User's Manual

NEC

μPD780024A, 780034A, 780024AY, 780034AY Subseries

8-Bit Single-Chip Microcontrollers

μPD780021A
μPD780022A
μPD780023A
μPD780024A
μPD780021A(A)
μPD780022A(A)
μPD780023A(A)
μPD780024A(A)

μPD780031A
μPD780032A
μPD780033A
μPD780034A
μPD780031A(A)
μPD780032A(A)
μPD780033A(A)
μPD780034A(A)
μPD78F0034A
μPD78F0034B
μPD78F0034B(A)

μPD780021AY
μPD780022AY
μPD780023AY
μPD780024AY
μPD780021AY(A)
μPD780022AY(A)
μPD780023AY(A)
μPD780024AY(A)

μPD780031AY
μPD780032AY
μPD780033AY
μPD780034AY
μPD780031AY(A)
μPD780032AY(A)
μPD780033AY(A)
μPD780034AY(A)
μPD78F0034AY
μPD78F0034BY
μPD78F0034BY(A)

Document No. U14046EJ3V0UD00 (3rd edition)
Date Published October 2003 N CP(K)

[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.

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^2C components conveys a license under the Philips I^2C Patent Rights to use these components in an I^2C system, provided that the system conforms to the I^2C Standard Specification as defined by Philips.

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.

[GLOBAL SUPPORT]
http://www.necel.com/en/support/support.html

NEC Electronics America, Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782

NEC Electronics (Europe) GmbH
Duesseldorf, Germany
Tel: 0211-65 03 01

- **Sucursal en España**
  Madrid, Spain
  Tel: 091-504 27 87
- **Succursale Française**
  Vélizy-Villacoublay, France
  Tel: 01-30-67 58 00
- **Filiale Italiana**
  Milano, Italy
  Tel: 02-66 75 41
- **Branch The Netherlands**
  Eindhoven, The Netherlands
  Tel: 040-244 58 45
- **Tyskland Filial**
  Taeby, Sweden
  Tel: 08-63 80 820
- **United Kingdom Branch**
  Milton Keynes, UK
  Tel: 01908-691-133

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-558-3737

NEC Electronics Shanghai, Ltd.
Shanghai, P.R. China
Tel: 021-6841-1138

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-2719-2377

NEC Electronics Singapore Pte. Ltd.
Novena Square, Singapore
Tel: 6253-8311

J03.4

Major Revisions in This Edition (1/3)

Page	Description
Throughout	Addition of the following products μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A), μPD780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A), μPD78F0034B, 78F0034B(A), 78F0034BY, 78F0034BY(A)
	Addition of the following packages • 64-pin plastic LQFP (GC-8BS type) • 73-pin plastic FBGA (F1-CN3 type)
	Addition of expanded-specification products to μPD780024A, 780034A Subseries
p.34	Addition of **1.1 Expanded-Specification Products and Conventional Products**
p.54	Addition of **1.10 Correspondence Between Mask ROM Versions and Flash Memory Versions**
p.54	Modification of **1.11 Differences Between Standard Grade Products and Special Grade Products**
p.55	Addition of **1.12 Correspondence Between Products and Packages**
p.74	Addition of **2.9 Correspondence Between Mask ROM Versions and Flash Memory Versions**
p.75	Modification of **2.10 Differences Between Standard Grade Products and Special Grade Products**
p.75	Addition of **2.11 Correspondence Between Products and Packages**
p.84	Addition of description of pin processing in **3.2.18 V_{PP} (flash memory versions only)**
p.85	Modification of **Table 3-1 Pin I/O Circuit Types**
p.96	Addition of description of pin processing in **4.2.18 V_{PP} (flash memory versions only)**
p.97	Modification of **Table 4-1 Pin I/O Circuit Types**
p.108	Addition of description of program area in **5.1.2 Internal data memory space**
pp.116, 117	Modification of **Figure 5-14 Data to Be Saved to Stack Memory** and **Figure 5-15 Data to Be Restored from Stack Memory**
p.130	Modification of **[Description example]** in **5.4.4 Short direct addressing**
pp.133 to 135	Addition of **[Illustration]** in **5.4.7 Based addressing**, **5.4.8 Based indexed addressing**, and **5.4.9 Stack addressing**
pp.140 to 160	Modification of port block diagram (**Figures 6-2 Block Diagram of P00 to P03** to **6-23 Block Diagram of P74 and P75**)
p.163	Addition of **Table 6-6 Port Mode Registers and Output Latch Settings When Alternate Function Is Used**
pp.171, 174	Addition of description of internal feedback resistor and oscillation stabilization time select register (OSTS) in **7.3 Clock Generator Control Registers**
p.185	Modification of **Figure 8-1 Block Diagram of 16-Bit Timer/Event Counter 0**
pp.186, 187	Modification of **Tables 8-2 TI00/TO0/P70 Pin Valid Edge and CR00, CR01 Capture Trigger** and **8-3 TI01/P71 Pin Valid Edge and CR00 Capture Trigger** in 2nd edition to **Table 8-2 CR00 Capture Trigger and Valid Edges of TI00 and TI01 Pins** and **Table 8-3 CR01 Capture Trigger and Valid Edge of TI00 Pin (CRC02 = 1)**
p.194	Modification of description procedure of each function in **8.4 Operation of 16-Bit Timer/Event Counter 0**
p.208	Addition of **Figure 8-26 PPG Output Configuration Diagram** and **Figure 8-27 PPG Output Operation Timing**
p.209	Addition of **8.5 Program List**
pp.216, 218	Modification of **8.6 (3) Capture register data retention timing** and addition of **(11) STOP mode or main system clock stop mode setting**

Major Revisions in This Edition (2/3)

Page	Description
p.220	Modification of **Figures 9-1 Block Diagram of 8-Bit Timer/Event Counter 50** and **9-2 Block Diagram of 8-Bit Timer/Event Counter 51**
pp.224, 225	Deletion of **Caution** in **Figures 9-5 Format of 8-Bit Timer Mode Control Register 50 (TMC50)** and **9-6 Format of 8-Bit Timer Mode Control Register 51 (TMC51)**
p.231	Addition of **[Setting]** in **9.4.2 External event counter operation**
p.232	Addition of description of frequency to **[Setting]** in **9.4.3 Square-wave output (8-bit resolution) operation**
p.233	Addition of description of cycle and duty ratio to **[Setting]** in **9.4.4 8-bit PWM output operation**
p.238	Addition of **9.5 Program List**
p.200 in 2nd edition	Deletion of **9.5 (2) Operation after compare register change during timer count operation** in 2nd edition
p.208 in 2nd edition	Deletion of oscillation stabilization time select register (OSTS) from **11.3 Registers to Control Watchdog Timer** in 2nd edition
p.252	Modification of **Figure 12-1 Block Diagram of Clock Output/Buzzer Output Controller**
pp.259, 260	Modification of description in **13.2 (3) Sample & hold circuit**, **(4) Voltage comparator**, and addition of **(10) ADTRG pin**
p.266	Addition of **Table 13-2 Sampling Time and A/D Conversion Start Delay Time of A/D Converter**
pp.277, 278	Deletion of **13.6 (4) Noise countermeasures** (contents of deletion are added to **Figure 13-18 Example of Connecting Capacitor to AV$_{REF}$ Pin** and **Figure 13-20 Example of Connection If Signal Source Impedance Is High**), and addition of **(14) Input impedance of ANI0 to ANI7 pins**
p.278	Modification of **Table 13-3 Resistances and Capacitances of Equivalent Circuit (Reference Values)**
p.281	Addition of **Figure 14-2 Format of A/D Conversion Result Register 0 (ADCR0)**
pp.281, 282	Modification of description in **14.2 (3) Sample & hold circuit**, **(4) Voltage comparator**, and addition of **(10) ADTRG pin**
p.288	Addition of **Table 14-2 Sampling Time and A/D Conversion Start Delay Time of A/D Converter**
pp.298, 299	Deletion of **14.6 (4) Noise countermeasures** (contents of deletion are added to **Figure 14-19 Example of Connecting Capacitor to AV$_{REF}$ Pin** and **Figure 14-21 Example of Connection If Signal Source Impedance Is High**), and addition of **(14) Input impedance of ANI0 to ANI7 pins**
p.299	Modification of **Table 14-3 Resistances and Capacitances of Equivalent Circuit (Reference Values)**
p.302	Modification of **Figure 16-1 Block Diagram of Serial Interface UART0**
p.304	Move of description of asynchronous serial interface status register 0 (ASIS0) in **16.3 Registers to Control Serial Interface UART0** to **16.2 Configuration of Serial Interface UART0**
p.315	Addition of **Caution** in **Figure 16-7 Error Tolerance (When k = 0), Including Sampling Errors**
p.319	Modification of **Caution** in **Figure 16-10 Timing of Asynchronous Serial Interface Receive Completion Interrupt Request**
pp.321, 326	Addition of **(1) Registers to be used** and **(3) Relationship between main system clock and baud rate** in **16.4.3 Infrared data transfer mode**
p.328	Addition of **Table 16-6 Register Settings**
p.329	Modification of **Figure 17-1 Block Diagram of Serial Interface SIO3n**
pp.332, 333	Addition of **Note 3** and **Caution** in **Figures 17-2 Format of Serial Operation Mode Register 30 (CSIM30)** and **17-3 Format of Serial Operation Mode Register 31 (CSIM31)**
p.339	Addition of **Table 17-2 Register Settings**
p.341	Modification of **Figure 18-1 Block Diagram of Serial Interface IIC0**

Major Revisions in This Edition (3/3)

Page	Description
p.343	Unification of **18.2 (1) IIC shift register 0 (IIC0)** and **(4) IIC shift register 0 (IIC0)** in 2nd edition, and **(2) Slave address register 0 (SVA0)** and **(3) Slave address register 0 (SVA0)** in 2nd edition
p.360	Addition of description to “Transfer lines” in **Figure 18-16 Wait Signal**
p.362	Addition of description to **Notes 1** and **2** in **Table 18-2 INTIIC0 Timing and Wait Control**
p.369	Modification of **Figure 18-21 Master Operation Flowchart**
p.374	Modification of **18.5.15 (2) Slave operation**
p.396	Modification of **(1) Start condition ~ address** and **(2) Data** in **Figure 18-23 Example of Master to Slave Communication (When 9-Clock Wait Is Selected for Both Master and Slave)**
p.399	Modification of **Figure 18-24 Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave)**
p.405	Modification of **(E) Software interrupt** in **Figure 19-1 Basic Configuration of Interrupt Function**
p.407	Addition of **Caution 5** in **Figure 19-2 Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L)**
p.410	Addition of **Caution** in **Figure 19-5 Format of External Interrupt Rising Edge Enable Register (EGP), External Interrupt Falling Edge Enable Register (EGN)**
p.412	Addition of description and **Remark** in **19.4.1 Non-maskable interrupt request acknowledgment operation**
p.415	Addition of description in **19.4.2 Maskable interrupt request acknowledgment operation**
p.418	Addition of an item in **Table 19-4 Interrupt Requests Enabled for Nesting During Interrupt Servicing**
p.422	Addition of description of using expanded-specification products in **CHAPTER 20 EXTERNAL DEVICE EXPANSION FUNCTION**
p.435	Addition of clock output and buzzer output in **Table 21-1 HALT Mode Operating Statuses**
p.438	Modification of clock output in **Table 21-3 STOP Mode Operating Statuses**
p.445	Revision of **CHAPTER 23 μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY**
p.487	Addition of **CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS: f_X = 1.0 TO 12 MHz)**
p.518	Addition of **CHAPTER 26 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS: f_X = 1.0 TO 8.38 MHz)**
p.550	Addition of **CHAPTER 27 PACKAGE DRAWINGS**
p.556	Addition of **CHAPTER 28 RECOMMENDED SOLDERING CONDITIONS**
p.561	Revision of **APPENDIX A DIFFERENCES BETWEEN μPD78018F, 780024A, 780034A, AND 780078 SUBSERIES**
p.564	Revision of **APPENDIX B DEVELOPMENT TOOLS**
p.581	Addition of **APPENDIX C NOTES ON TARGET SYSTEM DESIGN**

The mark ★ shows major revised points.

INTRODUCTION

Readers

This manual has been prepared for user engineers who understand the functions of the μPD780024A, 780034A, 780024AY, and 780034AY Subseries and wish to design and develop application systems and programs for these devices.

- μPD780024A Subseries
  μPD780021A, 780022A, 780023A, 780024A
  μPD780021A(A), 780022A(A), 780023A(A), 780024A(A)
- μPD780034A Subseries
  μPD780031A, 780032A, 780033A, 780034A, 78F0034A, 78F0034B
  μPD780031A(A), 780032A(A), 780033A(A), 780034A(A), 78F0034B(A)
- μPD780024AY Subseries
  μPD780021AY, 780022AY, 780023AY, 780024AY
  μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A)
- μPD780034AY Subseries
  μPD780031AY, 780032AY, 780033AY, 780034AY, 78F0034AY, 78F0034BY
  μPD780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A), 78F0034BY(A)

Purpose

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

Organization

The μPD780024A, 780034A, 780024AY, and 780034AY Subseries manual is separated into two parts: this manual and the instructions edition (common to the 78K/0 Series).

μPD780024A, 780034A, 780024AY, 780034AY Subseries User's Manual (This Manual)	**78K/0 Series User's Manual Instructions**
• Pin functions	• CPU functions
• Internal block functions	• Instruction set
• Interrupt	• Explanation of each instruction
• Other on-chip peripheral functions	
• Electrical specifications	

How To Read This Manual It is assumed that the reader of this manual has general knowledge in the fields of electrical engineering, logic circuits, and microcontrollers.

- For readers who use this as an (A) product:
  → Standard products differ from (A) products in their quality grade only. Re-read the product name as indicated below if your product is an (A) product.

  μPD780021A → μPD780021A(A) μPD780021AY → μPD780021AY(A)
  μPD780022A → μPD780022A(A) μPD780022AY → μPD780022AY(A)
  μPD780023A → μPD780023A(A) μPD780023AY → μPD780023AY(A)
  μPD780024A → μPD780024A(A) μPD780024AY → μPD780024AY(A)
  μPD780031A → μPD780031A(A) μPD780031AY → μPD780031AY(A)
  μPD780032A → μPD780032A(A) μPD780032AY → μPD780032AY(A)
  μPD780033A → μPD780033A(A) μPD780033AY → μPD780033AY(A)
  μPD780034A → μPD780034A(A) μPD780034AY → μPD780034AY(A)
  μPD78F0034B → μPD78F0034B(A) μPD78F0034BY → μPD78F0034BY(A)

- To gain a general understanding of functions:
  → Read this manual in the order of the contents.
- How to interpret the register format:
  → For the bit number enclosed in brackets, the bit name is defined as a reserved word in RA78K0, and in CC78K0, already defined in the header file named sfrbit.h.
- To check the details of a register when you know the register name:
  → See **APPENDIX D REGISTER INDEX.**
- To know the details of the 78K/0 Series instruction functions:
  → Refer to the **78K/0 Series Instructions User's Manual (U12326E).**
- To know the electrical specifications of the μPD780024A, 780034A, 780024AY, 780034AY Subseries:
  → See **CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS: fx = 1.0 TO 12 MHz)** and **CHAPTER 26 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS: fx = 1.0 TO 8.38 MHz).**

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.

Differences between μPD780024A, 780034A, 780024AY, and 780034AY Subseries

The configuration of the serial interface and the resolution of the A/D converter differ on μPD780024A, 780034A, 780024AY, and 780034AY Subseries products.

Item \ Subseries		μPD780024A	μPD780034A	μPD780024AY	μPD780034AY
Configuration of serial interface	3-wire serial I/O mode	2ch (SIO30, SIO31)		1ch (SIO30 only)	
	UART mode	1ch		1ch	
	I^2C mode	None		1ch	
A/D converter		8-bit resolution	10-bit resolution	8-bit resolution	10-bit resolution

Chapter Organization This manual divides the descriptions for the subseries into different chapters as shown below. Read only the chapters related to the device you use.

Chapter		μPD780024A Subseries	μPD780034A Subseries	μPD780024AY Subseries	μPD780034AY Subseries
Chapter 1	Outline (μPD780024A, 780034A Subseries)	√	√	–	–
Chapter 2	Outline (μPD780024AY, 780034AY Subseries)	–	–	√	√
Chapter 3	Pin Function (μPD780024A, 780034A Subseries)	√	√	–	–
Chapter 4	Pin Function (μPD780024AY, 780034AY Subseries)	–	–	√	√
Chapter 5	CPU Architecture	√	√	√	√
Chapter 6	Port Functions	√	√	√	√
Chapter 7	Clock Generator	√	√	√	√
Chapter 8	16-Bit Timer/Event Counter 0	√	√	√	√
Chapter 9	8-Bit Timer/Event Counters 50, 51	√	√	√	√
Chapter 10	Watch Timer	√	√	√	√
Chapter 11	Watchdog Timer	√	√	√	√
Chapter 12	Clock Output/Buzzer Output Controller	√	√	√	√
Chapter 13	8-Bit A/D Converter (μPD780024A, 780024AY Subseries)	√	–	√	–
Chapter 14	10-Bit A/D Converter (μPD780034A, 780034AY Subseries)	–	√	–	√
Chapter 15	Serial Interface Outline	√	√	√	√
Chapter 16	Serial Interface UART0	√	√	√	√
Chapter 17	Serial Interface SIO3	√	√	√	√
Chapter 18	Serial Interface IIC0 (μPD780024AY, 780034AY Subseries only)	–	–	√	√
Chapter 19	Interrupt Functions	√	√	√	√
Chapter 20	External Device Expansion Function	√	√	√	√
Chapter 21	Standby Function	√	√	√	√
Chapter 22	Reset Function	√	√	√	√
Chapter 23	μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY	√	√	√	√
Chapter 24	Instruction Set	√	√	√	√
Chapter 25	Electrical Specifications (Expanded-Specification Products: f_X = 1.0 to 12 MHz)	√	√	√	√
Chapter 26	Electrical Specifications (Conventional Products: f_X = 1.0 to 8.38 MHz)	√	√	√	√
Chapter 27	Package Drawings	√	√	√	√
Chapter 28	Recommended Soldering Conditions	√	√	√	√

Conventions

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

Active low representation: $\overline{\times\times\times}$ (overscore over pin or signal name)

Note: Footnote for item marked with **Note** in the text

Caution: Information requiring particular attention

Remark: Supplementary information

Numerical representation: Binary ··· ×××× or ××××B

Decimal ··· ××××

Hexadecimal ··· ××××H

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.
μPD780024A, 780034A, 780024AY, 780034AY Subseries User's Manual	This manual
78K/0 Series Instructions User's Manual	U12326E
78K/0 Series Basic (I) Application Note	U12704E

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

Document Name		Document No.
RA78K0 Assembler Package	Operation	U14445E
	Language	U14446E
	Structured 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 Specifications	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 Development Tools (Hardware) (User's Manuals)**

Document Name	Document No.
IE-78K0-NS In-Circuit Emulator	U13731E
IE-78K0-NS-A In-Circuit Emulator	U14889E
IE-78K0-NS-PA Performance Board	To be prepared
IE-780034-NS-EM1 Emulation Board	U14642E
IE-78001-R-A In-Circuit Emulator	U14142E
IE-78K0-R-EX1 In-Circuit Emulator	To be prepared

★ **Documents Related to Flash Memory Programming**

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

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

★ Other Documents

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

Note See the “Semiconductor Device Mount Manual” website (http://www.necel.com/pkg/en/mount/index.html).

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

CONTENTS

LIST OF FIGURES (1/8)

LIST OF FIGURES (2/8)

LIST OF FIGURES (3/8)

LIST OF FIGURES (4/8)

LIST OF FIGURES (5/8)

LIST OF FIGURES (6/8)

LIST OF FIGURES (7/8)

LIST OF FIGURES (8/8)

LIST OF TABLES (1/3)

LIST OF TABLES (2/3)

LIST OF TABLES (3/3)

CHAPTER 1 OUTLINE (μPD780024A, 780034A SUBSERIES)

★ 1.1 Expanded-Specification Products and Conventional Products

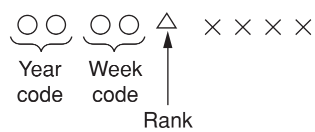
The expanded-specification products and conventional products refer to the following products.

Expanded-specification products: μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A for which orders were received after December 1, 2001
(Products with a rank[Note] other than K, E, P, X)
μPD78F0034B

Conventional products: Products other than the above expanded-specification products
(Products with rank[Note] K, E, P, X)
μPD78F0034A

Note The rank is indicated by the 5th digit from the left in the lot number marked on the package.

Lot number



Expanded-specification products and conventional products differ in the operating frequency ratings.

Power Supply Voltage (V_{DD})	Guaranteed Operating Speed (Operating Frequency)	
	Conventional Products	Expanded-Specification Products
4.5 to 5.5 V	8.38 MHz (0.238 μs)	12 MHz (0.166 μs)
4.0 to 5.5 V	8.38 MHz (0.238 μs)	8.38 MHz (0.238 μs)
3.0 to 5.5 V	5 MHz (0.4 μs)	8.38 MHz (0.238 μs)
2.7 to 5.5 V	5 MHz (0.4 μs)	5 MHz (0.4 μs)
1.8 to 5.5 V	1.25 MHz (1.6 μs)	1.25 MHz (1.6 μs)

Remark The parenthesized values indicate the minimum instruction execution time.

Caution Only the conventional products are available in the μPD780024AY and 780034AY Subseries (μPD780021AY, 780022AY, 780023AY, 780024AY, 780031AY, 780032AY, 780033AY, 780034AY, 78F0034AY, 78F0034BY).

1.2 Features

- Internal memory

Part Number \ Type	Program Memory (ROM/Flash Memory)	Data Memory (High-Speed RAM)
μPD780021A, 780031A	8 KB	512 bytes
μPD780022A, 780032A	16 KB	
μPD780023A, 780033A	24 KB	1024 bytes
μPD780024A, 780034A	32 KB	
μPD78F0034A, 78F0034B	32 KB[Note]	1024 bytes[Note]

Note The capacities of internal flash memory and internal high-speed RAM can be changed by means of the memory size switching register (IMS).

- External memory expansion space: 64 KB
- Minimum instruction execution time changeable from high speed (expanded-specification product (0.166 μs: @12 MHz operation with main system clock), conventional product (0.238 μs: @ 8.38 MHz operation with main system clock)) to ultra-low speed (122 μs: @ 32.768 kHz operation with subsystem clock)
- Instruction set suited to system control
  - Bit manipulation possible in all address spaces
  - Multiply and divide instructions
- Fifty-one I/O ports: (Four N-ch open-drain ports)
- 8-bit resolution A/D converter: 8 channels (μPD780024A Subseries only)
- 10-bit resolution A/D converter: 8 channels (μPD780034A Subseries only)
- Serial interface: 3 channels
  - 3-wire serial I/O mode: 2 channels
  - UART mode: 1 channel
- Timer: Five 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: 20
- Two types of on-chip clock oscillators (main system clock and subsystem clock)
- Power supply voltage: V_{DD} = 1.8 to 5.5 V

1.3 Applications

μPD780021A, 780022A, 780023A, 780024A
μPD780031A, 780032A, 780033A, 780034A, 78F0034A, 78F0034B

Home electric appliances, pagers, AV equipment, car audios, car electric equipment, office automation equipment, etc.

μPD780021A(A), 780022A(A), 780023A(A), 780024A(A)
μPD780031A(A), 780032A(A), 780033A(A), 780034A(A), 78F0034B(A)

Control of transportation equipment, gas detection breakers, safety devices, etc.

★
1.4 Ordering Information

(1) μPD780024A Subseries (1/2)

Part Number	Package	Internal ROM
μPD780021ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780021AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780021AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780021AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780021AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780021AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780022ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780022AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780022AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780022AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780022AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780022AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780023ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780023AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780023AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780023AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780023AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780023AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780024ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780024AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780024AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780024AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780024AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780024AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM

Remark ××× indicates ROM code suffix.

(1) μPD780024A Subseries (2/2)

Part Number	Package	Internal ROM
μPD780021ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780021AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780021AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780021AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780021AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780022ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780022AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780022AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780022AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780022AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780023ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780023AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780023AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780023AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780023AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780024ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780024AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780024AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780024AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780024AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM

Note Under development

Remark ××× indicates ROM code suffix.

(2) μPD780034A Subseries (1/2)

Part Number	Package	Internal ROM
μPD780031ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780031AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780031AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780031AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780031AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780031AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780032ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780032AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780032AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780032AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780032AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780032AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780033ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780033AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780033AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780033AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780033AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780033AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780034ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780034AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780034AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780034AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780034AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780034AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD78F0034ACW	64-pin plastic SDIP (19.05 mm (750))	Flash memory
μPD78F0034AGC-AB8[Note]	64-pin plastic QFP (14 × 14)	Flash memory
μPD78F0034AGC-8BS	64-pin plastic LQFP (14 × 14)	Flash memory
μPD78F0034AGK-9ET	64-pin plastic TQFP (12 × 12)	Flash memory
μPD78F0034AGB-8EU	64-pin plastic LQFP (10 × 10)	Flash memory
μPD78F0034BGC-8BS	64-pin plastic LQFP (14 × 14)	Flash memory
μPD78F0034BGK-9ET	64-pin plastic TQFP (12 × 12)	Flash memory
μPD78F0034BGB-8EU	64-pin plastic LQFP (10 × 10)	Flash memory
μPD78F0034BF1-CN3	73-pin plastic FBGA (9 × 9)	Flash memory

Note Maintenance product

Remark ××× indicates ROM code suffix.

(2) μPD780034A Subseries (2/2)

Part Number	Package	Internal ROM
μPD780031ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780031AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780031AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780031AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780031AGB(A)-×××-8EU^Note	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780032ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780032AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780032AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780032AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780032AGB(A)-×××-8EU^Note	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780033ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780033AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780033AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780033AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780033AGB(A)-×××-8EU^Note	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780034ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780034AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780034AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780034AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780034AGB(A)-×××-8EU^Note	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD78F0034BGC(A)-8BS	64-pin plastic LQFP (14 × 14)	Flash memory
μPD78F0034BGK(A)-9ET	64-pin plastic TQFP (12 × 12)	Flash memory
μPD78F0034BGB(A)-8EU	64-pin plastic LQFP (10 × 10)	Flash memory

Note Under development

Remark ××× indicates ROM code suffix.

★ ## 1.5 Quality Grade

(1) μPD780024A Subseries (1/2)

Part Number	Package	Quality Grades
μPD780021ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780021AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780021AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780021AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780021AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780021AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780022ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780022AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780022AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780022AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780022AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780022AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780023ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780023AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780023AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780023AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780023AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780023AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780024ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780024AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780024AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780024AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780024AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780024AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

(1) μPD780024A Subseries (2/2)

Part Number	Package	Quality Grades
μPD780021ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780021AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780021AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780021AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780021AGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Special
μPD780022ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780022AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780022AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780022AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780022AGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Special
μPD780023ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780023AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780023AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780023AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780023AGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Special
μPD780024ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780024AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780024AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780024AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780024AGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Special

Note Under development

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

(2) μPD780034A Subseries (1/2)

Part Number	Package	Quality Grades
μPD780031ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780031AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780031AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780031AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780031AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780031AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780032ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780032AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780032AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780032AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780032AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780032AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780033ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780033AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780033AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780033AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780033AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780033AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780034ACW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780034AGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780034AGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780034AGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780034AGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780034AF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD78F0034ACW	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD78F0034AGC-AB8[Note]	64-pin plastic QFP (14 × 14)	Standard
μPD78F0034AGC-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD78F0034AGK-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD78F0034AGB-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD78F0034BGC-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD78F0034BGK-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD78F0034BGB-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD78F0034BF1-CN3	73-pin plastic FBGA (9 × 9)	Standard

Note Maintenance product

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

(2) μPD780034A Subseries (2/2)

Part Number	Package	Quality Grades
μPD780031ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780031AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780031AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780031AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780031AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Special
μPD780032ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780032AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780032AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780032AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780032AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Special
μPD780033ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780033AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780033AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780033AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780033AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Special
μPD780034ACW(A)-×××	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780034AGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780034AGC(A)-×××-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD780034AGK(A)-×××-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD780034AGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Special
μPD78F0034BGC(A)-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD78F0034BGK(A)-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD78F0034BGB(A)-8EU	64-pin plastic LQFP (10 × 10)	Special

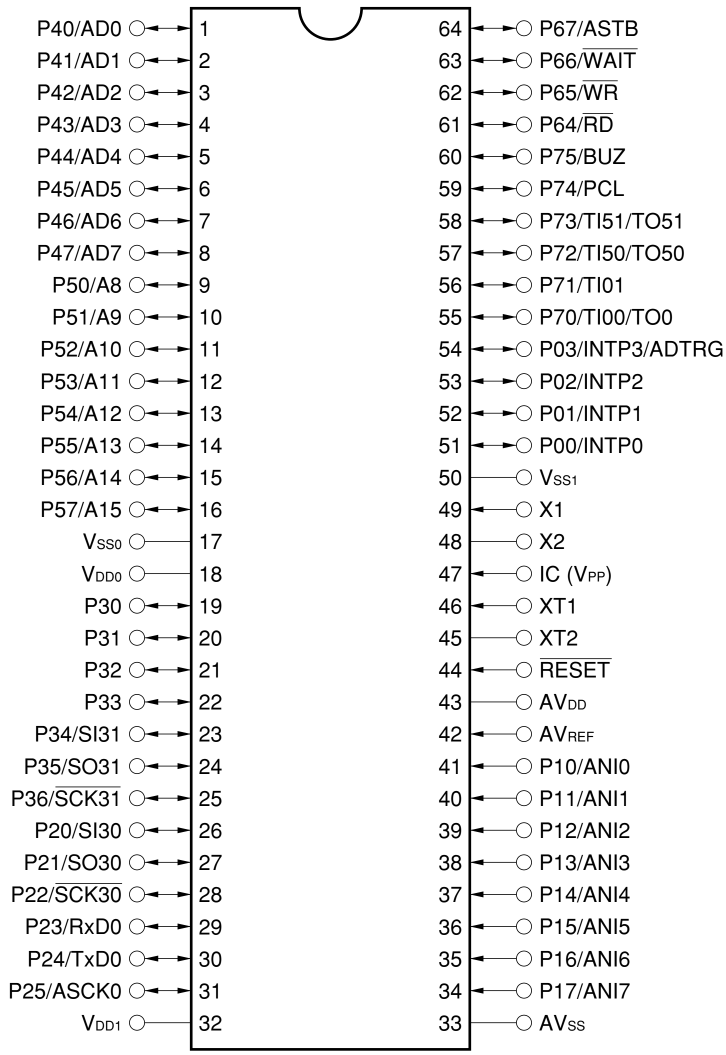
Note Under development

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

1.6 Pin Configuration (Top View)

- **64-pin plastic SDIP (19.05 mm (750))**



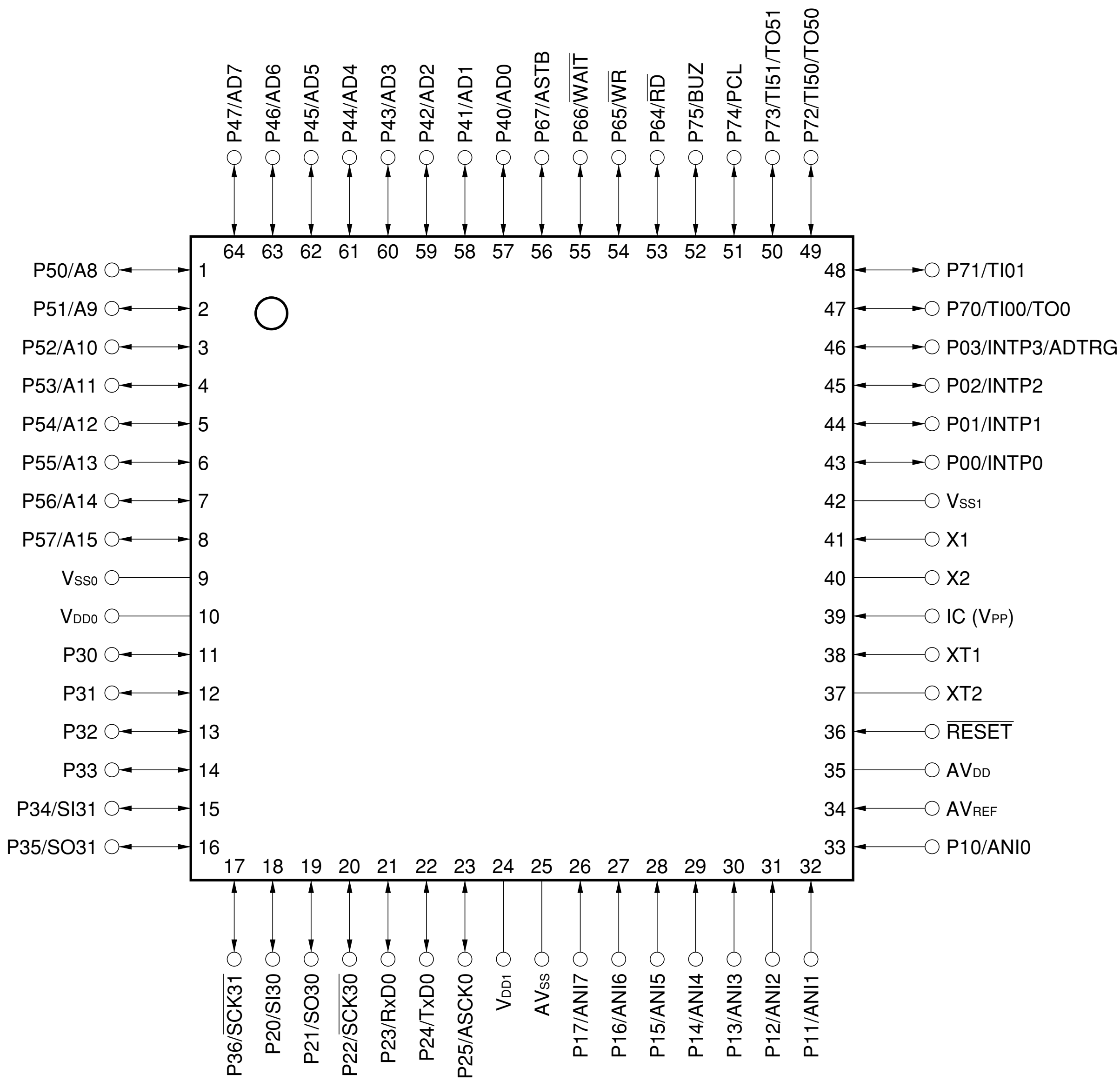
Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When the μPD780024A, 780034A Subseries products are used in applications where the noise generated inside the microcontroller needs to be reduced, the implementation of noise reduction measures, such as supplying voltage to V_{DD0} and V_{DD1} individually and connecting V_{SS0} and V_{SS1} to different ground lines, is recommended.

2. Pin connection in parentheses is intended for the μPD78F0034A.

- **64-pin plastic QFP (14 × 14)**
- **64-pin plastic LQFP (14 × 14)**
- **64-pin plastic TQFP (12 × 12)**
- **64-pin plastic LQFP (10 × 10)**



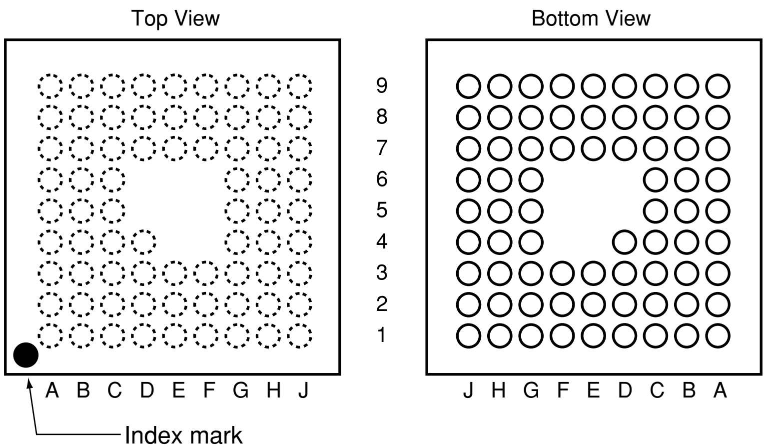
Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When the μPD780024A, 780034A Subseries products are used in applications where the noise generated inside the microcontroller needs to be reduced, the implementation of noise reduction measures, such as supplying voltage to V_{DD0} and V_{DD1} individually and connecting V_{SS0} and V_{SS1} to different ground lines, is recommended.

2. Pin connection in parentheses is intended for the μPD78F0034A, 78F0034B.

★ • **73-pin plastic FBGA (9 × 9)**



Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
A1	NC	C1	P52/A10	E1	P57/A15	G1	P33	J1	NC
A2	P46/AD6	C2	P53/A11	E2	V_{DD0}	G2	P32	J2	P36/$\overline{SCK31}$
A3	P44/AD4	C3	P45/AD5	E3	P54/A12	G3	P20/SI30	J3	NC
A4	P41/AD1	C4	P42/AD2	E4	–	G4	P21/SO30	J4	P25/ASCK0
A5	P67/ASTB	C5	P64/$\overline{RD}$	E5	–	G5	P24/TxD0	J5	NC
A6	P65/$\overline{WR}$	C6	P73/TI51/TO51	E6	–	G6	V_{DD1}	J6	P17/ANI7
A7	P74/PCL	C7	P03/INTP3/ADTRG	E7	P00/INTP0	G7	P16/ANI6	J7	P12/ANI2
A8	NC	C8	P01/INTP1	E8	XT1	G8	AV_{DD}	J8	P13/ANI3
A9	NC	C9	V_{SS1}	E9	X2	G9	NC	J9	NC
B1	P51/A9	D1	P55/A13	F1	P30	H1	P34/SI31		
B2	P47/AD7	D2	P56/A14	F2	P31	H2	P35/SO31		
B3	P43/AD3	D3	P50/A8	F3	V_{SS0}	H3	P23/RxD0		
B4	P40/AD0	D4	NC	F4	–	H4	P22/$\overline{SCK30}$		
B5	P66/$\overline{WAIT}$	D5	–	F5	–	H5	AV_{SS}		
B6	P75/BUZ	D6	–	F6	–	H6	P15/ANI5		
B7	P72/TI50/TO51	D7	P02/INTP2	F7	P14/ANI4	H7	P11/ANI1		
B8	P71/TI01	D8	IC (V_{PP})	F8	$\overline{RESET}$	H8	P10/ANI0		
B9	P70/TI00/TO0	D9	X1	F9	XT2	H9	AV_{REF}		

Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.
2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When the μPD780024A, 780034A Subseries products are used in applications where the noise generated inside the microcontroller needs to be reduced, the implementation of noise reduction measures, such as supplying voltage to V_{DD0} and V_{DD1} individually and connecting V_{SS0} and V_{SS1} to different ground lines, is recommended.

2. Pin connection in parentheses is intended for the μPD78F0034B.

A8 to A15:	Address bus
AD0 to AD7:	Address/data bus
ADTRG:	AD trigger input
ANI0 to ANI7:	Analog input
ASCK0:	Asynchronous serial clock
ASTB:	Address strobe
AV_{DD}:	Analog power supply
AV_{REF}:	Analog reference voltage
AV_{SS}:	Analog ground
BUZ:	Buzzer clock
IC:	Internally connected
INTP0 to INTP3:	External interrupt input
NC:	Non-connection
P00 to P03:	Port 0
P10 to P17:	Port 1
P20 to P25:	Port 2
P30 to P36:	Port 3
P40 to P47:	Port 4
P50 to P57:	Port 5
P64 to P67:	Port 6
P70 to P75:	Port 7
PCL:	Programmable clock
$\overline{\text{RD}}$:	Read strobe
$\overline{\text{RESET}}$:	Reset
RxD0:	Receive data
$\overline{\text{SCK30}}$, $\overline{\text{SCK31}}$:	Serial clock
SI30, SI31:	Serial input
SO30, SO31:	Serial output
TI00, TI01, TI50, TI51:	Timer input
TO0, TO50, TO51:	Timer output
TxD0:	Transmit data
V_{DD0}, V_{DD1}:	Power supply
V_{PP}:	Programming power supply
V_{SS0}, V_{SS1}:	Ground
$\overline{\text{WAIT}}$:	Wait
$\overline{\text{WR}}$:	Write strobe
X1, X2:	Crystal (main system clock)
XT1, XT2:	Crystal (subsystem clock)

★

1.7 78K/0 Series Lineup

The products in the 78K/0 Series are listed below. The names enclosed in boxes are subseries name.

Products in mass production (solid box) | Products under development (dashed box)

Y subseries products are compatible with I²C bus.

78K/0 Series

Control

100-pin	μPD78075B		EMI-noise reduced version of the μPD78078
100-pin	μPD78078	μPD78078Y	μPD78054 with timer and enhanced external interface
100-pin	μPD78070A	μPD78070AY	ROMless version of the μPD78078
100-pin		μPD780018AY	μPD78078Y with enhanced serial I/O and limited functions
80-pin	μPD780058	μPD780058Y	μPD78054 with enhanced serial I/O
80-pin	μPD78058F	μPD78058FY	EMI-noise reduced version of the μPD78054
80-pin	μPD78054	μPD78054Y	μPD78018F with UART and D/A converter, and enhanced I/O
80-pin	μPD780065		μPD780024A with expanded RAM
64-pin	μPD780078	μPD780078Y	μPD780034A with timer and enhanced serial I/O
64-pin	μPD780034A	μPD780034AY	μPD780024A with enhanced A/D converter
64-pin	μPD780024A	μPD780024AY	μPD78018F with enhanced serial I/O
52-pin	μPD780034AS		52-pin version of the μPD780034A
52-pin	μPD780024AS		52-pin version of the μPD780024A
64-pin	μPD78014H		EMI-noise reduced version of the μPD78018F
64-pin	μPD78018F	μPD78018FY	Basic subseries for control
42/44-pin	μPD78083		On-chip UART, capable of operating at low voltage (1.8 V)

Inverter control

64-pin	μPD780988		On-chip inverter controller and UART. EMI-noise reduced.

VFD drive

100-pin	μPD780208		μPD78044F with enhanced I/O and VFD C/D. Display output total: 53
80-pin	μPD780232		For panel control. On-chip VFD C/D. Display output total: 53
80-pin	μPD78044H		μPD78044F with N-ch open-drain I/O. Display output total: 34
80-pin	μPD78044F		Basic subseries for driving VFD. Display output total: 34

LCD drive

100-pin	μPD780354	μPD780354Y	μPD780344 with enhanced A/D converter
100-pin	μPD780344	μPD780344Y	μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780338		μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780328		μPD780308 with enhanced display function and timer. Segment signal output: 32 pins max.
120-pin	μPD780318		μPD780308 with enhanced display function and timer. Segment signal output: 24 pins max.
100-pin	μPD780308	μPD780308Y	μPD78064 with enhanced SIO, and expanded ROM and RAM
100-pin	μPD78064B		EMI-noise reduced version of the μPD78064
100-pin	μPD78064	μPD78064Y	Basic subseries for driving LCDs, on-chip UART

Bus interface supported

100-pin	μPD780948		On-chip CAN controller
80-pin	μPD78098B		μPD78054 with IEBus™ controller
80-pin		μPD780702Y	On-chip IEBus controller
80-pin		μPD780703AY	On-chip CAN controller
80-pin		μPD780833Y	On-chip controller compliant with J1850 (Class 2)
64-pin	μPD780816		Specialized for CAN controller function

Meter control

100-pin	μPD780958		For industrial meter control
80-pin	μPD780852		On-chip automobile meter controller/driver
80-pin	μPD780828B		For automobile meter driver. On-chip CAN controller

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.

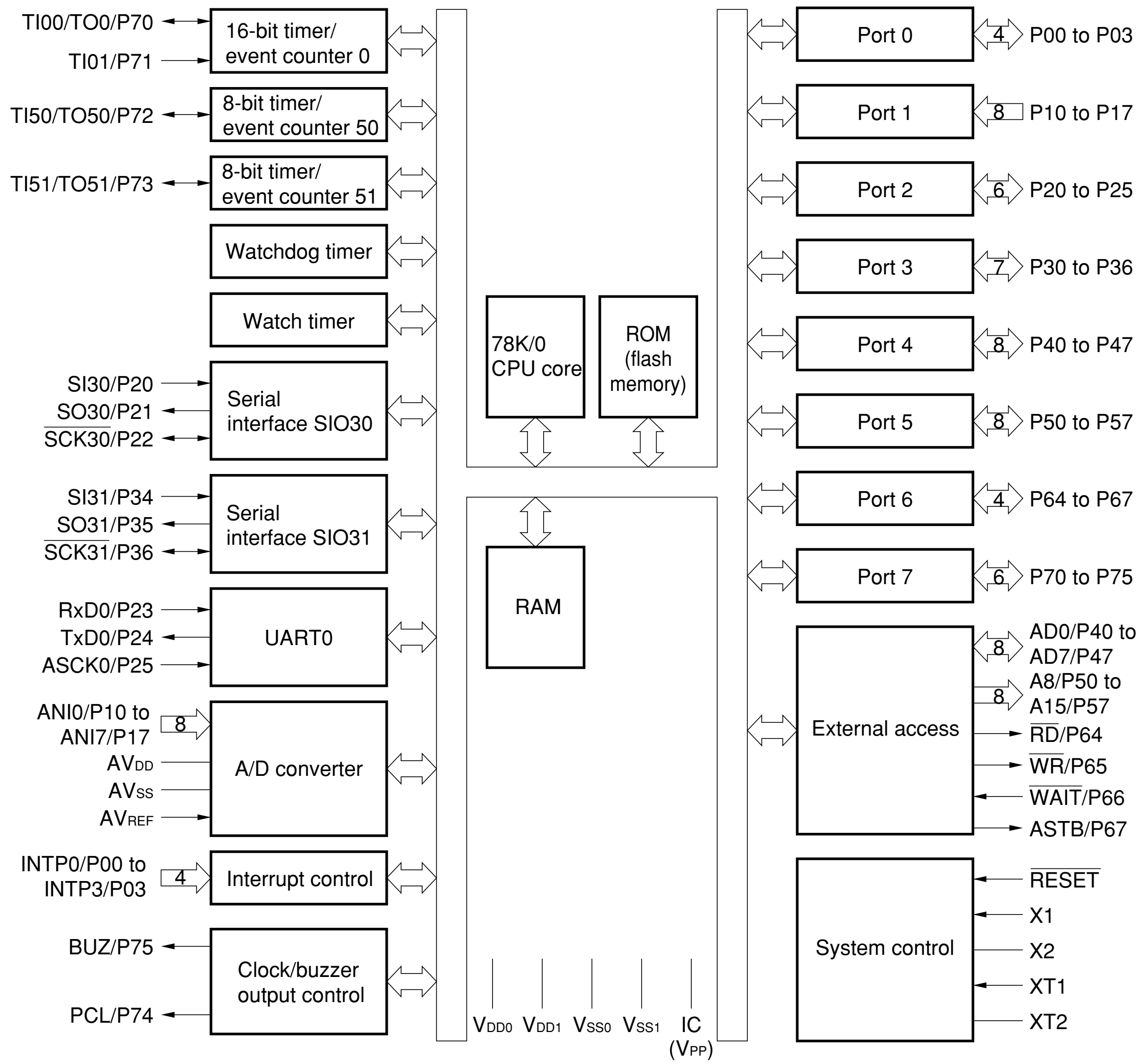
The major functional differences between the subseries are shown below.

- **Subseries without the suffix Y**

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

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

1.8 Block Diagram



Remarks 1. The internal ROM and RAM capacities depend on the product.

2. Pin connection in parentheses is intended for the μPD78F0034A, 78F0034B.

1.9 Outline of Function

Item	Part Number	μPD780021A μPD780031A	μPD780022A μPD780032A	μPD780023A μPD780033A	μPD780024A μPD780034A	μPD78F0034A μPD78F0034B
Internal memory	ROM	8 KB (Mask ROM)	16 KB (Mask ROM)	24 KB (Mask ROM)	32 KB (Mask ROM)	32 KB[Note] (Flash memory)
	High-speed RAM	512 bytes		1024 bytes		1024 bytes[Note]
Memory space		64 KB				
General-purpose register		8 bits × 32 registers (8 bits × 8 registers × 4 banks)				
Minimum instruction execution time		Minimum instruction execution time changeable function				
	When main system clock selected	• 0.166 μs/0.333 μs/0.666 μs/1.33 μs/2.66 μs (@ 12 MHz operation, expanded-specification product only) • 0.238 μs/0.477 μs/0.954 μs/1.90 μs/3.81 μs (@ 8.38 MHz operation)				
	When subsystem clock selected	122 μs (@ 32.768 kHz operation)				
Instruction set		• 16-bit operation • Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits) • Bit manipulate (set, reset, test, and Boolean operation) • BCD adjust, etc.				
I/O port		Total: 51 • CMOS input: 8 • CMOS I/O: 39 • N-ch open-drain I/O (5 V breakdown): 4				
A/D converter		• 8-bit resolution × 8 channels (μPD780021A, 780022A, 780023A, 780024A) • 10-bit resolution × 8 channels (μPD780031A, 780032A, 780033A, 780034A, 78F0034A, 78F0034B) • Low-voltage operation: AV_{DD} = 1.8 to 5.5 V				
Serial interface		• 3-wire serial I/O mode: 2 channels • UART mode: 1 channel				
Timer		• 16-bit timer/event counter: 1 channel • 8-bit timer/event counter: 2 channels • Watch timer: 1 channel • Watchdog timer: 1 channel				
Timer output		Three outputs (8-bit PWM output enable: 2)				
Clock output		• 93.7 kHz, 187 kHz, 375 kHz, 750 kHz, 1.5 MHz, 3 MHz, 6 MHz, 12 MHz (12 MHz with main system clock, expanded-specification product only) • 65.5 kHz, 131 kHz, 262 kHz, 524 kHz, 1.05 MHz, 2.10 MHz, 4.19 MHz, 8.38 MHz (8.38 MHz with main system clock) • 32.768 kHz (32.768 kHz with subsystem clock)				
Buzzer output		• 1.46 kHz, 2.92 kHz, 5.85 kHz, 11.7 kHz (12 MHz with main system clock, expanded-specification product only) • 1.02 kHz, 2.05 kHz, 4.10 kHz, 8.19 kHz (8.38 MHz with main system clock)				
Vectored interrupt source	Maskable	Internal: 13, External: 5				
	Non-maskable	Internal: 1				
	Software	1				

Note The capacities of internal flash memory and internal high-speed RAM can be changed by means of the memory size switching register (IMS).

Item \ Part Number	μPD780021A μPD780031A	μPD780022A μPD780032A	μPD780023A μPD780033A	μPD780024A μPD780034A	μPD78F0034A μPD78F0034B
Power supply voltage	V_{DD} = 1.8 to 5.5 V				
Operating ambient temperature	T_A = –40 to +85°C				
Package	• 64-pin plastic SDIP (19.05 mm (750)) • 64-pin plastic QFP (14 × 14) • 64-pin plastic LQFP (14 × 14) • 64-pin plastic TQFP (12 × 12) • 64-pin plastic LQFP (10 × 10) • 73-pin plastic FBGA (9 × 9) (standard grade product only)				

The outline of the timer/event counter is as follows (for details, see **CHAPTER 8 16-BIT TIMER/EVENT COUNTER 0**, **CHAPTER 9 8-BIT TIMER/EVENT COUNTERS 50, 51**, **CHAPTER 10 WATCH TIMER**, and **CHAPTER 11 WATCHDOG TIMER**).

		16-Bit Timer/ Event Counter 0	8-Bit Timer/ Event Counters 50, 51	Watch Timer	Watchdog Timer
Operation Mode	Interval timer	1 channel	2 channels	1 channel[Note 1]	1 channel[Note 2]
	External event counter	√	√	–	–
Function	Timer output	√	√	–	–
	PPG output	√	–	–	–
	PWM output	–	√	–	–
	Pulse width measurement	√	–	–	–
	Square-wave output	√	√	–	–
	Interrupt request	√	√	√	√

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.

★ 1.10 Correspondence Between Mask ROM Versions and Flash Memory Versions

Table 1-1. Correspondence Between Mask ROM Versions and Flash Memory Versions

Mask ROM Version / Flash Memory Version	μPD780021A/2A/3A/4A μPD780031A/2A/3A/4A		μPD780021A(A)/2A(A)/3A(A)/4A(A) μPD780031A(A)/2A(A)/3A(A)/4A(A)	
	Conventional Products	Expanded-Specification Products	Conventional Products	Expanded-Specification Products
μPD78F0034A	√	–	–	–
μPD78F0034B	–	√	–	–
μPD78F0034B(A)	–	–	√Note	√

Note The μPD78F0034B(A) and the conventional products of the μPD780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), and 780034A(A) differ in the operating frequency. When replacing a flash memory version with a mask ROM version, note the supply voltage and operating frequency.

Remarks 1. √: Supported, –: Not supported

2. The μPD780034A and μPD78F0034B, 78F0034B(A) differ in operating frequency ratings and the communication mode of flash memory programming. See **23.1 Differences Between μPD78F0034A, 78F0034AY and μPD78F0034B, 78F0034BY**.

3. Expanded-specification products and conventional products of the μPD780024A and 780034A Subseries differ in operating frequency ratings. For details, see **CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS: fx = 1.0 TO 12 MHz)** and **CHAPTER 26 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS: fx = 1.0 TO 8.38 MHz)**.

4. A special grade product of the μPD78F0034A is not available. Only a standard grade product is available.

★ 1.11 Differences Between Standard Grade Products and Special Grade Products

The differences between standard grade products (μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A, 78F0034A, 78F0034B) and special grade products (μPD780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), 780034A(A), 78F0034B(A)) are shown in Table 1-2.

Table 1-2. Differences Between Standard Grade Products and Special Grade Products

	μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A, 78F0034A, 78F0034B	μPD780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), 780034A(A), 78F0034B(A)
Quality grade	Standard	Special
Package	See **1.12 Correspondence Between Products and Packages**	
Other (functions, electrical specifications, etc.)	Same	

★ 1.12 Correspondence Between Products and Packages

The following table shows the correspondence between the products and packages.

Table 1-3. Correspondence Between Products and Packages

	Mask ROM Version		Flash Memory Version		
	μPD780021A/2A/3A/4A μPD780031A/2A/3A/4A		μPD78F0034A	μPD78F0034B	
	Standard	Special	Standard	Standard	Special
64-pin SDIP (CW type)	√	√	√	–	–
64-pin QFP (GC-AB8 type)	√	√	√**Note 1**	–	–
64-pin LQFP (GC-8BS type)	√	√	√	√	√
64-pin TQFP (GK-9ET type)	√	√	√	√	√
64-pin LQFP (GB-8EU type)	√	√**Note 2**	√	√	√
73-pin FBGA (F1-CN3 type)	√	–	–	√	–

Notes 1. Maintenance product
2. Under development

Remarks 1. √: Package available, –: Package not available
2. A special grade product of the μPD78F0034A is not available. Only a standard grade product is available.

1.13 Mask Options

The mask ROM versions (μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, and 780034A) 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 device production. Using the 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 μPD780024A and 780034A Subseries are shown in Table 1-4.

Table 1-4. Mask Options of Mask ROM Versions

Pin Names	Mask Option
P30 to P33	Pull-up resistor connection can be specified in 1-bit units.

CHAPTER 2 OUTLINE (μPD780024AY, 780034AY SUBSERIES)

2.1 Features

- Internal memory

Part Number \ Type	Program Memory (ROM/Flash Memory)	Data Memory (High-Speed RAM)
μPD780021AY, 780031AY	8 KB	512 bytes
μPD780022AY, 780032AY	16 KB	
μPD780023AY, 780033AY	24 KB	1024 bytes
μPD780024AY, 780034AY	32 KB	
μPD78F0034AY, 78F0034BY	32 KB[Note]	1024 bytes[Note]

Note The capacities of internal flash memory and internal high-speed RAM can be changed by means of the memory size switching register (IMS).

- External memory expansion space: 64 KB
- Minimum instruction execution time changeable from high speed (0.238 μs: @ 8.38 MHz operation with main system clock) to ultra-low speed (122 μs: @ 32.768 kHz operation with subsystem clock)
- Instruction set suited to system control
  - Bit manipulation possible in all address spaces
  - Multiply and divide instructions
- Fifty-one I/O ports: (Four N-ch open-drain ports)
- 8-bit resolution A/D converter: 8 channels (μPD780024AY Subseries only)
- 10-bit resolution A/D converter: 8 channels (μPD780034AY Subseries only)
- Serial interface: 3 channels
  - 3-wire serial mode: 1 channel
  - UART mode: 1 channel
  - I^2C mode: 1 channel
- Timer: Five 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: 20
- Two types of on-chip clock oscillators (main system clock and subsystem clock)
- Power supply voltage: V_{DD} = 1.8 to 5.5 V

Caution Only the conventional products are available in the μPD780024AY and 780034AY Subseries (for details of conventional products, see 1.1 Expanded-Specification Products and Conventional Products).

2.2 Applications

μPD780021AY, 780022AY, 780023AY, 780024AY
μPD780031AY, 780032AY, 780033AY, 780034AY, 78F0034AY, 78F0034BY

Home electric appliances, pagers, AV equipment, car audios, car electric equipment, office automation equipment, etc.

μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A)
μPD780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A), 78F0034BY(A)

Control of transportation equipment, gas detection breakers, safety devices, etc.

★ 2.3 Ordering Information

(1) μPD780024AY Subseries (1/2)

Part Number	Package	Internal ROM
μPD780021AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780021AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780021AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780021AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780021AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780021AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780022AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780022AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780022AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780022AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780022AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780022AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780023AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780023AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780023AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780023AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780023AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780023AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780024AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780024AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780024AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780024AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780024AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780024AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM

Remark ××× indicates ROM code suffix.

(1) μPD780024AY Subseries (2/2)

Part Number	Package	Internal ROM
μPD780021AYCW(A)-×××[Note]	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780021AYGC(A)-×××-AB8[Note]	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780021AYGC(A)-×××-8BS[Note]	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780021AYGK(A)-×××-9ET[Note]	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780021AYGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780022AYCW(A)-×××[Note]	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780022AYGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780022AYGC(A)-×××-8BS[Note]	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780022AYGK(A)-×××-9ET[Note]	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780022AYGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780023AYCW(A)-×××[Note]	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780023AYGC(A)-×××-AB8[Note]	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780023AYGC(A)-×××-8BS[Note]	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780023AYGK(A)-×××-9ET[Note]	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780023AYGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780024AYCW(A)-×××[Note]	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780024AYGC(A)-×××-AB8[Note]	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780024AYGC(A)-×××-8BS[Note]	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780024AYGK(A)-×××-9ET[Note]	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780024AYGB(A)-×××-8EU[Note]	64-pin plastic LQFP (10 × 10)	Mask ROM

Note Under development

Remark ××× indicates ROM code suffix.

(2) μPD780034AY Subseries (1/2)

Part Number	Package	Internal ROM
μPD780031AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780031AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780031AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780031AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780031AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780031AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780032AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780032AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780032AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780032AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780032AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780032AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780033AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780033AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780033AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780033AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780033AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780033AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD780034AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780034AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780034AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780034AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780034AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780034AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Mask ROM
μPD78F0034AYCW	64-pin plastic SDIP (19.05 mm (750))	Flash memory
μPD78F0034AYGC-AB8	64-pin plastic QFP (14 × 14)	Flash memory
μPD78F0034AYGC-8BS	64-pin plastic LQFP (14 × 14)	Flash memory
μPD78F0034AYGK-9ET	64-pin plastic TQFP (12 × 12)	Flash memory
μPD78F0034AYGB-8EU	64-pin plastic LQFP (10 × 10)	Flash memory
μPD78F0034BYGC-8BS	64-pin plastic LQFP (14 × 14)	Flash memory
μPD78F0034BYGK-9ET	64-pin plastic TQFP (12 × 12)	Flash memory
μPD78F0034BYGB-8EU	64-pin plastic LQFP (10 × 10)	Flash memory
μPD78F0034BYF1-CN3	73-pin plastic FBGA (9 × 9)	Flash memory

Remark ××× indicates ROM code suffix.

(2) μPD780034AY Subseries (2/2)

Part Number	Package	Internal ROM
μPD780031AYCW(A)-×××**Note**	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780031AYGC(A)-×××-AB8**Note**	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780031AYGC(A)-×××-8BS**Note**	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780031AYGK(A)-×××-9ET**Note**	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780031AYGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780032AYCW(A)-×××**Note**	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780032AYGC(A)-×××-AB8**Note**	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780032AYGC(A)-×××-8BS**Note**	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780032AYGK(A)-×××-9ET**Note**	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780032AYGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780033AYCW(A)-×××**Note**	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780033AYGC(A)-×××-AB8**Note**	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780033AYGC(A)-×××-8BS**Note**	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780033AYGK(A)-×××-9ET**Note**	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780033AYGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD780034AYCW(A)-×××**Note**	64-pin plastic SDIP (19.05 mm (750))	Mask ROM
μPD780034AYGC(A)-×××-AB8**Note**	64-pin plastic QFP (14 × 14)	Mask ROM
μPD780034AYGC(A)-×××-8BS**Note**	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD780034AYGK(A)-×××-9ET**Note**	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD780034AYGB(A)-×××-8EU**Note**	64-pin plastic LQFP (10 × 10)	Mask ROM
μPD78F0034BYGC(A)-8BS	64-pin plastic LQFP (14 × 14)	Mask ROM
μPD78F0034BYGK(A)-9ET	64-pin plastic TQFP (12 × 12)	Mask ROM
μPD78F0034BYGB(A)-8EU	64-pin plastic LQFP (10 × 10)	Mask ROM

Note Under development

Remark ××× indicates ROM code suffix.

★ 2.4 Quality Grade

(1) μPD780024AY Subseries (1/2)

Part Number	Package	Quality Grades
μPD780021AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780021AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780021AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780021AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780021AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780021AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780022AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780022AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780022AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780022AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780022AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780022AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780023AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780023AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780023AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780023AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780023AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780023AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780024AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780024AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780024AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780024AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780024AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780024AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

(1) μPD780024AY Subseries (2/2)

Part Number	Package	Quality Grades
μPD780021AYCW(A)-×××Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780021AYGC(A)-×××-AB8Note	64-pin plastic QFP (14 × 14)	Special
μPD780021AYGC(A)-×××-8BSNote	64-pin plastic LQFP (14 × 14)	Special
μPD780021AYGK(A)-×××-9ETNote	64-pin plastic TQFP (12 × 12)	Special
μPD780021AYGB(A)-×××-8EUNote	64-pin plastic LQFP (10 × 10)	Special
μPD780022AYCW(A)-×××Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780022AYGC(A)-×××-AB8	64-pin plastic QFP (14 × 14)	Special
μPD780022AYGC(A)-×××-8BSNote	64-pin plastic LQFP (14 × 14)	Special
μPD780022AYGK(A)-×××-9ETNote	64-pin plastic TQFP (12 × 12)	Special
μPD780022AYGB(A)-×××-8EUNote	64-pin plastic LQFP (10 × 10)	Special
μPD780023AYCW(A)-×××Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780023AYGC(A)-×××-AB8Note	64-pin plastic QFP (14 × 14)	Special
μPD780023AYGC(A)-×××-8BSNote	64-pin plastic LQFP (14 × 14)	Special
μPD780023AYGK(A)-×××-9ETNote	64-pin plastic TQFP (12 × 12)	Special
μPD780023AYGB(A)-×××-8EUNote	64-pin plastic LQFP (10 × 10)	Special
μPD780024AYCW(A)-×××Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780024AYGC(A)-×××-AB8Note	64-pin plastic QFP (14 × 14)	Special
μPD780024AYGC(A)-×××-8BSNote	64-pin plastic LQFP (14 × 14)	Special
μPD780024AYGK(A)-×××-9ETNote	64-pin plastic TQFP (12 × 12)	Special
μPD780024AYGB(A)-×××-8EUNote	64-pin plastic LQFP (10 × 10)	Special

Note Under development

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

(2) μPD780034AY Subseries (1/2)

Part Number	Package	Quality Grades
μPD780031AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780031AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780031AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780031AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780031AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780031AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780032AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780032AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780032AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780032AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780032AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780032AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780033AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780033AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780033AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780033AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780033AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780033AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD780034AYCW-×××	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD780034AYGC-×××-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD780034AYGC-×××-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD780034AYGK-×××-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD780034AYGB-×××-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD780034AYF1-×××-CN3	73-pin plastic FBGA (9 × 9)	Standard
μPD78F0034AYCW	64-pin plastic SDIP (19.05 mm (750))	Standard
μPD78F0034AYGC-AB8	64-pin plastic QFP (14 × 14)	Standard
μPD78F0034AYGC-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD78F0034AYGK-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD78F0034AYGB-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD78F0034BYGC-8BS	64-pin plastic LQFP (14 × 14)	Standard
μPD78F0034BYGK-9ET	64-pin plastic TQFP (12 × 12)	Standard
μPD78F0034BYGB-8EU	64-pin plastic LQFP (10 × 10)	Standard
μPD78F0034BYF1-CN3	73-pin plastic FBGA (9 × 9)	Standard

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

(2) μPD780034AY Subseries (2/2)

Part Number	Package	Quality Grades
μPD780031AYCW(A)-××× Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780031AYGC(A)-×××-AB8 Note	64-pin plastic QFP (14 × 14)	Special
μPD780031AYGC(A)-×××-8BS Note	64-pin plastic LQFP (14 × 14)	Special
μPD780031AYGK(A)-×××-9ET Note	64-pin plastic TQFP (12 × 12)	Special
μPD780031AYGB(A)-×××-8EU Note	64-pin plastic LQFP (10 × 10)	Special
μPD780032AYCW(A)-××× Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780032AYGC(A)-×××-AB8 Note	64-pin plastic QFP (14 × 14)	Special
μPD780032AYGC(A)-×××-8BS Note	64-pin plastic LQFP (14 × 14)	Special
μPD780032AYGK(A)-×××-9ET Note	64-pin plastic TQFP (12 × 12)	Special
μPD780032AYGB(A)-×××-8EU Note	64-pin plastic LQFP (10 × 10)	Special
μPD780033AYCW(A)-××× Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780033AYGC(A)-×××-AB8 Note	64-pin plastic QFP (14 × 14)	Special
μPD780033AYGC(A)-×××-8BS Note	64-pin plastic LQFP (14 × 14)	Special
μPD780033AYGK(A)-×××-9ET Note	64-pin plastic TQFP (12 × 12)	Special
μPD780033AYGB(A)-×××-8EU Note	64-pin plastic LQFP (10 × 10)	Special
μPD780034AYCW(A)-××× Note	64-pin plastic SDIP (19.05 mm (750))	Special
μPD780034AYGC(A)-×××-AB8 Note	64-pin plastic QFP (14 × 14)	Special
μPD780034AYGC(A)-×××-8BS Note	64-pin plastic LQFP (14 × 14)	Special
μPD780034AYGK(A)-×××-9ET Note	64-pin plastic TQFP (12 × 12)	Special
μPD780034AYGB(A)-×××-8EU Note	64-pin plastic LQFP (10 × 10)	Special
μPD78F0034BYGC(A)-8BS	64-pin plastic LQFP (14 × 14)	Special
μPD78F0034BYGK(A)-9ET	64-pin plastic TQFP (12 × 12)	Special
μPD78F0034BYGB(A)-8EU	64-pin plastic LQFP (10 × 10)	Special

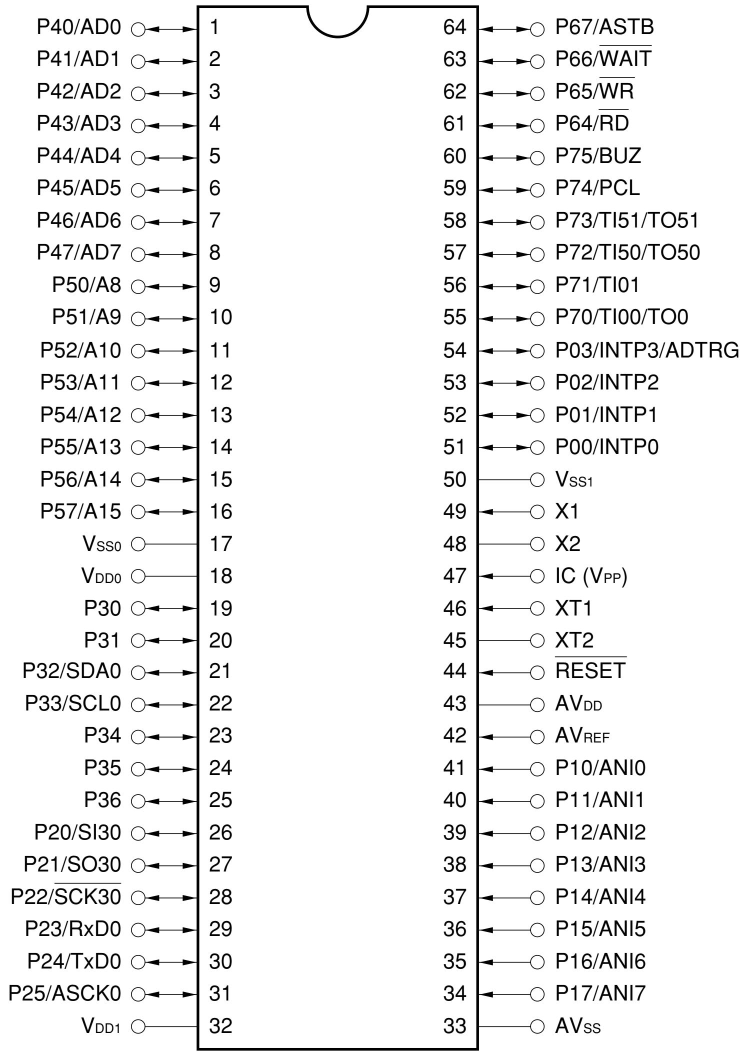
Note Under development

Remark ××× indicates ROM code suffix.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

2.5 Pin Configuration (Top View)

- **64-pin plastic SDIP (19.05 mm (750))**



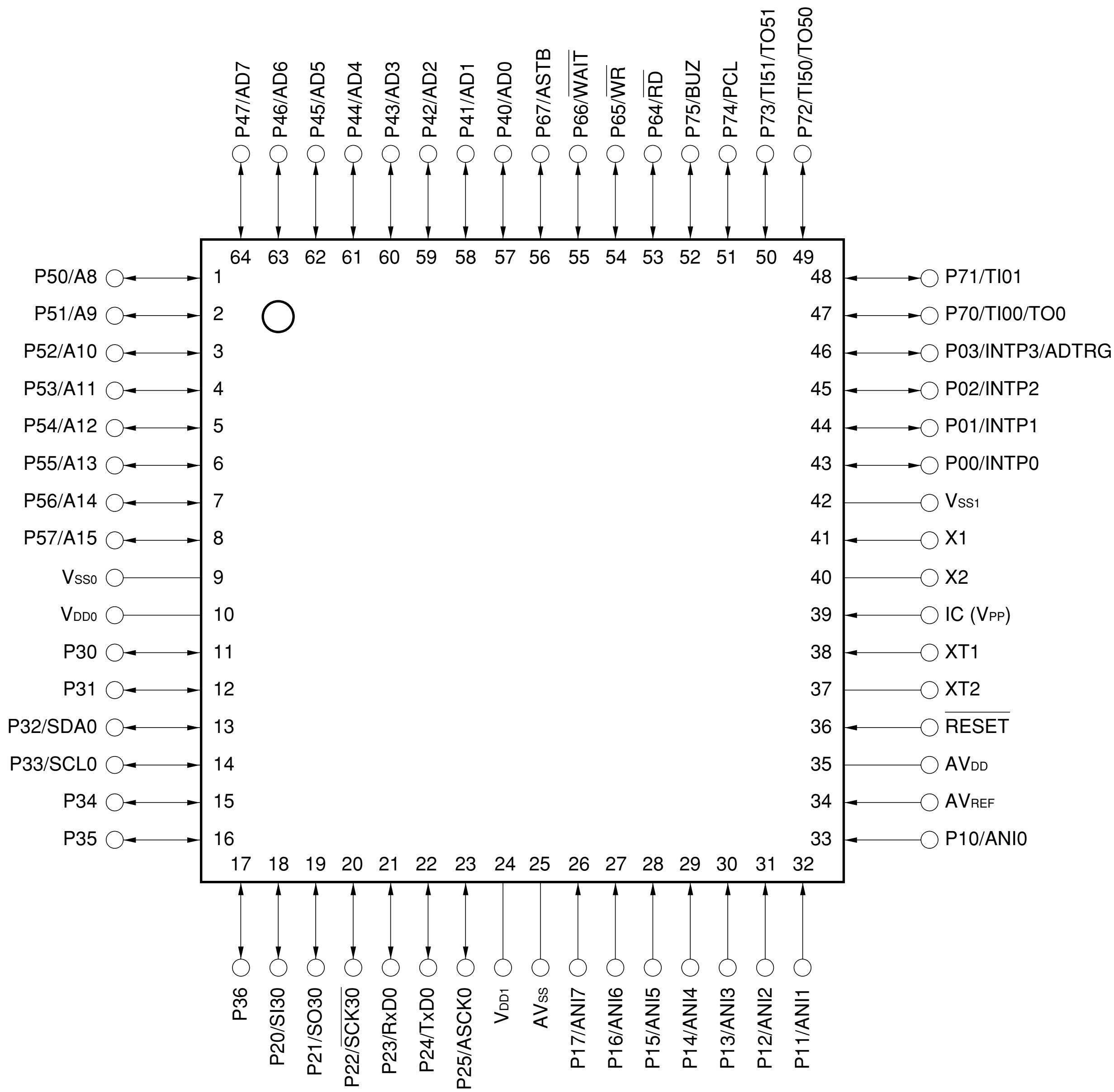
Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When the μPD780024AY, 780034AY Subseries products are used in applications where the noise generated inside the microcontroller needs to be reduced, the implementation of noise reduction measures, such as supplying voltage to V_{DD0} and V_{DD1} individually and connecting V_{SS0} and V_{SS1} to different ground lines, is recommended.

2. Pin connection in parentheses is intended for the μPD78F0034AY.

- **64-pin plastic QFP (14 × 14)**
- **64-pin plastic LQFP (14 × 14)**
- **64-pin plastic TQFP (12 × 12)**
- **64-pin plastic LQFP (10 × 10)**



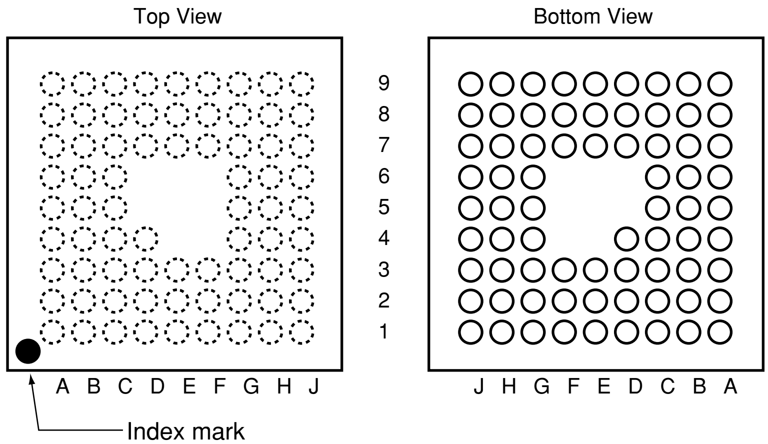
Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When the μPD780024AY, 780034AY Subseries products are used in applications where the noise generated inside the microcontroller needs to be reduced, the implementation of noise reduction measures, such as supplying voltage to V_{DD0} and V_{DD1} individually and connecting V_{SS0} and V_{SS1} to different ground lines, is recommended.

2. Pin connection in parentheses is intended for the μPD78F0034AY, 78F0034BY.

★ • **73-pin plastic FBGA (9 × 9)**



Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
A1	NC	C1	P52/A10	E1	P57/A15	G1	P33/SCL0	J1	NC
A2	P46/AD6	C2	P53/A11	E2	V_{DD0}	G2	P32/SDA0	J2	P36
A3	P44/AD4	C3	P45/AD5	E3	P54/A12	G3	P20/SI30	J3	NC
A4	P41/AD1	C4	P42/AD2	E4	–	G4	P21/SO30	J4	P25/ASCK0
A5	P67/ASTB	C5	P64/$\overline{\text{RD}}$	E5	–	G5	P24/TxD0	J5	NC
A6	P65/$\overline{\text{WR}}$	C6	P73/TI51/TO51	E6	–	G6	V_{DD1}	J6	P17/ANI7
A7	P74/PCL	C7	P03/INTP3/ADTRG	E7	P00/INTP0	G7	P16/ANI6	J7	P12/ANI2
A8	NC	C8	P01/INTP1	E8	XT1	G8	AV_{DD}	J8	P13/ANI3
A9	NC	C9	V_{SS1}	E9	X2	G9	NC	J9	NC
B1	P51/A9	D1	P55/A13	F1	P30	H1	P34		
B2	P47/AD7	D2	P56/A14	F2	P31	H2	P35		
B3	P43/AD3	D3	P50/A8	F3	V_{SS0}	H3	P23/RxD0		
B4	P40/AD0	D4	NC	F4	–	H4	P22/$\overline{\text{SCK30}}$		
B5	P66/$\overline{\text{WAIT}}$	D5	–	F5	–	H5	AV_{SS}		
B6	P75/BUZ	D6	–	F6	–	H6	P15/ANI5		
B7	P72/TI50/TO51	D7	P02/INTP2	F7	P14/ANI4	H7	P11/ANI1		
B8	P71/TI01	D8	IC (V_{PP})	F8	$\overline{\text{RESET}}$	H8	P10/ANI0		
B9	P70/TI00/TO0	D9	X1	F9	XT2	H9	AV_{REF}		

Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. When the μPD780024AY, 780034AY Subseries products are used in applications where the noise generated inside the microcontroller needs to be reduced, the implementation of noise reduction measures, such as supplying voltage to V_{DD0} and V_{DD1} individually and connecting V_{SS0} and V_{SS1} to different ground lines, is recommended.

2. Pin connection in parentheses is intended for the μPD78F0034BY.

A8 to A15:	Address bus
AD0 to AD7:	Address/data bus
ADTRG:	AD trigger input
ANI0 to ANI7:	Analog input
ASCK0:	Asynchronous serial clock
ASTB:	Address strobe
AV_{DD}:	Analog power supply
AV_{REF}:	Analog reference voltage
AV_{SS}:	Analog ground
BUZ:	Buzzer clock
IC:	Internally connected
INTP0 to INTP3:	External interrupt input
NC:	Non-connection
P00 to P03:	Port 0
P10 to P17:	Port 1
P20 to P25:	Port 2
P30 to P36:	Port 3
P40 to P47:	Port 4
P50 to P57:	Port 5
P64 to P67:	Port 6
P70 to P75:	Port 7
PCL:	Programmable clock
$\overline{RD}$:	Read strobe
$\overline{RESET}$:	Reset
RxD0:	Receive data
$\overline{SCK30}$:	Serial clock
SCL0:	Serial clock
SDA0:	Serial data
SI30:	Serial input
SO30:	Serial output
TI00, TI01, TI50, TI51:	Timer input
TO0, TO50, TO51:	Timer output
TxD0:	Transmit data
V_{DD0}, V_{DD1}:	Power supply
V_{PP}:	Programming power supply
V_{SS0}, V_{SS1}:	Ground
$\overline{WAIT}$:	Wait
$\overline{WR}$:	Write strobe
X1, X2:	Crystal (main system clock)
XT1, XT2:	Crystal (subsystem clock)

★ 2.6 78K/0 Series Lineup

The products in the 78K/0 Series are listed below. The names enclosed in boxes are subseries name.

Products in mass production

Products under development

Y subseries products are compatible with I²C bus.

78K/0 Series

Control

100-pin	μPD78075B		EMI-noise reduced version of the μPD78078
100-pin	μPD78078	μPD78078Y	μPD78054 with timer and enhanced external interface
100-pin	μPD78070A	μPD78070AY	ROMless version of the μPD78078
100-pin		μPD780018AY	μPD78078Y with enhanced serial I/O and limited functions
80-pin	μPD780058	μPD780058Y	μPD78054 with enhanced serial I/O
80-pin	μPD78058F	μPD78058FY	EMI-noise reduced version of the μPD78054
80-pin	μPD78054	μPD78054Y	μPD78018F with UART and D/A converter, and enhanced I/O
80-pin	μPD780065		μPD780024A with expanded RAM
64-pin	μPD780078	μPD780078Y	μPD780034A with timer and enhanced serial I/O
64-pin	μPD780034A	μPD780034AY	μPD780024A with enhanced A/D converter
64-pin	μPD780024A	μPD780024AY	μPD78018F with enhanced serial I/O
52-pin	μPD780034AS		52-pin version of the μPD780034A
52-pin	μPD780024AS		52-pin version of the μPD780024A
64-pin	μPD78014H		EMI-noise reduced version of the μPD78018F
64-pin	μPD78018F	μPD78018FY	Basic subseries for control
42/44-pin	μPD78083		On-chip UART, capable of operating at low voltage (1.8 V)

Inverter control

64-pin	μPD780988		On-chip inverter controller and UART. EMI-noise reduced.

VFD drive

100-pin	μPD780208		μPD78044F with enhanced I/O and VFD C/D. Display output total: 53
80-pin	μPD780232		For panel control. On-chip VFD C/D. Display output total: 53
80-pin	μPD78044H		μPD78044F with N-ch open-drain I/O. Display output total: 34
80-pin	μPD78044F		Basic subseries for driving VFD. Display output total: 34

LCD drive

100-pin	μPD780354	μPD780354Y	μPD780344 with enhanced A/D converter
100-pin	μPD780344	μPD780344Y	μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780338		μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780328		μPD780308 with enhanced display function and timer. Segment signal output: 32 pins max.
120-pin	μPD780318		μPD780308 with enhanced display function and timer. Segment signal output: 24 pins max.
100-pin	μPD780308	μPD780308Y	μPD78064 with enhanced SIO, and expanded ROM and RAM
100-pin	μPD78064B		EMI-noise reduced version of the μPD78064
100-pin	μPD78064	μPD78064Y	Basic subseries for driving LCDs, on-chip UART

Bus interface supported

100-pin	μPD780948		On-chip CAN controller
80-pin	μPD78098B		μPD78054 with IEBus controller
80-pin		μPD780702Y	On-chip IEBus controller
80-pin		μPD780703AY	On-chip CAN controller
80-pin		μPD780833Y	On-chip controller compliant with J1850 (Class 2)
64-pin	μPD780816		Specialized for CAN controller function

Meter control

100-pin	μPD780958		For industrial meter control
80-pin	μPD780852		On-chip automobile meter controller/driver
80-pin	μPD780828B		For automobile meter driver. On-chip CAN controller

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.

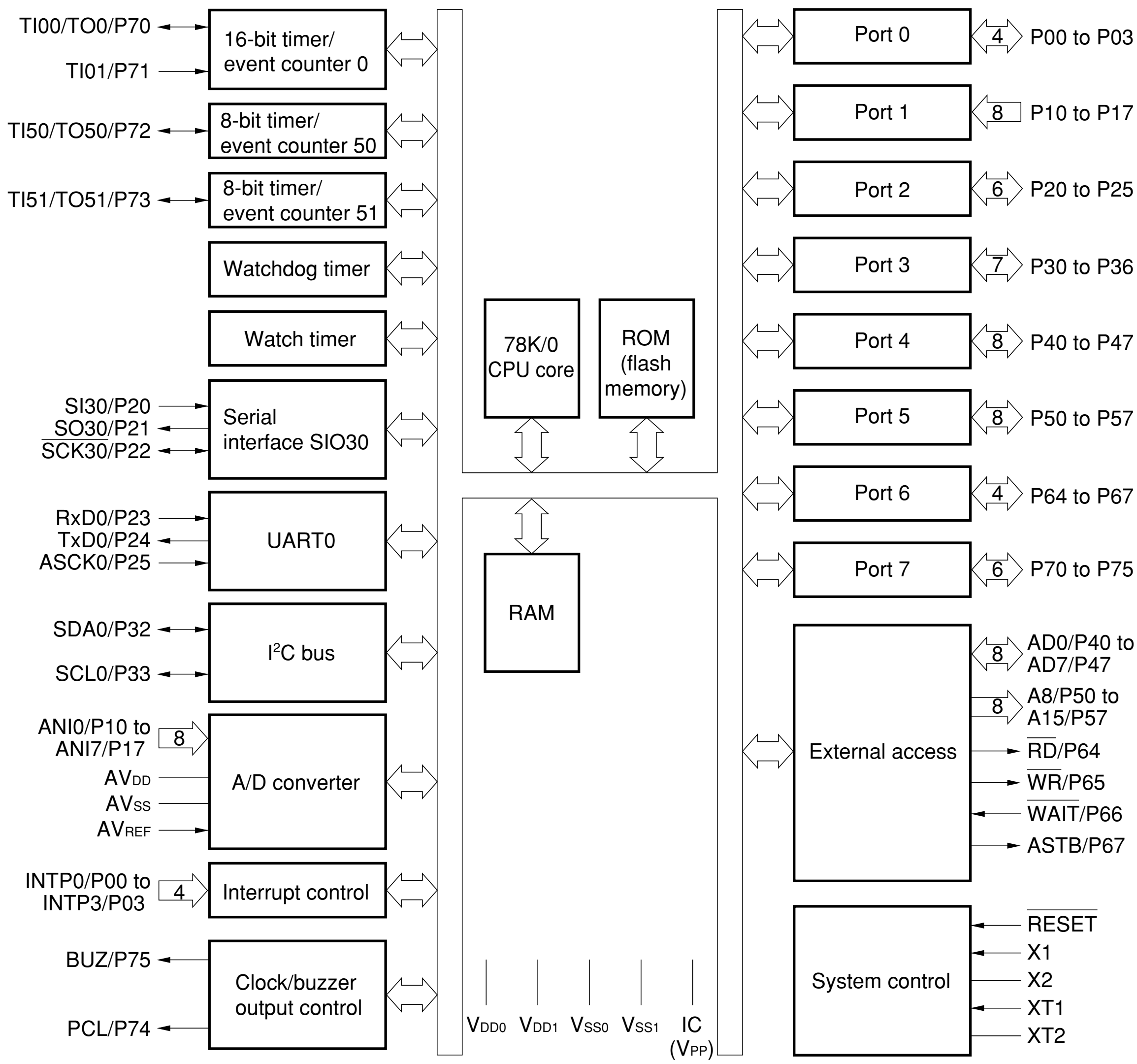
The major functional differences between the subseries are shown below.

- **Subseries with the suffix Y**

Subseries Name \ Function		ROM Capacity	Timer				8-Bit A/D	10-Bit A/D	8-Bit D/A	Serial Interface	I/O	V_{DD} MIN. Value	External Expansion
			8-Bit	16-Bit	Watch	WDT							
Control	μPD78078Y	48 KB to 60 KB	4 ch	1 ch	1 ch	1 ch	8 ch	–	2 ch	3 ch (UART: 1 ch, I^2C: 1 ch)	88	1.8 V	Yes
	μPD78070AY	–									61	2.7 V	
	μPD780018AY	48 KB to 60 KB							–	3 ch (I^2C: 1 ch)	88		
	μPD780058Y	24 KB to 60 KB	2 ch						2 ch	3 ch (time-division UART: 1 ch, I^2C: 1 ch)	68	1.8 V	
	μPD78058FY	48 KB to 60 KB								3 ch (UART: 1 ch, I^2C: 1 ch)	69	2.7 V	
	μPD78054Y	16 KB to 60 KB										2.0 V	
	μPD780078Y	48 KB to 60 KB		2 ch			–	8 ch	–	4 ch (UART: 2 ch, I^2C: 1 ch)	52	1.8 V	
	μPD780034AY	8 KB to 32 KB		1 ch						3 ch (UART: 1 ch, I^2C: 1 ch)	51		
	μPD780024AY						8 ch	–					
	μPD78018FY	8 KB to 60 KB								2 ch (I^2C: 1 ch)	53		
LCD drive	μPD780354Y	24 KB to 32 KB	4 ch	1 ch	1 ch	1 ch	–	8 ch	–	4 ch (UART: 1 ch, I^2C: 1 ch)	66	1.8 V	–
	μPD780344Y						8 ch	–					
	μPD780308Y	48 KB to 60 KB	2 ch							3 ch (time-division UART: 1 ch, I^2C: 1 ch)	57	2.0 V	
	μPD78064Y	16 KB to 32 KB								2 ch (UART: 1 ch, I^2C: 1 ch)			
Bus interface supported	μPD780702Y	60 KB	3 ch	2 ch	1 ch	1 ch	16 ch	–	–	4 ch (UART: 1 ch, I^2C: 1 ch)	67	3.5 V	–
	μPD780703AY	59.5 KB											
	μPD780833Y	60 KB									65	4.5 V	

Remark The functions of the subseries without the suffix Y and the subseries with the suffix Y are the same, except for the serial interface (if a subseries without the suffix Y is available).

2.7 Block Diagram



Remarks 1. The internal ROM and RAM capacities depend on the product.

