge	Falling edge
0	1	Rising edge	Rising edge
1	0	Setting prohibited	
1	1	Both rising and falling edges	Both rising and falling edges

CR00 is set with a 16-bit memory manipulation instruction.  
 RESET input makes CR00 undefined.

- Cautions**
1. Set the data of PWM (14 bits) to the higher 14 bits of CR00. At this time, clear the lower 2 bits to 00.
  2. Set CR00 to a value other than 0000H in the clear & start mode entered on a match between TM0 and CR00. However, in the free-running mode and in the clear mode using the valid edge of TI00, if CR00 is set to 0000H, an interrupt request (INTTM00) is generated following overflow (FFFFH).
  3. If the new value of CR00 is less than the value of the 16-bit timer register (TM0), TM0 continues counting, overflows, and then starts counting again from 0. If the new value of CR00 is less than the old value, the timer must be restarted after changing the value of CR00.

**(2) Capture/compare register 01 (CR01)**

CR01 is a 16-bit register which has the functions of both a capture register and a compare register. Whether it is used as a capture register or a compare register is set by bit 2 (CRC02) of capture/compare control register 0.

**(a) When CR01 is used as a compare register**

The value set in CR01 is constantly compared with the 16-bit timer register (TM0) count value, and an interrupt request (INTTM01) is generated if they match.

**(b) When CR01 is used as a capture register**

It is possible to select the valid edge of the INTP0/TI00 pin as the capture trigger. The INTP0/TI00 valid edge is set by external interrupt mode register 0 (INTM0).

★

**Table 8-6. INTP0/TI00 Pin Valid Edge and CR01 Capture Trigger Valid Edge**

ES11	ES10	INTP0/TI00 Pin Valid Edge	CR01 Capture Trigger Valid Edge
0	0	Falling edge	Falling edge
0	1	Rising edge	Rising edge
1	0	Setting prohibited	
1	1	Both rising and falling edges	Both rising and falling edges

CR01 is set with a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input makes CR01 undefined.

**Caution** Set CR01 to a value other than 0000H in the clear & start mode entered on a match between TM0 and CR00. However, in the free-running mode and in the clear mode using the valid edge of TI00, if CR01 is set to 0000H, an interrupt request (INTTM01) is generated following overflow (FFFFH).

**(3) 16-bit timer register (TM0)**

TM0 is a 16-bit register which counts the count pulses.

TM0 is read with a 16-bit memory manipulation instruction. When TM0 is read, the capture/compare register (CR01) should first be set as a capture register.

$\overline{\text{RESET}}$  input clears TM0 to 0000H.

**Caution** As reading of the value of TM0 is performed via CR01, the previously set value of CR01 is lost.



### 8.3 16-Bit Timer/Event Counter Control Registers

The following seven registers are used to control the 16-bit timer/event counter.

- Timer clock select register 0 (TCL0)
- 16-bit timer mode control register (TMC0)
- Capture/compare control register 0 (CRC0)
- 16-bit timer output control register (TOC0)
- Port mode register 3 (PM3)
- External interrupt mode register 0 (INTM0)
- Sampling clock select register (SCS)

#### (1) Timer clock select register 0 (TCL0)

This register is used to set the count clock of the 16-bit timer register.

TCL0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TCL0 to 00H.

**Remark** TCL0 has the function of setting the PCL output clock in addition to that of setting the count clock of the 16-bit timer register.

Figure 8-3. Format of Timer Clock Select Register 0

Symbol	<7>	6	5	4	3	2	1	0	Address	After reset	R/W
TCL0	CLOE	TCL06	TCL05	TCL04	TCL03	TCL02	TCL01	TCL00	FF40H	00H	R/W

TCL03	TCL02	TCL01	TCL00	PCL output clock selection			
				MCS = 1		MCS = 0	
0	0	0	0	f <sub>XT</sub> (32.768 kHz)			
0	1	0	1	f <sub>xx</sub>	f <sub>x</sub> (5.0 MHz)	f <sub>x/2</sub>	(2.5 MHz)
0	1	1	0	f <sub>xx/2</sub>	f <sub>x/2</sub> (2.5 MHz)	f <sub>x/2<sup>2</sup></sub>	(1.25 MHz)
0	1	1	1	f <sub>xx/2<sup>2</sup></sub>	f <sub>x/2<sup>2</sup></sub> (1.25 MHz)	f <sub>x/2<sup>3</sup></sub>	(625 kHz)
1	0	0	0	f <sub>xx/2<sup>3</sup></sub>	f <sub>x/2<sup>3</sup></sub> (625 kHz)	f <sub>x/2<sup>4</sup></sub>	(313 kHz)
1	0	0	1	f <sub>xx/2<sup>4</sup></sub>	f <sub>x/2<sup>4</sup></sub> (313 kHz)	f <sub>x/2<sup>5</sup></sub>	(156 kHz)
1	0	1	0	f <sub>xx/2<sup>5</sup></sub>	f <sub>x/2<sup>5</sup></sub> (156 kHz)	f <sub>x/2<sup>6</sup></sub>	(78.1 kHz)
1	0	1	1	f <sub>xx/2<sup>6</sup></sub>	f <sub>x/2<sup>6</sup></sub> (78.1 kHz)	f <sub>x/2<sup>7</sup></sub>	(39.1 kHz)
1	1	0	0	f <sub>xx/2<sup>7</sup></sub>	f <sub>x/2<sup>7</sup></sub> (39.1 kHz)	f <sub>x/2<sup>8</sup></sub>	(19.5 kHz)
Other than above				Setting prohibited			

TCL06	TCL05	TCL04	16-bit timer register count clock selection				
			MCS = 1		MCS = 0		
0	0	0	TI00 (valid edge specifiable)				
0	0	1	2f <sub>xx</sub>	Setting prohibited		f <sub>x</sub>	(5.0 MHz)
0	1	0	f <sub>xx</sub>	f <sub>x</sub> (5.0 MHz)	f <sub>x/2</sub>	(2.5 MHz)	
0	1	1	f <sub>xx/2</sub>	f <sub>x/2</sub> (2.5 MHz)	f <sub>x/2<sup>2</sup></sub>	(1.25 MHz)	
1	0	0	f <sub>xx/2<sup>2</sup></sub>	f <sub>x/2<sup>2</sup></sub> (1.25 MHz)	f <sub>x/2<sup>3</sup></sub>	(625 kHz)	
1	1	1	Watch timer output (INTTM 3)				
Other than above			Setting prohibited				

CLOE	PCL output control	
0	Output disabled	
1	Output enabled	

- Cautions**
1. The TI00/INTP0 pin valid edge is set by external interrupt mode register 0 (INTM0), and the sampling clock frequency is selected by the sampling clock selection register (SCS).
  2. When enabling PCL output, set TCL00 to TCL03, then set CLOE to 1 with a 1-bit memory manipulation instruction.
  3. To read the count value when TI00 has been specified as the TM0 count clock, the value should be read from TM0, not from 16-bit capture/compare register 01 (CR01).
  4. When rewriting TCL0 to other data, stop the timer operation beforehand.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3.  $f_{xT}$ : Subsystem clock oscillation frequency
  4. TI00: 16-bit timer/event counter input pin
  5. TM0: 16-bit timer register
  6. MCS: Bit 0 of oscillation mode select register (OSMS)
  7. Values in parentheses apply to operation with  $f_x = 5.0$  MHz or  $f_{xT} = 32.768$  kHz.

**(2) 16-bit timer mode control register (TMC0)**

This register sets the 16-bit timer operating mode, the 16-bit timer register clear mode and output timing, and detects an overflow.

TMC0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TMC0 to 00H.

**Caution** The 16-bit timer register starts operation at the moment TMC01 to TMC03 are set to values other than 0, 0, 0 (operation stop mode). Set TMC01 to TMC03 to 0, 0, 0 to stop the operation.

Figure 8-4. Format of 16-Bit Timer Mode Control Register

Symbol	7	6	5	4	3	2	1	<0>	Address	After reset	R/W
TMC0	0	0	0	0	TMC03	TMC02	TMC01	OVF0	FF48H	00H	R/W

OVF0	16-bit timer register overflow detection
0	Overflow not detected
1	Overflow detected

TMC03	TMC02	TMC01	Operating mode or clear mode selection	TO0 output timing selection	Interrupt request generation
0	0	0	Operation stopped (TM0 cleared to 0)	No change	Not generated
0	0	1	PWM mode (free running)	PWM pulse output	Generated on match between TM0 and CR00, and match between TM0 and CR01
0	1	0	Free-running mode	Match between TM0 and CR00 or match between TM0 and CR01	
0	1	1		Match between TM0 and CR00, match between TM0 and CR01 or T100 valid edge	
1	0	0	Clear & start on T100 valid edge	Match between TM0 and CR00 or match between TM0 and CR01	
1	0	1		Match between TM0 and CR00, match between TM0 and CR01 or T100 valid edge	
1	1	0	Clear & start on match between TM0 and CR00	Match between TM0 and CR00 or match between TM0 and CR01	
1	1	1		Match between TM0 and CR00, match between TM0 and CR01 or T100 valid edge	

- Cautions**
1. Switch the clear mode and the TO0 output timing after stopping the timer operation (by clearing TMC01 to TMC03 to 0, 0, 0).
  2. Set the valid edge of the T100/INTP0 pin using external interrupt mode register 0 (INTM0) and select the sampling clock frequency using the sampling clock select register (SCS).
  3. When using the PWM mode, set the PWM mode and then set data to CR00.
  4. If clear & start mode entered on a match between TM0 and CR00 is selected, when the set value of CR00 is FFFFH and the TM0 value changes from FFFFH to 0000H, the OVF0 flag is set to 1.

**Remark**

- TO0: 16-bit timer/event counter output pin
- T100: 16-bit timer/event counter input pin
- TM0: 16-bit timer register
- CR00: Compare register 00
- CR01: Compare register 01

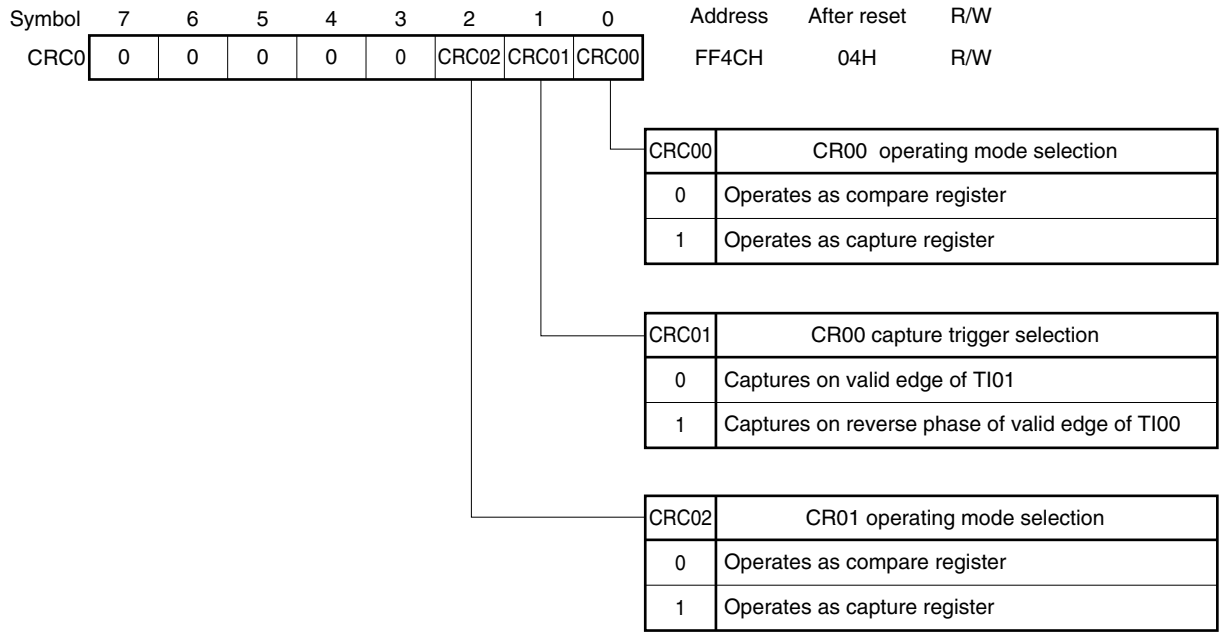
**(3) Capture/compare control register 0 (CRC0)**

This register controls the operation of capture/compare registers 00 and 01 (CR00 and CR01).

CRC0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets CRC0 to 04H.

**Figure 8-5. Format of Capture/Compare Control Register 0**



- Cautions**
1. Timer operation must be stopped before setting CRC0.
  2. When clear & start mode entered on a match between TM0 and CR00 is selected by the 16-bit timer mode control register (TMC0), CR00 should not be specified as a capture register.

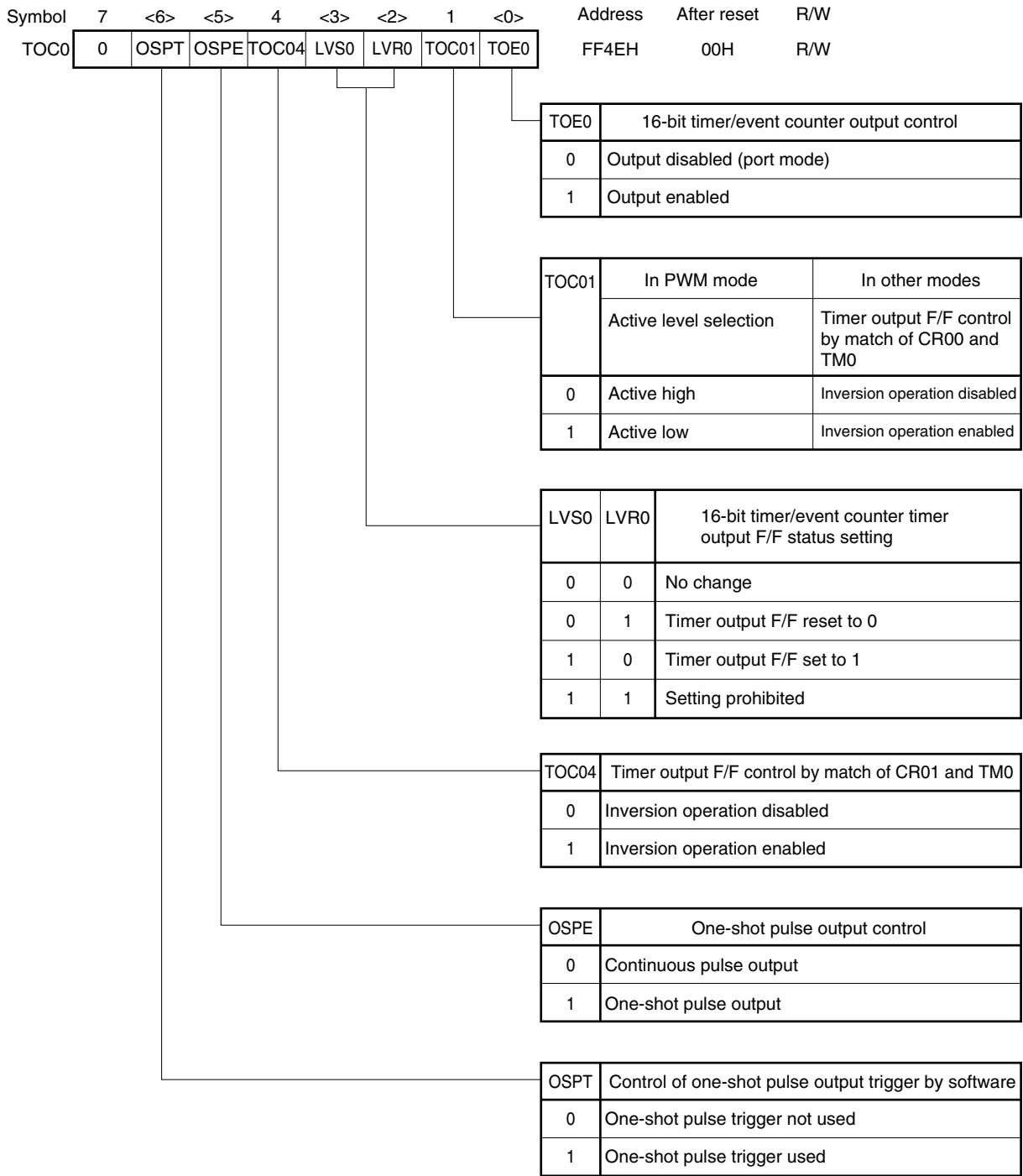
**(4) 16-bit timer output control register (TOC0)**

This register controls the operation of the 16-bit timer/event counter output controller. It sets R-S type flip-flop (LV0) setting/resetting, the active level in PWM mode, inversion enabling/disabling in modes other than PWM mode, 16-bit timer/event counter timer output enabling/disabling, one-shot pulse output operation enabling/disabling, and the output trigger for a one-shot pulse by software.

TOC0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TOC0 to 00H.

Figure 8-6. Format of 16-Bit Timer Output Control Register



- Cautions**
1. Timer operation must be stopped before setting TOC0 (except OSPT).
  2. If LVS0 and LVR0 are read after data is set, they will be 0.
  3. OSPT is cleared automatically after data setting, and will therefore be 0 if read.
  4. OSPT can be set only when OSPE = 1.

**(5) Port mode register 3 (PM3)**

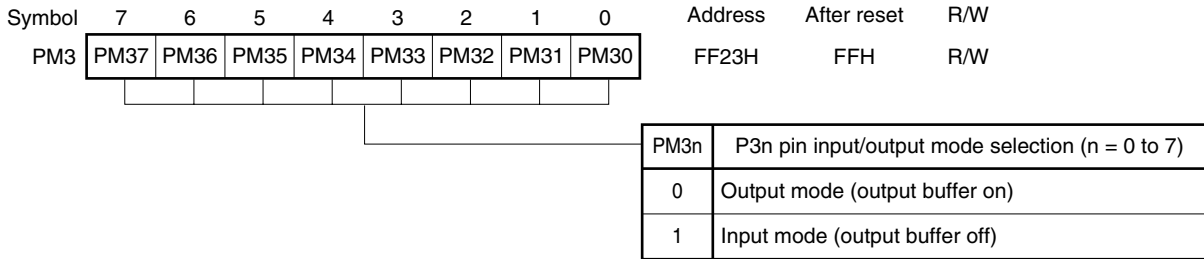
This register sets port 3 input/output in 1-bit units.

When using the P30/TO0 pin for timer output, set PM30 and the output latch of P30 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets PM3 to FFH.

**Figure 8-7. Format of Port Mode Register 3**





★

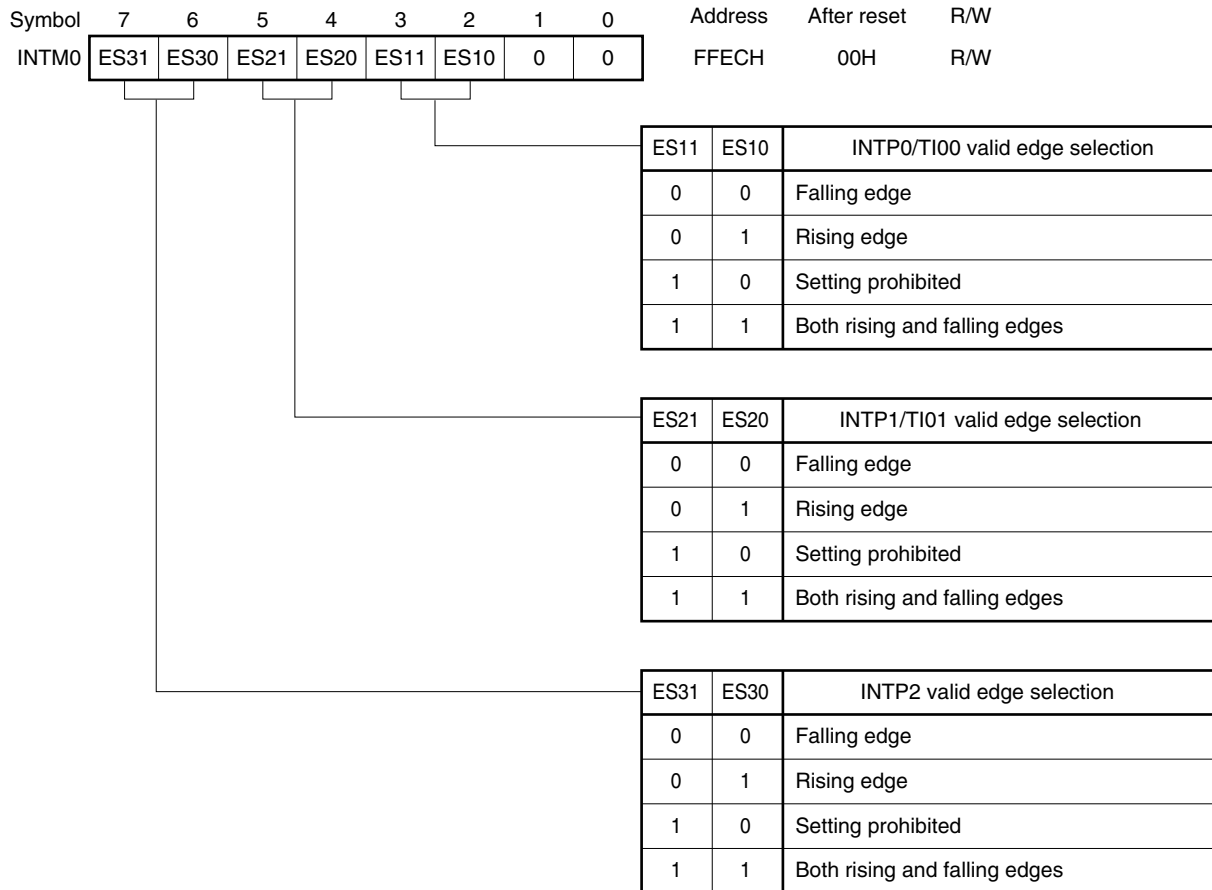
**(6) External interrupt mode register 0 (INTM0)**

This register is used to set the valid edges of INTP0 to INTP2, TI00, and TI01.

INTM0 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears INTM0 to 00H.

**Figure 8-8. Format of External Interrupt Mode Register 0**



★

**Caution** When using the INTP0/TI00/P00 and INTP1/TI01/P01 pins as timer input pins (TI00 and TI01), stop the operation of 16-bit timer 0 by clearing bits 1 to 3 (TMC01 to TMC03) of the 16-bit timer mode control register (TMC0) to 0, 0, 0, before setting the valid edge of TI00 and TI01. When using the INTP0/TI00/P00 and INTP1/TI01/P01 pins as external interrupt input pins (INTP0 and INTP1), the valid edge of INTP0 and INTP1 may be set while 16-bit timer 0 is operating.

**(7) Sampling clock select registers (SCS)**

This register sets the clock used as the clock for sampling the valid edges input to INTP0. When remote controlled reception is carried out using INTP0, digital noise is eliminated by the sampling clock.

SCS is set with an 8-bit memory manipulation instruction.

RESET input clears SCS to 00H.

**Figure 8-9. Format of Sampling Clock Select Register**



**Caution**  $f_{xx}/2^N$  is the clock supplied to the CPU, and  $f_{xx}/2^5$ ,  $f_{xx}/2^6$ , and  $f_{xx}/2^7$  are clocks supplied to peripheral hardware. The  $f_{xx}/2^N$  clock is stopped in HALT mode.

- Remarks**
1. N: Value set to bits 0 to 2 (PCC0 to PCC2) of the processor clock control register (PCC) (N = 0 to 4)
  2.  $f_{xx}$ : Main system clock frequency (fx or fx/2)
  3.  $f_x$ : Main system clock oscillation frequency
  4. MCS: Bit 0 of oscillation mode select register (OSMS)
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

## 8.4 16-Bit Timer/Event Counter Operations

### 8.4.1 Interval timer operations

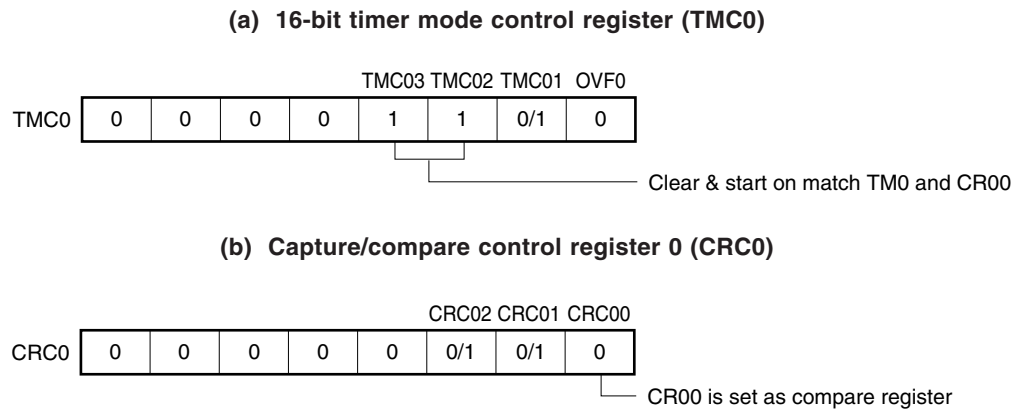
Setting the 16-bit timer mode control register (TMC0) and capture/compare control register 0 (CRC0) as shown in Figure 8-10 allows operation as an interval timer. Interrupt requests are generated repeatedly using the count value set to 16-bit capture/compare register 00 (CR00) beforehand as the interval.

When the count value of the 16-bit timer register (TM0) matches the value set to CR00, counting continues with the TM0 value cleared to 0 and the interrupt request signal (INTTM00) is generated.

The count clock of the 16-bit timer/event counter can be selected using bits 4 to 6 (TCL04 to TCL06) of timer clock select register 0 (TCL0).

For the operation when the value of the compare register is changed during the timer/counter operation, see **8.6 (3) Operation after compare register change during timer count operation.**

**Figure 8-10. Control Register Settings for Interval Timer Operation**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with the interval timer. See the description of the respective control registers for details.

Figure 8-11. Interval Timer Configuration Diagram

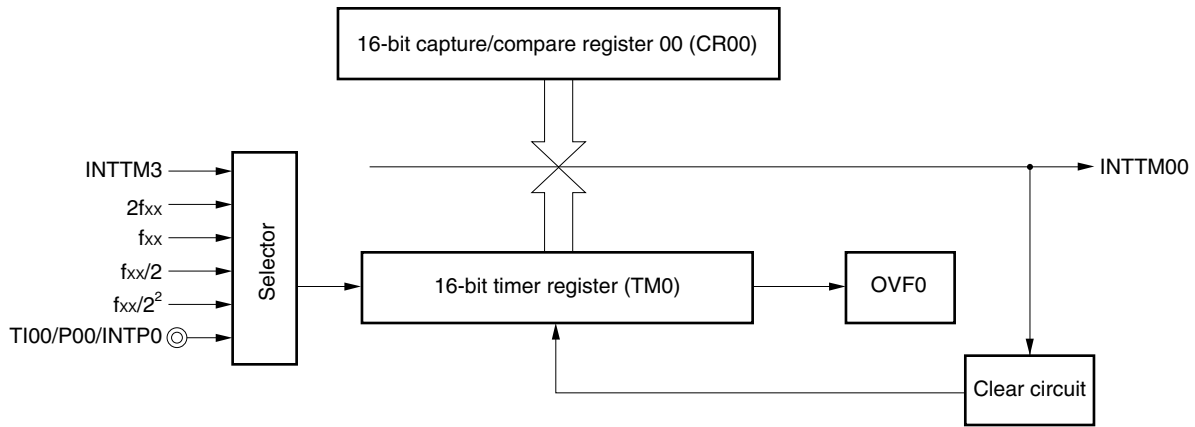
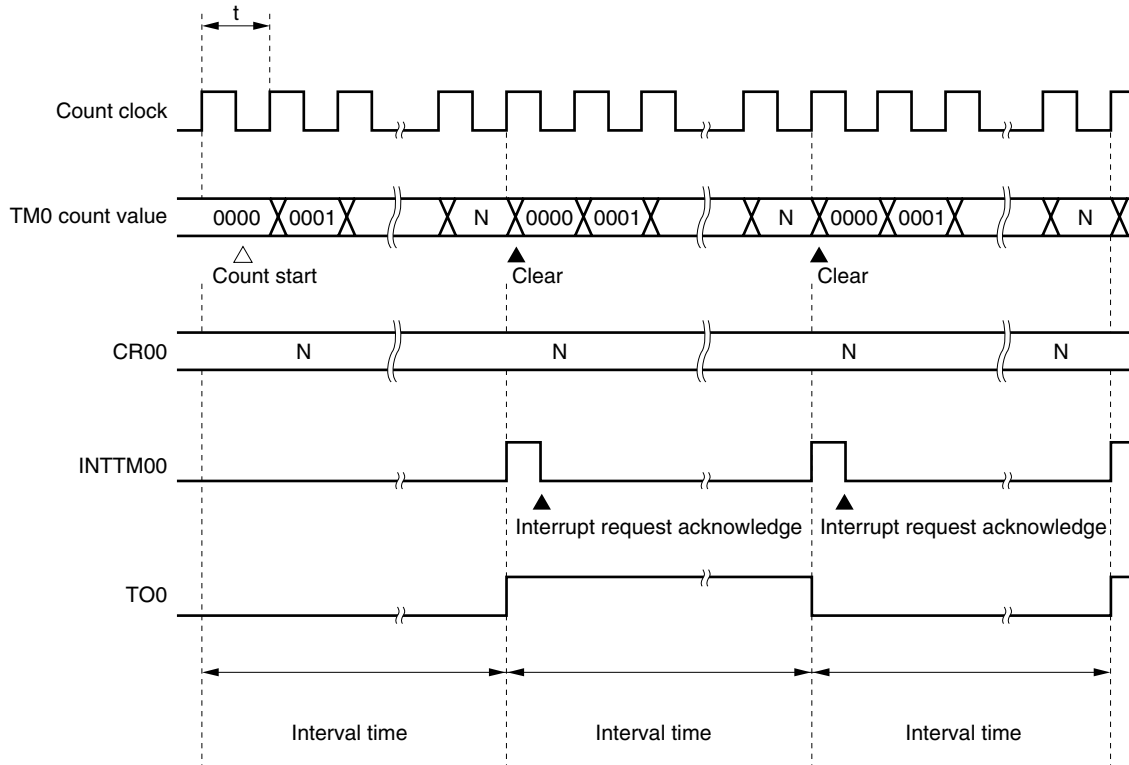


Figure 8-12. Interval Timer Operation Timings



**Remark** Interval time =  $(N + 1) \times t$  :  $N = 0001H$  to  $FFFFH$ .

Table 8-7. 16-Bit Timer/Event Counter Interval Times

TCL06	TCL05	TCL04	Minimum Interval Time		Maximum Interval Time		Resolution	
			MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	2 × TI00 input cycle		2 <sup>16</sup> × TI00 input cycle		TI00 input edge cycle	
0	0	1	Setting prohibited	2 × 1/fx (400 ns)	Setting prohibited	2 <sup>16</sup> × 1/fx (13.1 ms)	Setting prohibited	1/fx (200 ns)
0	1	0	2 × 1/fx (400 ns)	2 <sup>2</sup> × 1/fx (800 ns)	2 <sup>16</sup> × 1/fx (13.1 ms)	2 <sup>17</sup> × 1/fx (26.2 ms)	1/fx (200 ns)	2 × 1/fx (400 ns)
0	1	1	2 <sup>2</sup> × 1/fx (800 ns)	2 <sup>3</sup> × 1/fx (1.6 μs)	2 <sup>17</sup> × 1/fx (26.2 ms)	2 <sup>18</sup> × 1/fx (52.4 ms)	2 × 1/fx (400 ns)	2 <sup>2</sup> × 1/fx (800 ns)
1	0	0	2 <sup>3</sup> × 1/fx (1.6 μs)	2 <sup>4</sup> × 1/fx (3.2 μs)	2 <sup>18</sup> × 1/fx (52.4 ms)	2 <sup>19</sup> × 1/fx (104.9 ms)	2 <sup>2</sup> × 1/fx (800 ns)	2 <sup>3</sup> × 1/fx (1.6 μs)
1	1	1	2 × watch timer output cycle		2 <sup>16</sup> × watch timer output cycle		Watch timer output edge cycle	
Other than above			Setting prohibited					

- Remarks**
1. fx: Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. TCL04 to TCL06: Bits 4 to 6 of timer clock select register (TCL0)
  4. Values in parentheses apply to operation with fx = 5.0 MHz

#### 8.4.2 PWM output operations

Setting the 16-bit timer mode control register (TMC0), capture/compare control register 0 (CRC0), and the 16-bit timer output control register (TOC0) as shown in Figure 8-13 allows operation as PWM output. Pulses with the duty rate determined by the value set to 16-bit capture/compare register 00 (CR00) beforehand are output from the TO0/P30 pin.

Set the active level width of the PWM pulse to the higher 14 bits of CR00. Select the active level using bit 1 (TOC01) of the 16-bit timer output control register (TOC0).

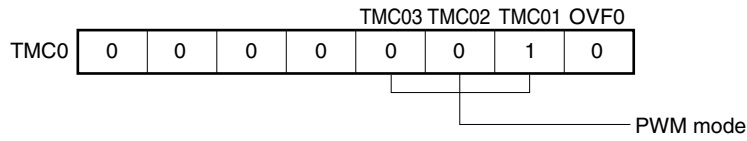
This PWM pulse has a 14-bit resolution. The pulse can be converted to an analog voltage by integrating it with an external low-pass filter (LPF). The PWM pulse is formed by a combination of the basic cycle determined by 2<sup>8</sup>/Φ and the sub-cycle determined by 2<sup>14</sup>/Φ so that the time constant of the external LPF can be shortened. The count clock Φ can be selected using bits 4 to 6 (TCL04 to TCL06) of timer clock select register 0 (TCL0).

PWM output enable/disable can be selected using bit 0 (TOE0) of TOC0.

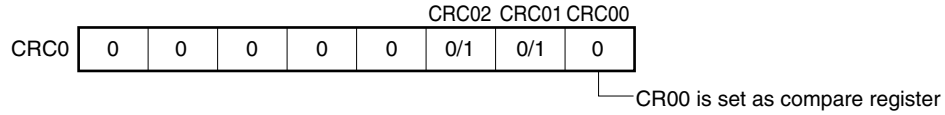
- Cautions**
1. PWM operation mode should be selected before setting CR00.
  2. Be sure to clear bits 0 and 1 of CR00 to 0.
  3. Do not select PWM operation mode for external clock input from the TI00/P00/INTP0 pin.

Figure 8-13. Control Register Settings for PWM Output Operation

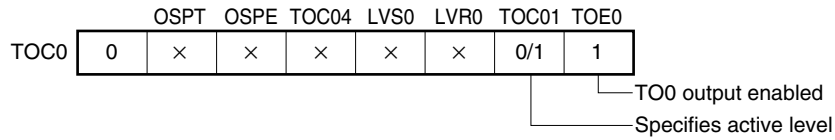
(a) 16-bit timer mode control register (TMC0)



(b) Capture/compare control register 0 (CRC0)



(c) 16-bit timer output control register (TOC0)



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with PWM output. See the description of the respective control registers for details.

x: don't care

By integrating 14-bit resolution PWM pulses with an external low-pass filter, they can be converted to an analog voltage and used for electronic tuning and D/A converter applications, etc.

The analog output voltage ( $V_{AN}$ ) used for D/A conversion with the configuration shown in Figure 8-14 is as follows.

$$V_{AN} = V_{REF} \times \frac{\text{Capture/compare register 00 (CR00) value}}{2^{16}}$$

$V_{REF}$ : External switching circuit reference voltage

Figure 8-14. Example of D/A Converter Configuration with PWM Output

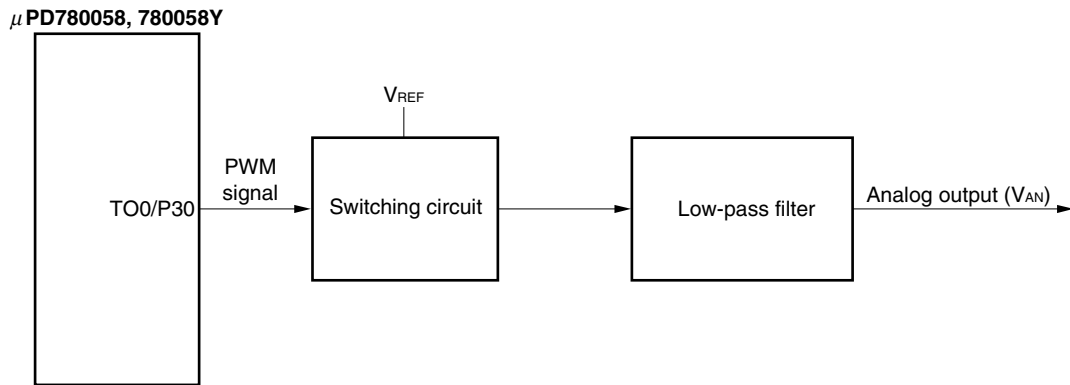
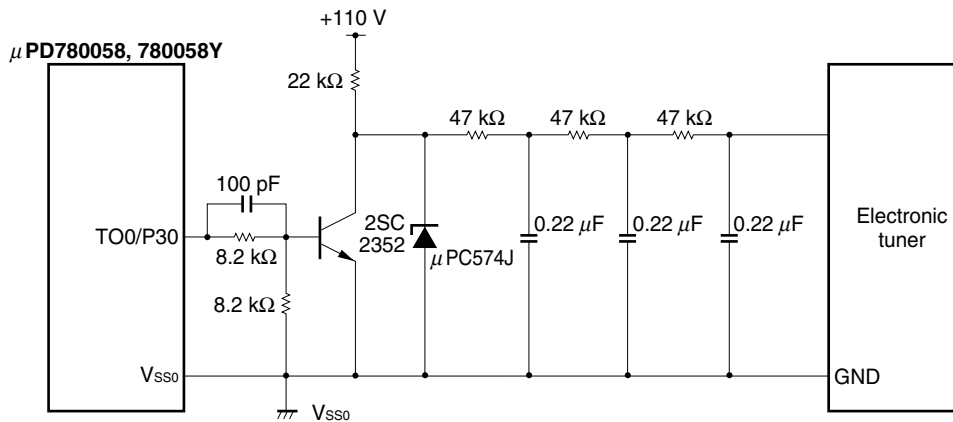


Figure 8-15 shows an example in which PWM output is converted to an analog voltage and used in a voltage synthesizer type TV tuner.

Figure 8-15. TV Tuner Application Circuit Example

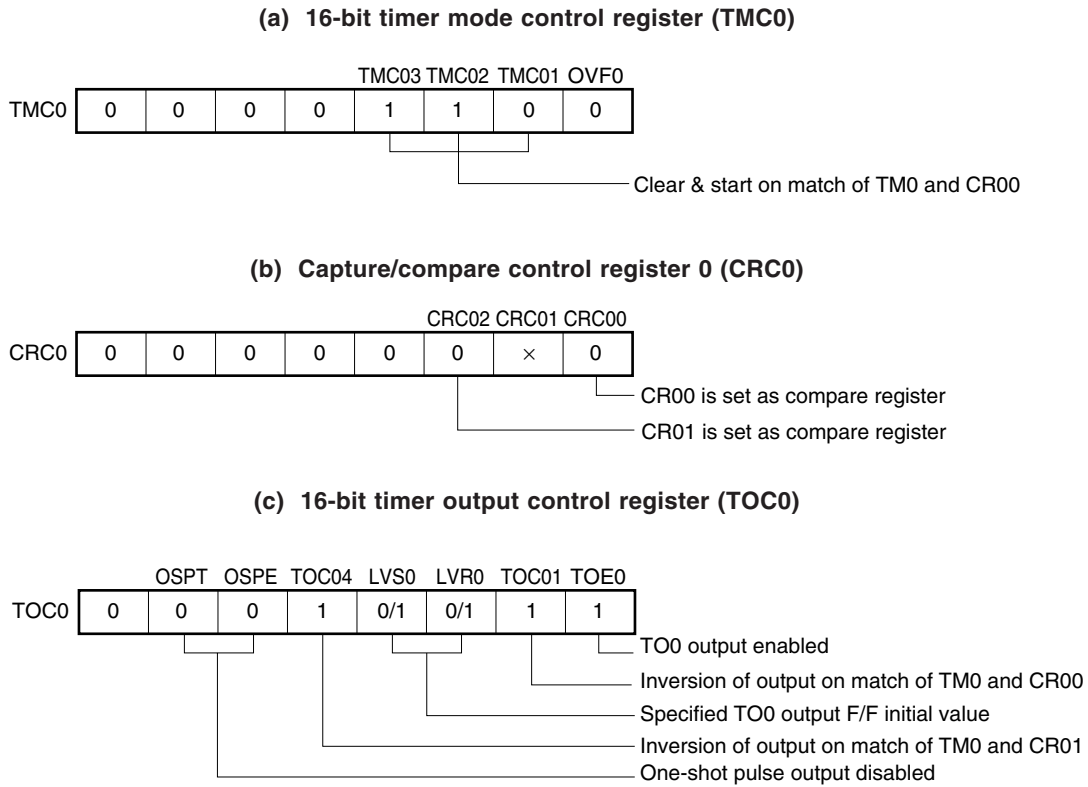


8.4.3 PPG output operations

Setting the 16-bit timer mode control register (TMC0) and capture/compare control register 0 (CRC0) as shown in Figure 8-16 allows operation as PPG (Programmable Pulse Generator) output.

In the PPG output operation, square waves are output from the TO0/P30 pin with the pulse width and the cycle that correspond to the count values set beforehand to 16-bit capture/compare register 01 (CR01) and 16-bit capture/compare register 00 (CR00), respectively.

Figure 8-16. Control Register Settings for PPG Output Operation



Remark × : don't care

- ★ **Cautions**
  1. CR00 and CR01 should be set to values in the following range:  
 $0000H \leq CR01 < CR00n \leq FFFFH$
  2. The cycle of the pulse generated through PPG output (CR00 setting value + 1) has a duty of  $(CR01 \text{ setting value} + 1)/(CR00 \text{ setting value} + 1)$ .



Figure 8-17. Configuration of PPG Output

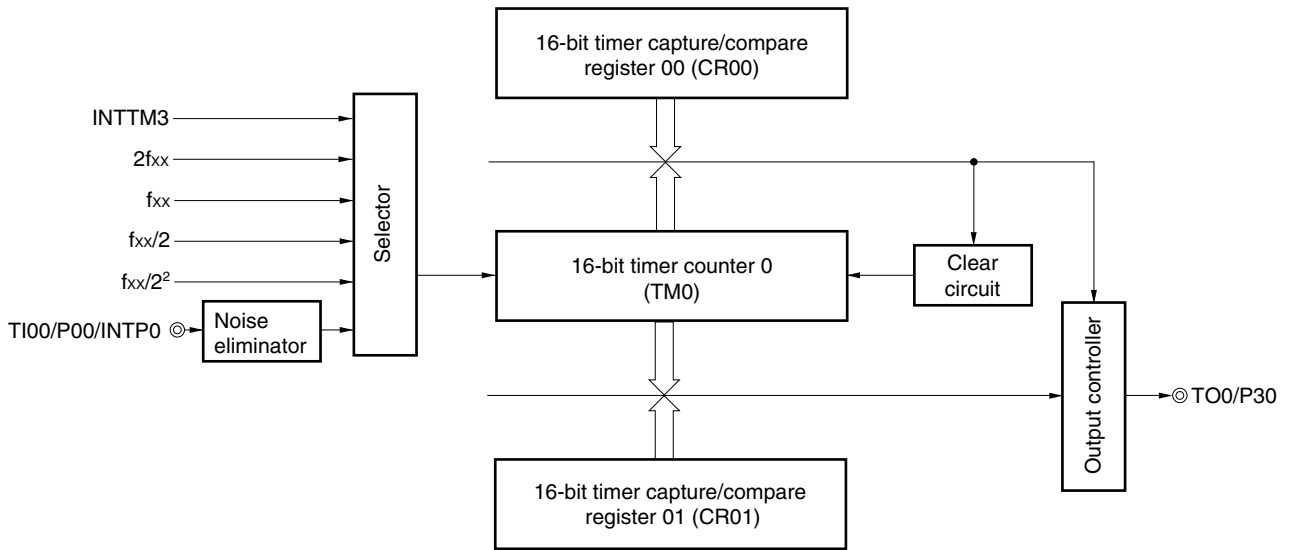
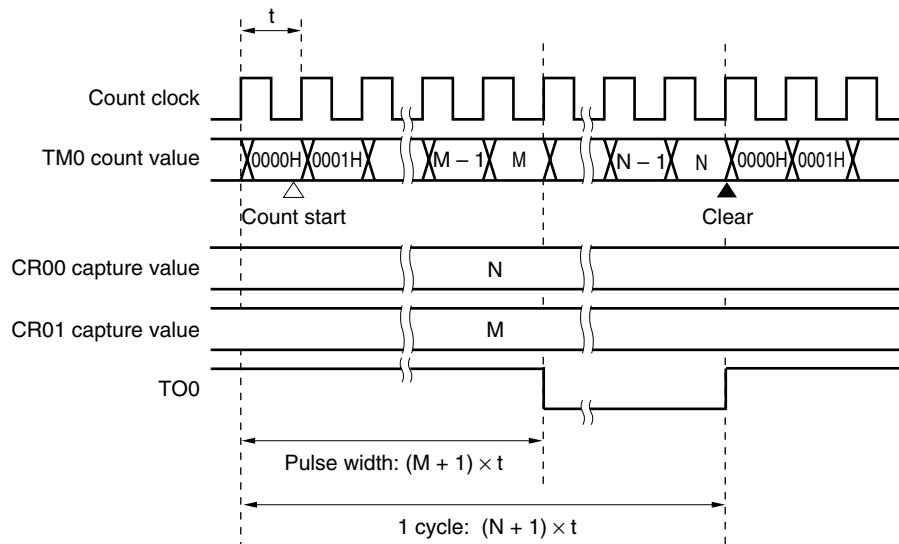


Figure 8-18. PPG Output Operation Timing



**Remark** 0000H < M < N ≤ FFFFH

**8.4.4 Pulse width measurement operations**

It is possible to measure the pulse width of the signals input to the TI00/P00 pin and TI01/P01 pin using the 16-bit timer register (TM0).

There are two measurement methods: measuring with TM0 used in free-running mode, and measuring by restarting the timer in synchronization with the edge of the signal input to the TI00/P00 pin.

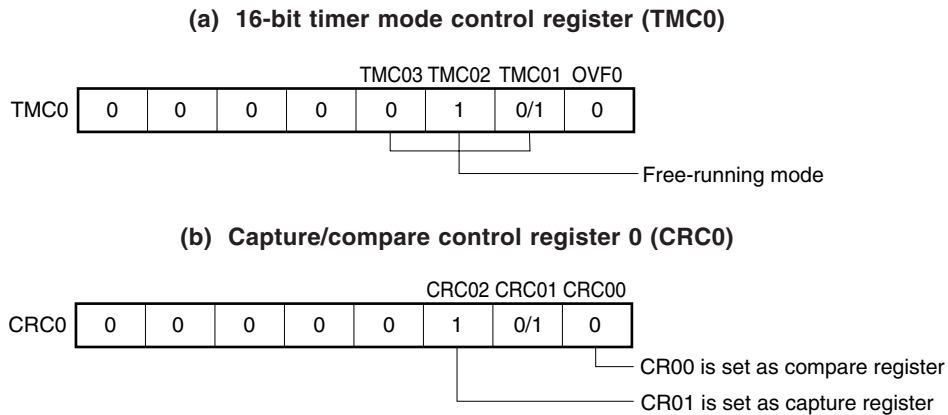
**(1) Pulse width measurement with free-running counter and one capture register**

When the 16-bit timer register (TM0) is operated in free-running mode (see register settings in Figure 8-17), and the edge specified by external interrupt mode register 0 (INTM0) is input to the TI00/P00 pin, the value of TM0 is taken into 16-bit capture/compare register 01 (CR01) and an external interrupt request signal (INTP0) is set.

Any of three edge specifications can be selected—rising, falling, or both edges—by bits 2 and 3 (ES10 and ES11) of INTM0.

For valid edge detection, sampling is performed at the interval selected by the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

**Figure 8-19. Control Register Settings for Pulse Width Measurement with Free-Running Counter and One Capture Register**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

Figure 8-20. Configuration Diagram for Pulse Width Measurement by Free-Running Counter

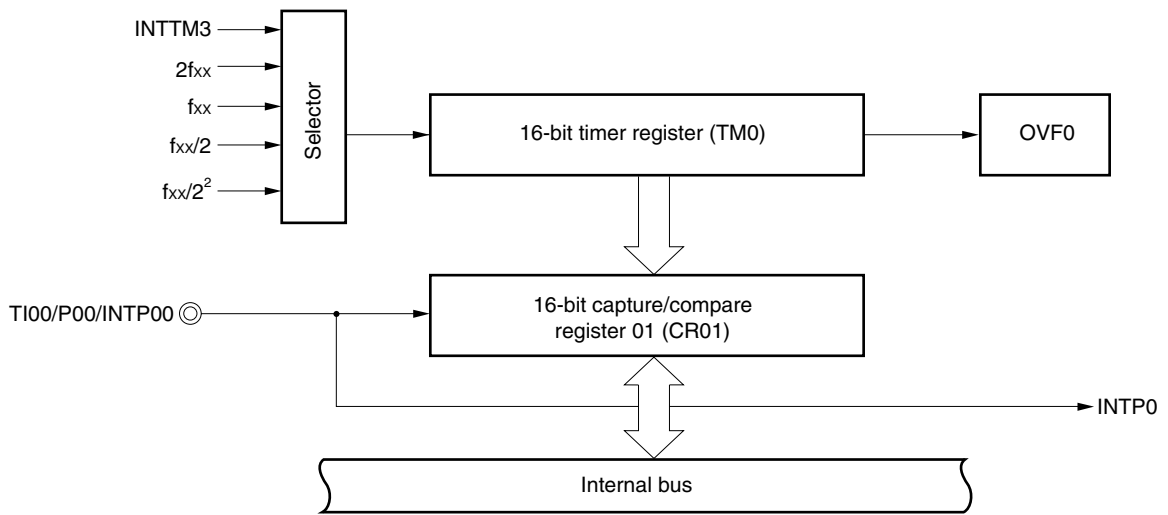
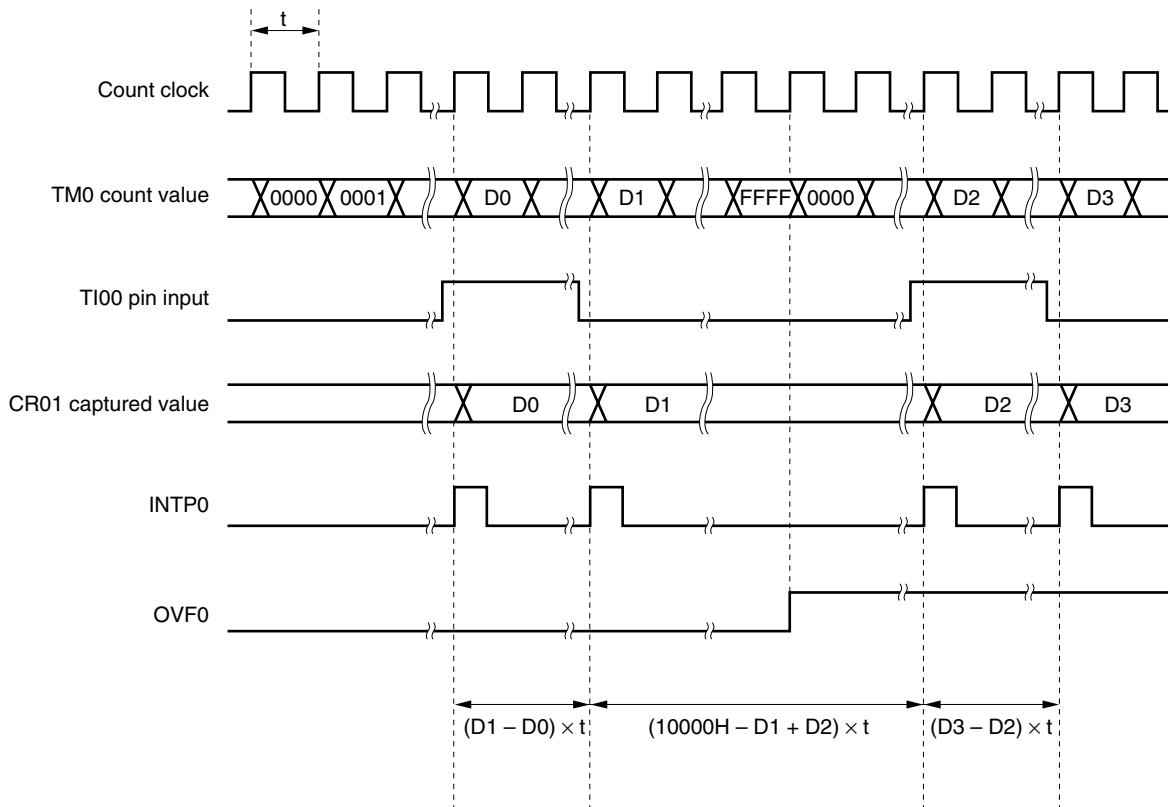


Figure 8-21. Timing of Pulse Width Measurement Operation by Free-Running Counter and One Capture Register (with Both Edges Specified)



**(2) Measurement of two pulse widths with free-running counter**

When the 16-bit timer register (TM0) is operated in free-running mode (see register settings in Figure 8-20), it is possible to simultaneously measure the pulse widths of the two signals input to the TI00/P00 pin and the TI01/P01 pin.

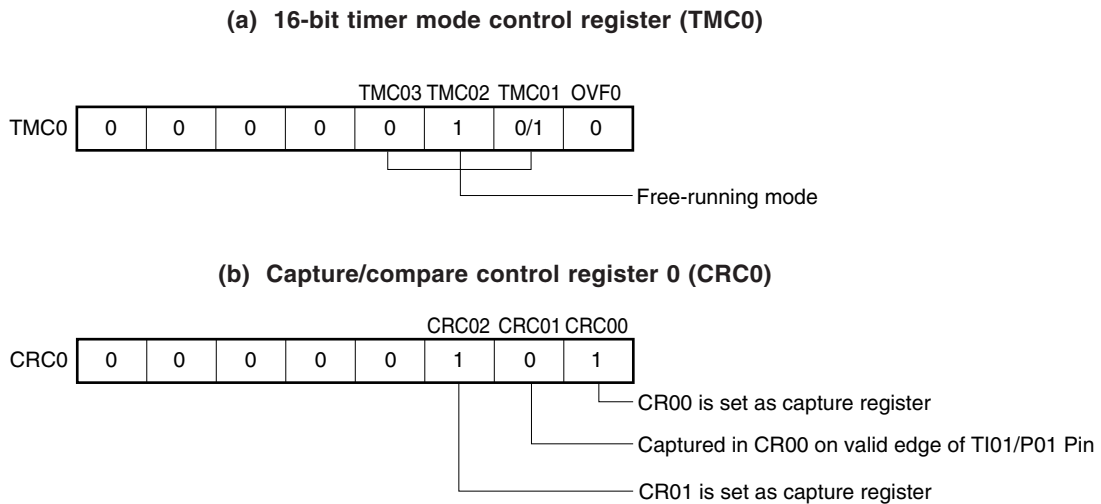
When the edge specified by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0) is input to the TI00/P00 pin, the value of TM0 is taken into 16-bit capture/compare register 01 (CR01) and an external interrupt request signal (INTP0) is set.

Also, when the edge specified by bits 4 and 5 (ES20 and ES21) of INTM0 is input to the TI01/P01 pin, the value of TM0 is taken into 16-bit capture/compare register 00 (CR00) and an external interrupt request signal (INTP1) is set.

Any of three edge specifications can be selected—rising, falling, or both edges—as the valid edges for the TI00/P00 pin and the TI01/P01 pin by bits 2 and 3 (ES10 and ES11) and bits 4 and 5 (ES20 and ES21) of INTM0, respectively.

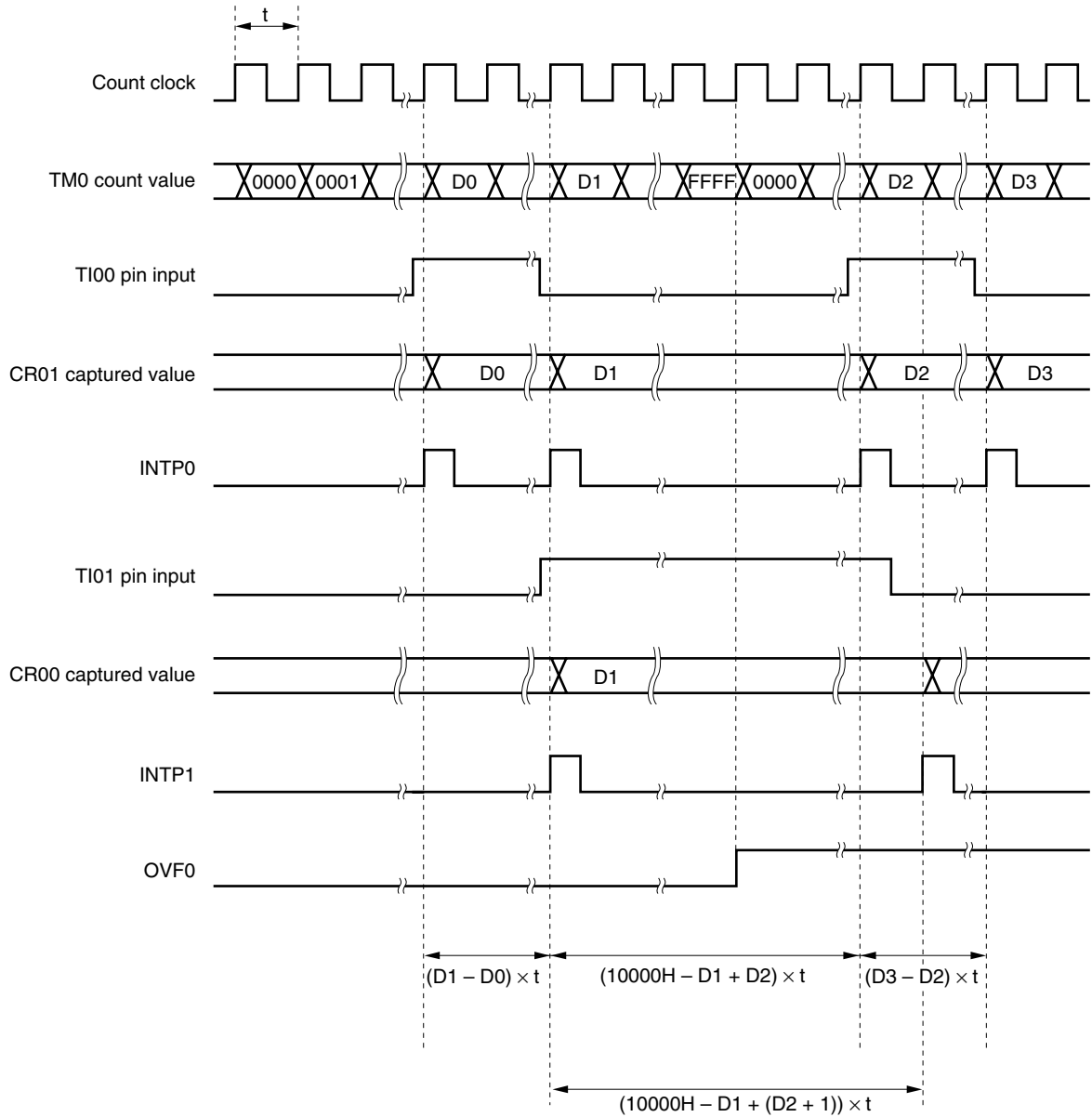
For TI00/P00 pin valid edge detection, sampling is performed at the interval selected by the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

**Figure 8-22. Control Register Settings for Two Pulse Width Measurements with Free-Running Counter**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

**Figure 8-23. Timing of Pulse Width Measurement Operation with Free-Running Counter (with Both Edges Specified)**



**(3) Pulse width measurement with free-running counter and two capture registers**

When the 16-bit timer register (TM0) is operated in free-running mode (see register settings in Figure 8-22), it is possible to measure the pulse width of the signal input to the TI00/P00 pin.

When the edge specified by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0) is input to the TI00/P00 pin, the value of TM0 is taken into 16-bit capture/compare register 01 (CR01) and an external interrupt request signal (INTP0) is set.

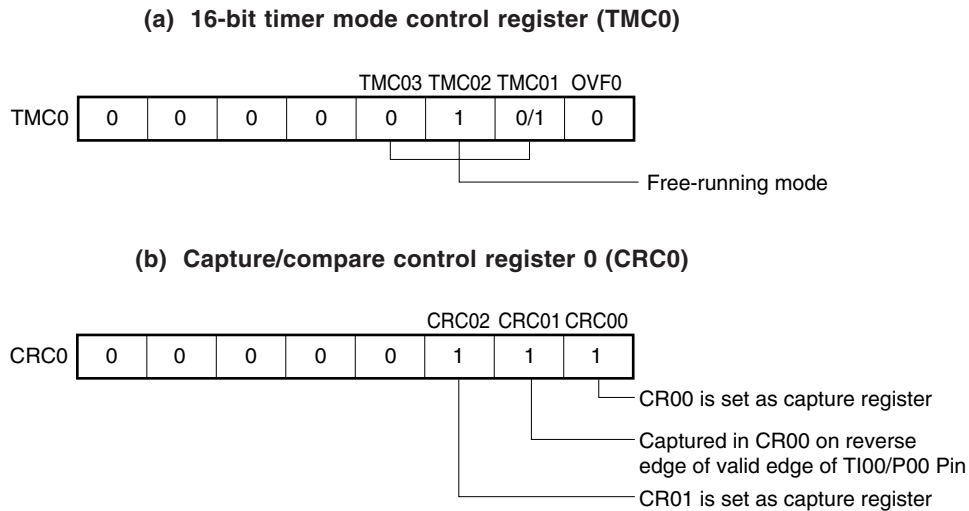
Also, on the reverse input edge to that of the capture operation into CR01, the value of TM0 is taken into 16-bit capture/compare register 00 (CR00).

Either of two edge specifications can be selected—rising or falling—as the valid edges for the TI00/P00 pin by bits 2 and 3 (ES10 and ES11) of INTM0.

For TI00/P00 pin valid edge detection, sampling is performed at the interval selected by the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

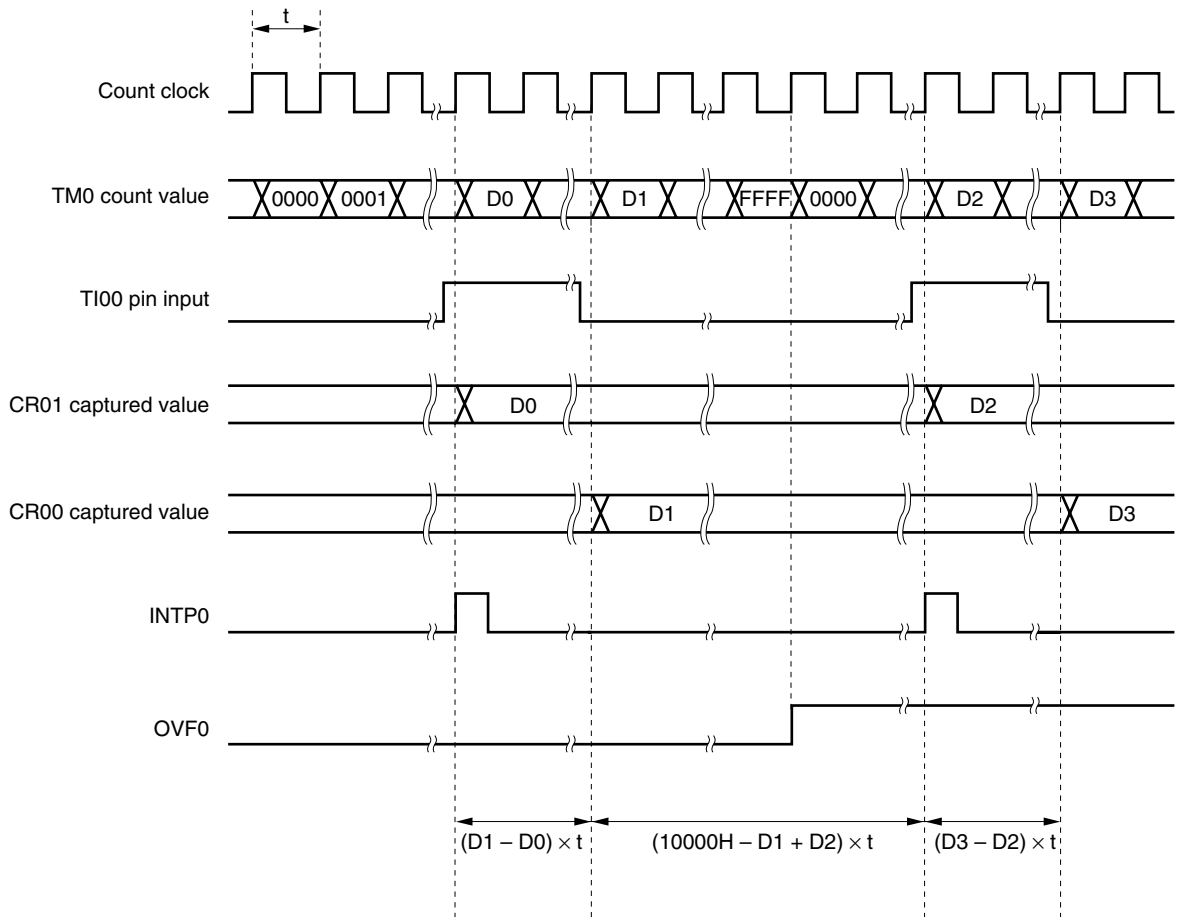
**Caution** If the valid edge of TI00/P00 is specified to be both the rising and falling edges, capture/compare register 00 (CR00) cannot perform the capture operation.

**Figure 8-24. Control Register Settings for Pulse Width Measurement with Free-Running Counter and Two Capture Registers**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

**Figure 8-25. Timing of Pulse Width Measurement Operation by Free-Running Counter and Two Capture Registers (with Rising Edge Specified)**



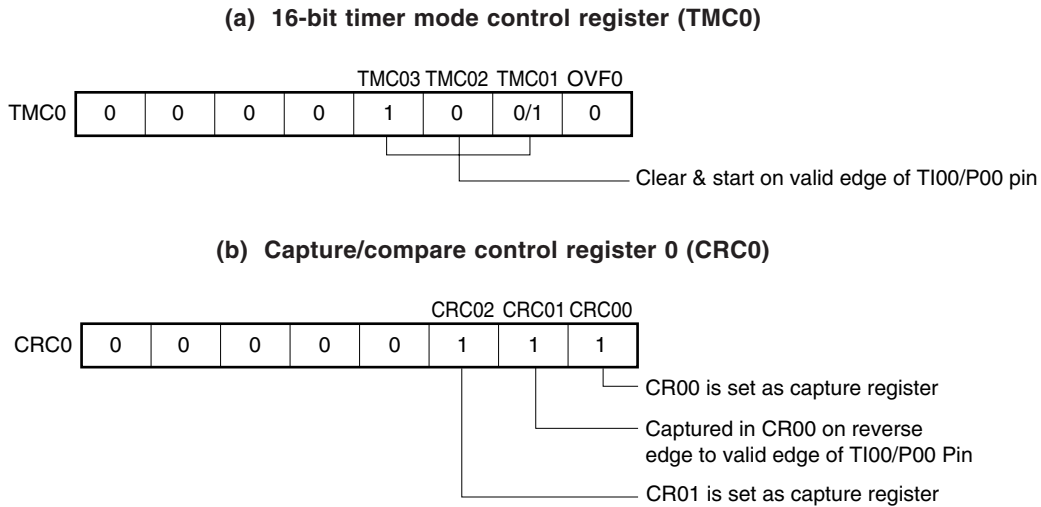
**(4) Pulse width measurement by means of restart**

When input of a valid edge to the TI00/P00 pin is detected, the count value of the 16-bit timer register (TM0) is taken into 16-bit capture/compare register 01 (CR01), and then the pulse width of the signal input to the TI00/P00 pin is measured by clearing TM0 and restarting the count (see register settings in Figure 8-24). The edge specification can be selected from two types, rising and falling edges by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0).

In a valid edge detection, sampling is performed on the cycle selected by the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

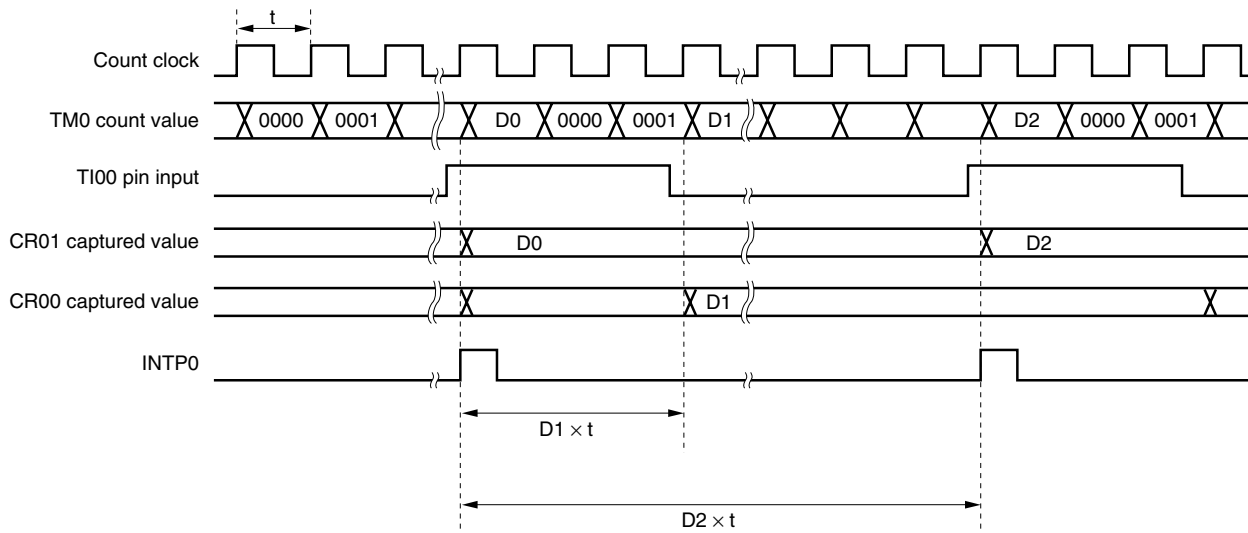
**Caution** If the valid edge of TI00/P00 is specified to be both the rising and falling edges, 16-bit capture/compare register 00 (CR00) cannot perform the capture operation.

**Figure 8-26. Control Register Settings for Pulse Width Measurement by Means of Restart**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

**Figure 8-27. Timing of Pulse Width Measurement Operation by Means of Restart (with Rising Edge Specified)**





**8.4.5 External event counter operation**

The external event counter counts the number of external clock pulses to be input to the T100/P00 pin with the 16-bit timer register (TM0).

TM0 is incremented each time the valid edge specified by external interrupt mode register 0 (INTM0) is input.

When the TM0 counted value matches the 16-bit capture/compare register 00 (CR00) value, TM0 is cleared to 0 and the interrupt request signal (INTTM00) is generated.

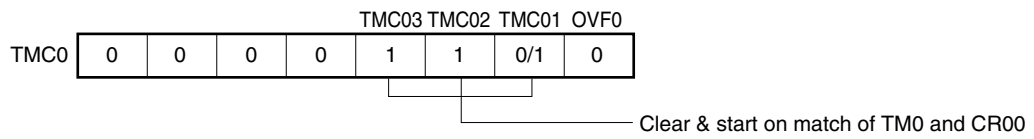
Set CR00 to a value other than 0000H (1-pulse count operation cannot be performed).

The rising edge, falling edge or both edges can be selected using bits 2 and 3 (ES10 and ES11) of INTM0.

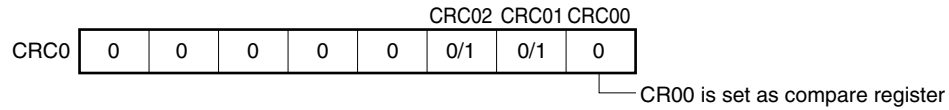
Because operations are carried out only after the valid edge is detected twice by sampling at the interval selected by the sampling clock select register (SCS), noise with short pulse widths can be eliminated.

**Figure 8-28. Control Register Settings in External Event Counter Mode**

**(a) 16-bit timer mode control register (TMC0)**



**(b) Capture/compare control register 0 (CRC0)**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with the external event counter. See the description of the respective control registers for details.

Figure 8-29. External Event Counter Configuration Diagram

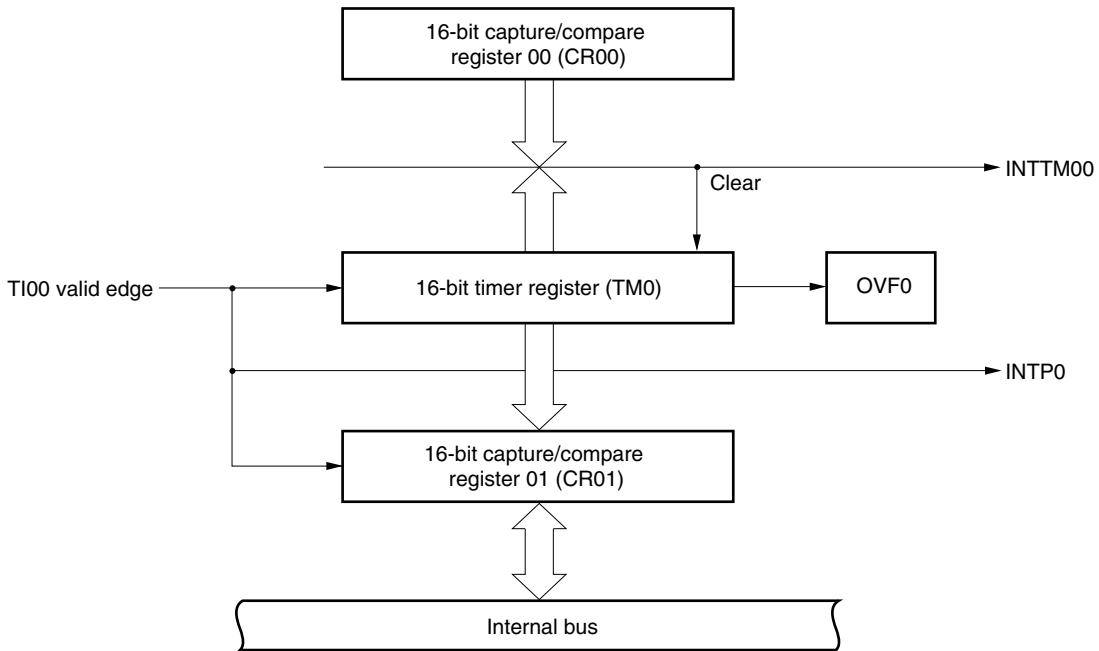
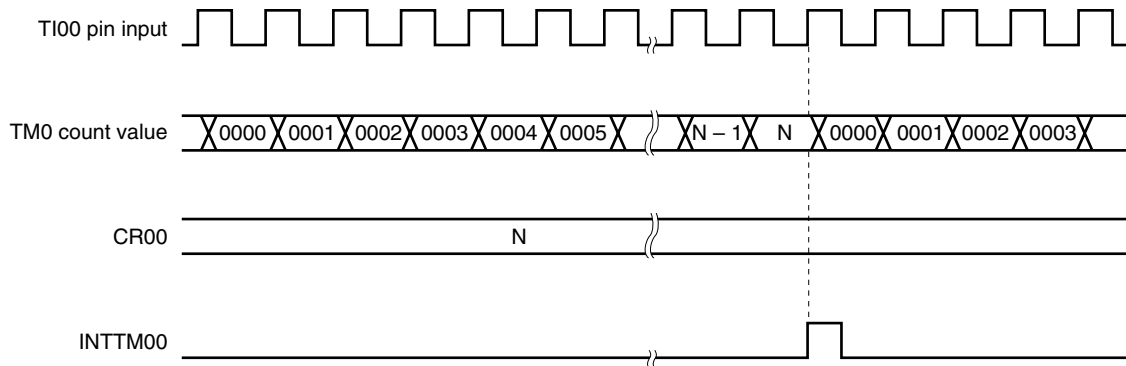


Figure 8-30. External Event Counter Operation Timing (with Rising Edge Specified)



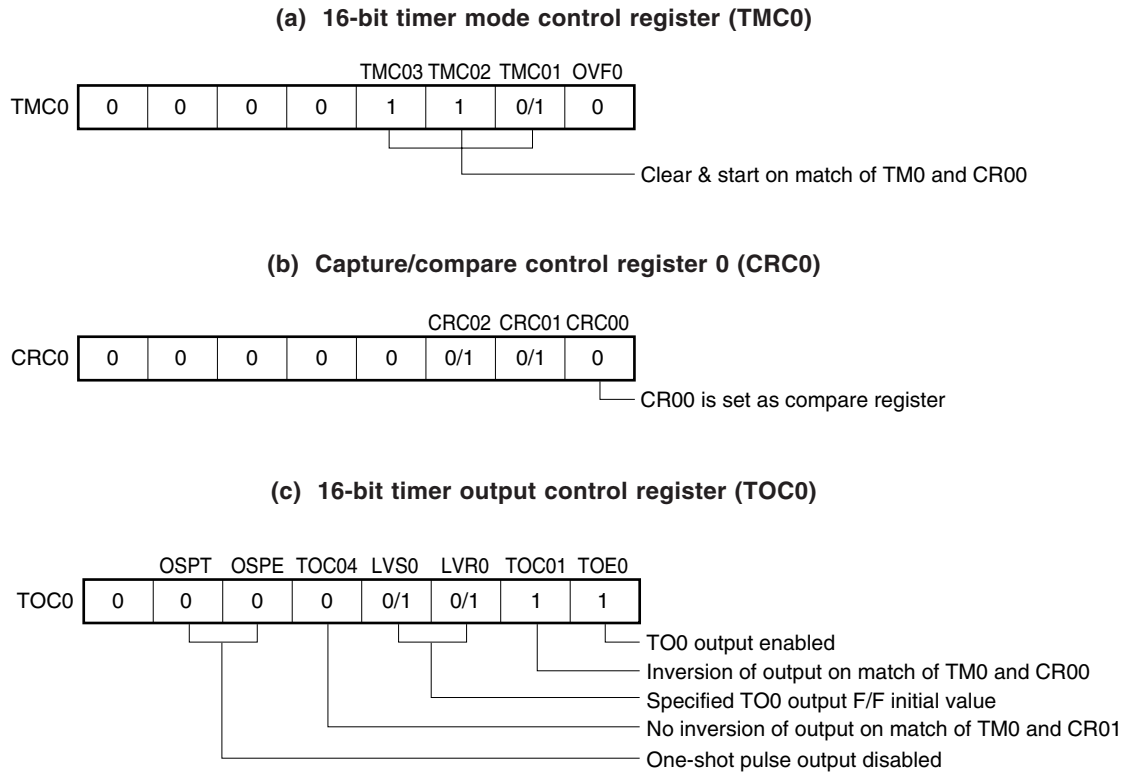
**Caution** When reading the external event counter count value, TM0 should be read.

**8.4.6 Square-wave output operation**

The 16-bit timer/event counter outputs a square wave with any selected frequency at intervals specified by the count value set in advance to 16-bit capture/compare register 00 (CR00).

The TO0/P30 pin output status is reversed at intervals of the count value preset to CR00 by setting bit 0 (TOE0) and bit 1 (TOC01) of the 16-bit timer output control register (TOC0) to 1. This enables a square wave with any selected frequency to be output.

**Figure 8-31. Control Register Settings in Square-Wave Output Mode**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with square-wave output. See the description of the respective control registers for details.

Figure 8-32. Square-Wave Output Operation Timing

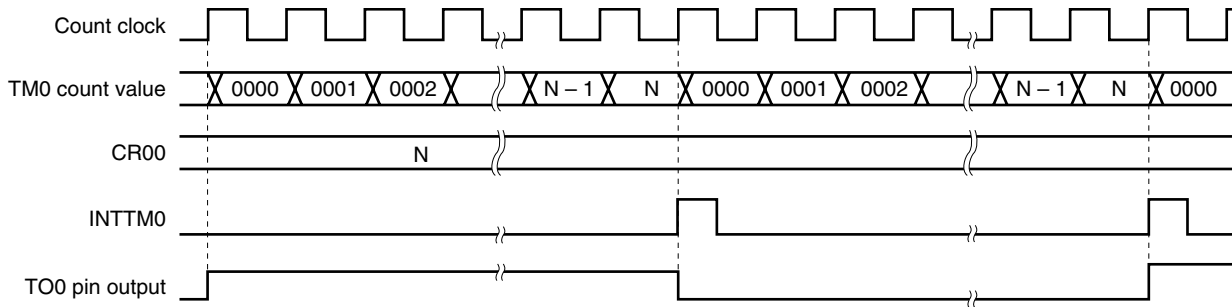


Table 8-8. 16-Bit Timer/Event Counter Square-Wave Output Ranges

Minimum Pulse Time		Maximum Pulse Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × T100 input cycle		2 <sup>16</sup> × T100 input cycle		T100 input edge cycle	
—	2 × 1/f <sub>x</sub> (400 ns)	—	2 <sup>16</sup> × 1/f <sub>x</sub> (13.1 ms)	—	1/f <sub>x</sub> (200 ns)
2 × 1/f <sub>x</sub> (400 ns)	2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)	2 <sup>16</sup> × 1/f <sub>x</sub> (13.1 ms)	2 <sup>17</sup> × 1/f <sub>x</sub> (26.2 ms)	1/f <sub>x</sub> (200 ns)	2 × 1/f <sub>x</sub> (400 ns)
2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)	2 <sup>3</sup> × 1/f <sub>x</sub> (1.6 μs)	2 <sup>17</sup> × 1/f <sub>x</sub> (26.2 ms)	2 <sup>18</sup> × 1/f <sub>x</sub> (52.4 ms)	2 × 1/f <sub>x</sub> (400 ns)	2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)
2 <sup>3</sup> × 1/f <sub>x</sub> (1.6 μs)	2 <sup>4</sup> × 1/f <sub>x</sub> (3.2 μs)	2 <sup>18</sup> × 1/f <sub>x</sub> (52.4 ms)	2 <sup>19</sup> × 1/f <sub>x</sub> (104.9 ms)	2 <sup>2</sup> × 1/f <sub>x</sub> (800 ns)	2 <sup>3</sup> × 1/f <sub>x</sub> (1.6 μs)
2 × watch timer output cycle		2 <sup>16</sup> × watch timer output cycle		Watch timer output edge cycle	

- Remarks**
1. f<sub>x</sub>: Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with f<sub>x</sub> = 5.0 MHz

8.4.7 One-shot pulse output operation

The 16-bit timer/event counter can be started in synchronization with a software trigger or external trigger (TI00/P00 pin input) and output a one-shot pulse that ends on overflow of TM0.

(1) One-shot pulse output using software trigger

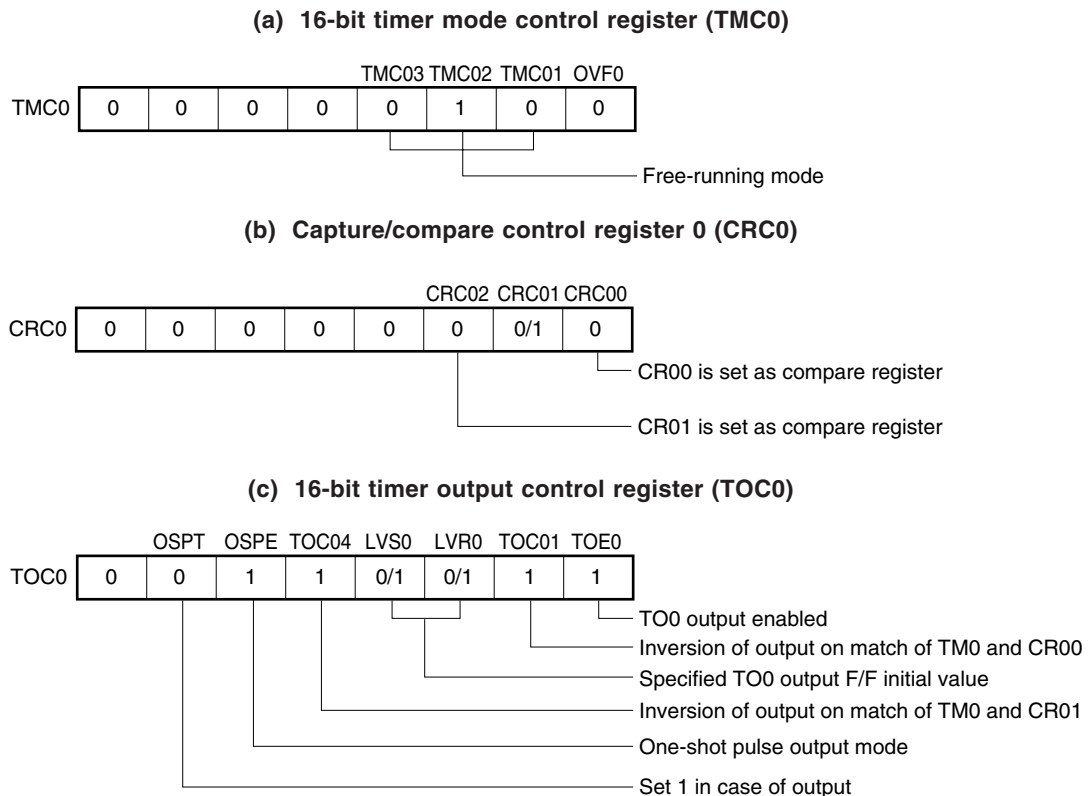
If the 16-bit timer mode control register (TMC0), capture/compare control register 0 (CRC0), and the 16-bit timer output control register (TOC0) are set as shown in Figure 8-33, and bit 6 (OSPT) of TOC0 is set to 1 by software, a one-shot pulse is output from the TO0/P30 pin.

By setting OSPT to 1, the 16-bit timer/event counter is cleared and started, and output is activated by the count value (N) set beforehand to 16-bit capture/compare register 01 (CR01). Thereafter, output is inactivated<sup>Note</sup> by the count value (M) set beforehand in 16-bit capture/compare register 00 (CR00).

TM0 continues to operate after a one-shot pulse is output. To stop TM0, TMC0 must be set to 00H.

- ★ **Note** The case where  $N < M$  is described here. When  $N > M$ , the output becomes active with the CR00 register and inactive with the CR01 register.
- ★ **Cautions**
  1. When a one-shot pulse is output by a software trigger, fix the TI00/P00 pin to either the high or low level.
  2. When outputting a one-shot pulse, do not set OSPT to 1. To output a one-shot pulse again, wait until the current one-shot pulse output is completed.

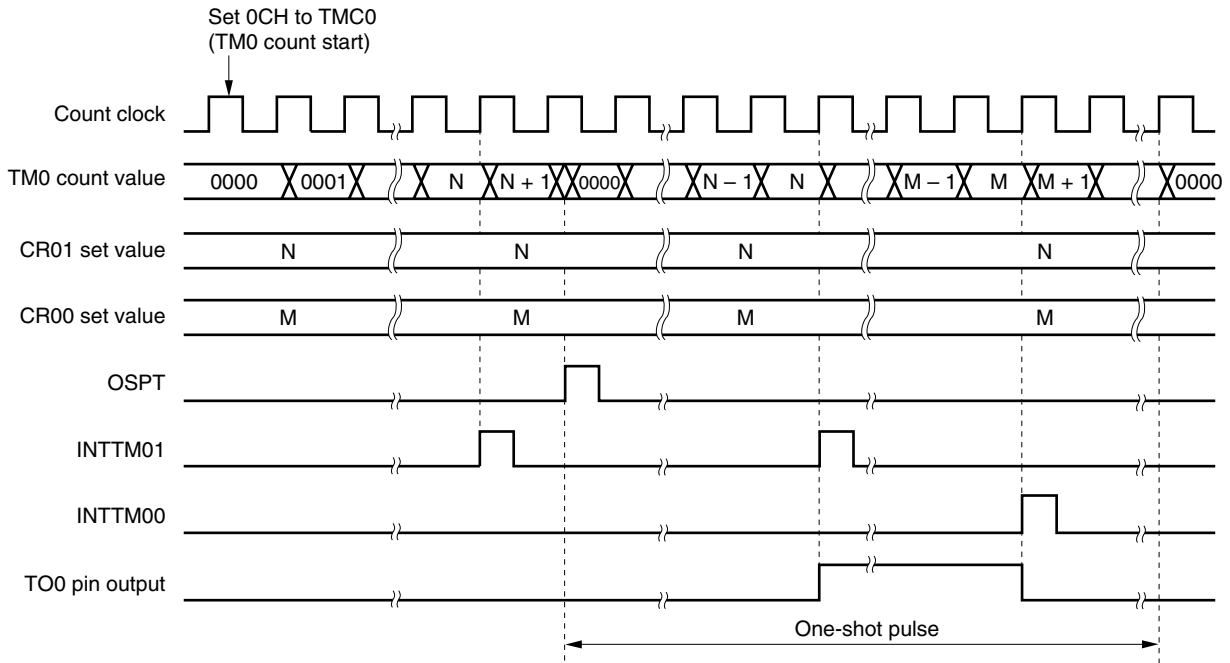
Figure 8-33. Control Register Settings for One-Shot Pulse Output Operation Using Software Trigger



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with one-shot pulse output. See the description of the respective control registers for details.

- ★ **Caution** Do not clear CR00 and CR01 to 0000H.

★ **Figure 8-34. One-Shot Pulse Output Operation Timing Using Software Trigger**



**Caution** The 16-bit timer register starts operation at the moment TMC01 to TMC03 are set to values other than 0, 0, 0 (operation stop mode).

**Remark**  $N < M$

**(2) One-shot pulse output using external trigger**

If the 16-bit timer mode control register (TMC0), capture/compare control register 0 (CRC0), and the 16-bit timer output control register (TOC0) are set as shown in Figure 8-33, a one-shot pulse is output from the TO0/P30 pin with a TI00/P00 valid edge as an external trigger.

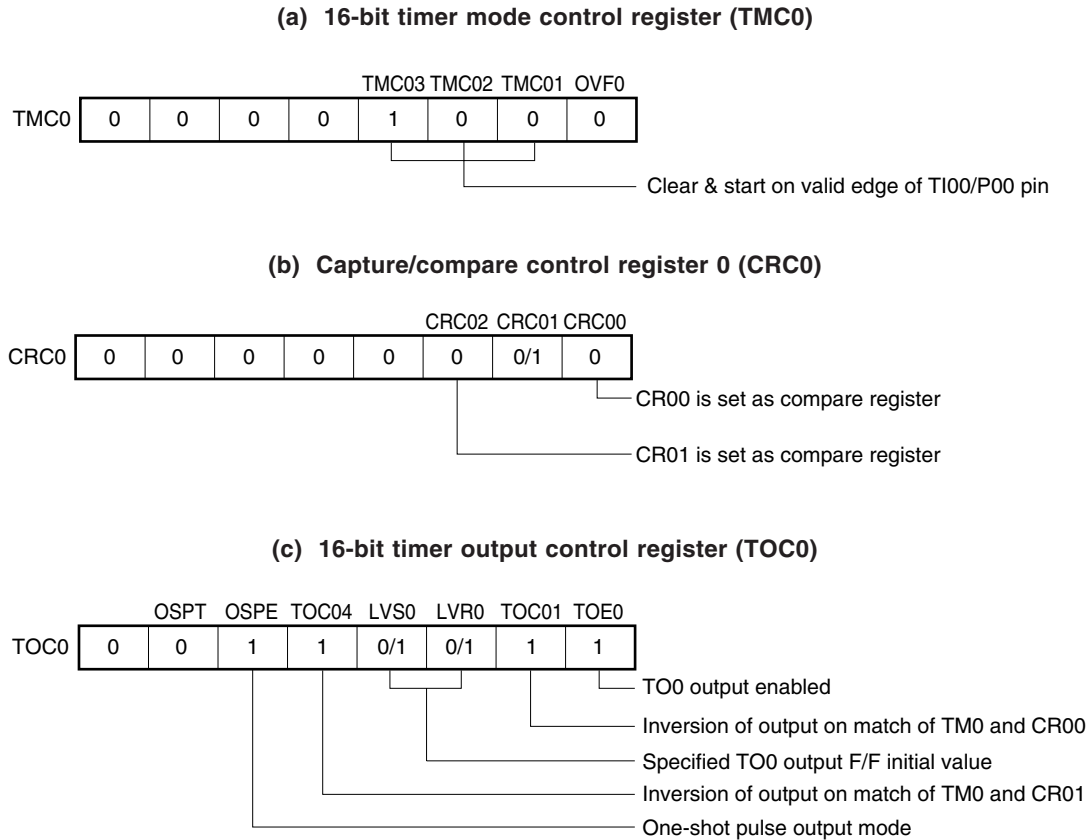
Any of three edge specifications can be selected—rising, falling, or both edges—as the valid edges for the TI00/P00 pin by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0).

When a valid edge is input to the TI00/P00 pin, the 16-bit timer/event counter is cleared and started, and output is activated by the count values (N) set beforehand in 16-bit capture/compare register 01 (CR01). Thereafter, output is inactivated<sup>Note</sup> by the count value (M) set beforehand in 16-bit capture/compare register 00 (CR00).

★ **Note** The case where  $N < M$  is described here. When  $N > M$ , the output becomes active with the CR00 register and inactive with the CR01 register.

**Caution** When outputting one-shot pulses, the external trigger is ignored if generated again.

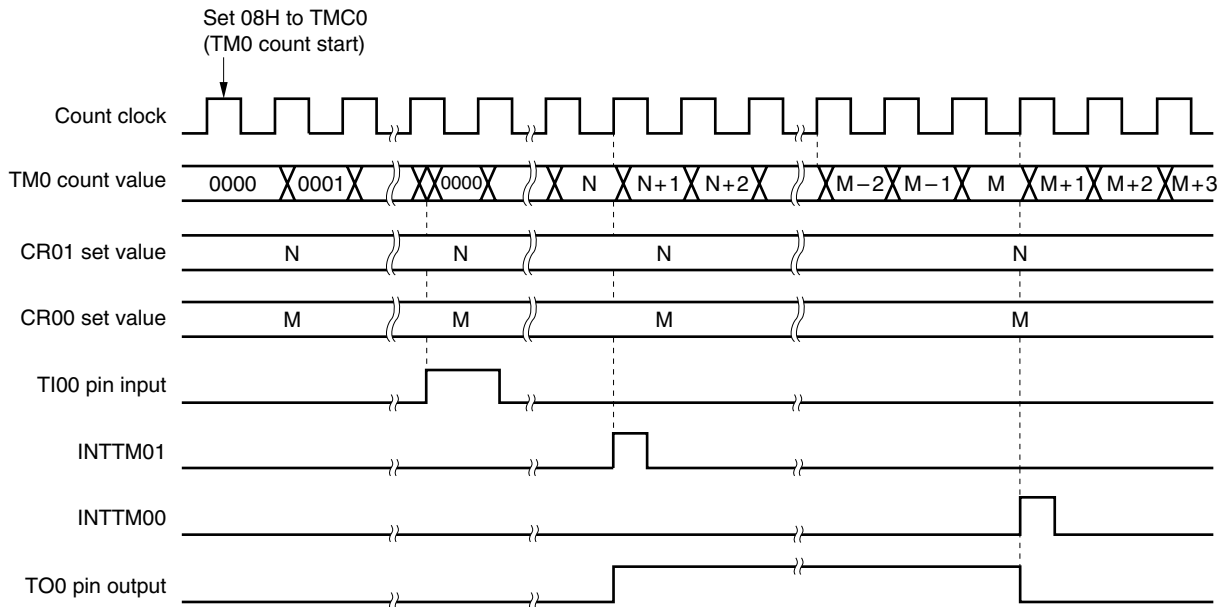
**Figure 8-35. Control Register Settings for One-Shot Pulse Output Operation Using External Trigger**



**Remark** 0/1: Setting 0 or 1 allows another function to be used simultaneously with one-shot pulse output. See the description of the respective control registers for details.

★ **Caution** Do not clear CR00 and CR01 to 0000H.

**Figure 8-36. One-Shot Pulse Output Operation Timing Using External Trigger (with Rising Edge Specified)**



**Caution** The 16-bit timer register starts operation at the moment TMC01 to TMC03 are set to values other than 0, 0, 0 (operation stop mode).

**Remark**  $N < M$

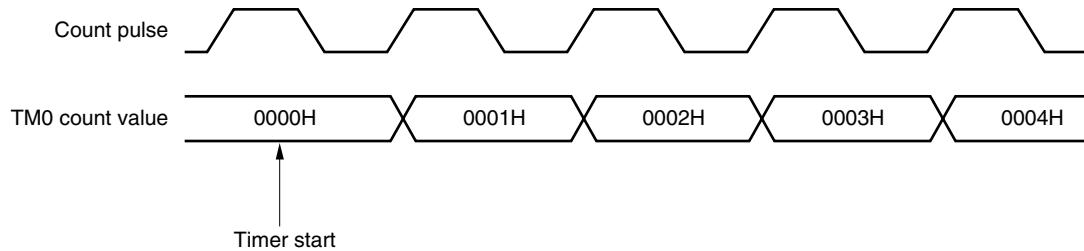


## 8.5 16-Bit Timer/Event Counter Operating Cautions

### (1) 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 the 16-bit timer register (TM0) is started asynchronously to the count pulse.

Figure 8-37. 16-Bit Timer Register Start Timing



### (2) 16-bit compare register setting (when in the clear & start mode entered on a match between TM0 and CR00)

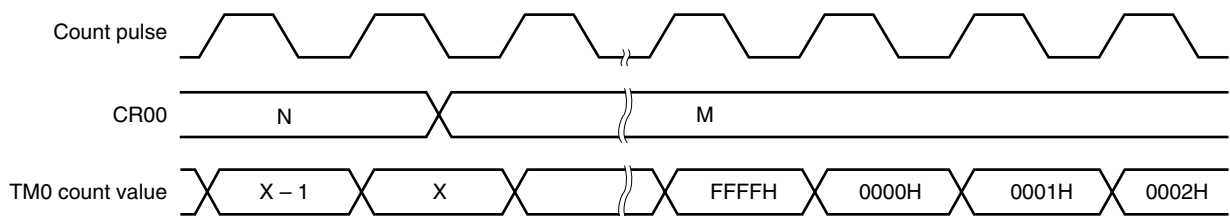
Set 16-bit capture/compare register 00 (CR00) to the a value other than 0000H.

Thus, when using the 16-bit capture/compare register as event counter, one-pulse count operation cannot be carried out.

### (3) Operation after compare register change during timer count operation

If the value after the 16-bit capture/compare register (CR00) is changed is smaller than that of the 16-bit timer register (TM0), TM0 continues counting, overflows and then restarts counting from 0. Thus, if the value after CR00 change (M) is smaller than that before change (N), it is necessary to reset and restart the timer after changing CR00.

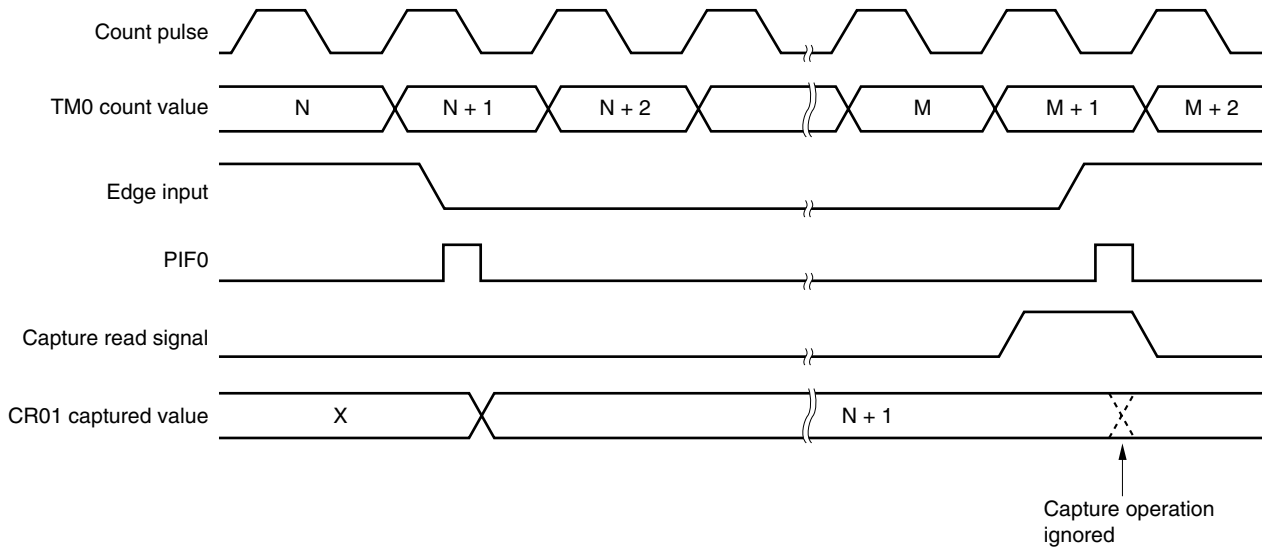
Figure 8-38. Timing After Change of Compare Register During Timer Count Operation



**Remark**  $N > X > M$

**(4) Capture register data retention timing**

If the valid edge of the TI00/P00 pin is input during 16-bit capture/compare register 01 (CR01) read, CR01 holds the data without carrying out a capture operation. However, the interrupt request signal (PIF0) is set upon detection of the valid edge.

**Figure 8-39. Capture Register Data Retention Timing****★ (5) Valid edge setting**

When using the TI00/P00/INTP0 and TI01/P01/INTP1 pins as timer input pins (TI00 and TI01), stop the operation of 16-bit timer 0 by clearing bits 1 to 3 (TMC01 to TMC03) of the 16-bit timer mode control register (TMC0) to 0, 0, 0, before setting the valid edge of TI00 and TI01. The valid edge is set by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0). When using the TI00/P00/INTP0 and TI01/P01/INTP1 pins as external interrupt input pins (INTP0 and INTP1), the valid edge of INTP0 and INTP1 may be set while 16-bit timer 0 is operating.

**(6) Re-trigger of one-shot pulse****(a) One-shot pulse output using software**

When outputting a one-shot pulse, do not set OSPT to 1. To output a one-shot pulse again, wait until the current one-shot pulse output is completed.

**(b) One-shot pulse output using external trigger**

When outputting one-shot pulses, the external trigger is ignored if generated again.

**★ (c) One-shot pulse output function**

When using the software trigger for one-shot pulse output, fix the level of the TI00/P00/INTP0 and TI01/P01/INTP1 pins to either the high or low level. Otherwise, the external trigger will remain valid even when the software trigger is used, and the timer will be cleared and started when the level of the TI00/P00/INTP0 or TI01/P01/INTP1 pin changes. In consequence, the pulse will be output unexpectedly.

**(7) Operation of OVF0 flag****(a) OVF0 flag setting**

OVF0 flag is set to 1 in the following case.

When one of clear & start mode on match between TM0 and CR00, clear & start mode on T100 valid edge, or free-running mode is selected.

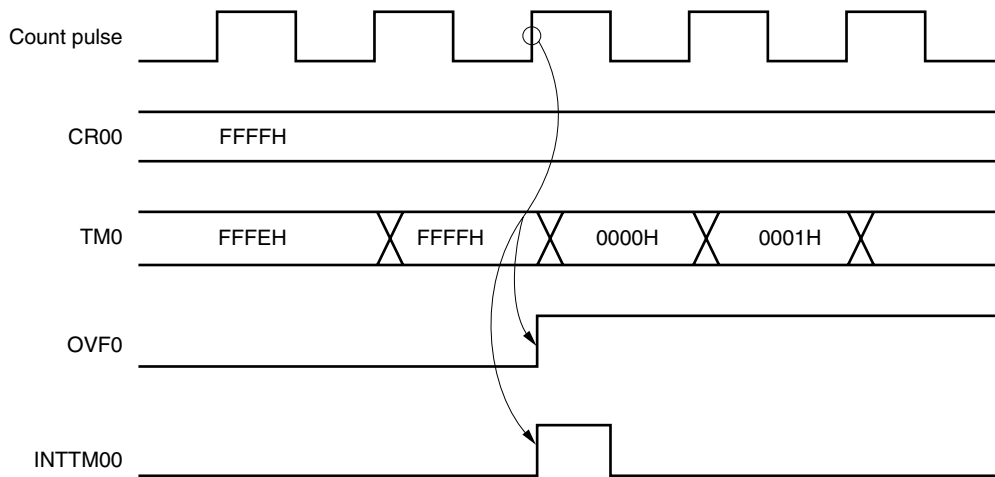


CR00 is set to FFFFH.



TM0 is counted up from FFFFH to 0000H.

**Figure 8-40. Operation Timing of OVF0 Flag**

**(b) Clear OVF0 flag**

Even if the OVF0 flag is cleared before the next count clock is counted (before TM0 becomes 0001H) after TM0 has overflowed, the OVF0 flag is set again and the clear becomes invalid.

**(8) Conflict operation****(a) If the read period and capture trigger input conflict**

If the read period and inputting a capture trigger conflict while 16-bit capture/compare registers 00 and 01 (CR00 and CR01) are used as capture registers, the registers do not perform a capture operation but hold data. However, the interrupt request flag (PIF0) is set when the valid edge is detected.

**(b) If the match timing of the write period and TM0 conflict**

When 16-bit capture/compare registers 00 and 01 (CR00, CR01) are used as capture registers, because match detection cannot be performed correctly if the match timing of the write period and 16-bit timer register 0 (TM0) conflict, do not write to CR00 and CR01 close to the match timing.

**(9) Timer operation****(a) CR01 capture**

Even if 16-bit timer register 0 (TM0) is read, a capture to 16-bit capture/compare register 01 (CR01) is not performed.

**(b) Acknowledgement of TI00 and TI01 pins**

When the timer is stopped, input signals to the TI00 and TI01 pins are not acknowledged, regardless of the CPU operation.

**(10) Capture operation****(a) If the valid edge of TI00 is specified for the count clock**

When the valid edge of TI00 is specified for the count clock, the capture register with TI00 specified as a trigger will not operate correctly.

**(b) If both rising and falling edges are selected as the valid edge of TI00.**

When both the rising and falling edges are selected as the valid edge of TI00, CR00 cannot perform a capture operation with TI00 specified as the capture trigger.

**(c) To use signal from TI00 as capture trigger**

For an accurate capture operation, a pulse longer than twice the width of the count clock selected by the sampling clock select register (SCS) is necessary.

**(11) Compare operation****(a) When rewriting CR00 and CR01 during timer operation**

When rewriting 16-bit timer capture/compare registers 00 and 01 (CR00, CR01), if the value is close to or larger than the timer value, the match interrupt request generation or clear operation may not be performed correctly.

**(b) When CR00 and CR01 are set to compare mode**

When CR00 and CR01 are set to compare mode, they do not perform a capture operation even if a capture trigger is input.

**(12) Edge detection****(a) When the TI00 or TI01 pin is high level immediately after a system reset**

When the TI00 or TI01 pin is high level immediately after a system reset, if the valid edge of the TI00 or TI01 pin is specified as the rising edge or both rising and falling edges, and the operation of 16-bit timer/counter 0 (TM0) is then enabled, the rising edge will be detected immediately. Care is therefore needed when the TI00 or TI01 pin is pulled up. However, when operation is enabled after being stopped, the rising or falling edge is not detected.

## CHAPTER 9 8-BIT TIMER/EVENT COUNTER

### 9.1 8-Bit Timer/Event Counter Functions

For the 8-bit timer/event counter, two modes are available. One is a mode for the two 8-bit timer/event counter channels to be used separately (the 8-bit timer/event counter mode) and the other is a mode for the 8-bit timer/event counter to be used as 16-bit timer/event counter (the 16-bit timer/event counter mode).

#### 9.1.1 8-bit timer/event counter mode

The 8-bit timer/event counters 1 and 2 (TM1 and TM2) have the following functions.

- Interval timer
- External event counter
- Square-wave output

(1) 8-bit interval timer

Interrupt requests are generated at the preset time intervals.

Table 9-1. Interval Times of 8-Bit Timer/Event Counters 1 and 2

Minimum Interval Time		Maximum Interval Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

**(2) External event counter**

The number of pulses of an externally input signal can be measured.

**(3) Square-wave output**

A square wave with any selected frequency can be output.

**Table 9-2. Square-Wave Output Ranges of 8-Bit Timer/Event Counters 1 and 2**

Minimum Pulse Time		Maximum Pulse Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

## 9.1.2 16-bit timer/event counter mode

## (1) 16-bit interval timer

Interrupt requests can be generated at the preset time intervals.

**Table 9-3. Interval Times When 8-Bit Timer/Event Counters 1 and 2 Are Used as 16-Bit Timer/Event Counter**

Minimum Interval Time		Maximum Interval Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^{23} \times 1/f_x$ (1.7 s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{23} \times 1/f_x$ (1.7 s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^{26} \times 1/f_x$ (13.4 s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{27} \times 1/f_x$ (26.8 s)	$2^{28} \times 1/f_x$ (53.7 s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.



**(2) External event counter**

The number of pulses of an externally input signal can be measured.

**(3) Square-wave output**

A square wave with any selected frequency can be output.

**Table 9-4. Square-Wave Output Ranges When 8-Bit Timer/Event Counters 1 and 2 Are Used as 16-Bit Timer/Event Counter**

Minimum Pulse Time		Maximum Pulse Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^{23} \times 1/f_x$ (1.7 s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{23} \times 1/f_x$ (1.7 s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^{26} \times 1/f_x$ (13.4 s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{27} \times 1/f_x$ (26.8 s)	$2^{28} \times 1/f_x$ (53.7 s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

## 9.2 8-Bit Timer/Event Counter Configuration

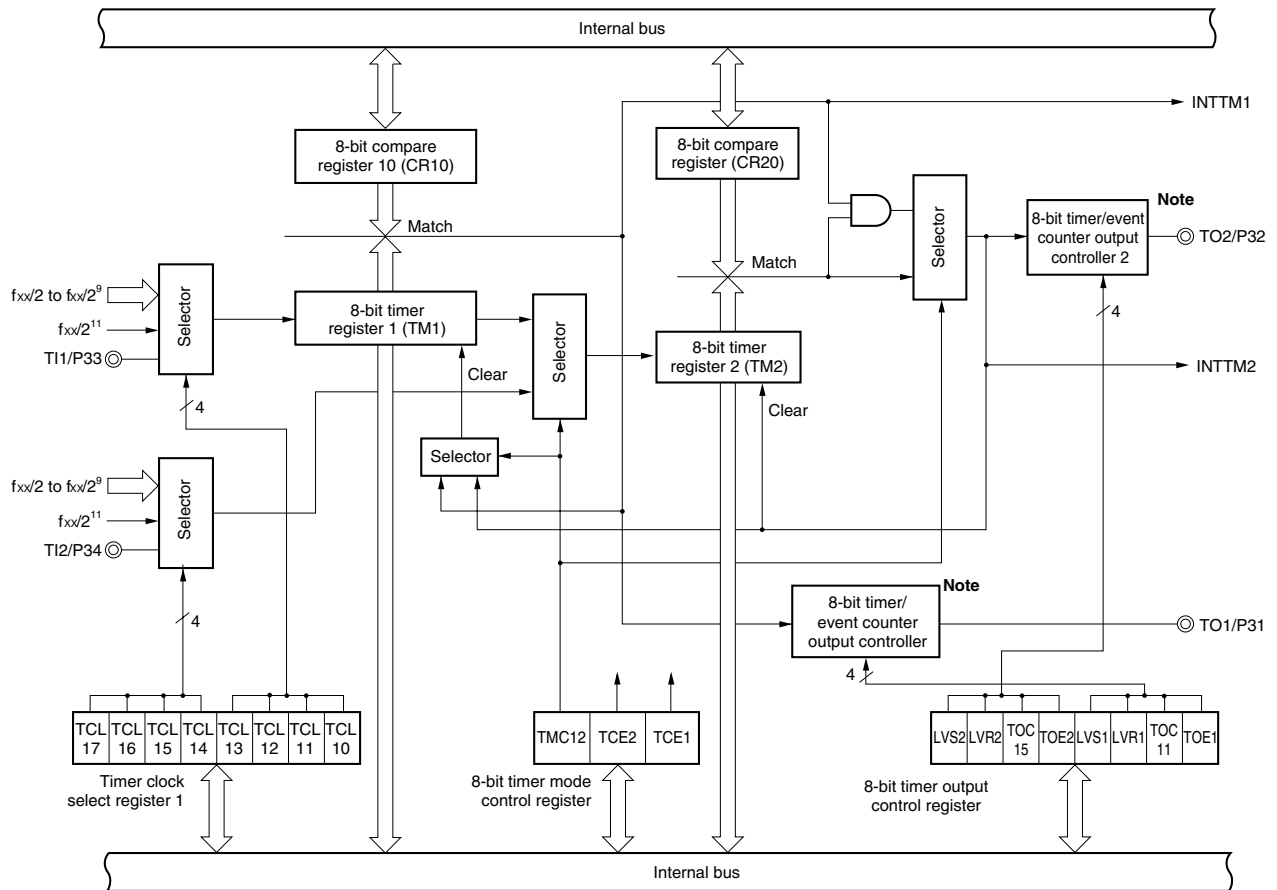
The 8-bit timer/event counter consists of the following hardware.

**Table 9-5. 8-Bit Timer/Event Counter Configuration**

Item	Configuration
Timer register	8 bits × 2 (TM1, TM2)
Register	Compare register: 8 bits × 2 (CR10, CR20)
Timer outputs	2 (TO1, TO2)
Control registers	Timer clock select register 1 (TCL1) 8-bit timer mode control register 1 (TMC1) 8-bit timer output control register (TOC1) Port mode register 3 (PM3) <sup>Note</sup>

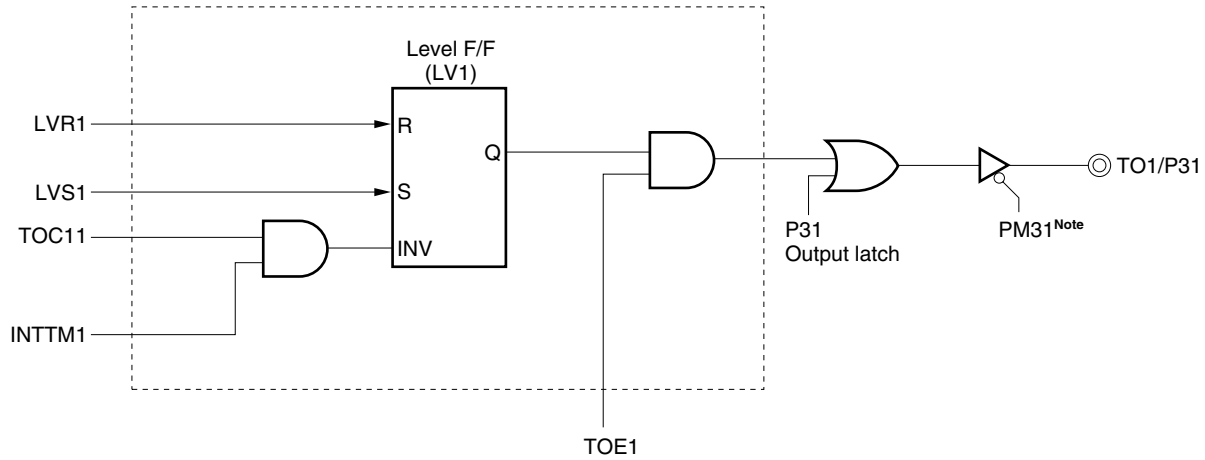
**Note** See **Figure 6-9 Block Diagram of P30 to P37**.

Figure 9-1. Block Diagram of 8-Bit Timer/Event Counter



**Note** See Figures 9-2 and 9-3 for details of 8-bit timer/event counter output controllers 1 and 2, respectively.

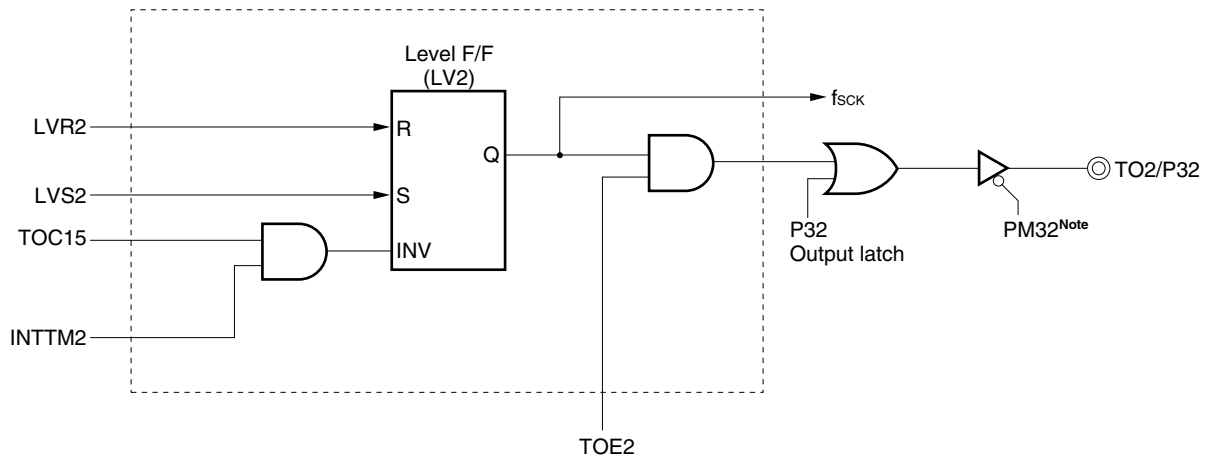
Figure 9-2. Block Diagram of 8-Bit Timer/Event Counter Output Controller 1



**Note** Bit 1 of port mode register 3 (PM3)

**Remark** The section in the broken lines is the output controller.

Figure 9-3. Block Diagram of 8-Bit Timer/Event Counter Output Controller 2



**Note** Bit 2 of port mode register 3 (PM3)

**Remarks** 1. The section in the broken lines is the output controller.  
 2. f<sub>sck</sub>: Serial clock frequency

**(1) Compare registers 10, 20 (CR10, CR20)**

CR10 and CR20 are 8-bit registers used to compare the value set to CR10 to the 8-bit timer register 1 (TM1) count value, and the value set to CR20 to the 8-bit timer register 2 (TM2) count value, and, if they match, generate an interrupt request (INTTM1 and INTTM2, respectively).

CR10 and CR20 are set with an 8-bit memory manipulation instruction. They cannot be set with a 16-bit memory manipulation instruction. When the compare register is used as 8-bit timer/event counter, between values 00H and FFH can be set. When the compare register is used as 16-bit timer/event counter, between values 0000H and FFFFH can be set.

$\overline{\text{RESET}}$  input makes CR10 and CR20 undefined.

- ★ **Cautions**
- 1. Before changing the set value of 8-bit compare registers 10 and 20 (CR10 and CR20) while the 16-bit timer/counter is being used, stop the operation of each of the 8-bit timer/event counters.**
  - 2. When the new values of CR10 and CR20 are less than the count values of the 8-bit timer registers (TM1 and TM2), TM1 and TM2 continue counting, overflow, and start counting again from 0. If the new values of CR10 and CR20 are less than the old values, therefore, it is necessary to restart the timers after changing the values of CR10 and CR20.**

**(2) 8-bit timer registers 1, 2 (TM1, TM2)**

TM1 and TM2 are 8-bit registers used to count count pulses.

When TM1 and TM2 are used in the 8-bit timer  $\times$  2-channel mode, they are read with an 8-bit memory manipulation instruction. When TM1 and TM2 are used as 16-bit timer  $\times$  1-channel mode, 16-bit timer register (TMS) is read with a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TM1 and TM2 to 00H.

### 9.3 8-Bit Timer/Event Counter Control Registers

The following four registers are used to control the 8-bit timer/event counter.

- Timer clock select register 1 (TCL1)
- 8-bit timer mode control register 1 (TMC1)
- 8-bit timer output control register (TOC1)
- Port mode register 3 (PM3)

#### (1) Timer clock select register 1 (TCL1)

This register sets the count clock of 8-bit timer registers 1 and 2.

TCL1 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TCL1 to 00H.

Figure 9-4. Format of Timer Clock Select Register 1

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL1	TCL17	TCL16	TCL15	TCL14	TCL13	TCL12	TCL11	TCL10	FF41H	00H	R/W

TCL13	TCL12	TCL11	TCL10	8-bit timer register 1 count clock selection		
				MCS = 1		MCS = 0
0	0	0	0	TI1 falling edge		
0	0	0	1	TI1 rising edge		
0	1	1	0	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
1	1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	1	1	1	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)
Other than above				Setting prohibited		

TCL17	TCL16	TCL15	TCL14	8-bit timer register 2 count clock selection		
				MCS = 1		MCS = 0
0	0	0	0	TI2 falling edge		
0	0	0	1	TI2 rising edge		
0	1	1	0	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
1	1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	1	1	1	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)
Other than above				Setting prohibited		

**Caution** When rewriting TCL1 to other data, stop the timer operation beforehand.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. TI1: 8-bit timer register 1 input pin
  4. TI2: 8-bit timer register 2 input pin
  5. MCS: Bit 0 of oscillation mode select register (OSMS)
  6. Values in parentheses apply to operation with  $f_x = 5.0$  MHz

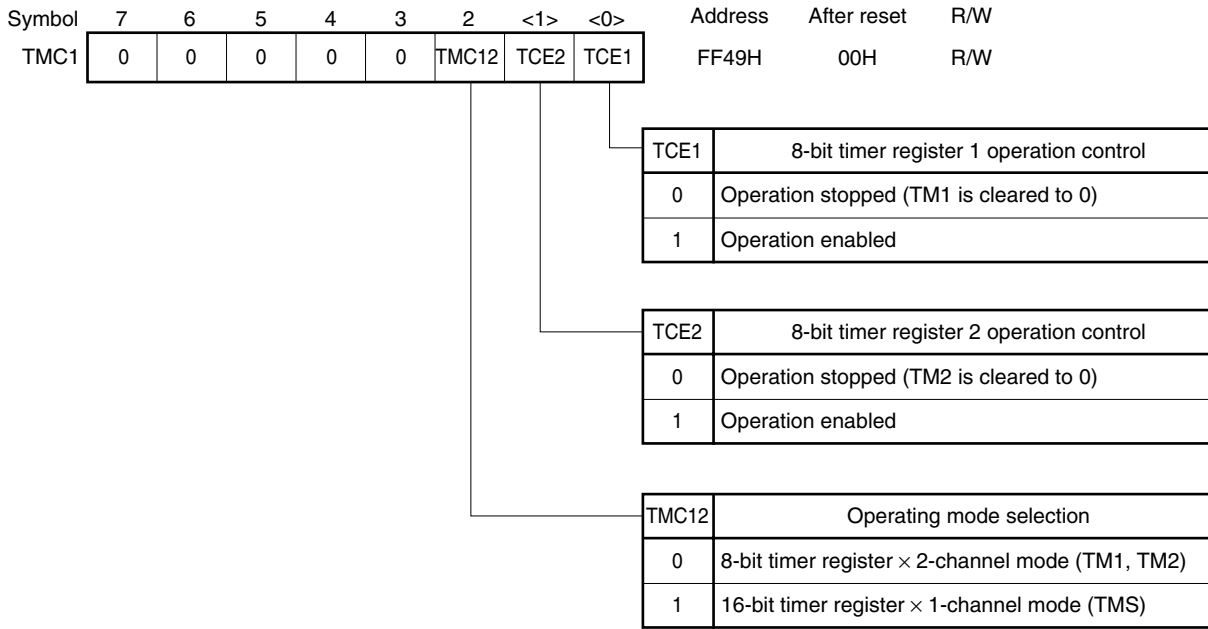
**(2) 8-bit timer mode control register (TMC1)**

This register enables/stops operation of 8-bit timer registers 1 and 2 and sets the operating mode of 8-bit timer registers 1 and 2.

TMC1 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears TMC1 to 00H.

**Figure 9-5. Format of 8-Bit Timer Mode Control Register 1**



- Cautions**
1. Switch the operating mode after stopping timer operation.
  2. When used as a 16-bit timer register (TMS), TCE1 should be used for operation enable/stop.



**(3) 8-bit timer output control register (TOC1)**

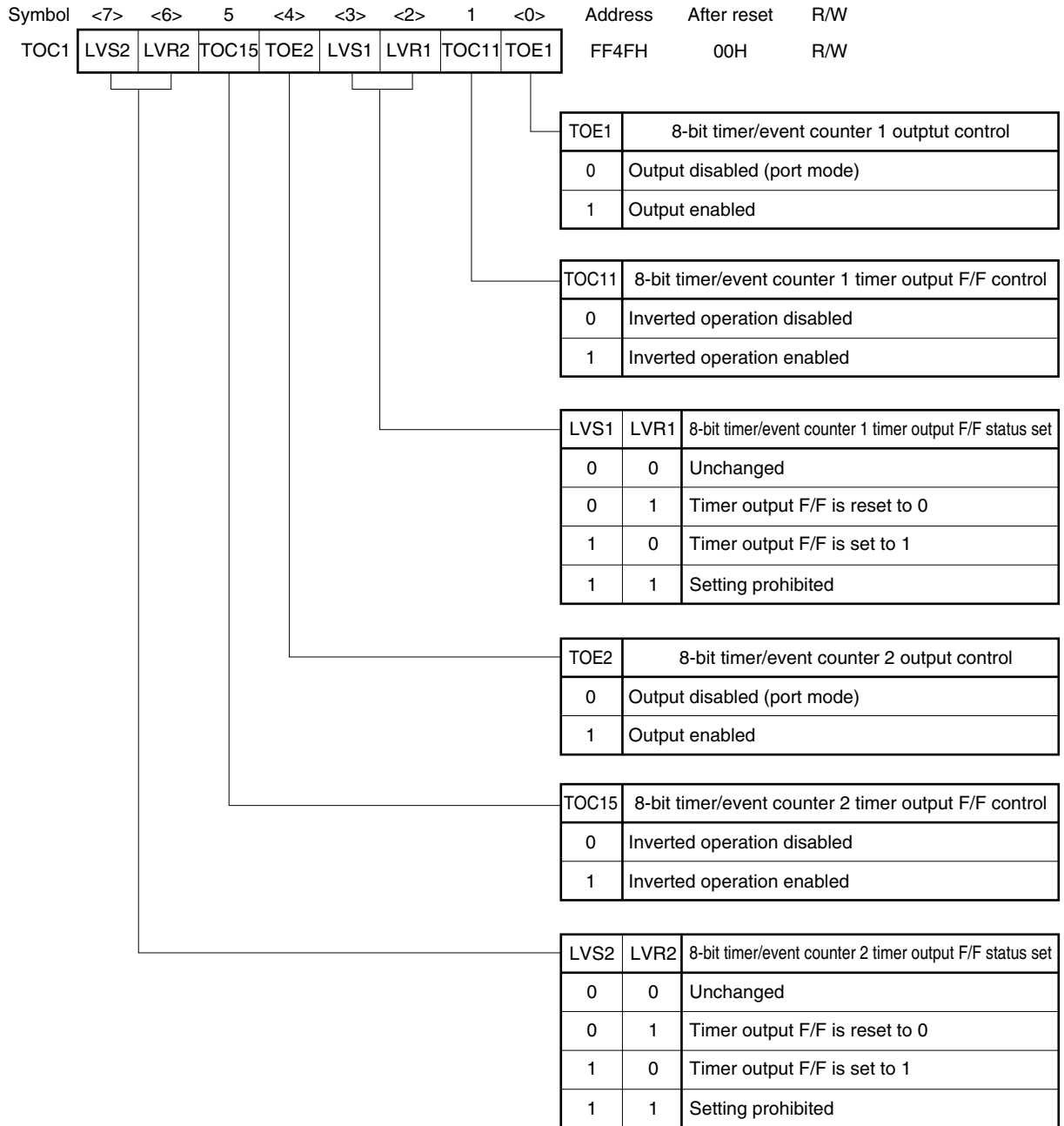
This register controls operation of 8-bit timer/event counter output controllers 1 and 2.

It sets/resets the R-S flip-flops (LV1 and LV2) and enables/disables inversion and 8-bit timer output of 8-bit timer registers 1 and 2.

TOC1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TOC1 to 00H.

**Figure 9-6. Format of 8-Bit Timer Output Control Register**



**Cautions 1. Be sure to set TOC1 after stopping timer operation.**

**2. After data setting, 0 is read from LVS1, LVS2, LVR1, and LVR2 when they are read.**

**(4) Port mode register 3 (PM3)**

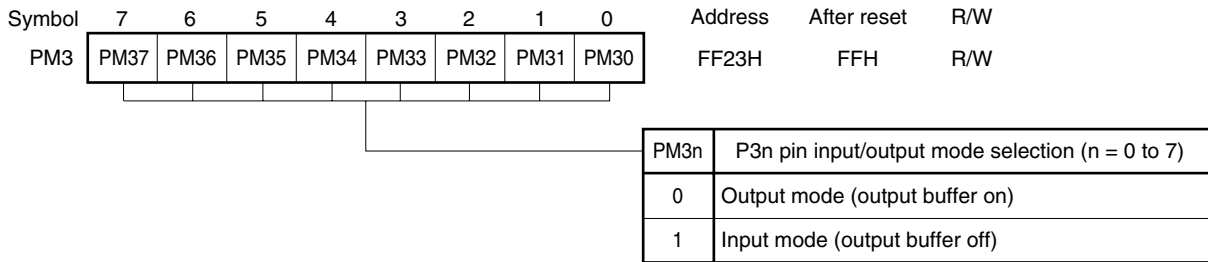
This register sets port 3 input/output in 1-bit units.

When using the P31/TO1 and P32/TO2 pins for timer output, set PM31, PM32, and the output latches of P31 and P32 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets PM3 to FFH.

**Figure 9-7. Format of Port Mode Register 3**



## 9.4 Operations of 8-Bit Timer/Event Counters 1 and 2

### 9.4.1 8-bit timer/event counter mode

#### (1) Interval timer operations

8-bit timer/event counters 1 and 2 operate as interval timers that generate interrupt requests repeatedly at intervals of the count value preset to 8-bit compare registers 10 and 20 (CR10 and CR20).

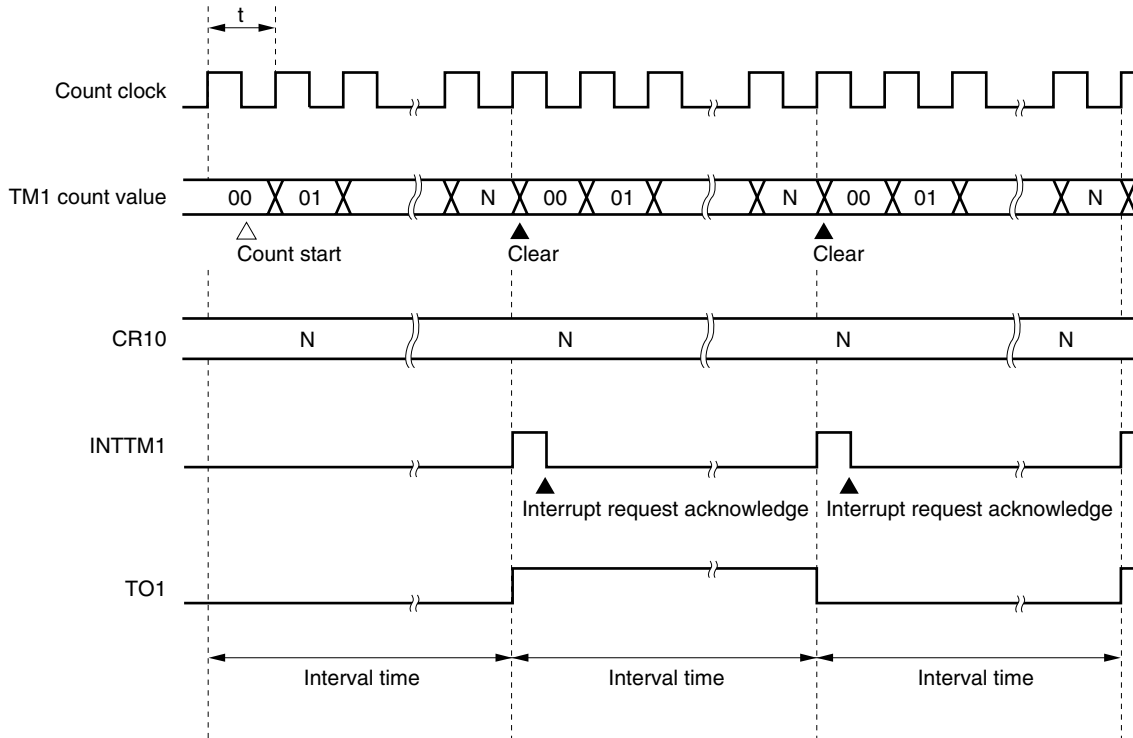
When the count values of 8-bit timer registers 1 and 2 (TM1 and TM2) match the values set to CR10 and CR20, counting continues with the TM1 and TM2 values cleared to 0 and the interrupt request signals (INTTM1 and INTTM2) are generated.

The count clock of TM1 can be selected using bits 0 to 3 (TCL10 to TCL13) of timer clock select register 1 (TCL1). The count clock of TM2 can be selected using bits 4 to 7 (TCL14 to TCL17) of timer clock select register 1 (TCL1).

For the operation when the value of the compare register is changed during a timer count operation, see 9.5

#### (3) Operation after compare register change during timer count operation.

Figure 9-8. Interval Timer Operation Timing



**Remark** Interval time =  $(N + 1) \times t$  : N = 00H to FFH

Table 9-6. Interval Time of 8-Bit Timer/Event Counter 1

TCL13	TCL12	TCL11	TCL10	Minimum Interval Time		Maximum Interval Time		Resolution	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	0	TI1 input cycle		$2^8 \times$ TI1 input cycle		TI1 input edge cycle	
0	0	0	1	TI1 input cycle		$2^8 \times$ TI1 input cycle		TI1 input edge cycle	
0	1	1	0	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
0	1	1	1	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
1	0	0	0	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
1	0	0	1	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
1	0	1	0	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
1	0	1	1	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
1	1	0	0	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
1	1	0	1	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
1	1	1	0	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
1	1	1	1	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)
Other than above				Setting prohibited					

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. TCL10 to TCL13: Bits 0 to 3 of timer clock select register 1 (TCL1)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

Table 9-7. Interval Time of 8-Bit Timer/Event Counter 2

TCL17	TCL16	TCL15	TCL14	Minimum Interval Time		Maximum Interval Time		Resolution	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	0	T12 input cycle		$2^8 \times$ T12 input cycle		T12 input edge cycle	
0	0	0	1	T12 input cycle		$2^8 \times$ T12 input cycle		T12 input edge cycle	
0	1	1	0	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
0	1	1	1	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
1	0	0	0	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
1	0	0	1	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
1	0	1	0	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
1	0	1	1	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
1	1	0	0	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
1	1	0	1	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
1	1	1	0	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
1	1	1	1	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)
Other than above				Setting prohibited					

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. TCL14 to TCL17: Bits 4 to 7 of timer clock select register 1 (TCL1)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz

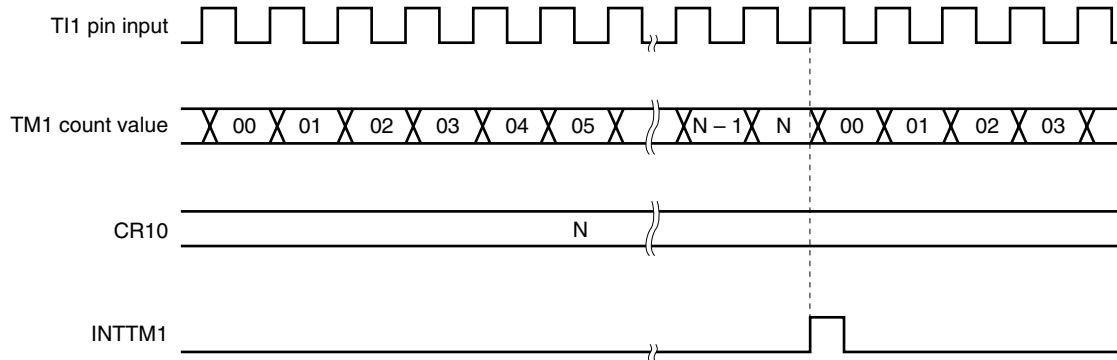
**(2) External event counter operation**

The external event counter counts the number of external clock pulses to be input to the T11/P33 and T12/P34 pins using 8-bit timer registers 1 and 2 (TM1 and TM2).

TM1 and TM2 are incremented each time the valid edge specified by the timer clock select register (TCL1) is input. Either the rising or falling edge can be selected.

When the TM1 and TM2 counted values match the values of 8-bit compare registers 10 and 20 (CR10 and CR20), TM1 and TM2 are cleared to 0 and the interrupt request signals (INTTM1 and INTTM2) are generated.

**Figure 9-9. External Event Counter Operation Timing (with Rising Edge Specified)**



**Remark** N = 00H to FFH

**(3) Square-wave output operation**

8-bit timer/event counters 1 and 2 output a square wave with any selected frequency at intervals specified by the value set in advance to 8-bit compare registers 10 and 20 (CR10 and CR20).

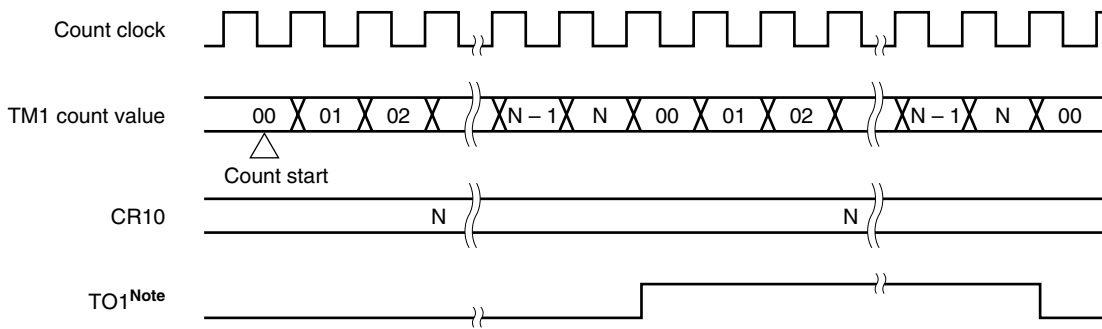
The TO1/P31 or TO2/P32 pin output status is reversed at intervals of the count value preset to CR10 or CR20 by setting bit 0 (TOE1) or bit 4 (TOE2) of the 8-bit timer output control register (TOC1) to 1. This enables a square wave with any selected frequency to be output.

**Table 9-8. Square-Wave Output Ranges of 8-Bit Timer/Event Counters 1 and 2**

Minimum Pulse Time		Maximum Pulse Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

Figure 9-10. Square-Wave Output Operation Timing



**Note** The initial value of the TO1 output can be set by bits 2 and 3 (LVS1 and LVR1) of the 8-bit timer output control register (TOC1).



**9.4.2 16-bit timer/event counter mode**

When bit 2 (TMC12) of the 8-bit timer mode control register (TMC1) is set to 1, the 16-bit timer/event counter mode is set.

In this mode, the count clock is selected by using bits 0 to 3 (TCL10 to TCL13) of the timer clock select register (TCL1), and the overflow signal of 8-bit timer/event counter 1 (TM1) is used as the count clock for 8-bit timer/event counter 2 (TM2).

The counting operation is enabled or disabled in this mode by using bit 0 (TCE1) of TMC1.

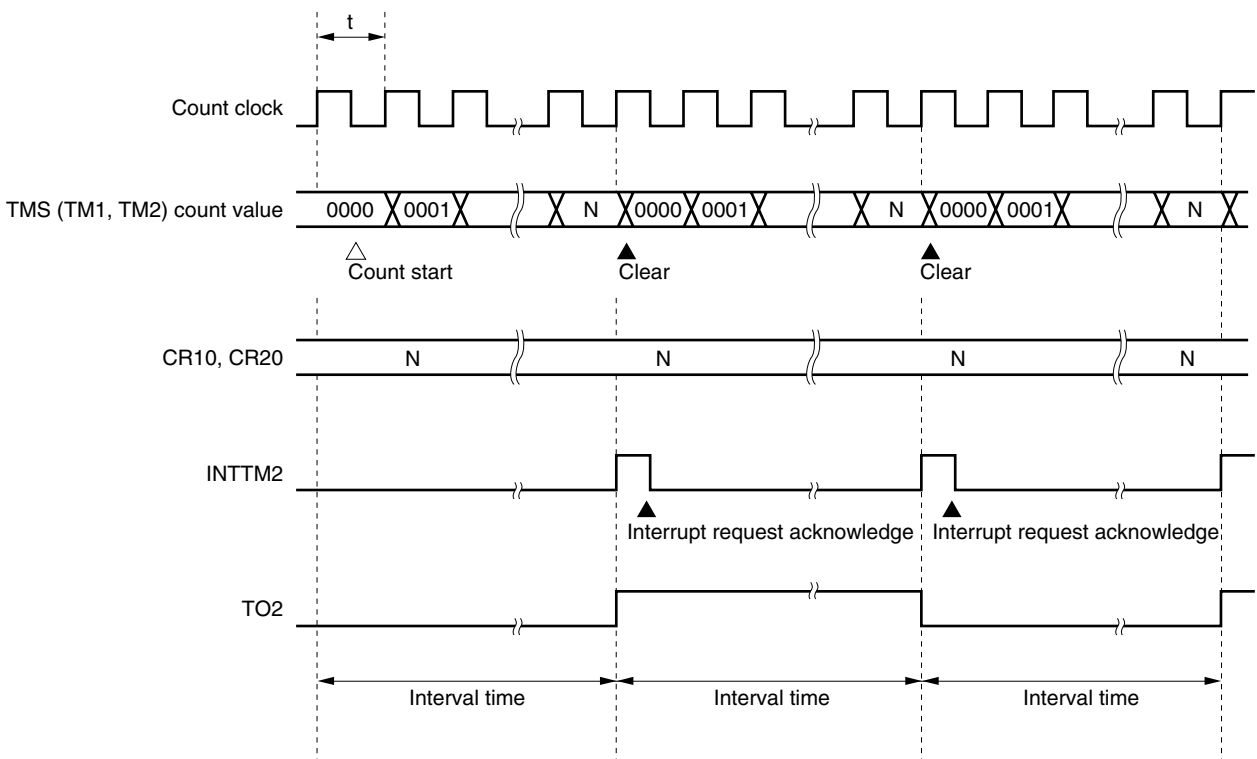
**(1) Operation as interval timer**

The 16-bit timer/event counter operates as an interval timer that repeatedly generates an interrupt request at intervals of the count values set in advance to the 2 channels of the 8-bit compare registers (CR10 and CR20). When setting a count value, assign the value of the higher 8 bits to CR20 and the value of the lower 8 bits to CR10. For the count values that can be set (interval time), see **Table 9-9**.

When the value of 8-bit timer register 1 (TM1) matches the value of CR10 and the value of 8-bit timer register 2 (TM2) matches the value of CR20, the values of TM1 and TM2 are cleared to 0, and at the same time, an interrupt request signal (INTTM2) is generated. For the operation timing of the interval timer, see **Figure 9-11**.

Select the count clock by using bits 0 to 3 (TCL10 to TCL13) of timer clock select register 1 (TCL1). The overflow signal of TM1 is used as the count clock for TM2.

**Figure 9-11. Interval Timer Operation Timing**



**Remark** Interval time = (N + 1) × t : N = 0000H to FFFFH

**Caution** Even if the 16-bit timer/event counter mode is used, when the TM1 count value matches the CR10 value, an interrupt request (INTTM1) is generated and the F/F of 8-bit timer/event counter output controller 1 is inverted. Thus, when using the 8-bit timer/event counter as a 16-bit interval timer, set the INTTM1 mask flag TMMK1 to 1 to disable INTTM1 acknowledgment. When reading the 16-bit timer register (TMS) count value, use a 16-bit memory manipulation instruction.

**Table 9-9. Interval Times When 2-Channel 8-Bit Timer/Event Counters (TM1 and TM2) Are Used as 16-Bit Timer/Event Counter**

TCL13	TCL12	TCL11	TCL10	Minimum Interval Time		Maximum Interval Time		Resolution	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	0	T11 input cycle		$2^8 \times$ T11 input cycle		T11 input edge cycle	
0	0	0	1	T11 input cycle		$2^8 \times$ T11 input cycle		T11 input edge cycle	
0	1	1	0	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
0	1	1	1	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
1	0	0	0	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
1	0	0	1	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
1	0	1	0	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
1	0	1	1	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^{23} \times 1/f_x$ (1.7 s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
1	1	0	0	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{23} \times 1/f_x$ (1.7 s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
1	1	0	1	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
1	1	1	0	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^{26} \times 1/f_x$ (13.4 s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
1	1	1	1	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{27} \times 1/f_x$ (26.8 s)	$2^{28} \times 1/f_x$ (53.7 s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)
Other than above				Setting prohibited					

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. TCL10 to TCL13: Bits 0 to 3 of timer clock select register 1 (TCL1)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

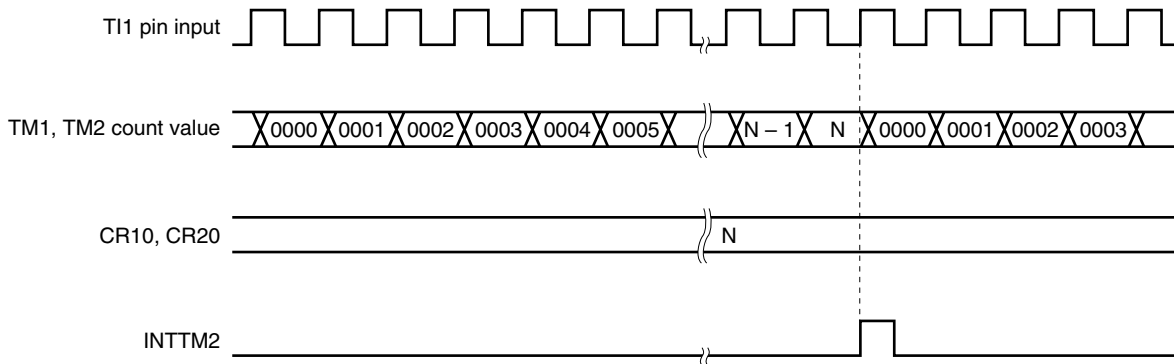
**(2) External event counter operations**

The external event counter counts the number of external clock pulses to be input to the TI1/P33 pin using 2-channel 8-bit timer registers 1 and 2 (TM1 and TM2).

TM1 is incremented each time the valid edge specified by timer clock select register 1 (TCL1) is input. When TM1 overflows as a result, TM2 is incremented with the overflow signal used as its count clock. Either the rising or falling edge can be selected.

When the TM1 and TM2 counted values match the values of 8-bit compare registers 10 and 20 (CR10 and CR20), TM1 and TM2 are cleared to 0 and the interrupt request signal (INTTM2) is generated.

**Figure 9-12. External Event Counter Operation Timing (with Rising Edge Specified)**



**Caution** Even if the 16-bit timer/event counter mode is used, when the TM1 count value matches the CR10 value, an interrupt request (INTTM1) is generated and the F/F of 8-bit timer/event counter output controller 1 is inverted. Thus, when using the 8-bit timer/event counter as a 16-bit interval timer, set the INTTM1 mask flag TMMK1 to 1 to disable INTTM1 acknowledgment. When reading the 16-bit timer register (TMS) count value, use a 16-bit memory manipulation instruction.

**(3) Square-wave output operation**

8-bit timer/event counters 1 and 2 output a square wave with any selected frequency at intervals specified by the value set in advance to 8-bit compare registers 10 and 20 (CR10 and CR20). To set a count value, set the value of the higher 8 bits to CR20, and the value of the lower 8 bits to CR10.

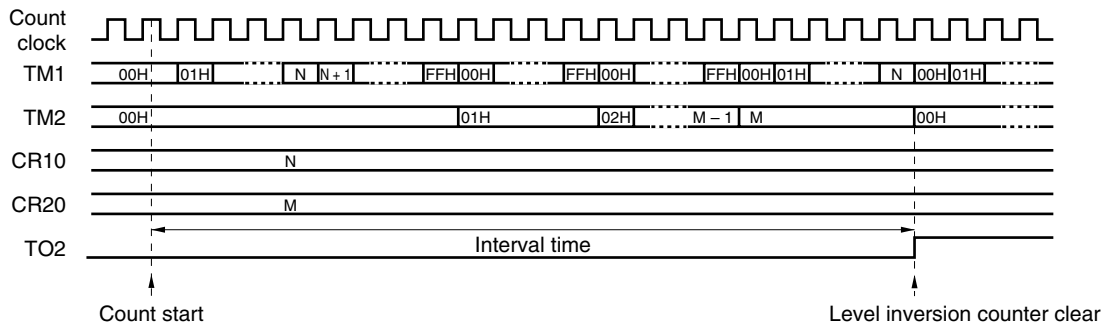
The TO2/P32 pin output status is reversed at intervals of the count value preset to CR10 and CR20 by setting bit 4 (TOE2) of the 8-bit timer output control register (TOC1) to 1. This enables a square wave with any selected frequency to be output.

**Table 9-10. Square-Wave Output Ranges When 2-Channel 8-Bit Timer/Event Counters (TM1 and TM2) Are Used as 16-Bit Timer/Event Counter**

Minimum Pulse Time		Maximum Pulse Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)	$2 \times 1/f_x$ (400 ns)	$2^2 \times 1/f_x$ (800 ns)
$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^{18} \times 1/f_x$ (52.4 ms)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^2 \times 1/f_x$ (800 ns)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)
$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^3 \times 1/f_x$ (1.6 $\mu$ s)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)
$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^{20} \times 1/f_x$ (209.7 ms)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^4 \times 1/f_x$ (3.2 $\mu$ s)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)
$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^{21} \times 1/f_x$ (419.4 ms)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^5 \times 1/f_x$ (6.4 $\mu$ s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)
$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^{22} \times 1/f_x$ (838.9 ms)	$2^{23} \times 1/f_x$ (1.7 s)	$2^6 \times 1/f_x$ (12.8 $\mu$ s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)
$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^{23} \times 1/f_x$ (1.7 s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^7 \times 1/f_x$ (25.6 $\mu$ s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)
$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{24} \times 1/f_x$ (3.4 s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^8 \times 1/f_x$ (51.2 $\mu$ s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)
$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)	$2^{25} \times 1/f_x$ (6.7 s)	$2^{26} \times 1/f_x$ (13.4 s)	$2^9 \times 1/f_x$ (102.4 $\mu$ s)	$2^{10} \times 1/f_x$ (204.8 $\mu$ s)
$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)	$2^{27} \times 1/f_x$ (26.8 s)	$2^{28} \times 1/f_x$ (53.7 s)	$2^{11} \times 1/f_x$ (409.6 $\mu$ s)	$2^{12} \times 1/f_x$ (819.2 $\mu$ s)

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)
  3. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

Figure 9-13. Square-Wave Output Operation Timing

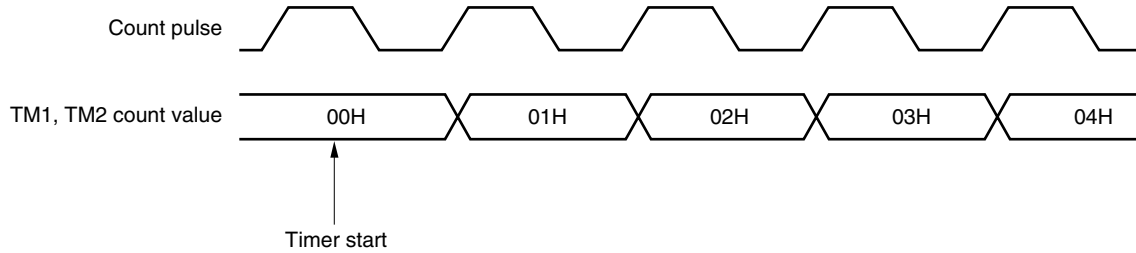


9.5 Cautions on 8-Bit Timer/Event Counters 1 and 2

(1) 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 8-bit timer registers 1 and 2 (TM1 and TM2) are started asynchronously to the count pulse.

Figure 9-14. Start Timing of 8-Bit Timer Registers 1 and 2

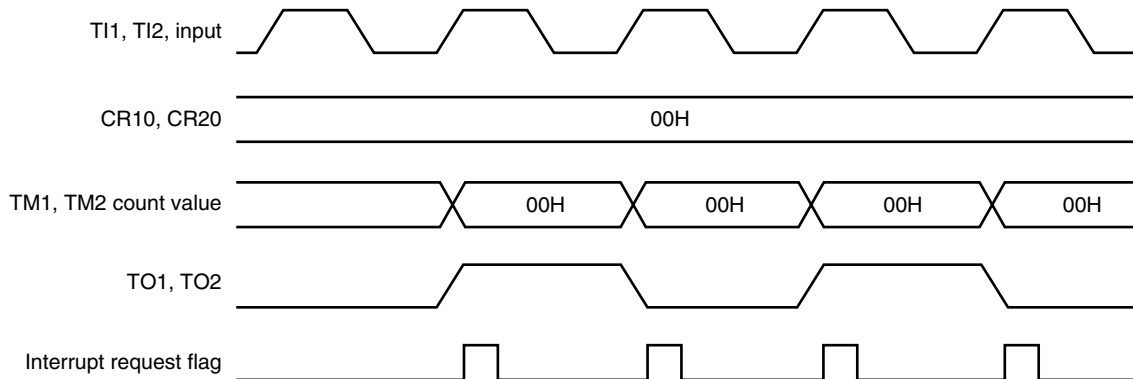


(2) 8-bit compare register 10 and 20 setting

8-bit compare registers 10 and 20 (CR10 and CR20) can be set to 00H. Thus, when these 8-bit compare registers are used as event counters, a one-pulse count operation can be carried out.

When the 8-bit compare register is used as 16-bit timer/event counter, write data to CR10 and CR20 after setting bit 0 (TCE1) of the 8-bit timer mode control register (TMC1) and stopping timer operation.

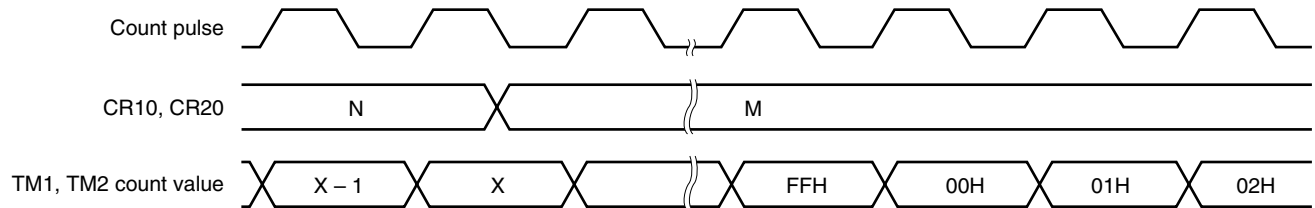
Figure 9-15. External Event Counter Operation Timing



**(3) Operation after compare register change during timer count operation**

If the values after 8-bit compare registers 10 and 20 (CR10 and CR20) are changed are smaller than those of the 8-bit timer registers (TM1 and TM2), TM1 and TM2 continue counting, overflow and then restart counting from 0. Thus, if the value after CR10 and CR20 change (M) is smaller than value before the change (N), it is necessary to restart the timer after changing CR10 and CR20.

**Figure 9-16. Timing After Compare Register Change During Timer Count Operation**



**Remark**  $N > X > M$

## CHAPTER 10 WATCH TIMER

### 10.1 Watch Timer Functions

The watch timer has the following functions.

- Watch timer
- Interval timer

The watch timer and the interval timer can be used simultaneously.

#### (1) Watch timer

When the 32.768 kHz subsystem clock is used, a flag (WTIF) is set at 0.5-second or 0.25-second intervals. When the 4.19 MHz (standard: 4.194304 MHz) main system clock is used, a flag (WTIF) is set at 0.5-second or 0.25-second intervals.

**Caution** 0.5-second intervals cannot be generated with the 5.0 MHz main system clock. Switch to the 32.768 kHz subsystem clock to generate 0.5-second intervals.

**Remark**  $f_{xx}$ : Watch timer clock frequency ( $f_x/2^7$  or  $f_{XT}$ )  
 $f_x$ : Main system clock oscillation frequency  
 $f_{XT}$ : Subsystem clock oscillation frequency

#### (2) Interval timer

Interrupt requests (INTTM3) are generated at the preset time interval.

**Table 10-1. Interval Timer Interval Time**

Interval Time	When Operated at $f_{xx} = 5.0 \text{ MHz}$	When Operated at $f_{xx} = 4.19 \text{ MHz}$	When Operated at $f_{XT} = 32.768 \text{ kHz}$
$2^4 \times 1/f_w$	410 $\mu\text{s}$	488 $\mu\text{s}$	488 $\mu\text{s}$
$2^5 \times 1/f_w$	819 $\mu\text{s}$	977 $\mu\text{s}$	977 $\mu\text{s}$
$2^6 \times 1/f_w$	1.64 ms	1.95 ms	1.95 ms
$2^7 \times 1/f_w$	3.28 ms	3.91 ms	3.91 ms
$2^8 \times 1/f_w$	6.55 ms	7.81 ms	7.81 ms
$2^9 \times 1/f_w$	13.1 ms	15.6 ms	15.6 ms

**Remark**  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )  
 $f_x$ : Main system clock oscillation frequency  
 $f_{XT}$ : Subsystem clock oscillation frequency  
 $f_w$ : Watch timer clock frequency ( $f_{xx}/2^7$  or  $f_{XT}$ )



## 10.2 Watch Timer Configuration

The watch timer consists of the following hardware.

**Table 10-2. Watch Timer Configuration**

Item	Configuration
Counter	5 bits × 1
Control registers	Timer clock select register 2 (TCL2) Watch timer mode control register (TMC2)

## 10.3 Watch Timer Control Registers

The following two registers are used to control the watch timer.

- Timer clock select register 2 (TCL2)
- Watch timer mode control register (TMC2)

### (1) Timer clock select register 2 (TCL2) (see Figure 10-2.)

This register sets the watch timer count clock.

TCL2 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TCL2 to 00H.

**Remark** Besides setting the watch timer count clock, TCL2 sets the watchdog timer count clock and buzzer output frequency.

Figure 10-1. Watch Timer Block Diagram

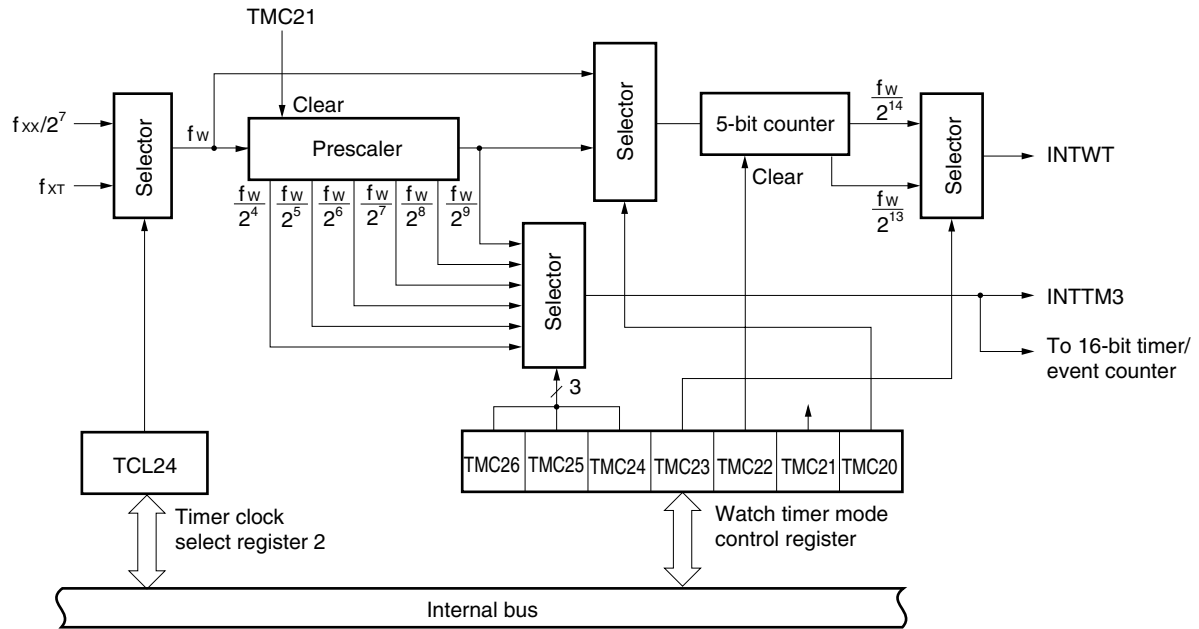


Figure 10-2. Format of Timer Clock Select Register 2

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog timer count clock selection (see CHAPTER 11 WATCHDOG TIMER)		
				MCS = 1	MCS = 0
0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	1	1	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)

TCL24	Watch timer count clock selection		
		MCS = 1	MCS = 0
0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	$f_{XT}$ (32.768 kHz)		

TCL27	TCL26	TCL25	Buzzer output frequency selection (see CHAPTER 13 BUZZER OUTPUT CONTROLLER)		
				MCS = 1	MCS = 0
0	×	×	Buzzer output disabled		
1	0	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	0	1	$f_{xx}/2^{10}$	$f_x/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)
1	1	0	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)
1	1	1	Setting prohibited		

★

**Caution** When changing the count clock, be sure to stop operation of the watch timer before rewriting TCL2 (stopping operation is not necessary when rewriting the same data).

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3.  $f_{XT}$ : Subsystem clock oscillation frequency
  4. ×: don't care
  5. MCS: Bit 0 of oscillation mode select register (OSMS)
  6. Values in parentheses apply to operation with  $f_x = 5.0$  MHz or  $f_{XT} = 32.768$  kHz.

**(2) Watch timer mode control register (TMC2)**

This register sets the watch timer operating mode, watch flag set time and prescaler interval time and enables/disables prescaler and 5-bit counter operations.

TMC2 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears TMC2 to 00H.

**Figure 10-3. Format of Watch Timer Mode Control Register**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TMC2	0	TMC26	TMC25	TMC24	TMC23	TMC22	TMC21	TMC20	FF4AH	00H	R/W

TMC20	Watch operating mode selection		
0	Normal operating mode (flag set at $f_w/2^{14}$ )		
1	Fast feed operating mode (flag set at $f_w/2^5$ )		

TMC21	Prescaler operation control		
0	Clear after operation stop		
1	Operation enable		

TMC22	5-bit counter operation control		
0	Clear after operation stop		
1	Operation enable		

TMC23	Watch flag set time selection		
	$f_{xx} = 5.0 \text{ MHz operation}$	$f_{xx} = 4.19 \text{ MHz operation}$	$f_{XT} = 32.768 \text{ kHz operation}$
	0	$2^{14}/f_w$ (0.4 sec)	$2^{14}/f_w$ (0.5 sec)
1	$2^{13}/f_w$ (0.2 sec)	$2^{13}/f_w$ (0.25 sec)	$2^{13}/f_w$ (0.25 sec)

TMC26	TMC25	TMC24	Prescaler interval time selection		
			$f_{xx} = 5.0 \text{ MHz operation}$	$f_{xx} = 4.19 \text{ MHz operation}$	$f_{XT} = 32.768 \text{ kHz operation}$
0	0	0	$2^4/f_w$ (410 $\mu\text{s}$ )	$2^4/f_w$ (488 $\mu\text{s}$ )	$2^4/f_w$ (488 $\mu\text{s}$ )
0	0	1	$2^5/f_w$ (819 $\mu\text{s}$ )	$2^5/f_w$ (977 $\mu\text{s}$ )	$2^5/f_w$ (977 $\mu\text{s}$ )
0	1	0	$2^6/f_w$ (1.64 ms)	$2^6/f_w$ (1.95 ms)	$2^6/f_w$ (1.95 ms)
0	1	1	$2^7/f_w$ (3.28 ms)	$2^7/f_w$ (3.91 ms)	$2^7/f_w$ (3.91 ms)
1	0	0	$2^8/f_w$ (6.55 ms)	$2^8/f_w$ (7.81 ms)	$2^8/f_w$ (7.81 ms)
1	0	1	$2^9/f_w$ (13.1 ms)	$2^9/f_w$ (15.6 ms)	$2^9/f_w$ (15.6 ms)
Other than above			Setting prohibited		

**Caution** When the watch timer is used, the prescaler should not be cleared frequently.

- Remarks**
1.  $f_w$ : Watch timer clock frequency ( $f_{xx}/2^7$  or  $f_{XT}$ )
  2.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  3.  $f_x$ : Main system clock oscillation frequency
  4.  $f_{XT}$ : Subsystem clock oscillation frequency

## 10.4 Watch Timer Operations

### 10.4.1 Watch timer operation

When the 32.768 kHz subsystem clock or 4.19 MHz main system clock is used, the timer operates as a watch timer with a 0.5-second or 0.25-second interval.

The watch timer sets the test input flag (WTIF) to 1 at a constant time interval. When WTMK = 0, the standby state (STOP mode/HALT mode) can be cleared by setting WTIF to 1.

When bit 2 (TMC22) of the watch timer mode control register (TMC2) is cleared to 0, the 5-bit counter is cleared and the count operation stops.

For simultaneous operation of the interval timer, zero-second start can be achieved by clearing TMC22 to 0 (maximum error: 26.2 ms when operated at  $f_{xx} = 5.0$  MHz).

### 10.4.2 Interval timer operation

The watch timer operates as interval timer which generates interrupt requests repeatedly at an interval of the preset count value.

The interval time can be selected using bits 4 to 6 (TMC24 to TMC26) of the watch timer mode control register (TMC2).

**Table 10-3. Interval Timer Interval Time**

TMC26	TMC25	TMC24	Interval Time	When Operated at $f_{xx} = 5.0$ MHz	When Operated at $f_{xx} = 4.19$ MHz	When Operated at $f_{XT} = 32.768$ kHz
0	0	0	$2^4 \times 1/f_w$	410 $\mu$ s	488 $\mu$ s	488 $\mu$ s
0	0	1	$2^5 \times 1/f_w$	819 $\mu$ s	977 $\mu$ s	977 $\mu$ s
0	1	0	$2^6 \times 1/f_w$	1.64 ms	1.95 ms	1.95 ms
0	1	1	$2^7 \times 1/f_w$	3.28 ms	3.91 ms	3.91 ms
1	0	0	$2^8 \times 1/f_w$	6.55 ms	7.81 ms	7.81 ms
1	0	1	$2^9 \times 1/f_w$	13.1 ms	15.6 ms	15.6 ms
Other than above			Setting prohibited			

**Remark**  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )  
 $f_x$ : Main system clock oscillation frequency  
 $f_{XT}$ : Subsystem clock oscillation frequency  
 $f_w$ : Watch timer clock frequency ( $f_{xx}/2^7$  or  $f_{XT}$ )  
TMC24 to TMC26: Bits 4 to 6 of watch timer mode control register (TMC2)

## CHAPTER 11 WATCHDOG TIMER

### 11.1 Watchdog Timer Functions

The watchdog timer has the following functions.

- Watchdog timer
- Interval timer

**Caution** Select the watchdog timer mode or the interval timer mode using the watchdog timer mode register (WDTM) (the watchdog timer and interval timer cannot be used at the same time).

#### (1) Watchdog timer mode

An inadvertent program loop is detected. Upon detection of the program loop, a non-maskable interrupt request or **RESET** can be generated.

**Table 11-1. Watchdog Timer Program Loop Detection Times**

Runaway Detection Time	MCS = 1	MCS = 0
$2^{11} \times 1/f_{xx}$	$2^{11} \times 1/f_x$ (410 $\mu$ s)	$2^{12} \times 1/f_x$ (819 $\mu$ s)
$2^{12} \times 1/f_{xx}$	$2^{12} \times 1/f_x$ (819 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)
$2^{13} \times 1/f_{xx}$	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)
$2^{14} \times 1/f_{xx}$	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)
$2^{15} \times 1/f_{xx}$	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)
$2^{16} \times 1/f_{xx}$	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)
$2^{17} \times 1/f_{xx}$	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)
$2^{19} \times 1/f_{xx}$	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

**(2) Interval timer mode**

Interrupt requests are generated at the preset time intervals.

**Table 11-2. Interval Times**

Interval Time	MCS = 1	MCS = 0
$2^{11} \times 1/f_{xx}$	$2^{11} \times 1/f_x$ (410 $\mu$ s)	$2^{12} \times 1/f_x$ (819 $\mu$ s)
$2^{12} \times 1/f_{xx}$	$2^{12} \times 1/f_x$ (819 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)
$2^{13} \times 1/f_{xx}$	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)
$2^{14} \times 1/f_{xx}$	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)
$2^{15} \times 1/f_{xx}$	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)
$2^{16} \times 1/f_{xx}$	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)
$2^{17} \times 1/f_{xx}$	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)
$2^{19} \times 1/f_{xx}$	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

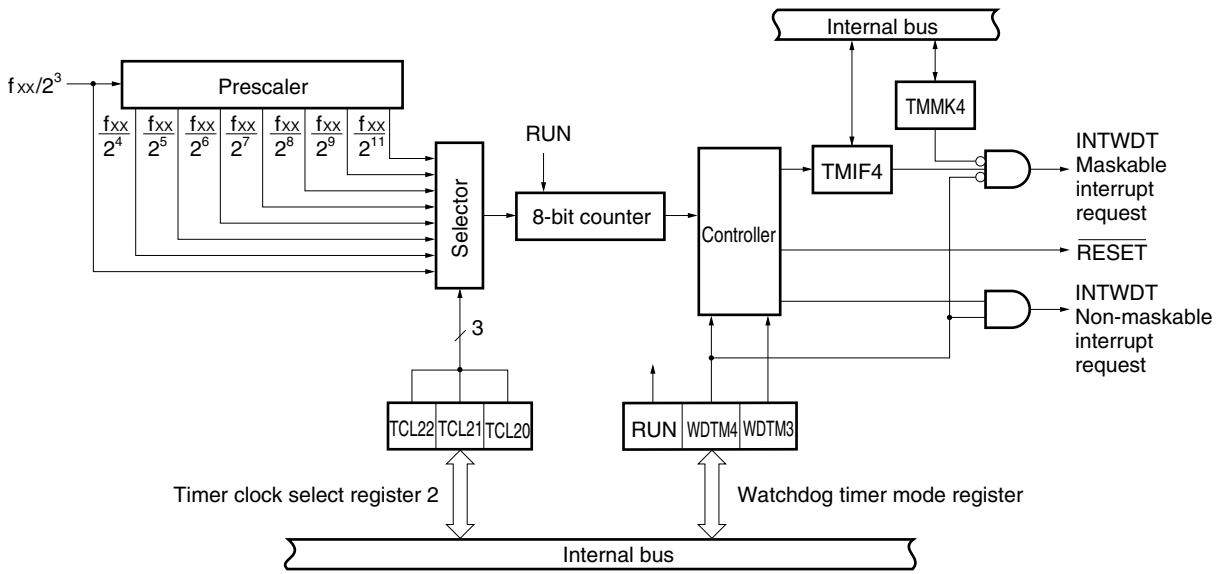
## 11.2 Watchdog Timer Configuration

The watchdog timer consists of the following hardware.

**Table 11-3. Watchdog Timer Configuration**

Item	Configuration
Control registers	Timer clock select register 2 (TCL2) Watchdog timer mode register (WDTM)

**Figure 11-1. Watchdog Timer Block Diagram**





### 11.3 Watchdog Timer Control Registers

The following two registers are used to control the watchdog timer.

- Timer clock select register 2 (TCL2)
- Watchdog timer mode register (WDTM)

#### (1) Timer clock select register 2 (TCL2)

This register sets the watchdog timer count clock.

TCL2 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TCL2 to 00H.

**Remark** Besides setting the watchdog timer count clock, TCL2 sets the watch timer count clock and buzzer output frequency.

Figure 11-2. Format of Timer Clock Select Register 2

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog timer count clock selection		
				MCS = 1	MCS = 0
0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	1	1	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)

TCL24	Watch timer count clock selection (see CHAPTER 10 WATCH TIMER)		
		MCS = 1	MCS = 0
0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	$f_{XT}$ (32.768 kHz)		

TCL27	TCL26	TCL25	Buzzer output frequency selection (see CHAPTER 13 BUZZER OUTPUT CONTROLLER)		
				MCS = 1	MCS = 0
0	×	×	Buzzer output disable		
1	0	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	0	1	$f_{xx}/2^{10}$	$f_x/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)
1	1	0	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)
1	1	1	Setting prohibited		

★ **Caution** Changing the count clock (rewriting TCL20 to TCL22) after watchdog timer operation has started is prohibited.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3.  $f_{XT}$ : Subsystem clock oscillation frequency
  4. ×: don't care
  5. MCS: Bit 0 of oscillation mode select register (OSMS)
  6. Values in parentheses apply to operation with  $f_x = 5.0$  MHz or  $f_{XT} = 32.768$  kHz.

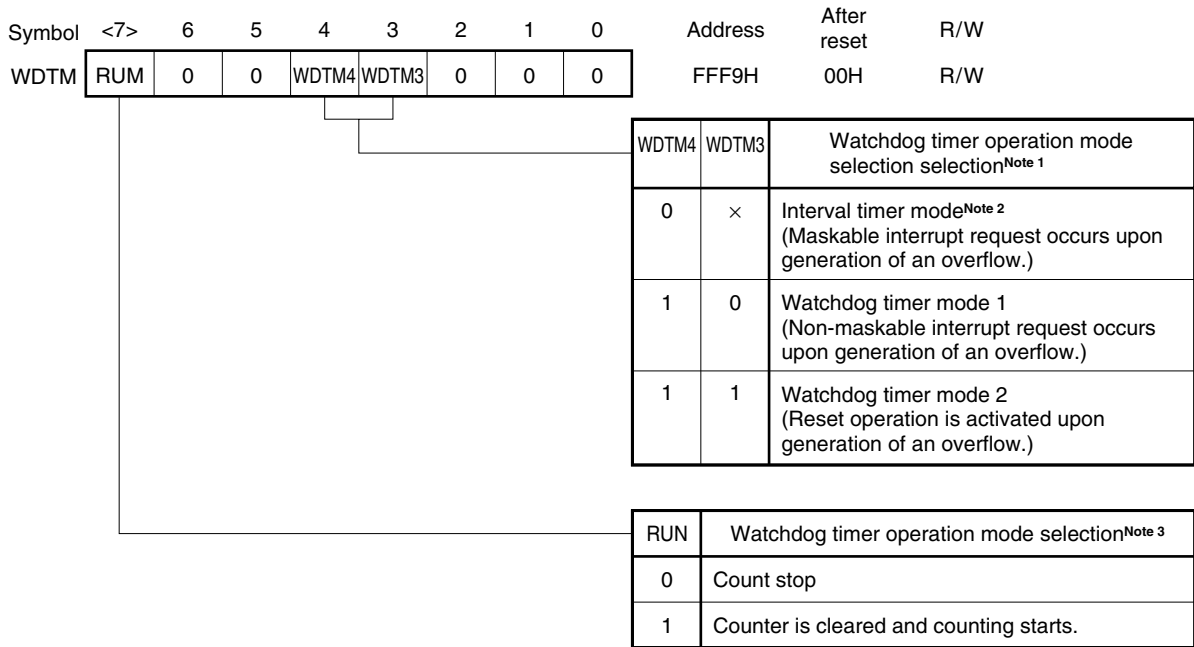
**(2) Watchdog timer mode register (WDTM)**

This register sets the watchdog timer operating mode and enables/disables counting.

WDTM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears WDTM to 00H.

**Figure 11-3. Format of Watchdog Timer Mode Register**



- Notes**
- Once set to 1, WDTM3 and WDTM4 cannot be cleared to 0 by software.
  - The watchdog timer starts operating as an interval timer as soon as RUN has been set to 1.
  - Once set to 1, RUN cannot be cleared to 0 by software.  
Thus, once counting starts, it can only be stopped by  $\overline{\text{RESET}}$  input.

- Cautions**
- When RUN is set to 1 so that the watchdog timer is cleared, the actual overflow time is up to 0.5% shorter than the time set by timer clock select register 2 (TCL2).
  - To use watchdog timer modes 1 and 2, make sure that the interrupt request flag (TMIF4) is 0, and then set WDTM4 to 1.  
If WDTM4 is set to 1 when TMIF4 is 1, the non-maskable interrupt request occurs, regardless of the contents of WDTM3.

**Remark** ×: don't care

## 11.4 Watchdog Timer Operations

### 11.4.1 Watchdog timer operation

When bit 4 (WDTM4) of the watchdog timer mode register (WDTM) is set to 1, the watchdog timer operates to detect an inadvertent program loop.

The watchdog timer count clock (program loop detection time interval) can be selected using bits 0 to 2 (TCL20 to TCL22) of timer clock select register 2 (TCL2).

The watchdog timer starts by setting bit 7 (RUN) of WDTM to 1. After the watchdog timer is started, set RUN to 1 within the set loop detection time interval. The watchdog timer can be cleared and counting is started by setting RUN to 1. If RUN is not set to 1 and the program loop detection time has elapsed, system reset or a non-maskable interrupt request is generated according to the value of WDTM bit 3 (WDTM3).

By setting RUN to 1, the watchdog timer can be cleared.

The watchdog timer continues operating in the HALT mode but it stops in the STOP mode. Thus, set RUN to 1 before the STOP mode is set, clear the watchdog timer and then execute the STOP instruction.

**Cautions 1. The actual loop detection time may be shorter than the set time by a maximum of 0.5%.**

**2. When the subsystem clock is selected for the CPU clock, the watchdog timer count operation is stopped.**

**Table 11-4. Watchdog Timer Program Loop Detection Time**

TCL22	TCL21	TCL20	Runaway Detection Time	MCS = 1	MCS = 0
0	0	0	$2^{11} \times 1/f_{xx}$	$2^{11} \times 1/f_x$ (410 $\mu$ s)	$2^{12} \times 1/f_x$ (819 $\mu$ s)
0	0	1	$2^{12} \times 1/f_{xx}$	$2^{12} \times 1/f_x$ (819 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)
0	1	0	$2^{13} \times 1/f_{xx}$	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)
0	1	1	$2^{14} \times 1/f_{xx}$	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)
1	0	0	$2^{15} \times 1/f_{xx}$	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)
1	0	1	$2^{16} \times 1/f_{xx}$	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)
1	1	0	$2^{17} \times 1/f_{xx}$	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)
1	1	1	$2^{19} \times 1/f_{xx}$	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. TCL20 to TCL22: Bits 0 to 2 of timer clock select register 2 (TCL2)
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

**11.4.2 Interval timer operation**

The watchdog timer operates as an interval timer which generates interrupt requests repeatedly at an interval of the preset count value when bit 4 (WDTM4) of the watchdog timer mode register (WDTM) is cleared to 0.

The count clock (interval time) can be selected by bits 0 to 2 (TCL20 to TCL22) of timer clock select register 2 (TCL2). By setting bit 7 (RUN) of WDTM to 1, the watchdog timer starts operating as an interval timer.

When the watchdog timer operates as interval timer, the interrupt mask flag (TMMK4) and priority specification flag (TMPR4) are validated and a maskable interrupt request (INTWDT) can be generated. Among the maskable interrupt requests, the INTWDT default has the highest priority.

The interval timer continues operating in the HALT mode but it stops in STOP mode. Thus, set bit 7 (RUN) of WDTM to 1 before the STOP mode is set, clear the interval timer and then execute the STOP instruction.

- Cautions**
1. Once bit 4 (WDTM4) of WDTM is set to 1 (with the watchdog timer mode selected), the interval timer mode is not set unless RESET input is applied.
  2. The interval time just after setting by WDTM may be shorter than the set time by a maximum of 0.5%.
  3. When the subsystem clock is selected for the CPU clock, the watchdog timer count operation is stopped.

**Table 11-5. Interval Timer Interval Time**

TCL22	TCL21	TCL20	Interval Time	MCS = 1	MCS = 0
0	0	0	$2^{11} \times 1/f_{xx}$	$2^{11} \times 1/f_x$ (410 $\mu$ s)	$2^{12} \times 1/f_x$ (819 $\mu$ s)
0	0	1	$2^{12} \times 1/f_{xx}$	$2^{12} \times 1/f_x$ (819 $\mu$ s)	$2^{13} \times 1/f_x$ (1.64 ms)
0	1	0	$2^{13} \times 1/f_{xx}$	$2^{13} \times 1/f_x$ (1.64 ms)	$2^{14} \times 1/f_x$ (3.28 ms)
0	1	1	$2^{14} \times 1/f_{xx}$	$2^{14} \times 1/f_x$ (3.28 ms)	$2^{15} \times 1/f_x$ (6.55 ms)
1	0	0	$2^{15} \times 1/f_{xx}$	$2^{15} \times 1/f_x$ (6.55 ms)	$2^{16} \times 1/f_x$ (13.1 ms)
1	0	1	$2^{16} \times 1/f_{xx}$	$2^{16} \times 1/f_x$ (13.1 ms)	$2^{17} \times 1/f_x$ (26.2 ms)
1	1	0	$2^{17} \times 1/f_{xx}$	$2^{17} \times 1/f_x$ (26.2 ms)	$2^{18} \times 1/f_x$ (52.4 ms)
1	1	1	$2^{19} \times 1/f_{xx}$	$2^{19} \times 1/f_x$ (104.9 ms)	$2^{20} \times 1/f_x$ (209.7 ms)

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. TCL20 to TCL22: Bits 0 to 2 of timer clock select register 2 (TCL2)
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

## CHAPTER 12 CLOCK OUTPUT CONTROLLER

### 12.1 Clock Output Controller Functions

The clock output controller is used for carrier output during remote controlled transmission and clock output for supply to peripheral LSI devices. The clock selected by timer clock select register 0 (TCL0) is output from the PCL/P35 pin.

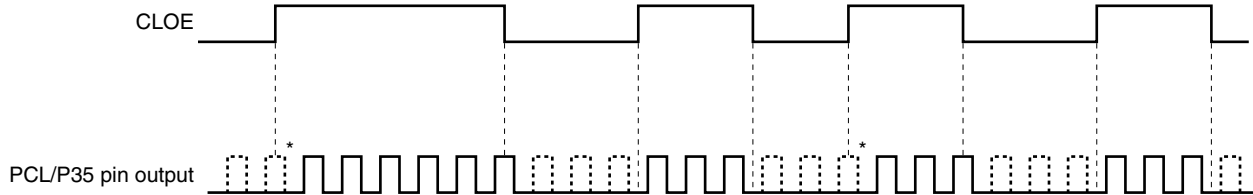
Follow the procedure below to output clock pulses.

- (1) Select the clock pulse output frequency (with clock pulse output disabled) using bits 0 to 3 (TCL00 to TCL03) of TCL0.
- (2) Set the P35 output latch to 0.
- (3) Set bit 5 (PM35) of port mode register 3 (PM3) to 0 (set to output mode).
- (4) Set bit 7 (CLOE) of timer clock select register 0 (TCL0) to 1.

**Caution** Clock output cannot be used when the P35 output latch is set to 1.

**Remark** When clock output enable/disable is switched, the clock output controller does not output pulses with small widths (see the portions marked with \* in **Figure 12-1**).

**Figure 12-1. Remote Controlled Output Application Example**



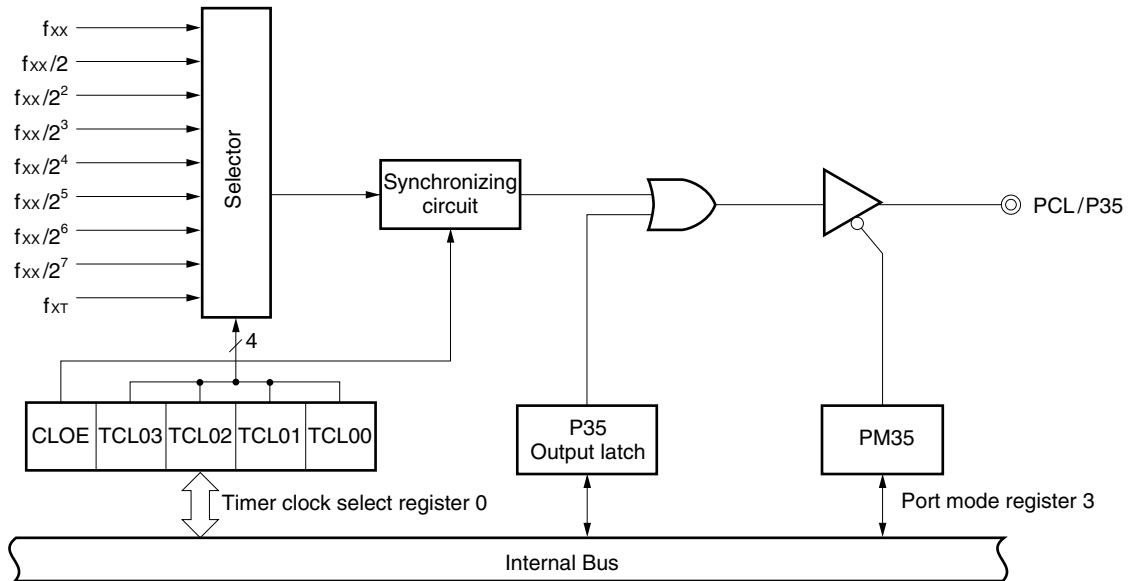
## 12.2 Clock Output Controller Configuration

The clock output controller consists of the following hardware.

**Table 12-1. Clock Output Controller Configuration**

Item	Configuration
Control registers	Timer clock select register 0 (TCL0) Port mode register 3 (PM3)

**Figure 12-2. Clock Output Controller Block Diagram**



## 12.3 Clock Output Function Control Registers

The following two registers are used to control the clock output function.

- Timer clock select register 0 (TCL0)
- Port mode register 3 (PM3)

### (1) Timer clock select register 0 (TCL0)

This register sets the PCL output clock.

TCL0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TCL0 to 00H.

**Remark** Besides setting the PCL output clock, TCL0 sets the 16-bit timer register count clock.

Figure 12-3. Format of Timer Clock Select Register 0

Symbol	<7>	6	5	4	3	2	1	0	Address	After reset	R/W
TCL0	CLOE	TCL06	TCL05	TCL04	TCL03	TCL02	TCL01	TCL00	FF40H	00H	R/W

TCL03	TCL02	TCL01	TCL00	PCL output clock selection		
				MCS = 1		MCS = 0
0	0	0	0	$f_{XT}$ (32.768 kHz)		
0	1	0	1	$f_{xx}$	$f_x$ (5.0 MHz)	$f_x/2$ (2.5 MHz)
0	1	1	0	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
Other than above				Setting prohibited		

TCL06	TCL05	TCL04	16-bit timer register count clock selection		
			MCS = 1		MCS = 0
0	0	0	TI00 (valid edge specifiable)		
0	0	1	$2f_{xx}$	Setting prohibited	$f_x$ (5.0 MHz)
0	1	0	$f_{xx}$	$f_x$ (5.0 MHz)	$f_x/2$ (2.5 MHz)
0	1	1	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)
1	0	0	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	1	1	Watch timer output (INTTM3)		
Other than above			Setting prohibited		

CLOE	PCL output control
0	Output disabled
1	Output enabled

- Cautions**
1. The TI00/P00/INTP0 pin valid edge is set by external interrupt mode register 0 (INTM0), and the sampling clock frequency is selected by the sampling clock select register (SCS).
  2. When enabling PCL output, set TCL00 to TCL03, then set CLOE to 1 with a 1-bit memory manipulation instruction.
  3. When reading the count value when TI00 has been specified as the TM0 count clock, the value should be read from TM0, not from the 16-bit capture/compare register (CR01).
  4. When rewriting TCL0 to other data, stop the clock operation beforehand.



- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3.  $f_{XT}$ : Subsystem clock oscillation frequency
  4. TI00: 16-bit timer/event counter input pin
  5. TM0: 16-bit timer register
  6. MCS: Bit 0 of oscillation mode select register (OSMS)
  7. Values in parentheses apply to operation with  $f_x = 5.0$  MHz or  $f_{XT} = 32.768$  kHz.

**(2) Port mode register 3 (PM3)**

This register sets port 3 input/output in 1-bit units.

When using the P35/PCL pin for clock output, set PM35 and the output latch of P35 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets PM3 to FFH.

**Figure 12-4. Format of Port Mode Register 3**



## CHAPTER 13 BUZZER OUTPUT CONTROLLER

### 13.1 Buzzer Output Controller Functions

The buzzer output controller outputs 1.2 kHz, 2.4 kHz, 4.9 kHz, or 9.8 kHz frequency square waves. The buzzer frequency selected by timer clock select register 2 (TCL2) is output from the BUZ/P36 pin.

Follow the procedure below to output the buzzer frequency.

- (1) Select the buzzer output frequency using bits 5 to 7 (TCL25 to TCL27) of TCL2.
- (2) Set the P36 output latch to 0.
- (3) Set bit 6 (PM36) of port mode register 3 (PM3) to 0 (set to output mode).

**Caution** Buzzer output cannot be used when the P36 output latch is set to 1.

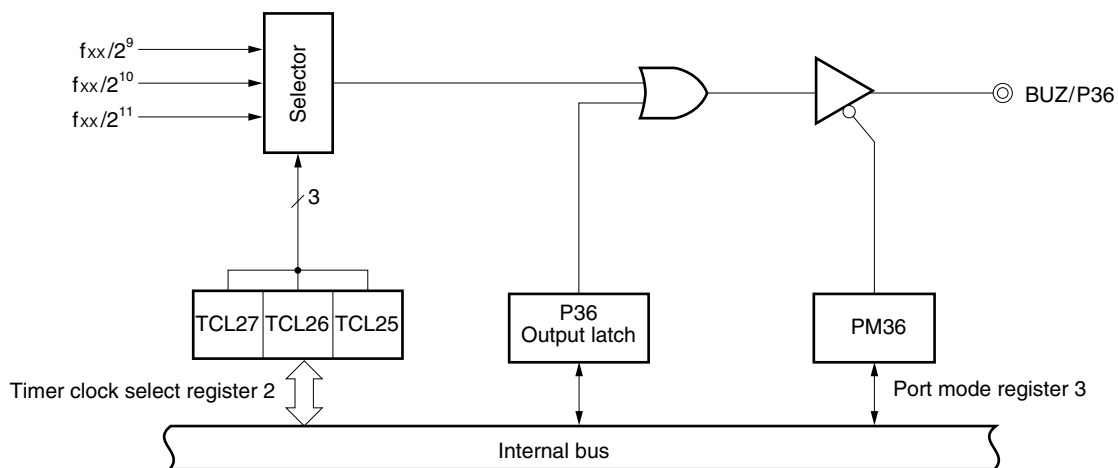
### 13.2 Buzzer Output Controller Configuration

The buzzer output controller consists of the following hardware.

**Table 13-1. Buzzer Output Controller Configuration**

Item	Configuration
Control registers	Timer clock select register 2 (TCL2) Port mode register 3 (PM3)

**Figure 13-1. Buzzer Output Controller Block Diagram**



### 13.3 Buzzer Output Function Control Registers

The following two registers are used to control the buzzer output function.

- Timer clock select register 2 (TCL2)
- Port mode register 3 (PM3)

#### (1) Timer clock select register 2 (TCL2)

This register sets the buzzer output frequency.

TCL2 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears TCL2 to 00H.

**Remark** Besides setting the buzzer output frequency, TCL2 sets the watch timer count clock and the watchdog timer count clock.

Figure 13-2. Format of Timer Clock Select Register 2

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog timer count clock selection (see CHAPTER 11 WATCHDOG TIMER)		
				MCS = 1	MCS = 0
0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	1	1	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)

TCL24	Watch timer count clock selection (see CHAPTER 10 WATCH TIMER)		
		MCS = 1	MCS = 0
0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	$f_{xt}$ (32.768 kHz)		

TCL27	TCL26	TCL25	Buzzer output frequency selection		
				MCS = 1	MCS = 0
0	×	×	Buzzer output disabled		
1	0	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)
1	0	1	$f_{xx}/2^{10}$	$f_x/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)
1	1	0	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.4 kHz)	$f_x/2^{12}$ (1.2 kHz)
1	1	1	Setting prohibited		

★ **Cautions** 1. Be sure to stop operation of the watch timer or buzzer to be changed before rewriting TCL2 (stop operation is not necessary when rewriting the same data). The operation is stopped by the following methods.

- Buzzer output: Input 0 to bit 7 of TCL2 (TCL27)
- Watch timer: Input 0 to bit 2 (TMC22) of watch timer mode control register 2 (TMC2)

★ 2. Changing the count clock (rewriting TCL20 to TCL22) after watchdog timer operation has started is prohibited.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3.  $f_{xt}$ : Subsystem clock oscillation frequency
  4. ×: don't care
  5. MCS: Bit 0 of oscillation mode select register (OSMS)
  6. Values in parentheses apply to operation with  $f_x = 5.0$  MHz or  $f_{xt} = 32.768$  kHz.

**(2) Port mode register 3 (PM3)**

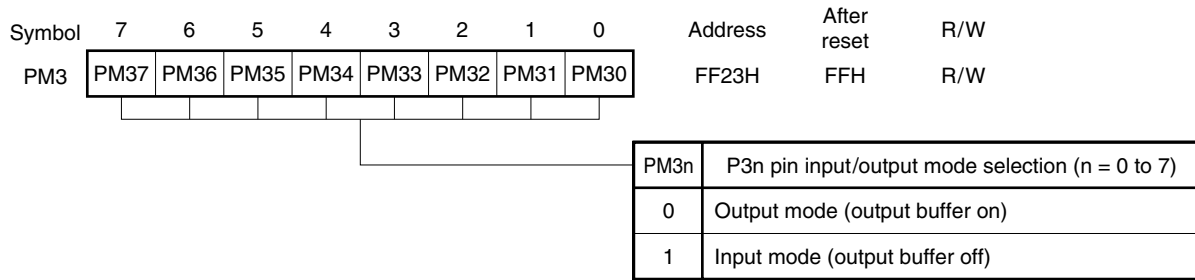
This register sets port 3 input/output in 1-bit units.

When using the P36/BUZ pin for buzzer output, clear PM36 and the output latch of P36 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets PM3 to FFH.

**Figure 13-3. Format of Port Mode Register 3**



## CHAPTER 14 A/D CONVERTER

### 14.1 A/D Converter Functions

The A/D converter converts an analog input into a digital value. It consists of 8 channels (ANI0 to ANI7) with an 8-bit resolution.

The conversion method is based on successive approximation and the conversion result is held in the 8-bit A/D conversion result register (ADCR).

A/D conversion can be started in the following two ways.

#### (1) Hardware start

Conversion is started by trigger input (INTP3).

#### (2) Software start

Conversion is started by setting the A/D converter mode register (ADM).

One analog input channel is selected from ANI0 to ANI7 and A/D conversion is carried out. In the case of hardware start, A/D conversion stops when an A/D conversion operation ends, and an interrupt request (INTAD) is generated. In the case of software start, A/D conversion is repeated. Each time an A/D conversion operation ends, an interrupt request (INTAD) is generated.

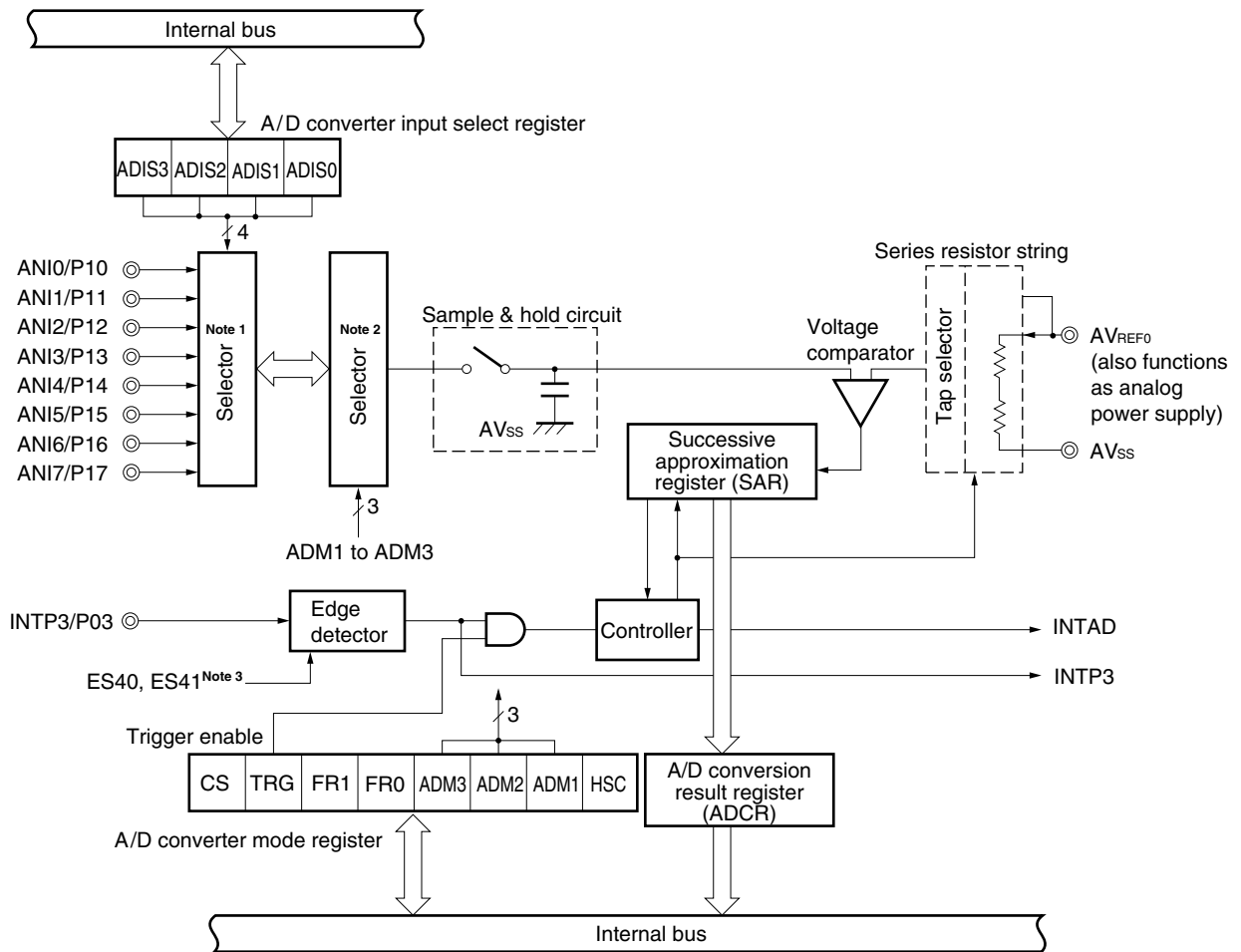
### 14.2 A/D Converter Configuration

The A/D converter consists of the following hardware.

**Table 14-1. A/D Converter Configuration**

Item	Configuration
Analog inputs	8 channels (ANI0 to ANI7)
Control registers	A/D converter mode register (ADM) A/D converter input select register (ADIS) External interrupt mode register 1 (INTM1)
Registers	Successive approximation register (SAR) A/D conversion result register (ADCR)

Figure 14-1. A/D Converter Block Diagram



- Notes**
1. Selector to select the number of channels to be used for analog input.
  2. Selector to select the channel for A/D conversion.
  3. ES40, ES41: Bits 0 and 1 of external interrupt mode register 1 (INTM1)

**(1) Successive approximation register (SAR)**

This register compares the analog input voltage value to the voltage tap (compare voltage) value applied from the series resistor string and holds the result from the most significant bit (MSB).

When up to the least significant bit (LSB) is held (end of A/D conversion), the SAR contents are transferred to the A/D conversion result register (ADCR).

**(2) A/D conversion result register (ADCR)**

This register holds the A/D conversion result. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR).

ADCR is read with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input makes ADCR undefined.

**(3) Sample & hold circuit**

The sample & hold circuit samples each analog input signal sequentially applied from the input circuit and sends it to the voltage comparator. This circuit holds the sampled analog input voltage value during A/D conversion.

**(4) Voltage comparator**

The voltage comparator compares the analog input to the series resistor string output voltage.

**(5) Series resistor string**

The series resistor string is connected between  $AV_{\text{REF0}}$  and  $AV_{\text{SS}}$ , and generates a voltage to be compared with the analog input.

**(6) ANI0 to ANI7 pins**

These are 8-channel analog input pins to input analog signals to undergo A/D conversion to the A/D converter. Pins other than those selected as analog input by the A/D converter input select register (ADIS) can be used as I/O ports.

**Cautions** 1. Use the ANI0 to ANI7 input voltages within the specified range. If a voltage higher than or equal to  $AV_{\text{REF0}}$  or lower than or equal to  $AV_{\text{SS}}$  is applied (even if within the absolute maximum ratings), the converted value of the corresponding channel becomes undefined and may adversely affect the converted values of other channels.

2. The analog input pins (ANI0 to ANI7) also function as I/O port pins (port 1). When A/D conversion is performed with any of pins ANI0 to ANI7 selected, be sure not to execute an instruction that inputs data to port 1 while conversion is in progress, as this may reduce the conversion resolution.

Also, if digital pulses are applied to a pin adjacent to the pin in the process of A/D conversion, the expected A/D conversion value may not be obtained due to coupling noise. Therefore, avoid applying pulses to pins adjacent to the pin undergoing A/D conversion.



**(7) AV<sub>REF0</sub> pin**

This pin inputs the A/D converter reference voltage.

It converts signals input to ANI0 to ANI7 into digital signals according to the voltage applied between AV<sub>REF0</sub> and AV<sub>SS</sub>.

The current flowing in the series resistor string can be reduced by setting the voltage to be input to the AV<sub>REF0</sub> pin to AV<sub>SS</sub> level in standby mode.

This pin also serves as an analog power supply pin. Supply power to this pin when the A/D converter is used.

**Caution** A series resistor string of approximately 10 k $\Omega$  is connected between the AV<sub>REF0</sub> pin and AV<sub>SS</sub> pin. Therefore, if the output impedance of the reference voltage source is high, this will result in series connection to the series resistor string between AV<sub>REF0</sub> pin and the AV<sub>SS</sub> pin, resulting in a large reference voltage error.

**(8) AV<sub>SS</sub> pin**

This is a GND potential pin of the A/D converter. Keep it at the same potential as the V<sub>SS0</sub> pin when not using the A/D converter.

### 14.3 A/D Converter Control Registers

The following three registers are used to control the A/D converter.

- A/D converter mode register (ADM)
- A/D converter input select register (ADIS)
- External interrupt mode register 1 (INTM1)

#### (1) A/D converter mode register (ADM)

This register sets the analog input channel for A/D conversion, conversion time, conversion start/stop and external trigger.

ADM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets ADM to 01H.

Figure 14-2. Format of A/D Converter Mode Register

Symbol	<7>	<6>	5	4	3	2	1	0	Address	After reset	R/W
ADM	CS	TRG	FR1	FR0	ADM3	ADM2	ADM1	HSC	FF80H	01H	R/W

ADM3	ADM2	ADM1	Analog input channel selection
0	0	0	ANI0
0	0	1	ANI1
0	1	0	ANI2
0	1	1	ANI3
1	0	0	ANI4
1	0	1	ANI5
1	1	0	ANI6
1	1	1	ANI7

FR1	FR0	HSC	A/D conversion time selection <sup>Note 1</sup>			
			fx = 5.0 MHz operation		fx = 4.19 MHz operation	
			MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	1	80/fx (16.0 μs)	160/fx (32.0 μs)	80/fx (19.1 μs)	160/fx (38.1 μs)
0	1	1	40/fx (Setting prohibited <sup>Note 2</sup> )	80/fx (16.0 μs)	40/fx (Setting prohibited <sup>Note 2</sup> )	80/fx (19.1 μs)
1	0	0	50/fx (Setting prohibited <sup>Note 2</sup> )	100/fx (20.0 μs)	50/fx (Setting prohibited <sup>Note 2</sup> )	100/fx (23.8 μs)
1	0	1	100/fx (20.0 μs)	200/fx (40.0 μs)	100/fx (23.8 μs)	200/fx (47.7 μs)
Other than above			Setting prohibited			

TRG	External trigger selection
0	No external trigger (software starts)
1	Conversion started by external trigger (hardware starts)

CS	A/D conversion operation control
0	Operation stop
1	Operation start

- Notes**
1. Set so that the A/D conversion time is 16 μs or more.
  2. Setting is prohibited because the A/D conversion time is less than 16 μs with fx set to this condition.

- Cautions**
1. The following sequence is recommended for reducing the power consumption of the A/D converter when the standby function is used: Clear bit 7 (CS) to 0 first to stop the A/D conversion operation, and then execute the HALT or STOP instruction.
  2. When restarting a stopped A/D conversion operation, start the A/D conversion operation after clearing the interrupt request flag (ADIF) to 0.

- Remarks**
1. fx: Main system clock oscillation frequency
  2. MCS: Bit 0 of oscillation mode select register (OSMS)

**(2) A/D converter input select register (ADIS)**

This register determines whether the ANI0/P10 to ANI7/P17 pins should be used for analog input channels or ports. Pins other than those selected as analog input can be used as I/O ports.

ADIS is set with an 8-bit memory manipulation instruction.

RESET input clears ADIS to 00H.

**Cautions 1. Set the analog input channel using the following procedure.**

**(1) Set the number of analog input channels using ADIS.**

**(2) Using the A/D converter mode register (ADM), select one channel to undergo A/D conversion from among the channels set to analog input by ADIS.**

**2. No internal pull-up resistor can be used for the channels set to analog input by ADIS, irrespective of the value of bit 1 (PUO1) of pull-up resistor option register L (PUOL).**

**Figure 14-3. Format of A/D Converter Input Select Register**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADIS	0	0	0	0	ADIS3	ADIS2	ADIS1	ADIS0	FF84H	00H	R/W

ADIS3	ADIS2	ADIS1	ADIS0	Number of analog input channel selection
0	0	0	0	No analog input channel (P10 to P17)
0	0	0	1	1 channels (ANI0, P11 to P17)
0	0	1	0	2 channels (ANI0, ANI1, P12 to P17)
0	0	1	1	3 channels (ANI0 to ANI2, P13 to P17)
0	1	0	0	4 channels (ANI0 to ANI3, P14 to P17)
0	1	0	1	5 channels (ANI0 to ANI4, P15 to P17)
0	1	1	0	6 channels (ANI0 to ANI5, P16, P17)
0	1	1	1	7 channels (ANI0 to ANI6, P17)
1	0	0	0	8 channels (ANI0 to ANI7)
Other than above				Setting prohibited

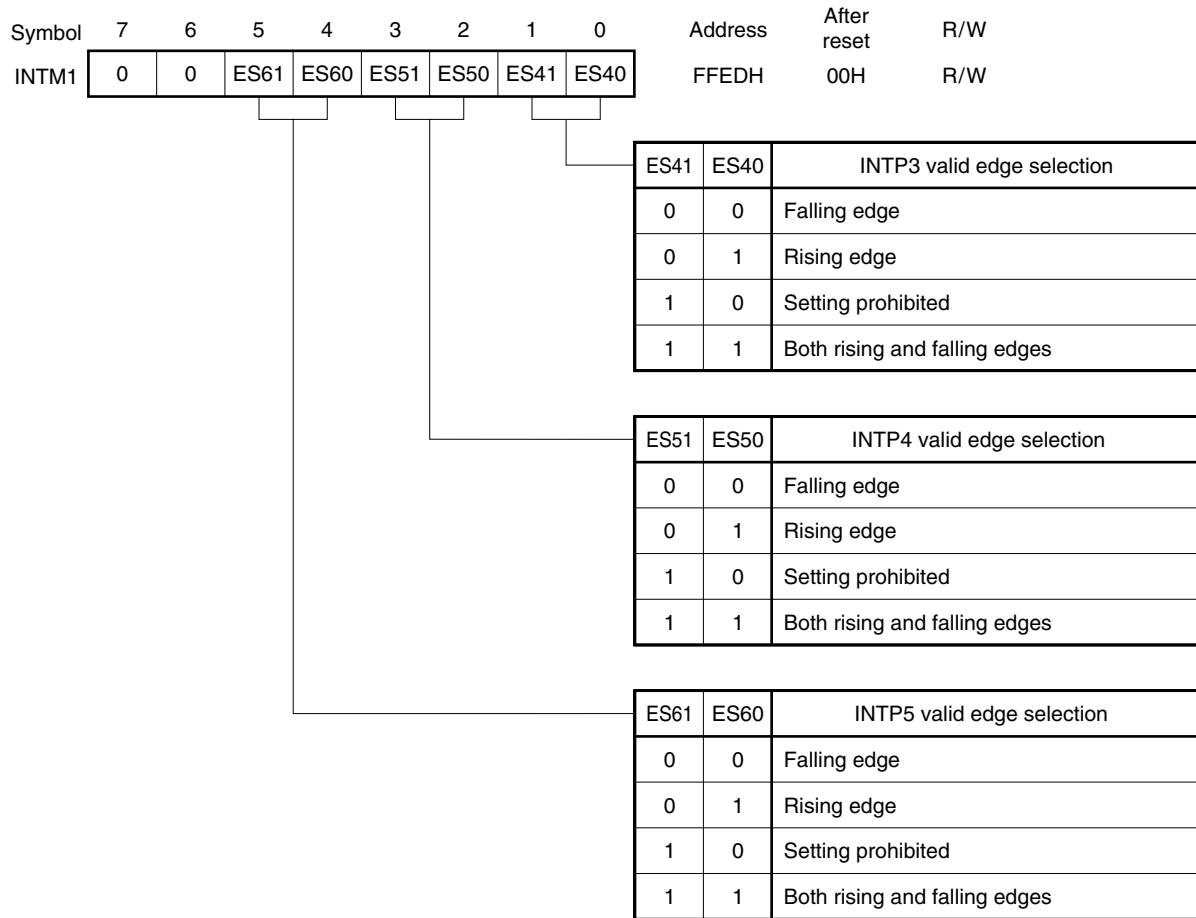
**(3) External interrupt mode register 1 (INTM1)**

This register sets the valid edge for INTP3 to INTP5.

INTM1 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears INTM1 to 00H.

**Figure 14-4. Format of External Interrupt Mode Register 1**



## 14.4 A/D Converter Operations

### 14.4.1 Basic operations of A/D converter

- (1) Set the number of analog input channels using the A/D converter input select register (ADIS).
- (2) From among the analog input channels set by ADIS, select one channel for A/D conversion using the A/D converter mode register (ADM).
- (3) Sample the voltage input to the selected analog input channel using the sample & hold circuit.
- (4) Sampling for the specified period of time sets the sample & hold circuit to the hold state so that the circuit holds the input analog voltage until the end of A/D conversion.
- (5) Bit 7 of the successive approximation register (SAR) is set. The series resistor string voltage tap is set to  $(1/2) AV_{REF0}$  by the tap selector.
- (6) The voltage difference between the series resistor string voltage tap and analog input is compared by the voltage comparator. If the analog input is greater than  $(1/2) AV_{REF0}$ , the MSB of SAR remains set. If the input is smaller than  $(1/2) AV_{REF0}$ , the MSB is reset.
- (7) Next, bit 6 of SAR is automatically set and the operation proceeds to the next comparison. In this case, the series resistor string voltage tap is selected according to the preset value of bit 7 as described below.
  - Bit 7 = 1:  $(3/4) AV_{REF0}$
  - Bit 7 = 0:  $(1/4) AV_{REF0}$

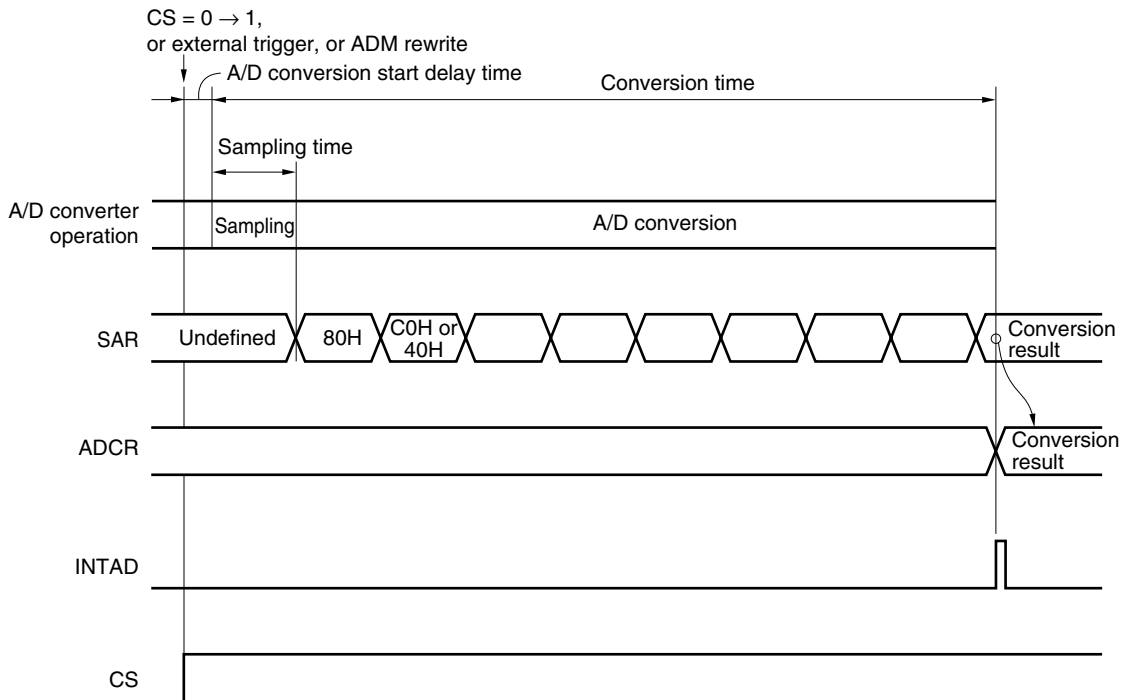
The voltage tap and analog input voltage are compared and bit 6 of SAR is manipulated with the result as follows.

- Analog input voltage  $\geq$  Voltage tap: Bit 6 = 1
  - Analog input voltage  $<$  Voltage tap: Bit 6 = 0
- (8) Comparison of this sort continues up to bit 0 of SAR.
  - (9) Upon completion of the comparison of 8 bits, a valid digital result remains in SAR and that value is transferred to and latched in the A/D conversion result register (ADCR).

At the same time, the A/D conversion end interrupt request (INTAD) can also be generated.

★

Figure 14-5. A/D Converter Basic Operation



A/D conversion operations are performed continuously until bit 7 (CS) of the AD converter mode register (ADM) is reset to 0 by software.

$\overline{\text{RESET}}$  input makes ADCR undefined.

Check the completion of A/D conversion by using the A/D conversion end interrupt request flag (ADIF).

Table 14-2. A/D Converter Sampling Time and A/D Conversion Start Delay Time

FR01	FR00	HSOC	Conversion Time <sup>Note 1</sup>		Sampling Time		A/D Conversion Start Delay Time	
			MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	1	80/fx (16.0 μs)	160/fx (32.0 μs)	9/fx	18/fx	6/fx	12/fx
0	1	1	40/fx (setting prohibited <sup>Note 2</sup> )	80/fx (16.0 μs)	4.5/fx	9/fx	3/fx	6/fx
1	0	0	50/fx (setting prohibited <sup>Note 2</sup> )	100/fx (20.0 μs)	5.25/fx	10.5/fx	4.5/fx	9/fx
1	0	1	100/fx (20.0 μs)	200/fx (40.0 μs)	10.5/fx	21/fx	9/fx	18/fx
Other than above			Setting prohibited		-		-	

**Notes 1.** Set so that the A/D conversion time is 16 μs or more.

**2.** Setting is prohibited because the A/D conversion time is less than 16 μs with fx set to this condition.

**Remarks 1.** fx: Main system clock oscillation frequency

**2.** Values in parentheses apply to operation with fx = 5.0 MHz.

**14.4.2 Input voltage and conversion results**

The relationship between the analog input voltage input to the analog input pins (ANI0 to ANI7) and the A/D conversion result (the value stored in the A/D conversion result register (ADCR)) is shown by the following expression.

$$ADCR = INT \left( \frac{V_{IN}}{AV_{REF0}} \times 256 + 0.5 \right)$$

or

$$(ADCR - 0.5) \times \frac{AV_{REF0}}{256} \leq V_{IN} < (ADCR + 0.5) \times \frac{AV_{REF0}}{256}$$

Where, INT( ): Function which returns integer part of value in parentheses.

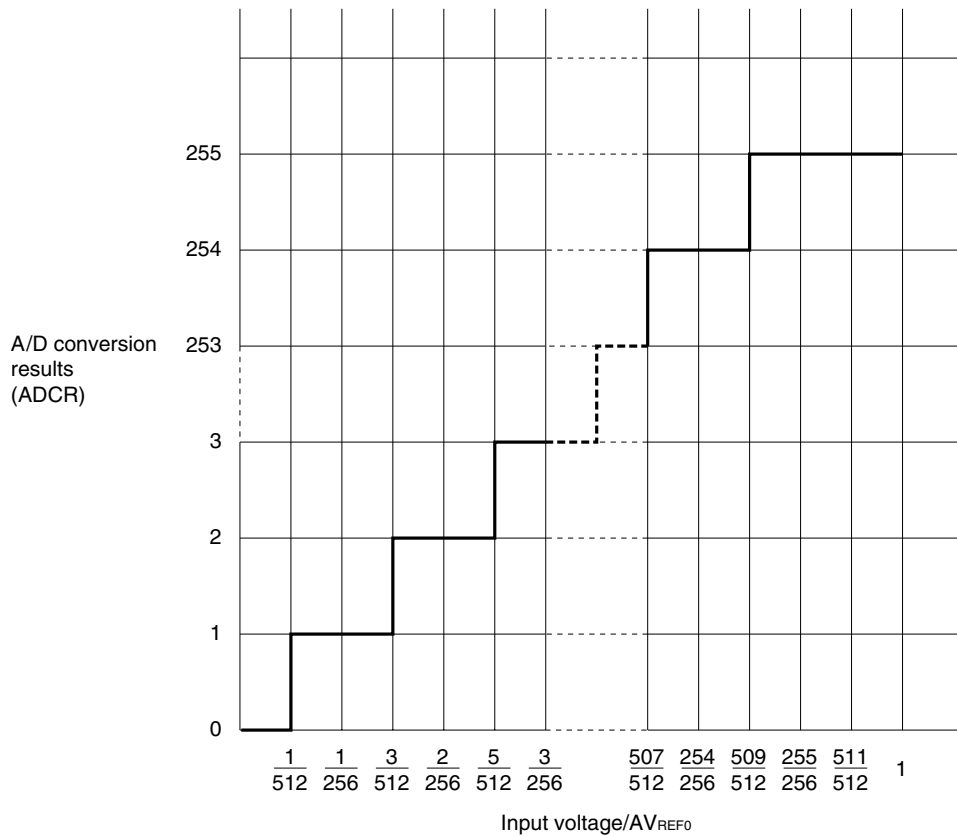
V<sub>IN</sub>: Analog input voltage

AV<sub>REF0</sub>: AV<sub>REF0</sub> pin voltage

ADCR Value of A/D conversion result register (ADCR)

Figure 14-6 shows the relationship between the analog input voltage and the A/D conversion result.

**Figure 14-6. Relationship Between Analog Input Voltage and A/D Conversion Result**





**14.4.3 A/D converter operating mode**

One analog input channel is selected from among ANI0 to ANI7 by the A/D converter input select register (ADIS) and A/D converter mode register (ADM) and A/D conversion is started.

A/D conversion can be started in the following two ways.

- Hardware start: Conversion is started by trigger input (INTP3).
- Software start: Conversion is started by setting ADM.

The A/D conversion result is stored in the A/D conversion result register (ADCR) and the interrupt request signal (INTAD) is simultaneously generated.

**(1) A/D conversion by hardware start**

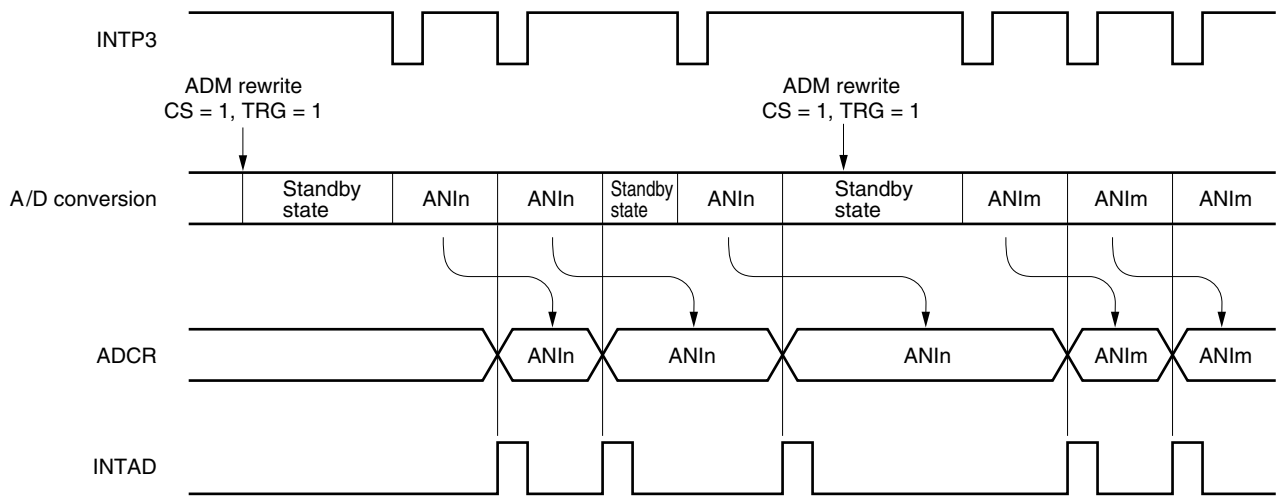
When bit 6 (TRG) and bit 7 (CS) of the A/D converter mode register (ADM) are set to 1, the A/D conversion standby state is set. When the external trigger signal (INTP3) is input, the A/D conversion starts on the voltage applied to the analog input pins specified by bits 1 to 3 (ADM1 to ADM3) of ADM.

Upon termination of A/D conversion, the conversion result is stored in the A/D conversion result register (ADCR) and the interrupt request signal (INTAD) is generated. After one A/D conversion operation is started and terminated, another operation is not started until a new external trigger signal is input.

If data with CS set to 1 is written to ADM again during A/D conversion, the converter suspends the A/D conversion operation and waits for a new external trigger signal to be input. When the external trigger input signal is input again, A/D conversion is carried out from the beginning.

If data with CS cleared to 0 is written to ADM during A/D conversion, the A/D conversion operation stops immediately.

**Figure 14-7. A/D Conversion by Hardware Start**



**Remark** n = 0, 1, ..., 7  
m = 0, 1, ..., 7

**(2) A/D conversion operation in software start**

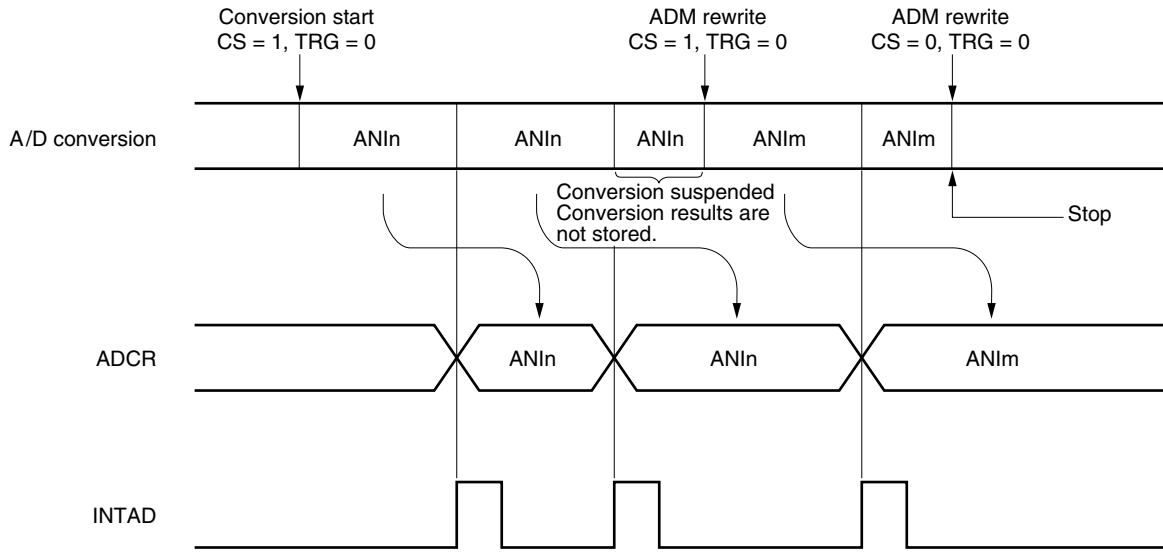
When bit 6 (TRG) and bit 7 (CS) of the A/D converter mode register (ADM) are set to 0 and 1, respectively, A/D conversion starts on the voltage applied to the analog input pins specified by bits 1 to 3 (ADM1 to ADM3) of ADM.

Upon termination of A/D conversion, the conversion result is stored in the A/D conversion result register (ADCR) and the interrupt request signal (INTAD) is generated. After one A/D conversion operation is started and terminated, the next A/D conversion operation starts immediately. The A/D conversion operation continues repeatedly until new data is written to ADM.

If data with CS set to 1 is written to ADM again during A/D conversion, the converter suspends the A/D conversion operation and starts A/D conversion on the newly written data.

If data with CS cleared to 0 is written to ADM during A/D conversion, the A/D conversion operation stops immediately.

**Figure 14-8. A/D Conversion by Software Start**



**Remark** n = 0, 1, ..., 7  
 m = 0, 1, ..., 7

★ **14.5 How to Read the 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 1 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).

When the resolution is 8 bits,

$$1\text{LSB} = 1/2^8 = 1/256$$

$$= 0.4\%\text{FSR}$$

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 offset, full-scale offset, integral linearity error, differential linearity error and errors which are combinations of these express the overall error.

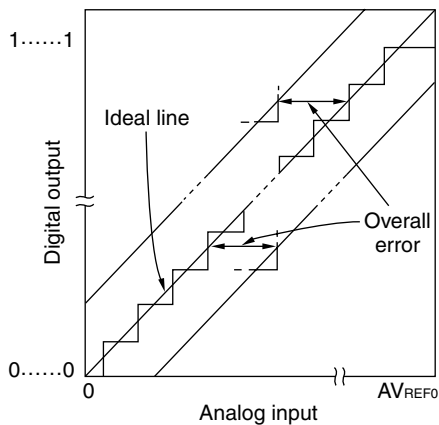
Note that the quantization error is not included in the overall error in the characteristics table.

**(3) Quantization error**

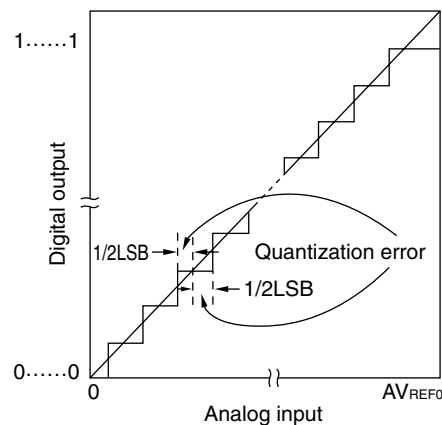
When analog values are converted to digital values, a  $\pm 1/2\text{LSB}$  error naturally occurs. In an A/D converter, an analog input voltage in a range of  $\pm 1/2\text{LSB}$  is converted to the same digital code, so a quantization error cannot be avoided.

Note that the quantization error is not included in the overall error, zero-scale offset, full-scale offset, integral linearity error, and differential linearity error in the characteristics table.

**Figure 14-9. Overall Error**



**Figure 14-10. Quantization Error**



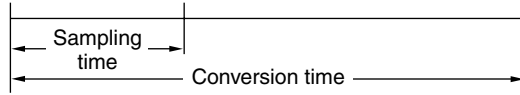
**(4) Conversion time**

This expresses the time from when the analog input voltage was applied to the time when the digital output was obtained.

The sampling time is included in the conversion time in the characteristics table.

**(5) Sampling time**

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.

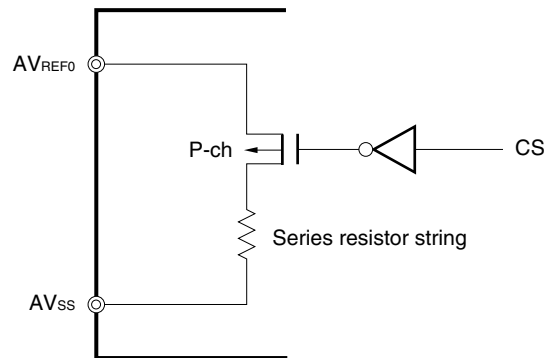


## 14.6 A/D Converter Cautions

### ★ (1) Power consumption in standby mode

A/D converter stops operating in the standby mode. At this time, current consumption can be reduced by stopping the conversion operation (by setting bit 7 (CS) of the A/D converter mode register (ADM) to 0). Figure 14-11 shows how to reduce the current consumption in the standby mode.

**Figure 14-11. Example of Method of Reducing Current Consumption in Standby Mode**



### (2) Input range of ANI0 to ANI7

The input voltages of ANI0 to ANI7 should be within the specification range. In particular, if a voltage of AVREF0 or above or AVSS or below is input (even if within the absolute maximum rating range), the conversion value for that channel will be undefined. The conversion values of the other channels may also be affected.

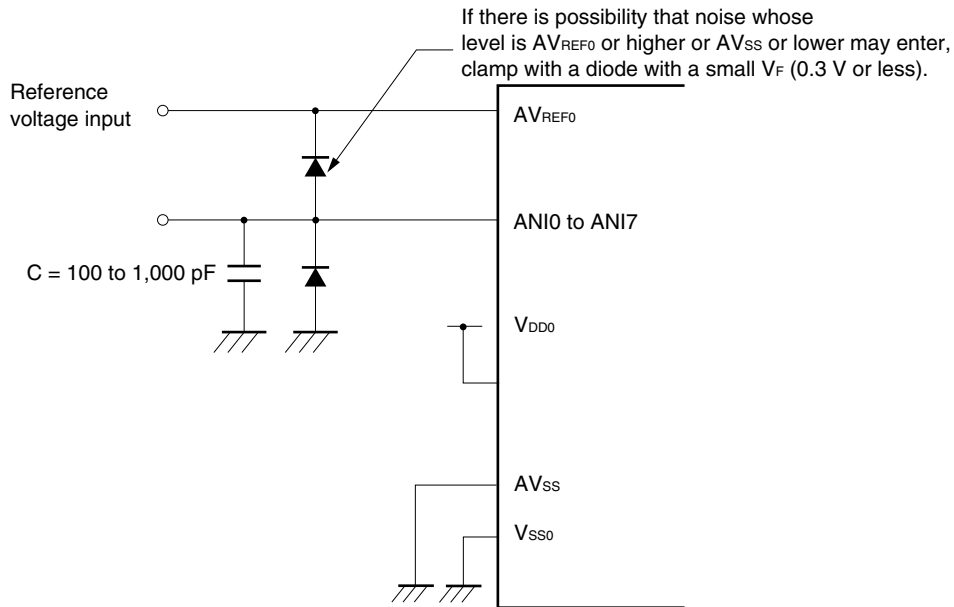
### ★ (3) Conflicting operations

- <1> Conflict between A/D conversion result register (ADCR) write and ADCR read by instruction upon end of conversion  
ADCR read is given priority. After the read operation, the new conversion result is written to ADCR.
- <2> Conflict between ADCR write and external trigger signal input upon end of conversion  
The external trigger signal is not acknowledged during A/D conversion. Therefore, the external trigger signal is not acknowledged during ADCR write.
- <3> Conflict between ADCR write and A/D converter mode register (ADM) write or A/D converter input select register (ADIS) write  
ADM or ADIS write is given priority. ADCR write is not performed, nor is the conversion end interrupt request signal (INTAD) generated.

**(4) Noise countermeasures**

In order to maintain 8-bit resolution, attention must be paid to noise on the  $AV_{REF0}$  and ANI0 to ANI7 pins. Since the effect increases in proportion to the output impedance of the analog input source, it is recommended that a capacitor be connected externally as shown in Figure 14-12 in order to reduce noise.

★

**Figure 14-12. Analog Input Pin Handling****(5) ANI0/P10 to ANI7/P17 pins**

The analog input pins ANI0 to ANI7 also function as I/O port pins (port 1). When A/D conversion is performed with any of pins ANI0 to ANI7 selected, be sure not to execute an instruction that inputs data to port 1 while conversion is in progress, as this may reduce the conversion resolution.

Also, if digital pulses are applied to a pin adjacent to the pin in the process of A/D conversion, the expected A/D conversion value may not be obtainable due to coupling noise. Therefore, avoid applying pulses to pins adjacent to the pin undergoing A/D conversion.

★ **(6) Input impedance of ANI0 to ANI7 pins**

In this A/D converter, the internal sampling capacitor is charged and sampling is performed for approx. one tenth of the conversion time.

Since only the leakage current flows other than during sampling and the current for charging the capacitor also flows during sampling, the input impedance fluctuates and has no meaning.

To perform sufficient sampling, however, it is recommended to make the output impedance of the analog input source 10 k $\Omega$  or lower, or attach a capacitor of around 100 pF to the ANI0 to ANI7 pins (see Figure 14-12).

**(7)  $AV_{REF0}$  pin input impedance**

A series resistor string of approximately 10 k $\Omega$  is connected between the  $AV_{REF0}$  pin and the  $AV_{SS}$  pin.

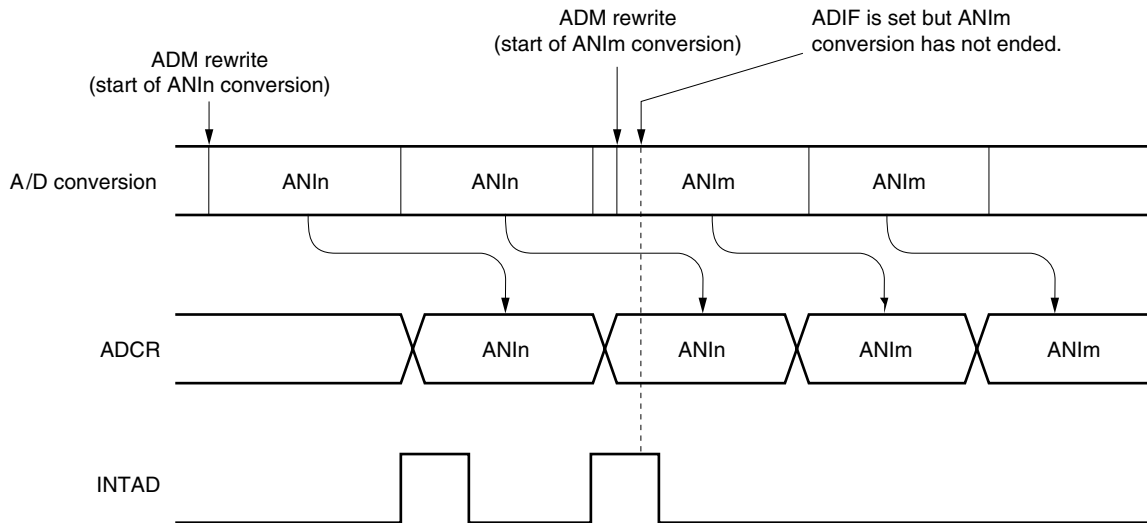
Therefore, if the output impedance of the reference voltage source is high, this will result in series connection to the series resistor string between the  $AV_{REF0}$  pin and the  $AV_{SS}$  pin, and there will be a large reference voltage error.

**(8) Interrupt request flag (ADIF)**

The interrupt request flag (ADIF) is not cleared even if the A/D converter mode register (ADM) is changed. Caution is therefore required since, if a change of analog input pin is performed during A/D conversion, the A/D conversion result and ADIF for the pre-change analog input may be set just before the ADM rewrite. At this time, when ADIF is read immediately after the ADM rewrite, ADIF may be set despite the fact that the A/D conversion for the post-change analog input has not ended.

When the A/D conversion is stopped and then resumed, clear ADIF before it is resumed.

**Figure 14-13. A/D Conversion End Interrupt Request Generation Timing**



**Remark**  $n = 0, 1, \dots, 7$   
 $m = 0, 1, \dots, 7$

**(9) Conversion result immediately after A/D converter start**

The first A/D conversion value immediately after A/D conversion is started may not satisfy ratings. Therefore, implement a countermeasure such as polling A/D conversion end interrupt requests (INTAD) to delete the first conversion result.

★ (10) Timing at which A/D conversion result is undefined

The A/D conversion value may be undefined if the timing of completion of A/D conversion and the timing of stopping the A/D conversion conflict with each other. Therefore, read the A/D conversion result during the A/D conversion operation. To read the conversion result after stopping the A/D conversion operation, be sure to stop the A/D conversion before the next conversion ends.

Figures 14-14 and 14-15 show the timing of reading the conversion result.

Figure 14-14. Timing of Reading Conversion Result (When Conversion Result is Undefined)

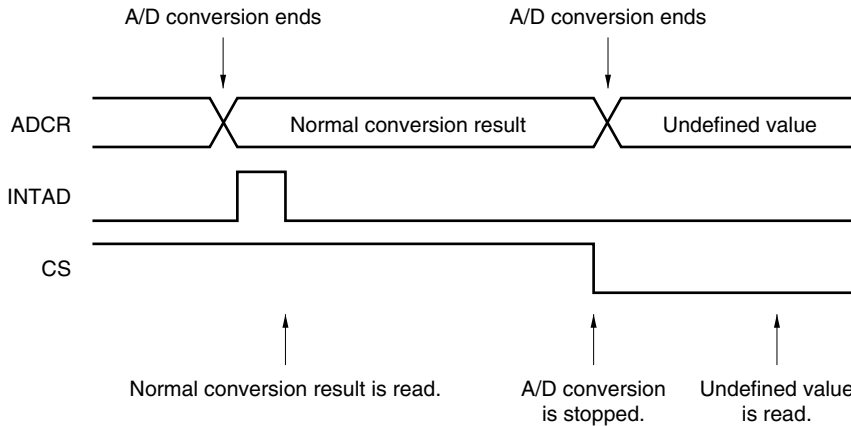
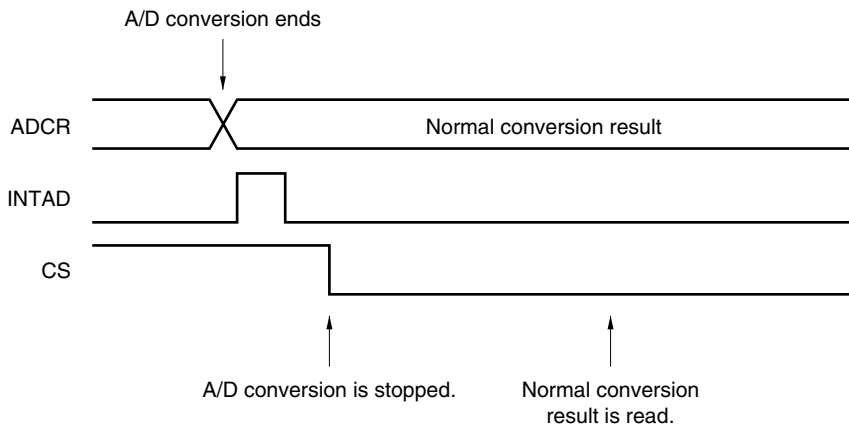


Figure 14-15. Timing of Reading Conversion Result (When Conversion Result is Normal)



★ (11) Notes on board design

Locate analog circuits as far away from digital circuits as possible on the board because the analog circuits may be affected by the noise of the digital circuits. In particular, do not cross an analog signal line with a digital signal line, or wire an analog signal line in the vicinity of a digital signal line. Otherwise, the A/D conversion characteristics may be affected by the noise of the digital line.

Connect AV<sub>SS</sub> and V<sub>SS0</sub> at one location on the board where the voltages are stable.

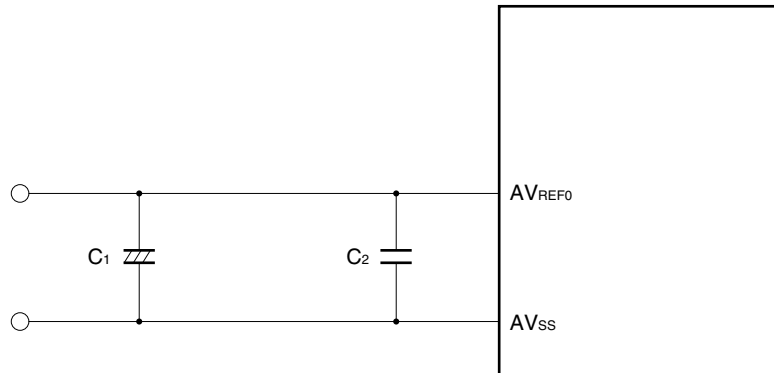


★ (12)  $AV_{REF0}$  pin

Connect a capacitor to the  $AV_{REF0}$  pin to minimize conversion errors due to noise. If an A/D conversion operation has been stopped and then is started, the voltage applied to the  $AV_{REF0}$  pin becomes unstable, causing the accuracy of the A/D conversion to drop. To prevent this, also connect a capacitor to the  $AV_{REF0}$  pin.

Figure 14-16 shows an example of connecting a capacitor. Capacitor C1 is effective for noise of low frequency and capacitor C2 is effective for noise of high frequency.

**Figure 14-16. Example of Connecting Capacitor to  $AV_{REF0}$  Pin**



**Remark** C1: 4.7  $\mu\text{F}$  to 10  $\mu\text{F}$  (reference value)  
 C2: 0.01  $\mu\text{F}$  to 0.1  $\mu\text{F}$  (reference value)  
 Connect C2 as close to the pin as possible.

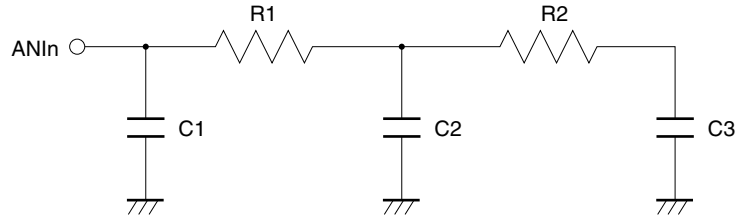
## ★ (13) Internal equivalent circuit of ANI0 to ANI7 pins and permissible signal source impedance

To complete sampling within the sampling time with sufficient A/D conversion accuracy, the impedance of the signal source such as a sensor must be sufficiently low. Figure 14-17 shows the internal equivalent circuit of the ANI0 to ANI7 pins.

If the impedance of the signal source is high, connect capacitors with a high capacitance to the ANI0 to ANI7 pins. An example of this is shown in Figure 14-18. In this case, however, the microcontroller cannot follow an analog signal with a high differential coefficient because a low-pass filter is created.

To convert a high-speed analog signal or to convert an analog signal in the scan mode, insert a low-impedance buffer.

Figure 14-17. Internal Equivalent Circuit of Pins ANI0 to ANI7



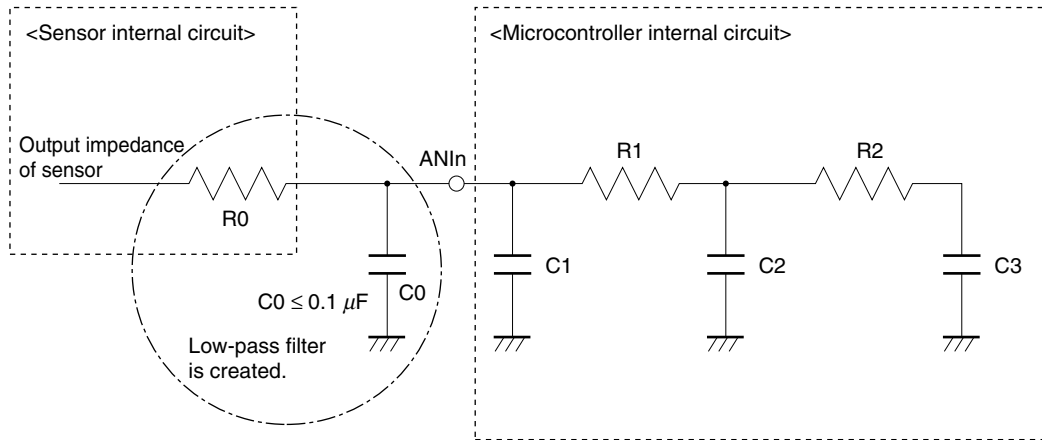
Remark n = 0 to 7

Table 14-3. Resistances and Capacitances of Equivalent Circuit (Reference Values)

$AV_{REF0}$	R1	R2	C1	C2	C3
1.8 V	75 k $\Omega$	30 k $\Omega$	8 pF	4 pF	2 pF
2.7 V	12 k $\Omega$	8 k $\Omega$	8 pF	3 pF	2 pF
4.5 V	4 k $\Omega$	2.7 k $\Omega$	8 pF	1.4 pF	2 pF

Caution The resistances and capacitances in Table 14-3 are not guaranteed values.

Figure 14-18. Example of Connection If Signal Source Impedance Is High



Remark n = 0 to 7

## CHAPTER 15 D/A CONVERTER

### 15.1 D/A Converter Functions

The D/A converter converts a digital input into an analog value. The D/A converter used is a 2-channel 8-bit resolution voltage output type D/A converter.

The conversion method used is the R-2R resistor ladder method.

Start D/A conversion by setting bits 0 and 1 (DACE0 and DACE1) of the D/A converter mode register (DAM).

There are two modes for the D/A converter, as follows.

**(1) Normal mode**

Outputs an analog voltage signal immediately after D/A conversion.

**(2) Real-time output mode**

Outputs an analog voltage signal synchronously with the output trigger after D/A conversion.

Since a sine wave can be generated in this mode, it is useful for an MSK modem for cordless telephone sets.

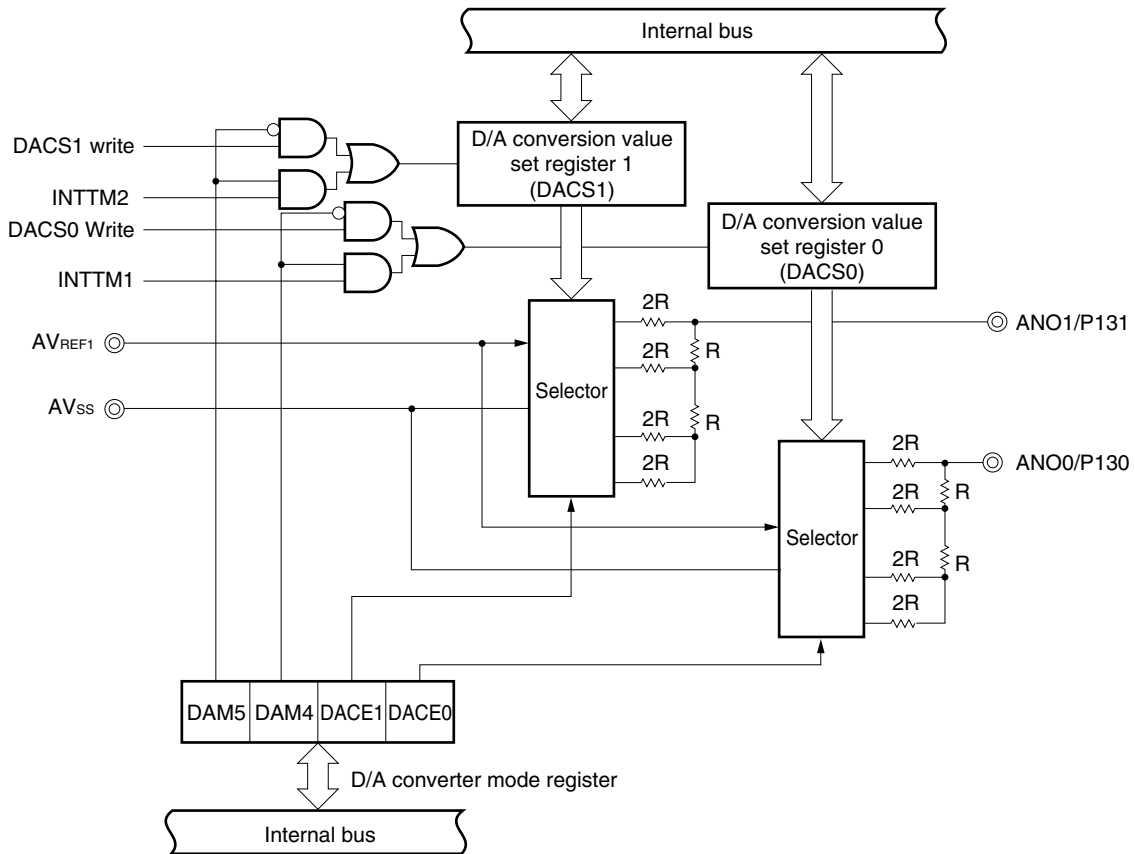
## 15.2 D/A Converter Configuration

The D/A converter consists of the following hardware.

**Table 15-1. D/A Converter Configuration**

Item	Configuration
Registers	D/A conversion value set register 0 (DACS0) D/A conversion value set register 1 (DACS1)
Control register	D/A converter mode register (DAM)

**Figure 15-1. D/A Converter Block Diagram**



**(1) D/A conversion value set registers 0, 1 (DACS0, DACS1)**

DACS0 and DACS1 are registers that set the values used to determine the analog voltages to be output to the ANO0 and ANO1 pins, respectively.

DACS0 and DACS1 are set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears DACS0 and DACS1 to 00H.

Analog voltage output to the ANO0 and ANO1 pins is determined by the following expression.

$$\text{ANOn output voltage} = AV_{\text{REF1}} \times \frac{\text{DACS}_n}{256}$$

where,  $n = 0, 1$

- Cautions**
1. In the real-time output mode, when data that is set in DACS0 and DACS1 is read before an output trigger is generated, the previous data is read rather than the set data.
  2. In the real-time output mode, data should be set to DACS0 and DACS1 after an output trigger and before the next output trigger.

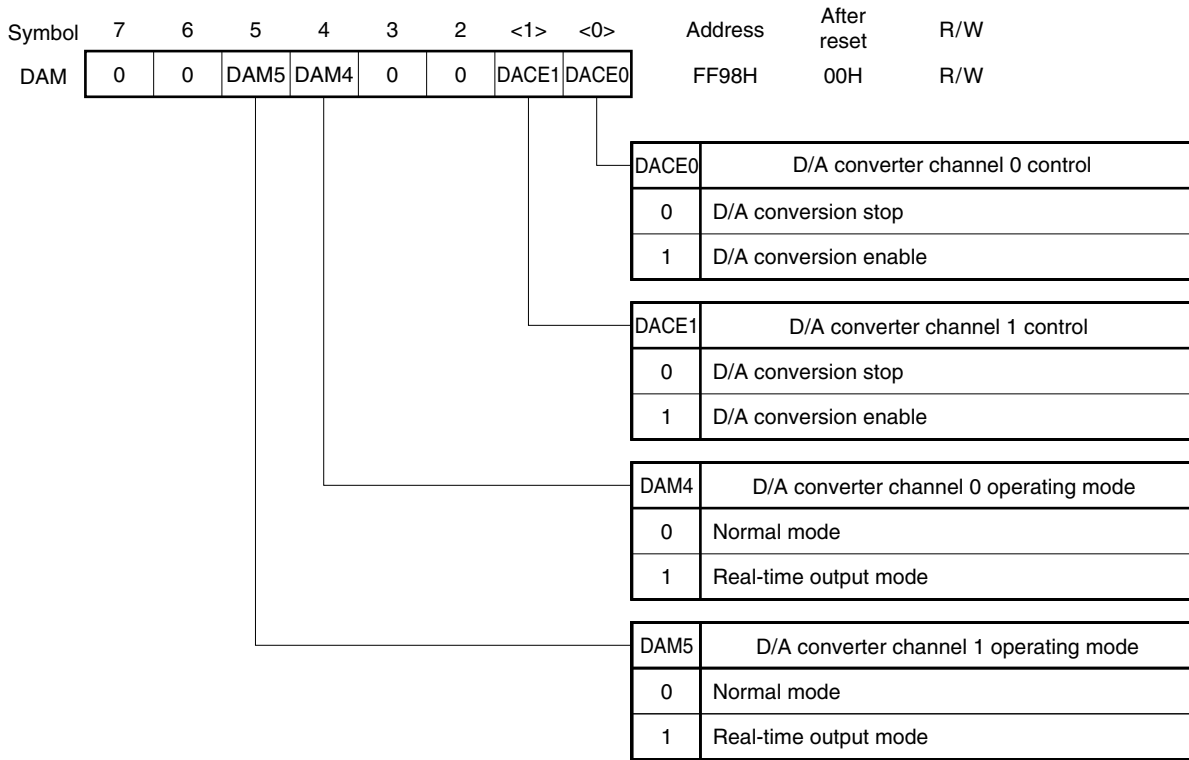
### 15.3 D/A Converter Control Registers

The D/A converter mode register (DAM) controls the D/A converter. This register sets D/A converter operation enable/stop.

DAM is set with a 1-bit or an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears DAM to 00H.

**Figure 15-2. Format of D/A Converter Mode Register**



- Cautions**
1. When using the D/A converter, alternate-function port pins should be set to the input mode, and pull-up resistors should be disconnected.
  2. Always set bits 2, 3, 6, and 7 to 0.
  3. When D/A conversion is stopped, the output state is high-impedance.
  4. The output triggers are INTTM1 and INTTM2 for channel 0 and channel 1, respectively, in the real-time output mode.

## 15.4 D/A Converter Operations

- (1) The channel 0 operating mode and channel 1 operating mode are selected by bits 4 and 5 (DAM4 and DAM5), respectively, of the D/A converter mode register (DAM).
- (2) Set the data corresponding to the analog voltages output to the ANO0/P130 and ANO1/P131 pins to D/A conversion value setting registers 0 and 1 (DACS0 and DACS1), respectively.
- (3) D/A conversion of channel 0 or channel 1 can be started by setting bits 0 or 1 (DACE0 or DACE1) of DAM, respectively.
- (4) In the normal mode, the analog voltage signals are output to the ANO0/P130 and ANO1/P131 pins immediately after D/A conversion. In the real-time output mode, the analog voltage signals are output synchronously with the output triggers.
- (5) In the normal mode, the analog voltage signals to be output are held until new data is set in DACS0 and DACS1. In the realtime output mode, new data is set in DACS0 and DACS1 and then held until the next trigger is generated.

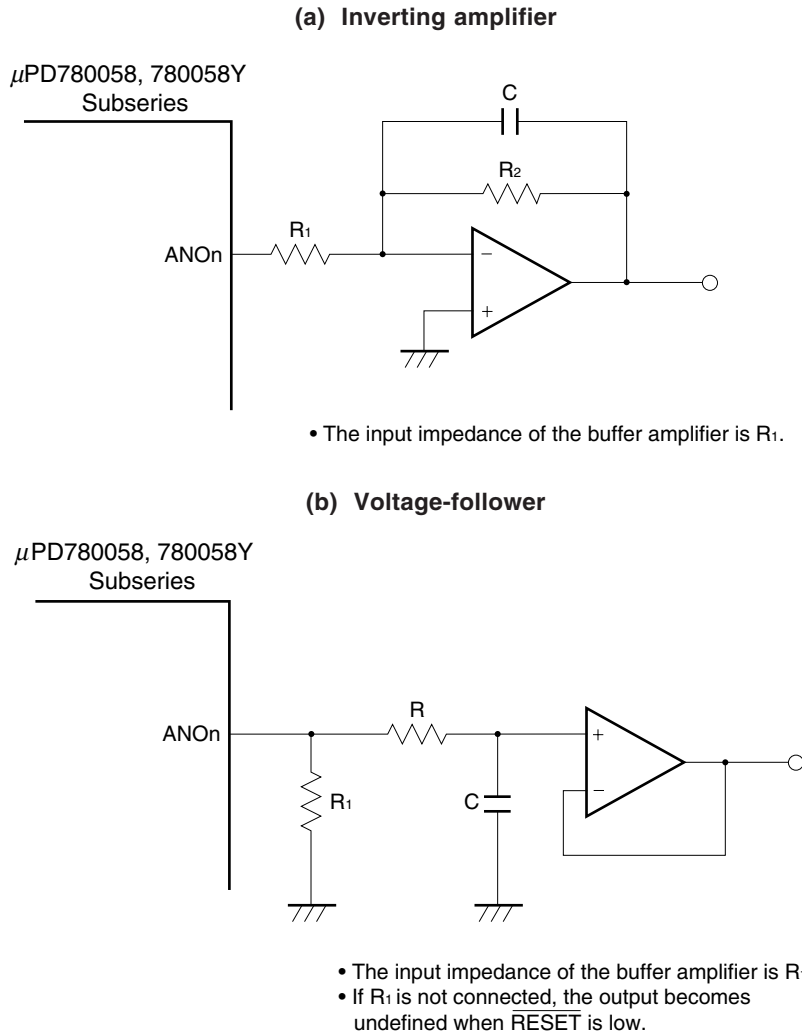
**Caution** Set DACE0 and DACE1 after setting data in DACS0 and DACS1.

## 15.5 D/A Converter Cautions

### (1) Output impedance of D/A converter

Because the output impedance of the D/A converter is high, use of current flowing from the ANOn pins ( $n = 0, 1$ ) is prohibited. If the input impedance of the load for the converter is low, insert a buffer amplifier between the load and the ANOn pins. In addition, wiring from the ANOn pins to the buffer amplifier or the load should be as short as possible (because of high output impedance). If the wiring may be long, design the ground pattern so as to be close to those lines or use some other expedient to achieve shorter wiring.

Figure 15-3. Use Example of Buffer Amplifier



### (2) Output voltage of D/A converter

Because the output voltage of the converter changes in steps, use the D/A converter output signals in general by connecting a low-pass filter.

### (3) AVREF1 pin

When only one of the D/A converter channels is used with  $AV_{REF1} < V_{DD0}$ , the other pins that are not used as analog outputs must be set as follows:

- Set the PM13x bit of port mode register 13 (PM13) to 1 (input mode) and connect the pin to  $V_{SS0}$ .
- Set the PM13x bit of port mode register 13 (PM13) to 0 (output mode) and the output latch to 0, and output a low level from the pin.

★ When not using the D/A converter, use  $AV_{REF1}$  with its potential the same as that of the  $V_{DD0}$ .



## CHAPTER 16 SERIAL INTERFACE CHANNEL 0 ( $\mu$ PD780058 SUBSERIES)

The  $\mu$ PD780058 Subseries incorporates three serial interface channels. Differences between channels 0, 1, and 2 are as follows (see **CHAPTER 18 SERIAL INTERFACE CHANNEL 1** for details of serial interface channel 1 and **CHAPTER 19 SERIAL INTERFACE CHANNEL 2** for details of serial interface channel 2).

**Table 16-1. Differences Between Channels 0, 1, and 2**

Serial Transfer Mode		Channel 0	Channel 1	Channel 2
3-wire serial I/O	Clock selection	$f_{xx}/2$ , $f_{xx}/2^2$ , $f_{xx}/2^3$ , $f_{xx}/2^4$ , $f_{xx}/2^5$ , $f_{xx}/2^6$ , $f_{xx}/2^7$ , $f_{xx}/2^8$ , external clock, TO2 output	$f_{xx}/2$ , $f_{xx}/2^2$ , $f_{xx}/2^3$ , $f_{xx}/2^4$ , $f_{xx}/2^5$ , $f_{xx}/2^6$ , $f_{xx}/2^7$ , $f_{xx}/2^8$ , external clock, TO2 output	External clock, baud rate generator output
	Transfer method	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit Automatic transmit/ receive function	MSB/LSB switchable as the start bit
	Transfer end flag	Serial transfer end interrupt request flag (CSIF0)	Serial transfer end interrupt request flag (CSIF1)	Serial transfer end interrupt request flag (SRIF)
SBI (serial bus interface)		Use possible	None	None
2-wire serial I/O				
UART (Asynchronous serial interface)		None		Use possible Time-division transfer function

## 16.1 Functions of Serial Interface Channel 0

Serial interface channel 0 employs the following four modes.

- Operation stop mode
- 3-wire serial I/O mode
- SBI (serial bus interface) mode
- 2-wire serial I/O mode

**Caution** Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or SBI) while serial interface channel 0 is enabled to operate. To change the operating mode, stop the serial operation first.

### (1) Operation stop mode

This mode is used when serial transfer is not carried out. Power consumption can be reduced in this mode.

### (2) 3-wire serial I/O mode (MSB-/LSB-first selectable)

This mode is used for 8-bit data transfer using three lines, one each for the serial clock ( $\overline{\text{SCK0}}$ ), serial output (SO0) and serial input (SI0). This mode enables simultaneous transmission/reception and therefore reduces the data transfer processing time.

The start bit of transferred 8-bit data is switchable between MSB and LSB, so that devices can be connected regardless of their start bit recognition.

This mode should be used when connecting with peripheral I/O devices or display controllers which incorporate a conventional clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

### (3) SBI (serial bus interface) mode (MSB-first)

This mode is used for 8-bit data transfer with two or more devices using the serial clock ( $\overline{\text{SCK0}}$ ) and serial data bus (SB0 or SB1) lines (see **Figure 16-1**).

The SBI mode conforms to the NEC Electronics serial bus format, and transmits or receives three types of transfer data: “addresses”, “commands”, and “data”.

- Address: Data to select the target device for serial communication
- Command: Data to give an instruction to the target device
- Data: Data actually transferred

Actually, the master device outputs an “address” to the serial bus to select one of the slave devices with which the master device is to communicate. After that, “commands” and “data” are transmitted or received between the master and slave devices (this is the serial transfer). The receiver can automatically identify the received data as an “address”, “command”, or “data” by hardware.

This function enables the I/O ports to be used effectively and the serial interface control portions of the application program to be simplified.

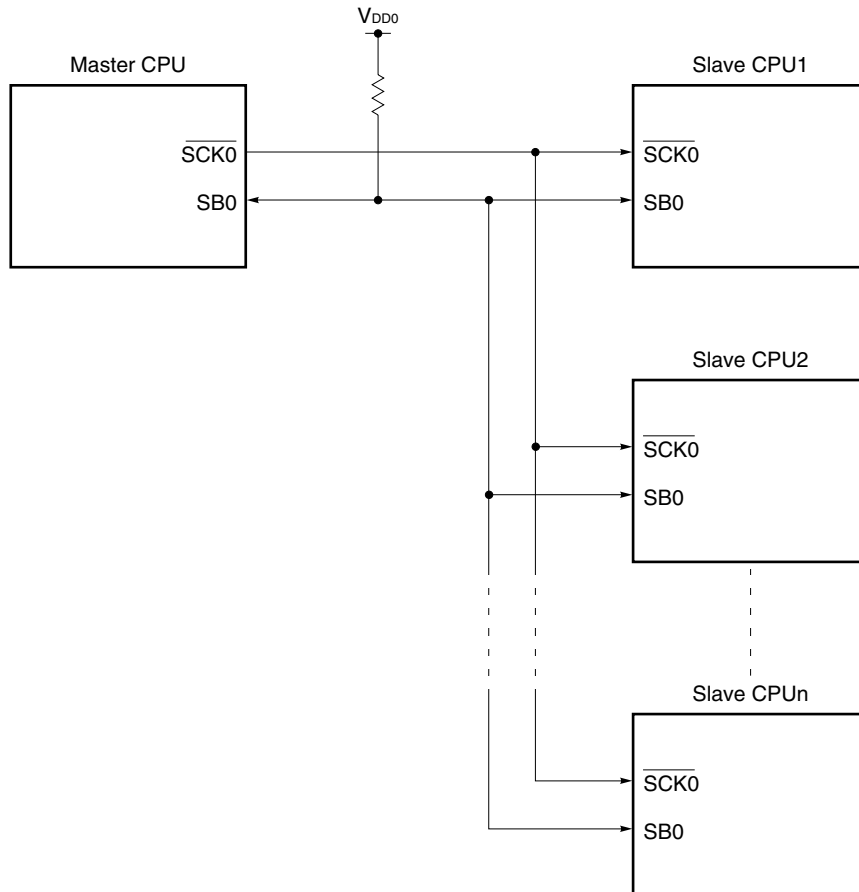
In this mode, the wakeup function for handshake and the output function of acknowledge and busy signals can also be used.

**(4) 2-wire serial I/O mode (MSB-first)**

This mode is used for 8-bit data transfer using the two lines of the serial clock ( $\overline{\text{SCK0}}$ ) and serial data bus (SB0 or SB1).

This mode enables support of any one of the possible data transfer formats by controlling the  $\overline{\text{SCK0}}$  level and the SB0 or SB1 output level. Thus, the handshake line previously necessary for connection of two or more devices can be removed, resulting in an increased number of available I/O ports.

**Figure 16-1. Serial Bus Interface (SBI) System Configuration Example**



## 16.2 Configuration of Serial Interface Channel 0

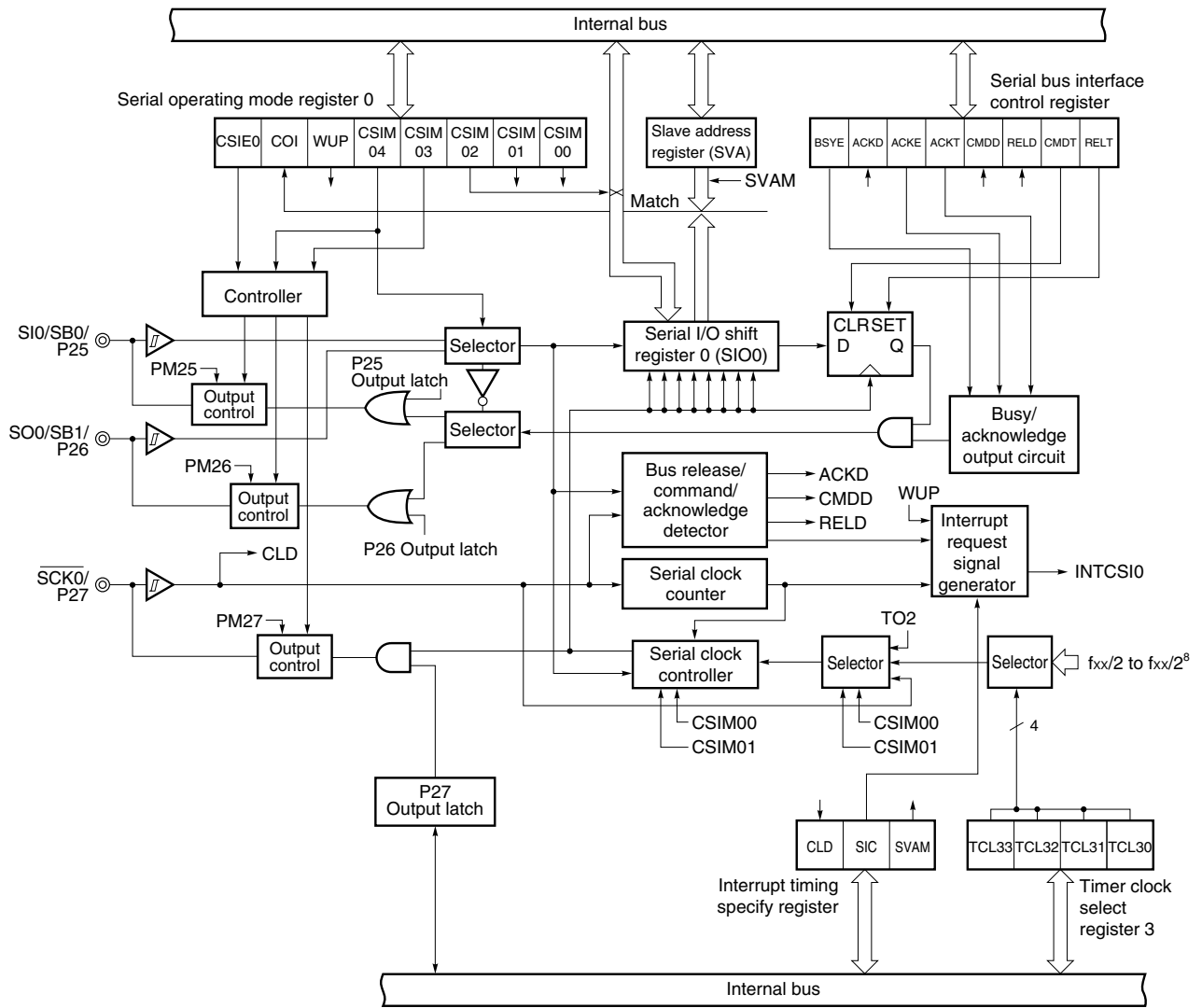
Serial interface channel 0 consists of the following hardware.

**Table 16-2. Configuration of Serial Interface Channel 0**

Item	Configuration
Registers	Serial I/O shift register 0 (SIO0) Slave address register (SVA)
Control registers	Timer clock select register 3 (TCL3) Serial operating mode register 0 (CSIM0) Serial bus interface control register (SBIC) Interrupt timing specify register (SINT) Port mode register 2 (PM2) <sup>Note</sup>

**Note** See **Figure 6-5 Block Diagram of P20, P21, and P23 to P26** and **Figure 6-6 Block Diagram of P22 and P27**.

Figure 16-2. Block Diagram of Serial Interface Channel 0



**Remark** The output control block performs selection between CMOS output and N-ch open-drain output.

**(1) Serial I/O shift register 0 (SIO0)**

SIO0 is an 8-bit register used to carry out parallel/serial conversion and to carry out serial transmission/reception (shift operation) in synchronization with the serial clock.

SIO0 is set with an 8-bit memory manipulation instruction.

When bit 7 (CSIE0) of serial operating mode register 0 (CSIM0) is 1, writing data to SIO0 starts a serial operation.

In transmission, data written to SIO0 is output to the serial output (SO0) or serial data bus (SB0/SB1). In reception, data is read from the serial input (SI0) or SB0/SB1 to SIO0.

Note that, if a bus is driven in the SBI mode or 2-wire serial I/O mode, the bus pins must serve for both input and output. Thus, in the case of a device for reception, write FFH to SIO0 in advance (except when address reception is carried out by setting bit 5 (WUP) of CSIM0 to 1).

In the SBI mode, the busy state can be cleared by writing data to SIO0. In this case, bit 7 (BSYE) of the serial bus interface control register (SBIC) is not cleared to 0.

RESET input makes SIO0 undefined.

**(2) Slave address register (SVA)**

SVA is an 8-bit register used to set the slave address value for connection of a slave device to the serial bus. The SVA is set with an 8-bit memory manipulation instruction. This register is not used in the 3-wire serial I/O mode.

The master device outputs a slave address to the connected slave devices for selection of a particular slave device. These two data (the slave address output from the master device and the SVA value) are compared by an address comparator. If they match, the slave device has been selected. In that case, bit 6 (COI) of serial operating mode register 0 (CSIM0) becomes 1.

Address comparison can also be executed on the data of LSB-masked higher 7 bits by setting bit 4 (SVAM) of the interrupt timing specify register (SINT) to 1.

If no matching is detected in address reception, bit 2 (RELD) of the serial bus interface control register (SBIC) is cleared to 0. In the SBI mode, the wakeup function can be used by setting bit 5 (WUP) of CSIM0 to 1. In this case, the interrupt request signal (INTCSI0) is generated only when the slave address output by the master matches with the SVA value, and it can be learned by this interrupt request that the master requests communication. If bit 5 (SIC) of the interrupt timing specify register (SINT) is set to 1, the wakeup function cannot be used even if WUP is set to 1 (an interrupt request signal is generated when bus release is detected). To use the wakeup function, clear SIC to 0.

Further, an error can be detected by using SVA when the device transmits data as a master or slave device in the SBI or 2-wire serial I/O mode.

RESET input makes SVA undefined.

**(3) SO0 latch**

This latch holds the SI0/SB0/P25 and SO0/SB1/P26 pin levels. It can be directly controlled by software. In the SBI mode, this latch is set upon termination of the 8th serial clock.

**(4) Serial clock counter**

This counter counts the serial clocks to be output and input during transmission/reception to check whether 8-bit data has been transmitted/received.

**(5) Serial clock controller**

This circuit controls serial clock supply to serial I/O shift register 0 (SIO0). When the internal system clock is used, the circuit also controls clock output to the SCK0/P27 pin.

**(6) Interrupt request signal generator**

This circuit controls interrupt request signal generation. It generates an interrupt request signal in the following cases.

**• In the 3-wire serial I/O mode and 2-wire serial I/O mode**

This circuit generates an interrupt request signal every eight serial clocks.

**• In the SBI mode**

When WUP is 0 ..... Generates an interrupt request signal every eight serial clocks.

When WUP is 1 ..... Generates an interrupt request signal when the serial I/O shift register 0 (SIO0) value matches the slave address register (SVA) value after address reception.

**Remark** WUP is the wakeup function specification bit. It is bit 5 of serial operating mode register 0 (CSIM0). When using the wakeup function (WUP = 1), clear bit 5 (SIC) of the interrupt timing specification register (SINT) to 0.

**(7) Busy/acknowledge output circuit and bus release/command/acknowledge detector**

These two circuits output and detect various control signals in the SBI mode.

These do not operate in the 3-wire serial I/O mode and 2-wire serial I/O mode.

### 16.3 Control Registers of Serial Interface Channel 0

The following four registers are used to control serial interface channel 0.

- Timer clock select register 3 (TCL3)
- Serial operating mode register 0 (CSIM0)
- Serial bus interface control register (SBIC)
- Interrupt timing specification register (SINT)

#### (1) Timer clock select register 3 (TCL3)

This register sets the serial clock of serial interface channel 0.

TCL3 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets TCL3 to 88H.



Figure 16-3. Format of Timer Clock Select Register 3

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL3	TCL37	TCL36	TCL35	TCL34	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33				TCL32				TCL31				TCL30				Serial interface channel 0 serial clock selection		
																MCS = 1		MCS = 0
0	1	1	0	$f_{xx}/2$	Setting prohibited		$f_{xx}/2^2$ (1.25 MHz)											
0	1	1	1	$f_{xx}/2^2$	$f_{xx}/2^2$ (1.25 MHz)		$f_{xx}/2^3$ (625 kHz)											
1	0	0	0	$f_{xx}/2^3$	$f_{xx}/2^3$ (625 kHz)		$f_{xx}/2^4$ (313 kHz)											
1	0	0	1	$f_{xx}/2^4$	$f_{xx}/2^4$ (313 kHz)		$f_{xx}/2^5$ (156 kHz)											
1	0	1	0	$f_{xx}/2^5$	$f_{xx}/2^5$ (156 kHz)		$f_{xx}/2^6$ (78.1 kHz)											
1	0	1	1	$f_{xx}/2^6$	$f_{xx}/2^6$ (78.1 kHz)		$f_{xx}/2^7$ (39.1 kHz)											
1	1	0	0	$f_{xx}/2^7$	$f_{xx}/2^7$ (39.1 kHz)		$f_{xx}/2^8$ (19.5 kHz)											
1	1	0	1	$f_{xx}/2^8$	$f_{xx}/2^8$ (19.5 kHz)		$f_{xx}/2^9$ (9.8 kHz)											
Other than above				Setting prohibited														

TCL37				TCL36				TCL35				TCL34				Serial interface channel 1 serial clock selection		
																MCS = 1		MCS = 0
0	1	1	0	$f_{xx}/2$	Setting prohibited		$f_{xx}/2^2$ (1.25 MHz)											
0	1	1	1	$f_{xx}/2^2$	$f_{xx}/2^2$ (1.25 MHz)		$f_{xx}/2^3$ (625 kHz)											
1	0	0	0	$f_{xx}/2^3$	$f_{xx}/2^3$ (625 kHz)		$f_{xx}/2^4$ (313 kHz)											
1	0	0	1	$f_{xx}/2^4$	$f_{xx}/2^4$ (313 kHz)		$f_{xx}/2^5$ (156 kHz)											
1	0	1	0	$f_{xx}/2^5$	$f_{xx}/2^5$ (156 kHz)		$f_{xx}/2^6$ (78.1 kHz)											
1	0	1	1	$f_{xx}/2^6$	$f_{xx}/2^6$ (78.1 kHz)		$f_{xx}/2^7$ (39.1 kHz)											
1	1	0	0	$f_{xx}/2^7$	$f_{xx}/2^7$ (39.1 kHz)		$f_{xx}/2^8$ (19.5 kHz)											
1	1	0	1	$f_{xx}/2^8$	$f_{xx}/2^8$ (19.5 kHz)		$f_{xx}/2^9$ (9.8 kHz)											
Other than above				Setting prohibited														

**Caution** When rewriting TCL3 to other data, stop the serial transfer operation beforehand.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

**(2) Serial operating mode register 0 (CSIM0)**

This register sets the serial interface channel 0 serial clock, operating mode, operation enable/stop wakeup function and displays the address comparator match signal.

CSIM0 is set with a 1-bit or an 8-bit memory manipulation instruction.

RESET input clears CSIM0 to 00H.

**Caution** Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or SBI) while serial interface channel 0 is enabled to operate. To change the operating mode, stop the serial operation first.

**Figure 16-4. Format of Serial Operating Mode Register 0 (1/2)**

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection
	0	×	Input clock to SCK0 pin from off-chip
	1	0	8-bit timer register 2 (TM2) output
	1	1	Clock specified by bits 0 to 3 of timer clock select register 3 (TCL3)

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SI0/SB0/P25 pin function	SO0/SB1/P26 pin function	SCK0/P27 pin function
	0	×	0	Note 2	Note 2	0	0	0	1	3-wire serial I/O mode	MSB	SI0 <sup>Note 2</sup> (input)	SO0 (CMOS output)	SCK0 (CMOS I/O)
			1	×	×						LSB			
	1	0	0	Note 3	Note 3	0	0	0	1	SBI mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	SCK0 (CMOS I/O)
				1	0							0	Note 3	
	1	1	0	Note 3	Note 3	0	0	0	1	2-wire serial I/O mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	SCK0 (N-ch open-drain I/O)
				1	0							0	Note 3	

(Cont'd)

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. Can be used as P25 (CMOS I/O) when used only for transmission.
  3. Can be used freely as a port function.

**Remark**

- ×: don't care
- PMxx: Port mode register
- Pxx: Port output latch

Figure 16-4. Format of Serial Operating Mode Register 0 (2/2)

R/W	WUP	Wakeup function control <sup>Note 1</sup>
	0	Interrupt request signal generation with each serial transfer in any mode
	1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register (SVA) data in SBI mode
R	COI	Slave address comparison result flag <sup>Note 2</sup>
	0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data
	1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data
R/W	CSIE0	Serial interface channel 0 operation control
	0	Operation stopped
	1	Operation enabled

- Notes**
1. When using the wakeup function (WUP = 1), clear bit 5 (SIC) of the interrupt timing specification register (SINT) to 0.
  2. When CSIE0 = 0, COI becomes 0.

**(3) Serial bus interface control register (SBIC)**

This register sets the serial bus interface operation and displays statuses.

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears SBIC to 00H.

**Figure 16-5. Format of Serial Bus Interface Control Register (1/2)**

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W <sup>Note</sup>
R/W	RELT	Used for bus release signal output. When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, is RELT automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R/W	CMDT	Used for command signal output. When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R	RELD	Bus release detection									
		Clear conditions (RELD = 0)					Set conditions (RELD = 1)				
		<ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• If SIO0 and SVA values do not match in address reception</li> <li>• When CSIE0 = 0</li> <li>• When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>					<ul style="list-style-type: none"> <li>• When bus release signal (REL) is detected</li> </ul>				
R	CMDD	Command detection									
		Clear conditions (CMDD = 0)					Set conditions (CMDD = 1)				
		<ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• When bus release signal (REL) is detected</li> <li>• When CSIE0 = 0</li> <li>• When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>					<ul style="list-style-type: none"> <li>• When command signal (CMD) is detected</li> </ul>				
R/W	ACKT	The acknowledge signal is output in synchronization with the falling edge of the $\overline{\text{SCK0}}$ clock just after execution of the instruction that sets this bit to 1, and after acknowledge signal output, ACKT is automatically cleared to 0. ACKT is used with ACKE = 0. ACKT is also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0.									

**Note** Bits 2, 3, and 6 (RELD, CMDD and ACKD) are read-only bits.

**Remarks 1.** Bits 0, 1, and 4 (RELT, CMDT, and ACKT) are 0 when read after data setting.

**2.** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

Figure 16-5. Format of Serial Bus Interface Control Register (2/2)

R/W	ACKE	Acknowledge signal automatic output control	
	0	Acknowledge signal automatic output disable (output by ACKT enabled)	
	1	Before completion of transfer	The acknowledge signal is output in synchronization with the falling edge of the 9th $\overline{\text{SCK0}}$ clock (automatically output when ACKE = 1).
		After completion of transfer	The acknowledge signal is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ just after execution of the instruction that sets this bit to 1 (automatically output when ACKE = 1). However, ACKE is not automatically cleared to 0 after acknowledge signal is output.
R	ACKD	Acknowledge detection	
		Clear conditions (ACKD = 0)	Set conditions (ACKD = 1)
		<ul style="list-style-type: none"> <li>Falling edge of <math>\overline{\text{SCK0}}</math> immediately after busy mode is released after executing the transfer start instruction</li> <li>When CSIE0 = 0</li> <li>When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>	<ul style="list-style-type: none"> <li>When acknowledge signal (<math>\overline{\text{ACK}}</math>) is detected at the rising edge of the <math>\overline{\text{SCK0}}</math> clock after completion of transfer</li> </ul>
R/W	Note BSYE	Synchronizing busy signal output control	
	0	Disable the busy signal which is output in synchronization with the falling edge the $\overline{\text{SCK0}}$ clock just after execution of the instruction that clears this bit to 0.	
	1	Output the busy signal at the falling edge of the $\overline{\text{SCK0}}$ clock following the acknowledge signal.	

**Note** The busy mode can be cleared by start of serial interface transfer. However, the BSYE flag is not cleared to 0.

**Remark** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

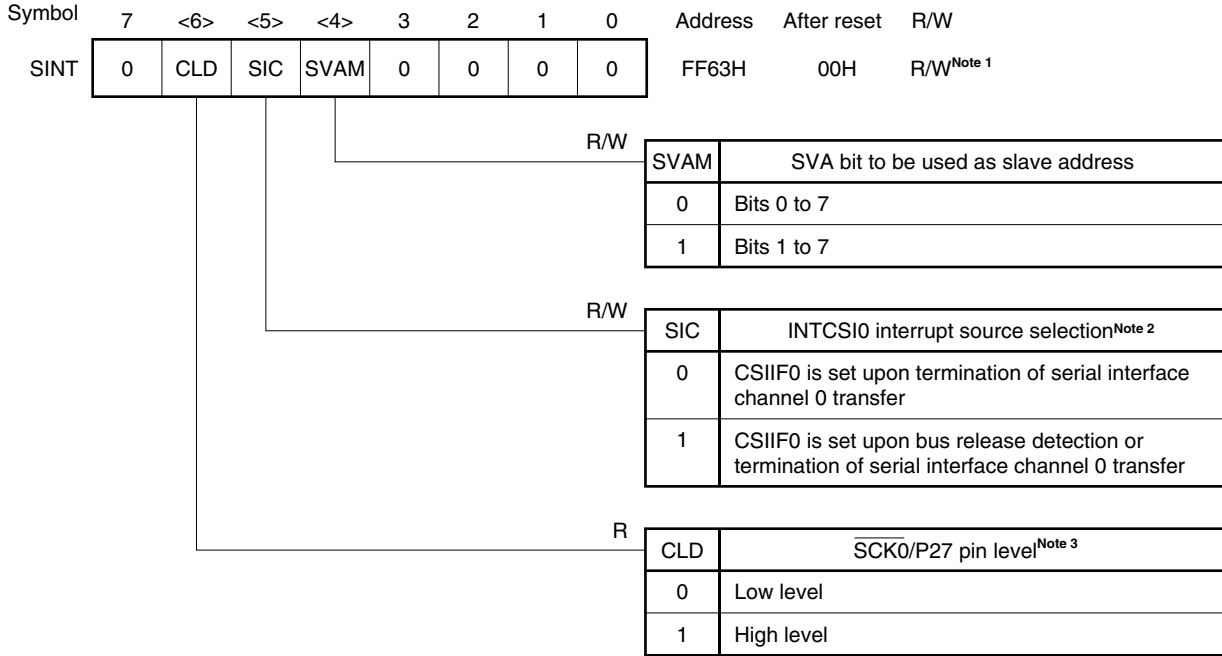
**(4) Interrupt timing specification register (SINT)**

This register sets the bus release interrupt and address mask functions and displays the  $\overline{\text{SCK0/P27}}$  pin level status.

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SINT to 00H.

**Figure 16-6. Format of Interrupt Timing Specification Register**



- Notes**
1. Bit 6 (CLD) is a read-only bit.
  2. When using wakeup function in the SBI mode, clear SIC to 0.
  3. When CSIE0 = 0, CLD becomes 0.

**Caution** Be sure to clear bits 0 to 3 to 0.

**Remark** SVA: Slave address register  
 CSIIF0: Interrupt request flag corresponding to INTCSI0  
 CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

## 16.4 Operations of Serial Interface Channel 0

The following four operating modes are available for serial interface channel 0.

- Operation stop mode
- 3-wire serial I/O mode
- SBI mode
- 2-wire serial I/O mode

### 16.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power consumption can be reduced. Serial I/O shift register 0 (SIO0) does not carry out shift operations either and thus it can be used as an ordinary 8-bit register.

In the operation stop mode, the P25/SI0/SB0, P26/SO0/SB1, and P27/ $\overline{\text{SCK0}}$  pins can be used as ordinary I/O ports.

#### (1) Register setting

The operation stop mode is set by serial operating mode register 0 (CSIM0).

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W

R/W	CSIE0	Serial interface channel 0 operation control
	0	Operation stopped
	1	Operation enabled

### 16.4.2 3-wire serial I/O mode operation

The 3-wire serial I/O mode is valid for connection of peripheral I/O units and display controllers which incorporate a conventional clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

Communication is carried out with the three lines of the serial clock ( $\overline{\text{SCK0}}$ ), serial output (SO0), and serial input (SI0).

#### (1) Register setting

The 3-wire serial I/O mode is set by serial operating mode register 0 (CSIM0) and the serial bus interface control register (SBIC).

##### (a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM0 to 00H.



Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock to SCK0 pin from off-chip								
	1	0	8-bit timer register 2 (TM2) output								
	1	1	Clock specified by bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SIO/SB0/P25 pin function	SO0/SB1/P26 pin function	SCK0/P27 pin function
	0	×	0	Note 2 1	Note 2 ×	0	0	0	1	3-wire serial I/O mode	MSB	SIO <sup>Note 2</sup> (input)	SO0 (CMOS output)	SCK0 (CMOS I/O)
			1								LSB			
	1	0	SBI mode (see 16.4.3 SBI mode operation.)											
	1	1	2-wire serial I/O mode (see 16.4.4 2-wire serial I/O mode operation.)											

R/W	WUP	Wakeup function control <sup>Note 3</sup>									
	0	Interrupt request signal generation with each serial transfer in any mode									
	1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register data in SBI mode									

R/W	CSIE0	Serial interface channel 0 operation control									
	0	Operation stopped									
	1	Operation enabled									

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. Can be used as P25 (CMOS input/output) when used only for transmission.
  3. Be sure to clear WUP to 0 when the 3-wire serial I/O mode is selected.

**Remark**

- ×: don't care
- PM××: Port mode register
- P××: Port output latch

**(b) Serial bus interface control register (SBIC)**

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	RELT	When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

R/W	CMDT	When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

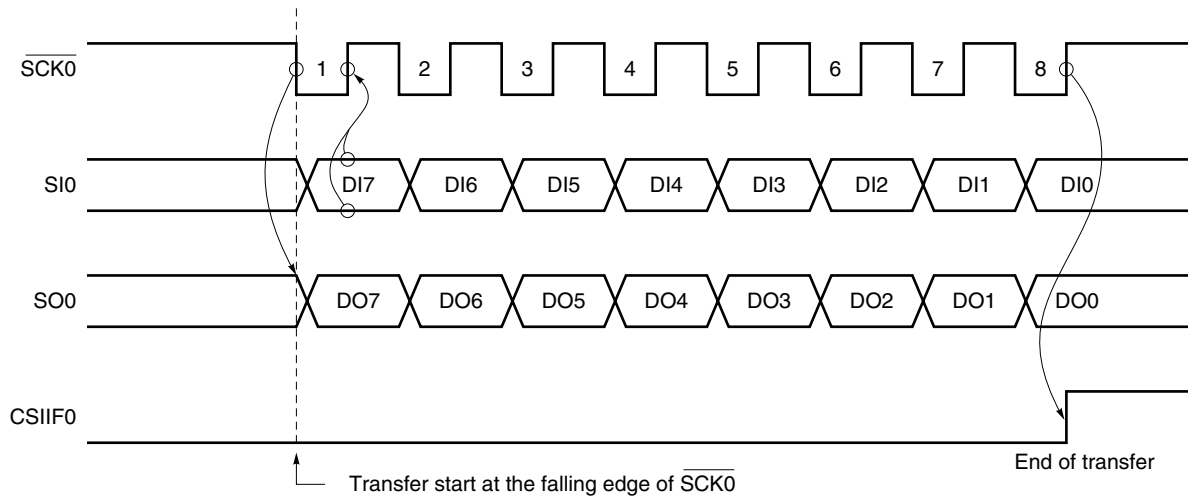
**(2) Communication operation**

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

Shift operations of serial I/O shift register 0 (SIO0) are carried out at the falling edge of the serial clock ( $\overline{\text{SCK0}}$ ). The transmitted data is held in the SO0 latch and is output from the SO0 pin. The received data input to the SIO pin is latched in SIO0 at the rising edge of  $\overline{\text{SCK0}}$ .

Upon termination of 8-bit transfer, SIO0 operation stops automatically and the interrupt request flag (CSIF0) is set.

**Figure 16-7. 3-Wire Serial I/O Mode Timing**



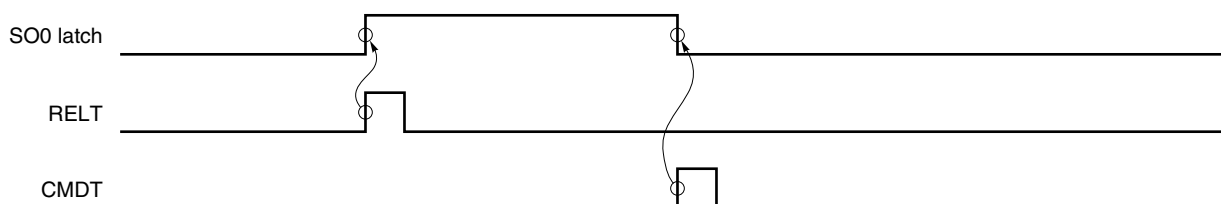
The SO0 pin is a CMOS output pin and outputs the current SO0 latch status. Thus, the SO0 pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the  $\overline{\text{SCK0}}$  pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (see **16.4.5  $\overline{\text{SCK0}}$ /P27 pin output manipulation**).

**(3) Other signals**

Figure 16-8 shows the RELT and CMDT operations.

**Figure 16-8. RELT and CMDT Operations**



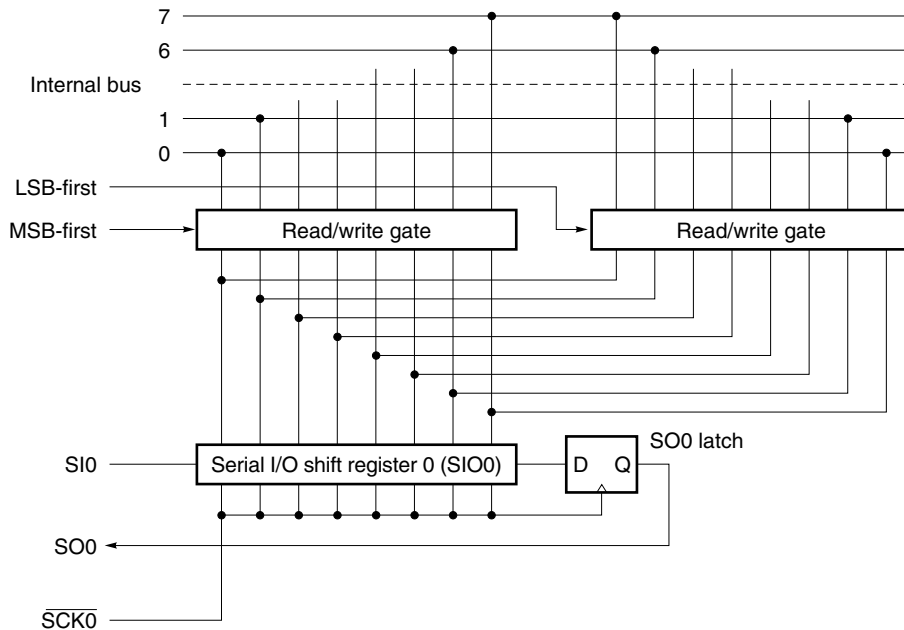
**(4) MSB/LSB switching as the start bit**

In the 3-wire serial I/O mode, it is possible to select transfer to start from the MSB or LSB.

Figure 16-9 shows the configuration of serial I/O shift register 0 (SIO0) and the internal bus. As shown in the figure, the MSB/LSB can be read or written in reverse form.

MSB/LSB switching as the start bit can be specified by bit 2 (CSIM02) of serial operating mode register 0 (CSIM0).

**Figure 16-9. Circuit for Switching Transfer Bit Order**



Start bit switching is realized by switching the bit order for data write to SIO0. The SIO0 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to SIO0.

**(5) Transfer start**

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1.
- Internal serial clock is stopped or  $\overline{\text{SCK0}}$  is a high level after 8-bit serial transfer.

**Caution** If CSIE0 is set to 1 after data write to SIO0, transfer does not start.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

### 16.4.3 SBI mode operation

SBI (Serial Bus Interface) is a high-speed serial interface that complies with the NEC Electronics serial bus format.

SBI uses a single master device and employs a clocked serial I/O format with the addition of a bus configuration function. This function enables devices to communicate using only two lines. Thus, when making up a serial bus with two or more microcontrollers and peripheral ICs, the number of ports to be used and the number of wires on the board can be decreased.

The master device outputs three kinds of data to slave devices on the serial data bus: “addresses” to select a device to be communicated with, “commands” to instruct the selected device, and “data” which is actually required.

The slave device can identify the received data as “address”, “command”, or “data” by hardware. An application program that controls serial interface channel 0 can be simplified by using this function.

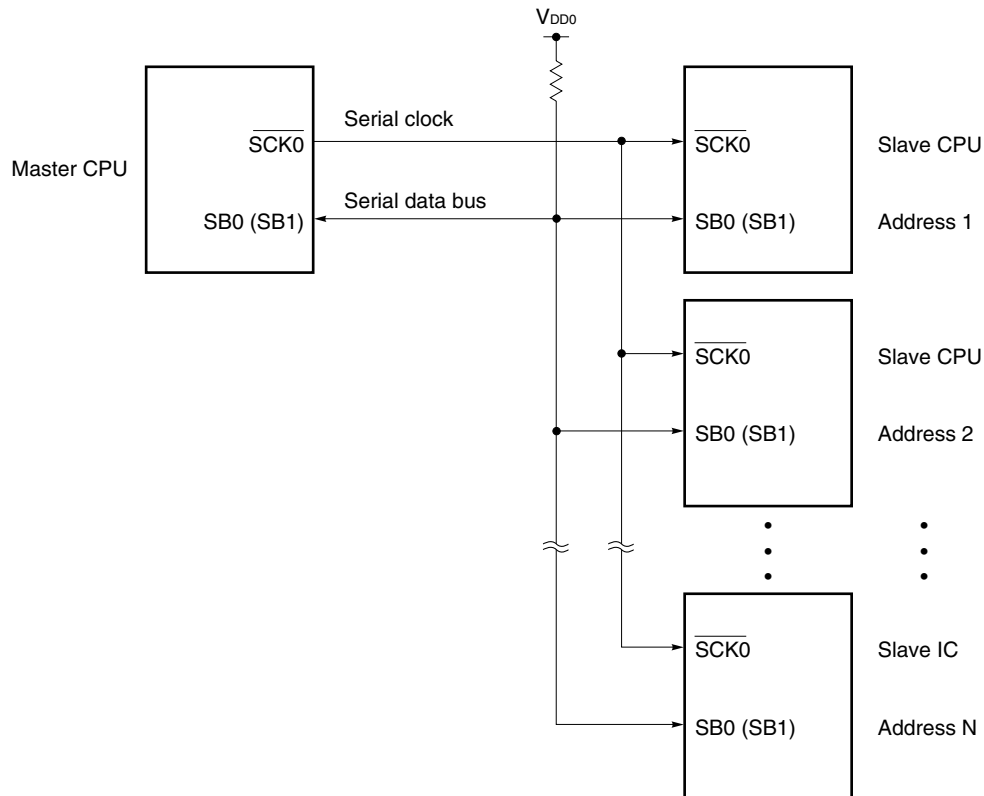
The SBI function is incorporated into various devices including the 75X/XL Series and 78K Series.

Figure 16-10 shows a serial bus configuration example when a CPU having a serial interface compliant with SBI and peripheral ICs are used.

In SBI, the SB0 (SB1) serial data bus pin is an open-drain output pin and therefore the serial data bus line behaves in the same way as a wired-OR configuration. In addition, a pull-up resistor must be connected to the serial data bus line.

When the SBI mode is used, see (11) **SBI mode precautions (d)** described later.

**Figure 16-10. Example of Serial Bus Configuration with SBI**



**Caution** When exchanging the master CPU/slave CPU, a pull-up resistor is necessary for the serial clock line ( $\overline{\text{SCK0}}$ ) as well because serial clock line ( $\overline{\text{SCK0}}$ ) input/output switching is carried out asynchronously between the master and slave CPUs.

**(1) SBI functions**

In the conventional serial I/O format, when a serial bus is configured by connecting two or more devices, many ports and wiring are necessary to provide chip select signals to identify commands and data, and to judge the busy state, because only the data transfer function is available. If these operations are to be controlled by software, the software load becomes very heavy.

In SBI, a serial bus can be configured with the two signal lines of the serial clock  $\overline{\text{SCK0}}$  and serial data bus SB0 (SB1). Thus, use of SBI leads to a reduction in the number of microcontroller ports and the amount of wiring and routing on the board.

The SBI functions are described below.

**(a) Address/command/data identification function**

Serial data is distinguished as addresses, commands, and data.

**(b) Chip select function by address transmission**

The master executes slave chip selection by address transmission.

**(c) Wakeup function**

The slave can easily judge address reception (chip select judgment) using the wakeup function (which can be set/reset by software).

When the wakeup function is set, the interrupt request signal (INTCSI0) is generated upon reception of a match address.

Thus, when communication is executed with two or more devices, the CPU except the selected slave device can operate regardless of serial communications.

**(d) Acknowledge signal ( $\overline{\text{ACK}}$ ) control function**

The acknowledge signal used to check serial data reception is controlled.

**(e) Busy signal ( $\overline{\text{BUSY}}$ ) control function**

The busy signal used to report the slave busy state is controlled.

**(2) SBI definition**

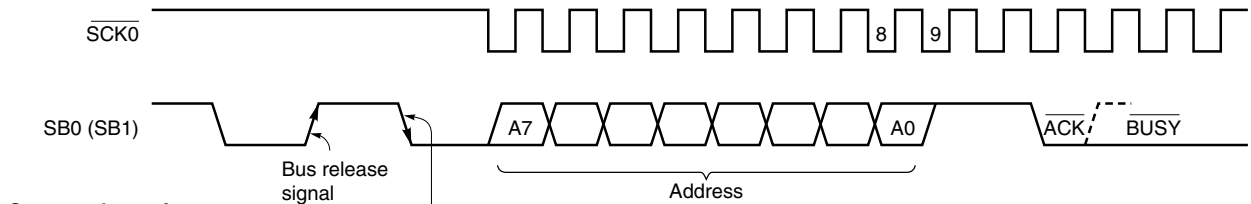
The SBI serial data format and the signals to be used are defined as follows.

Serial data to be transferred by SBI consists of three kinds of data: “address”, “command”, and “data”.

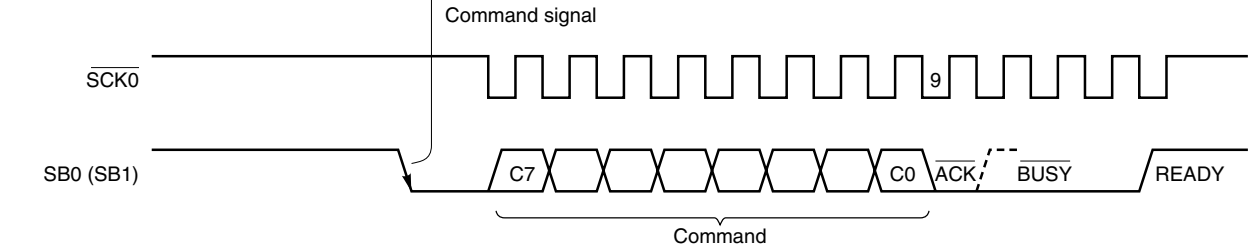
Figure 16-11 shows the address, command, and data transfer timing.

**Figure 16-11. SBI Transfer Timing**

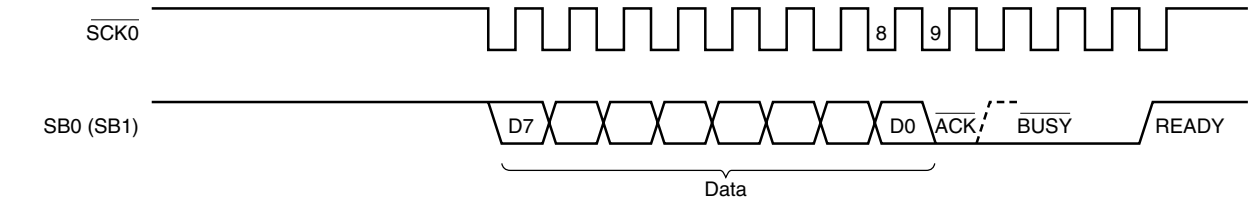
**Address transfer**



**Command transfer**



**Data transfer**



**Remark** The broken lines indicate the READY status.

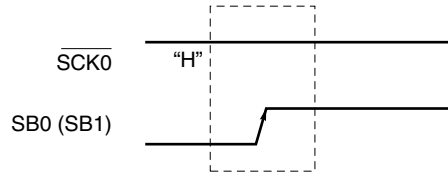
The bus release signal and the command signal are output by the master device.  $\overline{BUSY}$  is output by the slave device.  $\overline{ACK}$  can be output by either the master or slave device (normally, the 8-bit data receiver outputs). Serial clocks continue to be output by the master device from 8-bit data transfer start to  $\overline{BUSY}$  reset.

**(a) Bus release signal (REL)**

The bus release signal is recognized when the SB0 (SB1) line changes from low level to high level when the  $\overline{\text{SCK0}}$  line is at high level (without serial clock output).

This signal is output by the master device.

**Figure 16-12. Bus Release Signal**



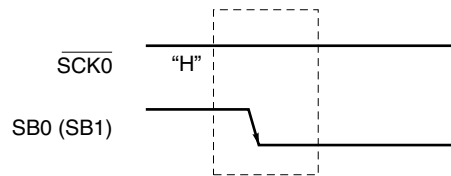
The bus release signal indicates that the master device is going to transmit an address to the slave device. The slave device incorporates hardware to detect the bus release signal.

**Caution** The transition of the SB0 (SB1) line from low to high when the  $\overline{\text{SCK0}}$  line is high is recognized as a bus release signal. If the transition timing of the bus is shifted due to the influence of board capacitance, transmitted data may be judged as a bus release signal. Exercise care in wiring so that noise is not superimposed on the signal lines.

**(b) Command signal (CMD)**

The command signal is recognized when the SB0 (SB1) line changes from high level to low level when the  $\overline{\text{SCK0}}$  line is at high level (without serial clock output). This signal is output by the master device.

**Figure 16-13. Command Signal**



The command signal indicates that the master is going to transmit a command to the slave (however, a command signal following a bus release signal indicates that an address is transmitted).

The slave device incorporates hardware to detect the command signal.

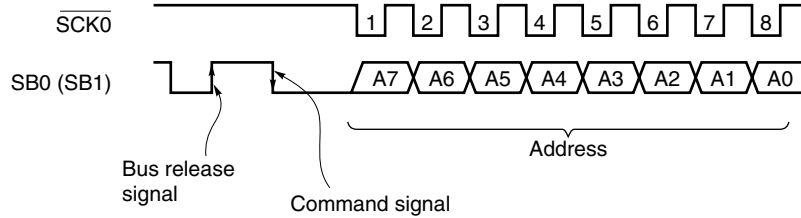
**Caution** The transition of the SB0 (SB1) line from high to low when the  $\overline{\text{SCK0}}$  line is high is recognized as a command signal. If the transition timing of the bus is shifted due to the influence of board capacitance, transmitted data may be judged as a command signal. Exercise care in wiring so that noise is not superimposed on the signal lines.



**(c) Address**

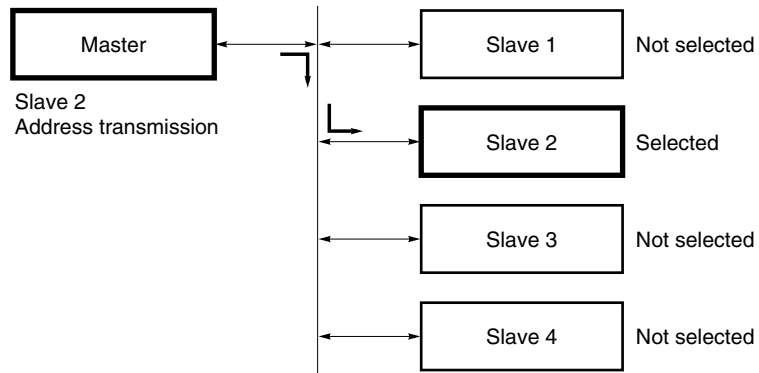
An address is 8-bit data which the master device outputs to the slave devices connected to the bus line in order to select a particular slave device.

**Figure 16-14. Addresses**



8-bit data following bus release and command signals is defined as an “address”. In the slave device, this condition is detected by hardware and whether or not 8-bit data matches the own specification number (slave address) is checked by hardware. If the 8-bit data matches the slave address, the slave device has been selected. After that, communication with the master device continues until a release instruction is received from the master device.

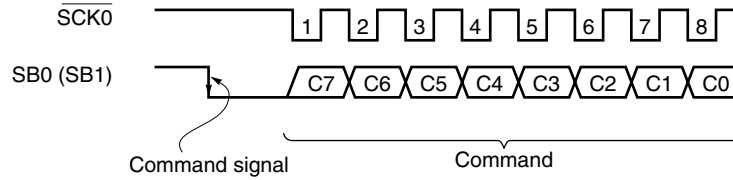
**Figure 16-15. Slave Selection by Address**



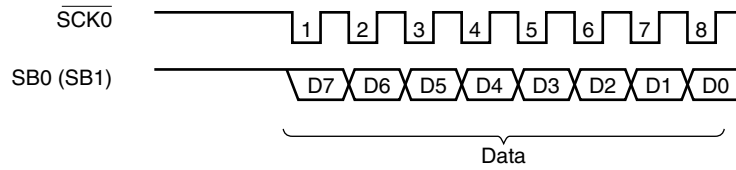
**(d) Commands and data**

The master device transmits commands to, and transmits/receives data to/from the slave device selected by address transmission.

**Figure 16-16. Commands**



**Figure 16-17. Data**

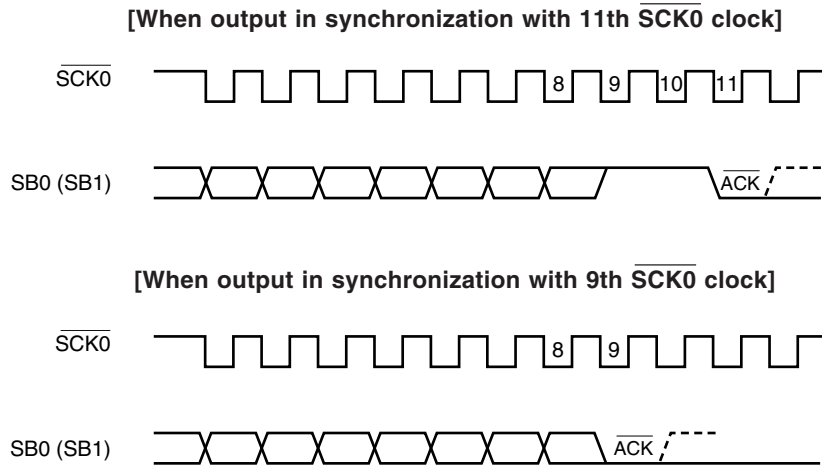


8-bit data following a command signal is defined as “command” data. 8-bit data without a command signal is defined as “data”. Command and data operation procedures can be determined by the user according to their communication specifications.

**(e) Acknowledge signal ( $\overline{\text{ACK}}$ )**

The acknowledge signal is used to check serial data reception between the transmitter and receiver.

**Figure 16-18. Acknowledge Signal**



**Remark** The broken lines indicate the READY status.

The acknowledge signal is one-shot pulse generated at the falling edge of  $\overline{\text{SCK0}}$  after 8-bit data transfer. It can be positioned anywhere and can be synchronized with any  $\overline{\text{SCK0}}$  clock.

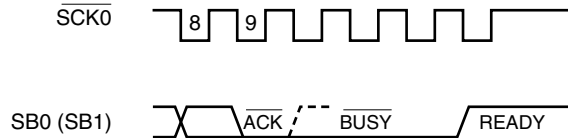
After 8-bit data transmission, the transmitter checks whether the receiver has returned the acknowledge signal. If the acknowledge signal is not returned for the preset period of time after data transmission, it can be judged that data reception has not been carried out correctly.

**(f) Busy signal ( $\overline{\text{BUSY}}$ ) and ready signal (READY)**

The  $\overline{\text{BUSY}}$  signal is used to report to the master device that the slave device is not ready for data transmission/reception.

The READY signal is used to report to the master device that the slave device is ready for data transmission/reception.

**Figure 16-19.  $\overline{\text{BUSY}}$  and READY Signals**



In SBI, the slave device notifies the master device of the busy state by setting the SB0 (SB1) line to low level.

$\overline{\text{BUSY}}$  signal output follows acknowledge signal output from the master or slave device. It is set/reset at the falling edge of  $\overline{\text{SCK0}}$ . When the  $\overline{\text{BUSY}}$  signal is reset, the master device automatically terminates the output of the  $\overline{\text{SCK0}}$  serial clock.

When the  $\overline{\text{BUSY}}$  signal is reset and the READY signal is set, the master device can start the next transfer.

**Caution** In the SBI mode, the  $\overline{\text{BUSY}}$  signal is output until the next serial clock ( $\overline{\text{SCK0}}$ ) falls after a command that resets the  $\overline{\text{BUSY}}$  signal has been issued. If WUP is set to 1 during this period by mistake, the  $\overline{\text{BUSY}}$  signal is not reset. Therefore, be sure to confirm that the SB0 (SB1) pin has gone high after resetting the  $\overline{\text{BUSY}}$  signal, by setting WUP to 1.

**(3) Register setting**

The SBI mode is set by serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specification register (SINT).

**(a) Serial operating mode register 0 (CSIM0)**

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock to SCK0 pin from off-chip								
	1	0	8-bit timer register 2 (TM2) output								
	1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SIO/SB0/P25 pin function	SO0/SB1/P26 pin function	$\overline{\text{SCK0}}$ /P27 pin function
	0	×	3-wire serial I/O mode (see 16.4.2 3-wire serial I/O mode operation.)											
	1	0	0	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	0	0	0	1	SBI mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ (CMOS I/O)
			1	0	0	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	0	1			SB0 (N-ch open-drain I/O)	P26 (CMOS I/O)	
	1	1	2-wire serial I/O mode (see 16.4.4 2-wire serial I/O mode operation.)											

R/W	WUP	Wakeup function control <sup>Note 3</sup>									
	0	Interrupt request signal generation with each serial transfer in any mode									
	1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register (SVA) data in SBI mode									

R	COI	Slave address comparison result flag <sup>Note 4</sup>									
	0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data									
	1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data									

R/W	CSIE0	Serial interface channel 0 operation control									
	0	Operation stopped									
	1	Operation enabled									

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. Can be used as a port function.
  3. When using the wakeup function (WUP = 1), clear bit 5 (SIC) of the interrupt timing specification register (SINT) to 0.
  4. When CSIE0 = 0, COI becomes 0.

**Remark**

- ×: don't care
- PMxx: Port mode register
- Pxx: Port output latch

**(b) Serial bus interface control register (SBIC)**

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W <sup>Note</sup>
R/W	RELT	Used for bus release signal output. When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R/W	CMDT	Used for command signal output. When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R	RELD	Bus release detection									
		Clear conditions (RELD = 0)					Set conditions (RELD = 1)				
		<ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• If SIO0 and SVA values do not match in address reception (only when WUP = 1)</li> <li>• When <math>\overline{\text{CSIE0}} = 0</math></li> <li>• When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>					<ul style="list-style-type: none"> <li>• When bus release signal (REL) is detected</li> </ul>				
R	CMDD	Command detection									
		Clear conditions (CMDD = 0)					Set conditions (CMDD = 1)				
		<ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• When bus release signal (REL) is detected</li> <li>• When <math>\overline{\text{CSIE0}} = 0</math></li> <li>• When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>					<ul style="list-style-type: none"> <li>• When command signal (CMD) is detected</li> </ul>				
R/W	ACKT	The acknowledge signal is output in synchronization with the falling edge of the $\overline{\text{SCK0}}$ clock just after execution of the instruction that sets this bit to 1 and, after acknowledge signal output, ACKT is automatically cleared to 0. ACKT is used with ACEK = 0. ACKT is also cleared to 0 upon start of serial interface transfer or when $\overline{\text{CSIE0}} = 0$ .									
R/W	ACKE	Acknowledge signal automatic output control									
	0	Acknowledge signal automatic output disable (output by ACKT enabled)									
	1	Before completion of transfer	The acknowledge signal is output in synchronization with the falling edge of the 9th $\overline{\text{SCK0}}$ clock (automatically output when ACEK = 1).								
		After completion of transfer	The acknowledge signal is output in synchronization with falling edge of the $\overline{\text{SCK0}}$ clock just after execution of the instruction that sets this bit to 1 (automatically output when ACEK = 1). However, ACEK is not automatically cleared to 0 after acknowledge signal output.								

(Cont'd)

**Note** Bits 2, 3, and 6 (RELD, CMDD and ACKD) are read-only bits.

**Remarks 1.** Bits 0, 1, and 4 (RELT, CMDT, and ACKT) are 0 when read after data setting.

**2.** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

R	ACKD	Acknowledge detection	
	Clear conditions (ACKD = 0)		Set conditions (ACKD = 1)
	<ul style="list-style-type: none"> <li>When <math>\overline{SCK0}</math> falls immediately after busy mode is released after transfer start instruction execution.</li> <li>When CSIE0 = 0</li> <li>When RESET input is applied</li> </ul>		<ul style="list-style-type: none"> <li>When the acknowledge signal (<math>\overline{ACK}</math>) is detected at the rising edge of the <math>\overline{SCK0}</math> clock after completion of transfer</li> </ul>

R/W	Note BSYE	Synchronizing busy signal output control	
	0	Disable the busy signal which is output in synchronization with the falling edge of $\overline{SCK0}$ clock just after execution of the instruction that clears this bit to 0 (set READY status).	
	1	Output the busy signal at the falling edge of the $\overline{SCK0}$ clock following the acknowledge signal.	

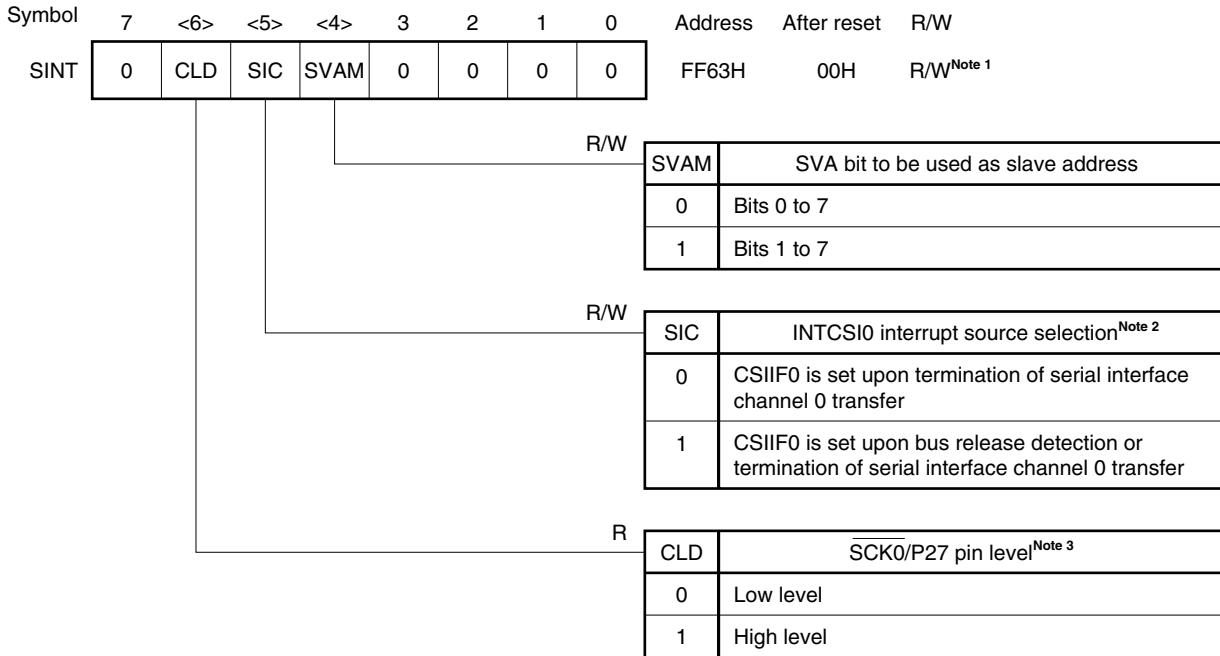
**Note** Busy mode can be cleared by start of serial interface transfer. However, the BSYE flag is not cleared to 0.

**Remark** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

**(c) Interrupt timing specification register (SINT)**

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SINT to 00H.



**Caution** Be sure to clear bits 0 to 3 to 0.

- Notes**
1. Bit 6 (CLD) is a read-only bit.
  2. When using wakeup function in the SBI mode, clear SIC to 0.
  3. When CSIE0 = 0, CLD becomes 0.

**Remark** SVA: Slave address register  
 CSIF0: Interrupt request flag corresponding to INTCSI0  
 CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)



(4) Various signals

Figures 16-20 to 16-25 show various signals and flag operations in SBI. Table 16-3 lists various signals in SBI.