2. Pin connection in parentheses is intended for the μPD78F0034AY, 78F0034BY.

2.8 Outline of Function

Item		μPD780021AY μPD780031AY	μPD780022AY μPD780032AY	μPD780023AY μPD780033AY	μPD780024AY μPD780034AY	μPD78F0034AY μPD78F0034BY
Internal memory	ROM	8 KB (Mask ROM)	16 KB (Mask ROM)	24 KB (Mask ROM)	32 KB (Mask ROM)	32 KB Note (Flash memory)
	High-speed RAM	512 bytes		1024 bytes		1024 bytes Note
Memory space		64 KB				
General-purpose register		8 bits × 32 registers (8 bits × 8 registers × 4 banks)				
Minimum instruction execution time		Minimum instruction execution time changeable function				
	When main system clock selected	0.238 μs/0.477 μs/0.954 μs/1.90 μs/3.81 μs (@ 8.38 MHz operation)				
	When subsystem clock selected	122 μs (@ 32.768 kHz operation)				
Instruction set		• 16-bit operation • Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits) • Bit manipulate (set, reset, test, and Boolean operation) • BCD adjust, etc.				
I/O port		Total: 51 • CMOS input: 8 • CMOS I/O: 39 • N-ch open-drain I/O (5 V breakdown): 4				
A/D converter		• 8-bit resolution × 8 channels (μPD780021AY, 780022AY, 780023AY, 780024AY) • 10-bit resolution × 8 channels (μPD780031AY, 780032AY, 780033AY, 780034AY, 78F0034AY, 78F0034BY) • Low-voltage operation: AV_{DD} = 1.8 to 5.5 V				
Serial interface		• 3-wire serial I/O mode: 1 channel • UART mode: 1 channel • I^2C bus mode: 1 channel				
Timer		• 16-bit timer/event counter: 1 channel • 8-bit timer/event counter: 2 channels • Watch timer: 1 channel • Watchdog timer: 1 channel				
Timer output		Three outputs (8-bit PWM output enable: 2)				
Clock output		• 65.5 kHz, 131 kHz, 262 kHz, 524 kHz, 1.05 MHz, 2.10 MHz, 4.19 MHz, 8.38 MHz (8.38 MHz with main system clock) • 32.768 kHz (32.768 kHz with subsystem clock)				
Buzzer output		1.02 kHz, 2.05 kHz, 4.10 kHz, 8.19 kHz (8.38 MHz with main system clock)				
Vectored interrupt source	Maskable	Internal: 13, External: 5				
	Non-maskable	Internal: 1				
	Software	1				
Power supply voltage		V_{DD} = 1.8 to 5.5 V				
Operating ambient temperature		T_A = –40 to +85°C				
Package		• 64-pin plastic SDIP (19.05 mm (750)) • 64-pin plastic QFP (14 × 14) • 64-pin plastic LQFP (14 × 14) • 64-pin plastic TQFP (12 × 12) • 64-pin plastic LQFP (10 × 10) • 73-pin plastic FBGA (9 × 9) (standard grade product only)				

Note The capacities of internal flash memory and internal high-speed RAM can be changed by means of the memory size switching register (IMS).

The outline of the timer/event counter is as follows (for details, see **CHAPTER 8 16-BIT TIMER/EVENT COUNTER 0**, **CHAPTER 9 8-BIT TIMER/EVENT COUNTERS 50, 51**, **CHAPTER 10 WATCH TIMER**, and **CHAPTER 11 WATCHDOG TIMER**).

		16-Bit Timer/ Event Counter 0	8-Bit Timer/ Event Counters 50, 51	Watch Timer	Watchdog Timer
Operation Mode	Interval timer	1 channel	2 channels	1 channel[Note 1]	1 channel[Note 2]
	External event counter	√	√	–	–
Function	Timer output	√	√	–	–
	PPG output	√	–	–	–
	PWM output	–	√	–	–
	Pulse width measurement	√	–	–	–
	Square-wave output	√	√	–	–
	Interrupt request	√	√	√	√

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.

★ 2.9 Correspondence Between Mask ROM Versions and Flash Memory Versions

Table 2-1. Correspondence Between Mask ROM Versions and Flash Memory Versions

Mask ROM Version / Flash Memory Version	μPD780021AY/2AY/3AY/4AY μPD780031AY/2AY/3AY/4AY	μPD780021AY(A)/2AY(A)/3AY(A)/4AY(A) μPD780031AY(A)/2AY(A)/3AY(A)/4AY(A)
μPD78F0034AY	√	–
μPD78F0034BY	√	–
μPD78F0034BY(A)	–	√

Remarks 1. √: Supported, –: Not supported

2. The μPD780034AY and μPD78F0034BY, 78F0034BY(A) differ in the communication mode of flash memory programming. See **23.1 Differences Between μPD78F0034A, 78F0034AY and μPD78F0034B, 78F0034BY**.

3. Expanded-specification products of the μPD780024AY and 780034AY Subseries are not available. Only conventional products are available.

4. A special grade product of the μPD78F0034AY is not available. Only a standard grade product is available.

★ 2.10 Differences Between Standard Grade Products and Special Grade Products

The differences between standard grade products (μPD780021AY, 780022AY, 780023AY, 780024AY, 780031AY, 780032AY, 780033AY, 780034AY, 78F0034AY, 78F0034BY) and special grade products (μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A), 780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A), 78F0034BY(A)) are shown in Table 2-2.

Table 2-2. Differences Between Standard Grade Products and Special Grade Products

	μPD780021AY, 780022AY, 780023AY, 780024AY, 780031AY, 780032AY, 780033AY, 780034AY, 78F0034AY, 78F0034BY	μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A), 780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A), 78F0034BY(A)
Quality grade	Standard	Special
Package	See **2.11 Correspondence Between Products and Packages**	
Other (functions, electrical specifications, etc.)	Same	

★ 2.11 Correspondence Between Products and Packages

The following table shows the correspondence between the products and packages.

Table 2-3. Correspondence Between Products and Packages

	Mask ROM Version		Flash Memory Version		
	μPD780021AY/2AY/3AY/4AY μPD780031AY/2AY/3AY/4AY		μPD78F0034AY	μPD78F0034BY	
	Standard	Special	Standard	Standard	Special
64-pin SDIP (CW type)	√	√Note 1	√	–	–
64-pin QFP (GC-AB8 type)	√	√Note 2	√	–	–
64-pin LQFP (GC-8BS type)	√	√Note 1	√	√	√
64-pin TQFP (GK-9ET type)	√	√Note 1	√	√	√
64-pin LQFP (GB-8EU type)	√	√Note 1	√	√	√
73-pin FBGA (F1-CN3 type)	√	–	–	√	–

Notes 1. Under development

2. Only the μPD780022AYGC(A)-AB8 is under mass production. The other models are still under development.

Remarks 1. √: Package available, –: Package not available

2. A special grade product of the μPD78F0034AY is not available. Only a standard grade product is available.

2.12 Mask Options

The mask ROM versions (μPD780021AY, 780022AY, 780023AY, 780024AY, 780031AY, 780032AY, 780033AY, 780034AY) 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 device production. Using the 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 μPD780024AY and 780034AY Subseries are shown in Table 2-4.

Table 2-4. Mask Options of Mask ROM Versions

Pin Names	Mask Option
P30, P31	Pull-up resistor connection can be specified in 1-bit units.

3.1 Pin Function List

(1) Port pins (1/2)

Pin Name	I/O	Function		After Reset	Alternate Function
P00	I/O	Port 0 4-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		Input	INTP0
P01					INTP1
P02					INTP2
P03					INTP3/ADTRG
P10 to P17	Input	Port 1 8-bit input-only port.		Input	ANI0 to ANI7
P20	I/O	Port 2 6-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		Input	SI30
P21					SO30
P22					$\overline{\text{SCK30}}$
P23					RxD0
P24					TxD0
P25					ASCK0
P30	I/O	Port 3 7-bit I/O port Input/output mode can be specified in 1-bit units.	N-ch open-drain I/O port On-chip pull-up resistor can be specified by mask option (mask ROM version only). LEDs can be driven directly.	Input	–
P31					
P32					
P33					
P34			An on-chip pull-up resistor can be used by software settings.		SI31
P35					SO31
P36					$\overline{\text{SCK31}}$
P40 to P47	I/O	Port 4 8-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings. Interrupt request 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 mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		Input	A8 to A15
P64	I/O	Port 6 4-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		Input	$\overline{\text{RD}}$
P65					$\overline{\text{WR}}$
P66					$\overline{\text{WAIT}}$
P67					ASTB

(1) Port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
P70	I/O	Port 7 6-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.	Input	TI00/TO0
P71				TI01
P72				TI50/TO50
P73				TI51/TO51
P74				PCL
P75				BUZ

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

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request input with specifiable valid edges (rising edge, falling edge, both rising and falling edges)	Input	P00
INTP1				P01
INTP2				P02
INTP3				P03/ADTRG
SI30	Input	Serial interface serial data input	Input	P20
SI31				P34
SO30	Output	Serial interface serial data output	Input	P21
SO31				P35
$\overline{SCK30}$	I/O	Serial interface serial clock input/output	Input	P22
$\overline{SCK31}$				P36
RxD0	Input	Asynchronous serial interface serial data input	Input	P23
TxD0	Output	Asynchronous serial interface serial data output	Input	P24
ASCK0	Input	Asynchronous serial interface serial clock input	Input	P25
TI00	Input	External count clock input to 16-bit timer/event counter 0 Capture trigger input to 16-bit timer/event counter 0 capture register (CR00, CR01)	Input	P70/TO0
TI01		Capture trigger input to 16-bit timer/event counter 0 capture register (CR00)		P71
TI50		External count clock input to 8-bit timer/event counter 50		P72/TO50
TI51		External count clock input to 8-bit timer/event counter 51		P73/TO51
TO0	Output	16-bit timer/event counter 0 output	Input	P70/TI00
TO50		8-bit timer/event counter 50 output (also used for 8-bit PWM output)	Input	P72/TI50
TO51		8-bit timer/event counter 51 output (also used for 8-bit PWM output)		P73/TI51
PCL	Output	Clock output (for main system clock and subsystem clock trimming)	Input	P74
BUZ	Output	Buzzer output	Input	P75
AD0 to AD7	I/O	Lower address/data bus when expanding external memory	Input	P40 to P47

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

Pin Name	I/O	Function	After Reset	Alternate Function
A8 to A15	Output	Higher address bus when expanding external memory	Input	P50 to P57
$\overline{RD}$	Output	Strobe signal output for read operation from external memory	Input	P64
$\overline{WR}$		Strobe signal output for write operation from external memory		P65
$\overline{WAIT}$	Input	Wait insertion when accessing external memory	Input	P66
ASTB	Output	Strobe output externally latching address information output to ports 4, 5 to access external memory	Input	P67
ANI0 to ANI7	Input	A/D converter analog input	Input	P10 to P17
ADTRG	Input	A/D converter trigger signal input	Input	P03/INTP3
AV_{REF}	Input	A/D converter reference voltage input	–	–
AV_{DD}	–	A/D converter analog power supply. Connect to V_{DD0} or V_{DD1}.	–	–
AV_{SS}	–	A/D converter ground potential. Connect to V_{SS0} or V_{SS1}.	–	–
$\overline{RESET}$	Input	System reset input	Input	–
X1	Input	Crystal/ceramic connection for main system clock oscillation	–	–
X2	–		–	–
XT1	Input	Crystal connection for subsystem clock oscillation	–	–
XT2	–		–	–
V_{DD0}	–	Positive power supply for ports	–	–
V_{DD1}	–	Positive power supply other than ports	–	–
V_{SS0}	–	Ground potential for ports	–	–
V_{SS1}	–	Ground potential other than ports	–	–
IC	–	Internally connected. Connect directly to V_{SS0} or V_{SS1}.	–	–
★ NC[Note]	–	Not internally connected. Leave open.	–	–
V_{PP}	–	High-voltage application for program write/verify.	–	–

Note The NC pin is available only for a 73-pin plastic FBGA.

3.2 Description of Pin Functions

3.2.1 P00 to P03 (Port 0)

These are 4-bit I/O ports. Besides serving as I/O ports, they function as an external interrupt input, and A/D converter external trigger input.

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

(1) Port mode

These ports function as 4-bit I/O ports.

P00 to P03 can be specified as input or output ports in 1-bit units with port mode register 0 (PM0). On-chip pull-up resistors can be used by setting pull-up resistor option register 0 (PU0).

(2) Control mode

These ports function as an external interrupt request input, and A/D converter external trigger input.

(a) INTP0 to INTP3

INTP0 to INTP3 are external interrupt request input pins which can specify valid edges (rising edge, falling edge, and both rising and falling edges).

(b) ADTRG

A/D converter external trigger input pin.

Caution **When P03 is used as an A/D converter external trigger input, specify the valid edge by bits 1, 2 (EGA00, EGA01) of A/D converter mode register (ADM0) and set interrupt mask flag (PMK3) to 1.**

3.2.2 P10 to P17 (Port 1)

These are 8-bit input-only ports. Besides serving as input ports, they function as an A/D converter analog input.

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

(1) Port mode

These ports function as 8-bit input-only ports.

(2) Control mode

These ports function as A/D converter analog input pins (ANI0 to ANI7).

3.2.3 P20 to P25 (Port 2)

These are 6-bit I/O ports. Besides serving as I/O ports, they function as data I/O and clock I/O of serial interface SIO30 or UART0.

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

(1) Port mode

These ports function as 6-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 2 (PM2). On-chip pull-up resistors can be used by setting pull-up resistor option register 2 (PU2).

(2) Control mode

These ports function as data I/O and clock I/O of serial interface SIO30 or UART0.

(a) SI30 and SO30

Serial data I/O pins of serial interface SIO30.

(b) $\overline{\text{SCK30}}$

Serial clock I/O pin of serial interface SIO30.

(c) RxD0 and TxD0

Serial data I/O pins of serial interface UART0.

(d) ASCK0

Serial clock input pin of serial interface UART0.

3.2.4 P30 to P36 (Port 3)

These are 7-bit I/O ports. Besides serving as I/O ports, they function as data I/O and clock I/O of serial interface SIO31.

P30 to P33 can drive LEDs directly.

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

(1) Port mode

These ports function as 7-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 3 (PM3). P30 to P33 are N-ch open-drain I/O port. On-chip pull-up resistor can be used by mask option (mask ROM version only). On-chip pull-up resistors of P34 to P36 can be used by setting pull-up resistor option register 3 (PU3).

(2) Control mode

These ports function as data I/O and clock I/O of serial interface SIO31.

(a) SI31 and SO31

Serial data I/O pins of serial interface SIO31.

(b) $\overline{\text{SCK31}}$

Serial clock I/O pin of serial interface SIO31.

3.2.5 P40 to P47 (Port 4)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as an address/data bus.
The interrupt request flag (KRIF) can be set to 1 by detecting a falling edge.
The following operating mode can be specified in 1-bit units.

Caution When using the falling edge detection interrupt (INTKR), be sure to set the memory expansion mode register (MEM) to 01H.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 4 (PM4). On-chip pull-up resistors can be used by setting pull-up resistor option register 4 (PU4).

(2) Control mode

These ports function as lower address/data bus pins (AD0 to AD7) in external memory expansion mode.

3.2.6 P50 to P57 (Port 5)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as an address bus.
Port 5 can drive LEDs directly.
The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 5 (PM5). On-chip pull-up resistors can be used by setting pull-up resistor option register 5 (PU5).

(2) Control mode

These ports function as higher address bus pins (A8 to A15) in external memory expansion mode.

3.2.7 P64 to P67 (Port 6)

These are 4-bit I/O ports. Besides serving as I/O ports, they are used for control in external memory expansion mode.
The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 4-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 6 (PM6).
On-chip pull-up resistors can be used by setting pull-up resistor option register 6 (PU6).

(2) Control mode

These ports function as control signal output pins ($\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WAIT}}$, ASTB) in external memory expansion mode.

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

3.2.8 P70 to P75 (Port 7)

These are 6-bit I/O ports. Besides serving as I/O ports, they function as a timer I/O, clock output, and buzzer output. The following operating modes can be specified in 1-bit units.

(1) Port mode

Port 7 functions as a 6-bit I/O port. They can be specified as an input port or output port in 1-bit units with port mode register 7 (PM7). On-chip pull-up resistors can be used by setting pull-up resistor option register 7 (PU7). P70 and P71 are also 16-bit timer/event counter 0 capture trigger signal input pins with a valid edge input.

(2) Control mode

Port 7 functions as timer I/O, clock output, and buzzer output.

(a) TI00

External count clock input pin to 16-bit timer/event counter 0 and capture trigger signal input pin to 16-bit timer/event counter capture register (CR01).

(b) TI01

Capture trigger signal input pin to 16-bit timer/event counter 0 capture register (CR00).

(c) TI50 and TI51

External count clock input pins to 8-bit timer/event counters 50 and 51.

(d) TO0, TO50, and TO51

Timer output pins.

(e) PCL

Clock output pin.

(f) BUZ

Buzzer output pin.

3.2.9 AV_{REF}

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

When no A/D converter is used, connect this pin directly to V_{SS0} or V_{SS1}.

3.2.10 AV_{DD}

This is an analog power supply pin of A/D converter. Always use the same potential as that of the V_{DD0} pin or V_{DD1} pin even when no A/D converter is used.

3.2.11 AV_{SS}

This is a ground potential pin of A/D converter. Always use the same potential as that of the V_{SS0} pin or V_{SS1} pin even when no A/D converter is used.

3.2.12 $\overline{\text{RESET}}$

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

★ 3.2.13 NC

NC (Non-connection) pin is not internally connected. Leave this pin open.

3.2.14 X1 and X2

Crystal/ceramic resonator connection pins for main system clock oscillation.
For external clock supply, input clock signal to X1 and its inverted signal to X2.

3.2.15 XT1 and XT2

Crystal resonator connection pins for subsystem clock oscillation.
For external clock supply, input the clock signal to XT1 and its inverted signal to XT2.

3.2.16 V_{DD0} and V_{DD1}

V_{DD0} is a positive power supply port pin.
V_{DD1} is a positive power supply pin other than port pin.

3.2.17 V_{SS0} and V_{SS1}

V_{SS0} is a ground potential port pin.
V_{SS1} is a ground potential pin other than port pin.

3.2.18 V_{PP} (flash memory versions only)

High-voltage apply pin for flash memory programming mode setting and program write/verify.
★ Handle in either of the following ways.

- Independently connect a 10 kΩ pull-down resistor.
- Set the jumper on the board so that this pin is connected directly to the dedicated flash programmer in programming mode and directly to V_{SS0} or V_{SS1} in normal operation mode.

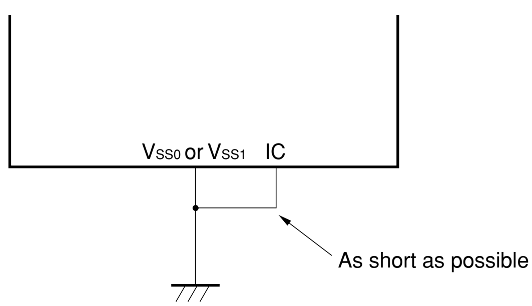
When there is a potential difference between the V_{PP} pin and V_{SS0} pin or V_{SS1} pin because the wiring between the two pins is too long or external noise is input to the V_{PP} pin, the user program may not operate normally.

3.2.19 IC (mask ROM version only)

The IC (Internally Connected) pin is provided to set the test mode to check the μPD780024A, 780034A Subseries at delivery. Connect it directly to the V_{SS0} or V_{SS1} pin with the shortest possible wire in the normal operating mode.

When a potential difference is produced between the IC pin and V_{SS0} pin or V_{SS1} 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 operate normally.

- **Connect IC pins to V_{SS0} pins or V_{SS1} pins directly.**



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

Table 3-1 shows the types of pin I/O circuit and the recommended connections of unused pins.
See 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 to P02/INTP2	8-C	I/O	Input: Independently connect to V_{SS0} or V_{SS1} via a resistor. Output: Leave open.
P03/INTP3/ADTRG			
P10/ANI0 to P17/ANI7	25	Input	Connect directly to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1}.
P20/SI30	8-C	I/O	Input: Independently connect to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1} via a resistor. Output: Leave open.
P21/SO30	5-H		
P22/$\overline{\text{SCK30}}$	8-C		
P23/RxD0			
P24/TxD0	5-H		
P25/ASCK0	8-C		
P30, P31 (for mask ROM version)	13-Q		Input: Connect directly to V_{SS0} or V_{SS1}. Output: Set the output latch of the port to 0, and leave these pins open at low level.
P30, P31 (for flash memory version)	13-P		
P32, P33 (for mask ROM version)	13-S		
P32, P33 (for flash memory version)	13-R		
P34/SI31	8-C		Input: Independently connect to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1} via a resistor. Output: Leave open.
P35/SO31	5-H		
P36/$\overline{\text{SCK31}}$	8-C		
P40/AD0 to P47/AD7	5-H		Input: Independently connect to V_{DD0} or V_{DD1} 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
P50/A8 to P57/A15	5-H	I/O	Input: Independently connect to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1} via a resistor. Output: Leave open.
P64/$\overline{RD}$			
P65/$\overline{WR}$			
P66/$\overline{WAIT}$			
P67/ASTB			
P70/TI00/TO0	8-C		
P71/TI01			
P72/TI50/TO50			
P73/TI51/TO51			
P74/PCL	5-H		
P75/BUZ			
$\overline{RESET}$	2	Input	–
NC[Note]	–	–	Leave open.
XT1	16	Input	Connect directly to V_{DD0} or V_{DD1}.
XT2		–	Leave open.
AV_{DD}	–		Connect directly to V_{DD0} or V_{DD1}.
AV_{REF}			Connect directly to V_{SS0} or V_{SS1}.
AV_{SS}			
IC (for mask ROM version)			
V_{PP} (for flash memory version)			Independently connect a 10 kΩ pull-down resistor to this pin, or connect directly to V_{SS0} or V_{SS1}.

Note The NC pin is available only for a 73-pin plastic FBGA.

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

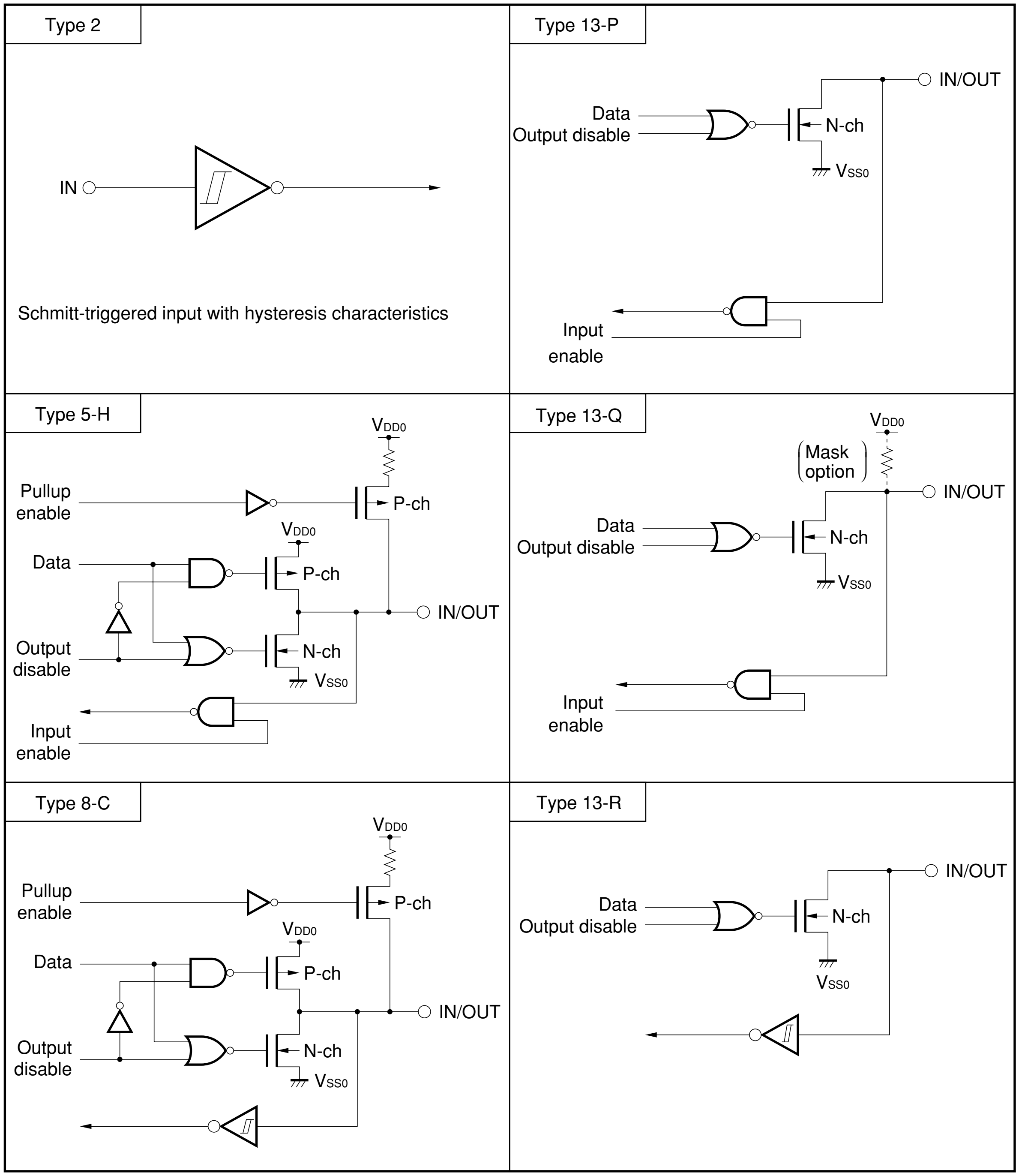
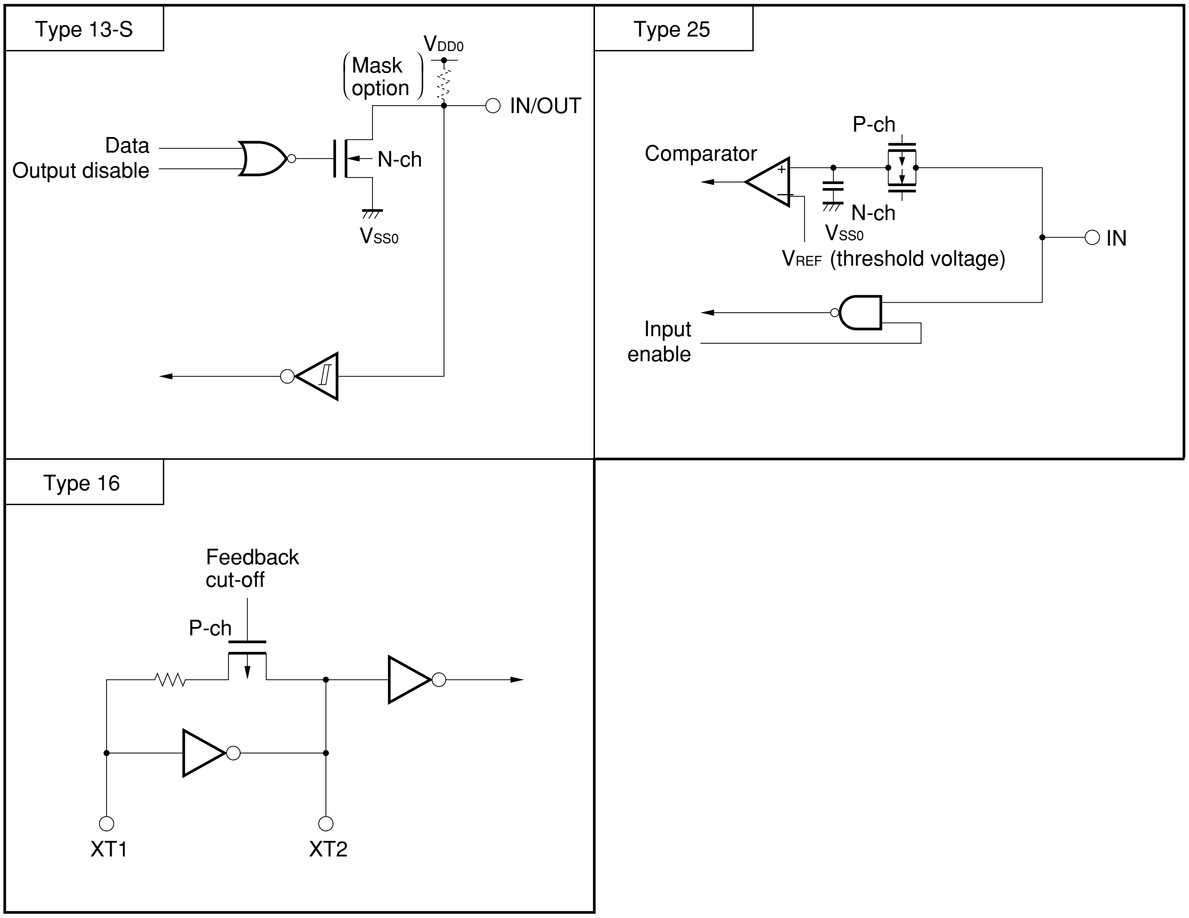


Figure 3-1. Pin I/O Circuit List (2/2)



4.1 Pin Function List

(1) Port pins (1/2)

<table>
<tr><th>Pin Name</th><th>I/O</th><th colspan="2">Function</th><th>After Reset</th><th>Alternate Function</th></tr>
<tr><td>P00</td><td rowspan="4">I/O</td><td colspan="2" rowspan="4">Port 0
4-bit I/O port
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td rowspan="4">Input</td><td>INTP0</td></tr>
<tr><td>P01</td><td>INTP1</td></tr>
<tr><td>P02</td><td>INTP2</td></tr>
<tr><td>P03</td><td>INTP3/ADTRG</td></tr>
<tr><td>P10 to P17</td><td>Input</td><td colspan="2">Port 1
8-bit input-only port.</td><td>Input</td><td>ANI0 to ANI7</td></tr>
<tr><td>P20</td><td rowspan="6">I/O</td><td colspan="2" rowspan="6">Port 2
6-bit I/O port
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td rowspan="6">Input</td><td>SI30</td></tr>
<tr><td>P21</td><td>SO30</td></tr>
<tr><td>P22</td><td>$\overline{\text{SCK30}}$</td></tr>
<tr><td>P23</td><td>RxD0</td></tr>
<tr><td>P24</td><td>TxD0</td></tr>
<tr><td>P25</td><td>ASCK0</td></tr>
<tr><td>P30</td><td rowspan="7">I/O</td><td rowspan="7">Port 3
7-bit I/O port
Input/output mode can be specified in 1-bit units.</td><td rowspan="4">N-ch open-drain I/O port
On-chip pull-up resistor can be specified by mask option
(P30 and P31 are mask ROM version only).
LEDs can be driven directly.</td><td rowspan="7">Input</td><td rowspan="2">–</td></tr>
<tr><td>P31</td></tr>
<tr><td>P32</td><td>SDA0</td></tr>
<tr><td>P33</td><td>SCL0</td></tr>
<tr><td>P34</td><td rowspan="3">An on-chip pull-up resistor can be used by software settings.</td><td rowspan="3">–</td></tr>
<tr><td>P35</td></tr>
<tr><td>P36</td></tr>
<tr><td>P40 to P47</td><td>I/O</td><td colspan="2">Port 4
8-bit I/O port
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.
Interrupt request flag (KRIF) is set to 1 by falling edge detection.</td><td>Input</td><td>AD0 to AD7</td></tr>
<tr><td>P50 to P57</td><td>I/O</td><td colspan="2">Port 5
8-bit I/O port
LEDs can be driven directly.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td>Input</td><td>A8 to A15</td></tr>
<tr><td>P64</td><td rowspan="4">I/O</td><td colspan="2" rowspan="4">Port 6
4-bit I/O port
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td rowspan="4">Input</td><td>$\overline{\text{RD}}$</td></tr>
<tr><td>P65</td><td>$\overline{\text{WR}}$</td></tr>
<tr><td>P66</td><td>$\overline{\text{WAIT}}$</td></tr>
<tr><td>P67</td><td>ASTB</td></tr>
</table>

(1) Port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
P70	I/O	Port 7 6-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.	Input	TI00/TO0
P71				TI01
P72				TI50/TO50
P73				TI51/TO51
P74				PCL
P75				BUZ

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

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request input with specifiable valid edges (rising edge, falling edge, both rising and falling edges)	Input	P00
INTP1				P01
INTP2				P02
INTP3				P03/ADTRG
SI30	Input	Serial interface serial data input	Input	P20
SO30	Output	Serial interface serial data output	Input	P21
SDA0	I/O	Serial interface serial data input/output	Input	P32
$\overline{SCK30}$	I/O	Serial interface serial clock input/output	Input	P22
SCL0				P33
RxD0	Input	Asynchronous serial interface serial data input	Input	P23
TxD0	Output	Asynchronous serial interface serial data output	Input	P24
ASCK0	Input	Asynchronous serial interface serial clock input	Input	P25
TI00	Input	External count clock input to 16-bit timer/event counter 0 Capture trigger input to 16-bit timer/event counter 0 capture register (CR00, CR01)	Input	P70/TO0
TI01		Capture trigger input to 16-bit timer/event counter 0 capture register (CR00)		P71
TI50		External count clock input to 8-bit timer/event counter 50		P72/TO50
TI51		External count clock input to 8-bit timer/event counter 51		P73/TO51
TO0	Output	16-bit timer/event counter 0 output	Input	P70/TI00
TO50		8-bit timer/event counter 50 output (also used for 8-bit PWM output)	Input	P72/TI50
TO51		8-bit timer/event counter 51 output (also used for 8-bit PWM output)		P73/TI51
PCL	Output	Clock output (for main system clock and subsystem clock trimming)	Input	P74
BUZ	Output	Buzzer output	Input	P75
AD0 to AD7	I/O	Lower address/data bus when expanding external memory	Input	P40 to P47
A8 to A15	Output	Higher address bus when expanding external memory	Input	P50 to P57
$\overline{RD}$	Output	Strobe signal output for read operation from external memory	Input	P64
$\overline{WR}$		Strobe signal output for write operation from external memory		P65

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

Pin Name	I/O	Function	After Reset	Alternate Function
$\overline{\text{WAIT}}$	Input	Wait insertion when accessing external memory	Input	P66
ASTB	Output	Strobe output externally latching address information output to ports 4, 5 to access external memory	Input	P67
ANI0 to ANI7	Input	A/D converter analog input	Input	P10 to P17
ADTRG	Input	A/D converter trigger signal input	Input	P03/INTP3
AV_{REF}	Input	A/D converter reference voltage input	–	–
AV_{DD}	–	A/D converter analog power supply. Connect to V_{DD0} or V_{DD1}.	–	–
AV_{SS}	–	A/D converter ground potential. Connect to V_{SS0} or V_{SS1}.	–	–
$\overline{\text{RESET}}$	Input	System reset input	Input	–
X1	Input	Crystal connection for main system clock oscillation	–	–
X2	–		–	–
XT1	Input	Crystal connection for subsystem clock oscillation	–	–
XT2	–		–	–
V_{DD0}	–	Positive power supply for ports	–	–
V_{DD1}	–	Positive power supply other than ports	–	–
V_{SS0}	–	Ground potential for ports	–	–
V_{SS1}	–	Ground potential other than ports	–	–
IC	–	Internally connected. Connect directly to V_{SS0} or V_{SS1}.	–	–
★ NC[Note]	–	Not internally connected. Leave open.	–	–
V_{PP}	–	High-voltage application for program write/verify.	–	–

Note The NC pin is available only for a 73-pin plastic FBGA.

4.2 Description of Pin Functions

4.2.1 P00 to P03 (Port 0)

These are 4-bit I/O ports. Besides serving as I/O ports, they function as an external interrupt input, and A/D converter external trigger input.

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

(1) Port mode

These ports function as 4-bit I/O ports.

P00 to P03 can be specified as input or output ports in 1-bit units with port mode register 0 (PM0). On-chip pull-up resistors can be used by setting pull-up resistor option register 0 (PU0).

(2) Control mode

These ports function as an external interrupt request input, and A/D converter external trigger input.

(a) INTP0 to INTP3

INTP0 to INTP3 are external interrupt request input pins which can specify valid edges (rising edge, falling edge, and both rising and falling edges).

(b) ADTRG

A/D converter external trigger input pin.

Caution When P03 is used as an A/D converter external trigger input, specify the valid edge by bits 1, 2 (EGA00, EGA01) of A/D converter mode register (ADM0) and set interrupt mask flag (PMK3) to 1.

4.2.2 P10 to P17 (Port 1)

These are 8-bit input-only ports. Besides serving as input ports, they function as an A/D converter analog input.

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

(1) Port mode

These ports function as 8-bit input-only ports.

(2) Control mode

These ports function as A/D converter analog input pins (ANI0 to ANI7).

4.2.3 P20 to P25 (Port 2)

These are 6-bit I/O ports. Besides serving as I/O ports, they 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

These ports function as 6-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 2 (PM2). On-chip pull-up resistors can be used by setting pull-up resistor option register 2 (PU2).

(2) Control mode

These ports function as data I/O and clock I/O of serial interface SIO30 or UART0.

(a) SI30 and SO30

Serial data I/O pins of serial interface SIO30.

(b) $\overline{\text{SCK30}}$

Serial clock I/O pin of serial interface SIO30.

(c) RxD0 and TxD0

Serial data I/O pins of serial interface UART0.

(d) ASCK0

Serial clock input pin of serial interface UART0.

4.2.4 P30 to P36 (Port 3)

These are 7-bit I/O ports. Besides serving as I/O ports, they function as data I/O and clock I/O of serial interface IIC0.

P30 to P33 can drive LEDs directly.

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

(1) Port mode

These ports function as 7-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 3 (PM3). P30 to P33 are N-ch open-drain I/O port. Mask ROM version can contain pull-up resistors in P30 and P31 with the mask option. On-chip pull-up resistors of P34 to P36 can be used by setting pull-up resistor option register 3 (PU3).

(2) Control mode

These ports function as data I/O and clock I/O of serial interface IIC0.

(a) SDA0

Serial data I/O pin of serial interface IIC0.

(b) SCL0

Serial clock I/O pin of serial interface IIC0.

4.2.5 P40 to P47 (Port 4)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as an address/data bus.

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

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

Caution When using the falling edge detection interrupt (INTKR), be sure to set the memory expansion mode register (MEM) to 01H.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 4 (PM4). On-chip pull-up resistors can be used by setting pull-up resistor option register 4 (PU4).

(2) Control mode

These ports function as lower address/data bus pins (AD0 to AD7) in external memory expansion mode.

4.2.6 P50 to P57 (Port 5)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as an address bus.

Port 5 can drive LEDs directly.

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

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 5 (PM5). On-chip pull-up resistors can be used by setting pull-up resistor option register 5 (PU5).

(2) Control mode

These ports function as higher address bus pins (A8 to A15) in external memory expansion mode.

4.2.7 P64 to P67 (Port 6)

These are 4-bit I/O ports. Besides serving as I/O ports, they are used for control in external memory expansion mode.

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

(1) Port mode

These ports function as 4-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 6 (PM6).

On-chip pull-up resistors can be used by setting pull-up resistor option register 6 (PU6).

(2) Control mode

These ports function as control signal output pins ($\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WAIT}}$, ASTB) in external memory expansion mode.

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

4.2.8 P70 to P75 (Port 7)

These are 6-bit I/O ports. Besides serving as I/O ports, they function as a timer I/O, clock output, and buzzer output. The following operating modes can be specified in 1-bit units.

(1) Port mode

Port 7 functions as a 6-bit I/O port. They can be specified as an input port or output port in 1-bit units with port mode register 7 (PM7). On-chip pull-up resistors can be used by setting pull-up resistor option register 7 (PU7). P70 and P71 are also 16-bit timer/event counter 0 capture trigger signal input pins with a valid edge input.

(2) Control mode

Port 7 functions as timer I/O, clock output, and buzzer output.

(a) TI00

External count clock input pin to 16-bit timer/event counter 0 and capture trigger signal input pin to 16-bit timer/event counter capture register (CR01).

(b) TI01

Capture trigger signal input pin to 16-bit timer/event counter 0 capture register (CR00).

(c) TI50 and TI51

External count clock input pins to 8-bit timer/event counters 50 and 51.

(d) TO0, TO50, and TO51

Timer output pins.

(e) PCL

Clock output pin.

(f) BUZ

Buzzer output pin.

4.2.9 AV_{REF}

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

When no A/D converter is used, connect this pin directly to V_{SS0} or V_{SS1}.

4.2.10 AV_{DD}

This is an analog power supply pin of A/D converter. Always use the same potential as that of the V_{DD0} pin or V_{DD1} pin even when no A/D converter is used.

4.2.11 AV_{SS}

This is a ground potential pin of A/D converter. Always use the same potential as that of the V_{SS0} pin or V_{SS1} pin even when no A/D converter is used.

4.2.12 $\overline{\text{RESET}}$

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

★ ### 4.2.13 NC

NC (Non-connection) pin is not internally connected. Leave this pin open.

4.2.14 X1 and X2

Crystal/ceramic resonator connection pins for main system clock oscillation.
For external clock supply, input the clock signal to X1 and its inverted signal to X2.

4.2.15 XT1 and XT2

Crystal resonator connection pins for subsystem clock oscillation.
For external clock supply, input the clock signal to XT1 and its inverted signal to XT2.

4.2.16 V_{DD0} and V_{DD1}

V_{DD0} is a positive power supply pin.
V_{DD1} is a positive power supply pin other than port pin.

4.2.17 V_{SS0} and V_{SS1}

V_{SS0} is a ground potential port pin.
V_{SS1} is a ground potential pin other than port pin.

4.2.18 V_{PP} (flash memory versions only)

High-voltage apply pin for flash memory programming mode setting and program write/verify.
★ Handle in either of the following ways.

- Independently connect a 10 kΩ pull-down resistor.
- Set the jumper on the board so that this pin is connected directly to the dedicated flash programmer in programming mode and directly to V_{SS0} or V_{SS1} in normal operation mode.

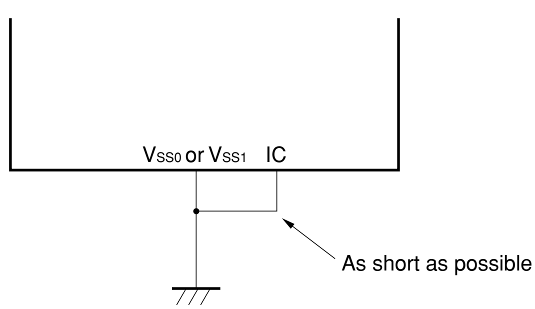
When there is a potential difference between the V_{PP} pin and V_{SS0} pin or V_{SS1} pin because the wiring between the two pins is too long or external noise is input to the V_{PP} pin, the user program may not operate normally.

4.2.19 IC (mask ROM version only)

The IC (Internally Connected) pin is provided to set the test mode to check the μPD780024AY, 780034AY Subseries at delivery. Connect it directly to the V_{SS0} or V_{SS1} pin with the shortest possible wire in the normal operating mode.

When a potential difference is produced between the IC pin and V_{SS0} pin or V_{SS1} 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 operate normally.

- **Connect IC pins to V_{SS0} pins or V_{SS1} pins directly.**



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

Table 4-1 shows the types of pin I/O circuit and the recommended connections of unused pins.
See 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 to P02/INTP2	8-C	I/O	Input: Independently connect to V_{SS0} or V_{SS1} via a resistor. Output: Leave open.
P03/INTP3/ADTRG			
P10/ANI0 to P17/ANI7	25	Input	Connect directly to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1}.
P20/SI30	8-C	I/O	Input: Independently connect to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1} via a resistor. Output: Leave open.
P21/SO30	5-H		
P22/$\overline{\text{SCK30}}$	8-C		
P23/RxD0			
P24/TxD0	5-H		
P25/ASCK0	8-C		
P30, P31 (for mask ROM version)	13-Q		Input: Connect directly to V_{SS0} or V_{SS1}. Output: Set the output latch of the port to 0, and leave these pins open at low level.
P30, P31 (for flash memory version)	13-P		
P32/SDA0	13-R		
P33/SCL0			
P34	8-C		Input: Independently connect to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1} via a resistor. Output: Leave open.
P35	5-H		
P36	8-C		
P40/AD0 to P47/AD7	5-H		Input: Independently connect to V_{DD0} or V_{DD1} 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
P50/A8 to P57/A15	5-H	I/O	Input: Independently connect to V_{DD0}, V_{DD1}, V_{SS0}, or V_{SS1} via a resistor. Output: Leave open.
P64/$\overline{\text{RD}}$			
P65/$\overline{\text{WR}}$			
P66/$\overline{\text{WAIT}}$			
P67/ASTB			
P70/TI00/TO0	8-C		
P71/TI01			
P72/TI50/TO50			
P73/TI51/TO51			
P74/PCL	5-H		
P75/BUZ			
$\overline{\text{RESET}}$	2	Input	–
NC[Note]	–	–	Leave open.
XT1	16	Input	Connect directly to V_{DD0} or V_{DD1}.
XT2		–	Leave open.
AV_{DD}	–		Connect directly to V_{DD0} or V_{DD1}.
AV_{REF}			Connect directly to V_{SS0} or V_{SS1}.
AV_{SS}			
IC (for mask ROM version)			
V_{PP} (for flash memory version)			Independently connect a 10 kΩ pull-down resistor to this pin, or connect directly to V_{SS0} or V_{SS1}.

Note The NC pin is available only for a 73-pin plastic FBGA.

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

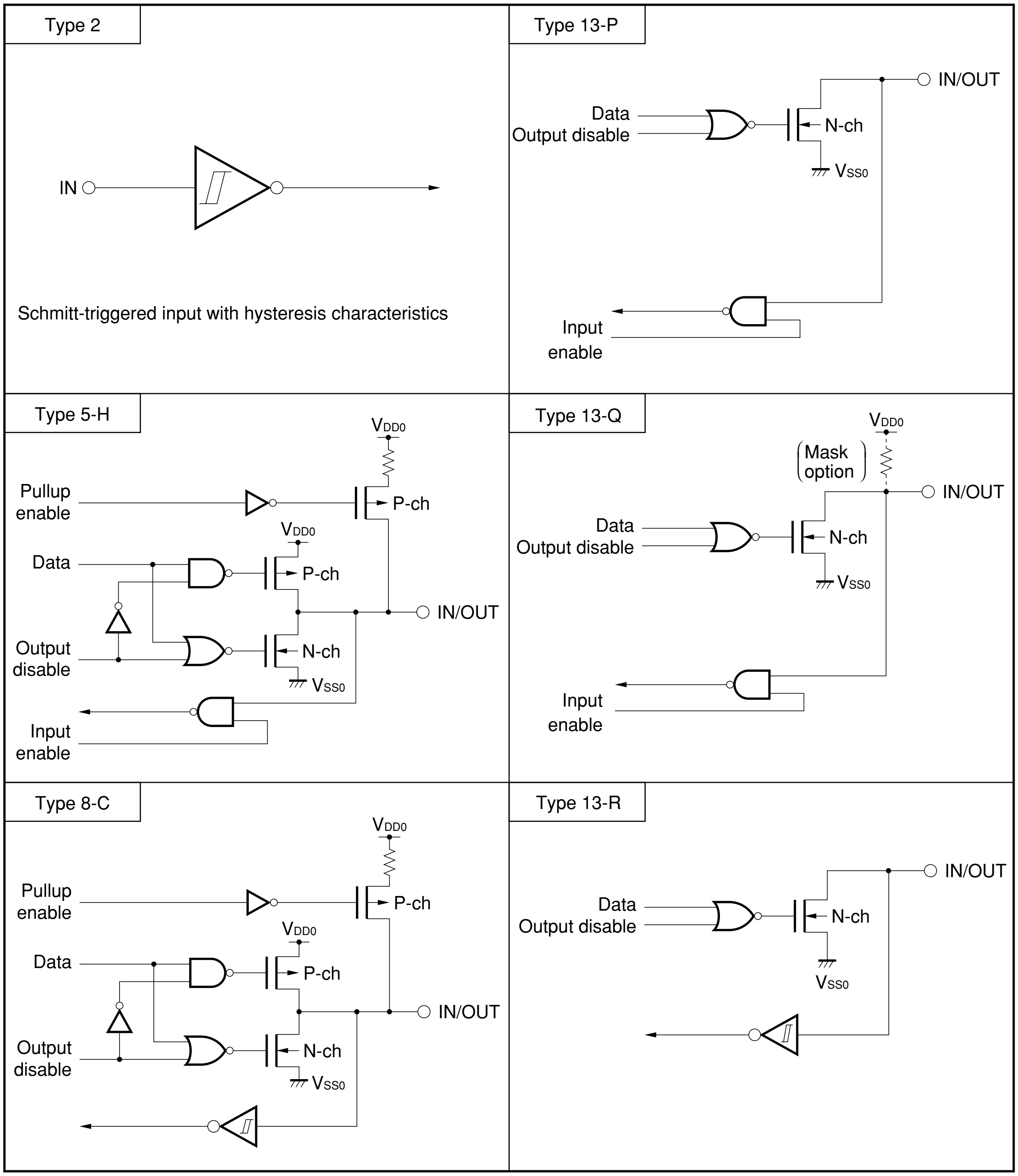
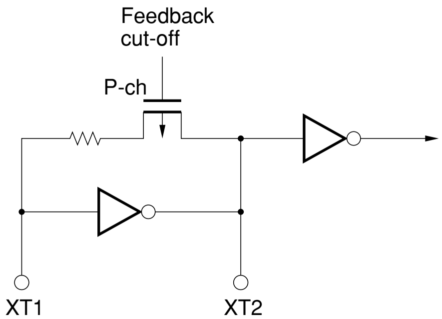
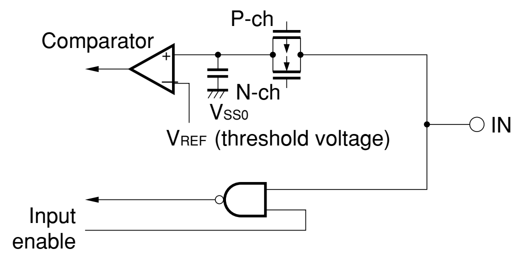


Figure 4-1. Pin I/O Circuit List (2/2)

Type 16



Type 25



5.1 Memory Spaces

μPD780024A, 780034A, 780024AY, 780034AY Subseries can access 64 KB memory space respectively. Figures 5-1 to 5-5 show memory maps.

Caution In case of the internal memory capacity, the initial value of memory size switching register (IMS) of all products (μPD780024A, 780034A, 780024AY, and 780034AY Subseries) is fixed (IMS = CFH). Therefore, set the value corresponding to each product indicated below.

μPD780021A, 780031A, 780021AY, 780031AY:	**42H**
μPD780022A, 780032A, 780022AY, 780032AY:	**44H**
μPD780023A, 780033A, 780023AY, 780033AY:	**C6H**
μPD780024A, 780034A, 780024AY, 780034AY:	**C8H**
μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY:	**Value for mask ROM version**

Figure 5-1. Memory Map (μPD780021A, 780031A, 780021AY, 780031AY)

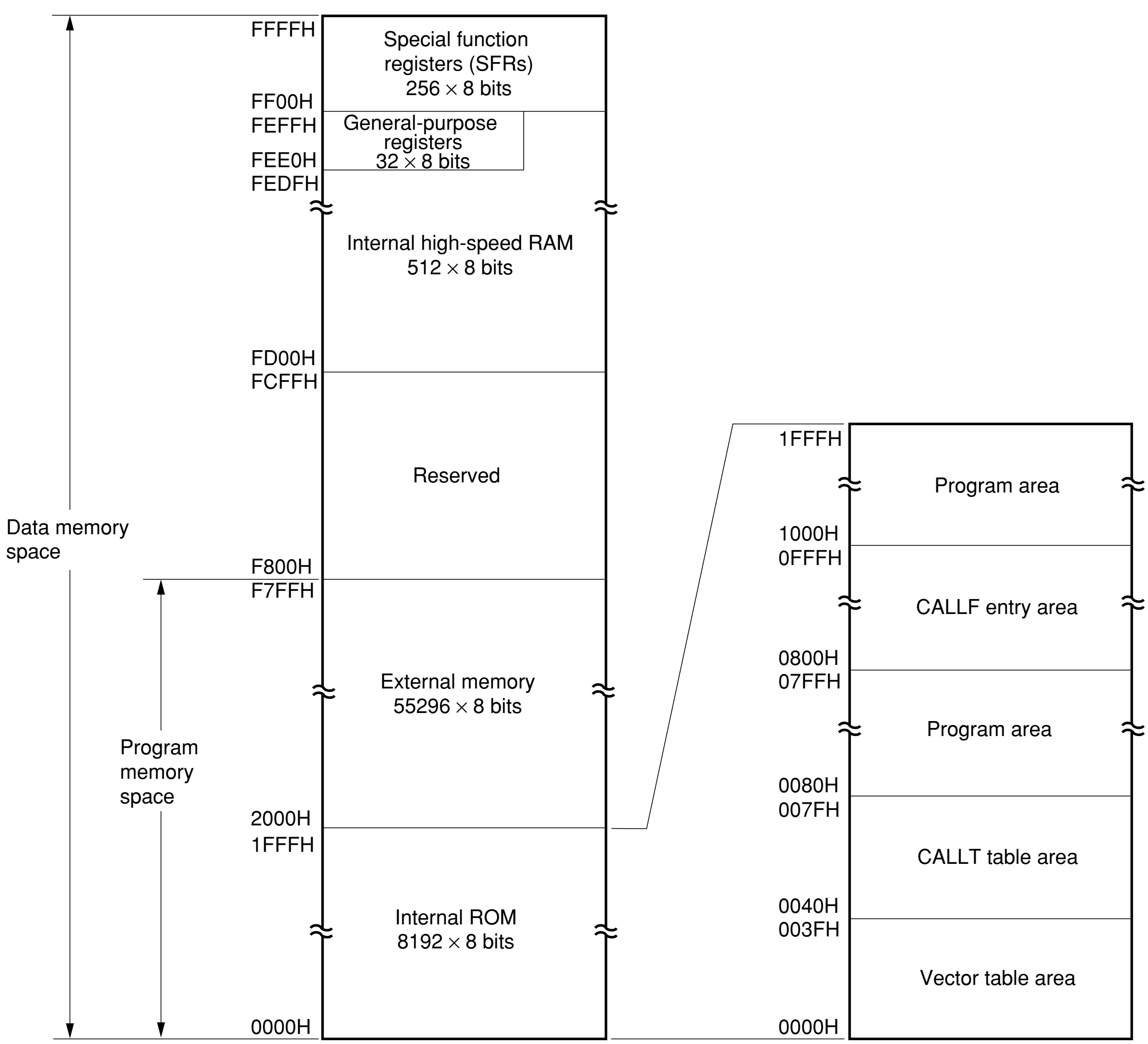


Figure 5-2. Memory Map (μPD780022A, 780032A, 780022AY, 780032AY)

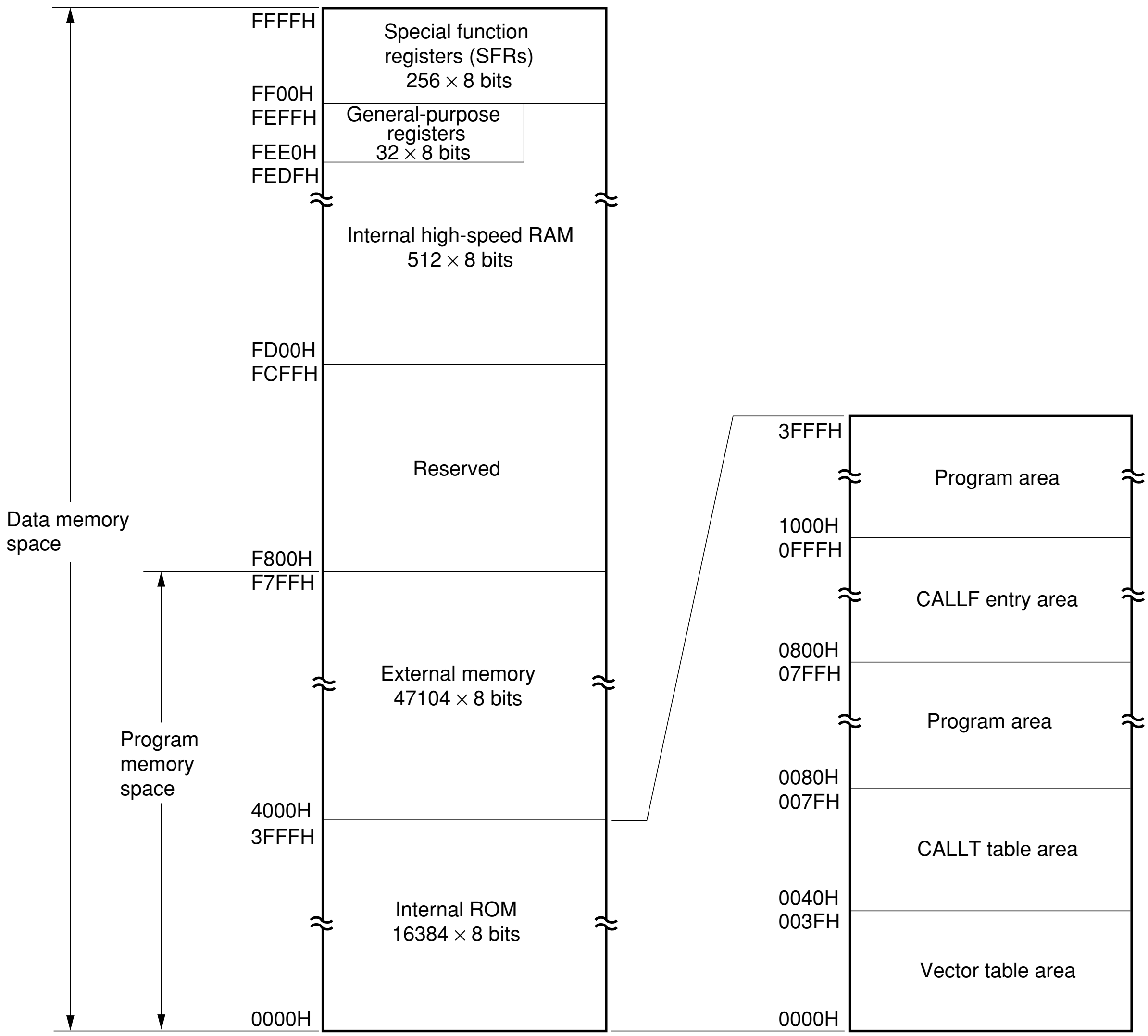


Figure 5-3. Memory Map (μPD780023A, 780033A, 780023AY, 780033AY)

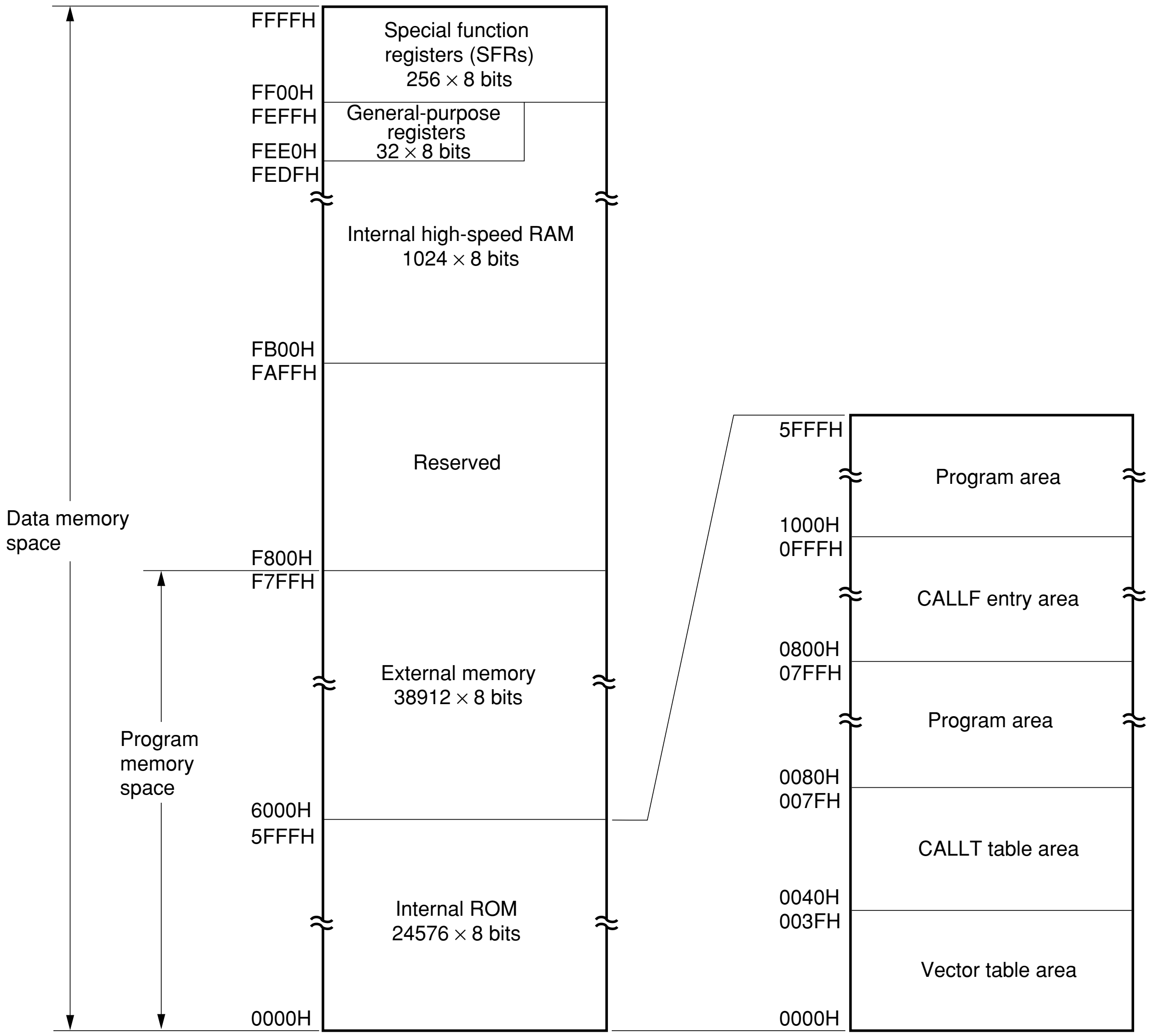


Figure 5-4. Memory Map (μPD780024A, 780034A, 780024AY, 780034AY)

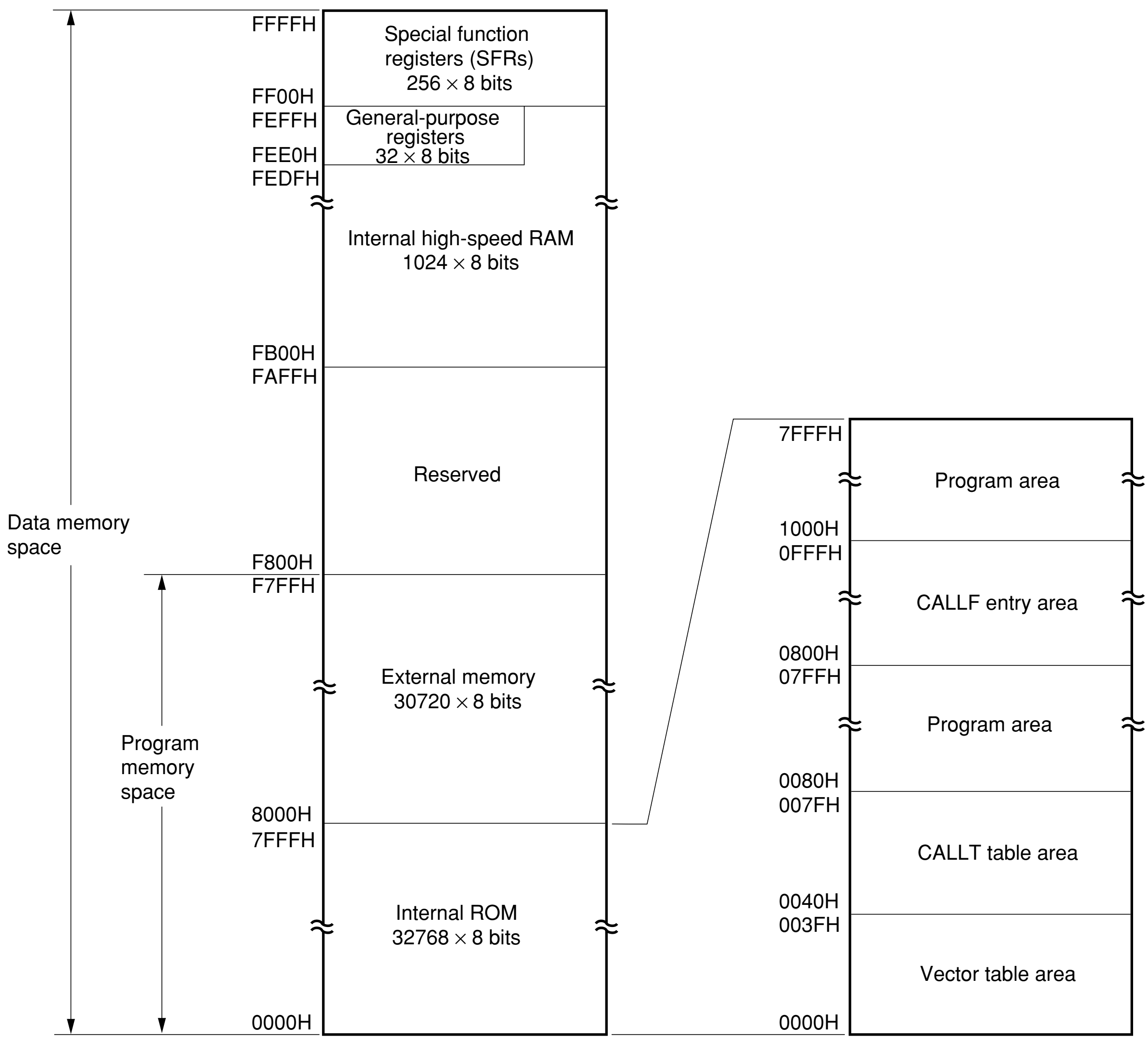
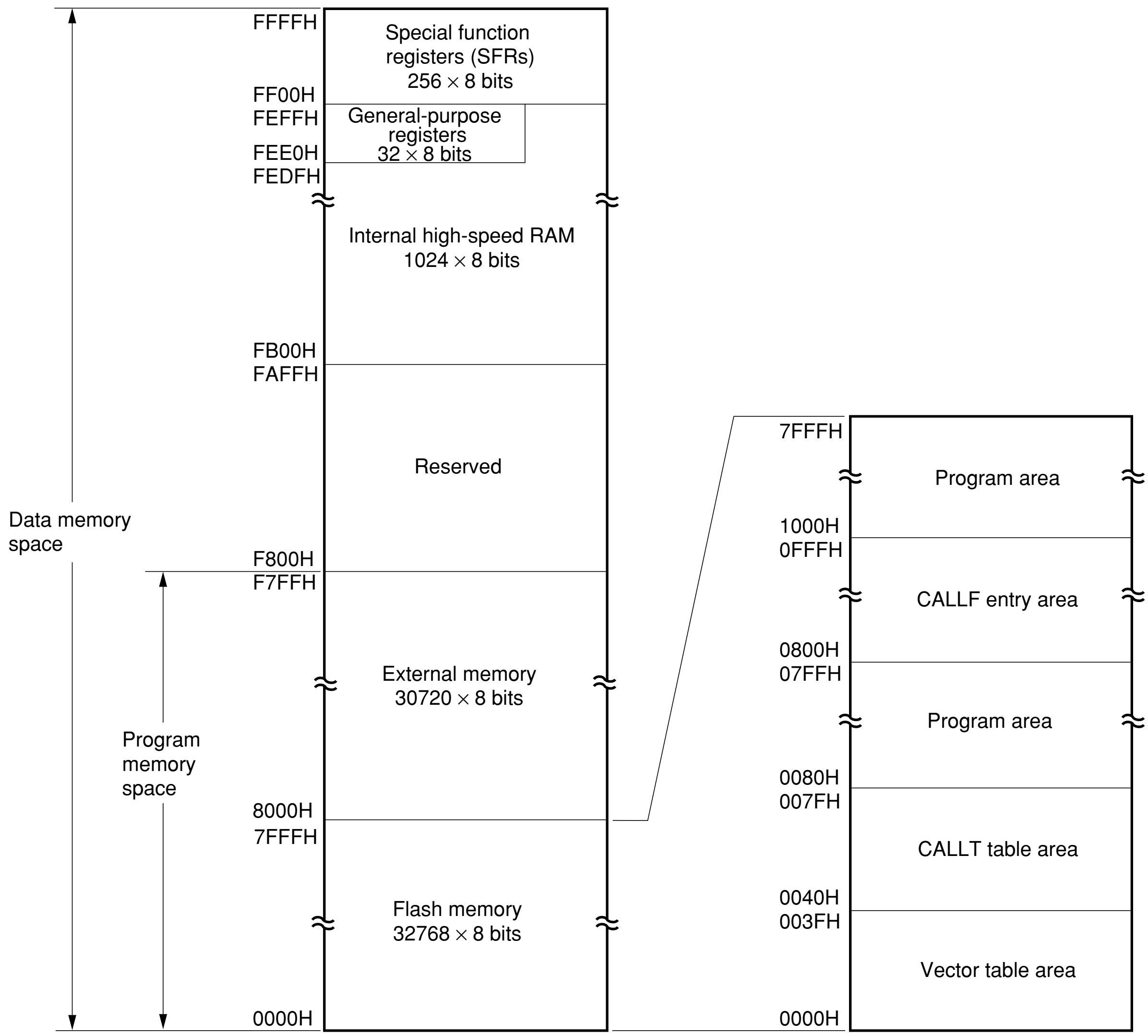


Figure 5-5. Memory Map (μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY)



5.1.1 Internal program memory space

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

The μPD780024A, 780034A, 780024AY, and 780034AY Subseries products incorporate an on-chip ROM (mask ROM or flash memory), as listed below.

Table 5-1. Internal ROM Capacity

Part Number	Type	Capacity
μPD780021A, 780031A, 780021AY, 780031AY	Mask ROM	8192 × 8 bits (0000H to 1FFFH)
μPD780022A, 780032A, 780022AY, 780032AY		16384 × 8 bits (0000H to 3FFFH)
μPD780023A, 780033A, 780023AY, 780033AY		24576 × 8 bits (0000H to 5FFFH)
μPD780024A, 780034A, 780024AY, 780034AY		32768 × 8 bits (0000H to 7FFFH)
μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY	Flash memory	32768 × 8 bits (0000H to 7FFFH)

The internal program memory space 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 $\overline{\text{RESET}}$ input and program start addresses for branch upon generation of each interrupt request are stored in the vector table area. Of the 16-bit address, lower 8 bits are stored at even addresses and higher 8 bits are stored at odd addresses.

Table 5-2. Vector Table

Vector Table Address	Interrupt Source
0000H	$\overline{\text{RESET}}$ input
0004H	INTWDT
0006H	INTP0
0008H	INTP1
000AH	INTP2
000CH	INTP3
000EH	INTSER0
0010H	INTSR0
0012H	INTST0
0014H	INTCSI30
0016H	INTCSI31**Note 1**
0018H	INTIIC0**Note 2**
001AH	INTWTI
001CH	INTTM00
001EH	INTTM01
0020H	INTTM50
0022H	INTTM51
0024H	INTAD0
0026H	INTWT
0028H	INTKR
003EH	BRK

Notes 1. μPD780024A, 780034A Subseries only

2. μPD780024AY, 780034AY Subseries only

(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 μPD780024A, 780034A, 780024AY, and 780034AY Subseries products incorporate an internal high-speed RAM, as listed below.

Table 5-3. Internal High-Speed RAM Capacity

Part Number	Internal High-Speed RAM
μPD780021A, 780031A, 780021AY, 780031AY	512 × 8 bits (FD00H to FEFFH)
μPD780022A, 780032A, 780022AY, 780032AY	
μPD780023A, 780033A, 780023AY, 780033AY	1024 × 8 bits (FB00H to FEFFH)
μPD780024A, 780034A, 780024AY, 780034AY	
μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY	

The 32-byte area FEE0H to FEFFH is allocated four general-purpose register banks composed of eight 8-bit registers.

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

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

5.1.3 Special function register (SFR) area

An on-chip peripheral hardware special function register (SFR) is allocated in the area FF00H to FFFFH (see **5.2.3 Special function register (SFR) Table 5-5 Special Function Register List**).

Caution Do not access addresses where the SFR is not assigned.

5.1.4 External memory space

The external memory space is accessible with memory expansion mode register (MEM). External memory space can store program, table data, etc., and allocate peripheral devices.

5.1.5 Data memory addressing

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

Several addressing modes are provided for addressing the memory relevant to the execution of instructions for the μPD780024A, 780034A, 780024AY, and 780034AY Subseries, based on operability and other considerations. For areas containing data memory in particular, special addressing methods designed for the functions of special function registers (SFR) and general-purpose registers are available for use. Correspondence between data memory and addressing is illustrated in Figures 5-6 to 5-10. For the details of each addressing mode, see **5.4 Operand Address Addressing**.

Figure 5-6. Correspondence Between Data Memory and Addressing (μPD780021A, 780031A, 780021AY, 780031AY)

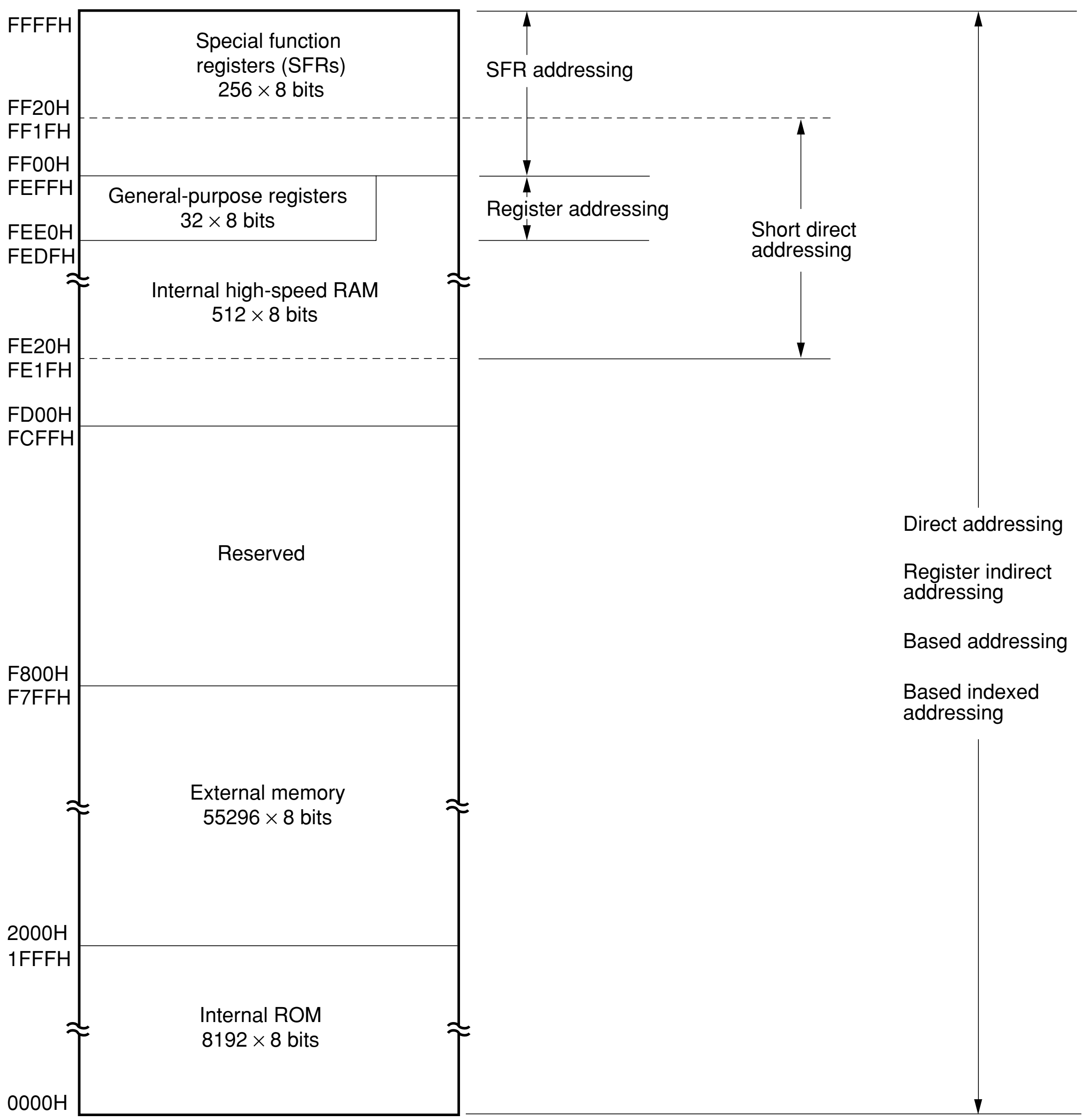


Figure 5-7. Correspondence Between Data Memory and Addressing (μPD780022A, 780032A, 780022AY, 780032AY)

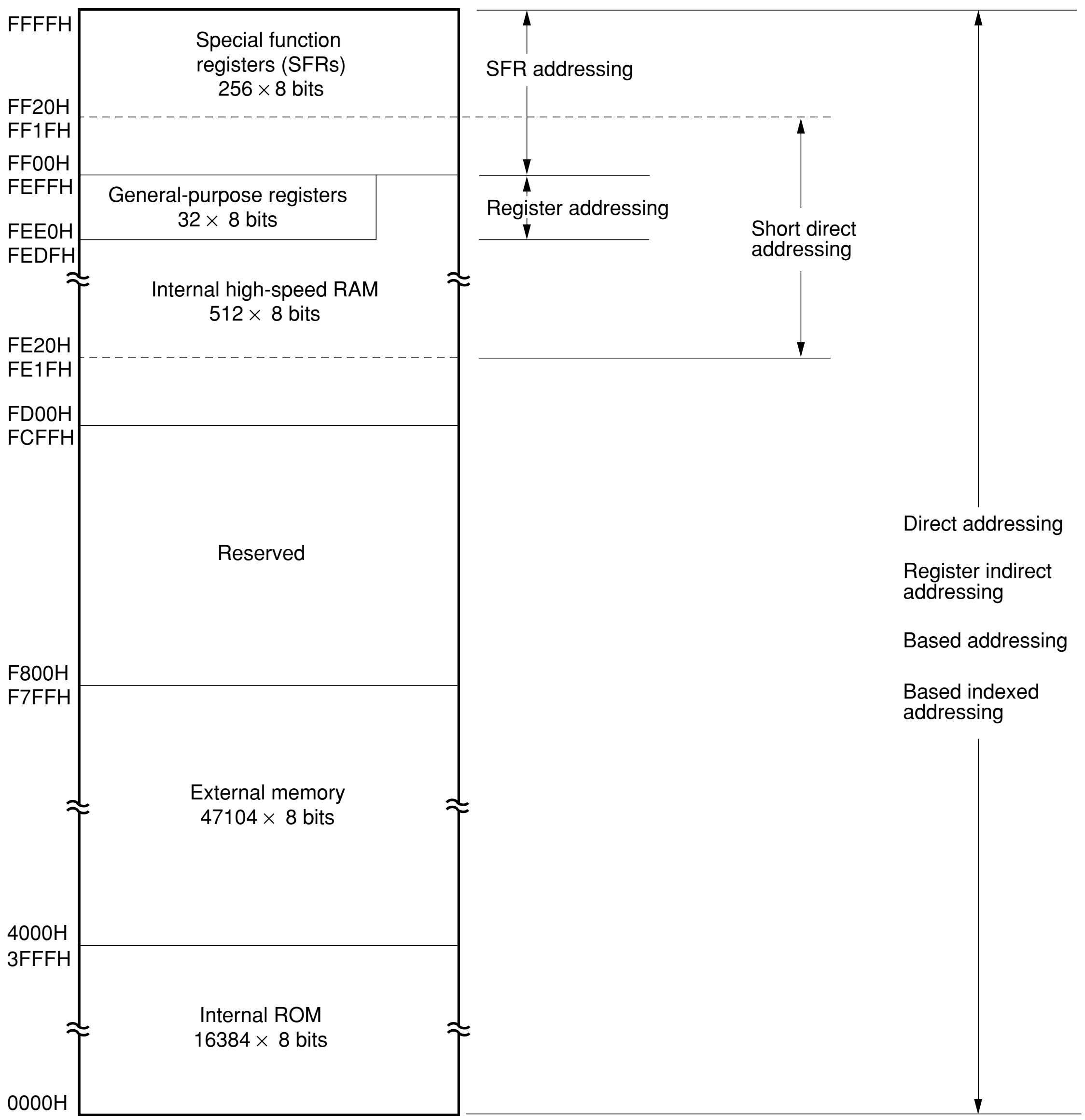


Figure 5-8. Correspondence Between Data Memory and Addressing (μPD780023A, 780033A, 780023AY, 780033AY)

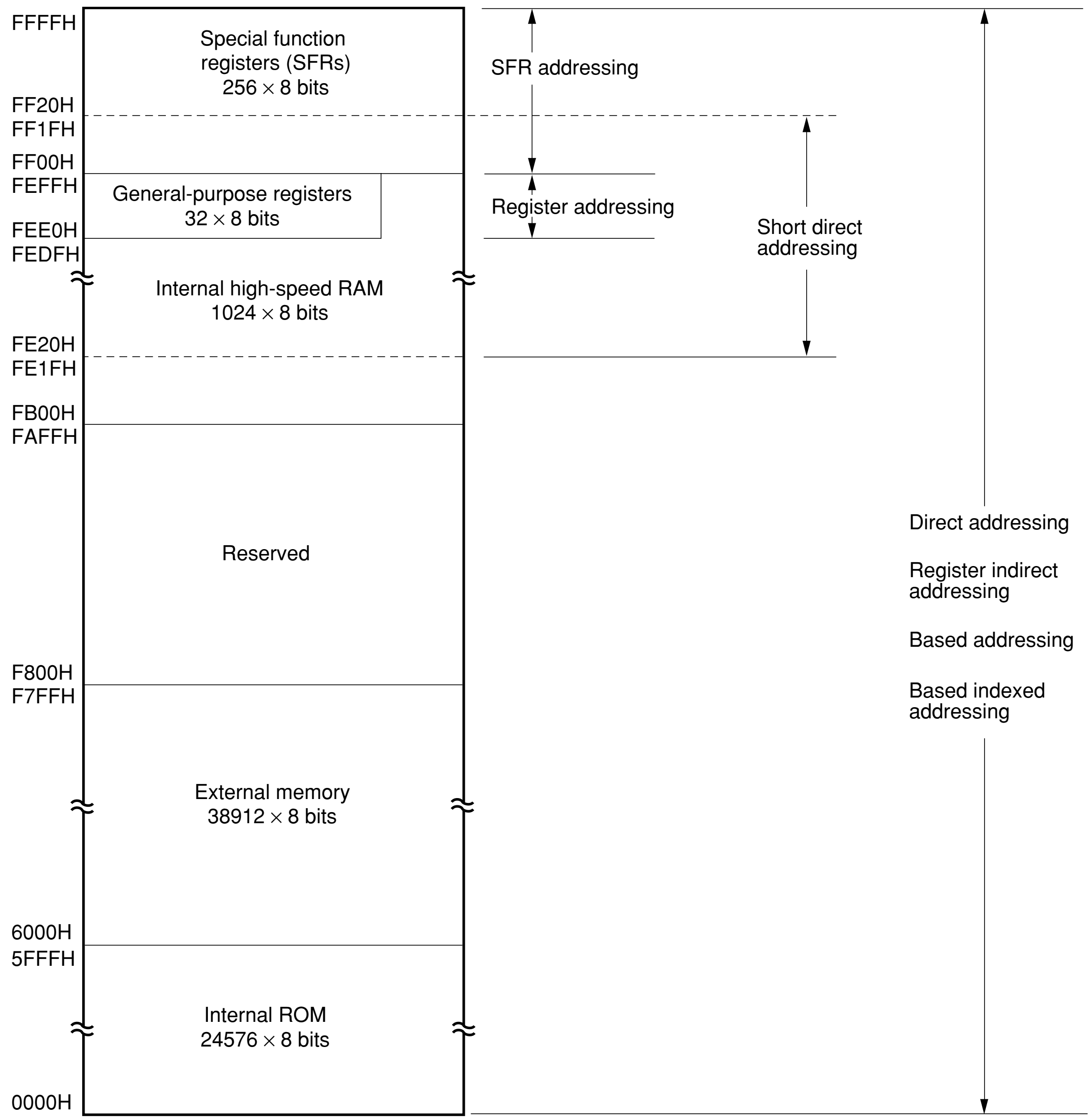


Figure 5-9. Correspondence Between Data Memory and Addressing
(μPD780024A, 780034A, 780024AY, 780034AY)

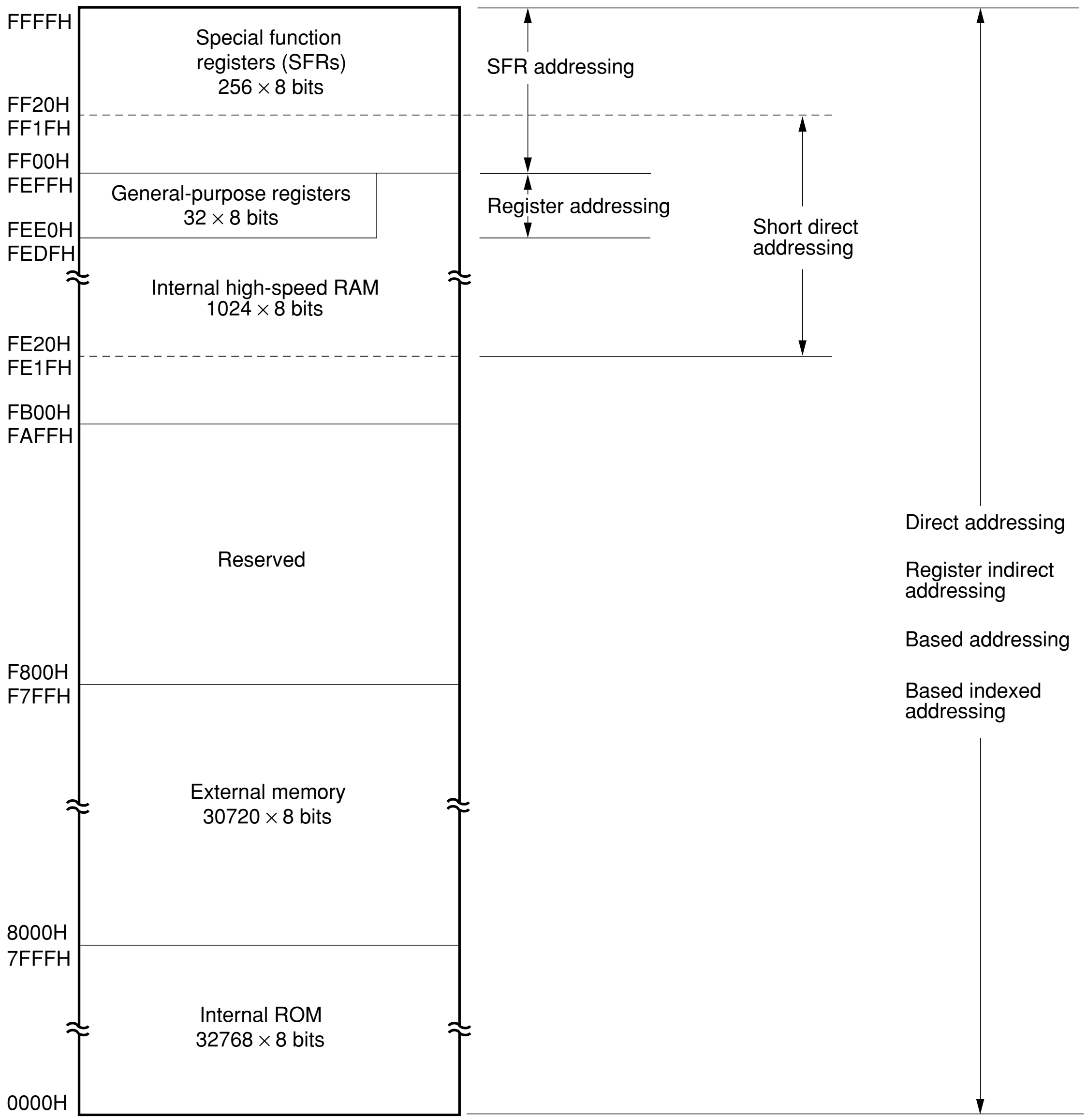
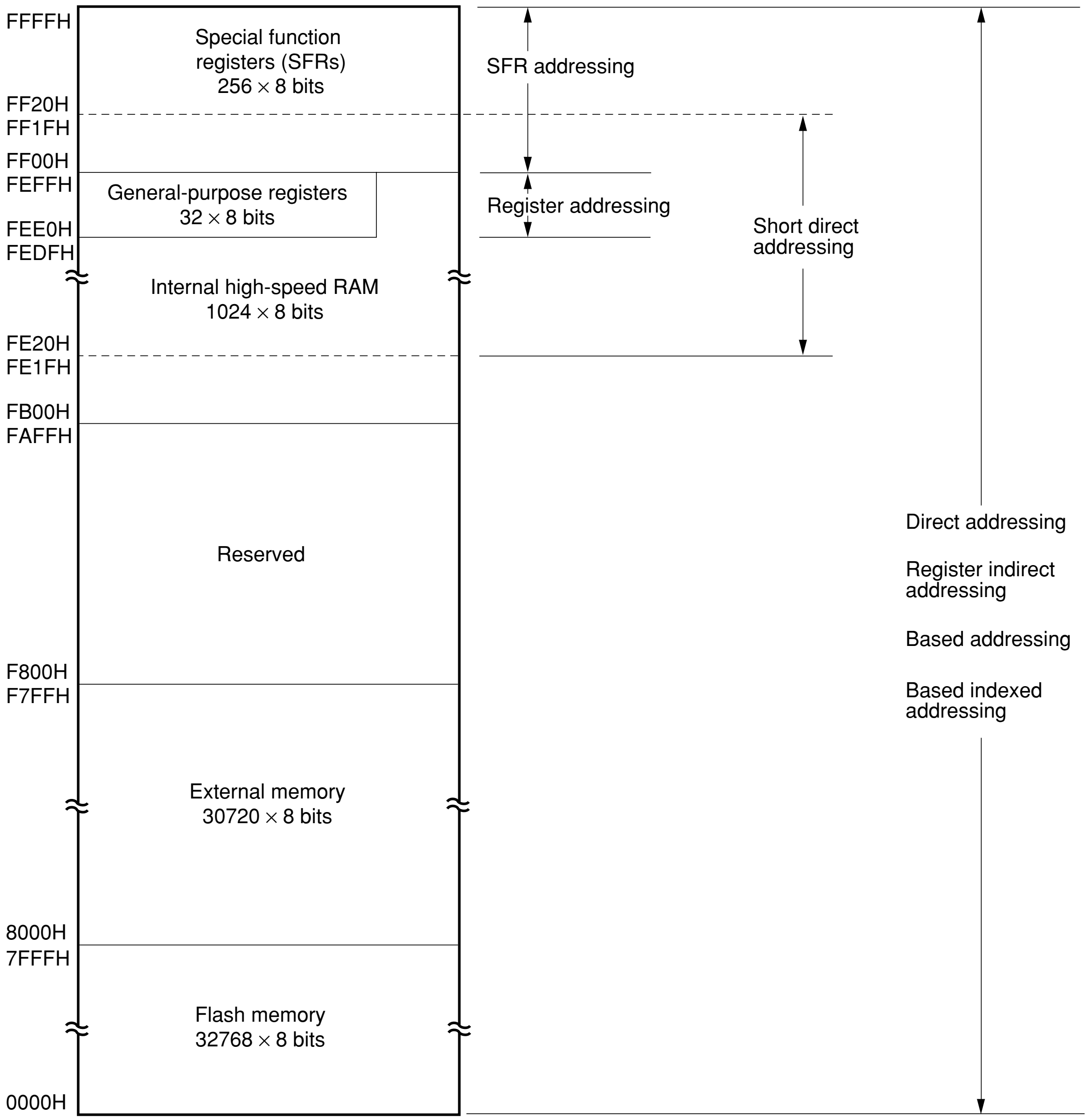


Figure 5-10. Correspondence Between Data Memory and Addressing (μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY)



5.2 Processor Registers

The μPD780024A, 780034A, 780024AY, 780034AY Subseries products 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 5-11. Program Counter Format

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

(2) Program status word (PSW)

The program status word is an 8-bit register consisting of various flags 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 5-12. Program Status Word Format

	7							0
PSW	IE	Z	RBS1	AC	RBS0	0	ISP	CY

(a) Interrupt enable flag (IE)

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

When 0, the IE is set to the disable interrupt (DI) state, and only non-maskable interrupt request becomes acknowledgeable. Other interrupt requests are all disabled.