Figure 16-20. RELT, CMDT, RELD, and CMDD Operations (Master)

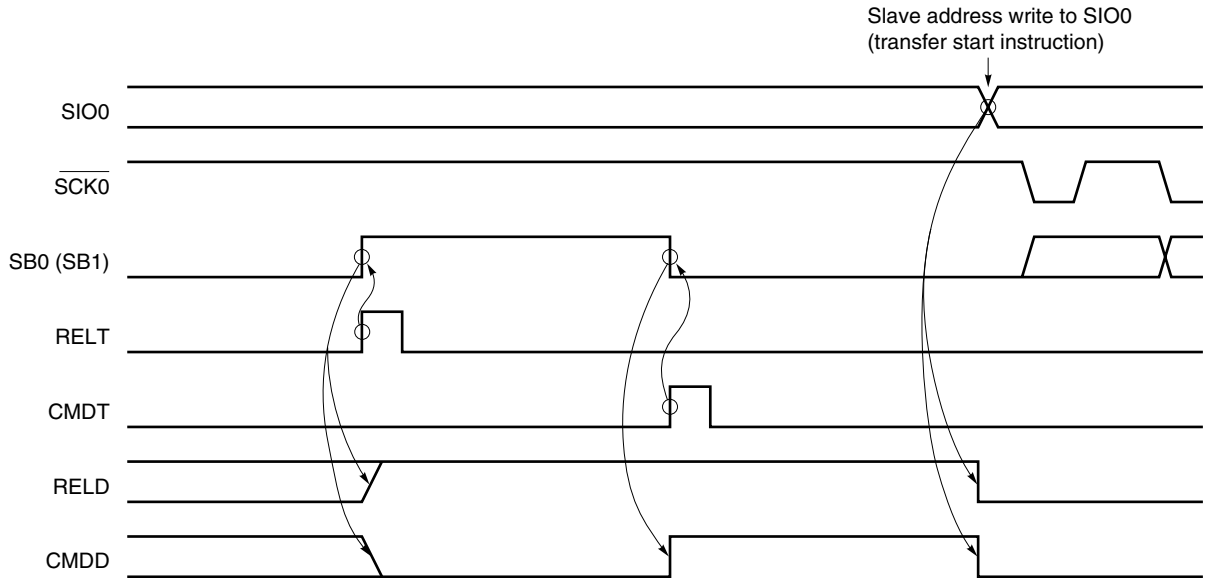


Figure 16-21. RELD and CMDD Operations (Slave)

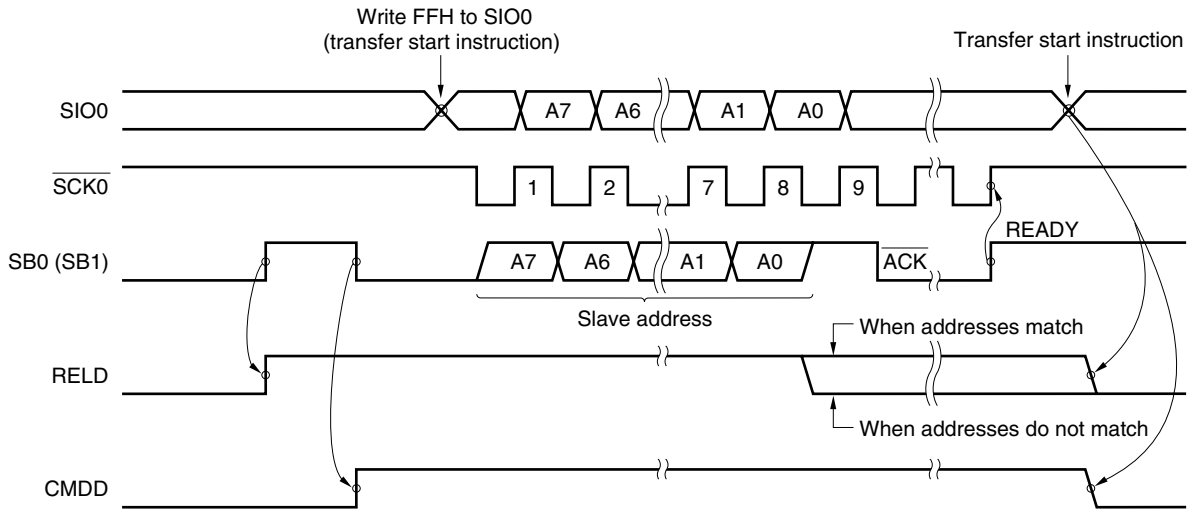
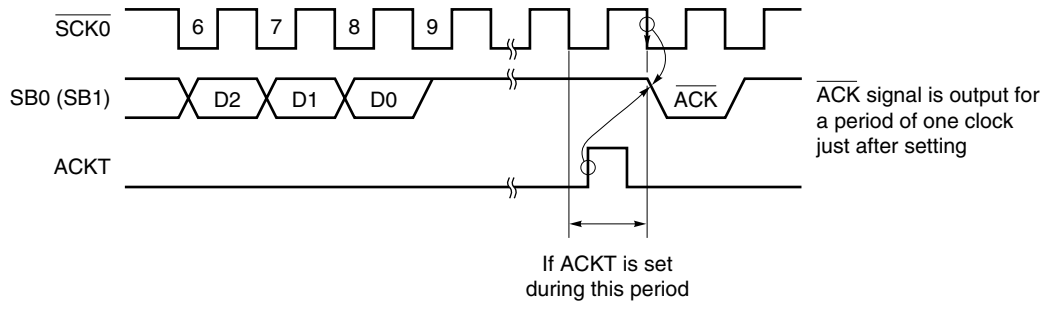


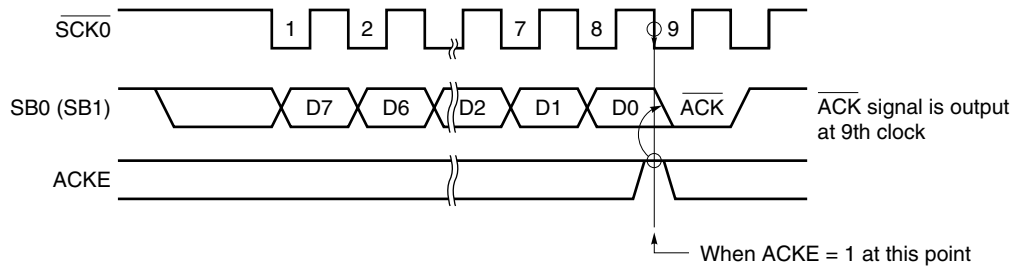
Figure 16-22. ACKT Operation



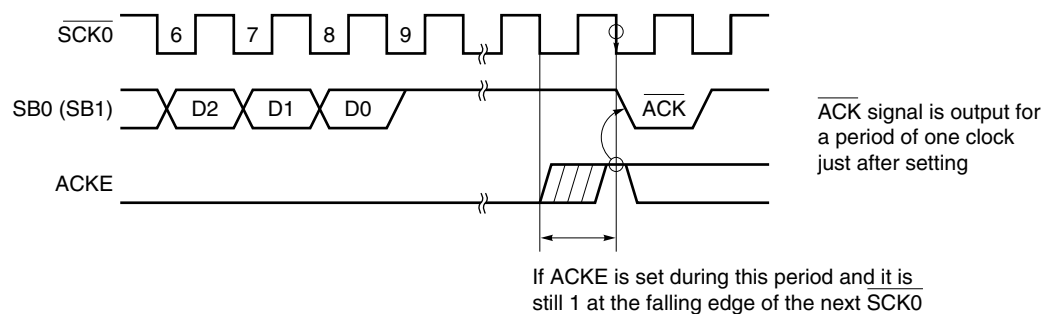
**Caution** Do not set ACKT before completion of transfer.

Figure 16-23. ACKE Operations

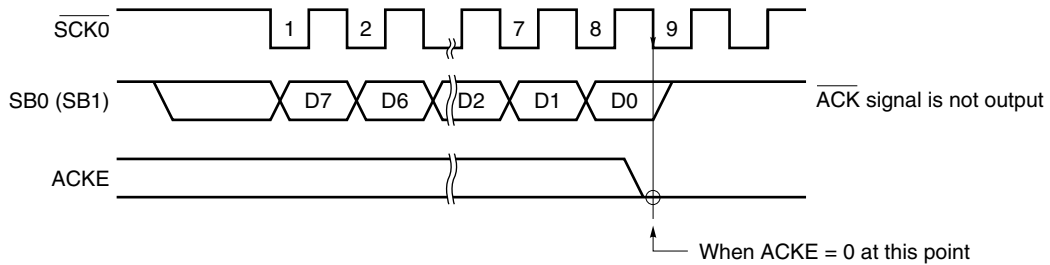
(a) When ACKE = 1 upon completion of transfer



(b) When set after completion of transfer



(c) When ACKE = 0 upon completion of transfer



(d) When ACKE = 1 period is short

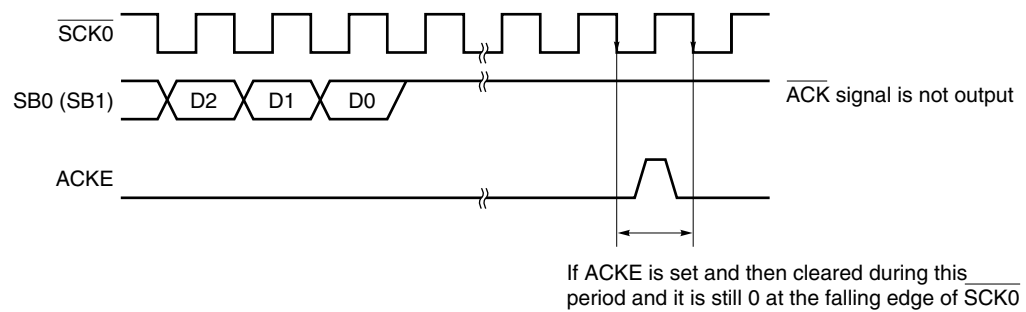


Figure 16-24. ACKD Operations

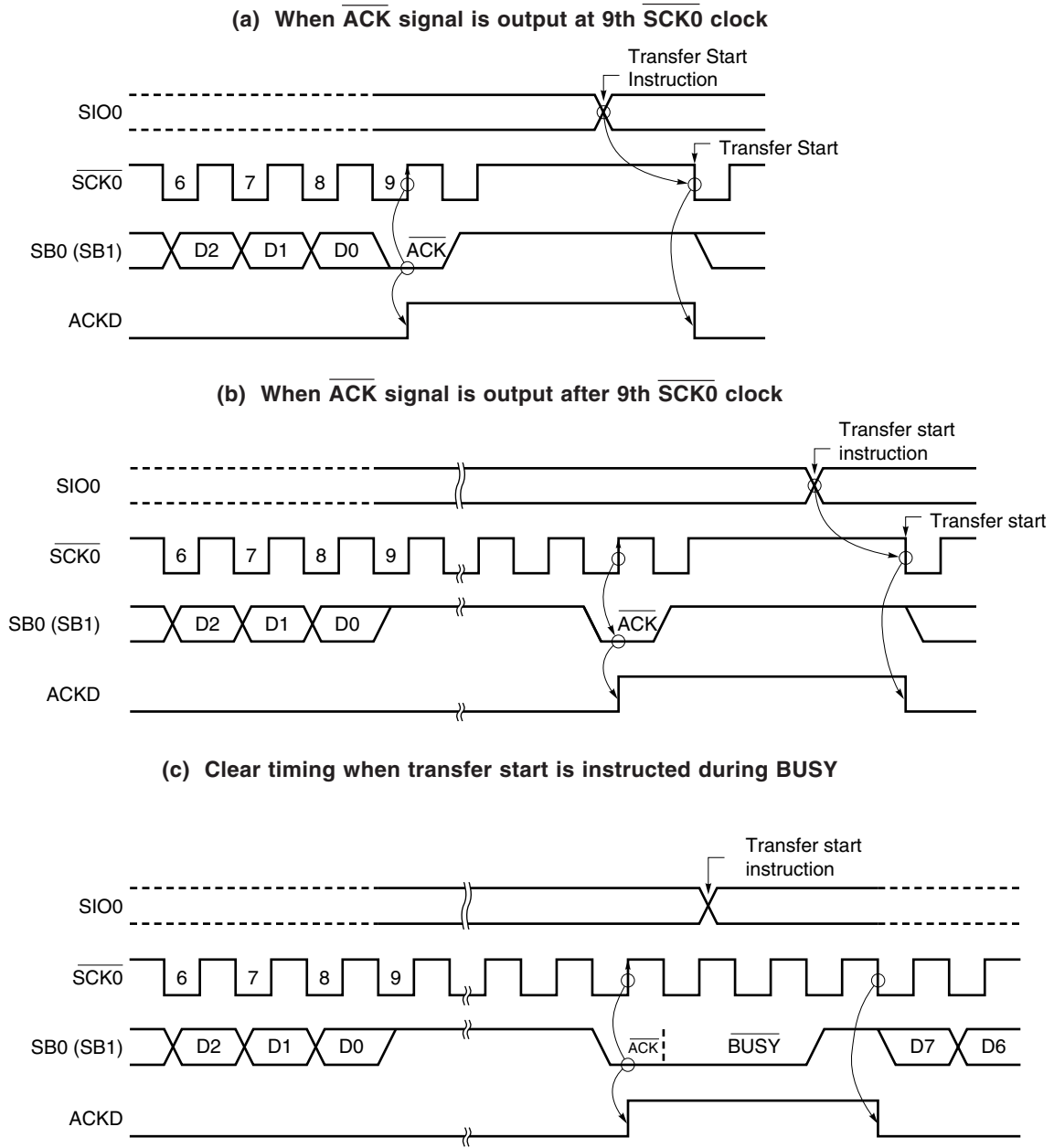
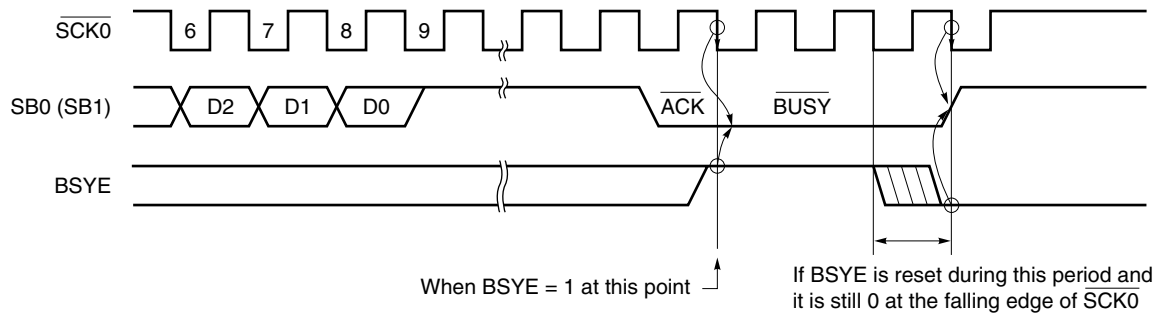


Figure 16-25. BSYE Operation



**Table 16-3. Various Signals in SBI Mode (1/2)**

Signal Name	Output Device	Definition	Timing Chart	Output Conditions	Effects on Flag	Meaning of Signal
Bus release signal (REL)	Master	SB0 (SB1) rising edge when $\overline{SCK0} = 1$		<ul style="list-style-type: none"> <li>• RELT set</li> </ul>	<ul style="list-style-type: none"> <li>• RELD set</li> <li>• CMDD clear</li> </ul>	CMD signal is output to indicate that transmit data is an address.
Command signal (CMD)	Master	SB0 (SB1) falling edge when $\overline{SCK0} = 1$		<ul style="list-style-type: none"> <li>• CMDT set</li> </ul>	<ul style="list-style-type: none"> <li>• CMDD set</li> </ul>	i) Transmit data is an address after REL signal output. ii) REL signal is not output and transmit data is an command.
Acknowledge signal ( $\overline{ACK}$ )	Master/slave	Low-level signal output to SB0 (SB1) during one-clock period of $\overline{SCK0}$ after completion of serial reception	[Synchronous BUSY output]	① $\overline{ACKE} = 1$ ② $\overline{ACKT}$ set	<ul style="list-style-type: none"> <li>• ACKD set</li> </ul>	Completion of reception
Busy signal ( $\overline{BUSY}$ )	Slave	[Synchronous BUSY signal] Low-level signal output to SB0 (SB1) following acknowledge signal		<ul style="list-style-type: none"> <li>• BSYE = 1</li> </ul>	—	Serial receive disabled because of processing
Ready signal (READY)	Slave	High-level signal output to SB0 (SB1) before serial transfer start and after completion of serial transfer		① $\overline{BSYE} = 0$ ② Execution of instruction for data write to SIO0 (transfer start instruction)	—	Serial receive enabled

Table 16-3. Various Signals in SBI Mode (2/2)

Signal Name	Output Device	Definition	Timing Chart	Output Conditions	Effects on Flag	Meaning of Signal
Serial clock (SCK0)	Master	Synchronous clock to output address/command/data, ACK signal, synchronous BUSY signal, etc. Address/command/data is transferred with the first eight synchronous clocks.		When CSIE0 = 1, execution of instruction for data write to SIO0 (serial transfer start instruction) <sup>Note 2</sup>	CSIIF0 set (rising edge of 9th clock of SCK0) <sup>Note 1</sup>	Timing of signal output to serial data bus
Address (A7 to A0)	Master	8-bit data transferred in synchronization with SCK0 after output of REL and CMD signals				Address value of slave device on the serial bus
Command (C7 to C0)	Master	8-bit data transferred in synchronization with SCK0 after output of only CMD signal without REL signal output				Instructions and messages to the slave device
Data (D7 to D0)	Master/slave	8-bit data transferred in synchronization with SCK0 without output of REL and CMD signals				Numeric values to be processed by slave or master device

**Notes** 1. When WUP = 0, CSIIF0 is set at the rising edge of the 9th clock of SCK0.

When WUP = 1, an address is received. Only when the address matches the slave address register (SVA) value, CSIIF0 is set (if the address does not match the value of SVA, RELD is cleared).

2. In the BUSY state, transfer starts after the READY state is set.

**(5) Pin configuration**

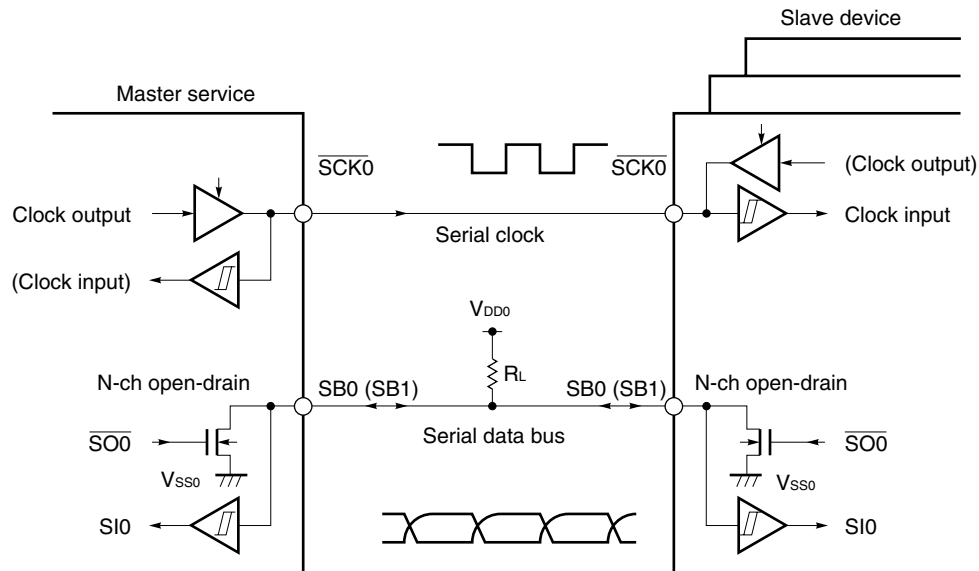
The serial clock pin  $\overline{\text{SCK0}}$  and serial data bus pin SB0 (SB1) have the following configurations.

- (a)  $\overline{\text{SCK0}}$  ..... Serial clock I/O pin
- ① Master ... CMOS and push-pull output
  - ② Slave ..... Schmitt input
- (b) SB0 (SB1) .... Serial data I/O alternate-function pin

Both master and slave devices have an N-ch open-drain output and a Schmitt input.

Because the serial data bus line has an N-ch open-drain output, an external pull-up resistor is necessary.

**Figure 16-26. Pin Configuration**



**Caution** Because the N-ch open-drain output must be made to go into a high-impedance state during data reception, write FFH to serial I/O shift register 0 (SIO0) in advance. The N-ch open-drain output can always go into a high-impedance state during transfer. However, when the wake-up function specification bit (WUP) = 1, the N-ch open-drain output always goes into a high-impedance state. Thus, it is not necessary to write FFH to SIO0 before reception.

**(6) Address match detection method**

In the SBI mode, the master transmits a slave address to select a specific slave device.

A match of the addresses can be automatically detected by hardware. CSIF0 is set only when the slave address transmitted by the master matches the address set to SVA when the wakeup function specification bit (WUP) = 1.

If bit 5 (SIC) of the interrupt timing specify register (SINT) is set, the wakeup function cannot be used even if WUP is set (an interrupt request signal is generated when bus release is detected). To use the wake-up function, clear SIC to 0.

**Cautions 1. Slave selection/non-selection is detected by matching of the slave address received after bus release (RELD = 1).**

**For this match detection, the match interrupt request (INTCSI0) of the address to be generated with WUP = 1 is normally used. Thus, execute selection/non-selection detection by slave address when WUP = 1.**

**2. When detecting selection/non-selection without the use of an interrupt request with WUP = 0, do so by means of transmission/reception of the command preset by program instead of using the address match detection method.**

**(7) Error detection**

In the SBI mode, the serial bus SB0 (SB1) status being transmitted is fetched into the destination device, that is, serial I/O shift register 0 (SIO0). Thus, transmit errors can be detected in the following ways.

**(a) Method of comparing SIO0 data before and after transmission**

In this case, if the two data differ from each other, a transmit error is judged to have occurred.

**(b) Method of using the slave address register (SVA)**

Transmit data is set to both SIO0 and SVA and is transmitted. After termination of transmission, the COI bit (match signal coming from the address comparator) of serial operating mode register 0 (CSIM0) is tested. If "1", normal transmission is judged to have been carried out. If "0", a transmit error is judged to have occurred.

**(8) Communication operation**

In the SBI mode, the master device normally selects one slave device as the communication target from among two or more devices by outputting an "address" to the serial bus.

After the communication target device has been determined, commands and data are transmitted/received and serial communication is realized between the master and slave device.

Figures 16-27 to 16-30 show data communication timing charts.

Shift operations of serial I/O shift register 0 (SIO0) are carried out at the falling edge of the serial clock ( $\overline{\text{SCK0}}$ ). Transmit data is latched into the SO0 latch and is output with the MSB set as the first bit from the SB0/P25 or SB1/P26 pin. Receive data input to the SB0 (or SB1) pin at the rising edge of  $\overline{\text{SCK0}}$  is latched into SIO0.



Figure 16-27. Address Transmission from Master Device to Slave Device (WUP = 1)

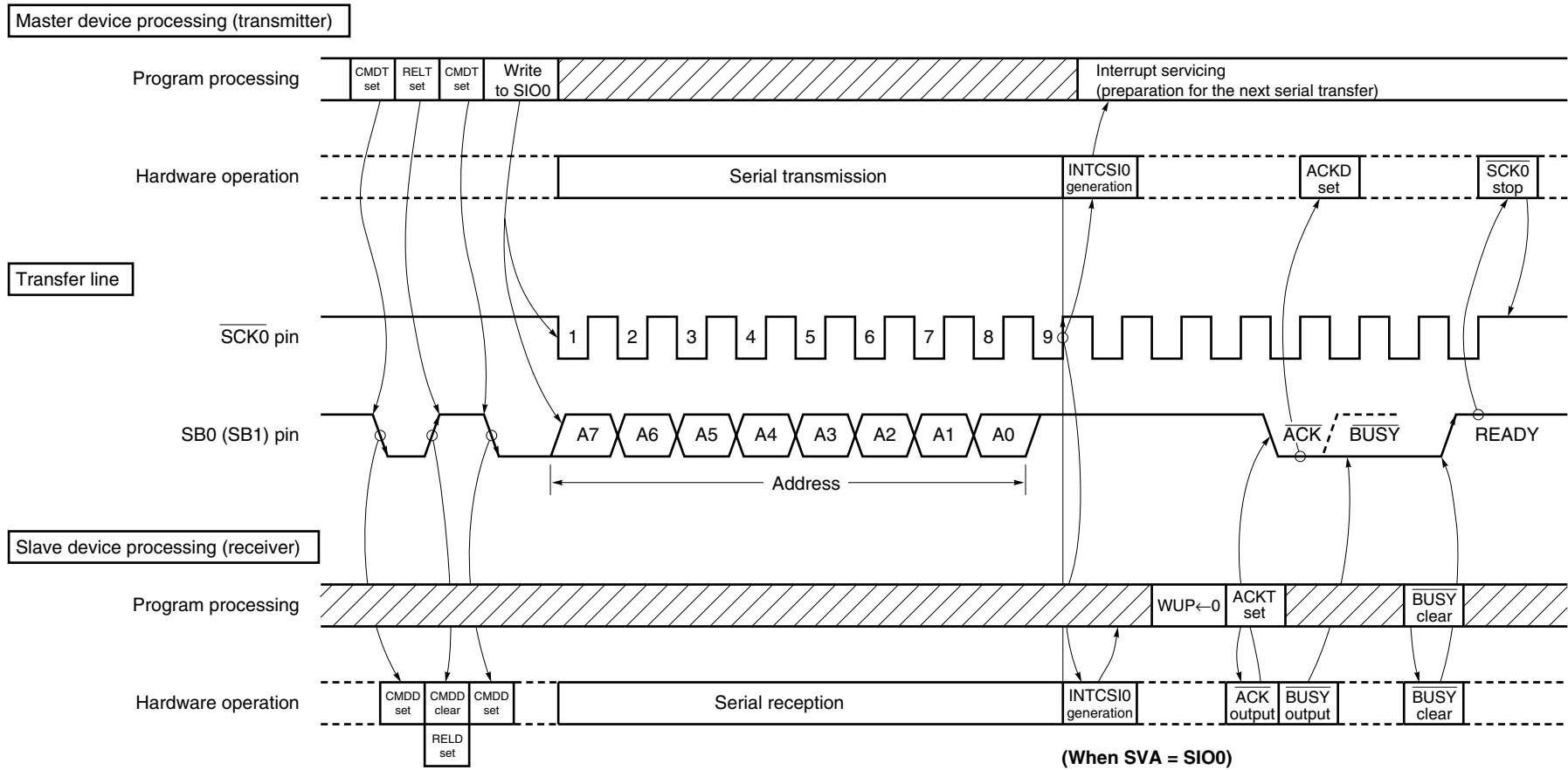


Figure 16-28. Command Transmission from Master Device to Slave Device

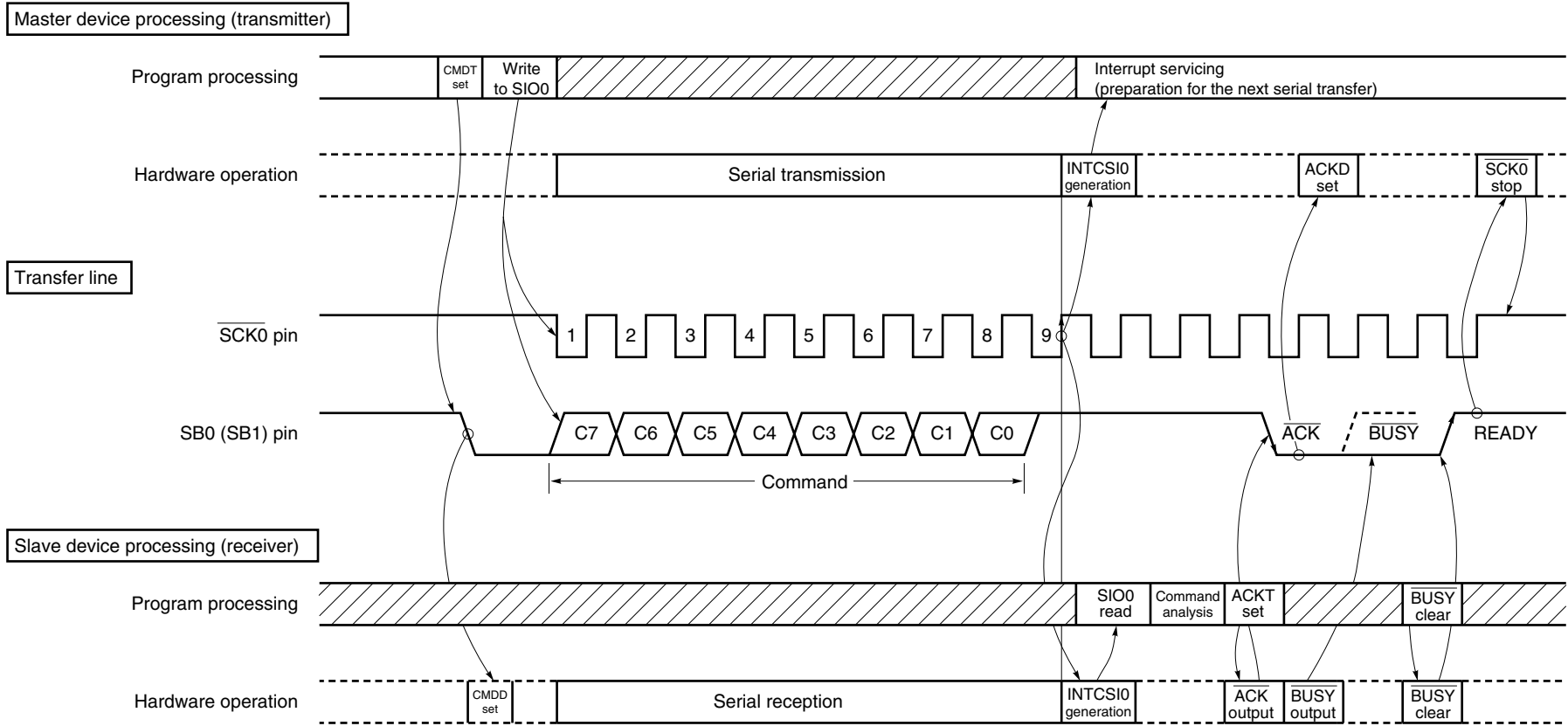


Figure 16-29. Data Transmission from Master Device to Slave Device

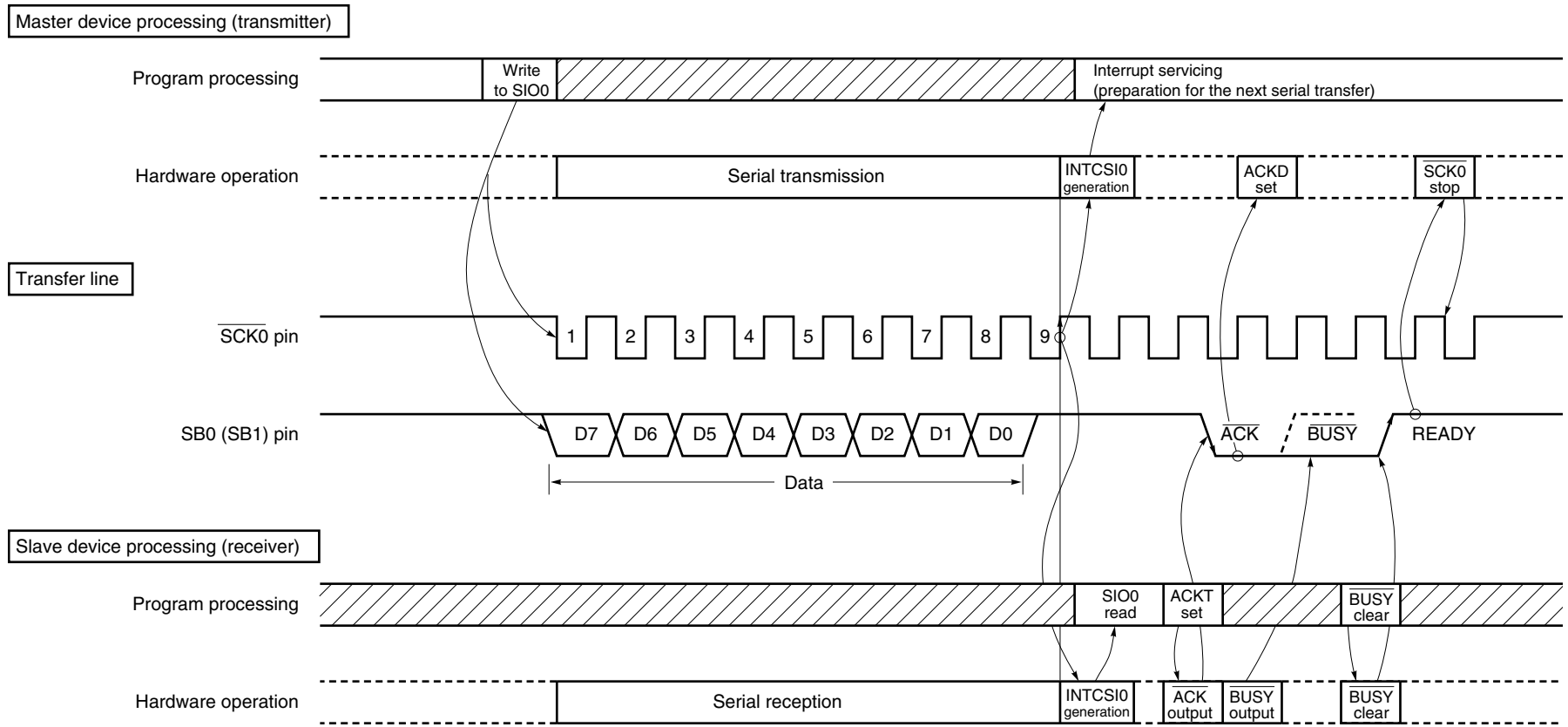
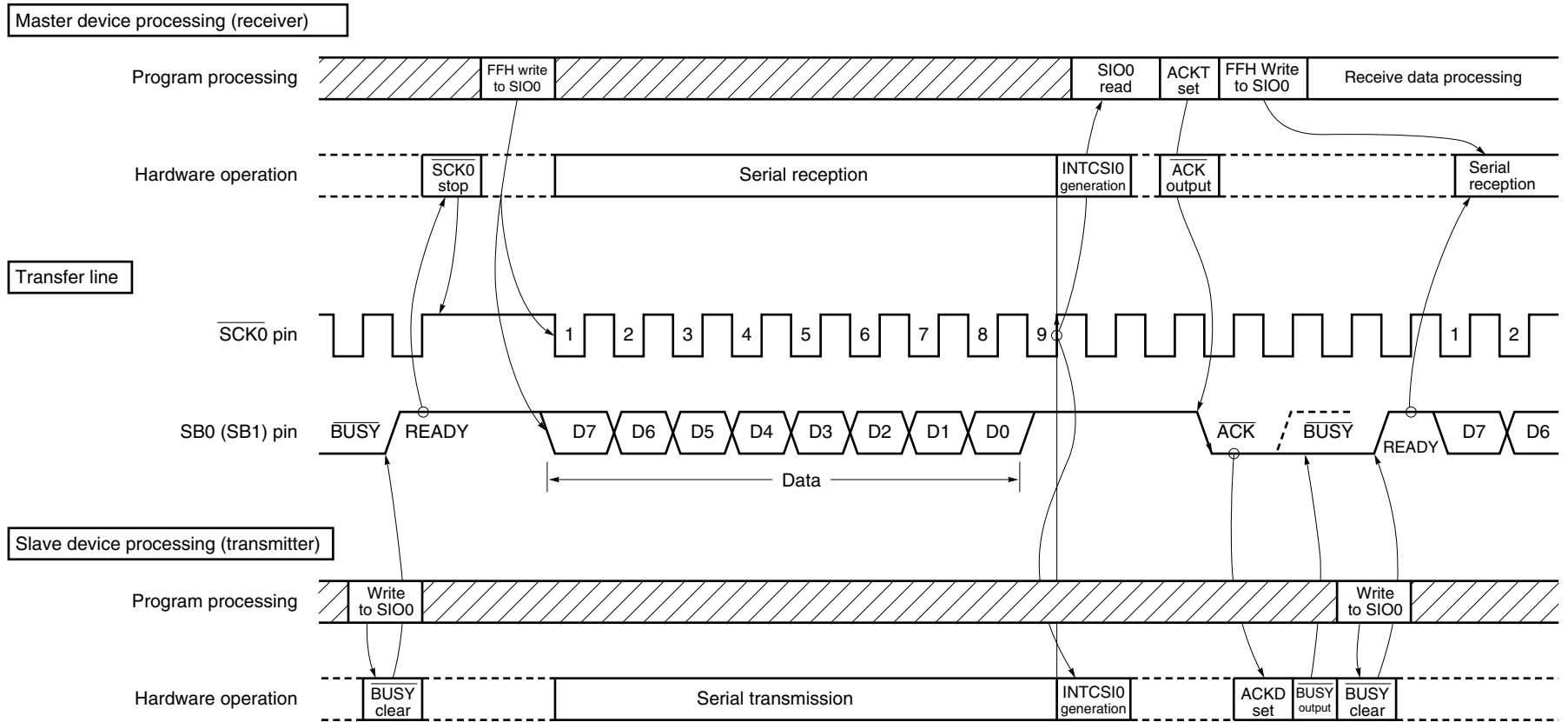


Figure 16-30. Data Transmission from Slave Device to Master Device



**(9) Transfer start**

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or  $\overline{\text{SCK0}}$  is at high level after 8-bit serial transfer.

**Cautions 1. If CSIE0 is set to “1” after data write to SIO0, transfer does not start.**

**2. Because the N-ch open-drain output must go into a high-impedance state during data reception, write FFH to SIO0 in advance.**

**However, when the wakeup function specification bit (WUP) = 1, the N-ch open-drain output always goes into a high-impedance state. Thus, it is not necessary to write FFH to SIO0 before reception.**

**3. If data is written to SIO0 when the slave is busy, the data is not lost.**

**When the busy state is cleared and SB0 (or SB1) input is set to the high level (READY) state, transfer starts.**

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIF0) is set.

For pins that are to be used for data I/O, be sure to carry out the following settings before serial transfer of the 1st byte after  $\overline{\text{RESET}}$  input.

- <1> Set the P25 and P26 output latches to 1.
- <2> Set bit 0 (RELT) of the serial bus interface control register (SBIC) to 1.
- <3> Reset the P25 and P26 output latches from 1 to 0.

**(10) Judging busy state of slave**

When the device is in the master mode, follow the procedure below to judge whether the slave device is in the busy state or not.

- <1> Detect acknowledge signal ( $\overline{\text{ACK}}$ ) or interrupt request signal generation.
- <2> Set the port mode register PM25 (or PM26) of the SB0/P25 (or SB1/P26) pin to the input mode.
- <3> Read out the pin state (when the pin level is high, the READY state is set).

After detection of the READY state, clear the port mode register to 0 and return to the output mode.

**(11) SBI mode precautions**

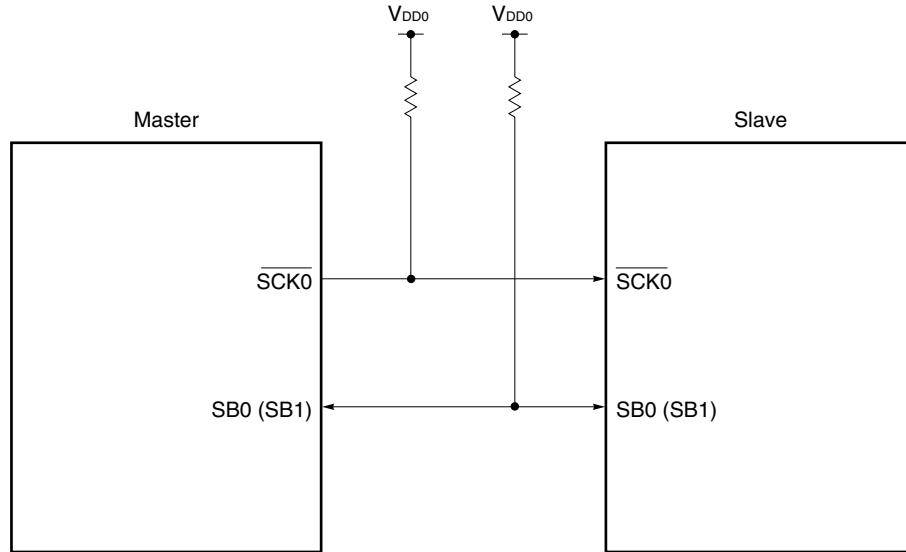
- (a) Slave selection/non-selection is detected by match detection of the slave address received after bus release ( $RELD = 1$ ).  
For this match detection, the match interrupt request (INTCSI0) of the address to be generated with  $WUP = 1$  is normally used. Thus, execute selection/non-selection detection by slave address when  $WUP = 1$ .
- (b) When detecting selection/non-selection without the use of an interrupt with  $WUP = 0$ , do so by means of transmission/reception of the command preset by program instead of using the address match detection method.
- (c) In the SBI mode, the  $\overline{BUSY}$  signal is output until the next serial clock falls after a command that resets the  $\overline{BUSY}$  signal has been issued. If  $WUP$  is set to 1 during this period by mistake, the  $\overline{BUSY}$  signal is not reset. Therefore, be sure to confirm that the SB0 (SB1) pin has gone high after resetting the  $\overline{BUSY}$  signal, by setting  $WUP$  to 1.
- (d) For pins that are to be used for data I/O, be sure to carry out the following settings before serial transfer of the 1st byte after  $\overline{RESET}$  input.
  - <1> Set the P25 and P26 output latches to 1.
  - <2> Set bit 0 (RELT) of the serial bus interface control register (SBIC) to 1.
  - <3> Reset the P25 and P26 output latches from 1 to 0.
- (e) The transition of the SB0 (SB1) line from low to high or from high to low when the  $\overline{SCK0}$  line is high is recognized as a bus release signal or a command signal, respectively. If the transition timing of the bus is shifted due to the influence of board capacitance, transmitted data may be judged as a bus release signal (or a command signal). Exercise care in wiring so that noise is not superimposed on the signal lines.

#### 16.4.4 2-wire serial I/O mode operation

The 2-wire serial I/O mode can cope with any communication format by program.

Communication is basically carried out with the two lines of the serial clock ( $\overline{\text{SCK0}}$ ) and serial data input/output (SB0 or SB1).

Figure 16-31. Serial Bus Configuration Example Using 2-Wire Serial I/O Mode



#### (1) Register setting

The 2-wire serial I/O mode is set by serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specification register (SINT).

##### (a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock to SCK0 pin from off-chip								
	1	0	8-bit timer register 2 (TM2) output								
	1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SIO/SB0/P25 pin function	SO0/SB1/P26 pin function	SCK0/P27 pin function
	0	×	3-wire serial I/O mode (see 16.4.2 3-wire serial I/O mode operation)											
	1	0	SBI mode (see 16.4.3 SBI mode operation)											
	1	1	0	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	0	0	0	1	2-wire serial I/O mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	SCK0 (N-ch open-drain I/O)
			1	0	0	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	0	1			SB0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W	WUP	Wakeup function control <sup>Note 3</sup>									
	0	Interrupt request signal generation with each serial transfer in any mode									
	1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register (SVA) data in SBI mode									

R	COI	Slave address comparison result flag <sup>Note 4</sup>									
	0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data									
	1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data									

R/W	CSIE0	Serial interface channel 0 operation control									
	0	Operation stopped									
	1	Operation enabled									

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. Can be used freely as a port function.
  3. Be sure to set WUP to 0 in the 2-wire serial I/O mode.
  4. When CSIE0 = 0, COI becomes 0.

**Remark**

- ×: don't care
- PMxx: Port mode register
- Pxx: Port output latch



**(b) Serial bus interface control register (SBIC)**

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	RELT	When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

R/W	CMDT	When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

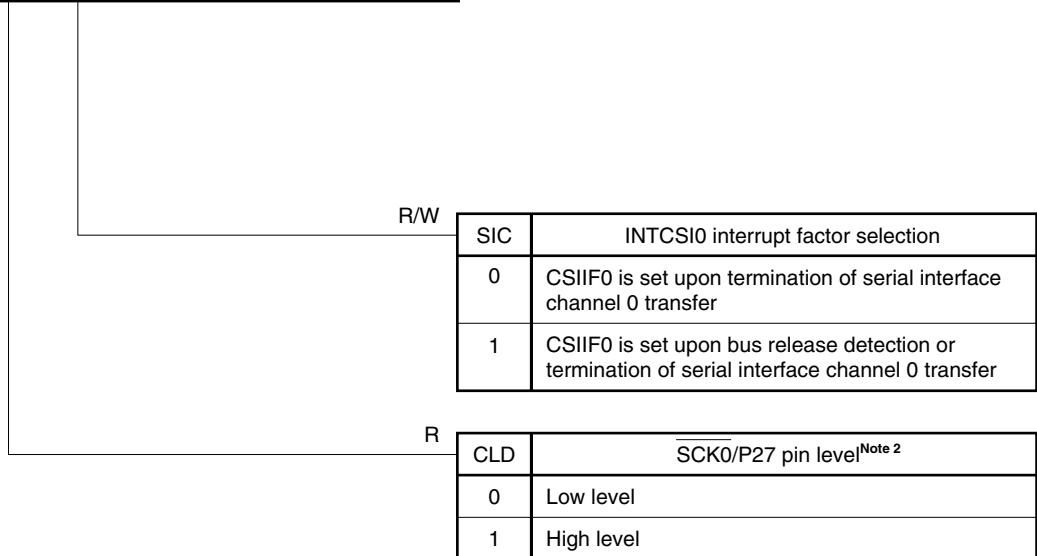
CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

**(c) Interrupt timing specification register (SINT)**

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SINT to 00H.

Symbol	7	<6>	<5>	<4>	3	2	1	0	Address	After reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W <sup>Note 1</sup>



**Caution** Be sure to clear bits 0 to 3 to 0.

- Notes**
1. Bit 6 (CLD) is a read-only bit.
  2. When CSIE0 = 0, CLD becomes 0.

**Remark** CSIIF0: Interrupt request flag corresponding to INTCSIO  
 CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

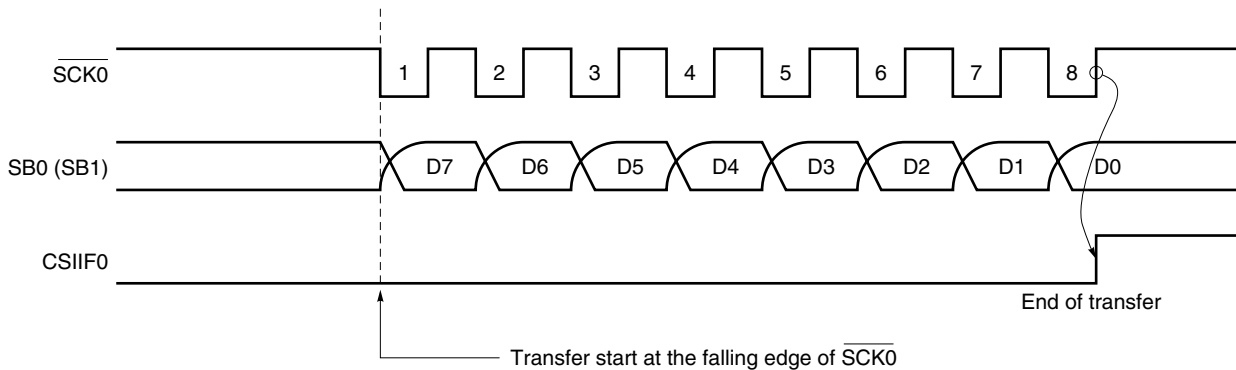
**(2) Communication operation**

The 2-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

Shift operations of serial I/O shift register 0 (SIO0) are carried out in synchronization with the falling edge of the serial clock ( $\overline{\text{SCK0}}$ ). The transmit data is held in the SO0 latch and is output from the SB0/P25 (or SB1/P26) pin on an MSB-first basis. The receive data input from the SB0 (or SB1) pin is latched into the SIO0 at the rising edge of  $\overline{\text{SCK0}}$ .

Upon termination of 8-bit transfer, the SIO0 operation stops automatically and the interrupt request flag (CSIF0) is set.

**Figure 16-32. 2-Wire Serial I/O Mode Timing**



The SB0 (or SB1) pin specified for the serial data bus is an N-ch open-drain I/O and thus it must be externally connected to a pull-up resistor. Because an N-ch open-drain output must go into a high-impedance state during data reception, write FFH to SIO0 in advance.

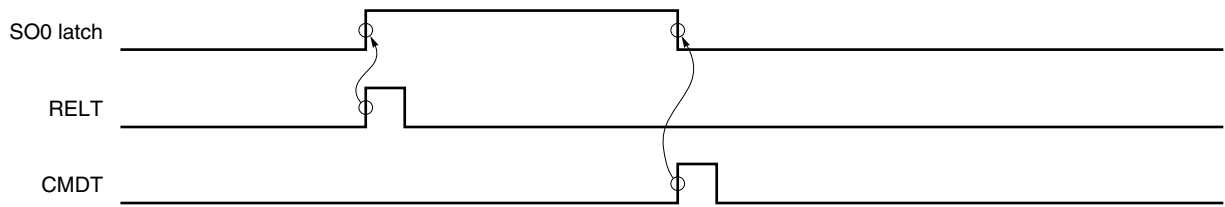
The SB0 (or SB1) pin generates the SO0 latch status and thus the SB0 (or SB1) pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the  $\overline{\text{SCK0}}$  pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (see **16.4.5  $\overline{\text{SCK0}}$ /P27 pin output manipulation**).

**(3) Other signals**

Figure 16-33 shows the RELT and CMDT operations.

**Figure 16-33. RELT and CMDT Operations**

**(4) Transfer start**

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or  $\overline{\text{SCK0}}$  is high level after 8-bit serial transfer.

**Cautions 1. If CSIE0 is set to 1 after data write to SIO0, transfer does not start.**

**2. Because the N-ch open-drain output must go into a high-impedance state during data reception, write FFH to SIO0 in advance.**

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIF0) is set.

**(5) Error detection**

In the 2-wire serial I/O mode, the serial bus SB0 (SB1) status being transmitted is fetched into the destination device, that is, serial I/O shift register 0 (SIO0). Thus, transmit errors can be detected in the following ways.

**(a) Method of comparing SIO0 data before and after transmission**

In this case, if the two data differ from each other, a transmit error is judged to have occurred.

**(b) Method of using the slave address register (SVA)**

Transmit data is set to both SIO0 and SVA and is transmitted. After termination of transmission, the COI bit (match signal coming from the address comparator) of serial operating mode register 0 (CSIM0) is tested. If "1", normal transmission is judged to have been carried out. If "0", a transmit error is judged to have occurred.

### 16.4.5 $\overline{\text{SCK0/P27}}$ pin output manipulation

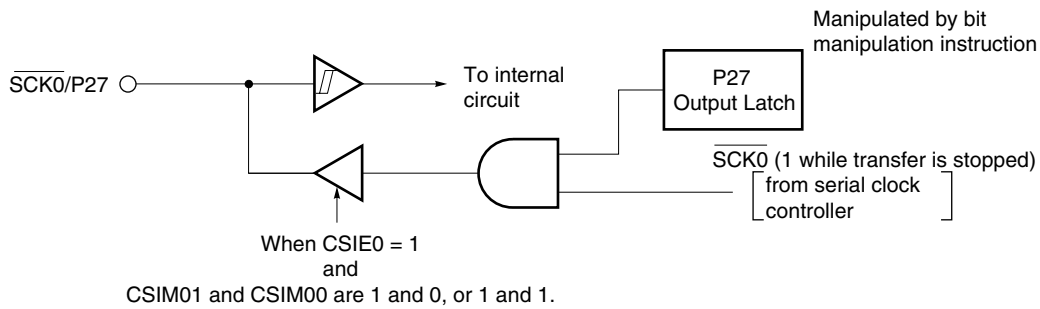
Because the  $\overline{\text{SCK0/P27}}$  pin incorporates an output latch, static output is also possible by software in addition to normal serial clock output.

P27 output latch manipulation enables any value of  $\overline{\text{SCK0}}$  to be set by software. (The SI0/SB0 and SO0/SB1 pins are controlled by bits 0 and 1 (RELT and CMDT) of the serial bus interface control register (SBIC).)

The procedure for manipulating the  $\overline{\text{SCK0/P27}}$  pin output is described below.

- ① Set serial operating mode register 0 (CSIM0) ( $\overline{\text{SCK0}}$  pin: Output mode, serial operation: Enabled).  $\overline{\text{SCK0}} = 1$  while serial transfer is suspended.
- ② Manipulate the P27 output latch with a bit manipulation instruction.

**Figure 16-34.  $\overline{\text{SCK0/P27}}$  Pin Configuration**



## CHAPTER 17 SERIAL INTERFACE CHANNEL 0 ( $\mu$ PD780058Y SUBSERIES)

The  $\mu$ PD780058Y Subseries incorporates three serial interface channels. Differences between channels 0, 1, and 2 are as follows (see **CHAPTER 18 SERIAL INTERFACE CHANNEL 1** for details of serial interface channel 1 and **CHAPTER 19 SERIAL INTERFACE CHANNEL 2** for details of serial interface channel 2).

**Table 17-1. Differences Between Channels 0, 1, and 2**

Serial Transfer Mode		Channel 0	Channel 1	Channel 2
3-wire serial I/O	Clock selection	$f_{xx}/2$ , $f_{xx}/2^2$ , $f_{xx}/2^3$ , $f_{xx}/2^4$ , $f_{xx}/2^5$ , $f_{xx}/2^6$ , $f_{xx}/2^7$ , $f_{xx}/2^8$ , external clock, TO2 output	$f_{xx}/2$ , $f_{xx}/2^2$ , $f_{xx}/2^3$ , $f_{xx}/2^4$ , $f_{xx}/2^5$ , $f_{xx}/2^6$ , $f_{xx}/2^7$ , $f_{xx}/2^8$ , external clock, TO2 output	External clock, baud rate generator output
	Transfer method	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit Automatic transmit/ receive function	MSB/LSB switchable as the start bit
	Transfer end flag	Serial transfer end interrupt request flag (CSIF0)	Serial transfer end interrupt request flag (CSIF1)	Serial transfer end interrupt request flag (SRIF)
2-wire serial I/O		Use possible	None	None
I <sup>2</sup> C bus (Inter IC Bus)				
UART (Asynchronous serial interface)		None		Use possible Timer-division transfer function

## 17.1 Functions of Serial Interface Channel 0

Serial interface channel 0 employs the following four modes.

- Operation stop mode
- 3-wire serial I/O mode
- 2-wire serial I/O mode
- I<sup>2</sup>C (Inter IC) bus mode

**Caution** Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or SBI) while serial interface channel 0 is enabled to operate. To change the operating mode, stop the serial operation first.

### (1) Operation stop mode

This mode is used when serial transfer is not carried out. Power consumption can be reduced in this mode.

### (2) 3-wire serial I/O mode (MSB-/LSB-first selectable)

This mode is used for 8-bit data transfer using three lines, one each for the serial clock ( $\overline{\text{SCK0}}$ ), serial output (SO0) and serial input (SI0). This mode enables simultaneous transmission/reception and therefore reduces the data transfer processing time.

The start bit of transferred 8-bit data is switchable between MSB and LSB, so that devices can be connected regardless of their start bit recognition.

This mode should be used when connecting with peripheral I/O devices or display controllers which incorporate a conventional clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

### (3) 2-wire serial I/O mode (MSB-first)

This mode is used for 8-bit data transfer using two lines of serial clock ( $\overline{\text{SCK0}}$ ) and serial data bus (SB0 or SB1).

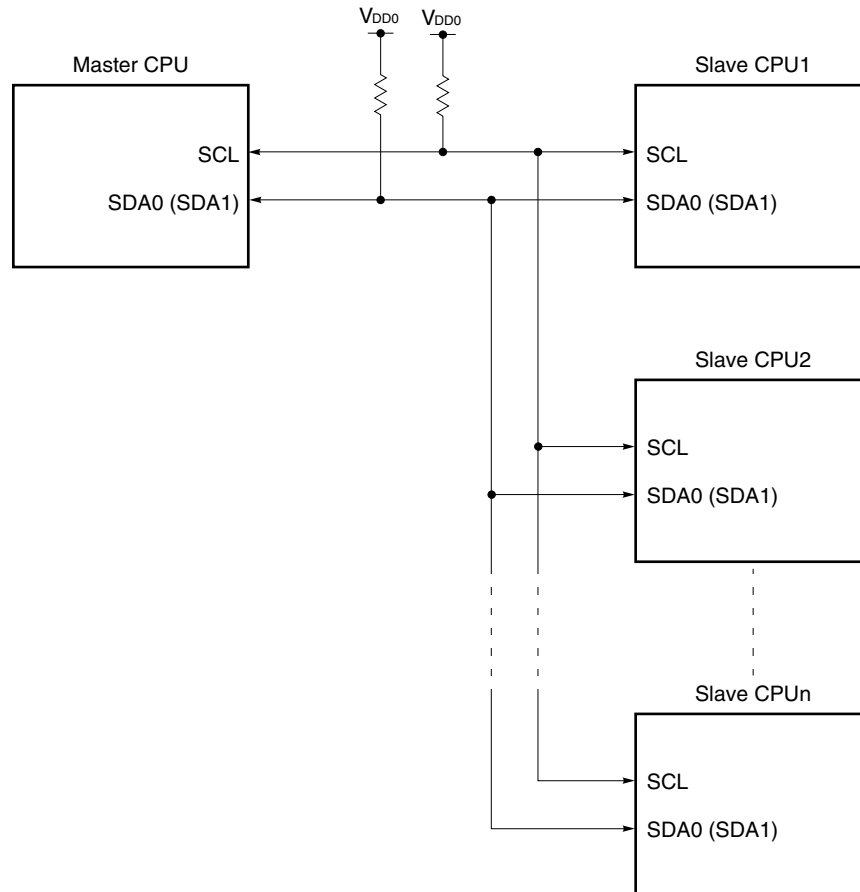
This mode enables to cope with any one of the possible data transfer formats by controlling the  $\overline{\text{SCK0}}$  level and the SB0 or SB1 output level. Thus, the handshake line previously necessary for connection of two or more devices can be removed, resulting in the increased number of available I/O ports.

**(4) I<sup>2</sup>C (Inter IC) bus mode (MSB-first)**

This mode is used for 8-bit data transfer with two or more devices using the two lines of the serial clock (SCL) and serial data bus (SDA0 or SDA1).

This mode complies with the I<sup>2</sup>C bus format. In this mode, the transmitter outputs three kinds of data onto the serial data bus: “start condition”, “data”, and “stop condition”, to be actually sent or received. The receiver automatically distinguishes the received data as “start condition”, “data”, or “stop condition”, by hardware.

**Figure 17-1. Serial Bus Configuration Example Using I<sup>2</sup>C Bus**



## 17.2 Configuration of Serial Interface Channel 0

Serial interface channel 0 consists of the following hardware.

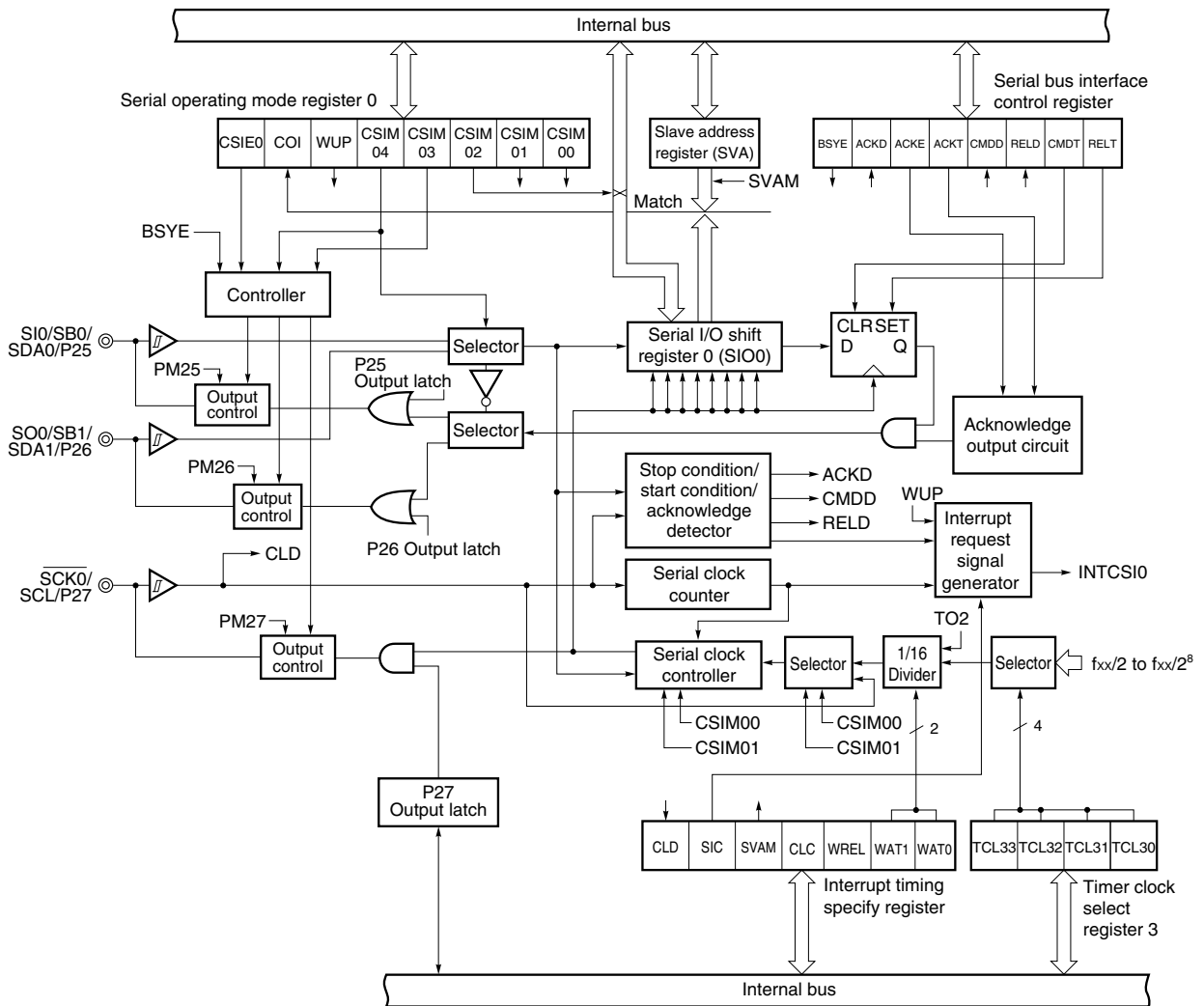
**Table 17-2. Configuration of Serial Interface Channel 0**

Item	Configuration
Registers	Serial I/O shift register 0 (SIO0) Slave address register (SVA)
Control registers	Timer clock select register 3 (TCL3) Serial operating mode register 0 (CSIM0) Serial bus interface control register (SBIC) Interrupt timing specify register (SINT) Port mode register 2 (PM2) <sup>Note</sup>

**Note** See **Figure 6-7 Block Diagram of P20, P21, and P23 to P26** and **Figure 6-8 Block Diagram of P22 and P27**.



Figure 17-2. Block Diagram of Serial Interface Channel 0



**Remark** The output control block performs selection between CMOS output and N-ch open-drain output.

**(1) Serial I/O shift register 0 (SIO0)**

SIO0 is an 8-bit register used to carry out parallel-serial conversion and to carry out serial transmission/reception (shift operation) in synchronization with the serial clock.

SIO0 is set with an 8-bit memory manipulation instruction.

When bit 7 (CSIE0) of serial operating mode register 0 (CSIM0) is 1, writing data to SIO0 starts a serial operation.

In transmission, data written to SIO0 is output to the serial output (SO0) or serial data bus (SB0/SB1). In reception, data is read from the serial input (SI0) or SB0/SB1 to SIO0.

Note that, if a bus is driven in the I<sup>2</sup>C bus mode or 2-wire serial I/O mode, the bus pins must serve for both input and output. Therefore, the transmission N-ch transistor of the device which will start reception of data must be turned off beforehand. Consequently, write FFH to SIO0 in advance.

In the I<sup>2</sup>C bus mode, set SIO0 to FFH with bit 7 (BSYE) of the serial bus interface control register (SBIC) set to 1.

$\overline{\text{RESET}}$  input makes SIO0 undefined.

**Caution** Do not execute an instruction that writes SIO0 in the I<sup>2</sup>C bus mode while WUP (bit 5 of serial operating mode register 0 (CSIM0)) = 1. Even if such an instruction is not executed, data can be received when the wake-up function is used (WUP = 1). For the detail of the wake-up function, see 17.4.4 (1) (c) Wake-up function.

**(2) Slave address register (SVA)**

SVA is an 8-bit register used to set the slave address value for connection of a slave device to the serial bus. SVA is set with an 8-bit memory manipulation instruction. This register is not used in the 3-wire serial I/O mode.

The master device outputs a slave address to the connected slave devices for selection of a particular slave device. These two data (the slave address output from the master device and the SVA value) are compared with an address comparator. If they match, the slave device has been selected. In that case, bit 6 (COI) of serial operating mode register 0 (CSIM0) becomes 1.

Address comparison can also be executed on the data of LSB-masked higher 7 bits by setting bit 4 (SVAM) of the interrupt timing specify register (SINT) to 1.

If no matching is detected in address reception, bit 2 (RELD) of the serial bus interface control register (SBIC) is cleared to 0. In the I<sup>2</sup>C bus mode, the wakeup function can be used by setting bit 5 (WUP) of CSIM0 to 1. In this case, the interrupt request signal (INTCSI0) is generated when the slave address output by the master matches the SVA value (the interrupt request signal is also generated when the stop condition is detected), and it can be learned by this interrupt request that the master requests for communication. To use the wakeup function, set SIC to 1.

Further, an error can be detected by using SVA when the device transmits data as a master or slave device in I<sup>2</sup>C bus mode or 2-wire serial I/O mode.

$\overline{\text{RESET}}$  input makes SVA undefined.

**(3) SO0 latch**

This latch holds the SI0/SB0/SDA0/P25 and SO0/SB1/SDA1/P26 pin levels. It can be directly controlled by software.

**(4) Serial clock counter**

This counter counts the serial clocks to be output and input during transmission/reception to check whether 8-bit data has been transmitted/received.

**(5) Serial clock controller**

This circuit controls serial clock supply to serial I/O shift register 0 (SIO0). When the internal system clock is used, the circuit also controls clock output to the  $\overline{\text{SCK0/SCL/P27}}$  pin.

**(6) Interrupt signal generator**

This circuit controls interrupt request signal generation. It generates interrupt request signals according to the settings of interrupt timing specification register (SINT) bits 0 and 1 (WAT0, WAT1) and serial operation mode register 0 (CSIM0) bit 5 (WUP), as shown in Table 17-3.

**(7) Acknowledge output circuit and stop condition/start condition/acknowledge detector**

These two circuits output and detect various control signals in the I<sup>2</sup>C mode.

These do not operate in the 3-wire serial I/O mode and 2-wire serial I/O mode.

**Table 17-3. Interrupt Request Signal Generation of Serial Interface Channel 0**

Serial Transfer Mode	BSYE	WUP	WAT1	WAT0	ACKE	Description
3-wire or 2-wire serial I/O mode	0	0	0	0	0	An interrupt request signal is generated each time 8 serial clocks are counted.
	Other than above					Setting prohibited
I <sup>2</sup> C bus mode (transmit)	0	0	1	0	0	An interrupt request signal is generated each time 8 serial clocks are counted (8-clock wait). Normally, during transmission the settings WAT21, WAT0 = 1, 0, are not used. They are used only when wanting to coordinate receive time and processing systematically using software. ACK information is generated by the receiving side, thus ACKE should be set to 0 (disable).
			1	1	0	An interrupt request signal is generated each time 9 serial clocks are counted (9-clock wait). ACK information is generated by the receiving side, thus ACKE should be set to 0 (disable).
	Other than above					Setting prohibited
I <sup>2</sup> C bus mode (receive)	1	0	1	0	0	An interrupt request signal is generated each time 8 serial clocks are counted (8-clock wait). ACK information is output by manipulating ACKT by software after an interrupt request is generated.
			1	1	0/1	An interrupt request signal is generated each time 9 serial clocks are counted (9-clock wait). To automatically generate ACK information, preset ACKE to 1 before transfer start. However, in the case of the master, set ACKE to 0 (disable) before receiving the last data.
	1	1	1	1	1	After an address is received, if the values of serial I/O shift register 0 (SIO0) and the slave address register (SVA) match, and if the stop condition is detected, an interrupt request signal is generated. To automatically generate ACK information, preset ACKE to 1 (enable) before transfer start.
	Other than above					Setting prohibited

**Remark** BSYE: Bit 7 of the serial bus interface control register (SBIC)

ACKE: Bit 5 of the serial bus interface control register (SBIC)

### 17.3 Control Registers of Serial Interface Channel 0

The following four registers are used to control serial interface channel 0.

- Timer clock select register 3 (TCL3)
- Serial operating mode register 0 (CSIM0)
- Serial bus interface control register (SBIC)
- Interrupt timing specification register (SINT)

#### (1) Timer clock select register 3 (TCL3)

This register sets the serial clock of serial interface channel 0.

TCL3 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets TCL3 to 88H.

Figure 17-3. Format of Timer Clock Select Register 3

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL3	TCL37	TCL36	TCL35	TCL34	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33	TCL32	TCL31	TCL30	Serial interface channel 0 serial clock selection					
				Serial clock in I <sup>2</sup> C bus mode			Serial clock in 2-wire or 3-wire serial I/O mode		
					MCS = 1	MCS = 0		MCS = 1	MCS = 0
0	1	1	0	$f_{xx}/2^5$	Setting prohibited	$f_x/2^6$ (78.1 kHz)	$f_{xx}/2$	Setting prohibited	$f_x/2^2$ (1.25 MHz)
0	1	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	0	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
1	0	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.77 kHz)	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
1	0	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.77 kHz)	$f_x/2^{10}$ (4.88 kHz)	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
1	0	1	1	$f_{xx}/2^{10}$	$f_x/2^{10}$ (4.88 kHz)	$f_x/2^{11}$ (2.44 kHz)	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	1	0	0	$f_{xx}/2^{11}$	$f_x/2^{11}$ (2.44 kHz)	$f_x/2^{12}$ (1.22 kHz)	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	1	0	1	$f_{xx}/2^{12}$	$f_x/2^{12}$ (1.22 kHz)	$f_x/2^{13}$ (0.61 kHz)	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
Other than above				Setting prohibited					

TCL37	TCL36	TCL35	TCL34	Serial interface channel 1 serial clock selection		
					MCS = 1	MCS = 0
0	1	1	0	$f_{xx}/2$	Setting prohibited	$f_x/2^2$ (1.25 MHz)
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
Other than above				Setting prohibited		

**Caution** When rewriting TCL3 to other data, stop the serial transfer operation beforehand.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

(2) Serial operating mode register 0 (CSIM0)

This register sets the serial interface channel 0 serial clock, operating mode, operation enable/stop wakeup function and displays the address comparator match signal.

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears CSIM0 to 00H.

**Caution** Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or SBI) while serial interface channel 0 is enabled to operate. To change the operating mode, stop the serial operation first.

Figure 17-4. Format of Serial Operating Mode Register 0

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock to $\overline{\text{SCK0}}$ /SCL pin from off-chip								
	1	0	8-bit timer register 2 (TM2) output <sup>Note 2</sup>								
	1	1	Clock specified by bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	PM25	PM26	PM26	PM27	PM27	Operation mode	Start Bit	SI0/SB0/SDA0/ P25 pin function	SO0/SB1/SDA1/ P26 pin function	$\overline{\text{SCK0}}$ /SCL/P27 pin function
	0	×	0	<sup>Note 3</sup> 1	<sup>Note 3</sup> ×	0	0	0	1	3-wire serial I/O mode	MSB	SI0 <sup>Note 3</sup> (Input)	SO0 (CMOS output)	$\overline{\text{SCK0}}$ (CMOS I/O)
			1							LSB				
	1	1	0	<sup>Note 4</sup> ×	<sup>Note 4</sup> ×	0	0	0	1	2-wire serial I/O mode or I <sup>2</sup> C bus mode	MSB	P25 (CMOS I/O)	SB1/SDA1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ /SCL (N-ch open-drain I/O)
			1	0	0	<sup>Note 4</sup> ×	<sup>Note 4</sup> ×	0	1			SB0/SDA0 (N-ch open-drain)	P26 (CMOS I/O)	

R/W	WUP	Wake-up function control <sup>Note 5</sup>									
	0	Interrupt request signal generation with each serial transfer in any mode									
	1	Interrupt request signal generation when the address received after detecting start condition (when CMDD = 1) matches the slave address register (SVA) data in I <sup>2</sup> C bus mode									

R	COI	Slave address comparison result flag <sup>Note 6</sup>									
	0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data									
	1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data									

R/W	CSIE0	Serial interface channel 0 operation control									
	0	Operation stopped									
	1	Operation enabled									

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. In I<sup>2</sup>C bus mode, the clock frequency becomes 1/16 of that output from TO2.
  3. Can be used as P25 (CMOS input/output) when used only for transmission.
  4. Can be used freely as a port function.
  5. To use the wakeup function (WUP = 1), set bit 5 (SIC) of the interrupt timing specification register (SINT) to 1. Do not execute an instruction that writes serial I/O shift register 0 (SIO0) while WUP = 1.
  6. When CSIE0 = 0, COI becomes 0.

**Remark** ×: don't care  
 PM××: Port mode register  
 P××: Port output latch

**(3) Serial bus interface control register (SBIC)**

This register sets the serial bus interface operation and displays statuses.

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears SBIC to 00H.

**Figure 17-5. Format of Serial Bus Interface Control Register (1/2)**

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W <sup>Note</sup>
R/W	RELT	Used for stop condition signal output. When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R/W	CMDT	Used for start condition signal output. When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R	RELD	Stop condition detection									
		Clear conditions (RELD = 0)					Set conditions (RELD = 1)				
		<ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• If SIO0 and SVA values do not match in address reception</li> <li>• When <math>\overline{\text{CSIE0}} = 0</math></li> <li>• When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>					<ul style="list-style-type: none"> <li>• When stop condition signal is detected</li> </ul>				
R	CMDD	Start condition detection									
		Clear conditions (CMDD = 0)					Set conditions (CMDD = 1)				
		<ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• When stop condition signal is detected</li> <li>• When <math>\overline{\text{CSIE0}} = 0</math></li> <li>• When <math>\overline{\text{RESET}}</math> input is applied</li> </ul>					<ul style="list-style-type: none"> <li>• When start condition signal is detected</li> </ul>				
R/W	ACKT	Used to generate the $\overline{\text{ACK}}$ signal by software when 8-clock wait mode is selected. Keeps SDA0 (SDA1) low from set instruction (ACKT = 1) execution to the next falling edge of SCL. ACKT is also cleared to 0 upon start of serial interface transfer or when $\overline{\text{CSIE0}} = 0$ .									

**Note** Bits 2, 3, and 6 (RELD, CMDD, and ACKD) are read-only bits.

**Remark** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

Figure 17-5. Format of Serial Bus Interface Control Register (2/2)

R/W	ACKE	Acknowledge signal output control <sup>Note 1</sup>	
	0	Acknowledge signal automatic output disable (However, output by ACKT enabled) Used for reception when 8-clock wait mode is selected or for transmission. <sup>Note 2</sup>	
	1	Enables acknowledge signal automatic output. Outputs the acknowledge signal in synchronization with the falling edge of the 9th SCL clock cycle (automatically output when ACKE = 1). However, ACKE is not automatically cleared to 0 after acknowledge signal is output. Used in reception with 9-clock wait mode selected.	
R	ACKD	Acknowledge detection	
		Clear conditions (ACKD = 0)	Set conditions (ACKD = 1)
		<ul style="list-style-type: none"> <li>• While executing the transfer start instruction</li> <li>• When CSIE0 = 0</li> <li>• When RESE<math>\overline{T}</math>input is applied</li> </ul>	<ul style="list-style-type: none"> <li>• When acknowledge signal (<math>\overline{ACK}</math>) is detected at the rising edge of the SCL clock after completion of transfer</li> </ul>
R/W	<sup>Note 3</sup> BSYE	Control of N-ch open-drain output for transmission in I <sup>2</sup> C Bus Mode <sup>Note 4</sup>	
	0	Output enabled (transmission)	
	1	Output disabled (reception)	

- Notes**
1. Setting should be performed before transfer.
  2. If 8-clock wait mode is selected, the acknowledge signal at reception must be output using ACKT.
  3. The busy mode can be cleared by start of serial interface transfer or reception of address signal. However, the BSYE flag is not cleared to 0.
  4. When using the wakeup function, be sure to set BSYE to 1.

**Remark** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)



**(4) Interrupt timing specification register (SINT)**

This register sets the bus release interrupt and address mask functions and displays the  $\overline{\text{SCK0}}$ /SCL pin level status.

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears SINT to 00H.

**Figure 17-6. Format of Interrupt Timing Specification Register (1/2)**

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	After reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W <sup>Note 1</sup>

R/W	WAT1	WAT0	Wait and interrupt control
	0	0	Generates interrupt servicing request at rising edge of 8th $\overline{\text{SCK0}}$ clock cycle (keeping clock output in high impedance).
	0	1	Setting prohibited
	1	0	Used in I <sup>2</sup> C bus mode (8-clock wait). Generates interrupt servicing request at rising edge of 8th SCK0 clock cycle. (In the case of master device, makes SCL output low to enter wait state after 8 clock pulses are output. In the case of slave device, makes SCL output low to request wait state after 8 clock pulses are input.)
	1	1	Used in I <sup>2</sup> C bus mode (9-clock wait). Generates interrupt servicing request at rising edge of 9th SCK0 clock cycle. (In the case of master device, makes SCL output low to enter wait state after 9 clock pulses are output. In the case of slave device, makes SCL output low to request wait state after 9 clock pulses are input.)

R/W	WREL	Wait state release control
	0	Wait state has been released.
	1	Release wait state. Automatically cleared to 0 when the state is released (Used to cancel wait state by means of WAT0 and WAT1.)

R/W	CLC	Clock level control <sup>Note 2</sup>
	0	Used in I <sup>2</sup> C bus mode. Make output level of SCL pin low unless serial transfer is being performed.
	1	Used in I <sup>2</sup> C bus mode. Make SCL pin enter high-impedance state unless serial transfer is being performed (except for clock line which is kept high). Used to enable master device to generate start condition and stop condition signals.

- Notes**
1. Bit 6 (CLD) is a read-only bit.
  2. When not using the I<sup>2</sup>C mode, clear CLC to 0.

Figure 17-6. Format of Interrupt Timing Specification Register (2/2)

R/W	SVAM	SVA bit to be used as slave address
	0	Bits 0 to 7
	1	Bits 1 to 7
R/W	SIC	INTCSI0 interrupt source selection <sup>Note 1</sup>
	0	CSIF0 is set to 1 upon termination of serial interface channel 0 transfer
	1	CSIF0 is set to 1 upon stop condition detection or termination of serial interface channel 0 transfer
R	CLD	$\overline{\text{SCK0/SCL}}$ pin level <sup>Note 2</sup>
	0	Low level
	1	High level

- Notes**
1. When using the wakeup function in the I<sup>2</sup>C mode, clear SIC to 0.
  2. When CSIE0 = 0, CLD becomes 0.

**Remark** SVA: Slave address register  
 CSIF0: Interrupt request flag corresponding to INTCSI0  
 CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

## 17.4 Operations of Serial Interface Channel 0

The following four operating modes are available for serial interface channel 0.

- Operation stop mode
- 3-wire serial I/O mode
- 2-wire serial I/O mode
- I<sup>2</sup>C (Inter IC) bus mode

### 17.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power consumption can be reduced. Serial I/O shift register 0 (SIO0) does not carry out shift operations either and thus it can be used as an ordinary 8-bit register.

In the operation stop mode, the P25/SI0/SB0/SDA0, P26/SO0/SB1/SDA1, and P27/ $\overline{\text{SCK0}}$ /SCL pins can be used as general I/O ports.

#### (1) Register setting

The operation stop mode is set by serial operating mode register 0 (CSIM0).

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W

R/W	CSIE0	Serial interface channel 0 operation control
	0	Operation stopped
	1	Operation enabled

**17.4.2 3-wire serial I/O mode operation**

The 3-wire serial I/O mode is valid for connection of peripheral I/O units and display controllers which incorporate a conventional clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

Communication is carried out with the three lines of the serial clock ( $\overline{SCK0}$ ), serial output (SO0), and serial input (SIO).

**(1) Register setting**

The 3-wire serial I/O mode is set by serial operating mode register 0 (CSIM0) and the serial bus interface control register (SBIC).

**(a) Serial operating mode register 0 (CSIM0)**

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{RESET}$  input clears CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock to $\overline{SCK0}$ pin from off-chip								
	1	0	8-bit timer register 2 (TM2) output								
	1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SIO/SB0/SDA0 /P25 pin function	SO0/SB1/SDA1 /P26 pin function	$\overline{SCK0}$ /SCL/P27 pin function		
	0	×	0	Note 2	1	Note 2	×	0	0	0	1	3-wire serial I/O mode	MSB	SIO <sup>Note 2</sup> (input)	SO0 (CMOS output)	$\overline{SCK0}$ (CMOS I/O)
	1	1	1	2-wire serial I/O mode (see 17.4.3 2-wire serial I/O mode operation.) or I <sup>2</sup> C bus mode (see 17.4.4 I <sup>2</sup> C bus mode operation.)												

R/W	WUP	Wake-up function control <sup>Note 3</sup>									
	0	Interrupt request signal generation with each serial transfer in any mode									
	1	Interrupt request signal generation when the address received after detecting start condition (when CMDD = 1) matches the slave address register (SVA) data in I <sup>2</sup> C bus mode									

R/W	CSIE0	Serial interface channel 0 operation control									
	0	Operation stopped									
	1	Operation enabled									

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. Can be used as P25 (CMOS input/output) when used only for transmission.
  3. Be sure to clear WUP to 0 when the 3-wire serial I/O mode is selected.

**Remark**

- ×: don't care
- PMxx: Port mode register
- Pxx: Port output latch

**(b) Serial bus interface control register (SBIC)**

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	RELT	When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

R/W	CMDT	When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

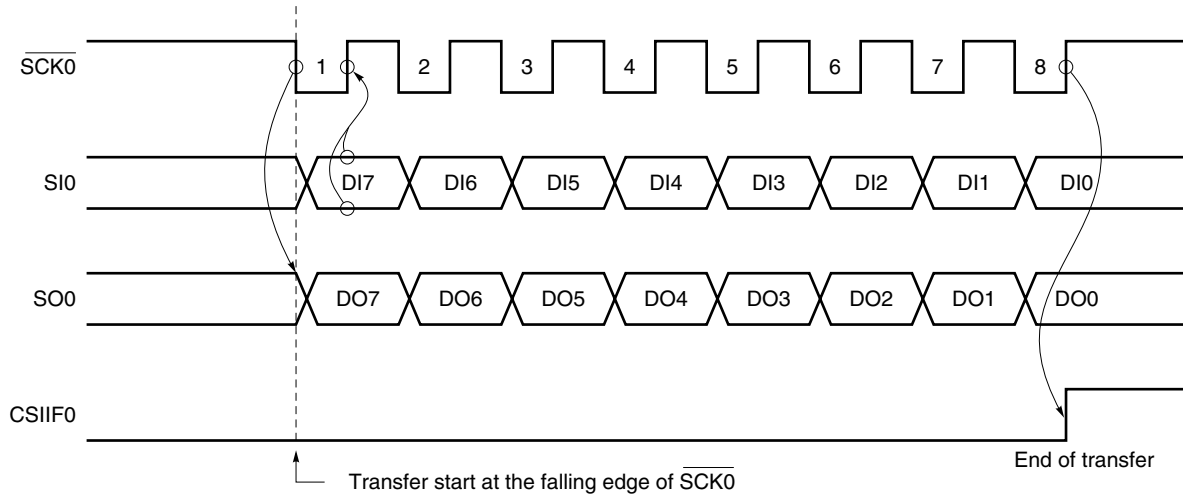
**(2) Communication operation**

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

Shift operations of serial I/O shift register 0 (SIO0) are carried out at the falling edge of the serial clock ( $\overline{\text{SCK0}}$ ). The transmitted data is held in the SO0 latch and is output from the SO0 pin. The received data input to the SIO pin is latched in SIO0 at the rising edge of  $\overline{\text{SCK0}}$ .

Upon termination of 8-bit transfer, SIO0 operation stops automatically and the interrupt request flag (CSIF0) is set.

**Figure 17-7. 3-Wire Serial I/O Mode Timing**



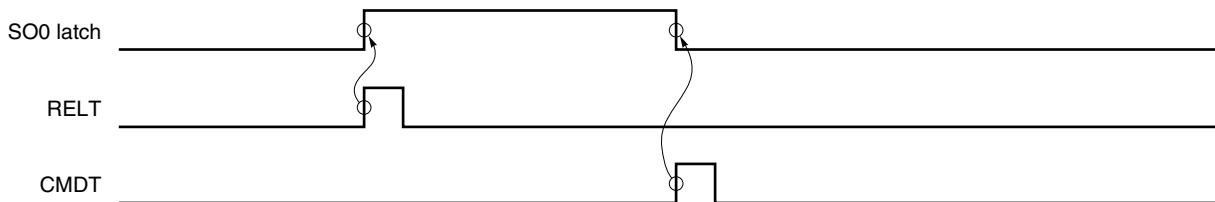
The SO0 pin is a CMOS output pin and outputs the current SO0 latch status. Thus, the SO0 pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the  $\overline{\text{SCK0}}$  pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (see 17.4.8 **SCK0/SCL/P27 pin output manipulation**).

**(3) Other signals**

Figure 17-8 shows RELT and CMDT operations.

**Figure 17-8. RELT and CMDT Operations**



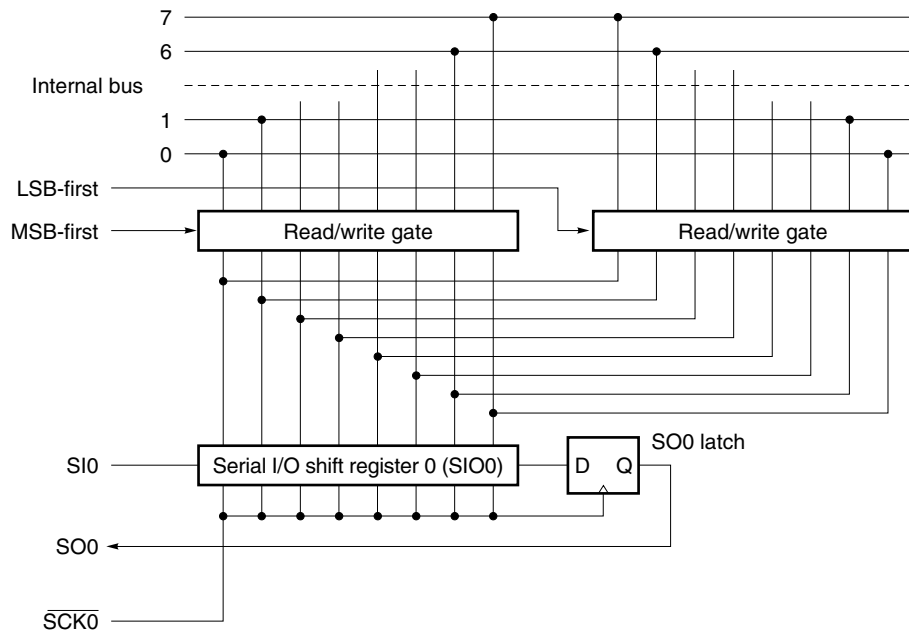
**(4) MSB/LSB switching as the start bit**

In the 3-wire serial I/O mode, it is possible to select transfer to start from the MSB or LSB.

Figure 17-9 shows the configuration of serial I/O shift register 0 (SIO0) and the internal bus. As shown in the figure, the MSB/LSB can be read or written in reverse form.

MSB/LSB switching as the start bit can be specified by bit 2 (CSIM02) of serial operating mode register 0 (CSIM0).

**Figure 17-9. Circuit for Switching Transfer Bit Order**



Start bit switching is realized by switching the bit order for data write to SIO0. The SIO0 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to SIO0.

**(5) Transfer start**

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1.
- Internal serial clock is stopped or  $\overline{\text{SCK0}}$  is a high level after 8-bit serial transfer.

**Caution** If CSIE0 is set to 1 after data write to SIO0, transfer does not start.

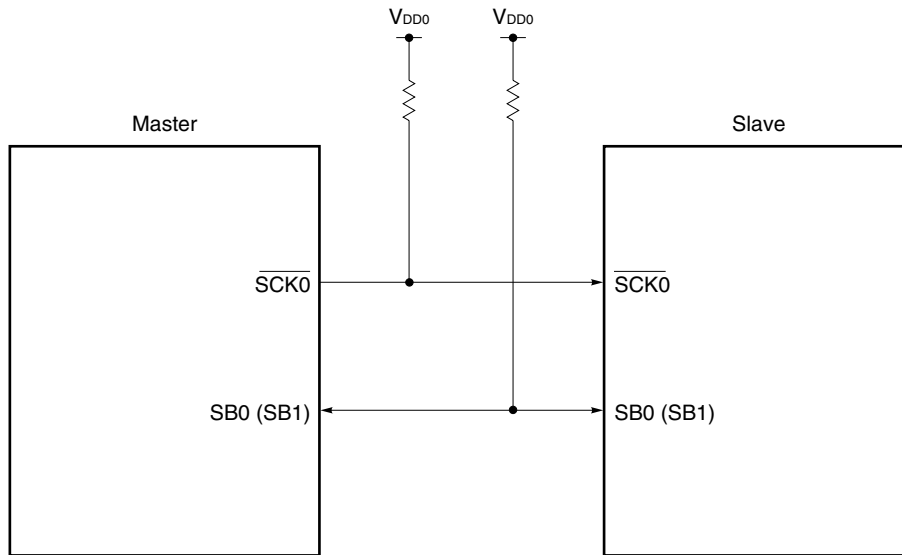
Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

### 17.4.3 2-wire serial I/O mode operation

The 2-wire serial I/O mode can cope with any communication format by program.

Communication is basically carried out with the two lines of the serial clock ( $\overline{\text{SCK0}}$ ) and serial data input/output (SB0 or SB1).

Figure 17-10. Serial Bus Configuration Example Using 2-Wire Serial I/O Mode



#### (1) Register setting

The 2-wire serial I/O mode is set by serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specify register (SINT).

##### (a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM0 to 00H.



Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip								
	1	0	8-bit timer register 2 (TM2) output								
	1	1	Clock specified by bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SI0/SB0/SDA0 /P25 pin function	SO0/SB1/SDA1 /P26 pin function	$\overline{\text{SCK0}}$ /SCL/P27 pin function
	0	×	3-wire serial I/O mode (see 17.4.2 3-wire serial I/O mode operation)											
	1	1	0	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	0	0	0	1	2-wire serial I/O mode or I <sup>2</sup> C bus mode	MSB	P25 (CMOS I/O)	SB1/SDA1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ /SCL (N-ch open-drain I/O)
			1	0	0	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	0	1			SB0/SDA0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W	WUP	Wakeup function control <sup>Note 3</sup>									
	0	Interrupt request signal generation with each serial transfer in any mode									
	1	Interrupt request signal generation when the address received after detecting start condition (when CMDD = 1) matches the slave address register (SVA) data in I <sup>2</sup> C bus mode									

R	COI	Slave address comparison result flag <sup>Note 4</sup>									
	0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data									
	1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data									

R/W	CSIE0	Serial interface channel 0 operation control									
	0	Operation stopped									
	1	Operation enabled									

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. Can be used freely as port function.
  3. Be sure to clear WUP to 0 when the 2-wire serial I/O mode.
  4. When CSIE0 = 0, COI becomes 0.

**Remark**

- ×: don't care
- PMxx: Port mode register
- Pxx: Port output latch

**(b) Serial bus interface control register (SBIC)**

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	RELT	When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

R/W	CMDT	When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.
-----	------	---

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

**(c) Interrupt timing specify register (SINT)**

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SINT to 00H.

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	After reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W <sup>Note 1</sup>

R/W	SIC	INTCSI0 interrupt source selection
	0	CSIF0 is set to 1 upon termination of serial interface channel 0 transfer
	1	CSIF0 is set to 1 upon bus release detection or termination of serial interface channel 0 transfer

R	CLD	SCK0 pin level <sup>Note 2</sup>
	0	Low level
	1	High level

- Notes**
1. Bit 6 (CLD) is a read-only bit.
  2. When CSIE0 = 0, CLD becomes 0.

**Caution** Be sure to clear bits 0 to 3 to 0 in the 2-wire serial I/O mode is used.

**Remark** CSIF0: Interrupt request flag corresponding to INTCSI0  
 CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

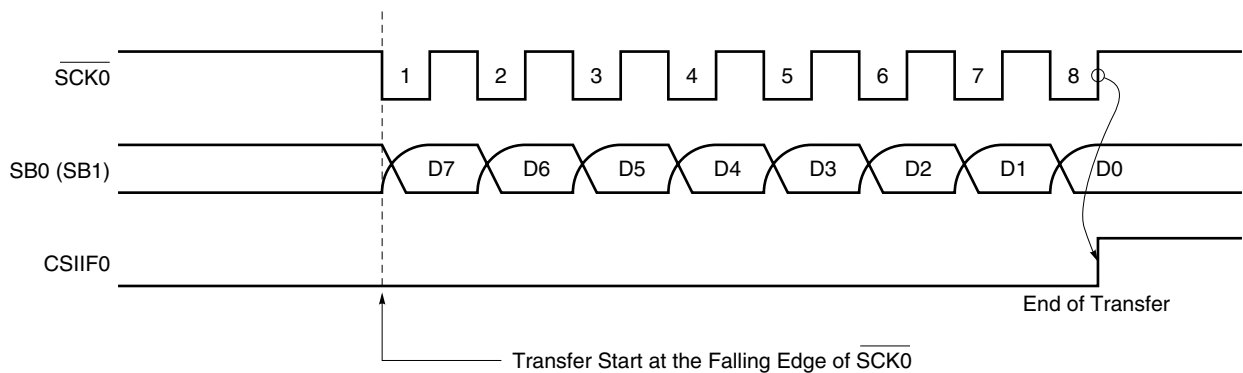
**(2) Communication operation**

The 2-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

Shift operations of serial I/O shift register 0 (SIO0) are carried out in synchronization with the falling edge of the serial clock ( $\overline{\text{SCK0}}$ ). The transmit data is held in the SO0 latch and is output from the SB0/SDA0/P25 (or SB1/SDA1/P26) pin on an MSB-first basis. The receive data input from the SB0 (or SB1) pin is latched into the SIO0 at the rising edge of  $\overline{\text{SCK0}}$ .

Upon termination of 8-bit transfer, the SIO0 operation stops automatically and the interrupt request flag (CSIF0) is set.

**Figure 17-11. 2-Wire Serial I/O Mode Timing**



The SB0 (or SB1) pin specified for the serial data bus is an N-ch open-drain input/output and thus it must be externally connected to a pull-up resistor. Because N-ch open-drain output must go into a high-impedance state during data reception, write FFH to SIO0 in advance.

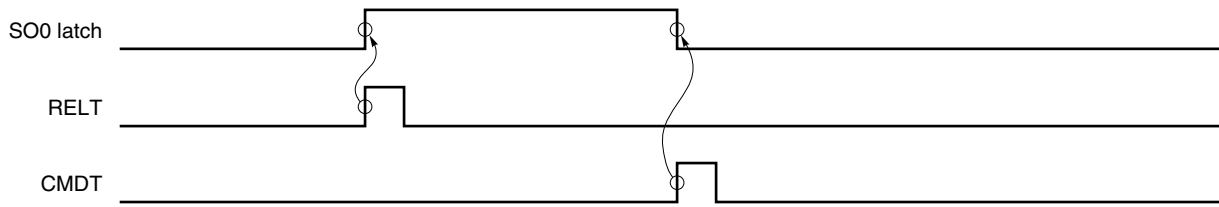
The SB0 (or SB1) pin generates the SO0 latch status and thus the SB0 (or SB1) pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the  $\overline{\text{SCK0}}$  pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (see **17.4.8  $\overline{\text{SCK0/SCL/P27}}$  pin output manipulation**).

**(3) Other signals**

Figure 17-12 shows the RELT and CMDT operations.

**Figure 17-12. RELT and CMDT Operations**

**(4) Transfer start**

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or  $\overline{\text{SCK0}}$  is at high level after 8-bit serial transfer.

**Cautions 1. If CSIE0 is set to 1 after data write to SIO0, transfer does not start.**

- 2. Because the N-ch open-drain output must go into a high-impedance state during data reception, write FFH to SIO0 in advance.**

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIF0) is set.

**(5) Error detection**

In the 2-wire serial I/O mode, the serial bus SB0 (SB1) status being transmitted is fetched into the destination device, that is, serial I/O shift register 0 (SIO0). Thus, transmit error can be detected in the following way.

**(a) Method of comparing SIO0 data before transmission to that after transmission**

In this case, if two data differ from each other, a transmit error is judged to have occurred.

**(b) Method of using the slave address register (SVA)**

Transmit data is set to both SIO0 and SVA and is transmitted. After termination of transmission, COI bit (match signal coming from the address comparator) of serial operating mode register 0 (CSIM0) is tested. If "1", normal transmission is judged to have been carried out. If "0", a transmit error is judged to have occurred.

#### 17.4.4 I<sup>2</sup>C bus mode operation

The I<sup>2</sup>C bus mode is provided for when communication operations are performed between a single master device and multiple slave devices. This mode configures a serial bus that includes only a single master device, and is based on the clocked serial I/O format with the addition of bus configuration functions, which allow the master device to communicate with a number of (slave) devices using only two lines: a serial clock (SCL) line and serial data bus (SDA0 or SDA1) line. Consequently, when the user plans to configure a serial bus which includes multiple microcontrollers and peripheral devices, using this configuration results in reduction of the required number of port pins and on-board wires.

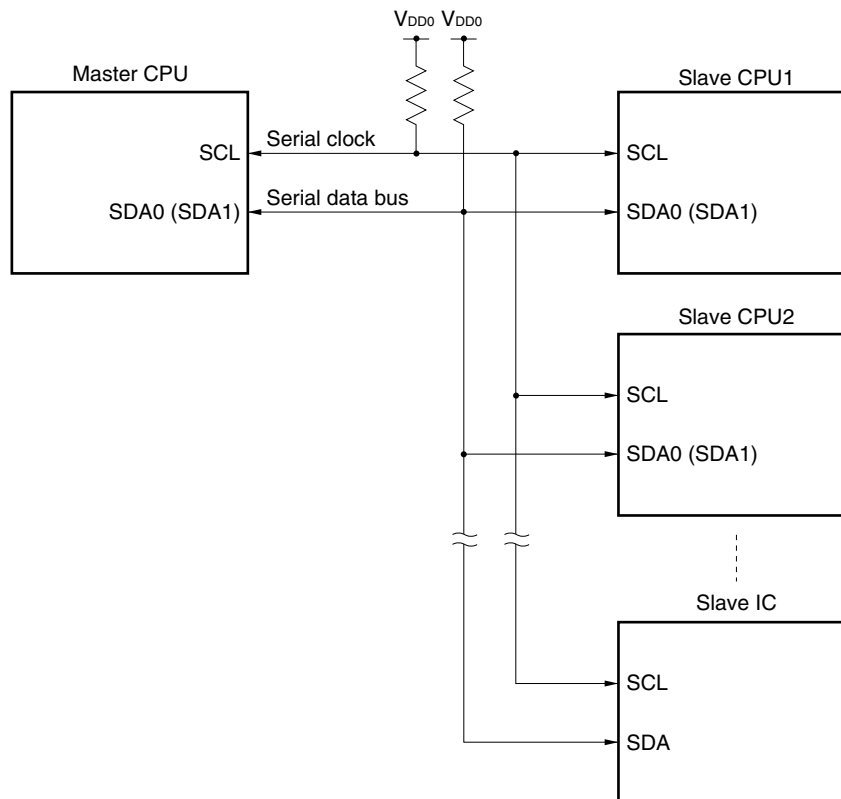
In the I<sup>2</sup>C bus specification, the master sends start condition, data, and stop condition signals to slave devices via the serial data bus, while slave devices automatically detect and distinguish the type of signals using a signal detection function incorporated as hardware. The application program that controls the I<sup>2</sup>C bus can be simplified by using this function.

An example of a serial bus configuration is shown in Figure 17-13. This system below is composed of CPUs and peripheral ICs having serial interface hardware that complies with the I<sup>2</sup>C bus specification.

Note that pull-up resistors are required to connect to both the serial clock line and serial data bus line, because open-drain buffers are used for the serial clock pin (SCL) and the serial data bus pin (SDA0 or SDA1) on the I<sup>2</sup>C bus.

The signals used in the I<sup>2</sup>C bus mode are described in Table 17-4.

**Figure 17-13. Example of Serial Bus Configuration Using I<sup>2</sup>C Bus**



**(1) I<sup>2</sup>C bus mode functions**

In the I<sup>2</sup>C bus mode, the following functions are available.

**(a) Automatic identification of serial data**

Slave devices automatically detect and identify start condition, data, and stop condition signals sent in series via the serial data bus.

**(b) Chip selection by specifying device addresses**

The master device can select a specific slave device connected to the I<sup>2</sup>C bus and communicate with it by sending in advance the address data corresponding to the destination device.

**(c) Wakeup function**

When address data is sent from the master device, slave devices compare it with the value registered in their internal slave address registers. If the values in one of the slave devices match, the slave device internally generates an interrupt request signal to terminate the current processing and communicates with the master device (the interrupt request also occurs when the stop condition is detected). Therefore, CPUs other than the selected slave device on the I<sup>2</sup>C bus can perform independent operations during the serial communication.

**(d) Acknowledge signal ( $\overline{\text{ACK}}$ ) control function**

The master device and a slave device send and receive acknowledge signals to confirm that the serial communication has been executed normally.

**(e) Wait signal ( $\overline{\text{WAIT}}$ ) control function**

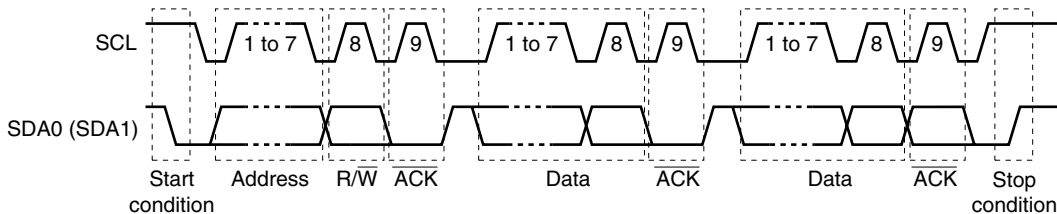
When a slave device is preparing for data transmission or reception and requires more waiting time, the slave device outputs a wait signal on the bus to inform the master device of the wait status.

**(2) I<sup>2</sup>C bus definition**

This section describes the format of serial data communications and functions of the signals used in the I<sup>2</sup>C bus mode.

First, the transfer timing of the “start condition”, “data”, and “stop condition” signals, which are output onto the signal data bus of the I<sup>2</sup>C bus, is shown in Figure 17-14.

**Figure 17-14. I<sup>2</sup>C Bus Serial Data Transfer Timing**



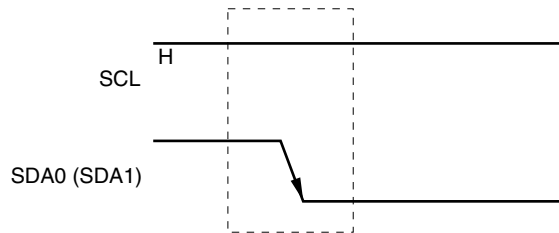
The start condition, slave address, and stop condition signals are output by the master. The acknowledge signal ( $\overline{\text{ACK}}$ ) is output by either the master or the slave device (normally by the device which has received the 8-bit data that was sent). A serial clock (SCL) is continuously supplied from the master device.

**(a) Start condition**

When the SDA0 (SDA1) pin level is changed from high to low while the SCL pin is high, this transition is recognized as the start condition signal. This start condition signal, which is created using the SCL and SDA0 (or SDA1) pins, is output from the master device to slave devices to initiate a serial transfer. See **17.4.5 Cautions on Use of I<sup>2</sup>C Bus Mode**, for details of the start condition output.

The start condition signal is detected by hardware incorporated in slave devices.

**Figure 17-15. Start Condition**

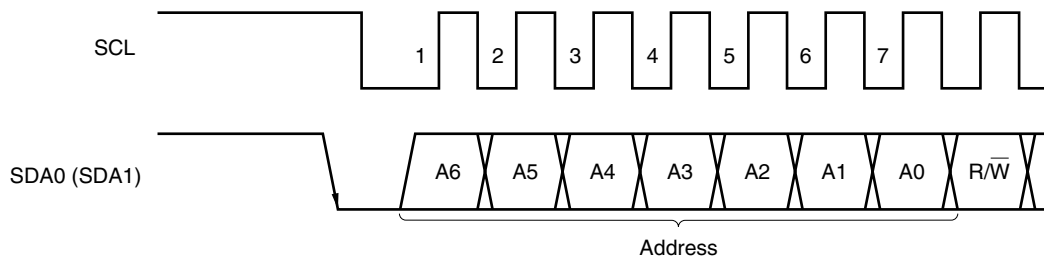


**(b) Address**

The 7 bits following the start condition signal are defined as an address.

The 7-bit address data is output by the master device to specify a specific slave from among those connected to the bus line. Each slave device on the bus line must therefore have a different address. Therefore, after a slave device detects the start condition, it compares the 7-bit address data received and the data of the slave address register (SVA). After the comparison, only the slave device in which the data are a match becomes the communication partner, and subsequently performs communication with the master device until the master device sends a start condition or stop condition signal.

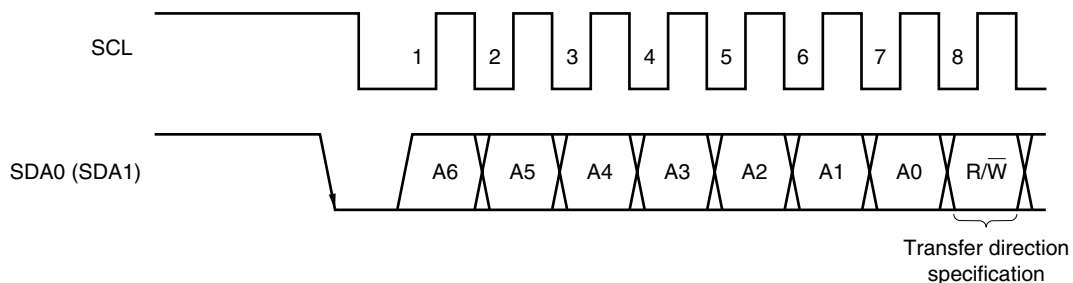
**Figure 17-16. Address**



**(c) Transfer direction specification**

The 1 bit that follows the 7-bit address data will be sent from the master device, and it is defined as the transfer direction specification bit. If this bit is 0, it is the master device which will send data to the slave. If it is 1, it is the slave device which will send data to the master.

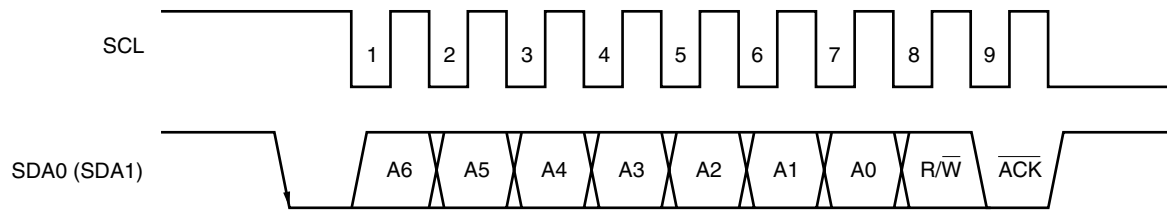
**Figure 17-17. Transfer Direction Specification**



**(d) Acknowledge signal ( $\overline{\text{ACK}}$ )**

The acknowledge signal indicates that the transferred serial data has definitely been received. This signal is used between the transmitting side and receiving side devices for confirmation of correct data transfer. In principle, the receiving side device returns an acknowledge signal to the transmitting device each time it receives 8-bit data. The only exception is when the receiving side is the master device and the 8-bit data is the last transfer data; the master device outputs no acknowledge signal in this case. The transmitting side that has transferred 8-bit data waits for the acknowledge signal which will be sent from the receiving side. If the transmitting side device receives the acknowledge signal, which means a successful data transfer, it proceeds to the next processing. If this signal is not sent back from the slave device, this means that the data sent has not been received by the slave device, and therefore the master device outputs a stop condition signal to terminate subsequent transmissions.

**Figure 17-18. Acknowledge Signal**

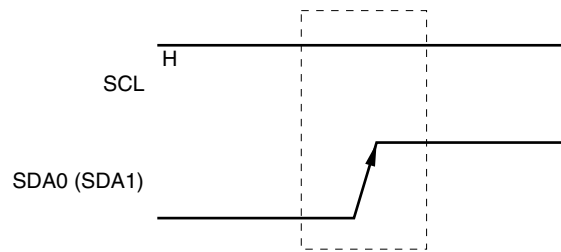


**(e) Stop condition**

If the SDA0 (SDA1) pin level changes from low to high while the SCL pin is high, this transition is defined as a stop condition signal.

The stop condition signal is output from the master to the slave device to terminate a serial transfer. The stop condition signal is detected by hardware incorporated in the slave device.

**Figure 17-19. Stop Condition**





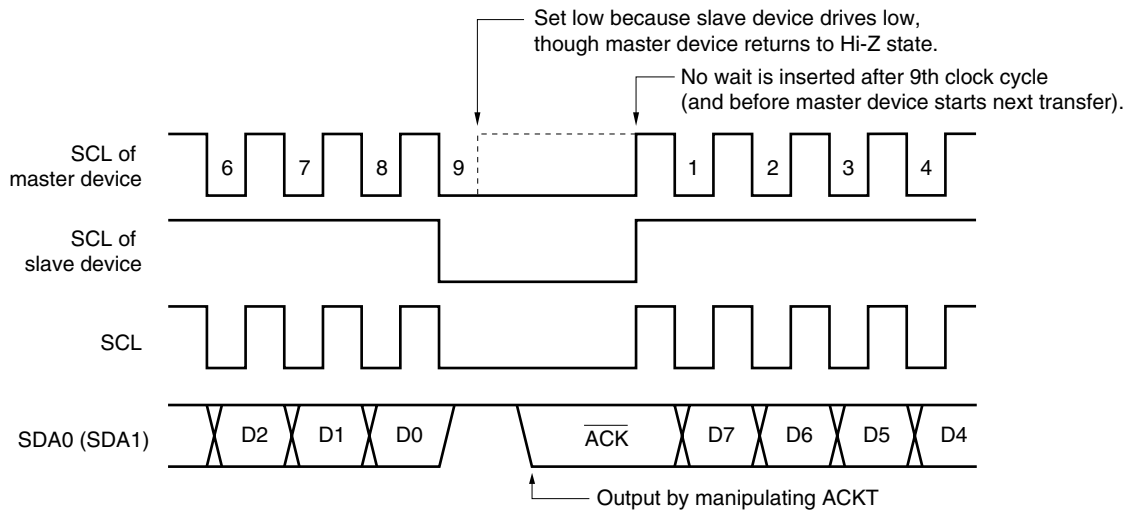
**(f) Wait signal ( $\overline{\text{WAIT}}$ )**

The wait signal is output by a slave device to inform the master device that the slave device is in a wait state due to preparing for transmitting or receiving data.

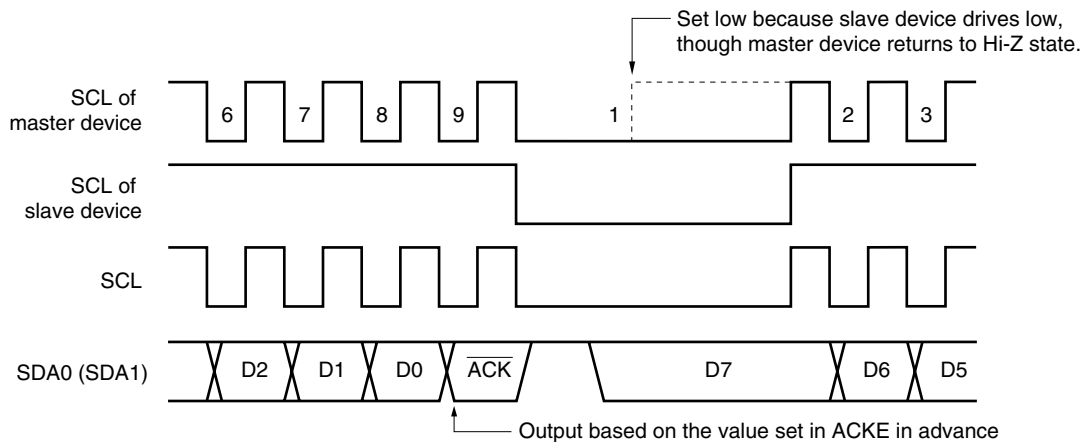
During the wait state, the slave device continues to output the wait signal by keeping the SCL pin low to delay subsequent transfers. When the wait state is released, the master device can start the next transfer. For the releasing operation of slave devices, see **17.4.5 Cautions on Use of I<sup>2</sup>C Bus Mode**.

**Figure 17-20. Wait Signal**

**(a) Wait of 8 clock cycles**



**(b) Wait of 9 clock cycles**



**(3) Register setting**

The I<sup>2</sup>C mode setting is performed by serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specification register (SINT).

**(a) Serial operating mode register 0 (CSIM0)**

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.  
 RESET input clears CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W <sup>Note 1</sup>

R/W	CSIM01	CSIM00	Serial interface channel 0 clock selection								
	0	×	Input clock from off-chip to SCL pin								
	1	0	8-bit timer register 2 (TM2) output <sup>Note 2</sup>								
	1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)								

R/W	CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operation mode	Start bit	SI0/SB0/SDA0/P25 pin function	SO0/SB1/SDA1/P26 pin function	SCK0/SCL/P27 pin function
	0	×	3-wire serial I/O mode (see 17.4.2 Operation in 3-wire serial I/O mode)											
	1	1	0	×	×	0	0	0	1	2-wire serial I/O or I <sup>2</sup> C bus mode	MSB	P25 (CMOS I/O)	SB1/SDA1 N-ch open-drain I/O	SCK0/SCL N-ch open-drain I/O

R/W	WUP	Wake-up function control <sup>Note 4</sup>													
	0	Interrupt request signal generation with each serial transfer in any mode													
	1	In I <sup>2</sup> C bus mode, interrupt request signal is generated when the address data received after start condition detection (when CMDD = 1) matches data in slave address register (SVA).													

R	COI	Slave address comparison result flag <sup>Note 5</sup>													
	0	Slave address register (SVA) not equal to data in serial I/O shift register 0 (SIO0)													
	1	Slave address register (SVA) equal to data in serial I/O shift register 0 (SIO0)													

R/W	CSIE0	Serial interface channel 0 operation control													
	0	Operation stopped.													
	1	Operation enabled.													

- Notes**
1. Bit 6 (COI) is a read-only bit.
  2. In the I<sup>2</sup>C bus mode, the clock frequency is 1/16 of the clock frequency output by TO2.
  3. Can be used freely as a port.
  4. To use the wakeup function (WUP = 1), set bit 5 (SIC) of the interrupt timing specification register (SINT) to 1. Do not execute an instruction that writes serial I/O shift register 0 (SIO0) while WUP = 1.
  5. When CSIE0 = 0, COI is 0.

**Remark**

- ×: don't care
- PM××: Port mode register
- P××: Port output latch

**(b) Serial bus interface control register (SBIC)**

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W <sup>Note 1</sup>
R/W	RELT	Use for stop condition output. When RELT = 1, the SO0 latch is set to 1. After the SO0 latch is set, RELT is automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.									
R/W	CMDT	Use for start condition output. When CMDT = 1, the SO0 latch is cleared to 0. After the SO0 latch is cleared, CMDT is automatically cleared to 0. It is also cleared to 0 when CSIE0 = 0.									
R	RELD	Stop condition detection									
	0	Clear conditions <ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• If SIO0 and SVA values do not match in address reception</li> <li>• When CSIE0 = 0</li> <li>• When RESET input is applied</li> </ul>									
	1	Setting condition <ul style="list-style-type: none"> <li>• When stop condition is detected</li> </ul>									
R	CMDD	Start condition detection									
	0	Clear conditions <ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• When stop condition is detected</li> <li>• When CSIE0 = 0</li> <li>• When RESET input is applied</li> </ul>									
	1	Setting condition <ul style="list-style-type: none"> <li>• When start condition is detected</li> </ul>									
R/W	ACKT	SDA0 (SDA1) is set to low after the Set instruction execution (ACKT = 1) before the next SCL falling edge. Used for generating an ACK signal by software if the 8-clock wait mode is selected. Cleared to 0 if CSIE0 = 0 when a transfer by the serial interface is started.									
R/W	ACKE	Acknowledge signal automatic output control <sup>Note 2</sup>									
	0	Disabled (with ACKT enabled). Used when receiving data in the 8-clock wait mode or when transmitting data. <sup>Note 3</sup>									
	1	Enabled. After completion of transfer, the acknowledge signal is output in ACKE is synchronization with the 9th falling edge of the SCL clock (automatically output when ACKE = 1). However, ACKE is not automatically cleared to 0 after acknowledge signal is output. It is used for reception when the 9-clock wait mode is selected.									
R	ACKD	Acknowledge detection									
	0	Clear Conditions <ul style="list-style-type: none"> <li>• When transfer start instruction is executed</li> <li>• When CSIE0 = 0</li> <li>• When RESET input is applied</li> </ul>									
	1	Set Conditions <ul style="list-style-type: none"> <li>• When the acknowledge signal is detected at the rising edge of SCL clock after completion of transfer</li> </ul>									
R/W	<sup>Note 4</sup> BSYE	Control of N-ch open-drain output for transmission in I <sup>2</sup> C bus mode <sup>Note 5</sup>									
	0	Output enabled (transmission)									
	1	Output disabled (reception)									

- Notes**
1. Bits 2, 3, and 6 (RELD, CMDD, ACKD) are read-only bits.
  2. This setting must be performed prior to transfer start.
  3. In the 8-clock wait mode, use ACKT for output of the acknowledge signal after normal data reception.
  4. The busy mode can be released by the start of a serial interface transfer or reception of an address signal. However, the BSYE flag is not cleared.
  5. When using the wakeup function, be sure to set BSYE to 1.

**Remark** CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

**(c) Interrupt timing specification register (SINT)**

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SINT to 00H.

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	After reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W <sup>Note 1</sup>

R/W	WAT1	WAT0	Interrupt control by wait <sup>Note 2</sup>
	0	0	Interrupt service request is generated on rise of 8th SCK0 clock cycle (clock output is high impedance).
	0	1	Setting prohibited
	1	0	Used in I <sup>2</sup> C bus mode (8-clock wait) Generates an interrupt service request on rise of 8th SCL clock cycle. (In case of master device, SCL pin is driven low after output of 8 clock cycles, to enter the wait state. In case of slave device, SCL pin is driven low after input of 8 clock cycles, to require the wait state.)
	1	1	Used in I <sup>2</sup> C bus mode (9-clock wait) Generates an interrupt service request on rise of 9th SCL clock cycle. (In case of master device, SCL pin is driven low after output of 9 clock cycles, to enter the wait state. In case of slave device, SCL pin is driven low after input of 9 clock cycles, to require the wait state.)

R/W	WREL	Wait release control
	0	Indicates that the wait state has been released.
	1	Releases the wait state. Automatically cleared to 0 after releasing the wait state. This bit is used to release the wait state set by means of WAT0 and WAT1.

R/W	CLC	Clock level control
	0	Used in I <sup>2</sup> C bus mode. In cases other than serial transfer, SCL pin output is driven low.
	1	Used in I <sup>2</sup> C bus mode. In cases other than serial transfer, SCL pin output is set to high impedance. (Clock line is held high.) Used by master device to generate the start condition and stop condition signals.

R/W	SVAM	SVA bits used as slave address
	0	Bits 0 to 7
	1	Bits 1 to 7

R/W	SIC	INTCSI0 interrupt source selection <sup>Note 3</sup>
	0	CSIF0 is set to 1 after end of serial interface channel 0 transfer.
	1	CSIF0 is set to 1 after end of serial interface channel 0 transfer or when stop condition is detected.

R	CLD	SCL pin level <sup>Note 4</sup>
	0	Low level
	1	High level

- Notes**
1. Bit 6 (CLD) is read-only.
  2. When the I<sup>2</sup>C bus mode is used, be sure to set WAT0 and WAT1 to 1 and 0, or 1 and 1, respectively.
  3. When using the wakeup function in I<sup>2</sup>C mode, be sure to set SIC to 1.
  4. When CSIE0 = 0, CLD is 0.

**Remark** SVA: Slave address register  
 CSIF0: Interrupt request flag corresponding to INTCSI0  
 CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

**(4) Various signals**

A list of signals in the I<sup>2</sup>C bus mode is given in Table 17-4.

**Table 17-4. Signals in I<sup>2</sup>C Bus Mode**

Signal name	Description
Start condition	Definition: SDA0 (SDA1) falling edge when SCL is high <sup>Note 1</sup>
	Function: Indicates that serial communication starts and subsequent data is address data.
	Signaled by: Master
	Signaled when: CMDT is set.
	Affected flag(s): CMDD (is set.)
Stop condition	Definition: SDA0 (SDA1) rising edge when SCL is high <sup>Note 1</sup>
	Function: Indicates end of serial transmission.
	Signaled by: Master
	Signaled when: RELT is set.
	Affected flag(s): RELD (is set) and CMDD (is cleared)
Acknowledge signal (ACK)	Definition: Low level of SDA0 (SDA1) pin during one SCL clock cycle after serial reception
	Function: Indicates completion of reception of 1 byte.
	Signaled by: Master or slave
	Signaled when: ACKT is set with ACKE = 1.
	Affected flag(s): ACKD (is set.)
Wait (WAIT)	Definition: Low-level signal output to SCL
	Function: Indicates state in which serial reception is not possible.
	Signaled by: Slave
	Signaled when: WAT1, WAT0 = 1x.
	Affected flag(s): None
Serial clock (SCL)	Definition: Synchronization clock for output of various signals
	Function: Serial communication synchronization signal.
	Signaled by: Master
	Signaled when: See <b>Note 2</b> below.
	Affected flag(s): CSIF0. Also see <b>Note 3</b> below.
Address (A6 to A0)	Definition: 7-bit data synchronized with SCL immediately after start condition signal
	Function: Indicates address value for specification of slave on serial bus.
	Signaled by: Master
	Signaled when: See <b>Note 2</b> below.
	Affected flag(s): CSIF0. Also see <b>Note 3</b> below.
Transfer direction (R/ $\bar{W}$ )	Definition: 1-bit data output in synchronization with SCL after address output
	Function: Indicates whether data transmission or reception is to be performed.
	Signaled by: Master
	Signaled when: See <b>Note 2</b> below.
	Affected flag(s): CSIF0. Also see <b>Note 3</b> below.
Data (D7 to D0)	Definition: 8-bit data synchronized with SCL, not immediately after start condition
	Function: Contains data to be actually sent.
	Signaled by: Master or slave
	Signaled when: See <b>Note 2</b> below.
	Affected flag(s): CSIF0. Also see <b>Note 3</b> below.

- Notes**
1. The level of the serial clock can be controlled with bit 3 (CLC) of interrupt timing specify register (SINT).
  2. Execution of instruction to write data to SIO0 when CSIE0 = 1 (serial transfer start directive). In the wait state, the serial transfer operation will be started after the wait state is released.
  3. If the 8-clock wait is selected when WUP = 0, CSIF0 is set at the rising edge of the 8th clock cycle of SCL. If the 9-clock wait is selected when WUP = 0, CSIF0 is set at the rising edge of the 9th clock cycle of SCL. CSIF0 is set if an address is received and that address matches the value of the slave address register (SVA) when WUP = 1, or if the stop condition is detected.

**(5) Pin configurations**

The configurations of the serial clock pin SCL and the serial data bus pins SDA0 (SDA1) are shown below.

**(a) SCL**

Pin for serial clock input/output alternate-function pin.

<1> Master ..... N-ch open-drain output

<2> Slave ..... Schmitt input

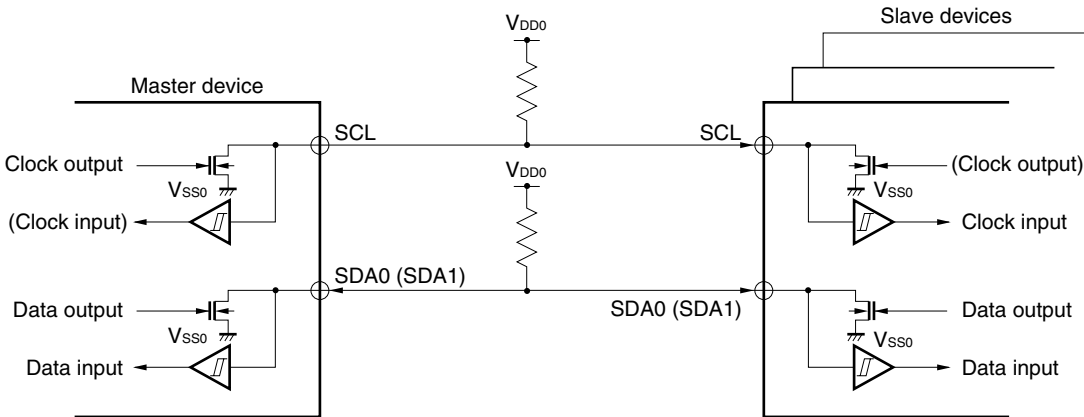
**(b) SDA0 (SDA1)**

Serial data I/O alternate-function pin.

Uses N-ch open-drain output and Schmitt-input buffers for both master and slave devices.

Note that pull-up resistors are required to be connected to both the serial clock line and serial data bus line, because open-drain buffers are used for the serial clock pin (SCL) and the serial data bus pin (SDA0 or SDA1) on the I<sup>2</sup>C bus.

**Figure 17-21. Pin Configuration**



**Caution** To receive data, the N-ch open-drain output must be made to go into a high-impedance state. Therefore, set bit 7 (BSYE) of the serial bus interface control register (SBIC) to 1 in advance, and write FFH to serial I/O shift register 0 (SIO0).

When the wakeup function is used (by setting bit 5 (WUP) of serial operating mode register 0 (CSIM0)), however, do not write FFH to SIO0 before reception. Even if FFH is not written to SIO0, the N-ch open-drain output always goes into a high-impedance state.

**(6) Address match detection method**

In the I<sup>2</sup>C mode, the master can select a specific slave device by sending slave address data.

A match of the addresses can be automatically detected by hardware. CSIF0 is set if the slave address transmitted by the master matches the value set to the slave address register (SVA) when a slave device address has a slave register (SVA), and the wakeup function specification bit (WUP) = 1 (CSIF0 is also set when the stop condition is detected).

When using the wakeup function, set SIC to 1.

**Caution** Slave selection/non-selection is detected by matching of the data (address) received after the start condition.

For this match detection, the match interrupt request (INTCSI0) of the address to be generated with WUP = 1 is normally used. Thus, execute selection/non-selection detection by slave address when WUP = 1.

**(7) Error detection**

In the I<sup>2</sup>C bus mode, transmission error detection can be performed by the following methods because the serial bus SDA0 (SDA1) status during transmission is also taken into the serial I/O shift register 0 (SIO0) register of the transmitting device.

**(a) Comparison of SIO0 data before and after transmission**

In this case, a transmission error is judged to have occurred if the two data values are different.

**(b) Using the slave address register (SVA)**

Transmit data is set in SIO0 and SVA before transmission is performed. After transmission, the COI bit (match signal from the address comparator) of serial operating mode register 0 (CSIM0) is tested: "1" indicates normal transmission, and "0" indicates a transmission error.

**(8) Communication operation**

In the I<sup>2</sup>C bus mode, the master selects the slave device to be communicated with from among multiple devices by outputting address data onto the serial bus.

After the slave address data, the master sends the R/W bit which indicates the data transfer direction, and starts serial communication with the selected slave device.

Data communication timing charts are shown in **Figures 17-22** and **17-23**.

In the transmitting device, serial I/O shift register 0 (SIO0) shifts transmission data to the SO latch in synchronization with the falling edge of the serial clock (SCL), the SO0 latch outputs the data on an MSB-first basis from the SDA0 or SDA1 pin to the receiving device.

In the receiving device, the data input from the SDA0 or SDA1 pin is taken into the SIO0 in synchronization with the rising edge of SCL.

Figure 17-22. Data Transmission from Master to Slave  
(Both Master and Slave Selected 9-Clock Wait) (1/3)

(a) Start condition to address

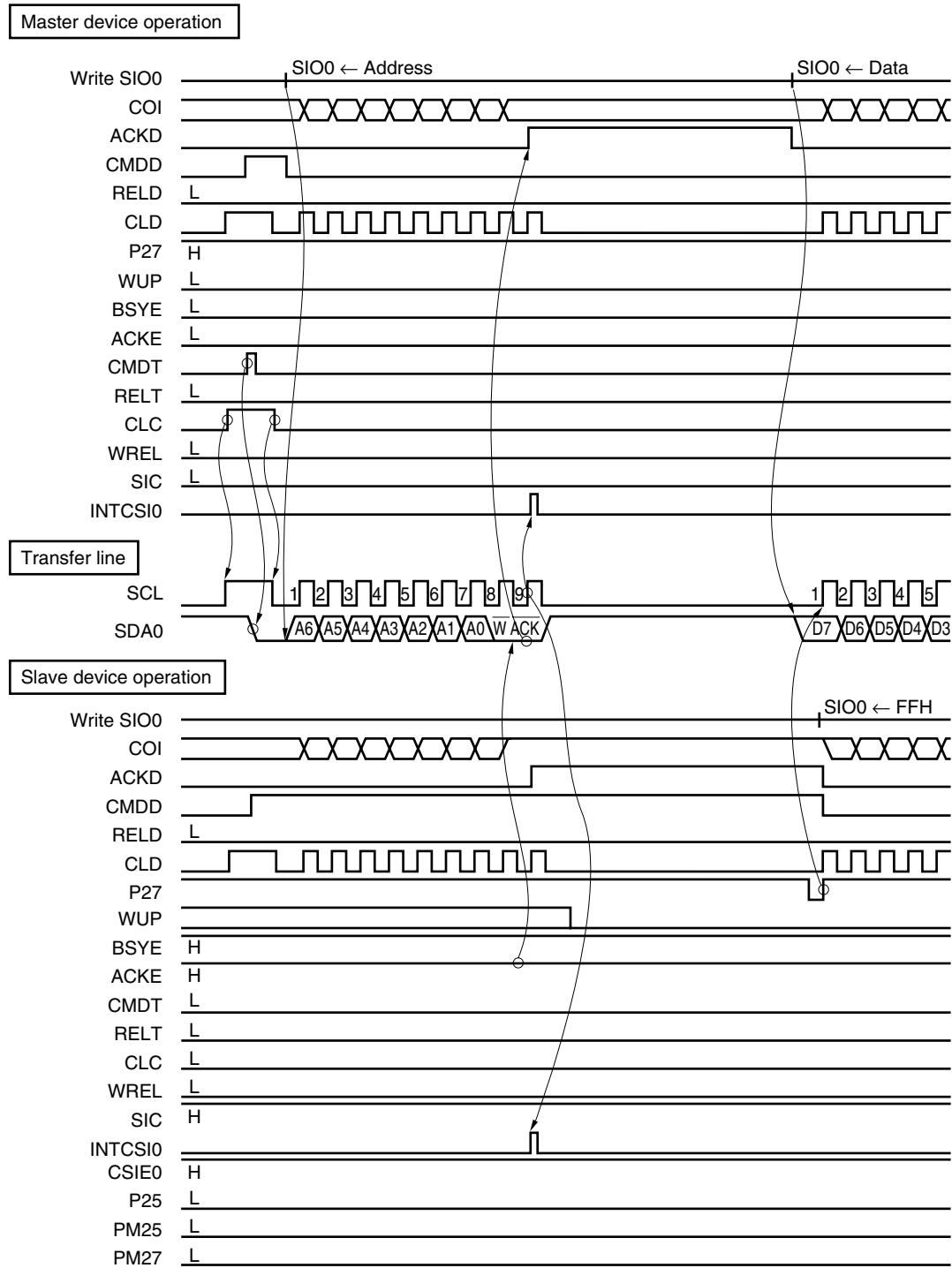




Figure 17-22. Data Transmission from Master to Slave  
(Both Master and Slave Selected 9-Clock Wait) (2/3)

(b) Data

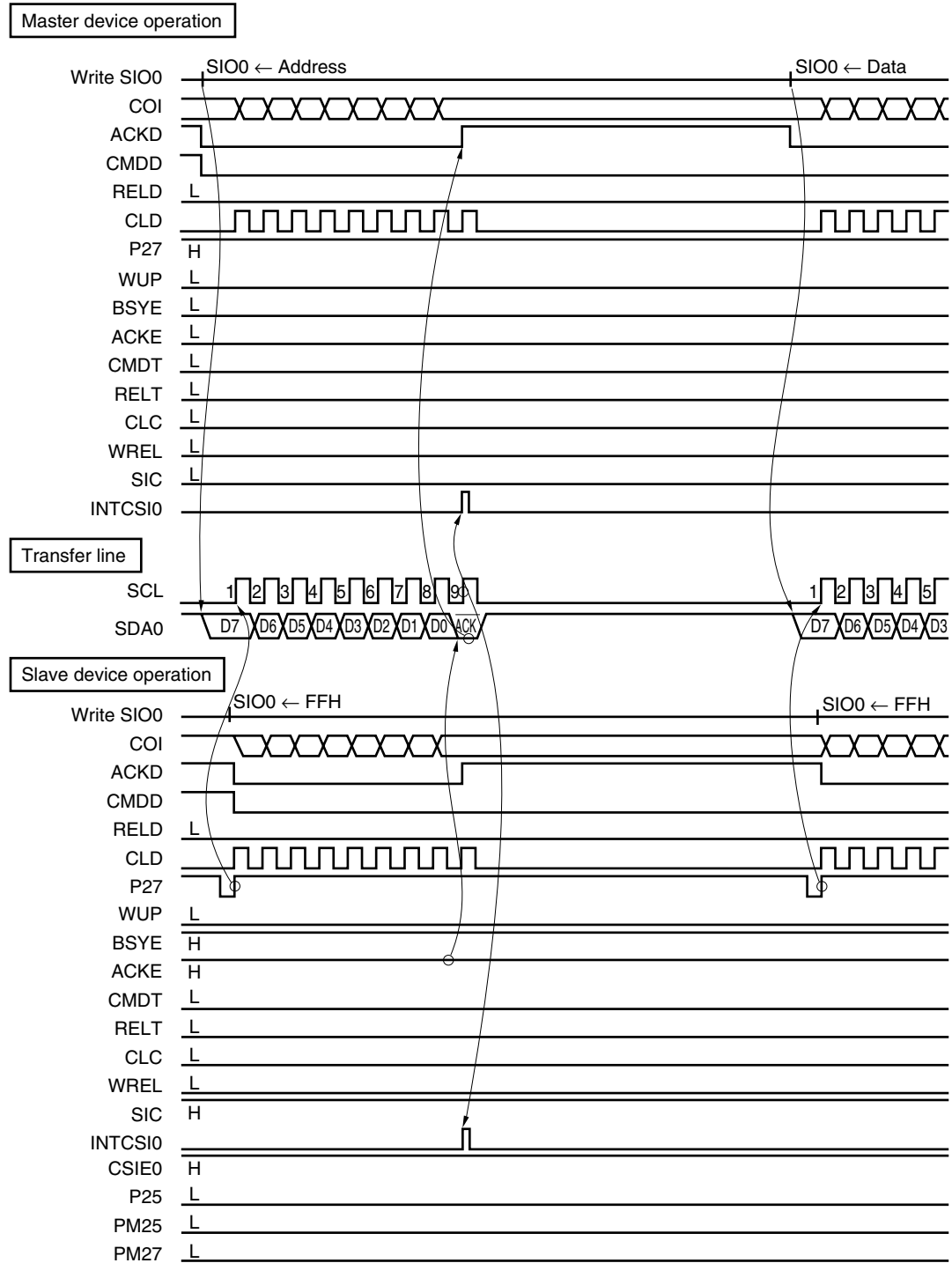


Figure 17-22. Data Transmission from Master to Slave  
(Both Master and Slave Selected 9-Clock Wait) (3/3)

(c) Stop condition

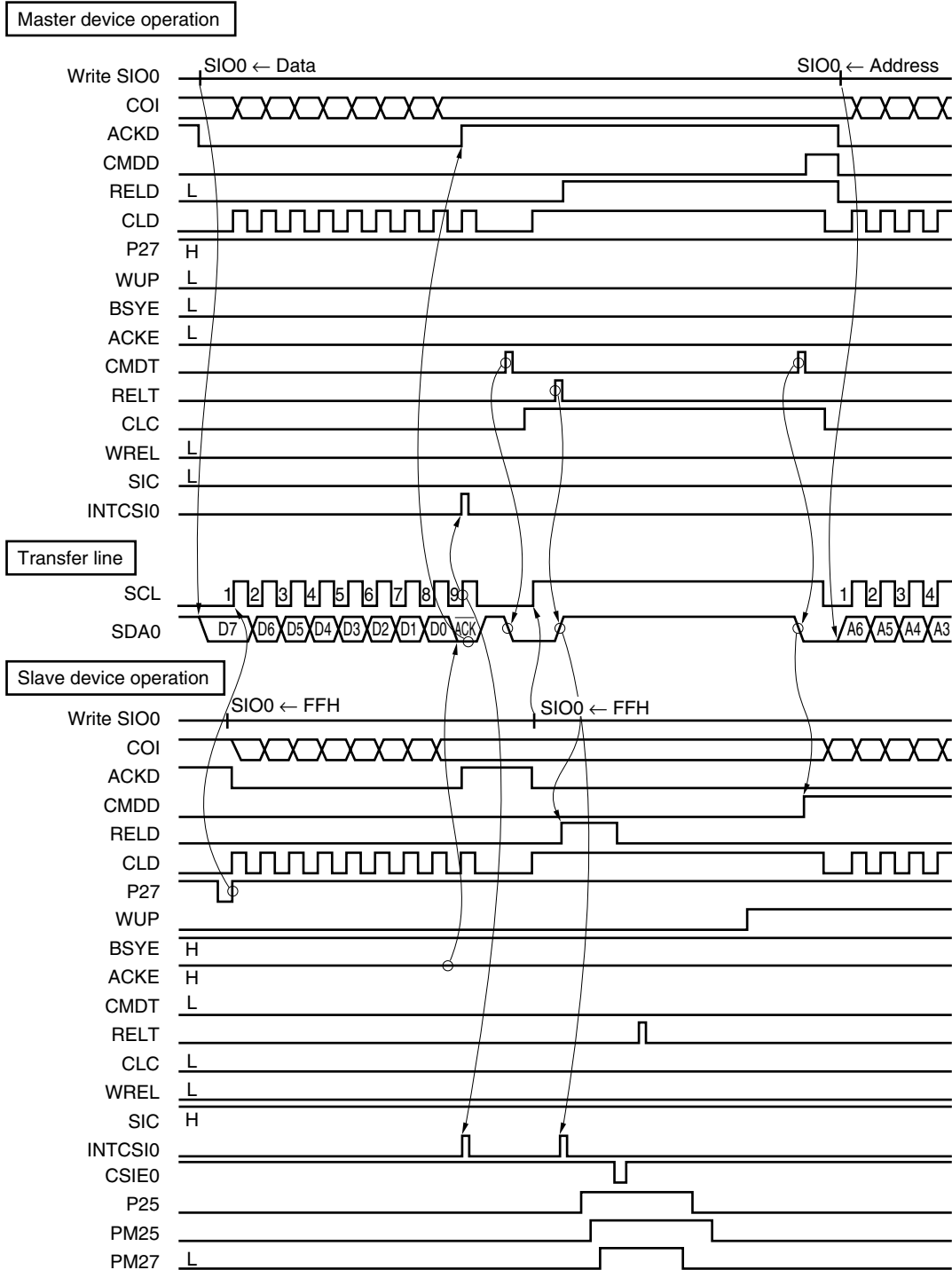
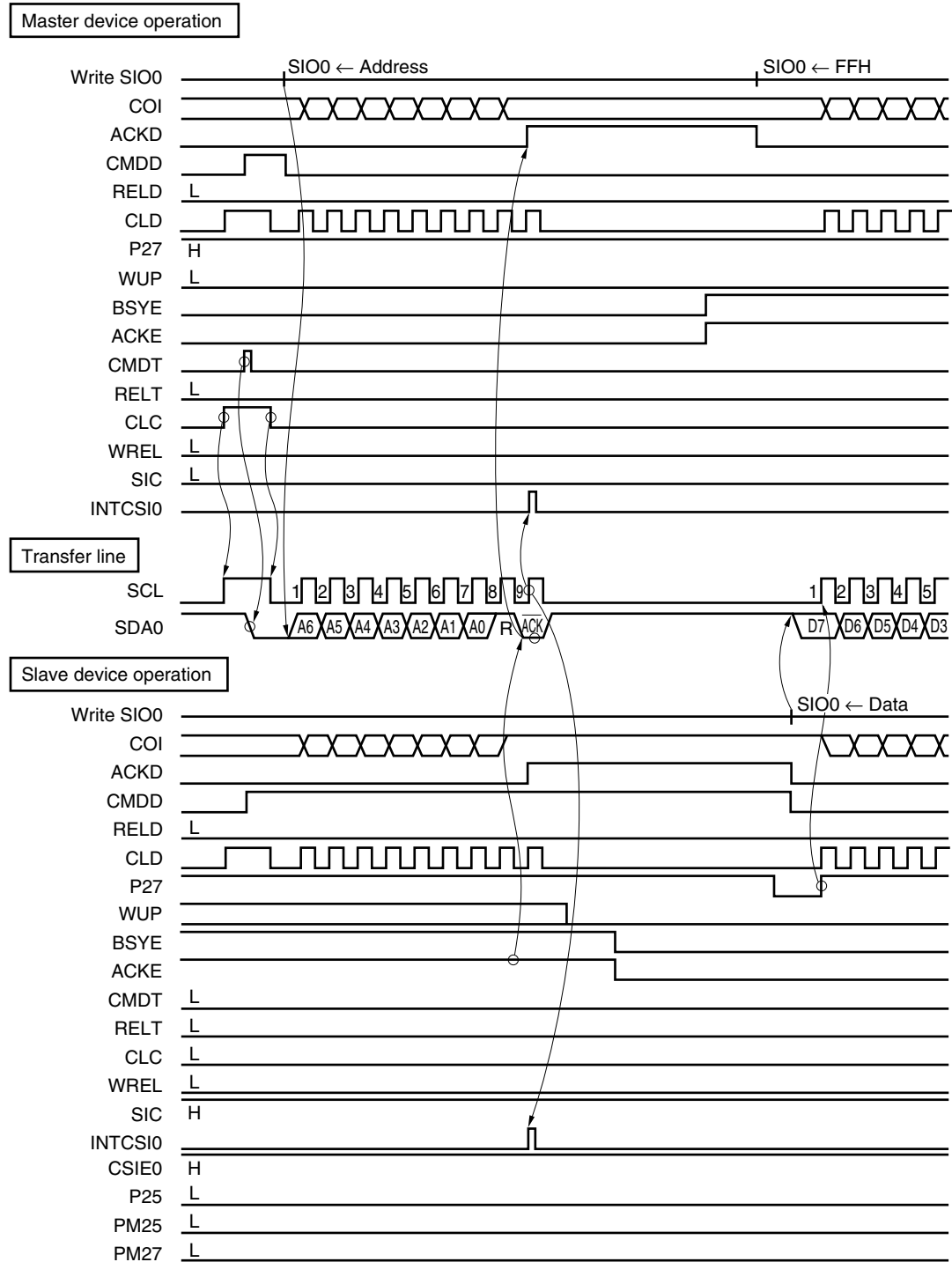


Figure 17-23. Data Transmission from Slave to Master  
(Both Master and Slave Selected 9-Clock Wait) (1/3)

(a) Start condition to address



**Figure 17-23. Data Transmission from Slave to Master  
(Both Master and Slave Selected 9-Clock Wait) (2/3)**

**(b) Data**

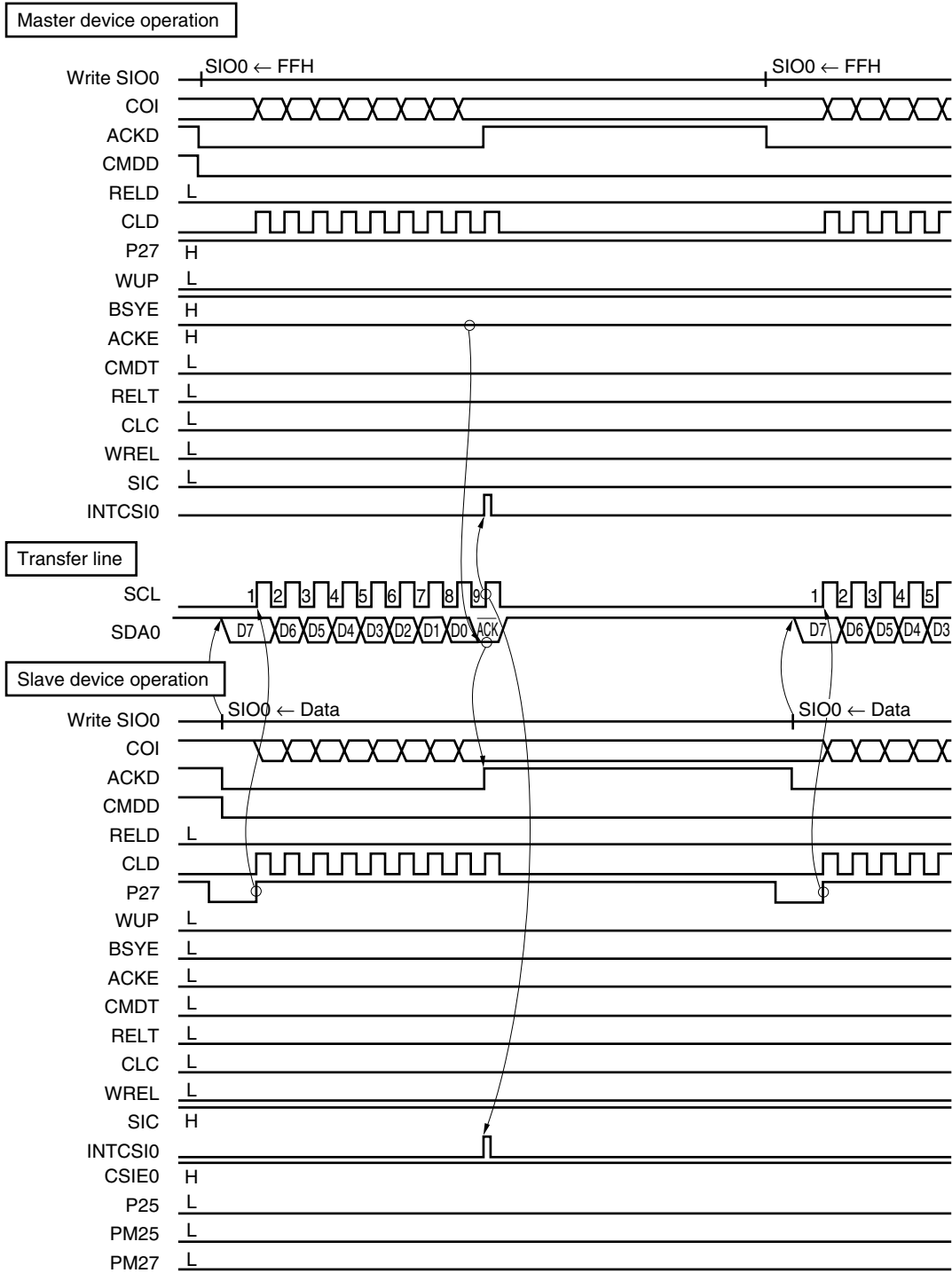
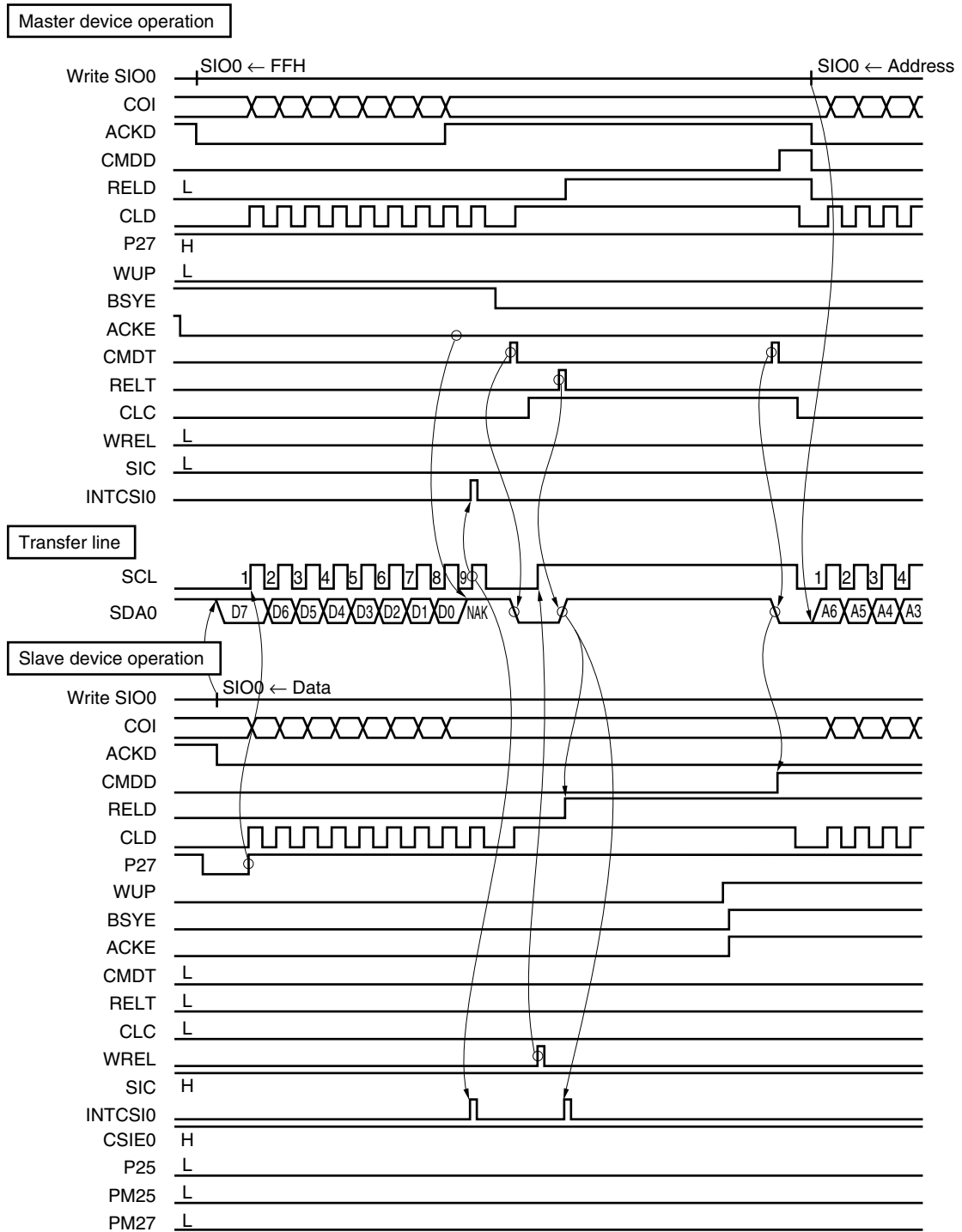


Figure 17-23. Data Transmission from Slave to Master  
(Both Master and Slave Selected 9-Clock Wait) (3/3)

(c) Stop condition



**(9) Transfer start**

A serial transfer is started by setting transfer data in serial I/O shift register 0 (SIO0) if the following two conditions have been satisfied.

- The serial interface channel 0 operation control bit (CSIE0) = 1.
- After an 8-bit serial transfer, the internal serial clock is stopped or SCL is low.

**Cautions 1. Be sure to set CSIE0 to 1 before writing data in SIO0. Setting CSIE0 to 1 after writing data in SIO0 does not initiate transfer operation.**

- 2. Because the N-ch open-drain output must be made to go into a high-impedance state during data reception, set bit 7 (BSYE) of the serial bus interface control register (SBIC) to 1 before writing FFH to SIO0.**

**Do not write FFH to SIO0 before reception when the wakeup function is used (by setting bit 5 (WUP) of serial operating mode register 0 (CSIM0)). Even if FFH is not written to SIO0, the N-ch open-drain output always goes into a high-impedance state.**

- 3. If data is written to SIO0 while the slave is in the wait state, that data is held. The transfer is started when SCL is output after the wait state is cleared.**

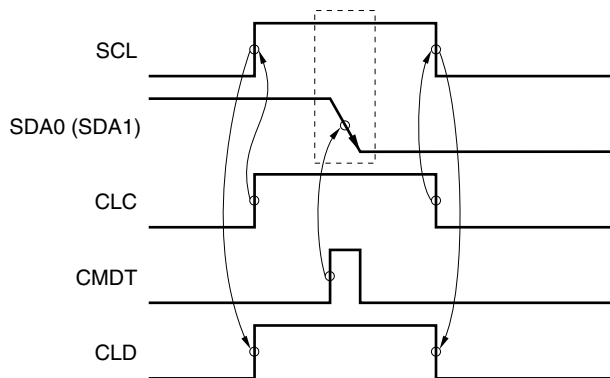
When an 8-bit data transfer ends, serial transfer is stopped automatically and the interrupt request flag (CSIF0) is set.

**17.4.5 Cautions on use of I<sup>2</sup>C bus mode****(1) Start condition output (master)**

The SCL pin normally outputs a low-level signal when no serial clock is output. It is necessary to change the SCL pin to high in order to output a start condition signal. Set CLC to 1 in the interrupt timing specification register (SINT) to drive the SCL pin high.

After setting CLC, clear CLC to 0 and return the SCL pin to low. If CLC remains 1, no serial clock is output.

If it is the master device which outputs the start condition and stop condition signals, confirm that CLD is set to 1 after setting CLC to 1; a slave device may have set SCL to low (wait state).

**Figure 17-24. Start Condition Output**

**(2) Slave wait release (slave transmission)**

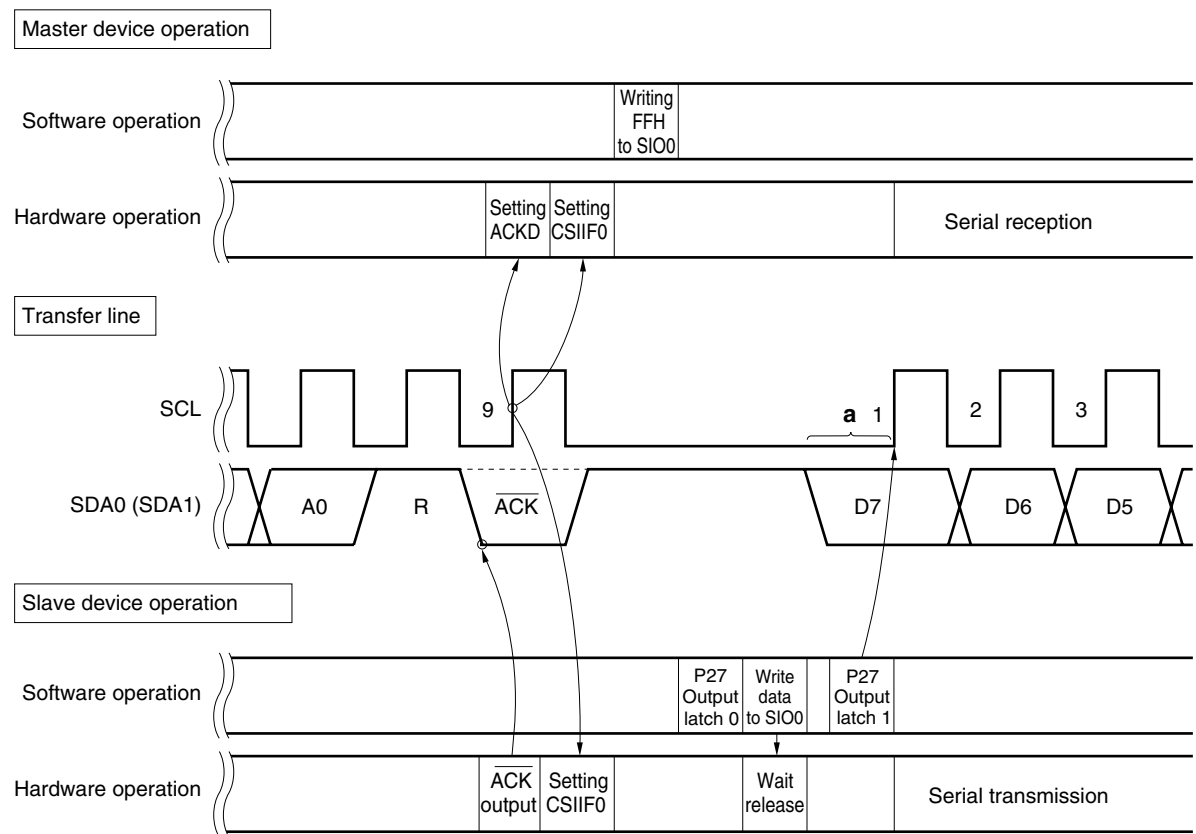
Slave wait status is released by WREL flag (bit 2 of interrupt timing specification register (SINT)) setting or execution of a serial I/O shift register 0 (SIO0) write instruction.

If the slave sends data, the wait is immediately released by execution of an SIO0 write instruction and the clock rises without the start transmission bit being output in the data line. Therefore, as shown in Figure 17-25, data should be transmitted by manipulating the P27 output latch through the program. At this time, control the low-level width (“a” in Figure 17-25) of the first serial clock at the timing used for setting the P27 output latch to 1 after execution of an SIO0 write instruction.

In addition, if the acknowledge signal from the master is not output (if data transmission from the slave is completed), set 1 in the WREL flag of SINT and release the wait.

For this timing, see Figure 17-23.

**Figure 17-25. Slave Wait Release (Transmission)**



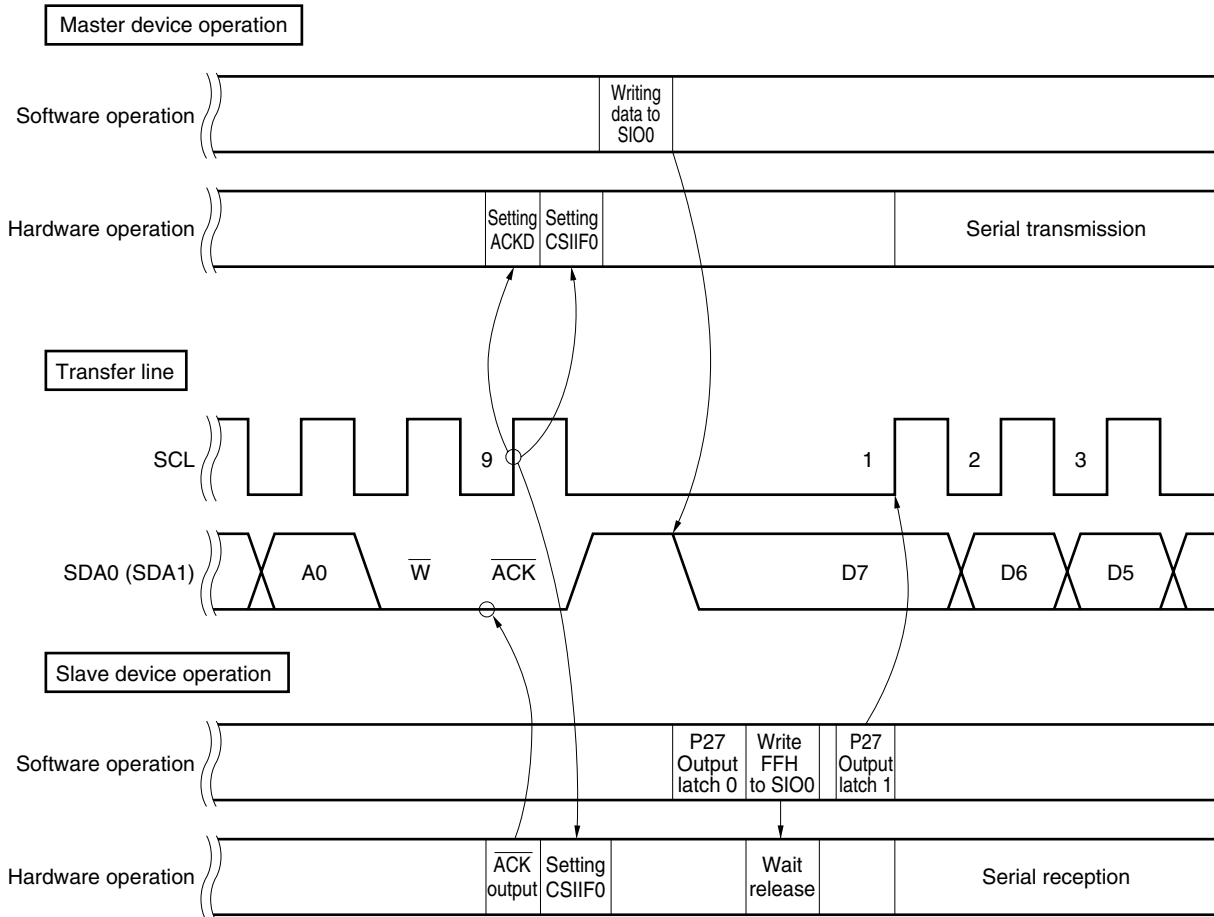
**(3) Slave wait release (slave reception)**

The slave is released from the wait status when the WREL flag (bit 2 of the interrupt timing specification register (SINT)) is set or when an instruction that writes data to serial I/O shift register 0 (SIO0) is executed. When the slave receives data, the first bit of the data sent from the master may not be received if the SCL line immediately goes into a high-impedance state after an instruction that writes data to SIO0 has been executed.

This is because SIO0 does not start operating if the SCL line is in the high-impedance state while the instruction that writes data to SIO0 is being executed (until the next instruction is executed).

Therefore, receive the data by manipulating the output latch of P27 by program, as shown in Figure 17-26. For this timing, see **Figure 17-22**.

**Figure 17-26. Slave Wait Release (Reception)**





**(4) Reception completion of slave**

In the reception completion processing of the slave, check bit 3 (CMDD) of the serial bus interface control register (SBIC) and bit 6 (COI) of serial operation mode register 0 (CSIM0) (when CMDD = 1). This is to avoid the situation where the slave cannot judge which of the start condition and data comes first and therefore the wakeup condition cannot be used when the slave receives an undefined number of data from the master.

**17.4.6 Restrictions in I<sup>2</sup>C bus mode 1**

The following restrictions are applied to the  $\mu$ PD780058Y Subseries.