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

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

(b) Zero flag (Z)

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

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

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

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

(d) Auxiliary carry flag (AC)

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

(e) In-service priority flag (ISP)

This flag manages the priority of acknowledgeable maskable vectored interrupts. When this flag is 0, low-level vectored interrupt requests specified with a priority specification flag register (PR0L, PR0H, PR1L) (see **19.3 (3) Priority specification flag registers (PR0L, PR0H, PR1L)**) are disabled for acknowledgment. When it is 1, all interrupts are acknowledgeable. Actual request acknowledgment is controlled with 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 can be set as the stack area. The internal high-speed RAM areas of each product are as follows.

Table 5-4. Internal High-Speed RAM Area

Part Number	Internal High-Speed RAM Area
μPD780021A, 780031A, 780021AY, 780031AY μPD780022A, 780032A, 780022AY, 780032AY	FD00H to FEFFH
μPD780023A, 780033A, 780023AY, 780033AY μPD780024A, 780034A, 780024AY, 780034AY μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY	FB00H to FEFFH

Figure 5-13. Stack Pointer Format

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

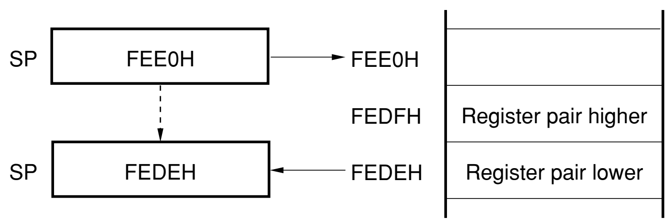
The SP is decremented ahead of write (save) to the stack memory and is incremented after read (restore) from the stack memory.

Each stack operation saves/restores data as shown in Figures 5-14 and 5-15.

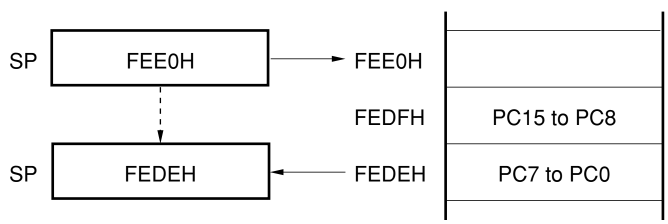
Caution Since $\overline{\text{RESET}}$ input makes the SP contents undefined, be sure to initialize the SP before using the stack.

★ **Figure 5-14. Data to Be Saved to Stack Memory**

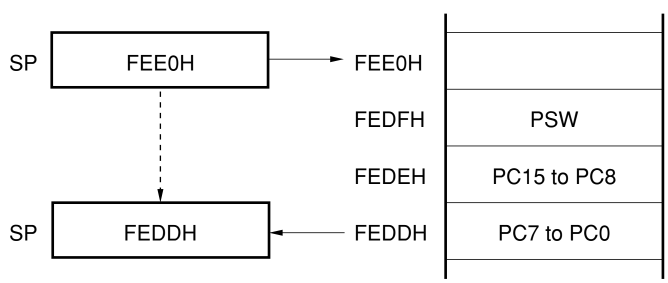
(a) PUSH rp instruction (when SP is FEE0H)



(b) CALL, CALLF, CALLT instructions (when SP is FEE0H)

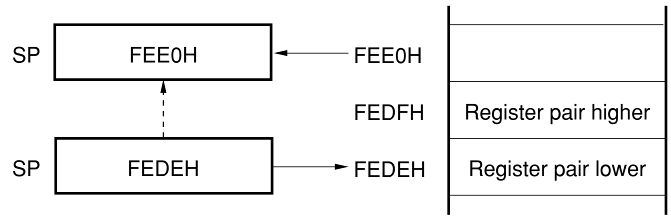


(c) Interrupt, BRK instruction (when SP is FEE0H)

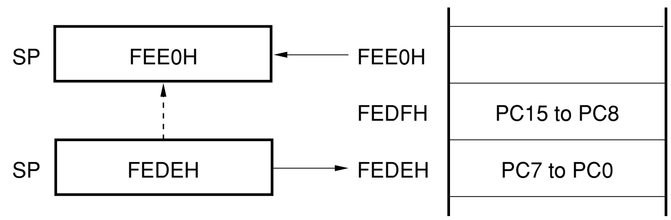


★ **Figure 5-15. Data to Be Restored from Stack Memory**

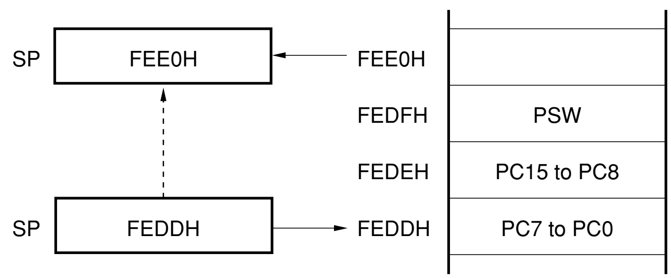
(a) POP rp instruction (when SP is FEDEH)



(b) RET instruction (when SP is FEDEH)



(c) RETI, RETB instructions (when SP is FEDDH)



5.2.2 General-purpose registers

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

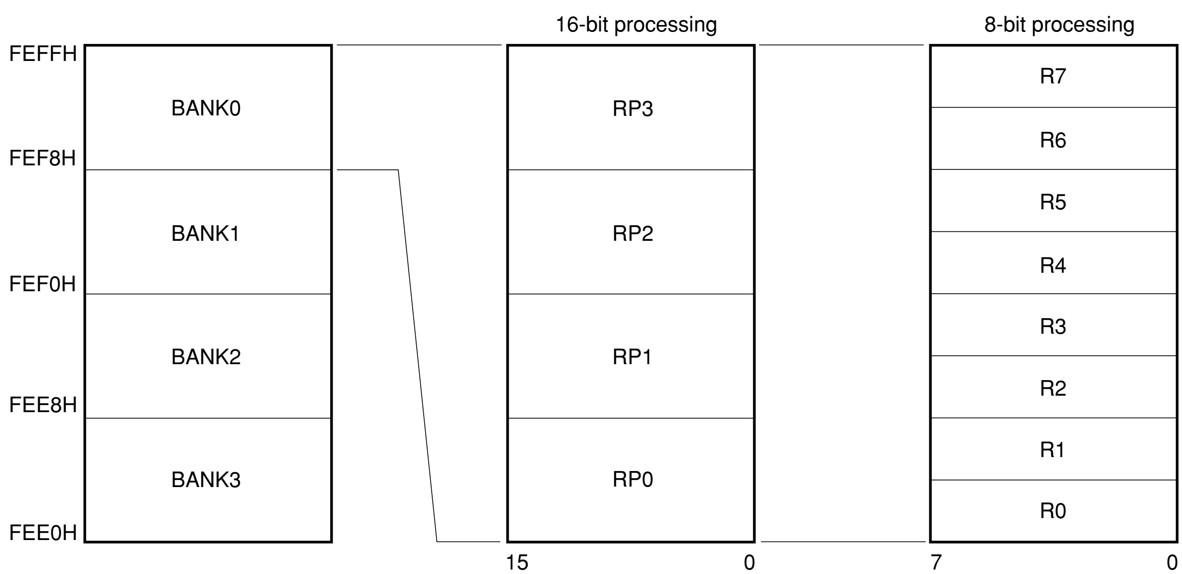
Each register can also be used as an 8-bit register. Two 8-bit registers can be used in pairs as a 16-bit register (AX, BC, DE, and HL).

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).

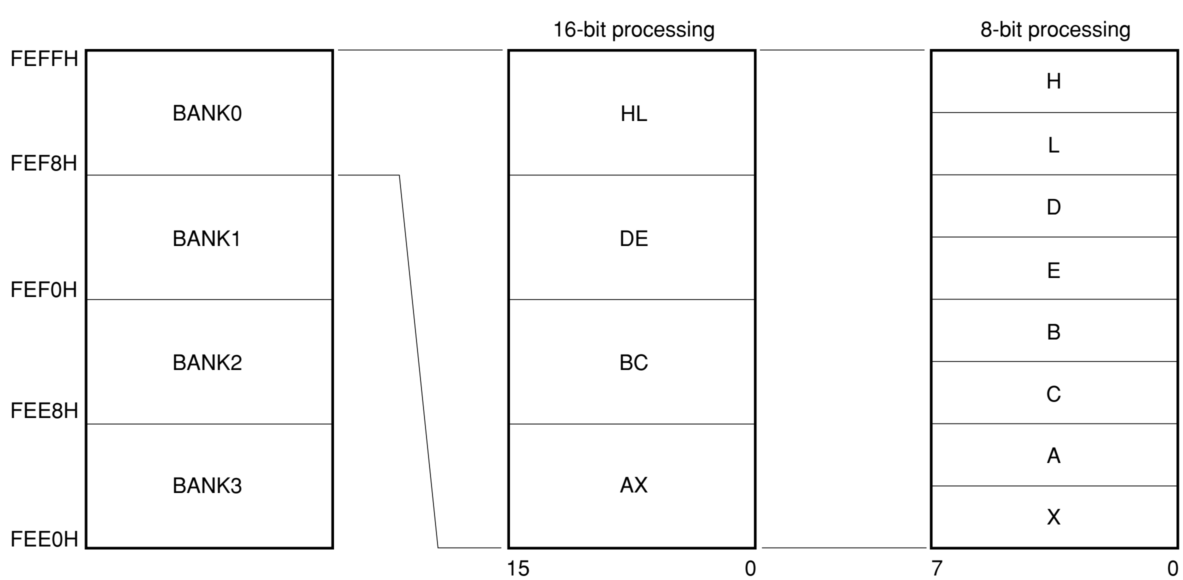
Register banks to be used for instruction execution are set with the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interrupts for each bank.

Figure 5-16. General-Purpose Register Configuration

(a) Absolute name



(b) Function name



5.2.3 Special function register (SFR)

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

It is allocated in the FF00H to FFFFH area.

The special function register can be manipulated like the general-purpose register, with the operation, transfer and bit manipulation instructions. 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 with assembler for the 1-bit manipulation instruction operand (sfr.bit).
  This manipulation can also be specified with an address.
- 8-bit manipulation
  Describe the symbol reserved with assembler for the 8-bit manipulation instruction operand (sfr).
  This manipulation can also be specified with an address.
- 16-bit manipulation
  Describe the symbol reserved with assembler for the 16-bit manipulation instruction operand (sfrp).
  When addressing an address, describe an even address.

Table 5-5 gives a list of special function registers. The meaning of items in the table is as follows.

- Symbol
  Symbol indicating the address of a special function register. It is a reserved word in the RA78K0, and is defined via the header file “sfrbit.h” in the CC78K0. When using the RA78K0, ID78K0-NS, ID78K0, or SM78K0, symbols can be written as an instruction operand.
- R/W
  Indicates whether the corresponding special function register can be read or written.
  R/W: Read/write enable
  R: Read only
  W: Write only
- Manipulatable bit units
  Indicates the manipulatable bit unit (1, 8, or 16). “–” indicates a bit unit for which manipulation is not possible.
- After reset
  Indicates each register status upon $\overline{\text{RESET}}$ input.

Table 5-5. 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		R	√	√	–	
FF02H	Port 2	P2		R/W	√	√	–	
FF03H	Port 3	P3			√	√	–	
FF04H	Port 4	P4			√	√	–	
FF05H	Port 5	P5			√	√	–	
FF06H	Port 6	P6			√	√	–	
FF07H	Port 7	P7			√	√	–	
FF0AH	16-bit timer capture/compare register 00	CR00			–	–	√	Undefined
FF0BH								
FF0CH	16-bit timer capture/compare register 01	CR01			–	–	√	
FF0DH								
FF0EH	16-bit timer counter 0	TM0		R	–	–	√	0000H
FF0FH								
FF10H	8-bit timer compare register 50	CR50		R/W	–	√	–	Undefined
FF11H	8-bit timer compare register 51	CR51			–	√	–	
FF12H	8-bit timer counter 50	TM5	TM50	R	–	√	√	00H
FF13H	8-bit timer counter 51		TM51		–	√		
FF16H	A/D conversion result register 0	ADCR0			–	–	√ Note 2	
FF17H					–	√ Note 1		
FF18H	Transmit shift register 0	TXS0		W	–	√	–	FFH
	Receive buffer register 0	RXB0		R	–	√	–	
FF1AH	Serial I/O shift register 30	SIO30		R/W	–	√	–	Undefined
FF1BH	Serial I/O shift register 31Note 3	SIO31			–	√	–	
FF1FH	IIC shift register 0Note 4	IIC0			–	√	–	00H

Notes 1. μPD780024A, 780024AY Subseries only

2. μPD780034A, 780034AY Subseries only, 16-bit access possible

3. μPD780024A, 780034A Subseries only

4. μPD780024AY, 780034AY Subseries only

Table 5-5. 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	
FF20H	Port mode register 0	PM0	R/W	√	√	–	FFH
FF22H	Port mode register 2	PM2		√	√	–	
FF23H	Port mode register 3	PM3		√	√	–	
FF24H	Port mode register 4	PM4		√	√	–	
FF25H	Port mode register 5	PM5		√	√	–	
FF26H	Port mode register 6	PM6		√	√	–	
FF27H	Port mode register 7	PM7		√	√	–	
FF30H	Pull-up resistor option register 0	PU0		√	√	–	00H
FF32H	Pull-up resistor option register 2	PU2		√	√	–	
FF33H	Pull-up resistor option register 3	PU3		√	√	–	
FF34H	Pull-up resistor option register 4	PU4		√	√	–	
FF35H	Pull-up resistor option register 5	PU5		√	√	–	
FF36H	Pull-up resistor option register 6	PU6		√	√	–	
FF37H	Pull-up resistor option register 7	PU7		√	√	–	
FF40H	Clock output select register	CKS		√	√	–	
FF41H	Watch timer operation mode register	WTM		√	√	–	
FF42H	Watchdog timer clock select register	WDCS		–	√	–	
FF47H	Memory expansion mode register	MEM		√	√	–	
FF48H	External interrupt rising edge enable register	EGP		√	√	–	
FF49H	External interrupt falling edge enable register	EGN		√	√	–	
FF60H	16-bit timer mode control register 0	TMC0		√	√	–	
FF61H	Prescaler mode register 0	PRM0		–	√	–	
FF62H	Capture/compare control register 0	CRC0		√	√	–	
FF63H	16-bit timer output control register 0	TOC0		√	√	–	
FF70H	8-bit timer mode control register 50	TMC50		√	√	–	
FF71H	Timer clock select register 50	TCL50		–	√	–	
FF78H	8-bit timer mode control register 51	TMC51		√	√	–	
FF79H	Timer clock select register 51	TCL51		–	√	–	
FF80H	A/D converter mode register 0	ADM0		√	√	–	
FF81H	Analog input channel specification register 0	ADS0		–	√	–	
FFA0H	Asynchronous serial interface mode register 0	ASIM0		√	√	–	
FFA1H	Asynchronous serial interface status register 0	ASIS0	R	–	√	–	
FFA2H	Baud rate generator control register 0	BRGC0	R/W	–	√	–	

Table 5-5. 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	
FFA8H	IIC control register 0[Note 1]	IICC0		R/W	√	√	–	00H
FFA9H	IIC status register 0[Note 1]	IICS0		R	√	√	–	
FFAAH	IIC transfer clock select register 0[Note 1]	IICCL0		R/W	√	√	–	
FFABH	Slave address register 0[Note 1]	SVA0			–	√	–	
FFB0H	Serial operation mode register 30	CSIM30			√	√	–	
FFB8H	Serial operation mode register 31[Note 2]	CSIM31			√	√	–	
FFE0H	Interrupt request flag register 0L	IF0	IF0L		√	√	√	
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 level specification flag register 0L	PR0	PR0L		√	√	√	
FFE9H	Priority level specification flag register 0H		PR0H		√	√		
FFEAH	Priority level specification flag register 1L	PR1L			√	√	–	
FFF0H	Memory size switching register	IMS			–	√	–	CFH[Note 3]
FFF8H	Memory expansion wait setting 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 1. μPD780024AY, 780034AY Subseries only

2. μPD780024A, 780034A Subseries only

3. The default is CFH, but set the value corresponding to each respective product as indicated below.

μPD780021A, 780031A, 780021AY, 780031AY: 42H
μPD780022A, 780032A, 780022AY, 780032AY: 44H
μPD780023A, 780033A, 780023AY, 780033AY: C6H
μPD780024A, 780034A, 780024AY, 780034AY: C8H
μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY: Value for mask ROM version

5.3 Instruction Address Addressing

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

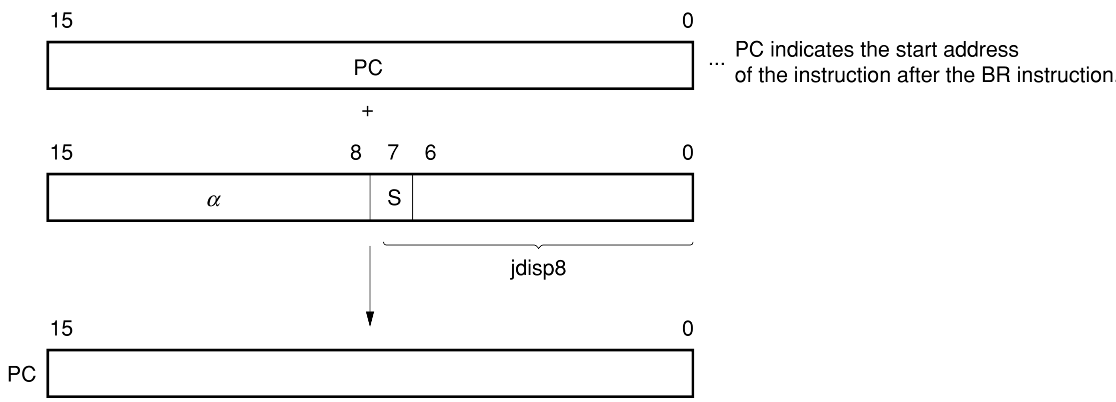
5.3.1 Relative addressing

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

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

[Illustration]



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

5.3.2 Immediate addressing

[Function]

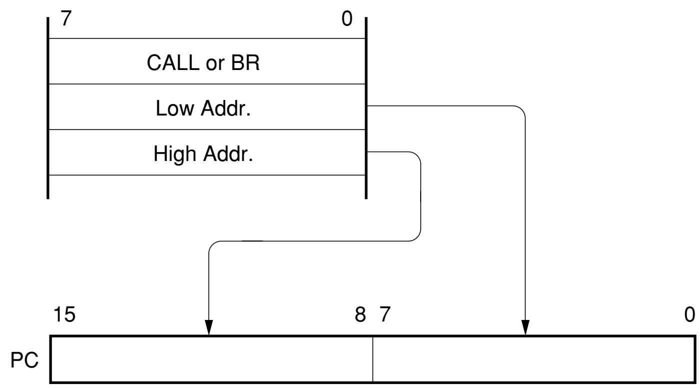
Immediate data in the instruction word is transferred to the program counter (PC) and branched.

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

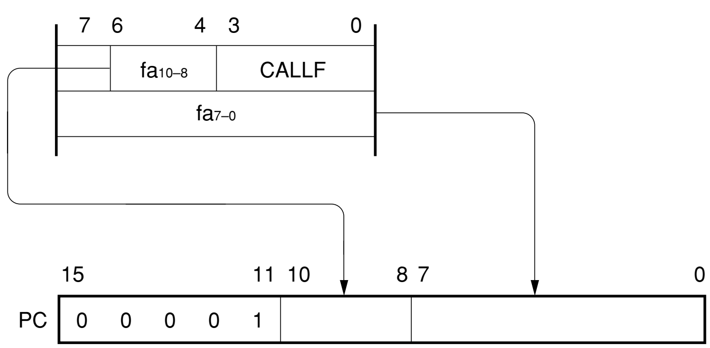
CALL !addr16 and BR !addr16 instructions can be branched to the entire memory space. The CALLF !addr11 instruction is branched to the 0800H to 0FFFH area.

[Illustration]

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



In the case of CALLF !addr11 instruction



5.3.3 Table indirect addressing

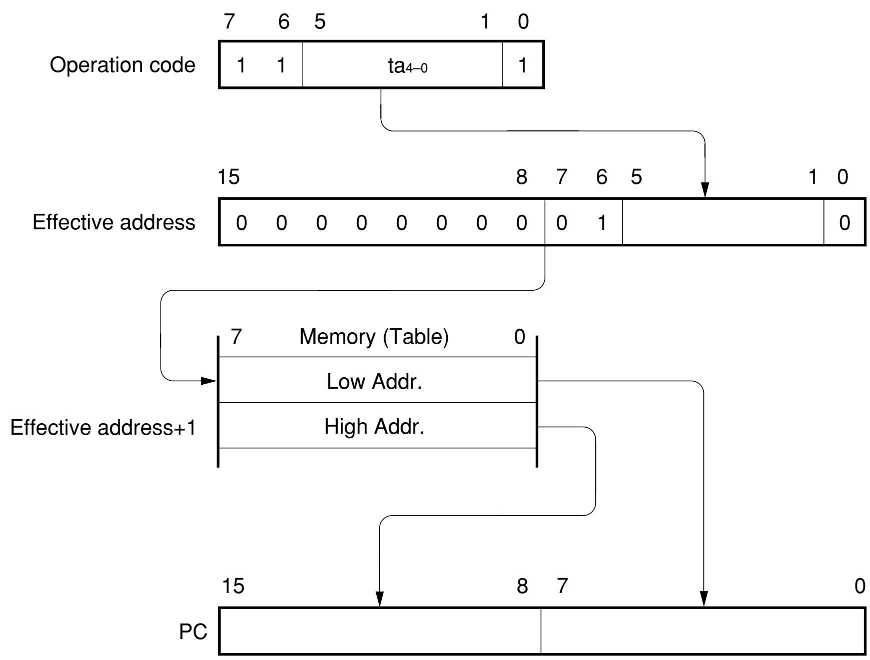
[Function]

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

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

This instruction references the address stored in the memory table from 40H to 7FH, and allows branching to the entire memory space.

[Illustration]



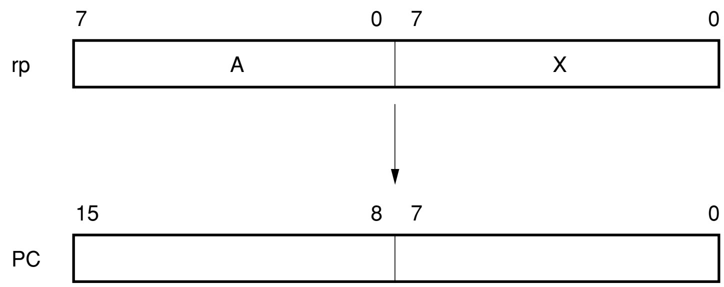
5.3.4 Register addressing

[Function]

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

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

[Illustration]



5.4 Operand Address Addressing

The following various 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 is automatically (implicitly) addressed.

Of the μPD780024A, 780034A, 780024AY, 780034AY Subseries instruction words, the following instructions employ implied addressing.

Instruction	Register to Be Specified by Implied Addressing
MULU	A register for multiplicand and AX register for product storage
DIVUW	AX register for dividend and quotient storage
ADJBA/ADJBS	A register for storage of numeric values which become decimal correction targets
ROR4/ROL4	A register 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 × 8-bit multiply instruction, the product of A register and X register is stored in AX. In this example, the A and AX registers are specified by implied addressing.

5.4.2 Register addressing

[Function]

The general-purpose register to be specified is accessed as an operand with the register specify code (Rn and RPn) of an instruction word in the registered bank specified with the register bank select flag (RBS0 and RBS1). Register addressing is carried out when an instruction with the following operand format is executed. When an 8-bit register is specified, one of the eight registers is specified with 3 bits in the operation code.

[Operand format]

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

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

[Description example]

MOV A, C; when selecting C register as r

Operation code 0 1 1 0 0 0 1 0

Register specify code

INCW DE; when selecting DE register pair as rp

Operation code 1 0 0 0 0 1 0 0

Register specify code

5.4.3 Direct addressing

[Function]

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

[Operand format]

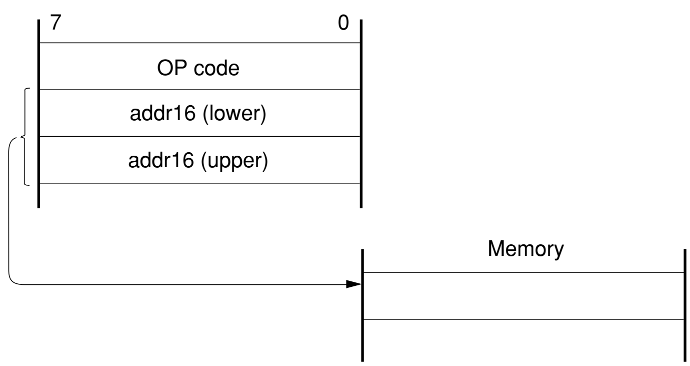
Identifier	Description
addr16	Label or 16-bit immediate data

[Description example]

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

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

[Illustration]



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

If the SFR area (FF00H to FF1FH) where short direct addressing is applied, ports which are frequently accessed in a program and compare and capture registers of the timer/event counter are mapped, and these SFRs can be manipulated with a small number of bytes and clocks.

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

[Operand format]

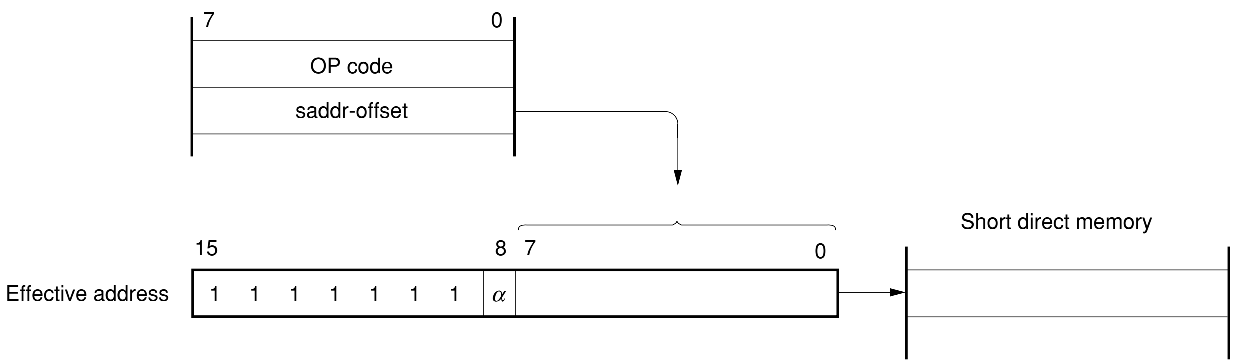
Identifier	Description
saddr	Label or immediate data indicating FE20H to FF1FH
saddrp	Label or immediate data indicating FE20H to FF1FH (even address only)

★ **[Description example]**

MOV 0FE30H, A; when transferring the value in register A to saddr (FE30H)

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

[Illustration]



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

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

5.4.5 Special function register (SFR) addressing

[Function]

The memory-mapped special function register (SFR) is addressed with 8-bit immediate data in an instruction word.

This addressing is applied to the 240-byte spaces FF00H to FFCFH and FFE0H to FFFFH. However, the SFR mapped at FF00H to FF1FH can be accessed with short direct addressing.

[Operand format]

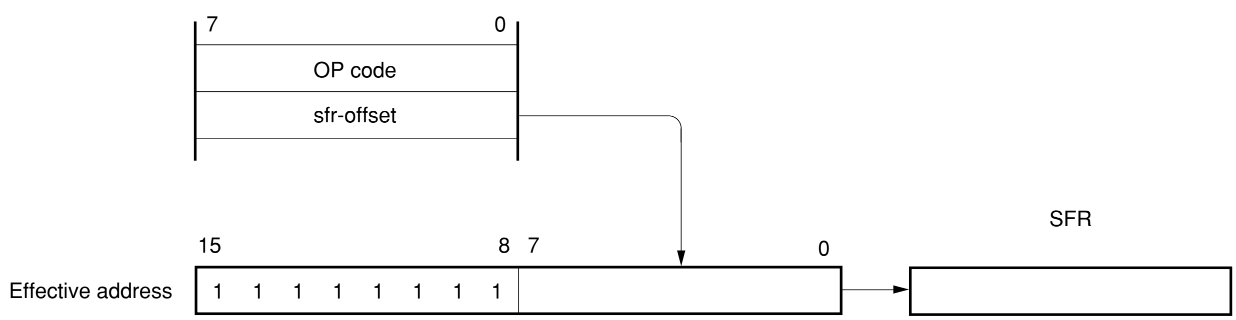
Identifier	Description
sfr	Special function register name
sfrp	16-bit manipulatable special function register name (even address only)

[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]

Register pair contents specified with a register pair specify code in an instruction word of the register bank specified with a register bank select flag (RBS0 and RBS1) serve as an operand address for addressing the memory to be manipulated. This addressing can be carried out for all the memory spaces.

[Operand format]

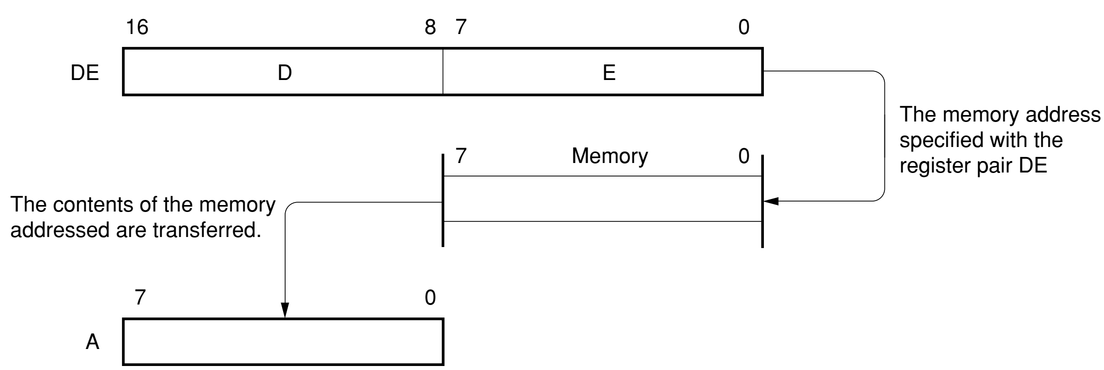
Identifier	Description
–	[DE], [HL]

[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]

8-bit immediate data is added as offset data to the contents of the base register, that is, the HL register pair in an instruction word of the register bank specified by the register bank select flags (RBS0 and RBS1) and the sum is used to address the memory. Addition is performed by expanding the offset data as a positive number to 16 bits. A carry from the 16th bit is ignored. This addressing can be carried out for all the memory spaces.

[Operand format]

Identifier	Description
–	[HL + byte]

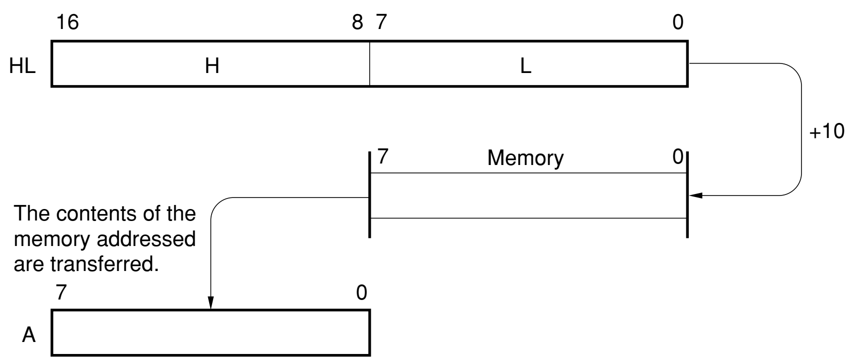
[Description example]

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

Operation code

1 0 1 0 1 1 1 0
0 0 0 1 0 0 0 0

★ **[Illustration]**



5.4.8 Based indexed addressing

[Function]

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

[Operand format]

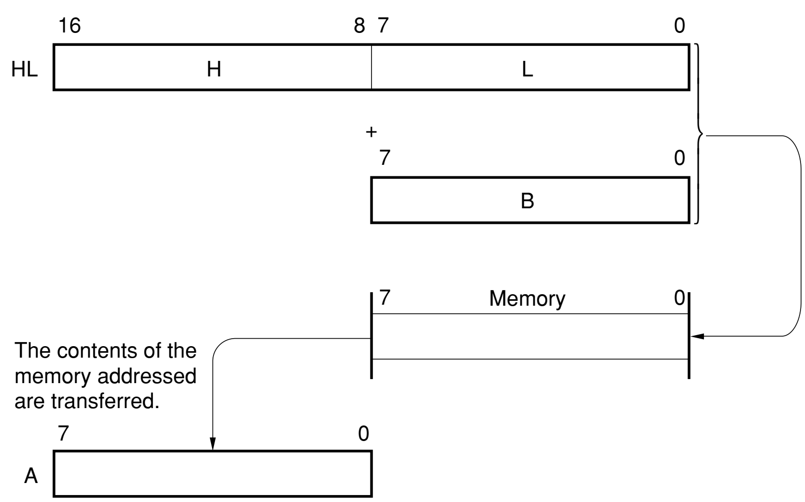
Identifier	Description
–	[HL + B], [HL + C]

[Description example]

In the case of MOV A, [HL + B] (selecting the B register)

Operation code: 1 0 1 0 1 0 1 1

★ **[Illustration]**



5.4.9 Stack addressing

[Function]

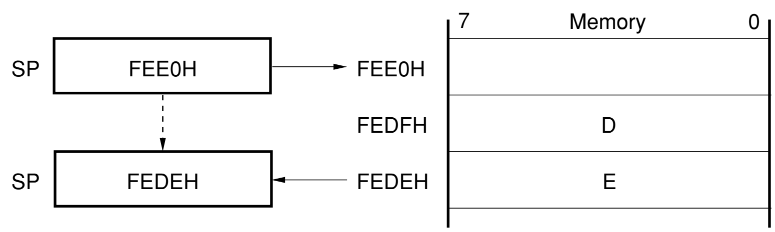
The stack area is indirectly addressed with the stack pointer (SP) contents.
This addressing method is automatically employed when the PUSH, POP, subroutine call and return instructions are executed or the register is saved/reset upon generation of an interrupt request.
Stack addressing can be used to address the internal high-speed RAM area only.

[Description example]

In the case of PUSH DE (saving the DE register)

Operation code	1 0 1 1 0 1 0 1

★ **[Illustration]**



CHAPTER 6 PORT FUNCTIONS

6.1 Port Functions

The μPD780024A, 780034A, 780024AY, and 780034AY Subseries products incorporate eight input ports and 43 I/O ports. Figure 6-1 shows the port configuration. Every port is capable of 1-bit and 8-bit manipulations 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

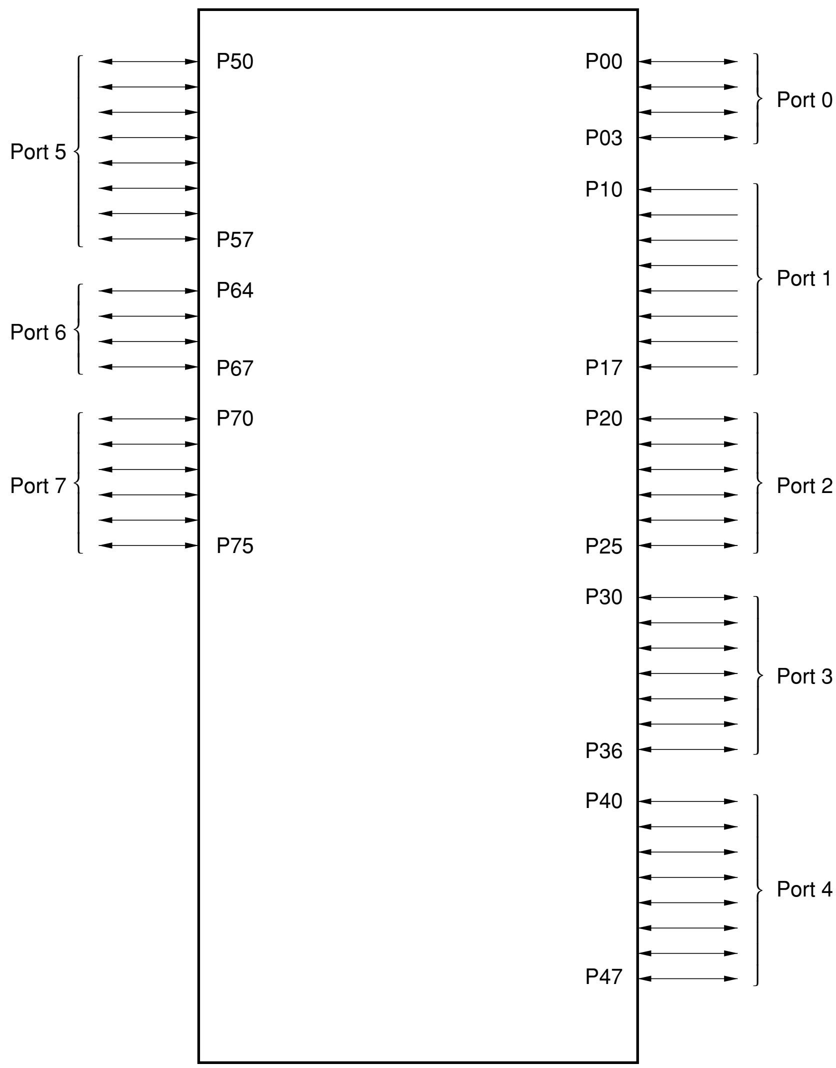


Table 6-1. Port Functions (μPD780024A, 780034A Subseries)

<table>
<tr><th>Pin Name</th><th colspan="2">Function</th><th>Alternate Function</th></tr>
<tr><td>P00</td><td colspan="2" rowspan="4">Port 0
4-bit I/O port.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td>INTP0</td></tr>
<tr><td>P01</td><td>INTP1</td></tr>
<tr><td>P02</td><td>INTP2</td></tr>
<tr><td>P03</td><td>INTP3/ADTRG</td></tr>
<tr><td>P10 to P17</td><td colspan="2">Port 1
8-bit input-only port.</td><td>ANI0 to ANI7</td></tr>
<tr><td>P20</td><td colspan="2" rowspan="6">Port 2
6-bit I/O port.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td>SI30</td></tr>
<tr><td>P21</td><td>SO30</td></tr>
<tr><td>P22</td><td>$\overline{SCK30}$</td></tr>
<tr><td>P23</td><td>RxD0</td></tr>
<tr><td>P24</td><td>TxD0</td></tr>
<tr><td>P25</td><td>ASCK0</td></tr>
<tr><td>P30</td><td rowspan="7">Port 3
7-bit I/O port.
Input/output mode can be specified in 1-bit units.</td><td rowspan="4">N-ch open-drain I/O port.
On-chip pull-up resistor can be specified by mask option (mask ROM version only).
LEDs can be driven directly.</td><td rowspan="4">–</td></tr>
<tr><td>P31</td></tr>
<tr><td>P32</td></tr>
<tr><td>P33</td></tr>
<tr><td>P34</td><td rowspan="3">An on-chip pull-up resistor can be specified by software settings.</td><td>SI31</td></tr>
<tr><td>P35</td><td>SO31</td></tr>
<tr><td>P36</td><td>$\overline{SCK31}$</td></tr>
<tr><td>P40 to P47</td><td colspan="2">Port 4
8-bit I/O port.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be specified by software settings.
Interrupt request flag (KRIF) is set to 1 by falling edge detection.</td><td>AD0 to AD7</td></tr>
<tr><td>P50 to P57</td><td colspan="2">Port 5
8-bit I/O port.
LEDs can be driven directly.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td>A8 to A15</td></tr>
<tr><td>P64</td><td colspan="2" rowspan="4">Port 6
4-bit I/O port.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td>$\overline{RD}$</td></tr>
<tr><td>P65</td><td>$\overline{WR}$</td></tr>
<tr><td>P66</td><td>$\overline{WAIT}$</td></tr>
<tr><td>P67</td><td>ASTB</td></tr>
<tr><td>P70</td><td colspan="2" rowspan="6">Port 7
6-bit I/O port.
Input/output mode can be specified in 1-bit units.
An on-chip pull-up resistor can be used by software settings.</td><td>TI00/TO0</td></tr>
<tr><td>P71</td><td>TI01</td></tr>
<tr><td>P72</td><td>TI50/TO50</td></tr>
<tr><td>P73</td><td>TI51/TO51</td></tr>
<tr><td>P74</td><td>PCL</td></tr>
<tr><td>P75</td><td>BUZ</td></tr>
</table>

Table 6-2. Port Functions (μPD780024AY, 780034AY Subseries)

Pin Name	Function		Alternate Function
P00	Port 0 4-bit I/O port. Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		INTP0
P01			INTP1
P02			INTP2
P03			INTP3/ADTRG
P10 to P17	Port 1 8-bit input-only port.		ANI0 to ANI7
P20	Port 2 6-bit I/O port Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		SI30
P21			SO30
P22			$\overline{SCK30}$
P23			RxD0
P24			TxD0
P25			ASCK0
P30	Port 3 7-bit I/O port. Input/output mode can be specified in 1-bit units.	N-ch open-drain I/O port. On-chip pull-up resistor can be specified by mask option (P30 and P31 are mask ROM version only). LEDs can be driven directly.	–
P31			
P32			SDA0
P33			SCL0
P34		An on-chip pull-up resistor can be used by software settings.	–
P35			
P36			
P40 to P47	Port 4 8-bit I/O port. Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings. Interrupt request 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 mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		A8 to A15
P64	Port 6 4-bit I/O port. Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		$\overline{RD}$
P65			$\overline{WR}$
P66			$\overline{WAIT}$
P67			ASTB
P70	Port 7 6-bit I/O port. Input/output mode can be specified in 1-bit units. An on-chip pull-up resistor can be used by software settings.		TI00/TO0
P71			TI01
P72			TI50/TO50
P73			TI51/TO51
P74			PCL
P75			BUZ

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, 2 to 7) Pull-up resistor option register (PUm: m = 0, 2 to 7)
Port	Total: 51 ports (8 inputs, 43 inputs/outputs)
Pull-up resistor	• Mask ROM version Total: 43 (software control: 39, mask option: 4[Note]) • Flash memory version Total: 39

Note Two mask options for the μPD780024AY and 780034AY Subseries.

6.2.1 Port 0

Port 0 is a 4-bit I/O port with output latch. P00 to P03 pins can specify the input mode/output mode in 1-bit units with port mode register 0 (PM0). An on-chip pull-up resistor of P00 to P03 pins can be used for them in 1-bit units with pull-up resistor option register 0 (PU0).

This port can also be used as an external interrupt request input and A/D converter external trigger input.

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

Figure 6-2 shows a block diagram of port 0.

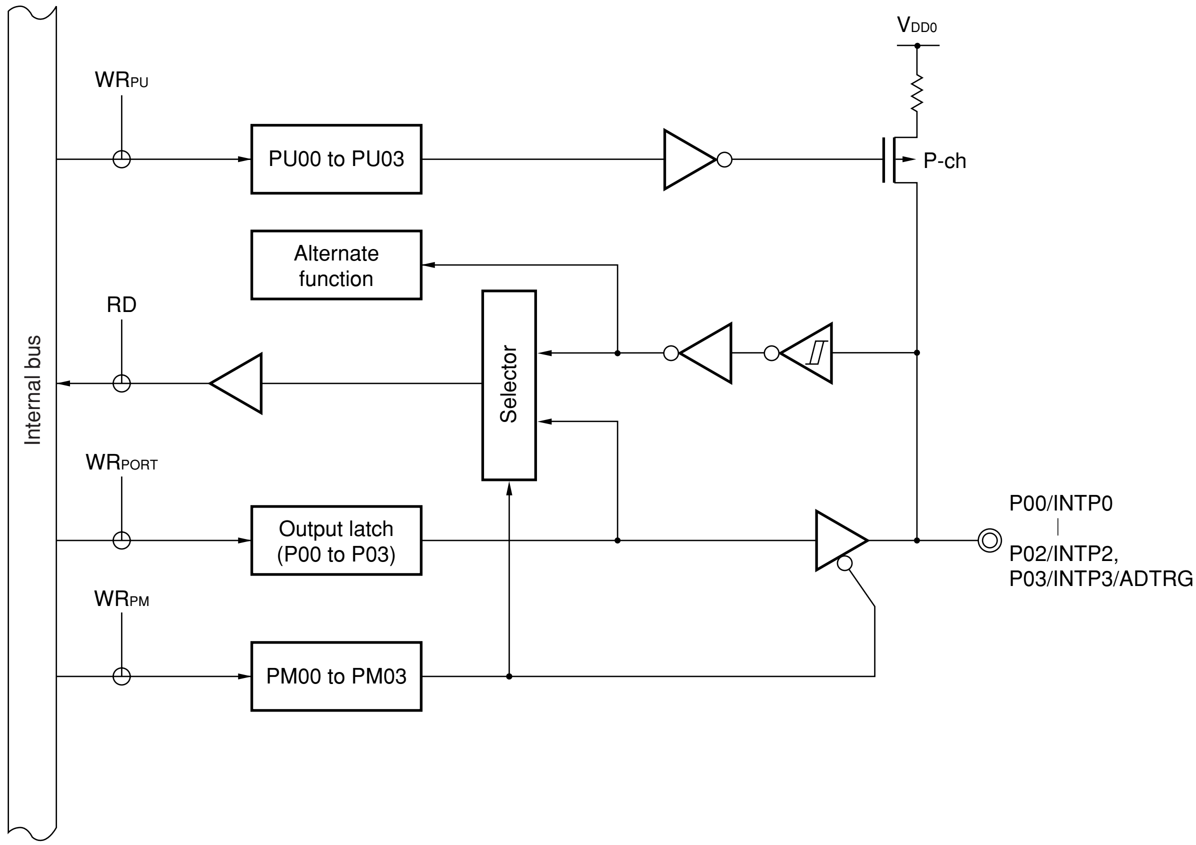
★ **Cautions 1. Port 0 functions alternately as an external interrupt request input pin. If the output mode of the port function is specified and the output level of the port is changed while interrupts are not disabled by the external interrupt rising edge enable register (EGP) and external interrupt falling edge enable register (EGN), the interrupt request flag is set. Thus, when the output mode is used, set the interrupt mask flag to 1.**

★ **2. When the external interrupt request function is switched to the port function, edge detection may be performed. Therefore, clear bit n (EGPn) of EGP and bit n (EGNn) of EGN to 0 before selecting the port mode.**

★ **3. When using P03/INTP3/ADTRG as an A/D converter external trigger input, specify valid edges by setting bits 1 and 2 (EGA00 and EGA01) of A/D converter mode register 0 (ADM0) and set the interrupt mask flag (PMK3) to 1.**

Remark n = 0 to 3

★ **Figure 6-2. Block Diagram of P00 to P03**



PU: Pull-up resistor option register

PM: Port mode register

RD: Port 0 read signal

WR: Port 0 write signal

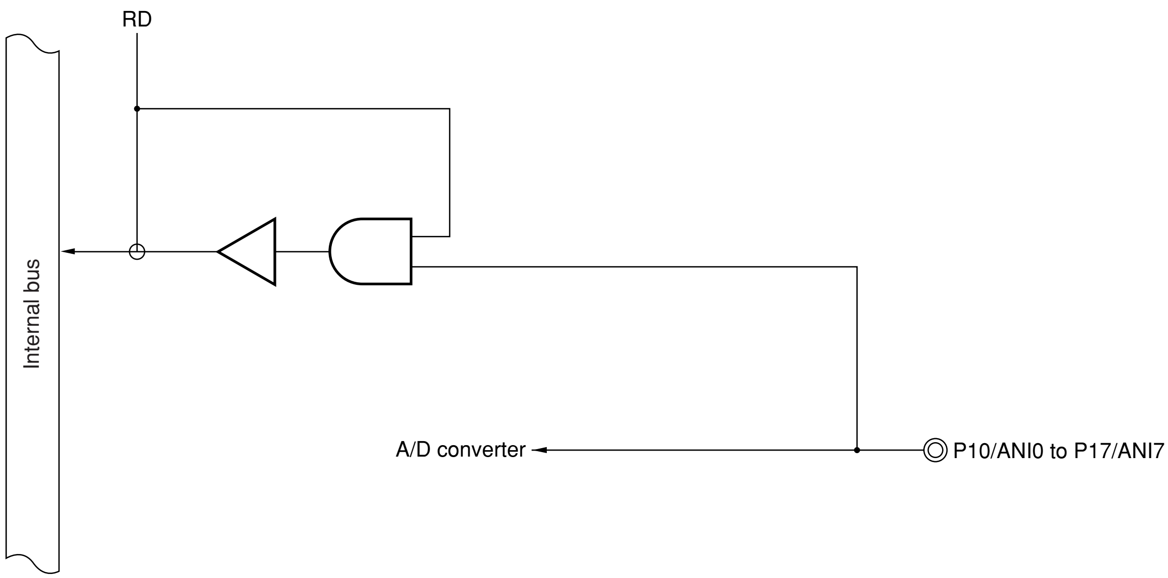
6.2.2 Port 1

Port 1 is an 8-bit input-only port.

This port can also be used as an A/D converter analog input.

Figure 6-3 shows a block diagram of port 1.

★ **Figure 6-3. Block Diagram of P10 to P17**



RD: Port 1 read signal

6.2.3 Port 2

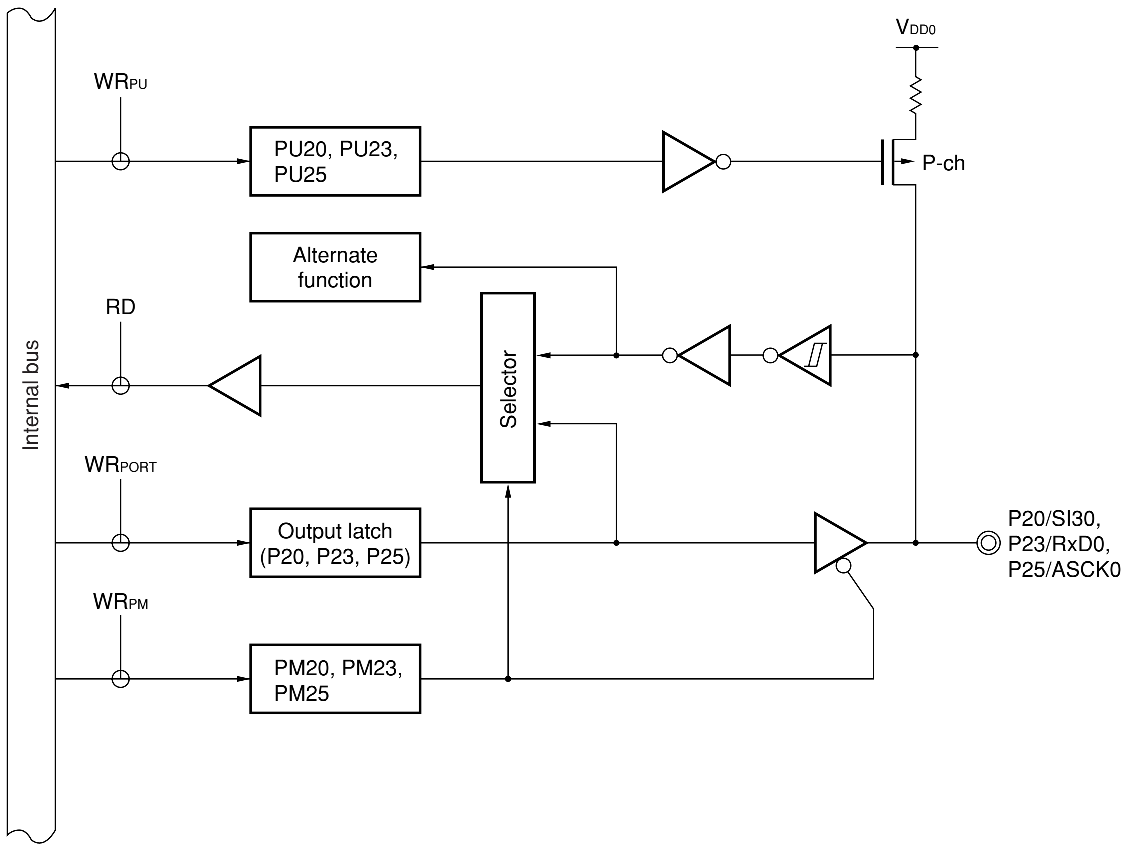
Port 2 is a 6-bit I/O port with output latch. P20 to P25 pins can specify the input mode/output mode in 1-bit units with port mode register 2 (PM2). An on-chip pull-up resistor of P20 to P25 pins can be used for them in 1-bit units with pull-up resistor option register 2 (PU2).

This port has also alternate functions as serial interface data I/O and clock I/O.

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

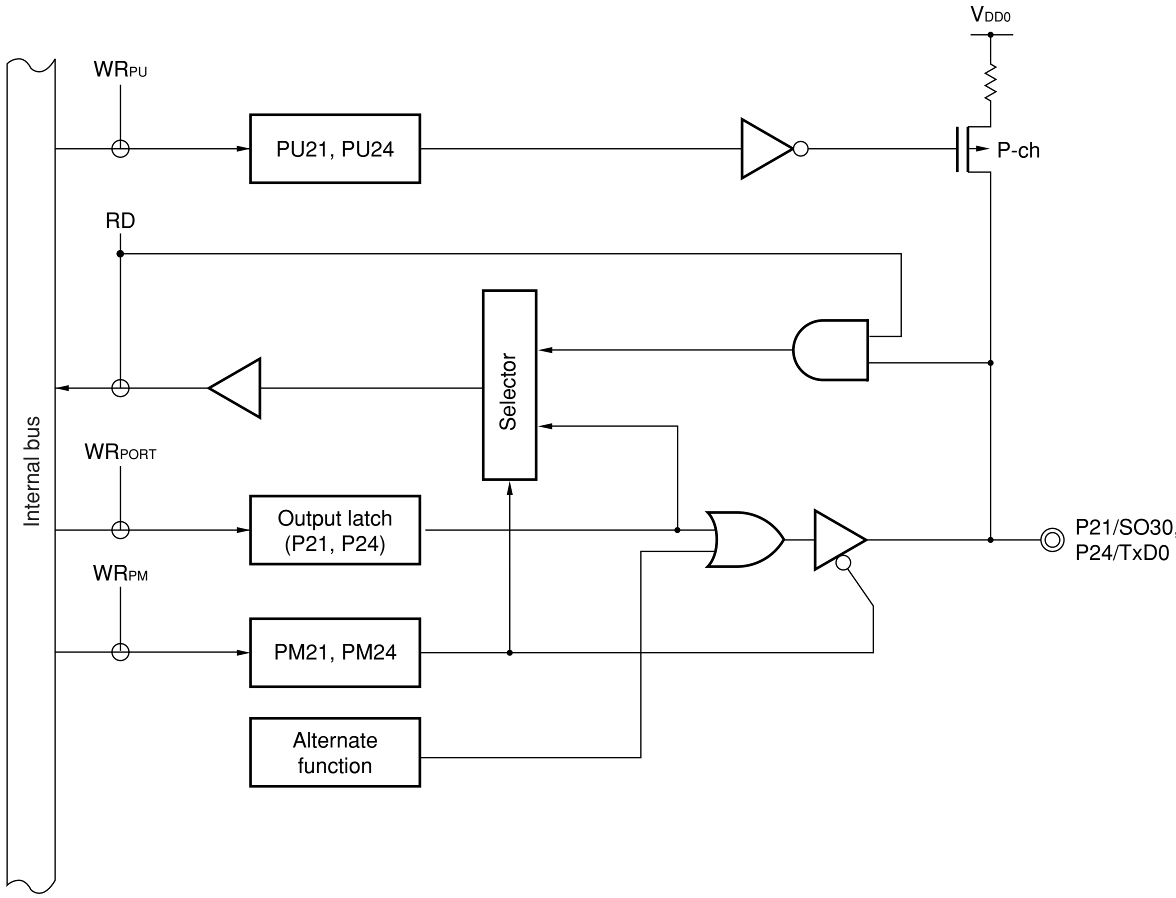
Figures 6-4 to 6-6 show block diagrams of port 2.

★ **Figure 6-4. Block Diagram of P20, P23, and P25**



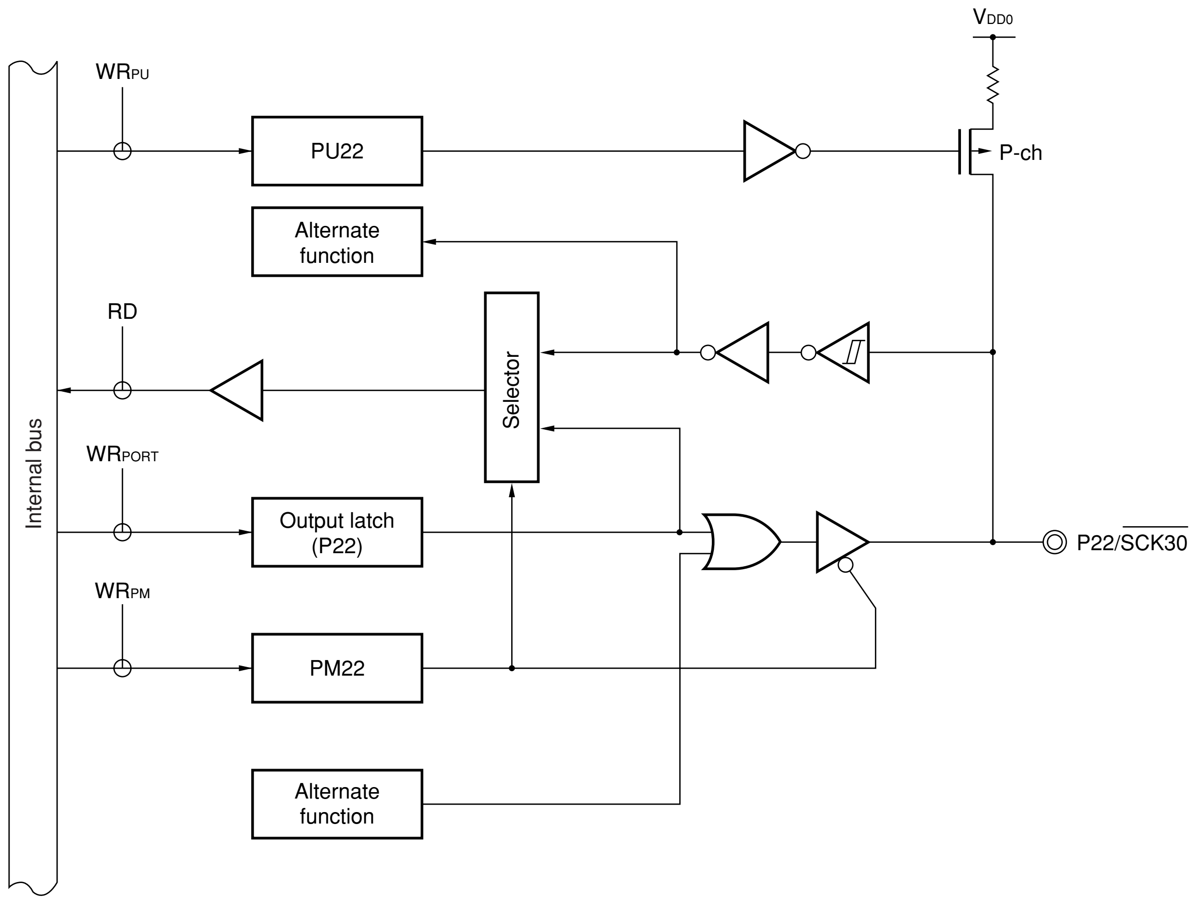
PU: Pull-up resistor option register
PM: Port mode register
RD: Port 2 read signal
WR: Port 2 write signal

★ **Figure 6-5. Block Diagram of P21 and P24**



PU: 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**



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

6.2.4 Port 3 (μPD780024A, 780034A Subseries)

Port 3 is a 7-bit I/O port with output latch. P30 to P36 pins can specify the input mode/output mode in 1-bit units with port mode register 3 (PM3).

This port has the following functions for pull-up resistors. These functions differ depending on the port's higher 3-bit/lower 4-bit, and whether the product is a mask ROM version or a flash memory version.

Table 6-4. Pull-Up Resistor of Port 3 (μPD780024A, 780034A Subseries)

	Higher 3-Bit (P34 to P36 Pins)	Lower 4-Bit (P30 to P33 Pins)
Mask ROM version	An on-chip pull-up resistor can be connected in 1-bit units by PU3	An on-chip pull-up resistor can be specified in 1-bit units by mask option
Flash memory version		On-chip pull-up resistor is not provided

PU3: Pull-up resistor option register 3

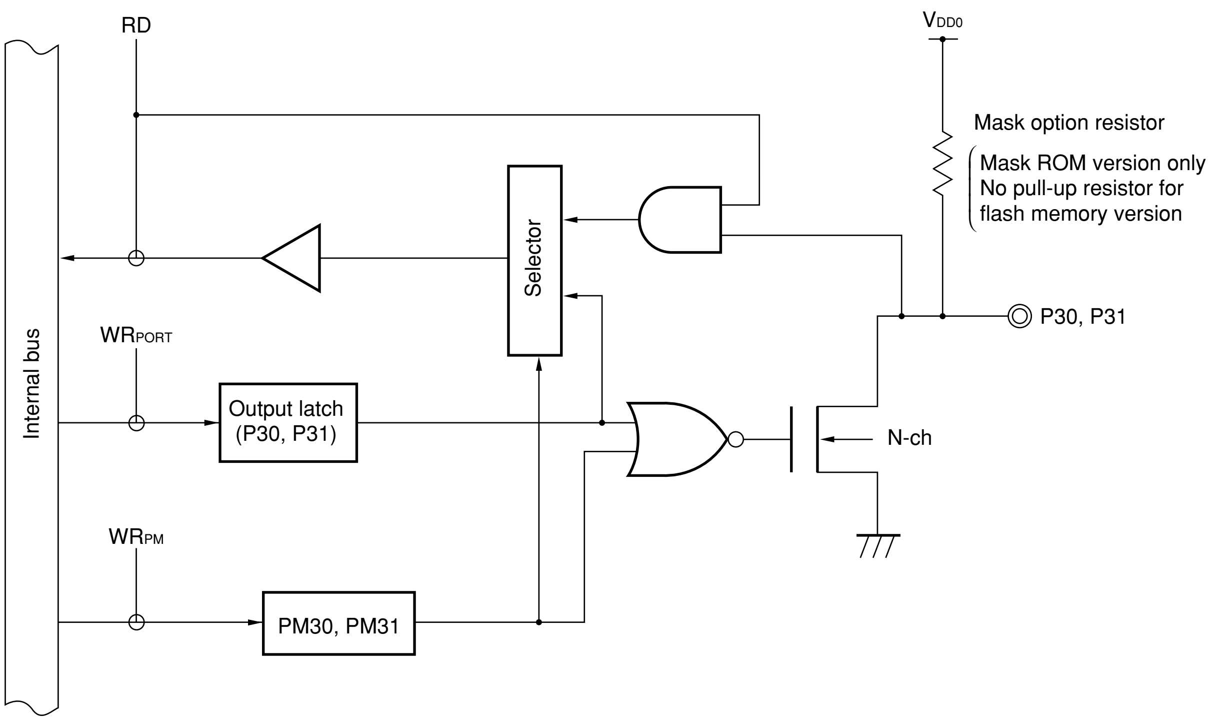
The P30 to P33 pins can drive LEDs directly.

The P34 to P36 pins can also be used for serial interface data I/O and clock I/O.

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

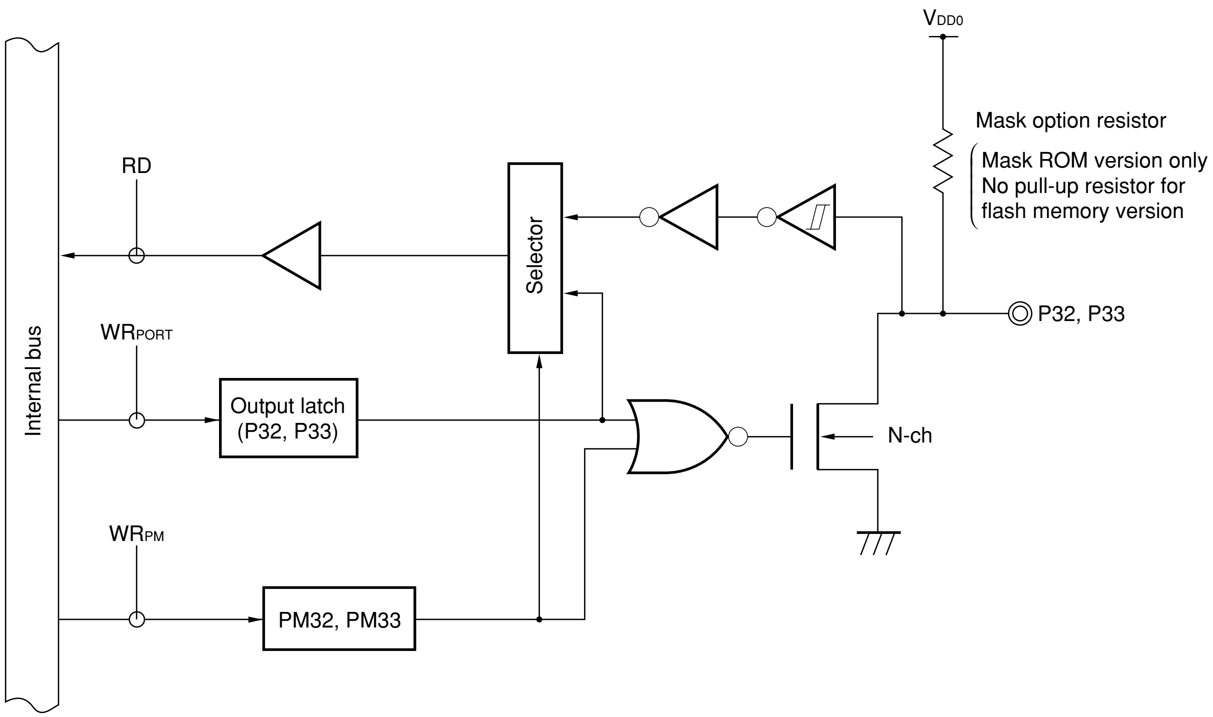
Figures 6-7 to 6-11 show block diagrams of port 3.

★ **Figure 6-7. Block Diagram of P30 and P31 (μPD780024A, 780034A Subseries)**



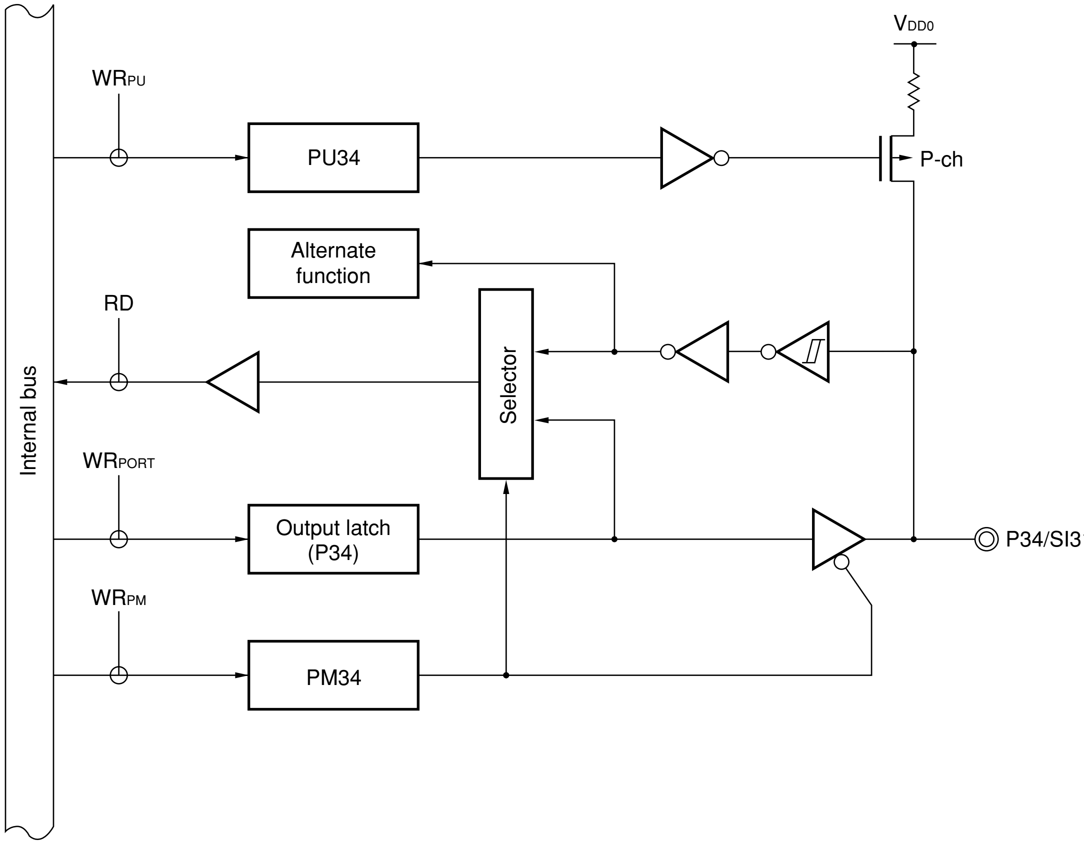
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-8. Block Diagram of P32 and P33 (μPD780024A, 780034A Subseries)**



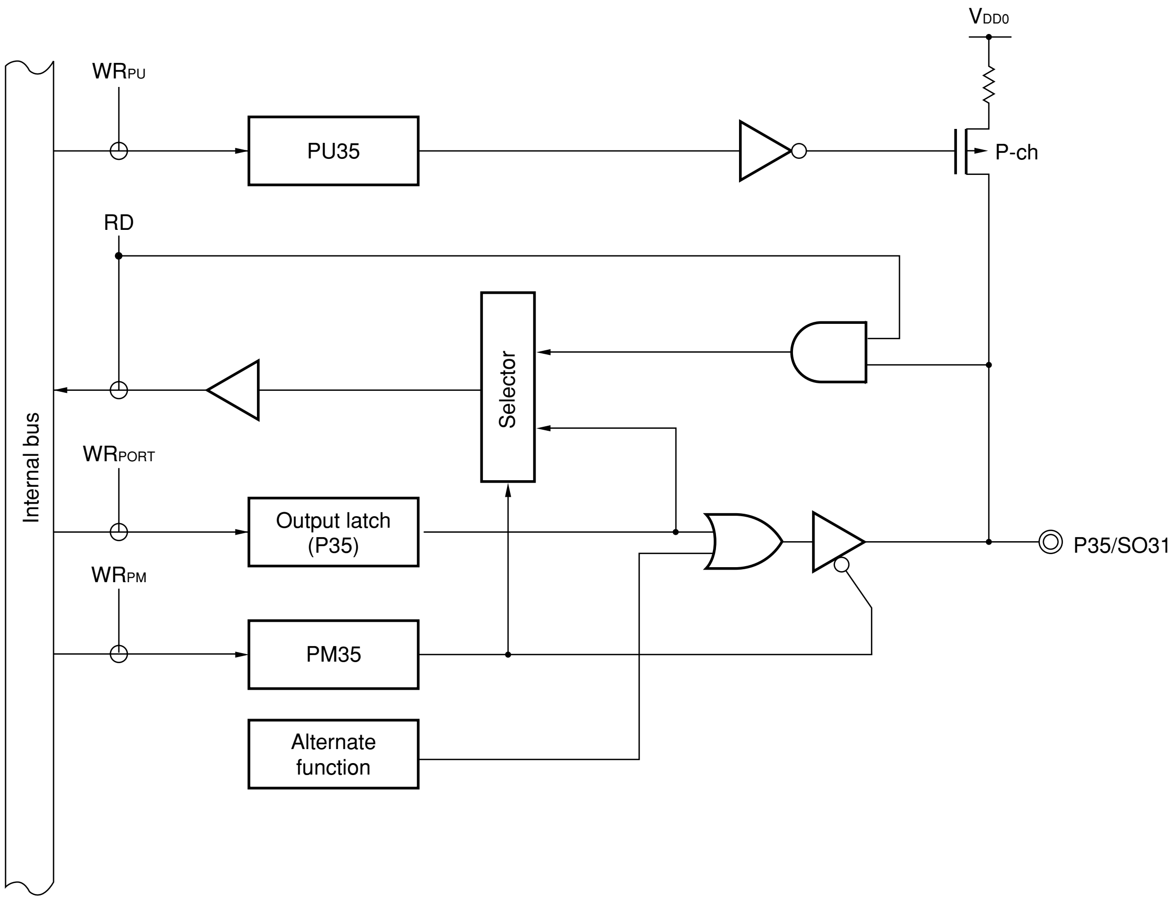
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-9. Block Diagram of P34 (μPD780024A, 780034A Subseries)**



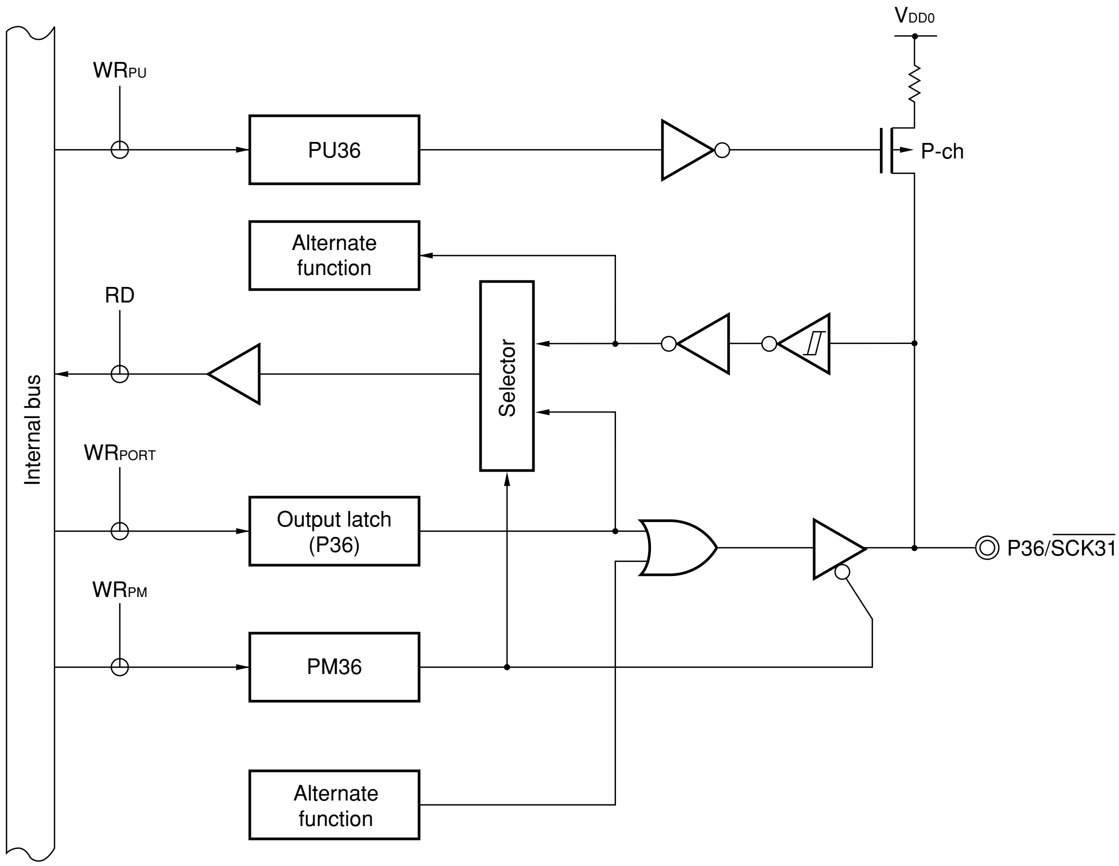
PU: Pull-up resistor option register
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-10. Block Diagram of P35 (μPD780024A, 780034A Subseries)**



PU: Pull-up resistor option register
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-11. Block Diagram of P36 (μPD780024A, 780034A Subseries)**



PU: Pull-up resistor option register
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

6.2.5 Port 3 (μPD780024AY, 780034AY Subseries)

Port 3 is a 7-bit I/O port with output latch. P30 to P36 pins can specify the input mode/output mode in 1-bit units with port mode register 3 (PM3).

This port has the following functions for pull-up resistors. These functions differ depending on port's bits location and mask ROM version/flash memory version.

Table 6-5. Pull-Up Resistor of Port 3 (μPD780024AY, 780034AY Subseries)

<table>
<tr><th></th><th>P34 to P36 Pins</th><th>P30 and P31 Pins</th></tr>
<tr><td>Mask ROM version</td><td rowspan="2">An on-chip pull-up resistor can be connected in 1-bit units by PU3</td><td>An on-chip pull-up resistor can be specified in 1-bit units by mask option</td></tr>
<tr><td>Flash memory version</td><td>On-chip pull-up resistor is not provided</td></tr>
</table>

PU3: Pull-up resistor option register 3

Caution P32 and P33 pins have no pull-up resistor.

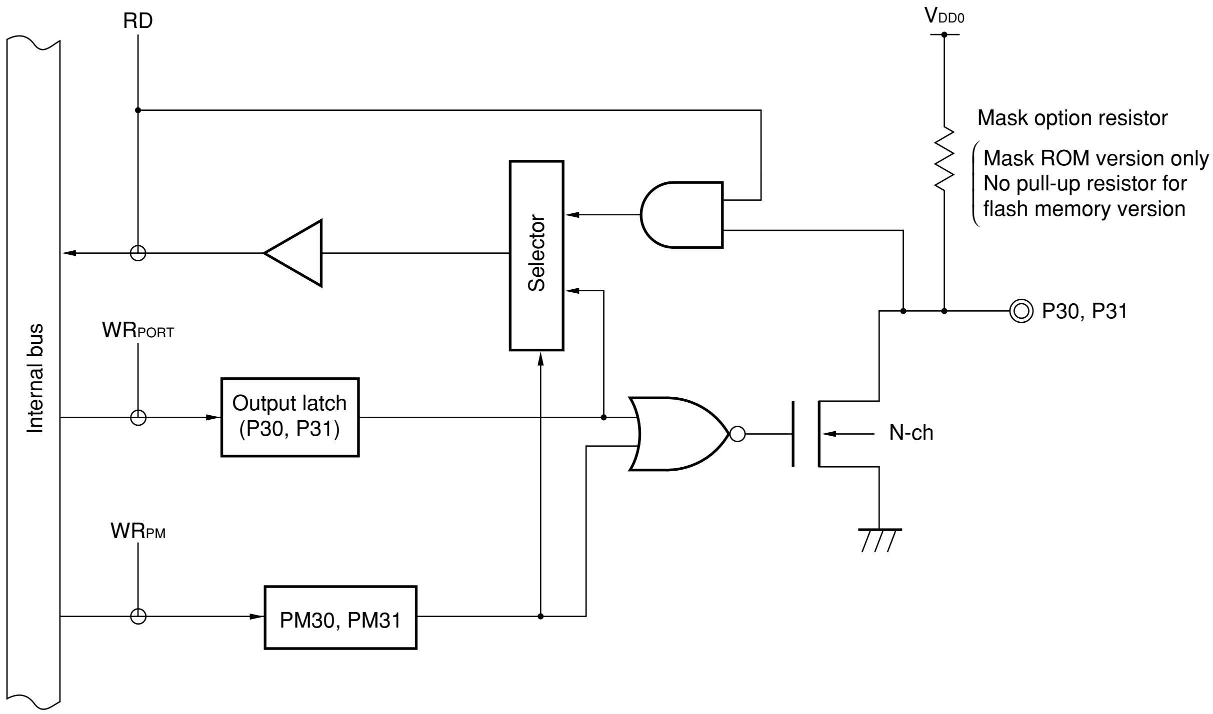
The P30 to P33 pins can drive LEDs directly.

The P32 and P33 pins can also be used for serial interface data I/O and clock I/O.

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

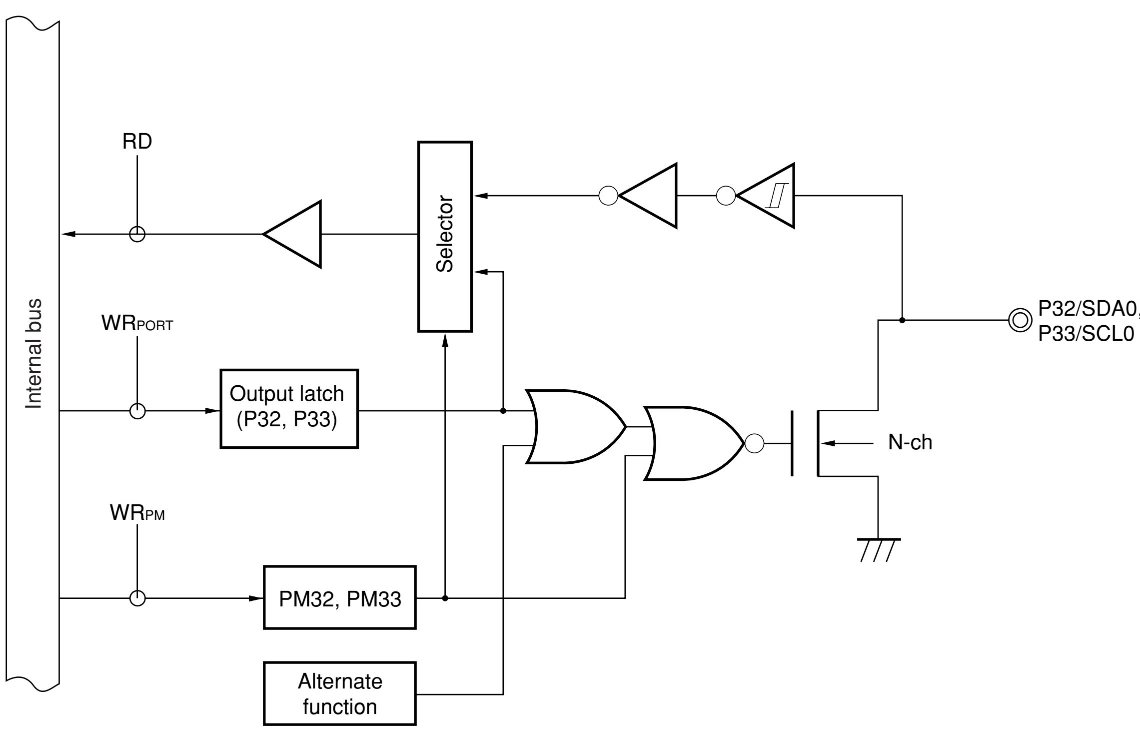
Figures 6-12 to 6-15 show block diagrams of port 3.

★ **Figure 6-12. Block Diagram of P30 and P31 (μPD780024AY, 780034AY Subseries)**



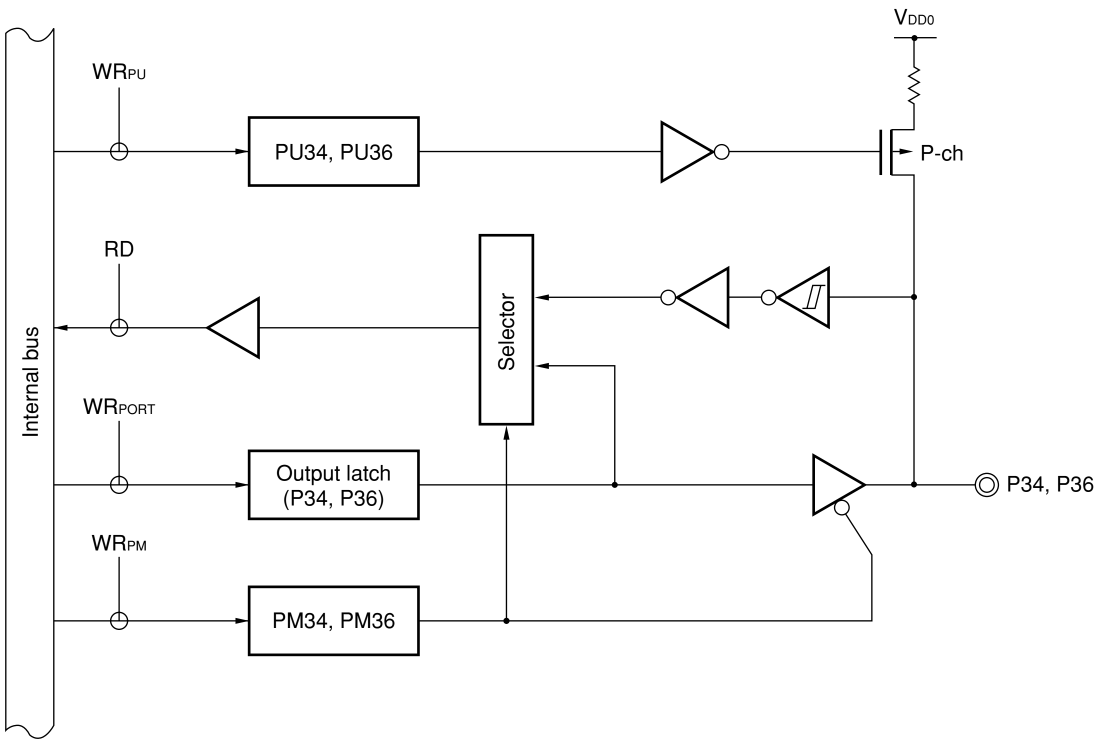
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-13. Block Diagram of P32 and P33 (μPD780024AY, 780034AY Subseries)**



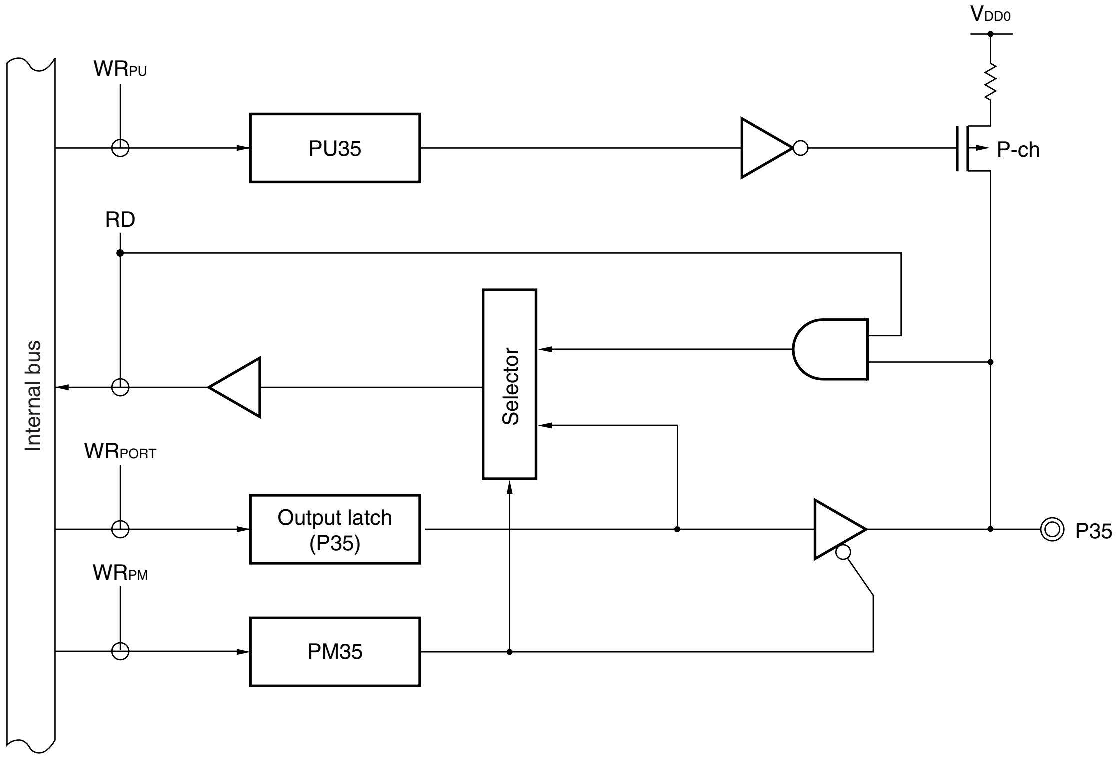
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-14. Block Diagram of P34 and P36 (μPD780024AY, 780034AY Subseries)**



PU: Pull-up resistor option register
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

★ **Figure 6-15. Block Diagram of P35 (μPD780024AY, 780034AY Subseries)**



PU: 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 output latch. The P40 to P47 pins can specify the input mode/output mode in 1-bit units with port mode register 4 (PM4). An on-chip pull-up resistor of P40 to P47 pins can be used for them in 1-bit units with pull-up resistor option register 4 (PU4).

The interrupt request flag (KRIF) can be set to 1 by detecting falling edges.

This port can also be used as an address/data bus in external memory expansion mode.

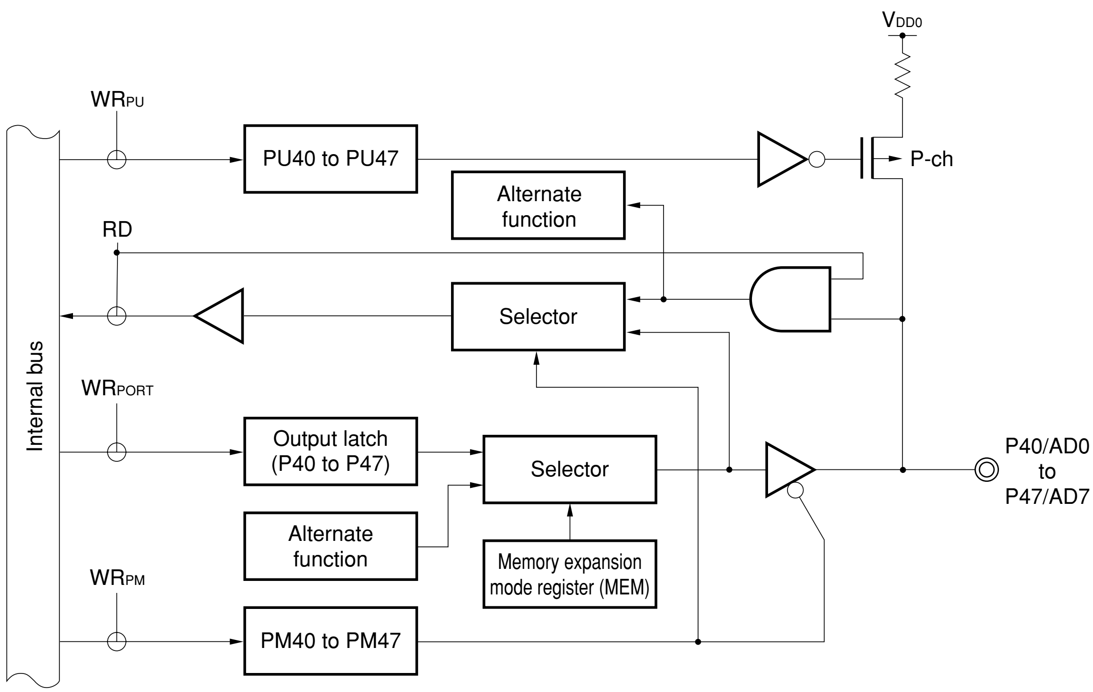
$\overline{\text{RESET}}$ input sets port 4 to input mode.

Figures 6-16 and 6-17 show a block diagram of port 4 and block diagram of the falling edge detector, respectively.

★ **Cautions 1. The internal pull-up resistor is not disconnected even if the external memory expansion mode is set when PU4n = 1 (n = 0 to 7).**

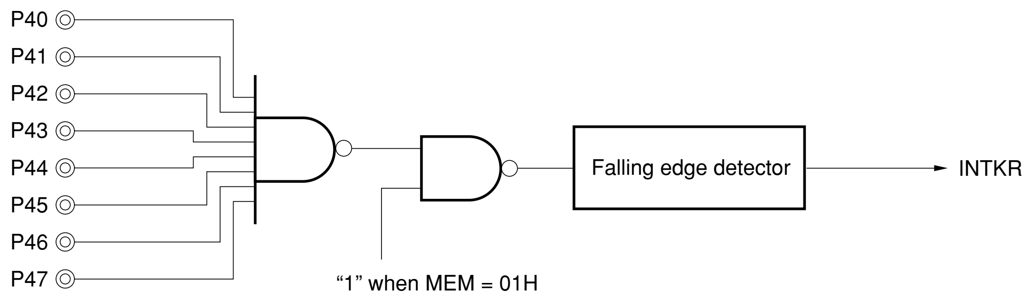
2. When using the falling edge detection interrupt (INTKR), be sure to set the memory expansion mode register (MEM) to 01H.

★ **Figure 6-16. Block Diagram of P40 to P47**



PU: Pull-up resistor option register
PM: Port mode register
RD: Port 4 read signal
WR: Port 4 write signal

Figure 6-17. Block Diagram of Falling Edge Detector



6.2.7 Port 5

Port 5 is an 8-bit I/O port with output latch. The P50 to P57 pins can specify the input mode/output mode in 1-bit units with port mode register 5 (PM5). An on-chip pull-up resistor of P50 to P57 pins can be used for them in 1-bit units with pull-up resistor option register 5 (PU5).

Port 5 can drive LEDs directly.

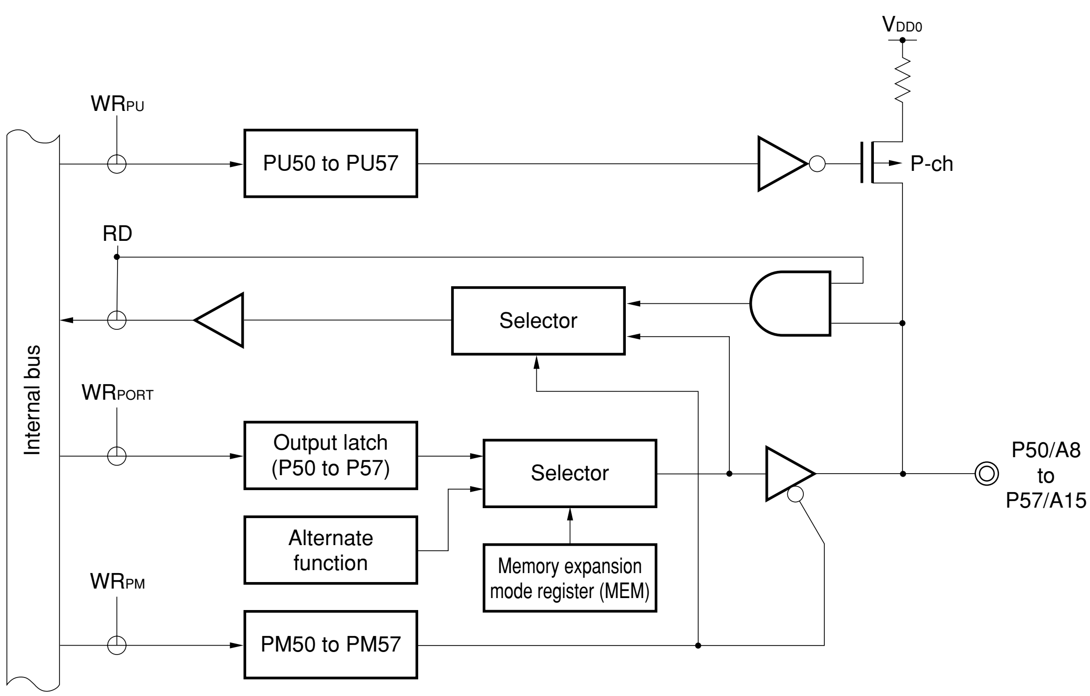
This port can also be used as an address bus in external memory expansion mode.

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

Figure 6-18 shows a block diagram of port 5.

★ **Caution The internal pull-up resistor is not disconnected even if the external memory expansion mode is set when PU5n = 1 (n = 0 to 7).**

★ **Figure 6-18. Block Diagram of P50 to P57**



PU: 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 a 4-bit I/O port with output latch. The P64 to P67 pins can specify the input mode/output mode in 1-bit units with port mode register 6 (PM6). An on-chip pull-up resistor of P64 to P67 pins can be used for them in 1-bit units with pull-up resistor option register 6 (PU6).

This port can also be used as a control signal output in external memory expansion mode.

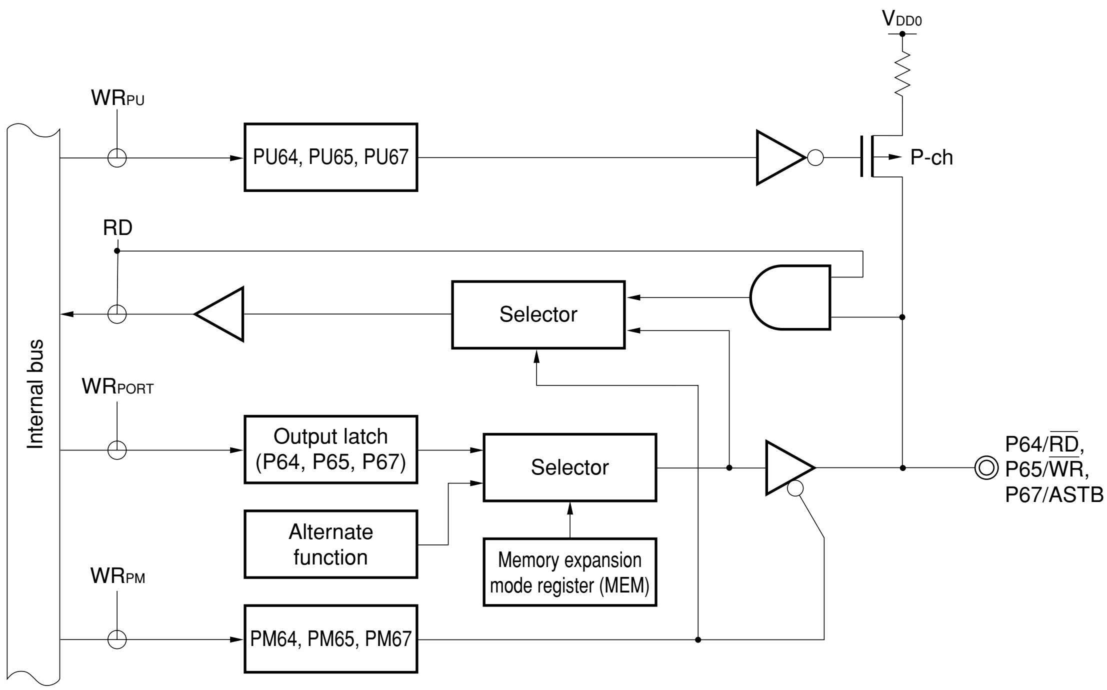
$\overline{\text{RESET}}$ input sets port 6 to input mode.

Figures 6-19 and 6-20 show block diagrams of port 6.

★ **Cautions 1. The internal pull-up resistor is not disconnected even if the external memory expansion mode is set when PU6n = 1 (n = 4 to 7).**

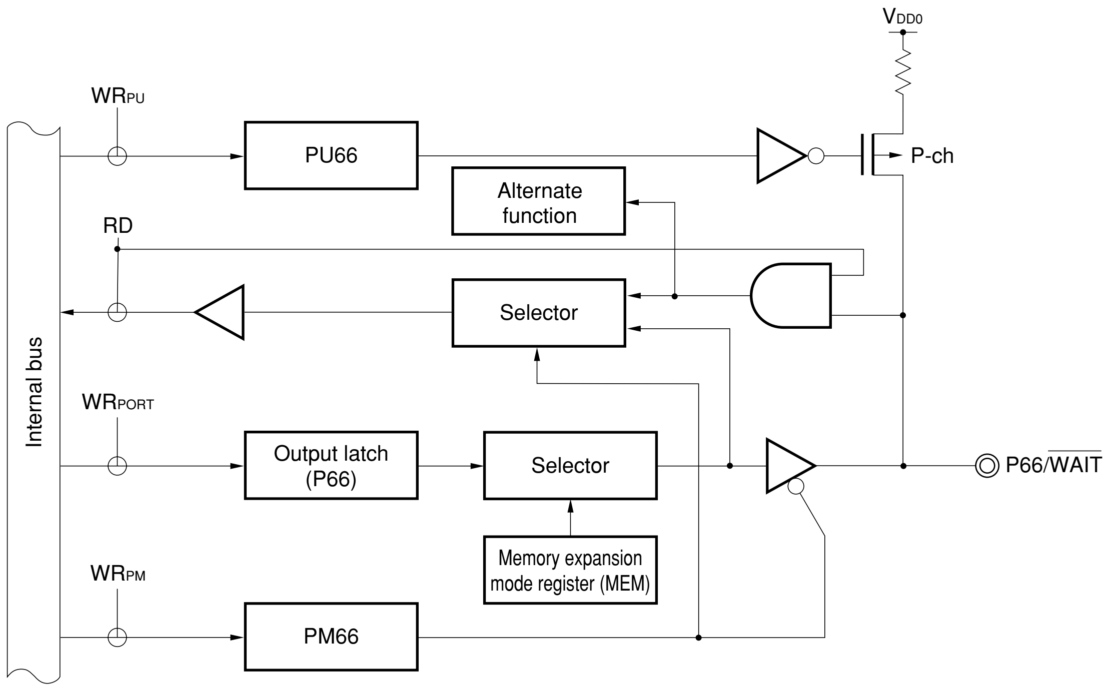
2. When external wait is not used in external memory expansion mode, P66 can be used as an I/O port.

★ **Figure 6-19. Block Diagram of P64, P65, and P67**



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

★ **Figure 6-20. Block Diagram of P66**



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

6.2.9 Port 7

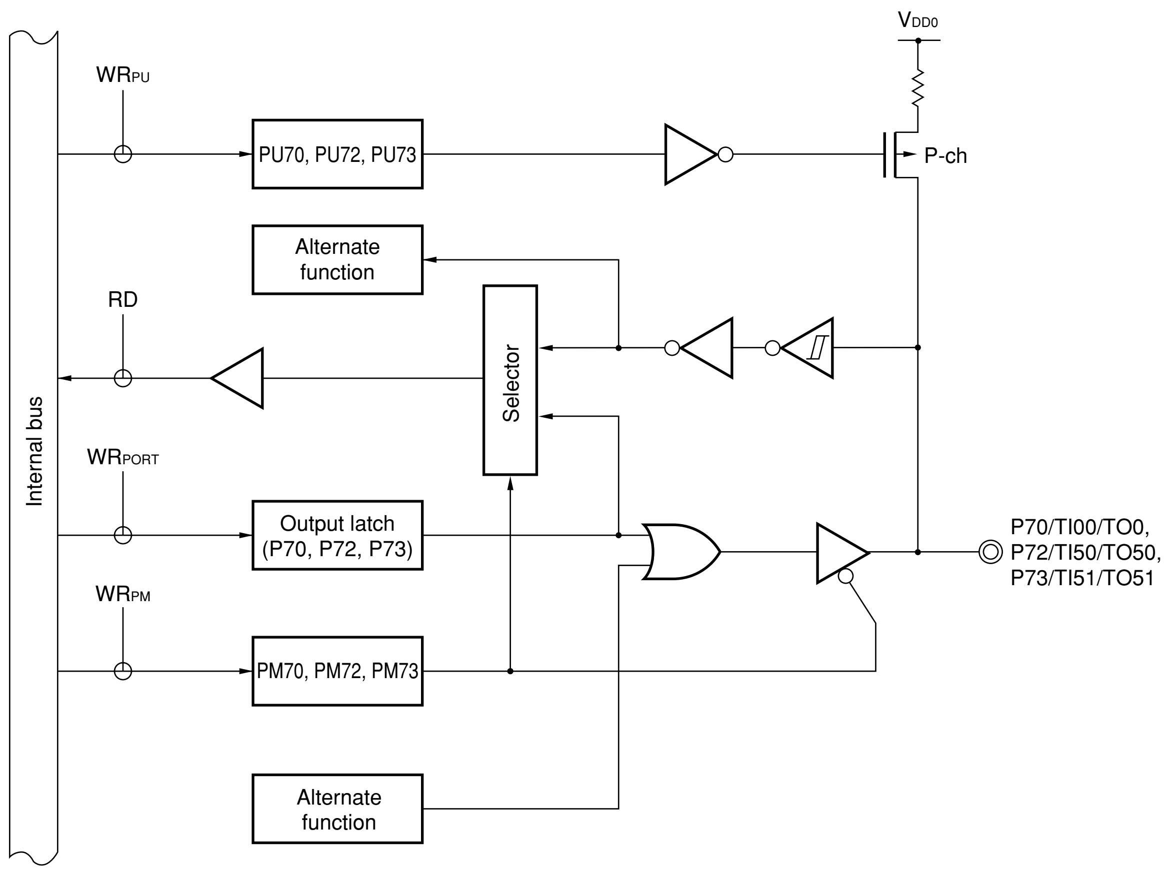
Port 7 is a 6-bit I/O port with output latch. The P70 to P75 pins can specify the input mode/output mode in 1-bit units with port mode register 7 (PM7). An on-chip pull-up resistor of P70 to P75 pins can be used for them in 1-bit units with pull-up resistor option register 7 (PU7).

This port can also be used as a timer I/O, clock output, and buzzer output.

$\overline{\text{RESET}}$ input sets the input mode.

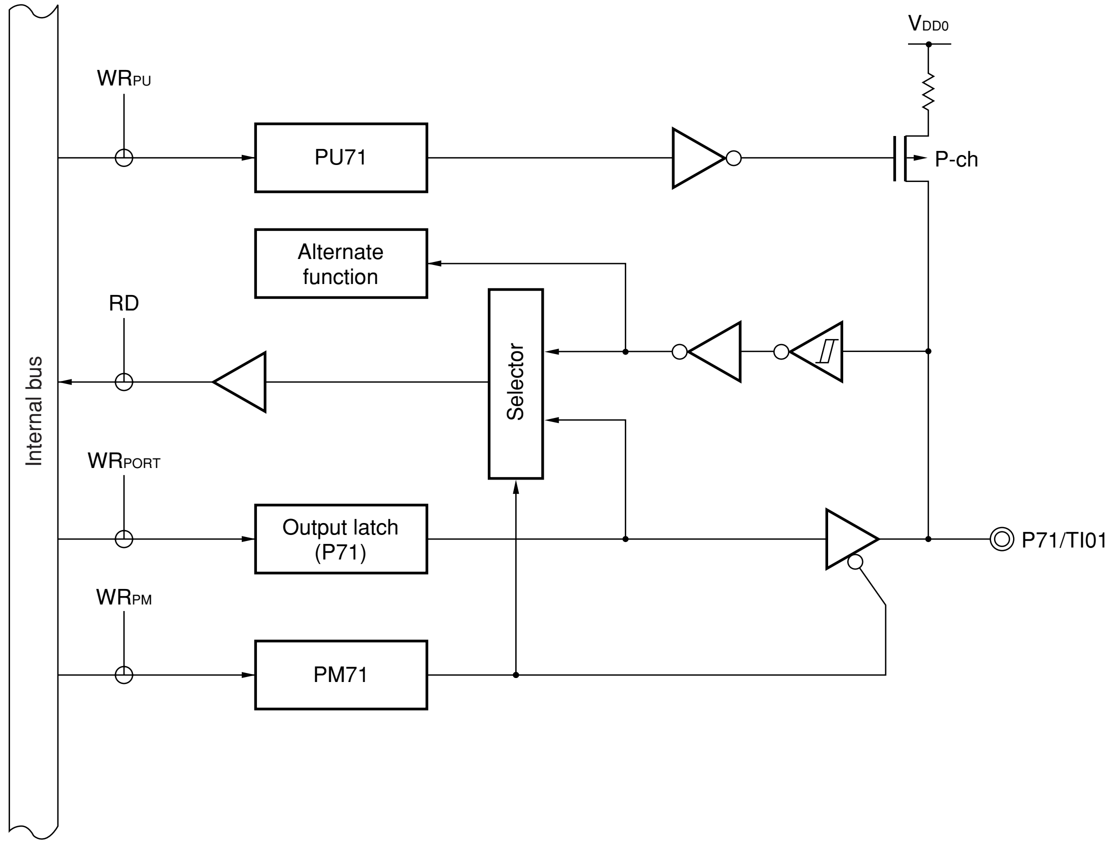
Figures 6-21 to 6-23 show block diagrams of port 7.

★ **Figure 6-21. Block Diagram of P70, P72, and P73**



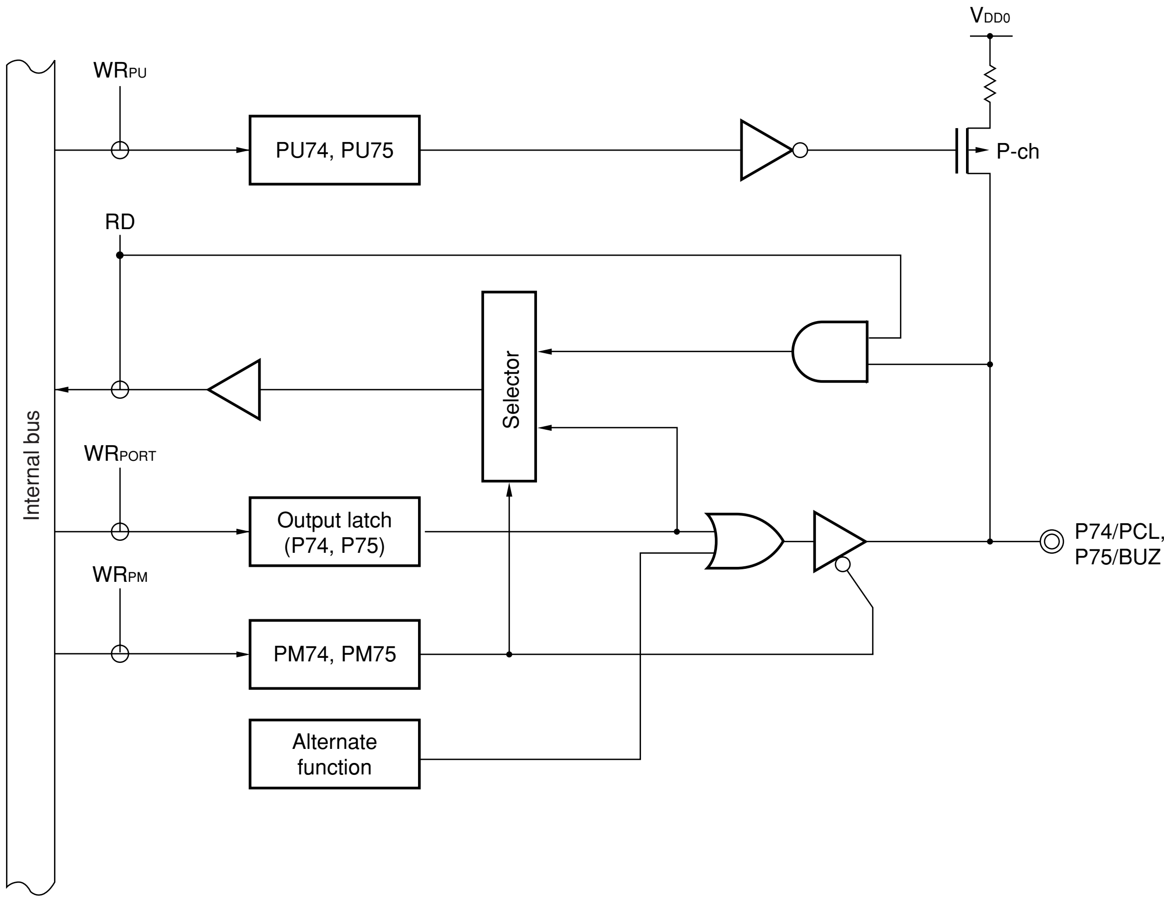
PU: Pull-up resistor option register
PM: Port mode register
RD: Port 7 read signal
WR: Port 7 write signal

★ **Figure 6-22. Block Diagram of P71**



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

★ **Figure 6-23. Block Diagram of P74 and P75**



PU: Pull-up resistor option register

PM: Port mode register

RD: Port 7 read signal

WR: Port 7 write signal

6.3 Port Function Control Registers

The following two types of registers control the ports.

- Port mode registers (PM0, PM2 to PM7)
- Pull-up resistor option registers (PU0, PU2 to PU7)

(1) Port mode registers (PM0, PM2 to PM7)

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

PM0 and PM2 to PM7 are independently set by a 1-bit or 8-bit memory manipulation instruction.

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

When using a port pin as its alternate-function pin, set the port mode registers and output latches as shown in Table 6-6.

★ **Cautions 1. Pins P10 to P17 are input-only pins.**

2. Port 0 functions alternately as an external interrupt request input pin. If the output mode of the port function is specified and the output level of the port is changed while interrupts are not disabled by the external interrupt rising edge enable register (EGP) and external interrupt falling edge enable register (EGN), the interrupt request flag is set. When the output mode is used, therefore, the interrupt mask flag should be set to 1 beforehand.

3. If a port has an alternate function pin and it is used as an alternate output function, clear the corresponding output latches (P0 and P2 to P7) to 0.

Figure 6-24. Format of Port Mode Register (PM0, PM2 to PM7)

Address: FF20H After Reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM0	1	1	1	1	PM03	PM02	PM01	PM00

Address: FF22H After Reset: FFH R/W

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

Address: FF23H After Reset: FFH R/W

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

Address: FF24H After Reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40

Address: FF25H After Reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50

Address: FF26H After Reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM6	PM67	PM66	PM65	PM64	1	1	1	1

Address: FF27H After Reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM7	1	1	PM75	PM74	PM73	PM72	PM71	PM70

PMmn	Pmn Pin I/O Mode Selection (m = 0, 2 to 7; n = 0 to 7)
0	Output mode (Output buffer on)
1	Input mode (Output buffer off)

★ **Table 6-6. Port Mode Registers and Output Latch Settings When Alternate Function Is Used (1/2)**

Pin Name	Alternate Function		PM××	P××
	Name	I/O		
P00 to P02	INTP0 to INTP2	Input	1	×
P03	INTP3	Input	1	×
	ADTRG	Input	1	×
P10 to P17	ANI0 to ANI7	Input	1 (fix)	×
P20	SI30	Input	1	×
P21	SO30	Output	0	0
P22	$\overline{\text{SCK30}}$	Input	1	×
		Output	0	0
P23	RxD0	Input	1	×
P24	TxD0	Output	0	0
P25	ASCK0	Input	1	×
P32	SDA0[Note 1]	I/O	0	0
P33	SCL0[Note 1]	I/O	0	0
P34	SI31	Input	1	×
P35	SO31	Output	0	0
P36	$\overline{\text{SCK31}}$	Input	1	×
		Output	0	0
P40 to P47	AD0 to AD7	I/O	×[Note 2]	
P50 to P57	A8 to A15	Output	×[Note 2]	
P64	$\overline{\text{RD}}$	Output	×[Note 2]	
P65	$\overline{\text{WR}}$	Output	×[Note 2]	
P66	$\overline{\text{WAIT}}$	Input	1[Note 2]	×[Note 2]
P67	ASTB	Output	×[Note 2]	

Notes 1. μPD780024AY, 780034AY Subseries only

2. When using the P40 to P47, P50 to P57, and P64 to P67 pins as alternate-function pins, set the function using the memory expansion mode register (MEM).

Remark ×: Don't care
PM××: Port mode register
P××: Port output latch

★ **Table 6-6. Port Mode Registers and Output Latch Settings When Alternate Function Is Used (2/2)**

Pin Name	Alternate Function		PM××	P××
	Name	I/O		
P70	TI00	Input	1	×
	TO0	Output	0	0
P71	TI01	Input	1	×
P72	TI50	Input	1	×
	TO50	Output	0	0
P73	TI51	Input	1	×
	TO51	Output	0	0
P74	PCL	Output	0	0
P75	BUZ	Output	0	0

Remark ×: Don’t care
PM××: Port mode register
P××: Port output latch

(2) Pull-up resistor option registers (PU0, PU2 to PU7)

These registers are used to set whether to use an on-chip pull-up resistor at each port or not. By setting PU0 and PU2 to PU7, the on-chip pull-up resistors of the port pins corresponding to the bits in PU0 and PU2 to PU7 can be used.

PU0 and PU2 to PU7 are independently set by a 1-bit or 8-bit memory manipulation instruction.

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

Cautions 1. The P10 to P17 pins do not incorporate a pull-up resistor.

2. Pins P30 to P33 (in μPD780024AY and 780034AY Subseries, P30 and P31 pins) can be used with pull-up resistor by mask option only for mask ROM version.

3. When PUm is set to 1, the on-chip pull-up resistor is connected irrespective of the input/output mode. When using in output mode, clear the bit of PUm to 0 (m = 0, 2 to 7).

Figure 6-25. Format of Pull-Up Resistor Option Register (PU0, PU2 to PU7)

Address: FF30H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PU0	0	0	0	0	PU03	PU02	PU01	PU00

Address: FF32H After Reset: 00H R/W

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

Address: FF33H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PU3	0	PU36	PU35	PU34	0	0	0	0

Address: FF34H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PU4	PU47	PU46	PU45	PU44	PU43	PU42	PU41	PU40

Address: FF35H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PU5	PU57	PU56	PU55	PU54	PU53	PU52	PU51	PU50

Address: FF36H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PU6	PU67	PU66	PU65	PU64	0	0	0	0

Address: FF37H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PU7	0	0	PU75	PU74	PU73	PU72	PU71	PU70

PUmn	Pmn Pin On-Chip Pull-Up Resistor Selection (m = 0, 2 to 7; n = 0 to 7)
0	On-chip pull-up resistor not used
1	On-chip pull-up resistor used

6.4 Port Function Operations

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

Caution In the case of 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.

6.4.1 Writing to I/O port

(1) Output mode

A value is written to the output latch by a transfer instruction, and the output latch contents are output from the pin.

Once data is written to the output latch, it is retained until data is written to the output latch again.

The output latch data is cleared by reset.

(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.

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.

The output latch data is cleared by reset.

(2) Input mode

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

6.5 Selection of Mask Option

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

Table 6-7. Comparison Between Mask ROM Version and Flash Memory Version

Pin Name	Mask ROM Version	Flash Memory Version
Mask option for pins P30 to P33[Note]	On-chip pull-up resistors can be specified in 1-bit units.	An on-chip pull-up resistor is not provided.

Note For μPD780024AY and 780034AY Subseries products, only the P30 and P31 pins can incorporate a 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 a clock with the following frequencies.

- 1.0 to 8.38 MHz: Conventional product of μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A and μPD780021AY, 780022AY, 780023AY, 780024AY, 780031AY, 780032AY, 780033AY, 780034AY, 78F0034A, 78F0034AY, 78F0034BY
- 1.0 to 12 MHz: Expanded-specification product of μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A and μPD78F0034B

Oscillation can be stopped by executing the STOP instruction or setting the processor clock control register (PCC).

(2) Subsystem clock oscillator

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

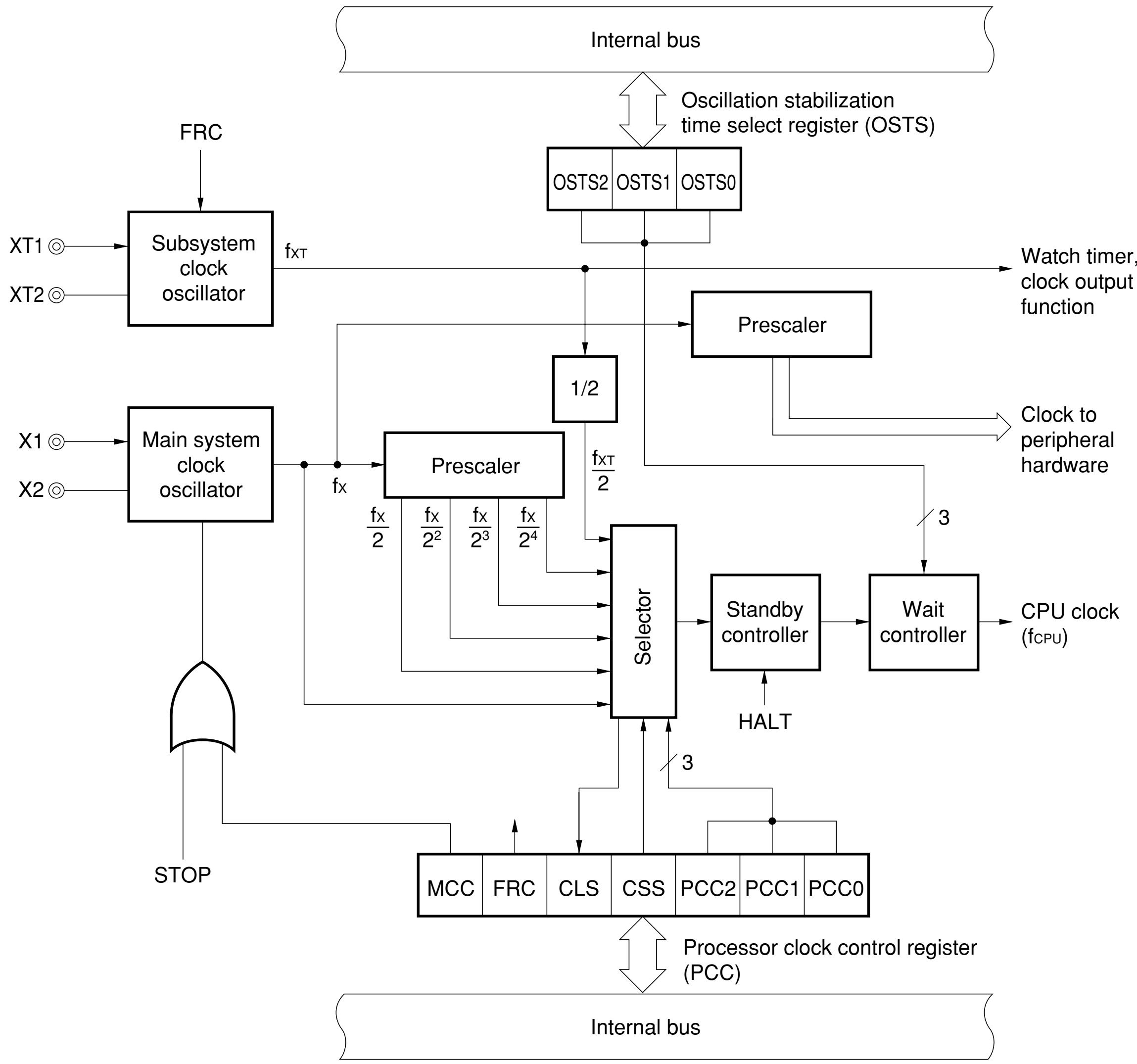
7.2 Clock Generator Configuration

The clock generator consists of the following hardware.

Table 7-1. Clock Generator Configuration

Item	Configuration
Control registers	Processor clock control register (PCC) Oscillation stabilization time select register (OSTS)
Oscillators	Main system clock oscillator Subsystem clock oscillator
Controllers	Prescaler Standby controller Wait controller

★ **Figure 7-1. Block Diagram of Clock Generator**



7.3 Clock Generator Control Registers

The clock generator is controlled by the following two registers.

- Processor clock control register (PCC)
- Oscillation stabilization time select register (OSTS)

(1) Processor clock control register (PCC)

This register selects the CPU clock and the division ratio, sets main system clock oscillator operation/stop and sets whether to use the subsystem clock oscillator internal feedback resistor[Note].

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

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

★ **Note** The feedback resistor is required to control the bias point of the oscillation waveform so that the bias point is in the middle of the power supply voltage.

When the subsystem clock is not used, the power consumption in the STOP mode can be reduced by setting bit 6 (FRC) of PCC to 1 (see **Figure 7-7 Subsystem Clock Feedback Resistor**).

Figure 7-2. Format of Processor Clock Control Register (PCC)

Address: FFFBH After reset: 04H R/W[Note 1]

Symbol	<7>	<6>	<5>	<4>	3	2	1	0
PCC	MCC	FRC	CLS	CSS	0	PCC2	PCC1	PCC0

MCC	Main system clock oscillation control[Note 2]
0	Oscillation possible
1	Oscillation stopped

FRC	Subsystem clock feedback resistor selection
0	Internal feedback resistor used
1	Internal feedback resistor not used[Note 3]

CLS	CPU clock status
0	Main system clock
1	Subsystem clock

CSS	PCC2	PCC1	PCC0	CPU clock (f_{CPU}) selection
0	0	0	0	f_X
	0	0	1	$f_X/2$
	0	1	0	$f_X/2^2$
	0	1	1	$f_X/2^3$
	1	0	0	$f_X/2^4$
1	0	0	0	$f_{XT}/2$
	0	0	1	
	0	1	0	
	0	1	1	
	1	0	0	
Other than above				Setting prohibited

Notes 1. Bit 5 is read-only.

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

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

Cautions 1. Be sure to clear bit 3 to 0.

2. When the external clock is input, MCC should not be set.
This is because the X2 pin is connected to V_{DD1} via a pull-up resistor.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

The fastest instructions of the μPD780024A, 780034A, 780024AY, and 780034AY Subseries are carried out in two CPU clocks. The relationship between the CPU clock (f_{CPU}) and minimum instruction execution time is 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 = 8.38 MHz	f_X = 12 MHz[Note]	f_{XT} = 32.768 kHz
f_X	0.238 μs	0.166 μs	–
$f_X/2$	0.477 μs	0.333 μs	–
$f_X/2^2$	0.954 μs	0.666 μs	–
$f_X/2^3$	1.90 μs	1.33 μs	–
$f_X/2^4$	3.81 μs	2.66 μs	–
$f_{XT}/2$	–	–	122 μs

Note Expanded-specification products of μPD780024A, 780034A Subseries only

Remark f_X: Main system clock oscillation frequency
f_{XT}: Subsystem clock oscillation frequency

(2) Oscillation stabilization time select register (OSTS)

This register is used to select the oscillation stabilization time from when reset is effected or STOP mode is released to when oscillation is stabilized.

OSTS is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets OSTS to 04H. Thus, when releasing the STOP mode by $\overline{\text{RESET}}$ input, the time required to release is $2^{17}/f_X$.

Figure 7-3. Format of Oscillation Stabilization Time Select Register (OSTS)

Address: FFFAH After reset: 04H R/W

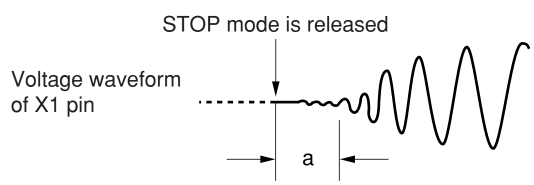
Symbol	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

★

OSTS2	OSTS1	OSTS0	Selection of oscillation stabilization time		
				f_X = 8.38 MHz	f_X = 12 MHz[Note]
0	0	0	$2^{12}/f_X$	488 μs	341 μs
0	0	1	$2^{14}/f_X$	1.95 ms	1.36 ms
0	1	0	$2^{15}/f_X$	3.91 ms	2.73 ms
0	1	1	$2^{16}/f_X$	7.82 ms	5.46 ms
1	0	0	$2^{17}/f_X$	15.6 ms	10.9 ms
Other than the above			Setting prohibited		

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

★ **Caution The wait time when STOP mode is released does not include the time ("a" in the figure below) from when STOP mode is released until the clock starts oscillation.**
This also applies when $\overline{\text{RESET}}$ is input and an interrupt request is generated.



Remark f_X: Main system clock oscillation frequency

7.4 System Clock Oscillator

7.4.1 Main system clock oscillator

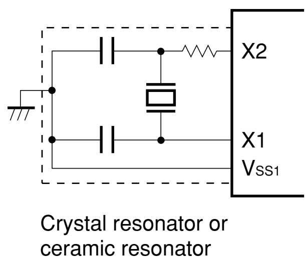
The main system clock oscillator oscillates with a crystal resonator or a ceramic resonator (8.38 MHz TYP.) connected to the X1 and X2 pins.

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

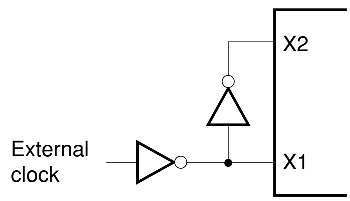
Figure 7-4 shows an external circuit of the main system clock oscillator.

Figure 7-4. External Circuit of Main System Clock Oscillator

(a) Crystal and ceramic oscillation



(b) External clock



Caution Do not execute the STOP instruction and do not set MCC (bit 7 of processor clock control register (PCC)) to 1 if an external clock is input. This is because when the STOP instruction is executed or MCC is set to 1, the main system clock operation stops and the X2 pin is connected to V_{DD1} via a pull-up resistor.

7.4.2 Subsystem clock oscillator

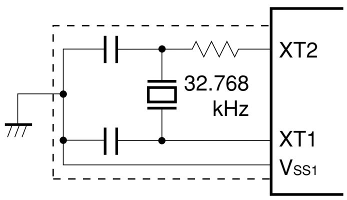
The subsystem clock oscillator oscillates with a crystal resonator (32.768 kHz TYP.) connected to the XT1 and XT2 pins.

External clocks can be input to the subsystem clock oscillator. In this case, input a clock signal to the XT1 pin and an inverted-phase clock signal to the XT2 pin.

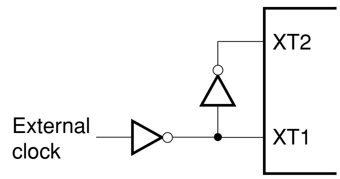
Figure 7-5 shows an external circuit of the subsystem clock oscillator.

Figure 7-5. External Circuit of Subsystem Clock Oscillator

(a) Crystal oscillation



(b) External clock



Cautions are listed on the next page.

Caution 1. When using the main system clock oscillator and subsystem clock oscillator, wire as follows in the area enclosed by broken lines in Figures 7-4 and 7-5 to avoid an adverse effect from wiring capacitance.

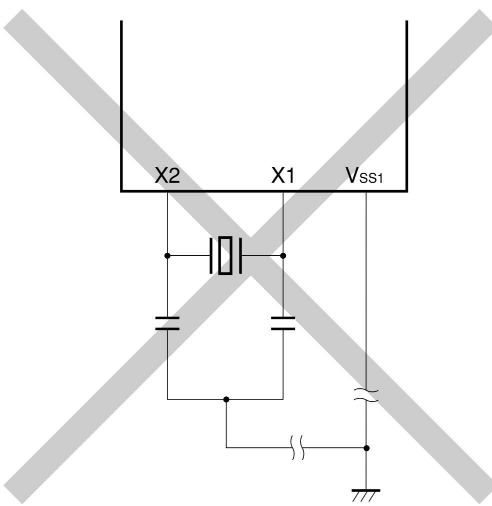
- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as V_{SS1}. Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

Note that the subsystem clock oscillator is designed as a low-amplitude circuit for reducing power consumption.

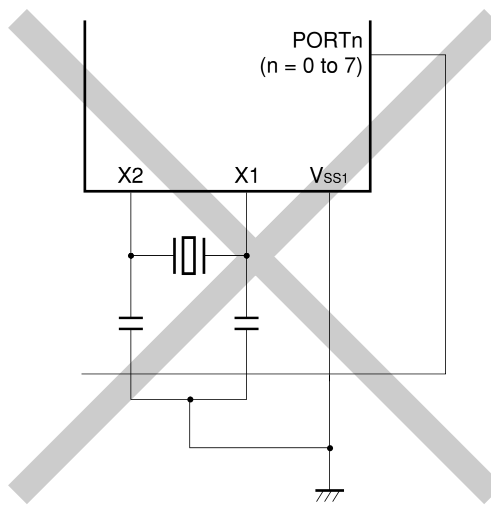
Figure 7-6 shows examples of incorrect oscillator connection.

Figure 7-6. Examples of Incorrect Oscillator Connection (1/2)

(a) Too long wiring



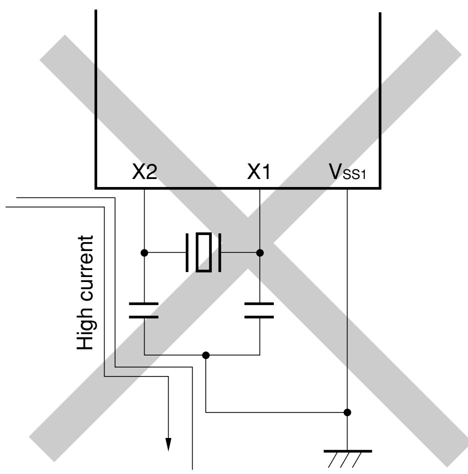
(b) Crossed signal line



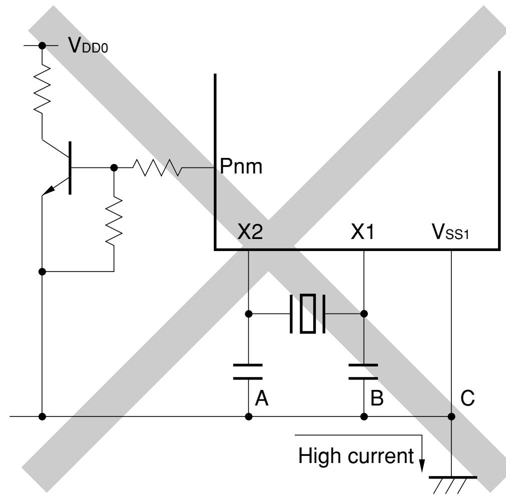
Remark When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Figure 7-6. Examples of Incorrect Oscillator Connection (2/2)

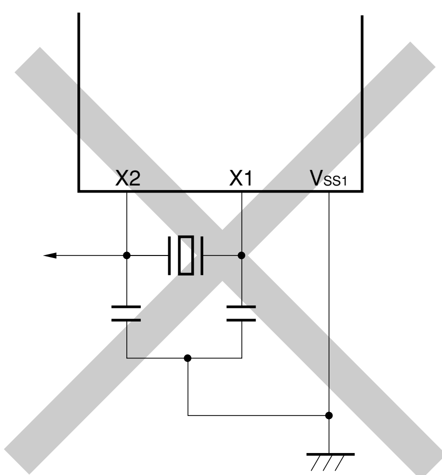
(c) Wiring near high fluctuating current



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



(e) Signals are fetched



Remark When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Caution 2. When X2 and XT1 are wired in parallel, the crosstalk noise of X2 may increase with XT1, resulting in malfunction.
To prevent that from occurring, it is recommended to wire X2 and XT1 so that they are not in parallel, and to connect the IC pin between X2 and XT1 directly to V_{SS1}.

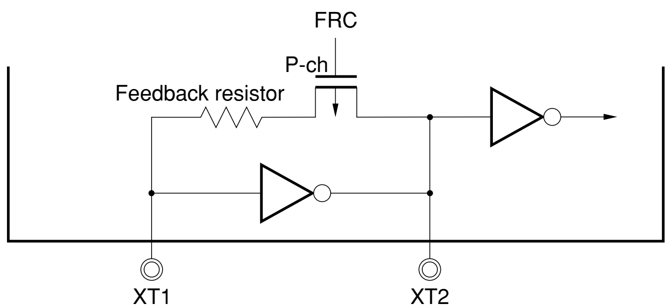
7.4.3 When subsystem clock is not used

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

★ XT1: Connect directly to V_{DD0} or V_{DD1}
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 minimize leakage current, the above internal feedback resistor can be removed by setting bit 6 (FRC) of the processor clock control register (PCC). In this case also, connect the XT1 and XT2 pins as described above.

Figure 7-7. Subsystem Clock Feedback Resistor



★ **Remark** The feedback resistor is required to control the bias point of the oscillation waveform so that the bias point is in the middle of the power supply voltage.

7.5 Clock Generator Operations

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

- Main system clock f_X
- 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).

(a) Upon generation of the $\overline{\text{RESET}}$ signal, the lowest speed mode of the main system clock (3.81 μs @ 8.38 MHz operation) is selected (PCC = 04H). Main system clock oscillation stops while a low level is applied to the $\overline{\text{RESET}}$ pin.

★ (b) With the main system clock selected, one of the five levels of minimum instruction execution time (0.166 μs, 0.333 μs, 0.666 μs, 1.33 μs, 2.66 μs: @ 12 MHz operation[Note], 0.238 μs, 0.476 μs, 0.954 μs, 1.90 μs, 3.81 μs: @ 8.38 MHz operation) can be selected by setting PCC.

(c) With the main system clock selected, two standby modes, the STOP and HALT modes, are available. To reduce power consumption in the STOP mode, the subsystem clock feedback resistor can be disconnected to stop the subsystem clock.

(d) PCC can be used to select the subsystem clock and to operate the system with low power consumption (122 μs @ 32.768 kHz operation).

(e) With the subsystem clock selected, main system clock oscillation can be stopped via PCC. The HALT mode can be used. However, the STOP mode cannot be used. (Subsystem clock oscillation cannot be stopped.)

(f) The main system clock is divided and supplied to the peripheral hardware. The subsystem clock is supplied to the watch timer and clock output functions only. Thus the watch function and the clock output function can also be continued in the standby state. However, since all other peripheral hardware operate with the main system clock, the peripheral hardware also stops if the main system clock is stopped (except external input clock operation).

Note Expanded-specification products of μPD780024A, 780034A Subseries only

7.5.1 Main system clock operations

When operating with 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 setting.

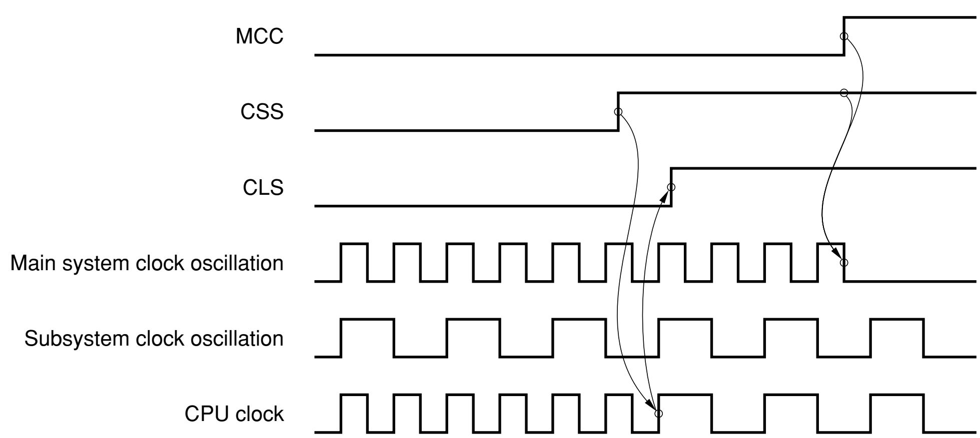
(a) Because the operation-guaranteed instruction execution speed depends on the power supply voltage, the minimum instruction execution time can be changed by bits 0 to 2 (PCC0 to PCC2) of PCC.

★ (b) When bit 4 (CSS) of PCC is set to 1 when operating with the main system clock, if bit 7 (MCC) of PCC is set to 1 after the operation has been switched to the subsystem clock (CLS = 1), the main system clock oscillation stops (see **Figure 7-8 (1)**).

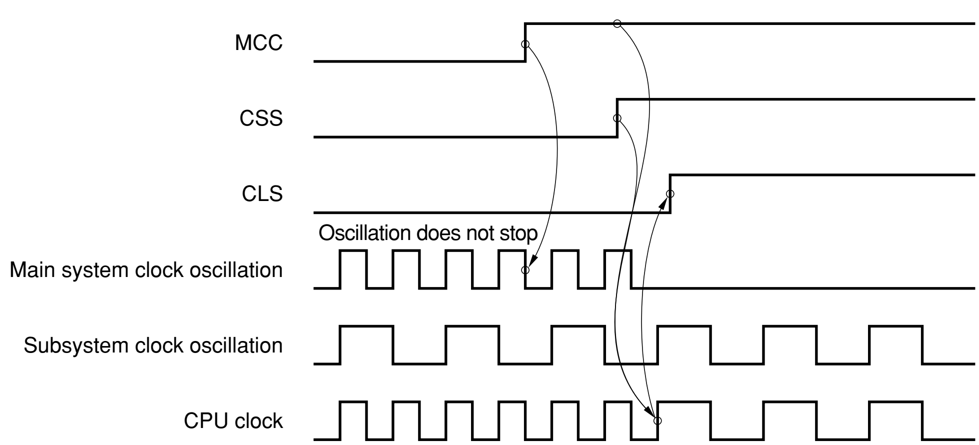
(c) If bit 7 (MCC) of PCC is set to 1 when operating with the main system clock, the main system clock oscillation does not stop. When bit 4 (CSS) of PCC is set to 1 and the operation is switched to the subsystem clock (CLS = 1) after that, the main system clock oscillation stops (see **Figure 7-8 (2)**).

Figure 7-8. Main System Clock Stop Function

(1) Operation when MCC is set after setting CSS with main system clock operation



★ **(2) Operation when CSS is set after setting MCC with main system clock operation**



7.5.2 Subsystem clock operations

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

(a) The minimum instruction execution time remains constant (122 μs @ 32.768 kHz operation) irrespective of bits 0 to 2 (PCC0 to PCC2) of PCC.

(b) Watchdog timer counting stops.

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 means of 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 by bit 5 (CLS) of the PCC register.

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

<table>
<tr><td colspan="4">Set Value Before Switchover</td><td colspan="24">Set Value After Switchover</td></tr>
<tr><td rowspan="2">CSS</td><td rowspan="2">PCC2</td><td rowspan="2">PCC1</td><td rowspan="2">PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>×</td><td>×</td><td>×</td></tr>
<tr><td rowspan="5">0</td><td>0</td><td>0</td><td>0</td><td colspan="4"></td><td colspan="4">16 instructions</td><td colspan="4">16 instructions</td><td colspan="4">16 instructions</td><td colspan="4">16 instructions</td><td colspan="4">$f_X/2f_{XT}$ instruction</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td colspan="4">8 instructions</td><td colspan="4"></td><td colspan="4">8 instructions</td><td colspan="4">8 instructions</td><td colspan="4">8 instructions</td><td colspan="4">$f_X/4f_{XT}$ instruction</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td colspan="4">4 instructions</td><td colspan="4">4 instructions</td><td colspan="4"></td><td colspan="4">4 instructions</td><td colspan="4">4 instructions</td><td colspan="4">$f_X/8f_{XT}$ instruction</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td colspan="4">2 instructions</td><td colspan="4">2 instructions</td><td colspan="4">2 instructions</td><td colspan="4"></td><td colspan="4">2 instructions</td><td colspan="4">$f_X/16f_{XT}$ instruction</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4"></td><td colspan="4">$f_X/32f_{XT}$ instruction</td></tr>
<tr><td>1</td><td>×</td><td>×</td><td>×</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4">1 instruction</td><td colspan="4"></td></tr>
</table>

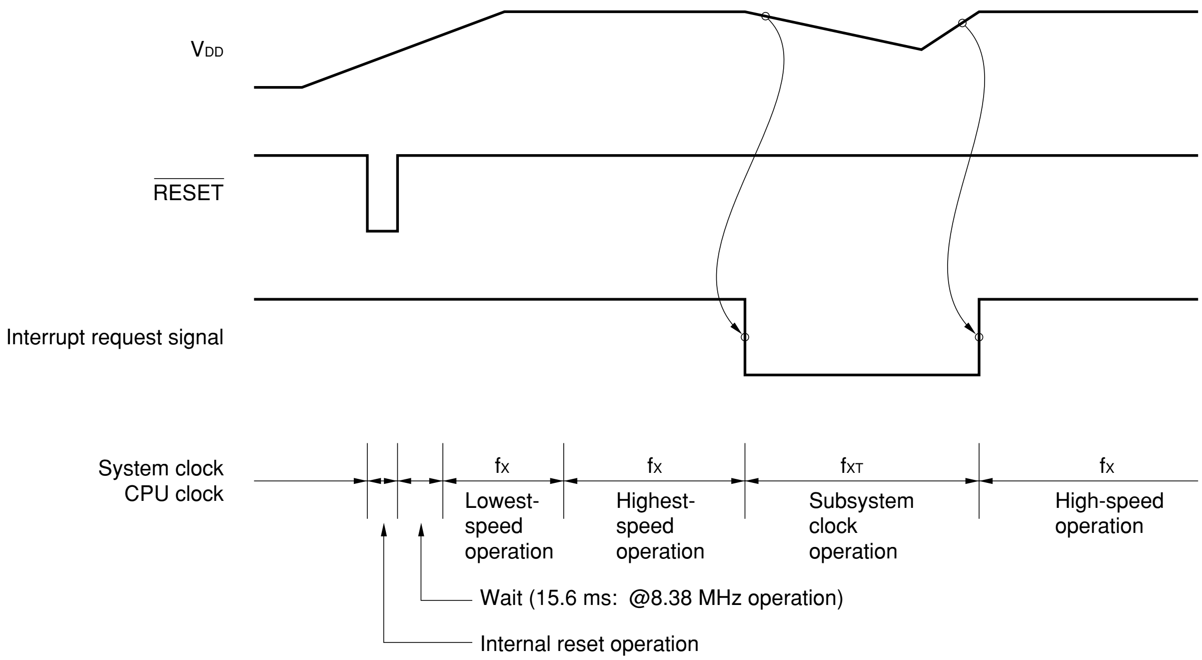
Remark One instruction is the minimum instruction execution time with the pre-switchover CPU clock.

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 set simultaneously. Simultaneous setting is possible, however, for selection of the CPU clock cycle division ratio (PCC0 to PCC2) and switch over 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 procedure for switching between the system clock and CPU clock.

Figure 7-9. System Clock and CPU Clock Switching



<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 instructions at the minimum speed of the main system clock (3.81 μs @ 8.38 MHz operation).

<2> After the lapse of sufficient time for the V_{DD} voltage to increase to enable operation at maximum speeds, PCC is rewritten and 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, 0 is set to the MCC and oscillation of the main system clock is started. After the lapse of the time required for stabilization of oscillation, PCC is 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 0

8.1 Functions of 16-Bit Timer/Event Counter 0

16-bit timer/event counter 0 has the following functions.

(1) Interval timer

16-bit timer/event counter 0 generates interrupt requests at the preset time interval.

- Number of counts: 2 to 65536

(2) External event counter

16-bit timer/event counter 0 can measure the number of pulses with a high-/low-level width of a signal input externally.

- Valid level pulse width: 16/fx or more

(3) Pulse width measurement

16-bit timer/event counter 0 can measure the pulse width of an externally input signal.

- Valid level pulse width: 2/fx or more

(4) Square-wave output

16-bit timer/event counter 0 can output a square wave with any selected frequency.

- Cycle: $(2 \times 2$ to $65536 \times 2) \times$ count clock cycle

(5) PPG output

16-bit timer/event counter 0 can output a square wave that have arbitrary cycle and pulse width.

- $2 <$ Pulse width $<$ Cycle $\leq$ (FFFF + 1) H

8.2 Configuration of 16-Bit Timer/Event Counter 0

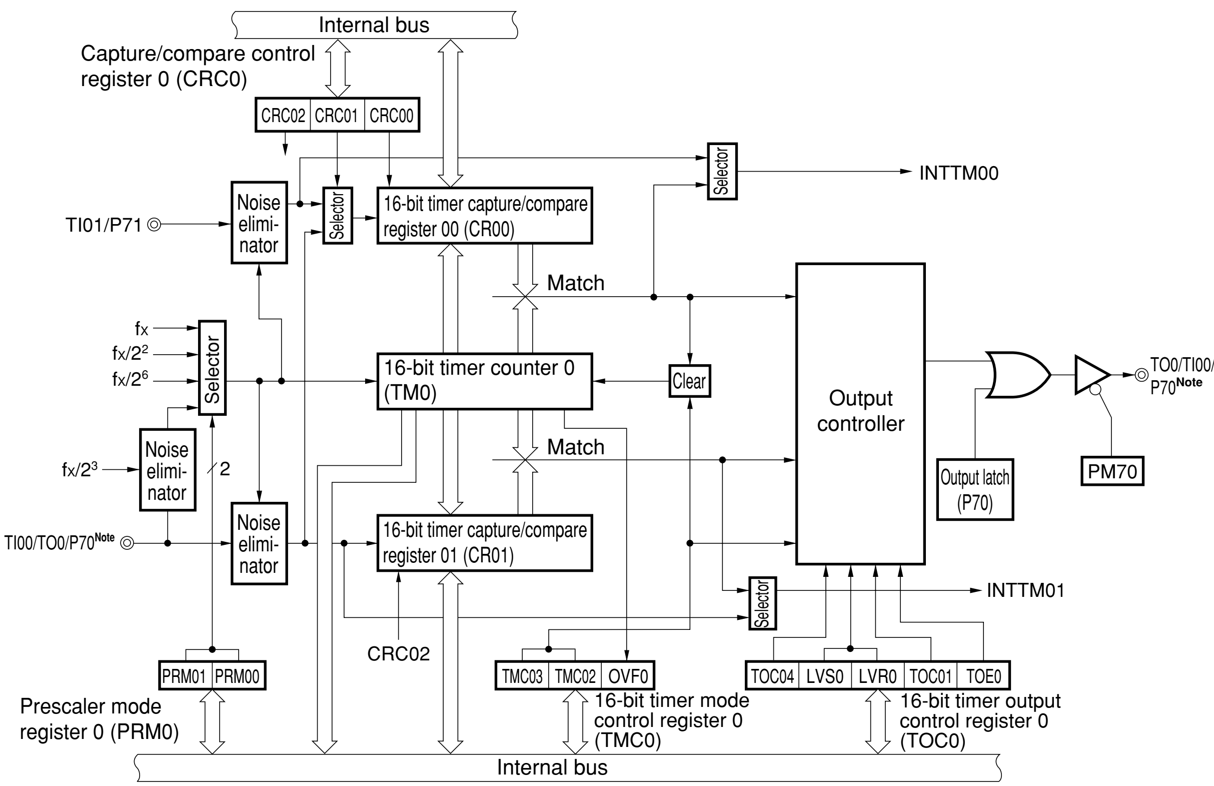
16-bit timer/event counter 0 consists of the following hardware.

Table 8-1. Configuration of 16-Bit Timer/Event Counter 0

Item	Configuration
Timer counter	16-bit timer counter 0 (TM0)
Register	16-bit timer capture/compare registers: 00, 01 (CR00, CR01)
Timer input	TI00, TI01
Timer output	TO0
Control registers	16-bit timer mode control register 0 (TMC0) Capture/compare control register 0 (CRC0) 16-bit timer output control register 0 (TOC0) Prescaler mode register 0 (PRM0) Port mode register 7 (PM7) Port 7 (P7)

Figure 8-1 shows block diagram of this counter.

★ **Figure 8-1. Block Diagram of 16-Bit Timer/Event Counter 0**



Note TI00 input and TO0 output cannot be used at the same time.

(1) 16-bit timer counter 0 (TM0)

TM0 is a 16-bit read-only register that counts count pulses.

The counter is incremented in synchronization with the rising edge of the count clock. If the count value is read during operation, input of the count clock is temporarily stopped, and the count value at that point is read. The count value is reset to 0000H in the following cases.

<1> At $\overline{\text{RESET}}$ input
<2> If TMC03 and TMC02 are cleared
<3> If the valid edge of TI00 is input in the clear & start mode entered by inputting the valid edge of TI00
<4> If TM0 and CR00 match in the clear & start mode entered on a match between TM0 and CR00

(2) 16-bit timer 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 (CRC0).

- **When CR00 is used as a compare register**

  The value set in CR00 is constantly compared with the 16-bit timer counter 0 (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 then TM0 is set to interval timer operation.

- **When CR00 is used as a capture register**

  It is possible to select the valid edge of the TI00 pin or the TI01 pin as the capture trigger. Setting of the TI00 or TI01 valid edge is performed by means of prescaler mode register 0 (PRM0) (see **Table 8-2**).

★ **Table 8-2. CR00 Capture Trigger and Valid Edges of TI00 and TI01 Pins**

(1) TI00 pin valid edge selected as capture trigger (CRC01 = 1, CRC00 = 1)

CR00 Capture Trigger	TI00 Pin Valid Edge		
		ES01	ES00
Falling edge	Rising edge	0	1
Rising edge	Falling edge	0	0
No capture operation	Both rising and falling edges	1	1

(2) TI01 pin valid edge selected as capture trigger (CRC01 = 0, CRC00 = 1)

CR00 Capture Trigger	TI01 Pin Valid Edge		
		ES11	ES10
Falling edge	Falling edge	0	0
Rising edge	Rising edge	0	1
Both rising and falling edges	Both rising and falling edges	1	1

Remarks 1. Setting ES01, ES00 = 1, 0 and ES11, ES10 = 1, 0 is prohibited.

2. ES01, ES00: Bits 5 and 4 of prescaler mode register 0 (PRM0)
ES11, ES10: Bits 7 and 6 of prescaler mode register 0 (PRM0)
CRC01, CRC00: Bits 1 and 0 of capture/compare control register 0 (CRC0)

CR00 is set by a 16-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input makes CR00 undefined.

Cautions 1. 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 cleared to 0000H, an interrupt request (INTTM00) is generated when CR00 changes from 0000H to 0001H following overflow (FFFFH).

2. If the new value of CR00 is less than the value of 16-bit timer counter 0 (TM0), TM0 continues counting, overflows, and then starts counting from 0 again. If the new value of CR00 is less than the old value, therefore, the timer must be reset to be restarted after the value of CR00 is changed.

3. When P70 is used as the input pin for the valid edge of TI00, it cannot be used as a timer output (TO0). Moreover, when P70 is used as TO0, it cannot be used as the input pin for the valid edge of TI00.

(3) 16-bit timer 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 (CRC0).

- **When CR01 is used as a compare register**

  The value set in CR01 is constantly compared with the 16-bit timer counter 0 (TM0) count value, and an interrupt request (INTTM01) is generated if they match.

- **When CR01 is used as a capture register**

  It is possible to select the valid edge of the TI00 pin as the capture trigger. The TI00 valid edge is set by means of prescaler mode register 0 (PRM0) (see **Table 8-3**).

★ **Table 8-3. CR01 Capture Trigger and Valid Edge of TI00 Pin (CRC02 = 1)**

CR01 Capture Trigger	TI00 Pin Valid Edge		
		ES01	ES00
Falling edge	Falling edge	0	0
Rising edge	Rising edge	0	1
Both rising and falling edges	Both rising and falling edges	1	1

Remarks 1. Setting ES01, ES00 = 1, 0 is prohibited.

2. ES01, ES00: Bits 5 and 4 of prescaler mode register 0 (PRM0)
CRC02: Bit 2 of capture/compare control register 0 (CRC0)

CR01 is set by a 16-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input makes CR01 undefined.

Caution Set CR01 to 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 cleared to 0000H, an interrupt request (INTTM01) is generated when CR01 changes from 0000H to 0001H following overflow (FFFFH).

8.3 Registers to Control 16-Bit Timer/Event Counter 0

The following six types of registers are used to control 16-bit timer/event counter 0.

- 16-bit timer mode control register 0 (TMC0)
- Capture/compare control register 0 (CRC0)
- 16-bit timer output control register 0 (TOC0)
- Prescaler mode register 0 (PRM0)
- Port mode register 7 (PM7)
- Port 7 (P7)

(1) 16-bit timer mode control register 0 (TMC0)

This register sets the 16-bit timer operating mode, the 16-bit timer counter 0 (TM0) clear mode, and output timing, and detects an overflow.
TMC0 is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TMC0 to 00H.

Caution 16-bit timer counter 0 (TM0) starts operation at the moment TMC02 and TMC03 (operation stop mode) are set to a value other than 0, 0, respectively. Clear TMC02 and TMC03 to 0, 0 to stop the operation.

★ **Figure 8-2. Format of 16-Bit Timer Mode Control Register 0 (TMC0)**

Address: FF60H After reset: 00H R/W

Symbol	7	6	5	4	<3>	<2>	1	<0>
TMC0	0	0	0	0	TMC03	TMC02	0	OVF0

TMC03	TMC02	Operating mode and clear mode selection	TO0 output timing selection	Interrupt request generation
0	0	Operation stop (TM0 cleared to 0)	No change	Not generated
0	1	Free-running mode	Match between TM0 and CR00 or match between TM0 and CR01	Generated on match between TM0 and CR00, or match between TM0 and CR01
1	0	Clear & start on TI00 valid edge[Note 1]	–	
1	1	Clear & start on match between TM0 and CR00[Note 2]	Match between TM0 and CR00 or match between TM0 and CR01	

OVF0	Overflow detection of 16-bit timer counter 0 (TM0)
0	Overflow not detected
1	Overflow detected

Notes 1. Set the valid edge of the TI00/TO0/P70 pin with prescaler mode register 0 (PRM0).

2. If the 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.

Cautions 1. To write different data to TMC0, stop the timer operation before writing.

2. The timer operation must be stopped before writing to bits other than the OVF0 flag.

Remarks 1. TO0: 16-bit timer/event counter 0 output pin

2. TI00: 16-bit timer/event counter 0 input pin

3. TM0: 16-bit timer counter 0

4. CR00: 16-bit timer capture/compare register 00

5. CR01: 16-bit timer capture/compare register 01

(2) Capture/compare control register 0 (CRC0)

This register controls the operation of the 16-bit timer capture/compare registers (CR00, CR01).

CRC0 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CRC0 to 00H.

Figure 8-3. Format of Capture/Compare Control Register 0 (CRC0)

Address: FF62H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
CRC0	0	0	0	0	0	CRC02	CRC01	CRC00

CRC02	CR01 operating mode selection
0	Operate as compare register
1	Operate as capture register

CRC01	CR00 capture trigger selection
0	Capture on valid edge of TI01
1	Capture on valid edge of TI00 by reverse phase[Note]

CRC00	CR00 operating mode selection
0	Operate as compare register
1	Operate as capture register

Note If both the rising and falling edges have been selected as the valid edges of TI00, capture is not performed.

Cautions 1. The timer operation must be stopped before setting CRC0.

2. When the clear & start mode entered on a match between TM0 and CR00 is selected by 16-bit timer mode control register 0 (TMC0), CR00 should not be specified as a capture register.

3. To ensure the reliability of the capture operation, the capture trigger requires a pulse longer than two cycles of the count clock selected by prescaler mode register 0 (PRM0) (see Figure 8-31).

(3) 16-bit timer output control register 0 (TOC0)

This register controls the operation of the 16-bit timer/event counter output controller. It sets R-S type flip-flop (LV0) set/reset, output inversion enable/disable, and 16-bit timer/event counter timer output enable/disable.

TOC0 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TOC0 to 00H.

★ **Figure 8-4. Format of 16-Bit Timer Output Control Register 0 (TOC0)**

Address: FF63H After reset: 00H R/W

Symbol	7	6	5	4	<3>	<2>	1	<0>
TOC0	0	0	0	TOC04	LVS0	LVR0	TOC01	TOE0

TOC04	Timer output F/F control by match of CR01 and TM0
0	Inversion operation disabled
1	Inversion operation enabled

LVS0	LVR0	16-bit timer/event counter 0 timer output F/F status setting
0	0	No change
0	1	Timer output F/F reset (0)
1	0	Timer output F/F set (1)
1	1	Setting prohibited

TOC01	Timer output F/F control by match of CR00 and TM0
0	Inversion operation disabled
1	Inversion operation enabled

TOE0	16-bit timer/event counter 0 output control
0	Output disabled (output set to level 0)
1	Output enabled

Cautions 1. The timer operation must be stopped before setting TOC0.

2. If LVS0 and LVR0 are read after data is set, they will be 0.

3. Be sure to clear bits 5 to 7 of TOC0 to 0.

(4) Prescaler mode register 0 (PRM0)

This register is used to set the 16-bit timer counter 0 (TM0) count clock and TI00, TI01 input valid edges.
PRM0 is set by an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears PRM0 to 00H.

Figure 8-5. Format of Prescaler Mode Register 0 (PRM0)

Address: FF61H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PRM0	ES11	ES10	ES01	ES00	0	0	PRM01	PRM00

ES11	ES10	TI01 valid edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES01	ES00	TI00 valid edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

★

PRM01	PRM00	Count clock selection		
			fx = 8.38 MHz	fx = 12 MHz Note 1
0	0	fx	8.38 MHz	12 MHz
0	1	$f_X/2^2$	2.09 MHz	3 MHz
1	0	$f_X/2^6$	130 kHz	187 kHz
1	1	TI00 valid edge Notes 2, 3		

Notes 1. Expanded-specification products of μPD780024A, 780034A Subseries only.

2. The external clock requires a pulse longer than two cycles of the internal count clock ($f_X/2^3$).

3. When the valid edge of TI00 is selected, the main system clock is used as the sampling clock for noise elimination. The valid edge of TI00 can be used only when the main system clock is operating.

Cautions 1. Always set data to PRM0 after stopping the timer operation.

2. If the valid edge of TI00 is to be set as the count clock, do not set the clear & start mode and the capture trigger at the valid edge of TI00.
Moreover, do not use the P70/TI00/TO0 pin as a timer output (TO0).

3. If the TI00 or TI01 pin is high level immediately after system reset, the rising edge is immediately detected after the rising edge or both the rising and falling edges are set as the valid edge(s) of the TI00 pin or TI01 pin to enable the operation of 16-bit timer counter 0 (TM0). Be careful when pulling up the TI00 pin or the TI01 pin. However, when re-enabling operation after the operation has been stopped once, the rising edge is not detected.

Remarks 1. fx: Main system clock oscillation frequency

2. TI00, TI01: 16-bit timer/event counter 0 input pin

(5) Port mode register 7 (PM7)

This register sets port 7 input/output in 1-bit units.

When using the P70/TO0/TI00 pin for timer output, clear PM70 and the output latch of P70 to 0.

★ When using the P70/TO0/TI00 pin for timer input, set PM70 to 1. At this time, the output latch of P70 can be either 0 or 1.

PM7 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM7 to FFH.

Figure 8-6. Format of Port Mode Register 7 (PM7)

Address: FF27H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM7	1	1	PM75	PM74	PM73	PM72	PM71	PM70

PM7n	P7n pin I/O mode selection (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

8.4 Operation of 16-Bit Timer/Event Counter 0

8.4.1 Interval timer operation

Setting 16-bit timer mode control register 0 (TMC0) and capture/compare control register 0 (CRC0) as shown in Figure 8-7 allows operation as an interval timer. Interrupt requests are generated repeatedly using the count value set in 16-bit timer capture/compare register 00 (CR00) beforehand as the interval.

When the count value of 16-bit timer counter 0 (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 0 and 1 (PRM00, PRM01) of prescaler mode register 0 (PRM0).

★ **Figure 8-7. Control Register Settings for Interval Timer Operation**

(a) 16-bit timer mode control register 0 (TMC0)

					TMC03	TMC02		OVF0
TMC0	0	0	0	0	1	1	0	0

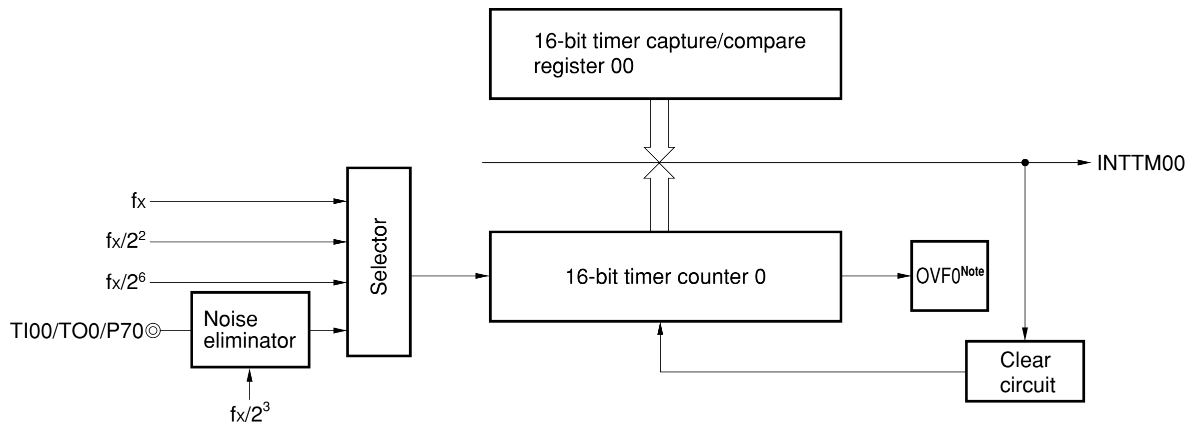
TMC03, TMC02 = 1, 1: Clears and starts on match between TM0 and CR00.

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

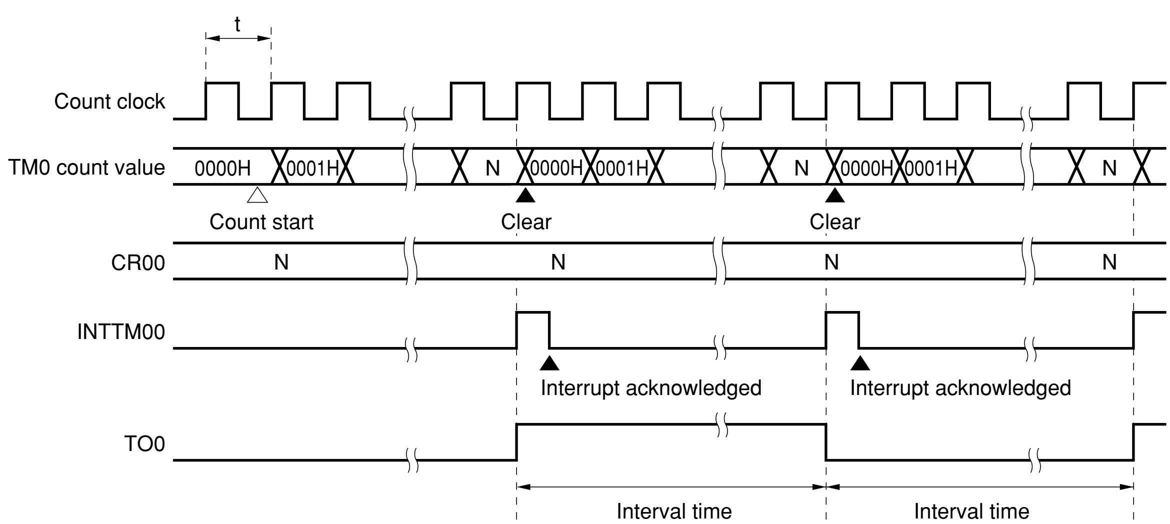
CRC00 = 0: CR00 used as compare register

Figure 8-8. Interval Timer Configuration Diagram



★ **Note** OVF0 is 1 only when 16-bit timer capture/compare register 00 is set to FFFFH.

Figure 8-9. Timing of Interval Timer Operation

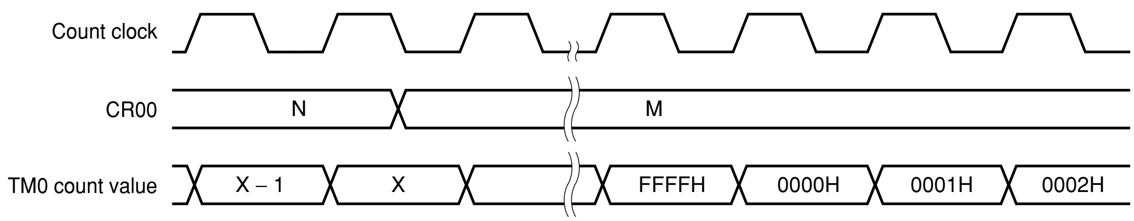


Remark Interval time = (N + 1) × t

★ N = 0001H to FFFFH

When the compare register is changed during timer count operation, if the value after 16-bit timer capture/compare register 00 (CR00) is changed is smaller than that of 16-bit timer counter 0 (TM0), TM0 continues counting, overflows and then restarts counting from 0. Thus, if the value (M) after the CR00 change is smaller than that (N) before the change, it is necessary to restart the timer after changing CR00.

Figure 8-10. Timing After Change of Compare Register During Timer Count Operation



Remark N > X > M

8.4.2 External event counter operation

The external event counter counts the number of external clock pulses to be input to the TI00 pin with using 16-bit timer counter 0 (TM0).

TM0 is incremented each time the valid edge specified by prescaler mode register 0 (PRM0) is input.

When the TM0 count value matches the 16-bit timer capture/compare register 00 (CR00) value, TM0 is cleared to 0 and the interrupt request signal (INTTM00) is generated.

Input a value other than 0000H to CR00. (A count operation with a pulse cannot be carried out.)

The rising edge, the falling edge, or both edges can be selected using bits 4 and 5 (ES00 and ES01) of prescaler mode register 0 (PRM0).

Because an operation is carried out only when the valid edge of the TI00 pin is detected twice after sampling with the internal clock ($f_X/2^3$), noise with a short pulse width can be removed.

Caution When used as an external event counter, the P70/TI00/TO0 pin cannot be used as a timer output (TO0).

★ **Figure 8-11. Control Register Settings in External Event Counter Mode**

(a) 16-bit timer mode control register 0 (TMC0)

					TMC03	TMC02		OVF0
TMC0	0	0	0	0	1	1	0	0

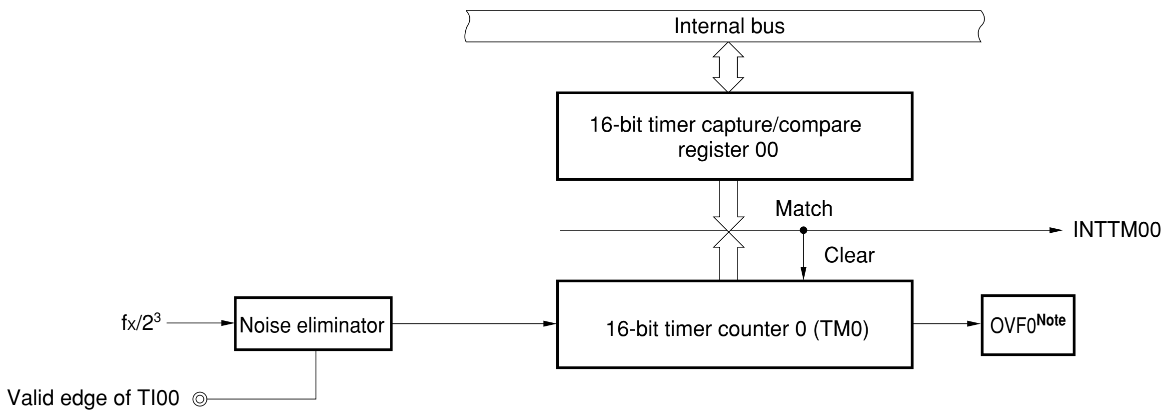
TMC03, TMC02: Clears and starts on match between TM0 and CR00.

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

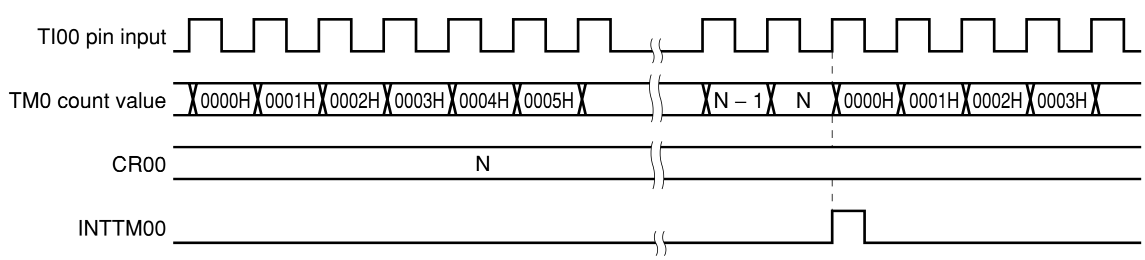
CRC00: CR00 used as compare register

★ **Figure 8-12. External Event Counter Configuration Diagram**



Note OVF0 is 1 only when 16-bit timer capture/compare register 00 is set to FFFFH.

Figure 8-13. External Event Counter Operation Timing (with Rising Edge Specified)



Caution When reading the external event counter count value, TM0 should be read.

8.4.3 Pulse width measurement operations

It is possible to measure the pulse width of the signals input to the TI00 pin and TI01 pin using 16-bit timer counter 0 (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 pin.

(1) Pulse width measurement with free-running counter and one capture register

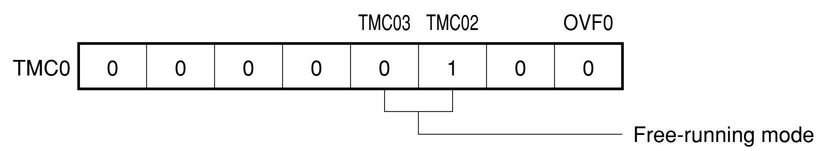
When 16-bit timer counter 0 (TM0) is operated in free-running mode (see register settings in **Figure 8-14**), and the edge specified by prescaler mode register 0 (PRM0) is input to the TI00 pin, the value of TM0 is taken into 16-bit timer capture/compare register 01 (CR01) and an external interrupt request signal (INTTM01) is set.

Any of three edges can be selected—rising, falling, or both edges—specified by bits 4 and 5 (ES00 and ES01) of PRM0.

Sampling is performed with the count clock selected by PRM0, and a capture operation is only performed when a valid level of the TI00 pin is detected twice, thus eliminating noise with a short pulse width.