- **Restrictions when used as slave device in I<sup>2</sup>C bus mode**

Target device:  $\mu$ PD780053Y, 780054Y, 780055Y, 780056Y, 780058BY, 78F0058Y, IE-780308-R-EM, IE-780308-NS-EM1

Description: If the wakeup function is executed (by setting bit 5 of serial operating mode register 0 (CSIM0) to 1) in the serial transfer status<sup>Note</sup>, the  $\mu$ PD780058Y Subseries checks the address of the data between the other slaves and the master. If that data happens to match the slave address of the  $\mu$ PD780058Y Subseries, the  $\mu$ PD780058Y Subseries takes part in communication, destroying the communication data.

**Note** The serial transfer status is the status from when data is written to serial I/O shift register 0 (SIO0) until the interrupt request flag (CSIF0) is set to 1 by completion of the serial transfer.

Preventive measure: The above phenomenon can be avoided by modifying the program. Before executing the wakeup function, execute the following program that clears the serial transfer status. When executing the wakeup function, do not execute an instruction that writes data to SIO0. Even if such an instruction is not executed, data can be received when the wakeup function is executed.

This program releases the serial transfer status. To release the serial transfer status, serial interface channel 0 must be disabled once (by clearing the CSIE0 flag (bit 7 of the serial operating mode register (CSIM0) to 0). If serial interface channel 0 is disabled in the I<sup>2</sup>C bus mode, however, the SCL pin outputs a high level, and the SDA0 (SDA1) pin outputs a low level, affecting communication of the I<sup>2</sup>C bus. Therefore, this program makes the SCL and SDA0 (SDA1) pins go into a high-impedance state to prevent the I<sup>2</sup>C bus from being affected.

In this example, the SDA0 (/P25) pin is used as a serial data input/output pin. When SDA1 (/P26) is used, take P2.5 and PM2.5 in the program example below as P2.6 and PM2.6.

For the timing of each signal when this program is executed, see **Figure 17-22**.

- Example of program releasing serial transfer status

```

SET1 P2.5; <1>
SET1 PM2.5; <2>
SET1 PM2.7; <3>
CLR1 CSIE0; <4>
SET1 CSIE0; <5>
SET1 RELT; <6>
CLR1 PM2.7; <7>
CLR1 P2.5; <8>
CLR1 PM2.5; <9>

```

- <1> This instruction prevents the SDA0 pin from outputting a low level when the I<sup>2</sup>C bus mode is restored by instruction <5>. The output of the SDA0 pin goes into a high-impedance state.
- <2> This instruction sets the P25 (/SDA0) pin in the input mode to protect the SDA0 line from adverse influence when the port mode is set by instruction <4>. The P25 pin is set in the input mode when instruction <2> is executed.
- <3> This instruction sets the P27 (/SCL) pin in the input mode to protect the SCL line from adverse influence when the port mode is set by instruction <4>. The P27 pin is set in the input mode when instruction <3> is executed.
- <4> This instruction changes the mode from I<sup>2</sup>C bus mode to port mode.
- <5> This instruction restores the I<sup>2</sup>C bus mode from the port mode.
- <6> This instruction prevents the SDA0 pin from outputting a low level when instruction <8> is executed.
- <7> This instruction sets the P27 pin in the output mode because the P27 pin must be in the output mode in the I<sup>2</sup>C bus mode.
- <8> This instruction clears the output latch of the P25 pin to 0 because the output latch of the P25 pin must be cleared to 0 in the I<sup>2</sup>C bus mode.
- <9> This instruction sets the P25 pin in the output mode because the P25 pin must be in the output mode in the I<sup>2</sup>C bus mode.

**Remark** RELT: Bit 0 of serial bus interface control register (SBIC)

★ 17.4.7 Restrictions in I<sup>2</sup>C bus mode 2

When using the I<sup>2</sup>C bus mode under the following conditions, the STOP condition is detected and an interrupt occurs when CSIE0 is set to 1. To enable the operation (by setting CSIE0 to 1), therefore, perform the following processing.

Condition: If a low level is used when CSIE0 is set to 1 when using the P26/SDA as the SDA line and P25/SDA0 as an input port

## (1) When operation is enabled

SET1	CSIMK0	; Disables <b>INTCSI0</b> interrupt.
SET1	CSIE0	; Enables IIC operation.
CLR1	CSIIF0	; Clears <b>INTCSI0</b> interrupt request flag.
CLR1	CSIMK0	; Enables <b>INTCSI0</b> interrupt.

- Cautions**
1. After that, RELD = 1 (stop condition is detected) until data that does not match the source station slave address (SVA) is received.
  2. Even if a start condition is satisfied while RELD = 1 (stop condition is detected), the interrupt occurs if it is enabled and CMDD = 1 (start condition is detected).

(2) When using as slave device in I<sup>2</sup>C bus mode (if restrictions in 17.4.6 apply)

Example of program releasing serial transfer status

SET1	CSIMK0	; Disables <b>INTCSI0</b> interrupt.
SET1	P2 . 6	
SET1	PM2 . 6	
SET1	PM2 . 7	
CLR1	CSIE0	; Stops IIC operation.
SET1	CSIE0	; Enables IIC operation.
CLR1	CSIIF0	; Clears <b>INTCSI0</b> interrupt request flag.
CLR1	CSIMK0	; Enables <b>INTCSI0</b> interrupt.
SET1	RELT	
CLR1	PM2 . 7	
CLR1	P2 . 6	
CLR1	PM2 . 6	

- Cautions**
1. After that, RELD = 1 (stop condition is detected) until data that does not match the source station slave address (SVA) is received.
  2. Even if a start condition is satisfied while RELD = 1 (stop condition is detected), the interrupt occurs if it is enabled and CMDD = 1 (start condition is detected).

### 17.4.8 $\overline{\text{SCK0/SCL/P27}}$ pin output manipulation

The  $\overline{\text{SCK0/SCL/P27}}$  pin can execute static output via software, in addition to outputting the normal serial clock.

The value of the serial clock can also be arbitrarily set by software (the  $\text{SI0/SB0/SDA0}$  and  $\text{SO0/SB1/SDA1}$  pins are controlled by bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC)).

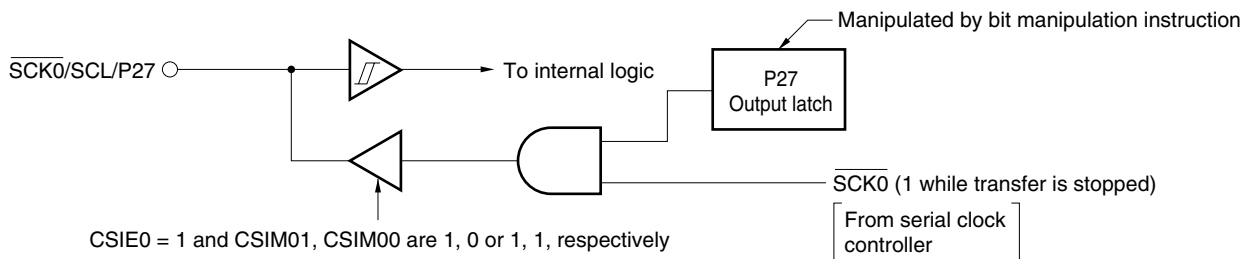
The  $\overline{\text{SCK0/SCL/P27}}$  pin output should be manipulated as described below.

#### (1) In 3-wire serial I/O mode and 2-wire serial I/O mode

The output level of the  $\overline{\text{SCK0/SCL/P27}}$  pin is manipulated by the P27 output latch.

- <1> Set serial operating mode register 0 (CSIM0) ( $\overline{\text{SCK0}}$  pin: Output mode, serial operation: Enabled).  $\overline{\text{SCK0}} = 1$  while serial transfer is stopped.
- <2> Manipulate the contents of the P27 output latch by executing a bit manipulation instruction.

Figure 17-27.  $\overline{\text{SCK0/SCL/P27}}$  Pin Configuration

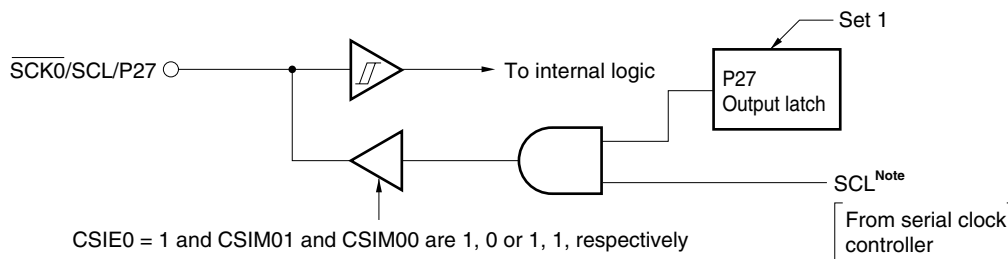


#### (2) In I<sup>2</sup>C bus mode

The output level of the  $\overline{\text{SCK0/SCL/P27}}$  pin is manipulated by the CLC bit of the interrupt timing specification register (SINT).

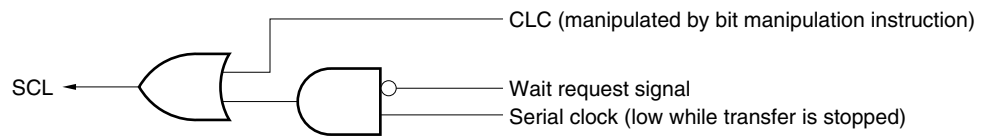
- <1> Set serial operating mode register 0 (CSIM0) (SCL pin: Output mode, serial operation: Enabled). Set the P27 output latch to 1.  $\text{SCL} = 0$  while serial transfer is stopped.
- <2> Manipulate the CLC bit of SINT by executing a bit manipulation instruction.

Figure 17-28.  $\overline{\text{SCK0/SCL/P27}}$  Pin Configuration



**Note** The level of the SCL signal is in accordance with the contents of the logic circuits shown in **Figure 17-29**.

Figure 17-29. Logic Circuit of SCL Signal



- Remarks**
1. This figure indicates the relationship of the signals and does not indicate the internal circuit.
  2. CLC: Bit 3 of interrupt timing specification register (SINT)

## CHAPTER 18 SERIAL INTERFACE CHANNEL 1

### 18.1 Functions of Serial Interface Channel 1

Serial interface channel 1 employs the following three modes.

- Operation stop mode
- 3-wire serial I/O mode
- 3-wire serial I/O mode with automatic transmit/receive function

#### (1) Operation stop mode

This mode is used when serial transfer is not carried out to reduce power consumption.

#### (2) 3-wire serial I/O mode (MSB-/LSB-first switchable)

This mode is used for 8-bit data transfer using three lines: a serial clock ( $\overline{\text{SCK1}}$ ), serial output (SO1), and serial input (SI1).

The 3-wire serial I/O mode enables simultaneous transmission/reception and so decreases the data transfer processing time.

Since the start bit of 8-bit data to undergo serial transfer is switchable between the MSB and LSB, connection is enabled with either start bit device.

The 3-wire serial I/O mode is valid for connection of peripheral I/O units and display controllers which incorporate a conventional synchronous serial interface such as the 75X/XL, 78K, and 17K Series.

#### (3) 3-wire serial I/O mode with automatic transmit/receive function (MSB-/LSB-first switchable)

This mode is equivalent to the 3-wire serial I/O mode with the addition of an automatic transmit/receive function. The automatic transmit/receive function is used to transmit/receive data with a maximum of 32 bytes. This function enables the hardware to transmit/receive data to/from the OSD (On Screen Display) device and a device with built-in display controller/driver independently of the CPU, thus alleviating the software load.

**Caution** When using the P23/STB/TxD1 and P24/BUSY/RxD1 pins in the asynchronous serial interface (UART) mode of serial interface channel 2, the busy control option and busy & strobe control option are invalid.

## 18.2 Configuration of Serial Interface Channel 1

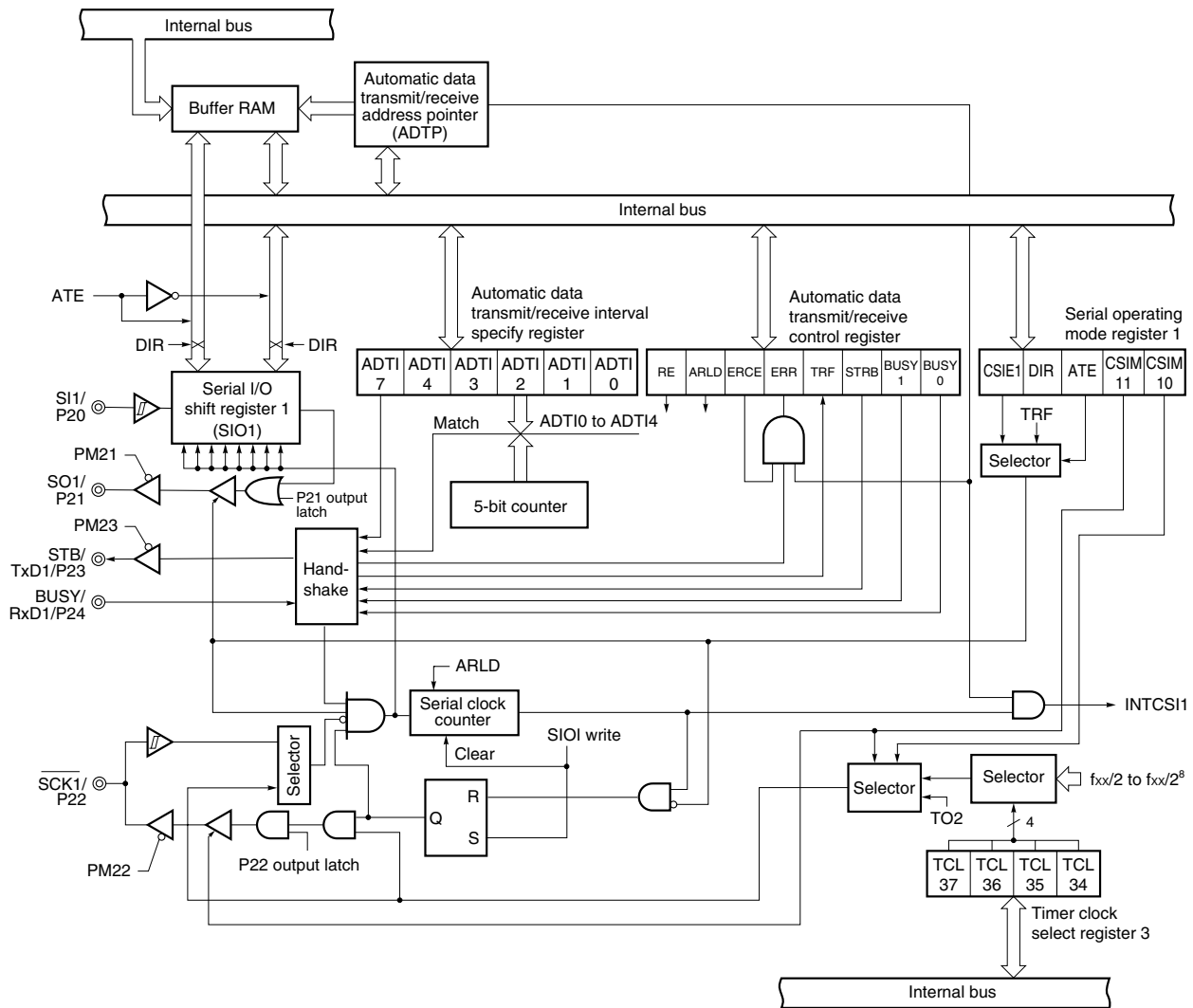
Serial interface channel 1 consists of the following hardware.

**Table 18-1. Configuration of Serial Interface Channel 1**

Item	Configuration
Registers	Serial I/O shift register 1 (SIO1) Automatic data transmit/receive address pointer (ADTP)
Control registers	Timer clock select register 3 (TCL3) Serial operating mode register 1 (CSIM1) Automatic data transmit/receive control register (ADTC) Automatic data transmit/receive interval specification register (ADTI) Port mode register 2 (PM2) <sup>Note</sup>

**Note** See **Figures 6-5 and 6-7 Block Diagram of P20, P21, and P23 to P26** and **Figures 6-6 and 6-8 Block Diagram of P22 and P27**.

Figure 18-1. Block Diagram of Serial Interface Channel 1





**(1) Serial I/O shift register 1 (SIO1)**

This is an 8-bit register used to carry out parallel/serial conversion and to carry out serial transmission/reception (shift operations) in synchronization with the serial clock.

SIO1 is set with an 8-bit memory manipulation instruction.

When the value in bit 7 (CSIE1) of serial operating mode register 1 (CSIM1) is 1, writing data to SIO1 starts serial operation.

In transmission, data written to SIO1 is output to the serial output (SO1). In reception, data is read from the serial input (SI1) to SIO1.

$\overline{\text{RESET}}$  input makes SIO1 undefined.

**Caution** Do not write data to SIO1 while the automatic transmit/receive function is activated.

**(2) Automatic data transmit/receive address pointer (ADTP)**

This register stores the value of (the number of transmit data bytes – 1) while the automatic transmit/receive function is activated. As data is transferred/received, the pointer is automatically decremented.

ADTP is set with an 8-bit memory manipulation instruction. The higher 3 bits must be cleared to 0.

$\overline{\text{RESET}}$  input clears ADTP to 00H.

**Caution** Do not write data to ADTP while the automatic transmit/receive function is activated.

**(3) Serial clock counter**

This counter counts the serial clocks to be output and input during transmission/reception to check whether 8-bit data has been transmitted/received.

### 18.3 Control Registers of Serial Interface Channel 1

The following four registers are used to control serial interface channel 1.

- Timer clock select register 3 (TCL3)
- Serial operating mode register 1 (CSIM1)
- Automatic data transmit/receive control register (ADTC)
- Automatic data transmit/receive interval specification register (ADTI)

#### (1) Timer clock select register 3 (TCL3)

This register sets the serial clock of serial interface channel 1.

TCL3 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets TCL3 to 88H.

**Remark** Besides setting the serial clock of serial interface channel 1, TCL3 sets the serial clock of serial interface channel 0.

Figure 18-2. Format of Timer Clock Select Register 3

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
TCL3	TCL37	TCL36	TCL35	TCL34	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL37	TCL36	TCL35	TCL34	Serial interface channel 1 serial clock selection		
					MCS = 1	MCS = 0
0	1	1	0	$f_{xx}/2$	Setting prohibited	$f_x/2^2$ (1.25 MHz)
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)
Other than above				Setting prohibited		

**Caution** When rewriting other data to TCL3 , stop the serial transfer operation beforehand.

- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of oscillation mode select register (OSMS)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz

(2) Serial operating mode register 1 (CSIM1)

This register sets the serial interface channel 1 serial clock, operating mode, operation enable/stop and automatic transmit/receive operation enable/stop.

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears CSIM1 to 00H.

Figure 18-3. Format of Serial Operation Mode Register 1

Symbol	<7>	6	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIM11	CSIM10	Serial interface channel 1 clock selection
0	×	External clock input to SCK1 pin <sup>Note 1</sup>
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified by bits 4 to 7 of timer clock select register 3 (TCL3)

ATE	Serial interface channel 1 operating mode selection
0	3-wire serial I/O mode
1	3-wire serial I/O mode with automatic transmit/receive function

DIR	Start bit	SI1 pin function	SO1 pin function
0	MSB	SI1/P20 (input)	SO1 (CMOS output)
1	LSB		

CSIE1	CSIM11	PM20	P20	PM21	P21	PM22	P22	Shift register 1 operation	Serial clock counter operation control	SI1/P20 pin function	SO1/P21 pin function	SCK1/P22 pin function
0	×	Note 2	Note 2	Note 2	Note 2	Note 2	Note 2	Operation stop	Clear	P20 (CMOS I/O)	P21 (CMOS I/O)	P22 (CMOS I/O)
1	0	Note 3	Note 3	0	0	1	×	Operation enable	Count operation	SI1 <sup>Note 3</sup> (input)	SO1 (CMOS output)	SCK1 (input)
	1				0	1	SCK1 (CMOS output)					

- Notes**
1. If the external clock input has been selected with CSIM11 cleared to 0, clear bit 1 (BUSY1) and bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) to 0, 0.
  2. Can be used freely as a port function.
  3. Can be used as P20 (CMOS I/O) when only transmission is performed (clear bit 7 (RE) of ADTC to 0).

**Remark**

- ×: don't care
- PM××: Port mode register
- P××: Port output latch

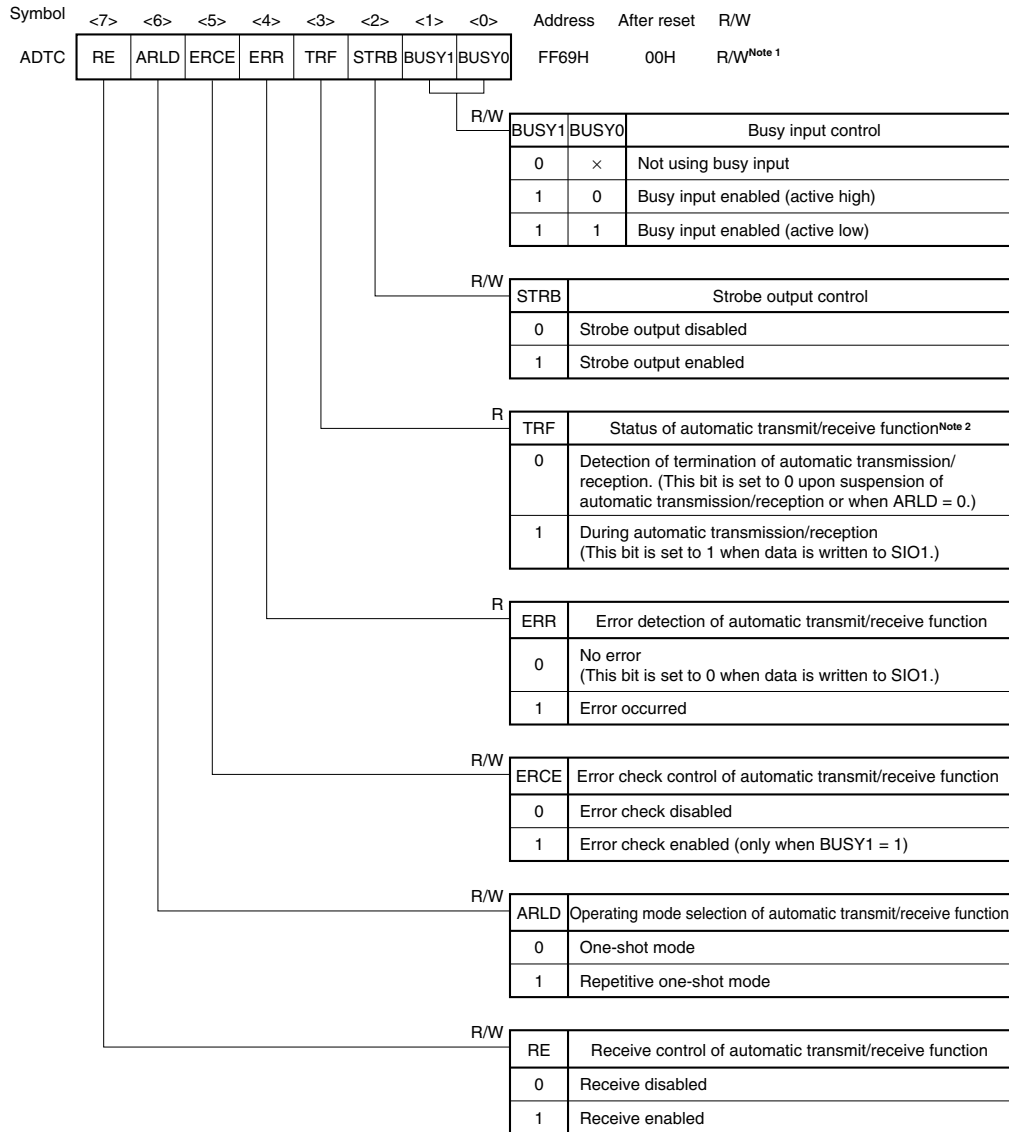
**(3) Automatic data transmit/receive control register (ADTC)**

This register sets automatic receive enable/disable, the operating mode, strobe output enable/disable, busy input enable/disable, error check enable/disable and displays automatic transmit/receive execution and error detection.

ADTC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ADTC to 00H.

**Figure 18-4. Format of Automatic Data Transmit/Receive Control Register**



**Notes** 1. Bits 3 and 4 (TRF and ERR) are read-only bits.

2. The termination of automatic transmission/reception should be judged by using TRF, not CSIF1 (interrupt request flag).

**Cautions** 1. When an external clock input is selected by clearing bit 1 (CSIM11) of serial operating mode register 1 (CSIM1) to 0, clear STRB and BUSY1 of ADTC to 0, 0.

2. When using the P23/STB/TxD1 and P24/BUSY/RxD1 pins in the asynchronous serial interface (UART) mode of serial interface channel 2, the busy control option and busy & strobe control option are invalid.

**Remark** x: don't care

**(4) Automatic data transmit/receive interval specification register (ADTI)**

This register sets the automatic data transmit/receive function data transfer interval.

ADTI is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ADTI to 00H.

**Figure 18-5. Format of Automatic Data Transmit/Receive Interval Specification Register (1/4)**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI7	Data transfer interval control
0	No control of interval by ADTI <sup>Note 1</sup>
1	Control of interval by ADTI (ADTI0 to ADTI4)

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 5.0 MHz operation)	
					Minimum <sup>Note 2</sup>	Maximum <sup>Note 2</sup>
0	0	0	0	0	18.4 μs + 0.5/f <sub>SCK</sub>	20.0 μs + 1.5/f <sub>SCK</sub>
0	0	0	0	1	31.2 μs + 0.5/f <sub>SCK</sub>	32.8 μs + 1.5/f <sub>SCK</sub>
0	0	0	1	0	44.0 μs + 0.5/f <sub>SCK</sub>	45.6 μs + 1.5/f <sub>SCK</sub>
0	0	0	1	1	56.8 μs + 0.5/f <sub>SCK</sub>	58.4 μs + 1.5/f <sub>SCK</sub>
0	0	1	0	0	69.6 μs + 0.5/f <sub>SCK</sub>	71.2 μs + 1.5/f <sub>SCK</sub>
0	0	1	0	1	82.4 μs + 0.5/f <sub>SCK</sub>	84.0 μs + 1.5/f <sub>SCK</sub>
0	0	1	1	0	95.2 μs + 0.5/f <sub>SCK</sub>	96.8 μs + 1.5/f <sub>SCK</sub>
0	0	1	1	1	108.0 μs + 0.5/f <sub>SCK</sub>	109.6 μs + 1.5/f <sub>SCK</sub>
0	1	0	0	0	120.8 μs + 0.5/f <sub>SCK</sub>	122.4 μs + 1.5/f <sub>SCK</sub>
0	1	0	0	1	133.6 μs + 0.5/f <sub>SCK</sub>	135.2 μs + 1.5/f <sub>SCK</sub>
0	1	0	1	0	146.4 μs + 0.5/f <sub>SCK</sub>	148.0 μs + 1.5/f <sub>SCK</sub>
0	1	0	1	1	159.2 μs + 0.5/f <sub>SCK</sub>	160.8 μs + 1.5/f <sub>SCK</sub>
0	1	1	0	0	172.0 μs + 0.5/f <sub>SCK</sub>	173.6 μs + 1.5/f <sub>SCK</sub>
0	1	1	0	1	184.8 μs + 0.5/f <sub>SCK</sub>	186.4 μs + 1.5/f <sub>SCK</sub>
0	1	1	1	0	197.6 μs + 0.5/f <sub>SCK</sub>	199.2 μs + 1.5/f <sub>SCK</sub>
0	1	1	1	1	210.4 μs + 0.5/f <sub>SCK</sub>	212.0 μs + 1.5/f <sub>SCK</sub>

- Notes**
- The interval is dependent only on the CPU processing.
  - The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>SCK</sub>, the minimum interval time is 2/f<sub>SCK</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{SCK}}, \text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{SCK}}$$

- Cautions**
- Do not write anything to ADTI while automatic transmission/reception is in progress (bit 3 (TRF) of the ADTC register = 1).
  - Be sure to clear bits 5 and 6 to 0.
  - When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
- f<sub>xx</sub>: Main system clock frequency (fx or fx/2)
  - fx: Main system clock oscillation frequency
  - f<sub>SCK</sub>: Serial clock frequency

Figure 18-5. Format of Automatic Data Transmit/Receive Interval Specification Register (2/4)

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 5.0 MHz operation)	
					Minimum <sup>Note</sup>	Maximum <sup>Note</sup>
1	0	0	0	0	223.2 μs + 0.5/f <sub>SCK</sub>	224.8 μs + 1.5/f <sub>SCK</sub>
1	0	0	0	1	236.0 μs + 0.5/f <sub>SCK</sub>	237.6 μs + 1.5/f <sub>SCK</sub>
1	0	0	1	0	248.8 μs + 0.5/f <sub>SCK</sub>	250.4 μs + 1.5/f <sub>SCK</sub>
1	0	0	1	1	261.6 μs + 0.5/f <sub>SCK</sub>	263.2 μs + 1.5/f <sub>SCK</sub>
1	0	1	0	0	274.4 μs + 0.5/f <sub>SCK</sub>	276.0 μs + 1.5/f <sub>SCK</sub>
1	0	1	0	1	287.2 μs + 0.5/f <sub>SCK</sub>	288.8 μs + 1.5/f <sub>SCK</sub>
1	0	1	1	0	300.0 μs + 0.5/f <sub>SCK</sub>	301.6 μs + 1.5/f <sub>SCK</sub>
1	0	1	1	1	312.8 μs + 0.5/f <sub>SCK</sub>	314.4 μs + 1.5/f <sub>SCK</sub>
1	1	0	0	0	325.6 μs + 0.5/f <sub>SCK</sub>	327.2 μs + 1.5/f <sub>SCK</sub>
1	1	0	0	1	338.4 μs + 0.5/f <sub>SCK</sub>	340.0 μs + 1.5/f <sub>SCK</sub>
1	1	0	1	0	351.2 μs + 0.5/f <sub>SCK</sub>	352.8 μs + 1.5/f <sub>SCK</sub>
1	1	0	1	1	364.0 μs + 0.5/f <sub>SCK</sub>	365.6 μs + 1.5/f <sub>SCK</sub>
1	1	1	0	0	376.8 μs + 0.5/f <sub>SCK</sub>	378.4 μs + 1.5/f <sub>SCK</sub>
1	1	1	0	1	389.6 μs + 0.5/f <sub>SCK</sub>	391.2 μs + 1.5/f <sub>SCK</sub>
1	1	1	1	0	402.4 μs + 0.5/f <sub>SCK</sub>	404.0 μs + 1.5/f <sub>SCK</sub>
1	1	1	1	1	415.2 μs + 0.5/f <sub>SCK</sub>	416.8 μs + 1.5/f <sub>SCK</sub>

**Note** The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>SCK</sub>, the minimum interval time is 2/f<sub>SCK</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{SCK}}$$

$$\text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{SCK}}$$

- Cautions**
1. Do not write to ADTI during operation of the automatic data transmit/receive function.
  2. Be sure to clear bits 5 and 6 to 0.
  3. When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
1. f<sub>xx</sub>: Main system clock frequency (f<sub>x</sub> or f<sub>x</sub>/2)
  2. f<sub>x</sub>: Main system clock oscillation frequency
  3. f<sub>SCK</sub>: Serial clock frequency

**Figure 18-5. Format of Automatic Data Transmit/Receive Interval Specification Register (3/4)**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI7	Data transfer interval control
0	No control of interval by ADTI <sup>Note 1</sup>
1	Control of interval by ADTI (ADTI0 to ADTI4)

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 2.5 MHz operation)	
					Minimum <sup>Note 2</sup>	Maximum <sup>Note 2</sup>
0	0	0	0	0	36.8 μs + 0.5/f <sub>sck</sub>	40.0 μs + 1.5/f <sub>sck</sub>
0	0	0	0	1	62.4 μs + 0.5/f <sub>sck</sub>	65.6 μs + 1.5/f <sub>sck</sub>
0	0	0	1	0	88.0 μs + 0.5/f <sub>sck</sub>	91.2 μs + 1.5/f <sub>sck</sub>
0	0	0	1	1	113.6 μs + 0.5/f <sub>sck</sub>	116.8 μs + 1.5/f <sub>sck</sub>
0	0	1	0	0	139.2 μs + 0.5/f <sub>sck</sub>	142.4 μs + 1.5/f <sub>sck</sub>
0	0	1	0	1	164.8 μs + 0.5/f <sub>sck</sub>	168.0 μs + 1.5/f <sub>sck</sub>
0	0	1	1	0	190.4 μs + 0.5/f <sub>sck</sub>	193.6 μs + 1.5/f <sub>sck</sub>
0	0	1	1	1	216.0 μs + 0.5/f <sub>sck</sub>	219.2 μs + 1.5/f <sub>sck</sub>
0	1	0	0	0	241.6 μs + 0.5/f <sub>sck</sub>	244.8 μs + 1.5/f <sub>sck</sub>
0	1	0	0	1	267.2 μs + 0.5/f <sub>sck</sub>	270.4 μs + 1.5/f <sub>sck</sub>
0	1	0	1	0	292.8 μs + 0.5/f <sub>sck</sub>	296.0 μs + 1.5/f <sub>sck</sub>
0	1	0	1	1	318.4 μs + 0.5/f <sub>sck</sub>	321.6 μs + 1.5/f <sub>sck</sub>
0	1	1	0	0	344.0 μs + 0.5/f <sub>sck</sub>	347.2 μs + 1.5/f <sub>sck</sub>
0	1	1	0	1	369.6 μs + 0.5/f <sub>sck</sub>	372.8 μs + 1.5/f <sub>sck</sub>
0	1	1	1	0	395.2 μs + 0.5/f <sub>sck</sub>	398.4 μs + 1.5/f <sub>sck</sub>
0	1	1	1	1	420.8 μs + 0.5/f <sub>sck</sub>	424.0 μs + 1.5/f <sub>sck</sub>

- Notes**
- The interval is dependent only on the CPU processing.
  - The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>sck</sub>, the minimum interval time is 2/f<sub>sck</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{sck}}$$

$$\text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{sck}}$$

- Cautions**
- Do not write to ADTI during operation of the automatic data transmit/receive function.
  - Be sure to clear bits 5 and 6 to 0.
  - When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
- f<sub>xx</sub>: Main system clock frequency (f<sub>x</sub> or f<sub>x</sub>/2)
  - f<sub>x</sub>: Main system clock oscillation frequency
  - f<sub>sck</sub>: Serial clock frequency



Figure 18-5. Format of Automatic Data Transmit/Receive Interval Specification Register (4/4)

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 2.5 MHz operation)	
					Minimum <sup>Note</sup>	Maximum <sup>Note</sup>
1	0	0	0	0	446.4 μs + 0.5/f <sub>sck</sub>	449.6 μs + 1.5/f <sub>sck</sub>
1	0	0	0	1	472.0 μs + 0.5/f <sub>sck</sub>	475.2 μs + 1.5/f <sub>sck</sub>
1	0	0	1	0	497.6 μs + 0.5/f <sub>sck</sub>	500.8 μs + 1.5/f <sub>sck</sub>
1	0	0	1	1	523.2 μs + 0.5/f <sub>sck</sub>	526.4 μs + 1.5/f <sub>sck</sub>
1	0	1	0	0	548.8 μs + 0.5/f <sub>sck</sub>	552.0 μs + 1.5/f <sub>sck</sub>
1	0	1	0	1	574.4 μs + 0.5/f <sub>sck</sub>	577.6 μs + 1.5/f <sub>sck</sub>
1	0	1	1	0	600.0 μs + 0.5/f <sub>sck</sub>	603.2 μs + 1.5/f <sub>sck</sub>
1	0	1	1	1	625.6 μs + 0.5/f <sub>sck</sub>	628.8 μs + 1.5/f <sub>sck</sub>
1	1	0	0	0	651.2 μs + 0.5/f <sub>sck</sub>	654.4 μs + 1.5/f <sub>sck</sub>
1	1	0	0	1	676.8 μs + 0.5/f <sub>sck</sub>	680.0 μs + 1.5/f <sub>sck</sub>
1	1	0	1	0	702.4 μs + 0.5/f <sub>sck</sub>	705.6 μs + 1.5/f <sub>sck</sub>
1	1	0	1	1	728.0 μs + 0.5/f <sub>sck</sub>	731.2 μs + 1.5/f <sub>sck</sub>
1	1	1	0	0	753.6 μs + 0.5/f <sub>sck</sub>	756.8 μs + 1.5/f <sub>sck</sub>
1	1	1	0	1	779.2 μs + 0.5/f <sub>sck</sub>	782.4 μs + 1.5/f <sub>sck</sub>
1	1	1	1	0	804.8 μs + 0.5/f <sub>sck</sub>	808.0 μs + 1.5/f <sub>sck</sub>
1	1	1	1	1	830.4 μs + 0.5/f <sub>sck</sub>	833.6 μs + 1.5/f <sub>sck</sub>

**Note** The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>sck</sub>, the minimum interval time is 2/f<sub>sck</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{sck}}$$

$$\text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{sck}}$$

- Cautions**
1. Do not write to ADTI during operation of the automatic data transmit/receive function.
  2. Be sure to clear bits 5 and 6 to 0.
  3. When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
1. f<sub>xx</sub>: Main system clock frequency (fx or fx/2)
  2. fx: Main system clock oscillation frequency
  3. f<sub>sck</sub>: Serial clock frequency

### 18.4 Operations of Serial Interface Channel 1

The following three operating modes are available for serial interface channel 1.

- Operation stop mode
- 3-wire serial I/O mode
- 3-wire serial I/O mode with automatic transmit/receive function

#### 18.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power consumption can be reduced. Serial I/O shift register 1 (SIO1) does not carry out shift operations either, and thus it can be used as an ordinary 8-bit register.

In the operation stop mode, the P20/SI1, P21/SO1, P22/ $\overline{\text{SCK1}}$ , P23/STB/TxD1, and P24/BUSY/RxD1 pins can be used as ordinary I/O ports.

##### (1) Register setting

The operation stop mode is set by serial operating mode register 1 (CSIM1).

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM1 to 00H.

Symbol	<7>	6	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIE1	CSIM11	PM20	P20	PM21	P21	PM22	P22	Shift register 1 operation	Serial clock counter operation control	S11/P20 pin function	SO1/P21 pin function	$\overline{\text{SCK1}}$ /P22 pin function
0	×	Note 1	Note 1	Note 1	Note 1	Note 1	Note 1	Operation stop	Clear	P20 (CMOS I/O)	P21 (CMOS I/O)	P22 (CMOS I/O)
1	0	Note 2	Note 2	0	0	1	×	Operation enable	Count operation	S11 Note 2 (input)	SO1 (CMOS output)	$\overline{\text{SCK1}}$ (input)
	1					0	1					$\overline{\text{SCK1}}$ (CMOS output)

**Notes** 1. Can be used freely as a port function.

2. Can be used as P20 (CMOS I/O) when only transmission is performed (clear bit 7 (RE) of the automatic data transmit/receive control register (ADTC) to 0).

**Remark** ×: don't care  
 PMxx: Port mode register  
 Pxx: Port output latch

**18.4.2 3-wire serial I/O mode operation**

The 3-wire serial I/O mode is useful for connection of peripheral I/O units and display controllers which incorporate a conventional clocked serial interface such as the 75X/XL, 78K and 17K Series.

Communication is carried out using the three lines of the serial clock ( $\overline{\text{SCK1}}$ ), serial output (SO1) and serial input (SI1).

**(1) Register setting**

The 3-wire serial I/O mode is set by serial operating mode register 1 (CSIM1).

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM1 to 00H.

Symbol	<7>	6	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIM11	CSIM10	Serial interface channel 1 clock selection
0	×	External clock input to $\overline{\text{SCK1}}$ pin <sup>Note</sup>
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified by bits 4 to 7 of timer clock select register 3 (TCL3)

ATE	Serial interface channel 1 operating mode selection
0	3-wire serial I/O mode
1	3-wire serial I/O mode with automatic transmit/receive function

DIR	Start bit	SO1 pin function	SO1 pin function
0	MSB	SI1/P20 (Input)	SO1 (CMOS output)
1	LSB		

**Note** If the external clock input has been selected by setting CSIM11 to 0, set bit 1 (BUSY1) and bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) to 0, 0.

**Remark** ×: don't care

CSIE1	CSIM11	PM20	P20	PM21	P21	PM22	P22	Shift register 1 operation	Serial clock counter operation control	SI1/P20 pin function	SO1/P21 pin function	$\overline{\text{SCK1}}$ /P22 pin function
0	x	Note 1 x	Note 1 x	Note 1 x	Note 1 x	Note 1 x	Note 1 x	Operation stop	Clear	P20 (CMOS I/O)	P21 (CMOS I/O)	P22 (CMOS I/O)
1	0	Note 2 1	Note 2 x	0	0	1	x	Operation enable	Count operation	SI1 Note 2 (input)	SO1 (CMOS output)	$\overline{\text{SCK1}}$ (input)
	1				0	1	$\overline{\text{SCK1}}$ (CMOS output)					

- Notes**
1. Can be used freely as a port function.
  2. Can be used as P20 (CMOS input/output) when only transmission is performed (clear bit 7 (RE) of ADTC to 0).

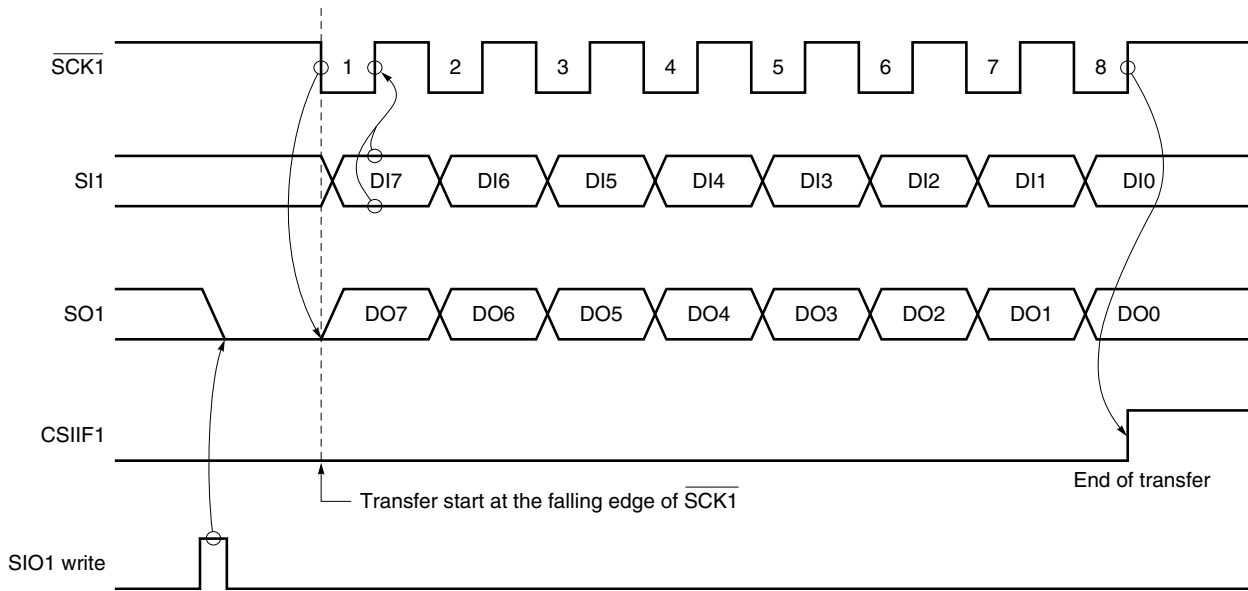
**Remark**

- x: don't care
- PMxx: Port mode register
- Pxx: Port output latch

**(2) Communication operation**

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock. Shift operations of serial I/O shift register 1 (SIO1) are carried out at the falling edge of the serial clock  $\overline{\text{SCK1}}$ . The transmit data is held in the SO1 latch and is output from the SO1 pin. The receive data input to the SI1 pin is latched into SIO1 at the rising edge of  $\overline{\text{SCK1}}$ . Upon termination of 8-bit transfer, the SIO1 operation stops automatically and the interrupt request flag (CSIF1) is set.

**Figure 18-6. 3-Wire Serial I/O Mode Timing**



**Caution** The SO1 pin becomes low level by writing SIO1.

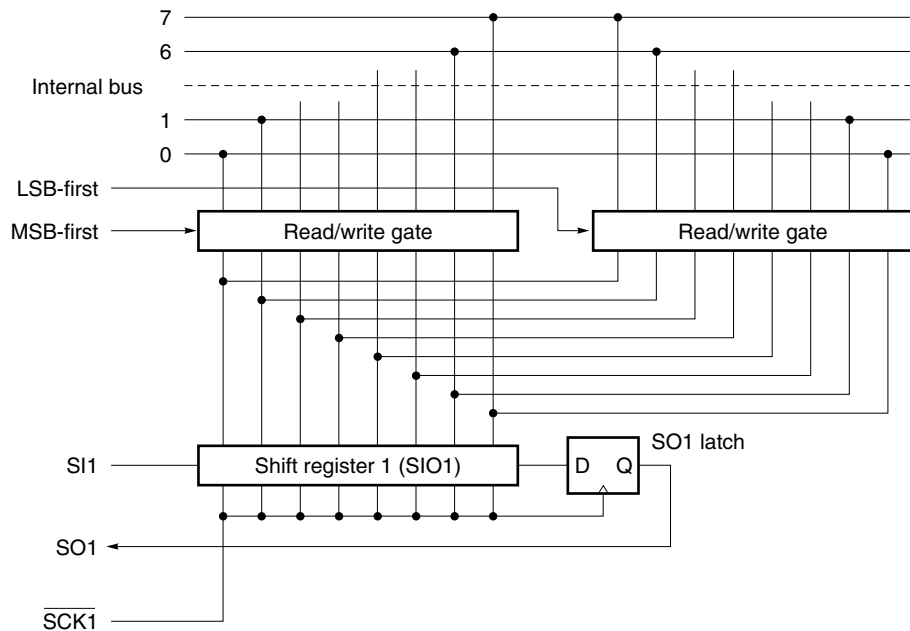
**(3) MSB/LSB switching as the start bit**

In 3-wire serial I/O mode, it is possible to select transfer to start from the MSB or LSB.

Figure 18-7 shows the configuration of serial I/O shift register 1 (SIO1) and the internal bus. As shown in the figure, the MSB/LSB can be read or written in reverse form.

MSB/LSB switching as the start bit can be specified by bit 6 (DIR) of serial operating mode register 1 (CSIM1).

**Figure 18-7. Circuit for Switching Transfer Bit Order**



Start bit switching is realized by switching the bit order for data write to SIO1. The SIO1 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to the shift register.

**(4) Transfer start**

Serial transfer is started by setting transfer data to serial I/O shift register 1 (SIO1) when the following two conditions are satisfied.

- Serial interface channel 1 operation control bit (CSIE1) = 1
- Internal serial clock is stopped or  $\overline{\text{SCK1}}$  is a high level after 8-bit serial transfer.

**Caution** If CSIE1 is set to 1 after data write to SIO1, transfer does not start.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIF1) is set.

**18.4.3 3-wire serial I/O mode operation with automatic transmit/receive function**

This 3-wire serial I/O mode is used for transmission/reception of a maximum of 32 bytes of data without the use of software. Once transfer is started, the set number of bytes of the data prestored in the RAM can be transmitted, and the set number of bytes of data can be received and stored in the RAM.

Handshake signals (STB and BUSY) are supported by hardware to transmit/receive data continuously. An OSD (On Screen Display) LSI and peripheral LSI including an LCD controller/driver can thus be connected without difficulty.

**(1) Register setting**

The 3-wire serial I/O mode with automatic transmit/receive function is set by serial operating mode register 1 (CSIM1), the automatic data transmit/receive control register (ADTC) and the automatic data transmit/receive interval specification register (ADTI).

**(a) Serial operating mode register 1 (CSIM1)**

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM1 to 00H.

Symbol	<7>	6	<5>	4	3	2	1	0	Address	After reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIM11	CSIM10	Serial interface channel 1 clock selection
0	×	External clock input to $\overline{\text{SCK1}}$ pin <sup>Note 1</sup>
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified by bits 4 to 7 of timer clock select register 3 (TCL3)

ATE	Serial interface channel 1 operating mode selection
0	3-wire serial I/O mode
1	3-wire serial I/O mode with automatic transmit/receive function

DIR	Start bit	SI1 pin function	SO1 pin function
0	MSB	SI1/P20 (input)	SO1 (CMOS output)
1	LSB		

CSIE1	CSIM11	PM20	P20	PM21	P21	PM22	P22	Shift register 1 operation	Serial clock counter operation control	SI1/P20 pin function	SO1/P21 pin function	$\overline{\text{SCK1}}$ /P22 pin function
0	×	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	<sup>Note 2</sup> ×	Operation stop	Clear	P20 (CMOS I/O)	P21 (CMOS I/O)	P22 (CMOS I/O)
1	0	<sup>Note 3</sup> 1	<sup>Note 3</sup> ×	0	0	1	×	Operation enable	Count operation	SI1 <sup>Note 3</sup> (input)	SO1 (CMOS output)	$\overline{\text{SCK1}}$ (input)
	1				0	1	$\overline{\text{SCK1}}$ (CMOS output)					

- Notes**
1. If the external clock input has been selected by clearing CSIM11 to 0, clear bit 1 (BUSY 1) and bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) to 0, 0.
  2. Can be used freely as a port function.
  3. Can be used as P20 (CMOS input/output) when only transmission is performed (clear bit 7 (RE) of ADTC to 0).

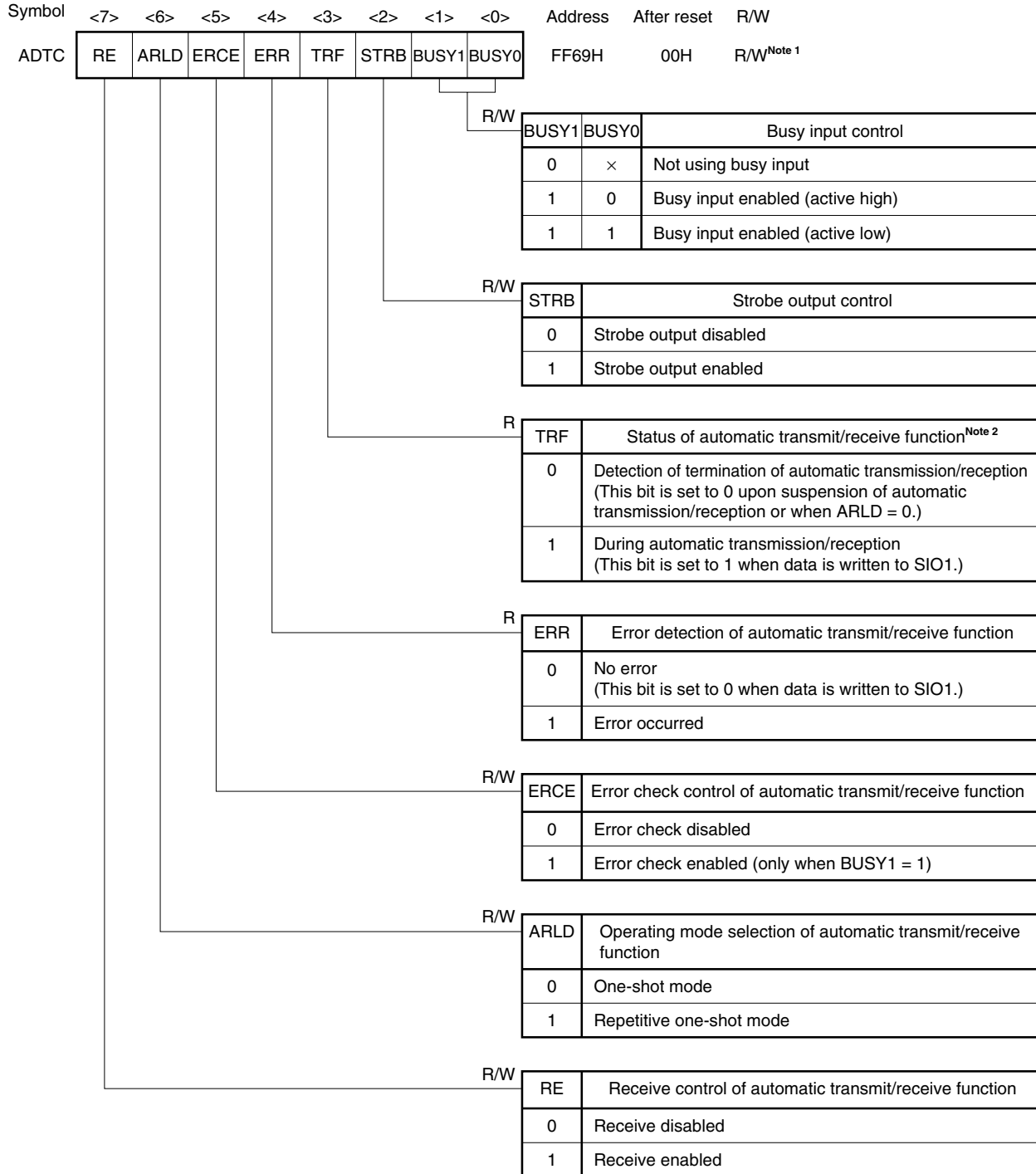
**Remark**

- ×: don't care
- PMxx: Port mode register
- Pxx: Port output latch

**(b) Automatic data transmit/receive control register (ADTC)**

ADTC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ADTC to 00H.



- Notes**
- Bits 3 and 4 (TRF and ERR) are read-only bits.
  - The termination of automatic transmission/reception should be judged by using TRF, not CSIF1 (interrupt request flag).

**Caution** When an external clock input is selected by clearing bit 1 (CSIM11) of serial operating mode register 1 (CSIM1) to 0, clear STRB and BUSY1 of ADTC to 0, 0 (handshake control cannot be executed when an external clock is input).

**Remark** ×: don't care



**(c) Automatic data transmit/receive interval specification register (ADTI)**

This register sets the automatic data transmit/receive function data transfer interval.

ADTI is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears ADTI to 00H.

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI7	Data transfer interval control
0	No control of interval by ADTI <sup>Note 1</sup>
1	Control of interval by ADTI (ADTI0 to ADTI4)

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 5.0 MHz operation)	
					Minimum <sup>Note 2</sup>	Maximum <sup>Note 2</sup>
0	0	0	0	0	18.4 μs + 0.5/f <sub>SCK</sub>	20.0 μs + 1.5/f <sub>SCK</sub>
0	0	0	0	1	31.2 μs + 0.5/f <sub>SCK</sub>	32.8 μs + 1.5/f <sub>SCK</sub>
0	0	0	1	0	44.0 μs + 0.5/f <sub>SCK</sub>	45.6 μs + 1.5/f <sub>SCK</sub>
0	0	0	1	1	56.8 μs + 0.5/f <sub>SCK</sub>	58.4 μs + 1.5/f <sub>SCK</sub>
0	0	1	0	0	69.6 μs + 0.5/f <sub>SCK</sub>	71.2 μs + 1.5/f <sub>SCK</sub>
0	0	1	0	1	82.4 μs + 0.5/f <sub>SCK</sub>	84.0 μs + 1.5/f <sub>SCK</sub>
0	0	1	1	0	95.2 μs + 0.5/f <sub>SCK</sub>	96.8 μs + 1.5/f <sub>SCK</sub>
0	0	1	1	1	108.0 μs + 0.5/f <sub>SCK</sub>	109.6 μs + 1.5/f <sub>SCK</sub>
0	1	0	0	0	120.8 μs + 0.5/f <sub>SCK</sub>	122.4 μs + 1.5/f <sub>SCK</sub>
0	1	0	0	1	133.6 μs + 0.5/f <sub>SCK</sub>	135.2 μs + 1.5/f <sub>SCK</sub>
0	1	0	1	0	146.4 μs + 0.5/f <sub>SCK</sub>	148.0 μs + 1.5/f <sub>SCK</sub>
0	1	0	1	1	159.2 μs + 0.5/f <sub>SCK</sub>	160.8 μs + 1.5/f <sub>SCK</sub>
0	1	1	0	0	172.0 μs + 0.5/f <sub>SCK</sub>	173.6 μs + 1.5/f <sub>SCK</sub>
0	1	1	0	1	184.8 μs + 0.5/f <sub>SCK</sub>	186.4 μs + 1.5/f <sub>SCK</sub>
0	1	1	1	0	197.6 μs + 0.5/f <sub>SCK</sub>	199.2 μs + 1.5/f <sub>SCK</sub>
0	1	1	1	1	210.4 μs + 0.5/f <sub>SCK</sub>	212.0 μs + 1.5/f <sub>SCK</sub>

**Notes** 1. The interval is dependent only on the CPU processing.

2. The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>SCK</sub>, the minimum interval time is 2/f<sub>SCK</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{SCK}}, \quad \text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{SCK}}$$

**Cautions** 1. Do not write to ADTI during operation of the automatic data transmit/receive function.

2. Be sure to clear bits 5 and 6 to 0.

3. When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

**Remarks** 1. f<sub>xx</sub>: Main system clock frequency (fx or fx/2)

2. fx: Main system clock oscillation frequency

3. f<sub>SCK</sub>: Serial clock frequency

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 5.0 MHz operation)	
					Minimum <sup>Note</sup>	Maximum <sup>Note</sup>
1	0	0	0	0	223.2 μs + 0.5/f <sub>sck</sub>	224.8 μs + 1.5/f <sub>sck</sub>
1	0	0	0	1	236.0 μs + 0.5/f <sub>sck</sub>	237.6 μs + 1.5/f <sub>sck</sub>
1	0	0	1	0	248.8 μs + 0.5/f <sub>sck</sub>	250.4 μs + 1.5/f <sub>sck</sub>
1	0	0	1	1	261.6 μs + 0.5/f <sub>sck</sub>	263.2 μs + 1.5/f <sub>sck</sub>
1	0	1	0	0	274.4 μs + 0.5/f <sub>sck</sub>	276.0 μs + 1.5/f <sub>sck</sub>
1	0	1	0	1	287.2 μs + 0.5/f <sub>sck</sub>	288.8 μs + 1.5/f <sub>sck</sub>
1	0	1	1	0	300.0 μs + 0.5/f <sub>sck</sub>	301.6 μs + 1.5/f <sub>sck</sub>
1	0	1	1	1	312.8 μs + 0.5/f <sub>sck</sub>	314.4 μs + 1.5/f <sub>sck</sub>
1	1	0	0	0	325.6 μs + 0.5/f <sub>sck</sub>	327.2 μs + 1.5/f <sub>sck</sub>
1	1	0	0	1	338.4 μs + 0.5/f <sub>sck</sub>	340.0 μs + 1.5/f <sub>sck</sub>
1	1	0	1	0	351.2 μs + 0.5/f <sub>sck</sub>	352.8 μs + 1.5/f <sub>sck</sub>
1	1	0	1	1	364.0 μs + 0.5/f <sub>sck</sub>	365.6 μs + 1.5/f <sub>sck</sub>
1	1	1	0	0	376.8 μs + 0.5/f <sub>sck</sub>	378.4 μs + 1.5/f <sub>sck</sub>
1	1	1	0	1	389.6 μs + 0.5/f <sub>sck</sub>	391.2 μs + 1.5/f <sub>sck</sub>
1	1	1	1	0	402.4 μs + 0.5/f <sub>sck</sub>	404.0 μs + 1.5/f <sub>sck</sub>
1	1	1	1	1	415.2 μs + 0.5/f <sub>sck</sub>	416.8 μs + 1.5/f <sub>sck</sub>

**Note** The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>sck</sub>, the minimum interval time is 2/f<sub>sck</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{sck}}$$

$$\text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{sck}}$$

- Cautions**
1. Do not write to ADTI during operation of the automatic data transmit/receive function.
  2. Be sure to clear bits 5 and 6 to 0.
  3. When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
1. f<sub>xx</sub>: Main system clock frequency (fx or fx/2)
  2. fx: Main system clock oscillation frequency
  3. f<sub>sck</sub>: Serial clock frequency

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI7	Data transfer interval control
0	No control of interval by ADTI   <sup>Note 1</sup>
1	Control of interval by ADTI (ADTI0 to ADTI4)

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 2.5 MHz operation)	
					Minimum <sup>Note 2</sup>	Maximum <sup>Note 2</sup>
0	0	0	0	0	36.8 μs + 0.5/f <sub>sck</sub>	40.0 μs + 1.5/f <sub>sck</sub>
0	0	0	0	1	62.4 μs + 0.5/f <sub>sck</sub>	65.6 μs + 1.5/f <sub>sck</sub>
0	0	0	1	0	88.0 μs + 0.5/f <sub>sck</sub>	91.2 μs + 1.5/f <sub>sck</sub>
0	0	0	1	1	113.6 μs + 0.5/f <sub>sck</sub>	116.8 μs + 1.5/f <sub>sck</sub>
0	0	1	0	0	139.2 μs + 0.5/f <sub>sck</sub>	142.4 μs + 1.5/f <sub>sck</sub>
0	0	1	0	1	164.8 μs + 0.5/f <sub>sck</sub>	168.0 μs + 1.5/f <sub>sck</sub>
0	0	1	1	0	190.4 μs + 0.5/f <sub>sck</sub>	193.6 μs + 1.5/f <sub>sck</sub>
0	0	1	1	1	216.0 μs + 0.5/f <sub>sck</sub>	219.2 μs + 1.5/f <sub>sck</sub>
0	1	0	0	0	241.6 μs + 0.5/f <sub>sck</sub>	244.8 μs + 1.5/f <sub>sck</sub>
0	1	0	0	1	267.2 μs + 0.5/f <sub>sck</sub>	270.4 μs + 1.5/f <sub>sck</sub>
0	1	0	1	0	292.8 μs + 0.5/f <sub>sck</sub>	296.0 μs + 1.5/f <sub>sck</sub>
0	1	0	1	1	318.4 μs + 0.5/f <sub>sck</sub>	321.6 μs + 1.5/f <sub>sck</sub>
0	1	1	0	0	344.0 μs + 0.5/f <sub>sck</sub>	347.2 μs + 1.5/f <sub>sck</sub>
0	1	1	0	1	369.6 μs + 0.5/f <sub>sck</sub>	372.8 μs + 1.5/f <sub>sck</sub>
0	1	1	1	0	395.2 μs + 0.5/f <sub>sck</sub>	398.4 μs + 1.5/f <sub>sck</sub>
0	1	1	1	1	420.8 μs + 0.5/f <sub>sck</sub>	424.0 μs + 1.5/f <sub>sck</sub>

- Notes**
- The interval is dependent only on the CPU processing.
  - The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>sck</sub>, the minimum interval time is 2/f<sub>sck</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{sck}}$$

$$\text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{sck}}$$

- Cautions**
- Do not write to ADTI during operation of the automatic data transmit/receive function.
  - Be sure to clear bits 5 and 6 to 0.
  - When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
- f<sub>xx</sub>: Main system clock frequency (fx or fx/2)
  - f<sub>x</sub>: Main system clock oscillation frequency
  - f<sub>sck</sub>: Serial clock frequency

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
ADTI	ADTI7	0	0	ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	FF6BH	00H	R/W

ADTI4	ADTI3	ADTI2	ADTI1	ADTI0	Data transfer interval specification (f <sub>xx</sub> = 2.5 MHz operation)	
					Minimum <sup>Note</sup>	Maximum <sup>Note</sup>
1	0	0	0	0	446.4 μs + 0.5/f <sub>sck</sub>	449.6 μs + 1.5/f <sub>sck</sub>
1	0	0	0	1	472.0 μs + 0.5/f <sub>sck</sub>	475.2 μs + 1.5/f <sub>sck</sub>
1	0	0	1	0	497.6 μs + 0.5/f <sub>sck</sub>	500.8 μs + 1.5/f <sub>sck</sub>
1	0	0	1	1	523.2 μs + 0.5/f <sub>sck</sub>	526.4 μs + 1.5/f <sub>sck</sub>
1	0	1	0	0	548.8 μs + 0.5/f <sub>sck</sub>	552.0 μs + 1.5/f <sub>sck</sub>
1	0	1	0	1	574.4 μs + 0.5/f <sub>sck</sub>	577.6 μs + 1.5/f <sub>sck</sub>
1	0	1	1	0	600.0 μs + 0.5/f <sub>sck</sub>	603.2 μs + 1.5/f <sub>sck</sub>
1	0	1	1	1	625.6 μs + 0.5/f <sub>sck</sub>	628.8 μs + 1.5/f <sub>sck</sub>
1	1	0	0	0	651.2 μs + 0.5/f <sub>sck</sub>	654.4 μs + 1.5/f <sub>sck</sub>
1	1	0	0	1	676.8 μs + 0.5/f <sub>sck</sub>	680.0 μs + 1.5/f <sub>sck</sub>
1	1	0	1	0	702.4 μs + 0.5/f <sub>sck</sub>	705.6 μs + 1.5/f <sub>sck</sub>
1	1	0	1	1	728.0 μs + 0.5/f <sub>sck</sub>	731.2 μs + 1.5/f <sub>sck</sub>
1	1	1	0	0	753.6 μs + 0.5/f <sub>sck</sub>	756.8 μs + 1.5/f <sub>sck</sub>
1	1	1	0	1	779.2 μs + 0.5/f <sub>sck</sub>	782.4 μs + 1.5/f <sub>sck</sub>
1	1	1	1	0	804.8 μs + 0.5/f <sub>sck</sub>	808.0 μs + 1.5/f <sub>sck</sub>
1	1	1	1	1	830.4 μs + 0.5/f <sub>sck</sub>	833.6 μs + 1.5/f <sub>sck</sub>

**Note** The data transfer interval includes an error. The data transfer minimum and maximum intervals are found from the following expressions (n: Value set in ADTI0 to ADTI4). However, if a minimum which is calculated by the following expression is smaller than 2/f<sub>sck</sub>, the minimum interval time is 2/f<sub>sck</sub>.

$$\text{Minimum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{28}{f_{xx}} + \frac{0.5}{f_{sck}}$$

$$\text{Maximum} = (n + 1) \times \frac{2^6}{f_{xx}} + \frac{36}{f_{xx}} + \frac{1.5}{f_{sck}}$$

- Cautions**
1. Do not write to ADTI during operation of the automatic data transmit/receive function.
  2. Be sure to clear bits 5 and 6 to 0.
  3. When controlling the data transfer interval by means of automatic transmission/reception using ADTI, busy control (see 18.4.3 (4) (a) Busy control option) is invalid.

- Remarks**
1. f<sub>xx</sub>: Main system clock frequency (fx or fx/2)
  2. fx: Main system clock oscillation frequency
  3. f<sub>sck</sub>: Serial clock frequency

**(2) Automatic transmit/receive data setting****(a) Transmit data setting**

- <1> Write transmit data from the least significant address FAC0H of buffer RAM (up to FADFH at maximum). The transmit data should be in order from higher address to lower address.
- <2> Set the value obtained by subtracting 1 from the number of transmit data bytes to the automatic data transmit/receive address pointer (ADTP).

**(b) Automatic transmit/receive mode setting**

- <1> Set CSIE1 and ATE of serial operating mode register 1 (CSIM1) to 1.
- <2> Set RE of the automatic data transmit/receive control register (ADTC) to 1.
- <3> Set a data transmit/receive interval in the automatic data transmit/receive interval specification register (ADTI).
- <4> Write any value to serial I/O shift register 1 (SIO1) (transfer start trigger).

**Caution** Writing any value to SIO1 orders the start of the automatic transmit/receive operation; the written value has no meaning.

The following operations are automatically carried out when (a) and (b) are carried out.

- After the buffer RAM data specified by ADTP is transferred to SIO1, transmission is carried out (start of automatic transmission/reception).
- The received data is written to the buffer RAM address specified by ADTP.
- ADTP is decremented and the next data transmission/reception is carried out. Data transmission/reception continues until the ADTP decremental output becomes 00H and the data at address FAC0H is output (end of automatic transmission/reception).
- When automatic transmission/reception is terminated, TRF is cleared to 0.

**(3) Communication operation****(a) Basic transmission/reception mode**

This transmission/reception mode is the same as the 3-wire serial I/O mode in which the specified number of data are transmitted/received in 8-bit units.

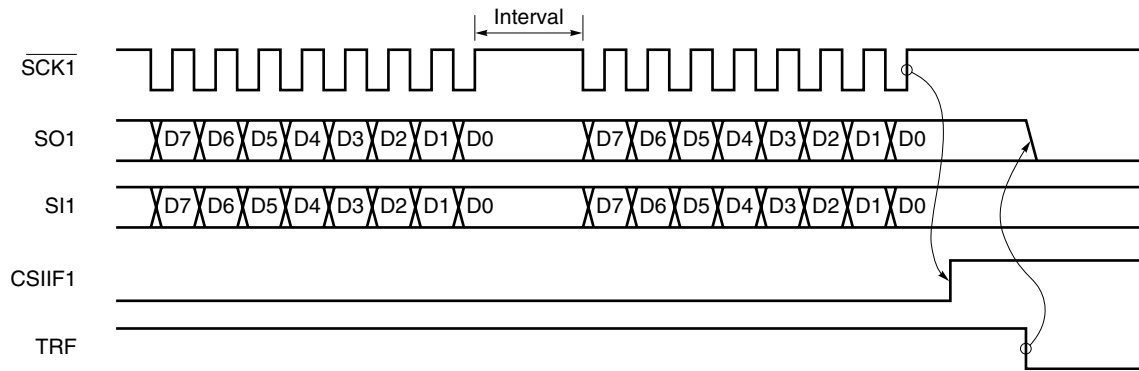
Serial transfer is started when any data is written to serial I/O shift register 1 (SIO1) while bit 7 (CSIE1) of serial operating mode register 1 (CSIM1) is set to 1.

When the final byte has been sent, an interrupt request flag (CSIF1) is set. However, judge the termination of auto transmit and receive not by CSIF1, but by bit 3 (TRF) of the automatic data transmit/receive control register (ADTC).

If busy control and strobe control are not executed, the P23/STB/TxD1 and P24/BUSY/RxD1 pins can be used as normal I/O ports.

Figure 18-8 shows the basic transmission/reception mode operation timing, and Figure 18-9 shows the operation flowchart. Figure 18-10 shows the operation of the internal buffer RAM when 6 bytes of data are transmitted or received.

**Figure 18-8. Basic Transmission/Reception Mode Operation Timing**



**Cautions**

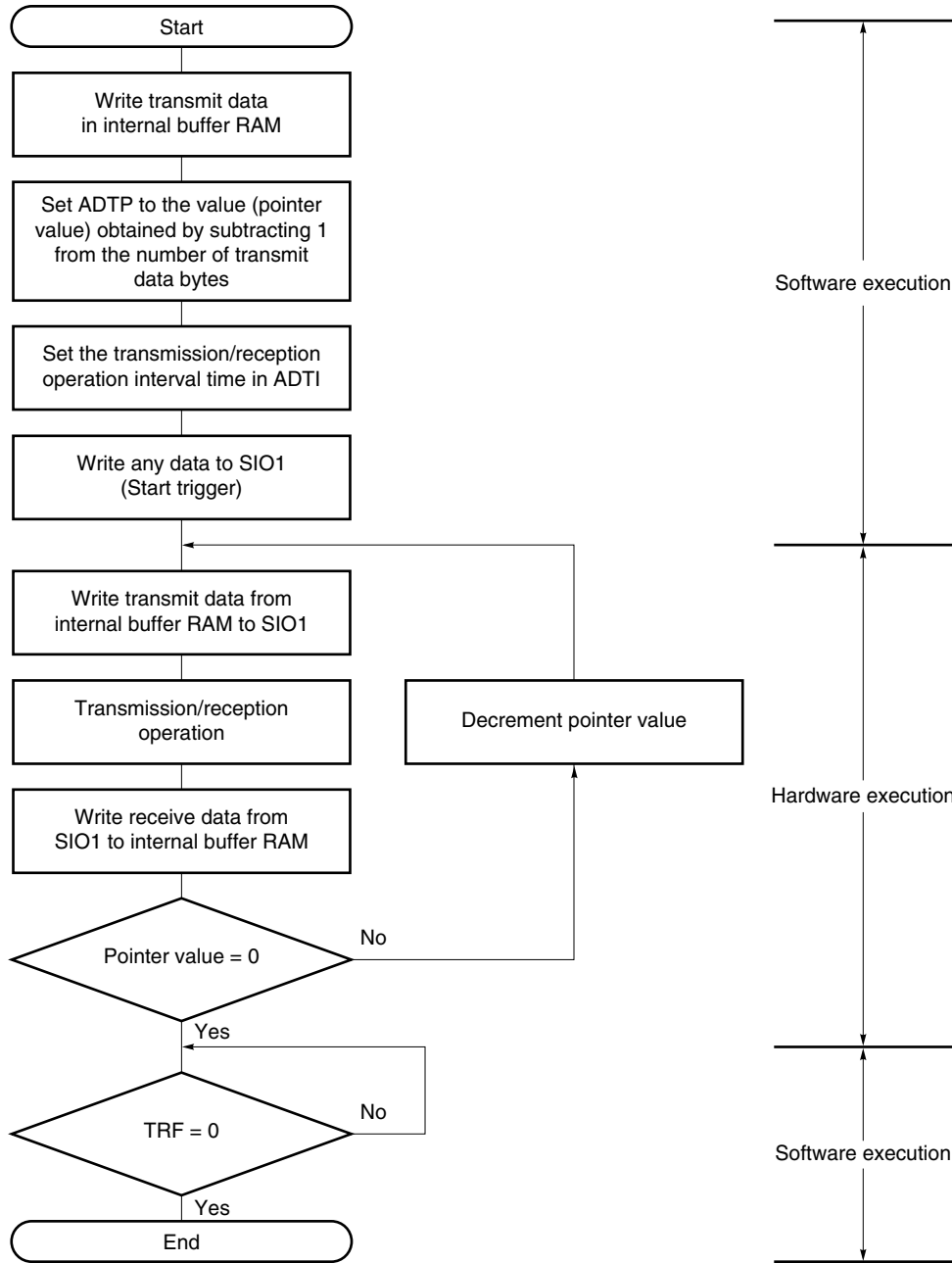
1. Because, in the basic transmission/reception mode, the automatic transmit/receive function writes/reads data to/from the internal buffer RAM after 1-byte transmission/reception, an interval is inserted until the next transmission/reception. As the buffer RAM write/read is performed at the same time as CPU processing, the maximum interval is dependent upon the CPU processing and the value of the automatic data transmit/receive interval specification register (ADTI) (see (5) Automatic data transmit/receive interval).

2. When TRF is cleared, the SO1 pin becomes low level.

**Remark** CSIF1: Interrupt request flag

TRF: Bit 3 of automatic data transmit/receive control register (ADTC)

Figure 18-9. Basic Transmission/Reception Mode Flowchart



- ADTP: Automatic data transmit/receive address pointer
- ADTI: Automatic data transmit/receive interval specification register
- SIO1: Serial I/O shift register 1
- TRF: Bit 3 of automatic data transmit/receive control register (ADTC)

In 6-byte transmission/reception (ARLD = 0, RE = 1) in basic transmit/receive mode, the internal buffer RAM operates as follows.

**(i) Before transmission/reception (see Figure 18-10 (a))**

After any data has been written to serial I/O shift register 1 (SIO1) (start trigger: this data is not transferred), transmit data 1 (T1) is transferred from the internal buffer RAM to SIO1. When transmission of the first byte is completed, receive data 1 (R1) is transferred from SIO1 to the buffer RAM, and the automatic data transmit/receive address pointer (ADTP) is decremented. Then transmit data 2 (T2) is transferred from the internal buffer RAM to SIO1.

**(ii) 4th byte transmission/reception point (see Figure 18-10 (b))**

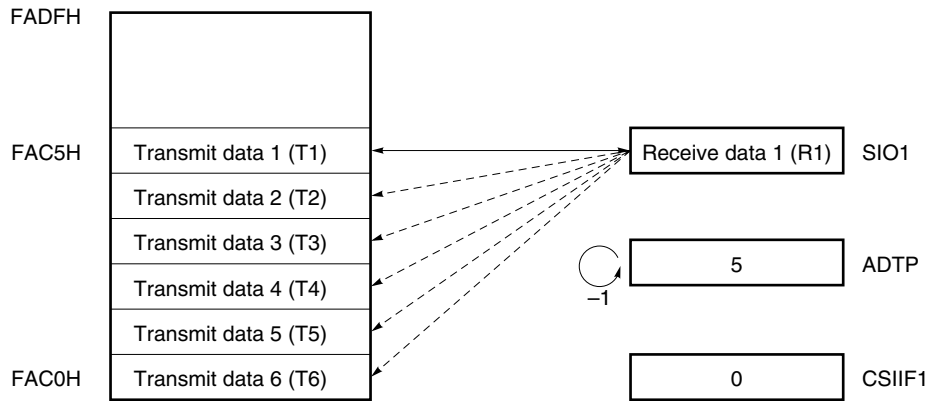
Transmission/reception of the third byte is completed, and transmit data 4 (T4) is transferred from the internal buffer RAM to SIO1. When transmission of the fourth byte is completed, receive data 4 (R4) is transferred from SIO1 to the internal buffer RAM, and ADTP is decremented.

**(iii) Completion of transmission/reception (see Figure 18-10 (c))**

When transmission of the sixth byte is completed, receive data 6 (R6) is transferred from SIO1 to the internal buffer RAM, and the interrupt request flag (CSIF1) is set (INTCSI1 generation).

**Figure 18-10. Internal Buffer RAM Operation in 6-Byte Transmission/Reception (in Basic Transmit/Receive Mode) (1/2)**

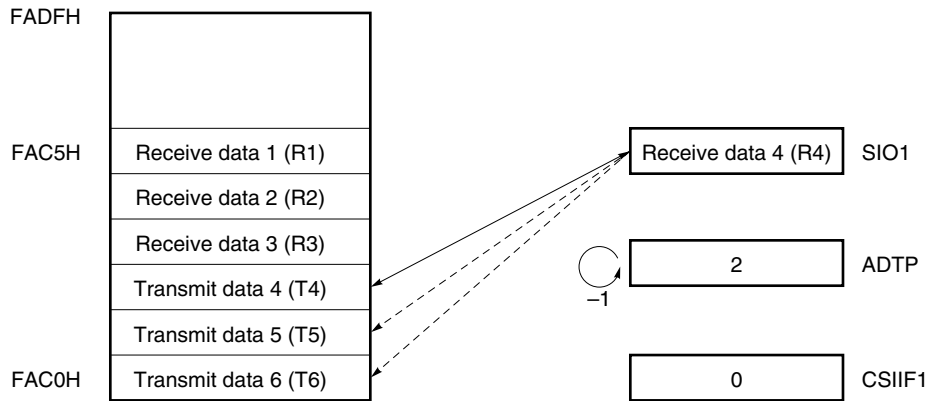
**(a) Before transmission/reception**



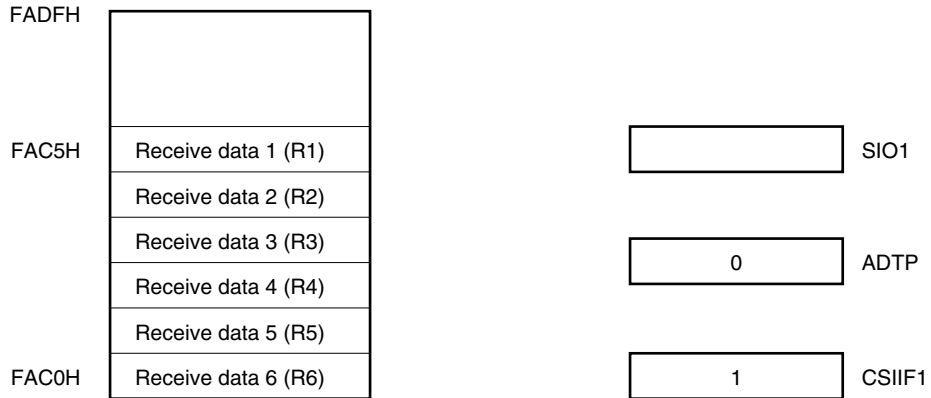


**Figure 18-10. Internal Buffer RAM Operation in 6-Byte Transmission/Reception (in Basic Transmit/Receive Mode) (2/2)**

**(b) 4th byte transmission/reception**



**(c) Completion of transmission/reception**



**(b) Basic transmission mode**

In this mode, 8-bit unit data is transmitted the specified number of times.

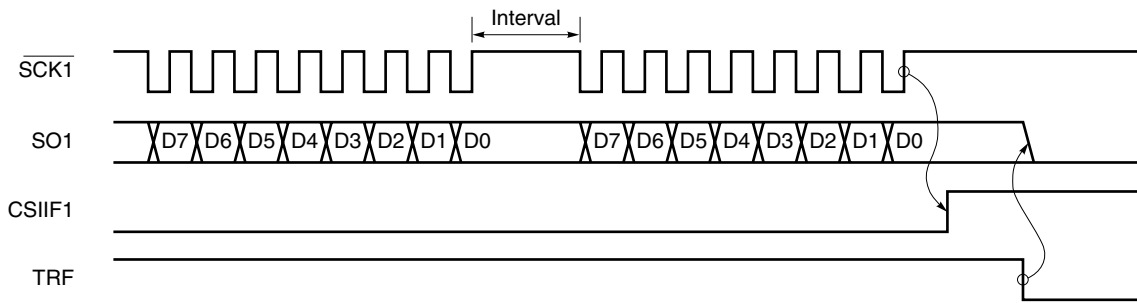
Serial transfer is started when any data is written to serial I/O shift register 1 (SIO1) while bit 7 (CSIE1) of serial operating mode register 1 (CSIM1) is set to 1.

When the final byte has been sent, an interrupt request flag (CSIIF1) is set. However, judge the termination of automatic transmit and receive not by CSIIF1, but by bit 3 (TRF) of the automatic data transmit/receive control register (ADTC).

If a receive operation, busy control and strobe control are not executed, the P20/SI1, P23/STB/TxD1 and P24/BUSY/RxD1 pins can be used as normal I/O ports.

Figure 18-11 shows the basic transmission mode operation timing, and Figure 18-12 shows the operation flowchart. Figure 18-13 shows the operation of the internal buffer RAM when 6 bytes of data are transmitted or received.

**Figure 18-11. Basic Transmission Mode Operation Timing**

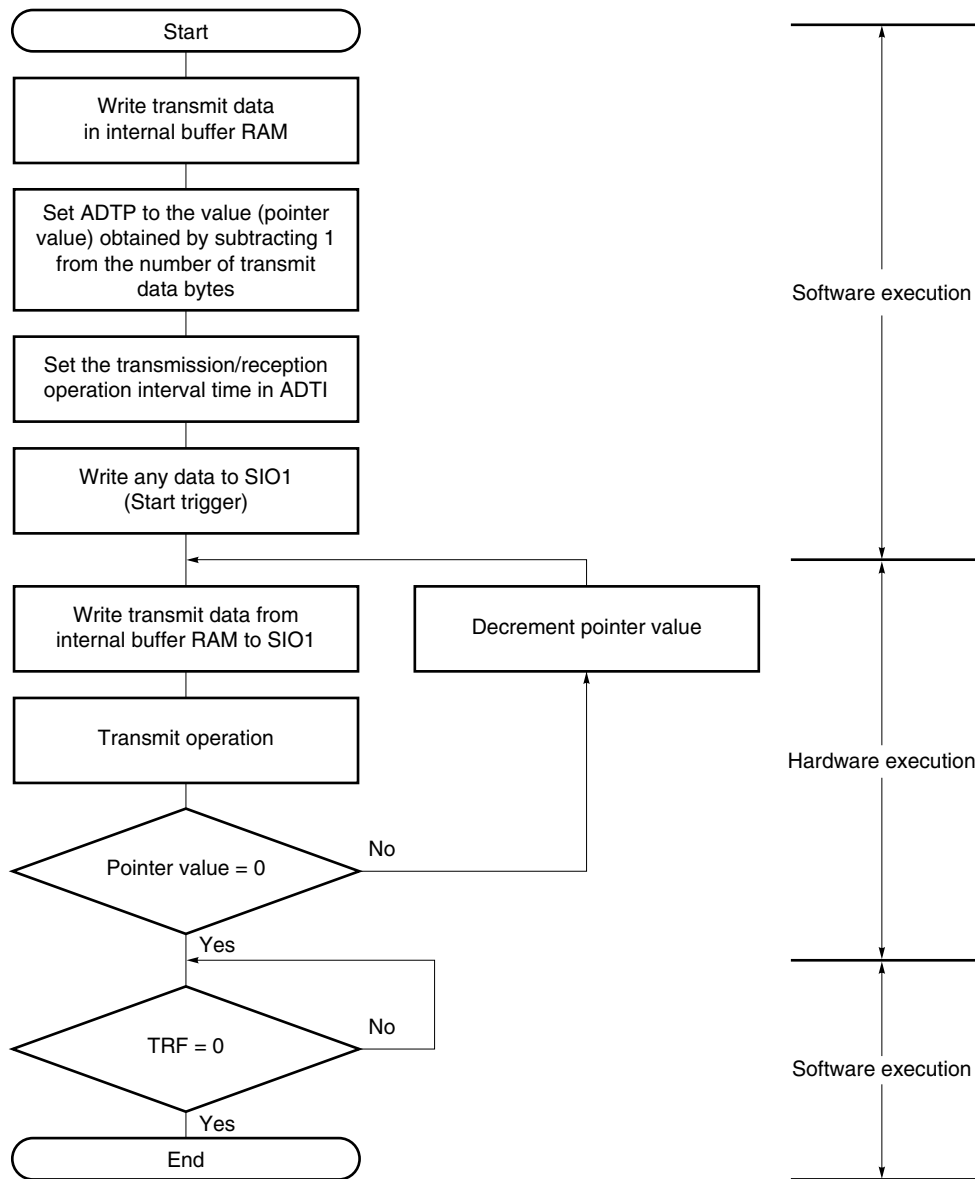


- Cautions**
1. Because, in the basic transmission mode, the automatic transmit/receive function reads data from the internal buffer RAM after 1-byte transmission, an interval is inserted until the next transmission. As buffer RAM read is performed at the same time as CPU processing, the maximum interval is dependent upon the CPU processing and the value of the automatic data transmit/receive interval specification register (ADTI) (see (5) Automatic data transmit/receive interval).
  2. When TRF is cleared, the SO1 pin becomes low level.

**Remark** CSIIF1: Interrupt request flag

TRF: Bit 3 of automatic data transmit/receive control register (ADTC)

Figure 18-12. Basic Transmission Mode Flowchart



ADTP: Automatic data transmit/receive address pointer  
 ADTI: Automatic data transmit/receive interval specification register  
 SIO1: Serial I/O shift register 1  
 TRF: Bit 3 of automatic data transmit/receive control register (ADTC)

In 6-byte transmission (ARLD = 0, RE = 0) in basic transmit mode, the internal buffer RAM operates as follows.

**(i) Before transmission (see Figure 18-13 (a).)**

After any data has been written to serial I/O shift register 1 (SIO1) (start trigger: this data is not transferred), transmit data 1 (T1) is transferred from the internal buffer RAM to SIO1. When transmission of the first byte is completed, the automatic data transmit/receive address pointer (ADTP) is decremented. Then transmit data 2 (T2) is transferred from the internal buffer RAM to SIO1.

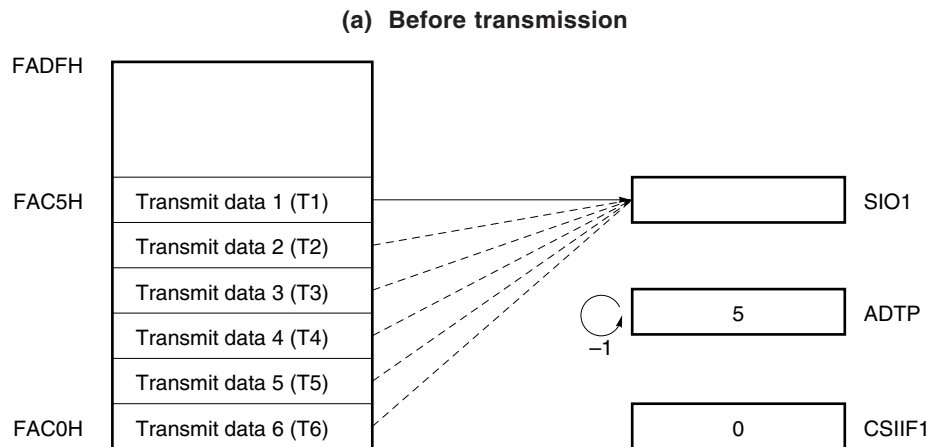
**(ii) 4th byte transmission point (see Figure 18-13 (b).)**

Transmission of the third byte is completed, and transmit data 4 (T4) is transferred from the internal buffer RAM to SIO1. When transmission of the fourth byte is completed, ADTP is decremented.

**(iii) Completion of transmission (see Figure 18-13 (c).)**

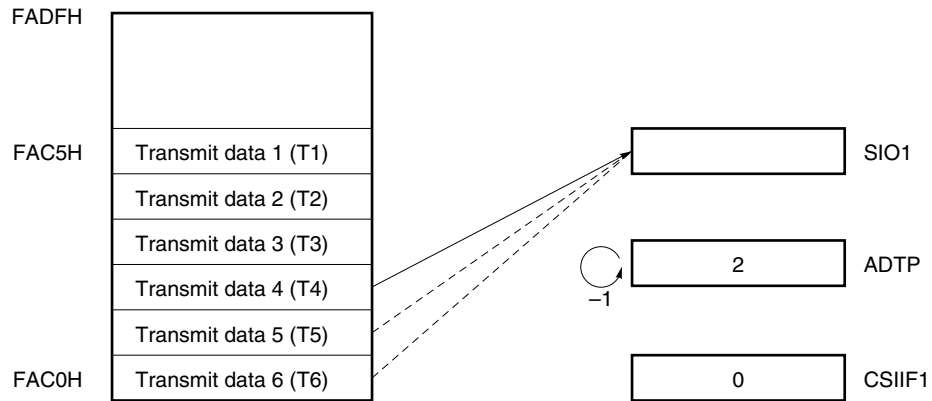
When transmission of the sixth byte is completed, the interrupt request flag (CSIF1) is set (INTCSI1 generation).

**Figure 18-13. Internal Buffer RAM Operation in 6-Byte Transmission (in Basic Transmit Mode) (1/2)**

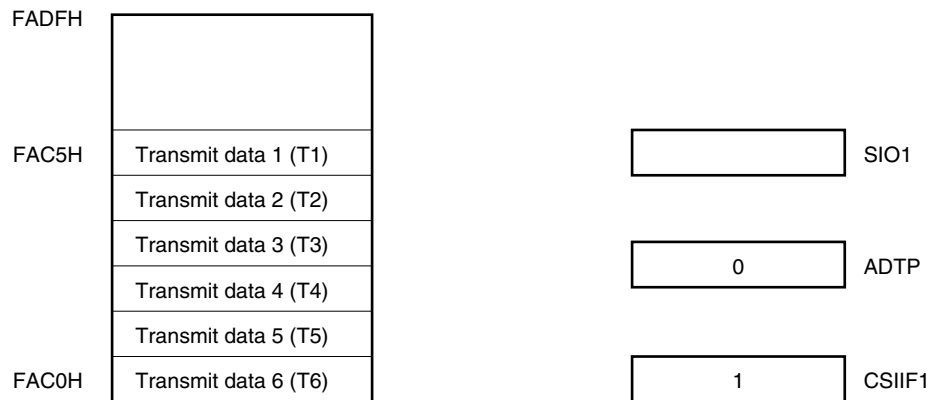


**Figure 18-13. Internal Buffer RAM Operation in 6-Byte Transmission (in Basic Transmit Mode) (2/2)**

**(b) 4th byte transmission point**



**(c) Completion of transmission**



**(c) Repeat transmission mode**

In this mode, data stored in the internal buffer RAM is transmitted repeatedly.

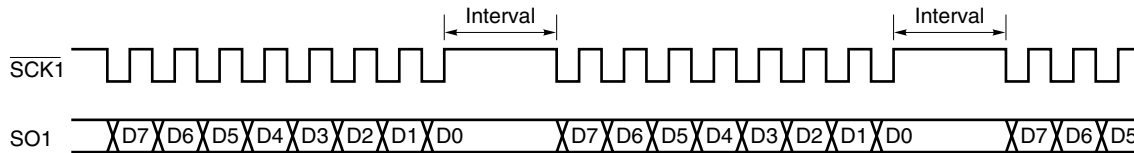
Serial transmission is started by writing any data to serial I/O shift register 1 (SIO1) when bit 7 (CSIE1) of serial operating mode register 1 (CSIM1) is set to 1.

Unlike the basic transmission mode, after the final byte (data at address FAC0H) has been transmitted, the interrupt request flag (CSIF1) is not set, the value at the time when transmission was started is set in the automatic data transmit/receive address pointer (ADTP) again, and the internal buffer RAM contents are transmitted again.

When a reception operation, busy control and strobe control are not performed, the P20/SI1, P23/STB/TxD1 and P24/BUSY/RxD1 pins can be used as normal I/O ports.

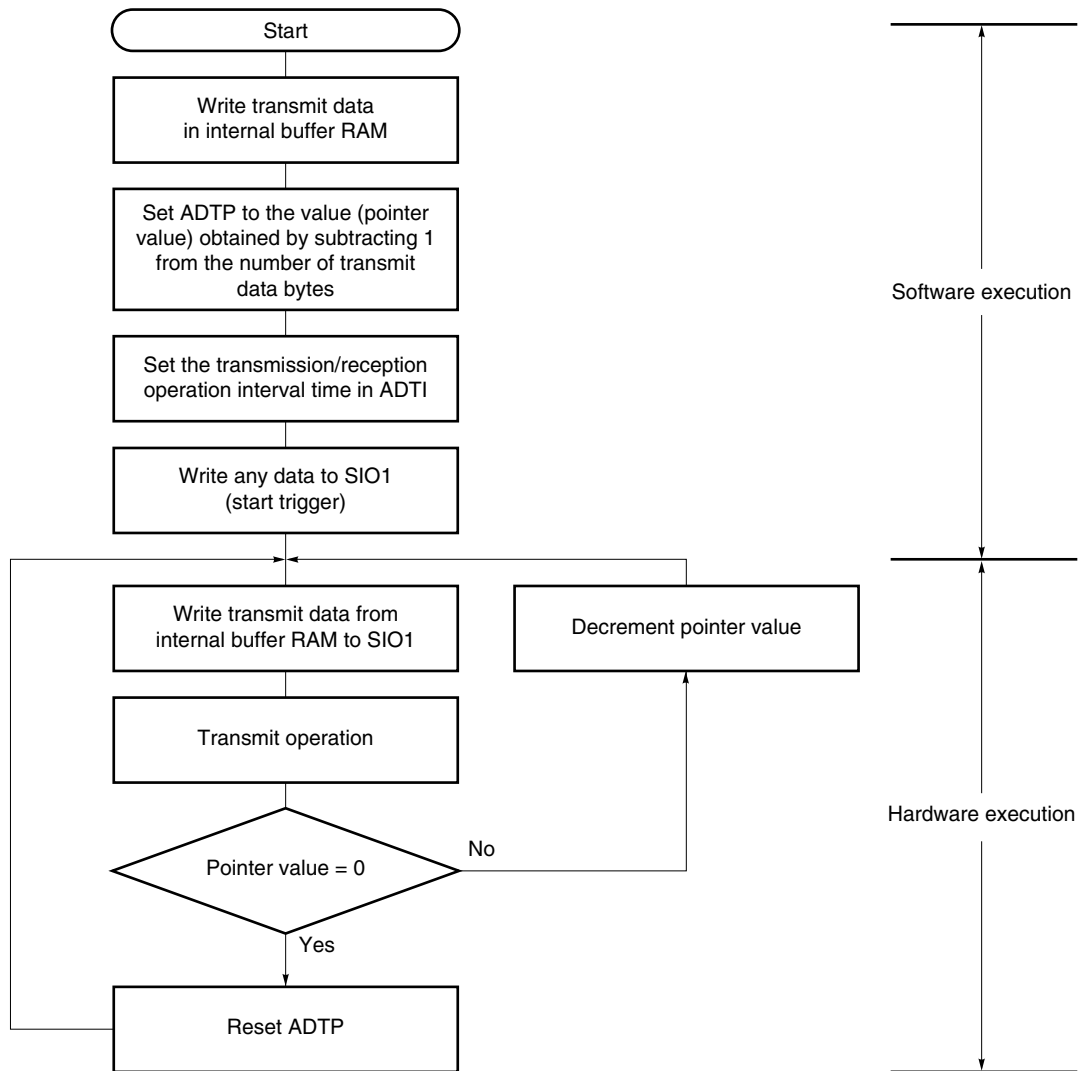
The repeat transmission mode operation timing is shown in Figure 18-14, and the operation flowchart in Figure 18-15. Figure 18-16 shows the operation of the internal buffer RAM when 6 bytes of data are transmitted in the repeat mode.

**Figure 18-14. Repeat Transmission Mode Operation Timing**



**Caution** Because, in the repeat transmission mode, a read is performed on the buffer RAM after the transmission of one byte, an interval is inserted in the period up to the next transmission. As buffer RAM read is performed at the same time as CPU processing, the maximum interval is dependent upon the CPU operation and the value of the automatic data transmit/receive interval specification register (ADTI) (see (5) Automatic data transmit/receive interval).

Figure 18-15. Repeat Transmission Mode Flowchart



ADTP: Automatic data transmit/receive address pointer  
 ADTI: Automatic data transmit/receive interval specification register  
 SIO1: Serial I/O shift register 1

In 6-byte transmission (ARLD = 1, RE = 0) in repeat transmit mode, the internal buffer RAM operates as follows.

**(i) Before transmission (see Figure 18-16 (a).)**

After any data has been written to serial I/O shift register 1 (SIO1) (start trigger: this data is not transferred), transmit data 1 (T1) is transferred from the internal buffer RAM to SIO1. When transmission of the first byte is completed, the automatic data transmit/receive address pointer (ADTP) is decremented. Then transmit data 2 (T2) is transferred from the internal buffer RAM to SIO1.

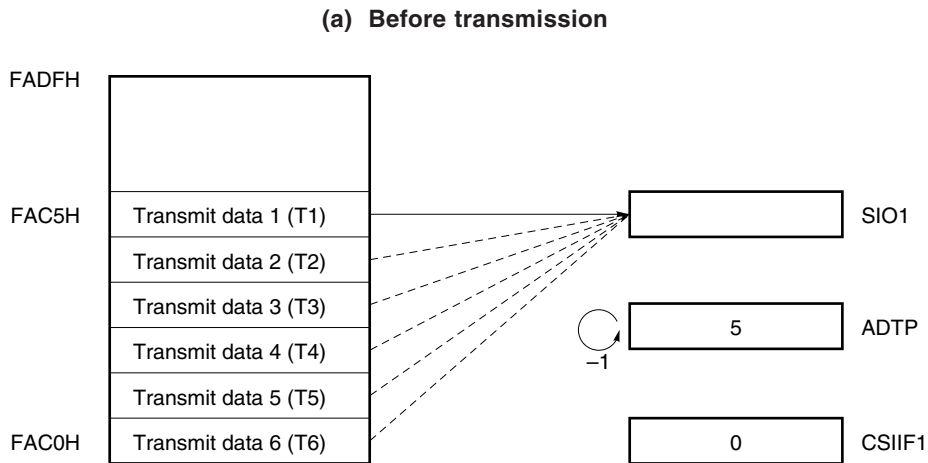
**(ii) Upon completion of transmission of 6 bytes (see Figure 18-16 (b).)**

When transmission of the sixth byte is completed, the interrupt request flag (CSIF1) is not set. The internal pointer value is reset in ADTP.

**(iii) 7th byte transmission point (see Figure 18-16 (c).)**

Transmit data 1 (T1) is transferred from the internal buffer RAM to SIO1 again. When transmission of the first byte is completed, ADTP is decremented. Then transmit data 2 (T2) is transferred from the internal buffer RAM to SIO1.

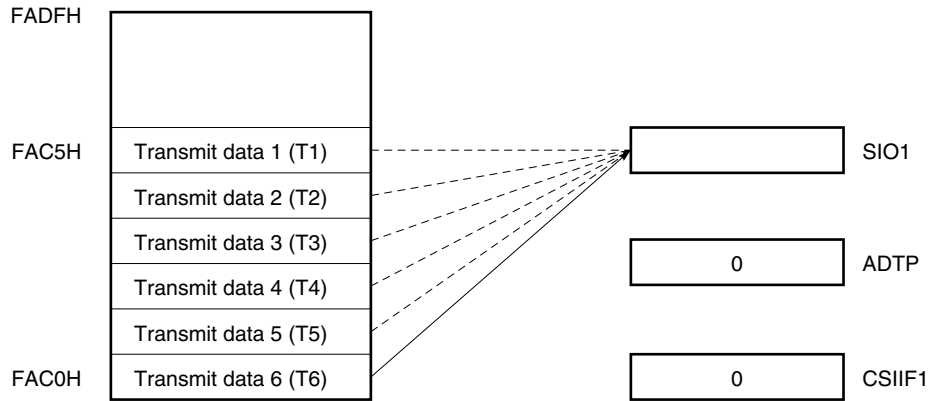
**Figure 18-16. Internal Buffer RAM Operation in 6-Byte Transmission (in Repeat Transmit Mode) (1/2)**



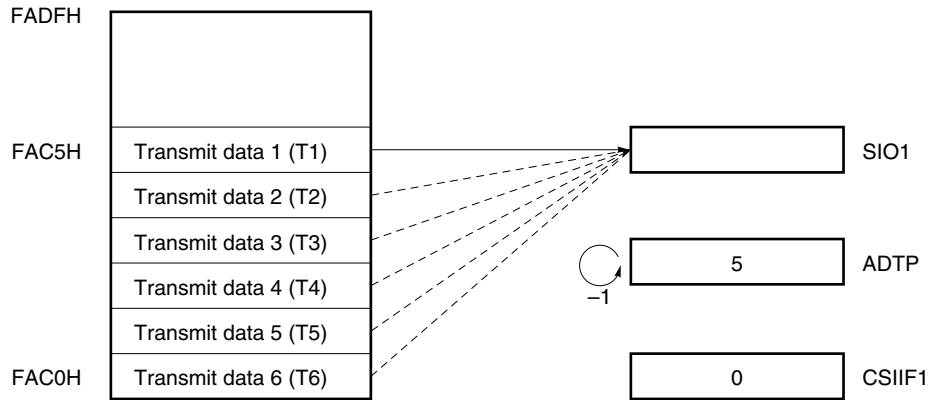


**Figure 18-16. Internal Buffer RAM Operation in 6-Byte Transmission (in Repeat Transmit Mode) (2/2)**

**(b) Upon completion of transmission of 6 bytes**



**(c) 7th byte transmission point**



**(d) Automatic transmission/reception suspending and restart**

Automatic transmission/reception can be temporarily suspended by clearing bit 7 (CSIE1) of serial operating mode register 1 (CSIM1) to 0.

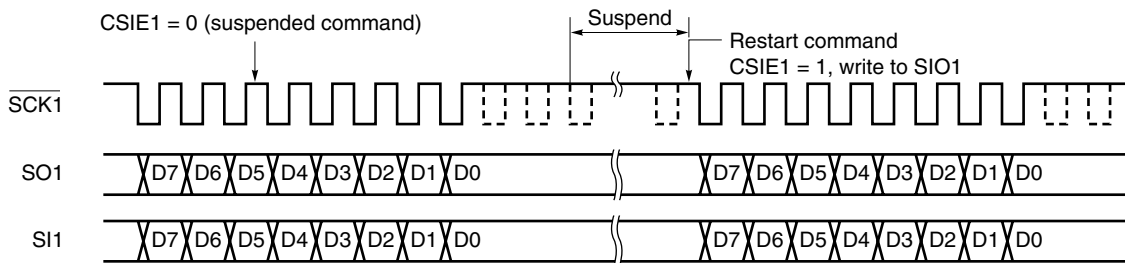
If during 8-bit data transfer, the transmission/reception is not suspended if bit 7 (CSIE1) is cleared to 0. It is suspended upon completion of 8-bit data transfer.

When suspended, bit 3 (TRF) of the automatic data transmit/receive control register (ADTC) is cleared to 0 after transfer of the 8th bit, and all the port pins used as serial interface alternate-function pins (P20/SI1, P21/SO1, P22/ $\overline{\text{SCK1}}$ , P23/STB/TxD1 and P24/BUSY/RxD1) are set to the port mode.

To restart automatic transmission/reception, set CSIE1 to 1 and write the desired value to serial I/O shift register 1 (SIO1). The remaining data can be transmitted in this way.

- Cautions**
1. If the HALT instruction is executed during automatic transmission/reception, transfer is suspended and the HALT mode is set, even if 8-bit data transfer is in progress. When the HALT mode is cleared, automatic transmission/reception is restarted from the suspended point.
  2. When suspending automatic transmission/reception, do not change the operating mode to 3-wire serial I/O mode while TRF = 1.

**Figure 18-17. Automatic Transmission/Reception Suspension and Restart**



CSIE1: Bit 7 of serial operating mode register 1 (CSIM1)

**(4) Synchronization control**

Busy control and strobe control are functions to synchronize transmission/reception between the master device and a slave device.

By using these functions, a shift in bits being transmitted or received can be detected.

**(a) Busy control option**

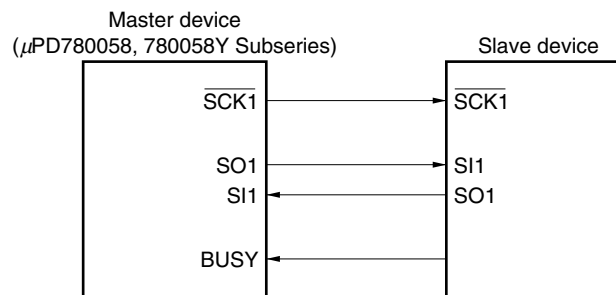
Busy control is a function to keep the serial transmission/reception by the master device waiting while the busy signal output by a slave device to the master is active.

When using this busy control option, the following conditions must be satisfied.

- Bit 5 (ATE) of serial operating mode register 1 (CSIM1) is set to 1.
- Bit 1 (BUSY1) of the automatic data transmit/receive control register (ADTC) is set to 1.

Figure 18-18 shows the system configuration of the master device and a slave device when the busy control option is used.

**Figure 18-18. System Configuration When Busy Control Option Is Used**



The master device inputs the busy signal output by the slave device to the BUSY/P24 pin. The master device samples the input busy signal in synchronization with the falling edge of the serial clock. Even if the busy signal becomes active while 8-bit data is being transmitted or received, transmission/reception by the master is not kept waiting. If the busy signal is active at the rising edge of the serial clock 2 clocks after completion of transmission/reception of the 8-bit data, the busy input becomes valid. After that, the master transmission/reception is kept waiting while the busy signal is active.

The active level of the busy signal is set by bit 0 (BUSY0) of ADTC.

BUSY0 = 0: Active high

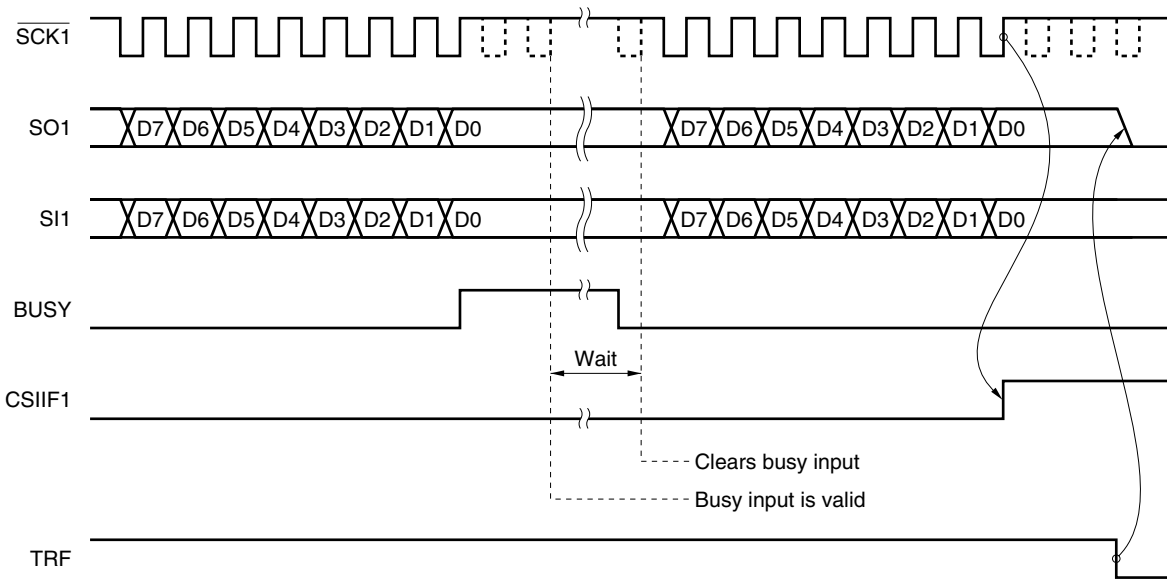
BUSY0 = 1: Active low

When using the busy control option, select the internal clock as the serial clock. Control with the busy signal cannot be implemented with an external clock.

Figure 18-19 shows the operation timing when the busy control option is used.

**Caution** Busy control cannot be used simultaneously with the interval time control function of the automatic data transmit/receive interval specification register (ADTI). If used, busy control is invalid.

Figure 18-19. Operation Timing When Busy Control Option Is Used (When BUSY0 = 0)



**Caution** If TRF is cleared, the SO1 pin goes low.

**Remark** CSIF1: Interrupt request flag

TRF: Bit 3 of the automatic data transmit/receive control register (ADTC)

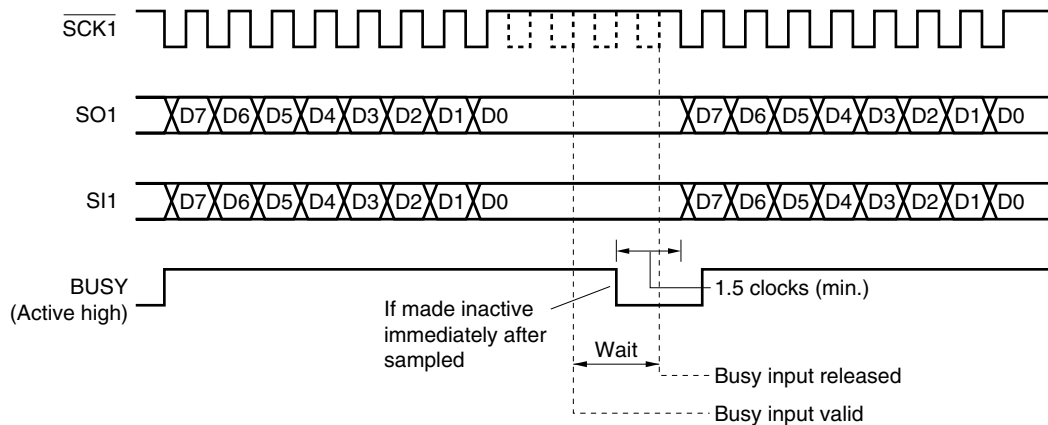
When the busy signal becomes inactive, waiting is released. If the sampled busy signal is inactive, transmission/reception of the next 8-bit data is started at the falling edge of the next clock.

Because the busy signal is asynchronous to the serial clock, it takes up to 1 clock until the busy signal is sampled, even if made inactive by the slave. It takes 0.5 clock until data transfer is started after the busy signal was sampled.

To accurately release waiting, the slave must keep the busy signal inactive at least for the duration of 1.5 clocks.

Figure 18-20 shows the timing of the busy signal and wait release. This figure shows an example where the busy signal is active as soon as transmission/reception has been started.

Figure 18-20. Busy Signal and Wait Release (When BUSY0 = 0)

**(b) Busy & strobe control option**

Strobe control is a function to synchronize data transmission/reception between the master and slave devices. The master device outputs the strobe signal from the STB/P23 pin when 8-bit transmission/reception has been completed. By this signal, the slave device can determine the timing of the end of data transmission. Therefore, synchronization is established even if a bit shift occurs because noise is superimposed on the serial clock, and transmission of the next byte is not affected by the bit shift. To use the strobe control option, the following conditions must be satisfied.

- Bit 5 (ATE) of serial operating mode register 1 (CSIM1) is set to 1.
- Bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) is set to 1.

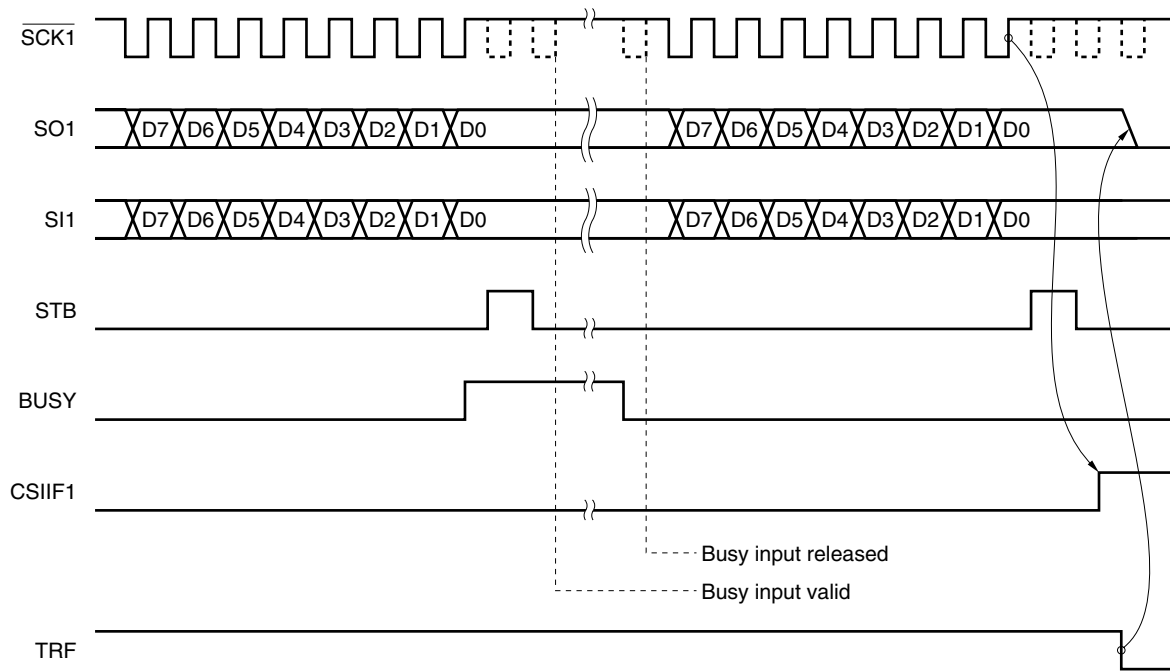
Usually, the busy control and strobe control options are simultaneously used as handshake signals. In this case, the strobe signal is output from the STB/P23 pin, the BUSY/P24 pin is sampled, and transmission/reception can be kept waiting while the busy signal is input.

When the strobe control option is not used, the P23/STB pin can be used as a normal I/O port pin.

Figure 18-21 shows the operation timing when the busy & strobe control options are used.

When the strobe control option is used, the interrupt request flag (CSIIF1) that is set on completion of transmission/reception is set after the strobe signal is output.

Figure 18-21. Operation Timing When Busy & Strobe Control Options Are Used (When BUSY0 = 0)



**Caution** When TRF is cleared, the SO1 pin goes low.

**Remark** CSIF1: Interrupt request flag

TRF: Bit 3 of the automatic data transmit/receive control register (ADTC)

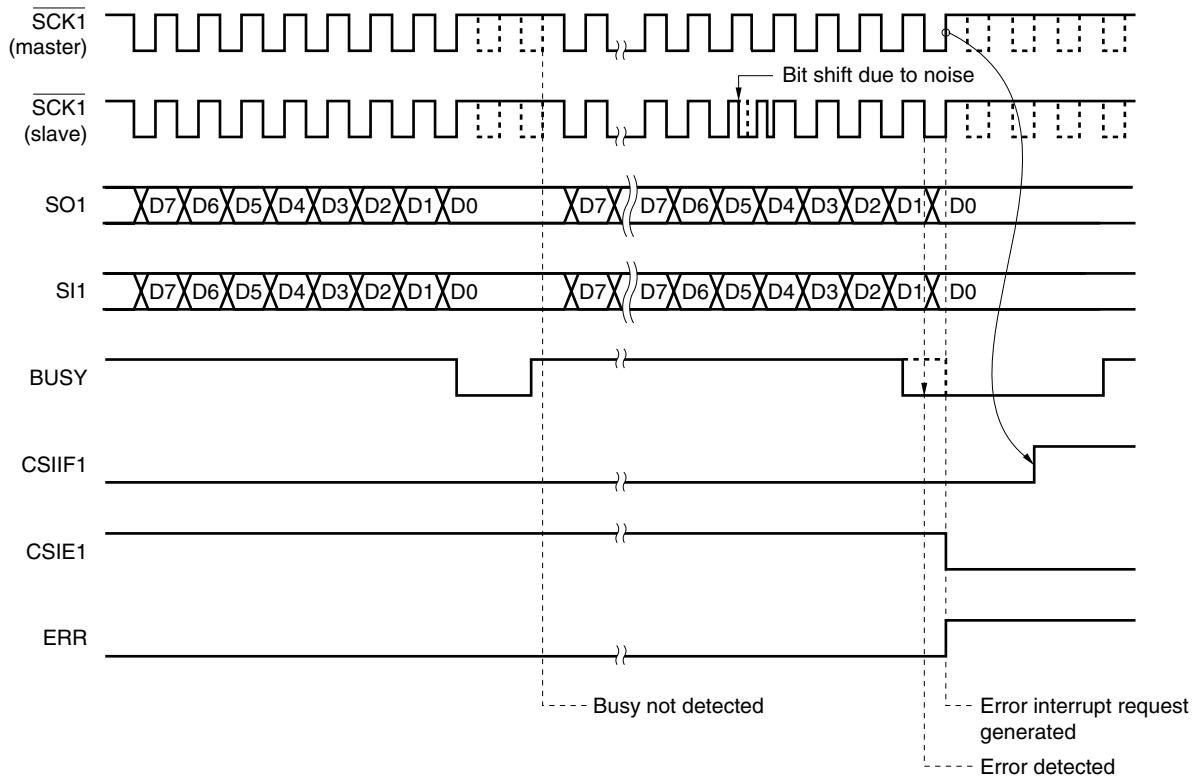
**(c) Bit shift detection by busy signal**

During automatic transmission/reception, a bit shift of the serial clock of the slave device may occur because noise is superimposed on the serial clock signal output by the master device. Unless the strobe control option is used at this time, the bit shift affects transmission of the next byte. In this case, the master can detect the bit shift by checking the busy signal during transmission by using the busy control option. A bit shift is detected by using the busy signal as follows.

The slave outputs the busy signal after the rising of the eighth serial clock during data transmission/reception (to not keep transmission/reception waiting by the busy signal at this time, make the busy signal inactive within 2 clocks).

The master samples the busy signal in synchronization with the falling edge of the leading side of the serial clock. If a bit shift does not occur, all the eight serial clocks that have been sampled are inactive. If the sampled serial clocks are active, it is assumed that a bit shift has occurred, and error processing is executed (by setting bit 4 (ERR) of the automatic data transmit/receive control register (ADTC) to 1). Figure 18-22 shows the operation timing of the bit shift detection function by the busy signal.

**Figure 18-22. Operation Timing of Bit Shift Detection Function by Busy Signal (When BUSY0 = 1)**



- CSIF1: Interrupt request flag
- CSIE1: Bit 7 of serial operating mode register 1 (CSIM1)
- ERR: Bit 4 of the automatic data transmit/receive control register (ADTC)

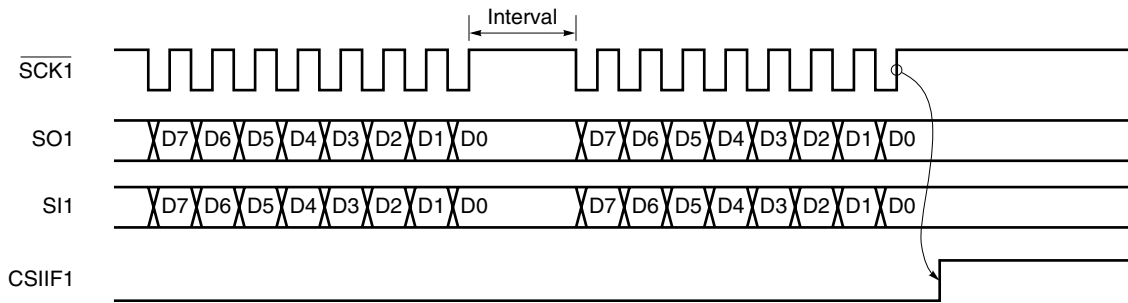
**(5) Automatic transmit/receive interval time**

When using the automatic transmit/receive function, the read/write operations from/to the internal buffer RAM are performed after transmitting/receiving one byte. Therefore, an interval is inserted before the next transmit/receive operation.

Since the read/write operations from/to the buffer RAM are performed in parallel with the CPU processing when using the automatic transmit/receive function with the internal clock, the interval depends on the value which is set in the automatic transmit/receive interval specification register (ADTI) and the CPU processing at the rising edge of the eighth serial clock. Whether it depends on the ADTI or not can be selected by setting bit 7 of ADTI (ADTI7). When it is cleared to 0, the interval depends only on the CPU processing. When it is set to 1, the interval depends on the contents of ADTI or the CPU processing, whichever is greater.

When the automatic transmit/receive function is used with an external clock, it must be selected so that the interval may be longer than the value indicated by paragraph (b).

**Figure 18-23. Automatic Data Transmit/Receive Interval Time**



CSIF1: Interrupt request flag



**(a) When the automatic transmit/receive function is used with the internal clock**

If bit 1 (CSIM11) of serial operating mode register 1 (CSIM1) is set to 1, the internal clock operates. If the automatic transmit/receive function is operated with the internal clock, the interval timing according to CPU processing is as follows.

When bit 7 (ADTI7) of the automatic data transmit/receive interval specification register (ADTI) is cleared to 0, the interval depends on the CPU processing. When ADTI7 is set to 1, it depends on the contents of ADTI or the CPU processing, whichever is greater.

See **Figure 18-5 Automatic Data Transmit/Receive Interval Specification Register Format** for the intervals set by ADTI.

**Table 18-2. Interval Timing According to CPU Processing (When Internal Clock Is Operating)**

CPU Processing	Interval Time
When using multiplication instruction	Max. ( $2.5T_{SCK}$ , $13T_{CPU}$ )
When using division instruction	Max. ( $2.5T_{SCK}$ , $20T_{CPU}$ )
External access 1 wait mode	Max. ( $2.5T_{SCK}$ , $9T_{CPU}$ )
Other than above	Max. ( $2.5T_{SCK}$ , $7T_{CPU}$ )

$T_{SCK}$ :  $1/f_{SCK}$

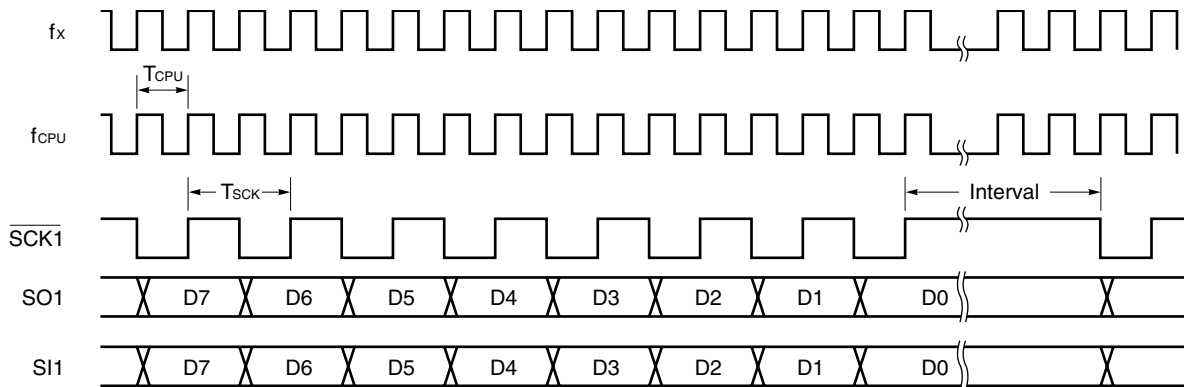
$f_{SCK}$ : Serial clock frequency

$T_{CPU}$ :  $1/f_{CPU}$

$f_{CPU}$ : CPU clock (set by bits 0 to 2 (PCC0 to PCC2) of the processor clock control register (PCC) and bit 0 (MCS) of the oscillation mode select register (OSMS))

MAX. (a, b): a or b, whichever is greater

**Figure 18-24. Operation Timing with Automatic Data Transmit/Receive Function Performed Using Internal Clock**



$f_x$ : Main system clock oscillation frequency

$f_{CPU}$ : CPU clock (set by bits 0 to 2 (PCC0 to PCC2) of the processor clock control register (PCC))

$T_{CPU}$ :  $1/f_{CPU}$

$T_{SCK}$ :  $1/f_{SCK}$

$f_{SCK}$ : Serial clock frequency

**(b) When using automatic transmit/receive function with external clock**

An external clock is used when bit 1 (CSIM11) of serial operating mode register 1 (CSIM1) is cleared to 0.

To use the automatic transmit/receive function with an external clock, the external clock must be input so that the interval time is as follows.

**Table 18-3. Interval Time According to CPU Processing (with External Clock)**

CPU Processing	Interval Time
When using multiplication instruction	$13T_{\text{CPU}}$ or more
When using division instruction	$20T_{\text{CPU}}$ or more
External access 1 wait mode	$9T_{\text{CPU}}$ or more
Other than above	$7T_{\text{CPU}}$ or more

$T_{\text{CPU}}$ :  $1/f_{\text{CPU}}$

$f_{\text{CPU}}$ : CPU clock (set by the bits 0 to 2 (PCC0 to PCC2) of the processor clock control register (PCC) and bit 0 (MCS) of the oscillation mode select register (OSMS))

## CHAPTER 19 SERIAL INTERFACE CHANNEL 2

### 19.1 Functions of Serial Interface Channel 2

Serial interface channel 2 has the following three modes.

- Operation stop mode
- Asynchronous serial interface (UART) mode (with time-division transfer function)
- 3-wire serial I/O mode

#### (1) Operation stop mode

This mode is used when serial transfer is not carried out to reduce power consumption.

#### (2) Asynchronous serial interface (UART) mode (with time-division transfer function)

In this mode, one byte of data is transmitted/received following the start bit, and full-duplex operation is possible.

A dedicated UART baud rate generator is incorporated, allowing communication over a wide range of baud rates. In addition, the baud rate can be defined by dividing the clock input to the ASCK pin.

The MIDI standard baud rate (31.25 kbps) can be used by employing the dedicated UART baud rate generator. Two sets of data I/O pins (RxD and TxD) are provided, and the pin to be used can be selected by software (time-division transfer function). However, only one set of pins can be used at one time.

**Cautions** 1. If it is not necessary to change the data I/O pin, use of the RxD0/SI2/P70 and TxD0/SO2/P71 pins is recommended. If only port 2 (RxD1/BUSY/P24 and TxD1/STB/P23) is used as data I/O pins, the function of port 7 is limited.

2. When using the busy control option or busy & strobe control option in the 3-wire serial I/O mode with automatic transmit/receive function of serial interface channel 1, the RxD1/BUSY/P24 and TxD1/STB/P23 pins cannot be used as data I/O pins.

#### (3) 3-wire serial I/O mode (MSB-first/LSB-first switchable)

In this mode, 8-bit data transfer is performed using three lines: the serial clock ( $\overline{\text{SCK2}}$ ), and serial data lines (SI2, SO2).

In the 3-wire serial I/O mode, simultaneous transmission and reception is possible, increasing the data transfer processing speed.

Either the MSB or LSB can be specified as the start bit for an 8-bit data serial transfer, allowing connection to devices using either as the start bit.

The 3-wire serial I/O mode is useful for connection to peripheral I/Os and display controllers, etc., which incorporate a conventional clocked serial interface, such as the 75X/XL Series, 78K Series, 17K Series, etc.

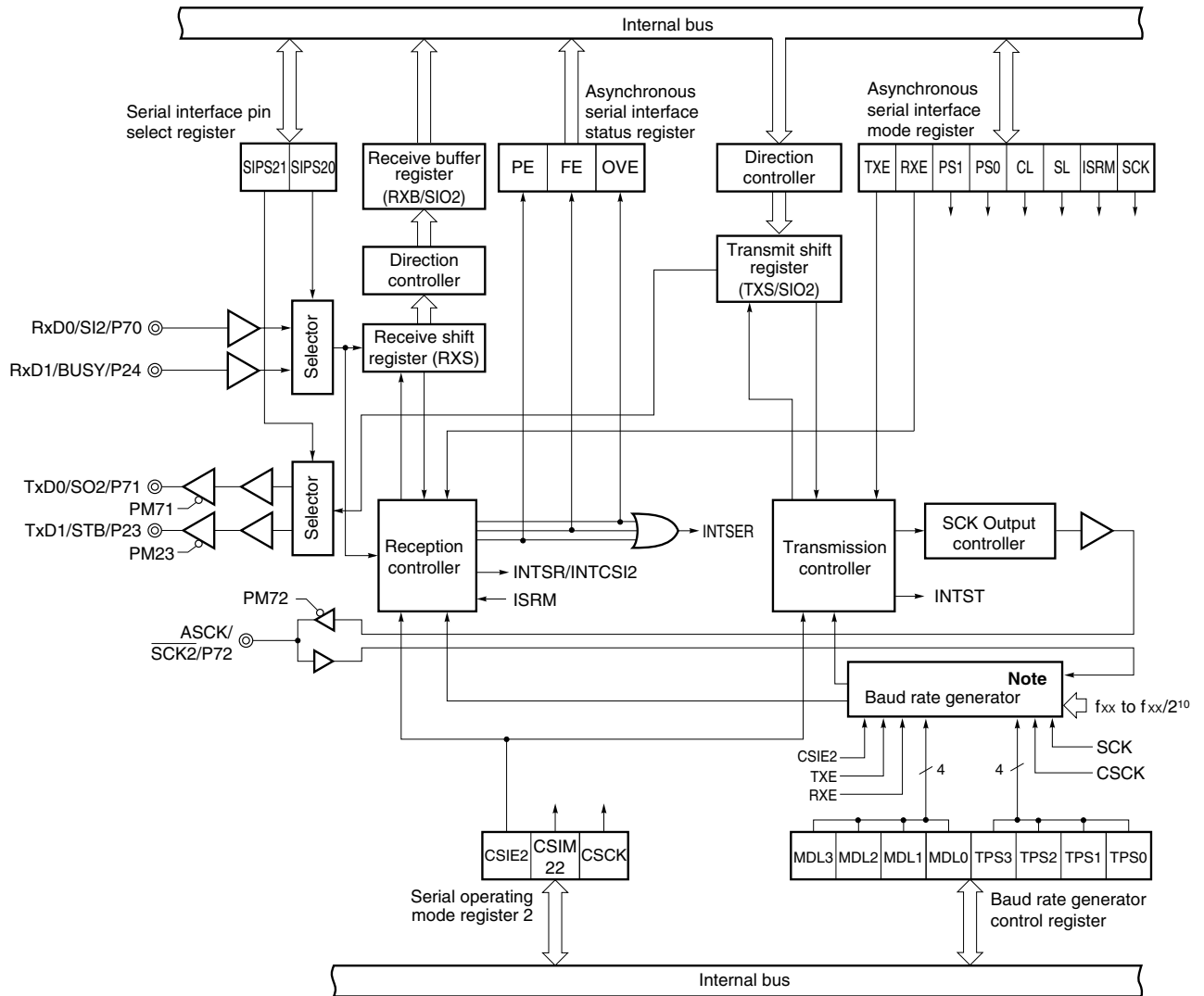
## 19.2 Configuration of Serial Interface Channel 2

Serial interface channel 2 consists of the following hardware.

**Table 19-1. Configuration of Serial Interface Channel 2**

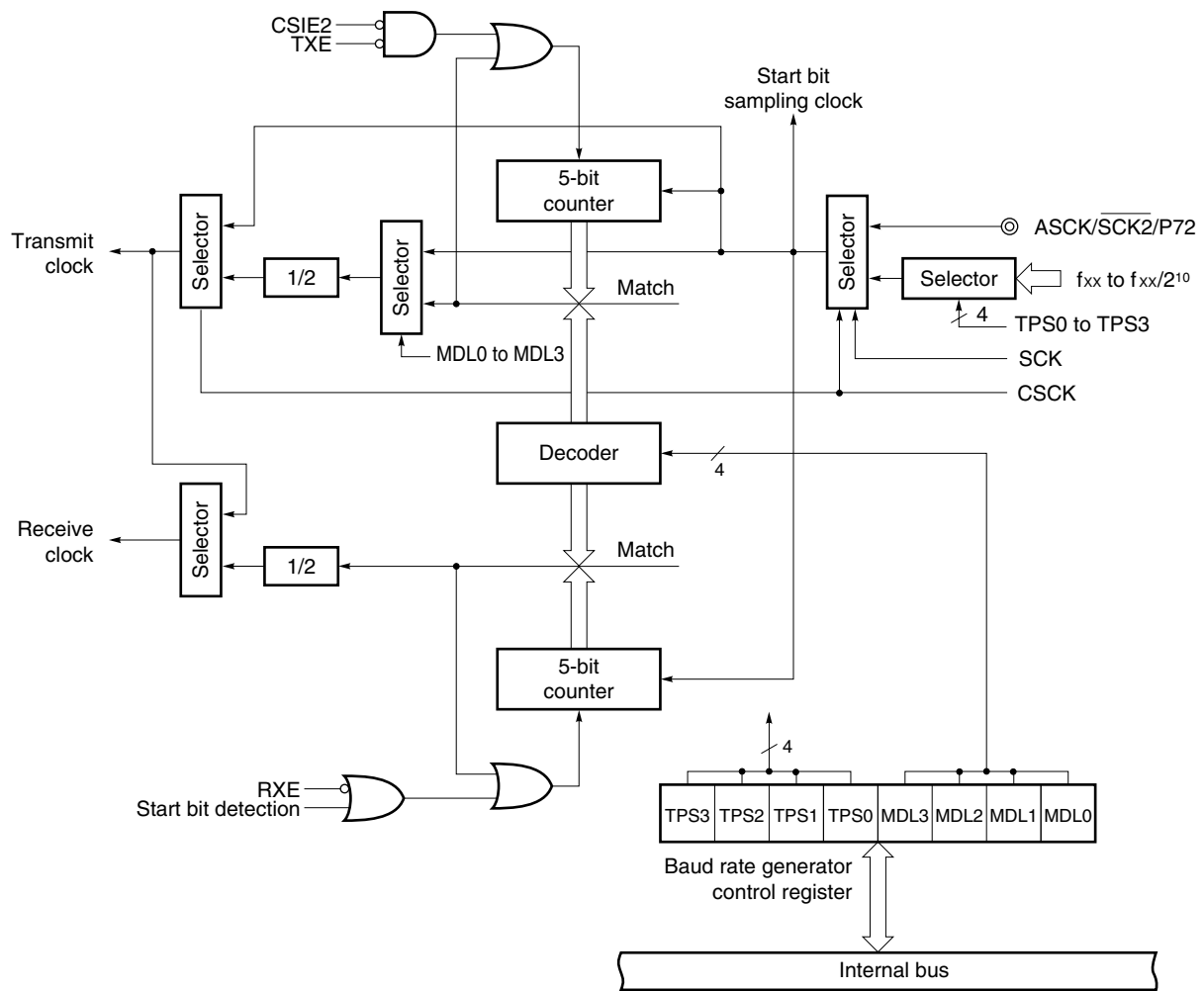
Item	Configuration
Registers	Transmit shift register (TXS) Receive shift register (RXS) Receive buffer register (RXB)
Control registers	Serial operating mode register 2 (CSIM2) Asynchronous serial interface mode register (ASIM) Asynchronous serial interface status register (ASIS) Baud rate generator control register (BRGC) Serial interface pin select register (SIPS)

Figure 19-1. Block Diagram of Serial Interface Channel 2



**Note** See Figure 19-2 for the baud rate generator configuration.

Figure 19-2. Baud Rate Generator Block Diagram



**(1) Transmit shift register (TXS)**

This register is used to set the transmit data. The data written in TXS is transmitted as serial data. If the data length is specified as 7 bits, bits 0 to 6 of the data written in TXS are transferred as transmit data. Writing data to TXS starts the transmit operation. TXS is written with an 8-bit memory manipulation instruction. It cannot be read.  $\overline{\text{RESET}}$  input sets TXS to FFH.

**Caution** TXS must not be written during a transmit operation. TXS and the receive buffer register (RXB) are allocated to the same address, and when a read is performed, the value of RXB is read.

**(2) Receive shift register (RXS)**

This register is used to convert serial data input to the RxD0 (RxD1) pin into parallel data. When one byte of data is received, the receive data is transferred to the receive buffer register (RXB). RXS cannot be directly manipulated by a program.

**(3) Receive buffer register (RXB)**

This register holds receive data. Each time one byte of data is received, new receive data is transferred from the receive shift register (RXS). If the data length is specified as 7 bits, the receive data is transferred to bits 0 to 6 of RXB, and the MSB of RXB is always cleared to 0. RXB is read with an 8-bit memory manipulation instruction. It cannot be written to.  $\overline{\text{RESET}}$  input sets RXB to FFH.

**Caution** RXB and the transmit shift register (TXS) are allocated to the same address, and when a write is performed, the value is written to TXS.

**(4) Transmission controller**

This circuit performs transmit operation control such as the addition of a start bit, parity bit, and stop bit to data written in the transmit shift register (TXS) in accordance with the contents set in the asynchronous serial interface mode register (ASIM).

**(5) Reception controller**

This circuit controls receive operations in accordance with the contents set in the asynchronous serial interface mode register (ASIM). It performs error checks for parity errors, etc., during a receive operation, and if an error is detected, sets a value in the asynchronous serial interface status register (ASIS) in accordance with the error contents.

### 19.3 Control Registers of Serial Interface Channel 2

Serial interface channel 2 is controlled by the following five registers.

- Serial operating mode register 2 (CSIM2)
- Asynchronous serial interface mode register (ASIM)
- Asynchronous serial interface status register (ASIS)
- Baud rate generator control register (BRGC)
- Serial interface pin select register (SIPS)

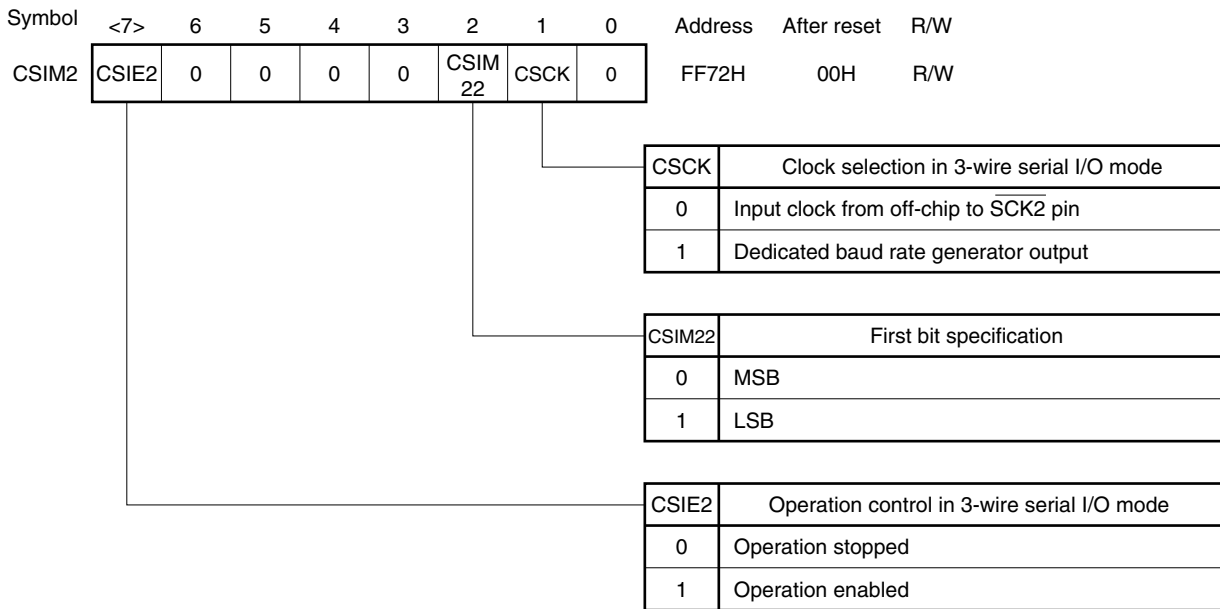
#### (1) Serial operating mode register 2 (CSIM2)

This register is set when serial interface channel 2 is used in the 3-wire serial I/O mode.

CSIM2 is set with a 1-bit or an 8-bit memory manipulation instruction.

RESET input sets CSIM2 to 00H.

**Figure 19-3. Format of Serial Operating Mode Register 2**



- Cautions**
1. Be sure to clear bits 0 and 3 to 6 to 0.
  2. When UART mode is selected, CSIM2 should be cleared to 00H.



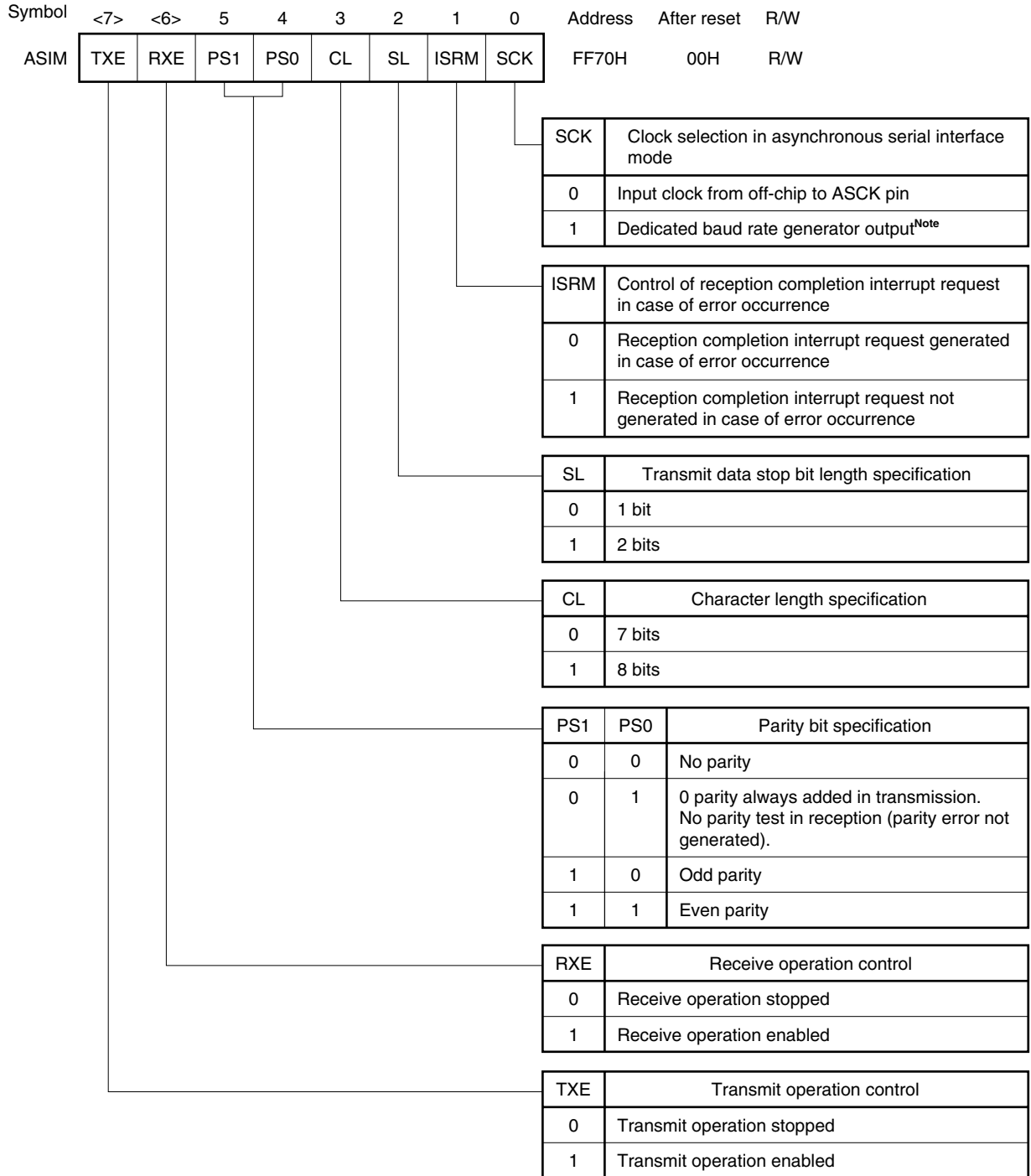
**(2) Asynchronous serial interface mode register (ASIM)**

This register is set when serial interface channel 2 is used in the asynchronous serial interface mode.

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ASIM to 00H.

**Figure 19-4. Format of Asynchronous Serial Interface Mode Register**



**Note** When SCK is set to 1 and the baud rate generator output is selected, the ASCK pin can be used as an I/O port.

- Cautions**
1. When the 3-wire serial I/O mode is selected, ASIM should be cleared to 00H.
  2. The serial transmit/receive operation must be stopped before changing the operating mode.

Table 19-2. Operating Mode Settings of Serial Interface Channel 2 (1/2)

(1) Operation stop mode

ASIM			CSIM2			SIPS		PM70	P70	PM71	P71	PM23	P23	PM24	P24	PM72	P72	Start bit	Shift clock	P70/SI2/RxD0 pin function	P71/SO2/TxD0 pin function	P23/STB/TxD1 pin function	P24/BUSY/RxD1 pin function	P72/SCK2/ASCK pin function
TXE	RXE	SCK	CSIE2	CSIM22	CSCK	SIPS21	SIPS20																	
0	0	x	0	x	x	x	x	Note 1	Note 1	Note 1	Note 1	Note 1	Note 1	Note 1	Note 1	Note 1	Note 1	—	—	P70	P71	P23/STB	P24/BUSY	P72
Other than above																		Setting prohibited						

(2) 3-wire serial I/O mode

ASIM			CSIM2			SIPS		PM70	P70	PM71	P71	PM23	P23	PM24	P24	PM72	P72	Start bit	Shift clock	P70/SI2/RxD0 pin function	P71/SO2/TxD0 pin function	P23/STB/TxD1 pin function	P24/BUSY/RxD1 pin function	P72/SCK2/ASCK pin function
TXE	RXE	SCK	CSIE2	CSIM22	CSCK	SIPS21	SIPS20																	
0	0	0	1	0	0	x	x	Note 2	Note 2	0	1	Note 1	Note 1	Note 1	Note 1	1	x	MSB	External clock	SI2 <sup>Note 2</sup>	SO2 (CMOS output)	P23/STB	P24/BUSY	SCK2 input
					1										0	1		Internal clock						SCK2 output
			1	1	0										1	x	LSB	External clock	SI2 <sup>Note 2</sup>	SO2 (CMOS output)				SCK2 input
					1										0	1		Internal clock						SCK2 output
Other than above																		Setting prohibited						

Notes 1. Can be used freely as a port function.

2. Can be used as P70 (CMOS I/O) when only transmission is performed.

Remark x: don't care  
 PMxx: Port mode register  
 Pxx: Port output latch

Table 19-2. Operating Mode Setting of Serial Interface Channel 2 (2/2)

(3) Asynchronous serial interface mode

ASIM			CSIM2			SIPS		PM70	P70	PM71	P71	PM23	P23	PM24	P24	PM72	P72	Start bit	Shift clock	P70/SI2/RxD0 pin function	P71/SO2/TxD0 pin function	P23/STB/TxD1 pin function	P24/BUSY/RxD1 pin function	P72/SCK2/ASCK pin function
TXE	RXE	SCK	CSIE2	CSIM22	CSCK	SIPS21	SIPS20																	
1	0	0	0	0	0	0	0	x <sup>Note 1</sup>	x <sup>Note 1</sup>	0	1	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	1	x	LSB	External clock	P70	TxD0 (CMOS output)	P23/STB	P24/BUSY	ASCK input
		1																						Internal clock
0	1	0	0	0	0	0	0	1	x	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	1	x	External clock	RxD0	P71	P23/STB	P24/BUSY	ASCK input	
		1																					Internal clock	P72
1	1	0	0	0	0	0	0	1	x	0	1	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	1	x	External clock	TxD0 (CMOS output)	P23/STB	P24/BUSY	ASCK input		
		1																				Internal clock	P72	
1	0	0	0	0	0	1	0	x <sup>Note 1</sup>	x <sup>Note 1</sup>	0	1	0	Note 2	x <sup>Note 1</sup>	x <sup>Note 1</sup>	1	x	External clock	P70	High output	TxD1	P24/BUSY	ASCK input	
		1																					Internal clock	P72
0	1	0	0	0	0	0	1	1	x	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	x <sup>Note 1</sup>	1	x	1	x	External clock	P70 (Input)	P71	P23/STB	RxD1	ASCK input
		1																						Internal clock
1	1	0	0	0	0	1	1	1	x	0	1	0	Note 2	1	x	1	x	External clock	P70 (Input)	High output	TxD1	RxD1	ASCK input	
		1																					Internal clock	P72
Other than above																		Setting prohibited						

**Notes 1.** Can be used freely as a port function.

**2.** The set value differs between when the actual device operates and when emulation is executed by the in-circuit emulator. For details, see **19.4.5 Restrictions in UART mode 2.**

**Remark** x: don't care  
 PMxx: Port mode register  
 Pxx: Port output latch

**(3) Asynchronous serial interface status register (ASIS)**

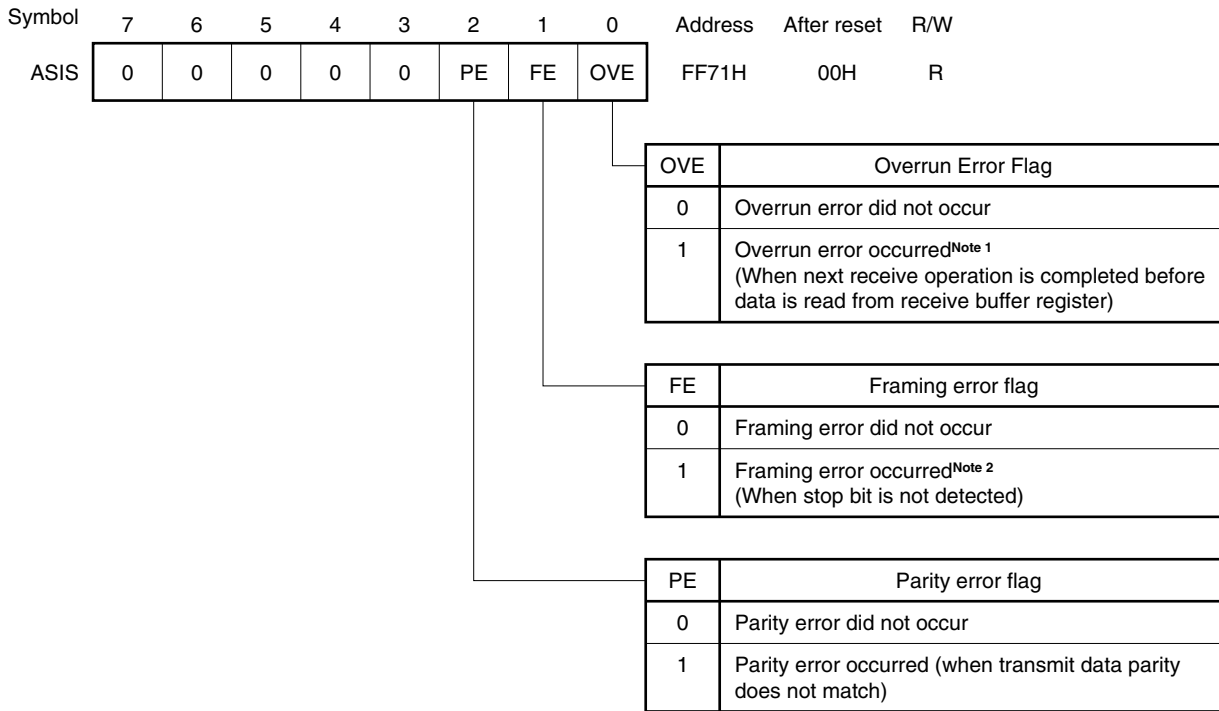
This is a register which displays the type of error when a reception error occurs in the asynchronous serial interface mode.

ASIS is read with a 1-bit or 8-bit memory manipulation instruction.

In 3-wire serial I/O mode, the contents of ASIS are undefined.

$\overline{\text{RESET}}$  input clears ASIS to 00H.

**Figure 19-5. Format of Asynchronous Serial Interface Status Register**



- Notes**
1. The receive buffer register (RXB) must be read when an overrun error occurs. Overrun errors will continue to occur until RXB is read.
  2. Even if the stop bit length has been set as 2 bits by bit 2 (SL) of the asynchronous serial interface mode register (ASIM), only single stop bit detection is performed during reception.

**(4) Baud rate generator control register (BRGC)**

This register sets the serial clock for serial interface channel 2.

BRGC is set with an 8-bit memory manipulation instruction.

RESET input clears BRGC to 00H.

**Figure 19-6. Format of Baud Rate Generator Control Register (1/2)**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
BRGC	TPS3	TPS2	TPS1	TPS0	MDL3	MDL2	MDL1	MDL0	FF73H	00H	R/W

MDL3	MDL2	MDL1	MDL0	Baud rate generator input clock selection	k
0	0	0	0	f <sub>sck</sub> /16	0
0	0	0	1	f <sub>sck</sub> /17	1
0	0	1	0	f <sub>sck</sub> /18	2
0	0	1	1	f <sub>sck</sub> /19	3
0	1	0	0	f <sub>sck</sub> /20	4
0	1	0	1	f <sub>sck</sub> /21	5
0	1	1	0	f <sub>sck</sub> /22	6
0	1	1	1	f <sub>sck</sub> /23	7
1	0	0	0	f <sub>sck</sub> /24	8
1	0	0	1	f <sub>sck</sub> /25	9
1	0	1	0	f <sub>sck</sub> /26	10
1	0	1	1	f <sub>sck</sub> /27	11
1	1	0	0	f <sub>sck</sub> /28	12
1	1	0	1	f <sub>sck</sub> /29	13
1	1	1	0	f <sub>sck</sub> /30	14
1	1	1	1	f <sub>sck</sub> <sup>Note</sup>	—

**Note** Can only be used in 3-wire serial I/O mode.

- Remarks**
1. f<sub>sck</sub>: 5-bit counter source clock
  2. k: Value set in MDL0 to MDL3 (0 ≤ k ≤ 14)

Figure 19-6. Format of Baud Rate Generator Control Register (2/2)

TPS3	TPS2	TPS1	TPS0	5-bit counter source clock selection				n
				MCS = 1		MCS = 0		
0	0	0	0	$f_{xx}/2^{10}$	$f_{xx}/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)		11
0	1	0	1	$f_{xx}$	$f_x$ (5.0 MHz)	$f_x/2$ (2.5 MHz)		1
0	1	1	0	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)		2
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)		3
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)		4
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)		5
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)		6
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)		7
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)		8
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)		9
1	1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)		10
Other than above				Setting prohibited				

**Caution** When BRGC is written during a communication operation, baud rate generator output is disrupted and communication cannot be performed normally. Therefore, BRGC must not be written during a communication operation.

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  3. MCS: Bit 0 of the oscillation mode select register (OSMS)
  4. n: Value set in TPS0 to TPS3 ( $1 \leq n \leq 11$ )
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz

The baud rate transmit/receive clock generated is either a signal divided from the main system clock, or a signal divided from the clock input from the ASCK pin.

**(a) Generation of baud rate transmit/receive clock from main system clock**

The transmit/receive clock is generated by dividing the main system clock. The baud rate generated from the main system clock is obtained from the following expression.

$$[\text{Baud rate}] = \frac{f_{xx}}{2^n \times (k + 16)} \text{ [Hz]}$$

- where,
- fx: Main system clock oscillation frequency
  - fx: Main system clock frequency (fx or fx/2)
  - n: Value set in TPS0 to TPS3 (1 ≤ n ≤ 11)
  - k: Value set in MDL0 to MDL3 (0 ≤ k ≤ 14)

**Table 19-3. Relationship Between Main System Clock and Baud Rate**

Baud Rate (bps)	fx = 5.0 MHz				fx = 4.19 MHz			
	MCS = 1		MCS = 0		MCS = 1		MCS = 0	
	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)
75	—		00H	1.73	0BH	1.14	EBH	1.14
110	06H	0.88	E6H	0.88	03H	-2.01	E3H	-2.01
150	00H	1.73	E0H	1.73	EBH	1.14	DBH	1.14
300	E0H	1.73	D0H	1.73	DBH	1.14	CBH	1.14
600	D0H	1.73	C0H	1.73	CBH	1.14	BBH	1.14
1,200	C0H	1.73	B0H	1.73	BBH	1.14	ABH	1.14
2,400	B0H	1.73	A0H	1.73	ABH	1.14	9BH	1.14
4,800	A0H	1.73	90H	1.73	9BH	1.14	8BH	1.14
9,600	90H	1.73	80H	1.73	8BH	1.14	7BH	1.14
19,200	80H	1.73	70H	1.73	7BH	1.14	6BH	1.14
31,250	74H	0	64H	0	71H	-1.31	61H	-1.31
38,400	70H	1.73	60H	1.73	6BH	1.14	5BH	1.14
76,800	60H	1.73	50H	1.73	5BH	1.14	—	—

**Remark** MCS: Bit 0 of the oscillation mode select register (OSMS)

**(b) Generation of baud rate transmit/receive clock from external clock input from ASCK pin**

The transmit/receive clock is generated by dividing the clock input from the ASCK pin. The baud rate generated from the clock input from the ASCK pin is obtained from the following expression.

$$[\text{Baud rate}] = \frac{f_{\text{ASCK}}}{2 \times (k + 16)} \text{ [Hz]}$$

$f_{\text{ASCK}}$ : Frequency of clock input to ASCK pin

k: Value set in MDL0 to MDL3 ( $0 \leq k \leq 14$ )

**Table 19-4. Relationship Between ASCK Pin Input Frequency and Baud Rate (When BRGC Is Set to 00H)**

Baud Rate (bps)	ASCK Pin Input Frequency
75	2.4 kHz
110	3.52 kHz
150	4.8 kHz
300	9.6 kHz
600	19.2 kHz
1,200	38.4 kHz
2,400	76.8 kHz
4,800	153.6 kHz
9,600	307.2 kHz
19,200	614.4 kHz
31,250	1,000.0 kHz
38,400	1,228.8 kHz



**(5) Serial interface pin select register (SIPS)**

This register selects the input/output pins when serial interface channel 2 is used in the asynchronous serial interface mode (with time-division transfer function).

SIPS is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SIPS to 00H.

To select the input/output pins, the port mode register and the output latch of the port must be set. For details, see **Table 19-2 Operating Mode Settings of Serial Interface Channel 2**.

**Figure 19-7. Format of Serial Interface Pin Select Register**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
SIPS	0	0	SIPS21	SIPS20	0	0	0	0	FF75H	00H	R/W

SIPS21	SIPS20	Selection input/output pin of asynchronous serial interface
0	0	Input pin: RxD0/SI2/P70 Output pin: TxD0/SO2/P71
0	1	Input pin: RxD1/BUSY/P24 Output pin: TxD0/SO2/P71
1	0	Input pin: RxD0/SI2/P70 Output pin: TxD1/STB/P23
1	1	Input pin: RxD1/BUSY/P24 Output pin: TxD1/STB/P23

- Cautions**
1. Select the input/output pins after stopping serial transmission/reception.
  2. When using the busy control option or busy & strobe control option in the 3-wire serial I/O mode with automatic transmit/receive function of serial interface channel 1, the RxD1/BUSY/P24 and TxD1/STB/P23 pins cannot be used as data I/O pins.
  3. SIPS21 is valid only when the TXE flag is “1” and SIPS20 is valid only when the RXE flag is “1”.
  4. There are restrictions when SIPS21 = 1 (when the TxD1 pin is used as an output pin for UART transmission). For details, see 19.4.5 Restrictions in UART mode 2.

## 19.4 Operation of Serial Interface Channel 2

The following three operating modes are available for serial interface channel 2.

- Operation stop mode
- Asynchronous serial interface (UART) mode (with time-division transfer function)
- 3-wire serial I/O mode

### 19.4.1 Operation stop mode

In the operation stop mode, serial transfer is not performed, and therefore power consumption can be reduced.