★ **Figure 8-14. Control Register Settings for Pulse Width Measurement with Free-Running Counter and One Capture Register**

(a) 16-bit timer mode control register 0 (TMC0)



(b) Capture/compare control register 0 (CRC0)

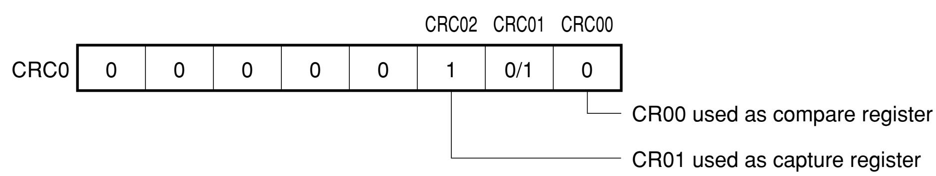
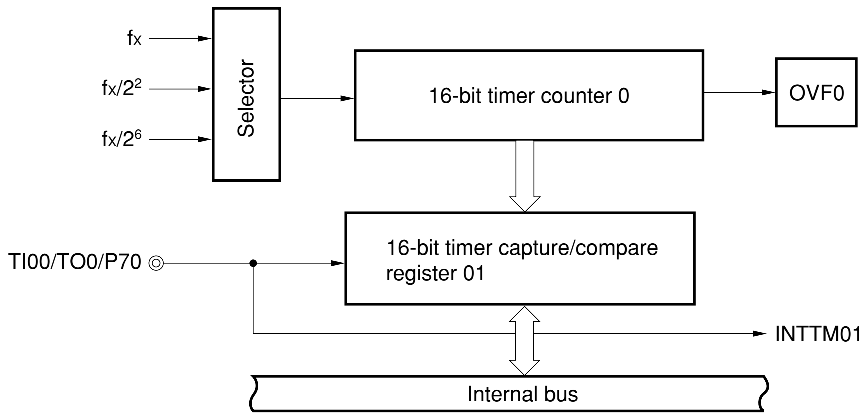
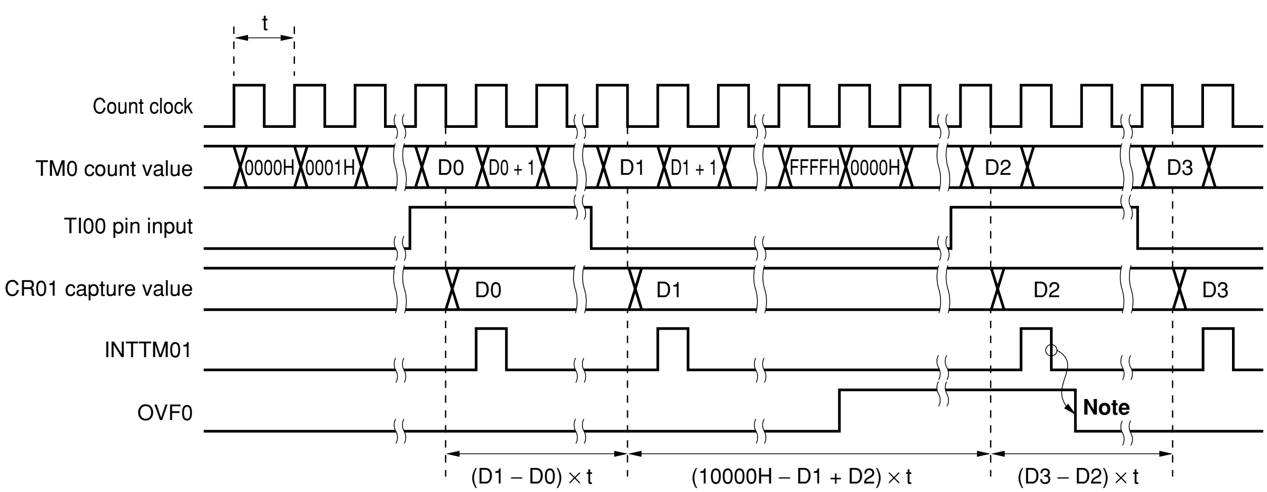


Figure 8-15. Configuration Diagram for Pulse Width Measurement with Free-Running Counter



★ **Figure 8-16. Timing of Pulse Width Measurement Operation with Free-Running Counter and One Capture Register (with Both Edges Specified)**



Note OVF0 must be cleared by software.

(2) Measurement of two pulse widths with free-running counter

When 16-bit timer counter 0 (TM0) is operated in free-running mode (see register settings in **Figure 8-17**), it is possible to simultaneously measure the pulse widths of the two signals input to the TI00 pin and the TI01 pin.

When the edge specified by bits 4 and 5 (ES00 and ES01) of prescaler mode register 0 (PRM0) is input to the TI00 pin, the value of TM0 is taken into 16-bit timer capture/compare register 01 (CR01) and an interrupt request signal (INTTM01) is set.

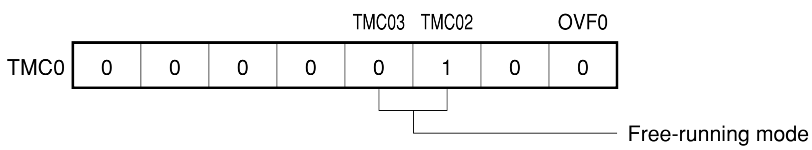
Also, when the edge specified by bits 6 and 7 (ES10 and ES11) of PRM0 is input to the TI01 pin, the value of TM0 is taken into 16-bit timer capture/compare register 00 (CR00) and an interrupt request signal (INTTM00) is set.

Any of three edges can be selected—rising, falling, or both edges—as the valid edges for the TI00 pin and the TI01 pin specified by bits 4 and 5 (ES00 and ES01) and bits 6 and 7 (ES10 and ES11) of PRM0, respectively.

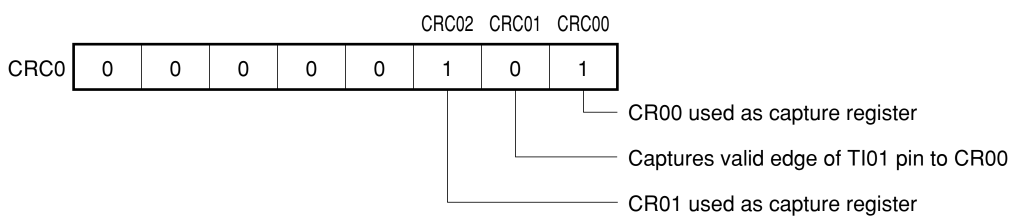
Sampling is performed at the interval selected by prescaler mode register 0 (PRM0), and a capture operation is only performed when a valid level of the TI00 pin or TI01 pin is detected twice, thus eliminating noise with a short pulse width.

★ **Figure 8-17. Control Register Settings for Measurement of Two Pulse Widths with Free-Running Counter**

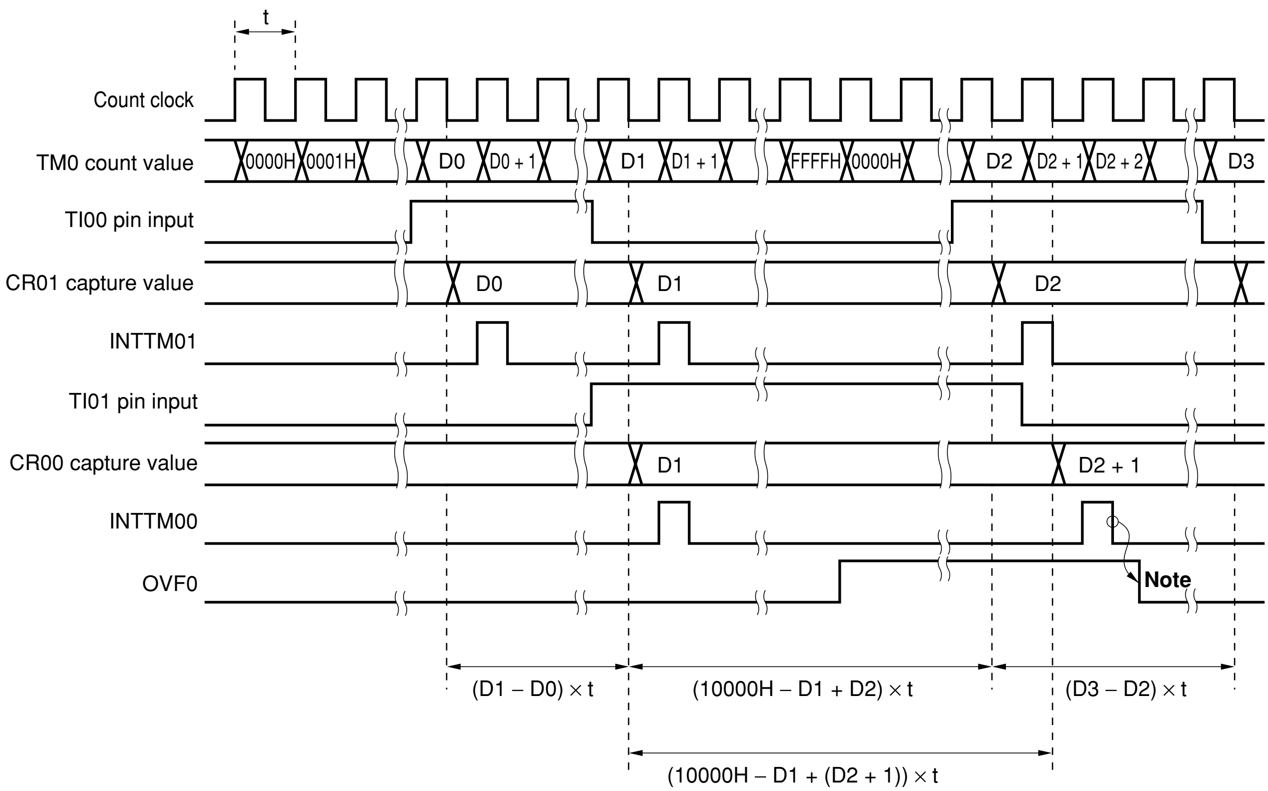
(a) 16-bit timer mode control register 0 (TMC0)



(b) Capture/compare control register 0 (CRC0)



★ **Figure 8-18. Timing of Pulse Width Measurement Operation with Free-Running Counter (with Both Edges Specified)**



Note OVF0 must be cleared by software.

(3) Pulse width measurement with free-running counter and two capture registers

When 16-bit timer counter 0 (TM0) is operated in free-running mode (see register settings in **Figure 8-19**), it is possible to measure the pulse width of the signal input to the TI00 pin.

When the edge specified by bits 4 and 5 (ES00 and ES01) of prescaler mode register 0 (PRM0) is input to the TI00 pin, the value of TM0 is taken into 16-bit timer capture/compare register 01 (CR01) and an interrupt request signal (INTTM01) is set.

Also, when the inverse edge to that of the capture operation to CR01 is input, the value of TM0 is taken into 16-bit timer capture/compare register 00 (CR00).

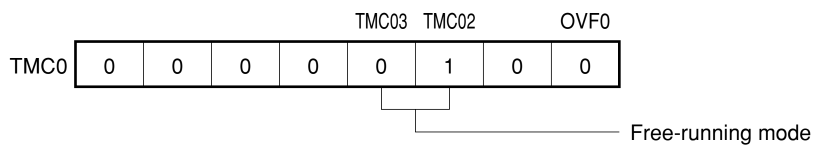
Either of two edges can be selected—rising or falling—as the valid edges for the TI00 pin specified by bits 4 and 5 (ES00 and ES01) of prescaler mode register 0 (PRM0).

Sampling is performed at the interval selected by prescaler mode register 0 (PRM0), and a capture operation is only performed when a valid level of the TI00 pin is detected twice, thus eliminating noise with a short pulse width.

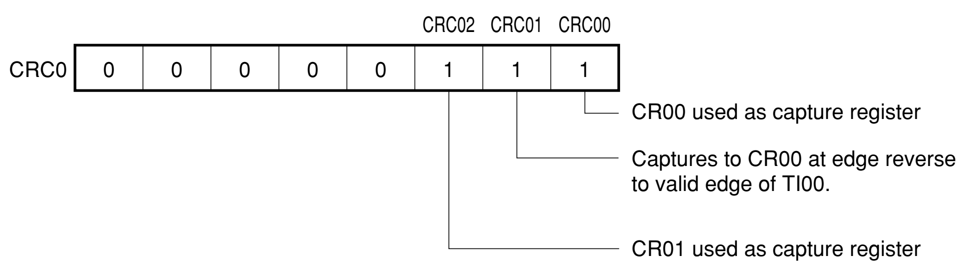
Caution If the valid edge of TI00 is specified to be both the rising and falling edges, 16-bit timer capture/compare register 00 (CR00) cannot perform the capture operation.

★ **Figure 8-19. Control Register Settings for Pulse Width Measurement with Free-Running Counter and Two Capture Registers**

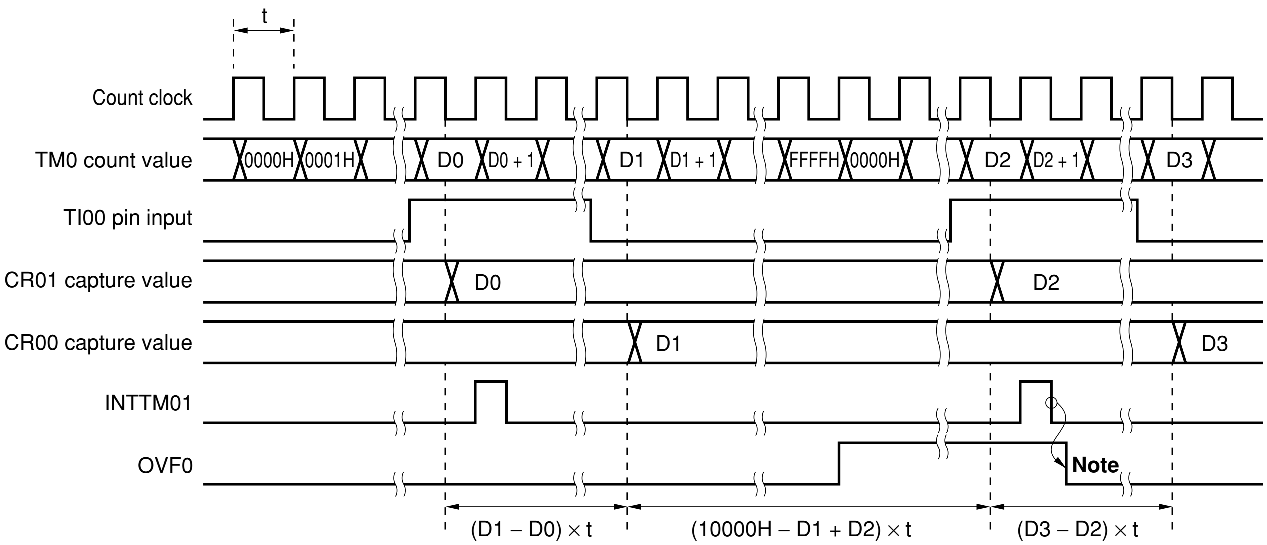
(a) 16-bit timer mode control register 0 (TMC0)



(b) Capture/compare control register 0 (CRC0)



★ **Figure 8-20. Timing of Pulse Width Measurement Operation with Free-Running Counter and Two Capture Registers (with Rising Edge Specified)**



Note OVF0 must be cleared by software.

(4) Pulse width measurement by means of restart

When input of a valid edge to the TI00 pin is detected, the count value of 16-bit timer counter 0 (TM0) is taken into 16-bit timer capture/compare register 01 (CR01), and then the pulse width of the signal input to the TI00 pin is measured by clearing TM0 and restarting the count (see register settings in **Figure 8-22**).

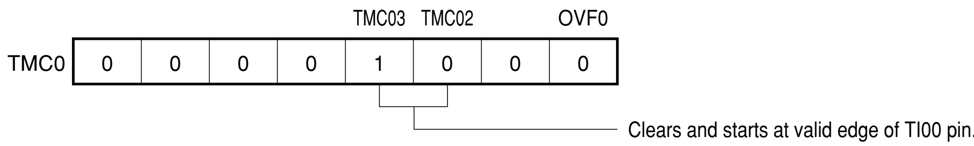
The edge specification can be selected from two types, rising or falling edges, by bits 4 and 5 (ES00 and ES01) of prescaler mode register 0 (PRM0).

Sampling is performed at the interval selected by prescaler mode register 0 (PRM0) and a capture operation is only performed when a valid level of the TI00 pin is detected twice, thus eliminating noise with a short pulse width.

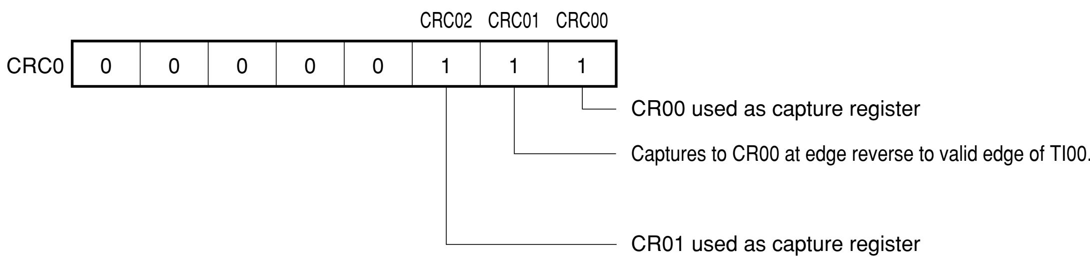
Caution If the valid edge of TI00 is specified to be both the rising and falling edges, 16-bit timer capture/compare register 00 (CR00) cannot perform the capture operation.

★ **Figure 8-21. Control Register Settings for Pulse Width Measurement by Means of Restart**

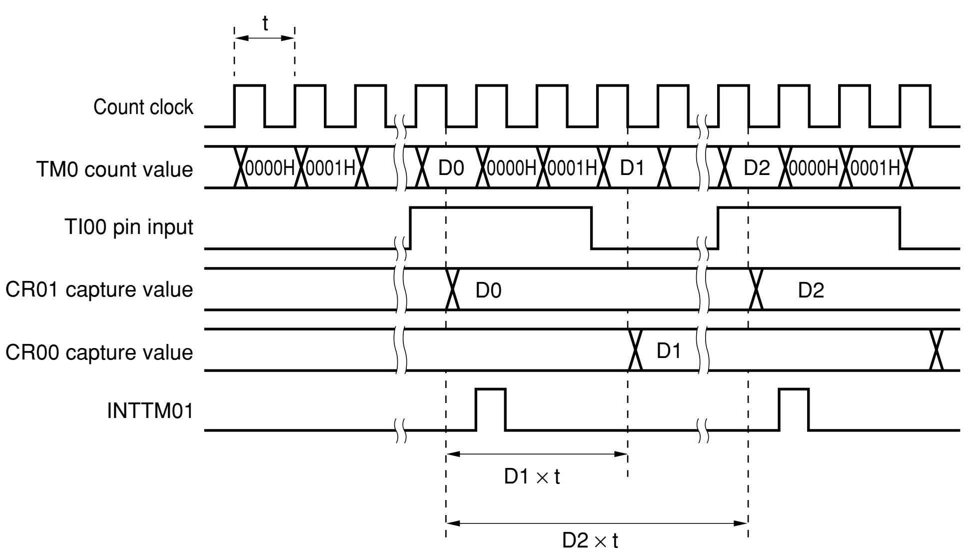
(a) 16-bit timer mode control register 0 (TMC0)



(b) Capture/compare control register 0 (CRC0)



★ **Figure 8-22. Timing of Pulse Width Measurement Operation by Means of Restart (with Rising Edge Specified)**



8.4.4 Square-wave output operation

A square wave with any selected frequency can be output at intervals determined by the count value preset to 16-bit timer capture/compare register 00 (CR00).

The TO0 pin output status is reversed at intervals determined by the count value preset to CR00 by setting bit 0 (TOE0) and bit 1 (TOC01) of 16-bit timer output control register 0 (TOC0) to 1. This enables a square wave with any selected frequency to be output.

★ **Figure 8-23. Control Register Settings in Square-Wave Output Mode**

(a) 16-bit timer mode control register 0 (TMC0)

					TMC03	TMC02		OVF0
TMC0	0	0	0	0	1	1	0	0

TMC03, TMC02: Clears and starts on match between TM0 and CR00.

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

CRC00: CR00 used as compare register

(c) 16-bit timer output control register 0 (TOC0)

				TOC04	LVS0	LVR0	TOC01	TOE0
TOC0	0	0	0	0	0/1	0/1	1	1

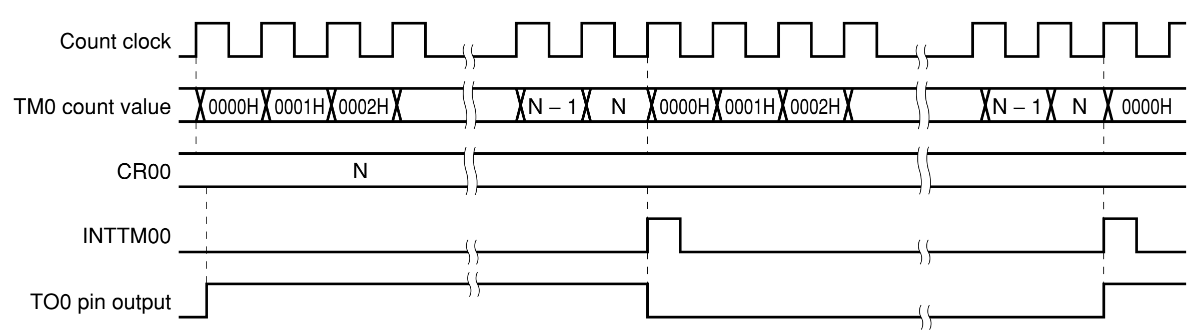
TOE0: Enables TO0 output.

TOC01: Reverses output on match between TM0 and CR00.

LVS0, LVR0: Specifies initial value of TO0 output F/F.

TOC04: Does not reverse output on match between TM0 and CR01.

Figure 8-24. Square-Wave Output Operation Timing



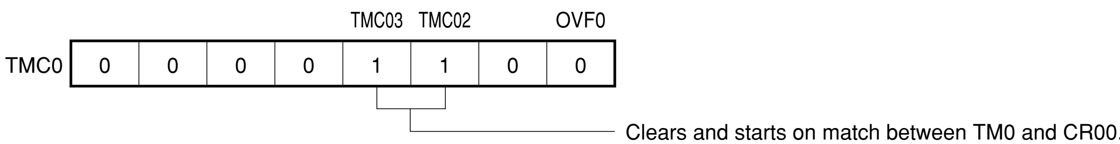
8.4.5 PPG output operation

Setting 16-bit timer mode control register 0 (TMC0) and capture/compare control register 0 (CRC0) as shown in Figure 8-25 allows operation as PPG (Programmable Pulse Generator) output.

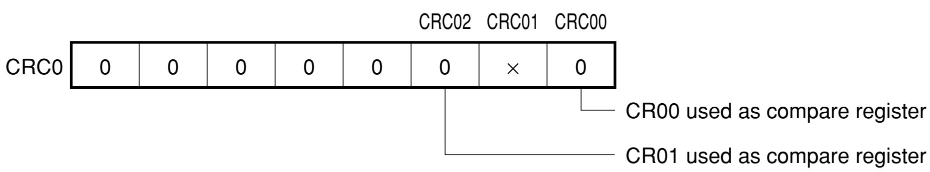
In the PPG output operation, square waves are output from the TO0 pin with the pulse width and the cycle that correspond to the count values set beforehand in 16-bit timer capture/compare register 01 (CR01) and in 16-bit timer capture/compare register 00 (CR00), respectively.

★ **Figure 8-25. Control Register Settings for PPG Output Operation**

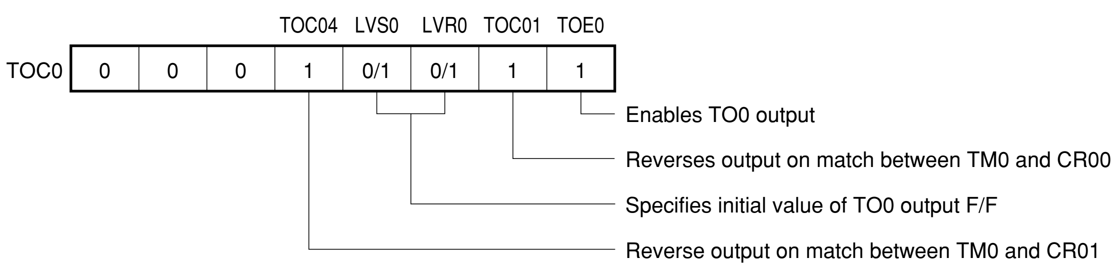
(a) 16-bit timer mode control register 0 (TMC0)



(b) Capture/compare control register 0 (CRC0)



(c) 16-bit timer output control register 0 (TOC0)

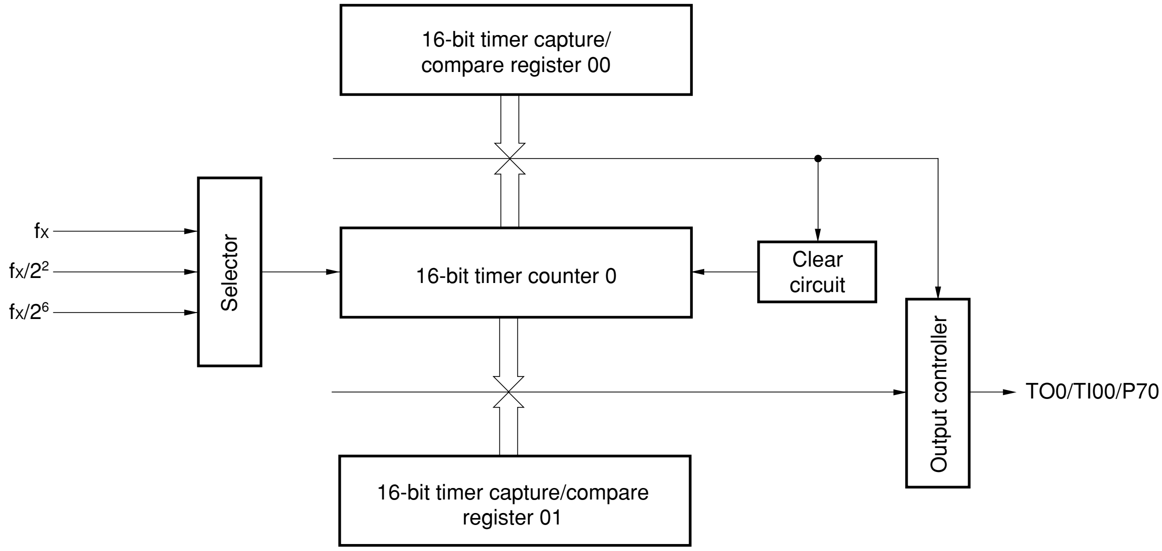


Cautions 1. CR00 and CR01 values in the following range should be set to:
0000H < CR01 < CR00 ≤ FFFFH

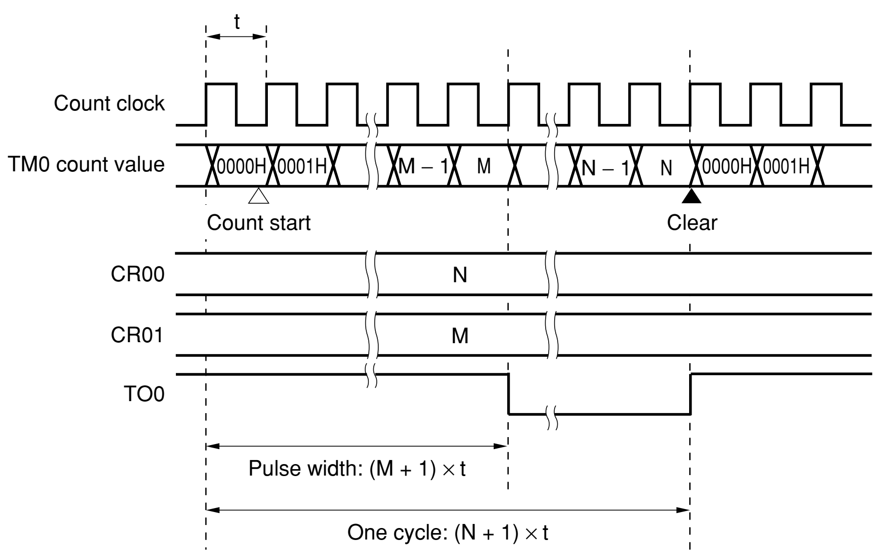
2. The cycle of the pulse generated via PPG output (CR00 setting value + 1) has a duty of (CR01 setting value + 1)/(CR00 setting value + 1).

Remark ×: Don't care

★ **Figure 8-26. PPG Output Configuration Diagram**



★ **Figure 8-27. PPG Output Operation Timing**



Remark 0000H < M < N ≤ FFFFH

★ ## 8.5 Program List

Caution The following sample program is shown as an example to describe the operation of semiconductor products and their applications. Therefore, when applying the following information to your devices, design the devices after performing evaluation under your own responsibility.

8.5.1 Interval timer

```
/******************************************************************************/
/*                                                                            */
/*           Setting example of timer 0 interval timer mode                   */
/*       Cycle set to 130 as intervalTM0 (at 8.38 MHz for 1 ms)               */
/*        Variable ppgdata prepared as rewrite data area                      */
/*         : Cycle (if 0000, no change)                                       */
/*        ppgdata to be checked at every INTTM00, and changed if required.    */
/*        Therefore, if change is required, set the change data in ppgdata.   */
/*        When changed, ppgdata cleared to 0000.                              */
/*                                                                            */
/******************************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#define intervalTM0 130                 /* Cycle data to be set to CR00 */
#pragma interrupt INTTM00 intervalint rb2
        unsigned int ppgdata;           /* Data area to be set to timer 0 */

void main(void)
{
        PCC = 0x0;                      /* Set high-speed operation mode */
        ppgdata = 0;
                                        /* Set port */
                                        /* Set the following to output */
        P7 = 0b11111110;                /* Clear P70 */
        PM7.0 = 0;                      /* Set P70 as output */
                                        /* Set interrupt */
        TMMK00 = 0;                     /* Cancel INTTM00 interrupt mask */
                                        /* Set timer 0 */
        PRM0 = 0b00000010;              /* Count clock is fx/2^6 */
        CRC0 = 0b00000000;              /* Set CR00 and CR01 to compare register */
        CR00 = intervalTM0;             /* Set cycle initial value to CR00 */
        TOC0 = 0b00000111;              /* Invert on match with CR00, initial value L */
        TMC0 = 0b00001100;              /* Clear & start on match between TM0 and CR00 */
        EI();

        while(1);                       /* Loop as dummy here */
}

/* Timer 0 interrupt function */
void intervalint()
{
        unsigned int work;
/***************************************************/
/*                                                 */
/* Define variables required for interrupt here    */
/*                                                 */
/***************************************************/
        work = ppgdata;
        if (work != 0)
        {
                CR00 = work;
                ppgdata = 0;
                if (work == 0xffff)
                {
                        TMC0 = 0b00000000;     /* Stop timer */
                }
          }
/***********************************************************/
/*                                                         */
/* Describe processing required for interrupt below        */
/*                                                         */
/***********************************************************/
}
```

8.5.2 Pulse width measurement by free-running counter and one capture register

```
/*****************************************************************************/
/*                                                                           */
/*           Timer 0 operation sample                                        */
/*       Pulse width measurement example by free-running and CR01            */
/*        Measurement results to be up to 16 bits and not checked for errors */
/*        data[0]: End flag                                                  */
/*        data[1]: Measurement results (pulse width)                         */
/*        data[2]: Previous read value                                       */
/*                                                                           */
/*****************************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#pragma interrupt INTTM01 intervalint rb2
        unsigned int data[3];           /* Data area */

void main(void)
{
        unsigned int length;
        PCC = 0x0;                      /* Set high-speed operation mode */
        data[0] = 0;
        data[1] = 0;
        data[2] = 0;
                                        /* Set port */
        PM7.0 = 1;                      /* Set P70 as input */
                                        /* Set interrupt */
        TMMK01 = 0;                     /* Cancel INTTM01 interrupt mask */
                                        /* Set timer 0 */
        PRM0 = 0b00110010;              /* Both rising and falling edges for TI00 */
                                        /* Count clock is fx/2^6              */
        CRC0 = 0b00000100;              /* Set CR01 to capture register */
        TMC0 = 0b00000100;              /* Start in free-run mode */
        EI();
        while(1){                       /* Dummy loop */
                while(data[0] == 0);    /* Wait for measurement completion */
                DI();                   /* Prohibit interrupt for exclusive operation */
                length = data[1];       /* Read measurement results */
                data[0] = 0;            /* Clear end flag */
                EI();                   /* Exclusive operation completed */
        }
}

/* Timer 0 interrupt function */
void intervalint()
{
        unsigned int work;
/***************************************************/
/*                                                 */
/* Define variables required for interrupt here    */
/*                                                 */
/***************************************************/
        work = CR01;                    /* Read capture value */
        data[1] = work - data[2];       /* Calculate and update interval */
        data[2] = work;                 /* Update read value */
        data[0] = 0xffff;               /* Set measurement completion flag */

/*********************************************************/
/*                                                       */
/* Describe processing required for interrupt below      */
/*                                                       */
/*********************************************************/
}
```

8.5.3 Two pulse widths measurement by free-running counter

```
/*****************************************************************************/
/*                                                                           */
/*          Timer 0 operation sample                                         */
/*      Two-pulse-width measurement sample by free-running                   */
/*      Measurement results to be up to 16 bits and not checked for errors   */
/*        Result area at TI00 side                                           */
/*       data[0]: End flag                                                   */
/*       data[1]: Measurement results (pulse width)                          */
/*       data[2]: Previous read value                                        */
/*        Result area at TI01 side                                           */
/*       data[3]: End flag                                                   */
/*       data[4]: Measurement results (pulse width)                          */
/*       data[5]: Previous read value                                        */
/*                                                                           */
/*****************************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#pragma interrupt INTTM00 intervalint rb2
#pragma interrupt INTTM01 intervalint2 rb2
        unsigned int data[6];           /* Data area */

void main(void)
{
        unsigned int length,length2;
        PCC = 0x0;                      /* Set high-speed operation mode */
        data[0] = 0;                    /* Clear data area */
        data[1] = 0;
        data[2] = 0;
        data[3] = 0;
        data[4] = 0;
        data[5] = 0;

                                        /* Set port */
        PM7.0 = 1;                      /* Set P70 as input */
        PM7.1 = 1;                      /* Set P71 as input */
                                        /* Set interrupt */
        TMMK01 = 0;                     /* Cancel INTTM01 interrupt mask */
        TMMK00 = 0;                     /* Cancel INTTM00 interrupt mask */
                                        /* Set timer 0 */
        PRM0 = 0b11110010;              /* Both rising and falling edges */
                                        /* Count clock is fx/2^6            */
        CRC0 = 0b00000101;              /* Set CR00 and CR01 to capture register */
        TMC0 = 0b00000100;              /* Start in free-run mode */
        EI();
        while(1){                       /* Dummy loop */
                if(data[0] != 0)        /* TI00 measurement completion check */
                {
                        TMMK01 = 1;             /* INTTM01 interrupt prohibited for
                                                   exclusive operation */
                        length = data[1];       /* Read measurement results */
                        data[0] = 0;            /* Clear end flag */
                        TMMK01 = 0;             /* Exclusive operation completed */
                }
                if(data[3] != 0)        /* TI01 measurement completion check */
                {
                        TMMK00 = 1;             /* INTTM00 interrupt prohibited for
                                                   exclusive operation */
                        length2 = data[4];      /* Read measurement results */
                        data[3] = 0;            /* Clear end flag */
                        TMMK00 = 0;             /* Exclusive operation completed */
                }
        }
}
```

```
/* INTTM00 interrupt function */
void intervalint()
{
        unsigned int work;
/*****************************************************/
/*                                                   */
/* Define variables required for interrupt here      */
/*                                                   */
/*****************************************************/
        work = CR00;                    /* Read capture value */
        data[4] = work - data[5];       /* Calculate and update interval */
        data[5] = work;                 /* Update read value */
        data[3] = 0xffff;               /* Set measurement completion flag */

/*******************************************************/
/*                                                     */
/* Describe processing required for interrupt below    */
/*                                                     */
/*******************************************************/
}
/* INTTM01 interrupt function */
void intervalint2()
{
        unsigned int work;
/*****************************************************/
/*                                                   */
/* Define variables required for interrupt here      */
/*                                                   */
/*****************************************************/
        work = CR01;                    /* Read capture value */
        data[1] = work - data[2];       /* Calculate and update interval */
        data[2] = work;                 /* Update read value */
        data[0] = 0xffff;               /* Set measurement completion flag */

/*******************************************************/
/*                                                     */
/* Describe processing required for interrupt below    */
/*                                                     */
/*******************************************************/
}
```

8.5.4 Pulse width measurement by restart

```
/***************************************************************************/
/*                                                                         */
/*          Timer 0 operation sample                                       */
/*      Pulse width measurement example by restart                         */
/*      Measurement results up to 16 bits, not to be checked for errors    */
/*       data[0]: End flag                                                 */
/*       data[1]: Measurement results (pulse width)                        */
/*       data[2]: Previous read value                                      */
/*                                                                         */
/***************************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#pragma interrupt INTTM01 intervalint rb2
        unsigned int data[3];           /* Data area */

void main(void)
{
        unsigned int length;
        PCC = 0x0;                      /* Set high-speed operation mode */
        data[0] = 0;
        data[1] = 0;
        data[2] = 0;
                                        /* Set port */
        PM7.0 = 1;                      /* Set P70 as input */
                                        /* Set interrupt */
        TMMK01 = 0;                     /* Cancel INTTM01 interrupt mask */
                                        /* Set timer 0 */
        PRM0 = 0b00110010;              /* Both rising and falling edges  */
                                        /* Count clock is fx/2^6        */
        CRC0 = 0b00000100;              /* Set CR01 to capture register   */
        TMC0 = 0b00001000;              /* Clear & start at TI00 valid edge */
        EI();
        while(1){                       /* Dummy loop */
                if(data[0] != 0)        /* Wait for TI00 measurement completion */
                {
                        TMMK01 = 1;             /* Prohibit INTTM01 interrupt for
                                                   exclusive operation */
                        length = data[1]+data[2];       /* Cycle calculation based on
                                                           measurement results */
                        data[0] = 0;            /* Clear end flag */
                        TMMK01 = 0;             /* Exclusive operation completed */
                }
        }
}

/* Timer 0 interrupt function */
void intervalint()
{
/****************************************************/
/*                                                  */
/* Define variables required for interrupt here     */
/*                                                  */
/****************************************************/
        data[2] = data[1];              /* Update old data */
        data[1] = CR01;                 /* Update read value */
        data[0] = 0xffff;               /* Set measurement completion flag */

/******************************************************/
/*                                                    */
/* Describe processing required for interrupt below   */
/*                                                    */
/******************************************************/
}
```

8.5.5 PPG output

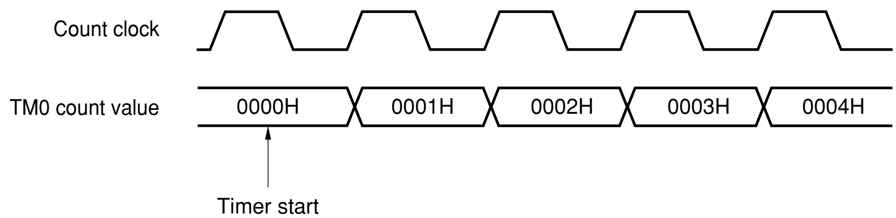
```
/*******************************************************************************/
/*                                                                             */
/*          Timer 0 PPG mode setting example                                   */
/*      Cycle set to 130 as intervalTM0                                        */
/*      Active period set to 65 as active_time                                 */
/*       Array ppgdata prepared as data area for rewriting                     */
/*       [0]: Active period (0000: no change, 0xffff: stop)                    */
/*       [1]: Cycle (0000: no change)                                          */
/*       ppgdata to be checked at every INTTM00, and changed if required.      */
/*       Therefore, if change is required, set the change data in ppgdata.     */
/*       When changed, ppgdata cleared to 0000.                                */
/*                                                                             */
/*******************************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#define intervalTM0 130             /* Cycle data to be set to CR00 */
#define active_time 65              /* Initial value data of CR01 */
#pragma interrupt INTTM00 ppgint rb2
        unsigned int ppgdata[2];    /* Data area to be set to timer 0 */

void main(void)
{
        PCC = 0x0;                  /* Set high-speed operation mode */
        ppgdata[0] = 0;
        ppgdata[1] = 0;
                                    /* Set port */
        P7 = 0b11111110;            /* Clear P70 */
        PM7.0 = 0;                  /* Set P70 to output */
                                    /* Set interrupt */
        TMMK00 = 0;                 /* Cancel INTTM00 interrupt mask */
                                    /* Set timer 0 */
        PRM0 = 0b00000010;          /* Count clock is fx/2^6 */
        CRC0 = 0b00000000;          /* Set CR00 and CR01 to compare register */
        CR00 = intervalTM0;         /* Set initial value of cycle */
        CR01 = active_time;         /* Set initial value of active period */
        TOC0 = 0b00010111;          /* Inverted on match between CR00 and CR01,
                                       initial value L */
        TMC0 = 0b00001100;          /* Clear & start on match between TM0 and CR00 */
        EI();

        while(1);
}

/* Timer 0 interrupt function */
void ppgint()
{
        unsigned int work;
        work = ppgdata[0];
        if (work != 0)
        {
                CR01 = work;
                ppgdata[0] = 0;
                if (work == 0xffff)
                {
                        TMC0 = 0b00000000;     /* Stop timer */
                }
        }
        work = ppgdata[1];
        if (work != 0)
        {
                CR00 = work;
                ppgdata[1]=0;
        }
}
```

8.6 Cautions Related to 16-Bit Timer/Event Counter 0

(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 16-bit timer counter 0 (TM0) is started asynchronously to the count clock.

Figure 8-28. Start Timing of 16-Bit Timer Counter 0 (TM0)



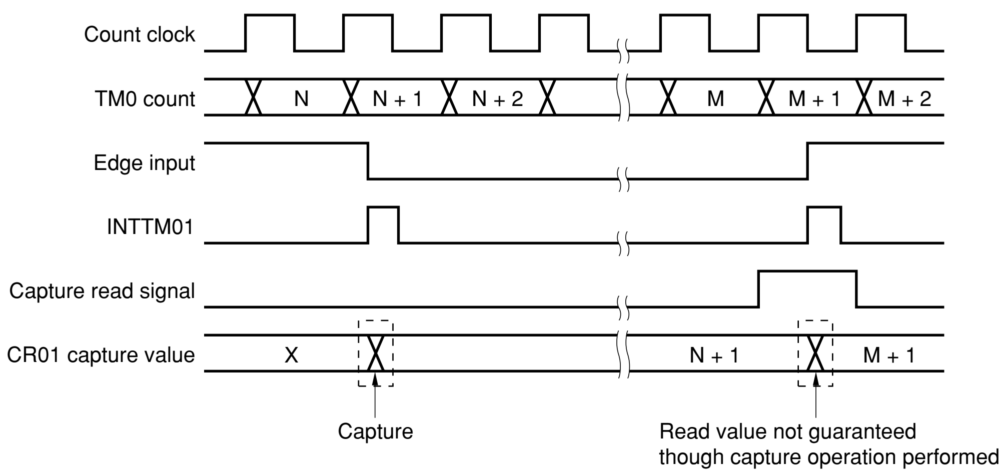
(2) 16-bit timer capture/compare register setting (clear & start mode entered on match between TM0 and CR00)

Set 16-bit timer capture/compare registers 00, 01 (CR00, CR01) to other than 0000H. This means a 1-pulse count operation cannot be performed.

★ **(3) Capture register data retention timing**

If the valid edge of the TI00 pin is input during 16-bit timer capture/compare register 01 (CR01) read, CR01 performs a capture operation but, the read value at this time is not guaranteed. The interrupt request signal (INTTM01) is generated upon detection of the valid edge.

Figure 8-29. Capture Register Data Retention Timing



(4) Valid edge setting

Set the valid edge of the TI00 pin after clearing bits 2 and 3 (TMC02 and TMC03) of 16-bit timer mode control register 0 (TMC0) to 0, 0, respectively, and then stopping the timer operation. The valid edge is set by bits 4 and 5 (ES00 and ES01) of prescaler mode register 0 (PRM0).

(5) Operation of OVF0 flag

<1> The OVF0 flag is also set to 1 in the following case.

★ Either of the clear & start mode entered on a match between TM0 and CR00, clear & start at the valid edge of TI00, or free-running mode is selected.

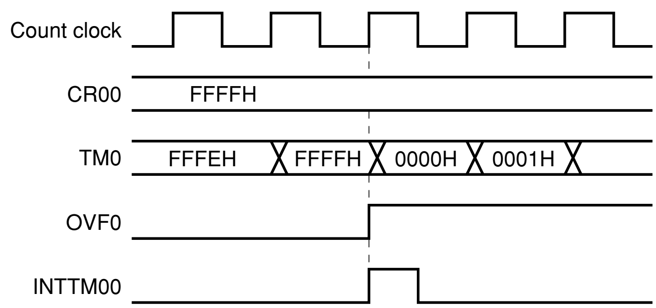
↓

CR00 is set to FFFFH.

↓

When TM0 is counted up from FFFFH to 0000H.

Figure 8-30. Operation Timing of OVF0 Flag



<2> Even if the OVF0 flag is cleared before the next count clock is counted (before TM0 becomes 0001H) after the occurrence of a TM0 overflow, the OVF0 flag is reset newly and clear is disabled.

(6) Conflicting operations

When the 16-bit timer capture/compare register (CR00/CR01) is used as a compare register, if the write period and the match timing of 16-bit timer counter 0 (TM0) conflict, match determination is not successfully done. Do not perform a write operation of CR00/CR01 near the match timing.

(7) Timer operation

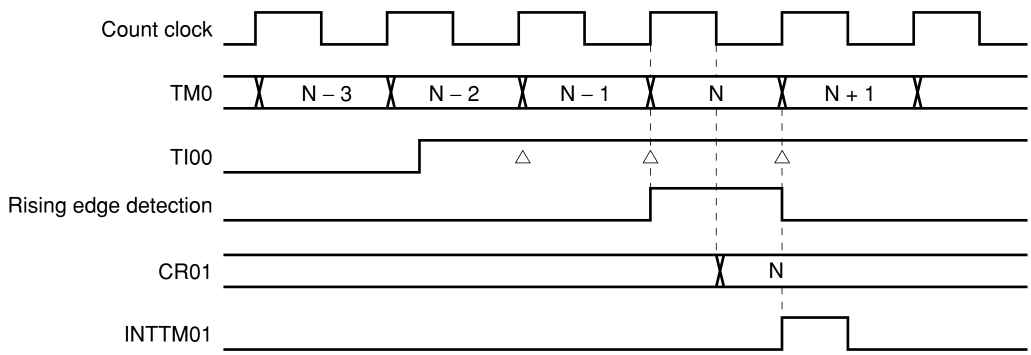
<1> Even if 16-bit timer counter 0 (TM0) is read, the value is not captured by 16-bit timer capture/compare register 01 (CR01).

<2> Regardless of the CPU's operation mode, when the timer stops, the signals input to pins TI00/TI01 are not acknowledged.

(8) Capture operation

<1> If TI00 is specified as the valid edge of the count clock, a capture operation by the capture register specified as the trigger for TI00 is not possible.
<2> If both the rising and falling edges are selected as the valid edges of TI00, capture is not performed.
<3> To ensure the reliability of the capture operation, the capture trigger requires a pulse longer than two cycles of the count clock selected by prescaler mode register 0 (PRM0).

Figure 8-31. CR01 Capture Operation with Rising Edge Specified



<4> The capture operation is performed at the fall of the count clock. An interrupt request input (INTTM0n), however, occurs at the rise of the next count clock.

(9) Compare operation

<1> When the 16-bit timer capture/compare register (CR00/CR01) is overwritten during timer operation, match interrupt may be generated or the clear operation may not be performed normally if that value is close to or large than the timer value.
<2> The capture operation may not be performed for CR00/CR01 set in compare mode even if a capture trigger is input.

(10) Edge detection

<1> If the TI00 pin or the TI01 pin is high level immediately after system reset and the rising edge or both the rising and falling edges are specified as the valid edge for the TI00 pin or TI01 pin to enable 16-bit timer counter 0 (TM0) operation, a rising edge is detected immediately. Be careful when pulling up the TI00 pin or the TI01 pin. However, the rising edge is not detected at restart after the operation has been stopped once.
<2> The sampling clock used to remove noise differs when a TI00 valid edge is used as the count clock and when it is used as a capture trigger. In the former case, the count clock is $f_X/2^3$, and in the latter case the count clock is selected by prescaler mode register 0 (PRM0). The capture operation is not performed until the valid edge is sampled and the valid level is detected twice, thus eliminating noise with a short pulse width.

★ **(11) STOP mode or main system clock stop mode setting**

Except when TI00, TI01 input is selected, stop the timer operation before setting STOP mode or main system clock stop mode; otherwise the timer may malfunction when the main system clock starts.

9.1 Functions of 8-Bit Timer/Event Counters 50, 51

8-bit timer/event counters 50, 51 (TM50, TM51) have the following two modes.

(1) Mode using 8-bit timer/event counters 50, 51 alone (discrete mode)

The timer operates as 8-bit timer/event counter 50 or 51.
It has the following functions.

<1> Interval timer
Interrupt requests are generated at the preset interval.
- Number of counts: 1 to 256

<2> External event counter
The number of pulses with high/low level widths of the signal input externally can be measured.

<3> Square-wave output
A square wave with an arbitrary frequency can be output.
- Cycle: $(1 \times 2$ to $256 \times 2) \times$ Cycles of count clock

<4> PWM output
A pulse with an arbitrary duty ratio can be output.
- Cycle: Count clock $\times$ 256
- Duty ratio: Set value of compare register/256

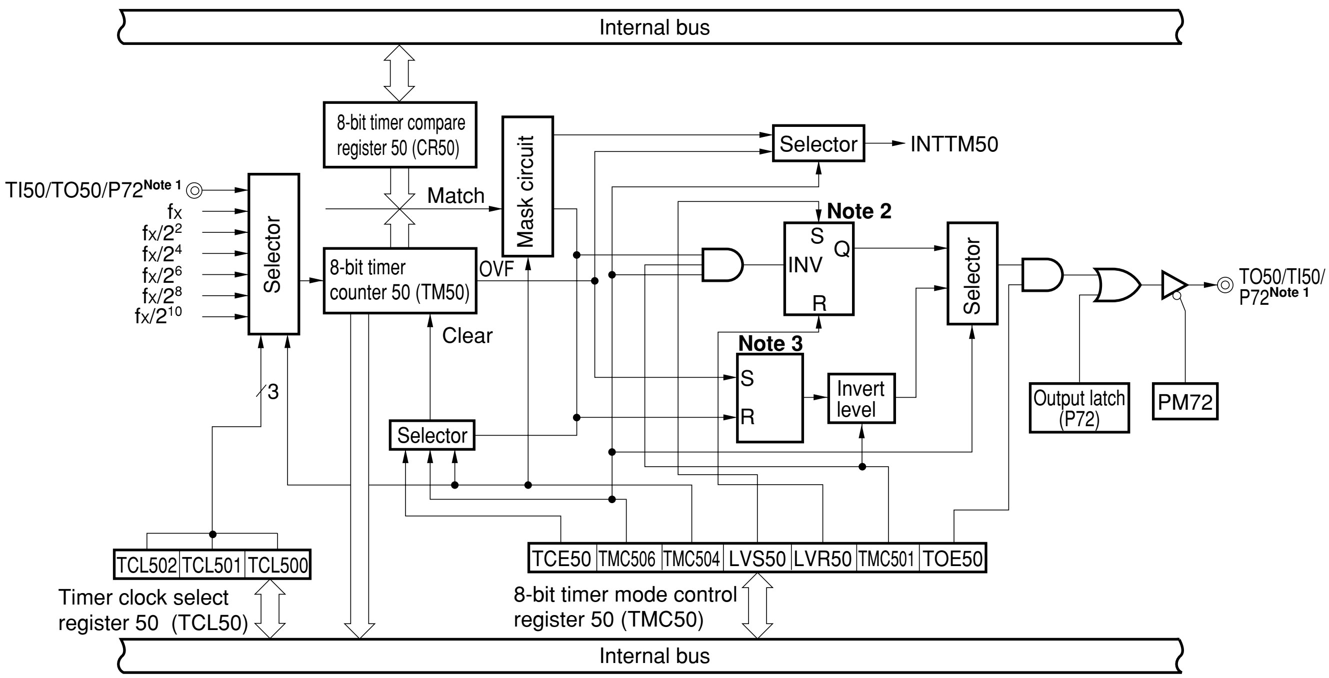
(2) Mode using cascade connection (16-bit resolution: cascade connection mode)

The timer operates as a 16-bit timer/event counter by combining two 8-bit timer/event counters.
It has the following functions.

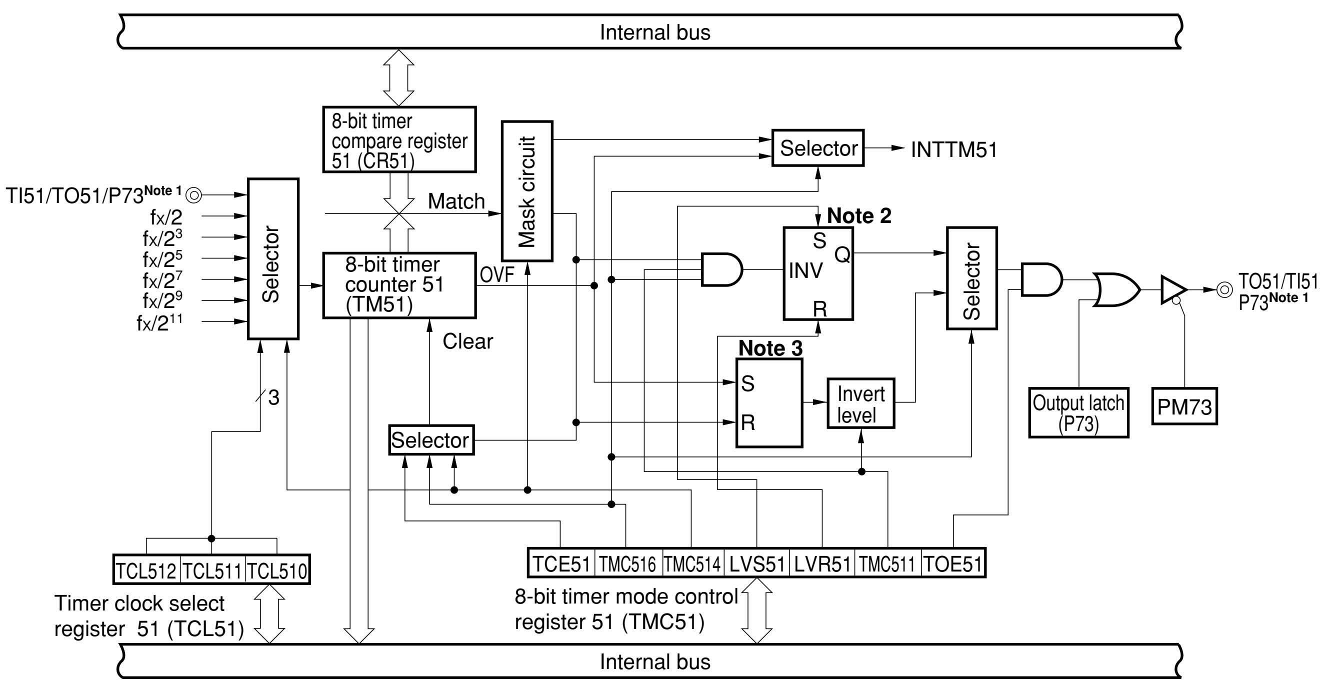
- Interval timer with 16-bit resolution
- External event counter with 16-bit resolution
- Square-wave output with 16-bit resolution

Figures 9-1 and 9-2 show block diagrams of 8-bit timer/event counters 50 and 51.

★ **Figure 9-1. Block Diagram of 8-Bit Timer/Event Counter 50**



★ **Figure 9-2. Block Diagram of 8-Bit Timer/Event Counter 51**



Notes 1. The respective combinations, TI50 input and TO50 output, and TI51 input and TO51 output, cannot be used at the same time.

2. Timer output F/F

3. PWM output F/F

9.2 Configuration of 8-Bit Timer/Event Counters 50, 51

8-bit timer/event counters 50, 51 consist of the following hardware.

Table 9-1. Configuration of 8-Bit Timer/Event Counters 50, 51

Item	Configuration
Timer counter	8-bit timer counter 5n (TM5n)
Register	8-bit timer compare register 5n (CR5n)
Timer input	TI5n
Timer output	TO5n
Control registers	Timer clock select register 5n (TCL5n) 8-bit timer mode control register 5n (TMC5n) Port mode register 7 (PM7) Port 7 (P7)

(1) 8-bit timer counter 5n (TM5n: n = 0, 1)

TM5n is an 8-bit read-only register that counts the count pulses.

The counter is incremented in synchronization with the rising edge of the count clock.

When TM50 and TM51 can be connected in cascade and used as a 16-bit timer, they can be read by a 16-bit memory manipulation instruction. However, since they are connected by an internal 8-bit bus, TM50 and TM51 are read separately twice in that order. Thus, take reading during the count change into consideration and compare them by reading twice. When the count value is read during operation, the count clock input is temporarily stopped[Note], and then the count value is read. In the following situations, count value is cleared to 00H.

<1> $\overline{\text{RESET}}$ input

<2> When TCE5n is cleared

<3> When TM5n and CR5n match in the clear & start mode entered on a match between TM5n and CR5n.

★ **Note** An error may occur in the count. Select a count clock with a high/low-level waveform longer than two cycles of the CPU clock.

Caution In cascade connection mode, the count value is reset to 0000H when TCE50 of the lowest timer is cleared.

(2) 8-bit timer compare register 5n (CR5n: n = 0, 1)

When CR5n is used as a compare register in other than PWM mode, the value set in CR5n is constantly compared with the 8-bit timer counter 5n (TM5n) count value, and an interrupt request (INTTM5n) is generated if they match.

★ In PWM mode, the TO5n pin goes to the active level by the overflow of TM5n. When the values of TM5n and CR5n match, the TO5n pin goes to the inactive level.

It is possible to rewrite the value of CR5n within 00H to FFH during a count operation.

When TM50 and TM51 can be connected in cascade and used as a 16-bit timer, CR50 and CR51 operate as a 16-bit compare register. This register compares the count value with the register value, and if the values match, an interrupt request (INTTM50) is generated. The INTTM51 interrupt request is also generated at this time. Thus, mask the INTTM51 interrupt request. CR5n is set by an 8-bit memory manipulation instruction. $\overline{\text{RESET}}$ input makes CR5n undefined.

Caution In cascade connection mode, stop the timer operation before setting data.

Remark n = 0, 1

9.3 Registers to Control 8-Bit Timer/Event Counters 50, 51

The following four types of registers are used to control 8-bit timer/event counters 50, 51.

- Timer clock select register 5n (TCL5n)
- 8-bit timer mode control register 5n (TMC5n)
- Port mode register 7 (PM7)
- Port 7 (P7)

Remark n = 0, 1

(1) Timer clock select register 5n (TCL5n: n = 0, 1)

This register sets the count clock of 8-bit timer/event counter 5n and the valid edge of TI50, TI51 input.
TCL5n is set by an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TCL5n to 00H.

Figure 9-3. Format of Timer Clock Select Register 50 (TCL50)

Address: FF71H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
TCL50	0	0	0	0	0	TCL502	TCL501	TCL500

★

TCL502	TCL501	TCL500	Count clock selection		
				fx = 8.38 MHz	fx = 12 MHz[Note]
0	0	0	TI50 falling edge	–	–
0	0	1	TI50 rising edge	–	–
0	1	0	fx	8.38 MHz	12 MHz
0	1	1	$f_X/2^2$	2.09 MHz	3 MHz
1	0	0	$f_X/2^4$	523 kHz	750 kHz
1	0	1	$f_X/2^6$	131 kHz	187 kHz
1	1	0	$f_X/2^8$	32.7 kHz	46.8 kHz
1	1	1	$f_X/2^{10}$	8.18 kHz	11.7 kHz

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Cautions 1. When rewriting TCL50 to other data, stop the timer operation beforehand.
2. Be sure to clear bits 3 to 7 to 0.

★ **Remarks 1.** When cascade connection is used, only TCL50 is valid for count clock setting.
2. fx: Main system clock oscillation frequency

Figure 9-4. Format of Timer Clock Select Register 51 (TCL51)

Address: FF79H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
TCL51	0	0	0	0	0	TCL512	TCL511	TCL510

★

TCL512	TCL511	TCL510	Count clock selection		
				f_X = 8.38 MHz	f_X = 12 MHz^Note
0	0	0	TI51 falling edge	–	–
0	0	1	TI51 rising edge	–	–
0	1	0	$f_X/2$	4.19 MHz	6 MHz
0	1	1	$f_X/2^3$	1.04 MHz	1.5 MHz
1	0	0	$f_X/2^5$	261 kHz	375 kHz
1	0	1	$f_X/2^7$	65.4 kHz	93.7 kHz
1	1	0	$f_X/2^9$	16.3 kHz	23.4 kHz
1	1	1	$f_X/2^{11}$	4.09 kHz	5.85 kHz

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Cautions 1. When rewriting TCL51 to other data, stop the timer operation beforehand.
2. Be sure to clear bits 3 to 7 to 0.

★ **Remarks 1.** When cascade connection is used, only TCL50 is valid for count clock setting.
2. f_X: Main system clock oscillation frequency

(2) 8-bit timer mode control register 5n (TMC5n: n = 0, 1)

TMC5n is a register that makes the following six settings.

<1> 8-bit timer counter 5n (TM5n) count operation control
<2> 8-bit timer counter 5n (TM5n) operating mode selection
<3> Discrete mode/cascade connection mode selection (TMC51 only)
<4> Timer output F/F (flip-flop) status setting
<5> Active level selection in timer F/F control or PWM (free-running) mode.
<6> Timer output control

TMC5n is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TMC5n to 00H.

Figure 9-5. Format of 8-Bit Timer Mode Control Register 50 (TMC50)

Address: FF70H After reset: 00H R/W

Symbol	<7>	6	5	4	<3>	<2>	1	<0>
★ TMC50	TCE50	TMC506	0	0	LVS50	LVR50	TMC501	TOE50

TCE50	TM50 count operation control
0	After clearing to 0, count operation disabled (prescaler disabled)
1	Count operation start

TMC506	TM50 operating mode selection
0	Clear and start mode by match between TM50 and CR50
1	PWM (free-running) mode

LVS50	LVR50	Timer output F/F status setting
0	0	No change
0	1	Timer output F/F reset (0)
1	0	Timer output F/F set (1)
1	1	Setting prohibited

TMC501	In other modes (TMC506 = 0)	In PWM mode (TMC506 = 1)
	Timer F/F control	Active level selection
0	Inversion operation disabled	Active high
1	Inversion operation enabled	Active low

TOE50	Timer output control
0	Output disabled (port mode)
1	Output enabled

Remarks 1. In PWM mode, PWM output will be inactive because TCE50 = 0.

2. If LVS50 and LVR50 are read after data is set, 0 is read.

Figure 9-6. Format of 8-Bit Timer Mode Control Register 51 (TMC51)

Address: FF78H After reset: 00H R/W

Symbol	<7>	6	5	4	<3>	<2>	1	<0>
TMC51	TCE51	TMC516	0	TMC514	LVS51	LVR51	TMC511	TOE51

TCE51	TM51 count operation control
0	After clearing to 0, count operation disabled (prescaler disabled)
1	Count operation start

TMC516	TM51 operating mode selection
0	Clear and start mode by match between TM51 and CR51
1	PWM (free-running) mode

TMC514	Discrete mode/cascade connection mode selection
0	Discrete mode
1	Cascade connection mode (TM50: Lower timer, TM51: Higher timer)

LVS51	LVR51	Timer output F/F status setting
0	0	No change
0	1	Timer output F/F reset (0)
1	0	Timer output F/F set (1)
1	1	Setting prohibited

TMC511	In other modes (TMC516 = 0)	In PWM mode (TMC516 = 1)
	Timer F/F control	Active level selection
0	Inversion operation disabled	Active high
1	Inversion operation enabled	Active low

TOE51	Timer output control
0	Output disabled (port mode)
1	Output enabled

Remarks 1. In PWM mode, PWM output will be inactive because TCE51 = 0.

2. If LVS51 and LVR51 are read after data is set, 0 is read.

(3) Port mode register 7 (PM7)

This register sets port 7 input/output in 1-bit units.

When using the P72/TO50/TI50 and P73/TI51/TO51 pins for timer output, clear PM72, PM73, and the output latches of P72 and P73 to 0.

★ When using the P72/TO50/TI50 and P73/TI51/TO51 pins for timer input, set PM72 and PM73 to 1. At this time, the output latches of P72 and P73 can be either 0 or 1.

PM7 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM7 to FFH.

Figure 9-7. Format of Port Mode Register 7 (PM7)

Address: FF27H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM7	1	1	PM75	PM74	PM73	PM72	PM71	PM70

PM7n	P7n pin I/O mode selection (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

9.4 Operation of 8-Bit Timer/Event Counters 50, 51

9.4.1 8-bit interval timer operation

The 8-bit timer/event counters operate as interval timers that generate interrupt requests repeatedly at intervals of the count value preset to 8-bit timer compare register 5n (CR5n).

When the count value of 8-bit timer counter 5n (TM5n) matches the value set to CR5n, counting continues with the TM5n value cleared to 0 and an interrupt request signal (INTTM5n) is generated.

The count clock of TM5n can be selected with bits 0 to 2 (TCL5n0 to TCL5n2) of timer clock select register 5n (TCL5n).

[Setting]

<1> Set the registers.

- TCL5n: Select count clock.
- CR5n: Compare value
- TMC5n: Count operation stop, clear & start mode on match between TM5n and CR5n.
  (TMC5n = 0000×××0B × = don't care)

<2> After TCE5n = 1 is set, count operation starts.

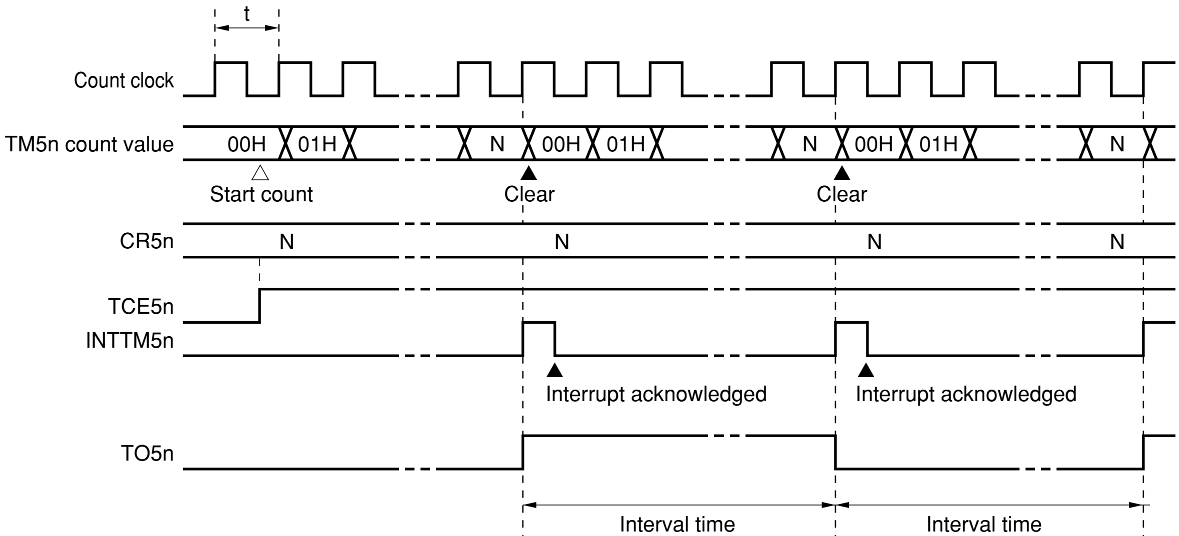
<3> If the values of TM5n and CR5n match, INTTM5n is generated (TM5n is cleared to 00H).

<4> INTTM5n is generated repeatedly at the same interval. Clear TCE5n to 0 to stop the count operation.

Remark n = 0, 1

Figure 9-8. Interval Timer Operation Timing (1/3)

(a) Basic operation



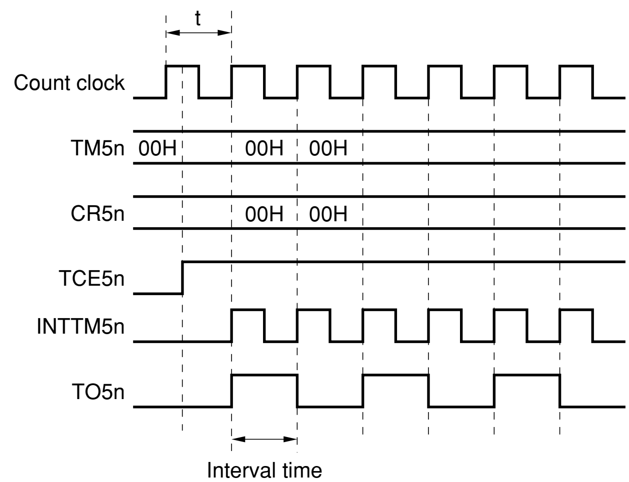
Remarks 1. Interval time = (N + 1) × t

N = 00H to FFH

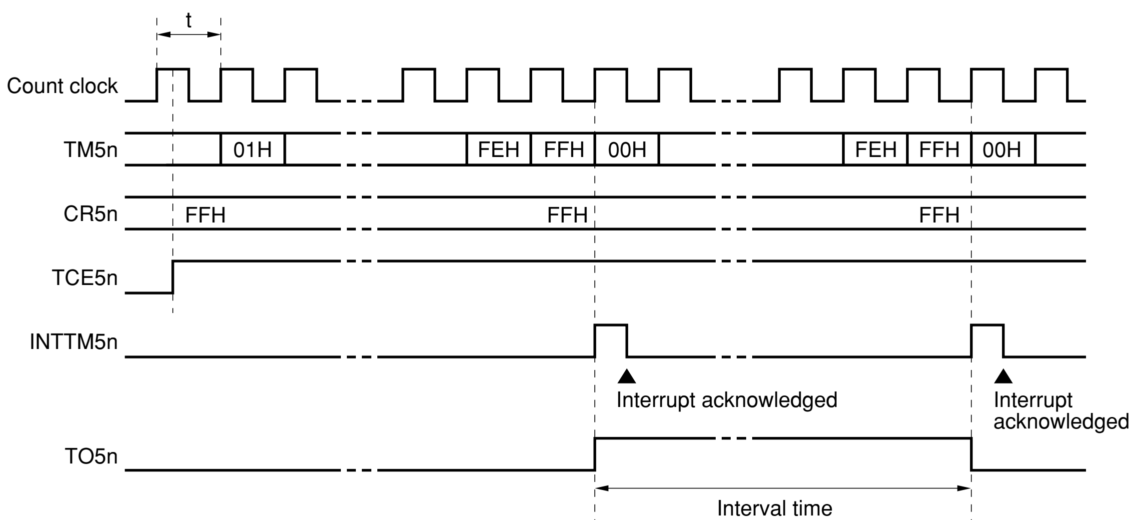
2. n = 0, 1

Figure 9-8. Interval Timer Operation Timing (2/3)

(b) When CR5n = 00H



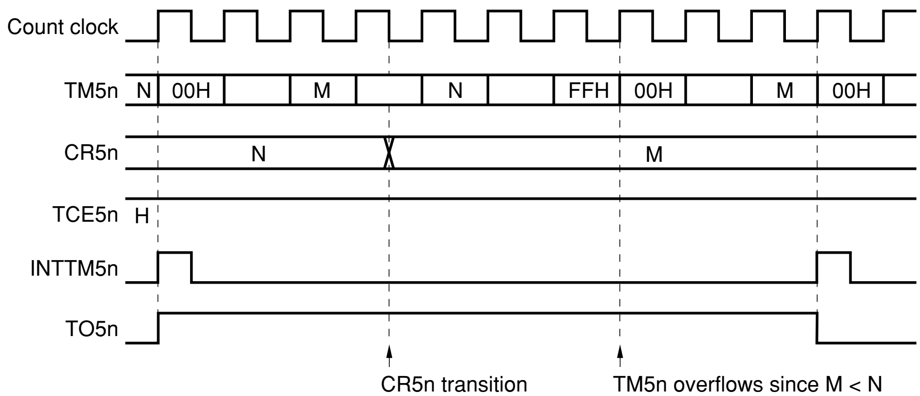
(c) When CR5n = FFH



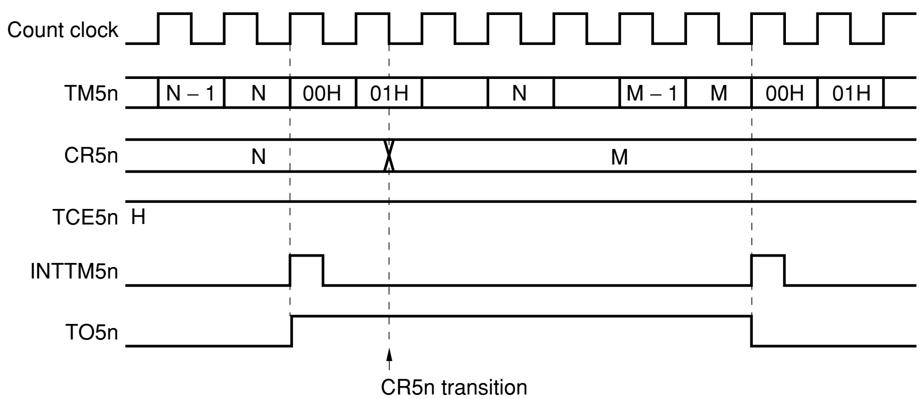
Remark n = 0, 1

Figure 9-8. Interval Timer Operation Timing (3/3)

(d) Operated by CR5n transition (M < N)



(e) Operated by CR5n transition (M > N)



Remark n = 0, 1

9.4.2 External event counter operation

The external event counter counts the number of external clock pulses to be input to TI5n using 8-bit timer counter 5n (TM5n).

TM5n is incremented each time the valid edge specified by timer clock select register 5n (TCL5n) is input. Either the rising or falling edge can be selected.

When the TM5n count value matches the value of 8-bit timer compare register 5n (CR5n), TM5n is cleared to 0 and an interrupt request signal (INTTM5n) is generated.

Whenever the TM5n count value matches the value of CR5n, INTTM5n is generated.

★ **[Setting]**

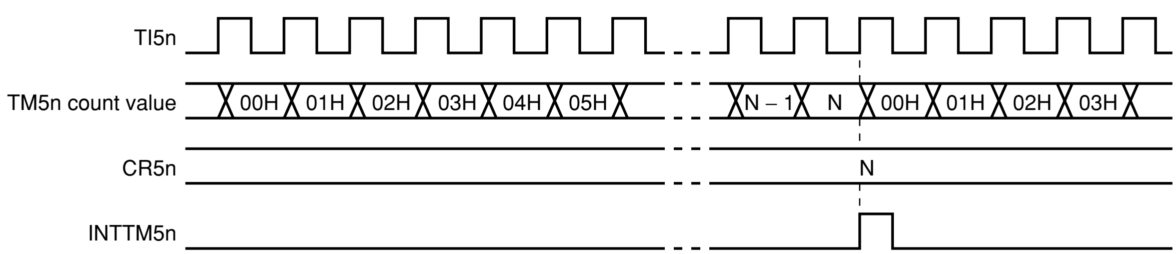
<1> Set each register
- TCL5n: Edge selection of TI5n input
  Falling edge of TI5n → TCL5n = 00H
  Rising edge of TI5n → TCL5n = 01H
- CR5n: Compare value
- TMC5n: Count operation stop, clear & start mode on match between TM5n and CR5n, timer F/F inverted operation disable, timer output disable
  (TMC5n = 0000××00B, × = don't care)

<2> When TCE5n = 1 is set, the number of pulses input from TI5n is counted.

<3> When the values of TM5n and CR5n match, INTTM5n is generated (TM5n is cleared to 00H).

<4> Each time the values of TM5n and CR5n match, INTTM5n is generated.

Figure 9-9. External Event Counter Operation Timing (with Rising Edge Specified)



Remarks 1. N = 00H to FFH

2. n = 0, 1

9.4.3 Square-wave output (8-bit resolution) operation

A square wave with any selected frequency is output at intervals determined by the value preset to 8-bit timer compare register 5n (CR5n).

The TO5n pin output status is reversed at intervals determined by the count value preset to CR5n by setting bit 0 (TOE5n) of 8-bit timer mode control register 5n (TMC5n) to 1. This enables a square wave with any selected frequency to be output (duty = 50%).

[Setting]

<1> Set each register

- Clear port latches (P72, P73)[Note] and port mode registers (PM72, PM73)[Note] to 0.
- TCL5n: Select count clock
- CR5n: Compare value
- TMC5n: Count operation stop, clear & start mode on match between TM5n and CR5n

LVS5n	LVR5n	Timer Output F/F Status Setting
1	0	High-level output
0	1	Low-level output

Timer output F/F reverse enable
Timer output enable → TOE5n = 1
★ (TMC5n = 00001011B or 00000111B)

<2> After TCE5n = 1 is set, the count operation starts.

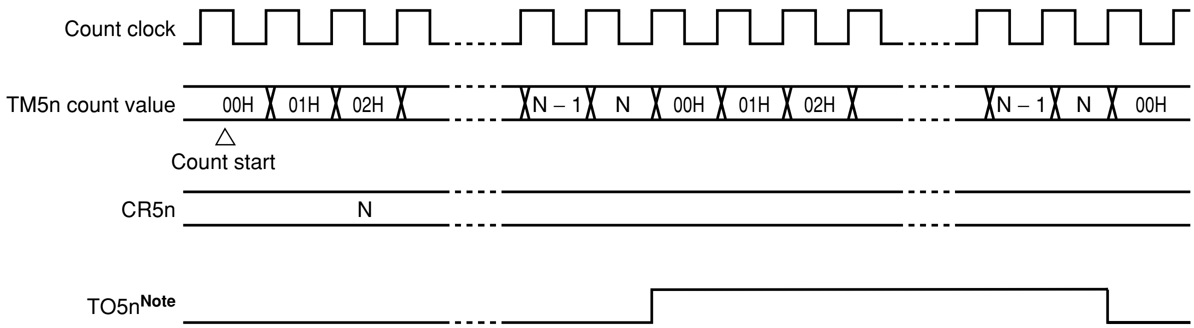
<3> Timer output F/F is reversed by match between TM5n and CR5n. After INTTM5n is generated, TM5n is cleared to 00H.

<4> Timer output F/F is reversed at the same interval and a square wave is output from TO5n.
★ The frequency is as follows.

- Frequency = $f_{CNT}/2\ (N + 1)$
  (N = 00H to FFH, f_{CNT}: Count clock)

Note 8-bit timer/event counter 50: P72, PM72
8-bit timer/event counter 51: P73, PM73

Figure 9-10. Square-Wave Output Operation Timing



Note The TO5n output initial value can be set by bits 2 and 3 (LVR5n, LVS5n) of 8-bit timer mode control register 5n (TMC5n).

Remarks 1. N = 00H to FFH

2. n = 0, 1

9.4.4 8-bit PWM output operation

The 8-bit timer/event counter operates as PWM output when bit 6 (TMC5n6) of 8-bit timer mode control register 5n (TMC5n) is set to 1.

The duty ratio pulse is determined by the value set to 8-bit timer compare register 5n (CR5n).

Set the active level width of the PWM pulse to CR5n. The active level can be selected with bit 1 (TMC5n1) of TMC5n.

The count clock can be selected with bits 0 to 2 (TCL5n0 to TCL5n2) of timer clock select register 5n (TCL5n).

PWM output enable/disable can be selected with bit 0 (TOE5n) of TMC5n.

★ • Cycle = Count clock × 256

★ • Duty ratio = $\dfrac{\text{Set value of compare register}}{256}$

Caution CR5n can be rewritten in PWM mode only once per cycle.

Remark n = 0, 1

(1) PWM output basic operation

★ **[Setting]**

<1> Set each register.

- Clear port latches (P72, P73)[Note] and port mode registers (PM72, PM73)[Note] to 0.
- TCL5n: Count clock selection
- CR5n: Compare value
- TMC5n: Count operation stop, PWM mode selection, timer output F/F not changed

TMC5n1	Active Level Selection
0	Active high
1	Active low

Timer output enabled
(TMC5n = 01000001B or 01000011B)

<2> When TCE5n = 1 is set, the count operation is started.
To stop the count operation, clear TCE5n to 0.

Note 8-bit timer/event counter 50: P72, PM72
8-bit timer/event counter 51: P73, PM73

[PWM output operation]

<1> PWM output (output from TO5n) outputs an inactive level after the count operation starts until an overflow occurs.

<2> When an overflow occurs, the active level is output.
The active level is output until CR5n matches the count value of 8-bit timer counter 5n (TM5n).

<3> After CR5n matches the count value, PWM output outputs the inactive level again until an overflow occurs.

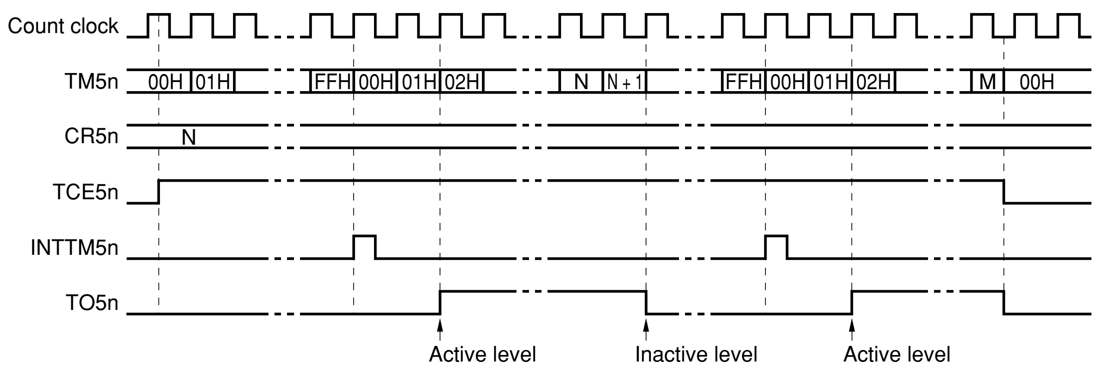
<4> Operations <2> and <3> are repeated until the count operation stops.

<5> When the count operation is stopped by setting TCE5n = 0, PWM output becomes the inactive level.

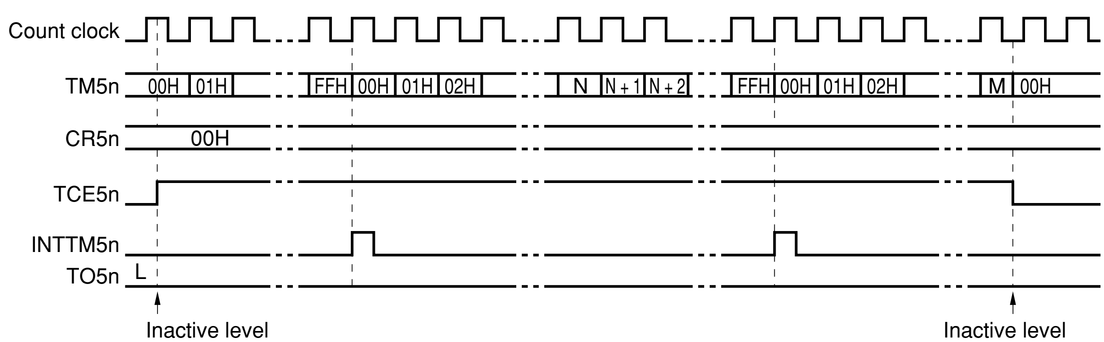
Remark n = 0, 1

Figure 9-11. PWM Output Operation Timing

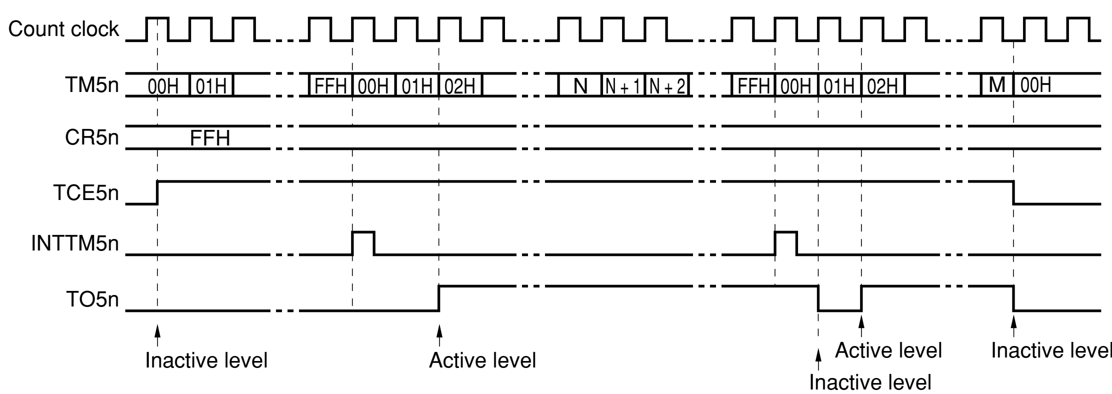
(a) Basic operation (active level = H)



(b) CR5n = 0



(c) CR5n = FFH

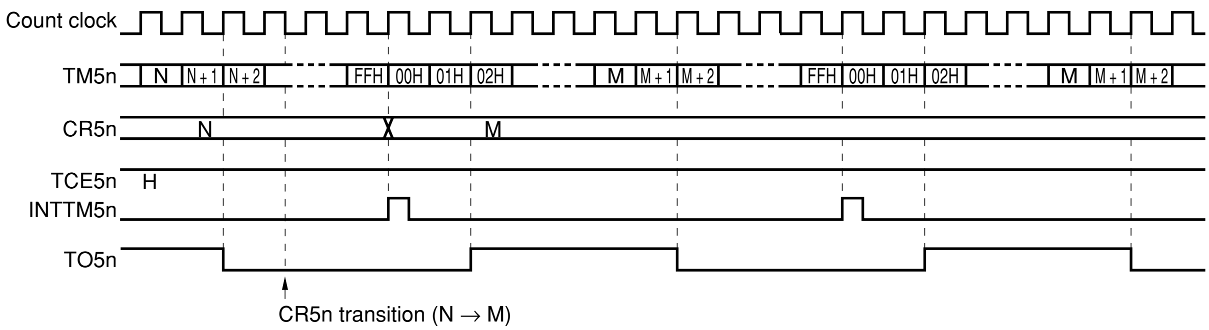


Remark n = 0, 1

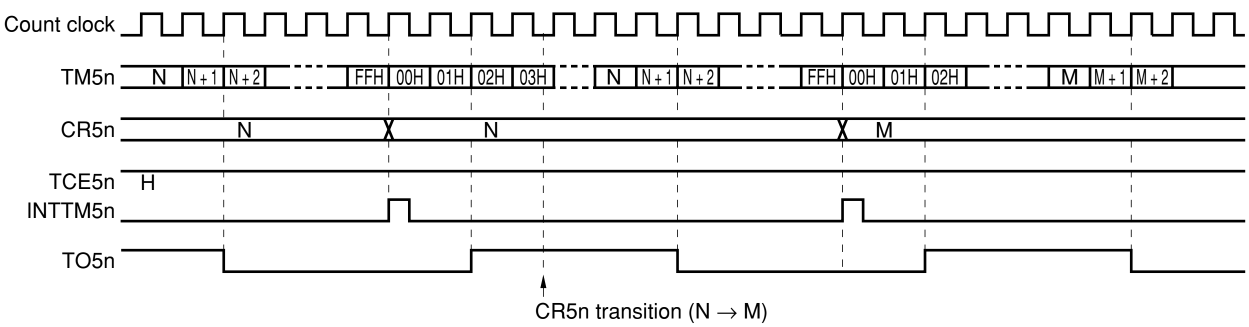
(2) Operated by CR5n transition

Figure 9-12. Timing of Operation by Change of CR5n

(a) CR5n value is changed from N to M when TM5n > CR5n



(b) CR5n value is changed from N to M when TM5n < CR5n



Remark n = 0, 1

9.4.5 Interval timer (16-bit) operations

When bit 4 (TMC514) of 8-bit timer mode control register 51 (TMC51) is set to 1, the 16-bit resolution timer/counter mode is entered.

The 8-bit timer/event counter operates as an interval timer that generates interrupt requests repeatedly at intervals of the count value preset to the 8-bit timer compare registers (CR50, CR51).

[Setting]

<1> Set each register

TCL50: Select count clock for TM50.
Cascade-connected TM51 need not be selected.

CR50, CR51: Compare value (each value can be set to 00H to FFH)

TMC50, TMC51: Select the clear & start mode entered on a match between TM50 and CR50 (TM51 and CR51).

TM50 → TMC50 = 0000×××0B ×: don't care

TM51 → TMC51 = 0001×××0B ×: don't care

<2> When TMC51 is set to TCE51 = 1 and then TMC50 is set to TCE50 = 1, the count operation starts.

<3> When the values of TM50 and CR50 of the cascade-connected timer match, INTTM50 of TM50 is generated (TM50 and TM51 are cleared to 00H).

<4> INTTM50 is generated repeatedly at the same interval.

Cautions 1. Stop the timer operation without fail before setting the compare registers (CR50, CR51).

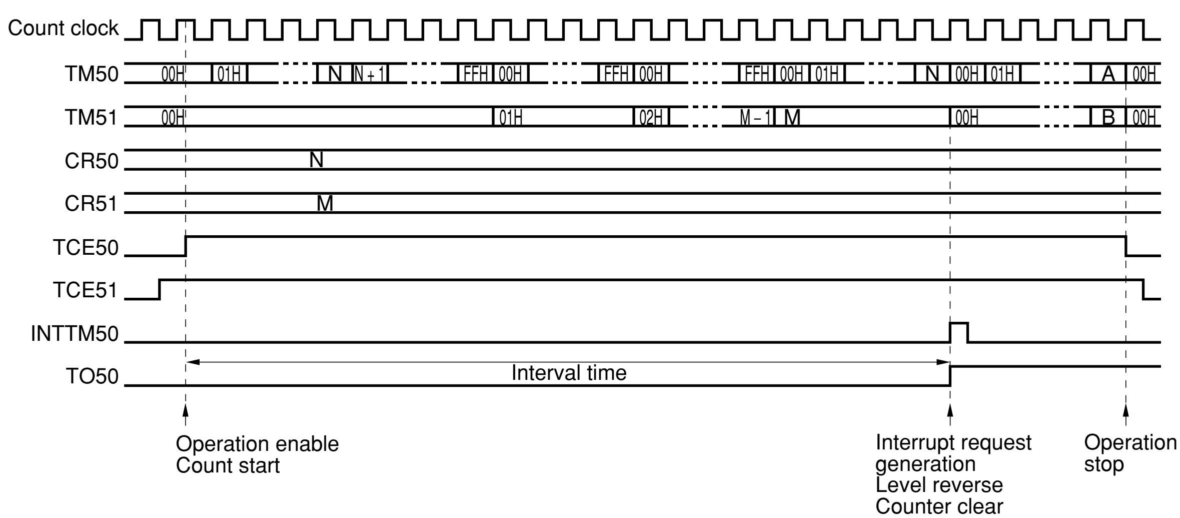
2. INTTM51 of TM51 is generated when the TM51 count value matches CR51, even if cascade connection is used. Be sure to mask TM51 to disable interrupts.

3. Set TCE50 and TCE51 in order of TM51 then TM50.

4. Count restart/stop can only be controlled by setting TCE50 of TM50 to 1/0.

Figure 9-13 shows an example of 16-bit resolution cascade connection mode timing.

Figure 9-13. 16-Bit Resolution Cascade Connection Mode



★ 9.5 Program List

Caution The following sample program is shown as an example to describe the operation of semiconductor products and their applications. Therefore, when applying the following information to your devices, design the devices after performing evaluation under your own responsibility.

9.5.1 Interval timer (8-bit)

```
/*************************************************************************************/
/*                                                                                   */
/*          Timer 50 operation sample                                                */
/*    Interval timer setting example (cycle change by interrupt servicing)           */
/*     data[0]: Data set flag (value changed when other than 00)                     */
/*     data[1]: Set data                                                             */
/*                                                                                   */
/*************************************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#pragma interrupt INTTM50 intervalint rb2
        unsigned char data[2];          /* Data area */

void main(void)
{
        PCC = 0x0;                      /* Set high-speed operation mode */
        data[0] = 0;                    /* Clear data area */
        data[1] = 0;

                                        /* Set port        */
        P7 = 0b11111011;                /* When using TO50 */
        PM7.2 = 0;                      /* Set P72 to output */

                                        /* Set interrupt                 */
        TMMK50 = 0;                     /* Clear INTTM50 interrupt mask */
                                        /* Set timer 50                       */
        TMC50 = 0b00000111;             /* Clear & start mode, initial value L */
        TCL50 = 0b00000101;             /* Both rising and falling edges */
                                        /* Count clock is fx/2^6 */
        CR50 = 131;                     /* Set interval to 1 ms as initial value */
        TCE50 = 1;                      /* Timer start */
        EI();
        while(1);                       /* Dummy loop */
}

/* INTTM50 interrupt function */
void intervalint()
{
        if(data[0] != 0)
        {
                CR50 = data[1];         /* Set new set value */
                data[0] = 0;            /* Clear request flag */
        }

}
```

9.5.2 External event counter

```
/***************************************************************/
/*                                                             */
/*           Timer 50 operation sample                         */
/*       Event counter setting example                         */
/*        data: Count up flag                                  */
/*                                                             */
/***************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#pragma interrupt INTTM50 intervalint rb2
        unsigned char data;             /* Data area */

void main(void)
{
        PCC = 0x0;                      /* Set high-speed operation mode */
        data = 0;                       /* Clear data area */

                                        /* Set port             */
        PM7.2 = 1;                      /* Set P72 to input */

                                        /* Set interrupt                  */
        TMMK50 = 0;                     /* Clear INTTM50 interrupt mask  */
                                        /* Set timer 50                      */
        TMC50 = 0b00000000;             /* Clear & start mode                  */
        TCL50 = 0b00000001;             /* Specify rising edge of TI50       */
        CR50 = 0x10;                    /* Set N = 16 as initial value     */
        TCE50 = 1;                      /* Timer start                         */
        EI();

/*************************************************************/
/*                                                           */
/*    Describe the processing to be executed                 */
/*                                                           */
/*************************************************************/

        while(data == 0);               /* Wait for count up */


/*************************************************************/
/*                                                           */
/*    Describe the processing after count up below           */
/*                                                           */
/*************************************************************/
}

/* INTTM50 interrupt function */
void intervalint()
{
        data = 0xff;                    /* Set count up flag   */
        TCE50 = 0;                      /* Timer stop          */
}
```

9.5.3 Interval timer (16-bit)

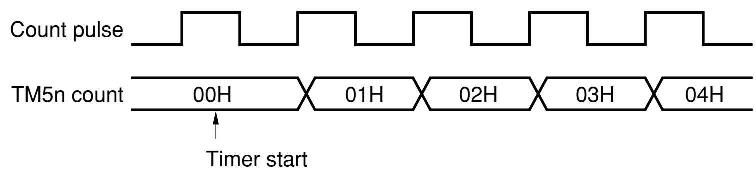
```
/***************************************************************/
/*                                                             */
/*              Timer 5 operation sample                       */
/*               Cascade connection setting example            */
/*                                                             */
/***************************************************************/
#pragma sfr
#pragma EI
#pragma DI
#define intervalTM5 130                 /* Cycle data to be set to CR */
#pragma interrupt INTTM50 ppgint rb2
        unsigned char ppgdata[2];       /* Data area to be set to timer 5 */

void main(void)
{
        int interval;
        interval = intervalTM5;
        PCC = 0x0;                      /* Select high-speed operation mode */
        ppgdata[0] = 0;                 /* Clear CR50 data */
        ppgdata[1] = 0;                 /* Clear CR51 data */
                                        /* Set port */
        P7 = 0b11111011;                /* Clear P72 */
        PM7.2 = 0;                      /* Set P72 to output */
                                        /* Set interrupt */
        TMMK50 = 0;                     /* Clear INTTM50 interrupt mask */
        TMMK51 = 1;                     /* Set INTTM51 interrupt mask */
                                        /* Set timer 5 */
        TCL50 = 0b00000101;             /* Count clock is fx/2^6 */
        CR50 = interval & 0xff;         /* Set lower compare register to CR50 */
        CR51 = interval >> 8;           /* Set higher compare register to CR51 */
        TMC50 = 0b00000111;             /* Inverted on match, initial value L */
        TMC51 = 0b00010000;             /* Cascade mode */
        TCE51 = 1;
        TCE50 = 1;                      /* Timer starts */
        EI();

        while(1);
}

/* Timer 5 interrupt function */
void ppgint()
{
        unsigned int work;
        work = ppgdata[0]+ppgdata[1]*0x100;
        if (work != 0)
        {
                TCE50 =0;
                CR51 = work >> 8;
                CR50 = work & 0xff;
                ppgdata[0] = 0;
                ppgdata[1] = 0;

                if (work != 0xffff)
                {
                        TCE50 = 1; /* Timer resumes */
                }
        }
}
```

9.6 Cautions Related to 8-Bit Timer/Event Counters 50, 51

(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 counter 5n (TM5n) is started asynchronously to the count pulse.

Figure 9-14. Start Timing of 8-Bit Timer Counter 5n (TM5n)



(2) Setting STOP mode or main system clock stop mode

Except when TI5n input is selected, always set TCE5n = 0 before setting the STOP mode or main system clock stop mode.

The timer may malfunction when the main system clock starts oscillating.

(3) TM5n (n = 0, 1) reading during timer operation

When reading TM5n during operation, the count clock stops temporarily, so select a count clock with a high/low-level waveform longer than two cycles of the CPU clock. For example, in the case where the CPU clock (f_{CPU}) is f_X, when the selected count clock is $f_X/4$ or below, it can be read.

Remark n = 0, 1

CHAPTER 10 WATCH TIMER

10.1 Watch Timer Functions

The watch timer has the following functions.

(1) Watch timer

When the main system clock or subsystem clock is used, interrupt requests (INTWT) are generated at $2^{14}/f_W$ second intervals.

(2) Interval timer

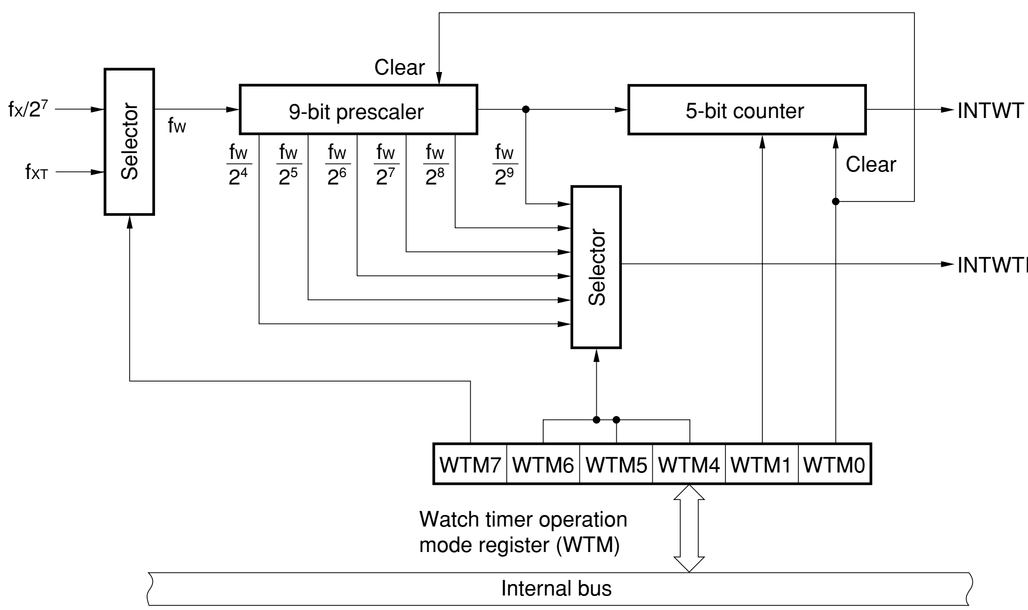
Interrupt requests (INTWTI) are generated at the preset time interval.

For the interval time, see **Table 10-2**.

The watch timer and the interval timer can be used simultaneously.

Figure 10-1 shows the watch timer block diagram.

Figure 10-1. Watch Timer Block Diagram



Remark f_W: Watch timer clock frequency ($f_X/2^7$ or f_{XT})

f_X: Main system clock oscillation frequency

f_{XT}: Subsystem clock oscillation frequency

10.2 Watch Timer Configuration

The watch timer consists of the following hardware.

Table 10-1. Watch Timer Configuration

Item	Configuration
Counter	5 bits × 1
Prescaler	9 bits × 1
Control register	Watch timer operation mode register (WTM)

10.3 Register to Control Watch Timer

The watch timer is controlled by the watch timer operation mode register (WTM).

- **Watch timer operation mode register (WTM)**
  This register sets the watch timer count clock, enables/disables operation, sets the prescaler interval time, and controls the 5-bit counter operation.
  WTM is set by a 1-bit or 8-bit memory manipulation instruction.
  $\overline{\text{RESET}}$ input clears WTM to 00H.

Figure 10-2. Format of Watch Timer Operation Mode Register (WTM)

Address: FF41H After reset: 00H R/W

Symbol	7	6	5	4	3	2	<1>	<0>
WTM	WTM7	WTM6	WTM5	WTM4	0	0	WTM1	WTM0

WTM7	Watch timer count clock selection
0	★ $f_X/2^7$ (65.4 kHz: f_X = 8.38 MHz, 93.7 kHz: f_X = 12 MHz[Note])
1	f_{XT} (32.768 kHz: f_{XT} = 32.768 kHz)

WTM6	WTM5	WTM4	Prescaler interval time selection
0	0	0	$2^4/f_W$
0	0	1	$2^5/f_W$
0	1	0	$2^6/f_W$
0	1	1	$2^7/f_W$
1	0	0	$2^8/f_W$
1	0	1	$2^9/f_W$
Other than above			Setting prohibited

WTM1	5-bit counter operation control
0	Clear after operation stop
1	Start

WTM0	Watch timer operation enable
0	Operation stopped (both prescaler and timer cleared)
1	Operation enabled

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Caution Do not change the count clock and interval time (by using bits 4 to 7 (WTM4 to WTM7) of WTM) while the watch timer is operating.

Remarks 1. f_W: Watch timer clock frequency ($f_X/2^7$ or f_{XT})

2. f_X: Main system clock oscillation frequency

3. f_{XT}: Subsystem clock oscillation frequency

10.4 Watch Timer Operations

10.4.1 Watch timer operation

The watch timer generates an interrupt request (INTWT) at specific time intervals ($2^{14}/f_W$ seconds) by using the main system clock or subsystem clock. The interrupt request is generated at the following time intervals:

- If main system clock (8.38 MHz) is selected: 0.25 seconds
- If subsystem clock (32.768 kHz) is selected: 0.5 seconds

When bit 0 (WTM0) and bit 1 (WTM1) of the watch timer operation mode register (WTM) are set to 1, the count operation starts, and when these bits are cleared to 0, the 5-bit counter is cleared and the count operation stops.

When the interval timer is simultaneously operated, a zero-second start can be achieved for the watch timer by setting WTM1 to 1 after clearing it to 0. In this case, however, the 9-bit prescaler is not cleared. Therefore, an error up to $2^9/f_W$ seconds occurs in the first overflow (INTWT) after the zero-second start.

Remark f_W: Watch timer clock frequency ($f_X/2^7$ or f_{XT})

10.4.2 Interval timer operation

The watch timer operates as interval timer that generates interrupt requests (INTWTI) repeatedly at an interval of the preset count value.

The interval time can be selected with bits 4 to 6 (WTM4 to WTM6) of the watch timer operation mode register (WTM).

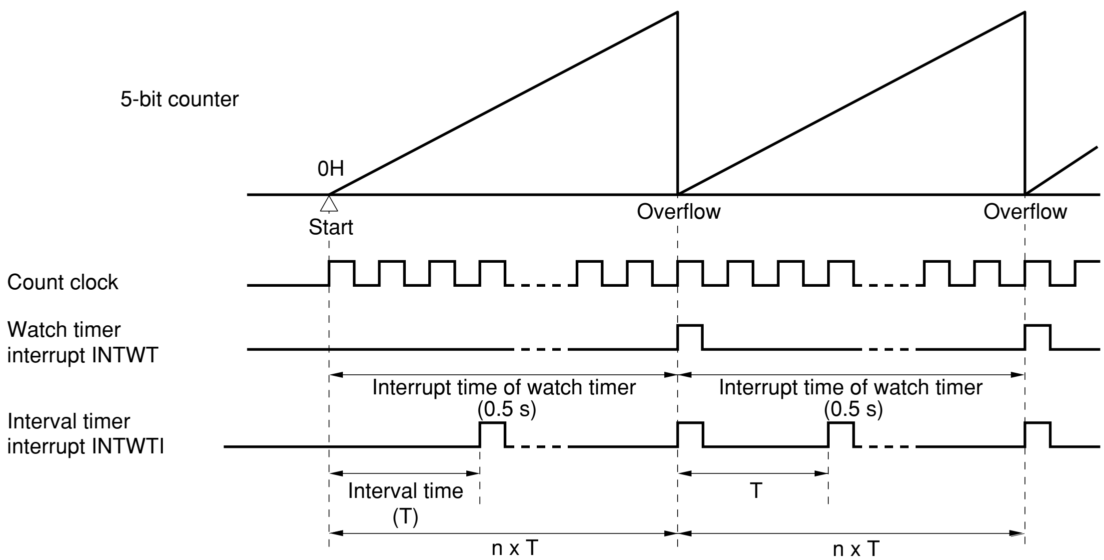
Table 10-2. Interval Timer Interval Time

WTM6	WTM5	WTM4	Interval Time	When Operated at f_X = 12 MHz[Note]	When Operated at f_X = 8.38 MHz	When Operated at f_X = 4.19 MHz	When Operated at f_{XT} = 32.768 kHz
0	0	0	$2^4/f_W$	170 μs	244 μs	488 μs	488 μs
0	0	1	$2^5/f_W$	341 μs	488 μs	977 μs	976 μs
0	1	0	$2^6/f_W$	682 μs	977 μs	1.95 ms	1.95 ms
0	1	1	$2^7/f_W$	1.36 ms	1.95 ms	3.91 ms	3.90 ms
1	0	0	$2^8/f_W$	2.73 ms	3.91 ms	7.82 ms	7.81 ms
1	0	1	$2^9/f_W$	5.46 ms	7.82 ms	15.6 ms	15.6 ms
Other than above			Setting prohibited				

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Remark f_W: Watch timer clock frequency ($f_X/2^7$ or f_{XT})
f_X: Main system clock oscillation frequency
f_{XT}: Subsystem clock oscillation frequency

Figure 10-3. Operation Timing of Watch Timer/Interval Timer



Caution **If the watch timer and 5-bit counter are enabled by the watch timer mode control register (WTM) (by setting bits 0 (WTM0) and 1 (WTM1) of WTM to 1), the time from this setting to the occurrence of the first interrupt request (INTWT) is not exactly the value set by bit 3 (WTM3) of WTM. This is because the 5-bit counter is late by one output cycle of the 9-bit prescaler in starting counting. The second INTWT signal and those that follow are generated exactly at the set time.**

Remark f_W: Watch timer clock frequency ($f_X/2^7$ or f_{XT})

n: The number of interval timer operations

Figures in parentheses are for operation with f_W = 32.768 kHz.

11.1 Watchdog Timer Functions

The watchdog timer has the following functions.

(1) Watchdog timer

The watchdog timer detects a program loop. Upon detection of a program loop, a non-maskable interrupt request or $\overline{\text{RESET}}$ can be generated.

For the loop detection time, see **Table 11-2**.

(2) Interval timer

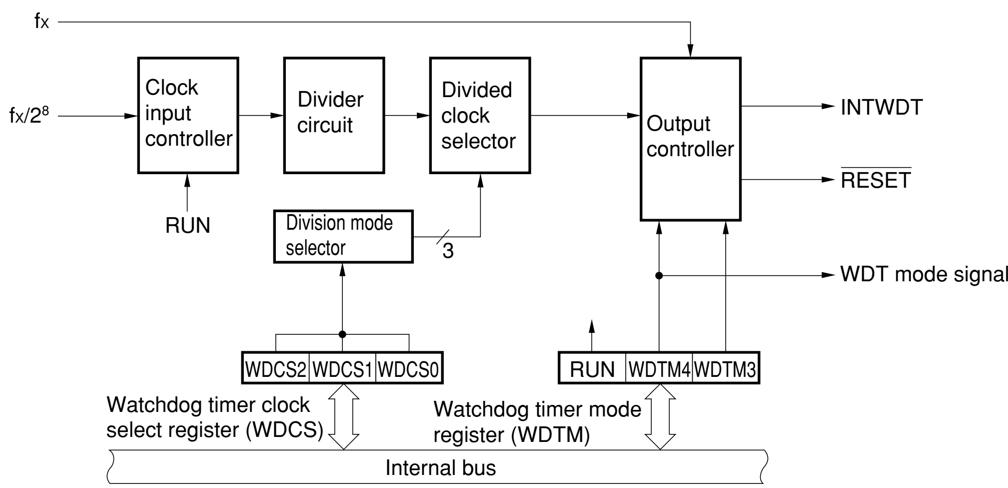
Interrupt requests are generated at the preset time intervals.

For the interval time, see **Table 11-3**.

Caution Select the watchdog timer mode or the interval timer mode using the watchdog timer mode register (WDTM). (The watchdog timer and the interval timer cannot be used simultaneously.)

Figure 11-1 shows a block diagram of the watchdog timer.

Figure 11-1. Watchdog Timer Block Diagram



11.2 Watchdog Timer Configuration

The watchdog timer consists of the following hardware.

Table 11-1. Watchdog Timer Configuration

Item	Configuration
Control registers	Watchdog timer clock select register (WDCS) Watchdog timer mode register (WDTM)

11.3 Registers to Control Watchdog Timer

The following two registers are used to control the watchdog timer.

- Watchdog timer clock select register (WDCS)
- Watchdog timer mode register (WDTM)

(1) Watchdog timer clock select register (WDCS)

This register sets the overflow time of the watchdog timer and the interval timer.

WDCS is set by an 8-bit memory manipulation instruction.

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

Figure 11-2. Format of Watchdog Timer Clock Select Register (WDCS)

Address: FF42H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
WDCS	0	0	0	0	0	WDCS2	WDCS1	WDCS0

★

WDCS2	WDCS1	WDCS0	Overflow time of watchdog timer/interval timer		
				f_X = 8.38 MHz	f_X = 12 MHz[Note]
0	0	0	$2^{12}/f_X$	488 μs	341 μs
0	0	1	$2^{13}/f_X$	977 μs	682 μs
0	1	0	$2^{14}/f_X$	1.95 ms	1.36 ms
0	1	1	$2^{15}/f_X$	3.91 ms	2.73 ms
1	0	0	$2^{16}/f_X$	7.82 ms	5.46 ms
1	0	1	$2^{17}/f_X$	15.6 ms	10.9 ms
1	1	0	$2^{18}/f_X$	31.2 ms	21.8 ms
1	1	1	$2^{20}/f_X$	125 ms	87.3 ms

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Remark f_X: Main system clock oscillation frequency

(2) Watchdog timer mode register (WDTM)

This register sets the watchdog timer operating mode and enables/disables counting.

WDTM is set by 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 (WDTM)

Address: FFF9H After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
WDTM	RUN	0	0	WDTM4	WDTM3	0	0	0

RUN	Watchdog timer operation mode selection[Note 1]
0	Count stop
1	Counter is cleared and counting starts

WDTM4	WDTM3	Watchdog timer operation mode selection[Note 2]
0	×	Interval timer mode[Note 3] (Maskable interrupt request occurs upon generation of overflow)
1	0	Watchdog timer mode 1 (Non-maskable interrupt request occurs upon generation of overflow)
1	1	Watchdog timer mode 2 (Reset operation is activated upon generation of overflow)

Notes 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.

2. Once set to 1, WDTM3 and WDTM4 cannot be cleared to 0 by software.

3. The watchdog timer starts operation as an interval timer when RUN is set to 1.

Caution When RUN is set to 1 so that the watchdog timer is cleared, the actual overflow time is up to 2^8/fx seconds shorter than the time set by the watchdog timer clock select register (WDCS).

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 is operated to detect a program loop.

The loop detection time interval is selected with bits 0 to 2 (WDCS0 to WDCS2) of the watchdog timer clock select register (WDCS). 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 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 loop detection time is exceeded, system reset or a non-maskable interrupt request is generated according to the value of WDTM bit 3 (WDTM3).

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 up to $2^8/f_X$ seconds.

2. When the subsystem clock is selected for the CPU clock, the watchdog timer count operation is stopped.

Table 11-2. Watchdog Timer Loop Detection Time

Loop Detection Time	When Operated at f_X = 8.38 MHz	When Operated at f_X = 12 MHz**Note**
$2^{12}/f_X$	488 μs	341 μs
$2^{13}/f_X$	977 μs	682 μs
$2^{14}/f_X$	1.95 ms	1.36 ms
$2^{15}/f_X$	3.91 ms	2.73 ms
$2^{16}/f_X$	7.82 ms	5.46 ms
$2^{17}/f_X$	15.6 ms	10.9 ms
$2^{18}/f_X$	31.2 ms	21.8 ms
$2^{20}/f_X$	125 ms	87.3 ms

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Remark f_X: Main system clock oscillation frequency

11.4.2 Interval timer operation

The watchdog timer operates as an interval timer that 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 interval time of the interval timer is selected with bits 0 to 2 (WDCS0 to WDCS2) of the watchdog timer clock select register (WDCS). When bit 7 (RUN) of WDTM is set to 1, the watchdog timer operates as an interval timer.

When the watchdog timer operates as an interval timer, the interrupt mask flag (WDTMK) and priority specification flag (WDTPR) are validated and the maskable interrupt request (INTWDT) can be generated. Among the maskable interrupts, INTWDT has the highest priority at default.

The interval timer continues operating in the HALT mode but it stops in STOP mode. Thus, set RUN 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 (this selects the watchdog timer mode), the interval timer mode is not set unless $\overline{\text{RESET}}$ is input.

2. The interval time just after setting WDTM may be shorter than the set time by up to $2^8/f_X$ seconds.

3. When the subsystem clock is selected for the CPU clock, the watchdog timer count operation is stopped.

Table 11-3. Interval Timer Interval Time

Interval Time	When Operated at f_X = 8.38 MHz	When Operated at f_X = 12 MHz^Note
$2^{12}/f_X$	488 μs	341 μs
$2^{13}/f_X$	977 μs	682 μs
$2^{14}/f_X$	1.95 ms	1.36 ms
$2^{15}/f_X$	3.91 ms	2.73 ms
$2^{16}/f_X$	7.82 ms	5.46 ms
$2^{17}/f_X$	15.6 ms	10.9 ms
$2^{18}/f_X$	31.2 ms	21.8 ms
$2^{20}/f_X$	125 ms	87.3 ms

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Remark f_X: Main system clock oscillation frequency

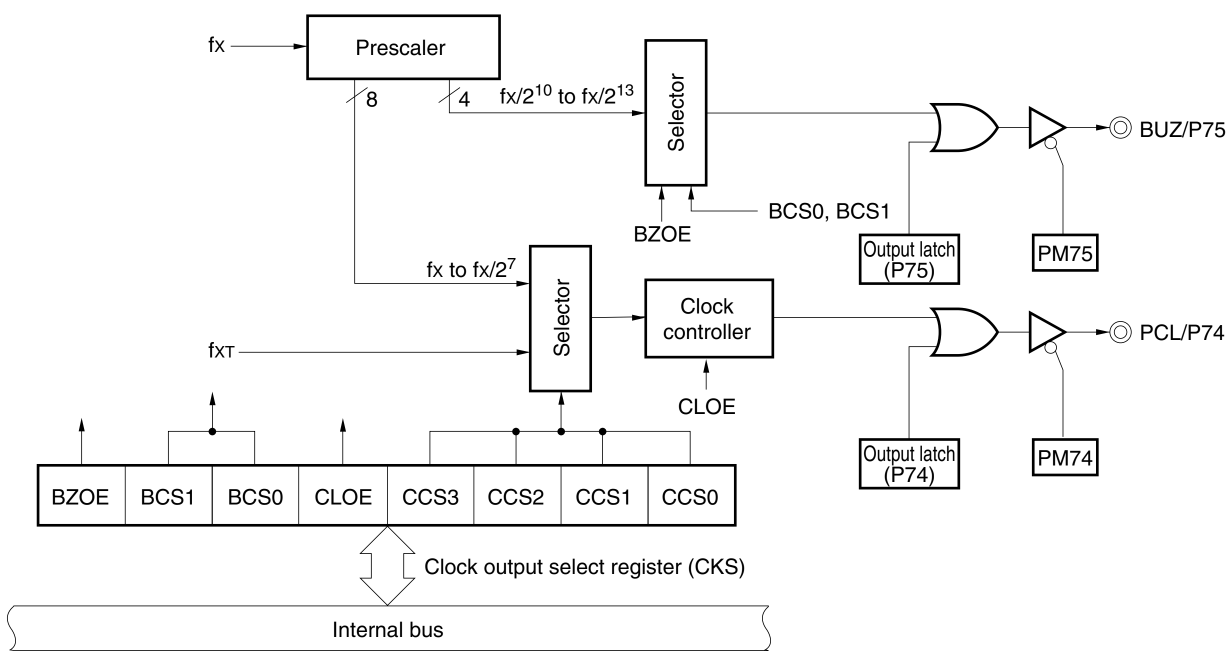
12.1 Clock Output/Buzzer Output Controller Functions

Clock output is used for carrier output during remote controlled transmission and clock output for supply to peripheral ICs. The clock selected by the clock output select register (CKS) is output.

In addition, buzzer output is used for square-wave output of the buzzer frequency selected by CKS.

Figure 12-1 shows the block diagram of the clock output/buzzer output controller.

★ **Figure 12-1. Block Diagram of Clock Output/Buzzer Output Controller**



12.2 Configuration of Clock Output/Buzzer Output Controller

The clock output/buzzer output controller consists of the following hardware.

Table 12-1. Configuration of Clock Output/Buzzer Output Controller

Item	Configuration
Control registers	Clock output select register (CKS) Port mode register (PM7) Port 7 (P7)

12.3 Register to Control Clock Output/Buzzer Output Controller

The following three registers are used to control the clock output/buzzer output controller.

- Clock output select register (CKS)
- Port mode register (PM7)
- Port 7 (P7)

(1) Clock output select register (CKS)

This register sets output enable/disable for clock output (PCL) and for the buzzer frequency output (BUZ), and sets the output clock.

CKS is set by a 1-bit or 8-bit memory manipulation instruction.

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

Figure 12-2. Format of Clock Output Select Register (CKS)

Address: FF40H After reset: 00H R/W

Symbol	<7>	6	5	<4>	3	2	1	0
CKS	BZOE	BCS1	BCS0	CLOE	CCS3	CCS2	CCS1	CCS0

BZOE	BUZ output enable/disable specification
0	Stop clock divider operation. BUZ fixed to low level.
1	Enable clock divider operation. BUZ output enabled.

★

BCS1	BCS0	BUZ output clock selection		
			f_X = 8.38 MHz	f_X = 12 MHz[Note]
0	0	$f_X/2^{10}$	8.18 kHz	11.7 kHz
0	1	$f_X/2^{11}$	4.09 kHz	5.85 kHz
1	0	$f_X/2^{12}$	2.04 kHz	2.92 kHz
1	1	$f_X/2^{13}$	1.02 kHz	1.46 kHz

CLOE	PCL output enable/disable specification
0	Stop clock divider operation. PCL fixed to low level.
1	Enable clock divider operation. PCL output enabled.

★

CCS3	CCS2	CCS1	CCS0	PCL output clock selection		
					f_X = 8.38 MHz	f_X = 12 MHz[Note]
0	0	0	0	f_X	8.38 MHz	12 MHz
0	0	0	1	$f_X/2$	4.19 MHz	6 MHz
0	0	1	0	$f_X/2^2$	2.09 MHz	3 MHz
0	0	1	1	$f_X/2^3$	1.04 MHz	1.5 MHz
0	1	0	0	$f_X/2^4$	523 kHz	750 kHz
0	1	0	1	$f_X/2^5$	261 kHz	375 kHz
0	1	1	0	$f_X/2^6$	130 kHz	187 kHz
0	1	1	1	$f_X/2^7$	65.4 kHz	93.7 kHz
1	0	0	0	f_{XT} (32.768 kHz)		
Other than above				Setting prohibited		

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

3. Figures in parentheses are for operation with f_{XT} = 32.768 kHz.

(2) Port mode register (PM7)

This register sets port 7 input/output in 1-bit units.

When using the P74/PCL pin for clock output and the P75/BUZ pin for buzzer output, clear PM74, PM75 and the output latches of P74 and P75 to 0.

PM7 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM7 to FFH.

Figure 12-3. Format of Port Mode Register 7 (PM7)

Address: FF27H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM7	1	1	PM75	PM74	PM73	PM72	PM71	PM70

PM7n	P7n pin I/O mode selection (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

12.4 Operation of Clock Output/Buzzer Output Controller

12.4.1 Operation as clock output

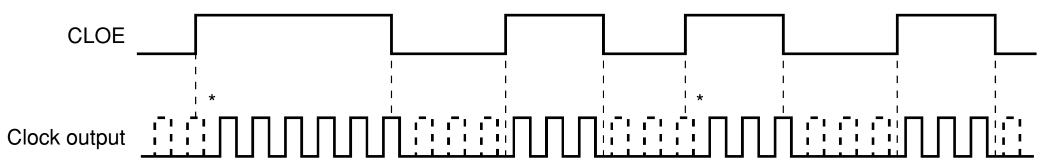
The clock pulse is output using the following procedure.

<1> Select the clock pulse output frequency using bits 0 to 3 (CCS0 to CCS3) of the clock output select register (CKS) (clock pulse output in disabled state).

<2> Set bit 4 (CLOE) of CKS to 1, and enable clock output.

Remark The clock output controller is designed not to output pulses with a small width during output enable/ disable switching of the clock output. As shown in Figure 12-4, be sure to start output from the low period of the clock (marked with ***** in the figure). When stopping output, do so after securing the high level of the clock.

Figure 12-4. Remote Control Output Application Example



12.4.2 Operation as buzzer output

The buzzer frequency is output using the following procedure.

<1> Select the buzzer output frequency using bits 5 and 6 (BCS0, BCS1) of the clock output select register (CKS) (buzzer output in disabled state).

<2> Set bit 7 (BZOE) of CKS to 1 to enable buzzer output.

CHAPTER 13 8-BIT A/D CONVERTER (μPD780024A, 780024AY SUBSERIES)

13.1 A/D Converter Functions

A/D converter is an 8-bit resolution converter that converts analog inputs into digital values. It can control up to 8 analog input channels (ANI0 to ANI7).

(1) Hardware start

Conversion is started by trigger input (ADTRG: rising edge, falling edge, or both rising and falling edges can be specified).

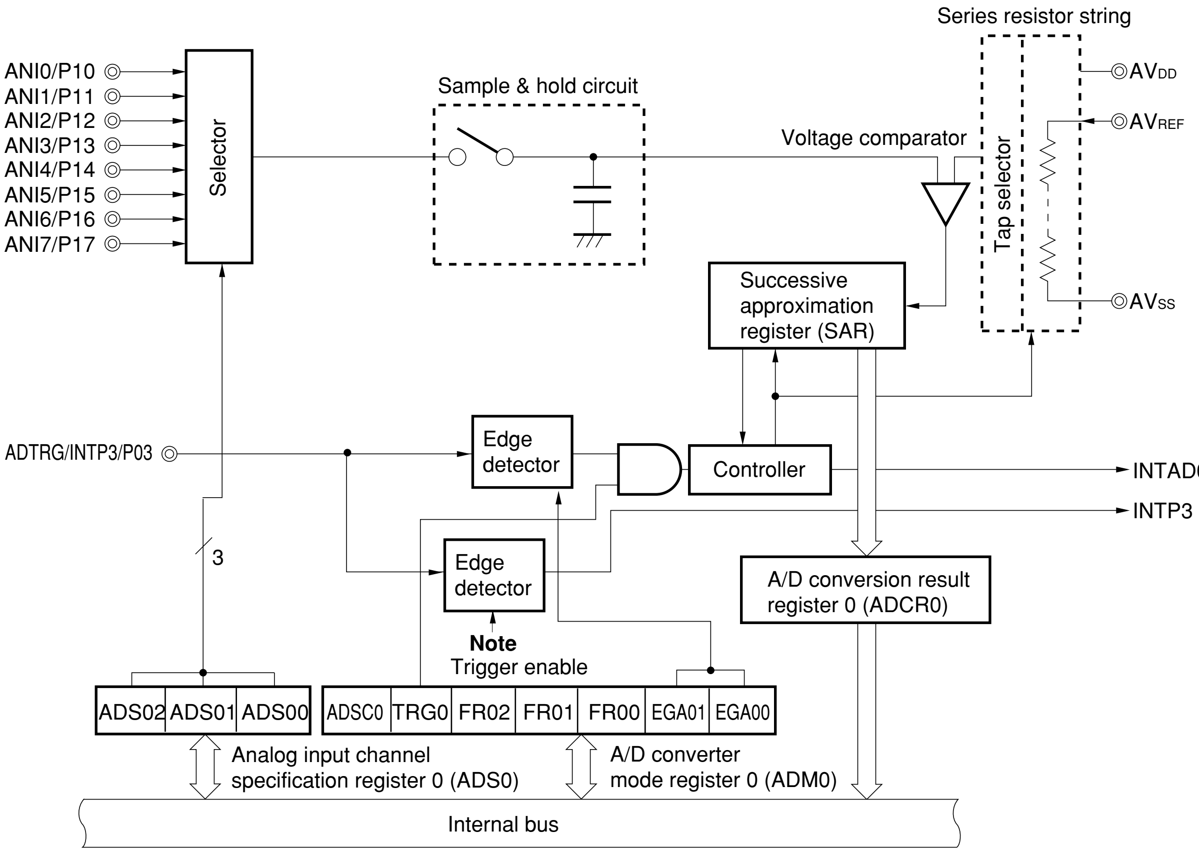
(2) Software start

Conversion is started by setting A/D converter mode register 0 (ADM0).

Select one channel for analog input from ANI0 to ANI7 to perform A/D conversion. In the case of hardware start, A/D conversion stops when an A/D conversion operation ends and an interrupt request (INTAD0) is generated. In the case of software start, A/D conversion is repeated. Each time an A/D conversion operation ends, an interrupt request (INTAD0) is generated.

Caution Although the μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY incorporate a 10-bit A/D converter, this converter can be operated as an 8-bit A/D converter by using the device file DF780024.

Figure 13-1. 8-Bit A/D Converter Block Diagram



Note The valid edge of external interrupt is specified by bit 3 of the EGP and EGN registers (see **Figure 19-5 Format of External Interrupt Rising Edge Enable Register (EGP), External Interrupt Falling Edge Enable Register (EGN)**).

13.2 A/D Converter Configuration

The A/D converter consists of the following hardware.

Table 13-1. A/D Converter Configuration

Item	Configuration
Analog input	8 channels (ANI0 to ANI7)
Hardware trigger input	1 (ADTRG)
Registers	Successive approximation register (SAR) A/D conversion result register 0 (ADCR0)
Control registers	A/D converter mode register 0 (ADM0) Analog input channel specification register 0 (ADS0)

(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 A/D conversion result register 0 (ADCR0).

(2) A/D conversion result register 0 (ADCR0)

The ADCR0 is an 8-bit register that stores the A/D conversion result. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register.

ADCR0 is read by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADCR0 to 00H.

Caution When writing is performed to A/D converter mode register 0 (ADM0) and analog input channel specification register 0 (ADS0), the contents of ADCR0 may become undefined. Read the conversion result following conversion completion before writing to ADM0, ADS0. Using a timing other than the above may cause an incorrect conversion result to be read.

(3) Sample & hold circuit

★ The sample & hold circuit samples the input signal of the analog input pin selected by the selector when A/D conversion is started and holds the sampled analog input voltage value during A/D conversion.

(4) Voltage comparator

★ The voltage comparator compares the sampled analog input voltage to the series resistor string output voltage.

(5) Series resistor string

The series resistor string is connected between AV_{REF} and AV_{SS}, and generates a voltage to be compared to the analog input.

(6) ANI0 to ANI7 pins

These are eight analog input pins to input analog signals to undergo A/D conversion to the A/D converter. ANI0 to ANI7 are alternate-function pins that can also be used for digital input.

Cautions 1. Use ANI0 to ANI7 input voltages within the specification range. If a voltage higher than AV_{REF} or lower than AV_{SS} is applied (even if within the absolute maximum rating range), the conversion value of that channel will be undefined and the conversion values of other channels may also be affected.

2. Analog input (ANI0 to ANI7) pins are alternate function pins that can also be used as input port (P10 to P17) pins. When A/D conversion is performed by selecting any one of ANI0 to ANI7, do not access port 1 during conversion. It may cause the lower conversion resolution.

3. When a digital pulse is applied to a pin adjacent to the pin in the process of A/D conversion, A/D conversion values may not be obtained as expected due to coupling noise. Thus, do not apply any pulse to a pin adjacent to the pin in the process of A/D conversion.

(7) AV_{REF} 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_{REF} and AV_{SS}.

Caution A series resistor string of several 10 kΩ is connected between the AV_{REF} pin and AV_{SS} pin. Therefore, when the output impedance of the reference voltage is too high, it seems as if the AV_{REF} pin and the series resistor string are connected in series. This may cause a greater reference voltage error.

(8) AV_{SS} pin

This is the ground potential pin of the A/D converter. Always keep it at the same potential as the V_{SS0} or V_{SS1} pin even when not using the A/D converter.

(9) AV_{DD} pin

This is the A/D converter analog power supply pin. Always keep it at the same potential as the V_{DD0} or V_{DD1} pin even when not using the A/D converter.

★ **(10) ADTRG pin**

This pin is a pin used to start the A/D converter by hardware.

13.3 Registers to Control A/D Converter

The following 2 types of registers are used to control the A/D converter.

- A/D converter mode register 0 (ADM0)
- Analog input channel specification register 0 (ADS0)

(1) A/D converter mode register 0 (ADM0)

This register sets the conversion time for analog input to be A/D converted, conversion start/stop, and external trigger.

ADM0 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADM0 to 00H.

Figure 13-2. Format of A/D Converter Mode Register 0 (ADM0)

Address: FF80H After reset: 00H R/W

Symbol	<7>	<6>	5	4	3	2	1	0
ADM0	ADCS0	TRG0	FR02	FR01	FR00	EGA01	EGA00	0

ADCS0	A/D conversion operation control
0	Stop conversion operation.
1	Enable conversion operation.

TRG0	Software start/hardware start selection
0	Software start
1	Hardware start

★

FR02	FR01	FR00	Conversion time selection[Note 1]		
				f_X = 8.38 MHz	f_X = 12 MHz[Note 2]
0	0	0	144/f_X	17.1 μs	12.0 μs
0	0	1	120/f_X	14.3 μs	10.0 μs[Note 4]
0	1	0	96/f_X	11.4 μs[Note 3]	8.0 μs[Note 4]
1	0	0	72/f_X	8.5 μs[Note 3]	6.0 μs[Note 4]
1	0	1	60/f_X	7.1 μs[Note 3]	5.0 μs[Note 4]
1	1	0	48/f_X	5.7 μs[Note 3]	4.0 μs[Note 4]
Other than above			Setting prohibited		

EGA01	EGA00	External trigger signal, edge specification
0	0	No edge detection
0	1	Falling edge detection
1	0	Rising edge detection
1	1	Both falling and rising edge detection

Notes 1. Set the A/D conversion time as follows.
- When operated at f_X = 12 MHz (V_{DD} = 4.5 to 5.5 V): 12 μs or more
- When operated at f_X = 8.38 MHz (V_{DD} = 4.0 to 5.5 V): 14 μs or more

2. Expanded-specification products of μPD780024A Subseries only.

3. Setting is prohibited because the A/D conversion time is less than 14 μs.

4. Setting is prohibited because the A/D conversion time is less than 12 μs.

Caution When rewriting FR00 to FR02 to other than the same data, stop A/D conversion operations once prior to performing rewrite.

Remark f_X: Main system clock oscillation frequency

(2) Analog input channel specification register 0 (ADS0)

This register specifies the analog voltage input port for A/D conversion.

ADS0 is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADS0 to 00H.

Figure 13-3. Format of Analog Input Channel Specification Register 0 (ADS0)

Address: FF81H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ADS0	0	0	0	0	0	ADS02	ADS01	ADS00

ADS02	ADS01	ADS00	Analog input channel specification
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

13.4 A/D Converter Operations

13.4.1 Basic operations of A/D converter

<1> Select one channel for A/D conversion with analog input channel specification register 0 (ADS0).
<2> The voltage input to the selected analog input channel is sampled by the sample & hold circuit.
<3> When sampling has been done for a certain time, the sample & hold circuit is placed in the hold state and the input analog voltage is held until the A/D conversion operation is ended.
<4> Bit 7 of the successive approximation register (SAR) is set. The series resistor string voltage tap is set to (1/2) AV_{REF} by the tap selector.
<5> 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_{REF}, the MSB of SAR remains set. If the analog input is smaller than (1/2) AV_{REF}, the MSB is reset.
<6> Next, bit 6 of SAR is automatically set, and the operation proceeds to the next comparison. 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_{REF}
- Bit 7 = 0: (1/4) AV_{REF}

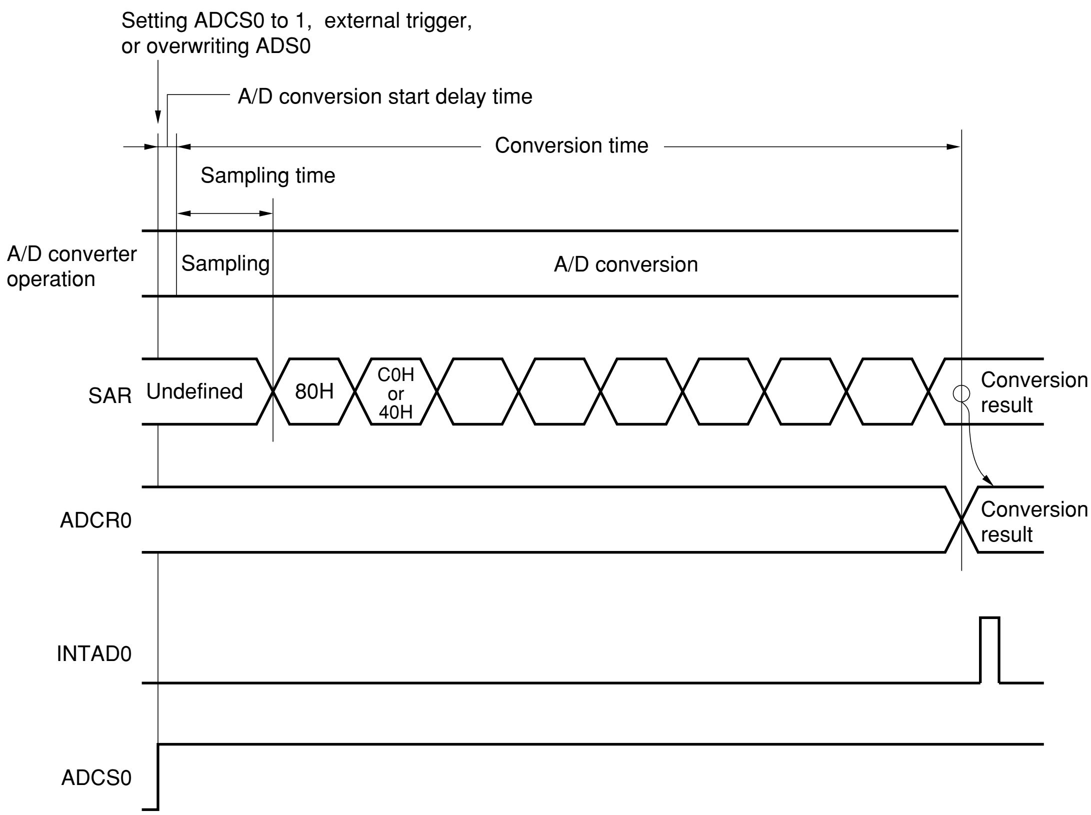
The voltage tap and analog input voltage are compared and bit 6 of SAR is manipulated as follows.
- Analog input voltage ≥ Voltage tap: Bit 6 = 1
- Analog input voltage < Voltage tap: Bit 6 = 0

<7> Comparison is continued in this way up to bit 0 of SAR.
<8> Upon completion of the comparison of 8 bits, an effective digital result value remains in SAR, and the result value is transferred to and latched in A/D conversion result register 0 (ADCR0).
At the same time, the A/D conversion end interrupt request (INTAD0) can also be generated.

★ **Cautions 1. The first A/D conversion value immediately after A/D conversion has been started may not satisfy the rated value. Take measures such as polling the A/D conversion end interrupt request (INTAD0) and removing the first conversion results.**

★ **2. The A/D converter stops operation in standby mode.**

★ **Figure 13-4. Basic Operation of 8-Bit A/D Converter**



A/D conversion operations are performed continuously until bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) is reset (0) by software.

$\overline{\text{RESET}}$ input clears A/D conversion result register 0 (ADCR0) to 00H.

★ Confirm the conversion results by referring to the A/D conversion end interrupt request flag (ADIF0).

The sampling time of the A/D converter varies depending on the values set in A/D converter mode register 0 (ADM0). There is a delay time from when the A/D converter is enabled for operation until sampling is actually performed.

For the sets in which a strict A/D conversion time is required, note the contents described in Table 13-2.

★ **Table 13-2. Sampling Time and A/D Conversion Start Delay Time of A/D Converter**

FR02	FR01	FR00	Conversion Time[Note 1]	Sampling Time	A/D Conversion Start Delay Time	
					MIN.	MAX.
0	0	0	$144/f_X$	$20/f_X$	$0.5/f_{CPU} + 6/f_X$	$0.5/f_{CPU} + 8/f_X$
0	0	1	$120/f_X$	$16/f_X$		
0	1	0	$96/f_X$	$12/f_X$		
1	0	0	$72/f_X$	$10/f_X$	$0.5/f_{CPU} + 3/f_X$	$0.5/f_{CPU} + 4/f_X$
1	0	1	$60/f_X$	$8/f_X$		
1	1	0	$48/f_X$	$6/f_X$		
Other than above			Setting prohibited	–	–	–

Notes 1. Set the A/D conversion time as follows.

- When operated at f_X = 12 MHz[Note 2] (V_{DD} = 4.5 to 5.5 V): 12 μs or more
- When operated at f_X = 8.38 MHz (V_{DD} = 4.0 to 5.5 V): 14 μs or more

2. Expanded-specification products of μPD780024A Subseries only.

Remark f_X: Main system clock oscillation frequency

f_{CPU}: CPU clock frequency

13.4.2 Input voltage and conversion results

The relationship between the analog input voltage input to the analog input pins (ANI0 to ANI7) and the theoretical A/D conversion result (stored in A/D conversion result register 0 (ADCR0)) is shown by the following expression.

$$ADCR0 = INT\left(\frac{V_{IN}}{AV_{REF}} \times 256 + 0.5\right)$$

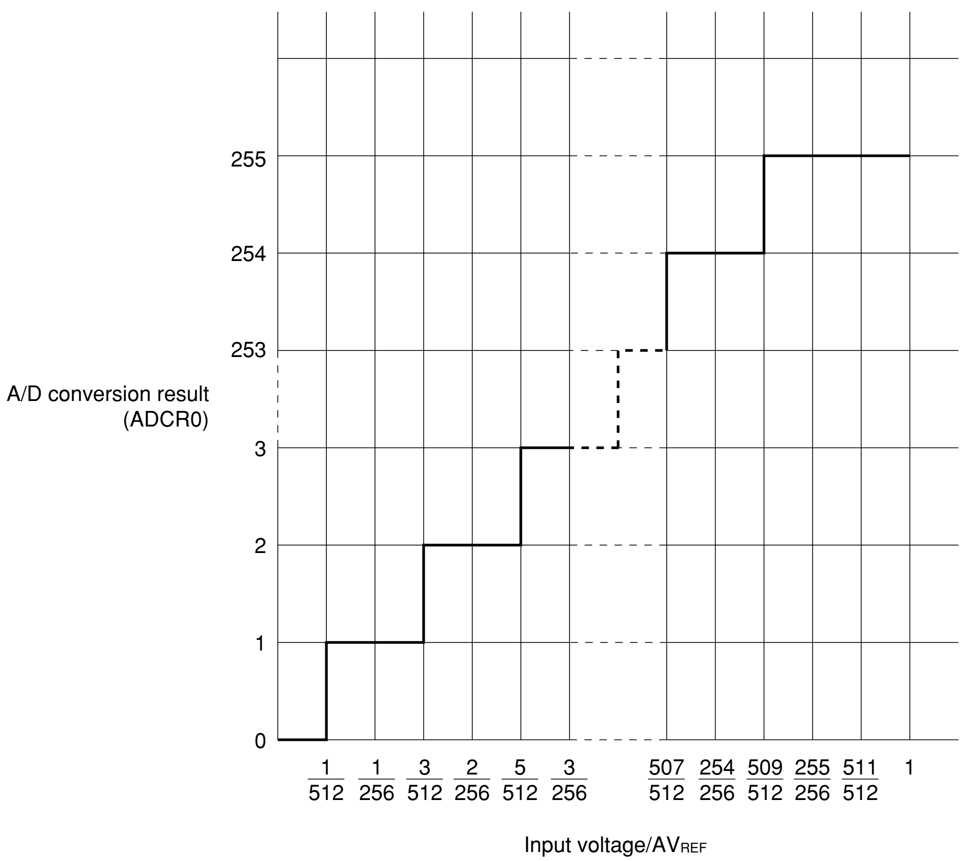
or

$$(ADCR0 - 0.5) \times \frac{AV_{REF}}{256} \leq V_{IN} < (ADCR0 + 0.5) \times \frac{AV_{REF}}{256}$$

where, INT( ): Function which returns integer part of value in parentheses
V_{IN}: Analog input voltage
AV_{REF}: AV_{REF} pin voltage
ADCR0: A/D conversion result register 0 (ADCR0) value

Figure 13-5 shows the relationship between the analog input voltage and the A/D conversion result.

Figure 13-5. Relationship Between Analog Input Voltage and A/D Conversion Result



13.4.3 A/D converter operation mode

Select one analog input channel from among ANI0 to ANI7 using analog input channel specification register 0 (ADS0) to start A/D conversion.

A/D conversion can be started in either of the following two ways.

- Hardware start: Conversion is started by trigger input (rising edge, falling edge, or both rising and falling edges enabled).
- Software start: Conversion is started by setting A/D converter mode register 0 (ADM0).

★ When A/D conversion is complete, the interrupt request signal (INTAD0) is generated.

(1) A/D conversion by hardware start

When bit 6 (TRG0) and bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) are set to 1, the A/D conversion standby state is set. When the external trigger signal (ADTRG) is input, A/D conversion of the voltage applied to the analog input pin specified by analog input channel specification register 0 (ADS0) starts.

Upon the end of A/D conversion, the conversion result is stored in A/D conversion result register 0 (ADCR0), and the interrupt request signal (INTAD0) is generated. After one A/D conversion operation is started and finished, the next conversion operation is not started until a new external trigger signal is input.

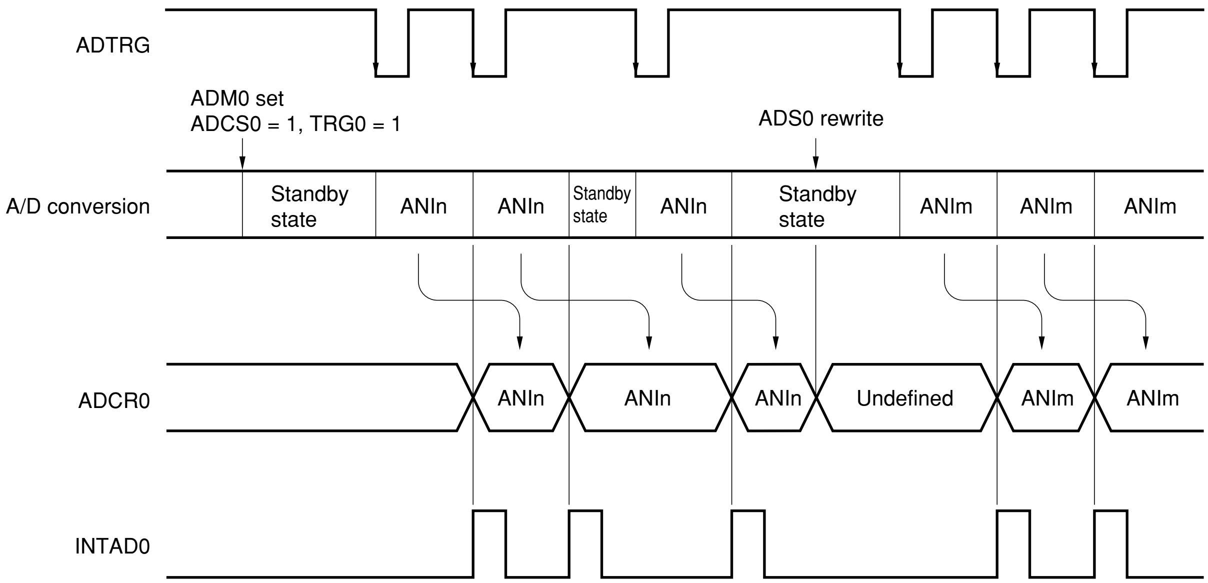
If ADS0 is rewritten during A/D conversion, the converter suspends A/D conversion and waits for a new external trigger signal to be input. When the external trigger input signal is reinput, A/D conversion is carried out from the beginning. If ADS0 is rewritten during A/D conversion standby, A/D conversion starts when the following external trigger input signal is input.

★ If 1 is written to ADCS0 again during A/D conversion, the A/D conversion in progress is discontinued and a new A/D conversion is started when the next external trigger input signal is input.

If 0 is written to ADCS0 during A/D conversion, the A/D conversion operation stops immediately.

Caution When P03/INTP3/ADTRG is used as the external trigger input (ADTRG), specify the valid edge using bits 1, 2 (EGA00, EGA01) of A/D converter mode register 0 (ADM0) and set the interrupt mask flag (PMK3) to 1.

★ **Figure 13-6. A/D Conversion by Hardware Start (When Falling Edge Is Specified)**



Remarks 1. n = 0, 1, ..., 7

2. m = 0, 1, ..., 7

(2) A/D conversion by software start

When bit 6 (TRG0) and bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) are set to 0 and 1, respectively, A/D conversion of the voltage applied to the analog input pin specified by analog input channel specification register 0 (ADS0) starts.

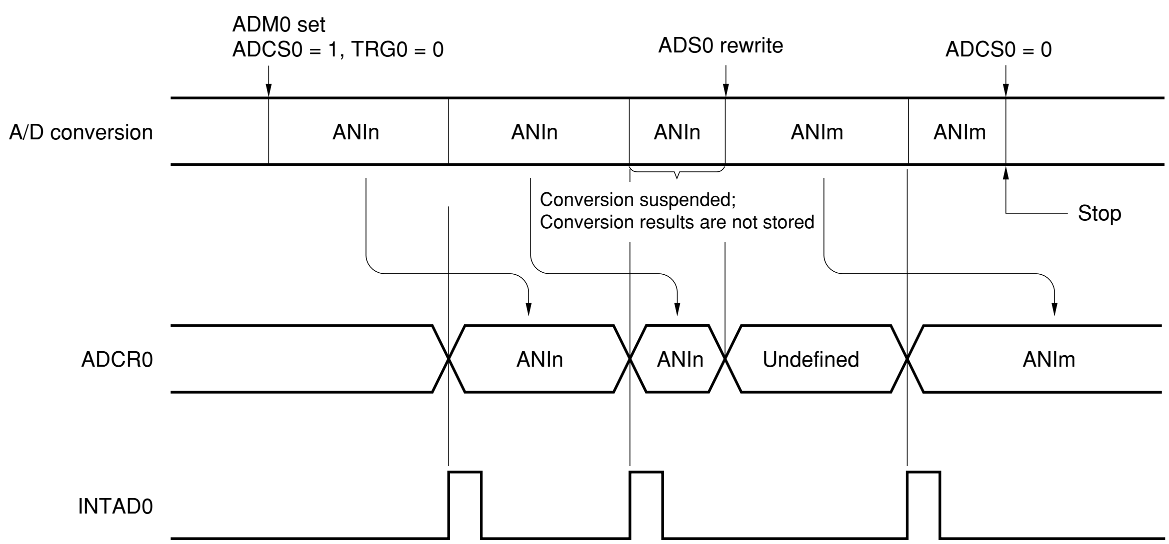
Upon the end of the A/D conversion, the conversion result is stored in A/D conversion result register 0 (ADCR0), and the interrupt request signal (INTAD0) is generated. After one A/D conversion operation is started and ended, the next conversion operation is immediately started. A/D conversion operations are repeated until new data is written to ADS0.

If ADS0 is rewritten during A/D conversion, the converter suspends A/D conversion and A/D conversion of the newly selected analog input channel is started.

If 1 is written to ADCS0 again during A/D conversion, the A/D conversion in progress is discontinued and a new A/D conversion is started.

If 0 is written to ADCS0 during A/D conversion, the A/D conversion operation stops immediately.

★ **Figure 13-7. A/D Conversion by Software Start**



Remarks 1. n = 0, 1, ..., 7

2. m = 0, 1, ..., 7

13.5 How to Read A/D Converter Characteristics Table

Here we will explain the special terms unique to A/D converters.

(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 error, full scale error, integral linearity error, differential linearity error and errors which are combinations of these express overall error.

Furthermore, quantization error is not included in overall error in the characteristics table.

(3) Quantization error

When analog values are converted to digital values, there naturally occurs a ±1/2LSB error. In an A/D converter, an analog input voltage in a range of ±1/2LSB are converted to the same digital code, so a quantization error cannot be avoided.

Furthermore, it is not included in the overall error, zero scale error, full scale error, integral linearity error, and differential linearity error in the characteristics table.

Figure 13-8. Overall Error

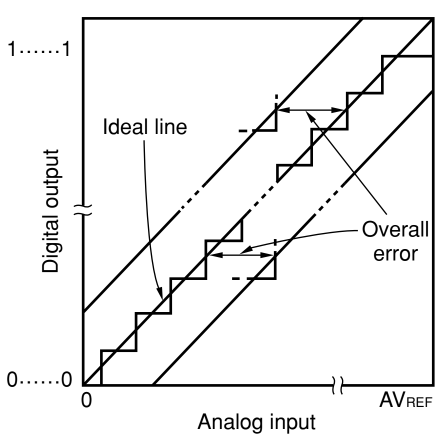
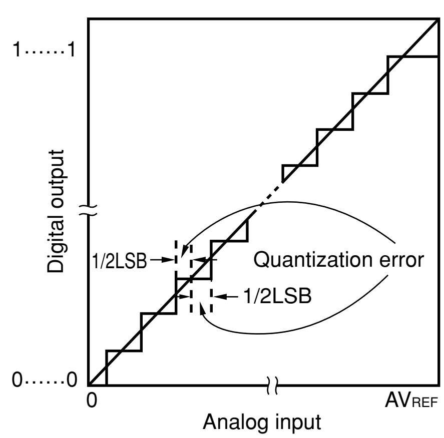


Figure 13-9. Quantization Error



(4) Zero scale error

This shows the difference between the actual measured value of the analog input voltage and the theoretical value (1/2LSB) when the digital output changes from 0……000 to 0……001. If the actual measured value is greater than the theoretical value, it shows the difference between the actual measured value of the analog input voltage and the theoretical value (3/2LSB) when the digital output changes from 0……001 to 0……010.

(5) Full scale error

This shows the difference between the actual measured value of the analog input voltage and the theoretical value (full scale –3/2LSB) when the digital output changes from 1……110 to 1……111.

(6) Integral linearity error

This shows the degree to which the conversion characteristics deviate from the ideal linear relationship. It expresses the maximum value of the difference between the actual measured value and the ideal straight line when the zero scale error and full scale error are 0.

(7) Differential linearity error

Although the ideal output width for a given code is 1LSB, this value shows the difference between the actual measured value and the ideal value of the width when outputting a particular code.

Figure 13-10. Zero Scale Error

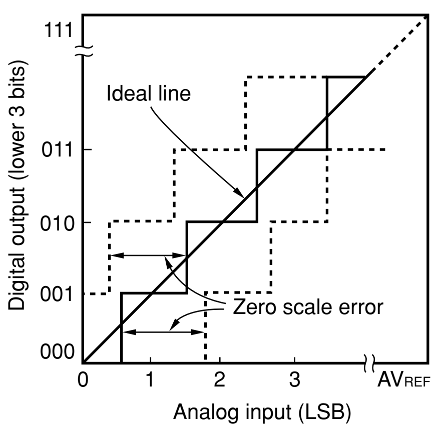


Figure 13-11. Full Scale Error

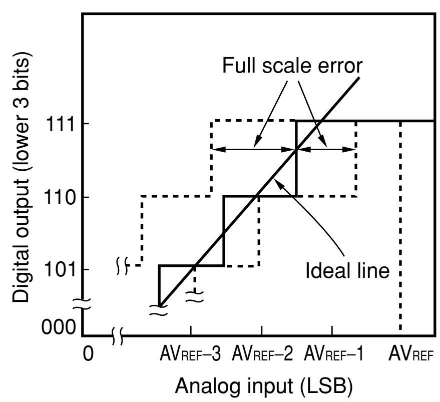


Figure 13-12. Integral Linearity Error

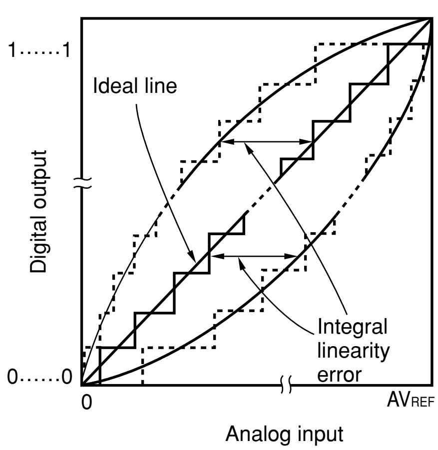
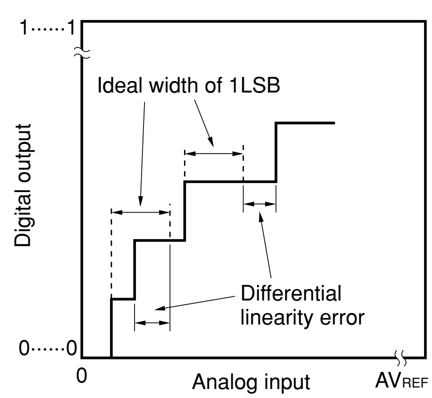


Figure 13-13. Differential Linearity Error

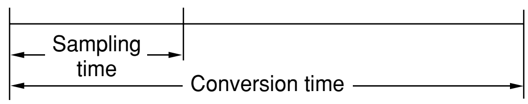


(8) Conversion time

This expresses the time from when the sampling was started to the time when the digital output was obtained. Sampling time is included in the conversion time in the characteristics table.

(9) Sampling time

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.

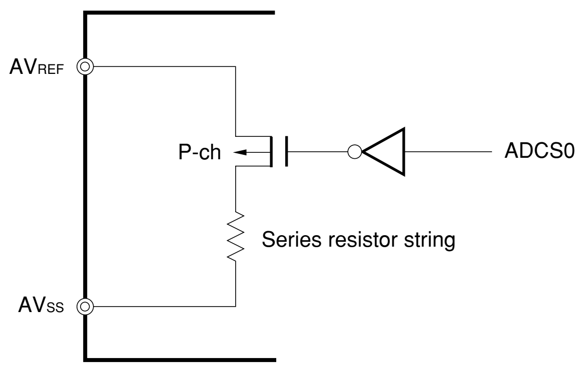


13.6 A/D Converter Cautions

(1) Power consumption in standby mode

A/D converter stops operating in the standby mode. At this time, power consumption can be reduced by stopping the conversion operation (by clearing bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) to 0).
Figure 13-14 shows the circuit configuration of a series resistor string.

Figure 13-14. Circuit Configuration of Series Resistor String



(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 higher than AV_{REF} or lower than AV_{SS} is input (even if within the absolute maximum rating range), the conversion value of that channel will be undefined and the conversion values of other channels may also be affected.

(3) Contending operations

<1> Contention between A/D conversion result register 0 (ADCR0) write and ADCR0 read by instruction upon the end of conversion
ADCR0 read is given priority. After the read operation, the new conversion result is written to ADCR0.

<2> Contention between ADCR0 write and external trigger signal input upon the end of conversion
The external trigger signal is not accepted during A/D conversion. Therefore, the external trigger signal is not accepted during ADCR0 write.

<3> Contention between ADCR0 write and A/D converter mode register 0 (ADM0) write or analog input channel specification register 0 (ADS0) write upon the end of conversion
ADM0 or ADS0 write is given priority. ADCR0 write is not performed, nor is the conversion end interrupt request signal (INTAD0) generated.

(4) ANI0/P10 to ANI7/P17

<1> The analog input pins (ANI0 to ANI7) also function as input port pins (P10 to P17).
When A/D conversion is performed with any of pins ANI0 to ANI7 selected, do not access port 1 while conversion is in progress, as this may reduce the conversion resolution.

<2> If digital pulses are applied to the pin adjacent to a 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 the pin adjacent to a pin undergoing A/D conversion.

(5) AVREF pin input impedance

A series resistor string is connected between the AVREF pin and the AVSS pin.

Therefore, when the output impedance of the reference voltage is too high, it seems as if the AVREF pin and the series resistor string are connected in series. This may cause a greater reference voltage error.

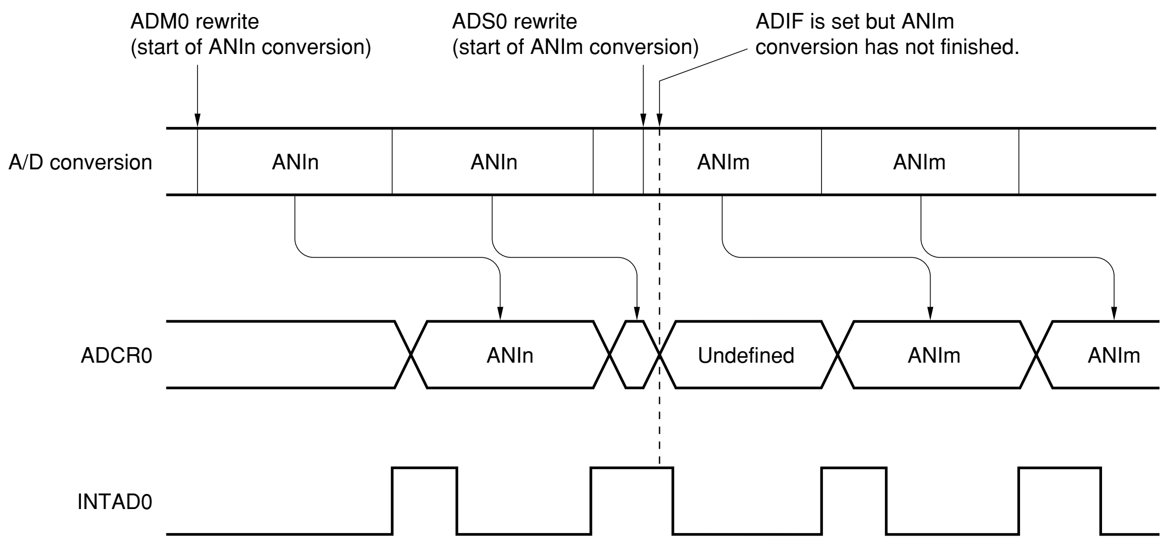
(6) Interrupt request flag (ADIF0)

The interrupt request flag (ADIF0) is not cleared even if analog input channel specification register 0 (ADS0) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and conversion end interrupt request flag for the pre-change analog input may be set just before the ADS0 rewrite. Caution is therefore required since, at this time, when ADIF0 is read immediately just after the ADS0 rewrite, ADIF0 is set despite the fact that the A/D conversion for the post-change analog input has not finished.

When A/D conversion is restarted after it is stopped, clear ADIF0 before restart.

★ **Figure 13-15. A/D Conversion End Interrupt Request Generation Timing**



Remarks 1. n = 0, 1, ..., 7

2. m = 0, 1, ..., 7

(7) Conversion results just after A/D conversion start

The A/D conversion value immediately after A/D conversion has been started may not satisfy the rated value. Take measures such as polling the A/D conversion end interrupt request (INTAD0) and removing the first conversion results.

(8) A/D conversion result register 0 (ADCR0) read operation

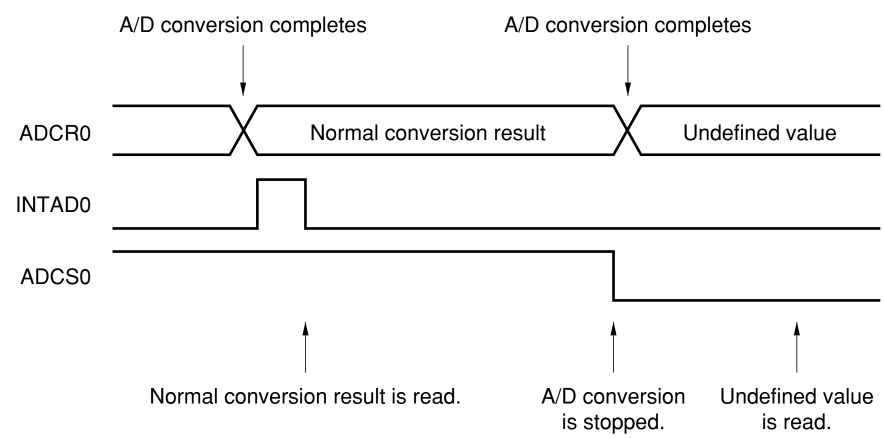
When A/D converter mode register 0 (ADM0) and analog input channel specification register 0 (ADS0) are written, the contents of ADCR0 may become undefined. Read the conversion result following conversion completion before writing to ADM0 and ADS0. Using a timing other than the above may cause an incorrect conversion result to be read.

(9) 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 before stopping the A/D conversion operation.

Figure 13-16 shows the timing of reading the conversion result.

Figure 13-16. Timing of Reading Conversion Result (When Conversion Result Is Undefined)



(10) 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.

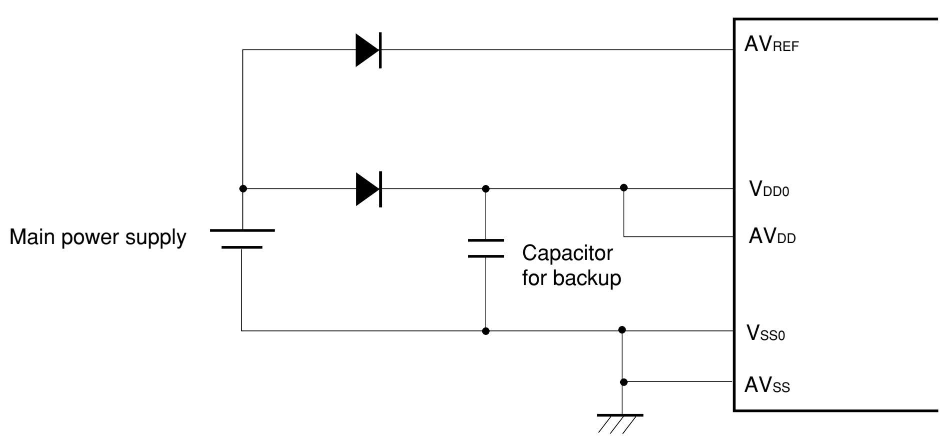
Connect AV_{SS0} and V_{SS0} at one location on the board where the voltages are stable.

(11) AV_{DD} pin

The AV_{DD} pin is the analog circuit power supply pin. It supplies power to the input circuits of the ANI0 to ANI7 pins.

Therefore, be sure to apply the same potential as V_{DD0} to this pin even for applications designed to switch to a backup battery for power supply.

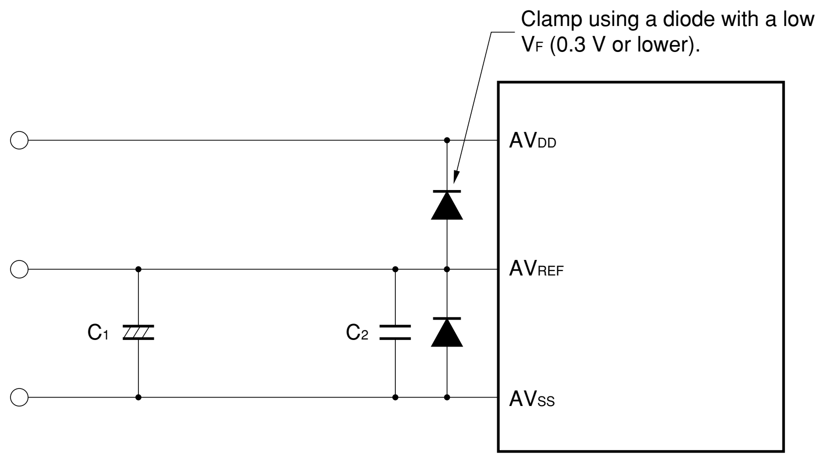
Figure 13-17. AV_{DD} Pin Connection



(12) AV$_{REF}$ pin

Connect a capacitor to the AV$_{REF}$ 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$_{REF}$ pin becomes unstable, causing the accuracy of the A/D conversion to drop. To prevent this, also connect a capacitor to the AV$_{REF}$ pin.
Figure 13-18 shows an example of connecting a capacitor.

★ **Figure 13-18. Example of Connecting Capacitor to AV$_{REF}$ Pin**

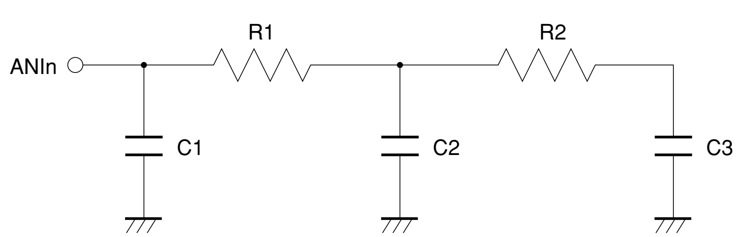


Remark C1: 4.7 μF to 10 μF (reference value)
C2: 0.01 μF to 0.1 μF (reference value)
Connect C2 as close to the pin as possible.

(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 13-19 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 pins ANI0 to ANI7. An example of this is shown in Figure 13-20. In this case, however, the microcontroller cannot follow an analog signal with a high differential coefficient because a lowpass 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 13-19. Internal Equivalent Circuit of Pins ANI0 to ANI7



Remark n = 0 to 7

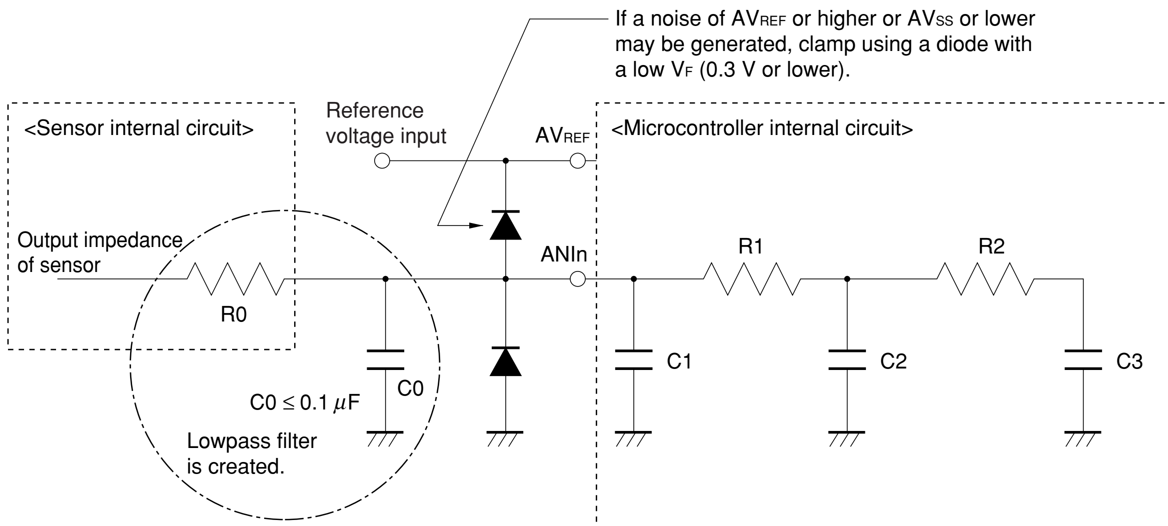
★ **Table 13-3. Resistances and Capacitances of Equivalent Circuit (Reference Values)**

(TYP.)

AV_{REF}	R1	R2	C1	C2	C3
2.7 V	12 kΩ	8.0 kΩ	3.0 pF	3.0 pF	2.0 pF
4.5 V	4 kΩ	2.7 kΩ	3.0 pF	1.4 pF	2.0 pF

Caution The resistances and capacitances in Table 13-3 are not guaranteed values.

★ **Figure 13-20. Example of Connection If Signal Source Impedance Is High**



Remark n = 0 to 7

★ **(14) Input impedance of ANI0 to ANI7 pins**

This A/D converter executes sampling by charging the internal sampling capacitor for approximately 1/10 of the conversion time.

Therefore, only the leakage current flows during other than sampling, and the current for charging the capacitor flows during sampling. The input impedance therefore varies and has no meaning.

To achieve sufficient sampling, it is recommended that the output impedance of the analog input source be 10 kΩ or less, or attach a capacitor of around 100 pF to the ANI0 to ANI7 pins (see **Figure 13-20**).

14.1 A/D Converter Functions

A/D converter is a 10-bit resolution converter that converts analog inputs into digital signals. It can control up to 8 analog input channels (ANI0 to ANI7).

(1) Hardware start

Conversion is started by trigger input (ADTRG: rising edge, falling edge, or both rising and falling edges can be specified).

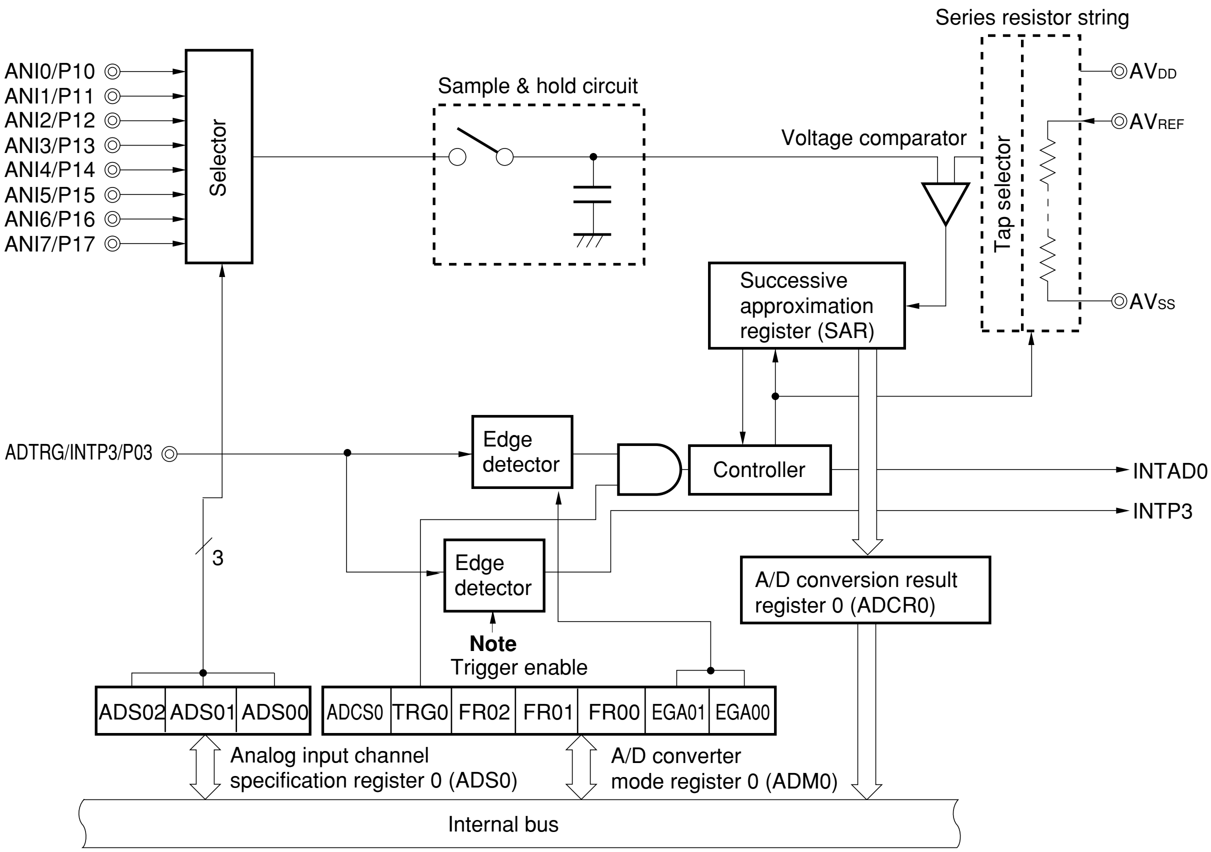
(2) Software start

Conversion is started by setting A/D converter mode register 0 (ADM0).

Select one channel for analog input from ANI0 to ANI7 to start A/D conversion. In the case of hardware start, the A/D converter stops when A/D conversion is completed, and an interrupt request (INTAD0) is generated. In the case of software start, A/D conversion is repeated. Each time as A/D conversion operation ends, an interrupt request (INTAD0) is generated.

Caution Although the μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY incorporate a 10-bit A/D converter, this converter can be operated as an 8-bit A/D converter by using the device file DF780024.

Figure 14-1. 10-Bit A/D Converter Block Diagram



Note The valid edge of external interrupt is specified by bit 3 of the EGP and EGN registers (see **Figure 19-5 Format of External Interrupt Rising Edge Enable Register (EGP), External Interrupt Falling Edge Enable Register (EGN)**).

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 input	8 channels (ANI0 to ANI7)
Hardware trigger input	1 (ADTRG)
Registers	Successive approximation register (SAR) A/D conversion result register 0 (ADCR0)
Control registers	A/D converter mode register 0 (ADM0) Analog input channel specification register 0 (ADS0)

(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 A/D conversion result register 0 (ADCR0).

(2) A/D conversion result register 0 (ADCR0)

This is a 16-bit register that stores the A/D conversion results. The lower 6 bits are fixed to 0. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR) and held ★ by this register. The most significant bit (MSB) is stored in ADCR0 first. The higher 8 bits of the conversion results are stored in FF17H. The lower 2 bits of the conversion results are stored in FF16H.

ADCR0 is read by a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADCR0 to 0000H.

★ **Figure 14-2. Format of A/D Conversion Result Register 0 (ADCR0)**

Address: FF16H, FF17H After reset: 0000H R

Symbol	FF17H								FF16H							
ADCR0											0	0	0	0	0	0

Caution When A/D converter mode register 0 (ADM0) and analog input channel specification register 0 (ADS0) are written, the contents of ADCR0 may become undefined. Read the conversion result following conversion completion before writing to ADM0 and ADS0. Using a timing other than the above may cause an incorrect conversion result to be read.

(3) Sample & hold circuit

★ The sample & hold circuit samples the input signal of the analog input pin selected by the selector when A/D conversion is started and holds the sampled analog input voltage value during A/D conversion.

(4) Voltage comparator

★ The voltage comparator compares the sampled analog input voltage to the series resistor string output voltage.

(5) Series resistor string

The series resistor string is connected between AV_{REF} and AV_{SS}, and generates the voltage to be compared to the analog input.

(6) ANI0 to ANI7 pins

These are eight analog input pins used to input analog signals to undergo A/D conversion to the A/D converter. ANI0 to ANI7 are alternate-function pins that can also be used for digital input.

Cautions 1. Use ANI0 to ANI7 input voltages within the specification range. If a voltage higher than or equal to AV_{REF} or lower than or equal to AV_{SS} is applied (even if within the absolute maximum rating range), the conversion value of that or equal to channel will be undefined and the conversion values of other channels may also be affected.

2. Analog input (ANI0 to ANI7) pins are alternate-function pins that can also be used as input port pins (P10 to P17). When A/D conversion is performed by selecting any one of ANI0 through ANI7, do not access port 1 during conversion, as this may cause a lower conversion resolution.

3. When a digital pulse is applied to a pin adjacent to the pin in the process of A/D conversion, A/D conversion values may not be obtained as expected due to coupling noise. Thus, do not apply a pulse to a pin adjacent to the pin in the process of A/D conversion.

(7) AV_{REF} pin

This is the A/D converter reference voltage pin and also the analog power supply pin.

It converts signals input to ANI0 to ANI7 into digital signals according to the voltage applied between AV_{REF} and AV_{SS}.

Caution A series resistor string is connected between the AV_{REF} and AV_{SS} pins. Therefore, when the output impedance of the reference voltage is too high, it seems as if the AV_{REF} pin and the series resistor string are connected in series. This may cause a greater reference voltage error.

(8) AV_{SS} pin

★ This is the ground potential pin of the A/D converter. Always keep it at the same potential as the V_{SS0} or V_{SS1} pin even when not using the A/D converter.

(9) AV_{DD} pin

This is the A/D converter analog power supply pin. Always keep it at the same potential as the V_{DD0} or V_{DD1} pin even when not using the A/D converter.

★ **(10) ADTRG pin**

This pin is a pin used to start the A/D converter by hardware.

14.3 Registers to Control A/D Converter

The following 2 types of registers are used to control the A/D converter.

- A/D converter mode register 0 (ADM0)
- Analog input channel specification register 0 (ADS0)

(1) A/D converter mode register 0 (ADM0)

This register sets the conversion time for analog input to be A/D converted, conversion start/stop, and external trigger.

ADM0 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADM0 to 00H.

Figure 14-3. Format of A/D Converter Mode Register 0 (ADM0)

Address: FF80H After reset: 00H R/W

Symbol	<7>	<6>	5	4	3	2	1	0
ADM0	ADCS0	TRG0	FR02	FR01	FR00	EGA01	EGA00	0

ADCS0	A/D conversion operation control
0	Stop conversion operation.
1	Enable conversion operation.