In the operation stop mode, the P70/SI2/RxD0, P71/SO2/TxD0, and P72/ $\overline{\text{SCK2}}$ /ASCK pins can be used as normal I/O ports and the P23/STB/TxD1, P24/BUSY/RxD1 pins can be used as normal I/O ports or as the strobe output and busy input for serial interface automatic transmit/receive.

#### (1) Register setting

Operation stop mode is set by serial operating mode register 2 (CSIM2) and the asynchronous serial interface mode register (ASIM).

##### (a) Serial operating mode register 2 (CSIM2)

CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM2 to 00H.

Symbol	<7>	6	5	4	3	2	1	0	Address	After reset	R/W
CSIM2	CSIE2	0	0	0	0	CSIM 22	CSCK	0	FF72H	00H	R/W

CSIE2	Operation control in 3-wire serial I/O mode
0	Operation stopped
1	Operation enabled

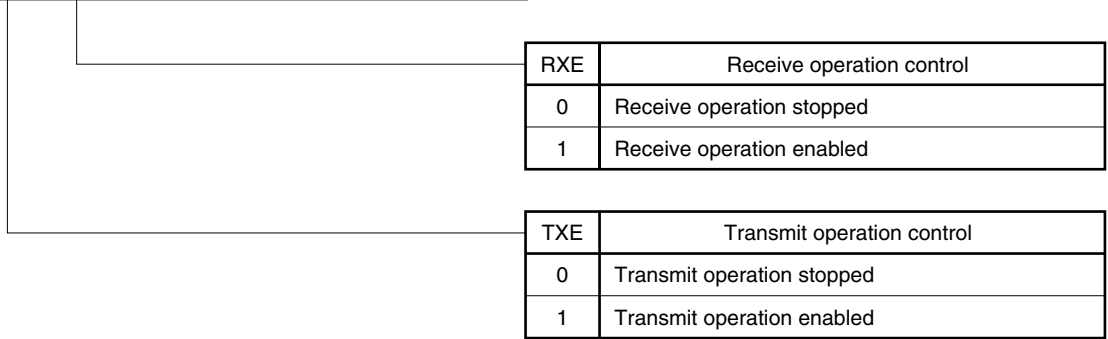
**Caution** Be sure to clear bits 0 and 3 to 6 to 0.

**(b) Asynchronous serial interface mode register (ASIM)**

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ASIM to 00H.

Symbol	<7>	<6>	5	4	3	2	1	0	Address	After reset	R/W
ASIM	TXE	RXE	PS1	PS0	CL	SL	ISRM	SCK	FF70H	00H	R/W



**19.4.2 Asynchronous serial interface (UART) mode (with time-division transfer function)**

In this mode, one byte of data is transmitted/received following the start bit, and full-duplex operation is possible. A dedicated UART baud rate generator is incorporated, allowing communication over a wide range of baud rates. In addition, the baud rate can be defined by dividing the clock input to the ASCK pin.

The MIDI standard baud rate (31.25 kbps) can be used by employing the dedicated UART baud rate generator. Two sets of data I/O pins (RxD and TxD) are provided, and the pin to be used can be selected by software (time-division transfer function). However, only one set of pins can be used at one time.

- Cautions**
1. If it is not necessary to change the data I/O pin, use of the RxD0/SI2/P70 and TxD0/SO2/P71 pins is recommended. If only port 2 (RxD1/BUSY/P24 and TxD1/STB/P23) is used as data I/O pins, the function of port 7 is limited.
  2. When using the busy control option or busy & strobe control option in the 3-wire serial I/O mode with automatic transmit/receive function of serial interface channel 1, the RxD1/BUSY/P24 and TxD1/STB/P23 pins cannot be used as data I/O pins.

**(1) Register setting**

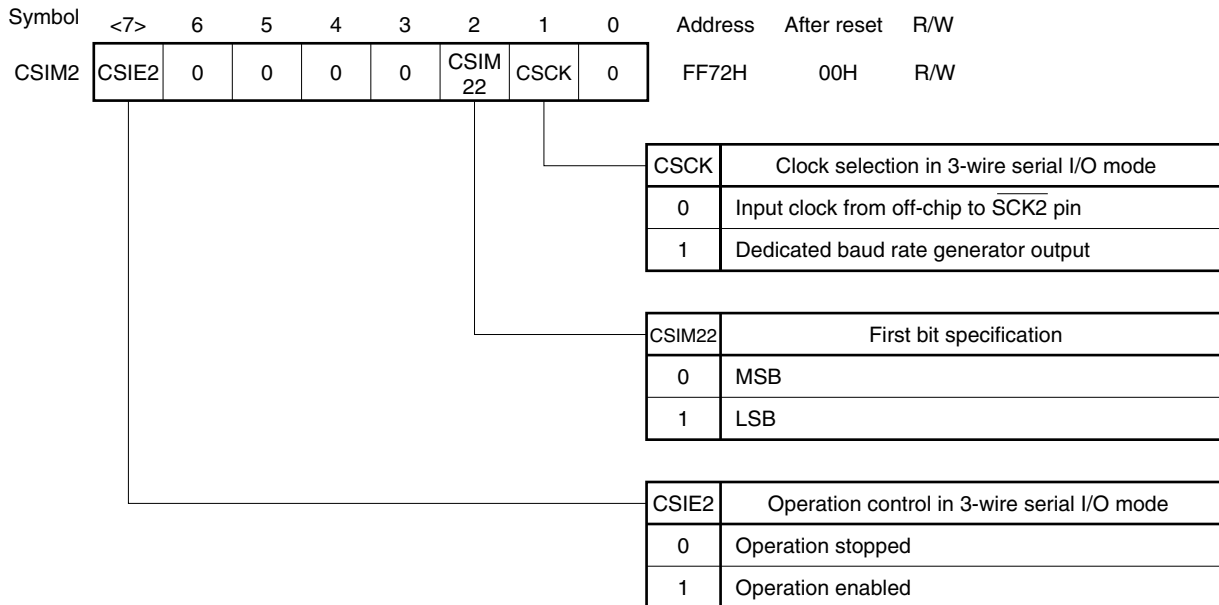
UART mode (with time-division transfer function) is set by serial operating mode register 2 (CSIM2), the asynchronous serial interface mode register (ASIM), the asynchronous serial interface status register (ASIS), the baud rate generator control register (BRGC), and the serial interface pin select register (SIPS).

**(a) Serial operating mode register 2 (CSIM2)**

CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM2 to 00H.

When the UART mode is selected, CSIM2 should be cleared to 00H.

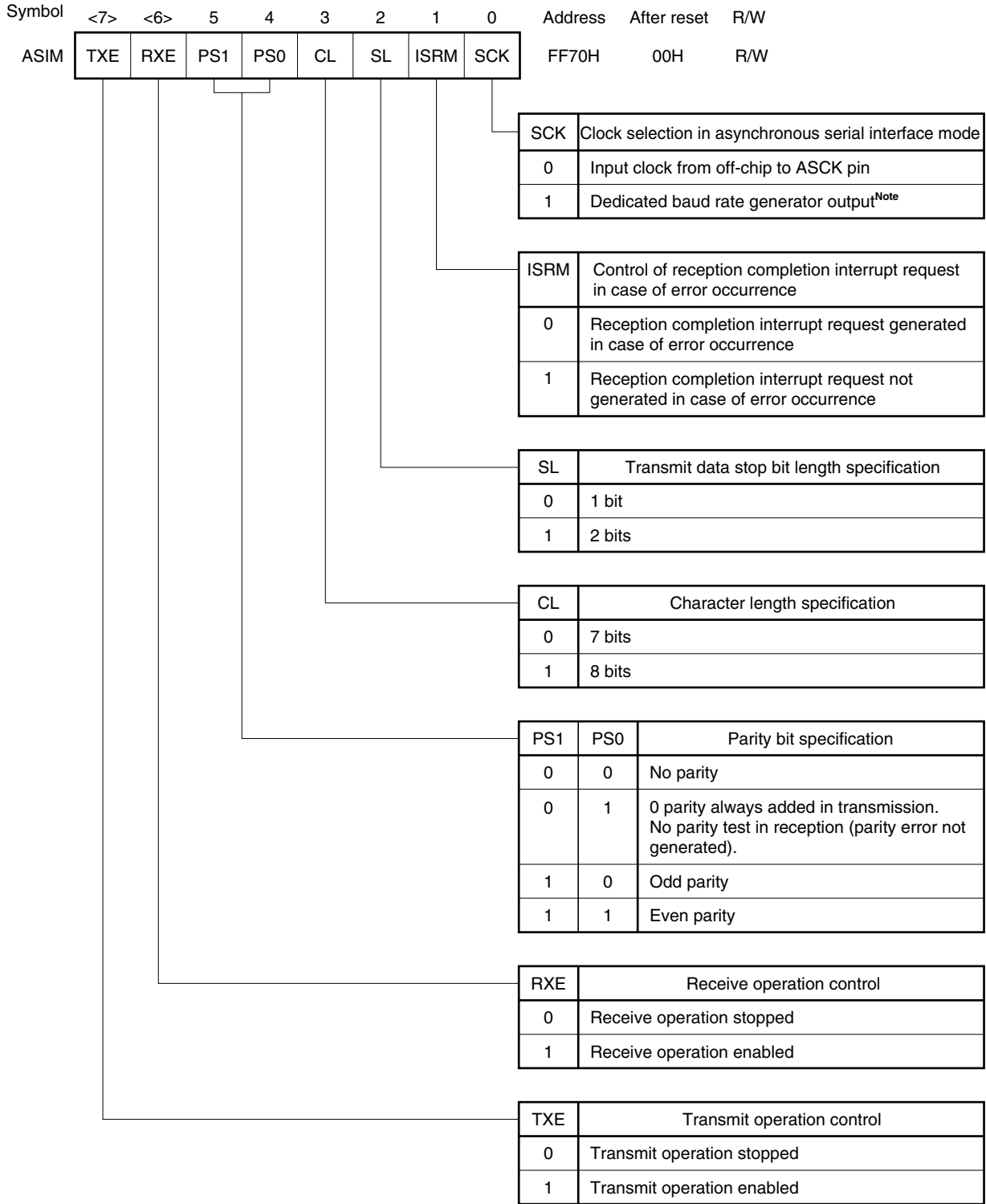


**Caution** Be sure to clear bits 0 and 3 to 6 to 0.

**(b) Asynchronous serial interface mode register (ASIM)**

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ASIM to 00H.



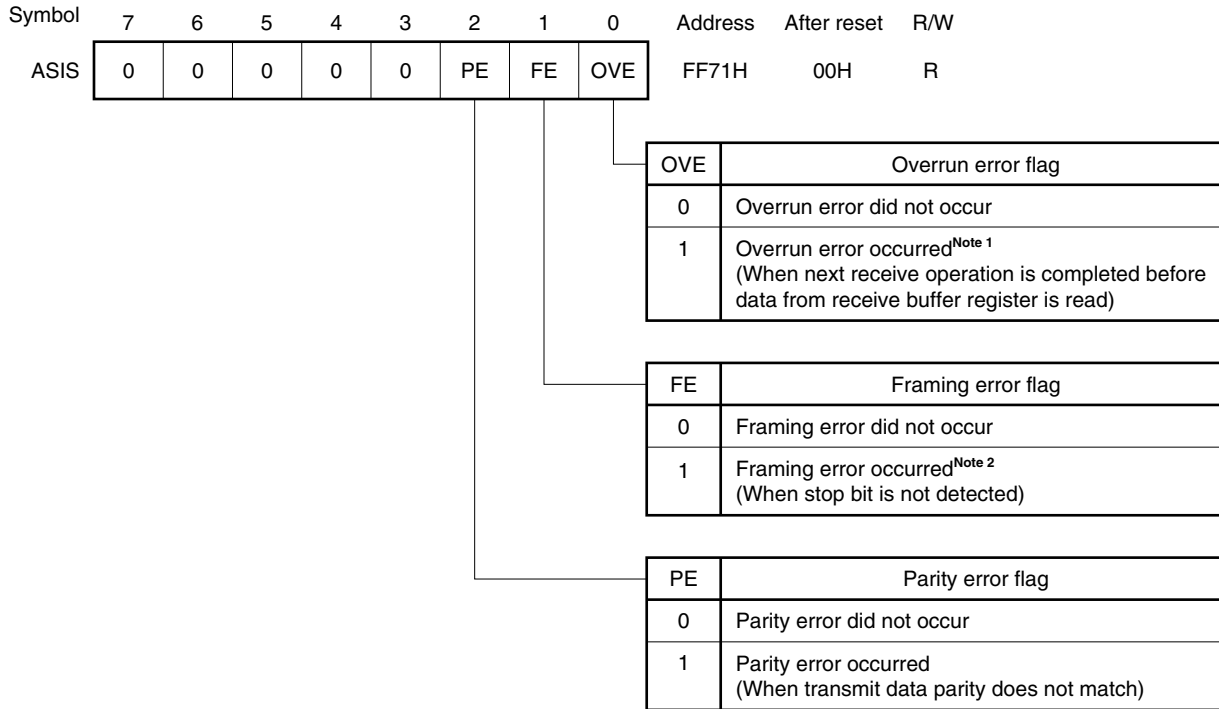
**Note** When SCK is set to 1 and the baud rate generator output is selected, the ASCK pin can be used as an I/O port.

**Caution** The serial transmit/receive operation must be stopped before changing the operating mode.

**(c) Asynchronous serial interface status register (ASIS)**

ASIS is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ASIS to 00H.



- Notes**
1. The receive buffer register (RXB) must be read when an overrun error occurs. Overrun errors will continue to occur until RXB is read.
  2. Even if the stop bit length has been set as 2 bits by bit 2 (SL) of the asynchronous serial interface mode register (ASIM), only single stop bit detection is performed during reception.

**(d) Baud rate generator control register (BRGC)**

BRGC is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears BRGC to 00H.

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
BRGC	TPS3	TPS2	TPS1	TPS0	MDL3	MDL2	MDL1	MDL0	FF73H	00H	R/W

MDL3	MDL2	MDL1	MDL0	Baud rate generator input clock selection	k
0	0	0	0	f <sub>sck</sub> /16	0
0	0	0	1	f <sub>sck</sub> /17	1
0	0	1	0	f <sub>sck</sub> /18	2
0	0	1	1	f <sub>sck</sub> /19	3
0	1	0	0	f <sub>sck</sub> /20	4
0	1	0	1	f <sub>sck</sub> /21	5
0	1	1	0	f <sub>sck</sub> /22	6
0	1	1	1	f <sub>sck</sub> /23	7
1	0	0	0	f <sub>sck</sub> /24	8
1	0	0	1	f <sub>sck</sub> /25	9
1	0	1	0	f <sub>sck</sub> /26	10
1	0	1	1	f <sub>sck</sub> /27	11
1	1	0	0	f <sub>sck</sub> /28	12
1	1	0	1	f <sub>sck</sub> /29	13
1	1	1	0	f <sub>sck</sub> /30	14

(Cont'd)

**Remark** f<sub>sck</sub>: 5-bit counter source clock

k: Value set in MDL0 to MDL3 (0 ≤ k ≤ 14)

TPS3	TPS2	TPS1	TPS0	5-Bit counter source clock selection				n
				MCS = 1		MCS = 0		
0	0	0	0	$f_{xx}/2^{10}$	$f_x/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)		11
0	1	0	1	$f_{xx}$	$f_x$ (5.0 MHz)	$f_x/2$ (2.5 MHz)		1
0	1	1	0	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)		2
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)		3
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)		4
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)		5
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)		6
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)		7
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)		8
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)		9
1	1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)		10
Other than above				Setting prohibited				

**Caution** When BRGC is written during a communication operation, baud rate generator output is disrupted and communication cannot be performed normally. Therefore, BRGC must not be written to during a communication operation.

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  3. MCS: Bit 0 of the oscillation mode select register (OSMS)
  4. n: Value set in TPS0 to TPS3 ( $1 \leq n \leq 11$ )
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.



The baud rate transmit/receive clock generated is either a signal divided from the main system clock, or a signal divided from the clock input from the ASCK pin.

**(i) Generation of baud rate transmit/receive clock from main system clock**

The transmit/receive clock is generated by dividing the main system clock. The baud rate generated from the main system clock is obtained from the following expression.

$$[\text{Baud rate}] = \frac{f_{xx}}{2^n \times (k + 16)} \text{ [Hz]}$$

- where,
- fx: Main system clock oscillation frequency
  - fx: Main system clock frequency (fx or fx/2)
  - n: Value set in TPS0 to TPS3 (1 ≤ n ≤ 11)
  - k: Value set in MDL0 to MDL3 (0 ≤ k ≤ 14)

**Table 19-5. Relationship Between Main System Clock and Baud Rate**

Baud Rate (bps)	fx = 5.0 MHz				fx = 4.19 MHz			
	MCS = 1		MCS = 0		MCS = 1		MCS = 0	
	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)
75	-		00H	1.73	0BH	1.14	EBH	1.14
110	06H	0.88	E6H	0.88	03H	-2.01	E3H	-2.01
150	00H	1.73	E0H	1.73	EBH	1.14	DBH	1.14
300	E0H	1.73	D0H	1.73	DBH	1.14	CBH	1.14
600	D0H	1.73	C0H	1.73	CBH	1.14	BBH	1.14
1,200	C0H	1.73	B0H	1.73	BBH	1.14	ABH	1.14
2,400	B0H	1.73	A0H	1.73	ABH	1.14	9BH	1.14
4,800	A0H	1.73	90H	1.73	9BH	1.14	8BH	1.14
9,600	90H	1.73	80H	1.73	8BH	1.14	7BH	1.14
19,200	80H	1.73	70H	1.73	7BH	1.14	6BH	1.14
31,250	74H	0	64H	0	71H	-1.31	61H	-1.31
38,400	70H	1.73	60H	1.73	6BH	1.14	5BH	1.14
76,800	60H	1.73	50H	1.73	5BH	1.14	—	—

**Remark** MCS: Bit 0 of the oscillation mode select register (OSMS)

**(ii) Generation of baud rate transmit/receive clock from external clock input from ASCK pin**

The transmit/receive clock is generated by dividing the clock input from the ASCK pin. The baud rate generated from the clock input from the ASCK pin is obtained from the following expression.

$$[\text{Baud rate}] = \frac{f_{\text{ASCK}}}{2 \times (k + 16)} \text{ [Hz]}$$

where,  $f_{\text{ASCK}}$ : Frequency of clock input to ASCK pin  
 k: Value set in MDL0 to MDL3 ( $0 \leq k \leq 14$ )

**Table 19-6. Relationship Between ASCK Pin Input Frequency and Baud Rate (When BRGC Is Set to 00H)**

Baud Rate (bps)	ASCK Pin Input Frequency
75	2.4 kHz
110	3.52 kHz
150	4.8 kHz
300	9.6 kHz
600	19.2 kHz
1,200	38.4 kHz
2,400	76.8 kHz
4,800	153.6 kHz
9,600	307.2 kHz
19,200	614.4 kHz
31,250	1,000.0 kHz
38,400	1,228.8 kHz

**(e) Serial interface pin select register (SIPS)**

SIPS is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears SIPS to 00H.

To select the input/output pins, the port mode register and the output latch of the port must be set. For details, see **Table 19-2 Operating Mode Settings of Serial Interface Channel 2.**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
SIPS	0	0	SIPS21	SIPS20	0	0	0	0	FF75H	00H	R/W

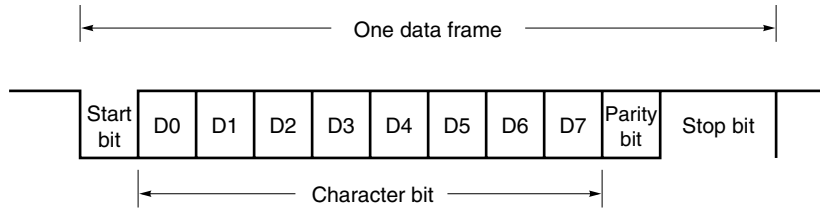
SIPS21	SIPS20	Selection of input/output pin of asynchronous serial interface
0	0	Input pin: RxD0/SI2/P70 Output pin: TxD0/SO2/P71
0	1	Input pin: RxD1/BUSY/P24 Output pin: TxD0/SO2/P71
1	0	Input pin: RxD0/SI2/P70 Output pin: TxD1/STB/P23
1	1	Input pin: RxD1/BUSY/P24 Output pin: TxD1/STB/P23

- Cautions**
1. Select the input/output pins after stopping serial transmission/reception.
  2. When using the busy control option or busy & strobe control option in the 3-wire serial I/O mode with automatic transmit/receive function of serial interface channel 1, the RxD1/BUSY/P24 and TxD1/STB/P23 pins cannot be used as data I/O pins.
  3. SIPS21 is valid only when the TXE flag is “1” and SIPS20 is valid only when the RXE flag is “1”.
  4. There are restrictions when SIPS21 = 1 (when the TxD1 pin is used as an output pin for UART transmission). For details, see 19.4.5 Restrictions in UART mode 2.

**(2) Communication operation****(a) Data format**

The transmit/receive data format is as shown in Figure 19-8.

**Figure 19-8. Format of Asynchronous Serial Interface Transmit/Receive Data**



One data frame consists of the following bits:

- Start bits ..... 1 bit
- Character bits ..... 7 bits/8 bits
- Parity bits ..... Even parity/odd parity/0 parity/no parity
- Stop bits ..... 1 bit/2 bits

The specification of character bit length, parity selection, and specification of stop bit length for each data frame is carried out by the asynchronous serial interface mode register (ASIM).

When 7 bits are selected as the number of character bits, only the lower 7 bits (bits 0 to 6) are valid; in transmission the most significant bit (bit 7) is ignored, and in reception the most significant bit (bit 7) is always "0".

The serial transfer rate is selected by means of ASIM and the baud rate generator control register (BRGC). If a serial data receive error occurs, the receive error contents can be determined by reading the status of the asynchronous serial interface status register (ASIS).

**(b) Parity types and operation**

The parity bit is used to detect a bit error in the communication data. Normally, the same kind of parity bit is used on the transmitting side and the receiving side. With even parity and odd parity, a 1-bit (odd number) error can be detected. With 0 parity and no parity, an error cannot be detected.

**(i) Even parity****• Transmission**

The number of bits with a value of “1”, including the parity bit, in the transmit data is controlled to be even.

The value of the parity bit is as follows:

Number of bits with a value of “1” in transmit data is odd: 1

Number of bits with a value of “1” in transmit data is even: 0

**• Reception**

The number of bits with a value of “1”, including the parity bit, in the receive data is counted. If it is odd, a parity error occurs.

**(ii) Odd parity****• Transmission**

Conversely to the situation with even parity, the number of bits with a value of “1”, including the parity bit, in the transmit data is controlled to be odd. The value of the parity bit is as follows:

Number of bits with a value of “1” in transmit data is odd: 0

Number of bits with a value of “1” in transmit data is even: 1

**• Reception**

The number of bits with a value of “1”, including the parity bit, in the receive data is counted. If it is even, a parity error occurs.

**(iii) 0 Parity**

When transmitting, the parity bit is set to “0” irrespective of the transmit data.

At reception, a parity bit check is not performed. Therefore, a parity error does not occur, irrespective of whether the parity bit is set to “0” or “1”.

**(iv) No parity**

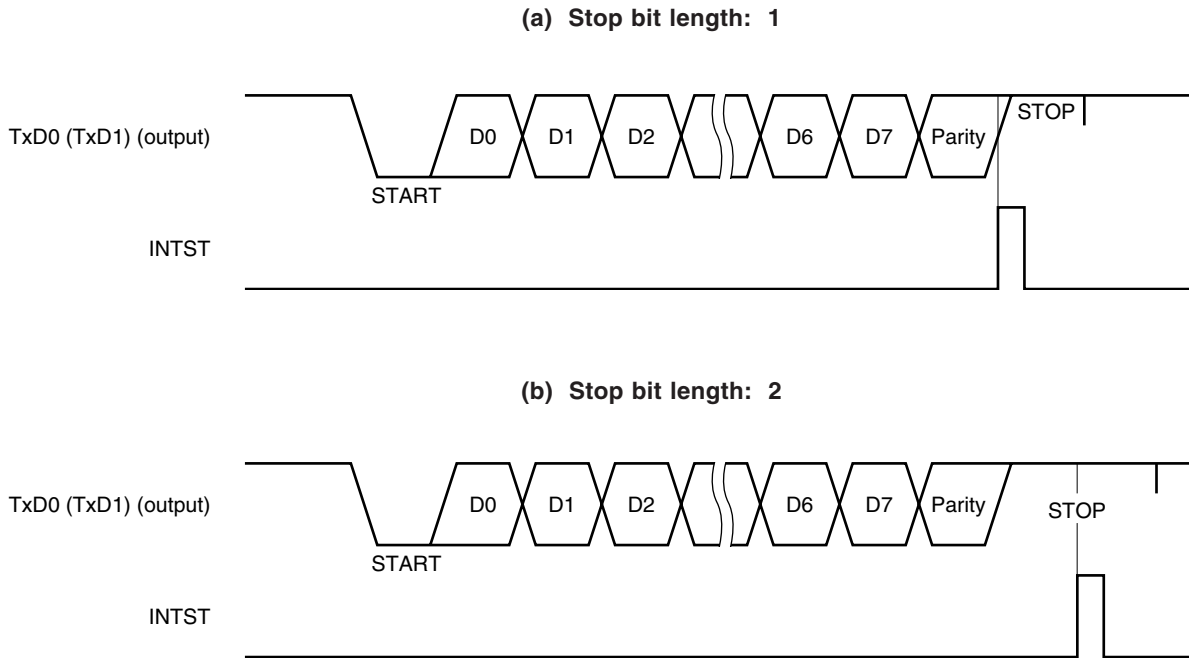
A parity bit is not added to the transmit data. At reception, data is received assuming that there is no parity bit. Since there is no parity bit, a parity error does not occur.

**(c) Transmission**

A transmit operation is started by writing transmit data to the transmit shift register (TXS). The start bit, parity bit and stop bit(s) are added automatically.

When the transmit operation starts, the data in the transmit shift register (TXS) is shifted out, and when the transmit shift register (TXS) is empty, a transmission completion interrupt request (INTST) is generated.

**Figure 19-9. Asynchronous Serial Interface Transmission Completion Interrupt Request Generation Timing**



**Caution** Rewriting the asynchronous serial interface mode register (ASIM) should not be performed during a transmit operation. If rewriting the ASIM is performed during transmission, subsequent transmit operations may not be possible (the normal state is restored by  $\overline{\text{RESET}}$  input).

It is possible to determine whether transmission is in progress by software by using a transmission completion interrupt request (INTST) or the interrupt request flag (STIF) set by INTST.

**(d) Reception**

When the RXE bit of the asynchronous serial interface mode register (ASIM) is set to 1, a receive operation is enabled and sampling of the RxD0 (RxD1) pin input is started.

RxD0 (RxD1) pin input sampling is performed using the serial clock specified by ASIM.

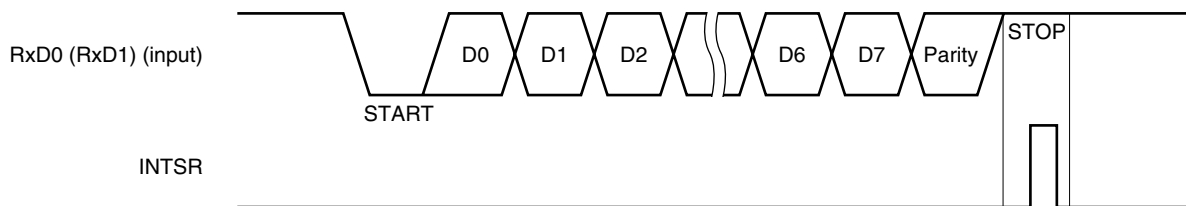
When the RxD0 (RxD1) pin input becomes low, the 5-bit counter of the baud rate generator (see **Figure 19-2**) starts counting, and at the time when the half time determined by specified baud rate has passed, the data sampling start timing signal is output. If the RxD0 (RxD1) pin input sampled again as a result of this start timing signal is low, it is identified as a start bit, the 5-bit counter is initialized and starts counting, and data sampling is performed. When character data, a parity bit, and one stop bit are detected after the start bit, reception of one frame of data ends.

When one frame of data has been received, the receive data in the shift register is transferred to the receive buffer register (RXB), and a reception completion interrupt request (INTSR) is generated.

If an error occurs, the receive data in which the error occurred is still transferred to RXB. If bit 1 (ISRM) of ASIM is cleared to 0 on occurrence of the error, INTSR is generated.

If the RXE bit is reset to 0 during the receive operation, the receive operation is stopped immediately. In this case, the contents of RXB and the asynchronous serial interface status register (ASIS) are not changed, and INTSR and INTSER are not generated.

**Figure 19-10. Asynchronous Serial Interface Reception Completion Interrupt Request Generation Timing**



**Caution** The receive buffer register (RXB) must be read even if a receive error occurs. If RXB is not read, an overrun error will occur when the next data is received, and the receive error state will continue indefinitely.

**(e) Receive errors**

Three kinds of errors can occur during a receive operation: a parity error, framing error, or overrun error. The data reception result error flag is set in the asynchronous serial interface status register (ASIS) and a receive error interrupt request (INTSER) is generated. The receive error interrupt is generated faster than receive completion interrupt (INTSR). Receive error causes are shown in Table 19-7.

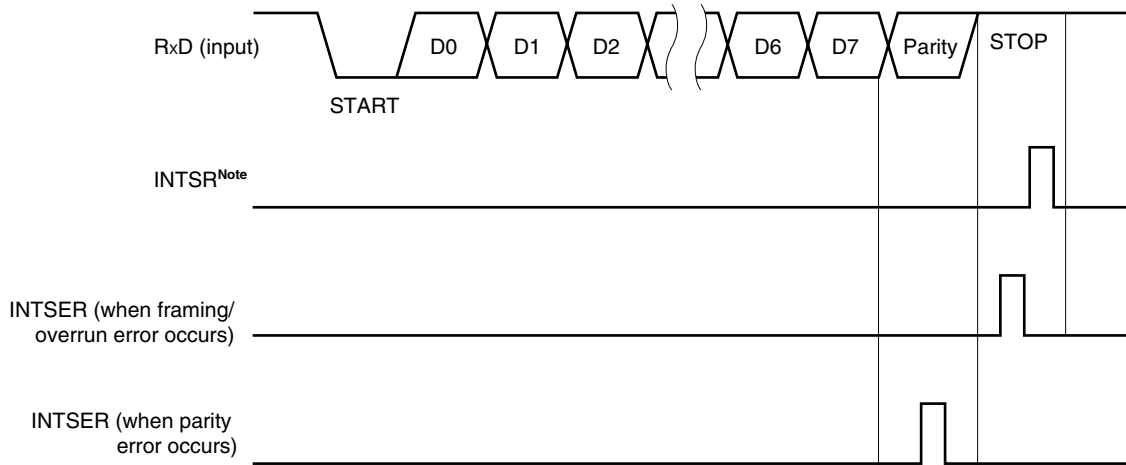
It is possible to determine what kind of error occurred during reception by reading the contents of ASIS in the reception error interrupt servicing (see **Figures 19-10** and **19-11**).

The contents of ASIS are reset to 0 by reading the receive buffer register (RXB) or receiving the next data (if there is an error in the next data, the corresponding error flag is set).

**Table 19-7. Receive Error Causes**

Receive Errors	Cause
Parity error	Transmission-time parity specification and reception data parity do not match
Framing error	Stop bit not detected
Overrun error	Reception of next data is completed before data is read from receive register buffer

**Figure 19-11. Receive Error Timing**



**Note** INTSR is not generated if a receive error occurs while bit 1 (ISRM) of the asynchronous serial interface mode register (ASIM) is set to 1.

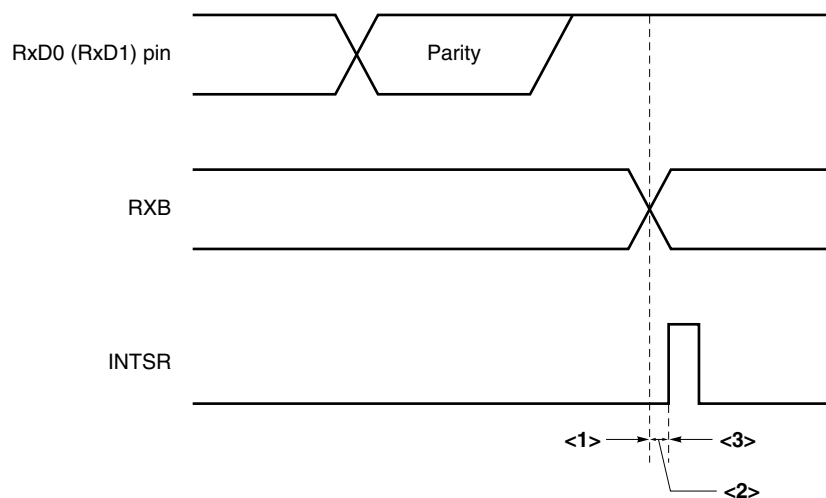
- Cautions**
1. The contents of the asynchronous serial interface status register (ASIS) are reset to 0 by reading the receive buffer register (RXB) or receiving the next data. To ascertain the error contents, ASIS must be read before reading RXB.
  2. The receive buffer register (RXB) must be read even if a receive error occurs. If RXB is not read, an overrun error will occur when the next data is received, and the receive error state will continue indefinitely.



**(3) UART mode cautions**

- (a) When the transmission under execution has been stopped by clearing bit 7 (TXE) of the asynchronous serial interface mode register (ASIM) to 0, be sure to set the transmit shift register (TXS) to FFH, then set TXE to 1 before executing the next transmission.
- (b) When the reception under execution has been stopped by clearing bit 6 (RXE) of the asynchronous serial interface mode register (ASIM) to 0, the status of the receive buffer register (RXB) and whether the receive completion interrupt request (INTSR) is generated differ depending on the timing at which reception is stopped. Figure 19-12 shows the timing.

**Figure 19-12. Status of Receive Buffer Register (RXB) and Generation of Interrupt Request (INTSR) When Reception Is Stopped**



When RXE is cleared to 0 at the time indicated by <1>, RXB holds the previous data and does not generate INTSR.

When RXE is cleared to 0 at the time indicated by <2>, RXB renews the data and does not generate INTSR.

When RXE is cleared to 0 at the time indicated by <3>, RXB renews the data and generates INTSR.

**19.4.3 3-wire serial I/O mode**

The 3-wire serial I/O mode is useful for connection of peripheral I/Os and display controllers, etc., which incorporate a conventional clocked serial interface, such as the 75X/XL Series, 78K Series, 17K Series, etc.

Communication is performed using three lines: the serial clock ( $\overline{\text{SCK2}}$ ), serial output (SO2), and serial input (SI2). In the 3-wire serial I/O mode, the P23/STB/TxD1, P24/BUSY/RxD1 pins can be used as normal I/O ports.

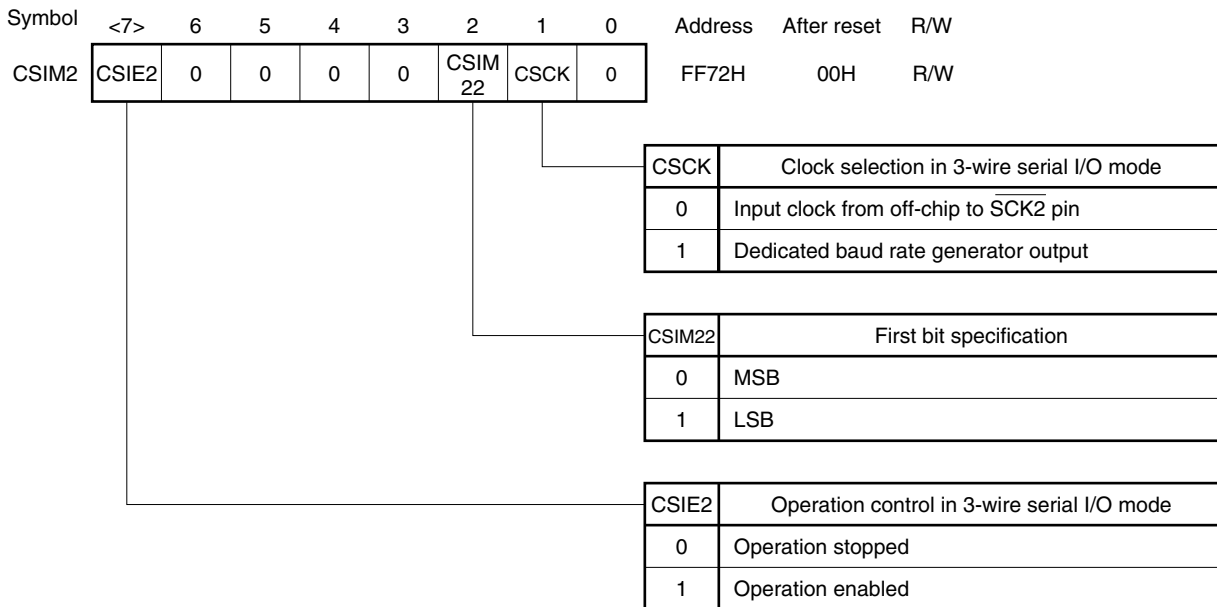
**(1) Register setting**

3-wire serial I/O mode is set by serial operating mode register 2 (CSIM2), the asynchronous serial interface mode register (ASIM), and the baud rate generator control register (BRGC).

**(a) Serial operating mode register 2 (CSIM2)**

CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CSIM2 to 00H.



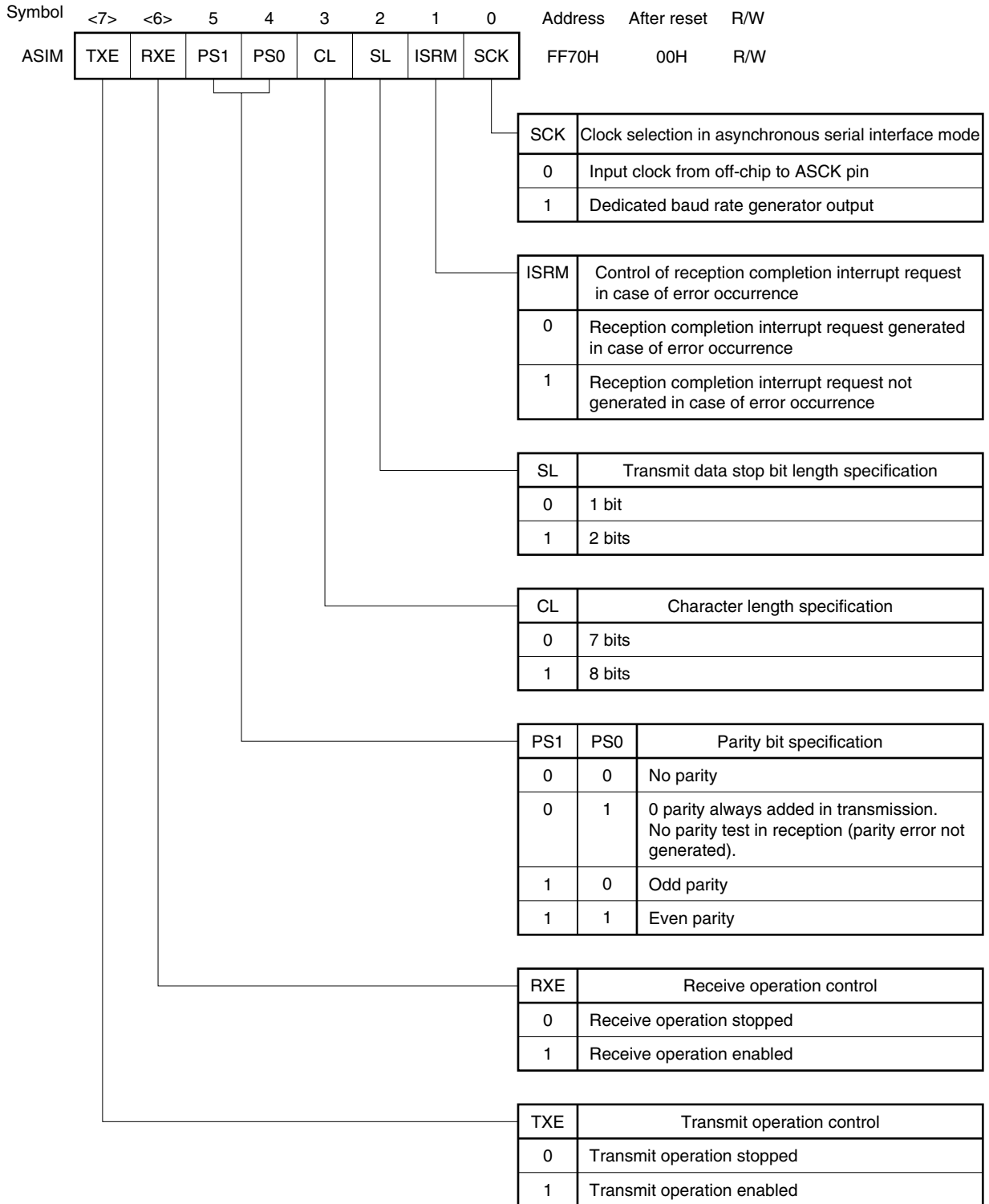
**Caution** Be sure to clear bits 0 and 3 to 6 to 0.

**(b) Asynchronous serial interface mode register (ASIM)**

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears ASIM to 00H.

When the 3-wire serial I/O mode is selected, ASIM should be cleared to 00H.



**(c) Baud rate generator control register (BRGC)**

BRGC is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears BRGC to 00H.

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
BRGC	TPS3	TPS2	TPS1	TPS0	MDL3	MDL2	MDL1	MDL0	FF73H	00H	R/W

MDL3	MDL2	MDL1	MDL0	Baud rate generator input clock selection	k
0	0	0	0	f <sub>sck</sub> /16	0
0	0	0	1	f <sub>sck</sub> /17	1
0	0	1	0	f <sub>sck</sub> /18	2
0	0	1	1	f <sub>sck</sub> /19	3
0	1	0	0	f <sub>sck</sub> /20	4
0	1	0	1	f <sub>sck</sub> /21	5
0	1	1	0	f <sub>sck</sub> /22	6
0	1	1	1	f <sub>sck</sub> /23	7
1	0	0	0	f <sub>sck</sub> /24	8
1	0	0	1	f <sub>sck</sub> /25	9
1	0	1	0	f <sub>sck</sub> /26	10
1	0	1	1	f <sub>sck</sub> /27	11
1	1	0	0	f <sub>sck</sub> /28	12
1	1	0	1	f <sub>sck</sub> /29	13
1	1	1	0	f <sub>sck</sub> /30	14
1	1	1	1	f <sub>sck</sub>	—

(Cont'd)

**Remark** f<sub>sck</sub>: 5-bit counter source clock  
 k: Value set in MDL0 to MDL3 (0 ≤ k ≤ 14)

TPS3	TPS2	TPS1	TPS0	5-bit counter source clock selection				n
				MCS = 1		MCS = 0		
0	0	0	0	$f_{xx}/2^{10}$	$f_x/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)	11	
0	1	0	1	$f_{xx}$	$f_x$ (5.0 MHz)	$f_x/2$ (2.5 MHz)	1	
0	1	1	0	$f_{xx}/2$	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)	2	
0	1	1	1	$f_{xx}/2^2$	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)	3	
1	0	0	0	$f_{xx}/2^3$	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)	4	
1	0	0	1	$f_{xx}/2^4$	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)	5	
1	0	1	0	$f_{xx}/2^5$	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)	6	
1	0	1	1	$f_{xx}/2^6$	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)	7	
1	1	0	0	$f_{xx}/2^7$	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)	8	
1	1	0	1	$f_{xx}/2^8$	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)	9	
1	1	1	0	$f_{xx}/2^9$	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)	10	
Other than above				Setting prohibited				

**Caution** When BRGC is written during a communication operation, baud rate generator output is disrupted and communication cannot be performed normally. Therefore, BRGC must not be written during a communication operation.

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  3. MCS: Bit 0 of the oscillation mode select register (OSMS)
  4. n: Value set in TPS0 to TPS3 ( $1 \leq n \leq 11$ )
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

When the internal clock is used as the serial clock in the 3-wire serial I/O mode, set BRGC as described below. BRGC setting is not required if an external serial clock is used.

**(i) When the baud rate generator is not used:**

Select the serial clock frequency using TPS0 to TPS3. Be sure then to set MDL0 to MDL3 to 1,1,1,1. The serial clock frequency becomes the same as the source clock frequency for the 5-bit counter.

**(ii) When the baud rate generator is used:**

Select the serial clock frequency using TPS0 to TPS3. Be sure then to set MDL0 to MDL3 to 1,1,1,1.

The serial clock frequency is calculated by the following formula:

$$\text{Serial clock frequency} = \frac{f_{xx}}{2^n \times (k + 16)} \text{ [Hz]}$$

- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  3.  $n$ : Value set in TPS0 to TPS3 ( $1 \leq n \leq 11$ )
  4.  $k$ : Value set in MDL0 to MDL3 ( $0 \leq k \leq 14$ )

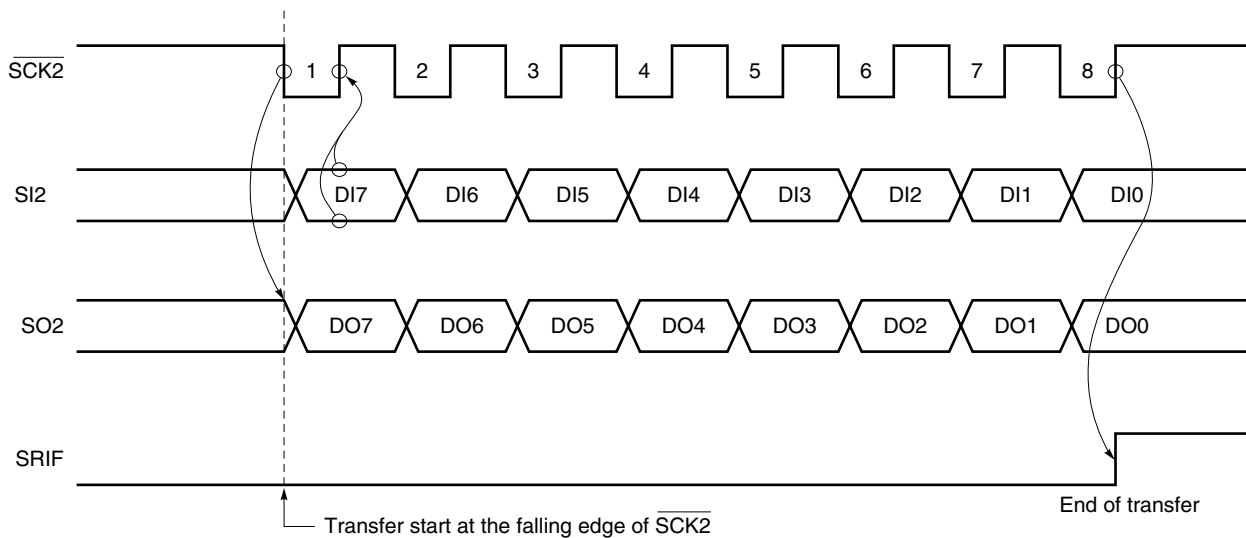
**(2) Communication operation**

In the 3-wire serial I/O mode, data transmission/reception is performed in 8-bit units. Data is transmitted/received bit-wise synchronization with the serial clock.

Transmit shift register (TXS/SIO2) and receive shift register (RXS) shift operations are performed in synchronization with the fall of the serial clock  $\overline{SCK2}$ . Then transmit data is held in the SO2 latch and output from the SO2 pin. Also, receive data input to the SI2 pin is latched in the receive buffer register (RXB/SIO2) on the rise of  $\overline{SCK2}$ .

At the end of an 8-bit transfer, the operation of TXS/SIO2 or RXS stops automatically, and the interrupt request flag (SRIF) is set.

**Figure 19-13. 3-Wire Serial I/O Mode Timing**

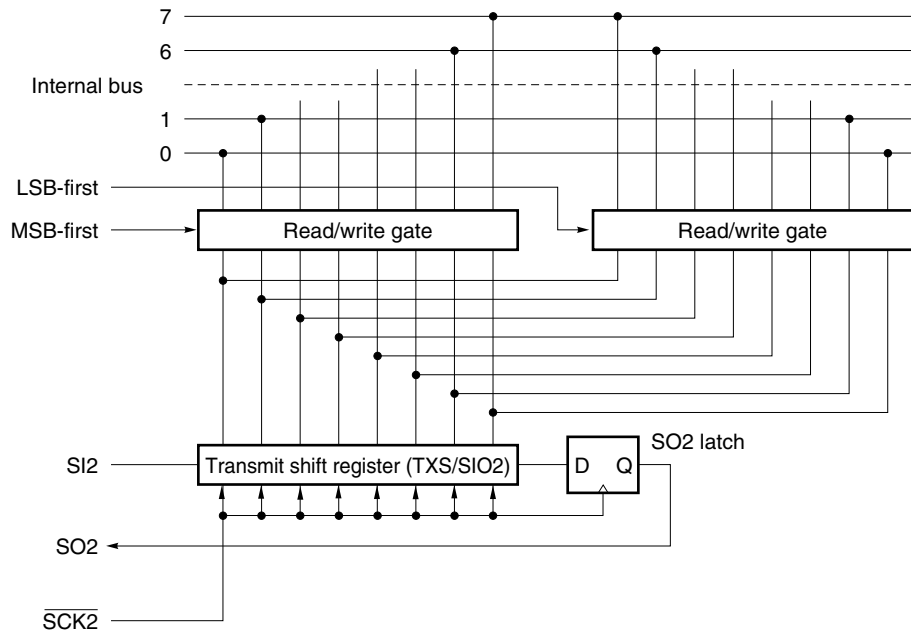
**(3) MSB/LSB switching as the start bit**

In the 3-wire serial I/O mode, it is possible to select transfer to start from the MSB or LSB.

Figure 19-14 shows the configuration of the transmit shift register (TXS/SIO2) and internal bus. As shown in the figure, the MSB/LSB can be read or written in reverse form.

MSB/LSB switching as the start bit can be specified by bit 2 (CSIM22) of serial operating mode register 2 (CSIM2).

Figure 19-14. Circuit for Switching Transfer Bit Order



Start bit switching is realized by switching the bit order for data write to SIO2. The SIO2 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to the shift register.

#### (4) Transfer start

Serial transfer is started by setting transfer data to the transmission shift register (TXS/SIO2) when the following two conditions are satisfied.

- Serial interface channel 2 operation control bit (CSIE2) = 1
- Internal serial clock is stopped or SCK2 is a high level after 8-bit serial transfer.

**Caution** If CSIE2 is set to 1 after data write to TXS/SIO2, transfer does not start.

**Remark** CSIE2: Bit 7 of serial operating mode register 2 (CSIM2)

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (SRIF) is set.



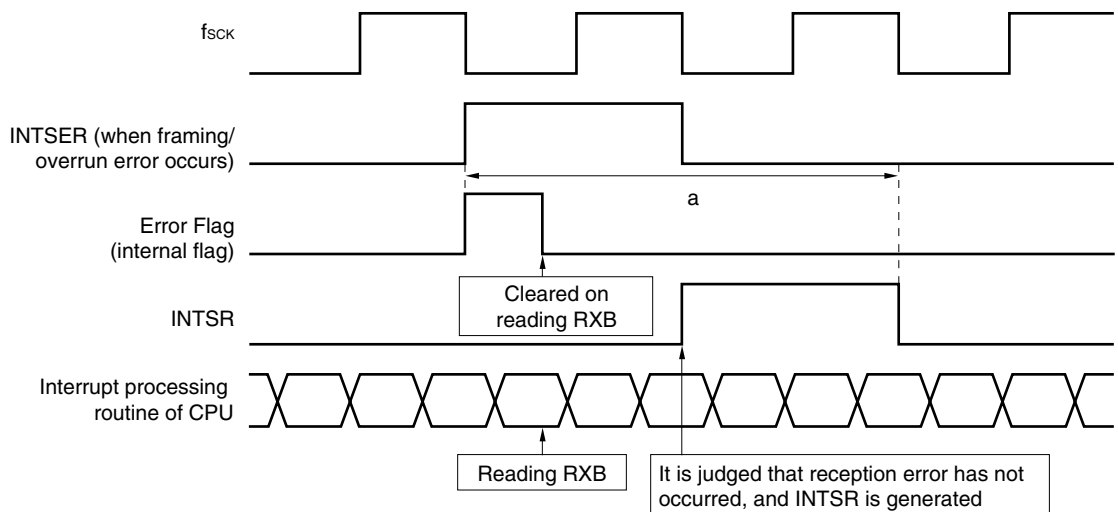
#### 19.4.4 Restrictions in UART mode 1

In the UART mode, the reception completion interrupt request (INTSR) is generated a certain time after the reception error interrupt (INTSER) is generated and then cleared. Consequently, the following phenomenon may occur.

- **Description**

If bit 1 (ISRM) of the asynchronous serial interface mode register (ASIM) is set to 1, the reception completion interrupt request (INTSR) is not generated on occurrence of a reception error. If the receive buffer register (RXB) is read at certain timing ("a" in Figure 19-15) during the reception error interrupt (INTSER) servicing, the internal error flag is cleared to 0. As a result, it is judged that no reception error has occurred, and INTSR, which should not be generated, is generated. Figure 19-15 illustrates this operation.

**Figure 19-15. Reception Completion Interrupt Request Generation Timing (When ISRM = 1)**



**Remark** ISRM: Bit 1 of the asynchronous serial interface mode register (ASIM)  
 f<sub>sck</sub>: Source clock of 5-bit counter of baud rate generator  
 RXB: Receive buffer register

To avoid this phenomenon, take the following measures.

- **Preventive measures**

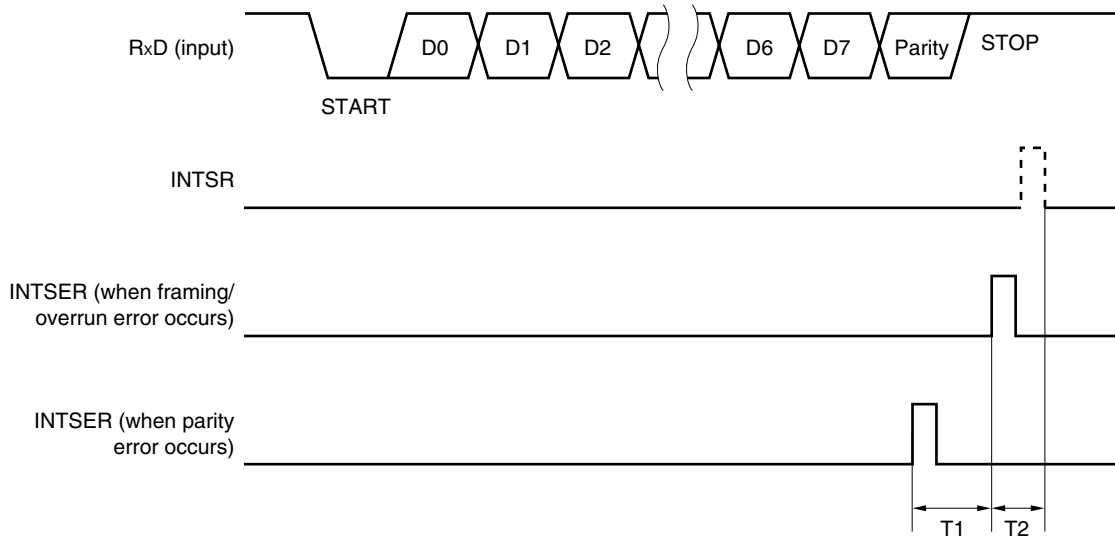
- **In case of framing error or overrun error**

Disable the receive buffer register (RXB) from being read for a certain period (T<sub>2</sub> in Figure 19-16) after the reception error interrupt request (INTSER) has been generated.

- **In case of parity error**

Disable the receive buffer register (RXB) from being read for a certain period (T<sub>1</sub> + T<sub>2</sub> in Figure 19-16) after the reception error interrupt request (INTSER) has been generated.

Figure 19-16. Receive Buffer Register Read Disable Period



T1: Time of one data of baud rate selected by baud rate generator control register (BRGC) (1/baud rate)

T2: Time of 2 clocks of source clock ( $f_{sck}$ ) of 5-bit counter selected by BRGC

• **Example of preventive measures**

Here is an example of the above preventive measures.

**[Conditions]**

$f_x = 5.0 \text{ MHz}$

Processor clock control register (PCC) = 00H

Oscillation mode select register (OSMS) = 01H

Baud rate generator control register (BRGC) = B0H (2,400 bps selected as baud rate)

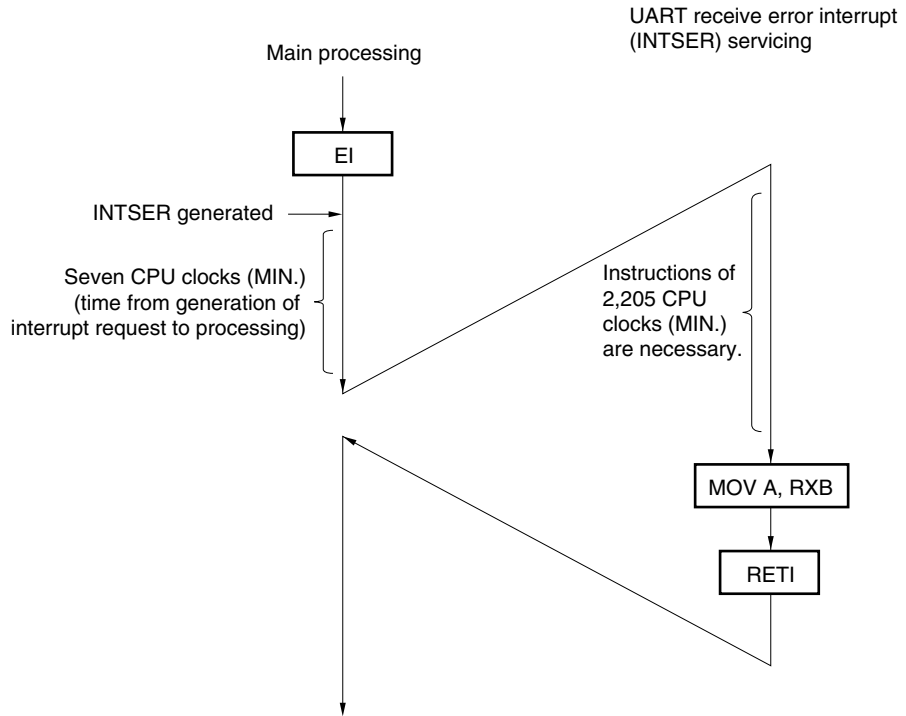
$T_{CY} = 0.4 \mu\text{s}$  ( $t_{CY} = 0.2 \mu\text{s}$ )

$$T1 = \frac{1}{2,400} = 416.7 \mu\text{s}$$

$$T2 = 12.8 \times 2 = 25.6 \mu\text{s}$$

$$\frac{T1 + T2}{t_{CY}} = 2,212 \text{ (clocks)}$$

[Example]



★ 19.4.5 Restrictions in UART mode 2

To use the TxD1/STB/P23 pin to output UART data by using the time-division transfer function, perform the following processing when the transmit operation is enabled and when the transmit/receive operation is stopped.

The output circuit of the alternate function of this TxD1/STB/P23 pin differs between the actual device and the in-circuit emulator (see Figure 19-17). Therefore, delete the underlined part in the examples below when executing emulation with the in-circuit emulator (IE).

Condition: Serial interface pin select register (SIPS) = 20H or 30H  
(when using TxD1 pin as output pin for UART transmission)

(1) When transmit operation is enabled

CLR1	PM2.3	; Sets P23 (TXD1) pin to output mode.
SET1	P2.3	; Sets output latch of P23 to "1".
SET1	ASIM.7	; Enables transmission (TXE = 1).
CLR1	P2.3	; This line is necessary only for the actual device. Delete it when the IE is used.
MOV	TXS, #BYTE	; Transfers transmit data (#BYTE) to transmit shift register.

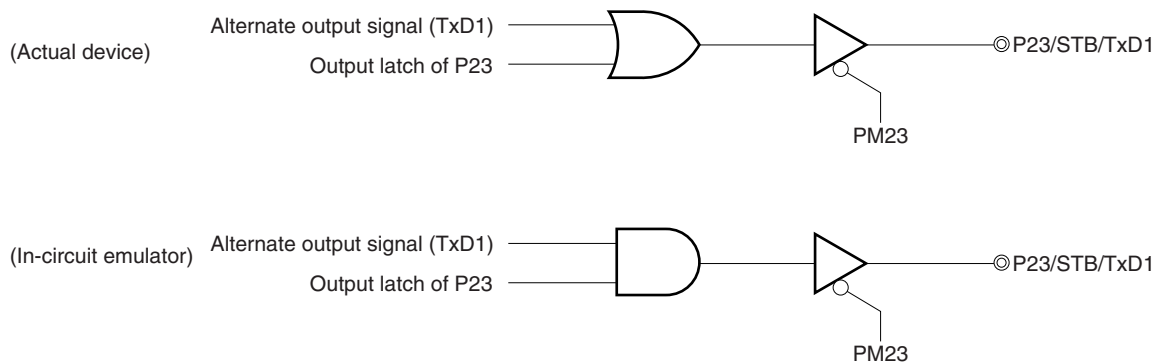
- Cautions**
1. Perform this processing each time a transmit operation is enabled by using the TxD1 pin as an output pin.
  2. Perform this processing each time the output pin is switched from the TxD0 pin to the TxD1 pin because the transmit operation must be stopped once and then enabled again.

(2) When transmit operation is stopped

SET1	P2.3	; This line is necessary only for the actual device. Delete it when the IE is used.
CLR1	ASIM.7	; Stops transmission (TXE = 0).

- Cautions**
1. Perform this processing each time a transmit operation is enabled by using the TxD1 pin as an output pin.
  2. Perform this processing each time the output pin is switched from the TxD0 pin to the TxD1 pin because the transmit operation must be stopped once and then enabled again.

Figure 19-17. P23 Output Selector



## CHAPTER 20 REAL-TIME OUTPUT PORT

### 20.1 Real-Time Output Port Functions

Data set previously in the real-time output buffer register can be transferred to the output latch by hardware concurrently with the generation of a timer interrupt request or external interrupt request, then output externally. This is called the real-time output function. The pins that output data externally are called real-time output ports.

By using a real-time output, a signal which has no jitter can be output. This port is therefore suitable for control of stepper motors, etc.

Port mode/real-time output port mode can be specified in 1-bit units.

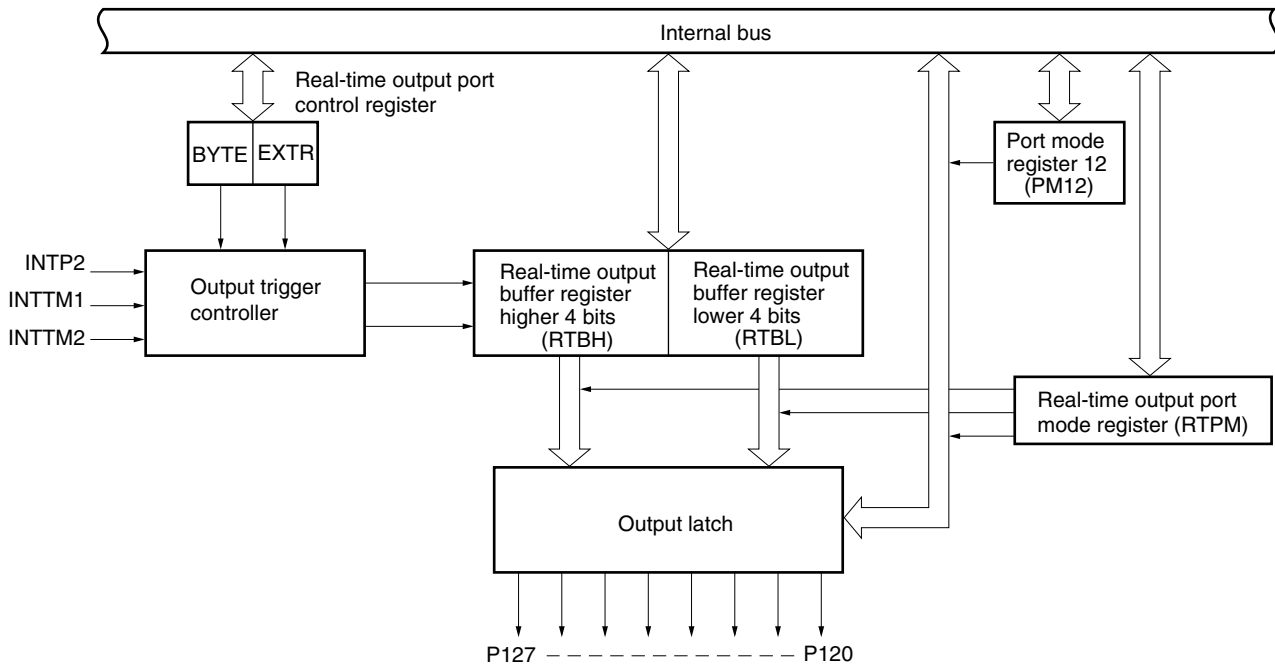
## 20.2 Real-Time Output Port Configuration

The real-time output port consists of the following hardware.

**Table 20-1. Real-Time Output Port Configuration**

Item	Configuration
Register	Real-time output buffer register (RTBL, RTBH)
Control registers	Port mode register 12 (PM12) Real-time output port mode register (RTPM) Real-time output port control register (RTPC)

**Figure 20-1. Real-Time Output Port Block Diagram**



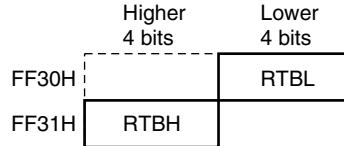
**(1) Real-time output buffer registers (RTBL, RTBH)**

The addresses of RTBL and RTBH are mapped individually in the special function register (SFR) area as shown in Figure 20-2.

When specifying 4 bits × 2 channels as the operating mode, data is set individually to RTBL and RTBH. When specifying 8 bits × 1 channel as the operating mode, data is set to both RTBL and RTBH by writing 8-bit data to either RTBL or RTBH.

Table 20-2 shows the operations during manipulation of RTBL and RTBH.

**Figure 20-2. Real-Time Output Buffer Register Configuration**



**Table 20-2. Operation in Real-Time Output Buffer Register Manipulation**

Operating Mode	Register to Be Manipulated	Reading <sup>Note 1</sup>		Writing <sup>Note 2</sup>	
		Higher 4 Bits	Lower 4 Bits	Higher 4 Bits	Lower 4 Bits
4 bits × 2 channels	RTBL	RTBH	RTBL	Invalid	RTBL
	RTBH	RTBH	RTBL	RTBH	Invalid
8 bits × 1 channel	RTBL	RTBH	RTBL	RTBH	RTBL
	RTBH	RTBH	RTBL	RTBH	RTBL

- Notes**
1. Only the bits set in the real-time output port mode can be read. When a bit set in the port mode is read, 0 is read.
  2. After setting data in the real-time output port, output data should be set to RTBL and RTBH by the time a real-time output trigger is generated.

### 20.3 Real-Time Output Port Control Registers

The following three registers control the real-time output port.

- Port mode register 12 (PM12)
- Real-time output port mode register (RTPM)
- Real-time output port control register (RTPC)

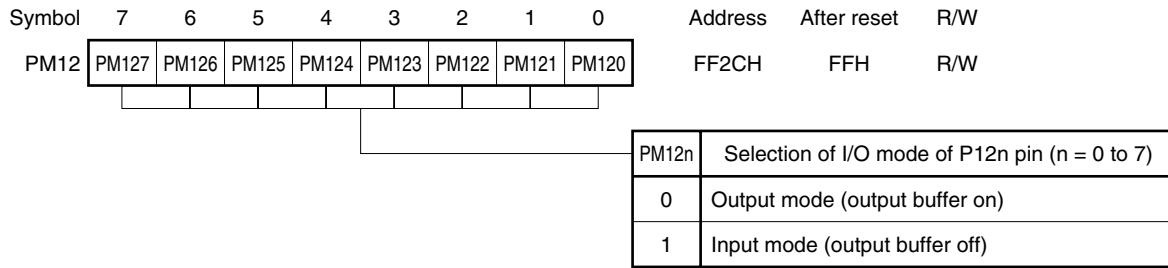
#### (1) Port mode register 12 (PM12)

This register sets the input or output mode of the port 12 pins (P120 to P127) which function alternately as real-time output pins (RTP0 to RTP7). To use port 12 as a real-time output port, the port pin that performs real-time output must be set in the output mode (PM12n = 0: n = 0 to 7).

PM12 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets PM12 to FFH.

Figure 20-3. Format of Port Mode Register 12



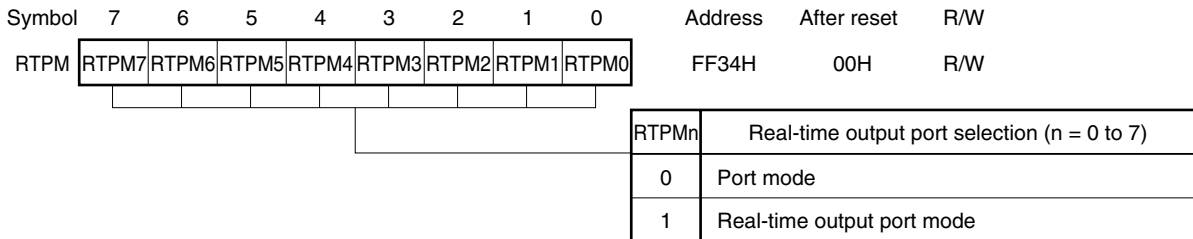
#### (2) Real-time output port mode register (RTPM)

This register selects the real-time output port mode/port mode in 1-bit units.

RTPM is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears RTPM to 00H.

Figure 20-4. Format of Real-Time Output Port Mode Register



- Cautions**
1. When using these bits as a real-time output port, set the ports at which real-time output is performed to the output mode (clear the corresponding bit of port mode register 12 (PM12) to 0).
  2. In ports specified as real-time output ports, data cannot be set to the output latch. Therefore, when setting an initial value, data should be set to the output latch before setting the real-time output mode.



**(3) Real-time output port control register (RTPC)**

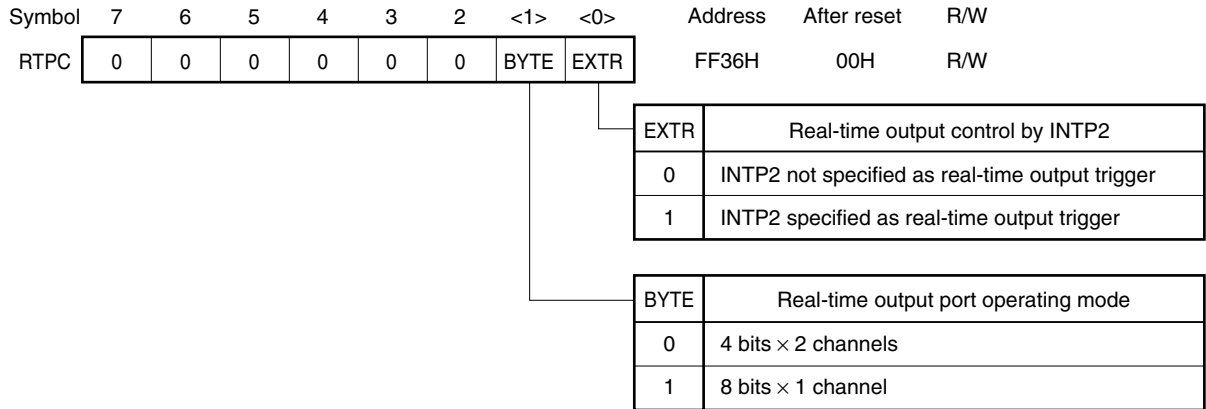
This register sets the real-time output port operating mode and output trigger.

Table 20-3 shows the relationship between the operating mode of the real-time output port and output trigger.

RTPC is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears RTPC to 00H.

**Figure 20-5. Format of Real-Time Output Port Control Register**



**Table 20-3. Real-Time Output Port Operating Mode and Output Trigger**

BYTE	EXTR	Operating Mode	RTBH → Port Output	RTBL → Port Output
0	0	4 bits × 2 channels	INTTM2	INTTM1
	1		INTTM1	INTP2
1	0	8 bits × 1 channel	INTTM1	
	1		INTP2	

## CHAPTER 21 INTERRUPT AND TEST FUNCTIONS

### 21.1 Interrupt Function Types

The following three types of interrupt functions are used.

#### (1) Non-maskable interrupt

This interrupt is acknowledged unconditionally even in the interrupt disabled status. It does not undergo interrupt priority control and is given top priority over all other interrupt requests.

It generates a standby release signal.

One interrupt source from the watchdog timer is provided as a non-maskable interrupt source.

#### (2) 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 (P0L, P0H, P1L).

High-priority interrupts can be given priority over to low priority interrupts by using multiple interrupt servicing. If two or more interrupts with the same priority are generated simultaneously, each interrupt has a predetermined priority (see **Table 21-1**).

A standby release signal is generated.

Six external interrupt sources and thirteen internal interrupt sources are provided as maskable interrupt sources.

#### (3) Software interrupt

This is a vectored interrupt that occurs when the BRK instruction is executed. It is acknowledged even in a disabled state. The software interrupt does not undergo interrupt priority control.

## 21.2 Interrupt Sources and Configuration

A total of 21 non-maskable, maskable, and software interrupts are provided as interrupt sources (see **Table 21-1**).

**Table 21-1. Interrupt Source List (1/2)**

Interrupt Type	Default Priority <sup>Note 1</sup>	Interrupt Source		Internal/ External	Vector Table Address	Basic Configuration Type <sup>Note 2</sup>	
		Name	Trigger				
Non-maskable	—	INTWDT	Watchdog timer overflow (with watchdog timer mode 1 selected)	Internal	0004H	(A)	
Maskable	0	INTWDT	Watchdog timer overflow (with interval timer mode selected)			External	0006H
	1	INTP0	Pin input edge detection	0008H	(C)		
	2	INTP1					
	3	INTP2					
	4	INTP3					
	5	INTP4					
	6	INTP5					
	7	INTCSI0				End of serial interface channel 0 transfer	Internal
	8	INTCSI1	End of serial interface channel 1 transfer	0016H			
	9	INTSER	Occurrence of serial interface channel 2 UART reception error	0018H			
	10	INTSR	End of serial interface channel 2 UART reception	001AH			
INTCSI2		End of serial interface channel 2 3-wire transfer					
11	INTST	End of serial interface channel 2 UART transfer	001CH				

- Notes**
- The default priority is the priority used when two or more maskable interrupt requests are generated simultaneously. 0 is the highest priority and 17 is the lowest.
  - Basic configuration types (A) to (E) correspond to (A) to (E) of Figure 21-1.

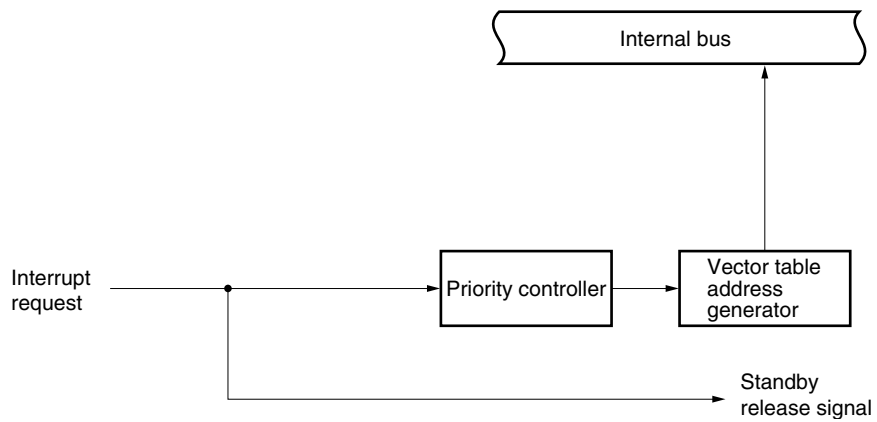
Table 21-1. Interrupt Source List (2/2)

Interrupt Type	Default Priority <sup>Note 1</sup>	Interrupt Source		Internal/External	Vector Table Address	Basic Configuration Type <sup>Note 2</sup>
		Name	Trigger			
Maskable	12	INTTM3	Reference time interval signal from watch timer	Internal	001EH	(B)
	13	INTTM00	Generation of 16-bit timer register, capture/compare register 00 (CR00) match signal		0020H	
	14	INTTM01	Generation of 16-bit timer register, capture/compare register 01 (CR01) match signal		0022H	
	15	INTTM1	Generation of 8-bit timer/event counter 1 match signal		0024H	
	16	INTTM2	Generation of 8 bit timer/event counter 2 match signal		0026H	
	17	INTAD	End of A/D converter conversion		0028H	
Software	—	BRK	BRK instruction execution	—	003EH	(E)

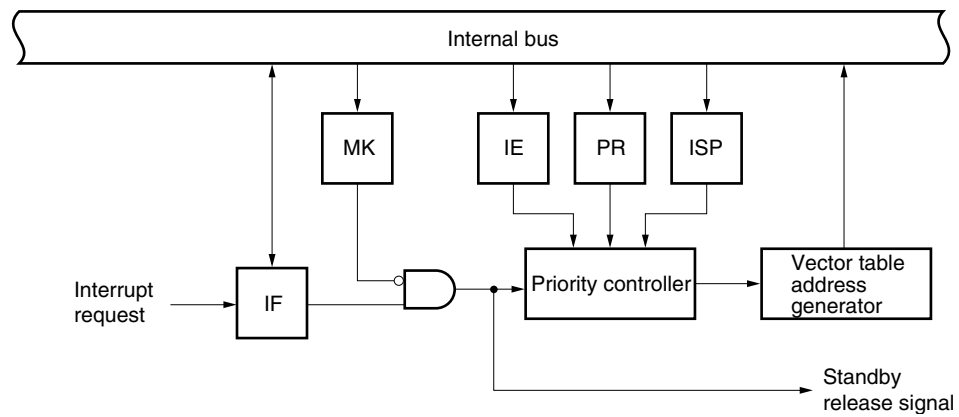
- Notes**
1. The default priority is the priority used when two or more maskable interrupt requests are generated simultaneously. 0 is the highest priority and 17 is the lowest.
  2. Basic configuration types (A) to (E) correspond to (A) to (E) of Figure 21-1.

Figure 21-1. Basic Configuration of Interrupt Function (1/2)

(A) Internal non-maskable interrupt



(B) Internal maskable interrupt



(C) External maskable interrupt (INTP0)

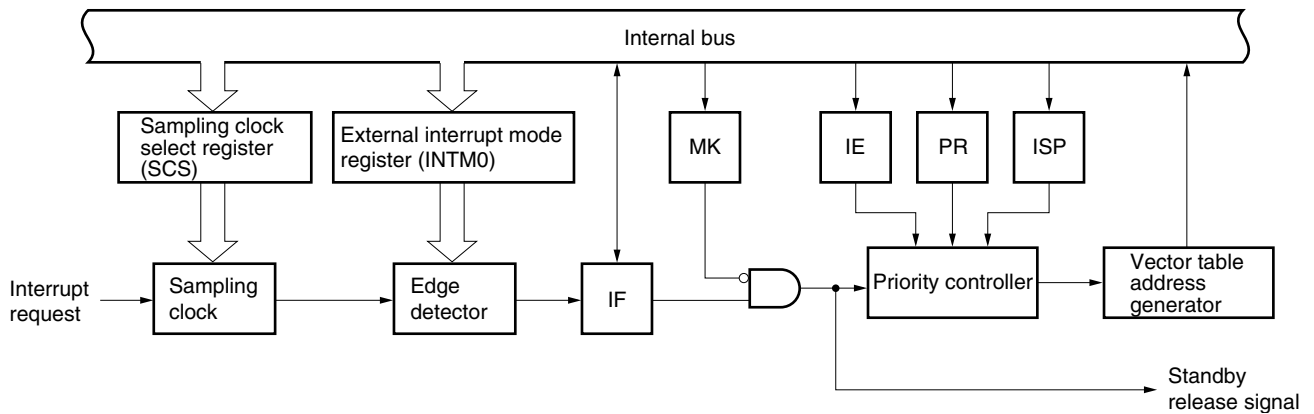
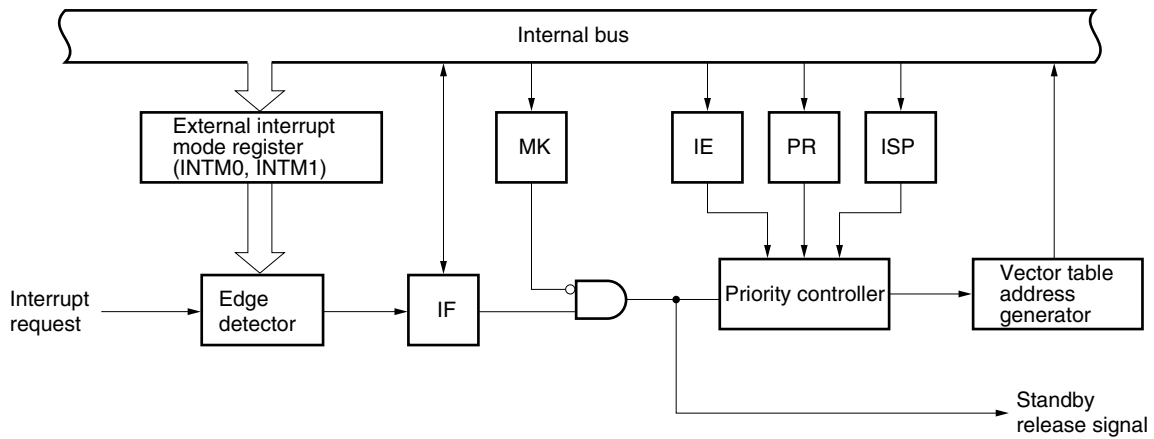
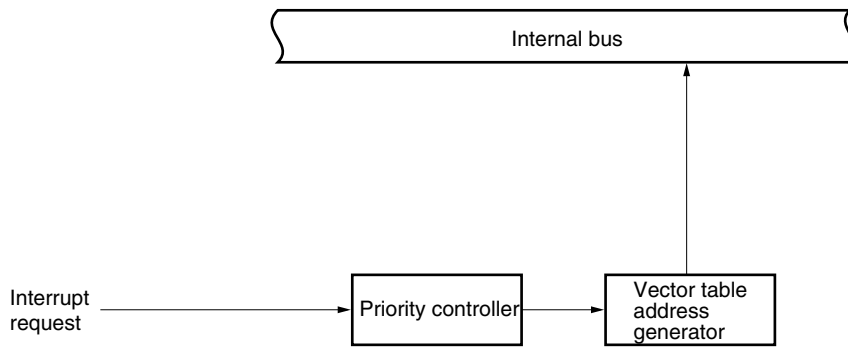


Figure 21-1. Basic Configuration of Interrupt Function (2/2)

(D) External maskable interrupt (except INTP0)



(E) Software interrupt



- Remark**
- IF: Interrupt request flag
  - IE: Interrupt enable flag
  - ISP: Inservice priority flag
  - MK: Interrupt mask flag
  - PR: Priority specification flag

### 21.3 Interrupt Function Control Registers

The following six types of registers are used to control the interrupt functions.

- Interrupt request flag register (IF0L, IF0H, IF1L)
- Interrupt mask flag register (MK0L, MK0H, MK1L)
- Priority specification flag register (PR0L, PR0H, PR1L)
- External interrupt mode register (INTM0, INTM1)
- Sampling clock select register (SCS)
- Program status word (PSW)

Table 21-2 gives a listing of interrupt request flags, interrupt mask flags, and priority specification flags corresponding to interrupt request sources.

**Table 21-2. Various Flags Corresponding to Interrupt Request Sources**

Interrupt Source	Interrupt Request Flag		Interrupt Mask Flag		Priority Specification Flag	
		Register		Register		Register
INTWDT	TMIF4	IF0L	TMMK4	MK0L	TMPR4	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			
INTCSI0	CSIF0	IF0H	CSIMK0	MK0H	CSIPR0	PR0H
INTCSI1	CSIF1		CSIMK1		CSIPR1	
INTSER	SERIF		SERMK		SERPR	
INTSR/INTCSI2	SRIF		SRMK		SRPR	
INTST	STIF		STMK		STPR	
INTTM3	TMIF3		TMMK3		TMPR3	
INTTM00	TMIF00		TMMK00		TMPR00	
INTTM01	TMIF01		TMMK01		TMPR01	
INTTM1	TMIF1	IF1L	TMMK1	MK1L	TMPR1	PR1L
INTTM2	TMIF2		TMMK2		TMPR2	
INTAD	ADIF		ADMK		ADPR	

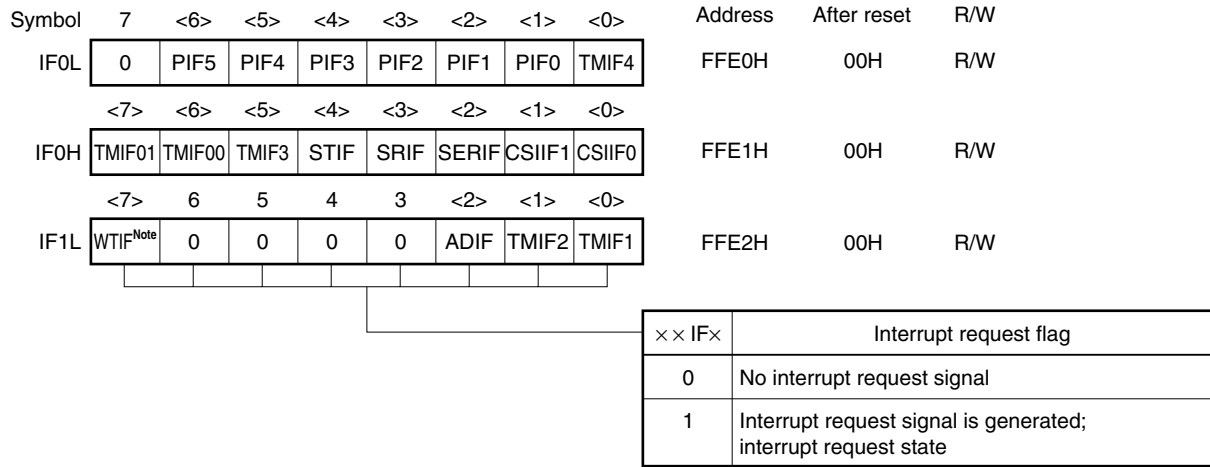
**(1) Interrupt request flag registers (IF0L, IF0H, IF1L)**

The interrupt request flag is set to 1 when the corresponding interrupt request is generated or an instruction is executed. It is cleared to 0 when an instruction is executed upon acknowledgment of an interrupt request or upon application of  $\overline{\text{RESET}}$  input.

IF0L, IF0H, and IF1L are set with a 1-bit or an 8-bit memory manipulation instruction. If IF0L and IF0H are used as the 16-bit register IF0, use a 16-bit memory manipulation instruction for setting.

$\overline{\text{RESET}}$  input clears these registers to 00H.

**Figure 21-2. Format of Interrupt Request Flag Register**



**Note** WTIF is the test input flag. A vectored interrupt request is not generated.

**Cautions** 1. The TMIF4 flag is R/W enabled only when the watchdog timer is used as an interval timer. If the watchdog timer is used in watchdog timer mode 1, clear the TMIF4 flag to 0.

2. Be sure to clear IF0L bit 7 and IF1L bits 3 to 6 to 0.

3. When an interrupt is acknowledged, the interrupt request flag is automatically cleared, and then servicing of the interrupt routine is started.

★



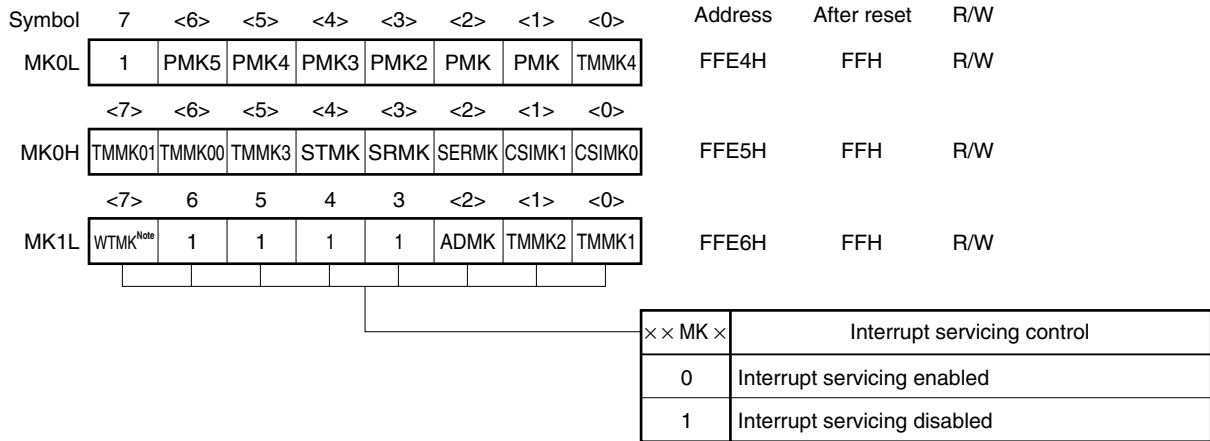
**(2) Interrupt mask flag registers (MK0L, MK0H, MK1L)**

The interrupt mask flag is used to enable/disable the corresponding maskable interrupt service and to set standby clear enable/disable.

MK0L, MK0H, and MK1L are set with a 1-bit or an 8-bit memory manipulation instruction. If MK0L and MK0H are used as the 16-bit register MK0, use a 16-bit memory manipulation instruction for setting.

$\overline{\text{RESET}}$  input sets these registers to FFH.

**Figure 21-3. Format of Interrupt Mask Flag Register**



**Note** WTMK controls standby mode release enable/disable; it does not control the interrupt function.

- Cautions**
1. If the TMMK4 flag is read when the watchdog timer is used in watchdog timer mode 1, the MK0 value becomes undefined.
  2. Because port 0 also functions as an external interrupt request input, when the output level is changed by specifying the output mode of the port function, an interrupt request flag is set. Therefore, the interrupt mask flag should be set to 1 before using the output mode.
  3. Be sure to set MK0L bit 7 and MK1L bits 3 to 6 to 1.

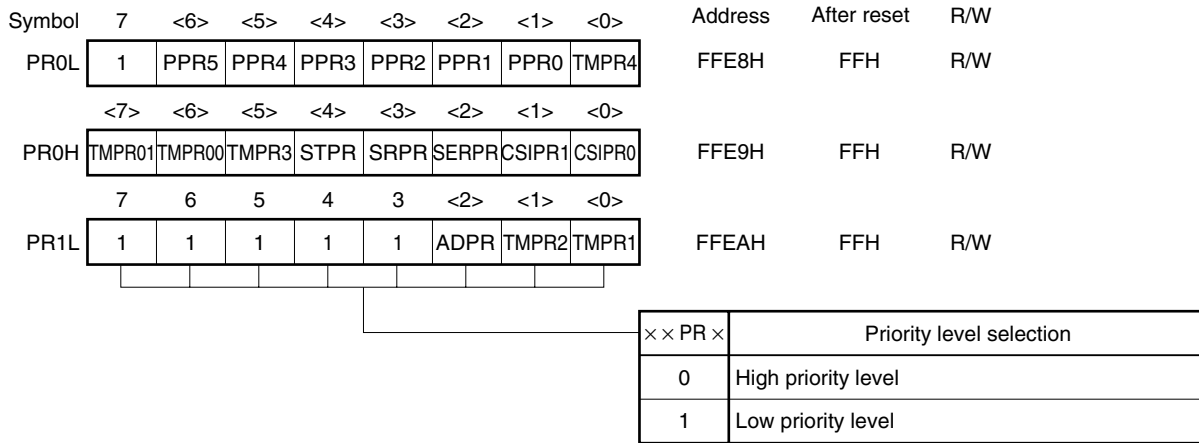
**(3) Priority specification flag registers (PR0L, PR0H, and PR1L)**

The priority specification flags are used to set the corresponding maskable interrupt priorities.

PR0L, PR0H, and PR1L are set with a 1-bit or an 8-bit memory manipulation instruction. If PR0L and PR0H are used as the 16-bit register PR0, use a 16-bit memory manipulation instruction for setting.

$\overline{\text{RESET}}$  input sets these registers to FFH.

**Figure 21-4. Format of Priority Specification Flag Register**



- Cautions**
1. If the watchdog timer is used in watchdog timer mode 1, set the TMPR4 flag to 1.
  2. Be sure to set PR0L bit 7 and PR1L bits 3 to 7 to 1.

**(4) External interrupt mode registers (INTM0, INTM1)**

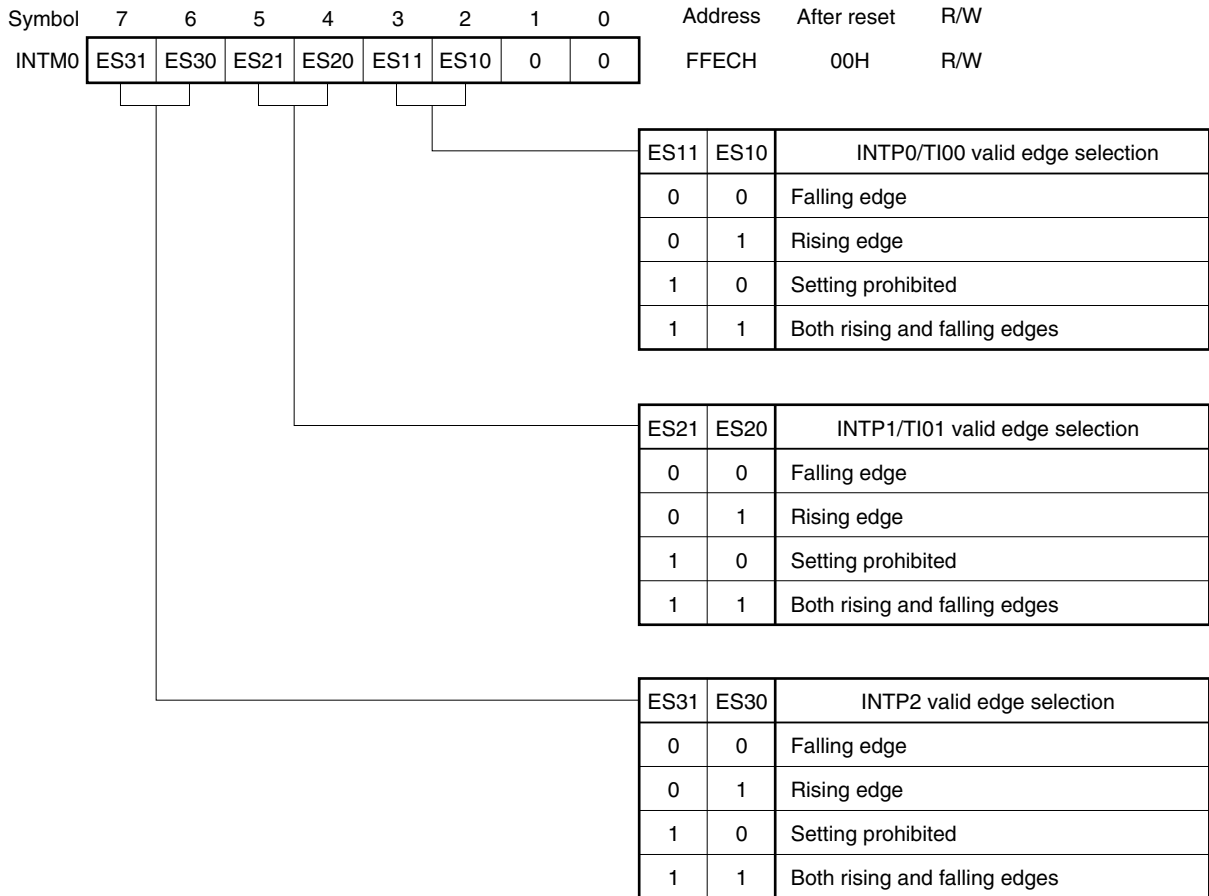
These registers set the valid edge for INTP0 to INTP5, TI00, and TI01.

INTM0 specifies the valid edges of interrupt pins INTP0 to INTP2, TI00, and TI01, and INTM1 specifies the valid edges of INTP3 to INTP5.

INTM0 and INTM1 are set with an 8-bit memory manipulation instruction.

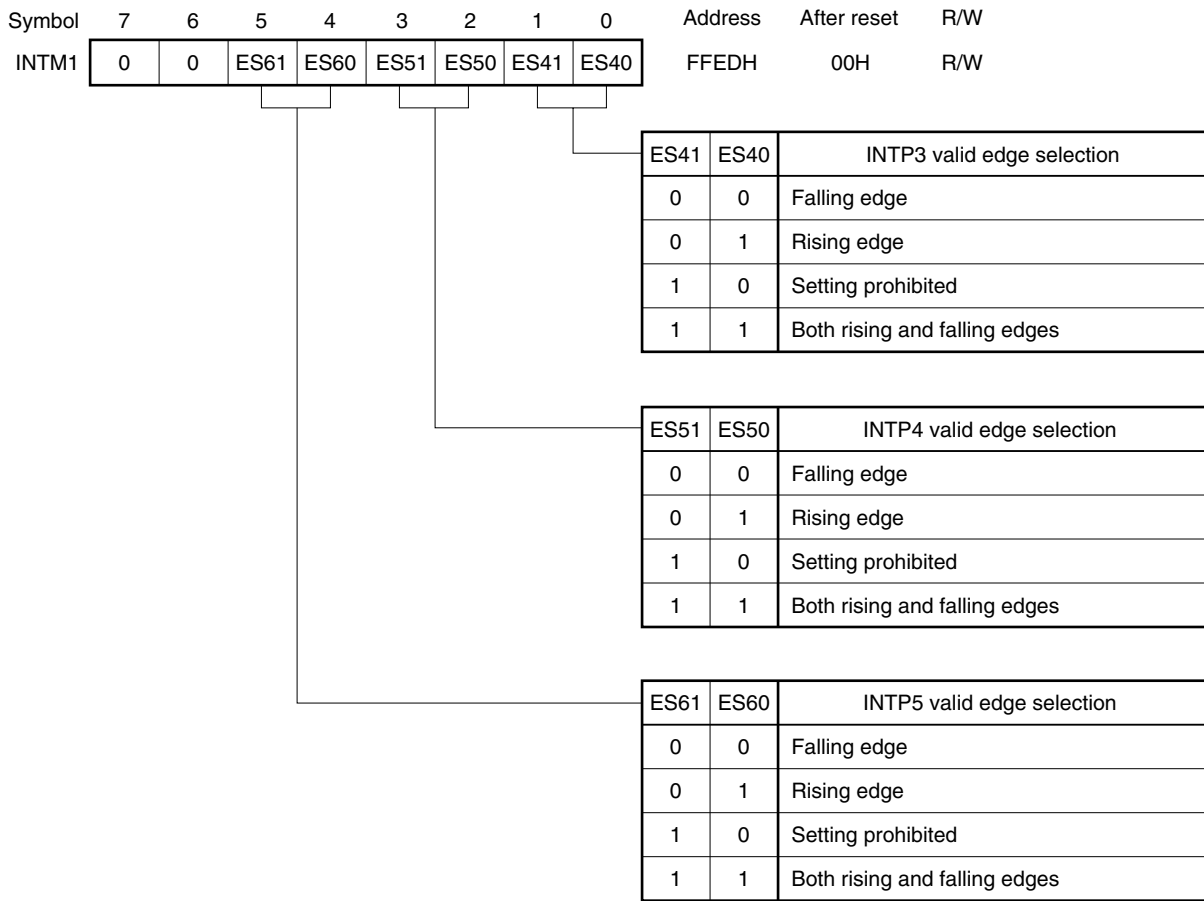
RESET input clears these registers to 00H.

**Figure 21-5. Format of External Interrupt Mode Register 0**



★ **Caution** When using the TI00/P00/INTP0 and TI01/P01/INTP1 pins as timer input pins (TI00 and TI01), stop the operation of 16-bit timer 0 by clearing bits 1 to 3 (TMC01 to TMC03) of the 16-bit timer mode control register (TMC0) to 0, 0, 0, before setting the valid edge of TI00 and TI01. The valid edge is set by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0). When using the TI00/P00/INTP0 and TI01/P01/INTP1 pins as external interrupt input pins (INTP0 and INTP1), the valid edge of INTP0 and INTP1 may be set while 16-bit timer 0 is operating.

Figure 21-6. Format of External Interrupt Mode Register 1



**(5) Sampling clock select register (SCS)**

This register is used to set the clock for sampling the valid edge input to INTP0. When remote controlled data reception is carried out using INTP0, digital noise is eliminated using the sampling clock.

SCS is set with an 8-bit memory manipulation instruction.

RESET input clears SCS to 00H.

**Figure 21-7. Format of Sampling Clock Select Register**



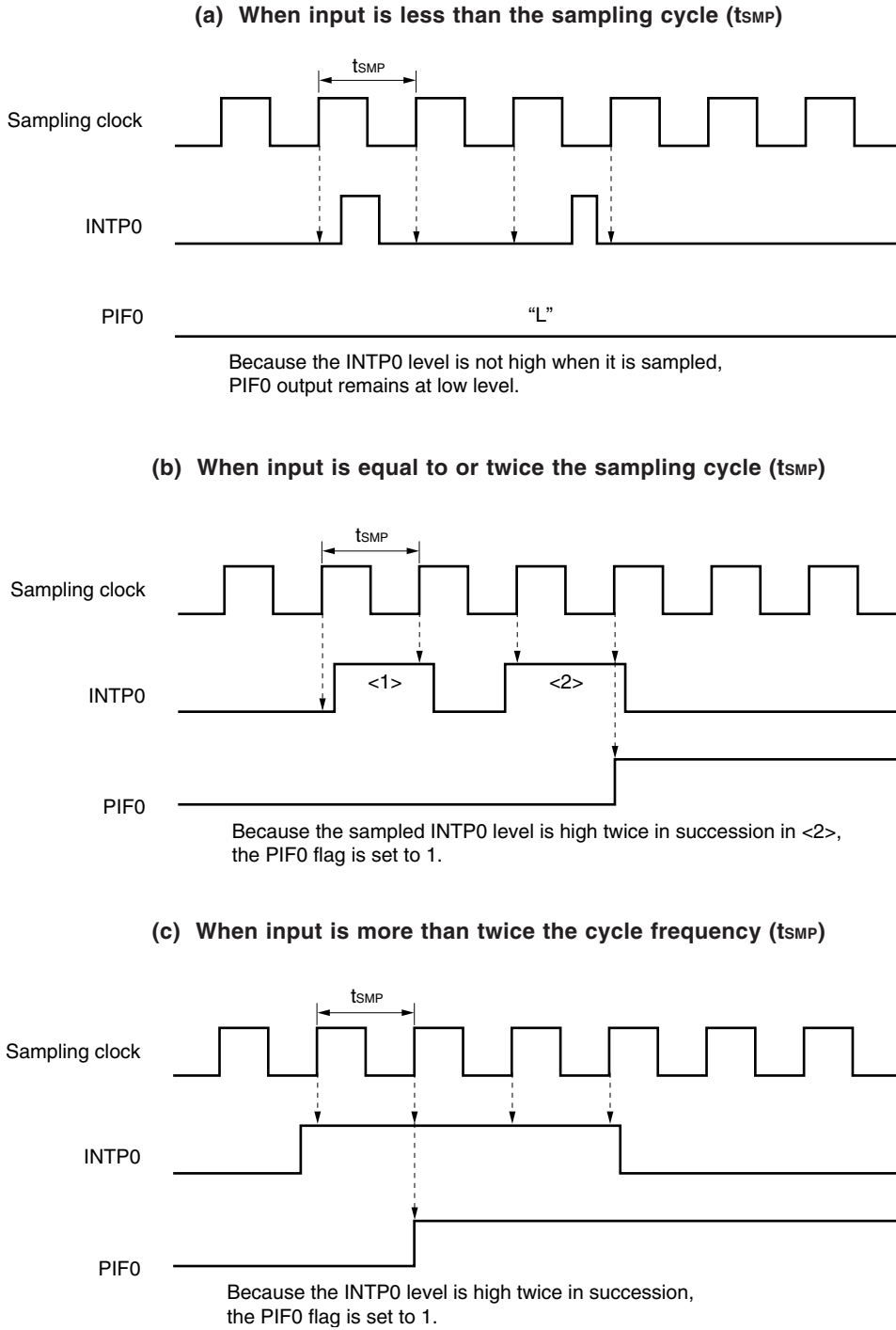
**Caution**  $f_{xx}/2^N$  is the clock supplied to the CPU and  $f_{xx}/2^5$ ,  $f_{xx}/2^6$ , and  $f_{xx}/2^7$  are clocks supplied to the peripheral hardware.  $f_{xx}/2^N$  stops in the HALT mode.

- Remarks**
1. N: Value (N = 0 to 4) of bits 0 to 2 (PCC0 to PCC2) of processor clock control register (PCC)
  2.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  3.  $f_x$ : Main system clock oscillation frequency
  4. MCS: Bit 0 of the oscillation mode select register (OSMS)
  5. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

When the sampled INTPO input level is the active level twice in succession, the noise eliminator sets the interrupt request flag (PIF0) to 1.

Figure 21-8 shows the I/O timing of the noise eliminator.

**Figure 21-8. Noise Eliminator I/O Timing (During Rising Edge Detection)**



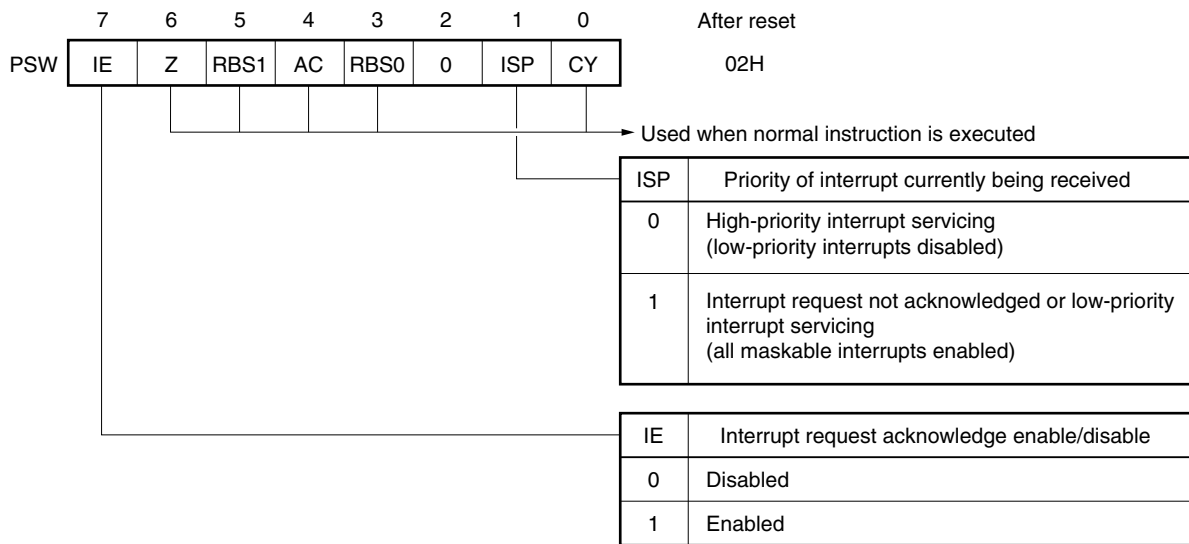
**(6) Program status word (PSW)**

The program status word is a register used to hold the instruction execution result and the current status of an interrupt request. The IE flag, which sets maskable interrupt enable/disable, and the ISP flag, which controls multiple interrupt servicing, are mapped to the PSW.

Besides 8-bit unit read/write, this register can also be manipulated by a bit manipulation instructions and dedicated instructions (EI and DI). When a vectored interrupt request is acknowledged or when the BRK instruction is executed, the contents of the PSW are automatically saved to the 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 contents of the PSW can also be saved into the stack with the PUSH PSW instruction. The contents are reset from the stack with the RETI, RETB, and POP PSW instructions.

$\overline{\text{RESET}}$  input sets the PSW to 02H.

**Figure 21-9. Format of Program Status Word**



## 21.4 Interrupt Servicing Operations

### 21.4.1 Non-maskable interrupt request acknowledgment operation

A non-maskable interrupt request is unconditionally acknowledged even if interrupt requests are in an acknowledgment disabled state. It does not undergo interrupt priority control and has the highest priority of all interrupts.

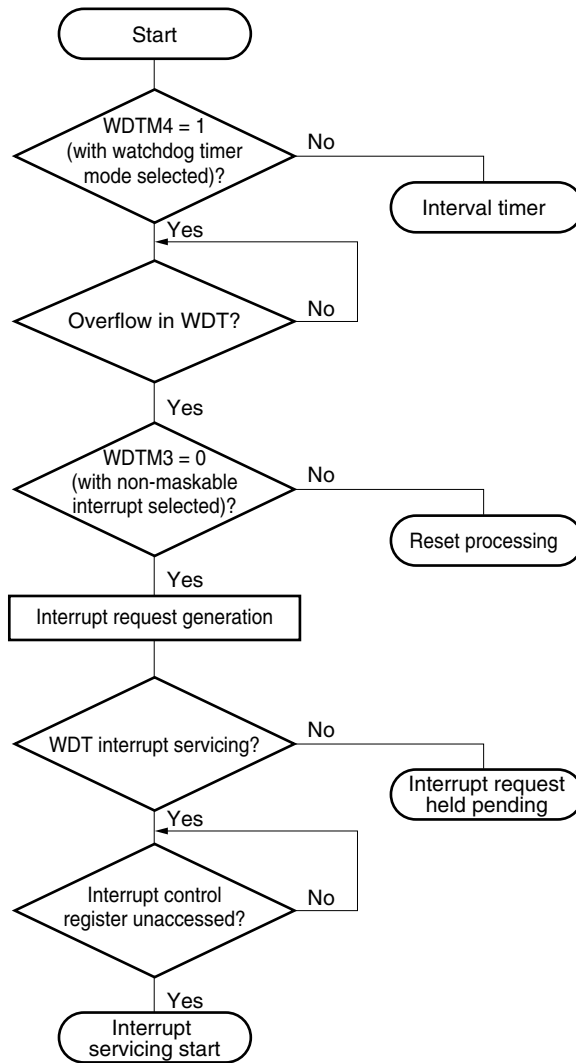
If a non-maskable interrupt request is acknowledged, the contents of the acknowledged interrupt are saved in the stack, PSW and PC, in that order, the IE and ISP flags are reset to 0, and the vector table contents are loaded into the PC and branched.

A new non-maskable interrupt request generated during execution of a non-maskable interrupt servicing program is acknowledged after the current execution of the non-maskable interrupt servicing program is terminated (following RETI instruction execution) and one main routine instruction is executed. If a new non-maskable interrupt request is generated twice or more during non-maskable interrupt service program execution, only one non-maskable interrupt request is acknowledged after termination of the non-maskable interrupt service program execution.

Figure 21-10 shows the flowchart illustrating how a non-maskable interrupt request occurs and is acknowledged. Figure 21-11 shows the acknowledgment timing of a non-maskable interrupt request. Figure 21-12 shows the acknowledgment operation of multiple non-maskable interrupt requests.



Figure 21-10. Non-Maskable Interrupt Request Occurrence and Acknowledgment Flowchart



WDTM: Watchdog timer mode register  
 WDT: Watchdog timer

Figure 21-11. Non-Maskable Interrupt Request Acknowledgment Timing

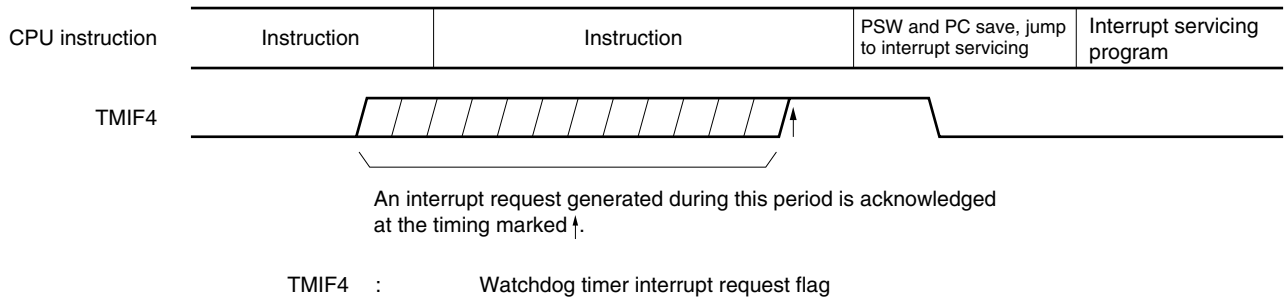
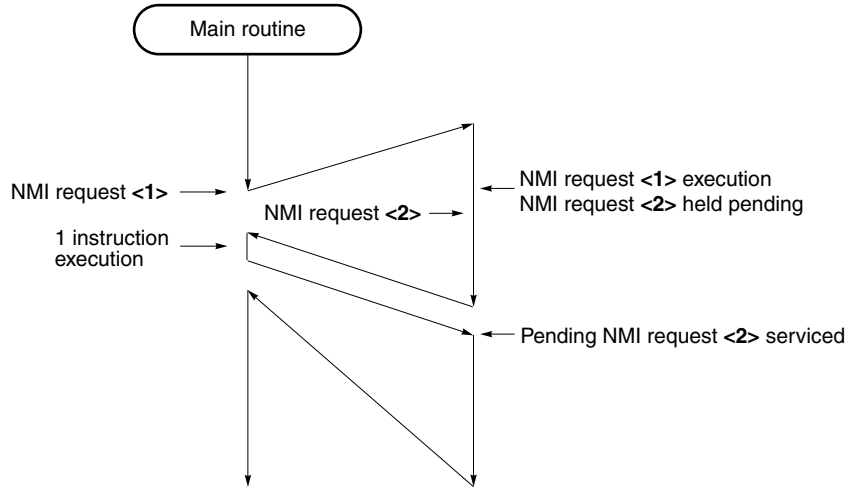
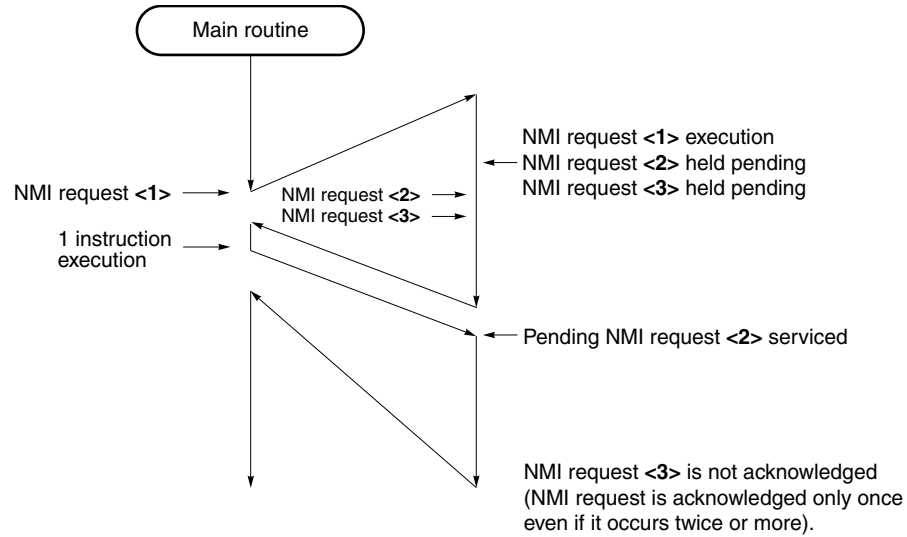


Figure 21-12. Non-Maskable Interrupt Request Acknowledgment Operation

- (a) If a new non-maskable interrupt request is generated during non-maskable interrupt servicing program execution



- (b) If two non-maskable interrupt requests are generated during non-maskable interrupt servicing program execution



**21.4.2 Maskable interrupt request acknowledgment operation**

A maskable interrupt request becomes acknowledgeable when the corresponding interrupt request flag is set to 1 and the corresponding interrupt mask (MK) flag is cleared to 0. A vectored interrupt request is acknowledged in an interrupt enabled state (with the IE flag set to 1). However, a low-priority interrupt is not acknowledged during high-priority interrupt servicing (with the ISP flag reset to 0).

Wait times from maskable interrupt request generation to interrupt servicing are shown in Table 21-3.

For the timing to acknowledge an interrupt request, see **Figures 21-14** and **21-15**.

**Table 21-3. Times from Maskable Interrupt Request Generation to Interrupt Servicing**

	Minimum Time	Maximum Time <sup>Note</sup>
When xxPR = 0	7 clocks	32 clocks
When xxPR = 1	8 clocks	33 clocks

**Note** If an interrupt request is generated just before a divide instruction, the wait time becomes the maximum.

**Remark** 1 clock:  $\frac{1}{f_{CPU}}$  (f<sub>CPU</sub>: CPU clock)

If two or more maskable interrupt requests are generated simultaneously, the request specified as higher priority with the priority specification flag is acknowledged first. If two or more requests specified as the same priority by the interrupt priority specification flag are generated simultaneously, the one with the higher default priority is acknowledged first.

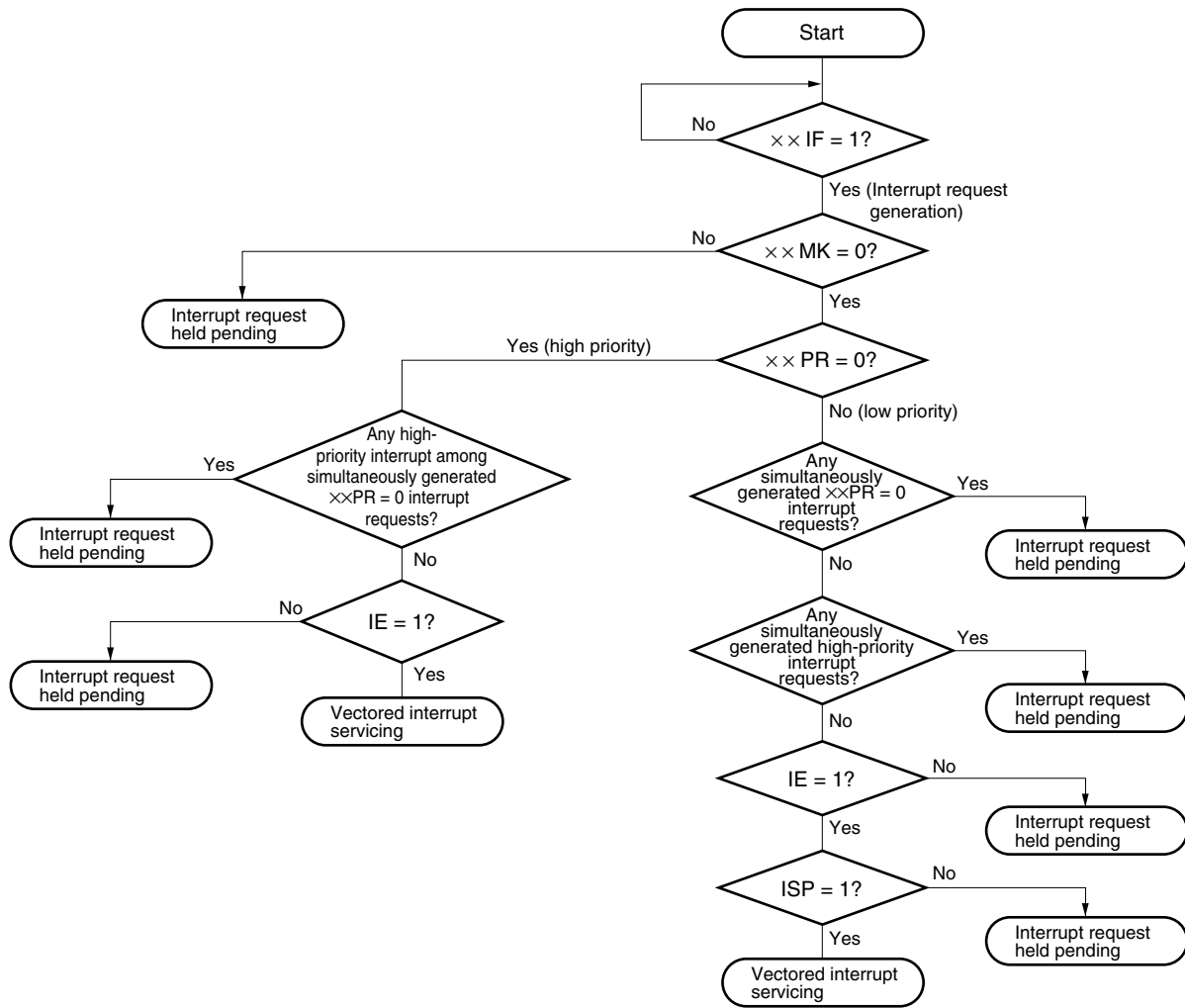
Any pending interrupt requests are acknowledged when they become acknowledgeable.

Figure 21-13 shows an interrupt request acknowledgment algorithm.

If a maskable interrupt request is acknowledged, the contents of acknowledged interrupt are saved in the stack, program status word (PSW) and program counter (PC), in that order, the IE flag is reset to 0, and the acknowledged interrupt priority specification flag contents are transferred to the ISP flag. Further, the vector table data determined for each interrupt request is loaded into the PC and branched.

Restoration from the interrupt is possible with the RETI instruction.

Figure 21-13. Interrupt Request Acknowledgment Processing Algorithm



xxIF: Interrupt request flag

xxMK: Interrupt mask flag

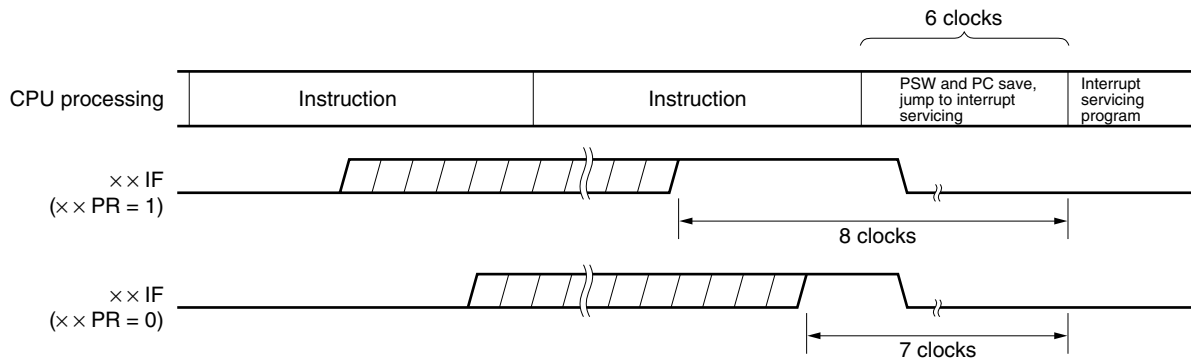
xxPR: Priority specification flag

IE: Flag that controls maskable interrupt request acknowledge (1 = enable, 0 = disable)

ISP: Flag indicating priority of interrupt currently being serviced (0 = interrupt with high priority is being serviced.

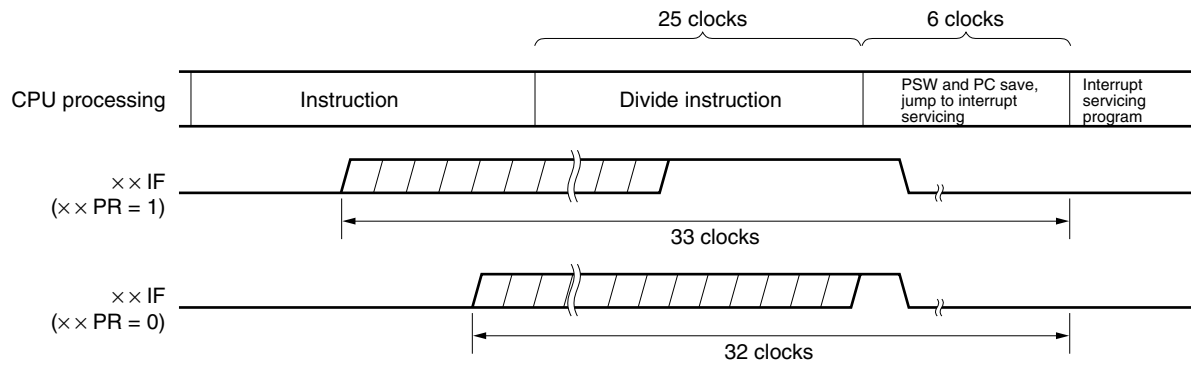
1 = interrupt request is not acknowledged or interrupt with low priority being serviced).

Figure 21-14. Interrupt Request Acknowledgment Timing (Minimum Time)



Remark 1 clock:  $\frac{1}{f_{CPU}}$  ( $f_{CPU}$ : CPU clock)

Figure 21-15. Interrupt Request Acknowledgment Timing (Maximum Time)



Remark 1 clock:  $\frac{1}{f_{CPU}}$  ( $f_{CPU}$ : CPU clock)

**21.4.3 Software interrupt request acknowledgment operation**

A software interrupt request is acknowledged by BRK instruction execution. Software interrupts cannot be disabled.

If a software interrupt request is acknowledged, the contents of the acknowledged interrupt are saved in the stack, program status word (PSW) and program counter (PC), in that order, the IE flag is reset to 0 and the contents of the vector tables (003EH and 003FH) are loaded into the PC and branched.

Restoration from the software interrupt is possible with the RETB instruction.

**Caution Do not use the RETI instruction for restoring from the software interrupt.**

**21.4.4 Multiple interrupt servicing**

Acknowledging another interrupt request while one interrupt is being serviced is called multiple interrupt servicing.

Multiple interrupt servicing does not occur unless interrupt requests are enabled (IE = 1) (except the non-maskable interrupt). When an interrupt request is acknowledged, the other interrupts are disabled (IE = 0). To enable multiple interrupt servicing, therefore, the IE flag must be set to 1 by executing the EI instruction during interrupt servicing and interrupts must be enabled. Even if interrupt requests are enabled, multiple interrupt servicing may not be possible. However, this is controlled by the programmable priority.

An interrupt has two types of priorities: the default priority and the programmable priority. Multiple interrupt servicing is controlled by the programmable priority.

In the EI status, if an interrupt request with a priority that is the same or higher than that of the interrupt currently being serviced is generated, the interrupt is acknowledged for multiple interrupt servicing. If an interrupt request with a priority lower than that of the interrupt currently being serviced is generated, the interrupt is not acknowledged for multiple interrupt servicing.

If interrupts are disabled, or if multiple interrupt servicing is not enabled because the interrupt has a low priority, the interrupt is held pending. After the servicing of the current interrupt has been completed, and after one instruction of the main processing has been executed, the pending interrupt is acknowledged.

Multiple interrupt servicing is not enabled while a non-maskable interrupt is being serviced.

Table 21-4 shows the interrupt requests enabled for multiple interrupt servicing. Figure 21-16 shows multiple interrupt servicing examples.

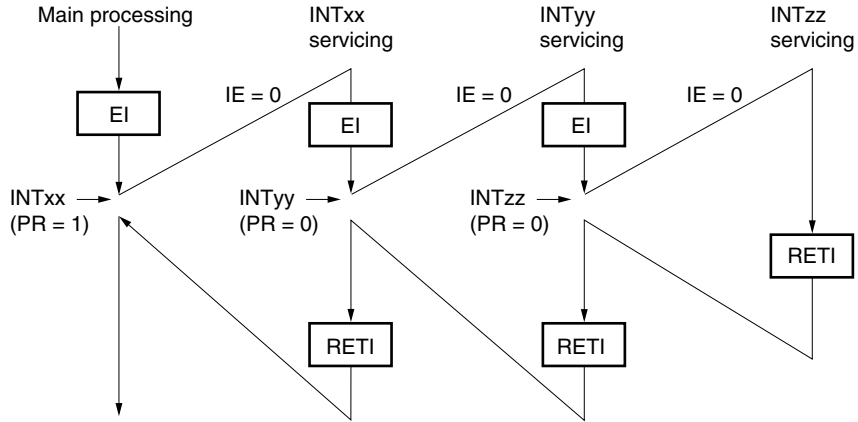
**Table 21-4. Interrupt Request Enabled for Multiple Interrupt Servicing During Interrupt Servicing**

Multiple Interrupt Request Interrupt Being Serviced	Non-maskable Interrupt Request	Maskable Interrupt Request			
		PR = 0		PR = 1	
		IE = 1	IE = 0	IE = 1	IE = 0
Non-maskable interrupt	D	D	D	D	D
Maskable interrupt	ISP = 0	E	D	D	D
	ISP = 1	E	D	E	D
Software interrupt	E	E	D	E	D

- Remarks**
1. E: Multiple interrupt servicing enabled
  2. D: Multiple interrupt servicing disabled
  3. ISP and IE are flags contained in the PSW
    - ISP = 0: An interrupt with a higher priority is being serviced
    - ISP = 1: An interrupt request is not 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, and PR1L
    - PR = 0: Higher priority level
    - PR = 1: Lower priority level

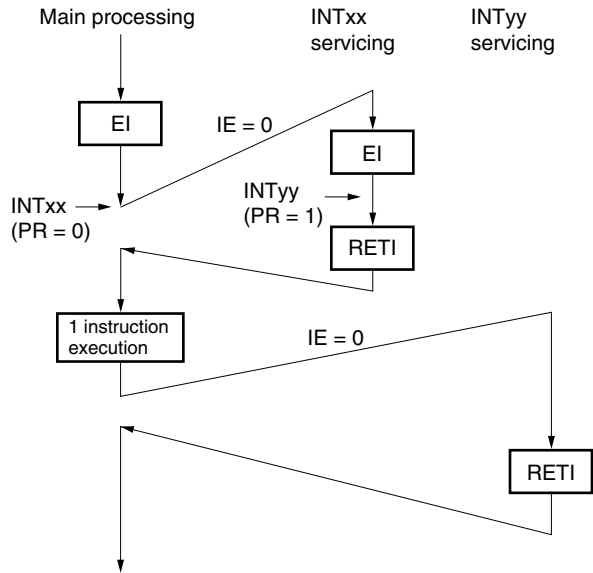
Figure 21-16. Multiple Interrupt Servicing Example (1/2)

**Example 1. Multiple interrupt servicing occurs twice**



Two interrupt requests, INTyy and INTzz, are acknowledged while the INTxx interrupt is being serviced. Before each interrupt request is acknowledged, the EI instruction is always issued and interrupt requests are enabled.

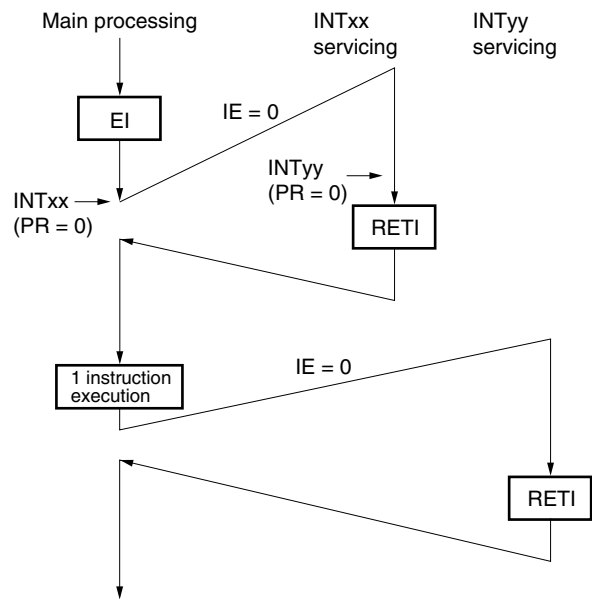
**Example 2. Multiple interrupt servicing does not occur because of interrupt priority**



INTyy, which occurs while INTxx is being serviced is not acknowledged for multiple interrupt servicing because the priority of INTyy is lower than that of INTxx. INTyy is held pending and is acknowledged after one instruction of the main processing has been executed.

- PR = 0: High-priority interrupt
- PR = 1: Low-priority interrupt
- IE = 0: Interrupt acknowledgment disabled

Figure 21-16. Multiple Interrupt Servicing Example (2/2)

**Example 3. Multiple interrupt servicing does not occur because interrupts are not enabled**

In the servicing of INTxx, other interrupts are not enabled (the EI instruction is not executed). Therefore, INTyy is not acknowledged for multiple interrupt servicing. This interrupt is held pending and acknowledged after one instruction of the main processing has been executed.

PR = 0: High-priority interrupt

IE = 0: Interrupt acknowledgment disabled



**21.4.5 Interrupt request pending**

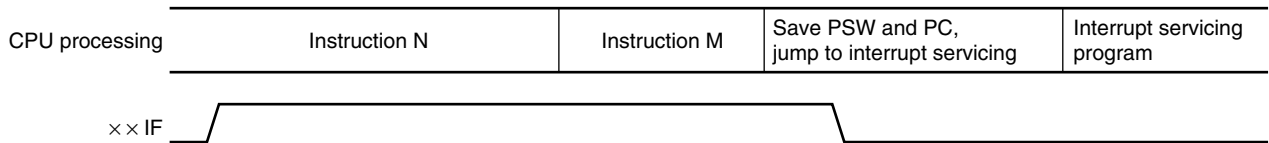
For some instructions, even if an interrupt is generated while that instruction is being executed, the interrupt is held pending until execution of the next instruction is completed. The instructions that hold interrupt requests pending (interrupt request pending) are shown below.

- 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 IF0L, IF0H, IF1L, MK0L, MK0H, MK1L, PR0L, PR0H, PR1L, INTM0, INTM1 registers

**Caution** The BRK instruction is not an interrupt request pending instruction. However, the IE flag is cleared to 0 by a software interrupt that is started by BRK instruction execution. Thus, even if a maskable interrupt request is generated during BRK instruction execution, that interrupt request is not acknowledged. However, a non-maskable interrupt request is acknowledged.

Figure 21-17 shows the timing at which an interrupt request is held pending.

**Figure 21-17. Interrupt Request Pending Timing**



- Remarks**
1. Instruction N: Instruction that holds interrupts requests pending
  2. Instruction M: Instructions other than instruction N
  3. The  $\times \times PR$  (priority level) values do not affect the operation of  $\times \times IF$  (interrupt request).

## 21.5 Test Function

When the watch timer overflows and the port 4 falling edge is detected, the internal test input flag is set to 1, and the standby release signal is generated.

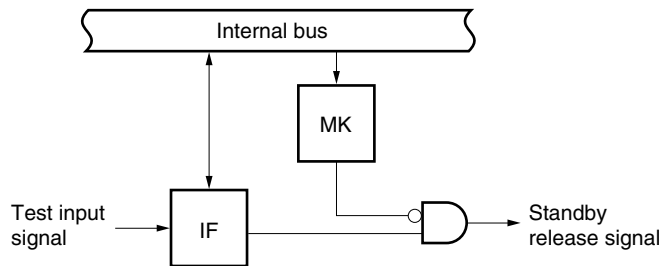
Unlike the interrupt function, vectored processing is not performed.

There are two test input sources as shown in Table 21-5. The basic configuration is shown in Figure 21-18.

**Table 21-5. Test Input Sources**

Test Input Sources		Internal/ External
Name	Trigger	
INTWT	Watch timer overflow	Internal
INTPT4	Falling edge detection at port 4	External

**Figure 21-18. Basic Configuration of Test Function**



**Remark** IF: Test input flag  
MK: Test mask flag

### 21.5.1 Registers controlling test function

The test function is controlled by the following three registers.

- Interrupt request flag register 1L (IF1L)
- Interrupt mask flag register 1L (MK1L)
- Key return mode register (KRM)

The names of the test input flags and test mask flags corresponding to the test input signals are listed in Table 21-6.

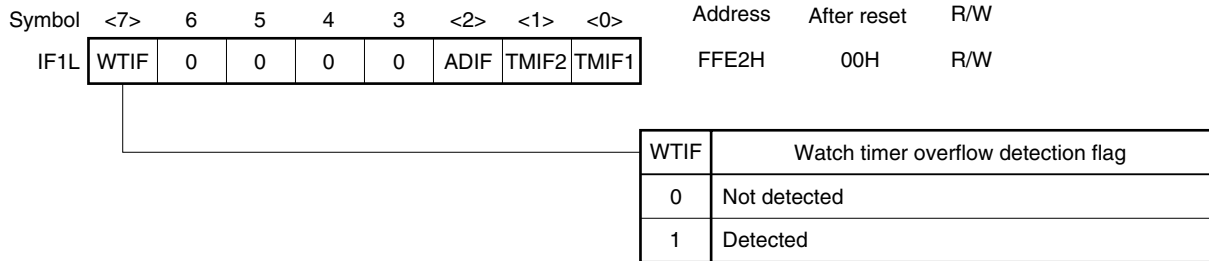
**Table 21-6. Flags Corresponding to Test Input Signals**

Test Input Signal Name	Test Input Flag	Test Mask Flag
INTWT	WTIF	WTMK
INTPT4	KRIF	KRMK

**(1) Interrupt request flag register 1L (IF1L)**

This register indicates whether a watch timer overflow is detected or not.  
 IF1L is set with a 1-bit or 8-bit memory manipulation instruction.  
 $\overline{\text{RESET}}$  input clears IF1L to 00H.

**Figure 21-19. Format of Interrupt Request Flag Register 1L**

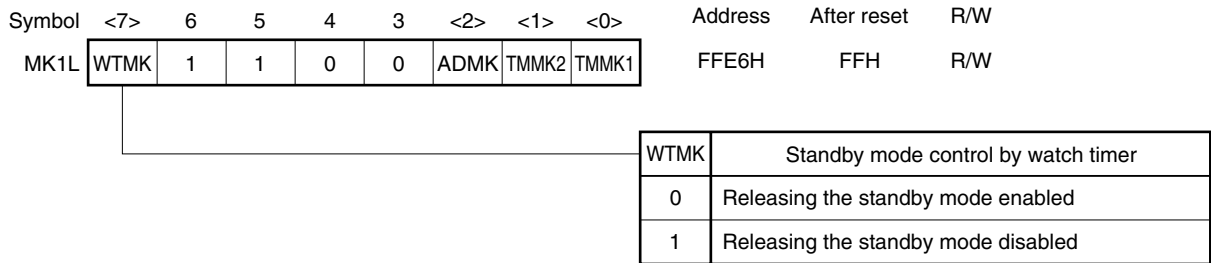


**Caution** Be sure to clear bits 3 to 6 to 0.

**(2) Interrupt mask flag register 1L (MK1L)**

This register is used to set the standby mode enable/disable at the time the standby mode is released by the watch timer.  
 MK1L is set with a 1-bit or 8-bit memory manipulation instruction.  
 $\overline{\text{RESET}}$  input sets MK1L to FFH.

**Figure 21-20. Format of Interrupt Mask Flag Register 1L**



**Caution** Be sure to set bits 3 to 6 to 1.

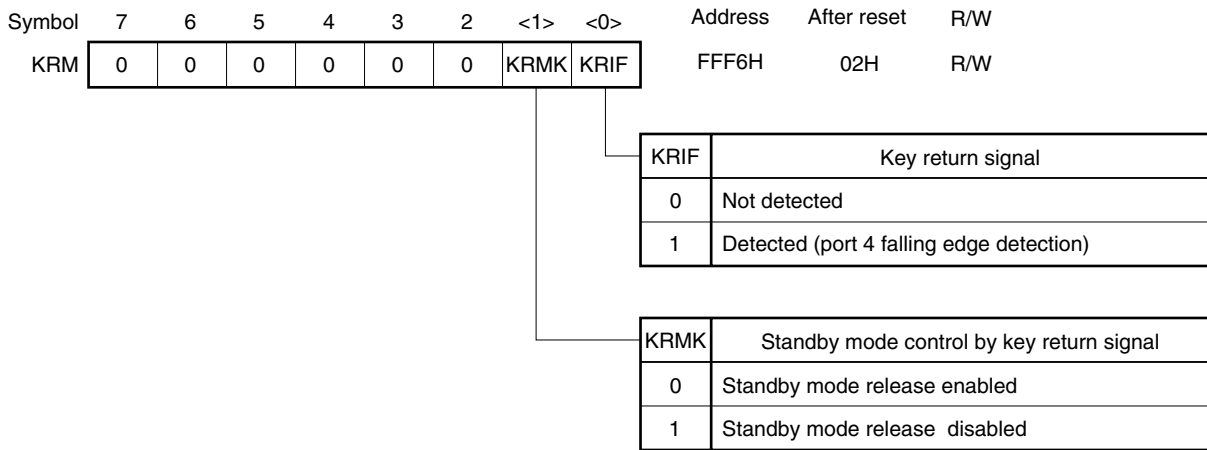
**(3) Key return mode register (KRM)**

This register is used to set enable/disable of standby function clear by the key return signal (port 4 falling edge detection).

KRM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets KRM to 02H.

**Figure 21-21. Format of Key Return Mode Register**



**Caution** When port 4 falling edge detection is used, be sure to clear KRIF to 0 (it is not cleared to 0 automatically).

**21.5.2 Test input signal acknowledgment operation**

**(1) Internal test signal (INTWT)**

INTWT is generated when the watch timer overflows, and sets the WTIF flag. Unless interrupts are masked by the interrupt mask flag (WTMK) at this time, the standby release signal is generated.

The watch function is realized by checking the WTIF flag at a shorter cycle than the watch timer overflow cycle.

**(2) External test input signal (INTPT4)**

INTPT4 is generated when a falling edge is input to the port 4 pins (P40 to P47), and KRIF is set. Unless interrupts are masked by the interrupt mask flag (KRMK) at this time, the standby release signal is generated. If port 4 is used as a key matrix return signal input, whether or not a key input has been applied can be checked from the KRIF status.

## CHAPTER 22 EXTERNAL DEVICE EXPANSION FUNCTION

### 22.1 External Device Expansion Function

The external device expansion function connects external devices to areas other than the internal ROM, RAM, and SFR. Ports 4 to 6 are used for connection of external devices. Ports 4 to 6 control addresses/data, the read/write strobe, wait, address strobe etc.

**Table 22-1. Pin Functions in External Memory Expansion Mode**

Pin Function When External Device Is Connected		Alternate Function
Name	Function	
AD0 to AD7	Multiplexed address/data bus	P40 to P47
A8 to A15	Address bus	P50 to P57
$\overline{\text{RD}}$	Read strobe signal	P64
$\overline{\text{WR}}$	Write strobe signal	P65
$\overline{\text{WAIT}}$	Wait signal	P66
ASTB	Address strobe signal	P67

**Table 22-2. State of Port 4 to 6 Pins in External Memory Expansion Mode**

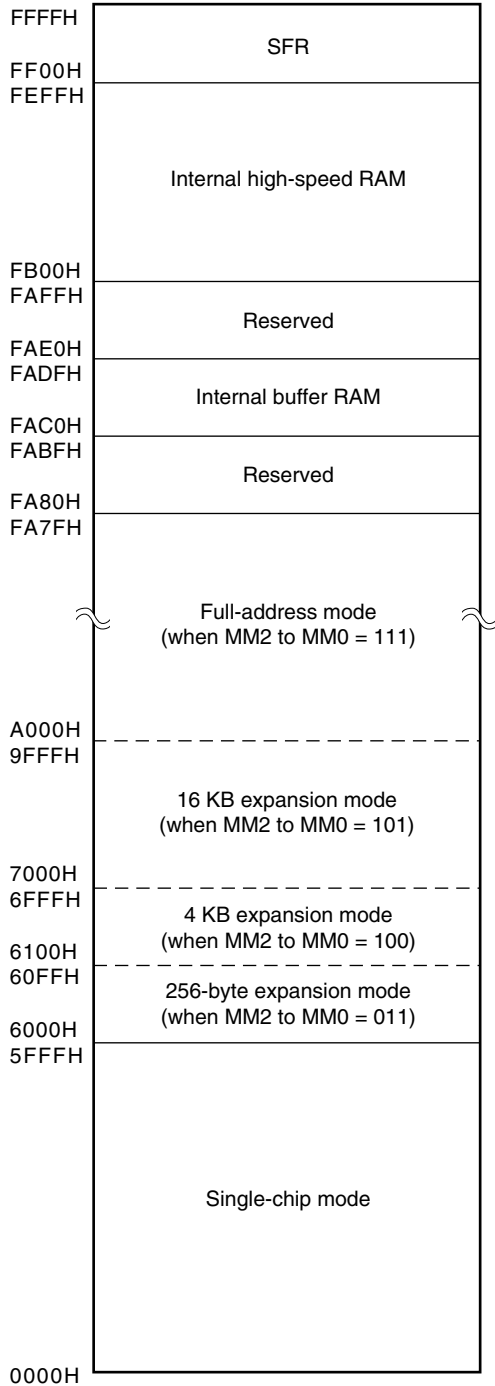
Ports and Bits External Expansion Modes	Port 4	Port 5							Port 6	
	0 to 7	0	1	2	3	4	5	6	7	0 to 3
Single-chip mode	Port	Port							Port	Port
256-byte expansion mode	Address/data	Port							Port	$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WAIT}}$ , ASTB
4 KB expansion mode	Address/data	Address			Port				Port	$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WAIT}}$ , ASTB
16 KB expansion mode	Address/data	Address				Port			Port	$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WAIT}}$ , ASTB
Full-address mode	Address/data	Address							Port	$\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{WAIT}}$ , ASTB

**Caution** When the external wait function is not used, the  $\overline{\text{WAIT}}$  pin can be used as a port in all modes.

Memory maps when using the external device expansion function are as follows.

Figure 22-1. Memory Map When Using External Device Expansion Function (1/3)

(a) Memory map of  $\mu$ PD780053 and 780053Y, and  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y with internal ROM (flash memory) set to 24 KB



(b) Memory map of  $\mu$ PD780054 and 780054Y, and  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y with internal ROM (flash memory) set to 32 KB

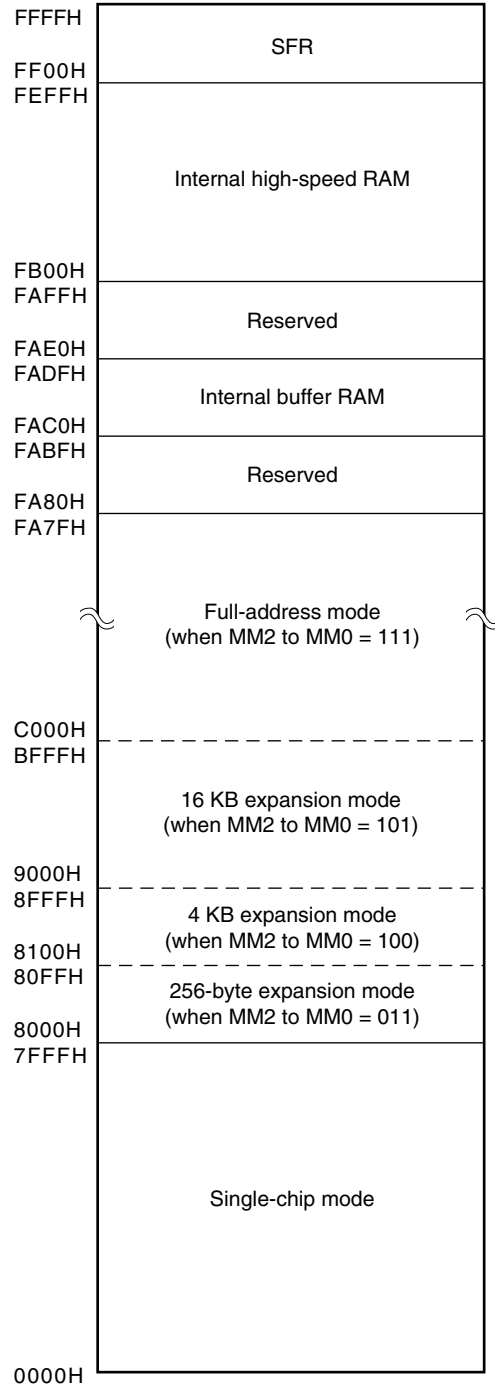
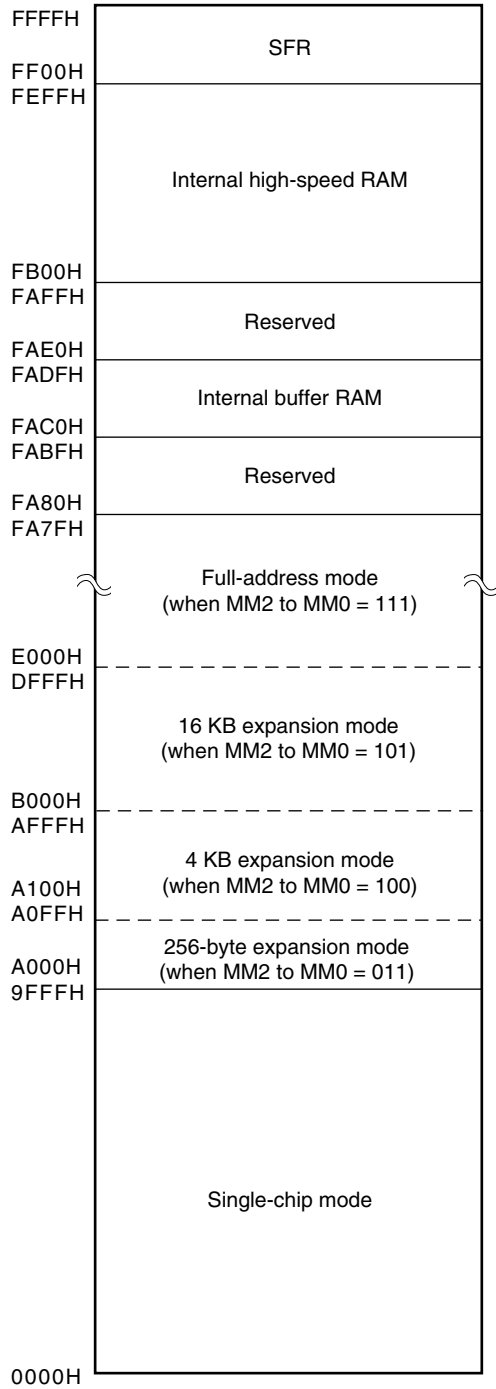


Figure 22-1. Memory Map When Using External Device Expansion Function (2/3)

(c) Memory map of  $\mu$ PD780055 and 780055Y, and  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y with internal ROM (flash memory) set to 40 KB



(d) Memory map of  $\mu$ PD780056 and 780056Y, and  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y with internal ROM (flash memory) set to 48 KB

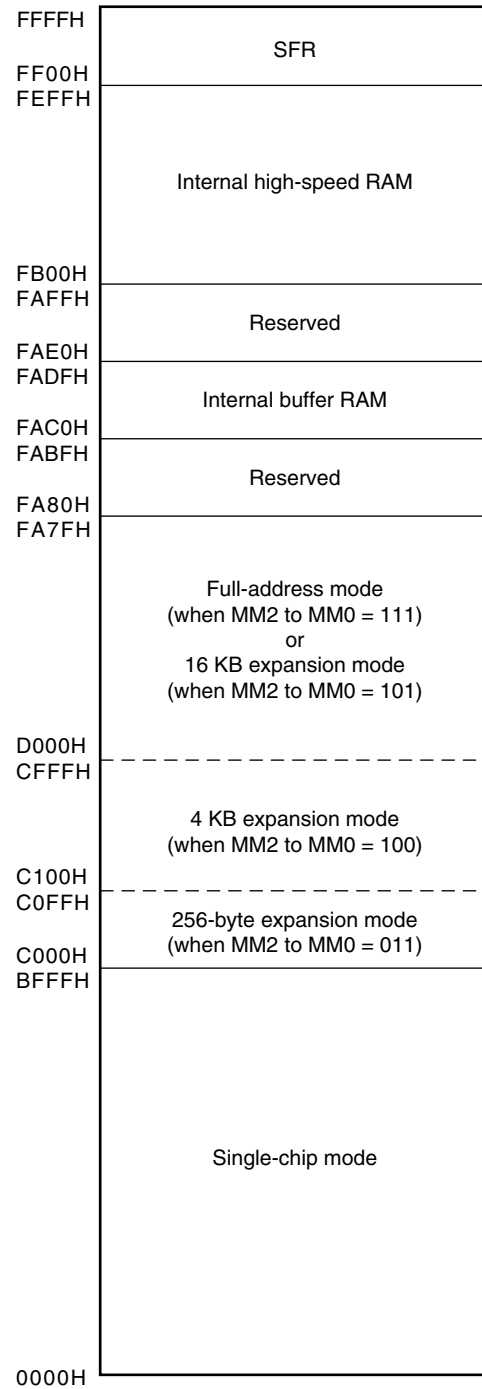
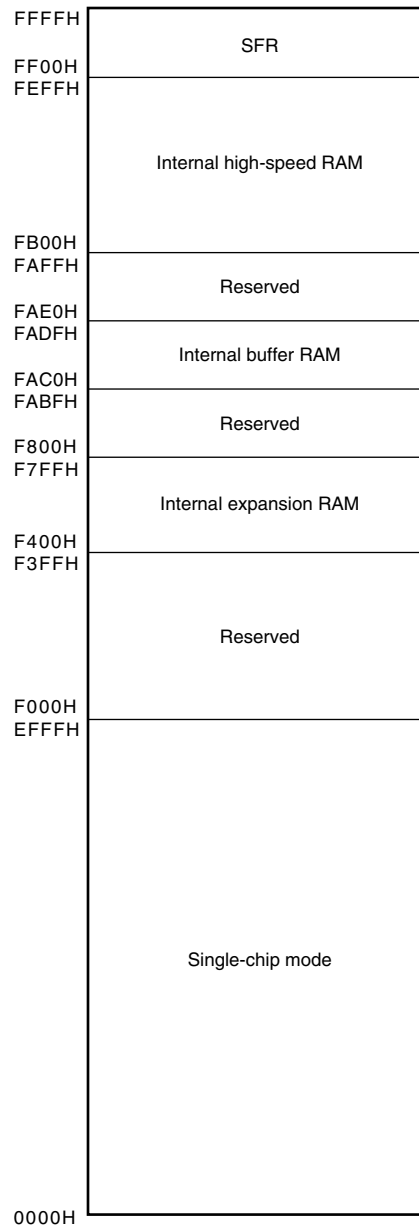
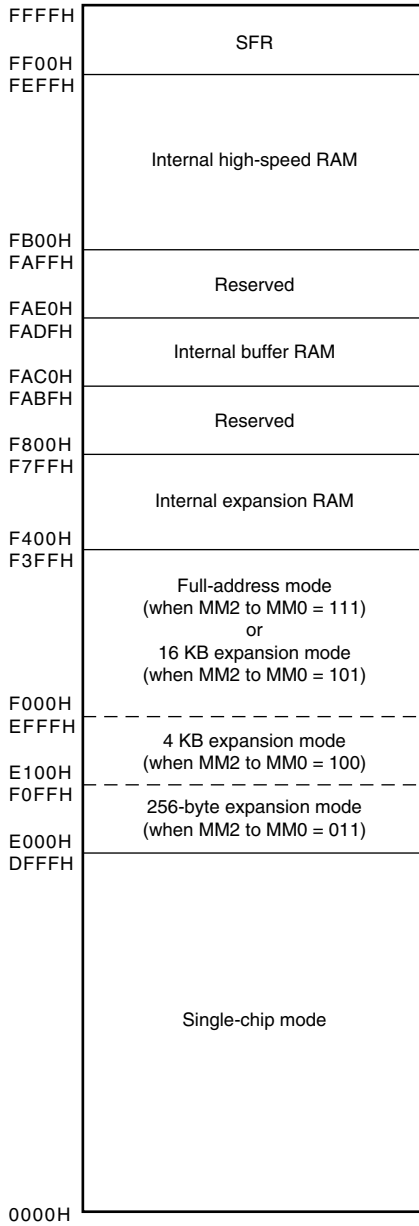


Figure 22-1. Memory Map When Using External Device Expansion Function (3/3)

(e)  $\mu$ PD780058, 780058B, 780058BY, 78F0058, 78F0058Y Memory map when internal ROM (flash memory) size is 56 KB

(f)  $\mu$ PD780058, 780058B, 780058BY, 78F0058, 78F0058Y Memory map when internal ROM (flash memory) size is 60 KB



**Caution** When the internal ROM (flash memory) size is 60 KB, the area from F000H to F3FFH cannot be used. F000H to F3FFH can be used as external memory by setting the internal ROM (flash memory) size to 56 KB or less using the internal memory size switching register (IMS).



## 22.2 External Device Expansion Function Control Register

The external device expansion function is controlled by the memory expansion mode register (MM) and internal memory size switching register (IMS).

### (1) Memory expansion mode register (MM)

MM sets the wait count and external expansion area, and also sets the input/output mode of port 4.

MM is set with a 1-bit memory or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets MM to 10H.

Figure 22-2. Format of Memory Expansion Mode Register

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
MM	0	0	PW1	PW0	0	MM2	MM1	MM0	FFF8H	10H	R/W

MM2	MM1	MM0	Single-chip/ memory expansion mode selection		P40 to P47, P50 to P57, P64 to P67 pin state					
					P40 to P47		P50 to P53	P54, P55	P56, P57	P64 to P67
0	0	0	Single-chip mode		Port mode	Input	Port mode			
0	0	1					Output			
0	1	1	Memory expansion mode	256-byte mode	AD0 to AD7	Port mode				
1	0	0		4 KB mode		A8 to A11	Port mode		P64 = $\overline{\text{RD}}$ P65 = $\overline{\text{WR}}$ P66 = $\overline{\text{WAIT}}$ P67 = $\overline{\text{ASTB}}$	
1	0	1		16 KB mode			A12, A13	Port mode		
1	1	1		Full- address mode <sup>Note</sup>		A14, A15				
Other than above			Setting prohibited							

PW1	PW0	Wait control
0	0	No wait
0	1	Wait (one wait state insertion)
1	0	Setting prohibited
1	1	Wait control by external wait pin

**Note** The full-address mode allows external expansion to the entire 64 KB address space except for the internal ROM, RAM, and SFR areas and the reserved areas.

**Remark** P60 to P63 are used as port pins without regard to the mode (single-chip mode or memory expansion mode).

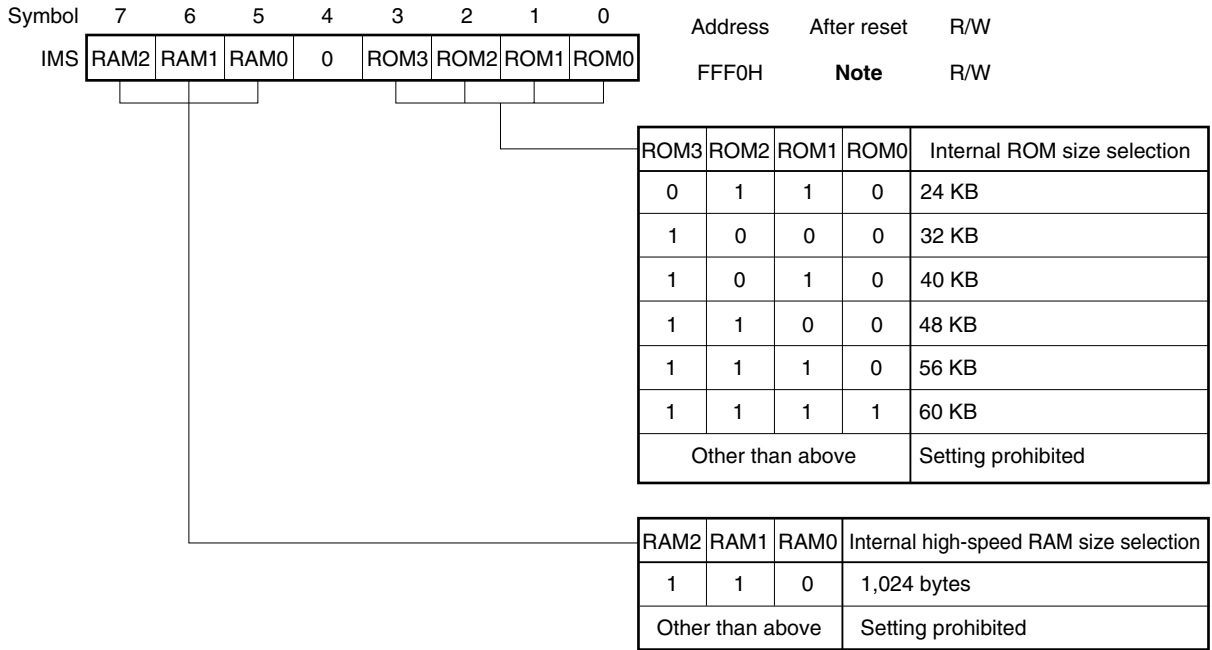
**(2) Internal memory size switching register (IMS)**

This register specifies the internal memory size. In principle, use IMS in the default status. However, when using the external device expansion function with the  $\mu$ PD780058, 780058B, and 780058BY, set IMS so that the internal ROM capacity is 56 KB or less.

IMS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets IMS to the value indicated in Table 22-3.

**Figure 22-3. Format of Internal Memory Size Switching Register**



**Note** The values after reset depend on the product (see Table 22-3).

**Table 22-3. Values After Internal Memory Size Switching Register Is Reset**

Part Number	Reset Value
$\mu$ PD780053, 780053Y	C6H
$\mu$ PD780054, 780054Y	C8H
$\mu$ PD780055, 780055Y	CAH
$\mu$ PD780056, 780056Y	CCH
$\mu$ PD780058, 780058B, 780058BY	CFH

### 22.3 External Device Expansion Function Timing

The timing control signal output pins in the external memory expansion mode are as follows.

**(1)  $\overline{\text{RD}}$  pin (alternate function: P64)**

Read strobe signal output pin. The read strobe signal is output upon the occurrence of data accesses and instruction fetches from external memory.

During internal memory access, the read strobe signal is not output (maintains high level).

**(2)  $\overline{\text{WR}}$  pin (alternate function: P65)**

Write strobe signal output pin. The write strobe signal is output upon the occurrence of data access to external memory.

During internal memory access, the write strobe signal is not output (maintains high level).

**(3)  $\overline{\text{WAIT}}$  pin (alternate function: P66)**

External wait signal input pin. When the external wait is not used, the  $\overline{\text{WAIT}}$  pin can be used as an I/O port.

During internal memory access, the external wait signal is ignored.

**(4)  $\overline{\text{ASTB}}$  pin (alternate function: P67)**

Address strobe signal output pin. The  $\overline{\text{ASTB}}$  signal is output without regard to data accesses and instruction fetches from external memory. The  $\overline{\text{ASTB}}$  signal is also output when the internal memory is accessed.

**(5)  $\overline{\text{AD0}}$  to  $\overline{\text{AD7}}$ ,  $\overline{\text{A8}}$  to  $\overline{\text{A15}}$  pins (alternate function: P40 to P47, P50 to P57)**

Address/data signal output pin. A valid signal is output or input during data accesses and instruction fetches from external memory.

These signals change when the internal memory is accessed (output values are undefined).

The timing charts are shown in Figures 22-4 to 22-7.

Figure 22-4. Instruction Fetch from External Memory

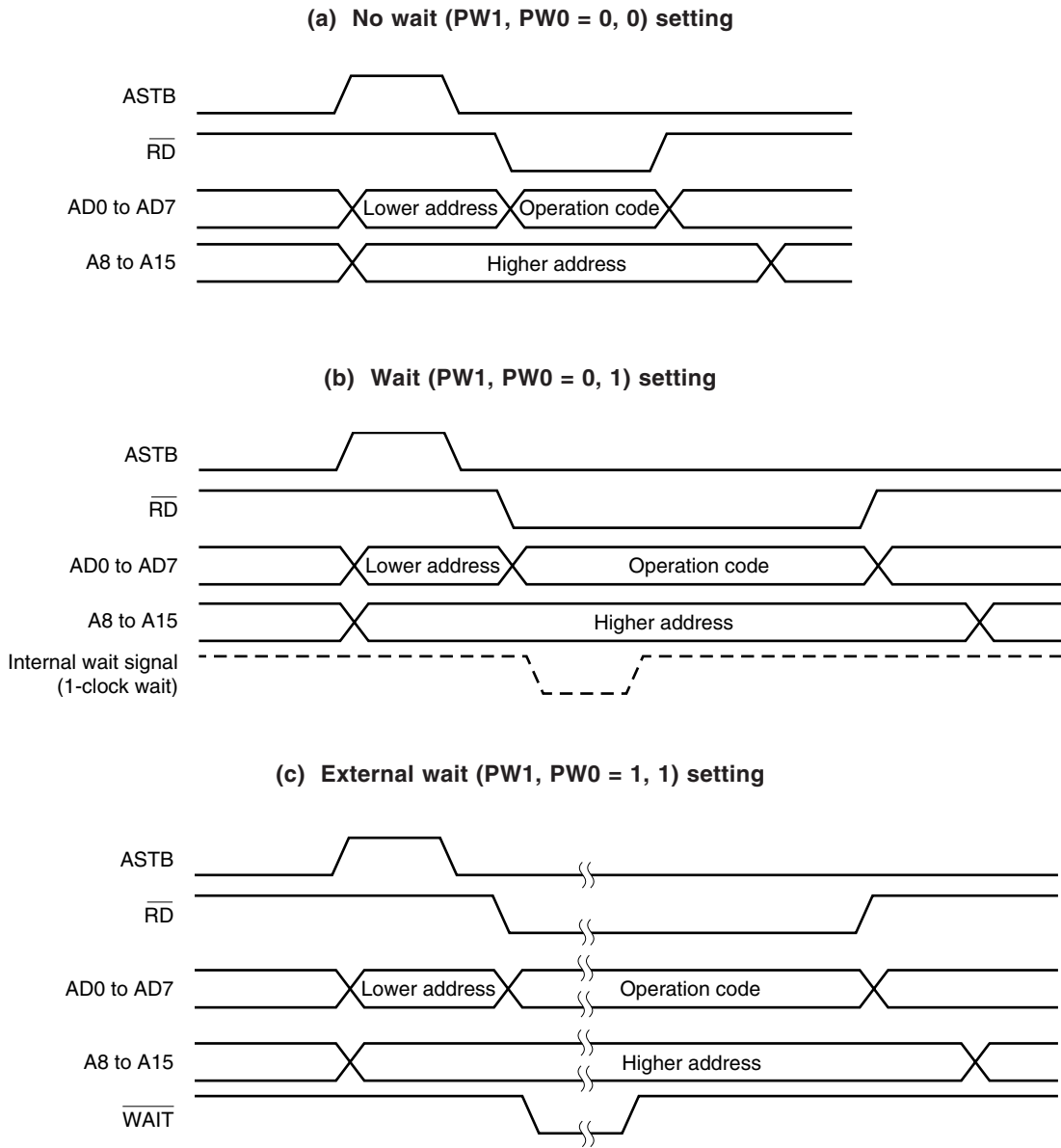
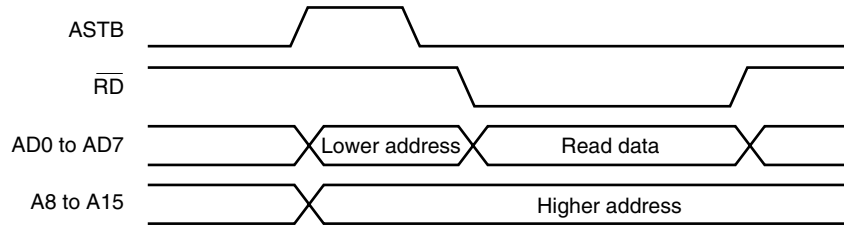
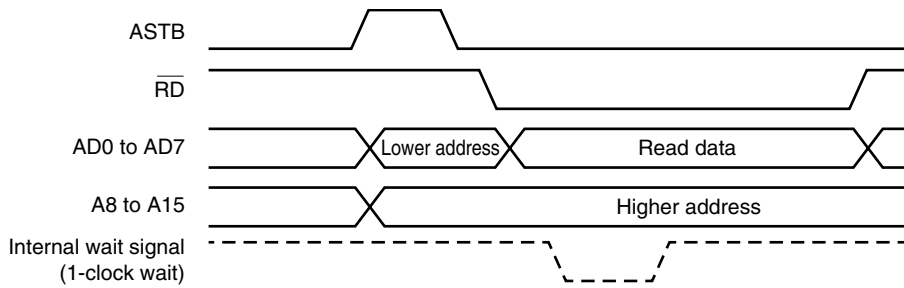


Figure 22-5. External Memory Read Timing

(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



(c) External wait (PW1, PW0 = 1, 1) setting

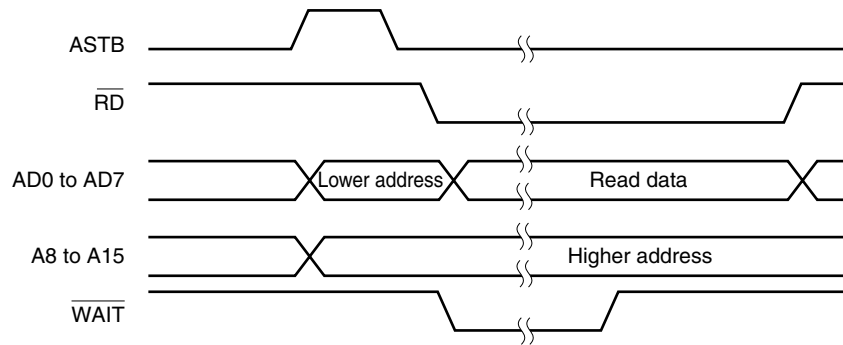
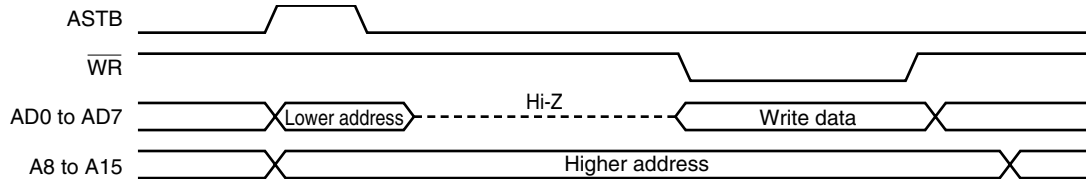
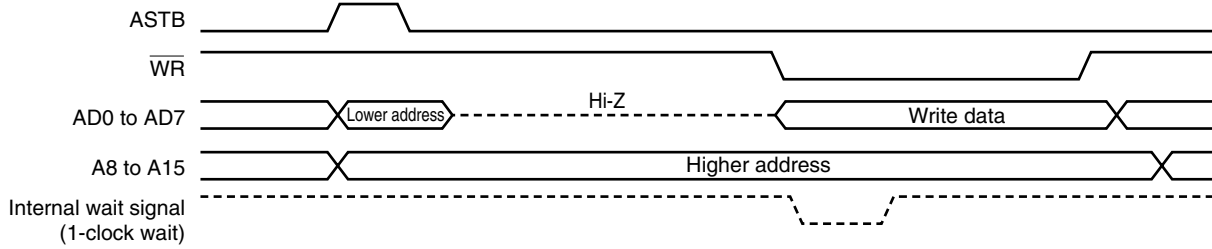


Figure 22-6. External Memory Write Timing

(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



(c) External wait (PW1, PW0 = 1, 1) setting

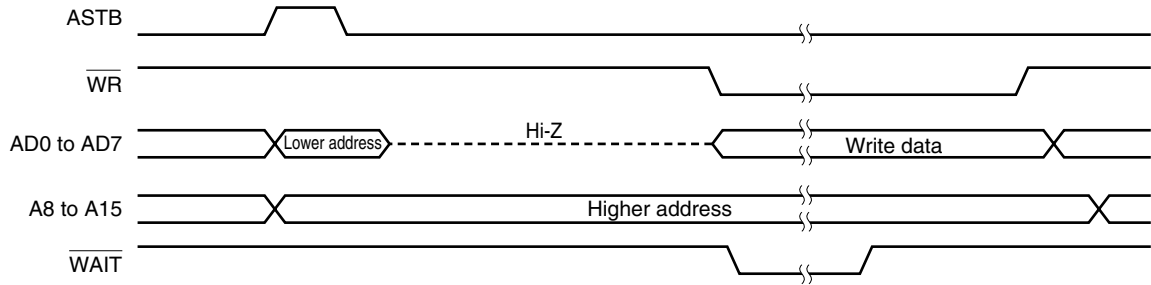
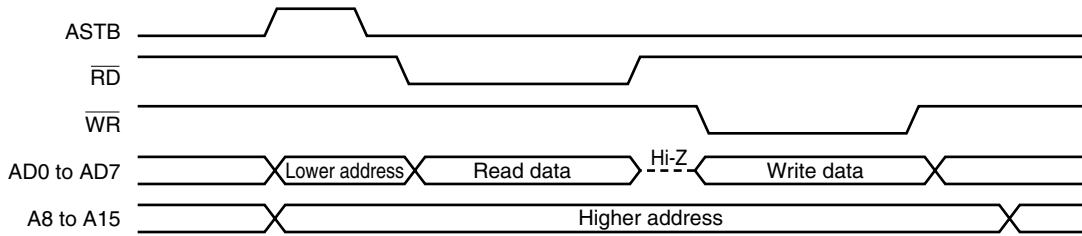
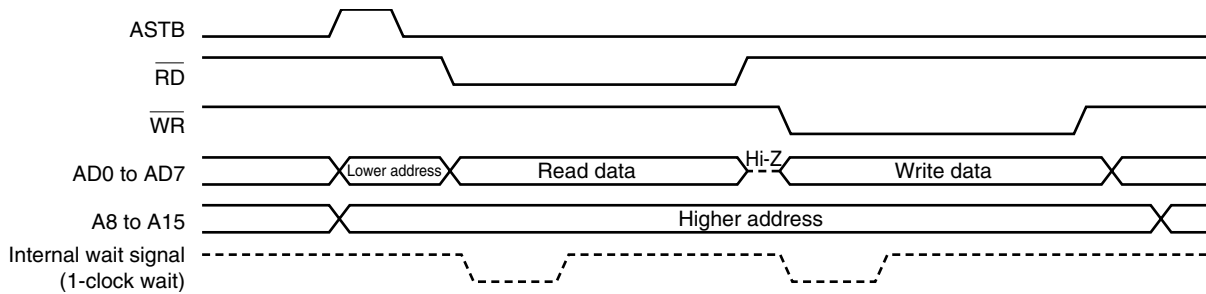


Figure 22-7. External Memory Read Modify Write Timing

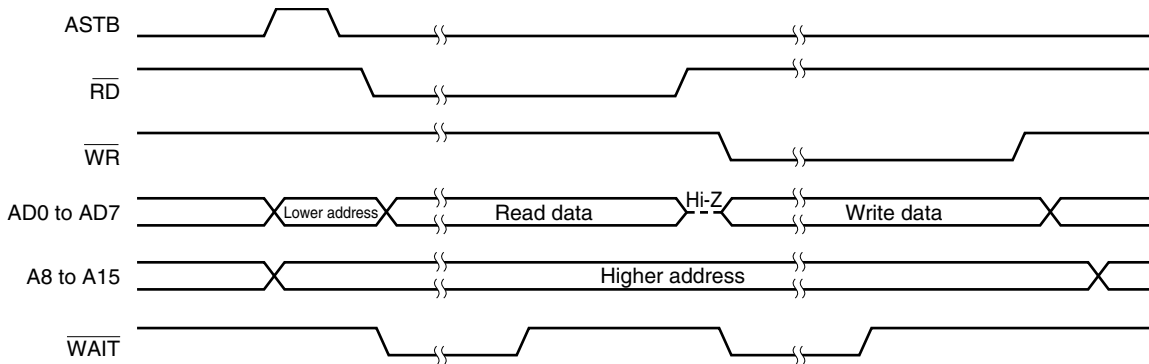
(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



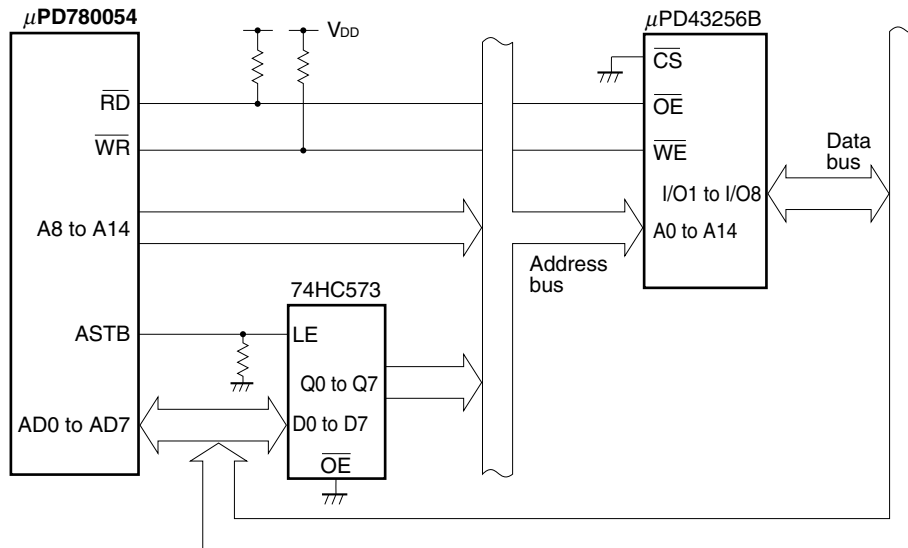
(c) External wait (PW1, PW0 = 1, 1) setting



### 22.4 Example of Connection with Memory

Figure 22-8 shows an example of the connection between the  $\mu$ PD780054 and external memory. SRAM is used as the external memory in this diagram. In addition, the external device expansion function is used in the full-address mode, and the addresses from 0000H to 7FFFH (32 KB) are allocated to internal ROM, and the addresses after 8000H to SRAM.

Figure 22-8. Example of Connection Between  $\mu$ PD780054 and Memory





### 23.1 Standby Function and Configuration

#### 23.1.1 Standby function

The standby function is designed to decrease the power consumption of the system. The following two modes are available.

##### (1) HALT mode

HALT instruction execution sets the HALT mode. The HALT mode is used to stop the CPU operation clock. The system clock oscillator continues oscillating. In this mode, the current consumption cannot be decreased as much as in the STOP mode, but the HALT mode is effective for restarting immediately upon interrupt request and to carry out intermittent operations such as in watch applications.

##### (2) STOP mode

STOP instruction execution sets the STOP mode. In the STOP mode, the main system clock oscillator stops and the whole system stops. The CPU current consumption can be considerably decreased.

Data memory low-voltage hold (down to  $V_{DD} = 1.8$  V) is possible. Thus, the STOP mode is effective for holding data memory contents with ultra-low current consumption. Because this mode can be cleared upon interrupt request, it enables intermittent operations to be carried out.

However, because a wait time is necessary to secure oscillation stabilization after the STOP mode is released, select the HALT mode if it is necessary to start processing immediately upon interrupt request.

In any mode, all the contents of the registers, flags and data memory just before 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 system operates on the main system clock (subsystem clock oscillation cannot be stopped). The HALT mode can be used with either the main system clock or the subsystem clock.**
  - 2. When shifting to the STOP mode, be sure to stop the peripheral hardware operation before executing the STOP instruction.**
  - 3. The following sequence is recommended for power consumption reduction of the A/D converter when the standby function is used: first clear bit 7 (CS) of the A/D converter mode register (ADM) to 0 to stop the A/D conversion operation, and then execute the HALT or STOP instruction.**

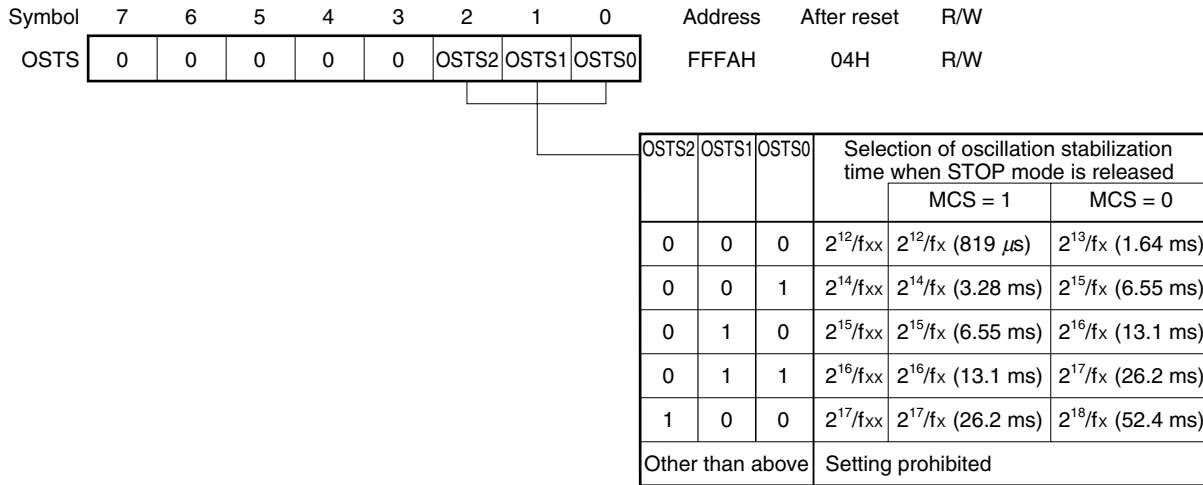
**23.1.2 Standby function control register**

The wait time after the STOP mode is released upon interrupt request until the oscillation stabilizes is controlled by the oscillation stabilization time select register (OSTS).

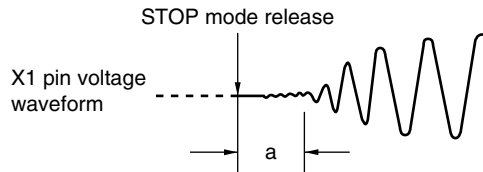
OSTS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets OSTS to 04H. However, it takes  $2^{17}/f_x$ , not  $2^{18}/f_x$ , until the STOP mode is released by  $\overline{\text{RESET}}$  input.

**Figure 23-1. Format of Oscillation Stabilization Time Select Register**



**Caution** The wait time that elapses when the STOP mode is released does not include the time required for the clock to start oscillation (see “a” in the illustration below) after the STOP mode is released. The same applies when the STOP mode is released by  $\overline{\text{RESET}}$  input and by generation of an interrupt request.



- Remarks**
1.  $f_{xx}$ : Main system clock frequency ( $f_x$  or  $f_x/2$ )
  2.  $f_x$ : Main system clock oscillation frequency
  3. MCS: Bit 0 of the oscillation mode select register (OSMS)
  4. Values in parentheses apply to operation with  $f_x = 5.0$  MHz

## 23.2 Standby Function Operations

### 23.2.1 HALT mode

#### (1) HALT mode setting and operating status

The HALT mode is set by executing the HALT instruction. It can be set when using either the main system clock or the subsystem clock.

The operating status in the HALT mode is described below.

**Table 23-1. HALT Mode Operating Status**

Setting of HALT Mode		On Execution of HALT Instruction During Main System Clock Operation		On Execution of HALT Instruction during Subsystem Clock Operation	
		Without subsystem clock <sup>Note 1</sup>	With subsystem clock <sup>Note 2</sup>	When main system clock continues oscillation	When main system clock stops oscillation
Item					
Clock generator		Both main system and subsystem clocks can be oscillated. Clock supply to the CPU stops.			
CPU		Operation stops			
Ports (output latches)		Status before HALT mode setting is held			
16-bit timer/event counter		Operable		Operable when watch timer output is selected as count clock ( $f_{XT}$ is selected as count clock of watch timer) or when T100 is selected	
8-bit timer/event counter		Operable		Operable when T11 or T12 is selected as count clock	
Watch timer		Operable when $f_{XX}/2^7$ is selected as count clock	Operable	Operable when $f_{XT}$ is selected as count clock	
Watchdog timer		Operable		Operation stops	
A/D converter		Operable			Operation stops
D/A converter		Operable			
Real-time output port		Operable			
Serial interface	Other than automatic transmit/receive function	Operable			Operable when external $\overline{SCK}$ is used
	Automatic transmit/receive function	Operation stops			
External interrupt requests	INTP0	INTP0 is operable when clock supplied for peripheral hardware is selected as sampling clock ( $f_{XX}/2^5$ , $f_{XX}/2^6$ , $f_{XX}/2^7$ )			Operation stops
	INTP1 to INTP5	Operable			
Bus line for external expansion	AD0 to AD7	High impedance			
	A0 to A15	Status before HALT mode setting is held			
	ASTB	Low level			
	$\overline{WR}$ , $\overline{RD}$	High level			
	$\overline{WAIT}$	High impedance			

**Notes** 1. Including when an external clock is not supplied

2. Including when an external clock is supplied

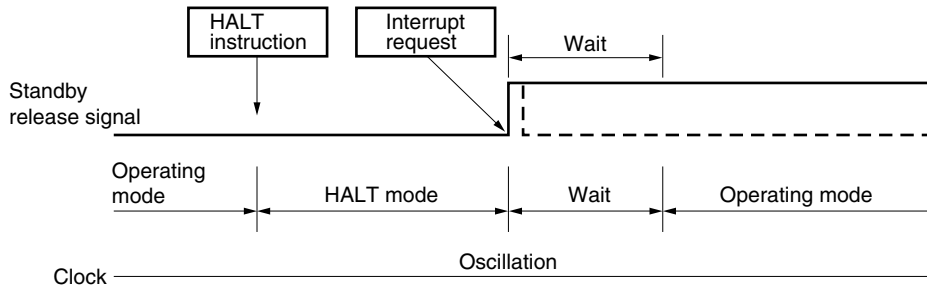
**(2) HALT mode release**

The HALT mode can be released by the following four types of sources.

**(a) Release by unmasked interrupt request**

If an unmasked interrupt request is generated, the HALT mode is released. If interrupt request acknowledgment is enabled, vectored interrupt servicing is carried out. If disabled, the next address instruction is executed.

**Figure 23-2. HALT Mode Release by Interrupt Request Generation**



**Remarks 1.** The broken lines indicate the case when the interrupt request which has released the standby status is acknowledged.

**2.** The wait time will be as follows:

- When the program branches to the vector table: 8 to 9 clocks
- When the program does not branch to the vector table: 2 to 3 clocks

**(b) Release by non-maskable interrupt request generation**

If a non-maskable interrupt request is generated, the HALT mode is released and vectored interrupt servicing is carried out irrespective of whether interrupt request acknowledgment is enabled or disabled.

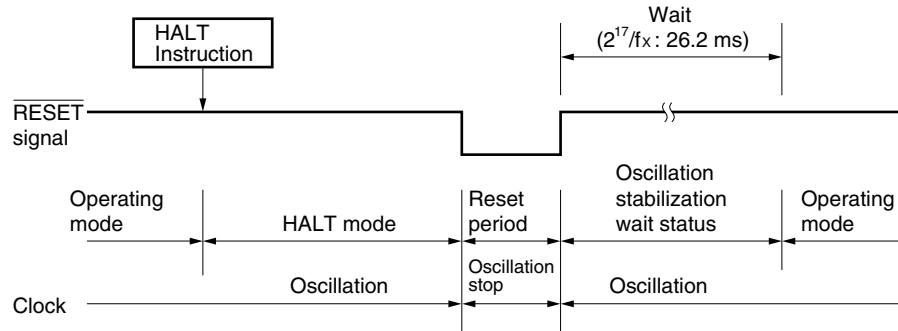
**(c) Release by unmasked test input**

If an unmasked test signal is input, the HALT mode is released, and the next address instruction of the HALT instruction is executed.

(d) Release by  $\overline{\text{RESET}}$  input

If the  $\overline{\text{RESET}}$  signal is input, the HALT mode is released. As is the case with a normal reset operation, the program is executed after branch to the reset vector address.

Figure 23-3. HALT Mode Release by  $\overline{\text{RESET}}$  Input



- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. Values in parentheses apply to operation with  $f_x = 5.0$  MHz.

Table 23-2. Operation After HALT Mode Release

Release Source	MK $\times\times$	PR $\times\times$	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 hold	
Non-maskable interrupt request	–	–	×	×	Interrupt servicing execution
Test input	0	–	×	×	Next address instruction execution
	1	–	×	×	HALT mode hold
$\overline{\text{RESET}}$ input	–	–	×	×	Reset processing

**Remark** ×: don't care

23.2.2 STOP mode

(1) STOP mode setting and operating status

The STOP mode is set by executing the STOP instruction. It can be set only when using the main system clock.

- Cautions**
1. When the STOP mode is set, the X2 pin is internally connected to V<sub>DD1</sub> via a pull-up resistor to minimize the leakage current at the crystal oscillator. Thus, do not use the STOP mode in a system where an external clock is used for the main system clock.
  2. 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. After the wait time set using the oscillation stabilization time select register (OSTS) elapses, the operating mode is set.

The operating status in the STOP mode is described below.

Table 23-3. STOP Mode Operating Status

Setting of STOP Mode		With Subsystem Clock	Without Subsystem Clock
Item			
Clock generator		Only main system clock stops oscillation	
CPU		Operation stops	
Ports (output latches)		Status before STOP mode setting is held	
16-bit timer/event counter		Operable when watch timer output is selected as count clock (f <sub>XT</sub> is selected as count clock of watch timer)	Operation stops
8-bit timer/event counter		Operable when TI1 and TI2 are selected for the count clock	
Watch timer		Operable when f <sub>XT</sub> is selected for the count clock	Operation stops
Watchdog timer		Operation stops	
A/D converter			
D/A converter		Operable	
Real-time output port		Operable when external trigger is used or TI1 and TI2 are selected for the 8-bit timer/event counter count clock	
Serial interface	Other than automatic transmit/receive function and UART	Operable when externally supplied clock is specified as the serial clock	
	Automatic transmit/receive function and UART	Operation stops	
External interrupt requests	INTP0	Not operable	
	INTP1 to INTP5	Operable	
Bus line for external expansion	AD0 to AD7	High impedance	
	A0 to A15	Status before STOP mode setting is held	
	ASTB	Low level	
	$\overline{WR}$ , $\overline{RD}$	High level	
	$\overline{WAIT}$	High impedance	

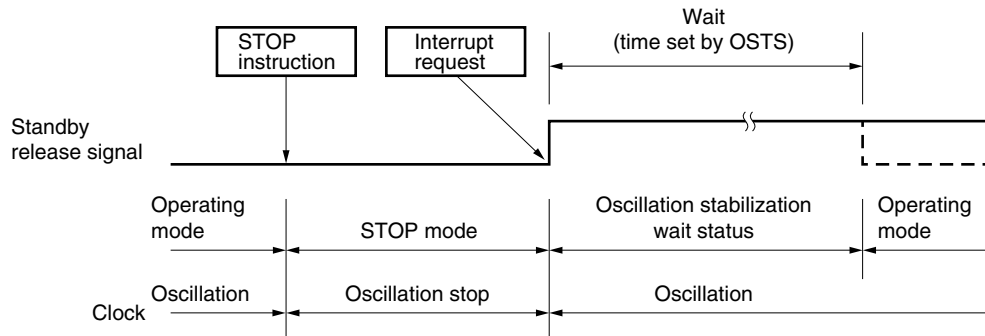
**(2) STOP mode release**

The STOP mode can be released by the following three types of sources.

**(a) Release by unmasked interrupt request**

If an unmasked interrupt request is generated, the STOP mode is released. If interrupt request acknowledgment is enabled after the lapse of oscillation stabilization time, vectored interrupt servicing is carried out. If interrupt request acknowledgment is disabled, the next address instruction is executed.

**Figure 23-4. STOP Mode Release by Interrupt Request Generation**



**Remark** The broken lines indicate the case when the interrupt request which has released the standby status is acknowledged.

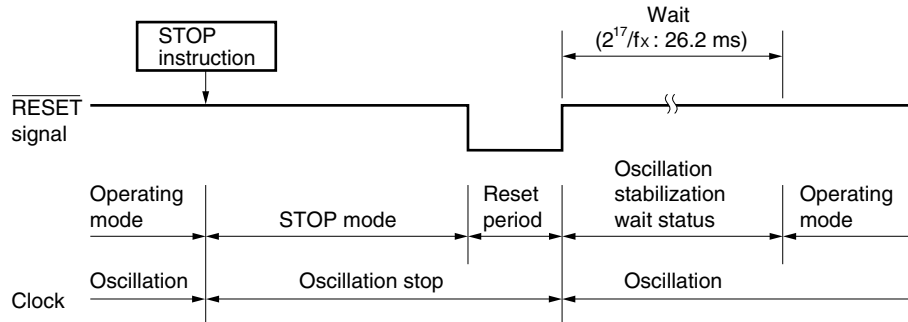
**(b) Release by unmasked test input**

If an unmasked test signal is input, the STOP mode is released. After the lapse of oscillation stabilization time, the instruction at the next address of the STOP instruction is executed.

(c) Release by  $\overline{\text{RESET}}$  input

If the  $\overline{\text{RESET}}$  signal is input, the STOP mode is released and after the lapse of oscillation stabilization time, a reset operation is carried out.

Figure 23-5. STOP Mode Release by  $\overline{\text{RESET}}$  Input



- Remarks**
1.  $f_x$ : Main system clock oscillation frequency
  2. Values in parentheses apply to operation with  $f_x = 5.0 \text{ MHz}$ .

Table 23-4. Operation After STOP Mode Release

Release Source	MK <sub>xx</sub>	PR <sub>xx</sub>	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 hold
Test input	0	–	×	×	Next address instruction execution
	1	–	×	×	STOP mode hold
$\overline{\text{RESET}}$ input	–	–	×	×	Reset processing

**Remark** ×: don't care



## CHAPTER 24 RESET FUNCTION

### 24.1 Reset Function

The following two operations are available to generate a reset signal.

- (1) External reset input by  $\overline{\text{RESET}}$  pin
- (2) Internal reset by watchdog timer program loop time detection

The external reset and internal reset have no functional differences. In both cases, program execution starts at the address at 0000H and 0001H by  $\overline{\text{RESET}}$  input.

When a low level is input to the  $\overline{\text{RESET}}$  pin or the watchdog timer overflows, a reset is applied and each hardware unit is set to the status shown in Table 24-1. Each pin is high impedance during reset input or during the oscillation stabilization time just after reset is released.

When a high level is input to the  $\overline{\text{RESET}}$  pin, the reset is cleared and program execution starts after the lapse of oscillation stabilization time ( $2^{17}/f_x$ ). The reset applied by watchdog timer overflow is automatically cleared after a reset and program execution starts after the lapse of oscillation stabilization time ( $2^{17}/f_x$ ) (see **Figures 24-2 to 24-4**).

- Cautions**
1. For an external reset, input a low level to the  $\overline{\text{RESET}}$  pin for 10  $\mu\text{s}$  or more.
  2. During reset input, main system clock oscillation remains stopped but subsystem clock oscillation continues.
  3. When the STOP mode is cleared by reset, the STOP mode contents are held during reset input. However, the port pin becomes high impedance.

Figure 24-1. Reset Function Block Diagram

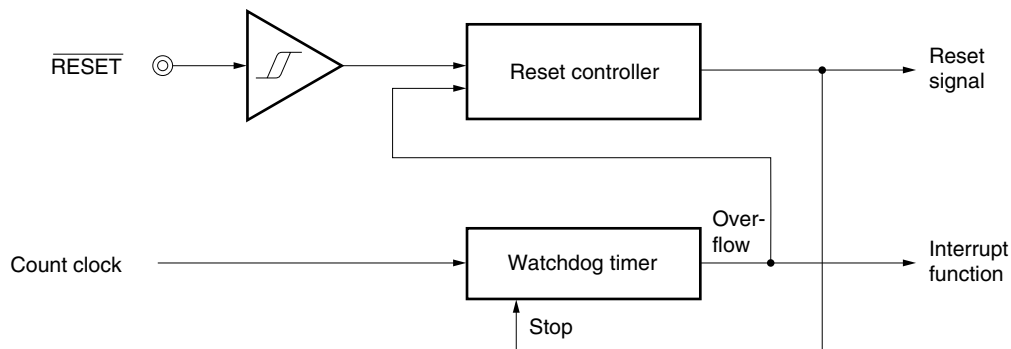


Figure 24-2. Reset Timing by  $\overline{\text{RESET}}$  Input

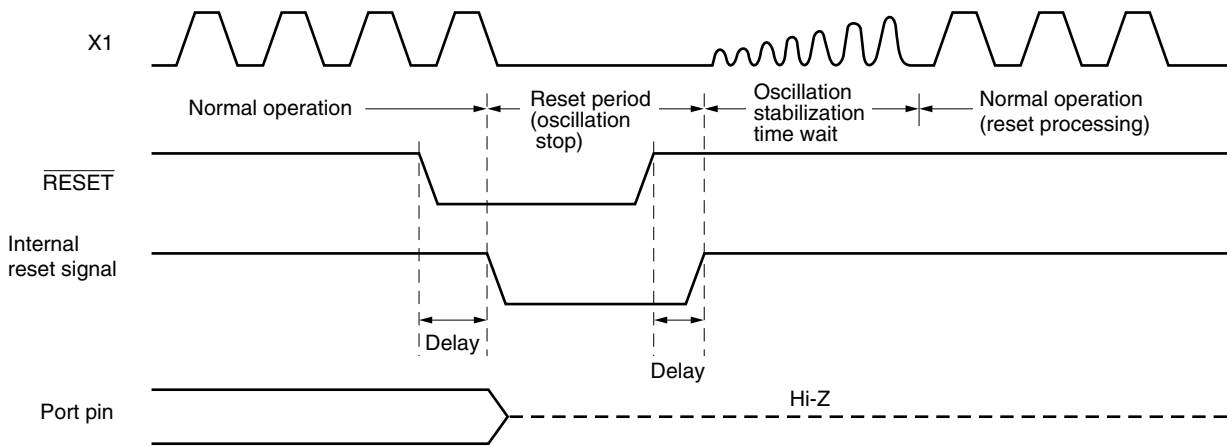


Figure 24-3. Reset Timing due to Watchdog Timer Overflow

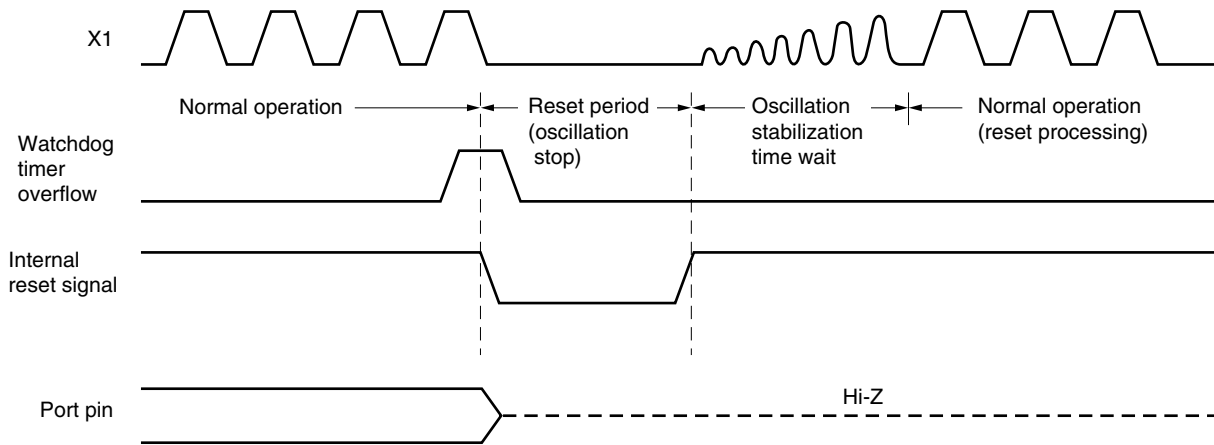
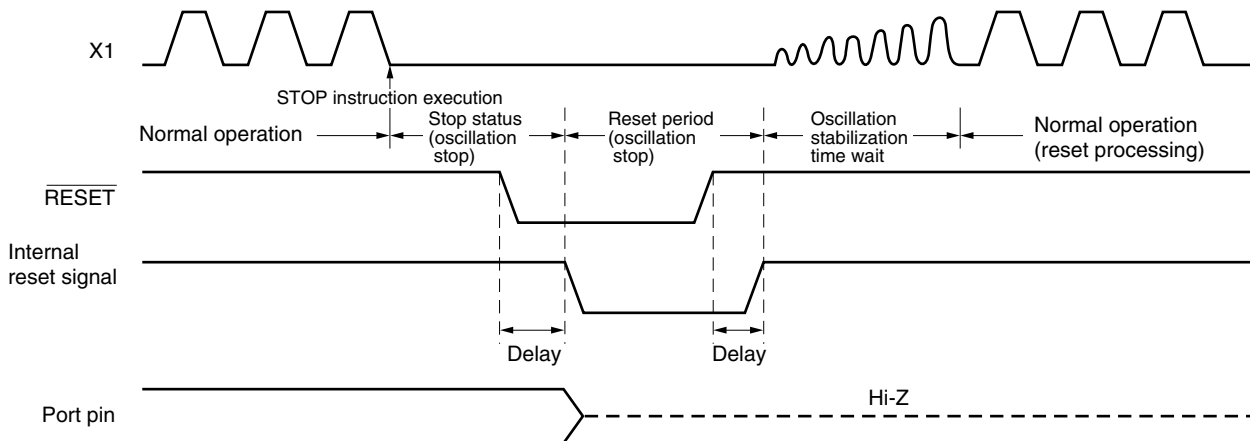


Figure 24-4. Reset Timing by  $\overline{\text{RESET}}$  Input in STOP Mode



**Table 24-1. Hardware Status After Reset (1/2)**

Hardware		Status After Reset
Program counter (PC) <sup>Note 1</sup>		The contents of the reset vector tables (0000H and 0001H) are set.
Stack pointer (SP)		Undefined
Program status word (PSW)		02H
RAM	Data memory	Undefined <sup>Note 2</sup>
	General register	Undefined <sup>Note 2</sup>
Ports (output latches)	Ports 0 to 3, 7, 12, 13 (P0 to P3, P7, P12, P13)	00H
	Ports 4 to 6 (P4 to P6)	Undefined
Port mode registers (PM0 to PM3, PM5 to PM7, PM12, PM13)		FFH
Pull-up resistor option registers (PUOH, PUOL)		00H
Processor clock control register (PCC)		04H
Oscillation mode select register (OSMS)		00H
Internal memory size switching register (IMS)		<b>Note 3</b>
Internal expansion RAM size switching register (IXS) <sup>Note 4</sup>		0AH
Memory expansion mode register (MM)		10H
Oscillation stabilization time select register (OSTS)		04H
16-bit timer/event counter	Timer register (TM0)	00H
	Capture/compare registers (CR00, CR01)	Undefined
	Clock select register (TCL0)	00H
	Mode control register (TMC0)	00H
	Capture/compare control register 0 (CRC0)	04H
	Output control register (TOC0)	00H
8-bit timer/event counters 1 and 2	Timer register (TM1, TM2)	00H
	Compare registers (CR10, CR20)	Undefined
	Clock select register (TCL1)	00H
	Mode control registers (TMC1)	00H
	Output control register (TOC1)	00H

- Notes**
1. During reset input or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.
  2. If the reset signal is input in the standby mode, the status before reset is retained even after reset.
  3. The values after reset depend on the product.  
 $\mu$ PD780053, 780053Y: C6H,  $\mu$ PD780054, 780054Y: C8H,  $\mu$ PD780055, 780055Y: CAH,  
 $\mu$ PD780056, 780056Y: CCH,  $\mu$ PD780058, 780058B, 780058BY: CFH,  $\mu$ PD78F0058, 78F0058Y: CFH
  4. Provided only in the  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y.

Table 24-1. Hardware Status After Reset (2/2)

	Hardware	Status After Reset
Watch timer	Mode control register (TMC2)	00H
	Clock select register (TCL2)	00H
Watchdog timer	Mode register (WDTM)	00H
	Clock select register (TCL3)	88H
Serial interface	Shift registers (SIO0, SIO1)	Undefined
	Mode registers (CSIM0, CSIM1, CSIM2)	00H
	Serial bus interface control register (SBIC)	00H
	Slave address register (SVA)	Undefined
	Automatic data transmit/receive control register (ADTC)	00H
	Automatic data transmit/receive address pointer (ADTP)	00H
	Automatic data transmit/receive interval specification register (ADTI)	00H
	Asynchronous serial interface mode register (ASIM)	00H
	Asynchronous serial interface status register (ASIS)	00H
	Baud rate generator control register (BRGC)	00H
	Serial interface pin select register (SIPS)	00H
	Transmit shift register (TXS)	FFH
	Receive buffer register (RXB)	
	Interrupt timing specification register (SINT)	00H
	A/D converter	Mode register (ADM)
Conversion result register (ADCR)		Undefined
Input select register (ADIS)		00H
D/A converter	Mode register (DAM)	00H
	Conversion value setting registers (DACs0, DACs1)	00H
Real-time output port	Mode register (RTPM)	00H
	Control register (RTPC)	00H
	Buffer registers (RTBL, RTBH)	00H
ROM correction <sup>Note</sup>	Correction address registers (CORAD0, CORAD1)	0000H
	Correction control register (CORCN)	00H
Interrupts	Request flag registers (IF0L, IF0H, IF1L)	00H
	Mask flag registers (MK0L, MK0H, MK1L)	FFH
	Priority specification flag registers (PR0L, PR0H, PR1L)	FFH
	External interrupt mode registers (INTM0, INTM1)	00H
	Key return mode register (KRM)	02H
	Sampling clock select register (SCS)	00H

**Note** Provided only in the  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y.

## CHAPTER 25 ROM CORRECTION

### 25.1 ROM Correction Function

★ The  $\mu$ PD780058, 780058B, 780058BY, 78F0058, 78F0058Y can replace part of a program in the mask ROM or flash memory with a program in the internal expansion RAM.

Instruction bugs found in the mask ROM or flash memory can be avoided, and program flow can be changed by using the ROM correction function.

The ROM correction function can be used to correct two places (max.) of the internal ROM or flash memory (program).

- ★ **Cautions**
1. ROM correction can be used only for the  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y.
  2. ROM correction function cannot be emulated by the in-circuit emulator (IE-78000-R, IE-78000-R-A, IE-78K0-NS, IE-78K0-NS-A, IE-78001-R-A).

### 25.2 ROM Correction Configuration

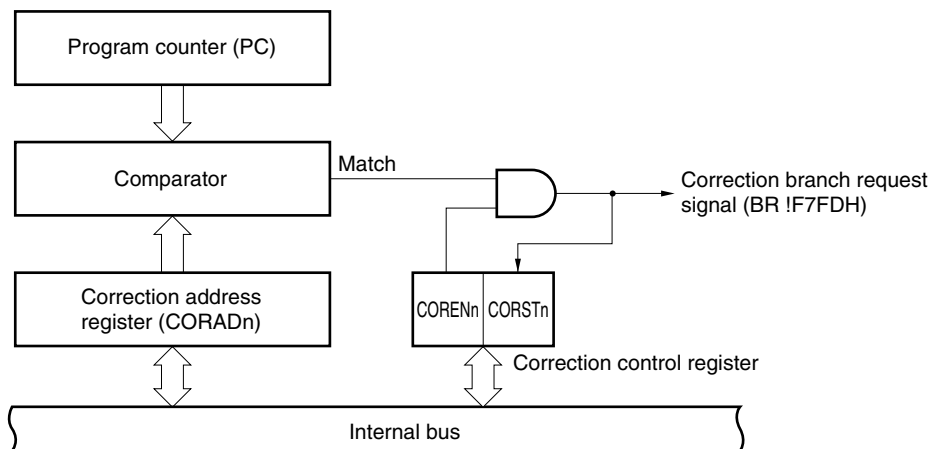
The ROM correction function consists of the following hardware.

**Table 25-1. ROM Correction Configuration**

Item	Configuration
Registers	Correction address registers 0, 1 (CORAD0, CORAD1)
Control register	Correction control register (CORCN)

Figure 25-1 shows a block diagram of the ROM correction function.

**Figure 25-1. ROM Correction Block Diagram**



**Remark** n = 0, 1



**(1) Correction address registers 0, 1 (CORAD0, CORAD1)**

These registers set the start address (correction address) of the instruction(s) to be corrected in the mask ROM or flash memory.

The ROM correction function corrects two places (max.) of the program. Addresses are set to two registers, CORAD0 and CORAD1. If only one place needs to be corrected, set the address to either of the registers. CORAD0 and CORAD1 are set with a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CORAD0 and CORAD1 to 0000H.

**Figure 25-2. Format of Correction Address Registers 0 and 1**

Symbol	15	0	Address	After reset	R/W
CORAD0			FF38H/FF39H	0000H	R/W
CORAD1			FF3AH/FF3BH	0000H	R/W

**Cautions** 1. Set CORAD0 and CORAD1 when bit 1 (COREN0) and bit 3 (COREN1) of the correction control register (CORCN: See Figure 25-3) are 0.

2. Only addresses where operation codes are stored can be set to CORAD0 and CORAD1.

3. Do not set the following addresses to CORAD0 and CORAD1.

- Address value in table area of table reference instruction (CALLT instruction): 0040H to 007FH
- Address value in vector table area: 0000H to 003FH

**(2) Comparator**

The comparator continuously compares the correction address value set in correction address registers 0 and 1 (CORAD0, CORAD1) with the fetch address value. When bit 1 (COREN0) or bit 3 (COREN1) of the correction control register (CORCN) is 1 and the correction address matches the fetch address value, the correction branch request signal (BR !F7FDH) is generated from the ROM correction circuit.

### 25.3 ROM Correction Control Registers

The ROM correction function is controlled by the correction control register (CORCN).

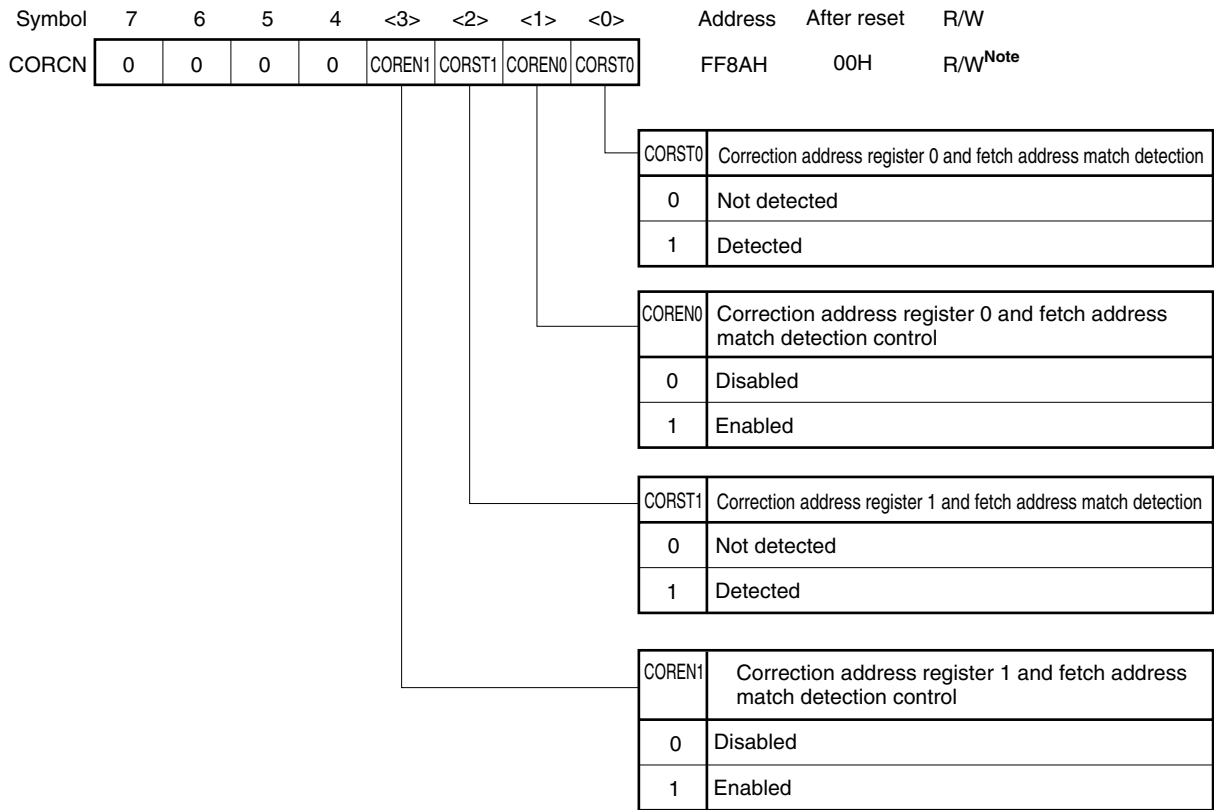
#### (1) Correction control register (CORCN)

This register controls whether or not the correction branch request signal is generated when the fetch address matches the correction address set in correction address registers 0 and 1. The correction control register consists of correction enable flags (COREN0, COREN1) and correction status flags (CORST0, CORST1). The correction enable flags enable or disable the comparator match detection signal, and correction status flags show that the values match.

CORCN is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input clears CORCN to 00H.

**Figure 25-3. Format of Correction Control Register**



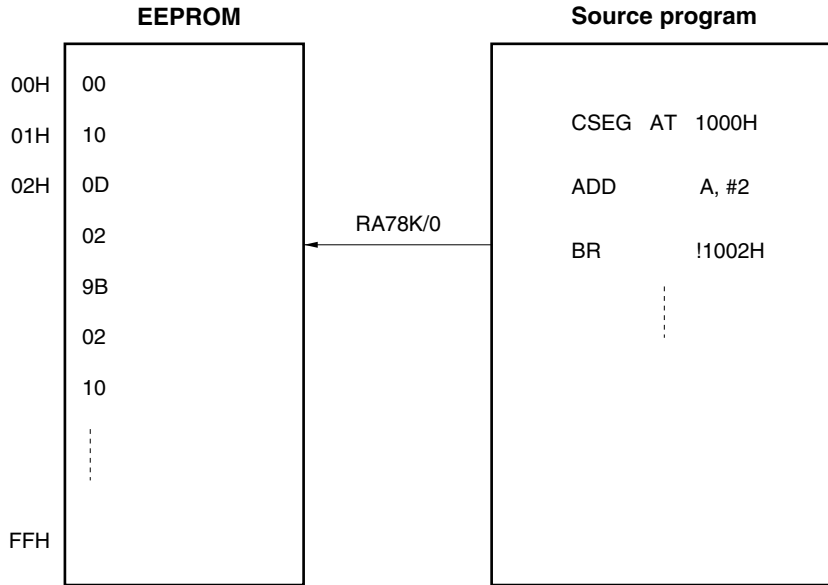
**Note** Bits 0 and 2 are read-only bits.

### 25.4 ROM Correction Application

- (1) Store the correction address and instruction after correction (patch program) to nonvolatile memory (such as EEPROM™) outside the microcontroller.

When two places should be corrected, store the branch destination judgment program as well. The branch destination judgment program checks which one of the addresses set to correction address registers 0 and 1 (CORAD0 or CORAD1) generates the correction branch.

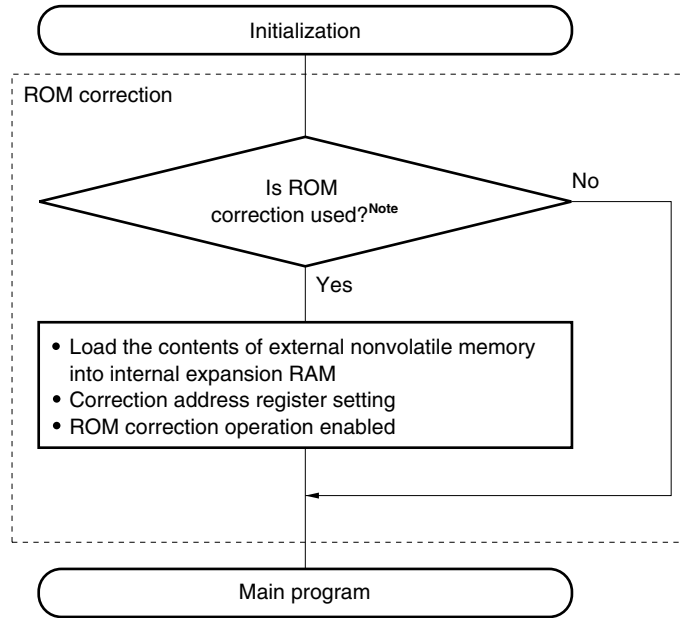
**Figure 25-4. Example of Storing to EEPROM (When One Place Is Corrected)**





- (2) Assemble in advance the initialization routine as shown in Figure 25-5 to correct the program.

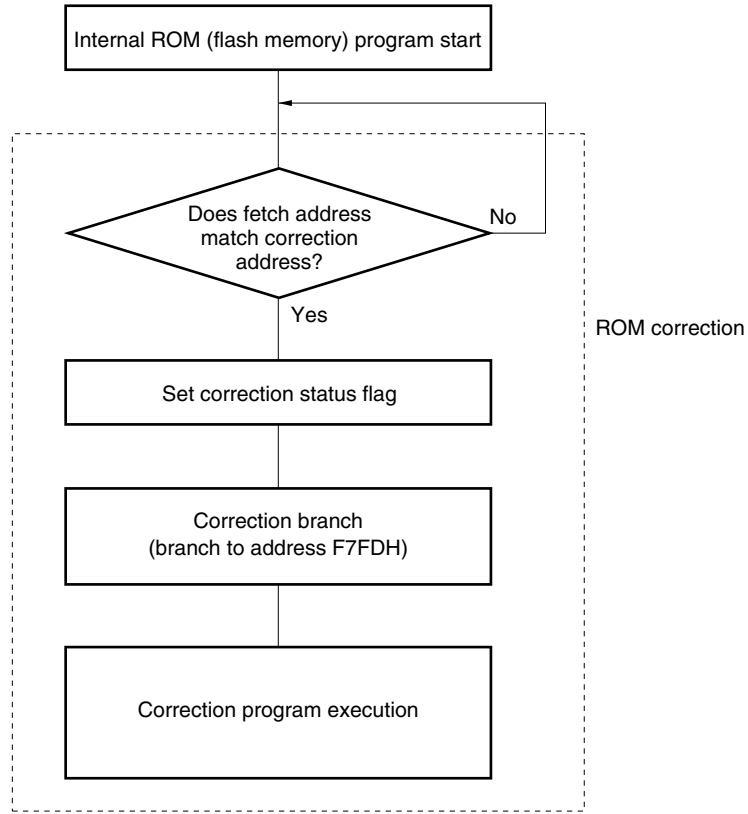
**Figure 25-5. Initialization Routine**



**Note** Whether the ROM correction function is used or not should be judged by the port input level. For example, when the P20 input level is high, ROM correction is used, otherwise, it is not used.

- (3) After reset, store the contents that were previously stored in the external nonvolatile memory by the user initialization routine for ROM correction to internal expansion RAM (see **Figure 25-5**). Set the start address of the instruction to be corrected to CORAD0 and CORAD1, and set bits 1 and 3 (COREN0, COREN1) of the correction control register (CORCN) to 1.
- (4) Set the entire-space branch instruction (BR !addr16) to the specified address (F7FDH) of the internal expansion RAM using the main program.
- (5) After the main program is started, the fetch address value and the values set in CORAD0 and CORAD1 are continuously compared by the comparator in the ROM correction circuit. When these values match, the correction branch request signal is generated. Simultaneously the corresponding correction status flag (CORST0 or CORST1) is set to 1.
- (6) Branch to the address F7FDH via the correction branch request signal.
- (7) Branch to the internal expansion RAM address set by the main program via the entire-space branch instruction of the address F7FDH.
- (8) When one place is corrected, the correction program is executed. When two places are corrected, the correction status flag is checked by the branch destination judgment program, and the program branches to the correction program.

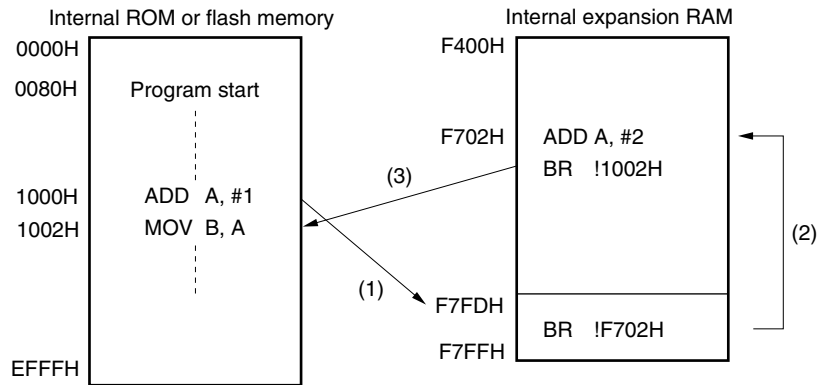
Figure 25-6. ROM Correction Operation



## 25.5 ROM Correction Usage Example

An example of ROM correction when the instruction at address 1000H “ADD A, #1” is changed to “ADD A, #2” is shown below.

**Figure 25-7. ROM Correction Usage Example**

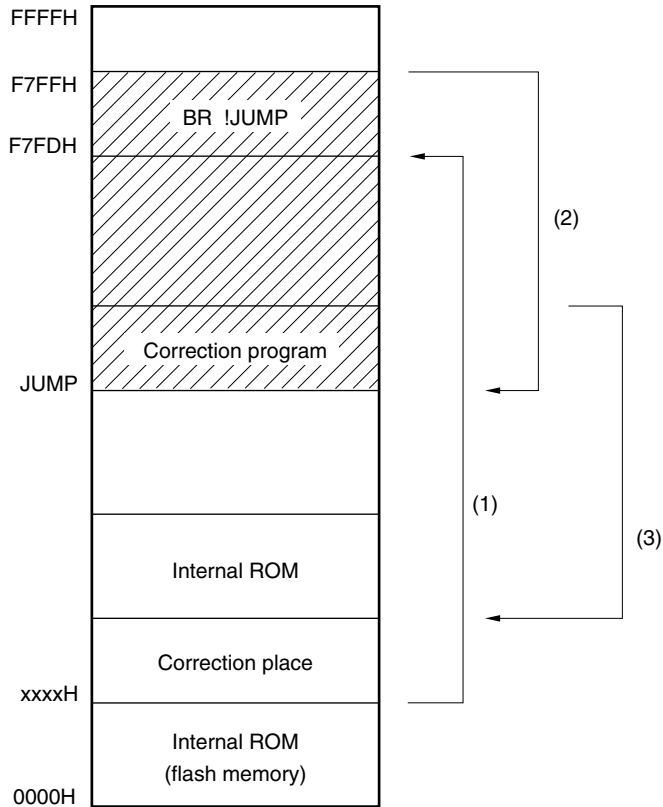


- (1) The program branches to address F7FDH when the preset value 1000H in the correction address register matches the fetch address value after the main program is started.
- (2) The program branches to any address (address F702H in this example) by setting the entire-space branch instruction (BR !addr16) to address F7FDH by the main program.
- (3) The program returns to the internal ROM (flash memory) program after executing the substitute instruction ADD A, #2.

25.6 Program Execution Flow

Figures 25-8 and 25-9 show the program transition diagrams when the ROM correction is used.

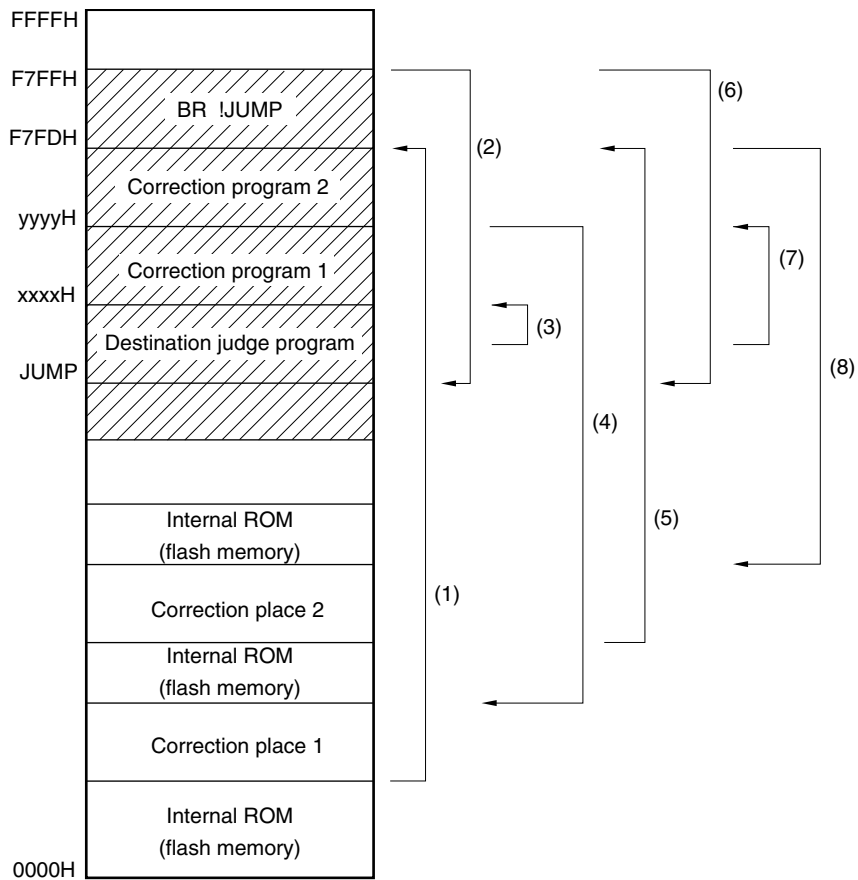
Figure 25-8. Program Transition Diagram (When One Place Is Corrected)



- (1) The program branches to address F7FDH when the fetch address matches the correction address
- (2) The program branches to the correction program
- (3) The program returns to the internal ROM (flash memory) program

**Remark** Shaded area: Internal expansion RAM  
 JUMP: Correction program start address

Figure 25-9. Program Transition Diagram (When Two Places Are Corrected)



- (1) The program branches to address F7FDH when the fetch address matches the correction address
- (2) The program branches to the branch destination judgment program
- (3) The program branches to correction program 1 via the branch destination judgment program (BTCLR !CORST0, \$xxxxH)
- (4) The program returns to the internal ROM (flash memory) program
- (5) The program branches to address F7FDH when the fetch address matches the correction address
- (6) The program branches to the branch destination judgment program
- (7) The program branches to correction program 2 via the branch destination judgment program (BTCLR !CORST1, \$yyyyH)
- (8) The program returns to the internal ROM (flash memory) program

**Remark** Shaded Area: Internal expansion RAM  
 JUMP: Destination judge program start address

## 25.7 ROM Correction Cautions

- (1) Address values set in correction address registers 0 and 1 (CORAD0, CORAD1) must be addresses where instruction codes are stored.
- (2) Correction address registers 0 and 1 (CORAD0, CORAD1) should be set when the correction enable flag (COREN0, COREN1) is 0 (when the correction branch is in the disabled state). If an address is set to CORAD0 or CORAD1 when COREN0 or COREN1 is 1 (when the correction branch is in the enabled state), the correction branch may start with a different address from the set address value.
- (3) Do not set the address value of an instruction immediately after the instruction that sets the correction enable flag (COREN0, COREN1) to 1, to correction address register 0 or 1 (CORAD0, CORAD1); otherwise the correction branch may not start.
- (4) Do not set the address value in the table area of the table reference instruction (CALLT instruction) (0040H to 007FH), and the address value in the vector table area (0000H to 003FH) to correction address registers 0 and 1 (CORAD0, CORAD1).
- (5) Do not set two addresses immediately after the instructions shown below to correction address registers 0 and 1 (CORAD0, CORAD1). (That is, when the mapped terminal address of these instructions is N, do not set the address values of N + 1 and N + 2.)
  - RET
  - RETI
  - RETB
  - BR \$addr16
  - STOP
  - HALT

## CHAPTER 26 $\mu$ PD78F0058, 78F0058Y

The  $\mu$ PD78F0058 and 78F0058Y have flash memory whose contents can be written, erased, rewritten with the device mounted on a PC board. Table 26-1 lists the differences between the flash memory versions ( $\mu$ PD78F0058 and 78F0058Y) and the mask ROM versions ( $\mu$ PD780053, 780054, 780055, 780056, 780058, 780058B, 780053Y, 780054Y, 780055Y, 780056Y, and 780058BY).

**Table 26-1. Differences Between  $\mu$ PD78F0058, 78F0058Y and Mask ROM Versions**

Item	$\mu$ PD78F0058	$\mu$ PD78F0058Y	Mask ROM Versions	
			$\mu$ PD780058 Subseries	$\mu$ PD780058Y Subseries
Internal ROM structure	Flash memory		Mask ROM	
Internal ROM capacity	60 KB		$\mu$ PD780053, 780053Y: 24 KB $\mu$ PD780054, 780054Y: 32 KB $\mu$ PD780055, 780055Y: 40 KB $\mu$ PD780056, 780056Y: 48 KB $\mu$ PD780058, 780058B, 780058BY: 60 KB	
Internal expansion RAM capacity	1,024 bytes		$\mu$ PD780053, 780053Y: None $\mu$ PD780054, 780054Y: None $\mu$ PD780055, 780055Y: None $\mu$ PD780056, 780056Y: None $\mu$ PD780058, 780058B, 780058BY: 1,024 bytes	
Internal ROM capacity changeable/not changeable using internal memory size switching register (IMS)	Changeable <sup>Note 1</sup>		Not changeable	
Internal expansion RAM capacity changeable/not changeable using internal expansion RAM size switching register (IXS)	Changeable <sup>Note 2</sup>		Not changeable	
★ Supply voltage	$V_{DD} = 2.7^{\text{Note 3}}$ to 5.5 V		$V_{DD} = 1.8$ to 5.5 V	
IC pin	Not provided		Provided	
$V_{PP}$ pin	Provided		Not provided	
P60 to P63 pin mask option with on-chip pull-up resistors	Not provided		Provided	
★ Serial interface (SBI)	Provided	Not provided	Provided	Not provided
★ Serial interface (I <sup>2</sup> C)	Not provided	Provided	Not provided	Provided

- Notes**
1. Flash memory is set to 60 KB by  $\overline{\text{RESET}}$  input.
  2. Internal expansion RAM is set to 1,024 bytes by  $\overline{\text{RESET}}$  input.
  3.  $V_{DD} = 2.2$  V can also be supplied. Contact an NEC Electronics sales representative for details.

**Caution** There are differences in noise immunity and noise radiation between the flash memory and mask ROM versions. When pre-producing an application set with the flash memory version and then mass-producing it with the mask ROM version, be sure to conduct sufficient evaluations for the commercial samples (not engineering samples) of the mask ROM version.

**Remark** Only the  $\mu$ PD780058, 780058B, 78F0058, 780058BY, and 78F0058Y are provided with an internal expansion RAM size switching register.

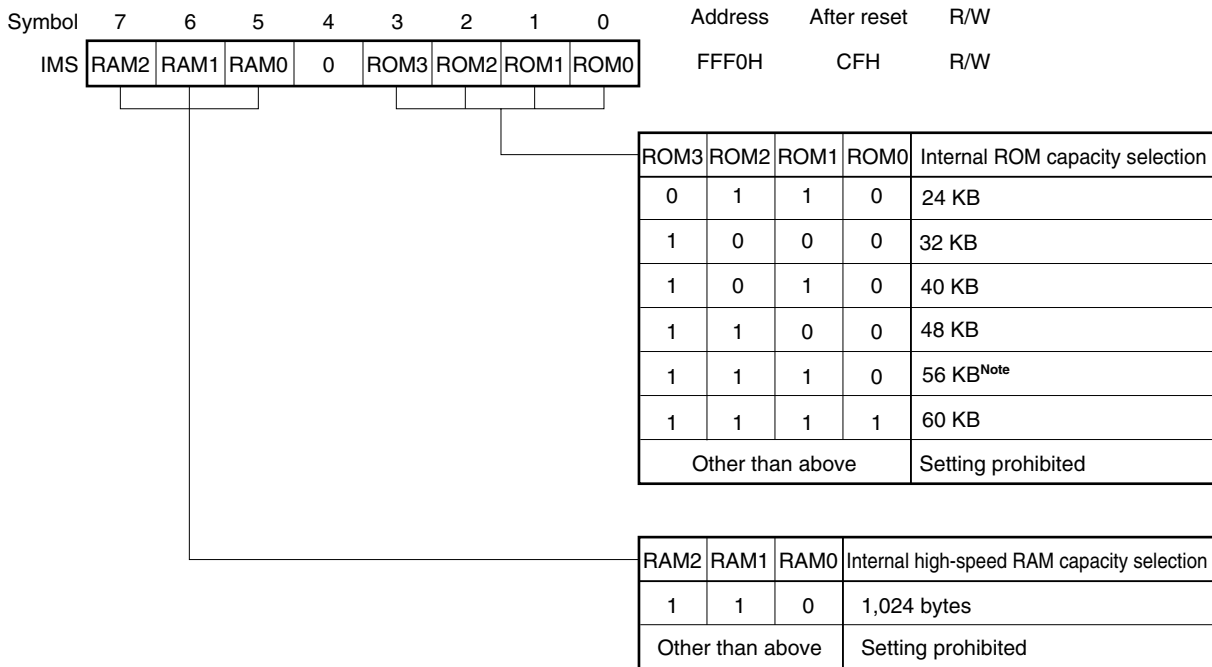
## 26.1 Internal memory Size Switching Register

The  $\mu$ PD78F0058 and 78F0058Y allow users to define the internal ROM size using the internal memory size switching register (IMS), so that the same memory mapping as that of a mask ROM version with a different-size internal ROM is possible.

IMS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$  input sets IMS to CFH.

**Figure 26-1. Format of Memory Size Switching Register**



**Note** When using the external device expansion function of the  $\mu$ PD780058, 780058B, 780058BY, 78F0058, and 78F0058Y, set the internal ROM capacity to 56 KB or less.

The IMS settings to give the same memory map as mask ROM versions are shown in Table 26-2.

**Table 26-2. Internal Memory Size Switching Register Setting Values**

Target Mask ROM Version	IMS Setting Value
$\mu$ PD780053, 780053Y	C6H
$\mu$ PD780054, 780054Y	C8H
$\mu$ PD780055, 780055Y	CAH
$\mu$ PD780056, 780056Y	CCH
$\mu$ PD780058, 780058B, 780058BY	CFH



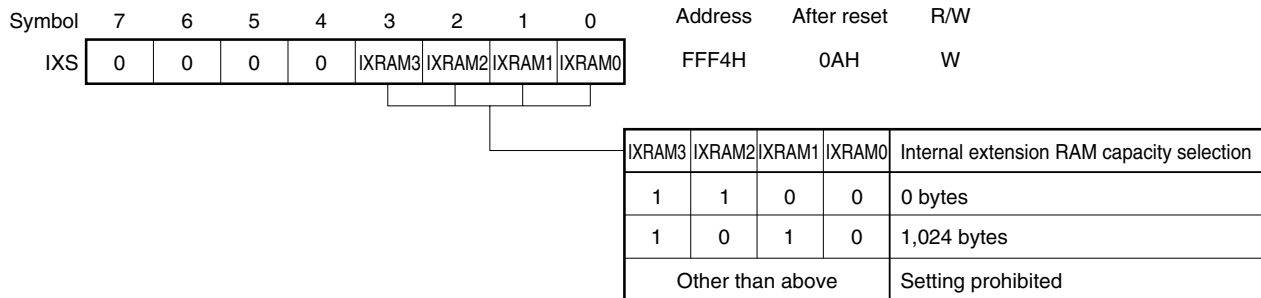
## 26.2 Internal Expansion RAM Size Switching Register

The  $\mu$ PD78F0058 and 78F0058Y allow users to define the internal expansion RAM size by using the internal expansion RAM size switching register (IXS), so that the same memory mapping as that of a mask ROM version with a different-size internal expansion RAM is possible.

IXS is set with an 8-bit memory manipulation instruction.

RESET input sets IXS to 0AH.

**Figure 26-2. Format of Internal Expansion RAM Size Switching Register**



The IXS settings that give the same memory map as the mask ROM versions are shown in Table 26-3.

**Table 26-3. Internal Expansion RAM Size Switching Register Setting Values**

Target Mask ROM Version	IXS Setting Value
$\mu$ PD780053, 780053Y	0CH
$\mu$ PD780054, 780054Y	
$\mu$ PD780055, 780055Y	
$\mu$ PD780056, 780056Y	
$\mu$ PD780058, 780058B, 780058BY	0AH

**Remark** If a program for the  $\mu$ PD78F0058 or 78F0058Y which includes “MOV IXS, #0CH” is implemented with the  $\mu$ PD780053, 780053Y, 780054, 780054Y, 780055, 780055Y, 780056, or 780056Y, this instruction is ignored and causes no malfunction.

## ★ 26.3 Flash Memory Characteristics

Flash memory programming is performed by connecting a dedicated flash programmer (Flashpro III (part no. FL-PR3, PG-FP3)/Flashpro IV (part no. FL-PR4, PG-FP4)) to the target system with the flash memory mounted on the target system (on-board). A flash memory writing adapter (program adapter), which is a target board used exclusively for programming, is also provided.

**Remark** FL-PR3, FL-PR4, and the program adapter are products of Naito Densai Machida Mfg. Co., Ltd. (TEL +81-45-475-4191).

Programming using flash memory has the following advantages.

- Software can be modified after the microcontroller is solder-mounted on the target system.
- Distinguishing software facilities low-quantity, varied model production
- Easy data adjustment when starting mass production

### 26.3.1 Programming environment

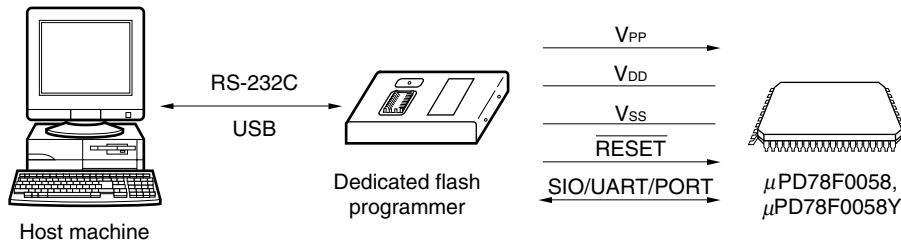
The following shows the environment required for  $\mu$ PD78F0058, 78F0058Y flash memory programming.

When Flashpro III (part no. FL-PR3, PG-FP3) or Flashpro IV (part no. FL-PR4, PG-FP4) is used as a dedicated flash programmer, a host machine is required to control the dedicated flash programmer. Communication between the host machine and flash programmer is performed via RS-232C/USB (Rev. 1.1).

For details, refer to the manuals for Flashpro III/Flashpro IV.

**Remark** USB is supported by Flashpro IV only.

**Figure 26-3. Environment for Writing Program to Flash Memory**



### 26.3.2 Communication mode

Use the communication mode shown in Table 26-4 to perform communication between the dedicated flash programmer and  $\mu$ PD78F0058, 78F0058Y.

**Table 26-4. Communication Mode List**

Communication Mode	TYPE Setting <sup>Note 1</sup>					Pins Used	Number of $V_{PP}$ Pulses
	COMM PORT	SIO Clock	CPU CLOCK	Flash Clock	Multiple Rate		
3-wire serial I/O	SIO ch-0 (3 wired, sync.)	100 Hz to 1.25 MHz <sup>Note 2</sup>	Optional	1 to 5 MHz <sup>Note 2</sup>	1.0	P27/ $\overline{\text{SCK0}}$ /SCL P26/SO0/SB1/SDA1 P25/SI0/SB0/SDA0	0
	SIO ch-1 (3 wired, sync.)					P22/ $\overline{\text{SCK1}}$ P21/SO1 P20/SI1	1
	SIO ch-2 (3 wired, sync.)					P72/ $\overline{\text{SCK2}}$ /ASCK P71/SO2/TxD0 P70/SI2/RxD0	2
UART (UART0)	UART ch-0 (Async.)	4,800 to 76,800 bps <sup>Notes 2, 3</sup>	Optional	1 to 5 MHz <sup>Note 2</sup>	1.0	P71/SO2/TxD0 P70/SI2/RxD0	8
	UART ch-1 (Async.)					P23/TxD1 P24/RxD1	9
Pseudo 3-wire serial I/O	Port A (Pseudo-3 wired)	100 Hz to 1 kHz	Optional	1 to 5 MHz <sup>Note 2</sup>	1.0	P32/TO2 (Serial clock I/O) P31/TO1 (Serial data output) P30/TO0 (Serial data input)	12

- Notes**
1. Selection items for TYPE settings on the dedicated flash programmer (Flashpro III (part no. FL-PR3, PG-FP3)/Flashpro IV (part no. FL-PR4, PG-FP4)).
  2. The possible setting range differs depending on the voltage. For details, see **CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)**, **CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION ( $V_{DD} = 2.2$  V))**.
  3. Because factors other than the baud rate error, such as the signal waveform slew, also affect UART communication, thoroughly evaluate the slew as well as the baud rate error.

**Figure 26-4. Communication Mode Selection Format**

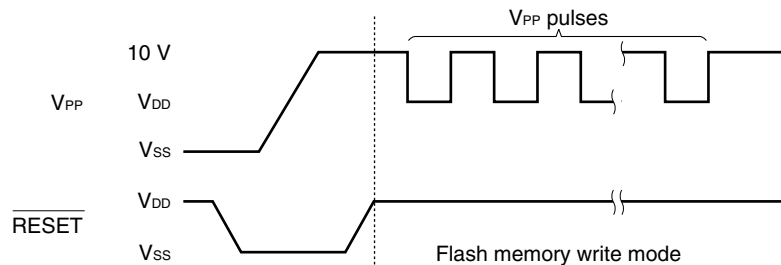
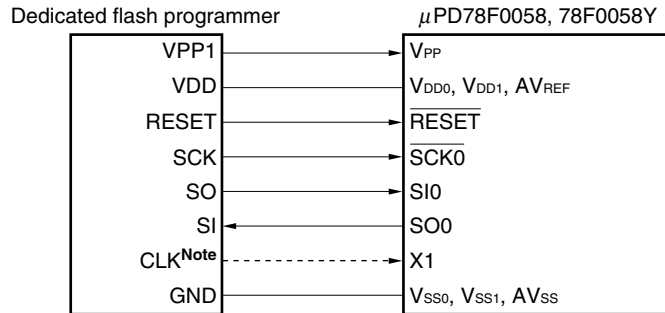
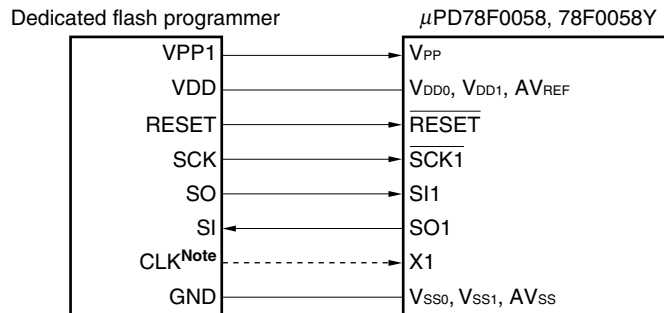


Figure 26-5. Example of Connection with Dedicated Flash Programmer (1/2)

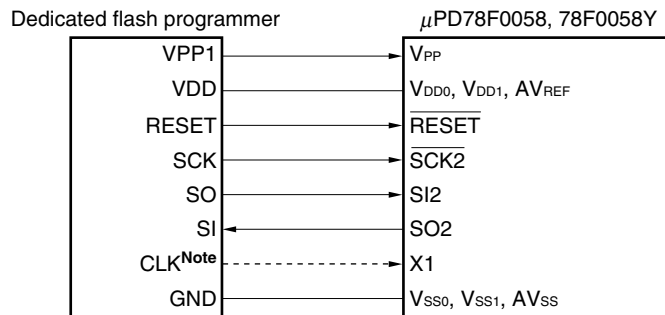
## (a) 3-wire serial I/O (SIO ch-0)



## (b) 3-wire serial I/O (SIO ch-1)



## (c) 3-wire serial I/O (SIO ch-2)

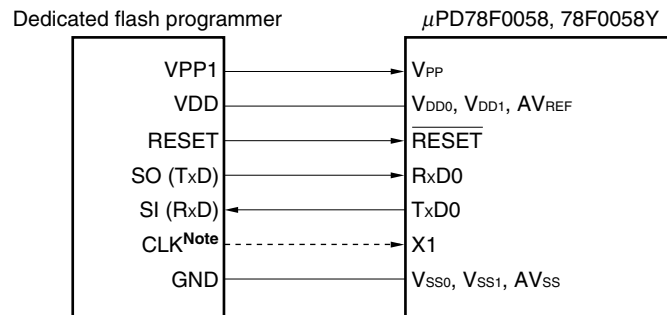


**Note** Connect this pin when the system clock is supplied from the dedicated flash programmer. If a resonator is already connected to the X1 pin, the CLK pin does not need to be connected.

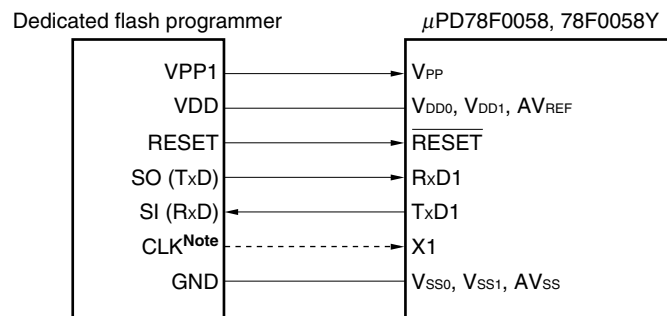
**Caution** The V<sub>DD0</sub> and V<sub>DD1</sub> pins, if already connected to the power supply, must be connected to the VDD pin of the dedicated flash programmer. When using the power supply connected to the V<sub>DD0</sub> and V<sub>DD1</sub> pins, supply voltage before starting programming.

Figure 26-5. Example of Connection with Dedicated Flash Programmer (2/2)

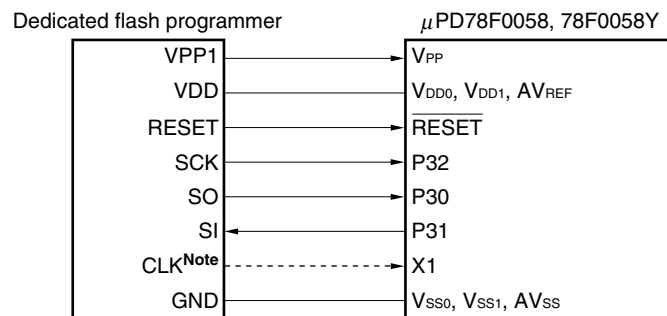
## (d) UART (UART ch-0)



## (e) UART (UART ch-1)



## (f) Pseudo 3-wire serial I/O



**Note** Connect this pin when the system clock is supplied from the dedicated flash programmer. If a resonator is already connected to the X1 pin, the CLK pin does not need to be connected.

**Caution** The V<sub>DD0</sub> and V<sub>DD1</sub> pins, if already connected to the power supply, must be connected to the VDD pin of the dedicated flash programmer. When using the power supply connected to the V<sub>DD0</sub> and V<sub>DD1</sub> pins, supply voltage before starting programming.

If Flashpro III (part no. FL-PR3, PG-FP3)/Flashpro IV is used as a dedicated flash programmer, the following signals are generated for the  $\mu$ PD78F0058, 78F0058Y. For details, refer to the manual of Flashpro III/Flashpro IV.

**Table 26-5. Pin Connection List**

Signal Name	I/O	Pin Function	Pin Name	3-Wire Serial I/O	UART	Pseudo 3-Wire Serial I/O
VPP1	Output	Write voltage	V <sub>PP</sub>	⊙	⊙	⊙
VPP2	–	–	–	×	×	×
VDD	I/O	V <sub>DD</sub> voltage generation/ voltage monitoring	V <sub>DD0</sub> , V <sub>DD1</sub> , AV <sub>REF</sub>	⊙ <b>Note</b>	⊙ <b>Note</b>	⊙ <b>Note</b>
GND	–	Ground	V <sub>SS0</sub> , V <sub>SS1</sub> , AV <sub>SS</sub>	⊙	⊙	⊙
CLK	Output	Clock output	X1	○	×	○
RESET	Output	Reset signal	$\overline{\text{RESET}}$	⊙	⊙	⊙
SI (Rx/D)	Input	Reception signal	SO0/SO1/SO2/TxD0/TxD1/P31	⊙	⊙	⊙
SO (Tx/D)	Output	Transmit signal	SI0/SI1/SI2/RxD0/RxD1/P30	⊙	⊙	⊙
SCK	Output	Transfer clock	$\overline{\text{SCK0/SCK1/SCK2/P32}}$	⊙	×	⊙
HS	Input	Handshake signal	–	×	×	×

**Note** V<sub>DD</sub> voltage must be supplied before programming is started.

**Remark** ⊙: Pin must be connected.

○: If the signal is supplied on the target board, pin does not need to be connected.

×: Pin does not need to be connected.

### 26.3.3 On-board pin processing

When performing programming on the target system, provide a connector on the target system to connect the dedicated flash programmer.

An on-board function that allows switching between normal operation mode and flash memory programming mode may be required in some cases.

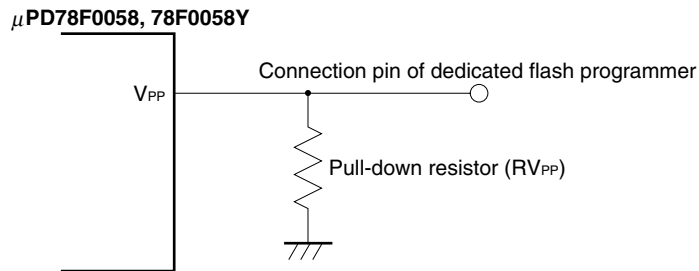
#### <V<sub>PP</sub> pin>

In normal operation mode, input 0 V to the V<sub>PP</sub> pin. In flash memory programming mode, a write voltage of 10.0 V (TYP.) is supplied to the V<sub>PP</sub> pin, so perform the following.

- (1) Connect a pull-down resistor (RV<sub>PP</sub> = 10 k $\Omega$ ) to the V<sub>PP</sub> pin.
- (2) Use the jumper on the board to switch the V<sub>PP</sub> pin input to either the programmer or directly to GND.

A V<sub>PP</sub> pin connection example is shown below.

**Figure 26-6. V<sub>PP</sub> Pin Connection Example**



#### <Serial interface pins>

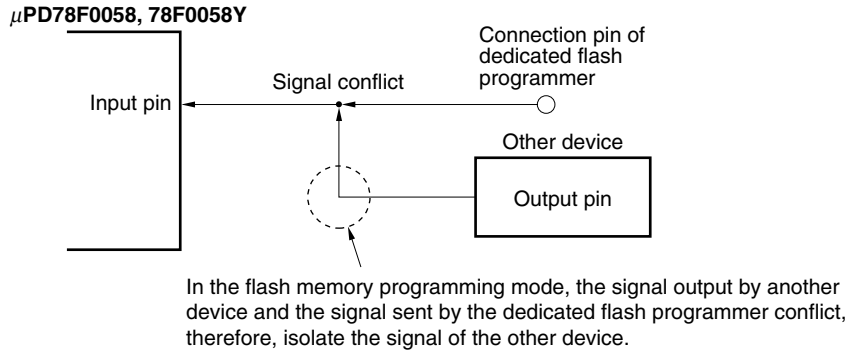
The following shows the pins used by the serial interface.

Serial Interface	Pins Used
3-wire serial I/O	SI0, SO0, $\overline{\text{SCK0}}$
	SI1, SO1, $\overline{\text{SCK1}}$
	SI2, SO2, $\overline{\text{SCK2}}$
UART	RxD0, TxD0
	RxD1, TxD1
Pseudo 3-wire serial I/O	P30, P31, P32

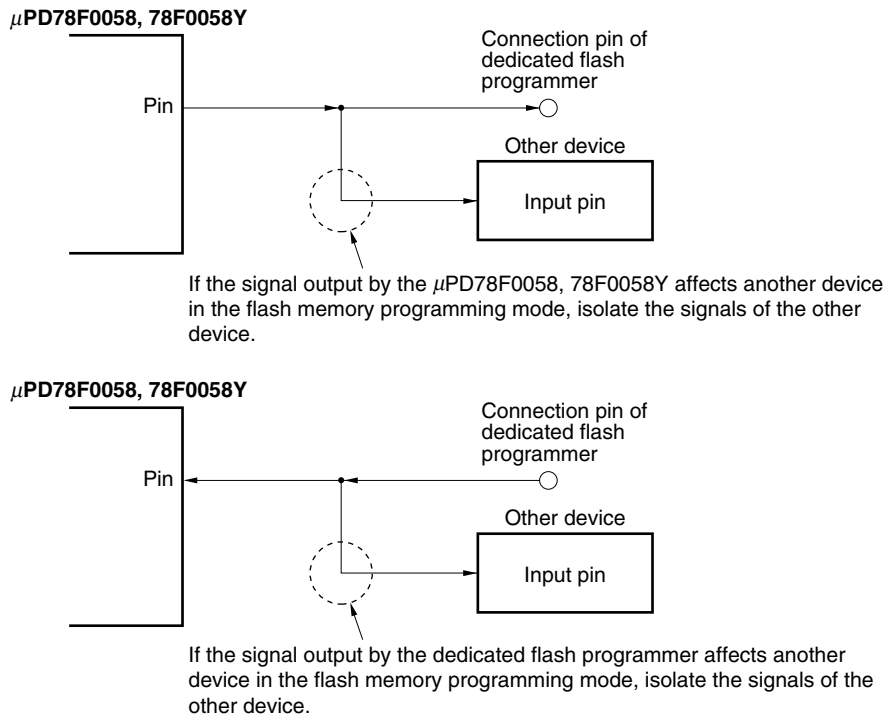
When connecting the dedicated flash programmer to a serial interface pin that is connected to another device on-board, signal conflict or abnormal operation of the other device may occur. Care must therefore be taken with such connections.

**(1) Signal conflict**

If the dedicated flash programmer (output) is connected to a serial interface pin (input) that is connected to another device (output), a signal conflict occurs. To prevent this, isolate the connection with the other device or set the other device to the output high impedance status.

**Figure 26-7. Signal Conflict (Input Pin of Serial Interface)****(2) Abnormal operation of other device**

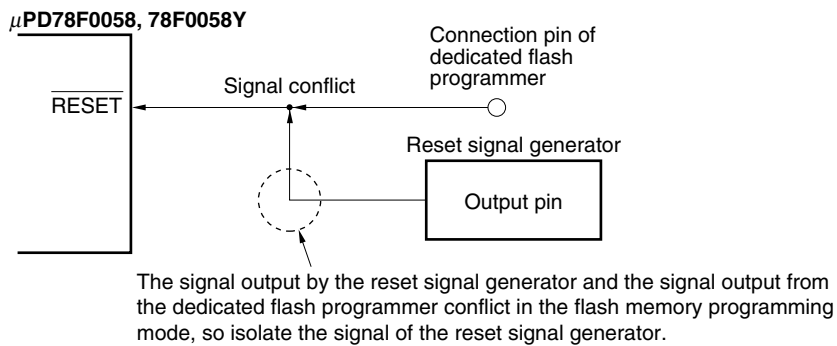
If the dedicated flash programmer (output or input) is connected to a serial interface pin (input or output) that is connected to another device (input), a signal is output to the device, and this may cause an abnormal operation. To prevent this abnormal operation, isolate the connection with the other device or set so that the input signals to the other device are ignored.

**Figure 26-8. Abnormal Operation of Other Device**



**<RESET pin>**

If the reset signal of the dedicated flash programmer is connected to the  $\overline{\text{RESET}}$  pin connected to the reset signal generator on-board, a signal conflict occurs. To prevent this, isolate the connection with the reset signal generator. If the reset signal is input from the user system in the flash memory programming mode, a normal programming operation cannot be performed. Therefore, do not input reset signals from other than the dedicated flash programmer.

**Figure 26-9. Signal Conflict ( $\overline{\text{RESET}}$  Pin)****<Port pins>**

When the  $\mu$ PD78F0058 and 78F0058Y enter the flash memory programming mode, all the pins other than those that communicate in flash memory programming are in the same status as immediately after reset.

If the external device does not recognize initial statuses such as the output high impedance status, therefore, connect the external device to  $V_{DD0}$  or  $V_{SS0}$  via a resistor.

**<Oscillator>**

When using the on-board clock, connect X1, X2, XT1, and XT2 as required in the normal operation mode.

When using the clock output of the flash programmer, connect it directly to X1, disconnecting the main oscillator on-board, and leave the X2 pin open. The subsystem clock conforms to the normal operation mode.

**<Power supply>**

To use the power output from the flash programmer, connect the  $V_{DD0}$  and  $V_{DD1}$  pins to VDD of the flash programmer, and the  $V_{SS0}$  and  $V_{SS1}$  pins to GND of the flash programmer.

To use the on-board power supply, make connections that accord with the normal operation mode. However, because the voltage is monitored by the flash programmer, be sure to connect VDD of the flash programmer. Supply the same power as in the normal operation mode to the other power supply pins ( $AV_{REF0}$ ,  $AV_{REF1}$ , and  $AV_{SS}$ ).

### 26.3.4 Connection of adapter for flash writing

The following figures show examples of the recommended connection when the adapter for flash writing is used.

**Figure 26-10. Wiring Example for Flash Writing Adapter in 3-Wire Serial I/O Mode (SIO ch-0)**

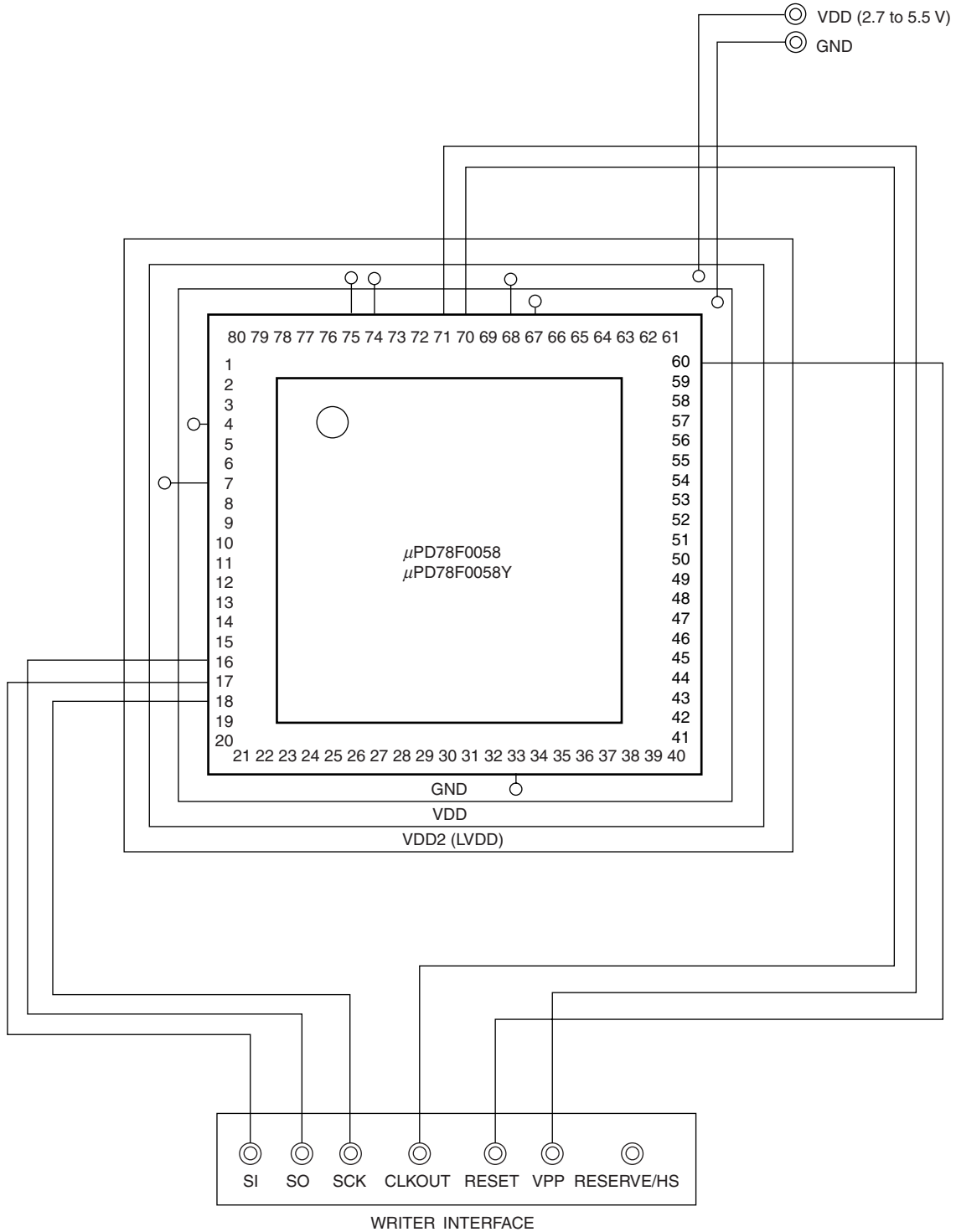


Figure 26-11. Wiring Example for Flash Writing Adapter in 3-Wire Serial I/O Mode (SIO ch-1)

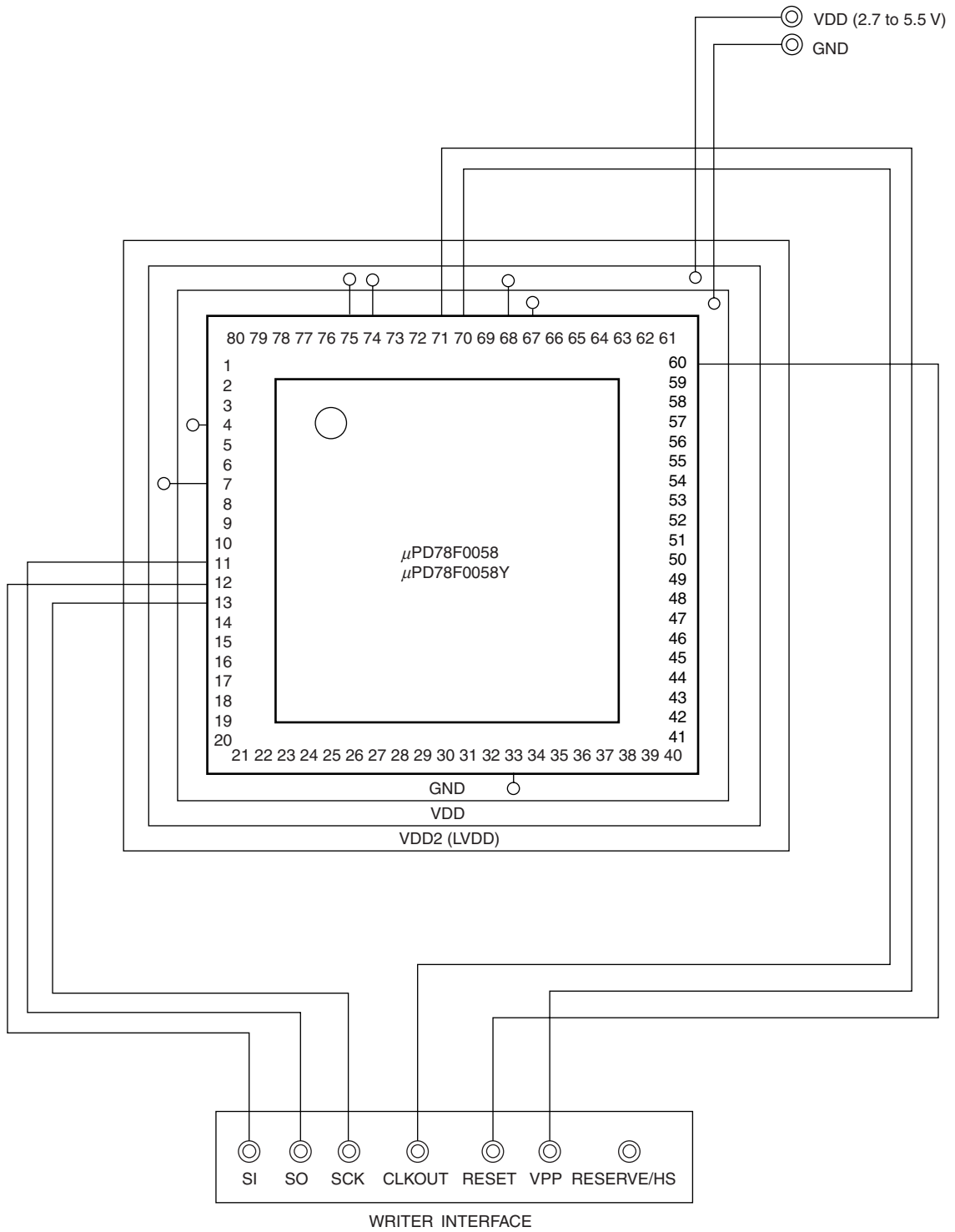


Figure 26-12. Wiring Example for Flash Writing Adapter in 3-Wire Serial I/O Mode (SIO ch-2)

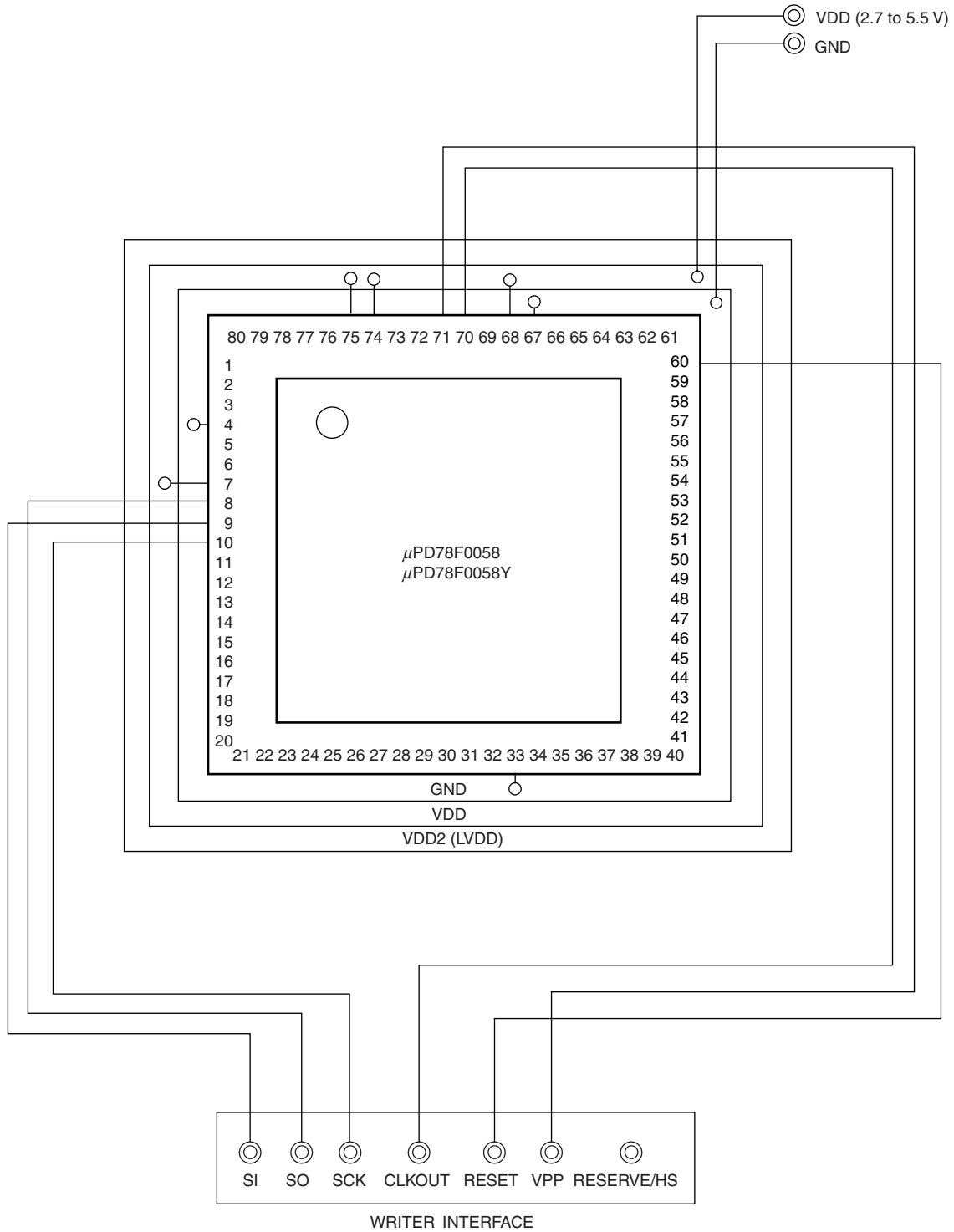


Figure 26-13. Wiring Example for Flash Writing Adapter in UART Mode (UART ch-0)

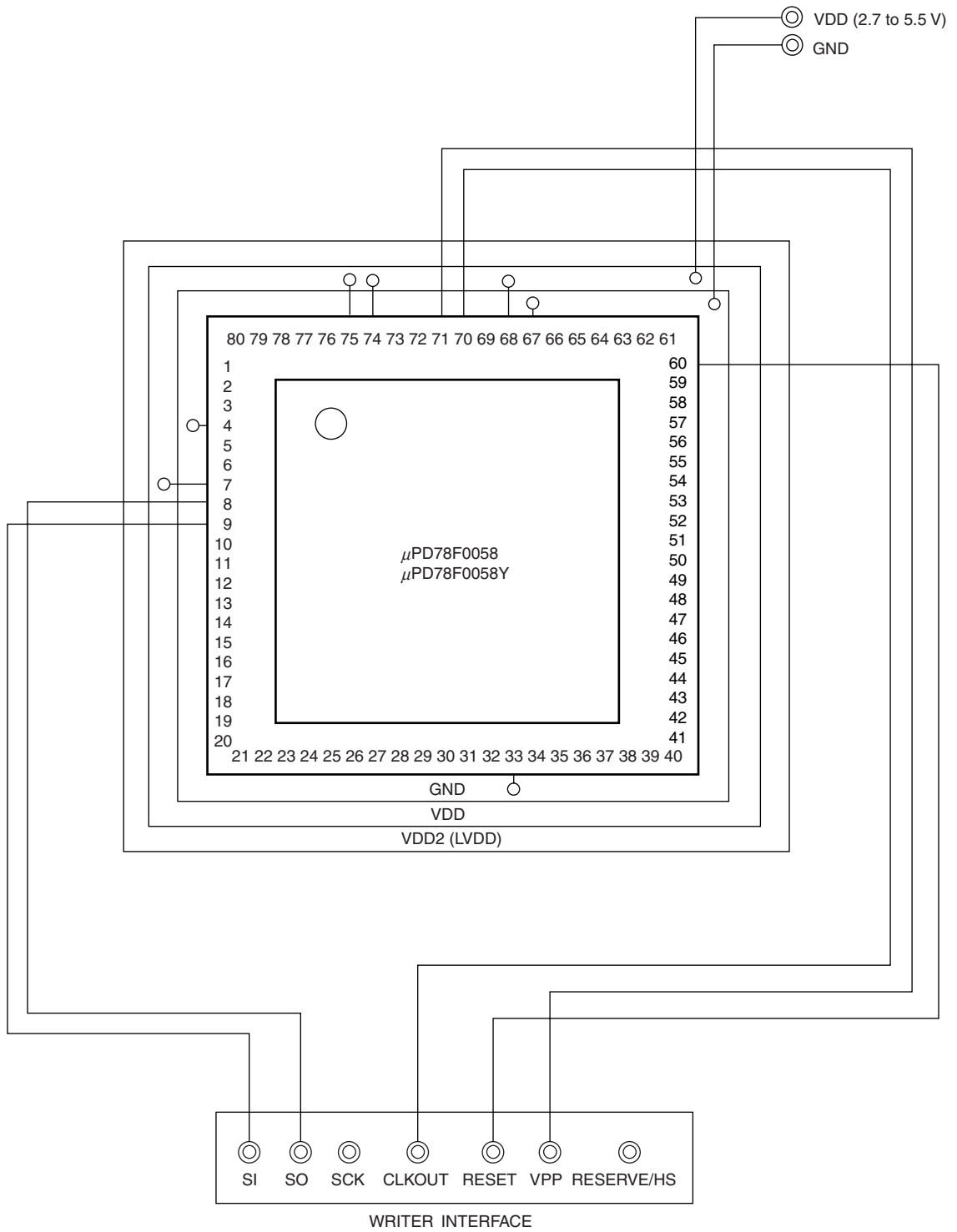


Figure 26-14. Wiring Example for Flash Writing Adapter in UART Mode (UART ch-1)

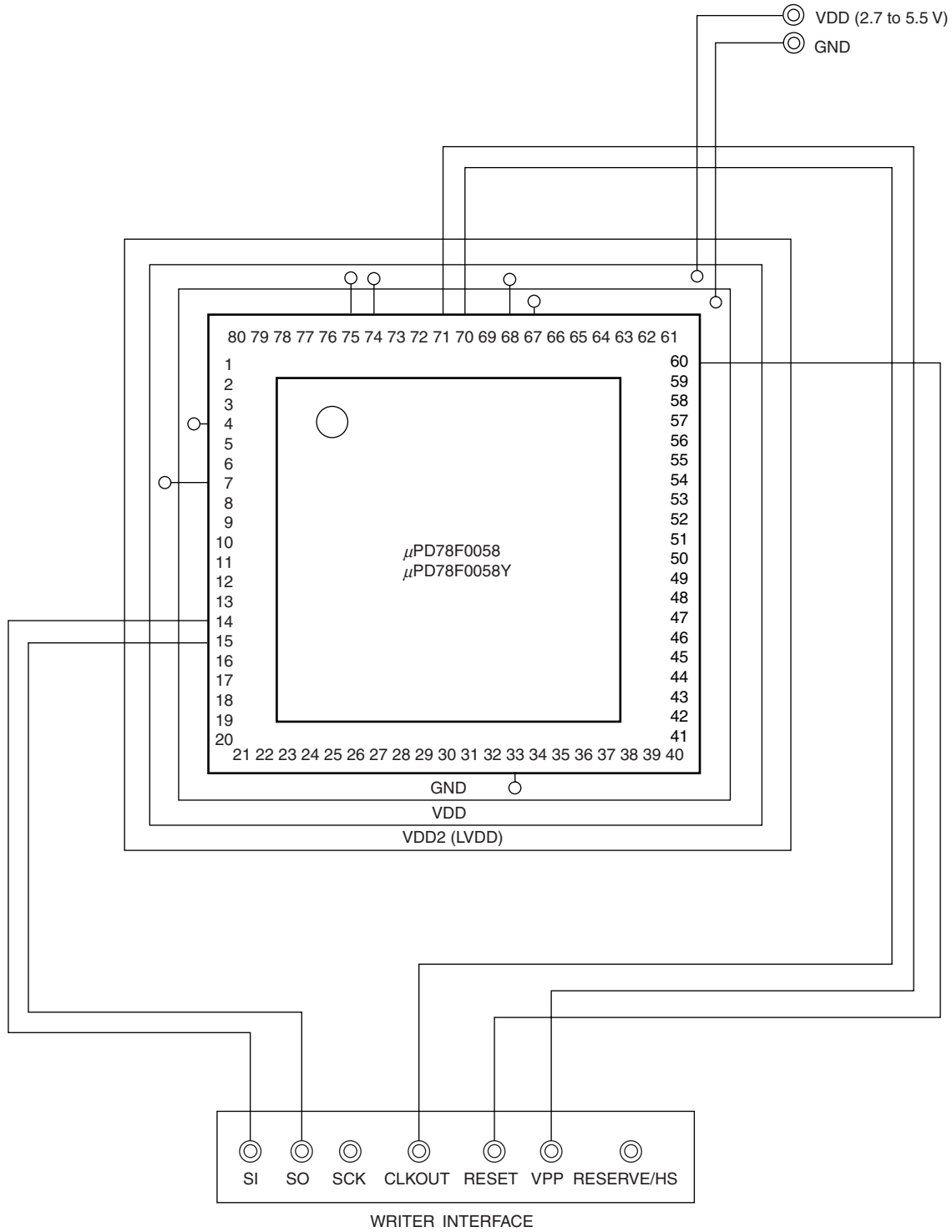
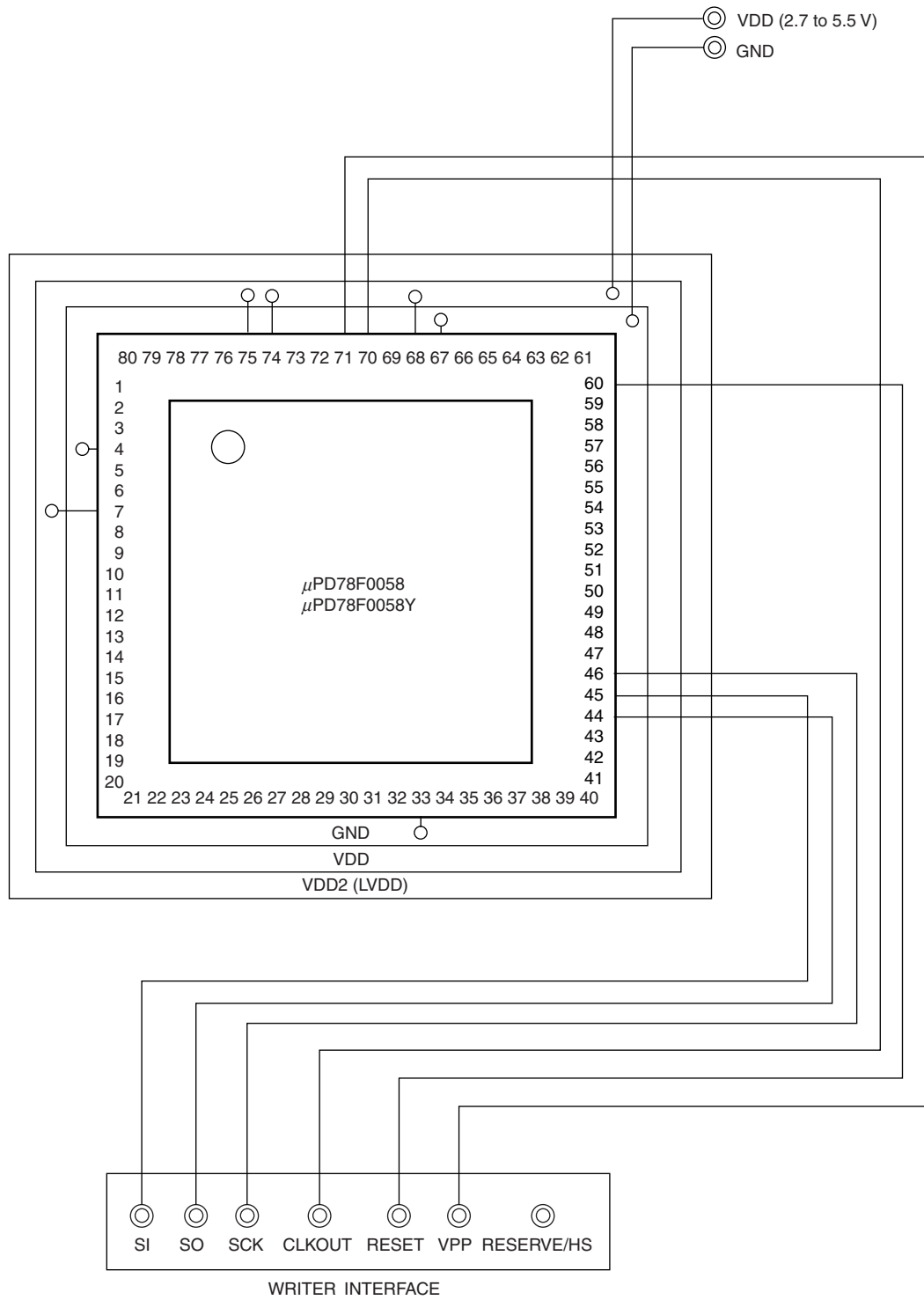


Figure 26-15. Wiring Example for Flash Writing Adapter in Pseudo 3-Wire Mode



## CHAPTER 27 INSTRUCTION SET OVERVIEW

This chapter describes each instruction set of the  $\mu$ PD780058 and 780058Y Subseries in table form. For details of the operations and operation codes, 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 description methods

Operands are described in the “Operand” column of each instruction in accordance with the description method of the instruction operand identifier (refer to the assembler specifications for details). When there are two or more description methods, select one of them. Uppercase letters and the symbols #, !, \$ and [ ] are keywords and must be described 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 describe 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 description.

**Table 27-1. Operand Identifiers and Description Methods**

Identifier	Description 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 <sup>Note</sup>
sfrp	Special-function register symbol (16-bit manipulatable register even addresses only) <sup>Note</sup>
saddr	FE20H to FF1FH Immediate data or label
saddrp	FE20H to FF1FH Immediate data or label (even address only)
addr16	0000H to FFFFH Immediate data or label (Only even addresses for 16-bit data transfer instructions)
addr11	0800H to 0FFFH Immediate data or label
addr5	0040H to 007FH Immediate data or label (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, see **Table 5-2 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
NMIS:	Non-maskable interrupt servicing flag
( ):	Memory contents indicated by address or register contents in parentheses
× <sub>H</sub> , × <sub>L</sub> :	Higher 8 bits and lower 8 bits of 16-bit register
∧:	Logical product (AND)
∨:	Logical sum (OR)
⊕:	Exclusive logical sum (exclusive OR)
—:	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):	Nt 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	<b>MOV</b>	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 + n	A ← (addr16)			
		!addr16, A		3	8	9 + m	(addr16) ← A			
		PSW, #byte		3	–	7	PSW ← byte	x	x	x
		A, PSW		2	–	5	A ← PSW			
		PSW, A		2	–	5	PSW ← A	x	x	x
		A, [DE]		1	4	5 + n	A ← (DE)			
		[DE], A		1	4	5 + m	(DE) ← A			
		A, [HL]		1	4	5 + n	A ← (HL)			
		[HL], A		1	4	5 + m	(HL) ← A			
		A, [HL + byte]		2	8	9 + n	A ← (HL + byte)			
		[HL + byte], A		2	8	9 + m	(HL + byte) ← A			
	A, [HL + B]		1	6	7 + n	A ← (HL + B)				
	[HL + B], A		1	6	7 + m	(HL + B) ← A				
	A, [HL + C]		1	6	7 + n	A ← (HL + C)				
	[HL + C], A		1	6	7 + m	(HL + C) ← A				
	<b>XCH</b>	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 + n + m	A ↔ (addr16)			
A, [DE]			1	4	6 + n + m	A ↔ (DE)				
A, [HL]			1	4	6 + n + m	A ↔ (HL)				
A, [HL + byte]			2	8	10 + n + m	A ↔ (HL + byte)				
A, [HL + B]			2	8	10 + n + m	A ↔ (HL + B)				
A, [HL + C]			2	8	10 + n + m	A ↔ (HL + C)				

- Notes**
1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.
  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<sub>CPU</sub>) selected by the processor clock control register (PCC).
  2. This clock cycle applies to the internal ROM program.
  3. n is the number of waits when external memory expansion area is read from.
  4. m is the number of waits when external memory expansion area is written to.

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag			
				Note 1	Note 2		Z	AC	CY	
16-bit data transfer	<b>MOVW</b>	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	<b>Note 3</b>	1	4	–	AX ← rp			
		rp, AX	<b>Note 3</b>	1	4	–	rp ← AX			
		AX, !addr16		3	10	12 + 2n	AX ← (addr16)			
	!addr16, AX		3	10	12 + 2m	(addr16) ← AX				
	<b>XCHW</b>	AX, rp	<b>Note 3</b>	1	4	–	AX ↔ rp			
8-bit operation	<b>ADD</b>	A, #byte	2	4	–	A, CY ← A + byte	x	x	x	
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte	x	x	x	
		A, r	<b>Note 4</b>	2	4	–	A, CY ← A + r	x	x	x
		r, A		2	4	–	r, CY ← r + A	x	x	x
		A, saddr		2	4	5	A, CY ← A + (saddr)	x	x	x
		A, !addr16		3	8	9 + n	A, CY ← A + (addr16)	x	x	x
		A, [HL]		1	4	5 + n	A, CY ← A + (HL)	x	x	x
		A, [HL + byte]		2	8	9 + n	A, CY ← A + (HL + byte)	x	x	x
		A, [HL + B]		2	8	9 + n	A, CY ← A + (HL + B)	x	x	x
		A, [HL + C]		2	8	9 + n	A, CY ← A + (HL + C)	x	x	x
		<b>ADDC</b>	A, #byte	2	4	–	A, CY ← A + byte + CY	x	x	x
			saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte + CY	x	x	x
	A, r		<b>Note 4</b>	2	4	–	A, CY ← A + r + CY	x	x	x
	r, A			2	4	–	r, CY ← r + A + CY	x	x	x
	A, saddr			2	4	5	A, CY ← A + (saddr) + CY	x	x	x
	A, !addr16			3	8	9 + n	A, CY ← A + (addr16) + CY	x	x	x
	A, [HL]			1	4	5 + n	A, CY ← A + (HL) + CY	x	x	x
	A, [HL + byte]			2	8	9 + n	A, CY ← A + (HL + byte) + CY	x	x	x

- Notes**
1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.
  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<sub>CPU</sub>) selected by the processor clock control register (PCC).
  2. This clock cycle applies to the internal ROM program.
  3. n is the number of waits when external memory expansion area is read from.
  4. m is the number of waits when external memory expansion area is written to.

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	<b>SUB</b>	A, #byte	2	4	–	A, CY ← A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte	×	×	×
		A, r <b>Note 3</b>	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 + n	A, CY ← A – (addr16)	×	×	×
		A, [HL]	1	4	5 + n	A, CY ← A – (HL)	×	×	×
		A, [HL + byte]	2	8	9 + n	A, CY ← A – (HL + byte)	×	×	×
		A, [HL + B]	2	8	9 + n	A, CY ← A – (HL + B)	×	×	×
		A, [HL + C]	2	8	9 + n	A, CY ← A – (HL + C)	×	×	×
	<b>SUBC</b>	A, #byte	2	4	–	A, CY ← A – byte – CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte – CY	×	×	×
		A, r <b>Note 3</b>	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 + n	A, CY ← A – (addr16) – CY	×	×	×
		A, [HL]	1	4	5 + n	A, CY ← A – (HL) – CY	×	×	×
		A, [HL + byte]	2	8	9 + n	A, CY ← A – (HL + byte) – CY	×	×	×
		A, [HL + B]	2	8	9 + n	A, CY ← A – (HL + B) – CY	×	×	×
		A, [HL + C]	2	8	9 + n	A, CY ← A – (HL + C) – CY	×	×	×
	<b>AND</b>	A, #byte	2	4	–	A ← A ∧ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ∧ byte	×		
		A, r <b>Note 3</b>	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 + n	A ← A ∧ (addr16)	×		
		A, [HL]	1	4	5 + n	A ← A ∧ (HL)	×		
		A, [HL + byte]	2	8	9 + n	A ← A ∧ (HL + byte)	×		
		A, [HL + B]	2	8	9 + n	A ← A ∧ (HL + B)	×		
	A, [HL + C]	2	8	9 + n	A ← A ∧ (HL + C)	×			

- Notes**
1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.
  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<sub>CPU</sub>) selected by the processor clock control register (PCC).
  2. This clock cycle applies to the internal ROM program.
  3. n is the number of waits when external memory expansion area is read from.

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	<b>OR</b>	A, #byte	2	4	–	$A \leftarrow A \vee \text{byte}$		x	
		saddr, #byte	3	6	8	$(\text{saddr}) \leftarrow (\text{saddr}) \vee \text{byte}$		x	
		A, r <b>Note 3</b>	2	4	–	$A \leftarrow A \vee r$		x	
		r, A	2	4	–	$r \leftarrow r \vee A$		x	
		A, saddr	2	4	5	$A \leftarrow A \vee (\text{saddr})$		x	
		A, !addr16	3	8	9 + n	$A \leftarrow A \vee (\text{addr16})$		x	
		A, [HL]	1	4	5 + n	$A \leftarrow A \vee (\text{HL})$		x	
		A, [HL + byte]	2	8	9 + n	$A \leftarrow A \vee (\text{HL} + \text{byte})$		x	
		A, [HL + B]	2	8	9 + n	$A \leftarrow A \vee (\text{HL} + B)$		x	
		A, [HL + C]	2	8	9 + n	$A \leftarrow A \vee (\text{HL} + C)$		x	
	<b>XOR</b>	A, #byte	2	4	–	$A \leftarrow A \nabla \text{byte}$		x	
		saddr, #byte	3	6	8	$(\text{saddr}) \leftarrow (\text{saddr}) \nabla \text{byte}$		x	
		A, r <b>Note 3</b>	2	4	–	$A \leftarrow A \nabla r$		x	
		r, A	2	4	–	$r \leftarrow r \nabla A$		x	
		A, saddr	2	4	5	$A \leftarrow A \nabla (\text{saddr})$		x	
		A, !addr16	3	8	9 + n	$A \leftarrow A \nabla (\text{addr16})$		x	
		A, [HL]	1	4	5 + n	$A \leftarrow A \nabla (\text{HL})$		x	
		A, [HL + byte]	2	8	9 + n	$A \leftarrow A \nabla (\text{HL} + \text{byte})$		x	
		A, [HL + B]	2	8	9 + n	$A \leftarrow A \nabla (\text{HL} + B)$		x	
		A, [HL + C]	2	8	9 + n	$A \leftarrow A \nabla (\text{HL} + C)$		x	
	<b>CMP</b>	A, #byte	2	4	–	$A - \text{byte}$	x	x	x
		saddr, #byte	3	6	8	$(\text{saddr}) - \text{byte}$	x	x	x
		A, r <b>Note 3</b>	2	4	–	$A - r$	x	x	x
		r, A	2	4	–	$r - A$	x	x	x
		A, saddr	2	4	5	$A - (\text{saddr})$	x	x	x
		A, !addr16	3	8	9 + n	$A - (\text{addr16})$	x	x	x
		A, [HL]	1	4	5 + n	$A - (\text{HL})$	x	x	x
		A, [HL + byte]	2	8	9 + n	$A - (\text{HL} + \text{byte})$	x	x	x
		A, [HL + B]	2	8	9 + n	$A - (\text{HL} + B)$	x	x	x
		A, [HL + C]	2	8	9 + n	$A - (\text{HL} + C)$	x	x	x

- Notes**
1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.
  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 ( $f_{\text{CPU}}$ ) selected by the processor clock control register (PCC).
  2. This clock cycle applies to the internal ROM program.
  3. n is the number of waits when external memory expansion area is read from.

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit operation	<b>ADDW</b>	AX, #word	3	6	–	AX, CY ← AX + word	×	×	×
	<b>SUBW</b>	AX, #word	3	6	–	AX, CY ← AX – word	×	×	×
	<b>CMPW</b>	AX, #word	3	6	–	AX – word	×	×	×
Multiply/divide	<b>MULU</b>	X	2	16	–	AX ← A × X			
	<b>DIVUW</b>	C	2	25	–	AX (Quotient), C (Remainder) ← AX ÷ C			
Increment/decrement	<b>INC</b>	r	1	2	–	r ← r + 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) + 1	×	×	
	<b>DEC</b>	r	1	2	–	r ← r – 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) – 1	×	×	
	<b>INCW</b>	rp	1	4	–	rp ← rp + 1			
<b>DECW</b>	rp	1	4	–	rp ← rp – 1				
Rotate	<b>ROR</b>	A, 1	1	2	–	(CY, A <sub>7</sub> ← A <sub>0</sub> , A <sub>m-1</sub> ← A <sub>m</sub> ) × 1 time			×
	<b>ROL</b>	A, 1	1	2	–	(CY, A <sub>0</sub> ← A <sub>7</sub> , A <sub>m+1</sub> ← A <sub>m</sub> ) × 1 time			×
	<b>RORC</b>	A, 1	1	2	–	(CY ← A <sub>0</sub> , A <sub>7</sub> ← CY, A <sub>m-1</sub> ← A <sub>m</sub> ) × 1 time			×
	<b>ROLC</b>	A, 1	1	2	–	(CY ← A <sub>7</sub> , A <sub>0</sub> ← CY, A <sub>m+1</sub> ← A <sub>m</sub> ) × 1 time			×
	<b>ROR4</b>	[HL]	2	10	12 + n + m	A <sub>3-0</sub> ← (HL) <sub>3-0</sub> , (HL) <sub>7-4</sub> ← A <sub>3-0</sub> , (HL) <sub>3-0</sub> ← (HL) <sub>7-4</sub>			
	<b>ROL4</b>	[HL]	2	10	12 + n + m	A <sub>3-0</sub> ← (HL) <sub>7-4</sub> , (HL) <sub>3-0</sub> ← A <sub>3-0</sub> , (HL) <sub>7-4</sub> ← (HL) <sub>3-0</sub>			
BCD adjust	<b>ADJBA</b>		2	4	–	Decimal Adjust Accumulator after Addition	×	×	×
	<b>ADJBS</b>		2	4	–	Decimal Adjust Accumulator after Subtract	×	×	×
Bit manipulation	<b>MOV1</b>	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 + n	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 + n + m	(HL).bit ← CY					

**Notes** 1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.

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<sub>cpu</sub>) selected by the processor clock control register (PCC).

2. This clock cycle applies to the internal ROM program.

3. n is the number of waits when external memory expansion area is read from.

4. m is the number of waits when external memory expansion area is written to.

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag			
				Note 1	Note 2		Z	AC	CY	
Bit manipulation	<b>AND1</b>	CY, saddr.bit	3	6	7	$CY \leftarrow CY \wedge (\text{saddr.bit})$			×	
		CY, sfr.bit	3	–	7	$CY \leftarrow CY \wedge \text{sfr.bit}$			×	
		CY, A.bit	2	4	–	$CY \leftarrow CY \wedge A.\text{bit}$			×	
		CY, PSW.bit	3	–	7	$CY \leftarrow CY \wedge \text{PSW.bit}$			×	
		CY, [HL].bit	2	6	7 + n	$CY \leftarrow CY \wedge (\text{HL}).\text{bit}$			×	
	<b>OR1</b>	CY, saddr.bit	3	6	7	$CY \leftarrow CY \vee (\text{saddr.bit})$			×	
		CY, sfr.bit	3	–	7	$CY \leftarrow CY \vee \text{sfr.bit}$			×	
		CY, A.bit	2	4	–	$CY \leftarrow CY \vee A.\text{bit}$			×	
		CY, PSW.bit	3	–	7	$CY \leftarrow CY \vee \text{PSW.bit}$			×	
		CY, [HL].bit	2	6	7 + n	$CY \leftarrow CY \vee (\text{HL}).\text{bit}$			×	
	<b>XOR1</b>	CY, saddr.bit	3	6	7	$CY \leftarrow CY \oplus (\text{saddr.bit})$			×	
		CY, sfr.bit	3	–	7	$CY \leftarrow CY \oplus \text{sfr.bit}$			×	
		CY, A.bit	2	4	–	$CY \leftarrow CY \oplus A.\text{bit}$			×	
		CY, PSW.bit	3	–	7	$CY \leftarrow CY \oplus \text{PSW.bit}$			×	
		CY, [HL].bit	2	6	7 + n	$CY \leftarrow CY \oplus (\text{HL}).\text{bit}$			×	
	<b>SET1</b>	saddr.bit	2	4	6	$(\text{saddr.bit}) \leftarrow 1$				
		sfr.bit	3	–	8	$\text{sfr.bit} \leftarrow 1$				
		A.bit	2	4	–	$A.\text{bit} \leftarrow 1$				
		PSW.bit	2	–	6	$\text{PSW.bit} \leftarrow 1$		×	×	×
		[HL].bit	2	6	8 + n + m	$(\text{HL}).\text{bit} \leftarrow 1$				
	<b>CLR1</b>	saddr.bit	2	4	6	$(\text{saddr.bit}) \leftarrow 0$				
		sfr.bit	3	–	8	$\text{sfr.bit} \leftarrow 0$				
		A.bit	2	4	–	$A.\text{bit} \leftarrow 0$				
		PSW.bit	2	–	6	$\text{PSW.bit} \leftarrow 0$		×	×	×
		[HL].bit	2	6	8 + n + m	$(\text{HL}).\text{bit} \leftarrow 0$				
<b>SET1</b>	CY	1	2	–	$CY \leftarrow 1$			1		
<b>CLR1</b>	CY	1	2	–	$CY \leftarrow 0$			0		
<b>NOT1</b>	CY	1	2	–	$CY \leftarrow \overline{CY}$			×		

**Notes** 1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.

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.

3. n is the number of waits when external memory expansion area is read from.

4. m is the number of waits when external memory expansion area is written to.



Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Call/return	<b>CALL</b>	!addr16	3	7	–	$(SP - 1) \leftarrow (PC + 3)_H, (SP - 2) \leftarrow (PC + 3)_L,$ $PC \leftarrow \text{addr16}, SP \leftarrow SP - 2$			
	<b>CALLF</b>	!addr11	2	5	–	$(SP - 1) \leftarrow (PC + 2)_H, (SP - 2) \leftarrow (PC + 2)_L,$ $PC_{15-11} \leftarrow 00001, PC_{10-0} \leftarrow \text{addr11},$ $SP \leftarrow SP - 2$			
	<b>CALLT</b>	[addr5]	1	6	–	$(SP - 1) \leftarrow (PC + 1)_H, (SP - 2) \leftarrow (PC + 1)_L,$ $PC_H \leftarrow (00000000, \text{addr5} + 1),$ $PC_L \leftarrow (00000000, \text{addr5}),$ $SP \leftarrow SP - 2$			
	<b>BRK</b>		1	6	–	$(SP - 1) \leftarrow PSW, (SP - 2) \leftarrow (PC + 1)_H,$ $(SP - 3) \leftarrow (PC + 1)_L, PC_H \leftarrow (003FH),$ $PC_L \leftarrow (003EH), SP \leftarrow SP - 3, IE \leftarrow 0$			
	<b>RET</b>		1	6	–	$PC_H \leftarrow (SP + 1), PC_L \leftarrow (SP),$ $SP \leftarrow SP + 2$			
	<b>RETI</b>		1	6	–	$PC_H \leftarrow (SP + 1), PC_L \leftarrow (SP),$ $PSW \leftarrow (SP + 2), SP \leftarrow SP + 3,$ $NMIS \leftarrow 0$	R	R	R
	<b>RETB</b>		1	6	–	$PC_H \leftarrow (SP + 1), PC_L \leftarrow (SP),$ $PSW \leftarrow (SP + 2), SP \leftarrow SP + 3$	R	R	R
Stack manipulation	<b>PUSH</b>	PSW	1	2	–	$(SP - 1) \leftarrow PSW, SP \leftarrow SP - 1$			
		rp	1	4	–	$(SP - 1) \leftarrow rp_H, (SP - 2) \leftarrow rp_L,$ $SP \leftarrow SP - 2$			
	<b>POP</b>	PSW	1	2	–	$PSW \leftarrow (SP), SP \leftarrow SP + 1$	R	R	R
		rp	1	4	–	$rp_H \leftarrow (SP + 1), rp_L \leftarrow (SP),$ $SP \leftarrow SP + 2$			
	<b>MOVW</b>	SP, #word	4	–	10	$SP \leftarrow \text{word}$			
		SP, AX	2	–	8	$SP \leftarrow AX$			
AX, SP		2	–	8	$AX \leftarrow SP$				
Unconditional branch	<b>BR</b>	!addr16	3	6	–	$PC \leftarrow \text{addr16}$			
		\$addr16	2	6	–	$PC \leftarrow PC + 2 + \text{jdisp8}$			
		AX	2	8	–	$PC_H \leftarrow A, PC_L \leftarrow X$			
Conditional branch	<b>BC</b>	\$addr16	2	6	–	$PC \leftarrow PC + 2 + \text{jdisp8}$ if $CY = 1$			
	<b>BNC</b>	\$addr16	2	6	–	$PC \leftarrow PC + 2 + \text{jdisp8}$ if $CY = 0$			
	<b>BZ</b>	\$addr16	2	6	–	$PC \leftarrow PC + 2 + \text{jdisp8}$ if $Z = 1$			
	<b>BNZ</b>	\$addr16	2	6	–	$PC \leftarrow PC + 2 + \text{jdisp8}$ if $Z = 0$			

**Notes** 1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.

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
Condi-tional branch	<b>BT</b>	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 + n	PC ← PC + 3 + jdisp8 if (HL).bit = 1			
	<b>BF</b>	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 + n	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
	<b>BTCLR</b>	saddr.bit, \$addr16	4	10	12	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)			
		sfr.bit, \$addr16	4	–	12	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
		A.bit, \$addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
		PSW.bit, \$addr16	4	–	12	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	×	×	×
		[HL].bit, \$addr16	3	10	12 + n + m	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			
<b>DBNZ</b>	B, \$addr16	2	6	–	B ← B – 1, then PC ← PC + 2 + jdisp8 if B ≠ 0				
	C, \$addr16	2	6	–	C ← C – 1, then PC ← PC + 2 + jdisp8 if C ≠ 0				
	saddr, \$addr16	3	8	10	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0				
CPU control	<b>SEL</b>	RBn	2	4	–	RBS1, 0 ← n			
	<b>NOP</b>		1	2	–	No Operation			
	<b>EI</b>		2	–	6	IE ← 1 (Enable Interrupt)			
	<b>DI</b>		2	–	6	IE ← 0 (Disable Interrupt)			
	<b>HALT</b>		2	6	–	Set HALT Mode			
	<b>STOP</b>		2	6	–	Set STOP Mode			

- Notes**
1. When the internal high-speed RAM area is accessed or instruction that performs no data access is executed.
  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<sub>CPU</sub>) selected by the processor clock control register (PCC).
  2. This clock cycle applies to the internal ROM program.
  3. n is the number of waits when external memory expansion area is read from.
  4. m is the number of waits when external memory expansion area is written to.

### 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

Second Operand First Operand	#byte	A	r>Note	sfr	saddr	!addr16	PSW	[DE]	[HL]	[HL + byte] [HL + B] [HL + C]	\$addr16	1	None
A	ADD ADDC SUB SUBC AND OR XOR CMP		MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP		ROR ROL RORC ROL4	
r	MOV	MOV ADD ADDC SUB SUBC AND OR XOR CMP											INC DEC
B, C											DBNZ		
sfr	MOV	MOV											
saddr	MOV ADD ADDC SUB SUBC AND OR XOR CMP	MOV									DBNZ		INC DEC
!addr16		MOV											
PSW	MOV	MOV											PUSH POP
[DE]		MOV											
[HL]		MOV											ROR4 ROL4
[HL + byte] [HL + B] [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 1st Operand	#word	AX	rp <sup>Note</sup>	sfrp	saddrp	!addr16	SP	None
AX	ADDW SUBW CMPW		MOVW XCHW	MOVW	MOVW	MOVW	MOVW	
rp	MOVW	MOVW <sup>Note</sup>						INCW DECW PUSH 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 BF BTCLR	SET1 CLR1
sfr.bit						MOV1	BT BF BTCLR	SET1 CLR1
saddr.bit						MOV1	BT BF BTCLR	SET1 CLR1
PSW.bit						MOV1	BT BF BTCLR	SET1 CLR1
[HL].bit						MOV1	BT BF BTCLR	SET1 CLR1
CY	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1			SET1 CLR1 NOT1

**(4) Call/instructions/branch instructions**

CALL, CALLF, CALLT, BR, BC, BNC, BZ, BNZ, BT, BF, BTCLR, DBNZ

Second Operand First Operand	AX	!addr16	!addr11	[addr5]	\$addr16
Basic instruction	BR	CALL BR	CALLF	CALLT	BR BC BNC BZ BNZ
Compound instruction					BT BF BTCLR DBNZ

**(5) Other instructions**

ADJBA, ADJBS, BRK, RET, RETI, RETB, SEL, NOP, EI, DI, HALT, STOP

Absolute Maximum Ratings (T<sub>A</sub> = 25°C)

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V <sub>DD</sub>			-0.3 to +6.5	V
	AV <sub>REF0</sub>			-0.3 to V <sub>DD</sub> + 0.3	V
	AV <sub>REF1</sub>			-0.3 to V <sub>DD</sub> + 0.3	V
	AV <sub>SS</sub>			-0.3 to +0.3	V
Input voltage	V <sub>I1</sub>	P00 to P05, P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131, X1, X2, XT2, RESET		-0.3 to V <sub>DD</sub> + 0.3	V
	V <sub>I2</sub>	P60 to P63	N-ch open drain	-0.3 to +16	V
Output voltage	V <sub>O</sub>			-0.3 to V <sub>DD</sub> + 0.3	V
Analog input voltage	V <sub>AN</sub>	P10 to P17	Analog input pin	AV <sub>SS</sub> - 0.3 to AV <sub>REF0</sub> + 0.3	V
Output current, high	I <sub>OH</sub>	Per pin		-10	mA
		Total for P01 to P05, P30 to P37, P56, P57, P60 to P67, P120 to P127		-15	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P50 to P55, P70 to P72, P130, P131		-15	mA
Output current, low	I <sub>OL</sub> <sup>Note</sup>	Per pin for other than P50 to P57, P60 to P63	Peak value	20	mA
			rms value	15	mA
		Per pin for P50 to P57, P60 to P63	Peak value	30	mA
			rms value	10	mA
		Total for P50 to P55	Peak value	100	mA
			rms value	70	mA
		Total for P56, P57, P60 to P63	Peak value	100	mA
			rms value	70	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P70 to P72, P130, P131	Peak value	50	mA
			rms value	20	mA
		Total for P01 to P05, P30 to P37, P64 to P67, P120 to P127	Peak value	50	mA
			rms value	20	mA
Operating ambient temperature	T <sub>A</sub>			-40 to +85	°C
Storage temperature	T <sub>stg</sub>			-65 to +150	°C

**Note** The rms value should be calculated as follows: [rms value] = [Peak value] × √Duty

**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.

**Main System Clock Oscillator Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 1.8 to 5.5 V)**

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f <sub>x</sub> ) <sup>Note 1</sup>	V <sub>DD</sub> = Oscillation voltage range	1.0		5.0	MHz
		Oscillation stabilization time <sup>Note 2</sup>	After V <sub>DD</sub> reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency (f <sub>x</sub> ) <sup>Note 1</sup>		1.0		5.0	MHz
		Oscillation stabilization time <sup>Note 2</sup>	V <sub>DD</sub> = 4.5 to 5.5 V V <sub>DD</sub> = 1.8 to 5.5 V			10 30	ms
External clock		X1 input frequency (f <sub>x</sub> ) <sup>Note 1</sup>		1.0		5.0	MHz
		X1 input high-/low-level width (t <sub>xH</sub> , t <sub>xL</sub> )		85		500	ns

**Notes** 1. Indicates only oscillator characteristics. See **AC Characteristics** for instruction execution time.

2. Time required to stabilize oscillation after reset or STOP mode release.

**Cautions** 1. When using the main system clock 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<sub>SS1</sub>.
- Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

2. When the main system clock is stopped and the system is operating on the subsystem clock, wait until the oscillation stabilization time has been secured by the program before switching back to the main system clock.

**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.



**Subsystem Clock Oscillator Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 1.8 to 5.5 V)**

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Crystal resonator		Oscillation frequency (f <sub>XT</sub> ) <sup>Note 1</sup>		32	32.768	35	kHz
		Oscillation stabilization time <sup>Note 2</sup>	V <sub>DD</sub> = 4.5 to 5.5 V V <sub>DD</sub> = 1.8 to 5.5 V		1.2	2 10	s
External clock		XT1 input frequency (f <sub>XT</sub> ) <sup>Note 1</sup>		32		35	kHz
		XT1 input high-/low-level width (t <sub>XTH</sub> , t <sub>XTL</sub> )		12		15	μs

- Notes**
1. Indicates only oscillator characteristics. See **AC Characteristics** for instruction execution time.
  2. Time required to stabilize oscillation after V<sub>DD</sub> reaches oscillation voltage MIN.

**Cautions** 1. When using the subsystem clock 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<sub>SS1</sub>.
- Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

2. The subsystem clock oscillator is designed as a low-amplitude circuit for reducing current consumption, and is more prone to malfunction due to noise than the main system clock oscillator. Particular care is therefore required with the wiring method when the subsystem 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.

**Capacitance (T<sub>A</sub> = 25°C, V<sub>DD</sub> = V<sub>SS</sub> = 0 V)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input capacitance	C <sub>IN</sub>	f = 1 MHz Unmeasured pins returned to 0 V.				15	pF
I/O capacitance	C <sub>IO</sub>	f = 1 MHz Unmeasured pins returned to 0 V.	P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131			15	pF
			P60 to P63			20	pF

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Input voltage, high	$V_{IH1}$	P10 to P17, P21, P23, P30 to P32, P35 to P37, P40 to P47, P50 to P57, P64 to P67, P71, P120 to P127, P130, P131	$V_{DD} = 2.7$ to $5.5$ V	$0.7V_{DD}$		$V_{DD}$	V
			$V_{DD} = 1.8$ to $5.5$ V	$0.8V_{DD}$		$V_{DD}$	V
	$V_{IH2}$	P00 to P05, P20, P22, P24 to P27, P33, P34, P70, P72, $\overline{\text{RESET}}$	$V_{DD} = 2.7$ to $5.5$ V	$0.8V_{DD}$		$V_{DD}$	V
			$V_{DD} = 1.8$ to $5.5$ V	$0.85V_{DD}$		$V_{DD}$	V
	$V_{IH3}$	P60 to P63 (N-ch open drain)	$V_{DD} = 2.7$ to $5.5$ V	$0.7V_{DD}$		15	V
			$V_{DD} = 1.8$ to $5.5$ V	$0.8V_{DD}$		15	V
	$V_{IH4}$	X1, X2	$V_{DD} = 2.7$ to $5.5$ V	$V_{DD} - 0.5$		$V_{DD}$	V
			$V_{DD} = 1.8$ to $5.5$ V	$V_{DD} - 0.2$		$V_{DD}$	V
	$V_{IH5}$	XT1/P07, XT2	$4.5 \text{ V} \leq V_{DD} \leq 5.5 \text{ V}$	$0.8V_{DD}$		$V_{DD}$	V
			$2.7 \text{ V} \leq V_{DD} < 4.5 \text{ V}$	$0.9V_{DD}$		$V_{DD}$	V
$1.8 \text{ V} \leq V_{DD} < 2.7 \text{ V}$ <sup>Note</sup>			$0.9V_{DD}$		$V_{DD}$	V	
Input voltage, low	$V_{IL1}$	P10 to P17, P21, P23, P30 to P32, P35 to P37, P40 to P47, P50 to P57, P64 to P67, P71, P120 to P127, P130, P131	$V_{DD} = 2.7$ to $5.5$ V	0		$0.3V_{DD}$	V
			$V_{DD} = 1.8$ to $5.5$ V	0		$0.2V_{DD}$	V
	$V_{IL2}$	P00 to P05, P20, P22, P24 to P27, P33, P34, P70, P72, $\overline{\text{RESET}}$	$V_{DD} = 2.7$ to $5.5$ V	0		$0.2V_{DD}$	V
			$V_{DD} = 1.8$ to $5.5$ V	0		$0.15V_{DD}$	V
	$V_{IL3}$	P60 to P63	$4.5 \text{ V} \leq V_{DD} \leq 5.5 \text{ V}$	0		$0.3V_{DD}$	V
			$2.7 \text{ V} \leq V_{DD} < 4.5 \text{ V}$	0		$0.2V_{DD}$	V
			$1.8 \text{ V} \leq V_{DD} < 2.7 \text{ V}$	0		$0.1V_{DD}$	V
	$V_{IL4}$	X1, X2	$V_{DD} = 2.7$ to $5.5$ V	0		0.4	V
			$V_{DD} = 1.8$ to $5.5$ V	0		0.2	V
	$V_{IL5}$	XT1/P07, XT2	$4.5 \text{ V} \leq V_{DD} \leq 5.5 \text{ V}$	0		$0.2V_{DD}$	V
$2.7 \text{ V} \leq V_{DD} < 4.5 \text{ V}$			0		$0.1V_{DD}$	V	
$1.8 \text{ V} \leq V_{DD} < 2.7 \text{ V}$ <sup>Note</sup>			0		$0.1V_{DD}$	V	
Output voltage, high	$V_{OH}$	$V_{DD} = 4.5$ to $5.5$ V, $I_{OH} = -1$ mA	$V_{DD} - 1.0$			V	
		$V_{DD} = 1.8$ to $5.5$ V, $I_{OH} = -100$ $\mu\text{A}$	$V_{DD} - 0.5$			V	
Output voltage, low	$V_{OL1}$	P50 to P57, P60 to P63	$V_{DD} = 4.5$ to $5.5$ V, $I_{OL} = 15$ mA		0.4	2.0	V
		P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P64 to P67, P70 to P72, P120 to P127, P130, P131	$V_{DD} = 4.5$ to $5.5$ V, $I_{OL} = 1.6$ mA			0.4	V
	$V_{OL2}$	SB0, SB1, $\overline{\text{SCK0}}$	$V_{DD} = 4.5$ to $5.5$ V, open drain, pulled-up ( $R = 1$ k $\Omega$ )			$0.2V_{DD}$	V
	$V_{OL3}$	$I_{OL} = 400$ $\mu\text{A}$				0.5	V

**Note** When P07/XT1 pin is used as P07, the inverse phase of P07 should be input to XT2 pin using an inverter.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**DC Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	$I_{LIH1}$	$V_{IN} = V_{DD}$	P00 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P70 to P72, P120 to P127, P130, P131, $\overline{\text{RESET}}$			3	$\mu\text{A}$
	$I_{LIH2}$		X1, X2, XT1/P07, XT2			20	$\mu\text{A}$
	$I_{LIH3}$	$V_{IN} = 15$ V	P60 to P63			80	$\mu\text{A}$
Input leakage current, low	$I_{LIL1}$	$V_{IN} = 0$ V	P00 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131, $\overline{\text{RESET}}$			-3	$\mu\text{A}$
	$I_{LIL2}$		X1, X2, XT1/P07, XT2			-20	$\mu\text{A}$
	$I_{LIL3}$		P60 to P63			-3 <sup>Note</sup>	$\mu\text{A}$
Mask option pull-up resistor	$R_1$	$V_{IN} = 0$ V, P60 to P63		20	40	120	k $\Omega$
Software pull-up resistor	$R_2$	$V_{IN} = 0$ V, P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131		15	30	90	k $\Omega$

**Note** When pull-up resistors are not connected to P60 to P63 (specified by the mask option), a low-level input leakage current of  $-200 \mu\text{A}$  (MAX.) flows only for 1.5 clocks (without wait) after a read instruction has been executed to port 6 (P6) or port mode register 6 (PM6). At times other than this 1.5-clock interval, a  $-3 \mu\text{A}$  (MAX.) current flows.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**DC Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output current, high	$I_{OH}$	Per pin				-1	mA
		Total for all pins				-15	mA
Output current, low	$I_{OL}$	Per pin for P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131				10	mA
		Per pin for P50 to P57, P60 to P63				15	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P70 to P72, P130, P131				10	mA
		Total for P01 to P05, P30 to P37, P64 to P67, P120 to P127				10	mA
		Total for P50 to P57, P60 to P63				70	mA

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Power supply current <sup>Note 5</sup>	I <sub>DD1</sub>	5.0 MHz crystal oscillation operating mode ( $f_{XX} = 2.5$ MHz) <sup>Note 3</sup>	$V_{DD} = 5.0$ V $\pm 10\%$ <sup>Note 1</sup>		3.5	7.7	mA
			$V_{DD} = 3.0$ V $\pm 10\%$ <sup>Note 2</sup>		0.92	2.2	mA
			$V_{DD} = 2.0$ V $\pm 10\%$ <sup>Note 2</sup>		0.47	1.2	mA
		5.0 MHz crystal oscillation operating mode ( $f_{XX} = 5.0$ MHz) <sup>Note 4</sup>	$V_{DD} = 5.0$ V $\pm 10\%$ <sup>Note 1</sup>		6.1	12.3	mA
			$V_{DD} = 3.0$ V $\pm 10\%$ <sup>Note 2</sup>		1.6	3.5	mA
		I <sub>DD2</sub>	5.0 MHz crystal oscillation HALT mode ( $f_{XX} = 2.5$ MHz) <sup>Note 3</sup>	$V_{DD} = 5.0$ V $\pm 10\%$			
	Peripheral functions operating					5.5	mA
	Peripheral functions not operating				0.97	2.4	mA
	$V_{DD} = 3.0$ V $\pm 10\%$						
	Peripheral functions operating					2.1	mA
	Peripheral functions not operating				0.38	0.92	mA
	5.0 MHz crystal oscillation HALT mode ( $f_{XX} = 5.0$ MHz) <sup>Note 4</sup>		$V_{DD} = 5.0$ V $\pm 10\%$				
			Peripheral functions operating			7.5	mA
			Peripheral functions not operating		1.2	2.9	mA
			$V_{DD} = 3.0$ V $\pm 10\%$				
			Peripheral functions operating			3.3	mA
			Peripheral functions not operating		0.48	1.2	mA
	I <sub>DD3</sub>	32.768 kHz crystal oscillation operating mode <sup>Note 6</sup>	$V_{DD} = 5.0$ V $\pm 10\%$		46	92	$\mu\text{A}$
			$V_{DD} = 3.0$ V $\pm 10\%$		25	50	$\mu\text{A}$
			$V_{DD} = 2.0$ V $\pm 10\%$		12.5	25	$\mu\text{A}$
	I <sub>DD4</sub>	32.768 kHz crystal oscillation HALT mode <sup>Note 6</sup>	$V_{DD} = 5.0$ V $\pm 10\%$		22.5	50	$\mu\text{A}$
$V_{DD} = 3.0$ V $\pm 10\%$				3.2	13.2	$\mu\text{A}$	
$V_{DD} = 2.0$ V $\pm 10\%$				1.5	11.5	$\mu\text{A}$	
I <sub>DD5</sub>	XT1 = $V_{DD}$ STOP mode When feedback resistor is used	$V_{DD} = 5.0$ V $\pm 10\%$		1.0	30	$\mu\text{A}$	
		$V_{DD} = 3.0$ V $\pm 10\%$		0.5	10	$\mu\text{A}$	
		$V_{DD} = 2.0$ V $\pm 10\%$		0.3	10	$\mu\text{A}$	
I <sub>DD6</sub>	XT1 = $V_{DD}$ STOP mode When feedback resistor is not used	$V_{DD} = 5.0$ V $\pm 10\%$		0.1	30	$\mu\text{A}$	
		$V_{DD} = 3.0$ V $\pm 10\%$		0.05	10	$\mu\text{A}$	
		$V_{DD} = 2.0$ V $\pm 10\%$		0.05	10	$\mu\text{A}$	

- Notes**
1. High-speed mode operation (when the processor clock control register (PCC) is cleared to 00H).
  2. Low-speed mode operation (when the PCC is set to 04H).
  3. Operation with main system clock  $f_{xx} = f_x/2$  (when the oscillation mode select register (OSMS) is cleared to 00H)
  4. Operation with main system clock  $f_{xx} = f_x$  (when OSMS is set to 01H)
  5. Refer to the current flowing to the  $V_{DD0}$  and  $V_{DD1}$  pins. The current flowing to the A/D converter, D/A converter, and on-chip pull-up resistor is not included.
  6. When the main system clock operation is stopped.