TRG0	Software start/hardware start selection
0	Software start
1	Hardware start

★

FR02	FR01	FR00	Conversion time selection[Note 1]		
				f_X = 8.38 MHz	f_X = 12 MHz[Note 2]
0	0	0	144/f_X	17.1 μs	12.0 μs
0	0	1	120/f_X	14.3 μs	10.0 μs[Note 4]
0	1	0	96/f_X	11.4 μs[Note 3]	8.0 μs[Note 4]
1	0	0	72/f_X	8.5 μs[Note 3]	6.0 μs[Note 4]
1	0	1	60/f_X	7.1 μs[Note 3]	5.0 μs[Note 4]
1	1	0	48/f_X	5.7 μs[Note 3]	4.0 μs[Note 4]
Other than above			Setting prohibited		

EGA01	EGA00	External trigger signal, edge specification
0	0	No edge detection
0	1	Falling edge detection
1	0	Rising edge detection
1	1	Both falling and rising edge detection

Notes 1. Set the A/D conversion time as follows.
- When operated at f_X = 12 MHz (V_{DD} = 4.5 to 5.5 V): 12 μs or more
- When operated at f_X = 8.38 MHz (V_{DD} = 4.0 to 5.5 V): 14 μs or more

2. Expanded-specification products of μPD780034A Subseries only.

3. Setting is prohibited because the A/D conversion time is less than 14 μs.

4. Setting is prohibited because the A/D conversion time is less than 12 μs.

Caution When rewriting FR00 to FR02 to other than the same data, stop A/D conversion operations once prior to performing rewrite.

Remark f_X: Main system clock oscillation frequency

(2) Analog input channel specification register 0 (ADS0)

This register specifies the analog voltage input port for A/D conversion.

ADS0 is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADS0 to 00H.

Figure 14-4. Format of Analog Input Channel Specification Register 0 (ADS0)

Address: FF81H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ADS0	0	0	0	0	0	ADS02	ADS01	ADS00

ADS02	ADS01	ADS00	Analog input channel specification
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

14.4 A/D Converter Operation

14.4.1 Basic operations of A/D converter

<1> Select one channel for A/D conversion with analog input channel specification register 0 (ADS0).

<2> The voltage input to the selected analog input channel is sampled by the sample & hold circuit.

<3> When sampling has been done for a certain time, the sample & hold circuit is placed in the hold state and the input analog voltage is held until the A/D conversion operation is ended.

<4> Bit 9 of the successive approximation register (SAR) is set. The series resistor string voltage tap is set to (1/2) AV_{REF} by the tap selector.

<5> 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_{REF}, the MSB of SAR remains set. If the analog input is smaller than (1/2) AV_{REF}, the MSB is reset.

<6> Next, bit 8 of SAR is automatically set, and the operation proceeds to the next comparison. The series resistor string voltage tap is selected according to the preset value of bit 9, as described below.

- Bit 9 = 1: (3/4) AV_{REF}
- Bit 9 = 0: (1/4) AV_{REF}

The voltage tap and analog input voltage are compared and bit 8 of SAR is manipulated as follows.

- Analog input voltage ≥ Voltage tap: Bit 8 = 1
- Analog input voltage < Voltage tap: Bit 8 = 0

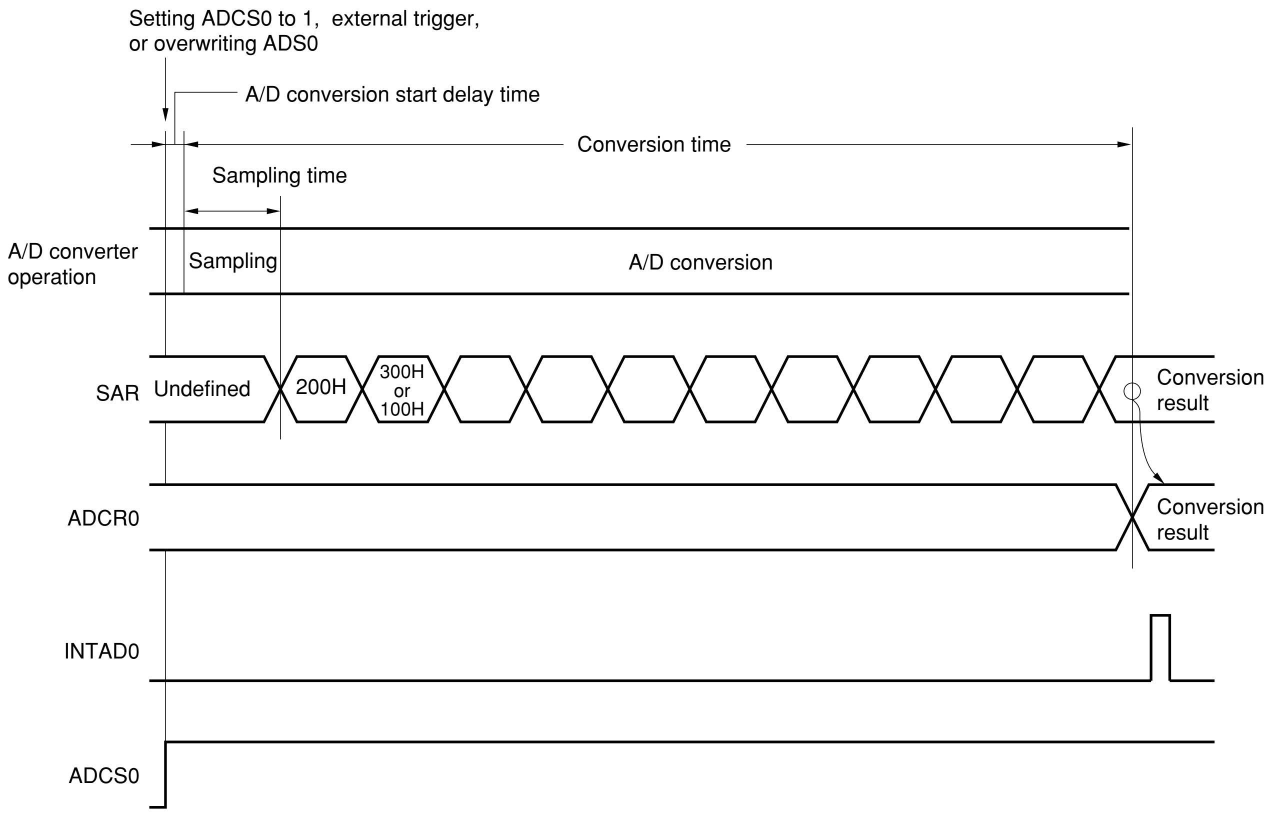
<7> Comparison is continued in this way up to bit 0 of SAR.

<8> Upon completion of the comparison of 10 bits, an effective digital result value remains in SAR, and the result value is transferred to and latched in A/D conversion result register 0 (ADCR0).
At the same time, the A/D conversion end interrupt request (INTAD0) can also be generated.

★ **Cautions 1. The first A/D conversion value immediately after A/D conversion has been started may not satisfy the rated value. Take measures such as polling the A/D conversion end interrupt request (INTAD0) and removing the first conversion results.**

★ **2. The A/D converter stops operation in standby mode.**

★ **Figure 14-5. Basic Operation of 10-Bit A/D Converter**



A/D conversion operations are performed continuously until bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) is reset (0) by software.

$\overline{\text{RESET}}$ input clears A/D conversion result register 0 (ADCR0) to 00H.

★ Confirm the conversion results by referring to the A/D conversion end interrupt request flag (ADIF0).

The sampling time of the A/D converter varies depending on the values set in A/D converter mode register 0 (ADM0). There is a delay time from when the A/D converter is enabled for operation until sampling is actually performed.

For the sets in which a strict A/D conversion time is required, note the contents described in Table 14-2.

★ **Table 14-2. Sampling Time and A/D Conversion Start Delay Time of A/D Converter**

FR02	FR01	FR00	Conversion Time[Note 1]	Sampling Time	A/D Conversion Start Delay Time	
					MIN.	MAX.
0	0	0	144/f_X	20/f_X	0.5/f_{CPU} + 6/f_X	0.5/f_{CPU} + 8/f_X
0	0	1	120/f_X	16/f_X		
0	1	0	96/f_X	12/f_X		
1	0	0	72/f_X	10/f_X	0.5/f_{CPU} + 3/f_X	0.5/f_{CPU} + 4/f_X
1	0	1	60/f_X	8/f_X		
1	1	0	48/f_X	6/f_X		
Other than above			Setting prohibited	–	–	–

Notes 1. Set the A/D conversion time as follows.

- When operated at f_X = 12 MHz[Note 2] (V_{DD} = 4.5 to 5.5 V): 12 μs or more
- When operated at f_X = 8.38 MHz (V_{DD} = 4.0 to 5.5 V): 14 μs or more

2. Expanded-specification products of μPD780034A Subseries only.

Remark f_X: Main system clock oscillation frequency
f_{CPU}: CPU clock frequency

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 theoretical A/D conversion result (stored in A/D conversion result register 0 (ADCR0)) is shown by the following expression.

$$\text{ADCR0} = \text{INT}\left(\frac{V_{IN}}{AV_{REF}} \times 1024 + 0.5\right)$$

or

$$(\text{ADCR0} - 0.5) \times \frac{AV_{REF}}{1024} \leq V_{IN} < (\text{ADCR0} + 0.5) \times \frac{AV_{REF}}{1024}$$

where, INT( ): Function which returns integer part of value in parentheses

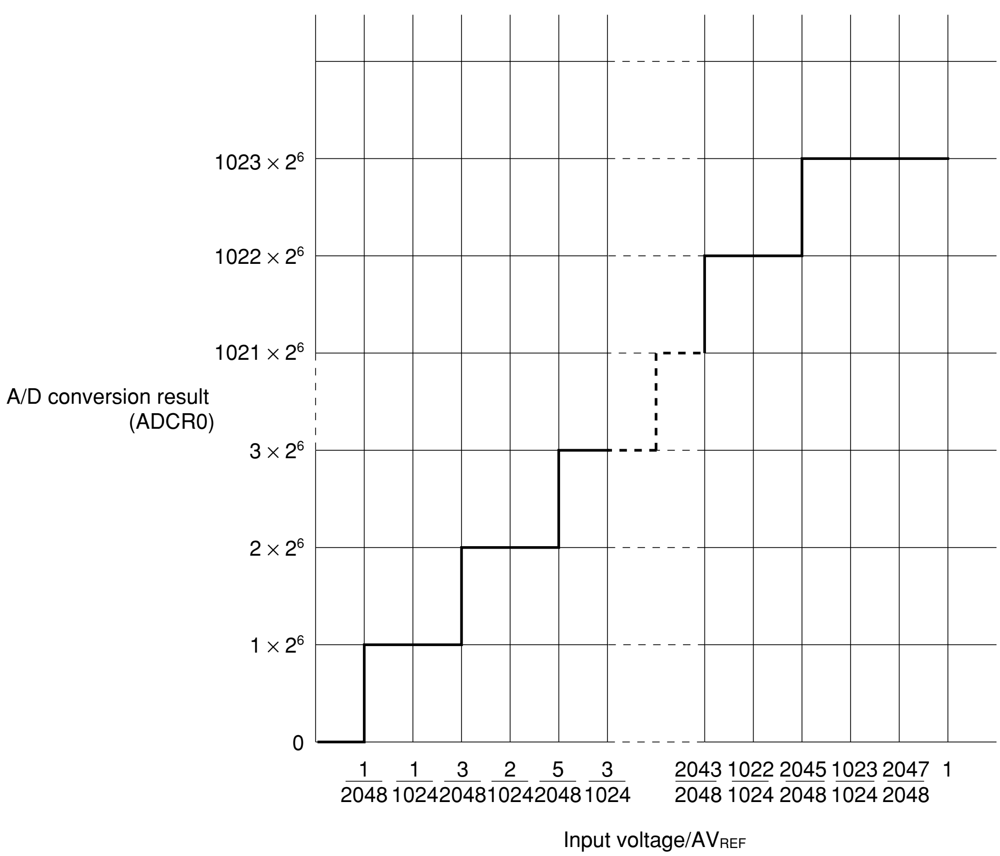
V_{IN}: Analog input voltage

AV_{REF}: AV_{REF} pin voltage

ADCR0: A/D conversion result register 0 (ADCR0) value

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 operation mode

Select one analog input channel from among ANI0 to ANI7 using analog input channel specification register 0 (ADS0) to start A/D conversion.

A/D conversion can be started in either of the following two ways.

- Hardware start: Conversion is started by trigger input (rising edge, falling edge, or both rising and falling edges enabled).
- Software start: Conversion is started by setting A/D converter mode register 0 (ADM0).

★ When A/D conversion is complete, the interrupt request signal (INTAD0) is generated.

(1) A/D conversion by hardware start

When bit 6 (TRG0) and bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) are set to 1, the A/D conversion standby state is set. When the external trigger signal (ADTRG) is input, A/D conversion of the voltage applied to the analog input pin specified by analog input channel specification register 0 (ADS0) starts.

Upon the end of A/D conversion, the conversion result is stored in A/D conversion result register 0 (ADCR0), and the interrupt request signal (INTAD0) is generated. After one A/D conversion operation is started and finished, the next conversion operation is not started until a new external trigger signal is input.

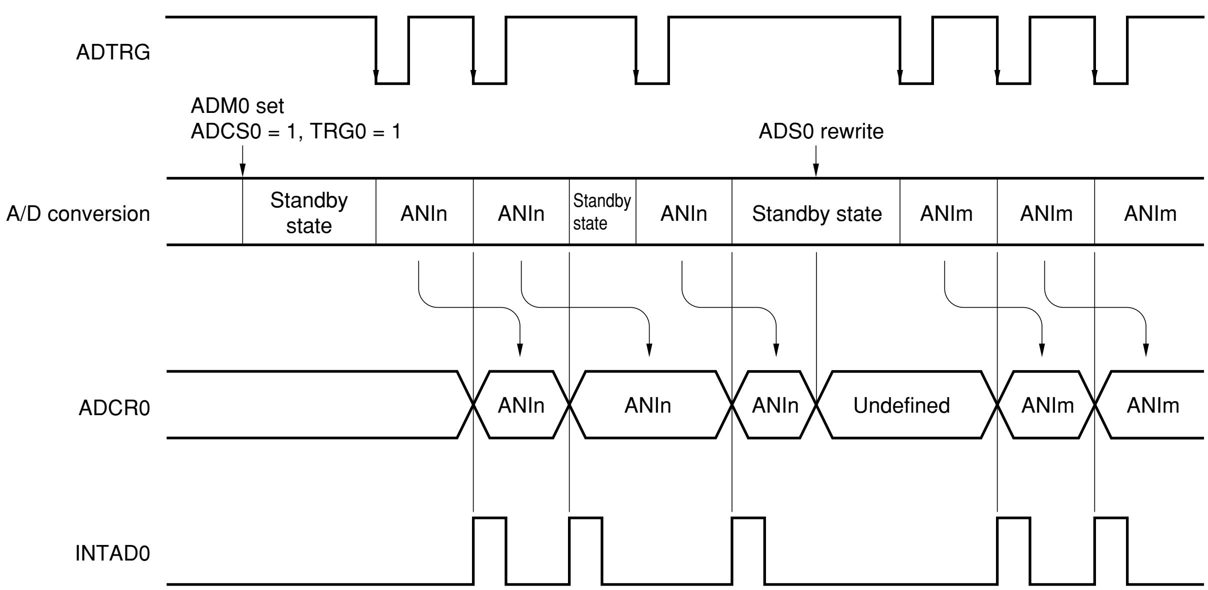
If ADS0 is rewritten during A/D conversion, the converter suspends A/D conversion and waits for a new external trigger signal to be input. When the external trigger input signal is reinput, A/D conversion is carried out from the beginning. If ADS0 is rewritten during A/D conversion standby, A/D conversion starts when the following external trigger input signal is input.

★ If 1 is written to ADCS0 again during A/D conversion, the A/D conversion in progress is discontinued and a new A/D conversion is started when the next external trigger input signal is input.

If 0 is written to ADCS0 during A/D conversion, the A/D conversion operation stops immediately.

Caution When P03/INTP3/ADTRG is used as the external trigger input (ADTRG), specify the valid edge using bits 1, 2 (EGA00, EGA01) of A/D converter mode register 0 (ADM0) and set the interrupt mask flag (PMK3) to 1.

★ **Figure 14-7. A/D Conversion by Hardware Start (When Falling Edge Is Specified)**



Remarks 1. n = 0, 1, ..., 7

2. m = 0, 1, ..., 7

(2) A/D conversion by software start

When bit 6 (TRG0) and bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) are set to 0 and 1, respectively, A/D conversion of the voltage applied to the analog input pin specified by analog input channel specification register 0 (ADS0) starts.

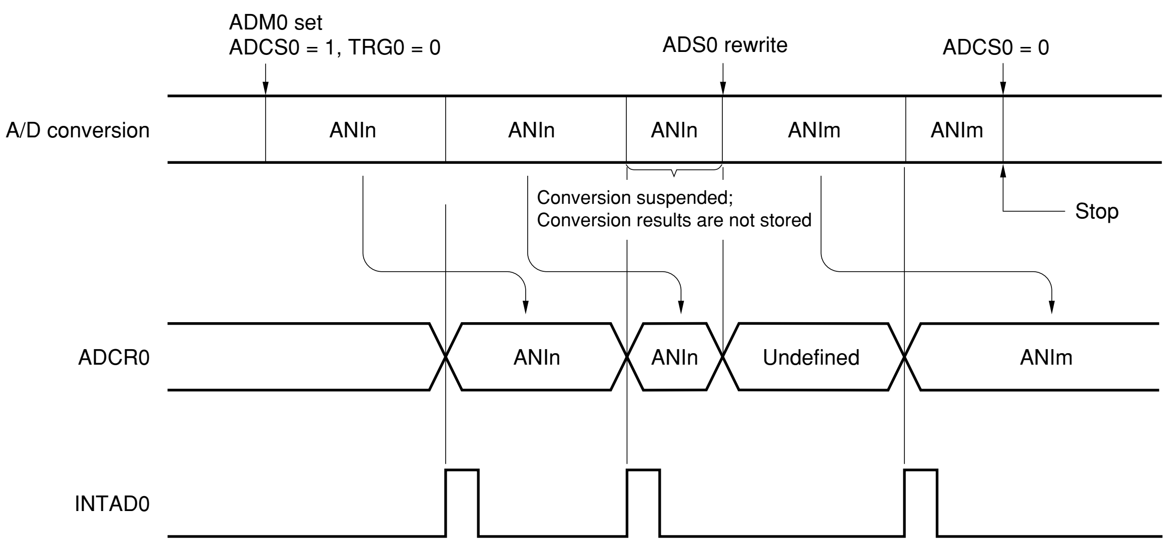
Upon the end of the A/D conversion, the conversion result is stored in A/D conversion result register 0 (ADCR0), and the interrupt request signal (INTAD0) is generated. After one A/D conversion operation is started and ended, the next conversion operation is immediately started. A/D conversion operations are repeated until new data is written to ADS0.

If ADS0 is rewritten during A/D conversion, the converter suspends A/D conversion and A/D conversion of the new selected analog input channel is started.

★ If 1 is written to ADCS0 again during A/D conversion, the A/D conversion in progress is discontinued and a new A/D conversion is started.

If 0 is written to ADCS0 during A/D conversion, the A/D conversion operation stops immediately.

★ **Figure 14-8. A/D Conversion by Software Start**



Remarks 1. n = 0, 1, ..., 7

2. m = 0, 1, ..., 7

14.5 How to Read A/D Converter Characteristics Table

Here we will explain the special terms unique to A/D converters.

(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 10 bits,

$$\begin{aligned} 1\text{LSB} &= 1/2^{10} = 1/1024 \\ &= 0.098\%\text{FSR} \end{aligned}$$

Accuracy has no relation to resolution, but is determined by overall error.

(2) Overall error

This shows the maximum error value between the actual measured value and the theoretical value.

Zero scale error, full scale error, integral linearity error, differential linearity error and errors which are combinations of these express overall error.

Furthermore, quantization error is not included in overall error in the characteristics table.

(3) Quantization error

When analog values are converted to digital values, there naturally occurs a ±1/2LSB error. In an A/D converter, an analog input voltage in a range of ±1/2LSB are converted to the same digital code, so a quantization error cannot be avoided.

Furthermore, it is not included in the overall error, zero scale error, full scale error, integral linearity error, and differential linearity error in the characteristics table.

Figure 14-9. Overall Error

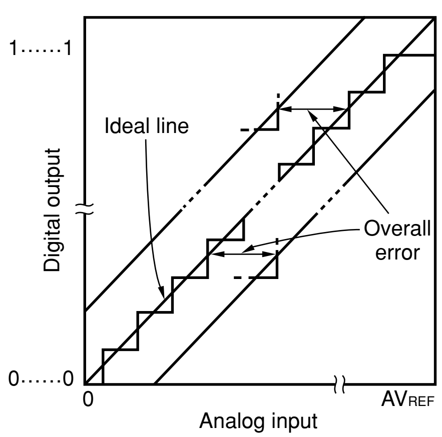
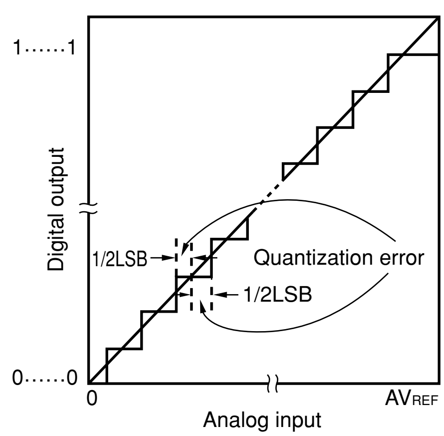


Figure 14-10. Quantization Error



(4) Zero scale error

This shows the difference between the actual measured value of the analog input voltage and the theoretical value (1/2LSB) when the digital output changes from 0……000 to 0……001. If the actual measured value is greater than the theoretical value, it shows the difference between the actual measured value of the analog input voltage and the theoretical value (3/2LSB) when the digital output changes from 0……001 to 0……010.

(5) Full scale error

This shows the difference between the actual measured value of the analog input voltage and the theoretical value (full scale –3/2LSB) when the digital output changes from 1……110 to 1……111.

(6) Integral linearity error

This shows the degree to which the conversion characteristics deviate from the ideal linear relationship. It expresses the maximum value of the difference between the actual measured value and the ideal straight line when the zero scale error and full scale error are 0.

(7) Differential linearity error

Although the ideal output width for a given code is 1LSB, this value shows the difference between the actual measured value and the ideal value of the width when outputting a particular code.

Figure 14-11. Zero Scale Error

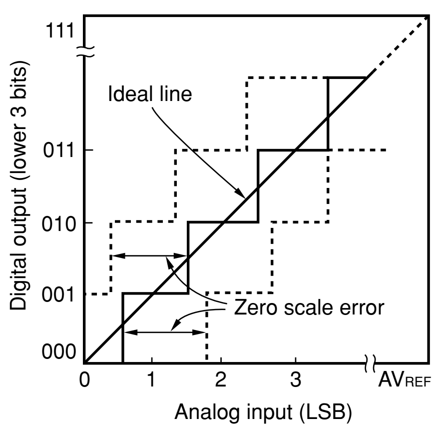


Figure 14-12. Full Scale Error

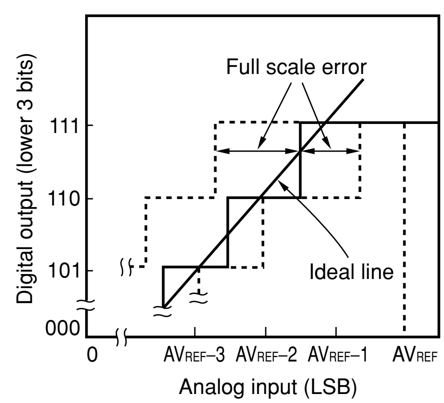


Figure 14-13. Integral Linearity Error

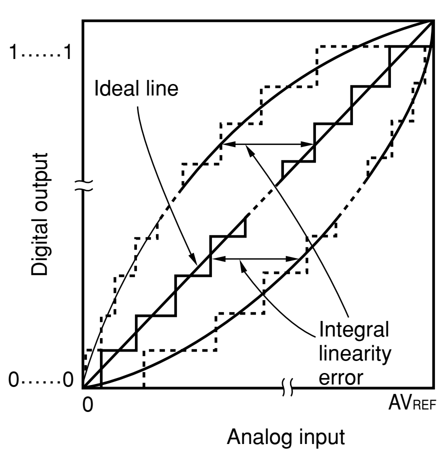
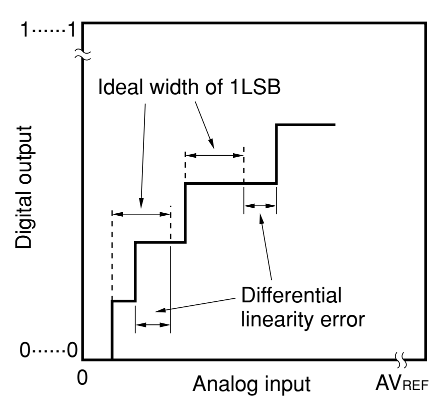


Figure 14-14. Differential Linearity Error

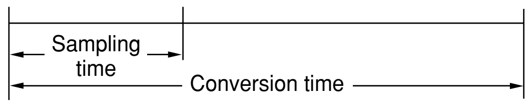


(8) Conversion time

This expresses the time from when the sampling was started to the time when the digital output was obtained. Sampling time is included in the conversion time in the characteristics table.

(9) Sampling time

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.

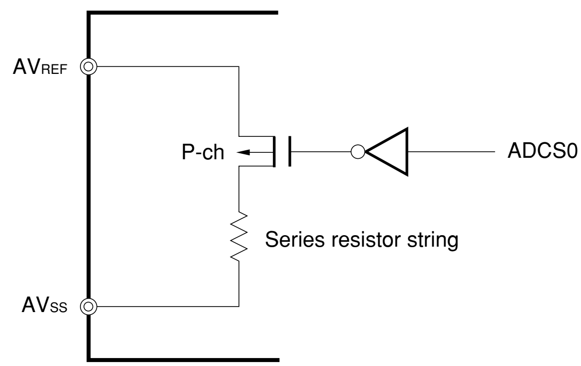


14.6 A/D Converter Cautions

(1) Power consumption in standby mode

A/D converter stops operating in the standby mode. At this time, power consumption can be reduced by stopping the conversion operation (by clearing bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) to 0).
Figure 14-15 shows the circuit configuration of a series resistor string.

Figure 14-15. Circuit Configuration of Series Resistor String



(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 higher than AV_{REF} or lower than AV_{SS} is input (even if within the absolute maximum rating range), the conversion value of that channel will be undefined and the conversion values of other channels may also be affected.

(3) Contending operations

<1> Contention between A/D conversion result register 0 (ADCR0) write and ADCR0 read by instruction upon the end of conversion
ADCR0 read is given priority. After the read operation, the new conversion result is written to ADCR0.

<2> Contention between ADCR0 write and external trigger signal input upon the end of conversion
The external trigger signal is not accepted during A/D conversion. Therefore, the external trigger signal is not accepted during ADCR0 write.

<3> Contention between ADCR0 write and A/D converter mode register 0 (ADM0) write or analog input channel specification register 0 (ADS0) write upon the end of conversion
ADM0 or ADS0 write is given priority. ADCR0 write is not performed, nor is the conversion end interrupt request signal (INTAD0) generated.

(4) ANI0/P10 to ANI7/P17

<1> The analog input pins (ANI0 to ANI7) also function as input port pins (P10 to P17).
When A/D conversion is performed with any of pins ANI0 to ANI7 selected, do not access port 1 while conversion is in progress, as this may reduce the conversion resolution.

<2> If digital pulses are applied to the pin adjacent to a 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 the pin adjacent to a pin undergoing A/D conversion.

(5) AV$_{REF}$ pin input impedance

A series resistor string is connected between the AV$_{REF}$ pin and the AV$_{SS}$ pin.

Therefore, when the output impedance of the reference voltage is too high, it seems as if the AV$_{REF}$ pin and the series resistor string are connected in series. This may cause a greater reference voltage error.

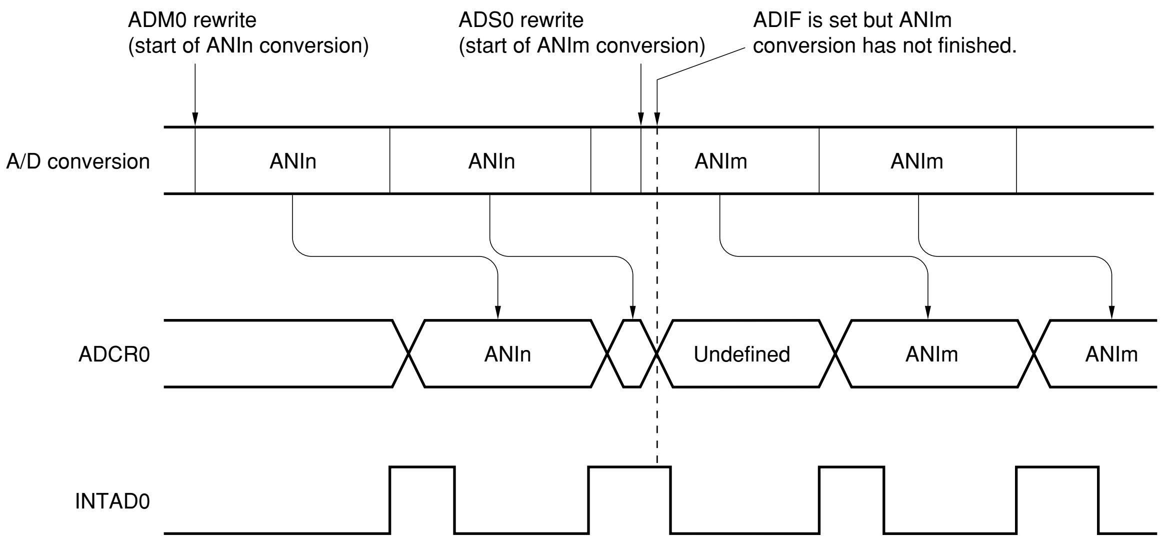
(6) Interrupt request flag (ADIF0)

The interrupt request flag (ADIF0) is not cleared even if analog input channel specification register 0 (ADS0) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and conversion end interrupt request flag for the pre-change analog input may be set just before the ADS0 rewrite. Caution is therefore required since, at this time, when ADIF0 is read immediately just after the ADS0 rewrite, ADIF0 is set despite the fact that the A/D conversion for the post-change analog input has not finished.

When A/D conversion is restarted after it is stopped, clear ADIF0 before restart.

Figure 14-16. A/D Conversion End Interrupt Request Generation Timing



Remarks 1. n = 0, 1, ..., 7

2. m = 0, 1, ..., 7

(7) Conversion results just after A/D conversion start

The A/D conversion value immediately after A/D conversion has been started may not satisfy the rated value. Take measures such as polling the A/D conversion end interrupt request (INTAD0) and removing the first conversion results.

(8) A/D conversion result register 0 (ADCR0) read operation

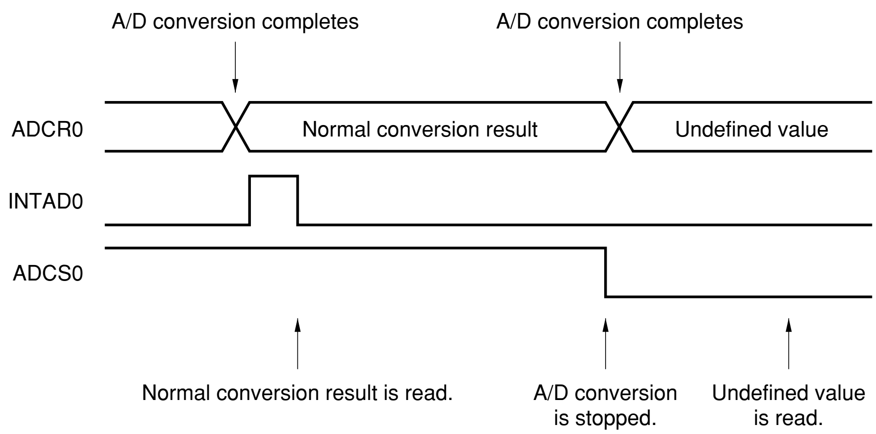
When A/D converter mode register 0 (ADM0) and analog input channel specification register 0 (ADS0) are written, the contents of ADCR0 may become undefined. Read the conversion result following conversion completion before writing to ADM0 and ADS0. Using a timing other than the above may cause an incorrect conversion result to be read.

(9) 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 before stopping the A/D conversion operation.

Figure 14-17 shows the timing of reading the conversion result.

Figure 14-17. Timing of Reading Conversion Result (When Conversion Result Is Undefined)



(10) 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.

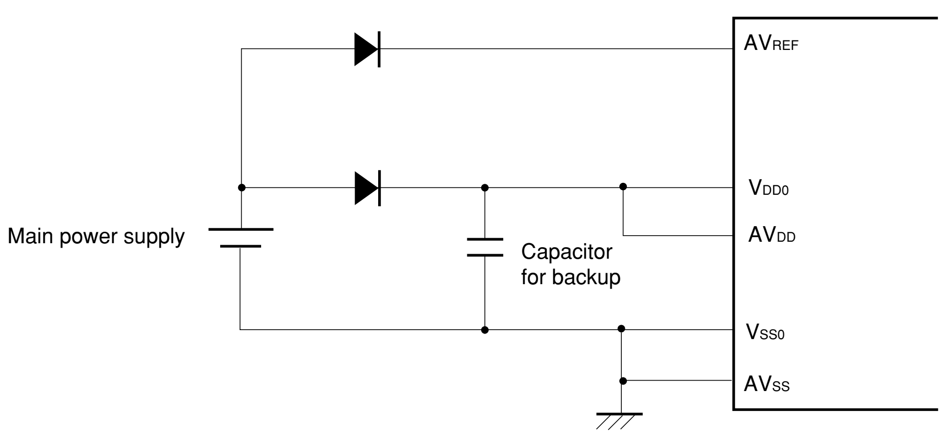
Connect AV_{SS0} and V_{SS0} at one location on the board where the voltages are stable.

(11) AV_{DD} pin

The AV_{DD} pin is the analog circuit power supply pin. It supplies power to the input circuits of the ANI0 to ANI7 pins.

Therefore, be sure to apply the same potential as V_{DD0} to this pin even for applications designed to switch to a backup battery for power supply.

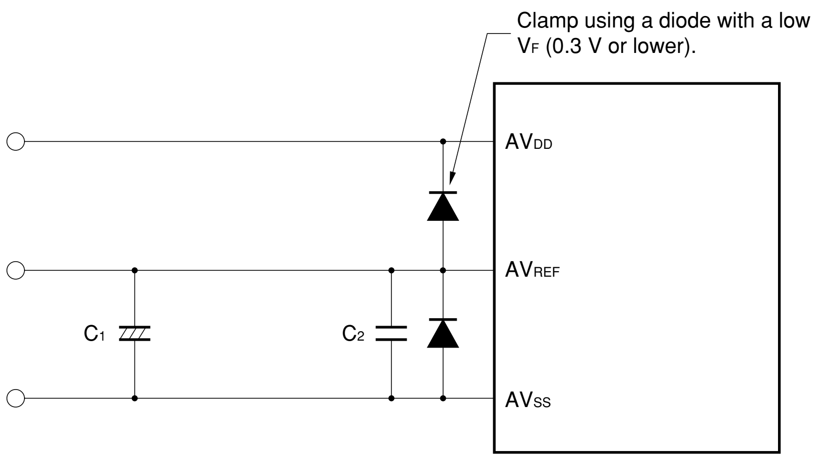
Figure 14-18. AV_{DD} Pin Connection



(12) AVREF pin

Connect a capacitor to the AVREF 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 AVREF pin becomes unstable, causing the accuracy of the A/D conversion to drop. To prevent this, also connect a capacitor to the AVREF pin.
Figure 14-19 shows an example of connecting a capacitor.

★ **Figure 14-19. Example of Connecting Capacitor to AVREF Pin**

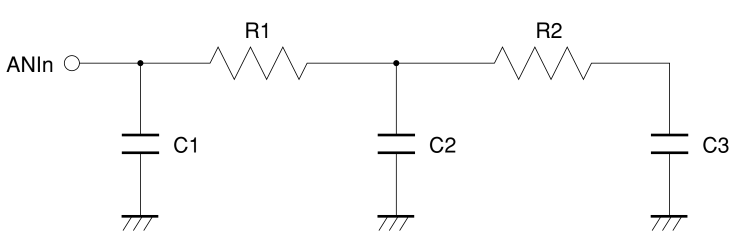


Remark C1: 4.7 μF to 10 μF (reference value)
C2: 0.01 μF to 0.1 μF (reference value)
Connect C2 as close to the pin as possible.

(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 14-20 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 pins ANI0 to ANI7. An example of this is shown in Figure 14-21. In this case, however, the microcontroller cannot follow an analog signal with a high differential coefficient because a lowpass 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 14-20. Internal Equivalent Circuit of Pins ANI0 to ANI7



Remark n = 0 to 7

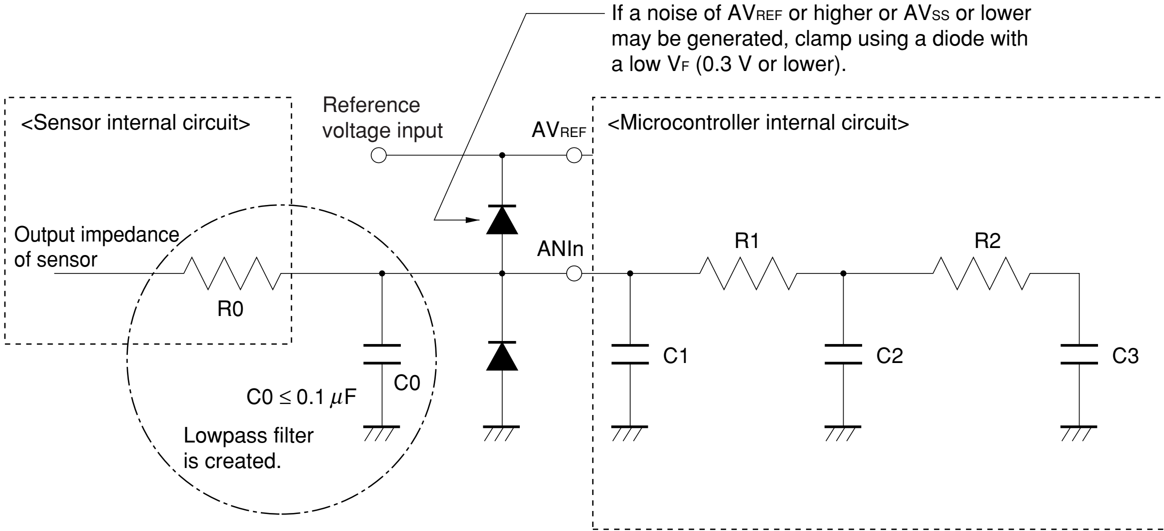
★ **Table 14-3. Resistances and Capacitances of Equivalent Circuit (Reference Values)**

(TYP.)

AV_{REF}	R1	R2	C1	C2	C3
2.7 V	12 kΩ	8.0 kΩ	3.0 pF	3.0 pF	2.0 pF
4.5 V	4 kΩ	2.7 kΩ	3.0 pF	1.4 pF	2.0 pF

Caution The resistances and capacitances in Table 14-3 are not guaranteed values.

Figure 14-21. Example of Connection If Signal Source Impedance Is High



Remark n = 0 to 7

★ **(14) Input impedance of ANI0 to ANI7 pins**

This A/D converter executes sampling by charging the internal sampling capacitor for approximately 1/10 of the conversion time.

Therefore, only the leakage current flows during other than sampling, and the current for charging the capacitor flows during sampling. The input impedance therefore varies and has no meaning.

To achieve sufficient sampling, it is recommended that the output impedance of the analog input source be 10 kΩ or less, or attach a capacitor of around 100 pF to the ANI0 to ANI7 pins (see **Figure 14-21**).

CHAPTER 15 SERIAL INTERFACE OUTLINE

The μPD780024A, 780034A Subseries and the μPD780024AY, 780034AY Subseries have differences in their serial interfaces. These differences are listed in Table 15-1.

Table 15-1. Differences Between μPD780024A, 780034A Subseries and μPD780024AY, 780034AY Subseries

Item		μPD780024A, 780034A	μPD780024AY, 780034AY	Relevant Section
UART0		√	√	CHAPTER 16
SIO3	SIO30	√	√	CHAPTER 17
	SIO31	√	–	
IIC0		–	√	CHAPTER 18

16.1 Functions of Serial Interface UART0

Serial interface UART0 has the following three modes.

(1) Operation stop mode

This mode is used when serial transfers are not performed to reduce power consumption.
For details, see **16.4.1 Operation stop mode**.

(2) Asynchronous serial interface (UART) mode (fixed to LSB first)

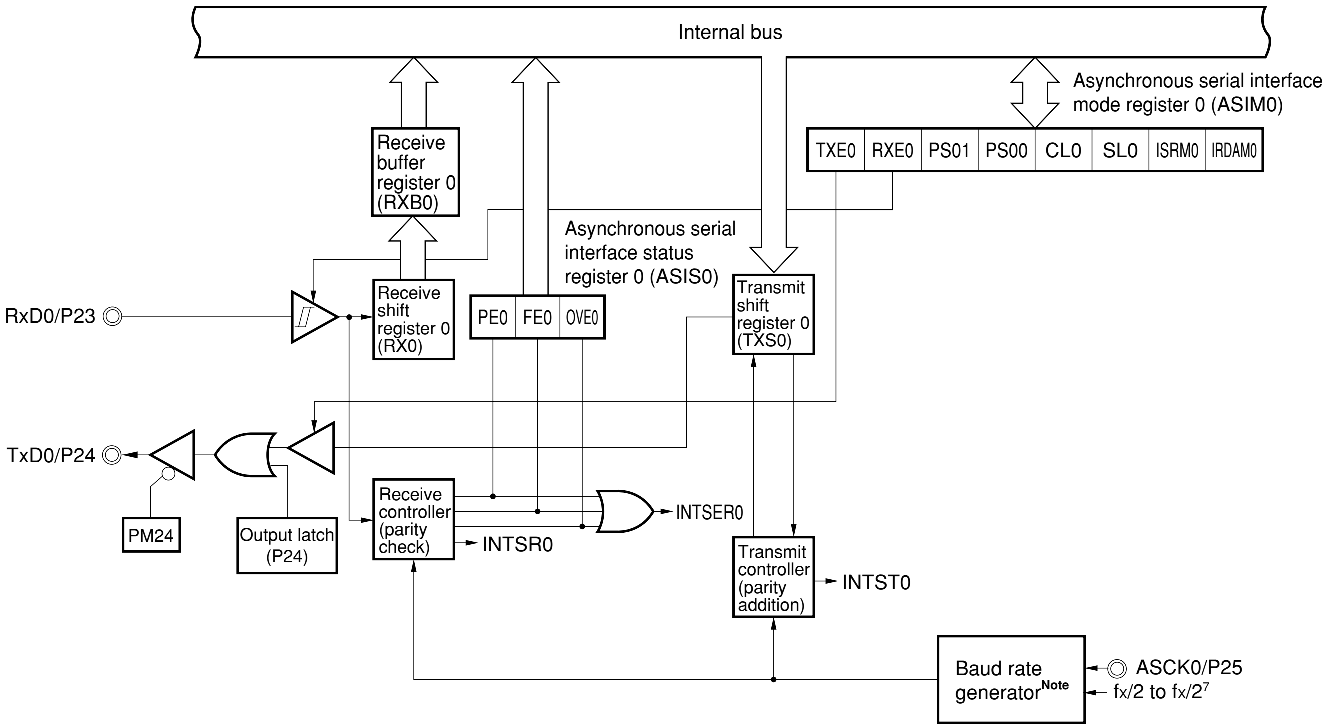
This mode enables full-duplex operation wherein one byte of data after the start bit is transmitted and received.
★ The on-chip baud rate generator dedicated to UART enables communications using a wide range of selectable baud rates. The communication range is between 1.2 kbps and 131 kbps (when operated at f_X = 8.38 MHz). In addition, a baud rate (39 kbps max. (when operated at f_X = 1.25 MHz)) can also be defined by dividing the clock input to the ASCK0 pin.
The UART baud rate generator can also be used to generate a MIDI-standard baud rate (31.25 kbps).
For details, see **16.4.2 Asynchronous serial interface (UART) mode**.

(3) Infrared data transfer mode

For details, see **16.4.3 Infrared data transfer mode**.

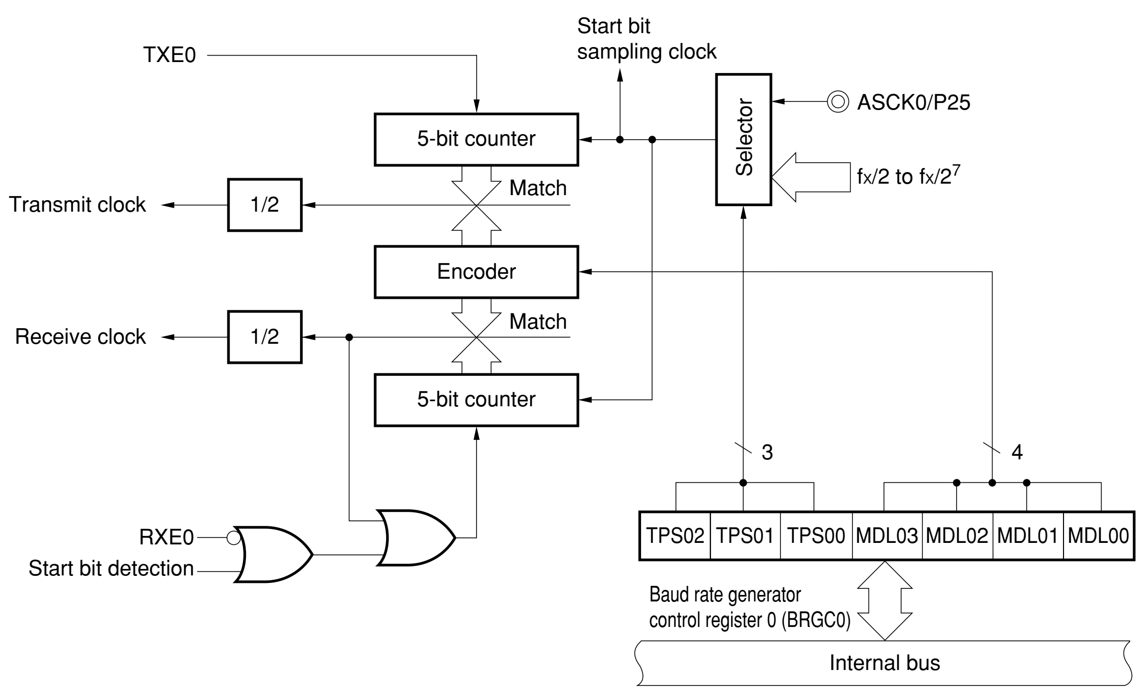
Figure 16-1 shows a block diagram of serial interface UART0.

★ **Figure 16-1. Block Diagram of Serial Interface UART0**



Note For the configuration of the baud rate generator, see **Figure 16-2**.

★ **Figure 16-2. Block Diagram of Baud Rate Generator**



Remark TXE0: Bit 7 of asynchronous serial interface mode register 0 (ASIM0)
RXE0: Bit 6 of asynchronous serial interface mode register 0 (ASIM0)

16.2 Configuration of Serial Interface UART0

Serial interface UART0 includes the following hardware.

★ **Table 16-1. Configuration of Serial Interface UART0**

Item	Configuration
Registers	Transmit shift register 0 (TXS0) Receive shift register 0 (RX0) Receive buffer register 0 (RXB0) Asynchronous serial interface status register 0 (ASIS0)
Control registers	Asynchronous serial interface mode register 0 (ASIM0) Baud rate generator control register 0 (BRGC0) Port mode register 2 (PM2) Port 2 (P2)

(1) Transmit shift register 0 (TXS0)

This is a register for setting transmit data. Data written to TXS0 is transmitted as serial data.
When the data length is set as 7 bits, bits 0 to 6 of the data written to TXS0 are transferred as transmit data.
Writing data to TXS0 starts the transmit operation.
TXS0 can be written by an 8-bit memory manipulation instruction. It cannot be read.
$\overline{\text{RESET}}$ input sets TXS0 to FFH.

Caution Do not write to TXS0 during a transmit operation.
The same address is assigned to TXS0 and receive buffer register 0 (RXB0), so a read operation reads values from RXB0.

(2) Receive shift register 0 (RX0)

This register converts serial data input via the RxD0 pin to parallel data. When one byte of data is received at this register, the receive data is transferred to receive buffer register 0 (RXB0).
RX0 cannot be manipulated directly by a program.

(3) Receive buffer register 0 (RXB0)

This register is used to hold receive data. When one byte of data is received, one byte of new receive data is transferred from the receive shift register (RX0).
When the data length is set as 7 bits, receive data is sent to bits 0 to 6 of RXB0. In this case, the MSB of RXB0 is always 0.
RXB0 can be read by an 8-bit memory manipulation instruction. It cannot be written.
$\overline{\text{RESET}}$ input sets RXB0 to FFH.

Caution The same address is assigned to RXB0 and transmit shift register 0 (TXS0), so during a write operation, values are written to TXS0.

★ (4) Asynchronous serial interface status register 0 (ASIS0)

When a receive error occurs in UART mode, this register indicates the type of error.
ASIS0 can be read by an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears ASIS0 to 00H.

Figure 16-3. Format of Asynchronous Serial Interface Status Register 0 (ASIS0)

Address: FFA1H After reset: 00H R

Symbol	7	6	5	4	3	2	1	0
ASIS0	0	0	0	0	0	PE0	FE0	OVE0

PE0	Parity error flag
0	No parity error
1	Parity error (Transmit data parity not matched)

FE0	Framing error flag
0	No framing error
1	Framing error[Note 1] (Stop bit not detected)

OVE0	Overrun error flag
0	No overrun error
1	Overrun error[Note 2] (Next receive operation was completed before data was read from receive buffer register 0 (RXB0))

Notes 1. Even if the stop bit length is set to two bits by setting bit 2 (SL0) of asynchronous serial interface mode register 0 (ASIM0), stop bit detection during a receive operation only applies to a stop bit length of 1 bit.

2. When an overrun error has occurred, further overrun errors will continue to occur until the contents of receive buffer register 0 (RXB0) are read.

(5) Transmission controller

The transmission controller controls transmit operations, such as adding a start bit, parity bit, and stop bit to data that is written to transmit shift register 0 (TXS0), based on the values set to asynchronous serial interface mode register 0 (ASIM0).

(6) Reception controller

The reception controller controls receive operations based on the values set to asynchronous serial interface mode register 0 (ASIM0). During a receive operation, it performs error checking, such as for parity errors, and sets various values to asynchronous serial interface status register 0 (ASIS0) according to the type of error that is detected.

16.3 Registers to Control Serial Interface UART0

Serial interface UART0 uses the following four registers for control functions.

- Asynchronous serial interface mode register 0 (ASIM0)
- Baud rate generator control register 0 (BRGC0)
- Port mode register 2 (PM2)
- Port 2 (P2)

(1) Asynchronous serial interface mode register 0 (ASIM0)

This is an 8-bit register that controls serial interface UART0's serial transfer operations.
ASIM0 is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears ASIM0 to 00H.
Figure 16-4 shows the format of ASIM0.

Figure 16-4. Format of Asynchronous Serial Interface Mode Register 0 (ASIM0)

Address: FFA0H After reset: 00H R/W

Symbol	<7>	<6>	5	4	3	2	1	0
ASIM0	TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0

TXE0	RXE0	Operation mode	RxD0/P23 pin function	TxD0/P24 pin function
0	0	Operation stop	Port function (P23)	Port function (P24)
0	1	UART mode (receive only)	Serial function (RxD0)	
1	0	UART mode (transmit only)	Port function (P23)	Serial function (TxD0)
1	1	UART mode (transmit and receive)	Serial function (RxD0)	

PS01	PS00	Parity bit specification
0	0	No parity
0	1	Zero parity always added during transmission No parity detection during reception (parity errors do not occur)
1	0	Odd parity
1	1	Even parity

CL0	Character length specification
0	7 bits
1	8 bits

SL0	Stop bit length specification for transmit data
0	1 bit
1	2 bits

ISRM0	Receive completion interrupt control when error occurs
0	Receive completion interrupt request is issued when an error occurs
1	Receive completion interrupt request is not issued when an error occurs

IRDAM0	Operation specified for infrared data transfer mode[Note 1]
0	UART (transmit/receive) mode
1	Infrared data transfer (transmit/receive) mode[Note 2]

Notes 1. The UART/infrared data transfer mode specification is controlled by TXE0 and RXE0.

2. When using infrared data transfer mode, be sure to set baud rate generator control register 0 (BRGC0) to “10H”.

★ **Caution Before writing different data to ASIM0, stop operation.**

(2) Baud rate generator control register 0 (BRGC0)

This register sets the serial clock for the serial interface.

BRGC0 is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BRGC0 to 00H.

Figure 16-5 shows the format of BRGC0.

Figure 16-5. Format of Baud Rate Generator Control Register 0 (BRGC0)

Address: FFA2H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
BRGC0	0	TPS02	TPS01	TPS00	MDL03	MDL02	MDL01	MDL00

(f_X = 8.38 MHz)

TPS02	TPS01	TPS00	Source clock selection for 5-bit counter	n
0	0	0	External clock input to ASCK0	0
0	0	1	$f_X/2$	1
0	1	0	$f_X/2^2$	2
0	1	1	$f_X/2^3$	3
1	0	0	$f_X/2^4$	4
1	0	1	$f_X/2^5$	5
1	1	0	$f_X/2^6$	6
1	1	1	$f_X/2^7$	7

MDL03	MDL02	MDL01	MDL00	Output clock selection for baud rate generator	k
0	0	0	0	$f_{SCK0}/16$	0
0	0	0	1	$f_{SCK0}/17$	1
0	0	1	0	$f_{SCK0}/18$	2
0	0	1	1	$f_{SCK0}/19$	3
0	1	0	0	$f_{SCK0}/20$	4
0	1	0	1	$f_{SCK0}/21$	5
0	1	1	0	$f_{SCK0}/22$	6
0	1	1	1	$f_{SCK0}/23$	7
1	0	0	0	$f_{SCK0}/24$	8
1	0	0	1	$f_{SCK0}/25$	9
1	0	1	0	$f_{SCK0}/26$	10
1	0	1	1	$f_{SCK0}/27$	11
1	1	0	0	$f_{SCK0}/28$	12
1	1	0	1	$f_{SCK0}/29$	13
1	1	1	0	$f_{SCK0}/30$	14
1	1	1	1	Setting prohibited	–

Cautions 1. Writing to BRGC0 during a communication operation may cause abnormal output from the baud rate generator and disable further communication operations. Therefore, do not write to BRGC0 during a communication operation.

2. Set BRGC0 to 10H when using in infrared data transfer mode.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{SCK0}: Source clock for 5-bit counter

3. n: Value set via TPS00 to TPS02 ($0 \le n \le 7$)

4. k: Value set via MDL00 to MDL03 ($0 \le k \le 14$)

★ **5.** The equation for the baud rate is as follows.

$$[\text{Baud rate}] = \frac{f_X}{2^{n+1}(k + 16)} \text{ [Hz]}$$

★ **(3) Port mode register 2 (PM2)**

Port mode register 2 is used to set input/output of port 2 in 1-bit units.

To use the P24/TxD0 pin as a serial data output, clear PM24 and the output latch of P24 to 0.

To use the P23/RxD0 pin as a serial data input, and the P25/ASCK0 pin as a clock input, set PM23 and PM25 to 1. At this time, the output latches of P23 and P25 can be either 0 or 1.

PM2 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM2 to FFH.

Figure 16-6. Format of Port Mode Register 2 (PM2)

Address: FF22H After reset: FFH R/W

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

PM2n	I/O mode selection of P2n pin (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

16.4 Operation of Serial Interface UART0

This section explains the three modes of serial interface UART0.

16.4.1 Operation stop mode

Because serial transfer is not performed in this mode, the power consumption can be reduced.
In addition, pins can be used as ordinary ports.

(1) Register to be used

Operation stop mode is set by asynchronous serial interface mode register 0 (ASIM0).
ASIM0 is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears ASIM0 to 00H.

Address: FFA0H After reset: 00H R/W

Symbol	<7>	<6>	5	4	3	2	1	0
ASIM0	TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0

TXE0	RXE0	Operation mode	RxD0/P23 pin function	TxD0/P24 pin function
0	0	Operation stop	Port function (P23)	Port function (P24)

16.4.2 Asynchronous serial interface (UART) mode

This mode enables full-duplex operation wherein one byte of data after the start bit is transmitted or received.

The on-chip baud rate generator dedicated to UART enables communications using a wide range of selectable ★ baud rates. The communication range is between 1.2 kbps and 131 kbps (when operated at f_X = 8.38 MHz). The baud rate (39 kbps max. (when operated of f_X = 1.25 MHz)) can be defined by dividing the input clock to the ASCK0 pin.

The UART baud rate generator can also be used to generate a MIDI-standard baud rate (31.25 kbps).

(1) Registers to be used

- Asynchronous serial interface mode register 0 (ASIM0)
- Asynchronous serial interface status register 0 (ASIS0)
- Baud rate generator control register 0 (BRGC0)
- Port mode register 2 (PM2)
- Port 2 (P2)

(a) Asynchronous serial interface mode register 0 (ASIM0)

ASIM0 is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears ASIM0 to 00H.

Address: FFA0H After reset: 00H R/W

Symbol	<7>	<6>	5	4	3	2	1	0
ASIM0	TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0

TXE0	RXE0	Operation mode	RxD0/P23 pin function	TxD0/P24 pin function
0	0	Operation stop	Port function (P23)	Port function (P24)
0	1	UART mode (receive only)	Serial function (RxD0)	
1	0	UART mode (transmit only)	Port function (P23)	Serial function (TxD0)
1	1	UART mode (transmit and receive)	Serial function (RxD0)	

PS01	PS00	Parity bit specification
0	0	No parity
0	1	Zero parity always added during transmission No parity detection during reception (parity errors do not occur)
1	0	Odd parity
1	1	Even parity

CL0	Character length specification
0	7 bits
1	8 bits

SL0	Stop bit length specification for transmit data
0	1 bit
1	2 bits

ISRM0	Receive completion interrupt control when error occurs
0	Receive completion interrupt request is issued when an error occurs
1	Receive completion interrupt request is not issued when an error occurs

IRDAM0	Operation specified for infrared data transfer mode[Note 1]
0	UART (transmit/receive) mode
1	Infrared data transfer (transmit/receive) mode[Note 2]

Notes 1. The UART/infrared data transfer mode specification is controlled by TXE0 and RXE0.

2. When using infrared data transfer mode, be sure to set baud rate generator control register 0 (BRGC0) to 10H.

★ **Caution Before writing different data to ASIM0, stop operation.**

(b) Asynchronous serial interface status register 0 (ASIS0)

ASIS0 can be read by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIS0 to 00H.

Address: FFA1H After reset: 00H R

Symbol	7	6	5	4	3	2	1	0
ASIS0	0	0	0	0	0	PE0	FE0	OVE0

PE0	Parity error flag
0	No parity error
1	Parity error (Transmit data parity not matched)

FE0	Framing error flag
0	No framing error
1	Framing error[Note 1] (Stop bit not detected)

OVE0	Overrun error flag
0	No overrun error
1	Overrun error[Note 2] (Next receive operation was completed before data was read from receive buffer register 0 (RXB0))

Notes 1. Even if the stop bit length is set to two bits by setting bit 2 (SL0) of asynchronous serial interface mode register 0 (ASIM0), stop bit detection during a receive operation only applies to a stop bit length of 1 bit.

2. When an overrun error has occurred, further overrun errors will continue to occur until the contents of receive buffer register 0 (RXB0) are read.

(c) Baud rate generator control register 0 (BRGC0)

BRGC0 is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BRGC0 to 00H.

Address: FFA2H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
BRGC0	0	TPS02	TPS01	TPS00	MDL03	MDL02	MDL01	MDL00

(f_X = 8.38 MHz)

TPS02	TPS01	TPS00	Source clock selection for 5-bit counter	n
0	0	0	External clock input to ASCK0	0
0	0	1	$f_X/2$	1
0	1	0	$f_X/2^2$	2
0	1	1	$f_X/2^3$	3
1	0	0	$f_X/2^4$	4
1	0	1	$f_X/2^5$	5
1	1	0	$f_X/2^6$	6
1	1	1	$f_X/2^7$	7

MDL03	MDL02	MDL01	MDL00	Output clock selection for baud rate generator	k
0	0	0	0	$f_{SCK0}/16$	0
0	0	0	1	$f_{SCK0}/17$	1
0	0	1	0	$f_{SCK0}/18$	2
0	0	1	1	$f_{SCK0}/19$	3
0	1	0	0	$f_{SCK0}/20$	4
0	1	0	1	$f_{SCK0}/21$	5
0	1	1	0	$f_{SCK0}/22$	6
0	1	1	1	$f_{SCK0}/23$	7
1	0	0	0	$f_{SCK0}/24$	8
1	0	0	1	$f_{SCK0}/25$	9
1	0	1	0	$f_{SCK0}/26$	10
1	0	1	1	$f_{SCK0}/27$	11
1	1	0	0	$f_{SCK0}/28$	12
1	1	0	1	$f_{SCK0}/29$	13
1	1	1	0	$f_{SCK0}/30$	14
1	1	1	1	Setting prohibited	–

Caution Writing to BRGC0 during a communication operation may cause abnormal output from the baud rate generator and disable further communication operations. Therefore, do not write to BRGC0 during a communication operation.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{SCK0}: Source clock for 5-bit counter

3. n: Value set via TPS00 to TPS02 ($0 \leq n \leq 7$)

4. k: Value set via MDL00 to MDL03 ($0 \leq k \leq 14$)

The transmit/receive clock that is used to generate the baud rate is obtained by dividing the main system clock.

- Transmit/receive clock generation for baud rate by using main system clock
  The main system clock is divided to generate the transmit/receive clock. The baud rate generated from the main system clock is determined according to the following formula.

$$[\text{Baud rate}] = \frac{f_X}{2^{n+1}(k + 16)} \quad [\text{Hz}]$$

f_X: Main system clock oscillation frequency
When ASCK0 is selected as the source clock of the 5-bit counter, substitute the input clock frequency to the ASCK0 pin for f_X in the above expression.
n: Value set via TPS00 to TPS02 ($0 \le n \le 7$)
k: Value set via MDL00 to MDL03 ($0 \le k \le 14$)

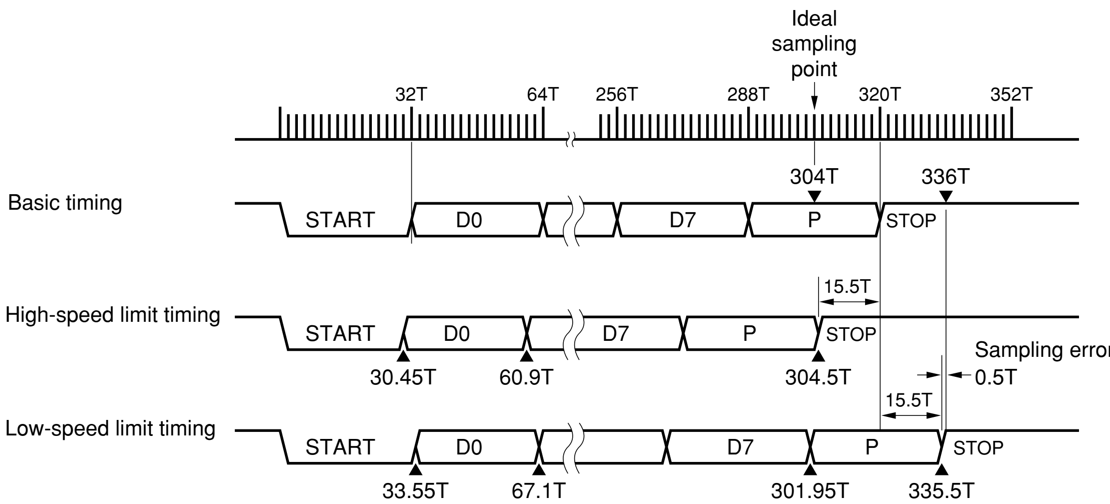
Table 16-2. Relationship Between Main System Clock and Baud Rate Error

Baud Rate (bps)	f_X = 8.3886 MHz		f_X = 8.000 MHz		f_X = 7.3728 MHz		f_X = 5.000 MHz		f_X = 4.1943 MHz	
	BRGC0	ERR (%)	BRGC0	ERR (%)	BRGC0	ERR (%)	BRGC0	ERR (%)	BRGC0	ERR (%)
600	–	–	–	–	–	–	–	–	7BH	1.14
1200	7BH	1.10	7AH	0.16	78H	0	70H	1.73	6BH	1.14
2400	6BH	1.10	6AH	0.16	68H	0	60H	1.73	5BH	1.14
4800	5BH	1.10	5AH	0.16	58H	0	50H	1.73	4BH	1.14
9600	4BH	1.10	4AH	0.16	48H	0	40H	1.73	3BH	1.14
19200	3BH	1.10	3AH	0.16	38H	0	30H	1.73	2BH	1.14
31250	31H	–1.3	30H	0	2DH	1.70	24H	0	21H	–1.3
38400	2BH	1.10	2AH	0.16	28H	0	20H	1.73	1BH	1.14
76800	1BH	1.10	1AH	0.16	18H	0	10H	1.73	–	–
115200	12H	1.10	11H	2.12	10H	0	–	–	–	–

Remark f_X: Main system clock oscillation frequency

- **Error tolerance range for baud rate**
  The error for the baud rate depends on the number of bits per frame and the 5-bit counter's division ratio [1/(16 + k)].
  Figure 16-7 shows an example of the baud rate error tolerance range.

Figure 16-7. Error Tolerance (When k = 0), Including Sampling Errors



Baud rate error tolerance (when k = 0) = $\frac{\pm 15.5}{320} \times 100 = 4.8438$ (%)

★ **Caution The above error tolerance value is the value calculated based on the ideal sample point. In the actual design, allow margins that include errors of timing for detecting a start bit.**

Remark T: 5-bit counter's source clock cycle

★ **(d) Port mode register 2 (PM2)**

Port mode register 2 is used to set input/output of port 2 in 1-bit units.

To use the P24/TxD0 pin as a serial data output, clear PM24 and the output latch of P24 to 0.

To use the P23/RxD0 pin as a serial data input, and the P25/ASCK0 pin as a clock input, set PM23 and PM25 to 1. At this time, the output latches of P23 and P25 can be either 0 or 1.

PM2 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM2 to FFH.

Address: FF22H After reset: FFH R/W

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

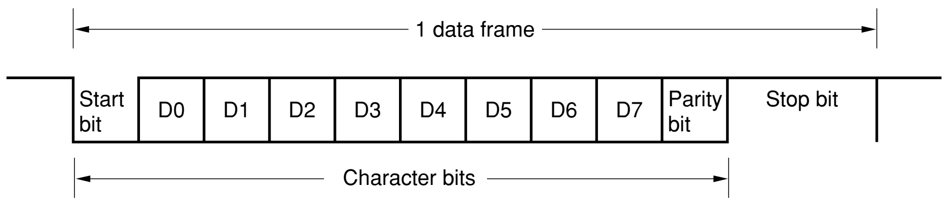
PM2n	I/O mode selection of P2n pin (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

(2) Communication operations

(a) Data format

Figure 16-8 shows the format of the transmit/receive data.

Figure 16-8. Example of Transmit/Receive Data Format in Asynchronous Serial Interface



1 data frame consists of the following bits.

- Start bit .... 1 bit
- Character bits ... 7 bits or 8 bits (LSB first)
- Parity bit .... Even parity, odd parity, zero parity, or no parity
- Stop bit(s) .... 1 bit or 2 bits

Asynchronous serial interface mode register 0 (ASIM0) is used to set the character bit length, parity selection, and stop bit length within each data frame.

When “7 bits” is selected as the number of character bits, only the lower 7 bits (bits 0 to 6) are valid, so that during a transmission the highest bit (bit 7) is ignored and during reception the highest bit (bit 7) must be cleared to 0.

Baud rate generator control register 0 (BRGC0) is used to set the serial transfer rate.

If a receive error occurs, information about the receive error can be ascertained by reading asynchronous serial interface status register 0 (ASIS0).

(b) Parity types and operations

The parity bit is used to detect bit errors in communication data. Usually, the same type of parity bit is used by the transmitting and receiving sides. When odd parity or even parity is set, errors in the parity bit (the odd-number bit) can be detected. When zero parity or no parity is set, errors are not detected.

(i) Even parity

- During transmission

  The number of bits in transmit data that includes a parity bit is controlled so that there are an even number of character bits whose value is 1. The value of the parity bit is as follows.

  If the transmit data contains an odd number of character bits whose value is 1: the parity bit is “1”
  If the transmit data contains an even number of character bits whose value is 1: the parity bit is “0”

- During reception

  The number of character bits whose value is 1 is counted in the receive data that includes a parity bit, and a parity error occurs when the counted result is an odd number.

(ii) Odd parity

- During transmission

  The number of bits in transmit data that includes a parity bit is controlled so that there is an odd number of character bits whose value is 1. The value of the parity bit is as follows.

  If the transmit data contains an odd number of character bits whose value is 1: the parity bit is “0”
  If the transmit data contains an even number of character bits whose value is 1: the parity bit is “1”

- During reception

  The number of character bits whose value is 1 is counted in the receive data that includes a parity bit, and a parity error occurs when the counted result is an even number.

(iii) Zero parity

During transmission, the parity bit is set to “0” regardless of the transmit data.
During reception, the parity bit is not checked. Therefore, no parity errors will occur regardless of whether the parity bit is a “0” or a “1”.

(iv) No parity

No parity bit is added to the transmit data.
During reception, receive data is regarded as having no parity bit. Since there is no parity bit, no parity errors will occur.

(c) Transmission

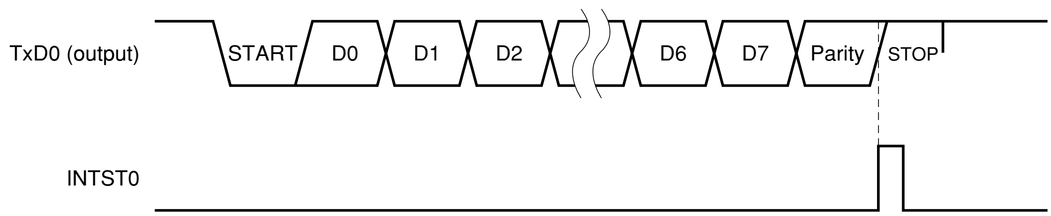
The transmit operation is enabled if bit 7 (TXE0) of asynchronous serial interface mode register 0 (ASIM0) is set to 1, and the transmit operation is started when transmit data is written to transmit shift register 0 (TXS0). A start bit, parity bit, and stop bit(s) are automatically added to the data.

Starting the transmit operation shifts out the data in TXS0, thereby emptying TXS0, after which a transmit completion interrupt request (INTST0) is issued.

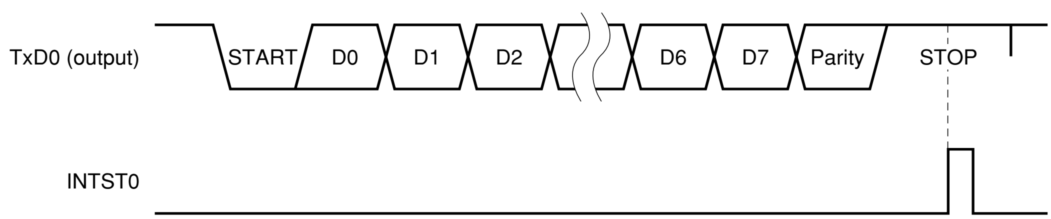
The timing of the transmit completion interrupt request is shown in Figure 16-9.

Figure 16-9. Timing of Asynchronous Serial Interface Transmit Completion Interrupt Request

(i) Stop bit length: 1 bit



(ii) Stop bit length: 2 bits



Caution Do not rewrite asynchronous serial interface mode register 0 (ASIM0) during a transmit operation. Rewriting the ASIM0 register during a transmit operation may disable further transmit operations (in such cases, enter a $\overline{\text{RESET}}$ to restore normal operation).

(d) Reception

★ The receive operation performs level detection.

The receive operation is enabled when "1" is set to bit 6 (RXE0) of asynchronous serial interface mode register 0 (ASIM0), and the input via the RxD0 pin is sampled.

The serial clock specified by baud rate generator control register 0 (BRGC0) is used to sample the RxD0 pin.

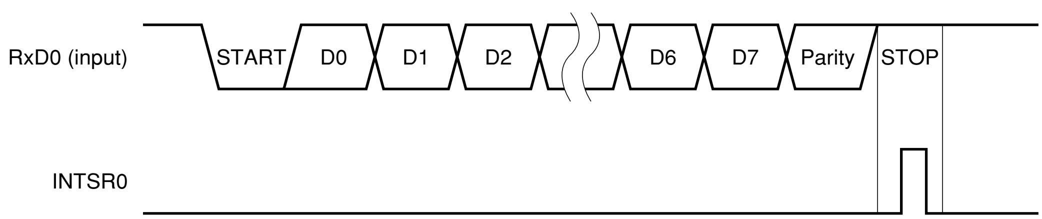
When the RxD0 pin goes low, the 5-bit counter of the baud rate generator begins counting and the start timing signal for data sampling is output when half of the specified baud rate time has elapsed. If sampling the RxD0 pin input at this start timing signal yields a low-level result, a start bit is recognized, after which the 5-bit counter is initialized and starts counting and data sampling begins. After the start bit is recognized, the character data, parity bit, and one-bit stop bit are detected, at which point reception of one data frame is completed.

Once reception of one data frame is completed, the receive data in the shift register is transferred to receive buffer register 0 (RXB0) and INTSR0 (receive completion interrupt request) occurs.

If the RXE0 bit is reset (to 0) during a receive operation, the receive operation is stopped immediately. At this time, the contents of RXB0 and ASIS0 do not change, nor does INTSR0 or INTSER0 (receive error interrupt request) occur.

Figure 16-10 shows the timing of the asynchronous serial interface receive completion interrupt request.

Figure 16-10. Timing of Asynchronous Serial Interface Receive Completion Interrupt Request



★ **Caution If the receive operation is enabled with the RxD0 pin input at the low level, the receive operation is immediately started. Make sure the RxD0 pin input is at the high level before enabling the receive operation.**

(e) Receive errors

Three types of errors can occur during a receive operation: a parity error, framing error, or overrun error. If, as the result of data reception, an error flag is set in asynchronous serial interface status register 0 (ASIS0), a receive error interrupt request (INTSER0) will occur. Receive error interrupt requests are generated before the receive completion interrupt request (INTSR0). Table 16-3 lists the causes behind receive errors.

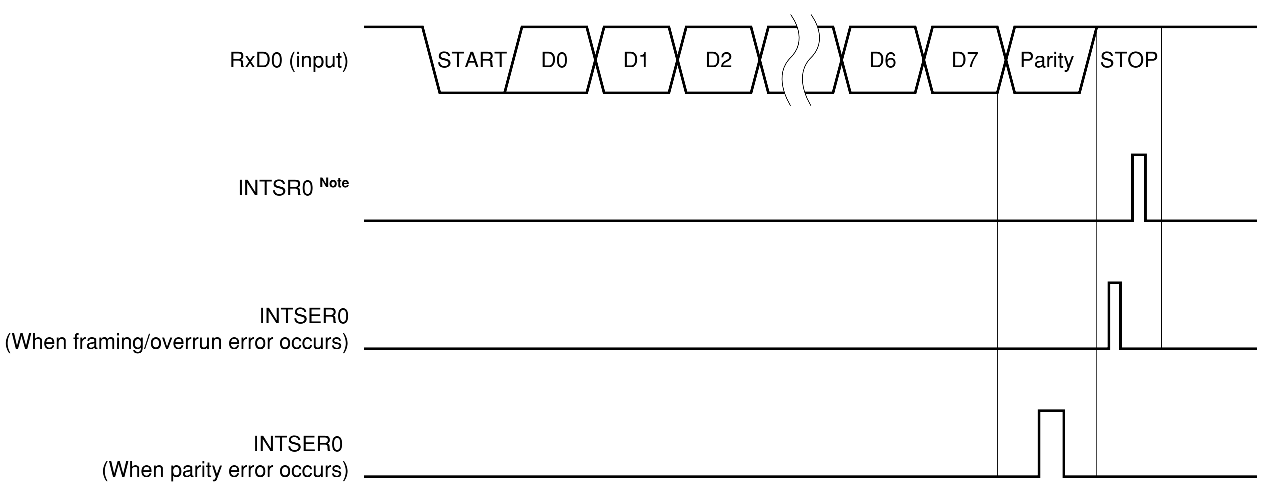
As part of receive error interrupt request (INTSER0) servicing, the contents of ASIS0 can be read to determine which type of error occurred during the receive operation (see **Table 16-3** and **Figure 16-11**).

The contents of ASIS0 are reset (to 0) when receive buffer register 0 (RXB0) is read or when the next data is received (if the next data contains an error, its error flag will be set).

Table 16-3. Causes of Receive Errors

Receive Error	Cause	ASIS0 Value
Parity error	Parity specified does not match parity of receive data	04H
Framing error	Stop bit was not detected	02H
Overrun error	Reception of the next data was completed before data was read from receive buffer register 0 (RXB0)	01H

Figure 16-11. Receive Error Timing



Note Even if a receive error occurs when the ISRM0 bit has been set (1), INTSR0 does not occur.

Cautions 1. The contents of asynchronous serial interface status register 0 (ASIS0) are reset (to 0) when receive buffer register 0 (RXB0) is read or when the next data is received. To obtain information about the error, be sure to read the contents of ASIS0 before reading RXB0.

★ **2. Be sure to read the contents of receive buffer register 0 (RXB0) after the receive completion interrupt request has occurred even when a receive error has occurred. If RXB0 is not read after the receive completion interrupt request has occurred, overrun errors will occur during the next data receive operations and the receive error status will remain until the contents of RXB0 are read.**

16.4.3 Infrared data transfer mode

In infrared data transfer mode, pulses can be output and received in the data format shown in (2).

★ **(1) Registers to be used**

- Asynchronous serial interface mode register 0 (ASIM0)
- Asynchronous serial interface status register 0 (ASIS0)
- Baud rate generator control register 0 (BRGC0)
- Port mode register 2 (PM2)
- Port 2 (P2)

(a) Asynchronous serial interface mode register 0 (ASIM0)

ASIM0 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIM0 to 00H.

Address: FFA0H After reset: 00H R/W

Symbol	<7>	<6>	5	4	3	2	1	0
ASIM0	TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0

TXE0	RXE0	Operation mode	RxD0/P23 pin function	TxD0/P24 pin function
0	0	Operation stop	Port function (P23)	Port function (P24)
0	1	UART mode (receive only)	Serial function (RxD0)	
1	0	UART mode (transmit only)	Port function (P23)	Serial function (TxD0)
1	1	UART mode (transmit and receive)	Serial function (RxD0)	

PS01	PS00	Parity bit specification
0	0	No parity
0	1	Zero parity always added during transmission No parity detection during reception (parity errors do not occur)
1	0	Odd parity
1	1	Even parity

CL0	Character length specification
0	7 bits
1	8 bits

SL0	Stop bit length specification for transmit data
0	1 bit
1	2 bits

ISRM0	Receive completion interrupt control when error occurs
0	Receive completion interrupt request is issued when an error occurs
1	Receive completion interrupt request is not issued when an error occurs

IRDAM0	Operation specified for infrared data transfer mode[Note 1]
0	UART (transmit/receive) mode
1	Infrared data transfer (transmit/receive) mode[Note 2]

Notes 1. The UART/infrared data transfer mode specification is controlled by TXE0 and RXE0.

2. When using infrared data transfer mode, be sure to set baud rate generator control register 0 (BRGC0) to “10H”.

★ **Caution Before writing different data to ASIM0, stop operation.**

(b) Asynchronous serial interface status register 0 (ASIS0)

ASIS0 can be read by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIS0 to 00H.

Address: FFA1H After reset: 00H R

Symbol	7	6	5	4	3	2	1	0
ASIS0	0	0	0	0	0	PE0	FE0	OVE0

PE0	Parity error flag
0	No parity error
1	Parity error (Transmit data parity not matched)

FE0	Framing error flag
0	No framing error
1	Framing error[Note 1] (Stop bit not detected)

OVE0	Overrun error flag
0	No overrun error
1	Overrun error[Note 2] (Next receive operation was completed before data was read from receive buffer register 0 (RXB0))

Notes 1. Even if the stop bit length is set to two bits by setting bit 2 (SL0) of asynchronous serial interface mode register 0 (ASIM0), stop bit detection during a receive operation only applies to a stop bit length of 1 bit.

2. When an overrun error has occurred, further overrun errors will continue to occur until the contents of receive buffer register 0 (RXB0) are read.

(c) Baud rate generator control register 0 (BRGC0)

BRGC0 is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BRGC0 to 00H.

Address: FFA2H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
BRGC0	0	TPS02	TPS01	TPS00	MDL03	MDL02	MDL01	MDL00

(f_X = 8.38 MHz)

TPS02	TPS01	TPS00	Source clock selection for 5-bit counter	n
0	0	0	External clock input to ASCK0	0
0	0	1	$f_X/2$	1
0	1	0	$f_X/2^2$	2
0	1	1	$f_X/2^3$	3
1	0	0	$f_X/2^4$	4
1	0	1	$f_X/2^5$	5
1	1	0	$f_X/2^6$	6
1	1	1	$f_X/2^7$	7

MDL03	MDL02	MDL01	MDL00	Output clock selection for baud rate generator	k
0	0	0	0	$f_{SCK0}/16$	0
0	0	0	1	$f_{SCK0}/17$	1
0	0	1	0	$f_{SCK0}/18$	2
0	0	1	1	$f_{SCK0}/19$	3
0	1	0	0	$f_{SCK0}/20$	4
0	1	0	1	$f_{SCK0}/21$	5
0	1	1	0	$f_{SCK0}/22$	6
0	1	1	1	$f_{SCK0}/23$	7
1	0	0	0	$f_{SCK0}/24$	8
1	0	0	1	$f_{SCK0}/25$	9
1	0	1	0	$f_{SCK0}/26$	10
1	0	1	1	$f_{SCK0}/27$	11
1	1	0	0	$f_{SCK0}/28$	12
1	1	0	1	$f_{SCK0}/29$	13
1	1	1	0	$f_{SCK0}/30$	14
1	1	1	1	Setting prohibited	–

Cautions 1. Writing to BRGC0 during a communication operation may cause abnormal output from the baud rate generator and disable further communication operations. Therefore, do not write to BRGC0 during a communication operation.

2. Set BRGC0 to 10H when using in infrared data transfer mode.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{SCK0}: Source clock for 5-bit counter

3. n: Value set via TPS00 to TPS02 ($0 \leq n \leq 7$)

4. k: Value set via MDL00 to MDL03 ($0 \leq k \leq 14$)

(d) Port mode register 2 (PM2)

Port mode register 2 is used to set input/output of port 2 in 1-bit units.

To use the P24/TxD0 pin as a serial data output, clear PM24 and the output latch of P24 to 0.

To use the P23/RxD0 pin as a serial data input, and the P25/ASCK0 pin as a clock input, set PM23 and PM25 to 1. At this time, the output latches of P23 and P25 can be either 0 or 1.

PM2 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM2 to FFH.

Address: FF22H After reset: FFH R/W

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

PM2n	I/O mode selection of P2n pin (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

(2) Data format

Figure 16-12 compares the data format used in UART mode with that used in infrared data transfer mode.

The IR (infrared) frame corresponds to the bit string of the UART frame, which consists of pulses that include a start bit, eight data bits, and a stop bit.

The length of the electrical pulses that are used to transmit and receive in an IR frame is 3/16 the length of the cycle time for one bit (i.e., the “bit time”). This pulse (whose width is 3/16 the length of one bit time) rises from the middle of the bit time (see the figure below).

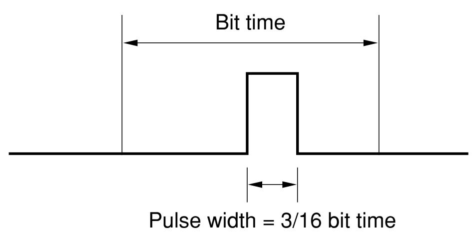
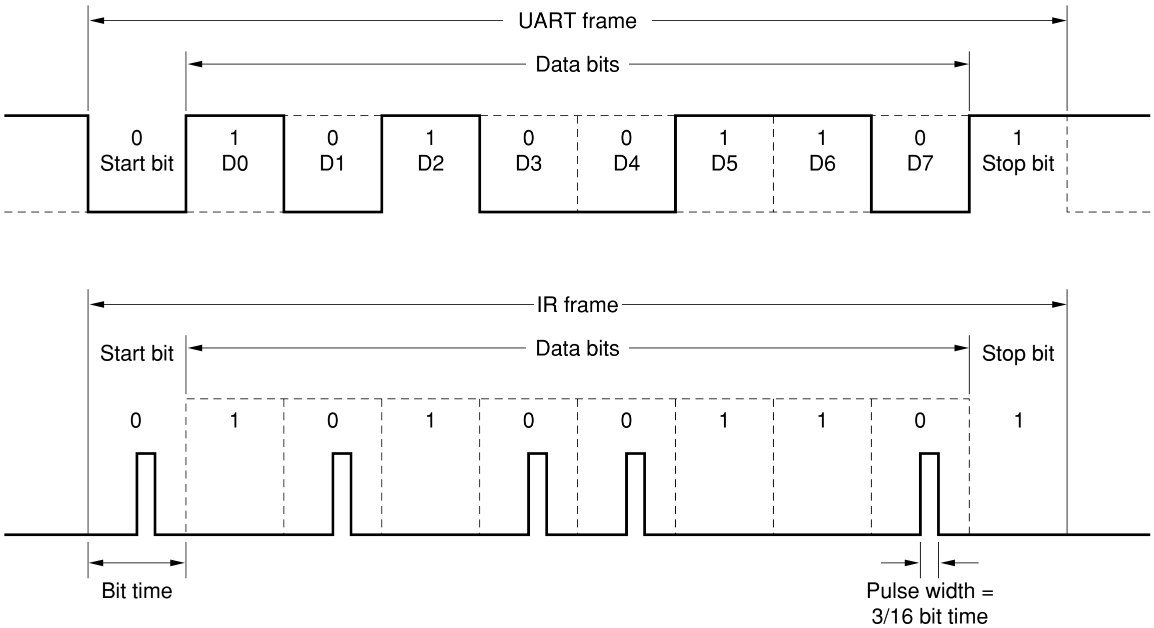


Figure 16-12. Data Format Comparison Between Infrared Data Transfer Mode and UART Mode



★ **(3) Relationship between main system clock and baud rate**

Table 16-4 shows the relationship between the main system clock and the baud rate.

Table 16-4. Relationship Between Main System Clock and Baud Rate

	fx = 8.3886 MHz	fx = 8.000 MHz	fx = 7.3728 MHz	fx = 5.000 MHz	fx = 4.1943 MHz
Baud rate	131031 bps	125000 bps	115200 bps	78125 bps	65536 bps

(4) Bit rate and pulse width

Table 16-5 lists the bit rate, bit rate error tolerance, and pulse width values.

Table 16-5. Bit Rate and Pulse Width Values

Bit Rate (kbps)	Bit Rate Error Tolerance (% of Bit Rate)	Pulse Width Minimum Value (μs)**Note 2**	3/16 Pulse Width <Nominal Value> (μs)	Maximum Pulse Width (μs)
115.2**Note 1**	+/– 0.87	1.41	1.63	2.71

Notes 1. Operation with fx = 7.3728 MHz

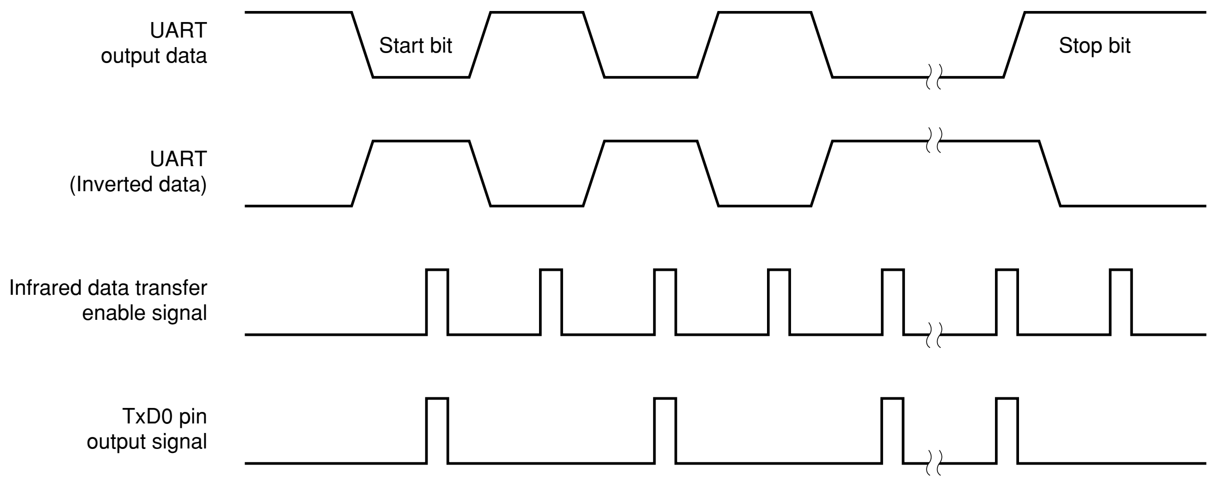
2. When a digital noise eliminator is used in a microcontroller operating at 1.41 MHz or above.