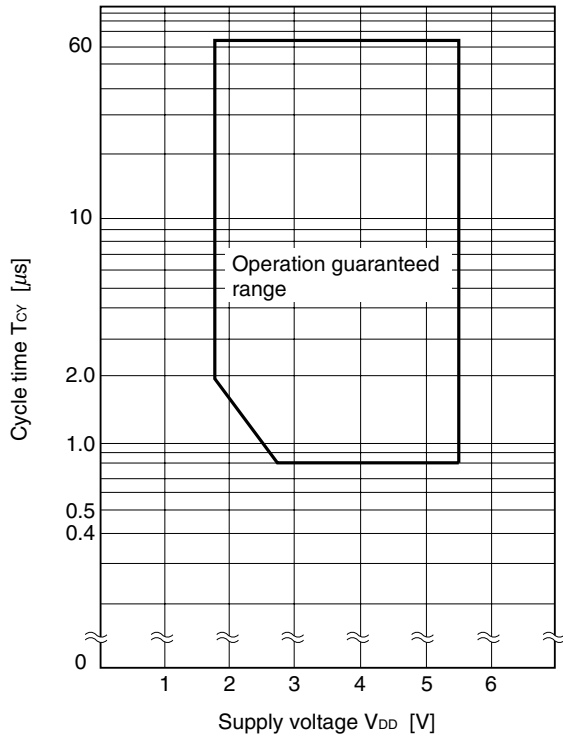
## AC Characteristics

### (1) Basic operation ( $T_A = -40$ to $+85^\circ\text{C}$ , $V_{DD} = 1.8$ to $5.5$ V)

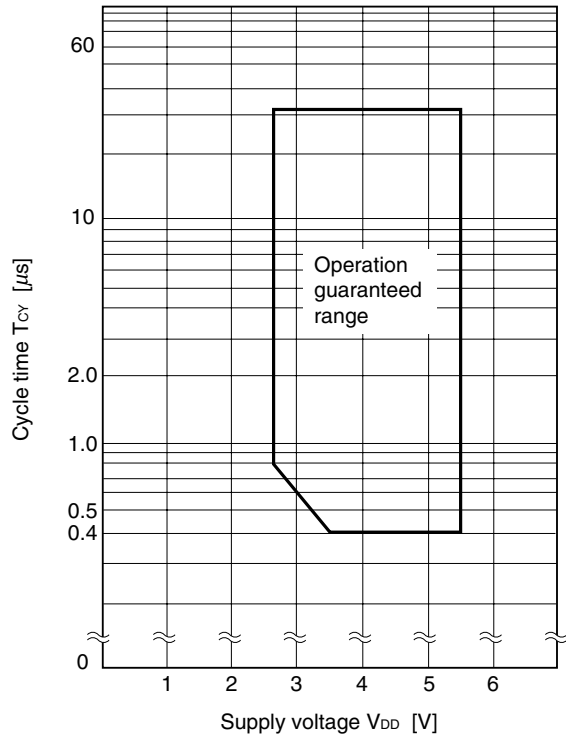
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Cycle time (Minimum instruction execution time)	$T_{CY}$	Operating with main system clock ( $f_{xx} = 2.5$ MHz) <sup>Note 1</sup>	$V_{DD} = 2.7$ to $5.5$ V	0.8		64	$\mu\text{s}$
			$V_{DD} = 1.8$ to $5.5$ V	2.0		64	$\mu\text{s}$
		Operating with main system clock ( $f_{xx} = 5.0$ MHz) <sup>Note 2</sup>	$3.5$ V $\leq V_{DD} \leq 5.5$ V	0.4		32	$\mu\text{s}$
			$2.7$ V $\leq V_{DD} < 3.5$ V	0.8		32	$\mu\text{s}$
		Operating on subsystem clock	40 <sup>Note 3</sup>	122	125	$\mu\text{s}$	
TI00 input high-/low-level width	$t_{TIH00}$	$3.5$ V $\leq V_{DD} \leq 5.5$ V	$2/f_{sam} + 0.1$ <sup>Note 4</sup>			$\mu\text{s}$	
	$t_{TIL00}$	$2.7$ V $\leq V_{DD} < 3.5$ V	$2/f_{sam} + 0.2$ <sup>Note 4</sup>			$\mu\text{s}$	
		$1.8$ V $\leq V_{DD} < 2.7$ V	$2/f_{sam} + 0.5$ <sup>Note 4</sup>			$\mu\text{s}$	
TI01 input high-/low-level width	$t_{TIH01}$	$V_{DD} = 2.7$ to $5.5$ V	10			$\mu\text{s}$	
	$t_{TIL01}$	$V_{DD} = 1.8$ to $5.5$ V	20			$\mu\text{s}$	
TI1, TI2 input frequency	$f_{TI1}$	$V_{DD} = 4.5$ to $5.5$ V	0		4	MHz	
		$V_{DD} = 1.8$ to $5.5$ V	0		275	kHz	
TI1, TI2 input high-/low-level width	$t_{TIH1}$	$V_{DD} = 4.5$ to $5.5$ V	100			ns	
	$t_{TIL1}$	$V_{DD} = 1.8$ to $5.5$ V	1.8			$\mu\text{s}$	
Interrupt request input high-/low-level width	$t_{INTH}$	INTP0	$3.5$ V $\leq V_{DD} \leq 5.5$ V	$2/f_{sam} + 0.1$ <sup>Note 4</sup>		$\mu\text{s}$	
			$2.7$ V $\leq V_{DD} < 3.5$ V	$2/f_{sam} + 0.2$ <sup>Note 4</sup>		$\mu\text{s}$	
			$1.8$ V $\leq V_{DD} < 2.7$ V	$2/f_{sam} + 0.5$ <sup>Note 4</sup>		$\mu\text{s}$	
	$t_{INTL}$	INTP1 to INTP5, P40 to P47	$V_{DD} = 2.7$ to $5.5$ V	10		$\mu\text{s}$	
			$V_{DD} = 1.8$ to $5.5$ V	20		$\mu\text{s}$	
RESET low-level width	$t_{RSL}$	$V_{DD} = 2.7$ to $5.5$ V	10			$\mu\text{s}$	
		$V_{DD} = 1.8$ to $5.5$ V	20			$\mu\text{s}$	

- Notes**
1. Operation with main system clock  $f_{xx} = f_x/2$  (when the oscillation mode select register (OSMS) is cleared to 00H)
  2. Operation with main system clock  $f_{xx} = f_x$  (when OSMS is set to 01H)
  3. Value when external clock is used. When a crystal resonator is used, it is  $114 \mu\text{s}$  (MIN.)
  4. Selection of  $f_{sam} = f_{xx}/2^N$ ,  $f_{xx}/32$ ,  $f_{xx}/64$ , and  $f_{xx}/128$  is possible with bits 0 and 1 (SCS0, SCS1) of the sampling clock select register (SCS) (when  $N = 0$  to  $4$ ).

$T_{CY}$  vs.  $V_{DD}$  (@ $f_{XX} = f_X/2$  main system clock operation)



$T_{CY}$  vs.  $V_{DD}$  (@ $f_{XX} = f_X$  main system clock operation)



**(2) Read/write operation**
**(a) When MCS = 1, PCC2 to PCC0 = 000B ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 3.5$  to  $5.5$  V)**

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	$t_{ASTH}$		$0.85t_{CY} - 50$		ns
Address setup time	$t_{ADS}$		$0.85t_{CY} - 50$		ns
Address hold time	$t_{ADH}$		50		ns
Time from address to data input	$t_{ADD1}$			$(2.85 + 2n)t_{CY} - 80$	ns
	$t_{ADD2}$			$(4 + 2n)t_{CY} - 100$	ns
Time from $\overline{RD}\downarrow$ to data input	$t_{RDD1}$			$(2 + 2n)t_{CY} - 100$	ns
	$t_{RDD2}$			$(2.85 + 2n)t_{CY} - 100$	ns
Read data hold time	$t_{RDH}$		0		ns
$\overline{RD}$ low-level width	$t_{RDL1}$		$(2 + 2n)t_{CY} - 60$		ns
	$t_{RDL2}$		$(2.85 + 2n)t_{CY} - 60$		ns
Time from $\overline{RD}\downarrow$ to $\overline{WAIT}\downarrow$ input	$t_{RDWT1}$			$0.85t_{CY} - 50$	ns
	$t_{RDWT2}$			$2t_{CY} - 60$	ns
Time from $\overline{WR}\downarrow$ to $\overline{WAIT}\downarrow$ input	$t_{WRWT}$			$2t_{CY} - 60$	ns
$\overline{WAIT}$ low-level width	$t_{WTL}$		$(1.15 + 2n)t_{CY}$	$(2 + 2n)t_{CY}$	ns
Write data setup time	$t_{WDS}$		$(2.85 + 2n)t_{CY} - 100$		ns
Write data hold time	$t_{WDH}$		20		ns
$\overline{WR}$ low-level width	$t_{WRL}$		$(2.85 + 2n)t_{CY} - 60$		ns
Delay time from $\overline{ASTB}\downarrow$ to $\overline{RD}\downarrow$	$t_{ASTRD}$		25		ns
Delay time from $\overline{ASTB}\downarrow$ to $\overline{WR}\downarrow$	$t_{ASTWR}$		$0.85t_{CY} + 20$		ns
Delay time from $\overline{RD}\uparrow$ to $\overline{ASTB}\uparrow$ at external fetch	$t_{RDAST}$		$0.85t_{CY} - 10$	$1.15t_{CY} + 20$	ns
Time from $\overline{RD}\uparrow$ to address hold at external fetch	$t_{RDADH}$		$0.85t_{CY} - 50$	$1.15t_{CY} + 50$	ns
Time from $\overline{RD}\uparrow$ to write data output	$t_{RDWD}$		40		ns
Time from $\overline{WR}\downarrow$ to write data output	$t_{WRWD}$		0	50	ns
Time from $\overline{WR}\uparrow$ to address hold	$t_{WRADH}$		$0.85t_{CY}$	$1.15t_{CY} + 40$	ns
Delay time from $\overline{WAIT}\uparrow$ to $\overline{RD}\uparrow$	$t_{WTRD}$		$1.15t_{CY} + 40$	$3.15t_{CY} + 40$	ns
Delay time from $\overline{WAIT}\uparrow$ to $\overline{WR}\uparrow$	$t_{WTWR}$		$1.15t_{CY} + 30$	$3.15t_{CY} + 30$	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3.  $t_{CY} = T_{CY}/4$
  4. n indicates the number of waits.

(b) When  $\overline{\text{MCS}} = 0$  or  $\text{PCC2 to PCC0} \neq 000\text{B}$  ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	$t_{\text{ASTH}}$		$t_{\text{CY}} - 80$		ns
Address setup time	$t_{\text{ADS}}$		$t_{\text{CY}} - 80$		ns
Address hold time	$t_{\text{ADH}}$		$0.4t_{\text{CY}} - 10$		ns
Time from address to data input	$t_{\text{ADD1}}$			$(3 + 2n)t_{\text{CY}} - 160$	ns
	$t_{\text{ADD2}}$			$(4 + 2n)t_{\text{CY}} - 200$	ns
Time from $\overline{\text{RD}}\downarrow$ to data input	$t_{\text{RDD1}}$			$(1.4 + 2n)t_{\text{CY}} - 70$	ns
	$t_{\text{RDD2}}$			$(2.4 + 2n)t_{\text{CY}} - 70$	ns
Read data hold time	$t_{\text{RDH}}$		0		ns
$\overline{\text{RD}}$ low-level width	$t_{\text{RDL1}}$		$(1.4 + 2n)t_{\text{CY}} - 20$		ns
	$t_{\text{RDL2}}$		$(2.4 + 2n)t_{\text{CY}} - 20$		ns
Time from $\overline{\text{RD}}\downarrow$ to $\overline{\text{WAIT}}\downarrow$ input	$t_{\text{RDWT1}}$			$t_{\text{CY}} - 100$	ns
	$t_{\text{RDWT2}}$			$2t_{\text{CY}} - 100$	ns
Time from $\overline{\text{WR}}\downarrow$ to $\overline{\text{WAIT}}\downarrow$ input	$t_{\text{WRWT}}$			$2t_{\text{CY}} - 100$	ns
$\overline{\text{WAIT}}$ low-level width	$t_{\text{WTL}}$		$(1 + 2n)t_{\text{CY}}$	$(2 + 2n)t_{\text{CY}}$	ns
Write data setup time	$t_{\text{WDS}}$		$(2.4 + 2n)t_{\text{CY}} - 60$		ns
Write data hold time	$t_{\text{WDH}}$		20		ns
$\overline{\text{WR}}$ low-level width	$t_{\text{WRL}}$		$(2.4 + 2n)t_{\text{CY}} - 20$		ns
Delay time from $\overline{\text{ASTB}}\downarrow$ to $\overline{\text{RD}}\downarrow$	$t_{\text{ASTRD}}$		$0.4t_{\text{CY}} - 30$		ns
Delay time from $\overline{\text{ASTB}}\downarrow$ to $\overline{\text{WR}}\downarrow$	$t_{\text{ASTWR}}$		$1.4t_{\text{CY}} - 30$		ns
Delay time from $\overline{\text{RD}}\uparrow$ to $\overline{\text{ASTB}}\uparrow$ at external fetch	$t_{\text{RDAST}}$		$t_{\text{CY}} - 10$	$t_{\text{CY}} + 20$	ns
Time from $\overline{\text{RD}}\uparrow$ to address hold at external fetch	$t_{\text{RDADH}}$		$t_{\text{CY}} - 50$	$t_{\text{CY}} + 50$	ns
Time from $\overline{\text{RD}}\uparrow$ to write data output	$t_{\text{RDWD}}$		$0.4t_{\text{CY}} - 20$		ns
Time from $\overline{\text{WR}}\downarrow$ to write data output	$t_{\text{WRWD}}$		0	60	ns
Time from $\overline{\text{WR}}\uparrow$ to address hold	$t_{\text{WRADH}}$		$t_{\text{CY}}$	$t_{\text{CY}} + 60$	ns
Delay time from $\overline{\text{WAIT}}\uparrow$ to $\overline{\text{RD}}\uparrow$	$t_{\text{WTRD}}$		$0.6t_{\text{CY}} + 180$	$2.6t_{\text{CY}} + 180$	ns
Delay time from $\overline{\text{WAIT}}\uparrow$ to $\overline{\text{WR}}\uparrow$	$t_{\text{WTWR}}$		$0.6t_{\text{CY}} + 120$	$2.6t_{\text{CY}} + 120$	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3.  $t_{\text{CY}} = T_{\text{CY}}/4$
  4. n indicates the number of waits.



(c) When MCS = 0 or PCC2 to PCC0 ≠ 000B (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 1.8 to 2.7 V)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t <sub>ASTH</sub>		t <sub>cy</sub> - 150		ns
Address setup time	t <sub>ADS</sub>		t <sub>cy</sub> - 150		ns
Address hold time	t <sub>ADH</sub>		0.37t <sub>cy</sub> - 40		ns
Time from address to data input	t <sub>ADD1</sub>			(3 + 2n)t <sub>cy</sub> - 320	ns
	t <sub>ADD2</sub>			(4 + 2n)t <sub>cy</sub> - 300	ns
Time from RD↓ to data input	t <sub>RDD1</sub>			(1.37 + 2n)t <sub>cy</sub> - 120	ns
	t <sub>RDD2</sub>			(2.37 + 2n)t <sub>cy</sub> - 120	ns
Read data hold time	t <sub>RDH</sub>		0		ns
RD low-level width	t <sub>RDL1</sub>		(1.37 + 2n)t <sub>cy</sub> - 20		ns
	t <sub>RDL2</sub>		(2.37 + 2n)t <sub>cy</sub> - 20		ns
Time from RD↓ to WAIT↓ input	t <sub>RDWT1</sub>			t <sub>cy</sub> - 200	ns
	t <sub>RDWT2</sub>			2t <sub>cy</sub> - 200	ns
Time from WR↓ to WAIT↓ input	t <sub>WRWT</sub>			2t <sub>cy</sub> - 200	ns
WAIT low-level width	t <sub>WTL</sub>		(1 + 2n)t <sub>cy</sub>	(2 + 2n)t <sub>cy</sub>	ns
Write data setup time	t <sub>WDS</sub>		(2.37 + 2n)t <sub>cy</sub> - 100		ns
Write data hold time	t <sub>WDH</sub>		20		ns
WR low-level width	t <sub>WRL</sub>		(2.37 + 2n)t <sub>cy</sub> - 20		ns
Delay time from ASTB↓ to RD↓	t <sub>ASTRD</sub>		0.37t <sub>cy</sub> - 50		ns
Delay time from ASTB↓ to WR↓	t <sub>ASTWR</sub>		1.37t <sub>cy</sub> - 50		ns
Delay time from RD↑ to ASTB at external fetch	t <sub>RDAST</sub>		t <sub>cy</sub> - 10	t <sub>cy</sub> + 20	ns
Time from RD↑ to address hold at external fetch	t <sub>RDADH</sub>		t <sub>cy</sub> - 50	t <sub>cy</sub> + 50	ns
Time from RD↑ to write data output	t <sub>RDWD</sub>		0.37t <sub>cy</sub> - 40		ns
Time from WR↓ to write data output	t <sub>WRWD</sub>		0	120	ns
Time from WR↑ to address hold	t <sub>WRADH</sub>		t <sub>cy</sub>	t <sub>cy</sub> + 120	ns
Delay time from WAIT↑ to RD↑	t <sub>WTRD</sub>		0.63t <sub>cy</sub> + 350	2.63t <sub>cy</sub> + 350	ns
Delay time from WAIT↑ to WR↑	t <sub>WTWR</sub>		0.63t <sub>cy</sub> + 240	2.63t <sub>cy</sub> + 240	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3. t<sub>cy</sub> = T<sub>cy</sub>/4
  4. n indicates the number of waits.

**(3) Serial interface ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V)**
**(a) Serial interface channel 0**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY1}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH1}}, t_{\text{KL1}}$	$V_{\text{DD}} = 4.5$ to $5.5 \text{ V}$	$t_{\text{KCY1}}/2 - 50$			ns
		$V_{\text{DD}} = 1.8$ to $5.5 \text{ V}$	$t_{\text{KCY1}}/2 - 100$			ns
SIO setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK1}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	400			ns
SIO hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI1}}$		400			ns
Delay time from $\overline{\text{SCK0}}\downarrow$ to SO0 output	$t_{\text{KSO1}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK0}}$  and SO0 output lines.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY2}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH2}}, t_{\text{KL2}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	2,400			ns
SIO setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK2}}$	$2.0 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	150			ns
SIO hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI2}}$		400			ns
Delay time from $\overline{\text{SCK0}}\downarrow$ to SO0 output	$t_{\text{KSO2}}$	$C = 100 \text{ pF}^{\text{Note}}$ $V_{\text{DD}} = 2.0$ to $5.5\text{V}$			300	ns
		$V_{\text{DD}} = 1.8$ to $5.5\text{V}$			500	ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{R2}}, t_{\text{F2}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO0 output line.

(iii) 2-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY3}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	1,600		ns
			2.0 V $\leq$ V <sub>DD</sub> < 2.7 V	3,200		ns
			1.8 V $\leq$ V <sub>DD</sub> < 2.0 V	4,800		ns
$\overline{\text{SCK0}}$ high-level width	$t_{\text{KH3}}$	V <sub>DD</sub> = 2.7 to 5.5 V	$t_{\text{KCY3}}/2 - 160$			ns
		V <sub>DD</sub> = 1.8 to 5.5 V	$t_{\text{KCY3}}/2 - 190$			ns
$\overline{\text{SCK0}}$ low-level width	$t_{\text{KL3}}$	V <sub>DD</sub> = 4.5 to 5.5 V	$t_{\text{KCY3}}/2 - 50$			ns
		V <sub>DD</sub> = 1.8 to 5.5 V	$t_{\text{KCY3}}/2 - 100$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK3}}$	4.5 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	300			ns
		2.7 V $\leq$ V <sub>DD</sub> < 4.5 V	350			ns
		2.0 V $\leq$ V <sub>DD</sub> < 2.7 V	400			ns
		1.8 V $\leq$ V <sub>DD</sub> < 2.0 V	500			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI3}}$		600			ns
						ns
Delay time from $\overline{\text{SCK0}}\downarrow$ to SB0, SB1 output	$t_{\text{KSO3}}$		0		300	ns

**Note** R and C are the load resistance and load capacitance of the  $\overline{\text{SCK0}}$ , SB0, and SB1 output lines.

 (iv) 2-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY4}}$	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	1,600			ns	
		2.0 V $\leq$ V <sub>DD</sub> < 2.7 V	3,200			ns	
		1.8 V $\leq$ V <sub>DD</sub> < 2.0 V	4,800			ns	
$\overline{\text{SCK0}}$ high-level width	$t_{\text{KH4}}$	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	650			ns	
		2.0 V $\leq$ V <sub>DD</sub> < 2.7 V	1,300			ns	
		1.8 V $\leq$ V <sub>DD</sub> < 2.0 V	2,100			ns	
$\overline{\text{SCK0}}$ low-level width	$t_{\text{KL4}}$	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	800			ns	
		2.0 V $\leq$ V <sub>DD</sub> < 2.7 V	1,600			ns	
		1.8 V $\leq$ V <sub>DD</sub> < 2.0 V	2,400			ns	
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK4}}$	V <sub>DD</sub> = 2.0 to 5.5 V	100			ns	
		V <sub>DD</sub> = 1.8 to 5.5 V	150			ns	
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI4}}$		$t_{\text{KCY4}}/2$			ns	
Delay time from $\overline{\text{SCK0}}\downarrow$ to SB0, SB1 output	$t_{\text{KSO4}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	4.5 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	0		300	ns
			2.0 V $\leq$ V <sub>DD</sub> < 4.5 V	0		500	ns
			1.8 V $\leq$ V <sub>DD</sub> < 2.0 V	0		800	ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{r4}}, t_{\text{f4}}$	When using external device expansion function			160	ns	
		When not using external device expansion function			1,000	ns	

**Note** R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

**(v) SBI mode ( $\overline{\text{SCK0}}$  ... Internal clock output) ( $\mu\text{PD78005x}$  only)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		800			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$		3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$		4,800			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH5}}, t_{\text{KL5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		$t_{\text{KCY5}}/2 - 50$			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$		$t_{\text{KCY5}}/2 - 150$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		100			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$		300			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$		400			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI5}}$			$t_{\text{KCY5}}/2$			ns
Delay time from $\overline{\text{SCK0}}\downarrow$ to SB0, SB1 output	$t_{\text{KSO5}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	0		250	ns
			$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	0		1,000	ns
SB0, SB1 $\downarrow$ from $\overline{\text{SCK0}}\uparrow$	$t_{\text{KSB}}$			$t_{\text{KCY5}}$			ns
$\overline{\text{SCK0}}\downarrow$ from SB0, SB1 $\downarrow$	$t_{\text{SBK}}$			$t_{\text{KCY5}}$			ns
SB0, SB1 high-level width	$t_{\text{SBH}}$			$t_{\text{KCY5}}$			ns
SB0, SB1 low-level width	$t_{\text{SBL}}$			$t_{\text{KCY5}}$			ns

**Note** R and C are the load resistance and load capacitance of the  $\overline{\text{SCK0}}$ , SB0, and SB1 output lines.

**(vi) SBI mode ( $\overline{\text{SCK0}}$  ... External clock input) ( $\mu\text{PD78005x}$  only)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		800			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$		3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$		4,800			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH6}}, t_{\text{KL6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		400			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$		1,600			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$		2,400			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		100			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$		300			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$		400			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI6}}$			$t_{\text{KCY6}}/2$			ns
Delay time from $\overline{\text{SCK0}}\downarrow$ to SB0, SB1 output	$t_{\text{KSO6}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	0		300	ns
			$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	0		1,000	ns
SB0, SB1 $\downarrow$ from $\overline{\text{SCK0}}\uparrow$	$t_{\text{KSB}}$			$t_{\text{KCY6}}$			ns
$\overline{\text{SCK0}}\downarrow$ from SB0, SB1 $\downarrow$	$t_{\text{SBK}}$			$t_{\text{KCY6}}$			ns
SB0, SB1 high-level width	$t_{\text{SBH}}$			$t_{\text{KCY6}}$			ns
SB0, SB1 low-level width	$t_{\text{SBL}}$			$t_{\text{KCY6}}$			ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{r6}}, t_{\text{f6}}$	When using external device expansion function				160	ns
		When not using external device expansion function				1,000	ns

**Note** R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

(vii) I<sup>2</sup>C bus mode (SCL ... Internal clock output) (μPD78005xY only)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
SCL cycle time	t <sub>KCY7</sub>	R = 1 KΩ, C = 100 pF <sup>Note</sup>	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	10		μs
			2.0 V ≤ V <sub>DD</sub> < 2.7 V	20		μs
			1.8 V ≤ V <sub>DD</sub> < 2.0 V	30		μs
SCL high-level width	t <sub>KH7</sub>		V <sub>DD</sub> = 2.7 to 5.5 V	t <sub>KCY7</sub> – 160		ns
			V <sub>DD</sub> = 1.8 to 5.5 V	t <sub>KCY7</sub> – 190		ns
SCL low-level width	t <sub>KL7</sub>		V <sub>DD</sub> = 4.5 to 5.5 V	t <sub>KCY7</sub> – 50		ns
			V <sub>DD</sub> = 1.8 to 5.5 V	t <sub>KCY7</sub> – 100		ns
SDA0, SDA1 setup time (to SCL↑)	t <sub>SIK7</sub>		2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	200		ns
			2.0 V ≤ V <sub>DD</sub> < 2.7 V	300		ns
			1.8 V ≤ V <sub>DD</sub> < 2.0 V	400		ns
SDA0, SDA1 hold time (from SCL↓)	t <sub>KSI7</sub>		0			ns
Delay time from SCL↓ to SDA0, SDA1 output	t <sub>KSO7</sub>		4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0	300	ns
			2.0 V ≤ V <sub>DD</sub> < 4.5 V	0	500	ns
			1.8 V ≤ V <sub>DD</sub> < 2.0 V	0	600	ns
SDA0, SDA1↓ from SCL↑ or SDA0, SDA1↑ from SCL↓	t <sub>KSB</sub>		200			ns
SCL↓ from SDA0, SDA1↓	t <sub>SBK</sub>		V <sub>DD</sub> = 2.0 to 5.5 V	400		ns
			V <sub>DD</sub> = 1.8 to 5.5 V	500		ns
SDA0, SDA1 high-level width	t <sub>SBH</sub>		500			ns

**Note** R and C are the load resistance and load capacitance of the SCL, SDA0, and SDA1 output lines.

 (viii) I<sup>2</sup>C bus mode (SCL ... External clock input) (μPD78005xY only)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
SCL cycle time	t <sub>KCY8</sub>		1,000			ns
SCL high-/low-level width	t <sub>KH8</sub> , t <sub>KL8</sub>	V <sub>DD</sub> = 2.0 to 5.5 V	400			ns
		V <sub>DD</sub> = 1.8 to 5.5 V	600			ns
SDA0, SDA1 setup time (to SCL↑)	t <sub>SIK8</sub>	V <sub>DD</sub> = 2.0 to 5.5 V	200			ns
		V <sub>DD</sub> = 1.8 to 5.5 V		300		ns
SDA0, SDA1 hold time (from SCL↓)	t <sub>KSI8</sub>		0			ns
Delay time from SCL↓ to SDA0, SDA1 output	t <sub>KSO8</sub>	R = 1 kΩ, C = 100 pF <sup>Note</sup>	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0	300	ns
			2.0 V ≤ V <sub>DD</sub> < 4.5 V	0	500	ns
			1.8 V ≤ V <sub>DD</sub> < 2.0 V	0	600	ns
SDA0, SDA1↓ from SCL↑ or SDA0, SDA1↑ from SCL↑	t <sub>KSB</sub>		200			ns
SCL↓ from SDA0, SDA1↓	t <sub>SBK</sub>	V <sub>DD</sub> = 2.0 to 5.5 V	400			ns
		V <sub>DD</sub> = 1.8 to 5.5 V	500			ns
SDA0, SDA1 high-level width	t <sub>SBH</sub>	V <sub>DD</sub> = 2.0 to 5.5 V	500			ns
		V <sub>DD</sub> = 1.8 to 5.5 V	800			ns
SCL rise/fall time	t <sub>RE</sub> , t <sub>FE</sub>	When using external device expansion function			160	ns
		When not using external device expansion function			1	μs

**Note** R and C are the load resistance and load capacitance of the SDA0 and SDA1 output lines.

**(b) Serial interface channel 1**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK1}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY9}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK1}}$ high/low-level width	$t_{\text{KH9}}, t_{\text{KL9}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY9}}/2 - 50$			ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY9}}/2 - 100$			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK9}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	400			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSI9}}$		400			ns
Delay time from $\overline{\text{SCK1}}\downarrow$ to SO1 output	$t_{\text{KSO9}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK1}}$  and SO1 output lines.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK1}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY10}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK1}}$ high/low-level width	$t_{\text{KH10}}, t_{\text{KL10}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	2,400			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK10}}$	$V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$	100			ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	150			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KIS10}}$		400			ns
Delay time from $\overline{\text{SCK1}}\downarrow$ to SO1 output	$t_{\text{KSO10}}$	$C = 100 \text{ pF}^{\text{Note}}$ $V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$			300	ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$			500	ns
$\overline{\text{SCK1}}$ rise/fall time	$t_{\text{R10}}, t_{\text{F10}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO1 output line.

(iii) 3-wire serial I/O mode with automatic transmit/receive function ( $\overline{\text{SCK1}}$  ... Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY11}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH11}}, t_{\text{KL11}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY11}}/2 - 50$			ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY11}}/2 - 100$			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK11}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	400			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SH11}}$		400			ns
Delay time from $\overline{\text{SCK1}}\downarrow$ to SO1 output	$t_{\text{KSO11}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
STB $\uparrow$ from $\overline{\text{SCK1}}\uparrow$	$t_{\text{SBD}}$		$t_{\text{KCY11}}/2 - 100$		$t_{\text{KCY11}}/2 + 100$	ns
Strobe signal high-level width	$t_{\text{SBW}}$	$2.7 \text{ V} \leq V_{\text{DD}} < 5.5 \text{ V}$	$t_{\text{KCY11}} - 30$		$t_{\text{KCY11}} + 30$	ns
		$2.0 \text{ V} < V_{\text{DD}} < 2.7 \text{ V}$	$t_{\text{KCY11}} - 60$		$t_{\text{KCY11}} + 60$	ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	$t_{\text{KCY11}} - 90$		$t_{\text{KCY11}} + 90$	ns
Busy signal setup time (to busy signal detection timing)	$t_{\text{BYS}}$		100			ns
Busy signal hold time (from busy signal detection timing)	$t_{\text{BYH}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	300			ns
$\overline{\text{SCK1}}\downarrow$ from busy inactive	$t_{\text{SPS}}$				$2t_{\text{KCY11}}$	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK1}}$  and SO1 output lines.

(iv) 3-wire serial I/O mode with automatic transmit/receive function ( $\overline{\text{SCK1}}$ ...External clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY12}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH12}}$ , $t_{\text{KL12}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	2,400			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK12}}$	$V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$	100			ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	150			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSH12}}$		400			ns
Delay time from $\overline{\text{SCK1}}\downarrow$ to SO1 output	$t_{\text{KSO12}}$	C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$		300	ns
			$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$		500	ns
$\overline{\text{SCK1}}$ rise/fall time	$t_{\text{R12}}, t_{\text{F12}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO1 output line.



## (c) Serial interface channel 2

 (i) 3-wire serial I/O mode ( $\overline{\text{SCK2}}$ ...Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK2}}$ cycle time	$t_{\text{KCY13}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK2}}$ high-/low-level width	$t_{\text{KH13}},$ $t_{\text{KL13}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY13}}/2 - 50$			ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY13}}/2 - 100$			ns
SI2 setup time (to $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{SIK13}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	400			ns
SI2 hold time (from $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{KSI13}}$		400			ns
Delay time from $\overline{\text{SCK2}}\downarrow$ to SO2 output	$t_{\text{KSO13}}$	$C = 100 \text{ pF}$ <sup>Note</sup>			300	ns

**Note** C is the load capacitance of the SO2 output line.

 (ii) 3-wire serial I/O mode ( $\overline{\text{SCK2}}$ ...External clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK2}}$ cycle time	$t_{\text{KCY14}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	4,800			ns
$\overline{\text{SCK2}}$ high-/low-level width	$t_{\text{KH14}},$ $t_{\text{KL14}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
		$2.0 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
		$1.8 \text{ V} \leq V_{\text{DD}} < 2.0 \text{ V}$	2,400			ns
SI2 setup time (to $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{SIK14}}$	$V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$	100			ns
		$V_{\text{DD}} = 1.8 \text{ to } 5.5 \text{ V}$	150			ns
SI2 hold time (from $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{KSI14}}$		400			ns
Delay time from $\overline{\text{SCK2}}\downarrow$ to SO2 output	$t_{\text{KSO14}}$	$C = 100 \text{ pF}$ <sup>Note</sup>	$V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$		300	ns
			$V_{\text{DD}} = 2.0 \text{ to } 5.5 \text{ V}$		500	ns
$\overline{\text{SCK2}}$ rise/fall time	$t_{\text{r14}},$ $t_{\text{f14}}$	Other than below			160	ns
		$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$ When not using external device expansion function			1	$\mu\text{s}$

**Note** C is the load capacitance of the SO2 output line.

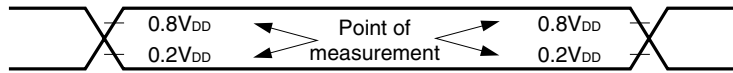
(iii) UART mode (dedicated baud rate generator output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			78,125	bps
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$			39,063	bps
		$2.0\text{ V} \leq V_{DD} < 2.7\text{ V}$			19,531	bps
		$1.8\text{ V} \leq V_{DD} < 2.0\text{ V}$			9,766	bps

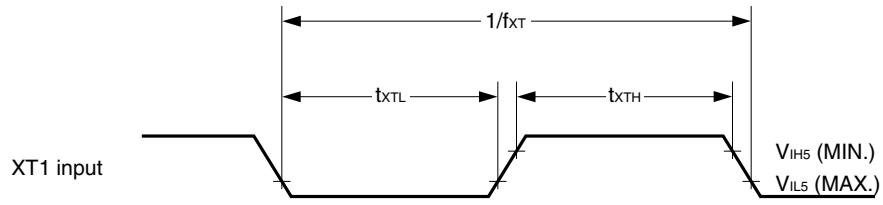
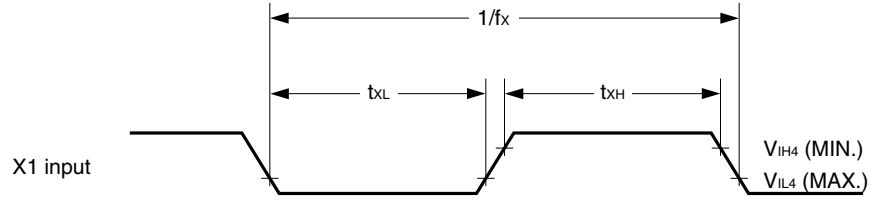
(iv) UART mode (external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK cycle time	$t_{KCY15}$	$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	800			ns
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$	1,600			ns
		$2.0\text{ V} \leq V_{DD} < 2.7\text{ V}$	3,200			ns
		$1.8\text{ V} \leq V_{DD} < 2.0\text{ V}$	4,800			ns
ASCK high-/low-level width	$t_{KH15}$ ,	$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	400			ns
	$t_{KL15}$	$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$	800			ns
		$2.0\text{ V} \leq V_{DD} < 2.7\text{ V}$	1,600			ns
		$1.8\text{ V} \leq V_{DD} < 2.0\text{ V}$	2,400			ns
Transfer rate		$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			39,063	bps
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$			19,531	bps
		$2.0\text{ V} \leq V_{DD} < 2.7\text{ V}$			9,766	bps
		$1.8\text{ V} \leq V_{DD} < 2.0\text{ V}$			6,510	bps
ASCK rise/fall time	$t_{R15}$ ,	$V_{DD} = 4.5\text{ to }5.5\text{ V}$ , when not using external device expansion function.			1,000	ns
	$t_{F15}$					
		Other than above			160	ns

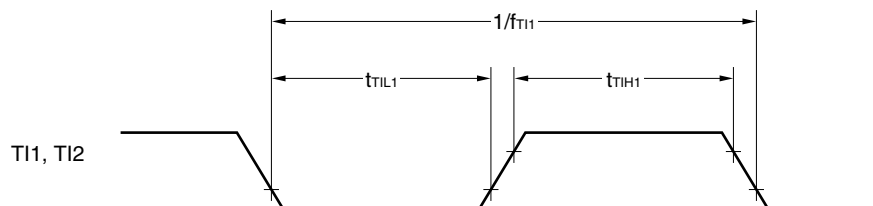
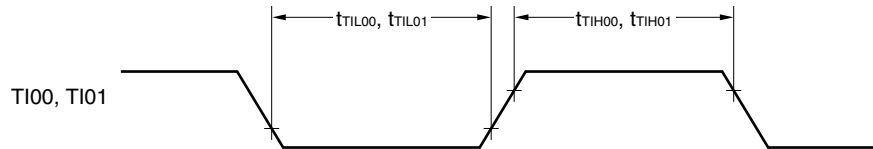
AC Timing Measurement Points (Excluding X1, XT1 Inputs)



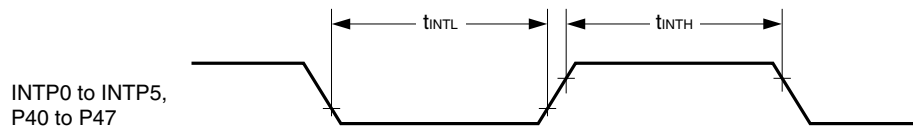
Clock Timing



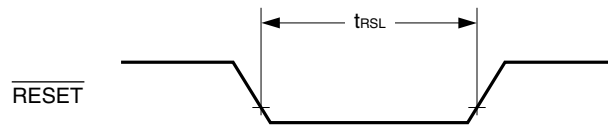
TI Timing



### Interrupt Request Input Timing

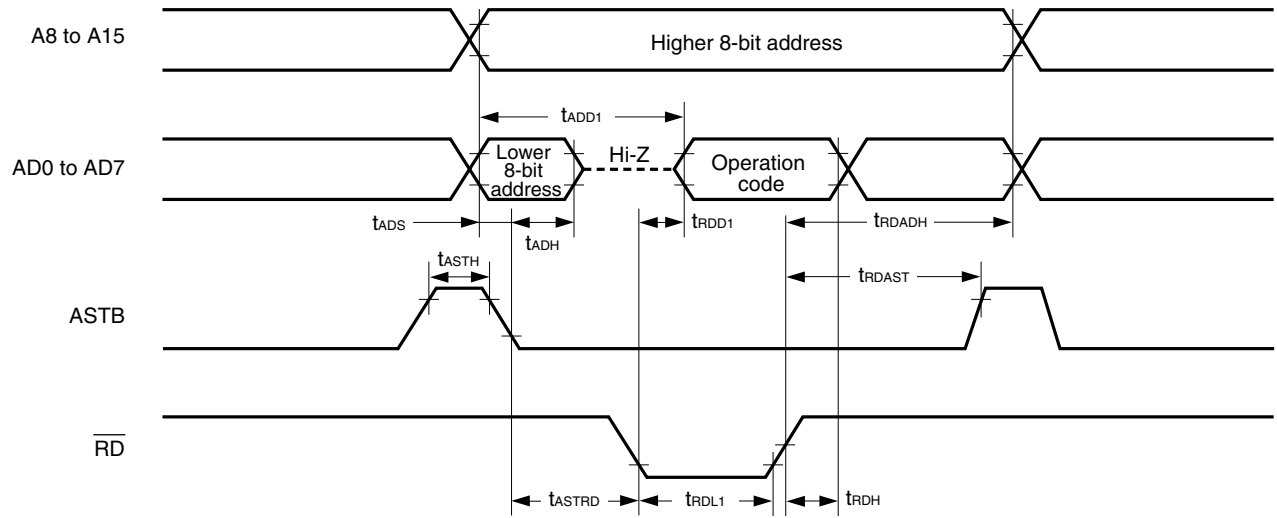


### $\overline{\text{RESET}}$ Input Timing

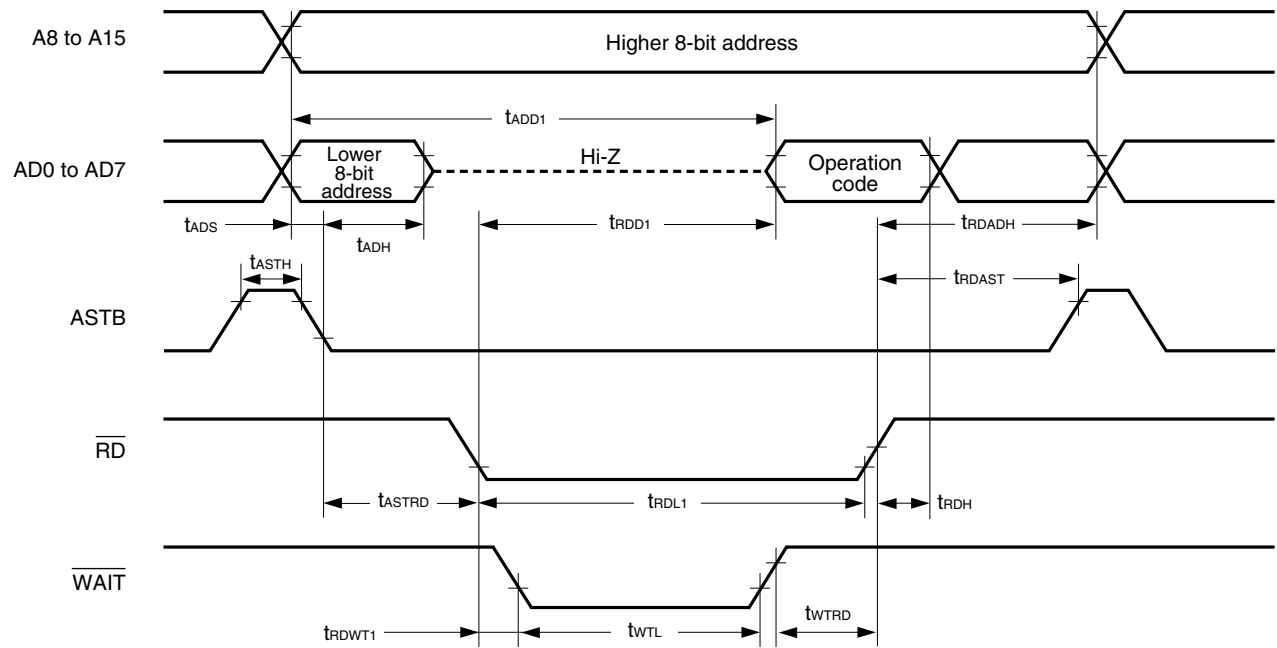


### Read/Write Operation

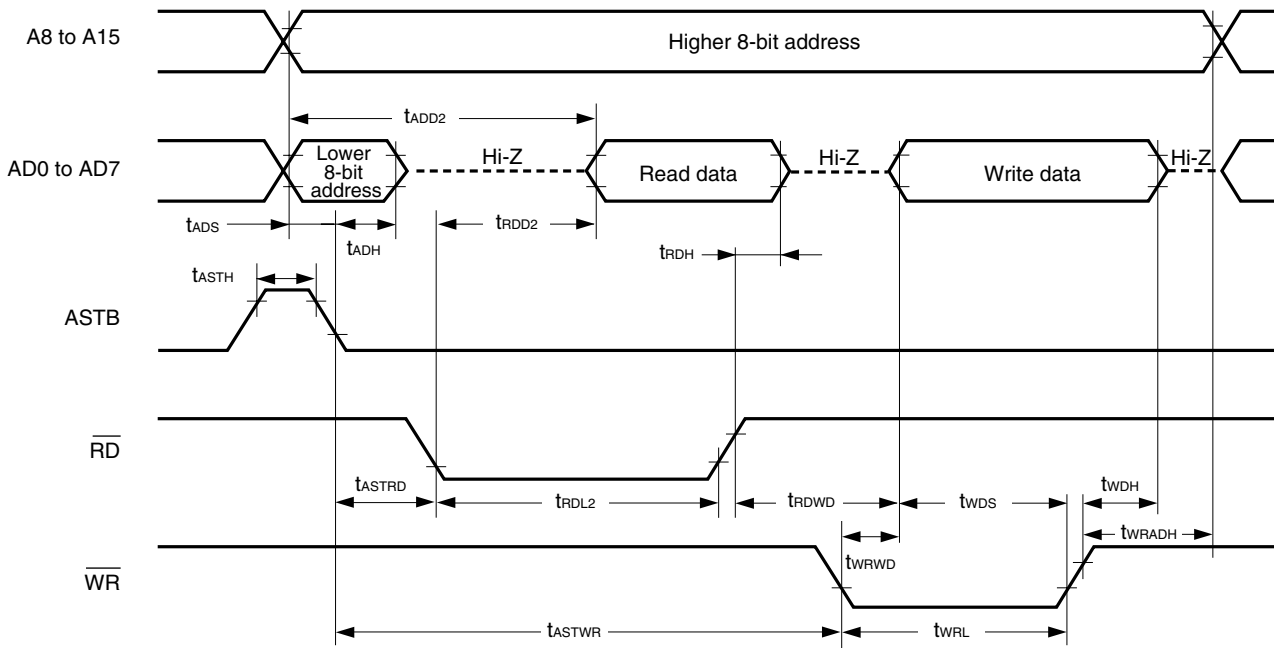
#### External fetch (no wait):



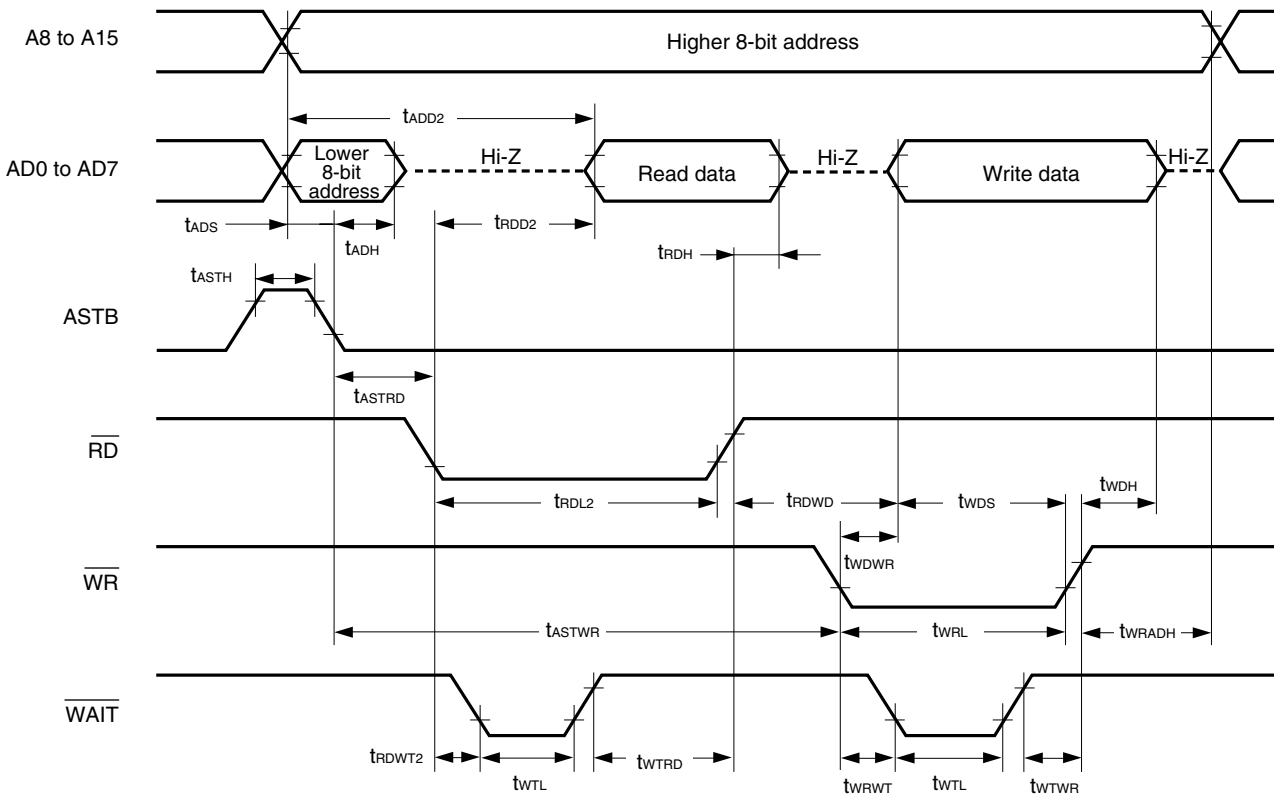
#### External fetch (wait insertion):



External data access (no wait):

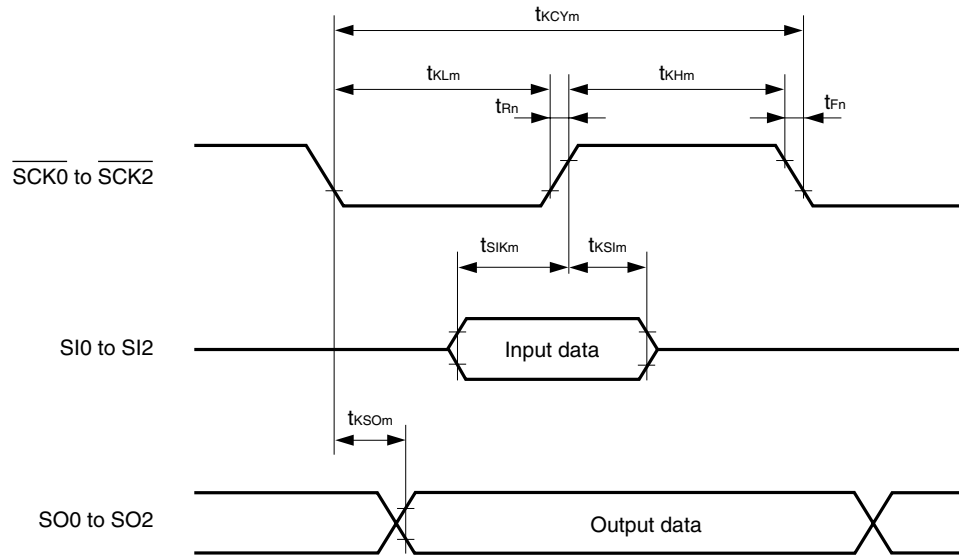


External data access (wait insertion):



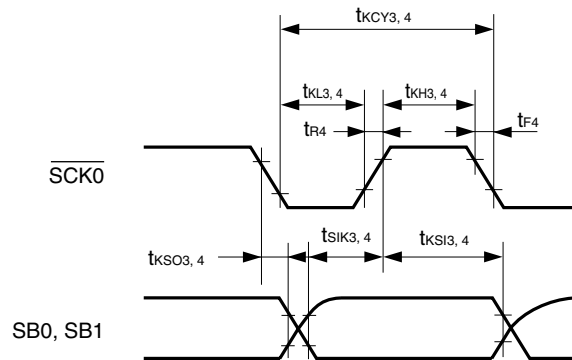
Serial Transfer Timing

3-wire serial I/O mode:

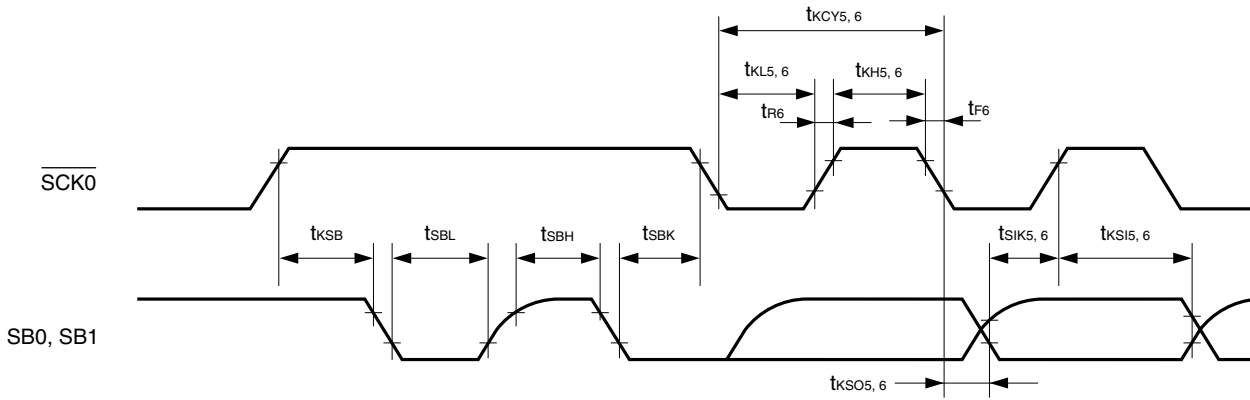


$m = 1, 2, 9, 10, 13, 14$   
 $n = 2, 10, 14$

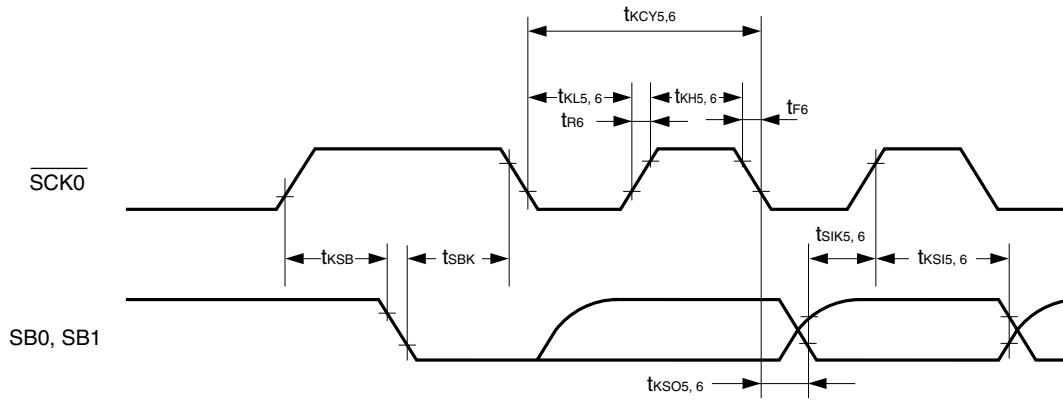
2-wire serial I/O mode:



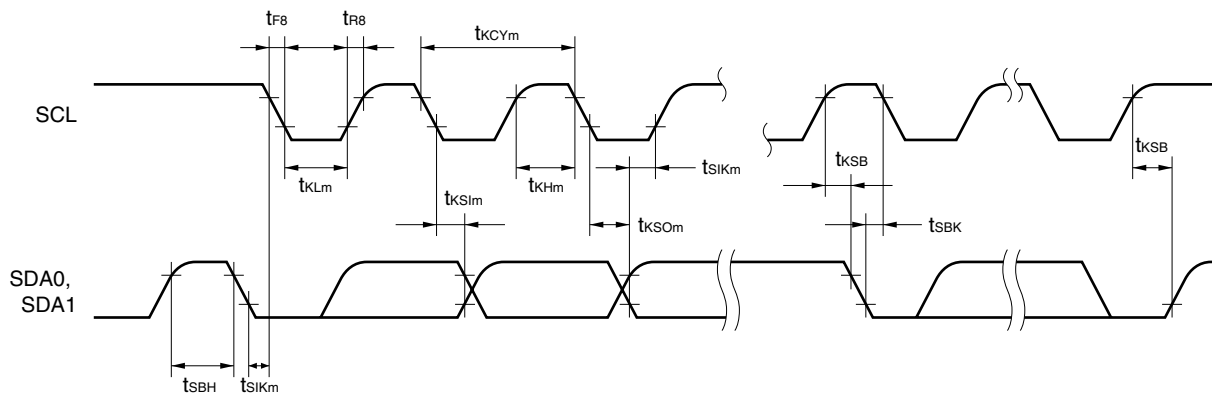
**SBI mode (bus release signal transfer):**



**SBI mode (command signal transfer):**



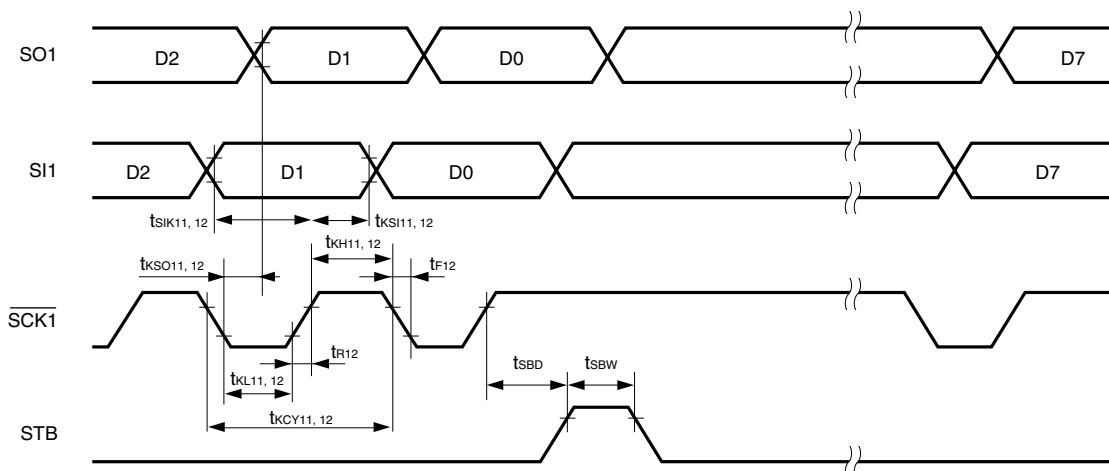
**I<sup>2</sup>C bus mode:**



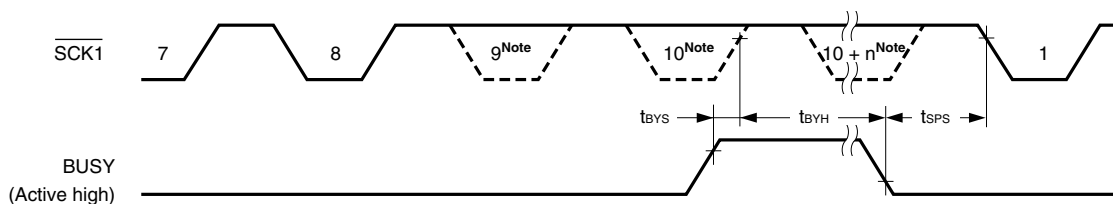
$m = 7, 8$



**3-wire serial I/O mode with automatic transmit/receive function:**

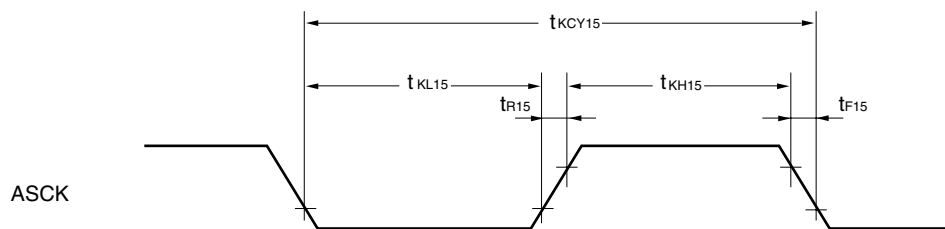


**3-wire serial I/O mode with automatic transmit/receive function (busy processing):**



**Note** The signal is not actually driven low here; it is shown as such to indicate the timing.

**UART mode (external clock input):**



**A/D Converter Characteristics**

( $\mu$ PD780053, 780053(A), 780054, 780054(A), 780055, 780055(A), 780056, 780056(A), 780058B, 780058B(A), 780053Y, 780053Y(A), 780054Y, 780054Y(A), 780055Y, 780055Y(A), 780056Y, 780056Y(A), 780058BY, 780058BY(A))

( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V,  $AV_{SS} = V_{SS} = 0$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error <sup>Note 1</sup>		$1.8\text{ V} \leq AV_{REF0} < 2.7\text{ V}$			$\pm 1.4$	%FSR
		$2.7\text{ V} \leq AV_{REF0} \leq 5.5\text{ V}$			$\pm 0.6$	%FSR
Conversion time	$T_{CONV1}$	$1.8\text{ V} \leq AV_{REF0} < 2.7\text{ V}$	40		100	$\mu\text{s}$
	$T_{CONV2}$	$2.7\text{ V} \leq AV_{REF0} \leq 5.5\text{ V}$	16		100	$\mu\text{s}$
Analog input voltage	$V_{IAN}$		$AV_{SS}$		$AV_{REF0}$	V
Reference voltage	$AV_{REF0}$		1.8		$V_{DD}$	V
$AV_{REF0}$ current	$I_{REF0}$	When A/D converter is operating <sup>Note 2</sup>		500	1,500	$\mu\text{A}$
		When A/D converter is not operating <sup>Note 3</sup>		0	3	$\mu\text{A}$

**Notes** 1. Excludes quantization error ( $\pm 1/2$  LSB). This value is indicated as a ratio to the full-scale value (%FSR).

2. The current flowing to the  $AV_{REF0}$  pin when bit 7 (CS) of the A/D converter mode register (ADM) is 1.

3. The current flowing to the  $AV_{REF0}$  pin when bit 7 (CS) of the A/D converter mode register (ADM) is 0.

**A/D Converter Characteristics ( $\mu$ PD780058)**

( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V,  $AV_{SS} = V_{SS} = 0$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error <sup>Note 1</sup>					$\pm 0.6$	%FSR
Conversion time	$T_{CONV}$		16		100	$\mu\text{s}$
Analog input voltage	$V_{IAN}$		$AV_{SS}$		$AV_{REF0}$	V
Reference voltage	$AV_{REF0}$		2.7		$V_{DD}$	V
$AV_{REF0}$ current	$I_{REF0}$	When A/D converter is operating <sup>Note 2</sup>		500	1,500	$\mu\text{A}$
		When A/D converter is not operating <sup>Note 3</sup>		0	3	$\mu\text{A}$

**Notes** 1. Excludes quantization error ( $\pm 1/2$  LSB). This value is indicated as a ratio to the full-scale value (%FSR).

2. The current flowing to the  $AV_{REF0}$  pin when bit 7 (CS) of the A/D converter mode register (ADM) is 1.

3. The current flowing to the  $AV_{REF0}$  pin when bit 7 (CS) of the A/D converter mode register (ADM) is 0.

**Caution** The operating voltage range of the A/D converter and D/A converter of the  $\mu$ PD780058 is  $V_{DD} = 2.7$  to  $5.5$  V.

**D/A Converter Characteristics**

( $\mu$ PD780053, 780053(A), 780054, 780054(A), 780055, 780055(A), 780056, 780056(A), 780058B, 780058B(A), 780053Y, 780053Y(A), 780054Y, 780054Y(A), 780055Y, 780055Y(A), 780056Y, 780056Y(A), 780058BY, 780058BY(A))

( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 1.8$  to  $5.5$  V,  $AV_{SS} = V_{SS} = 0$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Resolution					8	bit	
Overall error		$R = 2\text{ M}\Omega$ <sup>Note 1</sup>			$\pm 1.2$	%	
		$R = 4\text{ M}\Omega$ <sup>Note 1</sup>			$\pm 0.8$	%	
		$R = 10\text{ M}\Omega$ <sup>Note 1</sup>			$\pm 0.6$	%	
Settling time		$C = 30\text{ pF}$ <sup>Note 1</sup>	$AV_{REF1} = 1.8$ to $2.7$ V			10	$\mu\text{s}$
			$AV_{REF1} = 1.8$ to $5.5$ V			15	$\mu\text{s}$
Output resistance	$R_o$	<b>Note 2</b>		8		$\text{k}\Omega$	
Analog reference voltage	$AV_{REF1}$		1.8		$V_{DD}$	V	
$AV_{REF1}$ current	$I_{REF1}$	<b>Note 2</b>			2.5	mA	
Resistance between $AV_{REF1}$ and $AV_{SS}$	$R_{AIREF1}$	DACS0, DACS1 = $55\text{H}$ <sup>Note 2</sup>	4	8		$\text{k}\Omega$	

**Notes** 1. R and C are the D/A converter output pin load resistance and load capacitance, respectively.

2. Value for one D/A converter channel

**Remark** DACS0 and DACS1: D/A conversion value setting registers 0, 1

**D/A Converter Characteristics ( $\mu$ PD780058)**

( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V,  $AV_{SS} = V_{SS} = 0$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution					8	bit
Overall error		$R = 2\text{ M}\Omega$ <sup>Note 1</sup>			$\pm 1.2$	%
		$R = 4\text{ M}\Omega$ <sup>Note 1</sup>			0.8	%
		$R = 10\text{ M}\Omega$ <sup>Note 1</sup>			0.6	%
Settling time		$C = 30\text{ pF}$ <sup>Note 1</sup>			15	$\mu\text{s}$
Output resistance	$R_o$	<b>Note 2</b>		8		$\text{k}\Omega$
Analog reference voltage	$AV_{REF1}$		2.7		$V_{DD}$	V
$AV_{REF1}$ current	$I_{REF1}$	<b>Note 2</b>			2.5	mA
Resistance between $AV_{REF1}$ and $AV_{SS}$	$R_{AIREF1}$	DACS0, DACS1 = $55\text{H}$ <sup>Note 2</sup>	4	8		$\text{k}\Omega$

**Notes** 1. R and C are the D/A converter output pin load resistance and load capacitance, respectively.

2. Value for one D/A converter channel

**Remark** DACS0 and DACS1: D/A conversion value setting registers 0, 1

**Caution** The operating voltage range of the A/D converter and D/A converter of the  $\mu$ PD780058 is  $V_{DD} = 2.7$  to  $5.5$  V.

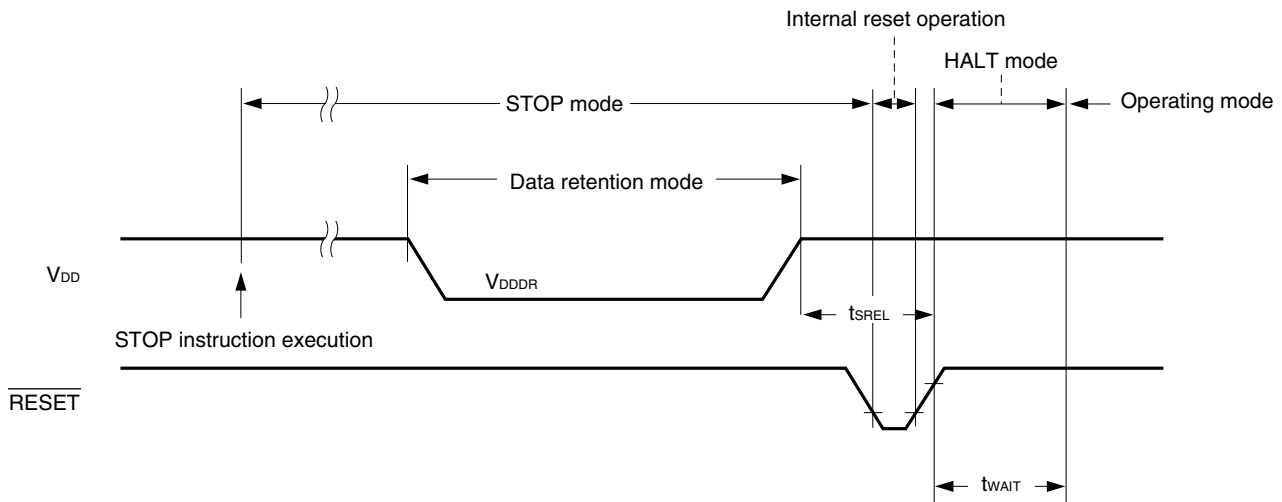
**Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (TA = -40 to +85°C)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention supply voltage	V <sub>DDDR</sub>		1.8		5.5	V
Data retention supply current	I <sub>DDDR</sub>	V <sub>DDDR</sub> = 1.8 V Subsystem clock stop and feed-back resistor disconnected		0.1	10	μA
Release signal set time	t <sub>SREL</sub>		0			μs
Oscillation stabilization wait time	t <sub>WAIT</sub>	Release by $\overline{\text{RESET}}$		2 <sup>17</sup> /f <sub>x</sub>		ms
		Release by interrupt request		<b>Note</b>		ms

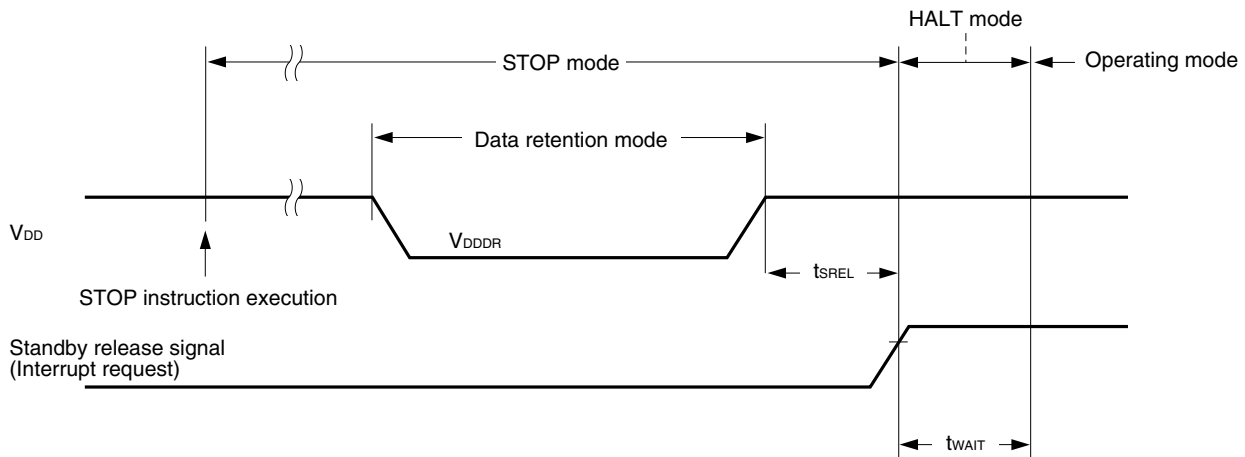
**Note** Selection of 2<sup>12</sup>/f<sub>xx</sub> and 2<sup>14</sup>/f<sub>xx</sub> to 2<sup>17</sup>/f<sub>xx</sub> is possible with bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register (OSTS).

**Remark** f<sub>xx</sub>: Main system clock frequency (f<sub>x</sub> or f<sub>x</sub>/2)  
f<sub>x</sub>: Main system clock oscillation frequency

**Data Retention Timing (STOP Mode Release by  $\overline{\text{RESET}}$ )**



**Data Retention Timing (Standby Release Signal: STOP Mode Release by Interrupt Request Signal)**





**CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)**

**Absolute Maximum Ratings (T<sub>A</sub> = 25°C)**

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V <sub>DD</sub>			-0.3 to +6.5	V
	V <sub>PP</sub>	<b>Note 1</b>		-0.3 to +10.5	V
	AV <sub>REF0</sub>			-0.3 to V <sub>DD</sub> + 0.3	V
	AV <sub>REF1</sub>			-0.3 to V <sub>DD</sub> + 0.3	V
	AV <sub>SS</sub>			-0.3 to +0.3	V
Input voltage	V <sub>I1</sub>	P00 to P05, P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131, X1, X2, XT2, RESET		-0.3 to V <sub>DD</sub> + 0.3	V
	V <sub>I2</sub>	P60 to P63	N-ch open drain	-0.3 to +16	V
Output voltage	V <sub>O</sub>			-0.3 to V <sub>DD</sub> + 0.3	V
Analog input voltage	V <sub>AN</sub>	P10 to P17	Analog input pin	AV <sub>SS</sub> - 0.3 to AV <sub>REF0</sub> + 0.3	V
Output current, high	I <sub>OH</sub>	Per pin		-10	mA
		Total for P01 to P05, P30 to P37, P56, P57, P60 to P67, P120 to P127		-15	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P50 to P55, P70 to P72, P130, P131		-15	mA
Output current, low	I <sub>OL</sub> <sup>Note 2</sup>	Per pin for other than P50 to P57, P60 to P63	Peak value	20	mA
			rms value	10	mA
		Per pin for P50 to P57, P60 to P63	Peak value	30	mA
			rms value	15	mA
		Total for P50 to P55	Peak value	100	mA
			rms value	70	mA
		Total for P56, P57, P60 to P63	Peak value	100	mA
			rms value	70	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P70 to P72, P130, P131	Peak value	50	mA
			rms value	20	mA
		Total for P01 to P05, P30 to P37, P64 to P67, P120 to P127	Peak value	50	mA
			rms value	20	mA
Operating ambient temperature	T <sub>A</sub>	During normal operation		-40 to +85	°C
		During flash memory programming		10 to 40	°C
Storage temperature	T <sub>stg</sub>			-65 to +125	°C

(The Note is described on the next page.)

**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.

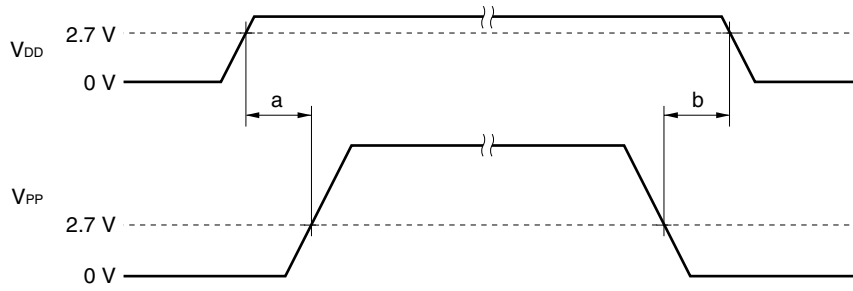
**Notes 1.** Make sure that the following conditions of the  $V_{PP}$  voltage application timing are satisfied when the flash memory is written.

- When supply voltage rises

$V_{PP}$  must exceed  $V_{DD}$  10  $\mu$ s or more after  $V_{DD}$  has reached the lower-limit value (2.7 V) of the operating voltage range (see a in the figure below).

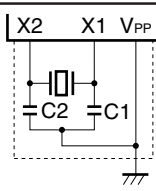
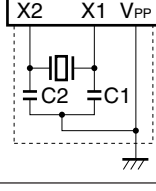
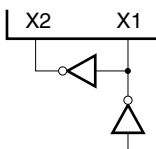
- When supply voltage drops

$V_{DD}$  must be lowered 10  $\mu$ s or more after  $V_{PP}$  falls below the lower-limit value (2.7 V) of the operating voltage range of  $V_{DD}$  (see b in the figure below).



2. The rms value should be calculated as follows: [rms value] = [Peak value]  $\times \sqrt{\text{Duty}}$

Main System Clock Oscillator Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency ( $f_x$ ) <sup>Note 1</sup>	$V_{DD}$ = Oscillation voltage range	1.0		5.0	MHz
		Oscillation stabilization time <sup>Note 2</sup>	After $V_{DD}$ reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency ( $f_x$ ) <sup>Note 1</sup>		1.0		5.0	MHz
		Oscillation stabilization time <sup>Note 2</sup>	$V_{DD} = 4.5$ to $5.5$ V $V_{DD} = 2.7$ to $5.5$ V			10 30	ms
External clock		X1 input frequency ( $f_x$ ) <sup>Note 1</sup>		1.0		5.0	MHz
		X1 input high-/low-level width ( $t_{xH}$ , $t_{xL}$ )		85		500	ns

**Notes** 1. Indicates only oscillator characteristics. See **AC Characteristics** for instruction execution time.

2. Time required to stabilize oscillation after reset or STOP mode release.

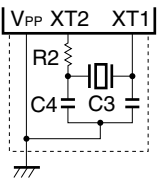
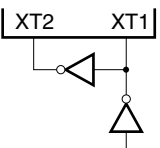
**Cautions** 1. When using the main system clock oscillator, wire as follows in the area enclosed by the broken lines in the above figures to avoid an adverse effect from wiring capacitance.

- 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  $V_{SS1}$ .
- Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

2. When the main system clock is stopped and the system is operating on the subsystem clock, wait until the oscillation stabilization time has been secured by the program before switching back to the main system clock.

**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.

**Subsystem Clock Oscillator Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V)**

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Crystal resonator		Oscillation frequency ( $f_{XT}$ ) <sup>Note 1</sup>		32	32.768	35	kHz
		Oscillation stabilization time <sup>Note 2</sup>	$V_{DD} = 4.5$ to $5.5$ V		1.2	2	s
			$V_{DD} = 4.5$ to $5.5$ V			10	
External clock		XT1 input frequency ( $f_{XT}$ ) <sup>Note 1</sup>		32		35	kHz
		XT1 input high-/low-level width ( $t_{XTH}$ , $t_{XTL}$ )		12		15	$\mu\text{s}$

**Notes** 1. Indicates only oscillator characteristics. See **AC Characteristics** for instruction execution time.

2. Time required to stabilize oscillation after  $V_{DD}$  reaches oscillation voltage range MIN.

**Cautions** 1. When using the subsystem clock 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  $V_{SS1}$ .
- Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

2. The subsystem clock oscillator is designed as a low-amplitude circuit for reducing current consumption, and is more prone to malfunction due to noise than the main system clock oscillator. Particular care is therefore required with the wiring method when the subsystem 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.

**Capacitance ( $T_A = 25^\circ\text{C}$ ,  $V_{DD} = V_{SS} = 0$  V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Input capacitance	$C_{IN}$	$f = 1$ MHz Unmeasured pins returned to 0 V.			15	pF	
I/O capacitance	$C_{IO}$	$f = 1$ MHz Unmeasured pins returned to 0 V.	P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131			15	pF
			P60 to P63			20	pF

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.



DC Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Input voltage, high	$V_{IH1}$	P10 to P17, P21, P23, P30 to P32, P35 to P37, P40 to P47, P50 to P57, P64-P67, P71, P120 to P127, P130, P131	$V_{DD} = 2.7$ to $5.5$ V	$0.7 V_{DD}$		$V_{DD}$	V
	$V_{IH2}$	P00 to P05, P20, P22, P24 to P27, P33, P34, P70, P72, RESET	$V_{DD} = 2.7$ to $5.5$ V	$0.8 V_{DD}$		$V_{DD}$	V
	$V_{IH3}$	P60 to P63 (N-ch open drain)	$V_{DD} = 2.7$ to $5.5$ V	$0.7 V_{DD}$		15	V
	$V_{IH4}$	X1, X2	$V_{DD} = 2.7$ to $5.5$ V	$V_{DD} - 0.5$		$V_{DD}$	V
	$V_{IH5}$	XT1/P07, XT2	$4.5 \text{ V} \leq V_{DD} \leq 5.5 \text{ V}$ $2.7 \text{ V} \leq V_{DD} < 4.5 \text{ V}$	$0.8 V_{DD}$ $0.9 V_{DD}$		$V_{DD}$ $V_{DD}$	V V
Input voltage, low	$V_{IL1}$	P10 to P17, P21, P23, P30 to P32, P35 to P37, P40 to P47, P50 to P57, P64 to P67, P71, P120 to P127, P130, P131	$V_{DD} = 2.7$ to $5.5$ V	0		$0.3 V_{DD}$	V
	$V_{IL2}$	P00 to P05, P20, P22, P24 to P27, P33, P34, P70, P72, RESET	$V_{DD} = 2.7$ to $5.5$ V	0		$0.2V_{DD}$	V
	$V_{IL3}$	P60 to P63	$4.5 \text{ V} \leq V_{DD} \leq 5.5 \text{ V}$	0		$0.3V_{DD}$	V
			$2.7 \text{ V} \leq V_{DD} < 4.5 \text{ V}$	0		$0.2V_{DD}$	V
	$V_{IL4}$	X1, X2	$V_{DD} = 2.7$ to $5.5$ V	0		0.4	V
	$V_{IL5}$	XT1/P07, XT2	$4.5 \text{ V} \leq V_{DD} \leq 5.5 \text{ V}$	0		$0.2V_{DD}$	V
$2.7 \text{ V} \leq V_{DD} < 4.5 \text{ V}$			0		$0.1V_{DD}$	V	
Output voltage, high	$V_{OH}$	$V_{DD} = 4.5$ to $5.5$ V, $I_{OH} = -1$ mA	$V_{DD} - 1.0$			V	
		$V_{DD} = 2.7$ to $5.5$ V, $I_{OH} = -100 \mu\text{A}$	$V_{DD} - 0.5$			V	
Output voltage, low	$V_{OL1}$	P50 to P57, P60 to P63	$V_{DD} = 4.5$ to $5.5$ V, $I_{OL} = 15$ mA		0.4	2.0	V
		P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P64 to P67, P70 to P72, P120-P127, P130, P131	$V_{DD} = 4.5$ to $5.5$ V, $I_{OL} = 1.6$ mA			0.4	V
	$V_{OL2}$	SB0, SB1, $\overline{\text{SCK0}}$	$V_{DD} = 4.5$ to $5.5$ V, open drain, pulled-up ( $R = 1 \text{ k}\Omega$ )			$0.2 V_{DD}$	V
	$V_{OL3}$	$I_{OL} = 400 \mu\text{A}$				0.5	V

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**DC Characteristics (T<sub>A</sub> = –40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	I <sub>LIH1</sub>	V <sub>IN</sub> = V <sub>DD</sub>	P00 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P70 to P72, P120 to P127, P130, P131, $\overline{\text{RESET}}$			3	μA
	I <sub>LIH2</sub>		X1, X2, XT1/P07, XT2			20	μA
	I <sub>LIH3</sub>	V <sub>IN</sub> = 15 V	P60 to P63			80	μA
Input leakage current, low	I <sub>LIL1</sub>	V <sub>IN</sub> = 0 V	P00 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131, $\overline{\text{RESET}}$			–3	μA
	I <sub>LIL2</sub>		X1, X2, XT1/P07, XT2			–20	μA
	I <sub>LIL3</sub>		P60-P63			–3 <sup>Note</sup>	μA
Software pull-up resistor	R	V <sub>IN</sub> = 0 V, P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131		15	30	90	kΩ

**Note** A low-level input leakage current of –200 μA (MAX.) flows only for 1.5 clocks (without wait) after a read instruction has been executed to port 6 (P6) or port mode register 6 (PM6). At times other than this 1.5-clock interval, a –3 μA (MAX.) current flows.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**DC Characteristics (T<sub>A</sub> = –40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Output current, high	I <sub>OH</sub>	Per pin			–1	mA
		Total for all pins			–15	mA
Output current, low	I <sub>OL</sub>	Per pin for P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131			10	mA
		Per pin for P50 to P57, P60 to P63			15	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P70 to P72, P130, P131			10	mA
		Total for P01 to P05, P30 to P37, P64 to P67, P120 to P127			10	mA
		Total for P50 to P57, P60 to P63			70	mA

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit			
Power supply current	I <sub>DD1</sub> Note 5	5.0 MHz crystal oscillation operating mode (f <sub>xx</sub> = 2.5 MHz)Note 3	V <sub>DD</sub> = 5.0 V ±10%Note 1		6.2	12.5	mA		
			V <sub>DD</sub> = 3.0 V ±10%Note 2		1.3	3.1	mA		
		5.0 MHz crystal oscillation operating mode (f <sub>xx</sub> = 5.0 MHz)Note 4	V <sub>DD</sub> = 5.0 V ±10%Note 1		13.1	25.7	mA		
			V <sub>DD</sub> = 3.0 V ±10%Note 2		2.1	4.9	mA		
	I <sub>DD2</sub>	5.0 MHz crystal oscillation HALT mode (f <sub>xx</sub> = 2.5 MHz)Note 3	V <sub>DD</sub> = 5.0 V ±10%	Peripheral functions operating			5.6	mA	
				Peripheral functions not operating		1.0	2.8	mA	
			V <sub>DD</sub> = 3.0 V ±10%	Peripheral functions operating			2.9	mA	
				Peripheral functions not operating		0.44	1.1	mA	
			5.0 MHz crystal oscillation HALT mode (f <sub>xx</sub> = 5.0 MHz)Note 4	V <sub>DD</sub> = 5.0 V ±10%	Peripheral functions operating			8.4	mA
					Peripheral functions not operating		1.3	3.1	mA
		V <sub>DD</sub> = 3.0 V ±10%		Peripheral functions operating			4.5	mA	
				Peripheral functions not operating		0.6	1.5	mA	
		I <sub>DD3</sub> Note 5	32.768 kHz crystal oscillation operating modeNote 6	V <sub>DD</sub> = 5.0 V ±10%		110	220	μA	
				V <sub>DD</sub> = 3.0 V ±10%		86	172	μA	
		I <sub>DD4</sub> Note 5	32.768 kHz crystal oscillation HALT modeNote 6	V <sub>DD</sub> = 5.0 V ±10%		22.5	50	μA	
				V <sub>DD</sub> = 3.0 V ±10%		3.2	13.2	μA	
I <sub>DD5</sub> Note 5	XT1 = V <sub>DD</sub> STOP mode When feedback resistor is used	V <sub>DD</sub> = 5.0 V ±10%		1.0	30	μA			
		V <sub>DD</sub> = 3.0 V ±10%		0.5	10	μA			
I <sub>DD6</sub> Note 5	XT1 = V <sub>DD</sub> STOP mode When feedback resistor is not used	V <sub>DD</sub> = 5.0 V ±10%		0.1	30	μA			
		V <sub>DD</sub> = 3.0 V ±10%		0.05	10	μA			

- Notes**
1. High-speed mode operation (when the processor clock control register (PCC) is cleared to 00H).
  2. Low-speed mode operation (when PCC is set to 04H).
  3. Operation with main system clock f<sub>xx</sub> = f<sub>x</sub>/2 (when the oscillation mode select register (OSMS) is cleared to 00H)
  4. Operation with main system clock f<sub>xx</sub> = f<sub>x</sub> (when OSMS is set to 01H)
  5. Refers to the current flowing to the V<sub>DD0</sub> and V<sub>DD1</sub> pins. The current flowing to the A/D converter, D/A converter, and on-chip pull-up resistor is not included.
  6. When the main system clock operation is stopped.

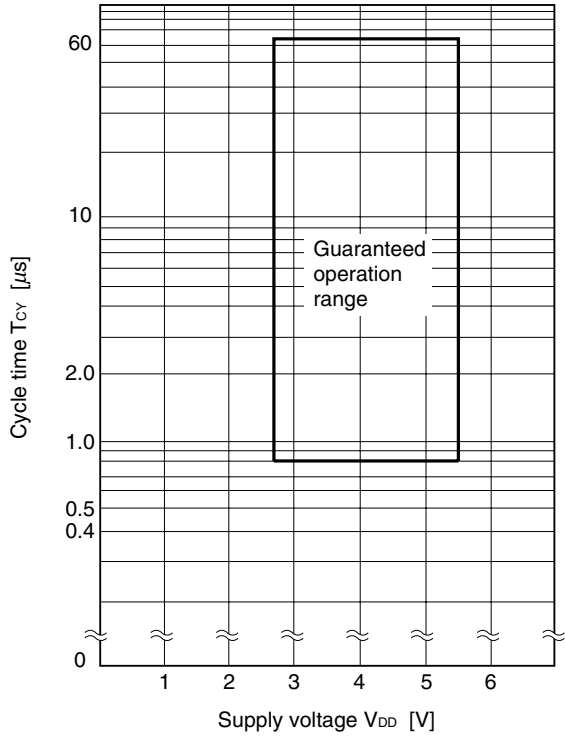
## AC Characteristics

### (1) Basic operation ( $T_A = -40$ to $+85^\circ\text{C}$ , $V_{DD} = 2.7$ to $5.5$ V)

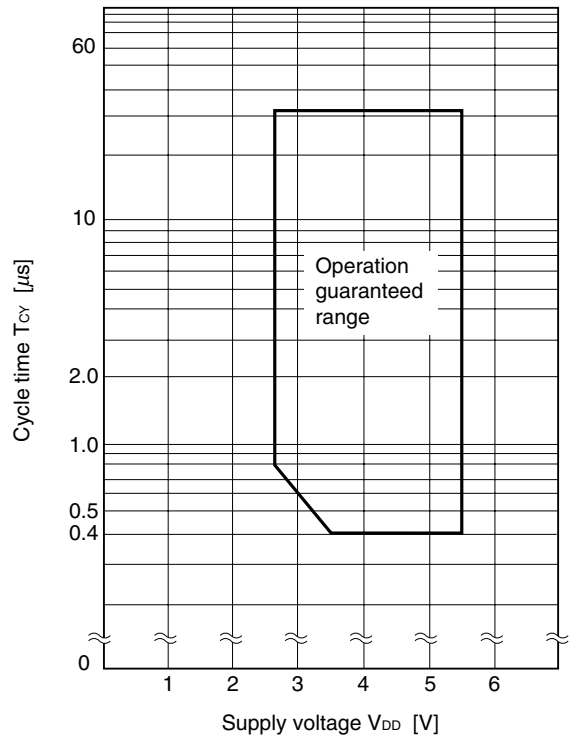
Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Cycle time (Min. instruction execution time)	$t_{CY}$	Operating with main system clock ( $f_{XX} = 2.5$ MHz) <b>Note 1</b>	$V_{DD} = 2.7$ to $5.5$ V	0.8		64	$\mu\text{s}$
		Operating with main system clock ( $f_{XX} = 5.0$ MHz) <b>Note 2</b>	$3.5$ V $\leq V_{DD} \leq 5.5$ V	0.4		32	$\mu\text{s}$
			$2.7$ V $\leq V_{DD} < 3.5$ V	0.8		32	$\mu\text{s}$
		Operating with subsystem clock		40 <b>Note 3</b>	122	125	$\mu\text{s}$
TI00 input high-/ low-level width	$t_{TIH00}$	$3.5$ V $\leq V_{DD} \leq 5.5$ V		$2/f_{sam} + 0.1$ <b>Note 4</b>			$\mu\text{s}$
	$t_{TIL00}$	$2.7$ V $\leq V_{DD} < 3.5$ V		$2/f_{sam} + 0.2$ <b>Note 4</b>			$\mu\text{s}$
TI01 input high-/ low-level width	$t_{TIH01}$	$V_{DD} = 2.7$ to $5.5$ V		10			$\mu\text{s}$
	$t_{TIL01}$						
TI1, TI2 input frequency	$f_{TI1}$	$V_{DD} = 4.5$ to $5.5$ V		0		4	MHz
		$V_{DD} = 2.7$ to $5.5$ V		0		275	kHz
TI1, TI2 input high-/low-level width	$t_{TIH1}$	$V_{DD} = 4.5$ to $5.5$ V		100			ns
	$t_{TIL1}$	$V_{DD} = 2.7$ to $5.5$ V		1.8			$\mu\text{s}$
Interrupt request input high-/ low-level width	$t_{INTH}$	INTP0	$3.5$ V $\leq V_{DD} \leq 5.5$ V	$2/f_{sam} + 0.1$ <b>Note 4</b>			$\mu\text{s}$
			$2.7$ V $\leq V_{DD} < 3.5$ V	$2/f_{sam} + 0.2$ <b>Note 4</b>			$\mu\text{s}$
	$t_{INTL}$	INTP1 to INTP5, P40 to P47	$V_{DD} = 2.7$ to $5.5$ V	10			$\mu\text{s}$
$\overline{\text{RESET}}$ low- level width	$t_{RSL}$	$V_{DD} = 2.7$ to $5.5$ V		10			$\mu\text{s}$

- Notes**
1. Operation with main system clock  $f_{XX} = f_X/2$  (when the oscillation mode select register (OSMS) is cleared to 00H)
  2. Operation with main system clock  $f_{XX} = f_X$  (when OSMS is set to 01H)
  3. Value when external clock is used. When a crystal resonator is used, it is  $114 \mu\text{s}$  (MIN.)
  4. Selection of  $f_{sam} = f_{XX}/2^N$ ,  $f_{XX}/32$ ,  $f_{XX}/64$ , and  $f_{XX}/128$  is possible with bits 0 and 1 (SCS0, SCS1) of the sampling clock select register (SCS) (when  $N = 0$  to  $4$ ).

$T_{CY}$  vs.  $V_{DD}$  (@ $f_{XX} = f_X/2$  main system clock operation)



$T_{CY}$  vs.  $V_{DD}$  (@ $f_{XX} = f_X$  main system clock operation)



**(2) Read/write operation**
**(a) When MCS = 1, PCC2 to PCC0 = 000B ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 3.5$  to  $5.5$  V)**

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	$t_{ASTH}$		$0.85t_{CY} - 50$		ns
Address setup time	$t_{ADS}$		$0.85t_{CY} - 50$		ns
Address hold time	$t_{ADH}$		50		ns
Data input time from address	$t_{ADD1}$			$(2.85 + 2n)t_{CY} - 80$	ns
	$t_{ADD2}$			$(4 + 2n)t_{CY} - 100$	ns
Data input time from $\overline{RD}\downarrow$	$t_{RDD1}$			$(2 + 2n)t_{CY} - 100$	ns
	$t_{RDD2}$			$(2.85 + 2n)t_{CY} - 100$	ns
Read data hold time	$t_{RDH}$		0		ns
$\overline{RD}$ low-level width	$t_{RD1}$		$(2 + 2n)t_{CY} - 60$		ns
	$t_{RD2}$		$(2.85 + 2n)t_{CY} - 60$		ns
$\overline{WAIT}\downarrow$ input time from $\overline{RD}\downarrow$	$t_{RDWT1}$			$0.85t_{CY} - 50$	ns
	$t_{RDWT2}$			$2t_{CY} - 60$	ns
$\overline{WAIT}\downarrow$ input time from $\overline{WR}\downarrow$	$t_{WRWT}$			$2t_{CY} - 60$	ns
$\overline{WAIT}$ low-level width	$t_{WTL}$		$(1.15 + 2n)t_{CY}$	$(2 + 2n)t_{CY}$	ns
Write data setup time	$t_{WDS}$		$(2.85 + 2n)t_{CY} - 100$		ns
Write data hold time	$t_{WDH}$		20		ns
$\overline{WR}$ low-level width	$t_{WRL}$		$(2.85 + 2n)t_{CY} - 60$		ns
$\overline{RD}\downarrow$ delay time from $ASTB\downarrow$	$t_{ASTRD}$		25		ns
$\overline{WR}\downarrow$ delay time from $ASTB\downarrow$	$t_{ASTWR}$		$0.85t_{CY} + 20$		ns
$ASTB\uparrow$ delay time from $\overline{RD}\uparrow$ at external fetch	$t_{RDAST}$		$0.85t_{CY} - 10$	$1.15t_{CY} + 20$	ns
Address hold time from $\overline{RD}\uparrow$ at external fetch	$t_{RDADH}$		$0.85t_{CY} - 50$	$1.15t_{CY} + 50$	ns
Write data output time from $\overline{RD}\uparrow$	$t_{RDWD}$		40		ns
Write data output time from $\overline{WR}\downarrow$	$t_{WRWD}$		0	50	ns
Address hold time from $\overline{WR}\uparrow$	$t_{WRADH}$		$0.85t_{CY}$	$1.15t_{CY} + 40$	ns
$\overline{RD}\uparrow$ delay time from $\overline{WAIT}\uparrow$	$t_{WTRD}$		$1.15t_{CY} + 40$	$3.15t_{CY} + 40$	ns
$\overline{WR}\uparrow$ delay time from $\overline{WAIT}\uparrow$	$t_{WTWR}$		$1.15t_{CY} + 30$	$3.15t_{CY} + 30$	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3.  $t_{CY} = T_{CY}/4$
  4. n indicates the number of waits.

(b) When MCS = 0 or PCC2 to PCC0 ≠ 000B (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t <sub>ASTH</sub>		t <sub>CY</sub> - 80		ns
Address setup time	t <sub>ADS</sub>		t <sub>CY</sub> - 80		ns
Address hold time	t <sub>ADH</sub>		0.4t <sub>CY</sub> - 10		ns
Data input time from address	t <sub>ADD1</sub>			(3 + 2n)t <sub>CY</sub> - 160	ns
	t <sub>ADD2</sub>			(4 + 2n)t <sub>CY</sub> - 200	ns
Data input time from $\overline{RD}\downarrow$	t <sub>RDD1</sub>			(1.4 + 2n)t <sub>CY</sub> - 70	ns
	t <sub>RDD2</sub>			(2.4 + 2n)t <sub>CY</sub> - 70	ns
Read data hold time	t <sub>RDH</sub>		0		ns
$\overline{RD}$ low-level width	t <sub>RDL1</sub>		(1.4 + 2n)t <sub>CY</sub> - 20		ns
	t <sub>RDL2</sub>		(2.4 + 2n)t <sub>CY</sub> - 20		ns
$\overline{WAIT}\downarrow$ input time from $\overline{RD}\downarrow$	t <sub>RDWT1</sub>			t <sub>CY</sub> - 100	ns
	t <sub>RDWT2</sub>			2t <sub>CY</sub> - 100	ns
$\overline{WAIT}\downarrow$ input time from $\overline{WR}\downarrow$	t <sub>WRWT</sub>			2t <sub>CY</sub> - 100	ns
$\overline{WAIT}$ low-level width	t <sub>WTL</sub>		(1 + 2n)t <sub>CY</sub>	(2 + 2n)t <sub>CY</sub>	ns
Write data setup time	t <sub>WDS</sub>		(2.4 + 2n)t <sub>CY</sub> - 60		ns
Write data hold time	t <sub>WDH</sub>		20		ns
$\overline{WR}$ low-level width	t <sub>WRL</sub>		(2.4 + 2n)t <sub>CY</sub> - 20		ns
$\overline{RD}\downarrow$ delay time from $\overline{ASTB}\downarrow$	t <sub>ASTRD</sub>		0.4t <sub>CY</sub> - 30		ns
$\overline{WR}\downarrow$ delay time from $\overline{ASTB}\downarrow$	t <sub>ASTWR</sub>		1.4t <sub>CY</sub> - 30		ns
ASTB $\uparrow$ delay time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDAST</sub>		t <sub>CY</sub> - 10	t <sub>CY</sub> + 20	ns
Address hold time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDADH</sub>		t <sub>CY</sub> - 50	t <sub>CY</sub> + 50	ns
Write data output time from $\overline{RD}\uparrow$	t <sub>RDWD</sub>		0.4t <sub>CY</sub> - 20		ns
Write data output time from $\overline{WR}\downarrow$	t <sub>WRWD</sub>		0	60	ns
Address hold time from $\overline{WR}\uparrow$	t <sub>WRADH</sub>		t <sub>CY</sub>	t <sub>CY</sub> + 60	ns
$\overline{RD}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTRD</sub>		0.6t <sub>CY</sub> + 180	2.6t <sub>CY</sub> + 180	ns
$\overline{WR}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTWR</sub>		0.6t <sub>CY</sub> + 120	2.6t <sub>CY</sub> + 120	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3. t<sub>CY</sub> = T<sub>CY</sub>/4
  4. n indicates the number of waits.

**(3) Serial interface ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.7$  to  $5.5$  V)**
**(a) Serial interface channel 0**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY1}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH1}}, t_{\text{KL1}}$	$V_{\text{DD}} = 4.5$ to $5.5 \text{ V}$	$t_{\text{KCY1}}/2 - 50$			ns
		$V_{\text{DD}} = 2.7$ to $5.5 \text{ V}$	$t_{\text{KCY1}}/2 - 100$			ns
SI0 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK1}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
SI0 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI1}}$		400			ns
SO0 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO1}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK0}}$  and SO0 output lines.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY2}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH2}}, t_{\text{KL2}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
SI0 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK2}}$	$2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
SI0 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI2}}$		400			ns
SO0 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO2}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{R2}}, t_{\text{F2}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO0 output line.



(iii) 2-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock output)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{CY3}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	1,600			ns
$\overline{\text{SCK0}}$ high-level width	$t_{\text{KH3}}$		$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	$t_{\text{CY3}}/2 - 160$			ns
$\overline{\text{SCK0}}$ low-level width	$t_{\text{KL3}}$		$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{CY3}}/2 - 50$			ns
			$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	$t_{\text{CY3}}/2 - 100$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK3}}$		$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	300			ns
			$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	350			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SI3}}$			600			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{SO3}}$		0		300	ns	

**Note** R and C are the load resistance and load capacitance of the  $\overline{\text{SCK0}}$ , SB0, and SB1 output lines.

(iv) 2-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... External clock input)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{CY4}}$	$2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		1,600			ns
$\overline{\text{SCK0}}$ high-level width	$t_{\text{KH4}}$	$2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		650			ns
$\overline{\text{SCK0}}$ low-level width	$t_{\text{KL4}}$	$2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$		800			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK4}}$	$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$		100			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SI4}}$			$t_{\text{CY4}}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{SO4}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	0		300	ns
			$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	0		500	ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{R4}}, t_{\text{F4}}$	When using external device expansion function				160	ns
		When not using external device expansion function				1,000	ns

**Note** R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

**(v) SBI mode ( $\overline{\text{SCK0}}$  ... Internal clock output) ( $\mu\text{PD78F0058}$  only)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	3,200			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH5}}, t_{\text{KL5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	$t_{\text{KCY5}}/2 - 50$			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	$t_{\text{KCY5}}/2 - 150$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	300			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI5}}$		$t_{\text{KCY5}}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO5}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	0	250	ns
			$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	0	1,000	ns
SB0, SB1 $\downarrow$ from $\overline{\text{SCK0}}\uparrow$	$t_{\text{KSB}}$		$t_{\text{KCY5}}$			ns
$\overline{\text{SCK0}}\downarrow$ from SB0, SB1 $\downarrow$	$t_{\text{SBK}}$		$t_{\text{KCY5}}$			ns
SB0, SB1 high-level width	$t_{\text{SBH}}$		$t_{\text{KCY5}}$			ns
SB0, SB1 low-level width	$t_{\text{SBL}}$		$t_{\text{KCY5}}$			ns

**Note** R and C are the load resistance and load capacitance of the  $\overline{\text{SCK0}}$ , SB0, and SB1 output lines.

**(vi) SBI mode ( $\overline{\text{SCK0}}$  ... External clock input) ( $\mu\text{PD78F0058}$  only)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	3,200			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH6}}, t_{\text{KL6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	300			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI6}}$		$t_{\text{KCY6}}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO6}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	0	300	ns
			$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	0	1,000	ns
SB0, SB1 $\downarrow$ from $\overline{\text{SCK0}}\uparrow$	$t_{\text{KSB}}$		$t_{\text{KCY6}}$			ns
$\overline{\text{SCK0}}\downarrow$ from SB0, SB1 $\downarrow$	$t_{\text{SBK}}$		$t_{\text{KCY6}}$			ns
SB0, SB1 high-level width	$t_{\text{SBH}}$		$t_{\text{KCY6}}$			ns
SB0, SB1 low-level width	$t_{\text{SBL}}$		$t_{\text{KCY6}}$			ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{r6}}, t_{\text{f6}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

(vii) I<sup>2</sup>C bus mode (SCL ... Internal clock output) ( $\mu$ PD78F0058Y only)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
SCL cycle time	t <sub>KCY7</sub>	R = 1 k $\Omega$ ,	2.7 V $\leq$ V <sub>DD</sub> < 5.5 V	10			$\mu$ s
SCL high-level width	t <sub>KH7</sub>	C = 100 pF <sup>Note</sup>	2.7 V $\leq$ V <sub>DD</sub> < 5.5 V	t <sub>KCY7</sub> – 160			$\mu$ s
SCL low-level width	t <sub>KL7</sub>		4.5 V $\leq$ V <sub>DD</sub> < 5.5 V	t <sub>KCY7</sub> – 50			ns
			2.7 V $\leq$ V <sub>DD</sub> < 4.5 V	t <sub>KCY7</sub> – 100			ns
SDA0, SDA1 setup time (to SCL $\uparrow$ )	t <sub>SIK7</sub>		2.7 V $\leq$ V <sub>DD</sub> < 5.5 V	200			ns
SDA0, SDA1 hold time (from SCL $\downarrow$ )	t <sub>KSI7</sub>			0			ns
SDA0, SDA1 output delay time from SCL $\downarrow$	t <sub>KSO7</sub>		4.5 V $\leq$ V <sub>DD</sub> < 5.5 V	0		300	ns
			2.7 V $\leq$ V <sub>DD</sub> < 4.5 V	0		500	ns
SDA0, SDA1 $\downarrow$ from SCL $\uparrow$ or SDA0, SDA1 $\uparrow$ from SCL $\uparrow$	t <sub>KSB</sub>			200			ns
SCL $\downarrow$ from SDA0, SDA1 $\downarrow$	t <sub>SBK</sub>			400			ns
SDA0, SDA1 high-level width	t <sub>SBH</sub>		500			ns	

**Note** R and C are the load resistance and load capacitance of the SCL, SDA0, and SDA1 output lines.

(viii) I<sup>2</sup>C bus mode (SCL ... External clock input) ( $\mu$ PD78F0058Y only)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
SCL cycle time	t <sub>KCY8</sub>			1			$\mu$ s
SCL high-level width	t <sub>KH8</sub>			400			ns
SDA0, SDA1 setup time (to SCL $\uparrow$ )	t <sub>SIK8</sub>			200			ns
SDA0, SDA1 hold time (from SCL $\downarrow$ )	t <sub>KSI8</sub>			0			ns
SDA0, SDA1 output delay time from SCL $\downarrow$	t <sub>KSO8</sub>	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	4.5 V $\leq$ V <sub>DD</sub> < 5.5 V	0		300	ns
			2.7 V $\leq$ V <sub>DD</sub> < 4.5 V	0		500	ns
SDA0, SDA1 $\downarrow$ from SCL $\uparrow$ or SDA0, SDA1 $\uparrow$ from SCL $\uparrow$	t <sub>KSB</sub>			200			ns
SCL $\downarrow$ from SDA0, SDA1 $\downarrow$	t <sub>SBK</sub>			400			ns
SDA0, SDA1 high-level width	t <sub>SBH</sub>			500			ns

**Note** R and C are the load resistance and load capacitance of the SDA0 and SDA1 output lines.

**(b) Serial interface channel 1**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK1}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY9}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH9}}, t_{\text{KL9}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY9}}/2 - 50$			ns
		$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY9}}/2 - 100$			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK9}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSI9}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO9}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK1}}$  and SO1 output lines.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK1}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY10}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH10}}, t_{\text{KL10}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK10}}$	$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	100			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KIS10}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO10}}$	$C = 100 \text{ pF}^{\text{Note}}$   $V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$			300	ns
$\overline{\text{SCK1}}$ rise/fall time	$t_{\text{R10}}, t_{\text{F10}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO1 output line.

(iii) 3-wire serial I/O mode with automatic transmit/receive function ( $\overline{\text{SCK1}}$  ... Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY11}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH11}}, t_{\text{KL11}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY11}}/2 - 50$			ns
		$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY11}}/2 - 100$			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK11}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSI11}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO11}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
$\text{STB}\uparrow$ from $\overline{\text{SCK1}}\uparrow$	$t_{\text{SBD}}$		$t_{\text{KCY11}}/2 - 100$		$t_{\text{KCY11}}/2 + 100$	ns
Strobe signal high-level width	$t_{\text{SBW}}$	$2.7 \text{ V} \leq V_{\text{DD}} < 5.5 \text{ V}$	$t_{\text{KCY11}} - 30$		$t_{\text{KCY11}} + 30$	ns
Busy signal setup time (to busy signal detection timing)	$t_{\text{BYS}}$		100			ns
Busy signal hold time (from busy signal detection timing)	$t_{\text{BYH}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
$\overline{\text{SCK1}}\downarrow$ from busy inactive	$t_{\text{SPS}}$				$2t_{\text{KCY11}}$	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK1}}$  and SO1 output lines.

 (iv) 3-wire serial I/O mode with automatic transmit/receive function ( $\overline{\text{SCK1}}$  ... External clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY12}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH12}}, t_{\text{KL12}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK12}}$	$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	100			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSI12}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO12}}$	$C = 100 \text{ pF}^{\text{Note}}$   $V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$			300	ns
$\overline{\text{SCK1}}$ rise/fall time	$t_{\text{R12}}, t_{\text{F12}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO1 output line.

**(c) Serial interface channel 2**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK2}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK2}}$ cycle time	$t_{\text{KCY13}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK2}}$ high-/low-level width	$t_{\text{KH13}},$ $t_{\text{KL13}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY13}}/2 - 50$			ns
		$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY13}}/2 - 100$			ns
SI2 setup time (to $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{SIK13}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
SI2 hold time (from $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{KSI13}}$		400			ns
SO2 output delay time from $\overline{\text{SCK2}}\downarrow$	$t_{\text{KSO13}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the SO2 output line.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK2}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK2}}$ cycle time	$t_{\text{KCY14}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
$\overline{\text{SCK2}}$ high-/low-level width	$t_{\text{KH14}},$ $t_{\text{KL14}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
SI2 setup time (to $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{SIK14}}$	$V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$	100			ns
SI2 hold time (from $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{KSI14}}$		400			ns
SO2 output delay time from $\overline{\text{SCK2}}\downarrow$	$t_{\text{KSO14}}$	$C = 100 \text{ pF}^{\text{Note}}$   $V_{\text{DD}} = 2.7 \text{ to } 5.5 \text{ V}$			300	ns
$\overline{\text{SCK2}}$ rise/fall time	$t_{\text{r14}},$ $t_{\text{f14}}$	Other than below			160	ns
		$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$ When not using external device expansion function			1	$\mu\text{s}$

**Note** C is the load capacitance of the SO2 output line.

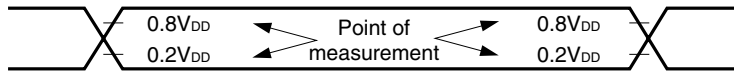
**(iii) UART mode (dedicated baud rate generator output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			78,125	bps
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$			39,063	bps

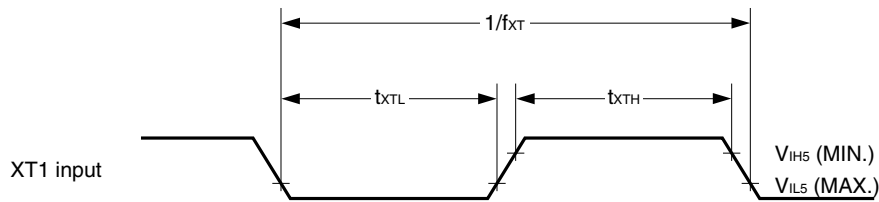
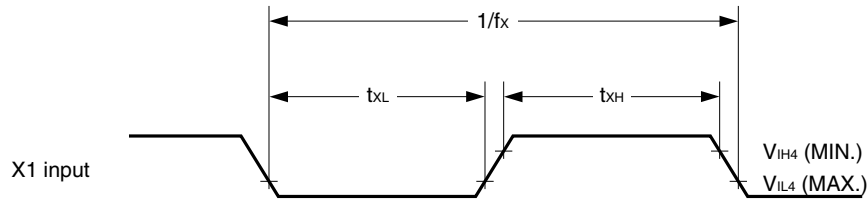
**(iv) UART mode (external clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK cycle time	$t_{KCY15}$	$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	800			ns
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$	1,600			ns
ASCK high-/low-level width	$t_{KH15}, t_{KL15}$	$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	400			ns
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$	800			ns
Transfer rate		$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			39,063	bps
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$			19,531	bps
ASCK rise/fall time	$t_{R15}, t_{F15}$	$V_{DD} = 4.5\text{ to }5.5\text{ V}$ , when not using external device expansion function.			1,000	ns
		Other than above			160	ns

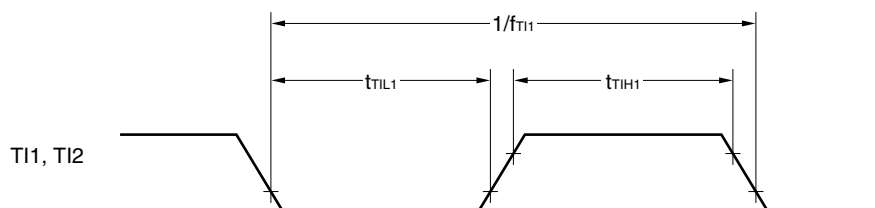
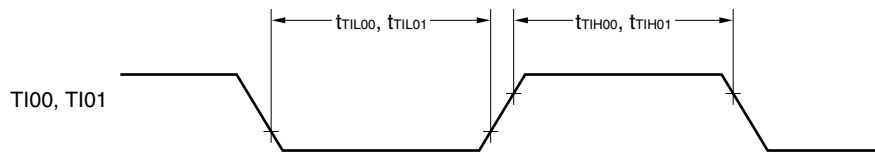
**AC Timing Measurement Points (Excluding X1, XT1 Inputs)**



**Clock Timing**

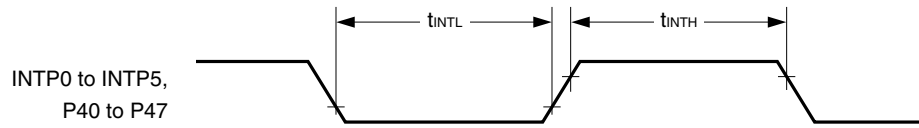


**TI Timing**

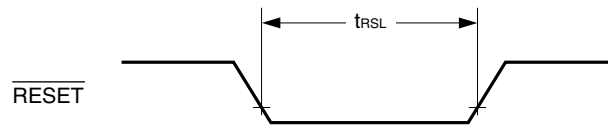




### Interrupt Request Input Timing

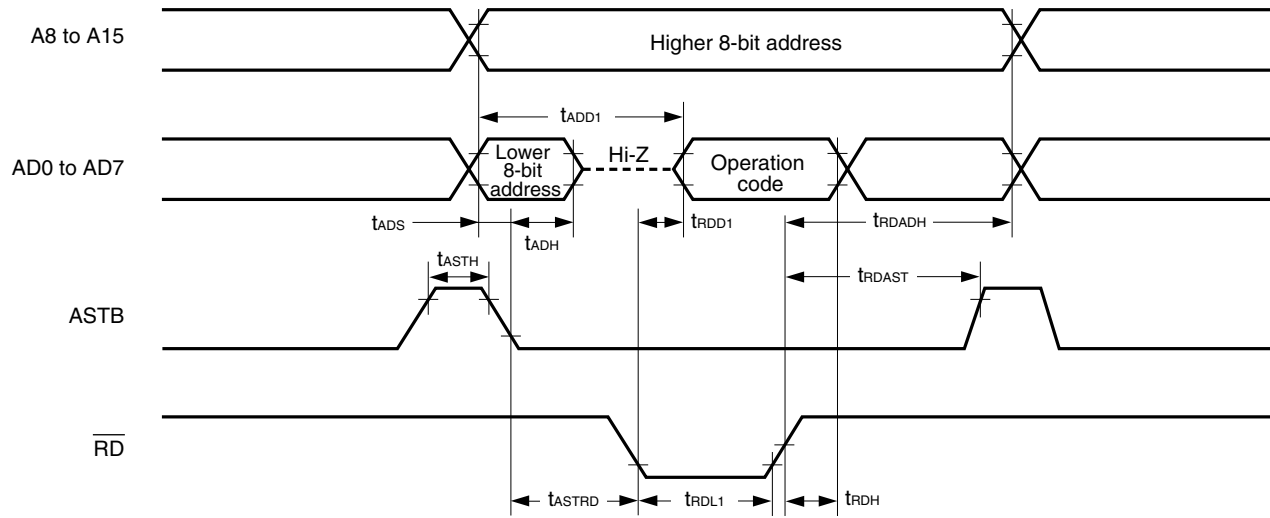


### $\overline{\text{RESET}}$ Input Timing

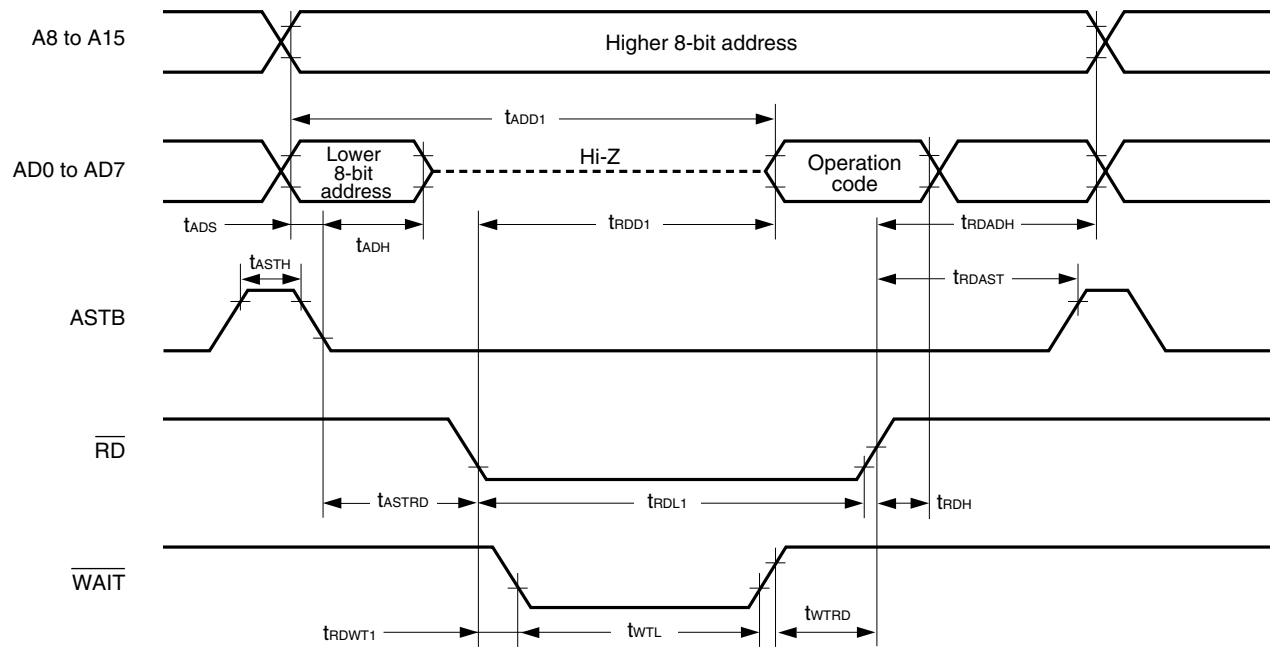


## Read/Write Operation

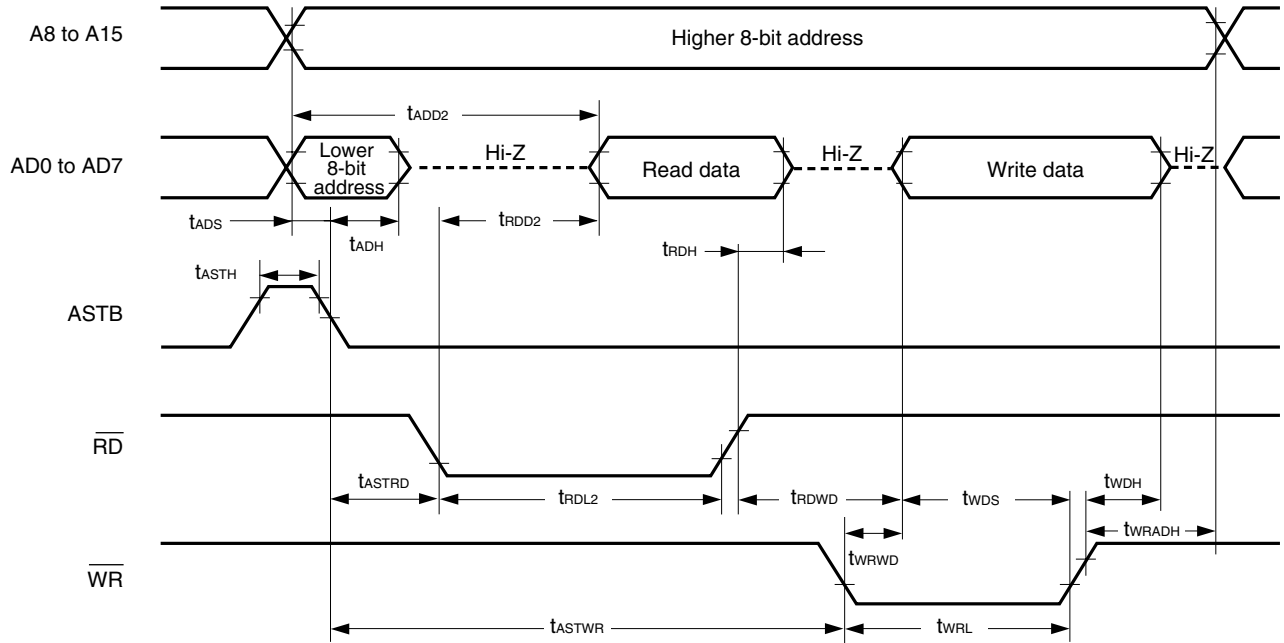
### External fetch (no wait):



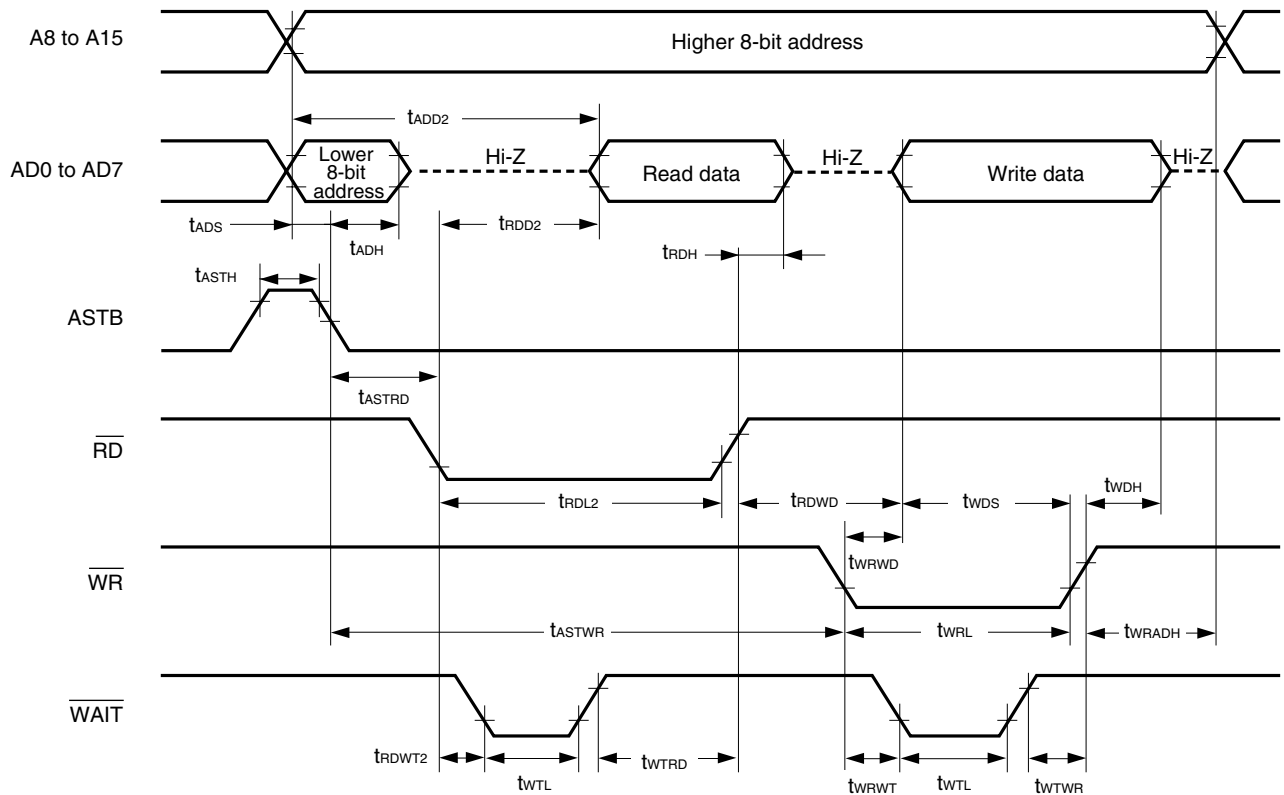
### External fetch (wait insertion):



External data access (no wait):

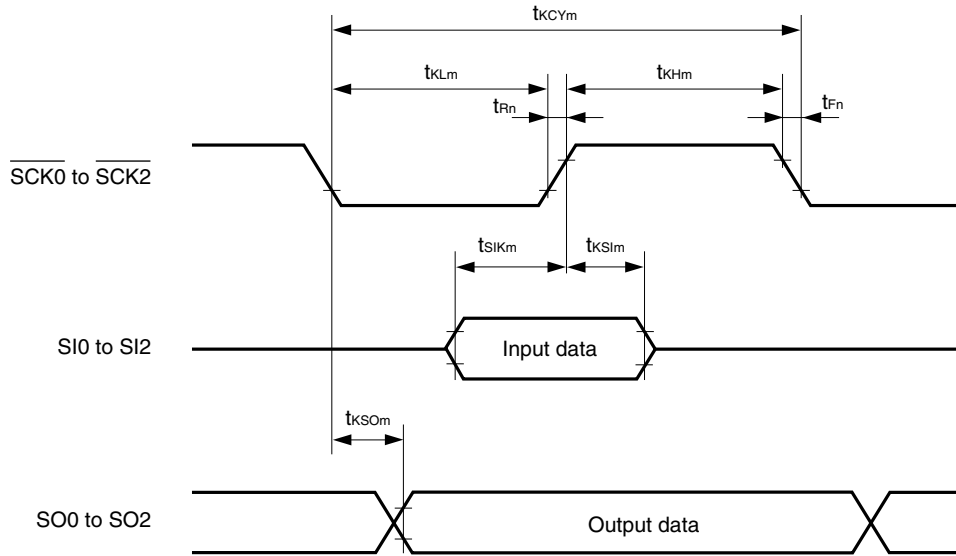


External data access (wait insertion):



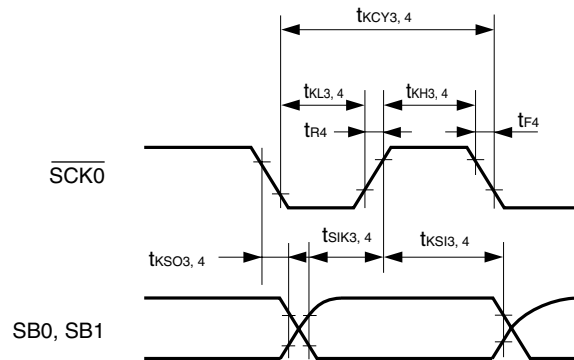
## Serial Transfer Timing

### 3-wire serial I/O mode:

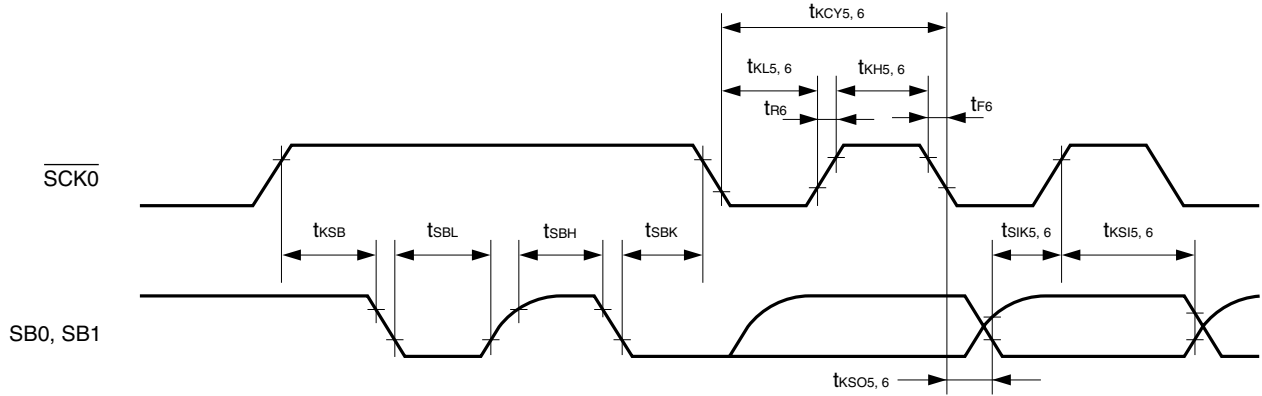


$m = 1, 2, 9, 10, 13, 14$   
 $n = 2, 10, 14$

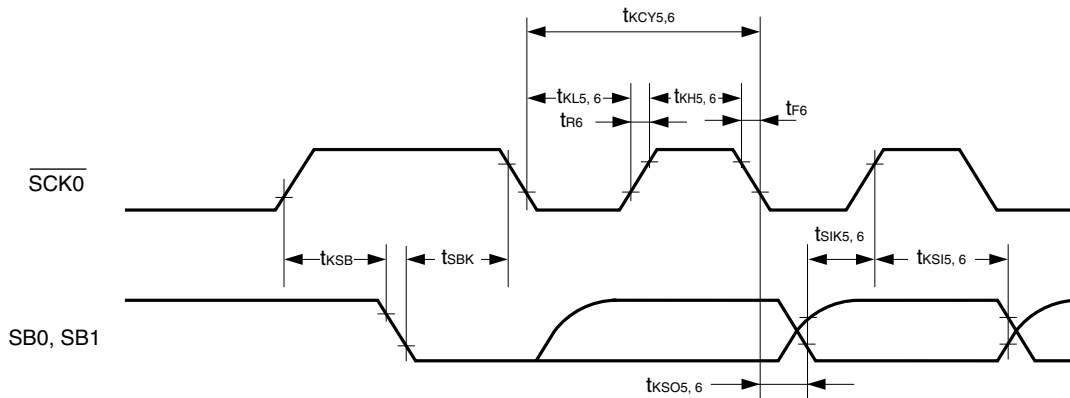
### 2-wire serial I/O mode:



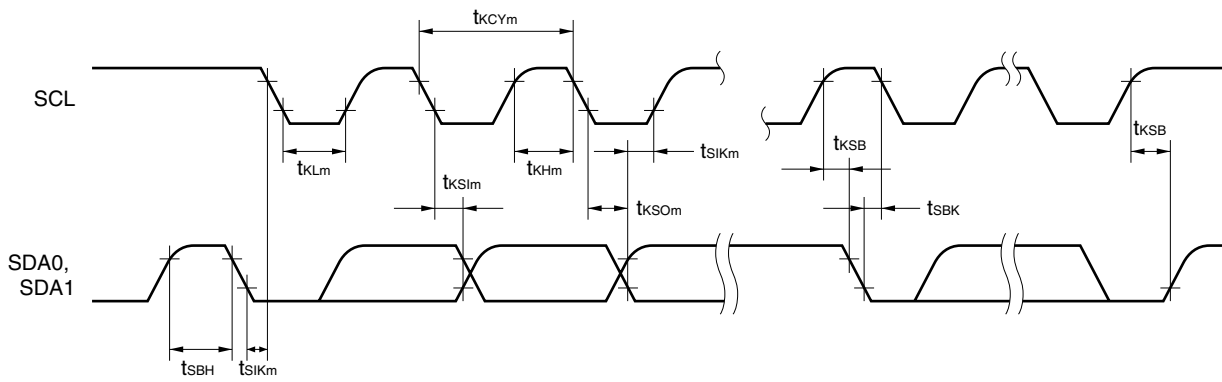
**SBI mode (bus release signal transfer):**



**SBI mode (command signal transfer):**

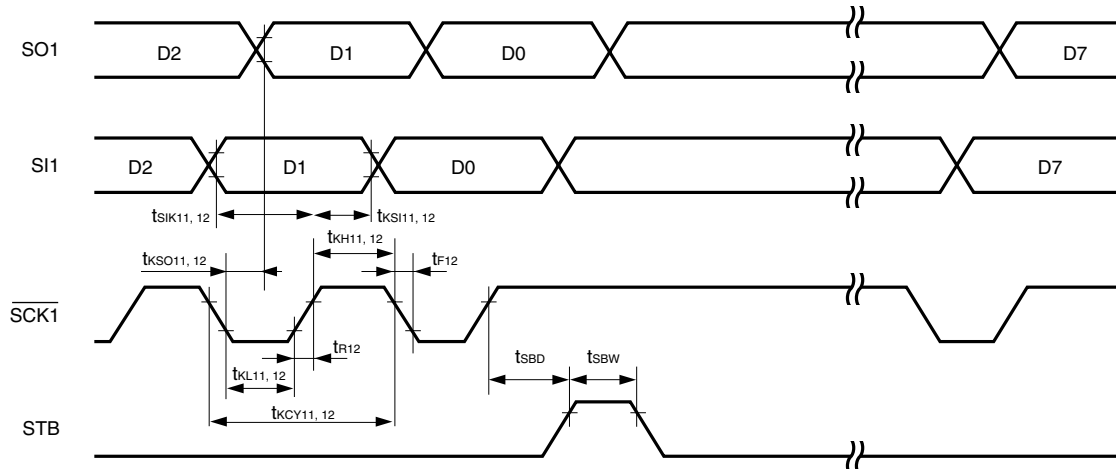


**I<sup>2</sup>C bus mode:**

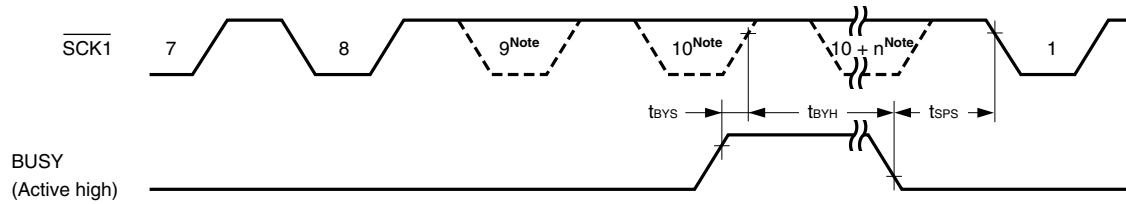


$m = 7, 8$

**3-wire serial I/O mode with automatic transmit/receive function:**

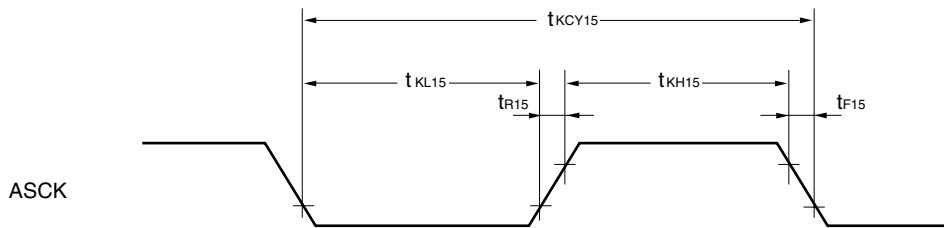


**3-wire serial I/O mode with automatic transmit/receive function (busy processing):**



**Note** The signal is not actually driven low here; it is shown as such to indicate the timing.

**UART mode (external clock input):**



**A/D Converter Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V, AV<sub>SS</sub> = V<sub>SS</sub> = 0 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error <sup>Note 1</sup>		2.7 V ≤ AV <sub>REF0</sub> < 4.5 V			±1.0	%FSR
		4.5 V ≤ AV <sub>REF0</sub> ≤ 5.5 V			±0.6	%FSR
Conversion time	T <sub>CONV</sub>	2.7 V ≤ AV <sub>REF0</sub> ≤ 5.5 V	16		100	μs
Analog input voltage	V <sub>IAN</sub>		AV <sub>SS</sub>		AV <sub>REF0</sub>	V
Reference voltage	AV <sub>REF0</sub>		2.7		V <sub>DD</sub>	V
AV <sub>REF0</sub> current	I <sub>REF0</sub>	When A/D converter is operating <sup>Note 2</sup>		500	1,500	μA
		When A/D converter is not operating <sup>Note 3</sup>		0	3	μA

**Notes** 1. Excludes quantization error (±1/2 LSB). This value is indicated as a ratio to the full-scale value (%FSR).

2. The current flowing to the AV<sub>REF0</sub> pin when bit 7 (CS) of the A/D converter mode register (ADM) is 1.

3. The current flowing to the AV<sub>REF0</sub> pin when bit 7 (CS) of the A/D converter mode register (ADM) is 0.

**D/A Converter Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V, AV<sub>SS</sub> = V<sub>SS</sub> = 0 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution					8	bit
Overall error		R = 2 MΩ <sup>Note 1</sup>			±1.2	%
		R = 4 MΩ <sup>Note 1</sup>			±0.8	%
		R = 10 MΩ <sup>Note 1</sup>			±0.6	%
Settling time		C = 30 pF <sup>Note 1</sup>			15	μs
Output resistance	R <sub>O</sub>	<b>Note 2</b>		8		kΩ
Analog reference voltage	AV <sub>REF1</sub>		1.8		V <sub>DD</sub>	V
AV <sub>REF1</sub> current	I <sub>REF1</sub>	<b>Note 2</b>			2.5	mA
Resistance between AV <sub>REF1</sub> and AV <sub>SS</sub>	R <sub>AIREF1</sub>	DACS0, DACS1 = 55H <sup>Note 2</sup>	4	8		kΩ

**Notes** 1. R and C are the D/A converter output pin load resistance and load capacitance, respectively.

2. Value for one D/A converter channel

**Remark** DACS0 and DACS1: D/A conversion value set registers 0, 1

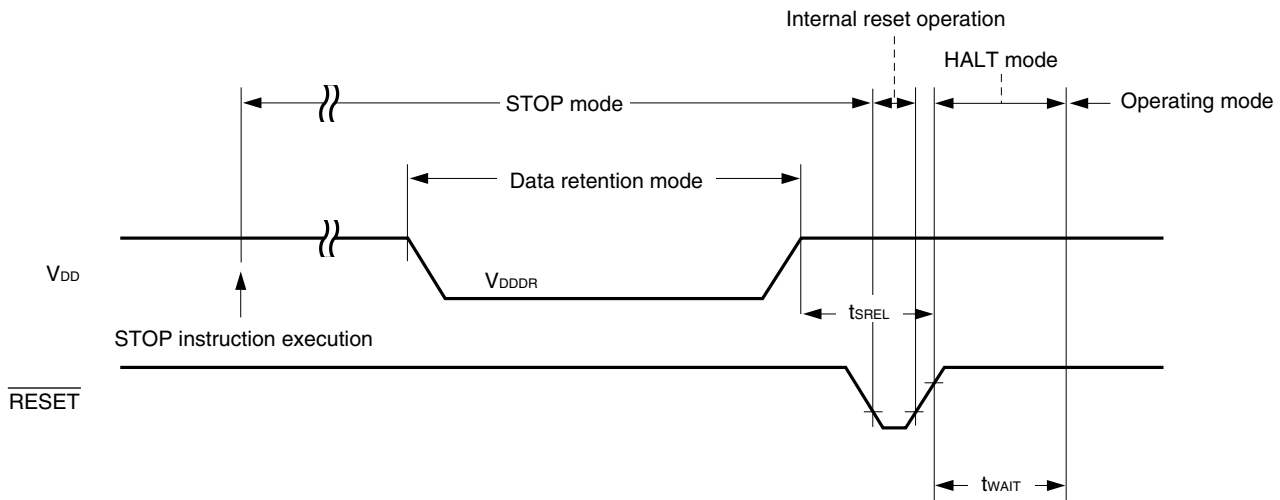
**Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (T<sub>A</sub> = -40 to +85°C)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention power supply voltage	V <sub>DDDR</sub>		1.8		5.5	V
Data retention power supply current	I <sub>DDDR</sub>	V <sub>DDDR</sub> = 1.8 V Subsystem clock stop and feed-back resistor disconnected		0.1	10	μA
Release signal set time	t <sub>SREL</sub>		0			μs
Oscillation stabilization wait time	t <sub>WAIT</sub>	Release by $\overline{\text{RESET}}$		2 <sup>17</sup> /f <sub>x</sub>		ms
		Release by interrupt request		<b>Note</b>		ms

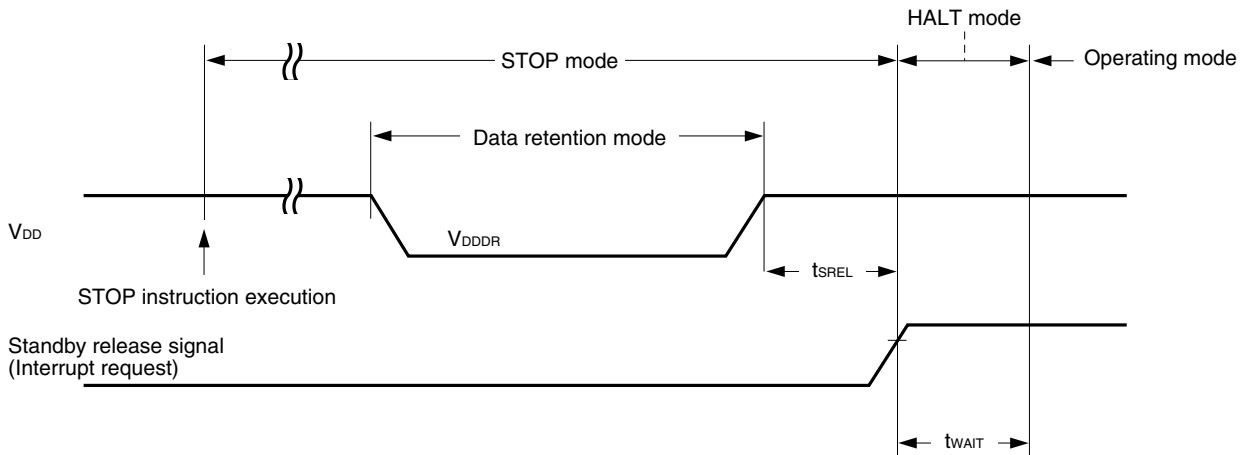
**Note** Selection of 2<sup>12</sup>/f<sub>xx</sub> and 2<sup>14</sup>/f<sub>xx</sub> to 2<sup>17</sup>/f<sub>xx</sub> is possible with bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register (OSTS).

**Remark** f<sub>xx</sub>: Main system clock frequency (f<sub>x</sub> or f<sub>x</sub>/2)  
f<sub>x</sub>: Main system clock oscillation frequency

**Data Retention Timing (STOP Mode Release by  $\overline{\text{RESET}}$ )**



**Data Retention Timing (Standby Release Signal: STOP Mode Release by Interrupt Request Signal)**





Flash Memory Programming Characteristics ( $V_{DD} = 2.7$  to  $5.5$  V,  $T_A = 10$  to  $40^\circ\text{C}$ )

## (1) Write/delete characteristics

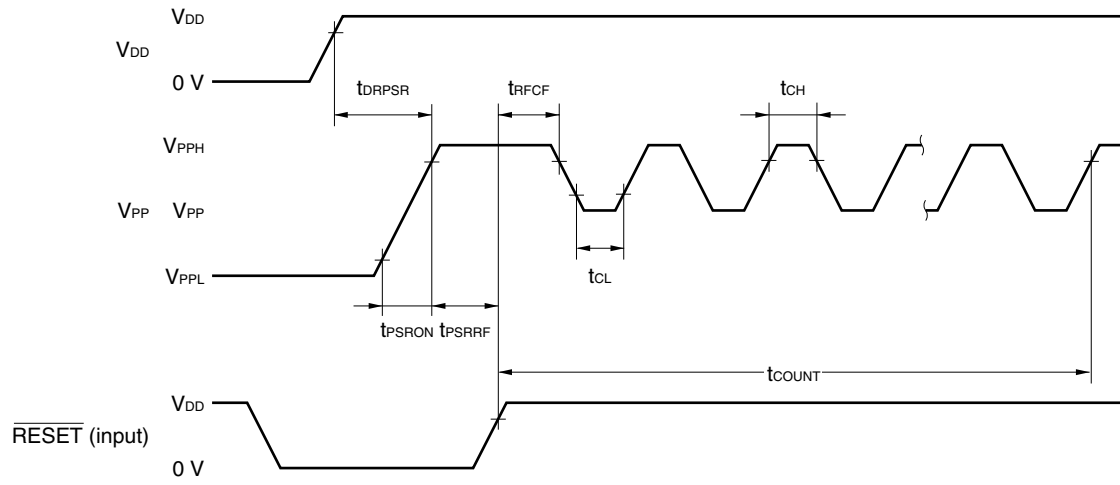
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Write current ( $V_{DD}$ pin) <sup>Note 1</sup>	$I_{DDW}$	When $V_{PP} = V_{PP1}$	5.0 MHz crystal oscillation operation mode ( $f_{XX} = 2.5$ MHz) <sup>Note 2</sup>			15.5	mA
			5.0 MHz crystal oscillation operation mode ( $f_{XX} = 5.0$ MHz) <sup>Note 3</sup>			28.7	mA
Write current ( $V_{PP}$ pin) <sup>Note 1</sup>	$I_{PPW}$	When $V_{PP} = V_{PP1}$	5.0 MHz crystal oscillation operation mode ( $f_{XX} = 2.5$ MHz) <sup>Note 2</sup>			19.5	mA
			5.0 MHz crystal oscillation operation mode ( $f_{XX} = 5.0$ MHz) <sup>Note 3</sup>			32.7	mA
Delete current ( $V_{DD}$ pin) <sup>Note 1</sup>	$I_{DDE}$	When $V_{PP} = V_{PP1}$	5.0 MHz crystal oscillation operation mode ( $f_{XX} = 2.5$ MHz) <sup>Note 2</sup>			15.5	mA
			5.0 MHz crystal oscillation operation mode ( $f_{XX} = 5.0$ MHz) <sup>Note 3</sup>			28.7	mA
Delete current ( $V_{PP}$ pin) <sup>Note 1</sup>	$I_{PPE}$	When $V_{PP} = V_{PP1}$			100	mA	
Unit delete time	$t_{ER}$		0.5	1	1	s	
Total delete time	$t_{ERA}$				20	s	
Number of overwrite	$C_{WRT}$	Delete and write are counted as one cycle			20	times	
$V_{PP}$ power supply voltage	$V_{PP0}$	In normal mode	0		$0.2 V_{DD}$	V	
	$V_{PP1}$	At flash memory programming	9.7	10.0	10.3	V	

- Notes**
1.  $AV_{REF}$  current and Port current (current flowing to internal pull-up resistor) are not included.
  2. When main system clock is operating at  $f_{XX} = f_{XX}/2$  (when oscillation mode select register (OSMS) is cleared to 00H).
  3. When main system clock is operating at  $f_{XX} = f_{XX}$  (when OSMS is set to 01H).

## 2) Serial write operation characteristics

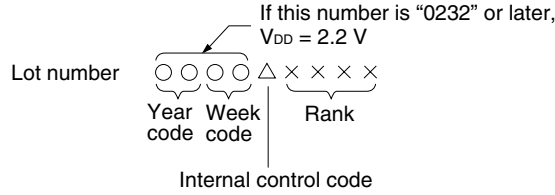
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$V_{PP}$ setup time	$t_{PSRON}$	$V_{PP}$ high voltage	1.0			$\mu\text{s}$
$V_{PP}\uparrow$ setup time from $V_{DD}\uparrow$	$t_{DRPSR}$	$V_{PP}$ high voltage	10			$\mu\text{s}$
$\overline{\text{RESET}}\uparrow$ setup time from $V_{PP}\uparrow$	$t_{PSRRF}$	$V_{PP}$ high voltage	1.0			$\mu\text{s}$
$V_{PP}$ count start time from $\overline{\text{RESET}}\uparrow$	$t_{RFCF}$		1.0			$\mu\text{s}$
Count execution time	$t_{COUNT}$				2.0	ms
$V_{PP}$ counter high-level width	$t_{CH}$		8.0			$\mu\text{s}$
$V_{PP}$ counter low-level width	$t_{CL}$		8.0			$\mu\text{s}$
$V_{PP}$ counter noise elimination width	$t_{NFW}$			40		ns

Flash Write Mode Setting Timing



★ CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION ( $V_{DD} = 2.2\text{ V}$ ))

**Caution** The product that can operate on  $V_{DD} = 2.2\text{ V}$  has “0232” or later as the first 4 digits of the lot number inscribed on the package.

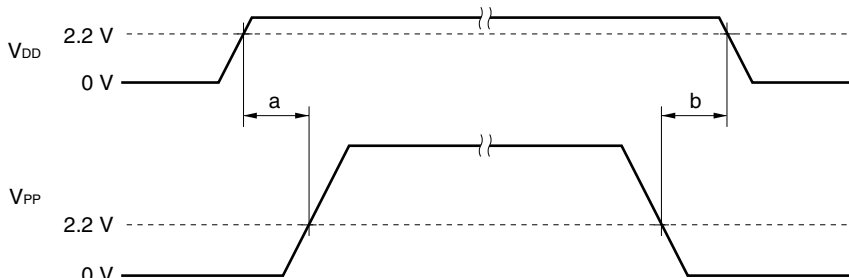


**Absolute Maximum Ratings ( $T_A = 25^\circ\text{C}$ )**

Parameter	Symbol	Conditions	Ratings	Unit	
Supply voltage	$V_{DD}$		-0.3 to +6.5	V	
	$V_{PP}$	<b>Note</b>	-0.3 to +10.5	V	
	$AV_{REF0}$		-0.3 to $V_{DD} + 0.3$	V	
	$AV_{REF1}$		-0.3 to $V_{DD} + 0.3$	V	
	$AV_{SS}$		-0.3 to +0.3	V	
Input voltage	$V_{I1}$	P00 to P05, P07, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131, X1, X2, XT2, RESET	-0.3 to $V_{DD} + 0.3$	V	
	$V_{I2}$	P60 to P63	N-ch open drain	-0.3 to +16	V
Output voltage	$V_O$		-0.3 to $V_{DD} + 0.3$	V	
Analog input voltage	$V_{AN}$	P10 to P17	Analog input pin	$AV_{SS} - 0.3$ to $AV_{REF0} + 0.3$	V

**Note** Make sure that the following conditions of the  $V_{PP}$  voltage application timing are satisfied when the flash memory is written.

- When supply voltage rises  
 $V_{PP}$  must exceed  $V_{DD}$   $10\ \mu\text{s}$  or more after  $V_{DD}$  has reached the lower-limit value (2.2 V) of the operating voltage range (see a in the figure below).
- When supply voltage drops  
 $V_{DD}$  must be lowered  $10\ \mu\text{s}$  or more after  $V_{PP}$  falls below the lower-limit value (2.2 V) of the operating voltage range of  $V_{DD}$  (see b in the figure below).



**Absolute Maximum Ratings (T<sub>A</sub> = 25°C)**

Parameter	Symbol	Conditions		Ratings	Unit
Output current, high	I <sub>OH</sub>	Per pin		-10	mA
		Total for P01 to P05, P30 to P37, P56, P57, P60 to P67, P120 to P127		-15	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P50 to P55, P70 to P72, P130, P131		-15	mA
Output current, low	I <sub>OL</sub> <sup>Note</sup>	Per pin for other than P50 to P57, P60 to P63	Peak value	20	mA
			rms value	10	mA
		Per pin for P50 to P57, P60 to P63	Peak value	30	mA
			rms value	15	mA
		Total for P50 to P55	Peak value	100	mA
			rms value	70	mA
		Total for P56, P57, P60 to P63	Peak value	100	mA
			rms value	70	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P70 to P72, P130, P131	Peak value	50	mA
			rms value	20	mA
		Total for P01 to P05, P30 to P37, P64 to P67, P120 to P127	Peak value	50	mA
			rms value	20	mA
Operating ambient temperature	T <sub>A</sub>	During normal operation		-40 to +85	°C
		During flash memory programming		10 to 40	°C
Storage temperature	T <sub>stg</sub>			-65 to +125	°C

**Note** The rms value should be calculated as follows: [rms value] = [Peak value] ×  $\sqrt{\text{Duty}}$

**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.

**Main System Clock Oscillator Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)**

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f <sub>x</sub> ) <sup>Note 1</sup>	V <sub>DD</sub> = Oscillation voltage range	1.0		5.0	MHz
		Oscillation stabilization time <sup>Note 2</sup>	After V <sub>DD</sub> reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency (f <sub>x</sub> ) <sup>Note 1</sup>		1.0		5.0	MHz
		Oscillation stabilization time <sup>Note 2</sup>	V <sub>DD</sub> = 4.5 to 5.5 V V <sub>DD</sub> = 2.2 to 5.5 V			10 30	ms
External clock		X1 input frequency (f <sub>x</sub> ) <sup>Note 1</sup>		1.0		5.0	MHz
		X1 input high-/low-level width (t <sub>xH</sub> , t <sub>xL</sub> )		85		500	ns

**Notes** 1. Indicates only oscillator characteristics. See **AC Characteristics** for instruction execution time.

2. Time required to stabilize oscillation after reset or STOP mode release.

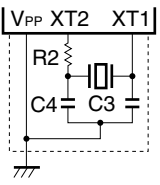
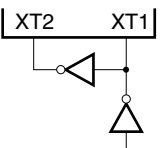
**Cautions** 1. When using the main system clock oscillator, wire as follows in the area enclosed by the broken lines in the above figures to avoid an adverse effect from wiring capacitance.

- 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<sub>SS1</sub>.
- Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

2. When the main system clock is stopped and the system is operating on the subsystem clock, wait until the oscillation stabilization time has been secured by the program before switching back to the main system clock.

**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.

**Subsystem Clock Oscillator Characteristics ( $T_A = -40$  to  $+85^\circ\text{C}$ ,  $V_{DD} = 2.2$  to  $5.5\text{ V}$ )**

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Crystal resonator		Oscillation frequency ( $f_{XT}$ ) <sup>Note 1</sup>		32	32.768	35	kHz
		Oscillation stabilization time <sup>Note 2</sup>	$V_{DD} = 4.5$ to $5.5\text{ V}$ $V_{DD} = 2.2$ to $5.5\text{ V}$		1.2	2	10
External clock		XT1 input frequency ( $f_{XT}$ ) <sup>Note 1</sup>		32		35	kHz
		XT1 input high-/low-level width ( $t_{XTH}$ , $t_{XTL}$ )		12		15	$\mu\text{s}$

**Notes 1.** Indicates only oscillator characteristics. See **AC Characteristics** for instruction execution time.

**2.** Time required to stabilize oscillation after  $V_{DD}$  reaches oscillation voltage range MIN.

**Cautions 1.** When using the subsystem clock 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  $V_{SS1}$ .
- Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

**2.** The subsystem clock oscillator is designed as a low-amplitude circuit for reducing current consumption, and is more prone to malfunction due to noise than the main system clock oscillator. Particular care is therefore required with the wiring method when the subsystem 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.

**Capacitance ( $T_A = 25^\circ\text{C}$ ,  $V_{DD} = V_{SS} = 0\text{ V}$ )**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Input capacitance	$C_{IN}$	$f = 1\text{ MHz}$ Unmeasured pins returned to $0\text{ V}$ .			15	pF	
I/O capacitance	$C_{IO}$	$f = 1\text{ MHz}$ Unmeasured pins returned to $0\text{ V}$ .	P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131			15	pF
			P60 to P63			20	pF

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Input voltage, high	V <sub>IH1</sub>	P10 to P17, P21, P23, P30 to P32, P35 to P37, P40 to P47, P50 to P57, P64 to P67, P71, P120 to P127, P130, P131	V <sub>DD</sub> = 2.7 to 5.5 V	0.7 V <sub>DD</sub>		V <sub>DD</sub>	V
			V <sub>DD</sub> = 2.2 to 5.5 V	0.8 V <sub>DD</sub>		V <sub>DD</sub>	V
	V <sub>IH2</sub>	P00 to P05, P20, P22, P24 to P27, P33, P34, P70, P72, RESET	V <sub>DD</sub> = 2.7 to 5.5 V	0.8 V <sub>DD</sub>		V <sub>DD</sub>	V
			V <sub>DD</sub> = 2.2 to 5.5 V	0.85 V <sub>DD</sub>		V <sub>DD</sub>	V
	V <sub>IH3</sub>	P60 to P63 (N-ch open drain)	V <sub>DD</sub> = 2.7 to 5.5 V	0.7 V <sub>DD</sub>		15	V
			V <sub>DD</sub> = 2.2 to 5.5 V	0.8 V <sub>DD</sub>		15	V
	V <sub>IH4</sub>	X1, X2	V <sub>DD</sub> = 2.7 to 5.5 V	V <sub>DD</sub> - 0.5		V <sub>DD</sub>	V
			V <sub>DD</sub> = 2.2 to 5.5 V	V <sub>DD</sub> - 0.2		V <sub>DD</sub>	V
	V <sub>IH5</sub>	XT1/P07, XT2	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.8 V <sub>DD</sub>		V <sub>DD</sub>	V
			2.7 V ≤ V <sub>DD</sub> < 4.5 V	0.9 V <sub>DD</sub>		V <sub>DD</sub>	V
2.2 V ≤ V <sub>DD</sub> < 2.7 V			0.9 V <sub>DD</sub>		V <sub>DD</sub>	V	
Input voltage, low	V <sub>IL1</sub>	P10 to P17, P21, P23, P30 to P32, P35 to P37, P40 to P47, P50 to P57, P64 to P67, P71, P120 to P127, P130, P131	V <sub>DD</sub> = 2.7 to 5.5 V	0		0.3 V <sub>DD</sub>	V
			V <sub>DD</sub> = 2.2 to 5.5 V	0		0.2 V <sub>DD</sub>	V
	V <sub>IL2</sub>	P00 to P05, P20, P22, P24 to P27, P33, P34, P70, P72, RESET	V <sub>DD</sub> = 2.7 to 5.5 V	0		0.2 V <sub>DD</sub>	V
			V <sub>DD</sub> = 2.2 to 5.5 V	0		0.15 V <sub>DD</sub>	V
	V <sub>IL3</sub>	P60 to P63	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0		0.3 V <sub>DD</sub>	V
			2.7 V ≤ V <sub>DD</sub> < 4.5 V	0		0.2 V <sub>DD</sub>	V
			2.2 V ≤ V <sub>DD</sub> < 2.7 V	0		0.1 V <sub>DD</sub>	V
	V <sub>IL4</sub>	X1, X2	V <sub>DD</sub> = 2.7 to 5.5 V	0		0.4	V
			V <sub>DD</sub> = 2.2 to 5.5 V	0		0.2	V
	V <sub>IL5</sub>	XT1/P07, XT2	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0		0.2 V <sub>DD</sub>	V
2.7 V ≤ V <sub>DD</sub> < 4.5 V			0		0.1 V <sub>DD</sub>	V	
2.2 V ≤ V <sub>DD</sub> < 2.7 V			0		0.1 V <sub>DD</sub>	V	
Output voltage, high	V <sub>OH</sub>	V <sub>DD</sub> = 4.5 to 5.5 V, I <sub>OH</sub> = -1 mA		V <sub>DD</sub> - 1.0		V	
		V <sub>DD</sub> = 2.2 to 5.5 V, I <sub>OH</sub> = -100 μA		V <sub>DD</sub> - 0.5		V	
Output voltage, low	V <sub>OL1</sub>	P50 to P57, P60 to P63	V <sub>DD</sub> = 4.5 to 5.5 V, I <sub>OL</sub> = 15 mA		0.4	2.0	V
		P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P64 to P67, P70 to P72, P120-P127, P130, P131	V <sub>DD</sub> = 4.5 to 5.5 V, I <sub>OL</sub> = 1.6 mA			0.4	V
	V <sub>OL2</sub>	SB0, SB1, SCK <sub>0</sub>	V <sub>DD</sub> = 4.5 to 5.5 V, open drain, pulled-up (R = 1 kΩ)			0.2V <sub>DD</sub>	V
	V <sub>OL3</sub>	I <sub>OL</sub> = 400 μA				0.5	V

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**DC Characteristics (T<sub>A</sub> = –40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	I <sub>LIH1</sub>	V <sub>IN</sub> = V <sub>DD</sub>	P00 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P60 to P67, P70 to P72, P120 to P127, P130, P131, $\overline{\text{RESET}}$			3	μA
	I <sub>LIH2</sub>		X1, X2, XT1/P07, XT2			20	μA
	I <sub>LIH3</sub>	V <sub>IN</sub> = 15 V	P60 to P63			80	μA
Input leakage current, low	I <sub>LIL1</sub>	V <sub>IN</sub> = 0 V	P00 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131, $\overline{\text{RESET}}$			–3	μA
	I <sub>LIL2</sub>		X1, X2, XT1/P07, XT2			–20	μA
	I <sub>LIL3</sub>		P60-P63			–3 <sup>Note</sup>	μA
Software pull-up resistor	R	V <sub>IN</sub> = 0 V, P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131		15	30	90	kΩ

**Note** A low-level input leakage current of –200 μA (MAX.) flows only for 1.5 clocks (without wait) after a read instruction has been executed to port 6 (P6) or port mode register 6 (PM6). At times other than this 1.5-clock interval, a –3 μA (MAX.) current flows.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

**DC Characteristics (T<sub>A</sub> = –40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Output current, high	I <sub>OH</sub>	Per pin			–1	mA
		Total for all pins			–15	mA
Output current, low	I <sub>OL</sub>	Per pin for P01 to P05, P10 to P17, P20 to P27, P30 to P37, P40 to P47, P50 to P57, P64 to P67, P70 to P72, P120 to P127, P130, P131			10	mA
		Per pin for P50 to P57, P60 to P63			15	mA
		Total for P10 to P17, P20 to P27, P40 to P47, P70 to P72, P130, P131			10	mA
		Total for P01 to P05, P30 to P37, P64 to P67, P120 to P127			10	mA
		Total for P50 to P57, P60 to P63			70	mA

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.



**DC Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Power supply current	I <sub>DD1</sub> Note 5	5.0 MHz crystal oscillation operating mode (f <sub>xx</sub> = 2.5 MHz) <sup>Note 3</sup>	V <sub>DD</sub> = 5.0 V ±10% <sup>Note 1</sup>	6.2	12.5	mA
			V <sub>DD</sub> = 3.0 V ±10% <sup>Note 2</sup>	1.3	3.1	mA
			V <sub>DD</sub> = 2.2 V <sup>Note 2</sup>	0.68	1.6	mA
		5.0 MHz crystal oscillation operating mode (f <sub>xx</sub> = 5.0 MHz) <sup>Note 4</sup>	V <sub>DD</sub> = 5.0 V ±10% <sup>Note 1</sup>	13.1	25.7	mA
			V <sub>DD</sub> = 3.0 V ±10% <sup>Note 2</sup>	2.1	4.9	mA
	I <sub>DD2</sub>	5.0 MHz crystal oscillation HALT mode (f <sub>xx</sub> = 2.5 MHz) <sup>Note 3</sup>	V <sub>DD</sub> = 5.0 V ±10%			
			Peripheral functions operating		5.6	mA
			Peripheral functions not operating	1.0	2.8	mA
			V <sub>DD</sub> = 3.0 V ±10%			
			Peripheral functions operating		2.9	mA
			Peripheral functions not operating	0.44	1.1	mA
		5.0 MHz crystal oscillation HALT mode (f <sub>xx</sub> = 5.0 MHz) <sup>Note 4</sup>	V <sub>DD</sub> = 2.2 V			
			Peripheral functions operating		1.5	mA
			Peripheral functions not operating	0.25	0.6	mA
			V <sub>DD</sub> = 5.0 V ±10%			
			Peripheral functions operating		8.4	mA
			Peripheral functions not operating	1.3	3.1	mA
	I <sub>DD3</sub> <sup>Note 5</sup>	32.768 kHz crystal oscillation operating mode <sup>Note 6</sup>	V <sub>DD</sub> = 5.0 V ±10%	110	220	μA
			V <sub>DD</sub> = 3.0 V ±10%	86	172	μA
V <sub>DD</sub> = 2.2 V			70	140	μA	
I <sub>DD4</sub> <sup>Note 5</sup>		32.768 kHz crystal oscillation HALT mode <sup>Note 6</sup>	V <sub>DD</sub> = 5.0 V ±10%	22.5	56	μA
			V <sub>DD</sub> = 3.0 V ±10%	3.2	13.2	μA
			V <sub>DD</sub> = 2.2 V	1.5	11.5	μA
I <sub>DD5</sub> <sup>Note 5</sup>	XT1 = V <sub>DD</sub> STOP mode When feedback resistor is used	V <sub>DD</sub> = 5.0 V ±10%	1.0	30	μA	
		V <sub>DD</sub> = 3.0 V ±10%	0.5	10	μA	
		V <sub>DD</sub> = 2.2 V	0.3	10	μA	
I <sub>DD6</sub> <sup>Note 5</sup>	XT1 = V <sub>DD</sub> STOP mode When feedback resistor is not used	V <sub>DD</sub> = 5.0 V ±10%	0.1	30	μA	
		V <sub>DD</sub> = 3.0 V ±10%	0.05	10	μA	
		V <sub>DD</sub> = 2.2 V	0.05	10	μA	

- Notes**
1. High-speed mode operation (when the processor clock control register (PCC) is cleared to 00H).
  2. Low-speed mode operation (when PCC is set to 04H).
  3. Operation with main system clock f<sub>xx</sub> = f<sub>x</sub>/2 (when the oscillation mode select register (OSMS) is cleared to 00H)
  4. Operation with main system clock f<sub>xx</sub> = f<sub>x</sub> (when OSMS is set to 01H)
  5. Refers to the current flowing to the V<sub>DD0</sub> and V<sub>DD1</sub> pins. The current flowing to the A/D converter, D/A converter, and on-chip pull-up resistor is not included.
  6. When the main system clock operation is stopped.

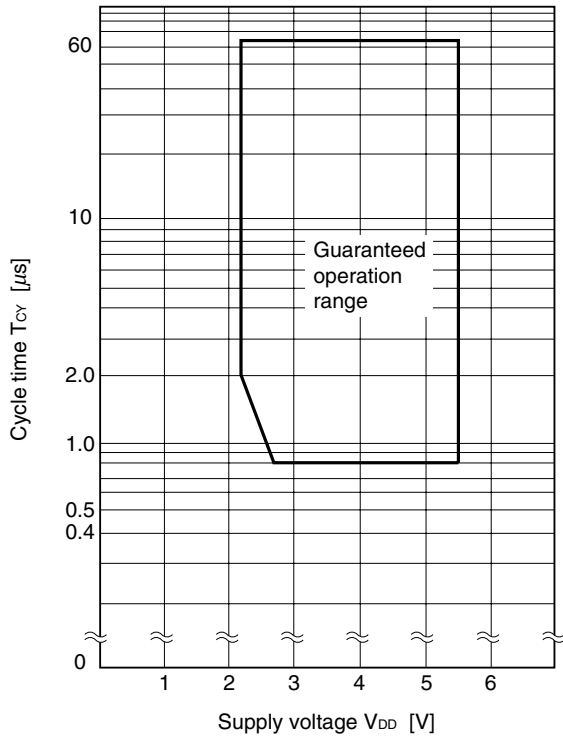
## AC Characteristics

### (1) Basic operation (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)

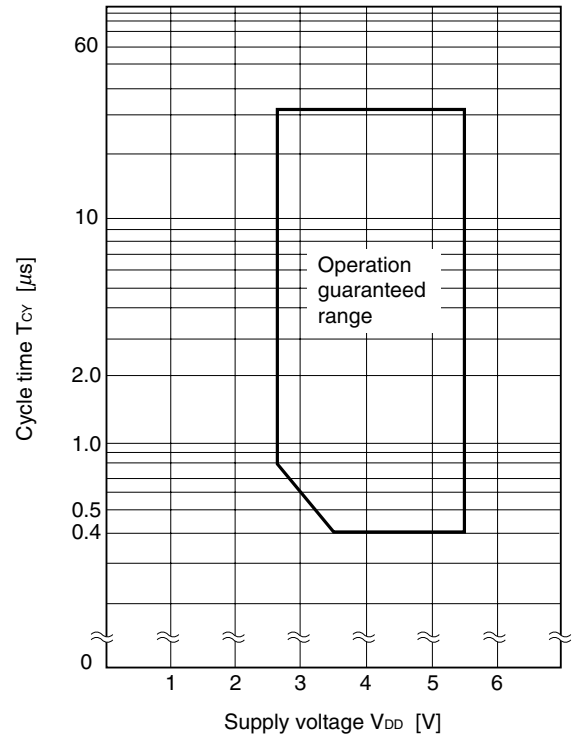
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Cycle time (Min. instruction execution time)	T <sub>CY</sub>	Operating with main system clock (f <sub>XX</sub> = 2.5 MHz) <sup>Note 1</sup>	V <sub>DD</sub> = 2.7 to 5.5 V	0.8		64	μs
			V <sub>DD</sub> = 2.2 to 5.5 V	2.0		64	μs
		Operating with main system clock (f <sub>XX</sub> = 5.0 MHz) <sup>Note 2</sup>	3.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.4		32	μs
			2.7 V ≤ V <sub>DD</sub> < 3.5 V	0.8		32	μs
		Operating with subsystem clock	40 <sup>Note 3</sup>	122	125	μs	
TI00 input high-/ low-level width	t <sub>TIH00</sub>	3.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	2/f <sub>sam</sub> + 0.1 <sup>Note 4</sup>			μs	
	t <sub>TIL00</sub>	2.7 V ≤ V <sub>DD</sub> < 3.5 V	2/f <sub>sam</sub> + 0.2 <sup>Note 4</sup>			μs	
		2.2 V ≤ V <sub>DD</sub> < 2.7 V	2/f <sub>sam</sub> + 0.5 <sup>Note 4</sup>			μs	
TI01 input high-/ low-level width	t <sub>TIH01</sub>	V <sub>DD</sub> = 2.7 to 5.5 V	10			μs	
	t <sub>TIL01</sub>	V <sub>DD</sub> = 2.2 to 5.5 V	20			μs	
TI1, TI2 input frequency	f <sub>TI1</sub>	V <sub>DD</sub> = 4.5 to 5.5 V	0		4	MHz	
		V <sub>DD</sub> = 2.2 to 5.5 V	0		275	kHz	
TI1, TI2 input high-/low-level width	t <sub>TIH1</sub>	V <sub>DD</sub> = 4.5 to 5.5 V	100			ns	
	t <sub>TIL1</sub>	V <sub>DD</sub> = 2.2 to 5.5 V	1.8			μs	
Interrupt request input high-/ low-level width	t <sub>INTH</sub>	INTP0	3.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	2/f <sub>sam</sub> + 0.1 <sup>Note 4</sup>		μs	
			2.7 V ≤ V <sub>DD</sub> < 3.5 V	2/f <sub>sam</sub> + 0.2 <sup>Note 4</sup>		μs	
			2.2 V ≤ V <sub>DD</sub> < 2.7 V	2/f <sub>sam</sub> + 0.5 <sup>Note 4</sup>		μs	
	t <sub>INTL</sub>	INTP1 to INTP5, P40 to P47	V <sub>DD</sub> = 2.7 to 5.5 V	10			μs
			V <sub>DD</sub> = 2.2 to 5.5 V	20			μs
RESET low- level width	t <sub>RSL</sub>	V <sub>DD</sub> = 2.7 to 5.5 V	10			μs	
		V <sub>DD</sub> = 2.2 to 5.5 V	20			μs	

- Notes**
1. Operation with main system clock f<sub>XX</sub> = f<sub>X</sub>/2 (when the oscillation mode select register (OSMS) is cleared to 00H)
  2. Operation with main system clock f<sub>XX</sub> = f<sub>X</sub> (when OSMS is set to 01H)
  3. Value when external clock is used. When a crystal resonator is used, it is 114 μs (MIN.)
  4. Selection of f<sub>sam</sub> = f<sub>XX</sub>/2<sup>N</sup>, f<sub>XX</sub>/32, f<sub>XX</sub>/64, and f<sub>XX</sub>/128 is possible with bits 0 and 1 (SCS0, SCS1) of the sampling clock select register (SCS) (when N = 0 to 4).

T<sub>CY</sub> vs. V<sub>DD</sub> (@f<sub>XX</sub> = f<sub>X</sub>/2 main system clock operation)



T<sub>CY</sub> vs. V<sub>DD</sub> (@f<sub>XX</sub> = f<sub>X</sub> main system clock operation)



**(2) Read/write operation**
**(a) When MCS = 1, PCC2 to PCC0 = 000B (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 3.5 to 5.5 V)**

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t <sub>ASTH</sub>		0.85t <sub>cy</sub> - 50		ns
Address setup time	t <sub>ADS</sub>		0.85t <sub>cy</sub> - 50		ns
Address hold time	t <sub>ADH</sub>		50		ns
Data input time from address	t <sub>ADD1</sub>			(2.85 + 2n)t <sub>cy</sub> - 80	ns
	t <sub>ADD2</sub>			(4 + 2n)t <sub>cy</sub> - 100	ns
Data input time from $\overline{RD}\downarrow$	t <sub>RDD1</sub>			(2 + 2n)t <sub>cy</sub> - 100	ns
	t <sub>RDD2</sub>			(2.85 + 2n)t <sub>cy</sub> - 100	ns
Read data hold time	t <sub>RDH</sub>		0		ns
$\overline{RD}$ low-level width	t <sub>RDL1</sub>		(2 + 2n)t <sub>cy</sub> - 60		ns
	t <sub>RDL2</sub>		(2.85 + 2n)t <sub>cy</sub> - 60		ns
$\overline{WAIT}\downarrow$ input time from $\overline{RD}\downarrow$	t <sub>RDWT1</sub>			0.85t <sub>cy</sub> - 50	ns
	t <sub>RDWT2</sub>			2t <sub>cy</sub> - 60	ns
$\overline{WAIT}\downarrow$ input time from $\overline{WR}\downarrow$	t <sub>WRWT</sub>			2t <sub>cy</sub> - 60	ns
$\overline{WAIT}$ low-level width	t <sub>WTL</sub>		(1.15 + 2n)t <sub>cy</sub>	(2 + 2n)t <sub>cy</sub>	ns
Write data setup time	t <sub>WDS</sub>		(2.85 + 2n)t <sub>cy</sub> - 100		ns
Write data hold time	t <sub>WDH</sub>		20		ns
$\overline{WR}$ low-level width	t <sub>WRL</sub>		(2.85 + 2n)t <sub>cy</sub> - 60		ns
$\overline{RD}\downarrow$ delay time from ASTB $\downarrow$	t <sub>ASTRD</sub>		25		ns
$\overline{WR}\downarrow$ delay time from ASTB $\downarrow$	t <sub>ASTWR</sub>		0.85t <sub>cy</sub> + 20		ns
ASTB $\uparrow$ delay time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDAST</sub>		0.85t <sub>cy</sub> - 10	1.15t <sub>cy</sub> + 20	ns
Address hold time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDADH</sub>		0.85t <sub>cy</sub> - 50	1.15t <sub>cy</sub> + 50	ns
Write data output time from $\overline{RD}\uparrow$	t <sub>RDWD</sub>		40		ns
Write data output time from $\overline{WR}\downarrow$	t <sub>WRWD</sub>		0	50	ns
Address hold time from $\overline{WR}\uparrow$	t <sub>WRADH</sub>		0.85t <sub>cy</sub>	1.15t <sub>cy</sub> + 40	ns
$\overline{RD}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTRD</sub>		1.15t <sub>cy</sub> + 40	3.15t <sub>cy</sub> + 40	ns
$\overline{WR}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTWR</sub>		1.15t <sub>cy</sub> + 30	3.15t <sub>cy</sub> + 30	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3. t<sub>cy</sub> = T<sub>cy</sub>/4
  4. n indicates the number of waits.

(b) When MCS = 0 or PCC2 to PCC0 ≠ 000B (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t <sub>ASTH</sub>		t <sub>CY</sub> - 80		ns
Address setup time	t <sub>ADS</sub>		t <sub>CY</sub> - 80		ns
Address hold time	t <sub>ADH</sub>		0.4t <sub>CY</sub> - 10		ns
Data input time from address	t <sub>ADD1</sub>			(3 + 2n)t <sub>CY</sub> - 160	ns
	t <sub>ADD2</sub>			(4 + 2n)t <sub>CY</sub> - 200	ns
Data input time from $\overline{RD}\downarrow$	t <sub>RDD1</sub>			(1.4 + 2n)t <sub>CY</sub> - 70	ns
	t <sub>RDD2</sub>			(2.4 + 2n)t <sub>CY</sub> - 70	ns
Read data hold time	t <sub>RDH</sub>		0		ns
$\overline{RD}$ low-level width	t <sub>RDL1</sub>		(1.4 + 2n)t <sub>CY</sub> - 20		ns
	t <sub>RDL2</sub>		(2.4 + 2n)t <sub>CY</sub> - 20		ns
$\overline{WAIT}\downarrow$ input time from $\overline{RD}\downarrow$	t <sub>RDWT1</sub>			t <sub>CY</sub> - 100	ns
	t <sub>RDWT2</sub>			2t <sub>CY</sub> - 100	ns
$\overline{WAIT}\downarrow$ input time from $\overline{WR}\downarrow$	t <sub>WRWT</sub>			2t <sub>CY</sub> - 100	ns
$\overline{WAIT}$ low-level width	t <sub>WTL</sub>		(1 + 2n)t <sub>CY</sub>	(2 + 2n)t <sub>CY</sub>	ns
Write data setup time	t <sub>WDS</sub>		(2.4 + 2n)t <sub>CY</sub> - 60		ns
Write data hold time	t <sub>WDH</sub>		20		ns
$\overline{WR}$ low-level width	t <sub>WRL</sub>		(2.4 + 2n)t <sub>CY</sub> - 20		ns
$\overline{RD}\downarrow$ delay time from $\overline{ASTB}\downarrow$	t <sub>ASTRD</sub>		0.4t <sub>CY</sub> - 30		ns
$\overline{WR}\downarrow$ delay time from $\overline{ASTB}\downarrow$	t <sub>ASTWR</sub>		1.4t <sub>CY</sub> - 30		ns
ASTB $\uparrow$ delay time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDAST</sub>		t <sub>CY</sub> - 10	t <sub>CY</sub> + 20	ns
Address hold time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDADH</sub>		t <sub>CY</sub> - 50	t <sub>CY</sub> + 50	ns
Write data output time from $\overline{RD}\uparrow$	t <sub>RDWD</sub>		0.4t <sub>CY</sub> - 20		ns
Write data output time from $\overline{WR}\downarrow$	t <sub>WRWD</sub>		0	60	ns
Address hold time from $\overline{WR}\uparrow$	t <sub>WRADH</sub>		t <sub>CY</sub>	t <sub>CY</sub> + 60	ns
$\overline{RD}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTRD</sub>		0.6t <sub>CY</sub> + 180	2.6t <sub>CY</sub> + 180	ns
$\overline{WR}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTWR</sub>		0.6t <sub>CY</sub> + 120	2.6t <sub>CY</sub> + 120	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3. t<sub>CY</sub> = T<sub>CY</sub>/4
  4. n indicates the number of waits.

(c) When MCS = 0 or PCC2 to PCC0 ≠ 000B (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t <sub>ASTH</sub>		t <sub>cy</sub> - 150		ns
Address setup time	t <sub>ADS</sub>		t <sub>cy</sub> - 150		ns
Address hold time	t <sub>ADH</sub>		0.37t <sub>cy</sub> - 40		ns
Data input time from address	t <sub>ADD1</sub>			(3 + 2n)t <sub>cy</sub> - 320	ns
	t <sub>ADD2</sub>			(4 + 2n)t <sub>cy</sub> - 300	ns
Data input time from $\overline{RD}\downarrow$	t <sub>RDD1</sub>			(1.37 + 2n)t <sub>cy</sub> - 120	ns
	t <sub>RDD2</sub>			(2.37 + 2n)t <sub>cy</sub> - 120	ns
Read data hold time	t <sub>RDH</sub>		0		ns
$\overline{RD}$ low-level width	t <sub>RDL1</sub>		(1.37 + 2n)t <sub>cy</sub> - 20		ns
	t <sub>RDL2</sub>		(2.37 + 2n)t <sub>cy</sub> - 20		ns
$\overline{WAIT}\downarrow$ input time from $\overline{RD}\downarrow$	t <sub>RDWT1</sub>			t <sub>cy</sub> - 200	ns
	t <sub>RDWT2</sub>			2t <sub>cy</sub> - 200	ns
$\overline{WAIT}\downarrow$ input time from $\overline{WR}\downarrow$	t <sub>WRWT</sub>			2t <sub>cy</sub> - 200	ns
$\overline{WAIT}$ low-level width	t <sub>WTL</sub>		(1 + 2n)t <sub>cy</sub>	(2 + 2n)t <sub>cy</sub>	ns
Write data setup time	t <sub>WDS</sub>		(2.37 + 2n)t <sub>cy</sub> - 100		ns
Write data hold time	t <sub>WDH</sub>		20		ns
$\overline{WR}$ low-level width	t <sub>WRL</sub>		(2.37 + 2n)t <sub>cy</sub> - 20		ns
$\overline{RD}\downarrow$ delay time from ASTB $\downarrow$	t <sub>ASTRD</sub>		0.37t <sub>cy</sub> - 50		ns
$\overline{WR}\downarrow$ delay time from ASTB $\downarrow$	t <sub>ASTWR</sub>		1.37t <sub>cy</sub> - 50		ns
ASTB $\uparrow$ delay time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDAST</sub>		t <sub>cy</sub> - 10	t <sub>cy</sub> + 20	ns
Address hold time from $\overline{RD}\uparrow$ at external fetch	t <sub>RDADH</sub>		t <sub>cy</sub> - 50	t <sub>cy</sub> + 50	ns
Write data output time from $\overline{RD}\uparrow$	t <sub>RDWD</sub>		0.37t <sub>cy</sub> - 40		ns
Write data output time from $\overline{WR}\downarrow$	t <sub>WRWD</sub>		0	120	ns
Address hold time from $\overline{WR}\uparrow$	t <sub>WRADH</sub>		t <sub>cy</sub>	t <sub>cy</sub> + 120	ns
$\overline{RD}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTRD</sub>		0.63t <sub>cy</sub> + 350	2.63t <sub>cy</sub> + 350	ns
$\overline{WR}\uparrow$ delay time from $\overline{WAIT}\uparrow$	t <sub>WTWR</sub>		0.63t <sub>cy</sub> + 240	2.63t <sub>cy</sub> + 240	ns

- Remarks**
1. MCS: Bit 0 of the oscillation mode select register (OSMS)
  2. PCC2 to PCC0: Bits 2 to 0 of the processor clock control register (PCC)
  3. t<sub>cy</sub> = T<sub>cy</sub>/4
  4. n indicates the number of waits.

**(3) Serial interface (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.7 to 5.5 V)**
**(a) Serial interface channel 0**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t <sub>KCY1</sub>	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	800			ns
		2.7 V ≤ V <sub>DD</sub> < 4.5 V	1,600			ns
		2.2 V ≤ V <sub>DD</sub> < 2.7 V	3,200			ns
$\overline{\text{SCK0}}$ high-/low-level width	t <sub>KH1</sub> , t <sub>KL1</sub>	V <sub>DD</sub> = 4.5 to 5.5 V	t <sub>KCY1</sub> /2 - 50			ns
		V <sub>DD</sub> = 2.2 to 5.5 V	t <sub>KCY1</sub> /2 - 100			ns
SIO setup time (to $\overline{\text{SCK0}}\uparrow$ )	t <sub>SIK1</sub>	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	100			ns
		2.7 V ≤ V <sub>DD</sub> < 4.5 V	150			ns
		2.2 V ≤ V <sub>DD</sub> < 2.7 V	300			ns
SIO hold time (from $\overline{\text{SCK0}}\uparrow$ )	t <sub>KSH1</sub>		400			ns
SO0 output delay time from $\overline{\text{SCK0}}\downarrow$	t <sub>KSO1</sub>	C = 100 pF <sup>Note</sup>			300	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK0}}$  and SO0 output lines.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t <sub>KCY2</sub>	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	800			ns
		2.7 V ≤ V <sub>DD</sub> < 4.5 V	1,600			ns
		2.2 V ≤ V <sub>DD</sub> < 2.7 V	3,200			ns
$\overline{\text{SCK0}}$ high-/low-level width	t <sub>KH2</sub> , t <sub>KL2</sub>	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	400			ns
		2.7 V ≤ V <sub>DD</sub> < 4.5 V	800			ns
		2.2 V ≤ V <sub>DD</sub> < 2.7 V	1,600			ns
SIO setup time (to $\overline{\text{SCK0}}\uparrow$ )	t <sub>SIK2</sub>		100			ns
SIO hold time (from $\overline{\text{SCK0}}\uparrow$ )	t <sub>KSH2</sub>		400			ns
SO0 output delay time from $\overline{\text{SCK0}}\downarrow$	t <sub>KSO2</sub>	C = 100 pF <sup>Note</sup>			300	ns
$\overline{\text{SCK0}}$ rise/fall time	t <sub>R2</sub> , t <sub>F2</sub>	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO0 output line.

**(iii) 2-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... Internal clock output)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY3}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	1,600			ns
			2.2 V $\leq$ V <sub>DD</sub> < 2.7 V	3,200			ns
$\overline{\text{SCK0}}$ high-level width	$t_{\text{KH3}}$		V <sub>DD</sub> = 2.7 to 5.5 V	$t_{\text{KCY3}}/2 - 160$			ns
			V <sub>DD</sub> = 2.2 to 5.5 V	$t_{\text{KCY3}}/2 - 190$			ns
$\overline{\text{SCK0}}$ low-level width	$t_{\text{KL3}}$		V <sub>DD</sub> = 4.5 to 5.5 V	$t_{\text{KCY3}}/2 - 50$			ns
			V <sub>DD</sub> = 2.2 to 5.5 V	$t_{\text{KCY3}}/2 - 100$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK3}}$		4.5 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	300			ns
			2.7 V $\leq$ V <sub>DD</sub> < 4.5 V	350			ns
			2.2 V $\leq$ V <sub>DD</sub> < 2.7 V	400			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI3}}$			600			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO3}}$			0		300	ns

**Note** R and C are the load resistance and load capacitance of the  $\overline{\text{SCK0}}$ , SB0, and SB1 output lines.

**(iv) 2-wire serial I/O mode ( $\overline{\text{SCK0}}$  ... External clock input)**

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY4}}$	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V		1,600			ns
		2.2 V $\leq$ V <sub>DD</sub> < 2.7 V		3,200			ns
$\overline{\text{SCK0}}$ high-level width	$t_{\text{KH4}}$	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V		650			ns
		2.2 V $\leq$ V <sub>DD</sub> < 2.7 V		1,300			ns
$\overline{\text{SCK0}}$ low-level width	$t_{\text{KL4}}$	2.7 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V		800			ns
		2.2 V $\leq$ V <sub>DD</sub> < 2.7 V		1,600			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK4}}$			100			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI4}}$			$t_{\text{KCY4}}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO4}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	4.5 V $\leq$ V <sub>DD</sub> $\leq$ 5.5 V	0		300	ns
			2.2 V $\leq$ V <sub>DD</sub> < 4.5 V	0		500	ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{R4}}, t_{\text{F4}}$	When using external device expansion function				160	ns
		When not using external device expansion function				1,000	ns

**Note** R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.



**(v) SBI mode ( $\overline{\text{SCK0}}$  ... Internal clock output) ( $\mu\text{PD78F0058}$ , 78F0058Y only)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	3,200			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH5}}, t_{\text{KL5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	$t_{\text{KCY5}}/2 - 50$			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	$t_{\text{KCY5}}/2 - 150$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK5}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	300			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI5}}$		$t_{\text{KCY5}}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO5}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	0	250	ns
			$V_{\text{DD}} = 2.2 \text{ to } 5.5 \text{ V}$	0	1,000	ns
SB0, SB1 $\downarrow$ from $\overline{\text{SCK0}}\uparrow$	$t_{\text{KSB}}$		$t_{\text{KCY5}}$			ns
$\overline{\text{SCK0}}\downarrow$ from SB0, SB1 $\downarrow$	$t_{\text{SBK}}$		$t_{\text{KCY5}}$			ns
SB0, SB1 high-level width	$t_{\text{SBH}}$		$t_{\text{KCY5}}$			ns
SB0, SB1 low-level width	$t_{\text{SBL}}$		$t_{\text{KCY5}}$			ns

**Note** R and C are the load resistance and load capacitance of the  $\overline{\text{SCK0}}$ , SB0, and SB1 output lines.

**(vi) SBI mode ( $\overline{\text{SCK0}}$  ... External clock input) ( $\mu\text{PD78F0058}$ , 78F0058Y only)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	$t_{\text{KCY6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	3,200			ns
$\overline{\text{SCK0}}$ high-/low-level width	$t_{\text{KH6}}, t_{\text{KL6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{SIK6}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	300			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}\uparrow$ )	$t_{\text{KSI6}}$		$t_{\text{KCY6}}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}\downarrow$	$t_{\text{KSO6}}$	R = 1 k $\Omega$ , C = 100 pF <sup>Note</sup>	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	0	300	ns
			$V_{\text{DD}} = 2.2 \text{ to } 5.5 \text{ V}$	0	1,000	ns
SB0, SB1 $\downarrow$ from $\overline{\text{SCK0}}\uparrow$	$t_{\text{KSB}}$		$t_{\text{KCY6}}$			ns
$\overline{\text{SCK0}}\downarrow$ from SB0, SB1 $\downarrow$	$t_{\text{SBK}}$		$t_{\text{KCY6}}$			ns
SB0, SB1 high-level width	$t_{\text{SBH}}$		$t_{\text{KCY6}}$			ns
SB0, SB1 low-level width	$t_{\text{SBL}}$		$t_{\text{KCY6}}$			ns
$\overline{\text{SCK0}}$ rise/fall time	$t_{\text{r6}}, t_{\text{f6}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

(vii) I<sup>2</sup>C bus mode (SCL ... Internal clock output) (μPD78F0058Y only)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
SCL cycle time	t <sub>KCY7</sub>	R = 1 kΩ, C = 100 pF <sup>Note</sup>	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	10			μs
			2.2 V ≤ V <sub>DD</sub> < 2.7 V	20			ns
SCL high-level width	t <sub>KH7</sub>		V <sub>DD</sub> = 2.7 to 5.5 V	t <sub>KCY7</sub> – 160			ns
			V <sub>DD</sub> = 2.2 to 5.5 V	t <sub>KCY7</sub> – 190			ns
SCL low-level width	t <sub>KL7</sub>		V <sub>DD</sub> = 4.5 to 5.5 V	t <sub>KCY7</sub> – 50			ns
			V <sub>DD</sub> = 2.2 to 5.5 V	t <sub>KCY7</sub> – 100			ns
SDA0, SDA1 setup time (to SCL↑)	t <sub>SIK7</sub>		2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	200			ns
			2.2 V ≤ V <sub>DD</sub> < 2.7 V	300			ns
SDA0, SDA1 hold time (from SCL↓)	t <sub>KSI7</sub>			0			ns
SDA0, SDA1 output delay time from SCL↓	t <sub>KSO7</sub>		4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0		300	ns
			2.2 V ≤ V <sub>DD</sub> < 4.5 V	0		500	ns
SDA0, SDA1↓ from SCL↑ or SDA0, SDA1↑ from SCL↑	t <sub>KSB</sub>			200			ns
SCL↓ from SDA0, SDA1↓	t <sub>SBK</sub>			400			ns
SDA0, SDA1 high-level width	t <sub>SBH</sub>			500			ns

**Note** R and C are the load resistance and load capacitance of the SCL, SDA0, and SDA1 output lines.

 (viii) I<sup>2</sup>C bus mode (SCL ... External clock input) (μPD78F0058Y only)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
SCL cycle time	t <sub>KCY8</sub>			1			μs
SCL high-level width	t <sub>KH8</sub>			400			ns
SDA0, SDA1 setup time (to SCL↑)	t <sub>SIK8</sub>			200			ns
SDA0, SDA1 hold time (from SCL↓)	t <sub>KSI8</sub>			0			ns
SDA0, SDA1 output delay time from SCL↓	t <sub>KSO8</sub>	R = 1 kΩ, C = 100 pF <sup>Note</sup>	4.5 V ≤ V <sub>DD</sub> ≤ 5.5 V	0		300	ns
			2.2 V ≤ V <sub>DD</sub> < 4.5 V	0		500	ns
SDA0, SDA1↓ from SCL↑ or SDA0, SDA1↑ from SCL↑	t <sub>KSB</sub>			200			ns
SCL↓ from SDA0, SDA1↓	t <sub>SBK</sub>			400			ns
SDA0, SDA1 high-level width	t <sub>SBH</sub>			500			ns

**Note** R and C are the load resistance and load capacitance of the SDA0 and SDA1 output lines.

**(b) Serial interface channel 1**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK1}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY9}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH9}}, t_{\text{KL9}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY9}}/2 - 50$			ns
		$V_{\text{DD}} = 2.2 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY9}}/2 - 100$			ns
S11 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK9}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
S11 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSI9}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO9}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK1}}$  and SO1 output lines.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK1}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY10}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH10}}, t_{\text{KL10}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
S11 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK10}}$		100			ns
S11 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KIS10}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO10}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
$\overline{\text{SCK1}}$ rise/fall time	$t_{\text{R10}}, t_{\text{F10}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO1 output line.

(iii) 3-wire serial I/O mode with automatic transmit/receive function ( $\overline{\text{SCK1}}$  ... Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY11}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH11}}, t_{\text{KL11}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY11}}/2 - 50$			ns
		$V_{\text{DD}} = 2.2 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY11}}/2 - 100$			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK11}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSH11}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO11}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
STB $\uparrow$ from $\overline{\text{SCK1}}\uparrow$	$t_{\text{SBD}}$		$t_{\text{KCY11}}/2 - 100$		$t_{\text{KCY11}}/2 + 100$	ns
Strobe signal high-level width	$t_{\text{SBW}}$	$2.7 \text{ V} \leq V_{\text{DD}} < 5.5 \text{ V}$	$t_{\text{KCY11}} - 30$		$t_{\text{KCY11}} + 30$	ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	$t_{\text{KCY11}} - 60$		$t_{\text{KCY11}} + 60$	ns
Busy signal setup time (to busy signal detection timing)	$t_{\text{BYS}}$		100			ns
Busy signal hold time (from busy signal detection timing)	$t_{\text{BYH}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	200			ns
$\overline{\text{SCK1}}\downarrow$ from busy inactive	$t_{\text{SPS}}$				$2t_{\text{KCY11}}$	ns

**Note** C is the load capacitance of the  $\overline{\text{SCK1}}$  and SO1 output lines.

 (iv) 3-wire serial I/O mode with automatic transmit/receive function ( $\overline{\text{SCK1}}$  ... External clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK1}}$ cycle time	$t_{\text{KCY12}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
$\overline{\text{SCK1}}$ high-/low-level width	$t_{\text{KH12}}, t_{\text{KL12}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
SI1 setup time (to $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{SIK12}}$		100			ns
SI1 hold time (from $\overline{\text{SCK1}}\uparrow$ )	$t_{\text{KSH12}}$		400			ns
SO1 output delay time from $\overline{\text{SCK1}}\downarrow$	$t_{\text{KSO12}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
$\overline{\text{SCK1}}$ rise/fall time	$t_{\text{R12}}, t_{\text{F12}}$	When using external device expansion function			160	ns
		When not using external device expansion function			1,000	ns

**Note** C is the load capacitance of the SO1 output line.

**(c) Serial interface channel 2**
**(i) 3-wire serial I/O mode ( $\overline{\text{SCK2}}$  ... Internal clock output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK2}}$ cycle time	$t_{\text{KCY13}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
$\overline{\text{SCK2}}$ high-/low-level width	$t_{\text{KH13}}$ ,	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY13}}/2 - 50$			ns
	$t_{\text{KL13}}$	$V_{\text{DD}} = 2.2 \text{ to } 5.5 \text{ V}$	$t_{\text{KCY13}}/2 - 100$			ns
SI2 setup time (to $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{SIK13}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	100			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	150			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	300			ns
SI2 hold time (from $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{KSI13}}$		400			ns
SO2 output delay time from $\overline{\text{SCK2}}\downarrow$	$t_{\text{KSO13}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns

**Note** C is the load capacitance of the SO2 output line.

**(ii) 3-wire serial I/O mode ( $\overline{\text{SCK2}}$  ... External clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK2}}$ cycle time	$t_{\text{KCY14}}$	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	800			ns
		$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	1,600			ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	3,200			ns
$\overline{\text{SCK2}}$ high-/low-level width	$t_{\text{KH14}}$ ,	$4.5 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$	400			ns
		$t_{\text{KL14}}$	$2.7 \text{ V} \leq V_{\text{DD}} < 4.5 \text{ V}$	800		ns
		$2.2 \text{ V} \leq V_{\text{DD}} < 2.7 \text{ V}$	1,600			ns
SI2 setup time (to $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{SIK14}}$		100			ns
SI2 hold time (from $\overline{\text{SCK2}}\uparrow$ )	$t_{\text{KSI14}}$		400			ns
SO2 output delay time from $\overline{\text{SCK2}}\downarrow$	$t_{\text{KSO14}}$	$C = 100 \text{ pF}^{\text{Note}}$			300	ns
$\overline{\text{SCK2}}$ rise/fall time	$t_{\text{R14}}$ ,	Other than below			160	ns
	$t_{\text{F14}}$	$V_{\text{DD}} = 4.5 \text{ to } 5.5 \text{ V}$ When not using external device expansion function			1	$\mu\text{s}$

**Note** C is the load capacitance of the SO2 output line.

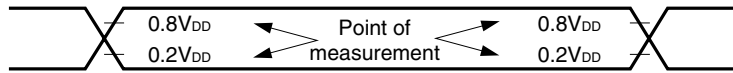
**(iii) UART mode (dedicated baud rate generator output)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			78,125	bps
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$			39,063	bps
		$2.2\text{ V} \leq V_{DD} < 2.7\text{ V}$			19,531	bps

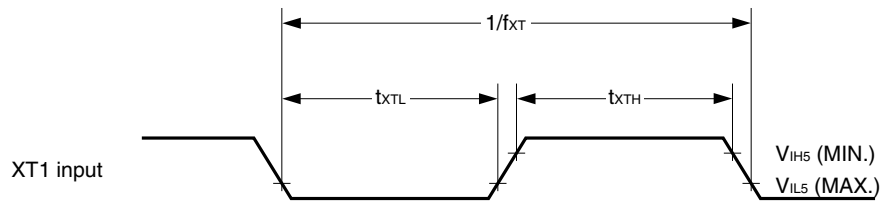
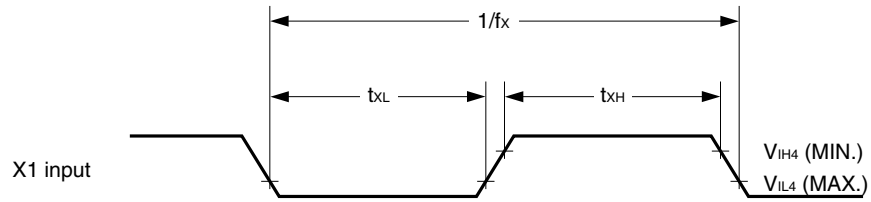
**(iv) UART mode (external clock input)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK cycle time	t <sub>KCY15</sub>	$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	800			ns
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$	1,600			ns
		$2.2\text{ V} \leq V_{DD} < 2.7\text{ V}$	3,200			ns
ASCK high-/low-level width	t <sub>KH15</sub> , t <sub>KL15</sub>	$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	400			ns
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$	800			ns
		$2.2\text{ V} \leq V_{DD} < 2.7\text{ V}$	1,600			ns
Transfer rate		$4.5\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			39,063	bps
		$2.7\text{ V} \leq V_{DD} < 4.5\text{ V}$			19,531	bps
		$2.2\text{ V} \leq V_{DD} < 2.7\text{ V}$			9,766	bps
ASCK rise/fall time	t <sub>R15</sub> , t <sub>F15</sub>	V <sub>DD</sub> = 4.5 to 5.5 V, when not using external device expansion function.			1,000	ns
		Other than above			160	ns

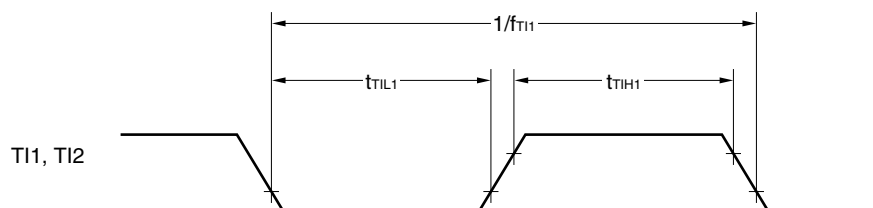
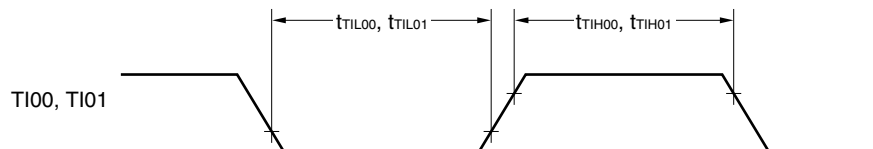
**AC Timing Measurement Points (Excluding X1, XT1 Inputs)**



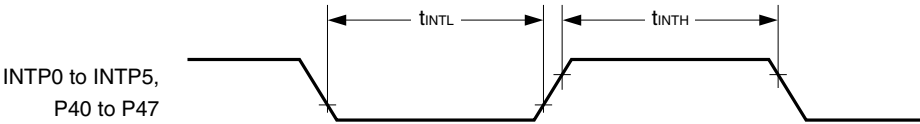
**Clock Timing**



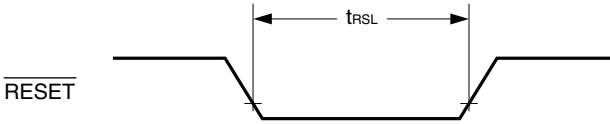
**TI Timing**



### Interrupt Request Input Timing



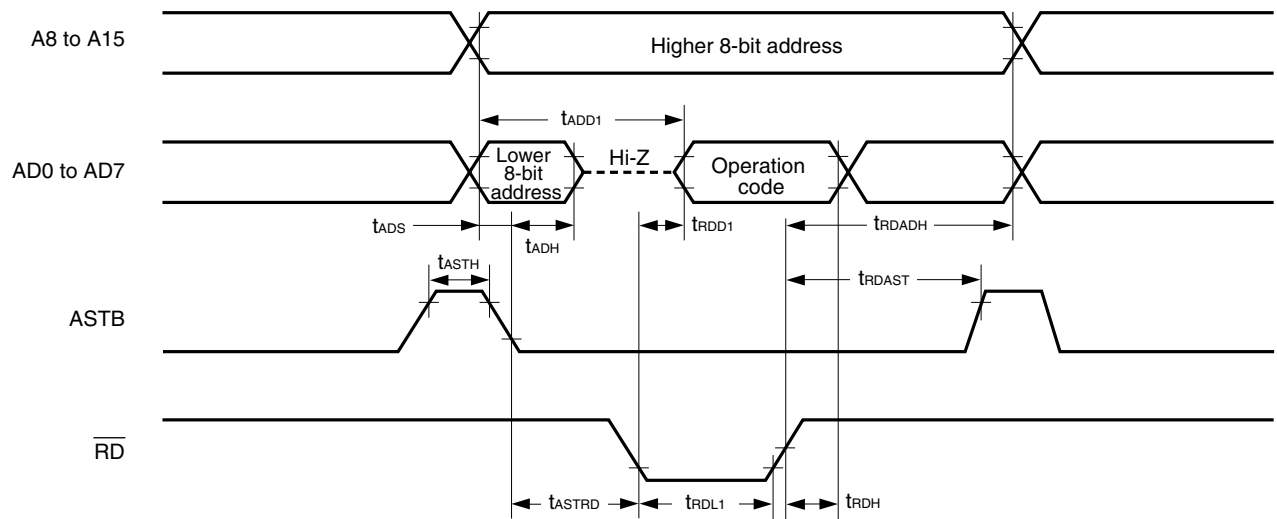
### RESET Input Timing



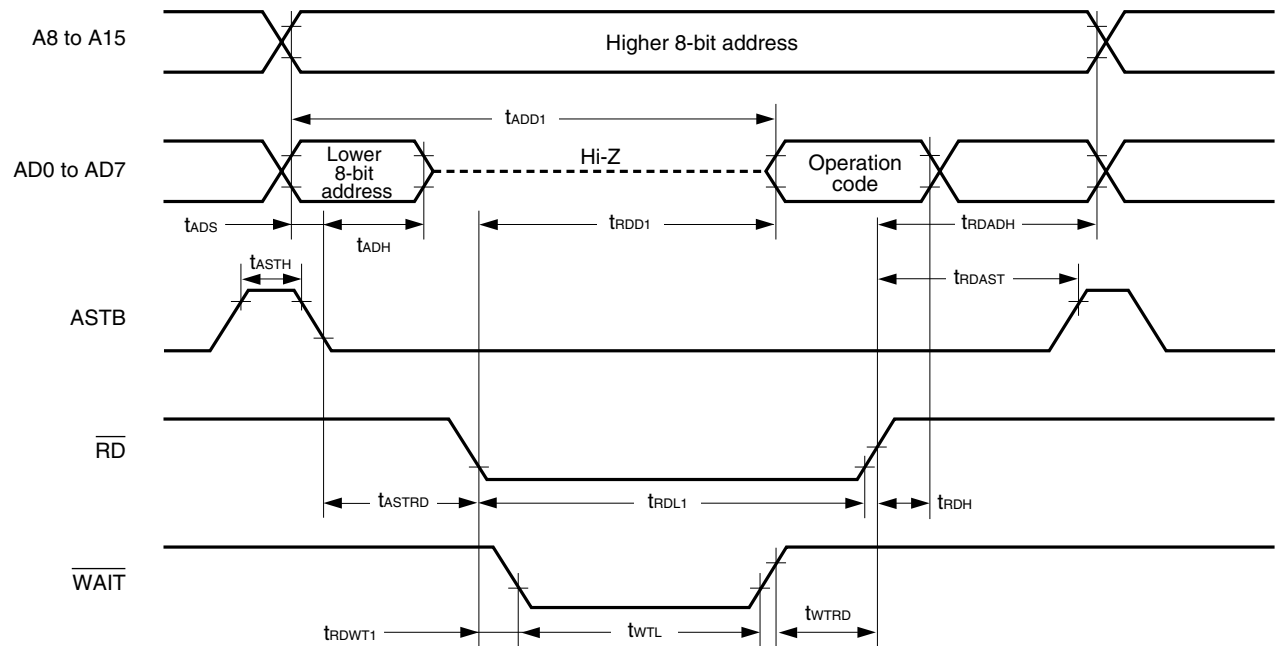


## Read/Write Operation

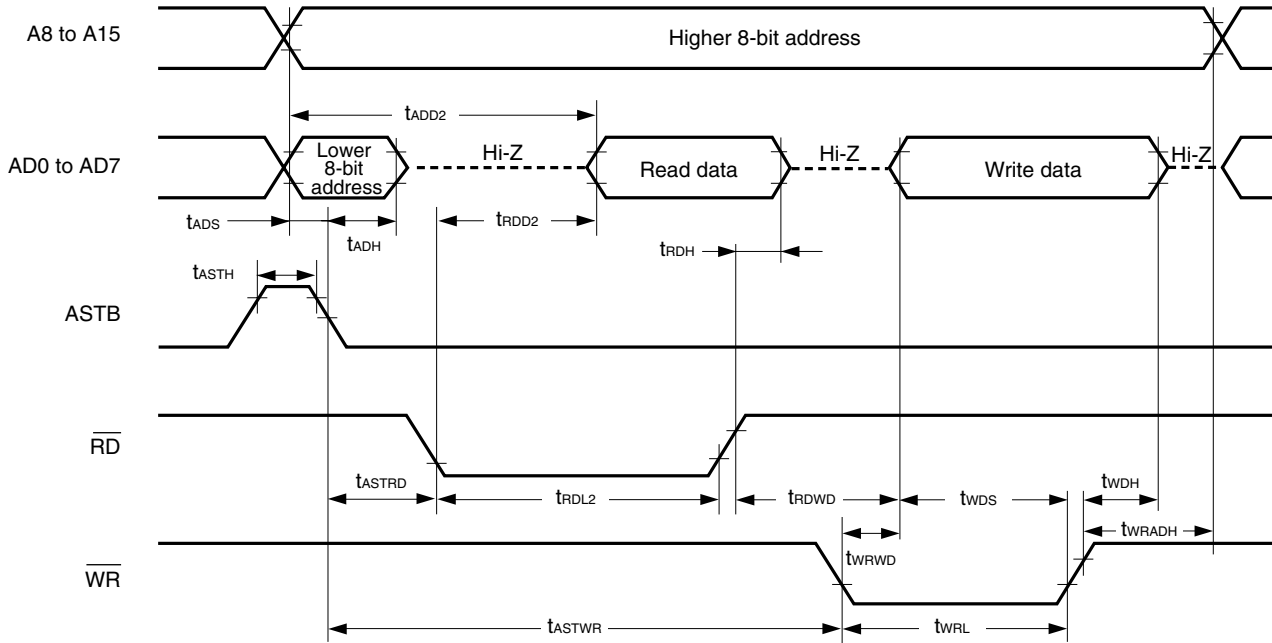
### External fetch (no wait):



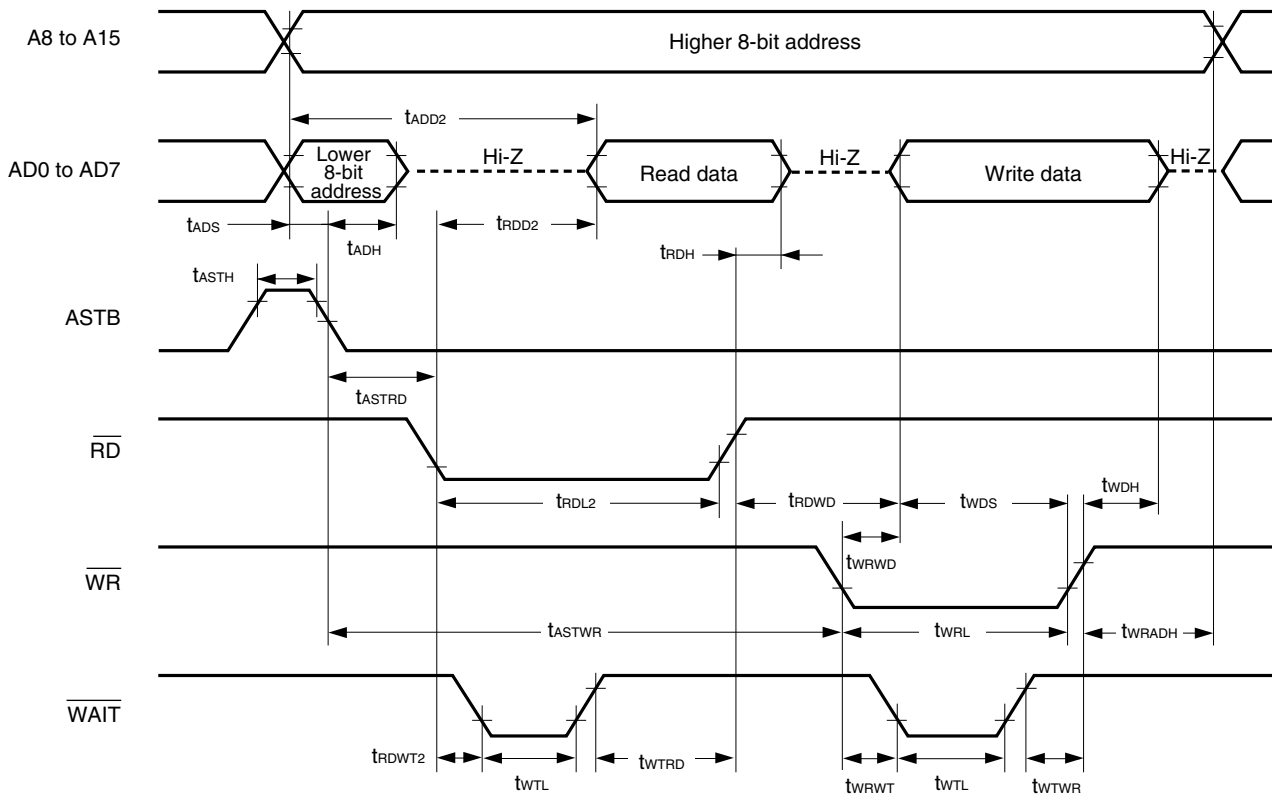
### External fetch (wait insertion):



External data access (no wait):

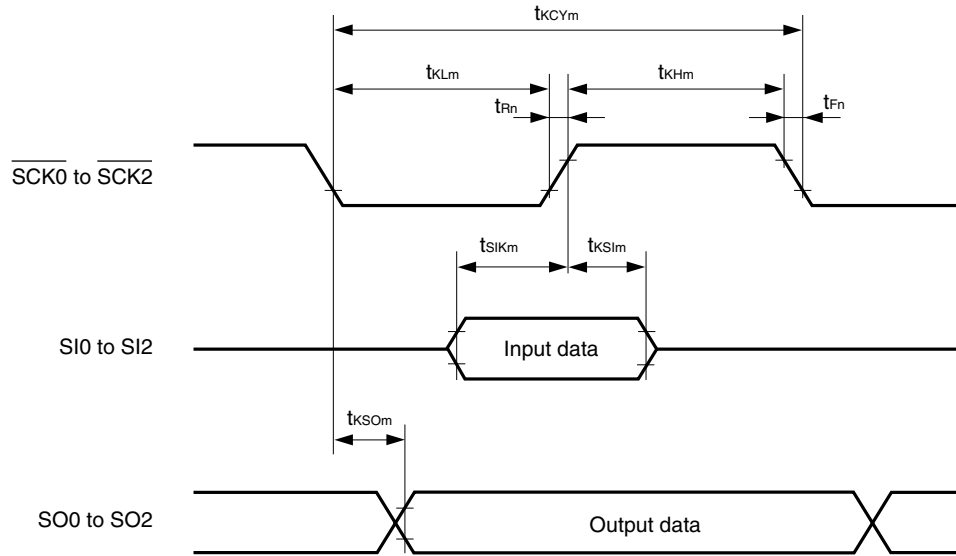


External data access (wait insertion):



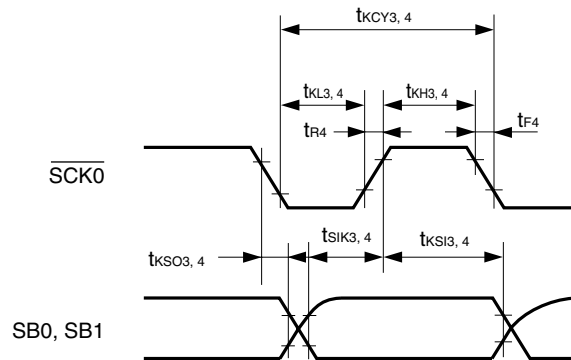
### Serial Transfer Timing

#### 3-wire serial I/O mode:

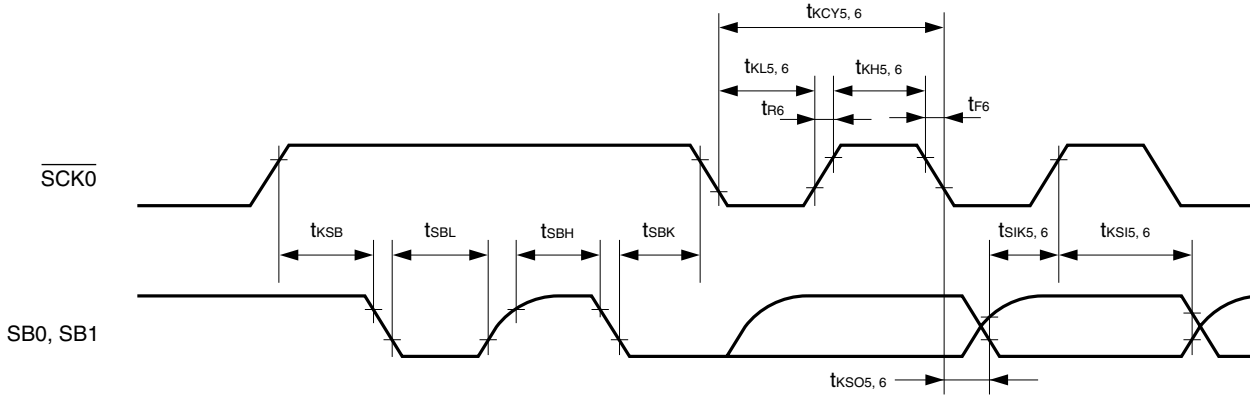


$m = 1, 2, 9, 10, 13, 14$   
 $n = 2, 10, 14$

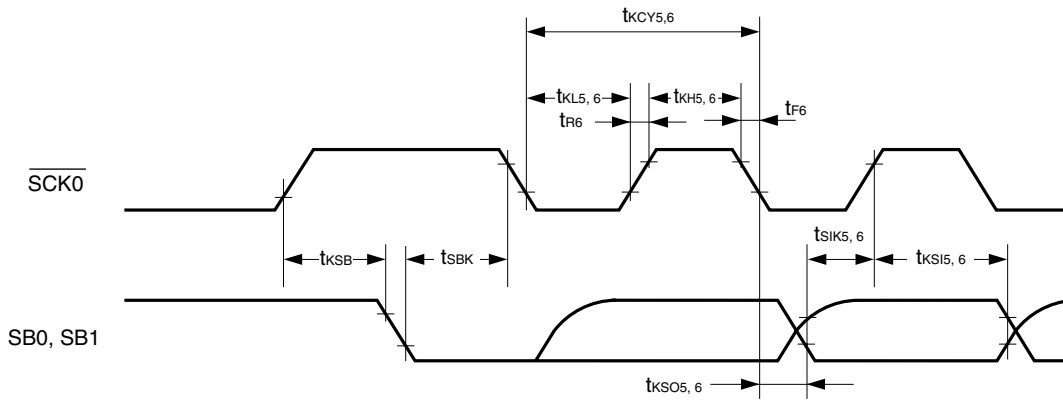
#### 2-wire serial I/O mode:



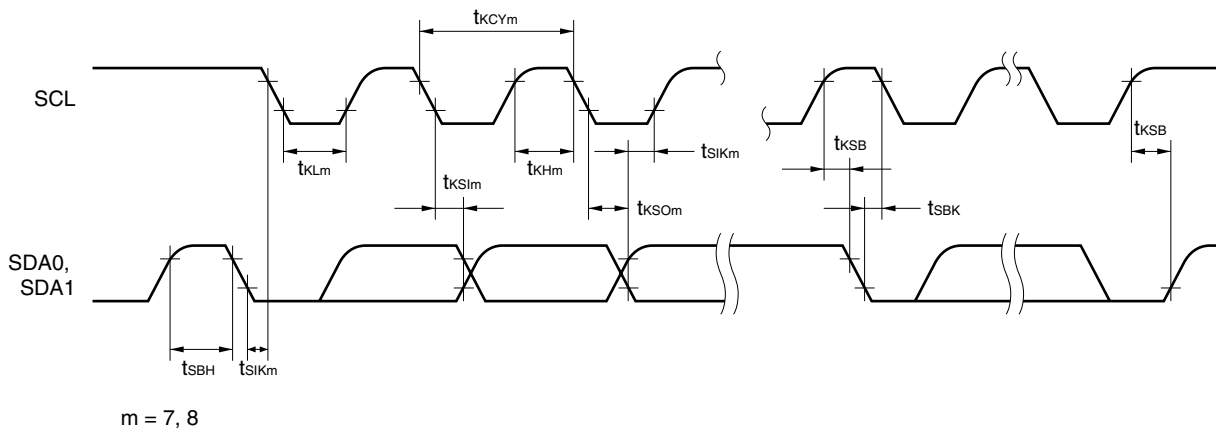
**SBI mode (bus release signal transfer):**



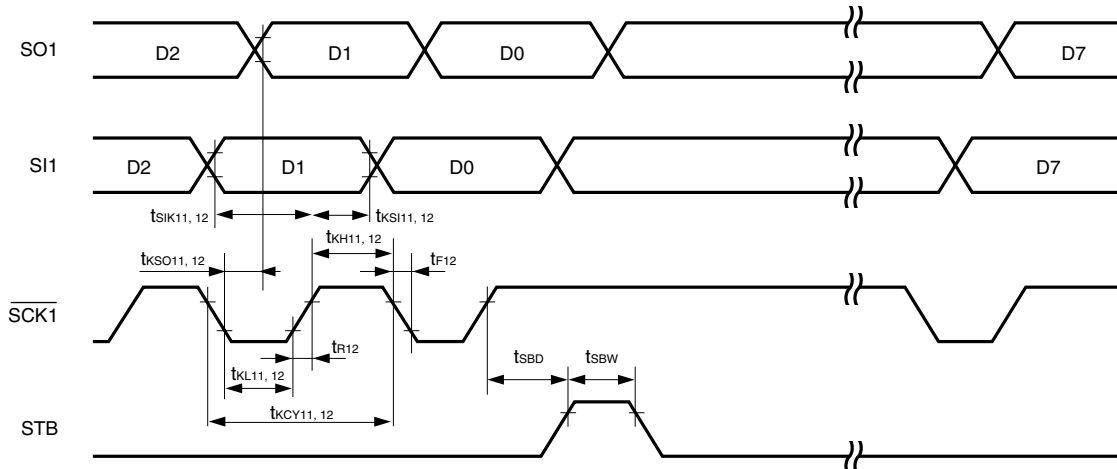
**SBI mode (command signal transfer):**



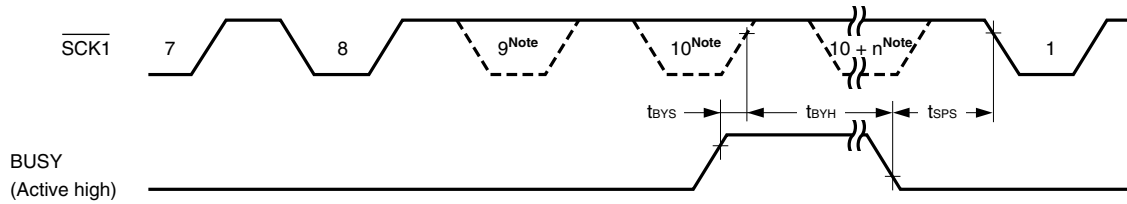
**I<sup>2</sup>C bus mode:**



**3-wire serial I/O mode with automatic transmit/receive function:**

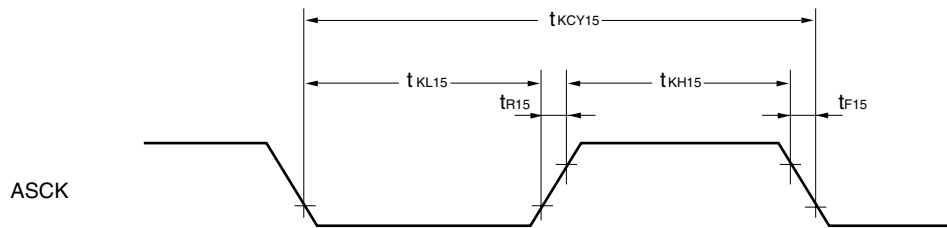


**3-wire serial I/O mode with automatic transmit/receive function (busy processing):**



**Note** The signal is not actually driven low here; it is shown as such to indicate the timing.

**UART mode (external clock input):**



**A/D Converter Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V, AV<sub>SS</sub> = V<sub>SS</sub> = 0 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error <sup>Note 1</sup>		2.7 V ≤ AV <sub>REF0</sub> ≤ 5.5 V			±0.6	%FSR
		2.2 V ≤ AV <sub>REF0</sub> < 2.7 V			±1.4	%FSR
Conversion time	T <sub>CONV1</sub>	2.2 V ≤ AV <sub>REF0</sub> < 2.7 V	40		100	μs
	T <sub>CONV2</sub>	2.7 V ≤ AV <sub>REF0</sub> < 5.5 V	16		100	μs
Analog input voltage	V <sub>IAN</sub>		AV <sub>SS</sub>		AV <sub>REF0</sub>	V
Reference voltage	AV <sub>REF0</sub>		2.2		V <sub>DD</sub>	V
AV <sub>REF0</sub> current	I <sub>REF0</sub>	When A/D converter is operating <sup>Note 2</sup>		500	1,500	μA
		When A/D converter is not operating <sup>Note 3</sup>		0	3.0	μA

- Notes**
1. Excludes quantization error (±1/2 LSB). This value is indicated as a ratio to the full-scale value (%FSR).
  2. The current flowing to the AV<sub>REF0</sub> pin when bit 7 (CS) of the A/D converter mode register (ADM) is 1.
  3. The current flowing to the AV<sub>REF0</sub> pin when bit 7 (CS) of the A/D converter mode register (ADM) is 0.

**D/A Converter Characteristics (T<sub>A</sub> = -40 to +85°C, V<sub>DD</sub> = 2.2 to 5.5 V, AV<sub>SS</sub> = V<sub>SS</sub> = 0 V)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution					8	bit
Overall error		R = 2 MΩ <sup>Note 1</sup>			±1.2	%
		R = 4 MΩ <sup>Note 1</sup>			±0.8	%
		R = 10 MΩ <sup>Note 1</sup>			±0.6	%
Overall error <sup>Note 1</sup>		C = 30 pF <sup>Note 1</sup>	AV <sub>REF1</sub> = 2.2 to 2.7 V		10	μs
			AV <sub>REF1</sub> = 2.2 to 5.5 V		15	μs
Output resistance	R <sub>O</sub>	<b>Note 2</b>		8		kΩ
Analog reference voltage	AV <sub>REF1</sub>		1.8		V <sub>DD</sub>	V
AV <sub>REF1</sub> current	I <sub>REF1</sub>	<b>Note 2</b>			2.5	mA
Resistance between AV <sub>REF1</sub> and AV <sub>SS</sub>	R <sub>AIREF1</sub>	DACS0, DACS1 = 55H <sup>Note 2</sup>	4	8		kΩ

- Notes**
1. R and C are the D/A converter output pin load resistance and load capacitance, respectively.
  2. Value for one D/A converter channel

**Remark** DACS0 and DACS1: D/A conversion value set registers 0, 1

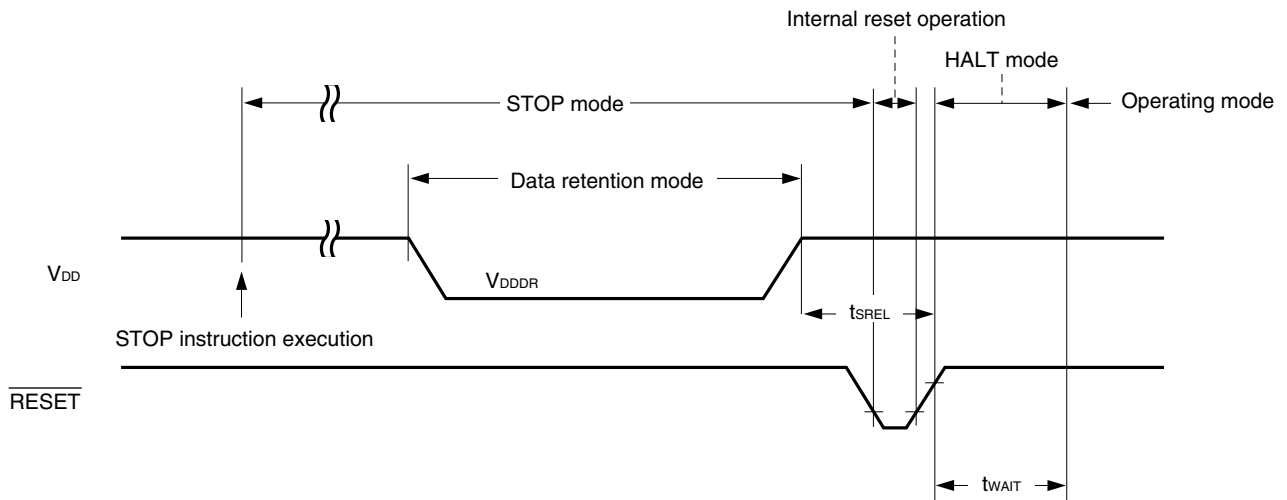
**Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (T<sub>A</sub> = -40 to +85°C)**

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention power supply voltage	V <sub>DDDR</sub>		1.8		5.5	V
Data retention power supply current	I <sub>DDDR</sub>	V <sub>DDDR</sub> = 1.8 V Subsystem clock stop and feed-back resistor disconnected		0.1	10	μA
Release signal set time	t <sub>SREL</sub>		0			μs
Oscillation stabilization wait time	t <sub>WAIT</sub>	Release by $\overline{\text{RESET}}$		2 <sup>17</sup> /f <sub>x</sub>		ms
		Release by interrupt request		<b>Note</b>		ms

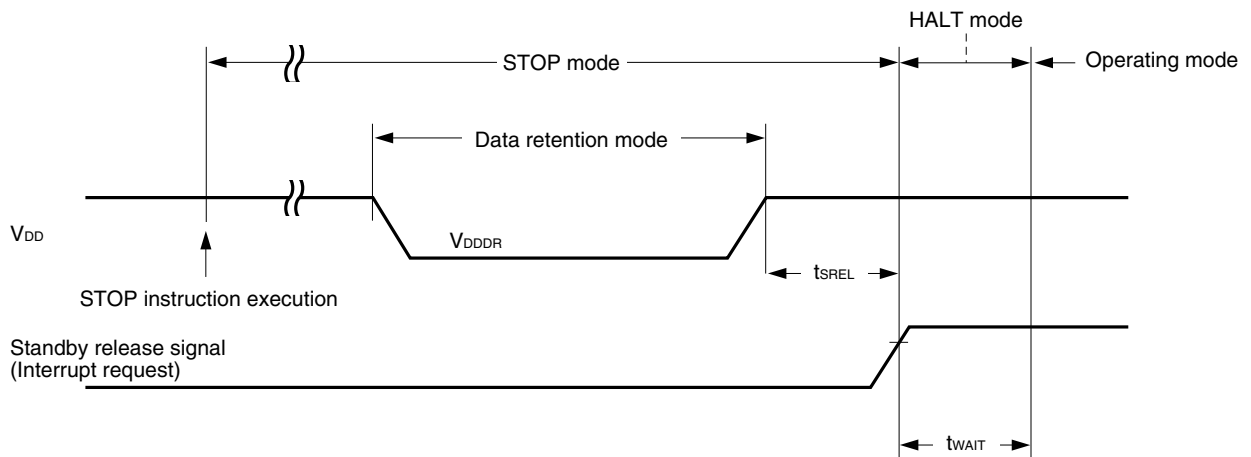
**Note** Selection of 2<sup>12</sup>/f<sub>xx</sub> and 2<sup>14</sup>/f<sub>xx</sub> to 2<sup>17</sup>/f<sub>xx</sub> is possible with bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register (OSTS).

**Remark** f<sub>xx</sub>: Main system clock frequency (f<sub>x</sub> or f<sub>x</sub>/2)  
f<sub>x</sub>: Main system clock oscillation frequency

**Data Retention Timing (STOP Mode Release by  $\overline{\text{RESET}}$ )**



**Data Retention Timing (Standby Release Signal: STOP Mode Release by Interrupt Request Signal)**



Flash Memory Programming Characteristics (V<sub>DD</sub> = 2.7 to 5.5 V, T<sub>A</sub> = 10 to 40°C)

## (1) Write/erase characteristics

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Write current (V <sub>DD</sub> pin) <sup>Note 1</sup>	I <sub>DDW</sub>	When V <sub>PP</sub> = V <sub>PP1</sub>	5.0 MHz crystal oscillation operation mode (f <sub>XX</sub> = 2.5 MHz) <sup>Note 2</sup>			15.5	mA
			5.0 MHz crystal oscillation operation mode (f <sub>XX</sub> = 5.0 MHz) <sup>Note 3</sup>			28.7	mA
Write current (V <sub>PP</sub> pin) <sup>Note 1</sup>	I <sub>PPW</sub>	When V <sub>PP</sub> = V <sub>PP1</sub>	5.0 MHz crystal oscillation operation mode (f <sub>XX</sub> = 2.5 MHz) <sup>Note 2</sup>			19.5	mA
			5.0 MHz crystal oscillation operation mode (f <sub>XX</sub> = 5.0 MHz) <sup>Note 3</sup>			32.7	mA
Erase current (V <sub>DD</sub> pin) <sup>Note 1</sup>	I <sub>DDE</sub>	When V <sub>PP</sub> = V <sub>PP1</sub>	5.0 MHz crystal oscillation operation mode (f <sub>XX</sub> = 2.5 MHz) <sup>Note 2</sup>			15.5	mA
			5.0 MHz crystal oscillation operation mode (f <sub>XX</sub> = 5.0 MHz) <sup>Note 3</sup>			28.7	mA
Erase current (V <sub>PP</sub> pin) <sup>Note 1</sup>	I <sub>PPE</sub>	When V <sub>PP</sub> = V <sub>PP1</sub>				100	mA
Unit erase time	t <sub>ER</sub>			0.5	1	1	s
Total erase time	t <sub>ERA</sub>					20	s
Number of overwrites	C <sub>WRT</sub>	Erase and write are counted as one cycle				20	times
V <sub>PP</sub> power supply voltage	V <sub>PP0</sub>	In normal mode		0		0.2 V <sub>DD</sub>	V
	V <sub>PP1</sub>	During flash memory programming		9.7	10.0	10.3	V

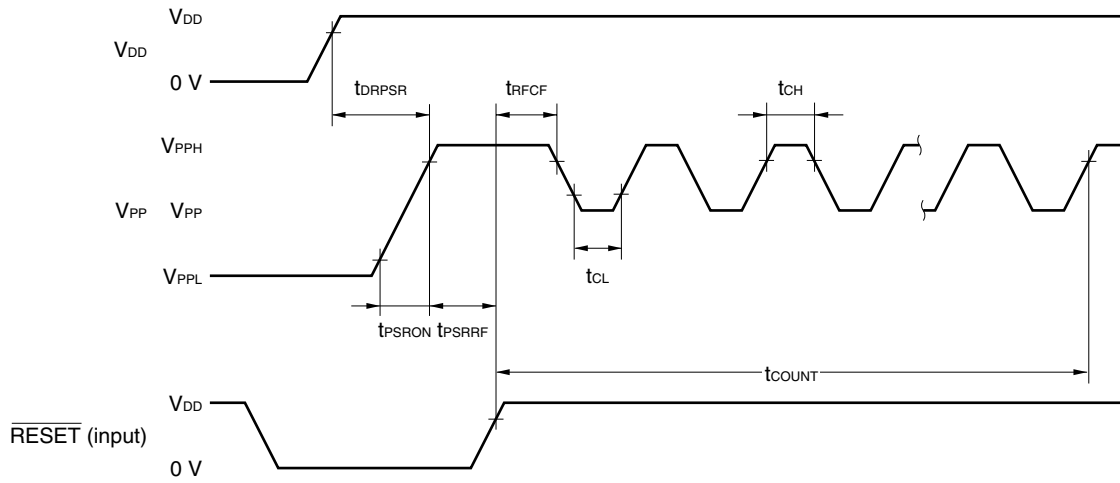
- Notes**
1. AV<sub>REF</sub> current and port current (current flowing to internal pull-up resistors) are not included.
  2. When main system clock is operating at f<sub>XX</sub> = f<sub>XX</sub>/2 (when oscillation mode select register (OSMS) is cleared to 00H).
  3. When main system clock is operating at f<sub>XX</sub> = f<sub>XX</sub> (when OSMS is set to 01H).

## 2) Serial write operation characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
V <sub>PP</sub> setup time	t <sub>PSRON</sub>	V <sub>PP</sub> high voltage	1.0			μs
V <sub>PP</sub> ↑ setup time from V <sub>DD</sub> ↑	t <sub>DRPSR</sub>	V <sub>PP</sub> high voltage	10			μs
RESET↑ setup time from V <sub>PP</sub> ↑	t <sub>PSRRF</sub>	V <sub>PP</sub> high voltage	1.0			μs
V <sub>PP</sub> count start time from RESET↑	t <sub>RFCF</sub>		1.0			μs
Count execution time	t <sub>COUNT</sub>				2.0	ms
V <sub>PP</sub> counter high-level width	t <sub>CH</sub>		8.0			μs
V <sub>PP</sub> counter low-level width	t <sub>CL</sub>		8.0			μs
V <sub>PP</sub> counter noise elimination width	t <sub>NFW</sub>			40		ns



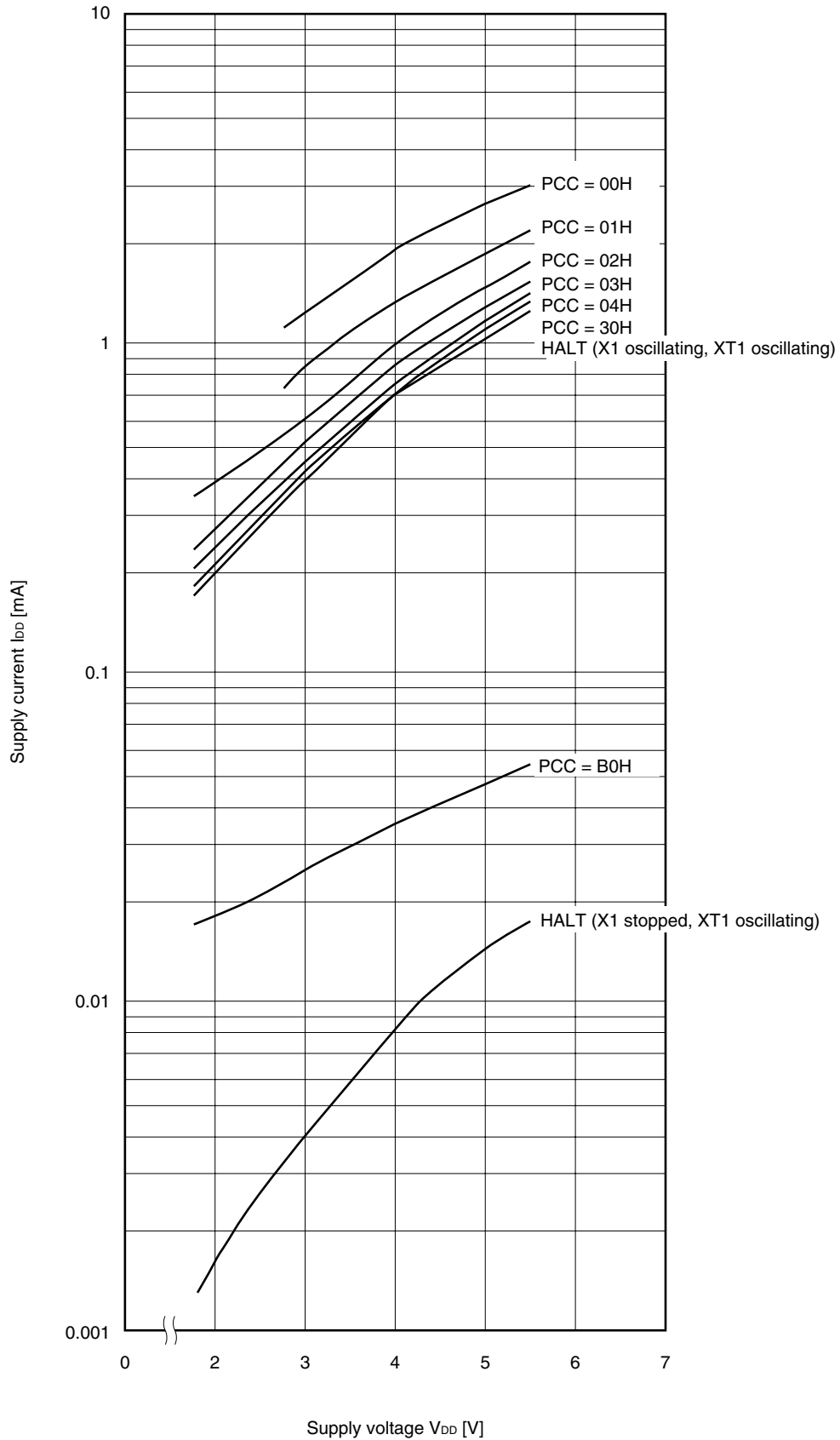
Flash Write Mode Setting Timing



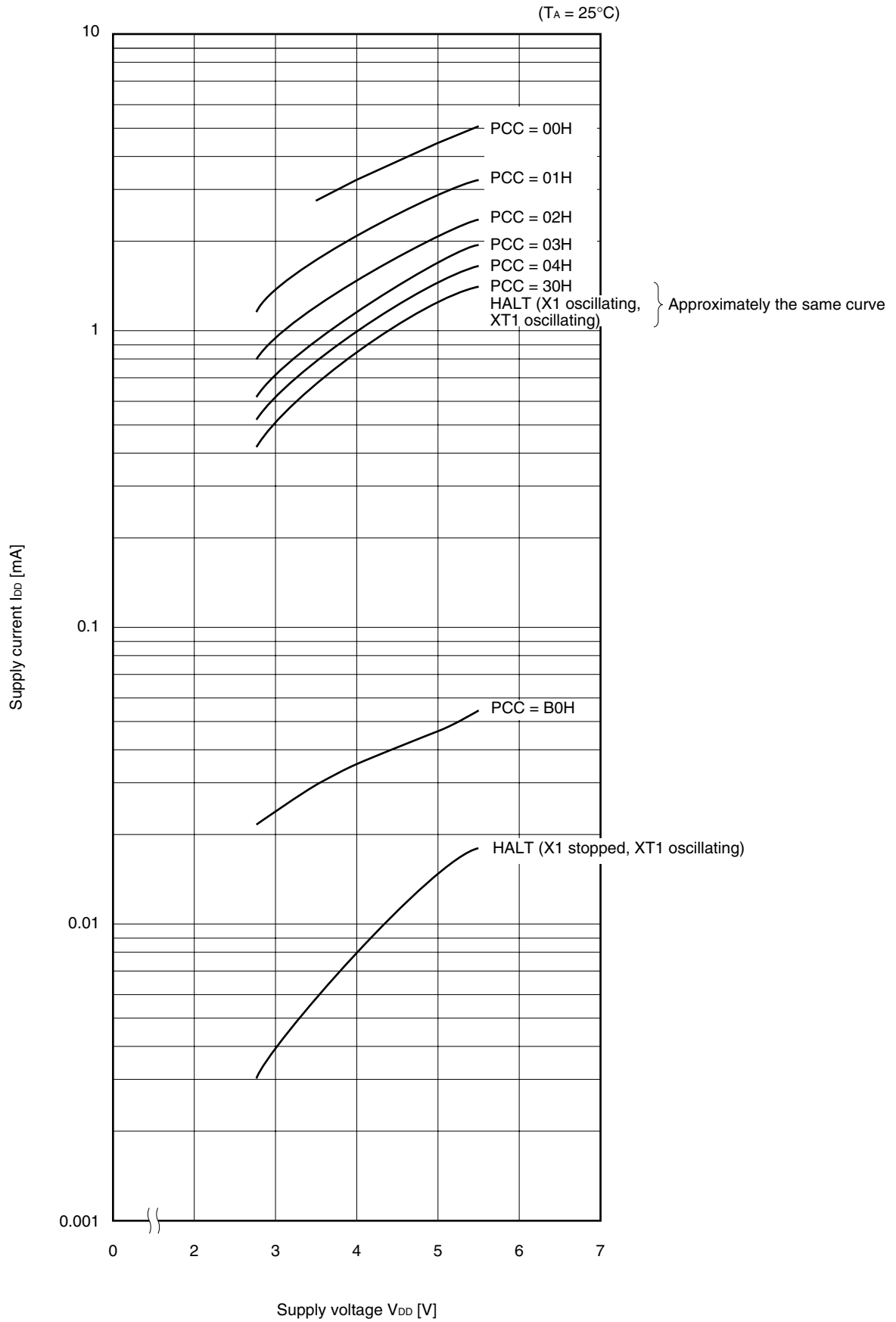
## CHAPTER 31 CHARACTERISTICS CURVES (REFERENCE VALUES)

$V_{DD}$  vs  $I_{DD}$  (mask ROM version,  $f_x = 5.0$  MHz,  $f_{xx} = 2.5$  MHz)

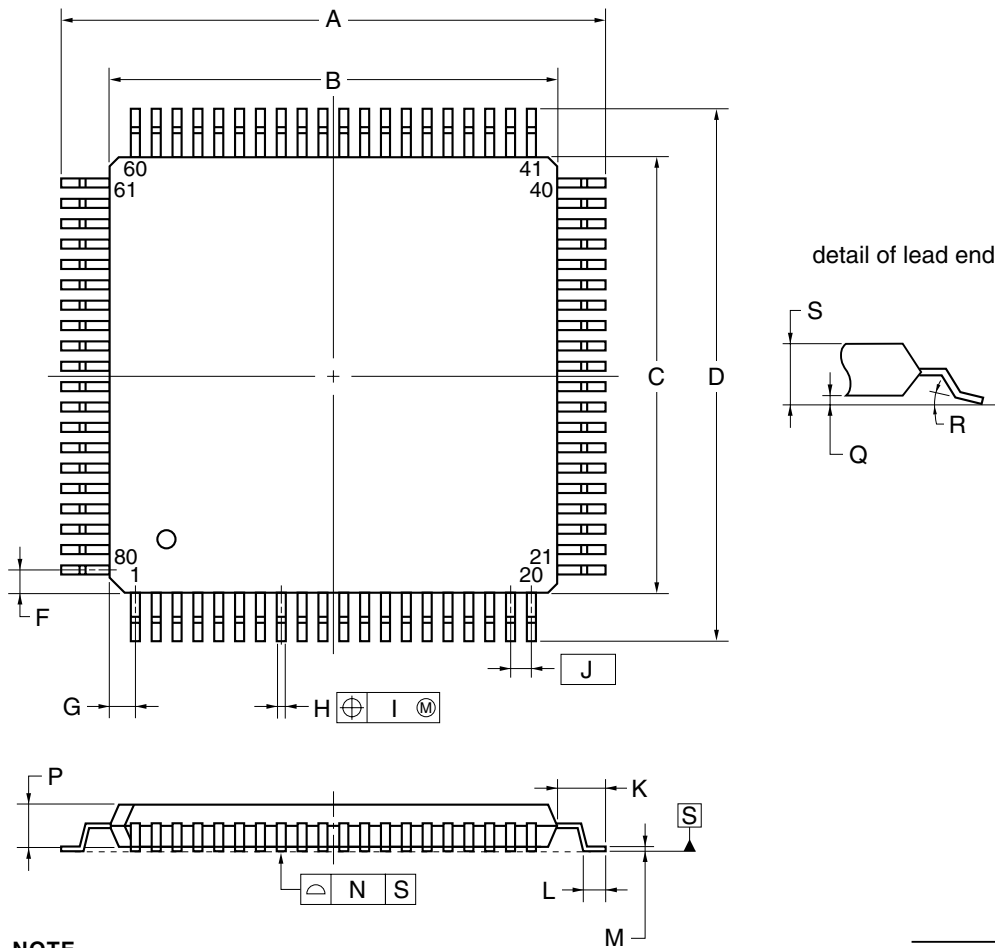
( $T_A = 25^\circ\text{C}$ )



V<sub>DD</sub> vs I<sub>DD</sub> (mask ROM version, f<sub>x</sub> = f<sub>xx</sub> = 5.0 MHz)



80-PIN PLASTIC QFP (14x14)



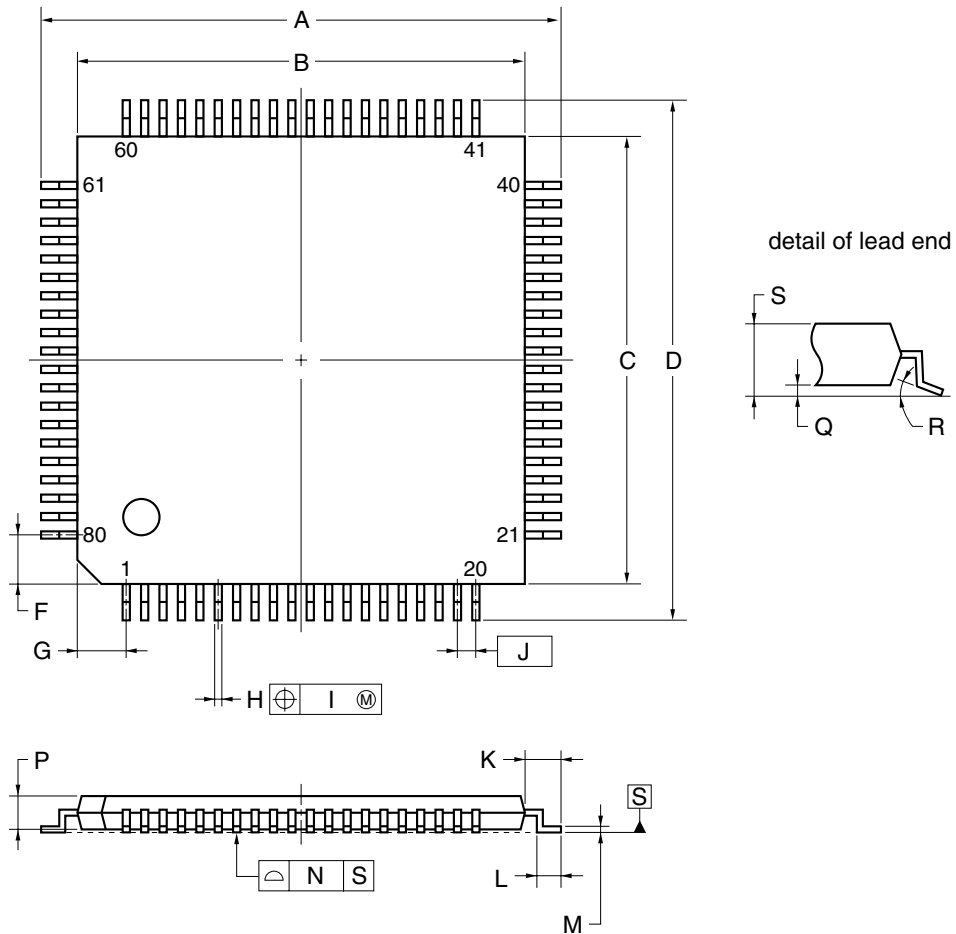
**NOTE**

Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	17.20±0.20
B	14.00±0.20
C	14.00±0.20
D	17.20±0.20
F	0.825
G	0.825
H	0.32±0.06
I	0.13
J	0.65 (T.P.)
K	1.60±0.20
L	0.80±0.20
M	0.17 <sup>+0.03</sup> <sub>-0.07</sub>
N	0.10
P	1.40±0.10
Q	0.125±0.075
R	3° <sup>+7°</sup> <sub>-3°</sub>
S	1.70 MAX.

**P80GC-65-8BT-1**

80-PIN PLASTIC TQFP (FINE PITCH) (12x12)



**NOTE**

Each lead centerline is located within 0.10 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	14.0±0.2
B	12.0±0.2
C	12.0±0.2
D	14.0±0.2
F	1.25
G	1.25
H	0.22±0.05
I	0.10
J	0.5 (T.P.)
K	1.0±0.2
L	0.5±0.2
M	0.145±0.05
N	0.10
P	1.0±0.05
Q	0.1±0.05
R	3° <sup>+7°</sup> <sub>-3°</sub>
S	1.2 MAX.

S80GK-50-9EU-1

The  $\mu$ PD780058 and 780058Y Subseries should be soldered and mounted under the following recommended conditions.

For details of the recommended soldering conditions, refer to the document **Semiconductor Device Mounting Technology Manual (C10535E)**.

For soldering methods and conditions other than those recommended below, contact an NEC Electronics sales representative.

**Table 33-1. Surface Mounting Type Soldering Conditions (1/4)**

(1) $\mu$ PD780053GC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780054GC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780055GC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780056GC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780058GC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780058BGC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780053YGC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780054YGC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780055YGC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780056YGC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780058BYGC-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780053GC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780054GC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780055GC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780056GC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780058BGC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780053YGC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780054YGC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780055YGC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780056YGC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)
$\mu$ PD780058BYGC(A)-xxx-8BT:	80-pin plastic QFP (14 × 14)

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Twice or less	IR35-00-2
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Twice or less	VP15-00-2
Wave soldering	Soldering bath temperature: 260°C or less, Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature)	WS60-00-1
Partial heating	Pin temperature: 300°C or less, Time: 3 seconds max. (per pin row)	—

**Caution** Do not use different soldering methods together (except for partial heating).

**Table 33-1. Surface Mounting Type Soldering Conditions (2/4)**(2)  $\mu$ PD78F0058GC-8BT: 80-pin plastic QFP (14 × 14) $\mu$ PD78F0058YGC-8BT: 80-pin plastic QFP (14 × 14)

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Twice or less, Exposure limit: 7 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	IR35-107-2
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Twice or less, Exposure limit: 7 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	VP15-107-2
Wave soldering	Soldering bath temperature: 260°C or less, Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature), Exposure limit: 7 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	WS60-107-1
Partial heating	Pin temperature: 300°C or less, Time: 3 seconds max. (per pin row)	–

**Note** After opening the dry pack, store it below 25°C and 65% RH for the allowable storage period.

**Caution** Do not use different soldering methods together (except for partial heating).

**Table 33-1. Surface Mounting Type Soldering Conditions (3/4)**

- (3)  $\mu$ PD780053GK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780054GK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780055GK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780056GK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780058GK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780058BGK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780053YGK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780054YGK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780055YGK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780056YGK-xxx-9EU: 80-pin plastic TQFP (12 × 12)
- $\mu$ PD780058BYGK-xxx-9EU: 80-pin plastic TQFP (12 × 12)

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Twice or less, Exposure limit: 7 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	IR35-107-2
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Twice or less, Exposure limit: 7 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	VP15-107-2
Wave soldering	–	–
Partial heating	Pin temperature: 300°C or less, Time: 3 seconds max. (per pin row)	–

**Note** After opening the dry pack, store it below 25°C and 65% RH for the allowable storage period.

**Caution** Do not use different soldering methods together (except for partial heating).



**Table 33-1. Surface Mounting Type Soldering Conditions (4/4)**

- (4)  $\mu$ PD78F0058GK-9EU: 80-pin plastic TQFP (12 × 12)  
 $\mu$ PD78F0058YGK-9EU: 80-pin plastic TQFP (12 × 12)

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Twice or less, Exposure limit: 3 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	IR35-103-2
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Twice or less, Exposure limit: 3 days <sup>Note</sup> (after that, prebake at 125°C for 10 hours)	VP15-103-2
Wave soldering	–	–
Partial heating	Pin temperature: 300°C or less, Time: 3 seconds max. (per pin row)	–

**Note** After opening the dry pack, store it below 25°C and 65% RH for the allowable storage period.

**Caution** Do not use different soldering methods together (except for partial heating).

## APPENDIX A DIFFERENCES BETWEEN $\mu$ PD78054, 78058F, AND 780058 SUBSERIES

Table A-1 shows the major differences between the  $\mu$ PD78054, 78058F, and 780058 Subseries.

**Table A-1. Major Differences Between  $\mu$ PD78054, 78058F, and 780058 Subseries (1/2)**

Product Name Item	$\mu$ PD78054 Subseries	$\mu$ PD78058F Subseries	$\mu$ PD780058 Subseries
EMI noise measures	None	Provided	Provided
Supply voltage	$V_{DD} = 2.0$ to $6.0$ V	$V_{DD} = 2.7$ to $6.0$ V	$V_{DD} = 1.8$ to $5.5$ V <sup>Note</sup>
PROM version	$\mu$ PD78P054, 78P058	$\mu$ PD78P058F	None
Flash memory version	None	None	$\mu$ PD78F0058
Internal ROM size	$\mu$ PD78052: 16 KB $\mu$ PD78053: 24 KB $\mu$ PD78054: 32 KB $\mu$ PD78P054: 32 KB $\mu$ PD78056: 48 KB $\mu$ PD78058: 60 KB $\mu$ PD78P058: 60 KB	$\mu$ PD78056F: 48 KB $\mu$ PD78058F: 60 KB $\mu$ PD78P058F: 60 KB	$\mu$ PD780053: 24 KB $\mu$ PD780054: 32 KB $\mu$ PD780055: 40 KB $\mu$ PD780056: 48 KB $\mu$ PD780058B: 60 KB $\mu$ PD780058: 60 KB $\mu$ PD78F0058: 60 KB
Internal high-speed RAM size	$\mu$ PD78052: 512 bytes $\mu$ PD78053, 78054, 78P054, 78056, 78058, 78P058: 1,024 bytes	1,024 bytes	1,024 bytes
I/O ports	Total: 69 pins • CMOS input: 2 pins • CMOS I/O: 63 pins • N-ch open-drain I/O: 4 pins		Total: 68 pins • CMOS input: 2 pins • CMOS I/O: 62 pins • N-ch open-drain I/O: 4 pins
$AV_{DD}$ pin	Power supply for A/D converter	Power supply for A/D converter and port output buffer	None (power supplied to port output buffer is $V_{DD0}$ )
$AV_{REF0}$ pin	Reference voltage input to A/D converter		Reference voltage input and analog power supply to A/D converter
★ Caution on operation immediately after A/D conversion starts	–	–	The results of the first A/D conversion immediately after the A/D conversion operation has started (CS set to 1) may not satisfy the ratings; therefore take appropriate measures such as discarding the results.
Serial interface channel 2	3-wire serial I/O/UART mode		3-wire serial I/O/UART mode with time division function
External maskable interrupts	7 sources		6 sources
★ Emulation probe	EP-78230GC-R, EP-78054GK-R		NP-80GC, NP-80GK, EP-78230GC-R, EP-78054GK-R
Device file	DF78054		DF780058

★ **Note**  $V_{DD}$  of flash memory version ( $\mu$ PD78F0058) = 2.7 to 5.5 V

**Table A-1. Major Differences Between  $\mu$ PD78054, 78058F, and 780058 Subseries (2/2)**

Item \ Product Name	$\mu$ PD78054 Subseries	$\mu$ PD78058F Subseries	$\mu$ PD780058 Subseries
★ Package	<ul style="list-style-type: none"> <li>• 80-pin plastic QFP (14 × 14)</li> <li>• 80-pin plastic QFP (14 × 14)</li> <li>• 80-pin ceramic WQFN (14 × 14) (<math>\mu</math>PD78P054, 78P058 only)</li> </ul>	<ul style="list-style-type: none"> <li>• 80-pin plastic QFP (14 × 14)</li> <li>• 80-pin plastic QFP (14 × 14)</li> <li>• 80-pin plastic TQFP (Fine pitch) (12 × 12) (<math>\mu</math>PD78058F only)</li> </ul>	<ul style="list-style-type: none"> <li>• 80-pin plastic QFP (14 × 14)</li> <li>• 80-pin plastic TQFP (Fine pitch) (12 × 12)</li> </ul>
★ Electrical specifications and recommended soldering conditions	Refer to data sheet of individual product.		See <b>CHAPTERS 28 to 30 ELECTRICAL SPECIFICATIONS</b> and <b>CHAPTER 33 RECOMMENDED SOLDERING CONDITIONS.</b>

## APPENDIX B DEVELOPMENT TOOLS

The following development tools are available for the development of systems which employ the  $\mu$ PD780058, 780058Y Subseries.

Figure B-1 shows a configuration example of the tools.

- **Support for PC98-NX series**

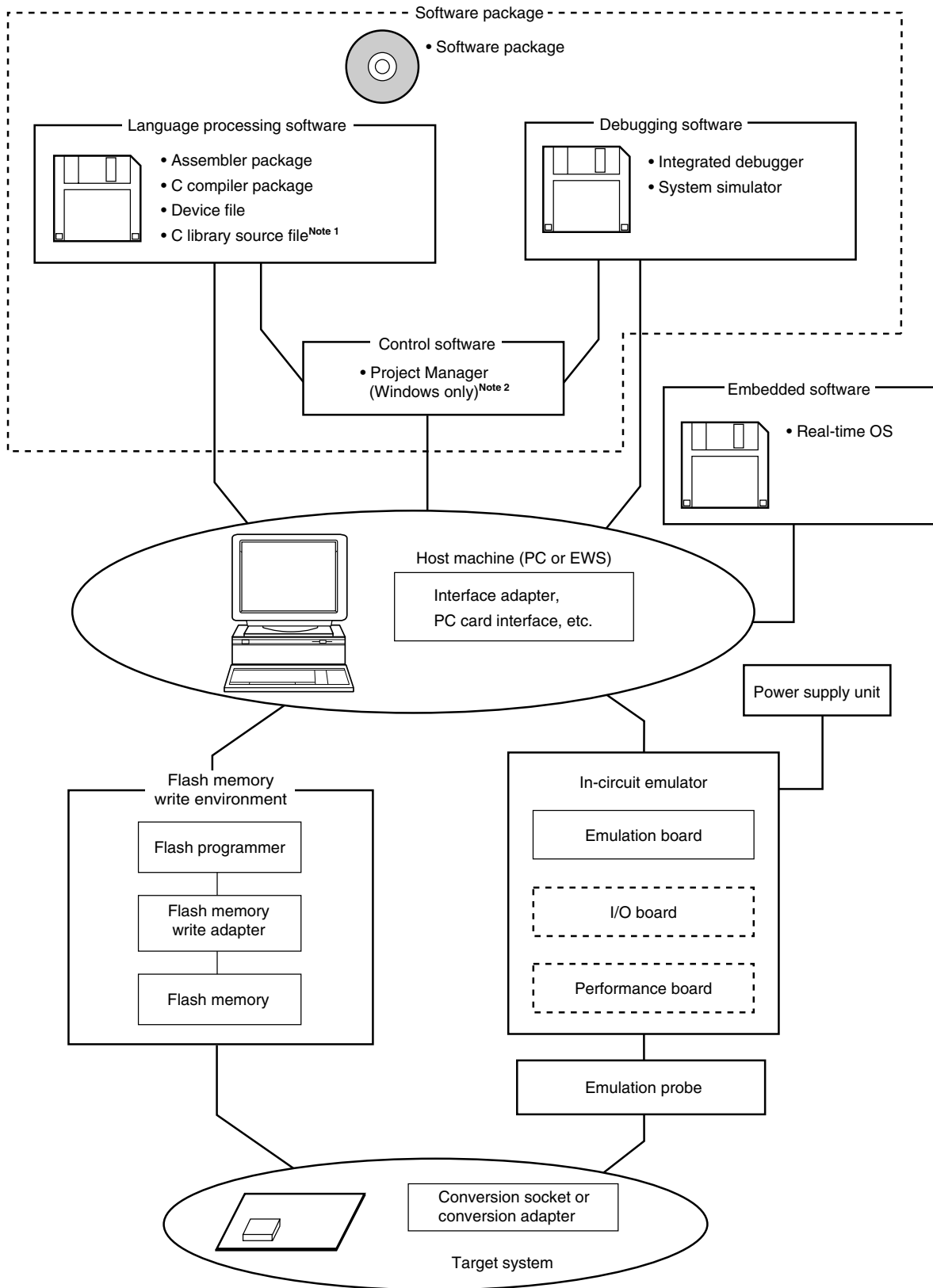
Unless otherwise specified, products supported by IBM PC/AT™ compatible machines can be used for PC98-NX series computers. When using PC98-NX series computers, refer to the description for IBM PC/AT compatible machines.

- **Windows**

Unless otherwise specified, "Windows" means the following OSs.

- Windows 3.1
- Windows 95
- Windows 98
- Windows 2000
- Windows NT™ Ver. 4.0

Figure B-1. Configuration of Development Tools



- Notes**
1. The C library source file is not included in the software package.
  2. The Project Manager is included in the assembler package.  
The Project Manager is only used for Windows.

**B.1 Software Package**

SP78K0 Software package	This package contains various software tools for 78K/0 Series development. The following tools are included. RA78K0, CC78K0, ID78K0-NS, SM78K0, and various device files
	Part Number: $\mu$ SxxxxSP78K0

**Remark** xxxx in the part number differs depending on the OS used.

$\mu$ SxxxxSP78K0

xxxx	Host Machine	OS	Supply Medium
AB17	PC-9800 series,	Windows (Japanese version)	CD-ROM
BB17	IBM PC/AT compatibles	Windows (English version)	

**B.2 Language Processing Software**

RA78K0 Assembler package	<p>This assembler converts programs written in mnemonics into object codes executable with a microcontroller.</p> <p>Further, this assembler is provided with functions capable of automatically creating symbol tables and branch instruction optimization.</p> <p>This assembler should be used in combination with a device file (DF780058) (sold separately).</p> <p><b>&lt;Precaution when using RA78K0 in PC environment&gt;</b> This assembler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (included in assembler package) in Windows.</p> <p>Part Number: <math>\mu</math>SxxxxRA78K0</p>
CC78K0 C compiler package	<p>This compiler converts programs written in C language into object codes executable with a microcontroller.</p> <p>This compiler should be used in combination with an assembler package and device file (both sold separately).</p> <p><b>&lt;Precaution when using CC78K0 in PC environment&gt;</b> This C compiler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (included in assembler package) in Windows.</p> <p>Part Number: <math>\mu</math>SxxxxCC78K0</p>
DF780058 <sup>Note 1</sup> Device file	<p>This file contains information peculiar to the device.</p> <p>This device file should be used in combination with tools (RA78K0, CC78K0, SM78K0, ID78K0-NS, ID78K0, and RX78K0) (sold separately).</p> <p>The corresponding OS and host machine differ depending on the tool used.</p> <p>Part Number: <math>\mu</math>SxxxxDF780058</p>
CC78K0-L <sup>Note 2</sup> C library source file	<p>This is a source file of functions configuring the object library included in the C compiler package.</p> <p>This file is required to match the object library included in C compiler package to the user's specifications.</p> <p>It does not depend on the operating environment because it is a source file.</p> <p>Part Number: <math>\mu</math>SxxxxCC78K0-L</p>

- Notes**
1. The DF780058 can be used in common with the RA78K0, CC78K0, SM78K0, ID78K0-NS, ID78K0, and RX78K0.
  2. CC78K0-L is not included in the software package (SP78K0).

**Remark** xxxx in the part number differs depending on the host machine and OS used.

μSxxxxRA78K0

μSxxxxCC78K0

xxxx	Host Machine	OS	Supply Medium
AB13	PC-9800 series,	Windows (Japanese version)	3.5-inch 2HD FD
BB13	IBM PC/AT and compatibles	Windows (English version)	
AB17			Windows (Japanese version)
BB17	Windows (English version)		
3P17	HP9000 series 700™	HP-UX™ (Rel. 10.10)	
3K17	SPARCstation™	SunOS™ (Rel. 4.1.4), Solaris™ (Rel. 2.5.1)	

μSxxxxDF780058

μSxxxxCC78K0-L

xxxx	Host Machine	OS	Supply Medium
AB13	PC-9800 series,	Windows (Japanese version)	3.5-inch 2HD FD
BB13	IBM PC/AT and compatibles	Windows (English version)	
3P16	HP9000 series 700	HP-UX (Rel. 10.10)	DAT
3K13	SPARCstation	SunOS (Rel. 4.1.4),	3.5-inch 2HD FD
3K15		Solaris (Rel. 2.5.1)	1/4-inch CGMT

### B.3 Control Software

Project Manager	<p>This is control software designed to enable efficient user program development in the Windows environment. All operations used in development of a user program, such as starting the editor, building, and starting the debugger, can be performed from the Project Manager.</p> <p><b>&lt;Caution&gt;</b> The Project Manager is included in the assembler package (RA78K0). It can only be used in Windows.</p>
-----------------	---

### B.4 Flash Memory Writing Tools

<p>Flashpro III (Part number: FL-PR3, PG-FP3) Flashpro IV (Part number: FL-PR4, PG-FP4) Flash programmer</p>	Flash programmer dedicated to microcontrollers with on-chip flash memory.
<p>FA-80GC-8BT FA-80GK-9EU Flash memory writing adapter</p>	<p>Flash memory writing adapter used connected to Flashpro III/Flashpro IV.</p> <ul style="list-style-type: none"> <li>FA-80GC-8BT: 80-pin plastic QFP (GC-8BT type)</li> <li>FA-80GK-9EU: 80-pin plastic TQFP (GK-9EU type)</li> </ul>

**Remark** FL-PR3, FL-PR4, FA-80GC-8EU, and FA-80GK-9EU are products of Naito Densai Machida Mfg. Co., Ltd.

Contact: +81-45-475-4191 Naito Densai Machida Mfg. Co., Ltd.

## B.5 Debugging Tools (Hardware)

### B.5.1 When using in-circuit emulator IE-78K0-NS, IE-78K0-NS-A

IE-78K0-NS In-circuit emulator	The in-circuit emulator serves to debug hardware and software when developing application systems using a 78K/0 Series product. It corresponds to an integrated debugger (ID78K0-NS). This emulator should be used in combination with a power supply unit, emulation probe, and interface adapter which is required to connect this emulator to the host machine.
IE-78K0-NS-PA Performance board	This board is used for extending the IE-78K0-NS functions, and is used connected to the IE-78K0-NS. With the addition of this board, the addition of a coverage function, enhancement of tracer and timer functions, and other such debugging function enhancements are possible.
IE-78K0-NS-A In-circuit emulator	In-circuit emulator that combines the IE-78K0-NS and IE-78K0-NS-PA
IE-70000-MC-PS-B Power supply unit	This adapter is used for supplying power from a 100 to 240 V AC output.
IE-70000-98-IF-C Interface adapter	This adapter is required when using a PC-9800 series computer (except notebook type) as the IE-78K0-NS host machine (C bus compatible).
IE-70000-CD-IF-A PC card interface	This is PC card and interface cable required when using a notebook-type computer as the IE-78K0-NS host machine (PCMCIA socket compatible).
IE-70000-PC-IF-C Interface adapter	This adapter is required when using an IBM PC/AT compatible computer as the IE-78K0-NS host machine (ISA bus compatible).
IE-70000-PCI-IF-A Interface adapter	This adapter is required when using a PC with a PCI bus as the IE-78K0-NS host machine.
IE-780308-NS-EM1 Emulation board	This board emulates the operations of the peripheral hardware peculiar to a device (common to $\mu$ PD780308 subseries). It should be used in combination with an in-circuit emulator.
NP-80GC-TQ NP-H80GC-TQ Emulation probe	This probe is used to connect the in-circuit emulator to the target system and is designed for an 80-pin plastic QFP (GC-8BT type). It should be used in combination with the TGC-080SBP.
	TGC-080SBP Conversion adapter (See <b>Figure B-2</b> )
NP-80GC Emulation Probe	This probe is for an 80-pin plastic QFP (GC-8BT type) and connects an in-circuit emulator and the target system.
	EV-9200GC-80 Conversion Socket (See <b>Figure B-2</b> )
NP-80GK Emulation Probe	This probe is for an 80-pin plastic TQFP (GK-9EU type) and connects an in-circuit emulator and the target system.
	TGK-080SDW Conversion Adapter (See <b>Figure B-3</b> )

**Remarks 1.** NP-80GC, NP-80GC-TQ, NP-H80GC-TQ, and NP-80GK are products of Naito Densai Machida Mfg. Co., Ltd.

Contact: +81-45-475-4191 Naito Densai Machida Mfg. Co., Ltd.

**2.** TGC-080SBP and TGK-080SDW are products of TOKYO ELETECH CORPORATION.

Inquiry: Daimaru Kogyo, Ltd. Phone: Tokyo +81-3-3820-7112 Electronics Dept.

Osaka +81-6-6244-6672 Electronics 2nd Dept.



**B.5.2 When using in-circuit emulator IE-78001-R-A**

IE-78001-R-A In-circuit emulator		This is an in-circuit emulator for debugging the hardware and software when an application system using the 78K/0 Series is developed. It supports an integrated debugger (ID78K0). This emulator is used with an emulation probe and interface adapter for connecting a host machine.
IE-70000-98-IF-C Interface adapter		This adapter is necessary when a PC-9800 series PC (except notebook type) is used as the host machine for the IE-78001-R-A (C bus compatible).
IE-70000-PC-IF-C Interface adapter		This adapter is necessary when an IBM PC/AT or compatible machine is used as the host machine for the IE-78001-R-A (ISA bus compatible).
IE-780308-R-EM Emulation board		This board is used with an in-circuit emulator to emulate device-specific peripheral hardware.
EP-78230GC-R Emulation probe		This probe is for an 80-pin plastic QFP (GC-8BT type) and connects an in-circuit emulator and the target system.
	EV-9200GC-80 Conversion socket (See <b>Figure B-2</b> )	This conversion socket connects the board of the target system created to mount an 80-pin plastic QFP (GC-8BT type) and EP-78230GC-R.
EP-78054GK-R Emulation probe		This probe is for an 80-pin plastic TQFP (GK-9EU type) and connects an in-circuit emulator and the target system.
	TGK-080SDW Conversion adapter (See <b>Figure B-3</b> )	This conversion adapter connects the board of the target system created to mount an 80-pin plastic TQFP (GK-9EU type) and EP-78054GK-R.

**Remarks 1.** TGK-080SDW is a product of TOKYO ELETECH CORPORATION.

Inquiry: Daimaru Kogyo, Ltd. Phone: Tokyo +81-3-3820-7112 Electronics Dept.

Osaka +81-6-6244-6672 Electronics 2nd Dept.

**2.** The EV-9200GC-80 is sold in sets of five units.

**3.** The TGK-080SDW is sold in single units.

**B.6 Debugging Tools (Software)**

SM78K0 System simulator	This is a system simulator for the 78K/0 Series. The SM78K0 is Windows-based software. 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. Use of the SM78K0 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. The SM78K0 should be used in combination with a device file (DF780058) (sold separately). Part Number: $\mu$ SxxxxSM78K0
ID78K0-NS Integrated debugger (supporting in-circuit emulators IE-78K0-NS and IE-78K0-NS-A)	This debugger supports the in-circuit emulators for the 78K/0 Series. The ID78K0-NS is Windows-based software. 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 a device file (sold separately).
ID78K0 Integrated debugger (supporting in-circuit emulator IE-78001-R-A)	Part Number: $\mu$ SxxxxID78K0-NS $\mu$ SxxxxID78K0

**Remark** xxxx in the part number differs depending on the host machine and OS used.

$\mu$ SxxxxSM78K0  
 $\mu$ SxxxxID78K0-NS  
 $\mu$ SxxxxID78K0

xxxx	Host Machine	OS	Supply Medium
AB13	PC-9800 series, IBM PC/AT and compatibles	Windows (Japanese version)	3.5-inch 2HD FD
BB13		Windows (English version)	
AB17		Windows (Japanese version)	CD-ROM
BB17		Windows (English version)	

**B.7 Embedded Software**

RX78K0 Real-time OS	The RX78K0 is a real-time OS conforming to the $\mu$ ITRON specifications. A tool (configurator) for generating the nucleus of the RX78K0 and multiple information tables is supplied. Used in combination with an assembler package (RA78K0) and device file (DF780058) (both sold separately). <b>&lt;Precaution when using RX78K0 in PC environment&gt;</b> The real-time OS is a DOS-based application. It should be used in the DOS prompt when using in Windows. <hr/> Part number: $\mu$ SxxxxRX78013- $\Delta\Delta\Delta\Delta$
------------------------	---

**Caution** When purchasing the RX78K0, fill in the purchase application form in advance and sign the user agreement.

**Remark** xxxx and  $\Delta\Delta\Delta\Delta$  in the part number differ depending on the host machine and OS used.

$\mu$ SxxxxRX78013- $\Delta\Delta\Delta\Delta$

$\Delta\Delta\Delta\Delta$	Product Outline	Maximum Number for Use in Mass Production
001	Evaluation object	Do not use for mass-produced product.
100K	Mass-production object	0.1 million units
001M		1 million units
010M		10 million units
S01	Source program	Source program for mass-produced object

xxxx	Host Machine	OS	Supply Medium
AA13	PC-9800 series	Windows (Japanese version)	3.5-inch 2HD FD
AB13	IBM PC/AT and compatibles	Windows (Japanese version)	
BB13		Windows (English version)	

**B.8 System-Upgrade Method from Former In-Circuit Emulator for 78K/0 Series to IE-78001-R-A**

If you already have a former in-circuit emulator for 78K/0 Series microcontrollers (IE-78000-R or IE-78000-R-A), that in-circuit emulator can operate as an equivalent to the IE-78001-R-A by replacing its internal break board with the IE-78001-R-BK.

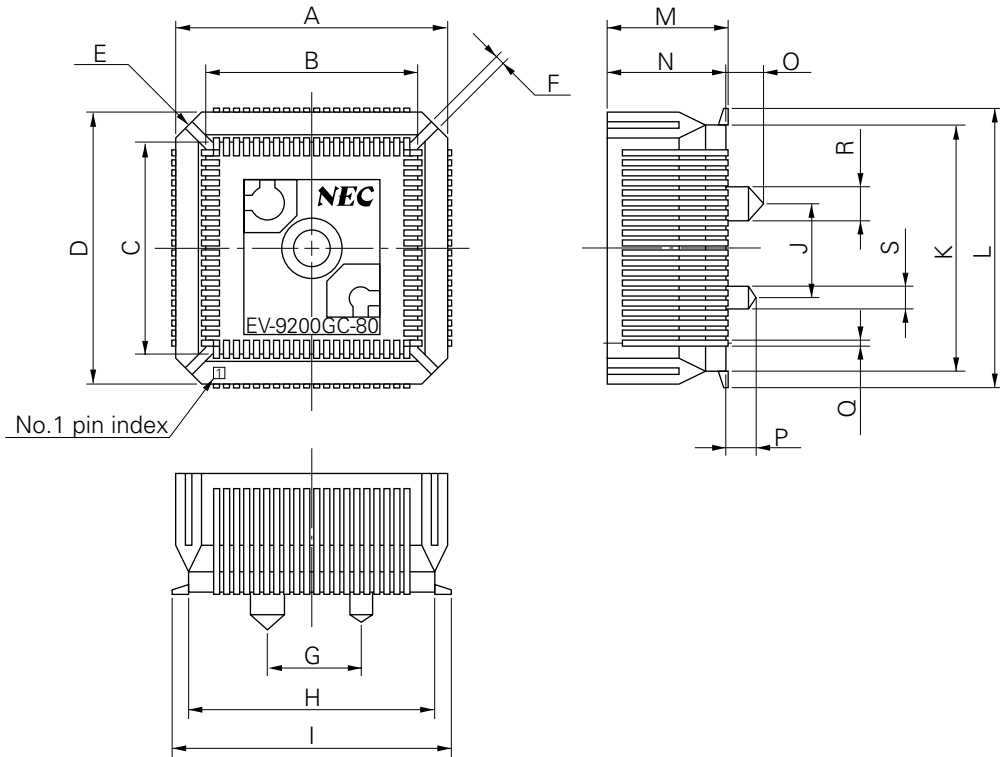
**Table B-1. System-Upgrade Method from Former In-Circuit Emulator for 78K/0 Series to IE-78001-R-A**

In-Circuit Emulator Owned	In-Circuit Emulator Cabinet System-Up <sup>Note</sup>	Board to Be Purchased
IE-78000-R	Required	IE-78001-R-BK
IE-78000-R-A	Not required	

**Note** For upgrading a cabinet, send your in-circuit emulator to NEC Electronics.

B.9 Drawing and Footprint for Conversion Socket (EV-9200GC-80)

Figure B-2. EV-9200GC-80 Drawing (For Reference Only)

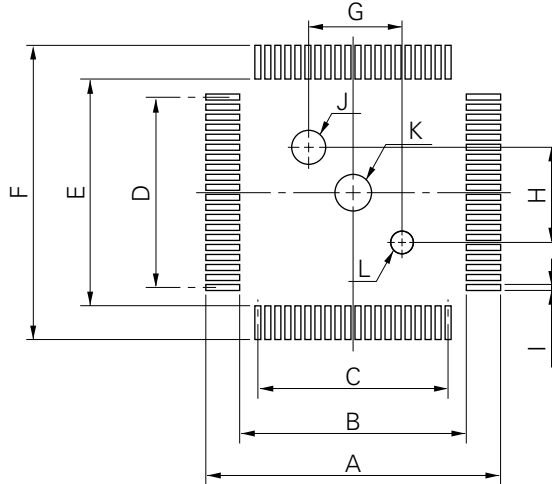


EV-9200GC-80-G0

ITEM	MILLIMETERS	INCHES
A	18.0	0.709
B	14.4	0.567
C	14.4	0.567
D	18.0	0.709
E	4-C 2.0	4-C 0.079
F	0.8	0.031
G	6.0	0.236
H	16.0	0.63
I	18.7	0.736
J	6.0	0.236
K	16.0	0.63
L	18.7	0.736
M	8.2	0.323
O	8.0	0.315
N	2.5	0.098
P	2.0	0.079
Q	0.35	0.014
R	φ2.3	φ0.091
S	φ1.5	φ0.059

Figure B-3. EV-9200GC-80 Footprint (For Reference Only)

Based on EV-9200GC-80  
(2) Pad drawing (in mm)



EV-9200GC-80-P1E

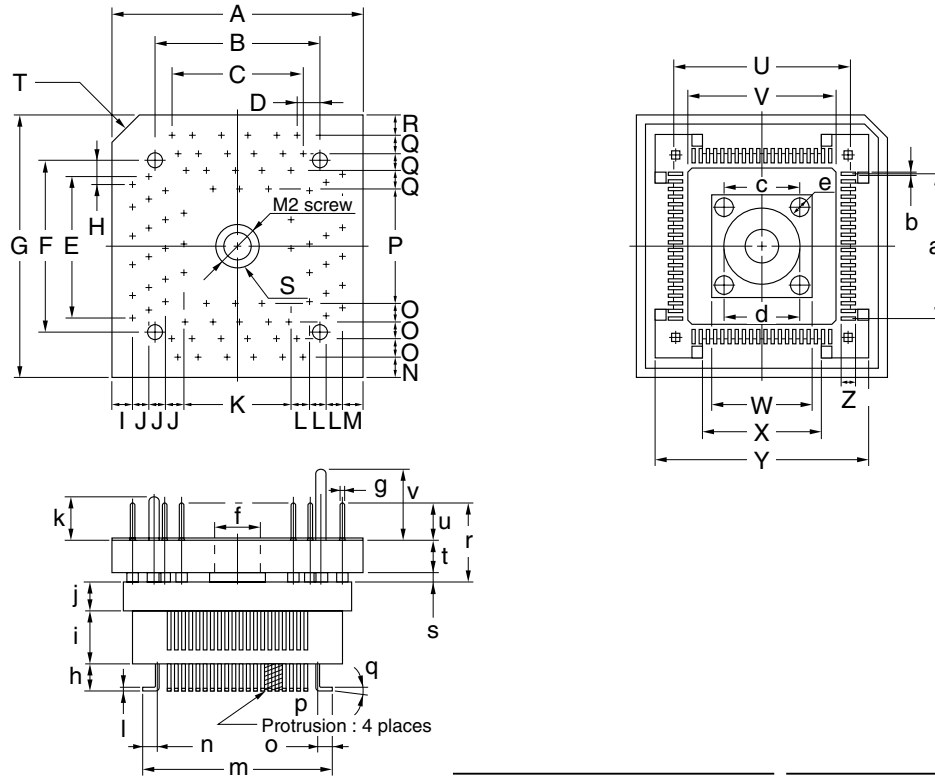
ITEM	MILLIMETERS	INCHES
A	19.7	0.776
B	15.0	0.591
C	$0.65 \pm 0.02 \times 19 = 12.35 \pm 0.05$	$0.026^{+0.001}_{-0.002} \times 0.748 = 0.486^{+0.003}_{-0.002}$
D	$0.65 \pm 0.02 \times 19 = 12.35 \pm 0.05$	$0.026^{+0.001}_{-0.002} \times 0.748 = 0.486^{+0.003}_{-0.002}$
E	15.0	0.591
F	19.7	0.776
G	$6.0 \pm 0.05$	$0.236^{+0.003}_{-0.002}$
H	$6.0 \pm 0.05$	$0.236^{+0.003}_{-0.002}$
I	$0.35 \pm 0.02$	$0.014^{+0.001}_{-0.001}$
J	$\phi 2.36 \pm 0.03$	$\phi 0.093^{+0.001}_{-0.002}$
K	$\phi 2.3$	$\phi 0.091$
L	$\phi 1.57 \pm 0.03$	$\phi 0.062^{+0.001}_{-0.002}$

**Caution** Dimensions of mount pad for EV-9200 and that for target device (QFP) may be different in some parts. For the recommended mount pad dimensions for QFP, refer to "SEMICONDUCTOR DEVICE MOUNTING TECHNOLOGY MANUAL" (C10535E).

B.10 Drawing of Conversion Adapter (TGK-080SDW, TGC-080SBP)

Figure B-4. TGK-080SDW Drawing (For Reference Only) (Unit: mm)

**TGK-080SDW (TQPACK080SD + TQSOCKET080SDW)**  
**Package dimension (unit: mm)**

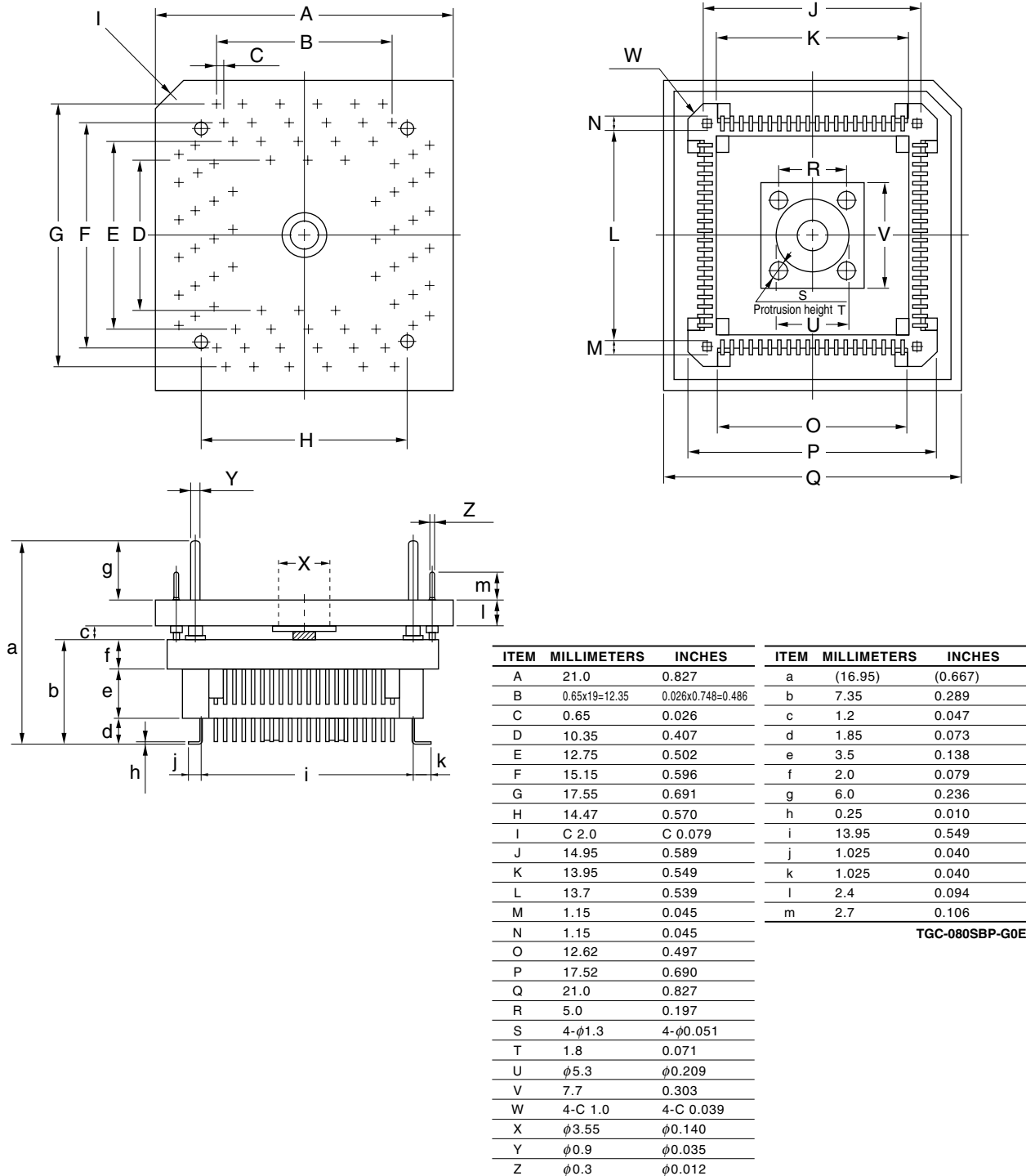


ITEM	MILLIMETERS	INCHES	ITEM	MILLIMETERS	INCHES
A	18.0	0.709	a	0.5x19=9.5±0.10	0.020x0.748-0.374±0.004
B	11.77	0.463	b	0.25	0.010
C	0.5x19=9.5	0.020x0.748=0.374	c	φ5.3	φ0.209
D	0.5	0.020	d	φ5.3	φ0.209
E	0.5x19=9.5	0.020x0.748=0.374	e	φ1.3	φ0.051
F	11.77	0.463	f	φ3.55	φ0.140
G	18.0	0.709	g	φ0.3	φ0.012
H	0.5	0.020	h	1.85±0.2	0.073±0.008
I	1.58	0.062	i	3.5	0.138
J	1.2	0.047	j	2.0	0.079
K	7.64	0.301	k	3.0	0.118
L	1.2	0.047	l	0.25	0.010
M	1.58	0.062	m	14.0	0.551
N	1.58	0.062	n	1.4±0.2	0.055±0.008
O	1.2	0.047	o	1.4±0.2	0.055±0.008
P	7.64	0.301	p	h=1.8 φ1.3	h=0.071 φ0.051
Q	1.2	0.047	q	0~5°	0.000~0.197°
R	1.58	0.062	r	5.9	0.232
S	φ3.55	φ0.140	s	0.8	0.031
T	C 2.0	C 0.079	t	2.4	0.094
U	12.31	0.485	u	2.7	0.106
V	10.17	0.400	v	3.9	0.154
W	6.8	0.268	<b>TGK-080SDW-G1E</b>		
X	8.24	0.324			
Y	14.8	0.583			
Z	1.4±0.2	0.055±0.008			

**note:** Product by TOKYO ELETECH CORPORATION.

Figure B-5. TGC-080SBP Drawing (For Reference Only) (Unit: mm)

Reference diagram: TGC-080SBP (TQPACK080SB+TQSOCKET080SBP)  
 Package dimension (unit: mm)



note: Product by TOKYO ELETECH CORPORATION.

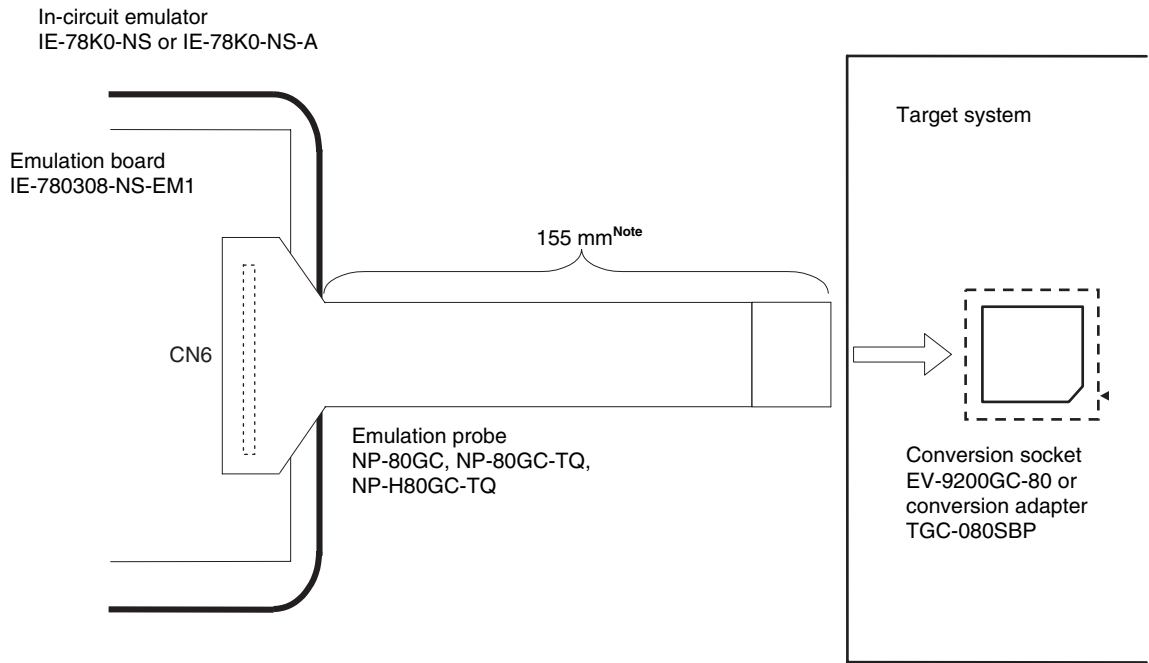


## ★ B.11 Cautions on Designing Target System

Figures B-6 to B-9 show the conditions when connecting the emulation probe to the conversion socket. Follow the configuration below and consider the shape of parts to be mounted on the target system when designing a system.

### (1) NP-80GC, NP-80GC-TQ, NP-H80GC-TQ

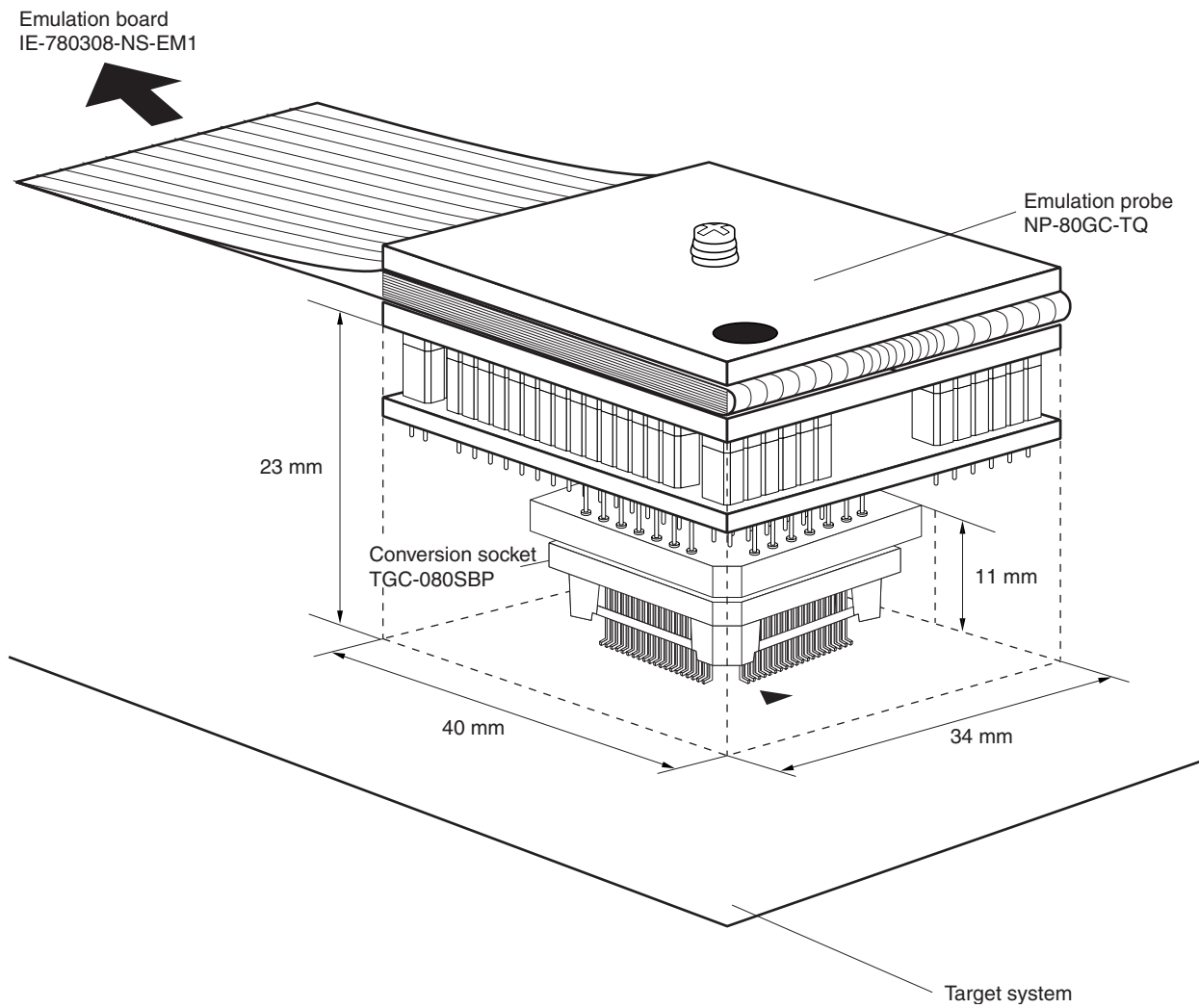
**Figure B-6. Distance Between In-Circuit Emulator and Conversion Socket (80GC)**



**Note** When NP-H80GC-TQ is used, the distance is 355 mm.

**Remark** NP-80GC, NP-80GC-TQ, and NP-H80GC-TQ are products of Naito Densetsu Machida Mfg. Co., Ltd. TGC-080SBP is a product of TOKYO ELETECH CORPORATION.

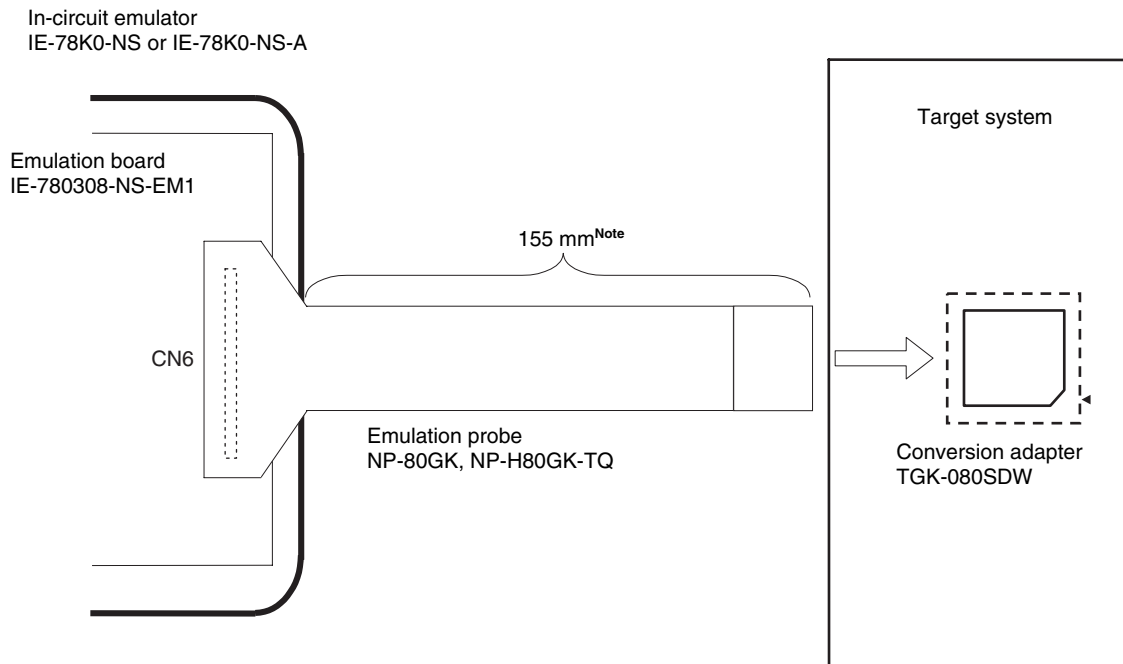
Figure B-7. Connection Condition of Target System (NP-80GC-TQ)



**Remark** NP-80GC-TQ is a product of Naito Densai Machida Mfg. Co., Ltd.  
TGC-080SBP is a product of TOKYO ELETECH CORPORATION.

## (2) NP-80GK, NP-H80GK-TQ

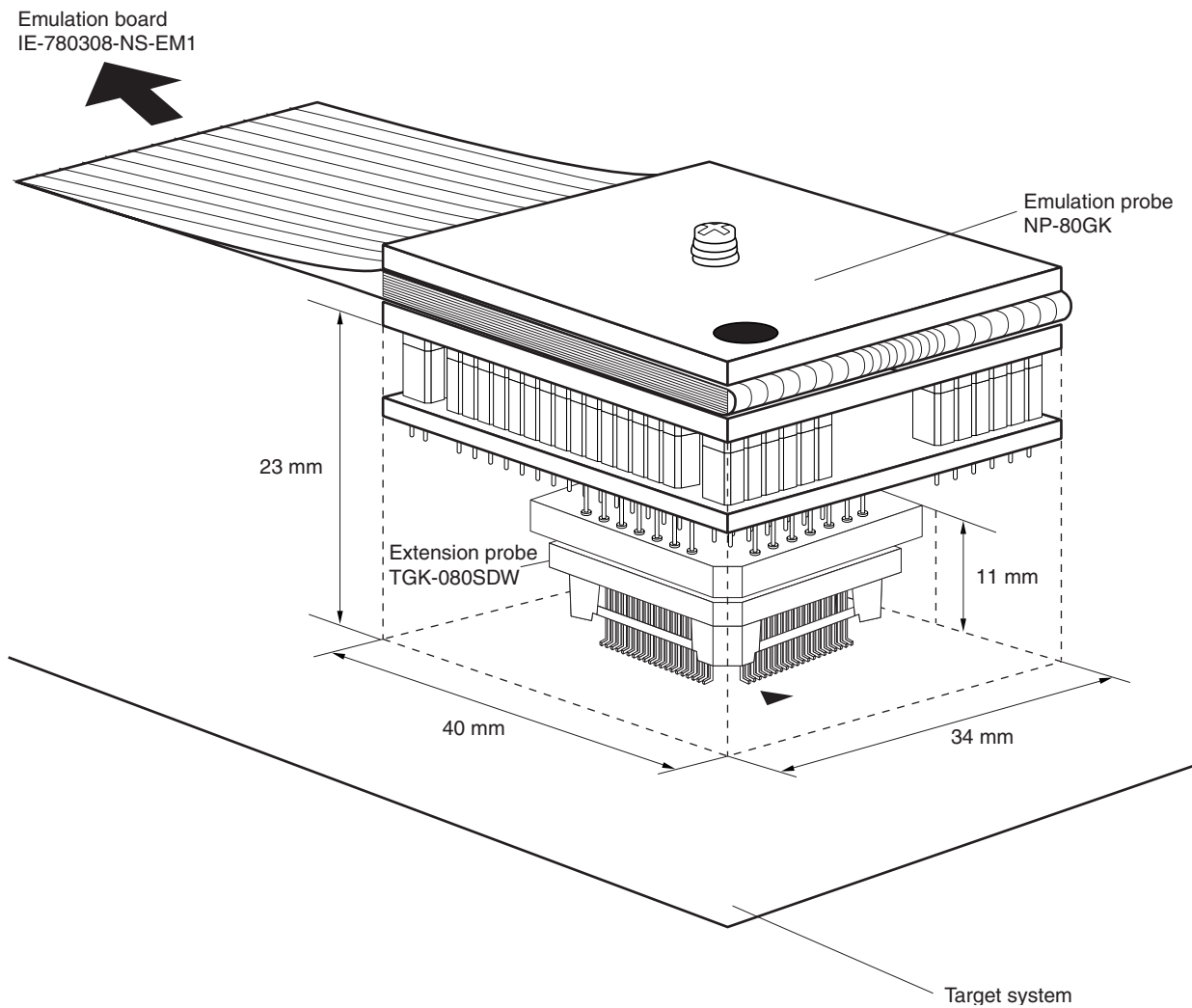
Figure B-8. Distance Between In-Circuit Emulator and Conversion Socket (80GK)



**Note** When NP-H80GK-TQ is used, the distance is 355 mm.

**Remark** NP-80GK and NP-H80GK-TQ are products of Naito Densai Machida Mfg. Co., Ltd.  
TGK-080SDW is a product of TOKYO ELETECH CORPORATION.

Figure B-9. Connection Condition of Target System (NP-80GK)



**Remark** NP-80GK is a product of Naito Densai Machida Mfg. Co., Ltd.  
TGK-080SDW is a product of TOKYO ELETECH CORPORATION.

## APPENDIX C REGISTER INDEX

### C.1 Register Index (Register Name)

16-bit timer mode control register (TMC0) .....	171
16-bit timer output control register (TOC0) .....	174
16-bit timer register (TM0) .....	168
8-bit timer mode control register (TMC1) .....	216
8-bit timer output control register (TOC1) .....	217
8-bit timer register 1 (TM1) .....	213
8-bit timer register 2 (TM2) .....	213
[A]	
A/D conversion result register (ADCR) .....	256
A/D converter input select register (ADIS) .....	260
A/D converter mode register (ADM) .....	258
Asynchronous serial interface mode register (ASIM) .....	433
Asynchronous serial interface status register (ASIS) .....	436
Automatic data transmit/receive address pointer (ADTP) .....	385
Automatic data transmit/receive control register (ADTC) .....	389
Automatic data transmit/receive interval specification register (ADTI) .....	390
[B]	
Baud rate generator control register (BRGC) .....	437
[C]	
Capture/compare control register 0 (CRC0) .....	173
Capture/compare register 00 (CR00) .....	167
Capture/compare register 01 (CR01) .....	168
Compare register 10 (CR10) .....	213
Compare register 20 (CR20) .....	213
Correction address register 0 (CORAD0) .....	526
Correction address register 1 (CORAD1) .....	526
Correction control register (CORCN) .....	527
[D]	
D/A conversion value set register 0 (DACS0) .....	277
D/A conversion value set register 1 (DACS1) .....	277
D/A converter mode register (DAM) .....	278
[E]	
External interrupt mode register 0 (INTM0) .....	177, 483
External interrupt mode register 1 (INTM1) .....	261, 483
[I]	
Internal expansion RAM size switching register (IXS) .....	537
Internal memory size switching register (IMS) .....	506, 536

Interrupt mask flag register 0H (MK0H) ..... 481  
 Interrupt mask flag register 0L (MK0L) ..... 481  
 Interrupt mask flag register 1L (MK1L) ..... 481, 499  
 Interrupt request flag register 0H (IF0H) ..... 480  
 Interrupt request flag register 0L (IF0L) ..... 480  
 Interrupt request flag register 1L (IF1L) ..... 480, 499  
 Interrupt timing specification register (SINT) ..... 294, 345

[K]

Key return mode register (KRM) ..... 143, 500

[M]

Memory expansion mode register (MM) ..... 142, 505

[O]

Oscillation mode select register (OSMS) ..... 151  
 Oscillation stabilization time select register (OSTS) ..... 514

[P]

Port 0 (P0) ..... 122  
 Port 1 (P1) ..... 124  
 Port 12 (P12) ..... 136  
 Port 13 (P13) ..... 137  
 Port 2 (P2) ..... 125, 127  
 Port 3 (P3) ..... 129  
 Port 4 (P4) ..... 130  
 Port 5 (P5) ..... 131  
 Port 6 (P6) ..... 132  
 Port 7 (P7) ..... 134  
 Port mode register 0 (PM0) ..... 138  
 Port mode register 1 (PM1) ..... 138  
 Port mode register 12 (PM12) ..... 138, 472  
 Port mode register 13 (PM13) ..... 138  
 Port mode register 2 (PM2) ..... 138  
 Port mode register 3 (PM3) ..... 138, 176, 218, 249, 253  
 Port mode register 5 (PM5) ..... 138  
 Port mode register 6 (PM6) ..... 138  
 Port mode register 7 (PM7) ..... 138  
 Priority specify flag register 0H (PR0H) ..... 482  
 Priority specify flag register 0L (PR0L) ..... 482  
 Priority specify flag register 1L (PR1L) ..... 482  
 Processor clock control register (PCC) ..... 148  
 Program status word (PSW) ..... 96, 487  
 Pull-up resistor option register H (PUOH) ..... 141  
 Pull-up resistor option register L (PUOL) ..... 141

[R]

Real-time output buffer register H (RTBH) .....	471
Real-time output buffer register L (RTBL) .....	471
Real-time output port control register (RTPC) .....	473
Real-time output port mode register (RTPM) .....	472
Receive buffer register (RXB) .....	431
Receive shift register (RXS) .....	431

[S]

Sampling clock select register (SCS) .....	178, 485
Serial bus interface control register (SBIC) .....	293, 343
Serial I/O shift register 0 (SIO0) .....	286, 338
Serial I/O shift register 1 (SIO1) .....	385
Serial interface pin select register (SIPS) .....	441
Serial operating mode register 0 (CSIM0) .....	290, 342
Serial operating mode register 1 (CSIM1) .....	388
Serial operating mode register 2 (CSIM2) .....	432
Slave address register (SVA) .....	286, 338

[T]

Timer clock select register 0 (TCL0) .....	169, 247
Timer clock select register 1 (TCL1) .....	214
Timer clock select register 2 (TCL2) .....	233, 241, 251
Timer clock select register 3 (TCL3) .....	288, 340, 386
Transmit shift register (TXS) .....	431

[W]

Watch timer mode control register (TMC2) .....	236
Watchdog timer mode register (WDTM) .....	243

**C.2 Register Index (Symbol)**

[A]

ADCR: A/D conversion result register ..... 256  
 ADIS: A/D converter input select register ..... 260  
 ADM: A/D converter mode register ..... 258  
 ADTC: Automatic data transmit/receive control register ..... 389  
 ADTI: Automatic data transmit/receive interval specification register ..... 390  
 ADTP: Automatic data transmit/receive address pointer ..... 385  
 ASIM: Asynchronous serial interface mode register ..... 433  
 ASIS: Asynchronous serial interface status register ..... 436

[B]

BRGC: Baud rate generator control register ..... 437

[C]

CORAD0: Correction address register 0 ..... 526  
 CORAD1: Correction address register 1 ..... 526  
 CORCN: Correction control register ..... 527  
 CR00: Capture/compare register 00 ..... 167  
 CR01: Capture/compare register 01 ..... 168  
 CR10: Compare register 10 ..... 213  
 CR20: Compare register 20 ..... 213  
 CRC0: Capture/compare control register 0 ..... 173  
 CSIM0: Serial operating mode register 0 ..... 290, 342  
 CSIM1: Serial operating mode register 1 ..... 388  
 CSIM2: Serial operating mode register 2 ..... 432

[D]

DACS0: D/A conversion value set register 0 ..... 277  
 DACS1: D/A conversion value set register 1 ..... 277  
 DAM: D/A converter mode register ..... 278

[I]

IF0H: Interrupt request flag register 0H ..... 480  
 IF0L: Interrupt request flag register 0L ..... 480  
 IF1L: Interrupt request flag register 1L ..... 480, 499  
 IMS: Internal memory size switching register ..... 506, 536  
 INTM0: External interrupt mode register 0 ..... 177, 483  
 INTM1: External interrupt mode register 1 ..... 261, 483  
 IXS: Internal expansion RAM size switching register ..... 537

[K]

KRM: Key return mode register ..... 143, 500

[M]

MK0H: Interrupt mask flag register 0H ..... 481  
 MK0L: Interrupt mask flag register 0L ..... 481



MK1L:	Interrupt mask flag register 1L .....	481, 499
MM:	Memory expansion mode register .....	142, 505
[O]		
OSMS:	Oscillation mode selection register .....	151
OSTS:	Oscillation stabilization time select register .....	514
[P]		
P0:	Port 0 .....	122
P1:	Port 1 .....	124
P12:	Port 12 .....	136
P13:	Port 13 .....	137
P2:	Port 2 .....	125, 127
P3:	Port 3 .....	129
P4:	Port 4 .....	130
P5:	Port 5 .....	131
P6:	Port 6 .....	132
P7:	Port 7 .....	134
PCC:	Processor clock control register .....	148
PM0:	Port mode register 0 .....	138
PM1:	Port mode register 1 .....	138
PM12:	Port mode register 12 .....	138, 472
PM13:	Port mode register 13 .....	138
PM2:	Port mode register 2 .....	138
PM3:	Port mode register 3 .....	138, 176, 218, 249, 253
PM5:	Port mode register 5 .....	138
PM6:	Port mode register 6 .....	138
PM7:	Port mode register 7 .....	138
PROH:	Priority specification flag register 0H .....	482
PROL:	Priority specification flag register 0L .....	482
PR1L:	Priority specification flag register 1L .....	482
PSW:	Program status word .....	96, 487
PUOH:	Pull-up resistor option register H .....	141
PUOL:	Pull-up resistor option register L .....	141
[R]		
RTBH:	Real-time output buffer register H .....	471
RTBL:	Real-time output buffer register L .....	471
RTPC:	Real-time output port control register .....	473
RTPM:	Real-time output port mode register .....	472
RXB:	Receive buffer register .....	431
RXS:	Receive shift register .....	431
[S]		
SBIC:	Serial bus interface control register .....	293, 343
SCS:	Sampling clock select register .....	178, 485
SFR:	Special-function register .....	115
SINT:	Interrupt timing specification register .....	294, 345

SIO0: Serial I/O shift register 0 ..... 286, 338  
 SIO1: Serial I/O shift register 1 ..... 385  
 SIPS: Serial interface pin select register ..... 441  
 SVA: Slave address register ..... 286, 338

[T]

TCL0: Timer clock select register 0 ..... 169, 247  
 TCL1: Timer clock select register 1 ..... 214  
 TCL2: Timer clock select register 2 ..... 233, 241, 251  
 TCL3: Timer clock select register 3 ..... 288, 340, 386  
 TM0: 16-bit timer register ..... 168  
 TM1: 8-bit timer register 1 ..... 213  
 TM2: 8-bit timer register 2 ..... 213  
 TMC0: 16-bit timer mode control register ..... 171  
 TMC1: 8-bit timer mode control register ..... 216  
 TMC2: Watch timer mode control register ..... 236  
 TOC0: 16-bit timer output control register ..... 174  
 TOC1: 8-bit timer output control register ..... 217  
 TXS: Transmit shift register ..... 431

[W]

WDTM: Watchdog timer mode register ..... 243

## APPENDIX D REVISION HISTORY

The revision history of this edition is listed in the table below. “Chapter” indicates the chapter of the previous edition where the revision was made.

Edition	Revisions	Chapter
2nd edition	Change of following block diagrams of ports: <b>Figures 6-5 and 6-7 P20, P21, and P23 to P26 Block Diagram, Figures 6-6 and 6-8 P22 and P27 Block Diagram, Figure 6-9 P30 to P37 Block Diagram, and Figure 6-16 P71 and P72 Block Diagram</b>	<b>CHAPTER 6 PORT FUNCTIONS</b>
	Addition of <b>Table 7-2 Relationships between CPU Clock and Minimum Instruction Execution Time</b>	<b>CHAPTER 7 CLOCK GENERATOR</b>
	Addition of <b>Figures 9-10 and 9-13 Square Wave Output Operation Timing</b>	<b>CHAPTER 9 8-BIT TIMER/EVENT COUNTER</b>
	Addition of <b>(7) Conversion result immediately after A/D converter start</b> to 14.5 How to Read the A/D Converter Characteristics Table	<b>CHAPTER 14 A/D CONVERTER</b>
	Correction of Note on BSYE in <b>Figure 16-5 Serial Bus Interface Control Register Format</b>	<b>CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (<math>\mu</math>PD780058 Subseries)</b>
	Addition of Caution to <b>16.4.3 (2) (a) Bus release signal (REL) and (b) Command signal (CMD)</b>	
	Addition of <b>(3) MSB/LSB switching as the start bit to 18.4.2 3-wire serial I/O mode operation</b>	<b>CHAPTER 18 SERIAL INTERFACE CHANNEL 1</b>
	Change of <b>18.4.3 (3) (d) Busy control option, (e) Busy &amp; strobe control option, and (f) Bit slippage detection function</b> in old edition to <b>(4) Synchronization control</b> , and improvement of explanation	
	Correction of <b>Figure 19-11 Receive Error Timing</b>	<b>CHAPTER 19 SERIAL INTERFACE CHANNEL 2</b>
	Addition of <b>(3) MSB/LSB switching as the start bit to 19.4.3 3-wire serial I/O mode</b>	
	Addition of <b>19.4.4 Restrictions in UART mode</b>	
	Addition of Note to <b>26.1 Memory Size Switching Register</b>	<b>CHAPTER 26 <math>\mu</math>PD78F0058, 78F0058Y</b>
	<b>26.3 Flash Memory Programming</b> Change of product name of flash programmer from Flashpro to Flashpro II	
	Addition of <b>APPENDIX A DIFFERENCES AMONG <math>\mu</math>PD78054, 78058F, AND 780058 SUBSERIES</b>	<b>APPENDIX A DIFFERENCES AMONG <math>\mu</math>PD78054, 78058F, AND 780058 SUBSERIES</b>
	Total revision: Support of in-circuit emulators IE-78K0-NS and IE-78001-R-A	<b>APPENDIX B DEVELOPMENT TOOLS</b>
Total revision: Deletion of fuzzy inference development support system	<b>APPENDIX C EMBEDDED SOFTWARE</b>	

Edition	Revisions	Chapter
3rd edition	Deletion of following product • $\mu$ PD780058Y	Throughout
	Addition of following products • $\mu$ PD780058B, 780058BY, 780053(A), 780053Y(A), 780054(A), 780054Y(A), 780055(A), 780055Y(A), 780056(A), 780056Y(A), 780058B(A), 780058BY(A)	
	Deletion of following packages • 80-pin plastic QFP (GC-3B9 type) • 80-pin plastic TQFP (GK-BE9 type)	
	Addition of following package • 80-pin plastic TQFP (GK-9EU type)	
	<b>1.1 Features, 1.7 Outline of Functions</b> • Change of operating voltage range of A/D and D/A converters of $\mu$ PD780058 and 78F0058 • Change of supply voltage of $\mu$ PD78F0058	<b>CHAPTER 1 OUTLINE (<math>\mu</math>PD780058 SUBSERIES)</b>
	Addition of <b>1.9 Differences Between Standard Model and (A) Model</b>  <b>2.1 Features, 2.7 Outline of Functions</b> • Change of operating voltage range of A/D and D/A converters of $\mu$ PD78F0058Y • Change of supply voltage of $\mu$ PD78F0058Y	<b>CHAPTER 2 OUTLINE (<math>\mu</math>PD780058Y SUBSERIES)</b>
	Addition of <b>2.9 Differences Between Standard Model and (A) Model</b>	
	Change of processing when A/D converter is not used in <b>3.2.11 AVREF0</b>	<b>CHAPTER 3 PIN FUNCTIONS (<math>\mu</math>PD780058 SUBSERIES)</b>
	Change of recommended connection of unused pins and connection of P60 to P63, AVREF1, and VPP pins in <b>Table 3-1 Pin I/O Circuit Types</b>	<b>CHAPTER 3 PIN FUNCTIONS (<math>\mu</math>PD780058 SUBSERIES)</b>
	Change of processing when A/D converter is not used in <b>4.2.11 AVREF0</b>	<b>CHAPTER 4 PIN FUNCTIONS (<math>\mu</math>PD780058Y SUBSERIES)</b>
	Change of recommended connection of unused pins and connection of P60 to P63, AVREF1, and VPP pins in <b>Table 4-1 Pin I/O Circuit Types</b>	<b>CHAPTER 4 PIN FUNCTIONS (<math>\mu</math>PD780058Y SUBSERIES)</b>
	Modification of Note 2 in <b>6.2.8 Port 6</b>	<b>CHAPTER 6 PORT FUNCTIONS</b>
	Addition of note on feedback resistor to <b>Figure 7-3 Format of Processor Clock Control Register</b>	<b>CHAPTER 7 CLOCK GENERATOR</b>
	Addition of <b>Table 8-5 INTP1/TI01 Pin Valid Edge and CR00 Capture Trigger Valid Edge</b>	<b>CHAPTER 8 16-BIT TIMER/EVENT COUNTER</b>
	Addition of <b>Table 8-6 INTP0/TI00 Pin Valid Edge and CR01 Capture Trigger Valid Edge</b>	
	Correction of note on valid edge of INTP0/TI00/P00 and INTP1/TI01/P01 pin in <b>Figure 8-8 Format of External Interrupt Mode Register 0</b>	
	Addition of <b>Figure 8-17 Configuration of PPG Output</b> Addition of <b>Figure 8-18 PPG Output Operation Timing</b>  <b>8.5 16-Bit Timer/Event Counter Operating Cautions</b> Addition of description on TI01/P01/INTP1 to <b>(5) Valid edge setting</b> Addition of <b>(c) One-shot pulse output function</b> to <b>(6) Re-trigger of one-shot pulse</b> Addition of <b>(8) Conflict operation</b> Addition of <b>(9) Timer operation</b> Addition of <b>(10) Capture operation</b> Addition of <b>(11) Compare operation</b> Addition of <b>(12) Edge detection</b>	
	Modification of note on changing count clock in <b>Figure 10-2 Format of Timer Clock Select Register 2</b>	<b>CHAPTER 10 WATCH TIMER</b>

Edition	Revisions	Chapter
3rd edition	Modification of note on changing count clock in <b>Figure 11-2 Format of Timer Clock Select Register 2</b>	<b>CHAPTER 11 WATCHDOG TIMER</b>
	Addition of note on rewriting TCL2 in <b>Figure 13-2 Format of Timer Clock Select Register 2</b>	<b>CHAPTER 13 BUZZER OUTPUT CONTROLLER</b>
	Modification of <b>Figure 14-5 A/D Converter Basic Operation</b> Addition of <b>Table 14-2 A/D Conversion Sampling Time and A/D Converter Start Delay Time</b>	<b>CHAPTER 14 A/D CONVERTER</b>
	Addition of <b>14.5 How to Read A/D Converter Characteristics Table</b>	
	<b>14.6 A/D Converter Cautions</b> Change of description in <b>(1) Power consumption in standby mode</b> Addition of <b>(3) Conflicting operations</b> Addition of <b>(6) Input impedance of ANI0 to ANI7 pins</b> Addition of <b>(10) Timing at which A/D conversion result is undefined</b> Addition of <b>(11) Notes on board design</b> Addition of <b>(12) AV<sub>REF0</sub> pin</b> Addition of <b>(13) Internal equivalent circuit of ANI0 to ANI7 pins and permissible signal source impedance</b>	
	Addition of description of processing when D/A converter is not used in <b>15.5 D/A Converter Cautions (3) AV<sub>REF1</sub> pin</b>	<b>CHAPTER 15 D/A CONVERTER</b>
	Addition of <b>17.4.7 Restrictions in I<sup>2</sup>C bus mode 2</b>	<b>CHAPTER 17 SERIAL INTERFACE CHANNEL 0 (μPD780058Y SUBSERIES)</b>
	Addition of <b>19.4.5 Restrictions in UART mode 2</b>	<b>CHAPTER 19 SERIAL INTERFACE CHANNEL 2</b>
	Addition of Caution when interrupt is acknowledged to <b>Figure 21-2 Format of Interrupt Request Flag Register</b>	<b>CHAPTER 21 INTERRUPT AND TEST FUNCTIONS</b>
	Addition of description on TI01/P01/INTP1 pin to <b>Figure 21-5 Format of External Interrupt Mode Register 0</b>	
	Addition of Caution to <b>25.1 ROM Correction Function</b>	<b>CHAPTER 25 ROM CORRECTION</b>
	Modification of <b>Table 26-1 Differences Between μPD78F0058, 78F0058Y and Mask ROM Versions</b>	<b>CHAPTER 26 μPD78F0058, 78F0058Y</b>
	Total revision of description on flash memory programming as <b>26.3 Flash Memory Characteristics</b>	
	Addition of <b>CHAPTER 28 ELECTRICAL SPECIFICATIONS (MASK ROM VERSION)</b>	<b>CHAPTER 28 ELECTRICAL SPECIFICATIONS (MASK ROM VERSION)</b>
	Addition of <b>CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)</b>	<b>CHAPTER 29 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION)</b>
	Addition of <b>CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION (V<sub>DD</sub> = 2.2 V))</b>	<b>CHAPTER 30 ELECTRICAL SPECIFICATIONS (FLASH MEMORY VERSION (V<sub>DD</sub> = 2.2 V))</b>
	Addition of <b>CHAPTER 31 CHARACTERISTICS CURVES (REFERENCE VALUES)</b>	<b>CHAPTER 31 CHARACTERISTICS CURVES (REFERENCE VALUES)</b>
	Addition of <b>CHAPTER 32 PACKAGE DRAWINGS</b>	<b>CHAPTER 32 PACKAGE DRAWINGS</b>

**APPENDIX D REVISION HISTORY**

---

Edition	Revisions	Chapter
3rd edition	Addition of <b>CHAPTER 33 RECOMMENDED SOLDERING CONDITIONS</b>	<b>CHAPTER 33 RECOMMENDED SOLDERING CONDITIONS</b>
	Correction of <b>APPENDIX A DIFFERENCES BETWEEN <math>\mu</math>PD78054, 78058F, AND 780058</b>	<b>APPENDIX A DIFFERENCES BETWEEN <math>\mu</math>PD78054, 78058F, AND 780058 SUBSERIES</b>
	Total revision of <b>APPENDIX B DEVELOPMENT TOOLS</b> Transfer of description of embedded software to <b>APPENDIX B DEVELOPMENT TOOLS</b>	<b>APPENDIX B DEVELOPMENT TOOLS</b>