Caution When using baud rate generator control register 0 (BRGC0) in infrared data transfer mode, set it to 10H.

Remark fx: Main system clock oscillation frequency

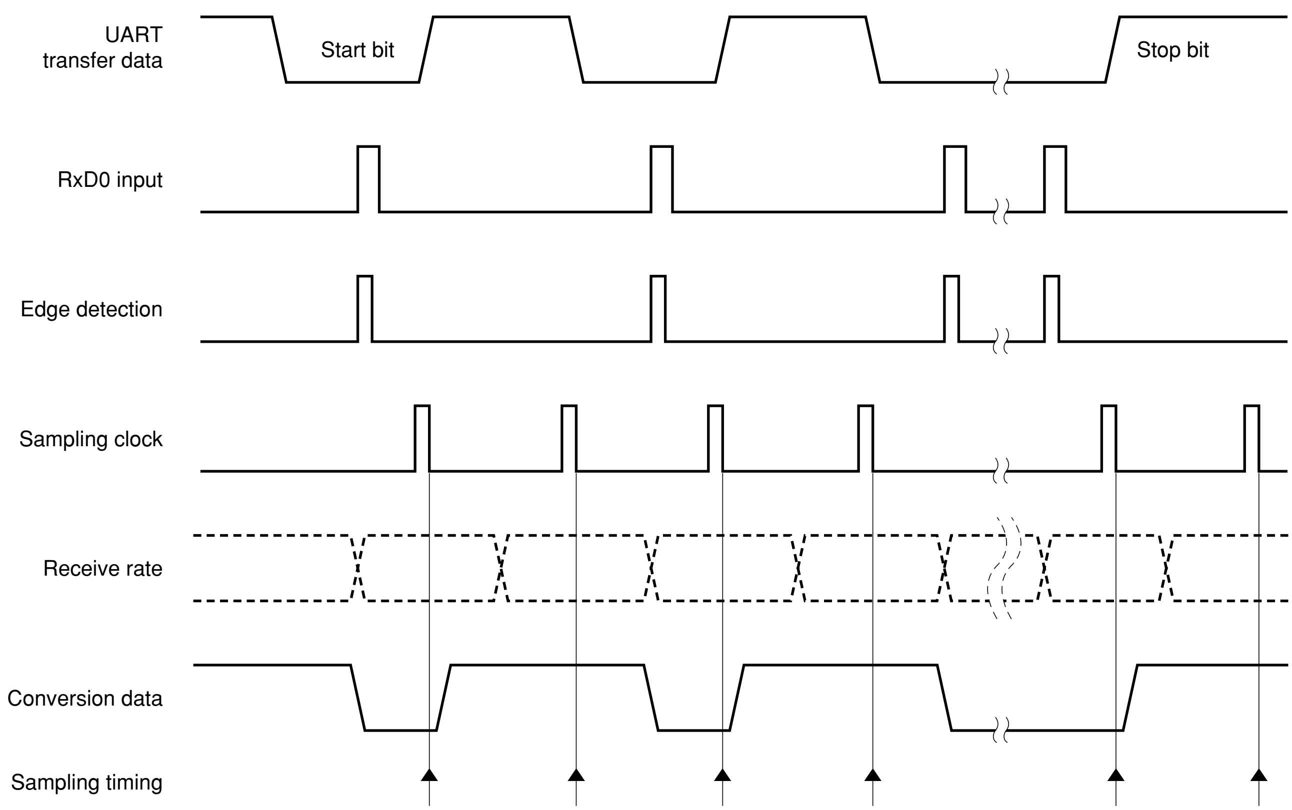
(5) Input data and internal signals

- **Transmit operation timing**



- **Receive operation timing**

  Data reception is delayed for one-half of the specified baud rate.



★ Table 16-6. Register Settings

(1) Operation stop mode

ASIM0								BRGC0							PM23	P23	PM24	P24	Pin Function		Operation
TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0	TPS02	TPS01	TPS00	MDL03	MDL02	MDL01	MDL00					P23/RxD0	P24/TxD0	Mode
0	0	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	P23	P24	Stop
Other than above																			Setting prohibited		

(2) Asynchronous serial interface (UART) mode

ASIM0								BRGC0							PM23	P23	PM24	P24	Pin Function		Operation
TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0	TPS02	TPS01	TPS00	MDL03	MDL02	MDL01	MDL00					P23/RxD0	P24/TxD0	Mode
0	1	0/1	0/1	0/1	×	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	1	×	×	×	RxD0	P24	Receive
1	0	0/1	0/1	0/1	0/1	×	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	×	×	0	0	P23	TxD0	Transmit
1	1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	1	×	0	0	RxD0	TxD0	Transmit /receive
Other than above																			Setting prohibited		

(3) Infrared data transfer mode

ASIM0								BRGC0							PM23	P23	PM24	P24	Pin Function		Operation
TXE0	RXE0	PS01	PS00	CL0	SL0	ISRM0	IRDAM0	TPS02	TPS01	TPS00	MDL03	MDL02	MDL01	MDL00					P23/RxD0	P24/TxD0	Mode
0	1	0/1	0/1	0/1	×	0/1	1	0	0	1	0	0	0	0	1	×	×	×	RxD0	P24	Receive
1	0	0/1	0/1	0/1	0/1	×	1	0	0	1	0	0	0	0	×	×	0	0	P23	TxD0	Transmit
1	1	0/1	0/1	0/1	0/1	0/1	1	0	0	1	0	0	0	0	1	×	0	0	RxD0	TxD0	Transmit /receive
Other than above																			Setting prohibited		

Caution When using the infrared data transfer mode, set the BRGC0 register to 10H.

Remark ×: Don’t care, ASIM0: Asynchronous serial interface mode register 0
BRGC0: Baud rate generator control register 0, PMxx: Port mode register, Pxx: Output latch of port

The μPD780024A, 780034A Subseries products have two 3-wire serial I/O mode channels (SIO30, SIO31). The μPD780024AY, 780034AY Subseries products have one 3-wire serial I/O mode channel (SIO30).

17.1 Functions of Serial Interfaces SIO30 and SIO31

Serial interface SIO3n has the following two modes.

(1) Operation stop mode

This mode is used when serial transfers are not performed. For details, see **17.4.1 Operation stop mode**.

(2) 3-wire serial I/O mode (fixed as MSB first)

This is an 8-bit data transfer mode using three lines: a serial clock line ($\overline{\text{SCK3n}}$), serial output line (SO3n), and serial input line (SI3n).

Since simultaneous transmit and receive operations are enabled in 3-wire serial I/O mode, the processing time for data transfers is reduced.

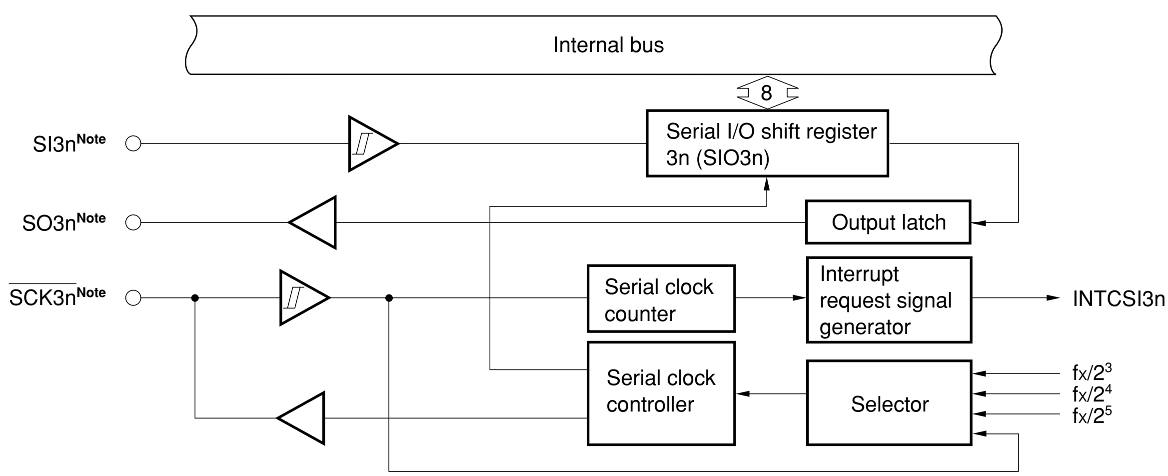
The first bit of the serial transferred 8-bit data is fixed as the MSB.

3-wire serial I/O mode is useful for connection to a peripheral IC incorporating a clocked serial interface, a display controller, etc. For details, see **17.4.2 3-wire serial I/O mode**.

Figure 17-1 shows a block diagram of serial interface SIO3n.

Remark n = 0, 1: μPD780024A, 780034A Subseries
n = 0: μPD780024AY, 780034AY Subseries

★ **Figure 17-1. Block Diagram of Serial Interface SIO3n**



Note SI30, SO30, and $\overline{\text{SCK30}}$ pins are alternate with P20, P21, and P22 pins. SI31, SO31, and $\overline{\text{SCK31}}$ pins are alternate with P34, P35, and P36 pins.

17.2 Configuration of Serial Interfaces SIO30 and SIO31

Serial interface SIO3n consists of the following hardware.

Table 17-1. Configuration of Serial Interface SIO3n

Item	Configuration
Register	Serial I/O shift register 3n (SIO3n)
Control registers	Serial operation mode register 3n (CSIM3n) Port mode registers 2, 3 (PM2, PM3) Ports 2, 3 (P2, P3)

(1) Serial I/O shift register 3n (SIO3n)

This is an 8-bit register that performs parallel-serial conversion and serial transmit/receive (shift operations) synchronized with the serial clock.
SIO3n is set by an 8-bit memory manipulation instruction.
When 1 is set to bit 7 (CSIE3n) of serial operation mode register 3n (CSIM3n), a serial operation can be started by writing data to or reading data from SIO3n.
When transmitting, data written to SIO3n is output to the serial output (SO3n).
When receiving, data is read from the serial input (SI3n) and written to SIO3n.
$\overline{\text{RESET}}$ input makes SIO3n undefined.

Caution Do not access SIO3n during a transfer operation unless the access is triggered by a transfer start (read operation is disabled when MODEn = 0 and write operation is disabled when MODEn = 1).

Remark n = 0, 1: μPD780024A, 780034A Subseries
n = 0: μPD780024AY, 780034AY Subseries

17.3 Registers to Control Serial Interfaces SIO30 and SIO31

Serial interface SIO3n is controlled by the following three registers.

- Serial operation mode register 3n (CSIM3n)
- Port mode registers 2, 3 (PM2, PM3)
- Ports 2, 3 (P2, P3)

(1) Serial operation mode register 3n (CSIM3n)

This register is used to enable or disable SIO3n's serial clock, operation modes, and specific operations.
CSIM3n is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears CSIM3n to 00H.

Remark n = 0, 1: μPD780024A, 780034A Subseries
n = 0: μPD780024AY, 780034AY Subseries

Figure 17-2. Format of Serial Operation Mode Register 30 (CSIM30)

Address: FFB0H After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CSIM30	CSIE30	0	0	0	0	MODE0	SCL301	SCL300

CSIE30	Enable/disable specification for SIO30		
	Shift register operation	Serial counter	Port
0	Operation stopped	Clear	Port function[Note 1]
1	Operation enabled	Count operation enable	Serial function + port function[Note 2]

MODE0	Transfer operation modes and flags		
	Operation mode	Transfer start trigger	SO30/P21 pin function
0	Transmit/transmit and receive mode	Write to SIO30	SO30
1	Receive-only mode	Read from SIO30	P21[Note 3]

★

SCL301	SCL300	Clock selection		
			f_X = 8.38 MHz	f_X = 12 MHz[Note 4]
0	0	External clock input to $\overline{SCK30}$	–	–
0	1	$f_X/2^3$	1.04 MHz	1.50 MHz
1	0	$f_X/2^4$	523 kHz	750 kHz
1	1	$f_X/2^5$	261 kHz	375 kHz

★ **Notes 1.** When CSIE30 = 0 (SIO30 operation stopped status), the SI30, SO30, and $\overline{SCK30}$ pins can be used as port functions.

2. When CSIE30 = 1 (SIO30 operation enabled status), the SI30 pin can be used as a port pin if only the transmit function is used, and the SO30 pin can be used as a port pin if only the receive-only mode is used.

★ **3.** When MODE0 = 1 (receive-only mode), the SO30 pin can be used for port functions.

4. Expanded-specification products of μPD780024A, 780034A Subseries only.

★ **Caution Do not rewrite the value of CSIM30 during transfer. However, CSIE30 can be rewritten using a 1-bit memory manipulation instruction.**

Remark f_X: Main system clock oscillation frequency

Figure 17-3. Format of Serial Operation Mode Register 31 (CSIM31)

Address: FFB8H After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CSIM31	CSIE31	0	0	0	0	MODE1	SCL311	SCL310

CSIE31	Enable/disable specification for SIO31		
	Shift register operation	Serial counter	Port
0	Operation stopped	Clear	Port function[Note 1]
1	Operation enabled	Count operation enable	Serial function + port function[Note 2]

MODE1	Transfer operation modes and flags		
	Operation mode	Transfer start trigger	SO31/P35 pin function
0	Transmit/transmit and receive mode	Write to SIO31	SO31
1	Receive-only mode	Read from SIO31	P35[Note 3]

★

SCL311	SCL310	Clock selection		
			f_X = 8.38 MHz	f_X = 12 MHz[Note 4]
0	0	External clock input to $\overline{SCK31}$	–	–
0	1	$f_X/2^3$	1.04 MHz	1.50 MHz
1	0	$f_X/2^4$	523 kHz	750 kHz
1	1	$f_X/2^5$	261 kHz	375 kHz

★ **Notes 1.** When CSIE31 = 0 (SIO31 operation stopped status), the SI31, SO31, and $\overline{SCK31}$ pins can be used as port functions.

2. When CSIE31 = 1 (SIO31 operation enabled status), the SI31 pin can be used as a port pin if only the transmit function is used, and the SO31 pin can be used as a port pin if only the receive-only mode is used.

★ **3.** When MODE1 = 1 (receive-only mode), the SO31 pin can be used for port functions.

4. Expanded-specification products of μPD780024A, 780034A Subseries only.

★ **Caution Do not rewrite the value of CSIM31 during transfer. However, CSIE31 can be rewritten using a 1-bit memory manipulation instruction.**

Remark f_X: Main system clock oscillation frequency

★ **(2) Port mode registers 2, 3 (PM2, PM3)**

These registers set the input/output of ports 2 and 3 in 1-bit units.

To use the P21/SO30 and P35/SO31 pins as serial data output, and the P22/$\overline{SCK30}$ and P36/$\overline{SCK31}$ pins as clock output, clear PM21, PM35, PM22, PM36, and the output latches of P21, P35, P22, and P36 to 0.

To use the P20/SI30 and P34/SI31 pins as serial data input, and the P22/$\overline{SCK30}$ and P36/$\overline{SCK31}$ pins as clock input, set PM20, PM34, PM22, and PM36 to 1.

At this time, the output latches of P20, P34, P22, and P36 can be either 0 or 1.

PM2 and PM3 are set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{RESET}$ input sets PM2 and PM3 to FFH.

Figure 17-4. Format of Port Mode Register 2 (PM2)

Address: FF22H After reset: FFH R/W

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

PM2n	I/O mode selection of P2n pin (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

Figure 17-5. Format of Port Mode Register 3 (PM3)

Address: FF23H After reset: FFH R/W

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

PM3n	I/O mode selection of P3n pin (n = 0 to 6)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

17.4 Operations of Serial Interfaces SIO30 and SIO31

This section explains the two modes of serial interface SIO3n.

17.4.1 Operation stop mode

Because the serial transfer is not performed during this mode, the power consumption can be reduced.
In addition, pins can be used as normal I/O ports.

(1) Register settings

Operation stop mode is set by serial operation mode register 3n (CSIM3n).
CSIM3n is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears CSIM3n to 00H.

Address: FFB0H (SIO30), FFB8H (SIO31) After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CSIM3n	CSIE3n	0	0	0	0	MODEn	SCL3n1	SCL3n0

CSIE3n	Enable/disable specification for SIO3n		
	Shift register operation	Serial counter	Port
0	Operation disabled	Clear	Port function[Note]

Note When CSIE3n = 0 (SIO3n operation stop status), the pins SI3n, SO3n, and $\overline{\text{SCK3n}}$ can be used for port functions.

Remark n = 0, 1

17.4.2 3-wire serial I/O mode

The 3-wire serial I/O mode can be used when connecting a peripheral IC incorporating a clocked serial interface, a display controller, etc.

This mode executes data transfers via three lines: a serial clock line ($\overline{\text{SCK3n}}$), serial output line (SO3n), and serial input line (SI3n).

(1) Registers to be used

- Serial operation mode register 3n (CSIM3n)
- Port mode registers 2, 3 (PM2, PM3)
- Ports 2, 3 (P2, P3)

(a) Serial operation mode register 3n (CSIM3n)

CSIM3n is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM3n to 00H.

Address: FFB0H (SIO30), FFB8H (SIO31) After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CSIM3n	CSIE3n	0	0	0	0	MODEn	SCL3n1	SCL3n0

CSIE3n	Enable/disable specification for SIO3n		
	Shift register operation	Serial counter	Port
0	Operation stopped	Clear	Port function Note 1
1	Operation enabled	Count operation enable	Serial function + port function Note 2

MODEn	Transfer operation modes and flags		
	Operation mode	Transfer start trigger	SO3n pin function
0	Transmit/transmit and receive mode	Write to SIO3n	SO3n
1	Receive-only mode	Read from SIO3n	Port function Note 3

★

SCL3n1	SCL3n0	Clock selection		
			fx = 8.38 MHz	fx = 12 MHz Note 4
0	0	External clock input to $\overline{\text{SCK3n}}$	–	–
0	1	$f_X/2^3$	1.04 MHz	1.50 MHz
1	0	$f_X/2^4$	523 kHz	750 kHz
1	1	$f_X/2^5$	261 kHz	375 kHz

★ **Notes 1.** When CSIE3n = 0 (SIO3n operation stopped status), the SI3n, SO3n, and $\overline{\text{SCK3n}}$ pins can be used as port functions.

2. When CSIE3n = 1 (SIO3n operation enabled status), the SI3n pin can be used as a port pin if only the transmit function is used, and the SO3n pin can be used as a port pin if only the receive-only mode is used.

★ **3.** When MODEn = 1 (receive-only mode), the SO3n pin can be used for port functions.

4. Expanded-specification products of μPD780024A, 780034A Subseries only.

★ **Caution Do not rewrite the value of CSIM3n during transfer. However, CSIE3n can be rewritten using a 1-bit memory manipulation instruction.**

Remarks 1. fx: Main system clock oscillation frequency

2. n = 0, 1: μPD780024A, 780034A Subseries

n = 0: μPD780024AY, 780034AY Subseries

★ **(b) Port mode registers 2, 3 (PM2, PM3)**

These registers set the input/output of ports 2 and 3 in 1-bit units.

To use the P21/SO30 and P35/SO31 pins as serial data output, and the P22/$\overline{\text{SCK30}}$ and P36/$\overline{\text{SCK31}}$ pins as clock output, clear PM21, PM35, PM22, PM36, and the output latches of P21, P35, P22, and P36 to 0.

To use the P20/SI30 and P34/SI31 pins as serial data input, and the P22/$\overline{\text{SCK30}}$ and P36/$\overline{\text{SCK31}}$ pins as clock input, set PM20, PM34, PM22, and PM36 to 1.

At this time, the output latches of P20, P34, P22, and P36 can be either 0 or 1.

PM2 and PM3 are set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM2 and PM3 to FFH.

Address: FF22H After reset: FFH R/W

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

PM2n	I/O mode selection of P2n pin (n = 0 to 5)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

Address: FF23H After reset: FFH R/W

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

PM3n	I/O mode selection of P3n pin (n = 0 to 6)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

(2) Transfer start

A serial transfer starts when the following two conditions have been satisfied and transfer data has been set (or read) to serial I/O shift register 3n (SIO3n).

<Transfer start conditions>

- SIO3n operation control bit (CSIE3n) = 1
- After an 8-bit serial transfer, either the internal serial clock is stopped or $\overline{\text{SCK3n}}$ is set to high level.

<Transfer start timing>

- Transmit/transmit and receive mode (MODEn = 0)
  Transfer starts when writing to SIO3n.
- Receive-only mode (MODEn = 1)
  Transfer starts when reading from SIO3n.

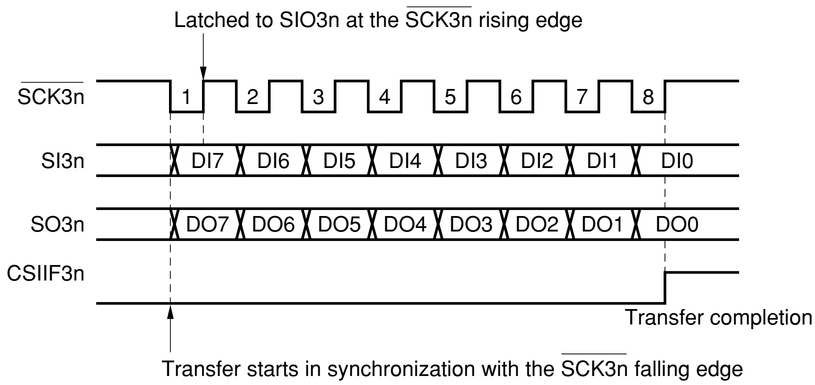
Caution After data has been written to SIO3n, transfer will not start even if the CSIE3n bit value is set to 1.

(3) Communication operations

In the 3-wire serial I/O mode, data is transmitted and received in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Serial I/O shift register 3n (SIO3n) is shifted in synchronization with the falling edge of the serial clock. Transmit data is held in the SO3n latch and is output from the SO3n pin. Data that is received via the SI3n pin in synchronization with the rising edge of the serial clock is latched to SIO3n.

Figure 17-6. Timing of 3-Wire Serial I/O Mode



(4) Transfer complete

Completion of an 8-bit transfer automatically stops the serial transfer operation and the interrupt request flag (CSIIF3n) is set.

Remark n = 0, 1: μPD780024A, 780034A Subseries
n = 0: μPD780024AY, 780034AY Subseries

★ Table 17-2. Register Settings

(1) Operation stop mode

• Serial interface SIO30

CSIM30				PM20	P20	PM21	P21	PM22	P22	Pin Function			Operation
CSIE30	MODE0	SCL301	SCL300							P20/SI30	P21/SO30	P22/$\overline{\text{SCK30}}$	
0	×	×	×	×	×	×	×	×	×	P20	P21	P22	Stop
Other than above										Setting prohibited			

• Serial interface SIO31

CSIM31				PM34	P34	PM35	P35	PM36	P36	Pin Function			Operation
CSIE31	MODE1	SCL311	SCL310							P34/SI31	P35/SO31	P36/$\overline{\text{SCK31}}$	
0	×	×	×	×	×	×	×	×	×	P34	P35	P36	Stop
Other than above										Setting prohibited			

(2) 3-wire serial I/O mode

• Serial interface SIO30

CSIM30				PM20	P20	PM21	P21	PM22	P22	Pin Function			Operation
CSIE30	MODE0	SCL301	SCL300							P20/SI30	P21/SO30	P22/$\overline{\text{SCK30}}$	
1	1	0	0	1	×	×	×	1	×	SI30	P21	$\overline{\text{SCK30}}$ input	Slave receive
1	0	0	0	1	×	0	0	1	×	SI30[Note]	SO30	$\overline{\text{SCK30}}$ input	Slave transmit/transmit and receive
1	1	Other than above		1	×	×	×	0	0	SI30	P21	$\overline{\text{SCK30}}$ output	Master receive
1	0			1	×	0	0	0	0	SI30[Note]	SO30	$\overline{\text{SCK30}}$ output	Master transmit/transmit and receive
Other than above										Setting prohibited			

• Serial interface SIO31

CSIM31				PM34	P34	PM35	P35	PM36	P36	Pin Function			Operation
CSIE31	MODE1	SCL311	SCL310							P21/SI31	P21/SO31	P22/$\overline{\text{SCK31}}$	
1	1	0	0	1	×	×	×	1	×	SI31	P35	$\overline{\text{SCK31}}$ input	Slave receive
1	0	0	0	1	×	0	0	1	×	SI31[Note]	SO31	$\overline{\text{SCK31}}$ input	Slave transmit/transmit and receive
1	1	Other than above		1	×	×	×	0	0	SI31	P35	$\overline{\text{SCK31}}$ output	Master receive
1	0			1	×	0	0	0	0	SI31[Note]	SO31	$\overline{\text{SCK31}}$ output	Master transmit/transmit and receive
Other than above										Setting prohibited			

Note When using for transmission only, it can be used as P20 or P34.

Remark ×: Don't care, CSIM30, CSIM31: Serial operation mode registers 30, 31, PM××: Port mode register, P××: Output latch of port

18.1 Functions of Serial Interface IIC0

Serial interface IIC0 has the following two modes.

(1) Operation stop mode

This mode is used when serial transfers are not performed. It can therefore be used to reduce power consumption.

(2) I^2C bus mode (multimaster supported)

This mode is used for 8-bit data transfers with several devices via two lines: a serial clock (SCL0) line and a serial data bus (SDA0) line.

The transfer rate is as follows.

- 97.5 kHz (standard mode) or 350 kHz (high-speed mode): When operated at f_X = 8.38 MHz

This mode complies with the I^2C bus format and can output “start condition”, “data”, and “stop condition” data segments when transmitting via the serial data bus. These data segments are automatically detected by hardware during reception.

Since SCL0 and SDA0 are open-drain outputs, the IIC0 requires pull-up resistors for the serial clock line (SCL0) and the serial data bus line (SDA0).

Figure 18-1 shows a block diagram of serial interface IIC0.

★ **Figure 18-1. Block Diagram of Serial Interface IIC0**

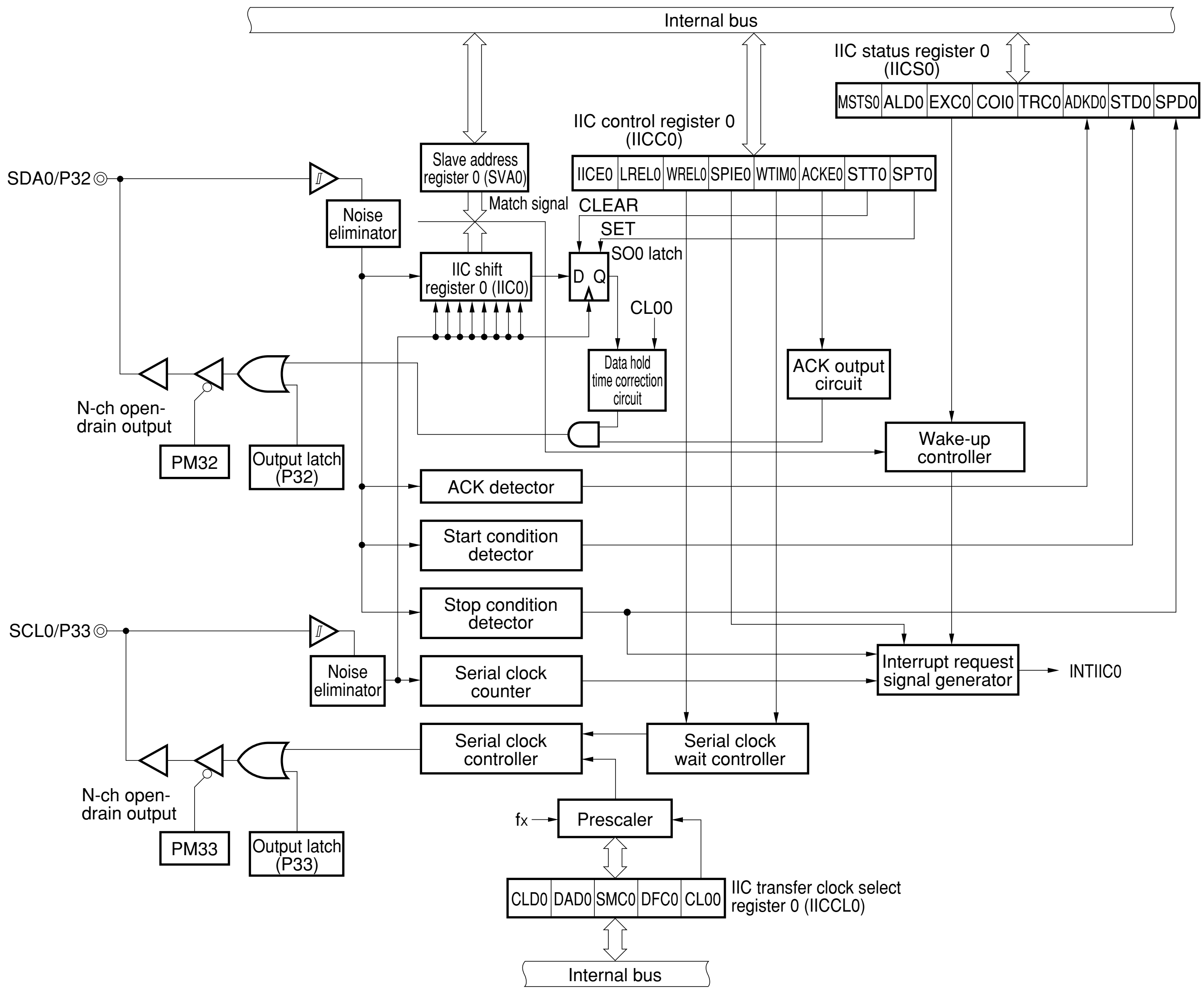
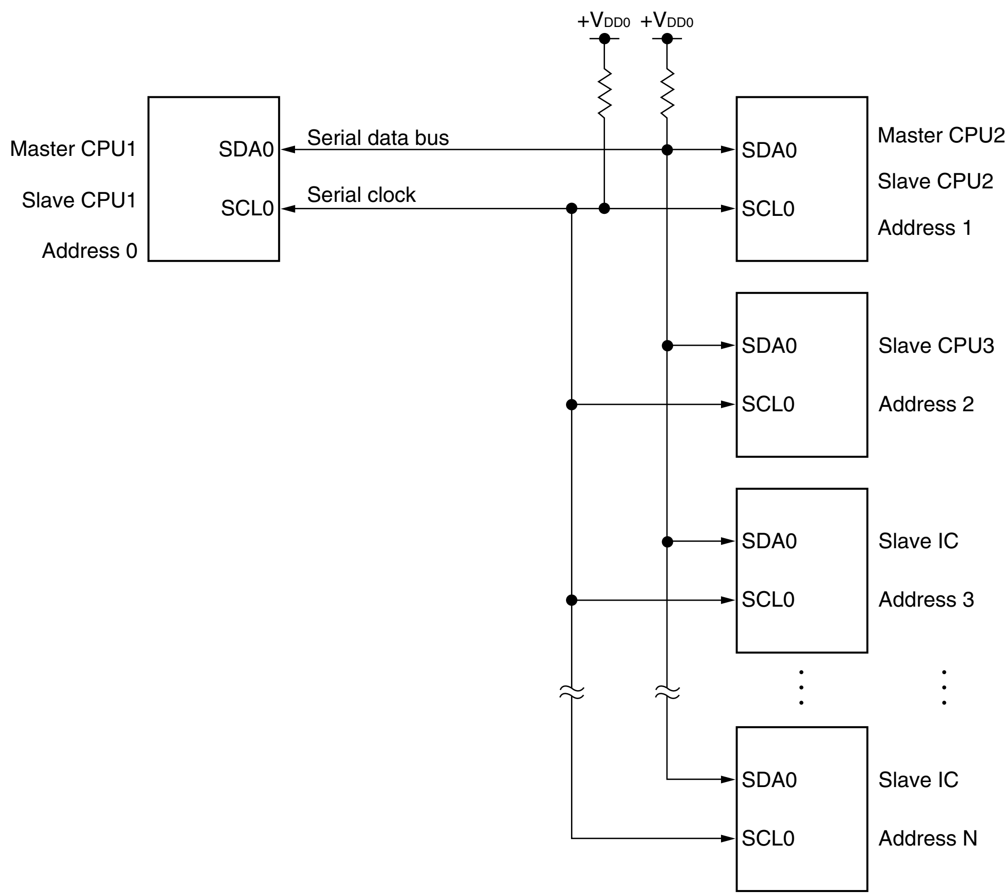


Figure 18-2 shows a serial bus configuration example.

Figure 18-2. Serial Bus Configuration Example Using I²C Bus



18.2 Configuration of Serial Interface IIC0

Serial interface IIC0 includes the following hardware.

★ **Table 18-1. Configuration of Serial Interface IIC0**

Item	Configuration
Registers	IIC shift register 0 (IIC0) Slave address register 0 (SVA0)
Control registers	IIC control register 0 (IICC0) IIC status register 0 (IICS0) IIC transfer clock select register 0 (IICCL0) Port mode register 3 (PM3) Port 3 (P3)

★ **(1) IIC shift register 0 (IIC0)**

IIC0 is used to convert 8-bit serial data to 8-bit parallel data and vice versa in synchronization with the serial clock.
IIC0 can be used for both transmission and reception.
Write and read operations to IIC0 are used to control the actual transmit and receive operations.
IIC0 is set by an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears IIC0 to 00H.

Figure 18-3. Format of IIC Shift Register 0 (IIC0)

Address: FF1FH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
IIC0								

Caution Do not write data to IIC0 during data transfer.

★ **(2) Slave address register 0 (SVA0)**

This register sets local addresses when in slave mode.
SVA0 is set by an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears SVA0 to 00H.

Figure 18-4. Format of Slave Address Register 0 (SVA0)

Address: FFABH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
SVA0								0[Note]

Note Bit 0 is fixed to 0.

(3) SO0 latch

The SO0 latch is used to retain the SDA0 pin's output level.

(4) Wake-up controller

This circuit generates an interrupt request when the address received by this register matches the address value set to slave address register 0 (SVA0) or when an extension code is received.

(5) Prescaler

This selects the sampling clock to be used.

(6) Serial clock counter

This counter counts the serial clocks that are output or input during transmit/receive operations and is used to verify that 8-bit data was transmitted or received.

(7) Interrupt request signal generator

This circuit controls the generation of interrupt request signals (INTIIC0).
An I^2C interrupt request is generated following either of two triggers.

- Falling of eighth or ninth clock of the serial clock (set by WTIM0 bit[Note])
- Interrupt request generated when a stop condition is detected (set by SPIE0 bit[Note])

Note WTIM0 bit: Bit 3 of IIC control register 0 (IICC0)
SPIE0 bit: Bit 4 of IIC control register 0 (IICC0)

(8) Serial clock controller

In master mode, this circuit generates the clock output via the SCL0 pin from a sampling clock.

(9) Serial clock wait controller

This circuit controls the wait timing.

(10) ACK output circuit, stop condition detector, start condition detector, and ACK detector

These circuits are used to output and detect various control signals.

(11) Data hold time correction circuit

This circuit generates the hold time for data corresponding to the falling edge of the serial clock.

18.3 Registers to Control Serial Interface IIC0

Serial interface IIC0 is controlled by the following five registers.

- IIC control register 0 (IICC0)
- IIC status register 0 (IICS0)
- IIC transfer clock select register 0 (IICCL0)
- Port mode register 3 (PM3)
- Port 3 (P3)

(1) IIC control register 0 (IICC0)

This register is used to enable/stop I^2C operations, set wait timing, and set other I^2C operations.
IICC0 is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears IICC0 to 00H.

Figure 18-5. Format of IIC Control Register 0 (IICC0) (1/4)

Address: FFA8H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICC0	IICE0	LREL0	WREL0	SPIE0	WTIM0	ACKE0	STT0	SPT0

IICE0	I^2C operation enable
0	Stop operation. Reset IIC status register 0 (IICS0). Stop internal operation.
1	Enable operation.
Condition for clearing (IICE0 = 0)	Condition for setting (IICE0 = 1)
• Cleared by instruction • When $\overline{RESET}$ is input	• Set by instruction

LREL0	Exit from communications
0	Normal operation
1	This exits from the current communications operation and sets standby mode. This setting is automatically cleared after being executed. Its uses include cases in which a locally irrelevant extension code has been received. The SCL0 and SDA0 lines go into the high impedance state. The following flags of IIC status register 0 (IICS0) and IIC control register 0 (IICC0) are cleared. • STD0 • ACKD0 • TRC0 • COI0 • EXC0 • MSTS0 • STT0 • SPT0
The standby mode following exit from communications remains in effect until the following communications entry conditions are met. • After a stop condition is detected, restart is in master mode. • An address match or extension code reception occurs after the start condition.	
Condition for clearing (LREL0 = 0)Note	Condition for setting (LREL0 = 1)
• Automatically cleared after execution • When $\overline{RESET}$ is input	• Set by instruction

WREL0	Cancel wait
0	Do not cancel wait
1	Cancel wait. This setting is automatically cleared after wait is canceled.
When WREL0 is set (wait canceled) during the wait period at the ninth clock pulse in the transmission status (TRC0 = 1), the SDA0 line goes into the high impedance state (TRC0 = 0).	
Condition for clearing (WREL0 = 0)Note	Condition for setting (WREL0 = 1)
• Automatically cleared after execution • When $\overline{RESET}$ is input	• Set by instruction

SPIE0	Enable/disable generation of interrupt request when stop condition is detected
0	Disable
1	Enable
Condition for clearing (SPIE0 = 0)Note	Condition for setting (SPIE0 = 1)
• Cleared by instruction • When $\overline{RESET}$ is input	• Set by instruction

Note This flag's signal is invalid when IICE0 = 0.

Figure 18-5. Format of IIC Control Register 0 (IICC0) (2/4)

<table>
<tr><th>WTIM0</th><th colspan="2">Control of wait and interrupt request generation</th></tr>
<tr><td>0</td><td colspan="2">Interrupt request is generated at the eighth clock’s falling edge.
Master mode: After output of eight clocks, clock output is set to low level and wait is set.
Slave mode: After input of eight clocks, the clock is set to low level and wait is set for master device.</td></tr>
<tr><td>1</td><td colspan="2">Interrupt request is generated at the ninth clock’s falling edge.
Master mode: After output of nine clocks, clock output is set to low level and wait is set.
Slave mode: After input of nine clocks, the clock is set to low level and wait is set for master device.</td></tr>
<tr><td colspan="3">This bit’s setting is invalid during an address transfer and is valid after the transfer is completed. When in master mode, a wait is inserted at the falling edge of the ninth clock during address transfers. For a slave device that has received a local address, a wait is inserted at the falling edge of the ninth clock after an ACK signal is issued. When the slave device has received an extension code, a wait is inserted at the falling edge of the eighth clock.</td></tr>
<tr><td colspan="2">Condition for clearing (WTIM0 = 0)[Note]</td><td>Condition for setting (WTIM0 = 1)</td></tr>
<tr><td colspan="2">• Cleared by instruction
• When $\overline{\text{RESET}}$ is input</td><td>• Set by instruction</td></tr>
</table>

<table>
<tr><th>ACKE0</th><th colspan="2">Acknowledgment control</th></tr>
<tr><td>0</td><td colspan="2">Disable acknowledgment.</td></tr>
<tr><td>1</td><td colspan="2">Enable acknowledgment. During the ninth clock period, the SDA0 line is set to low level. However, the ACK is invalid during address transfers and is valid when EXC0 = 1.</td></tr>
<tr><td colspan="2">Condition for clearing (ACKE0 = 0)[Note]</td><td>Condition for setting (ACKE0 = 1)</td></tr>
<tr><td colspan="2">• Cleared by instruction
• When $\overline{\text{RESET}}$ is input</td><td>• Set by instruction</td></tr>
</table>

Note This flag’s signal is invalid when IICE0 = 0.

Figure 18-5. Format of IIC Control Register 0 (IICC0) (3/4)

<table>
<tr><th>STT0</th><th colspan="2">Start condition trigger</th></tr>
<tr><td>0</td><td colspan="2">Do not generate a start condition.</td></tr>
<tr><td>1</td><td colspan="2">When bus is released (during STOP mode):
Generate a start condition (for starting as master). The SDA0 line is changed from high level to low level and then the start condition is generated. Next, after the rated amount of time has elapsed, SCL0 is changed to low level.
When bus is not used:
This trigger functions as a start condition reservation flag. When set, it releases the bus and then automatically generates a start condition.
Wait status (during master mode):
Generate a restart condition after wait is released.</td></tr>
<tr><td colspan="3">Cautions concerning set timing
• For master reception: Cannot be set during transfer. Can be set only in the waiting period when ACKE0 has been set to 0 and slave has been notified of final reception.
• For master transmission: A start condition may not be generated normally during the ACK period. Therefore, set it during the waiting period.
• Cannot be set at the same time as SPT0</td></tr>
<tr><td colspan="2">Condition for clearing (STT0 = 0)</td><td>Condition for setting (STT0 = 1)</td></tr>
<tr><td colspan="2">• Cleared by loss in arbitration
• Cleared after start condition is generated by master device
• Cleared by LREL0 = 1 (exit from communications)
• When IICE0 = 0 (operation stop)
• When $\overline{\text{RESET}}$ is input</td><td>• Set by instruction</td></tr>
</table>

★ **Remark** Bit 1 (STT0) is 0 when read after data has been set.

Figure 18-5. Format of IIC Control Register 0 (IICC0) (4/4)

<table>
<tr><th>SPT0</th><th colspan="2">Stop condition trigger</th></tr>
<tr><td>0</td><td colspan="2">Stop condition is not generated.</td></tr>
<tr><td>1</td><td colspan="2">Stop condition is generated (termination of master device’s transfer).
After the SDA0 line goes to low level, either set the SCL0 line to high level or wait until it goes to high level. Next, after the rated amount of time has elapsed, the SDA0 line changes from low level to high level and a stop condition is generated.</td></tr>
<tr><td colspan="3">Cautions concerning set timing
• For master reception: Cannot be set during transfer.
Can be set only in the waiting period when ACKE0 has been set to 0 and slave has been notified of final reception.
• For master transmission: A stop condition cannot be generated normally during the ACK0 period. Therefore, set it during the waiting period.
• Cannot be set at the same time as STT0.
• SPT0 can be set only when in master mode.Note
• When WTIM0 has been set to 0, if SPT0 is set during the wait period that follows output of eight clocks, note that a stop condition will be generated during the high level period of the ninth clock.
When a ninth clock must be output, WTIM0 should be changed from 0 to 1 during the wait period following output of eight clocks, and SPT0 should be set during the wait period that follows output of the ninth clock.</td></tr>
<tr><td colspan="2">Condition for clearing (SPT0 = 0)</td><td>Condition for setting (SPT0 = 1)</td></tr>
<tr><td colspan="2">• Cleared by loss in arbitration
• Automatically cleared after stop condition is detected
• Cleared by LREL0 = 1 (exit from communications)
• When IICE0 = 0 (operation stop)
• When $\overline{\text{RESET}}$ is input</td><td>• Set by instruction</td></tr>
</table>

Note Set SPT0 only in master mode. However, SPT0 must be set and a stop condition generated before the first stop condition is detected following the switch to the operation enabled status. For details, see **18.5.14 Other cautions**.

Caution When bit 3 (TRC0) of IIC status register 0 (IICS0) is set to 1, WREL0 is set during the ninth clock and wait is canceled, after which TRC0 is cleared and the SDA0 line is set to high impedance.

Remark Bit 0 (SPT0) becomes 0 when it is read after data setting.

(2) IIC status register 0 (IICS0)

This register indicates the status of I^2C.

IICS0 is read by a 1-bit or 8-bit memory manipulation instruction.

RESET input clears IICS0 to 00H.

Figure 18-6. Format of IIC Status Register 0 (IICS0) (1/3)

Address: FFA9H After reset: 00H R

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICS0	MSTS0	ALD0	EXC0	COI0	TRC0	ACKD0	STD0	SPD0

MSTS0	Master device status
0	Slave device status or communication standby status
1	Master device communication status
Condition for clearing (MSTS0 = 0)	Condition for setting (MSTS0 = 1)
• When a stop condition is detected • When ALD0 = 1 (arbitration loss) • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • When RESET is input	• When a start condition is generated

ALD0	Detection of arbitration loss
0	This status means either that there was no arbitration or that the arbitration result was a “win”.
1	This status indicates the arbitration result was a “loss”. MSTS0 is cleared.
Condition for clearing (ALD0 = 0)	Condition for setting (ALD0 = 1)
• Automatically cleared after IICS0 is read[Note] • When IICE0 changes from 1 to 0 (operation stop) • When RESET is input	• When the arbitration result is a “loss”.

EXC0	Detection of extension code reception
0	Extension code was not received.
1	Extension code was received.
Condition for clearing (EXC0 = 0)	Condition for setting (EXC0 = 1)
• When a start condition is detected • When a stop condition is detected • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • When RESET is input	• When the higher 4 bits of the received address data are either “0000” or “1111” (set at the rising edge of the eighth clock).

Note This register is also cleared when a bit manipulation instruction is executed for bits other than IICS0.

Remark LREL0: Bit 6 of IIC control register 0 (IICC0)
IICE0: Bit 7 of IIC control register 0 (IICC0)

Figure 18-6. Format of IIC Status Register 0 (IICS0) (2/3)

COI0	Detection of matching addresses
0	Addresses do not match.
1	Addresses match.

Condition for clearing (COI0 = 0)	Condition for setting (COI0 = 1)
• When a start condition is detected • When a stop condition is detected • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • When $\overline{\text{RESET}}$ is input	• When the received address matches the local address (slave address register 0 (SVA0)) (set at the rising edge of the eighth clock).

TRC0	Detection of transmit/receive status
0	Receive status (other than transmit status). The SDA0 line is set to high impedance.
1	Transmit status. The value in the SO0 latch is enabled for output to the SDA0 line (valid starting at the falling edge of the first byte's ninth clock).

Condition for clearing (TRC0 = 0)	Condition for setting (TRC0 = 1)
<Both master and slave> • When a stop condition is detected • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • Cleared by WREL0 = 1[Note] (wait cancel) • When ALD0 changes from 0 to 1 (arbitration loss) • When $\overline{\text{RESET}}$ is input <Master> • When "1" is output to the first byte's LSB (transfer direction specification bit) <Slave> • When a start condition is detected • When "0" is input to the first byte's LSB (transfer direction specification bit) <When not used for communication>	<Master> • When a start condition is generated <Slave> • When "1" is input to the first byte's LSB (transfer direction specification bit)

Note If the wait status is cleared by setting bit 5 (WREL0) of IIC control register 0 (IICC0) to 1 at the ninth clock when bit 3 (TRC0) of IIC status register 0 (IICS0) is 1, TRC0 is cleared, and the SDA0 line goes into a high-impedance state.

Remark LREL0: Bit 6 of IIC control register 0 (IICC0)
IICE0: Bit 7 of IIC control register 0 (IICC0)

Figure 18-6. Format of IIC Status Register 0 (IICS0) (3/3)

ACKD0	Detection of ACK
0	ACK was not detected.
1	ACK was detected.

Condition for clearing (ACKD0 = 0)	Condition for setting (ACKD0 = 1)
• When a stop condition is detected • At the rising edge of the next byte's first clock • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • When $\overline{\text{RESET}}$ is input	• After the SDA0 line is set to low level at the rising edge of the SCL0's ninth clock

STD0	Detection of start condition
0	Start condition was not detected.
1	Start condition was detected. This indicates that the address transfer period is in effect.

Condition for clearing (STD0 = 0)	Condition for setting (STD0 = 1)
• When a stop condition is detected • At the rising edge of the next byte's first clock following address transfer • Cleared by LREL0 = 1 (exit from communications) • When IICE0 changes from 1 to 0 (operation stop) • When $\overline{\text{RESET}}$ is input	• When a start condition is detected

SPD0	Detection of stop condition
0	Stop condition was not detected.
1	Stop condition was detected. The master device's communication was terminated and the bus was released.

Condition for clearing (SPD0 = 0)	Condition for setting (SPD0 = 1)
• At the rising edge of the address transfer byte's first clock following setting of this bit and detection of a start condition • When IICE0 changes from 1 to 0 (operation stop) • When $\overline{\text{RESET}}$ is input	• When a stop condition is detected

Remark LREL0: Bit 6 of IIC control register 0 (IICC0)
IICE0: Bit 7 of IIC control register 0 (IICC0)

(3) IIC transfer clock select register 0 (IICCL0)

This register is used to set the transfer clock for the I^{2}C bus.

IICCL0 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears IICCL0 to 00H.

Figure 18-7. Format of IIC Transfer Clock Select Register 0 (IICCL0) (1/2)

Address: FFAAH After reset: 00H R/W[Note]

Symbol	7	6	<5>	<4>	3	2	1	0
IICCL0	0	0	CLD0	DAD0	SMC0	DFC0	0	CL00

CLD0	Detection of SCL0 line level (valid only when IICE0 = 1)	
0	SCL0 line was detected at low level.	
1	SCL0 line was detected at high level.	
Condition for clearing (CLD0 = 0)		Condition for setting (CLD0 = 1)
• When the SCL0 line is at low level • When IICE0 = 0 (operation stop) • When $\overline{\text{RESET}}$ is input		• When the SCL0 line is at high level

DAD0	Detection of SDA0 line level (valid only when IICE0 = 1)	
0	SDA0 line was detected at low level.	
1	SDA0 line was detected at high level.	
Condition for clearing (DAD0 = 0)		Condition for setting (DAD0 = 1)
• When the SDA0 line is at low level • When IICE0 = 0 (operation stop) • When $\overline{\text{RESET}}$ is input		• When the SDA0 line is at high level

SMC0	Operation mode switching	
0	Operation in standard mode	
1	Operation in high-speed mode	
Condition for clearing (SMC0 = 0)		Condition for setting (SMC0 = 1)
• Cleared by instruction • When $\overline{\text{RESET}}$ is input		• Set by instruction

Note Bits 4 and 5 are read-only bits.

Remark IICE0: Bit 7 of IIC control register 0 (IICC0)

Figure 18-7. Format of IIC Transfer Clock Select Register 0 (IICCL0) (2/2)

DFC0	Control of digital filter operation[Note 1]
0	Digital filter off
1	Digital filter on

CL00	Selection of transfer rate			
	Standard mode		High-speed mode	
		f_X = 8.38 MHz		f_X = 8.38 MHz
0	f_X/44	190.4 kHz[Note 2]	f_X/24	350 kHz
1	f_X/86	97.5 kHz		

Notes 1. The digital filter can be used when in high-speed mode. The response time is slower when the digital filter is used.

2. The transfer rate in standard mode must not be set when f_X is more than 100 kHz.

Caution Stop serial transfer once before rewriting CL00 to other than the same value.

Remarks 1. f_X: Main system clock oscillation frequency

2. The transfer clock does not change in the high-speed mode even if DFC0 is turned on and off.

★ **(4) Port mode register 3 (PM3)**

PM3 is a register that sets the input/output of port 3 in 1-bit units.

To use the P32/SDA0 pin as serial data I/O and the P33/SCL0 pin as clock I/O, clear PM32, PM33, and the output latches of P32 and P33 to 0.

PM3 is set by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM3 to FFH.

Figure 18-8. Format of Port Mode Register 3 (PM3)

Address: FF23FH After reset: FFH R/W

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

PM3n	I/O mode selection of P3n pin (n = 0 to 6)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

18.4 I²C Bus Mode Functions

18.4.1 Pin configuration

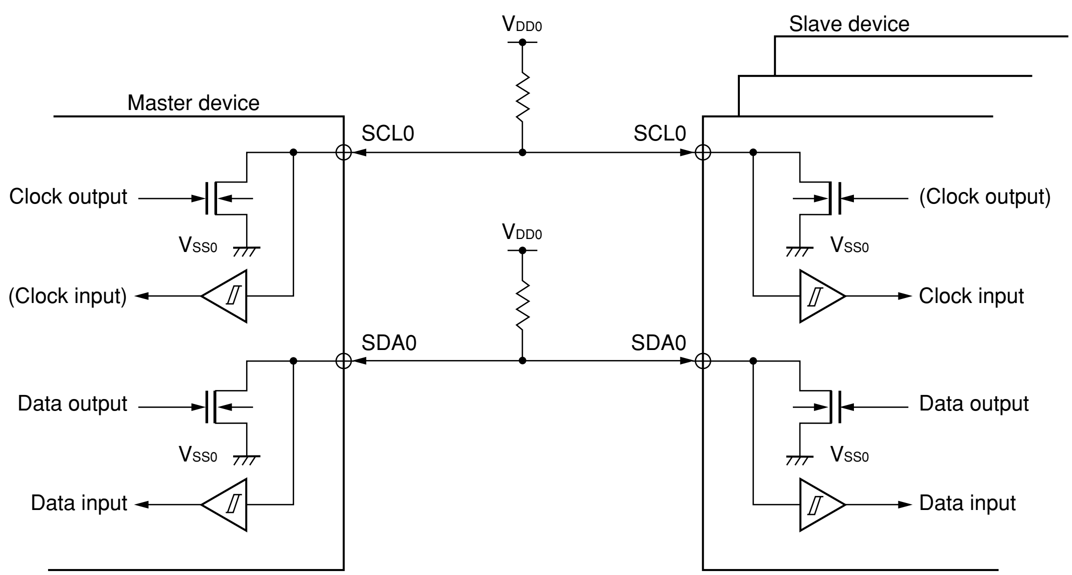
The serial clock pin (SCL0) and serial data bus pin (SDA0) are configured as follows.

(1) SCL0 ········· This pin is used for serial clock input and output.
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

(2) SDA0 ········· This pin is used for serial data input and output.
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

Since outputs from the serial clock line and the serial data bus line are N-ch open drain outputs, an external pull-up resistor is required.

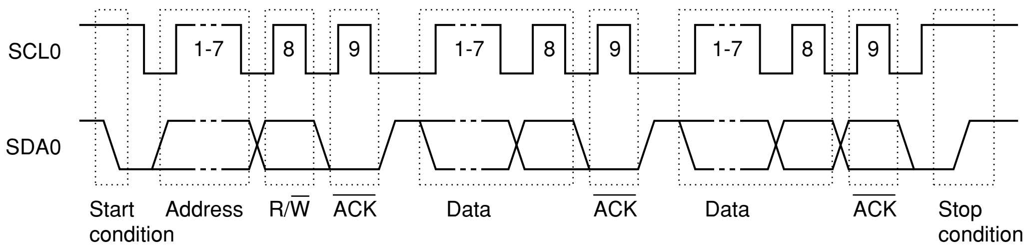
Figure 18-9. Pin Configuration Diagram



18.5 I²C Bus Definitions and Control Methods

The following section describes the I²C bus's serial data communication format and the signals used by the I²C bus. Figure 18-10 shows the transfer timing for the "start condition", "data", and "stop condition" output via the I²C bus's serial data bus.

Figure 18-10. I²C Bus Serial Data Transfer Timing



The master device outputs the start condition, slave address, and stop condition.

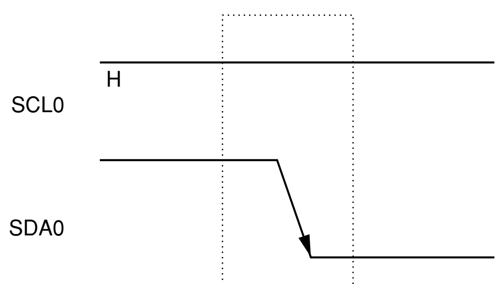
The acknowledge signal ($\overline{ACK}$) can be output by either the master or slave device (normally, it is output by the device that receives 8-bit data).

The serial clock (SCL0) is continuously output by the master device. However, in the slave device, the SCL0's low level period can be extended and a wait can be inserted.

18.5.1 Start conditions

A start condition is met when the SCL0 pin is at high level and the SDA0 pin changes from high level to low level. The start conditions for the SCL0 pin and SDA0 pin are signals that the master device outputs to the slave device when starting a serial transfer. When the device is used as a slave, start conditions can be detected.

Figure 18-11. Start Conditions



A start condition is output when bit 1 (STT0) of IIC control register 0 (IICC0) is set (to 1) after a stop condition has been detected (SPD0: Bit 0 = 1 in IIC status register 0 (IICS0)). When a start condition is detected, bit 1 (STD0) of IICS0 is set (to 1).

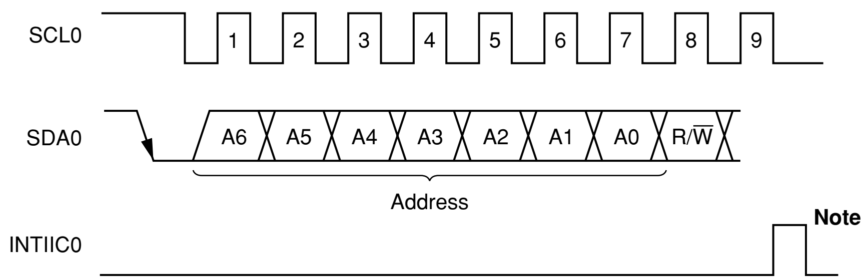
18.5.2 Addresses

The address is defined by the 7 bits of data that follow the start condition.

An address is a 7-bit data segment that is output in order to select one of the slave devices that are connected to the master device via the bus lines. Therefore, each slave device connected via the bus lines must have a unique address.

The slave devices include hardware that detects the start condition and checks whether or not the 7-bit address data matches the data values stored in slave address register 0 (SVA0). If the address data matches the SVA0 values, the slave device is selected and communicates with the master device until the master device transmits a start condition or stop condition.

Figure 18-12. Address



Note INTIIC0 is not issued if data other than a local address or extension code is received during slave device operation.

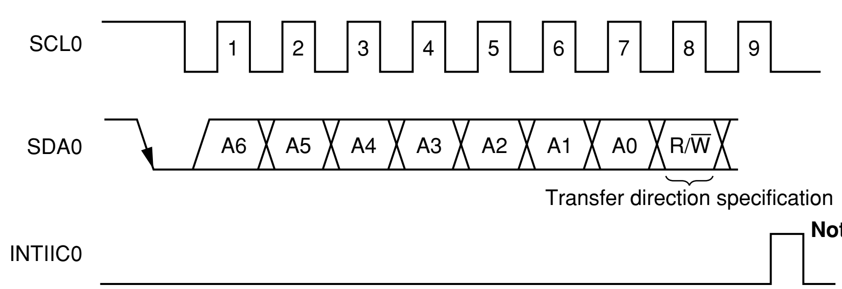
The slave address and the eighth bit, which specifies the transfer direction as described in **18.5.3 Transfer direction specification** below, are together written to IIC shift register 0 (IIC0) and are then output. Received addresses are written to IIC0.

The slave address is assigned to the higher 7 bits of IIC0.

18.5.3 Transfer direction specification

In addition to the 7-bit address data, the master device sends 1 bit that specifies the transfer direction. When this transfer direction specification bit has a value of “0”, it indicates that the master device is transmitting data to a slave device. When the transfer direction specification bit has a value of “1”, it indicates that the master device is receiving data from a slave device.

Figure 18-13. Transfer Direction Specification



Note INTIIC0 is not issued if data other than a local address or extension code is received during slave device operation.

18.5.4 Acknowledge ($\overline{\text{ACK}}$) signal

The acknowledge ($\overline{\text{ACK}}$) signal is used by the transmitting and receiving devices to confirm serial data reception.

The receiving device returns one $\overline{\text{ACK}}$ signal for each 8 bits of data it receives. The transmitting device normally receives an $\overline{\text{ACK}}$ signal after transmitting 8 bits of data. However, when the master device is the receiving device, it does not output an $\overline{\text{ACK}}$ signal after receiving the final data to be transmitted. The transmitting device detects whether or not an $\overline{\text{ACK}}$ signal is returned after it transmits 8 bits of data. When an $\overline{\text{ACK}}$ signal is returned, the reception is judged as normal and processing continues. If the slave device does not return an $\overline{\text{ACK}}$ signal, the master device outputs either a stop condition or a restart condition and then stops the current transmission. Failure to return an $\overline{\text{ACK}}$ signal may be caused by the following two factors.

(a) Reception was not performed normally.
(b) The final data was received.

When the receiving device sets the SDA0 line to low level during the ninth clock, the $\overline{\text{ACK}}$ signal becomes active (normal receive response).

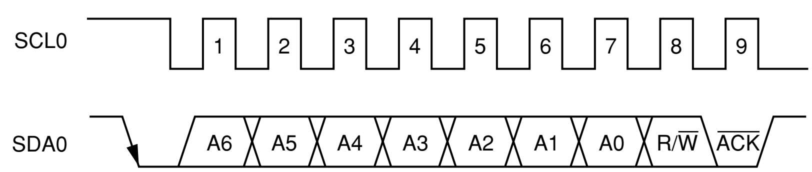
When bit 2 (ACKE0) of IIC control register 0 (IICC0) is set to 1, automatic $\overline{\text{ACK}}$ signal generation is enabled.

Transmission of the eighth bit following the 7 address data bits causes bit 3 (TRC0) of IIC status register 0 (IICS0) to be set. When this TRC0 bit's value is "0", it indicates receive mode. Therefore, ACKE0 should be set to 1.

When the slave device is receiving (when TRC0 = 0), if the slave device does not need to receive any more data after receiving several bytes, setting ACKE0 to 0 will prevent the master device from starting transmission of the subsequent data.

Similarly, when the master device is receiving (when TRC0 = 0) and the subsequent data is not needed and when either a restart condition or a stop condition should therefore be output, setting ACKE0 to 0 will prevent the $\overline{\text{ACK}}$ signal from being returned. This prevents the MSB data from being output via the SDA0 line (i.e., stops transmission) during transmission from the slave device.

Figure 18-14. $\overline{\text{ACK}}$ Signal



When the local address is received, an $\overline{\text{ACK}}$ signal is automatically output in sync with the falling edge of the SCL0's eighth clock regardless of the ACKE0 value. No $\overline{\text{ACK}}$ signal is output if the received address is not a local address.

The $\overline{\text{ACK}}$ signal output method during data reception is based on the wait timing setting, as described below.

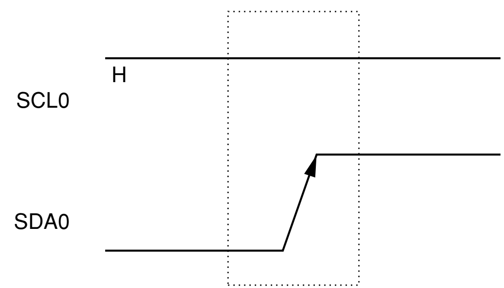
- When 8-clock wait is selected: (WTIM0 = 0) — $\overline{\text{ACK}}$ signal is output when ACKE0 is set to 1 before wait cancellation.
- When 9-clock wait is selected: (WTIM0 = 1) — $\overline{\text{ACK}}$ signal is automatically output at the falling edge of the SCL0's eighth clock if ACKE0 has already been set to 1.

18.5.5 Stop condition

When the SCL0 pin is at high level, changing the SDA0 pin from low level to high level generates a stop condition.

A stop condition is a signal that the master device outputs to the slave device when serial transfer has been completed. When the device is used as a slave, stop conditions can be detected.

Figure 18-15. Stop Condition



A stop condition is generated when bit 0 (SPT0) of IIC control register 0 (IICC0) is set (to 1). When the stop condition is detected, bit 0 (SPD0) of IIC status register 0 (IICS0) is set (to 1) and INTIIC0 is generated when bit 4 (SPIE0) of IICC0 is set (to 1).

18.5.6 Wait signal ($\overline{WAIT}$)

The wait signal ($\overline{WAIT}$) is used to notify the communication partner that a device (master or slave) is preparing to transmit or receive data (i.e., is in a wait state).

Setting the SCL0 pin to low level notifies the communication partner of the wait status. When wait status has been canceled for both the master and slave devices, the next data transfer can begin.

Figure 18-16. Wait Signal (1/2)

(1) When master device has a nine-clock wait and slave device has an eight-clock wait (master transmits, slave receives, and ACKE0 = 1)

★

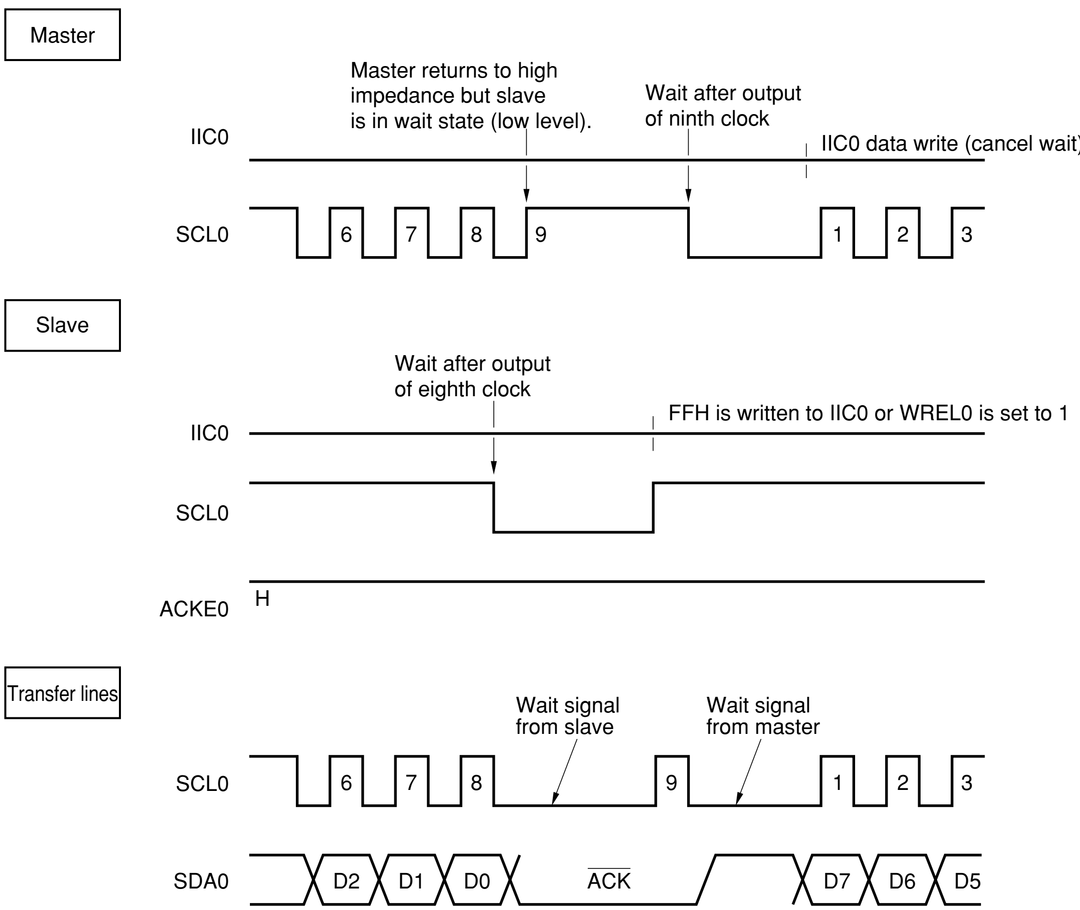
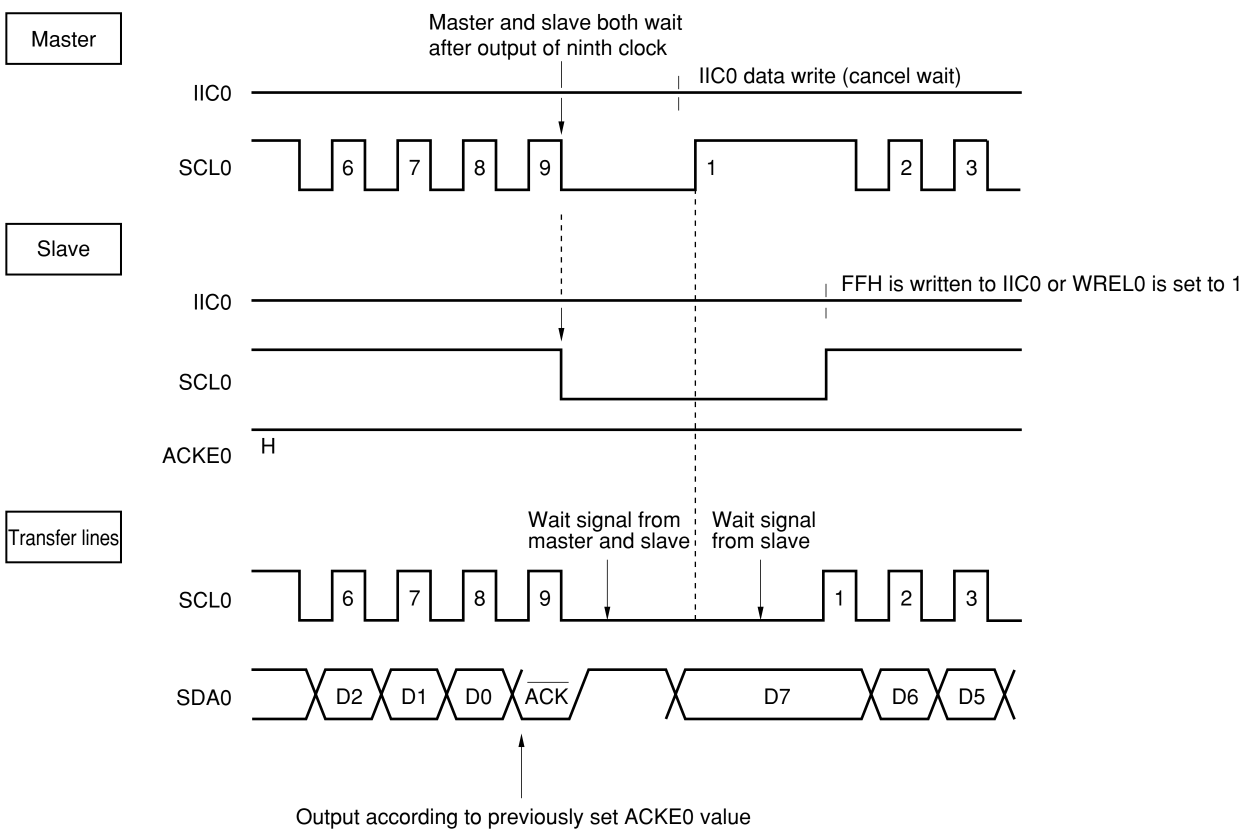


Figure 18-16. Wait Signal (2/2)

(2) When master and slave devices both have a nine-clock wait (master transmits, slave receives, and ACKE0 = 1)



★

Remark ACKE0: Bit 2 of IIC control register 0 (IICC0)
WREL0: Bit 5 of IIC control register 0 (IICC0)

A wait may be automatically generated depending on the setting of bit 3 (WTIM0) of IIC control register 0 (IICC0).

Normally, the receiving side cancels the wait status when bit 5 (WREL0) is set to 1 or when FFH is written to IIC shift register 0 (IIC0), and the transmitting side cancels the wait status when data is written to IIC0.

The master device can also cancel the wait status via either of the following methods.

- By setting bit 1 (STT0) of IICC0 to 1
- By setting bit 0 (SPT0) of IICC0 to 1

18.5.7 Interrupt request (INTIIC0) generation timing and wait control

The setting of bit 3 (WTIM0) of IIC control register 0 (IICC0) determines the timing by which INTIIC0 is generated and the corresponding wait control, as shown in Table 18-2.

Table 18-2. INTIIC0 Timing and Wait Control

WTIM0	During Slave Device Operation			During Master Device Operation		
	Address	Data Reception	Data Transmission	Address	Data Reception	Data Transmission
0	9 Notes 1, 2	8 Note 2	8 Note 2	9	8	8
1	9 Notes 1, 2	9 Note 2	9 Note 2	9	9	9

Notes 1. The slave device's INTIIC0 signal and wait period occurs at the falling edge of the ninth clock only when there is a match with the address set to slave address register 0 (SVA0).
At this point, $\overline{ACK}$ is output regardless of the value set to IICC0's bit 2 (ACKE0). For a slave device that has received an extension code, INTIIC0 occurs at the falling edge of the eighth clock.
★ However, if the address does not match after restart, INTIIC0 is generated at the falling of the 9th clock, but wait does not occur.

★ **2.** If the received address does not match the contents of slave address register 0 (SVA0) and extension code is not received, neither INTIIC0 nor a wait occurs.

Remark The numbers in the table indicate the number of the serial clock's clock signals. Interrupt requests and wait control are both synchronized with the falling edge of these clock signals.

(1) During address transmission/reception

- Slave device operation: Interrupt and wait timing are determined depending on the conditions described in Notes 1 and 2 above, regardless of the WTIM0 bit.
- Master device operation: Interrupt and wait timing occur at the falling edge of the ninth clock regardless of the WTIM0 bit.

(2) During data reception

- Master/slave device operation: Interrupt and wait timing are determined according to the WTIM0 bit.

(3) During data transmission

- Master/slave device operation: Interrupt and wait timing are determined according to the WTIM0 bit.

(4) Wait cancellation method

The four wait cancellation methods are as follows.

- By setting bit 5 (WREL0) of IIC control register 0 (IICC0) to 1
- By writing to the IIC shift register 0 (IIC0)
- By setting a start condition (setting bit 1 (STT0) of IICC0 to 1)Note
- By setting a stop condition (setting bit 0 (SPT0) of IICC0 to 1)Note

Note Master only.

When an 8-clock wait has been selected (WTIM0 = 0), the output level of $\overline{ACK}$ must be determined prior to wait cancellation.

(5) Stop condition detection

INTIIC0 is generated when a stop condition is detected.

18.5.8 Address match detection method

In I^2C bus mode, the master device can select a particular slave device by transmitting the corresponding slave address.

Address match can be detected automatically by hardware. An interrupt request (INTIIC0) occurs when a local address has been set to slave address register 0 (SVA0) and when the address set to SVA0 matches the slave address sent by the master device, or when an extension code has been received.

18.5.9 Error detection

In I^2C bus mode, the status of the serial data bus (SDA0) during data transmission is captured by IIC shift register 0 (IIC0) of the transmitting device, so the IIC0 data prior to transmission can be compared with the transmitted IIC0 data to enable detection of transmission errors. A transmission error is judged as having occurred when the compared data values do not match.

18.5.10 Extension code

(1) When the higher 4 bits of the receive address are either “0000” or “1111”, the extension code flag (EXC0) is set for extension code reception and an interrupt request (INTIIC0) is issued at the falling edge of the eighth clock. The local address stored in slave address register 0 (SVA0) is not affected.

(2) If “111110××” is set to SVA0 by a 10-bit address transfer and “111110××” is transferred from the master device, the results are as follows. Note that INTIIC0 occurs at the falling edge of the eighth clock.

- Higher four bits of data match: EXC0 = 1**Note**
- Seven bits of data match: COI0 = 1**Note**

Note EXC0: Bit 5 of IIC status register 0 (IICS0)
COI0: Bit 4 of IIC status register 0 (IICS0)

(3) Since the processing after the interrupt request occurs differs according to the data that follows the extension code, such processing is performed by software.
For example, after the extension code is received, if you do not wish to operate the target device as a slave device, you can set bit 6 (LREL0) of IIC control register 0 (IICC0) to 1 to set the standby mode for the next communication operation.

Table 18-3. Extension Code Bit Definitions

Slave Address	R/W Bit	Description
0000 000	0	General call address
0000 000	1	Start byte
0000 001	×	CBUS address
0000 010	×	Address that is reserved for different bus format
1111 0××	×	10-bit slave address specification

18.5.11 Arbitration

When several master devices simultaneously output a start condition (when STT0 is set to 1 before STD0 is set to 1[Note]), communication among the master devices is performed as the number of clocks are adjusted until the data differs. This kind of operation is called arbitration.

When one of the master devices loses in arbitration, an arbitration loss flag (ALD0) in IIC status register 0 (IICS0) is set (1) via the timing by which the arbitration loss occurred, and the SCL0 and SDA0 lines are both set to high impedance, which releases the bus.

The arbitration loss is detected based on the timing of the next interrupt request (the eighth or ninth clock, when a stop condition is detected, etc.) and the ALD0 = 1 setting that has been made by software.

For details of interrupt request timing, see **18.5.16 Timing of I^{2}C interrupt request (INTIIC0) occurrence**.

Note STD0: Bit 1 of IIC status register 0 (IICS0)
STT0: Bit 1 of IIC control register 0 (IICC0)

Figure 18-17. Arbitration Timing Example

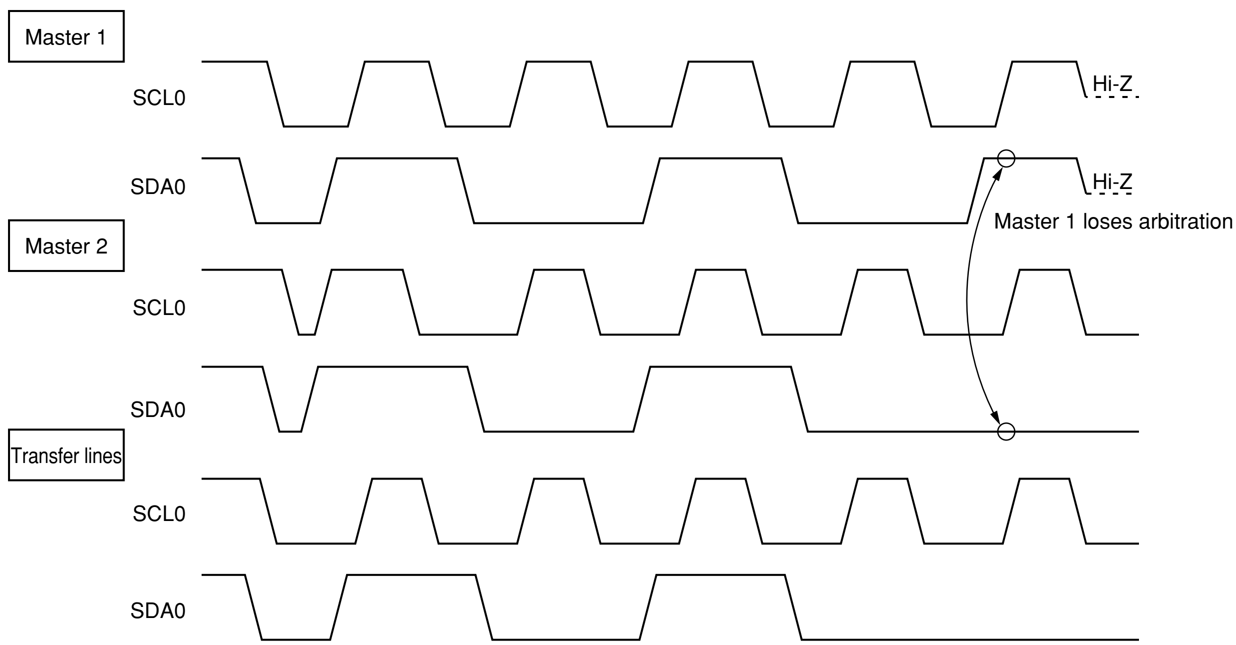


Table 18-4. Status During Arbitration and Interrupt Request Generation Timing

Status During Arbitration	Interrupt Request Generation Timing
During address transmission	At falling edge of eighth or ninth clock following byte transfer[Note 1]
Read/write data after address transmission	
During extension code transmission	
Read/write data after extension code transmission	
During data transmission	
During $\overline{\text{ACK}}$ signal transfer period after data transmission	
When restart condition is detected during data transfer	
When stop condition is detected during data transfer	When stop condition is output (when SPIE0 = 1)[Note 2]
When data is at low level while attempting to output a restart condition	At falling edge of eighth or ninth clock following byte transfer[Note 1]
When stop condition is detected while attempting to output a restart condition	When stop condition is output (when SPIE0 = 1)[Note 2]
When data is at low level while attempting to output a stop condition	At falling edge of eighth or ninth clock following byte transfer[Note 1]
When SCL0 is at low level while attempting to output a restart condition	

Notes 1. When WTIM0 (bit 3 of IIC control register 0 (IICC0)) = 1, an interrupt request occurs at the falling edge of the ninth clock. When WTIM0 = 0 and the extension code's slave address is received, an interrupt request occurs at the falling edge of the eighth clock.

2. When there is a chance that arbitration will occur, set SPIE0 = 1 for master device operation.

Remark SPIE0: Bit 4 of IIC control register 0 (IICC0)

18.5.12 Wake-up function

The I^{2}C bus slave function is a function that generates an interrupt request (INTIIC0) when a local address and extension code have been received.

This function makes processing more efficient by preventing unnecessary interrupt requests from occurring when addresses do not match.

When a start condition is detected, wake-up standby mode is set. This wake-up standby mode is in effect while addresses are transmitted due to the possibility that an arbitration loss may change the master device (which has output a start condition) to a slave device.

However, when a stop condition is detected, bit 4 (SPIE0) of IIC control register 0 (IICC0) is set regardless of the wake-up function, and this determines whether interrupt requests are enabled or disabled.

18.5.13 Communication reservation

To start master device communications when not currently using a bus, a communication reservation can be made to enable transmission of a start condition when the bus is released. There are two modes under which the bus is not used.

- When arbitration results in neither master nor slave operation
- When an extension code is received and slave operation is disabled ($\overline{\text{ACK}}$ is not returned and the bus was released when bit 6 (LREL0) of IIC control register 0 (IICC0) was set to 1).

If bit 1 (STT0) of IICC0 is set (1) while the bus is not used (after a stop condition is detected), a start condition is automatically generated and wait status is set. When the bus release is detected (when a stop condition is detected), writing to IIC shift register 0 (IIC0) causes the master address transfer to start. At this point, bit 4 (SPIE0) of IICC0 should be set (1).

When STT0 has been set (1), the operation mode (as start condition or as communication reservation) is determined according to the bus status.

- If the bus has been released .... a start condition is generated
- If the bus has not been released (standby mode) .... communication reservation

Check whether the communication reservation operates or not by using MSTS0 (bit 7 of IIC status register 0 (IICS0)) after SST0 is set and the wait time elapses.

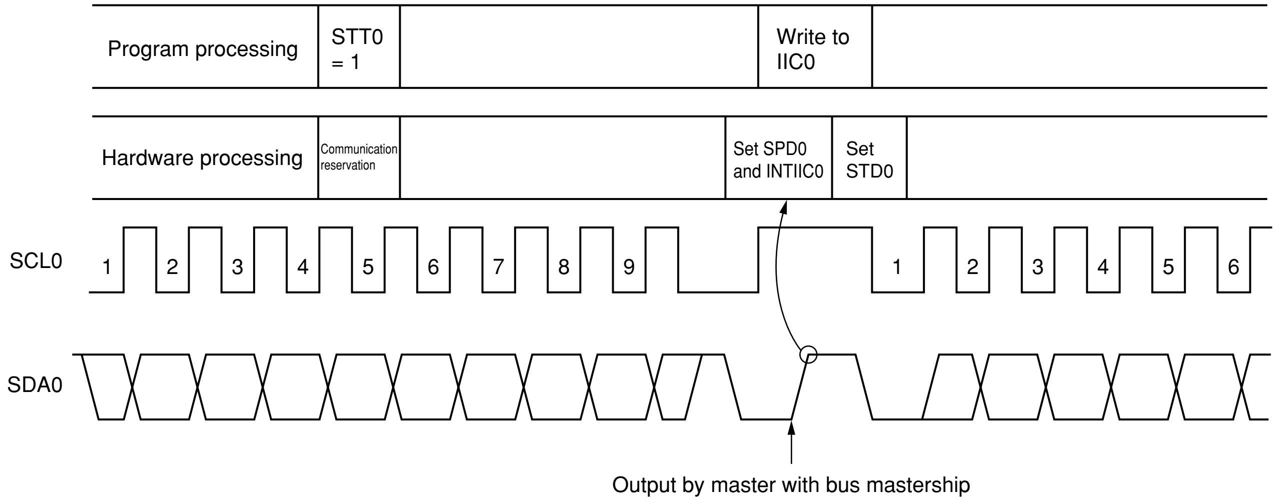
The wait periods, which should be set via software, are listed in Table 18-5. These wait periods can be set via the settings for bits 3 and 0 (SMC0 and CL00) in IIC transfer clock select register 0 (IICCL0).

Table 18-5. Wait Periods

<table>
<tr><th>SMC0</th><th>CL00</th><th>Wait Period</th></tr>
<tr><td>0</td><td>0</td><td>26 clocks</td></tr>
<tr><td>0</td><td>1</td><td>46 clocks</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">16 clocks</td></tr>
<tr><td>1</td><td>1</td></tr>
</table>

Figure 18-18 shows the communication reservation timing.

Figure 18-18. Communication Reservation Timing



Remark IIC0: IIC shift register 0
STT0: Bit 1 of IIC control register 0 (IICC0)
STD0: Bit 1 of IIC status register 0 (IICS0)
SPD0: Bit 0 of IIC status register 0 (IICS0)

Communication reservations are accepted via the following timing. After bit 1 (STD0) of IIC status register 0 (IICS0) is set to 1, a communication reservation can be made by setting bit 1 (STT0) of IIC control register 0 (IICC0) to 1 before a stop condition is detected.

Figure 18-19. Timing for Accepting Communication Reservations

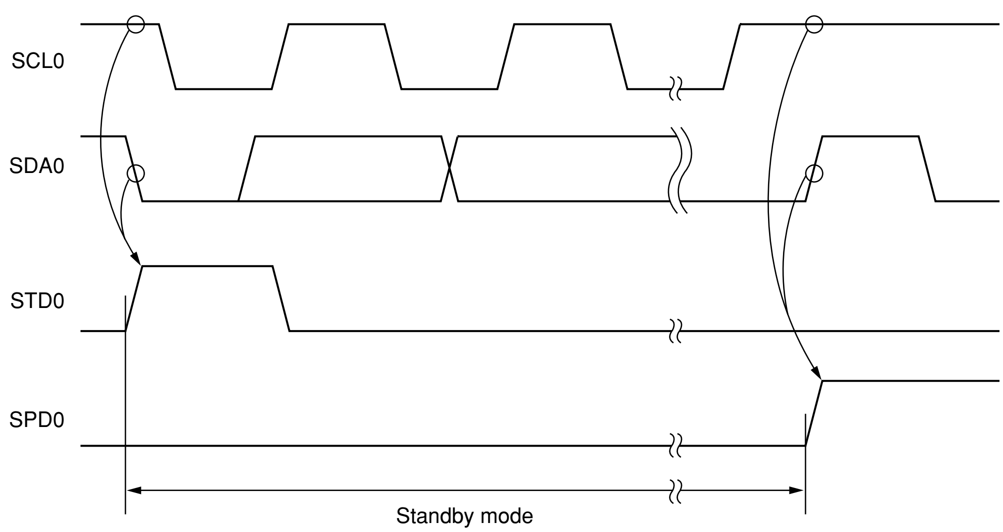
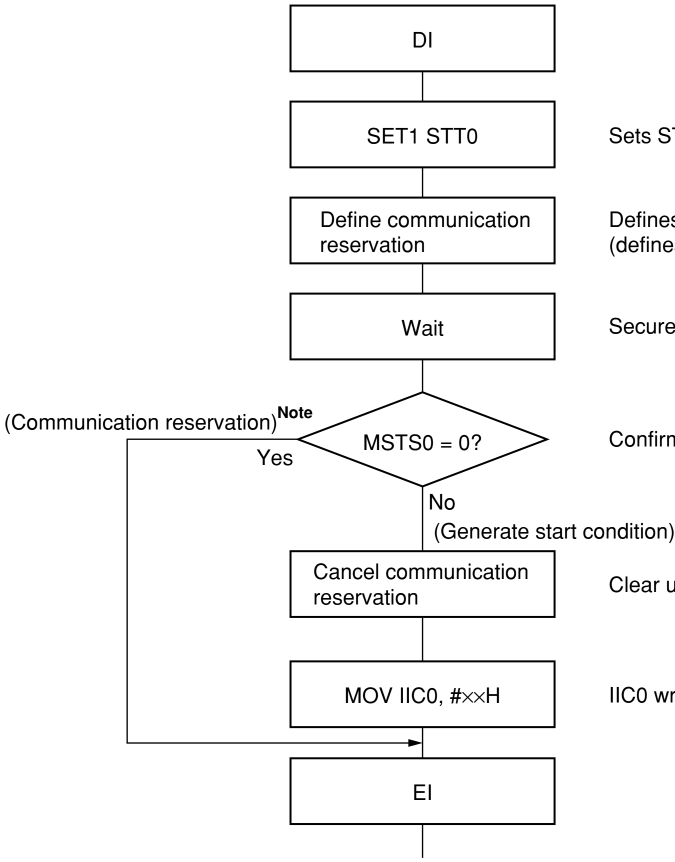


Figure 18-20 shows the communication reservation protocol.

Figure 18-20. Communication Reservation Protocol



Note The communication reservation operation executes a write to IIC shift register 0 (IIC0) when a stop condition interrupt request occurs.

Remark STT0: Bit 1 of IIC control register 0 (IICC0)
MSTS0: Bit 7 of IIC status register 0 (IICS0)
IIC0: IIC shift register 0

18.5.14 Other cautions

After a reset, when changing from a mode in which no stop condition has been detected (the bus has not been released) to a master device communication mode, first generate a stop condition to release the bus, then perform master device communication.

When using multiple masters, it is not possible to perform master device communication when the bus has not been released (when a stop condition has not been detected).

Use the following sequence for generating a stop condition.

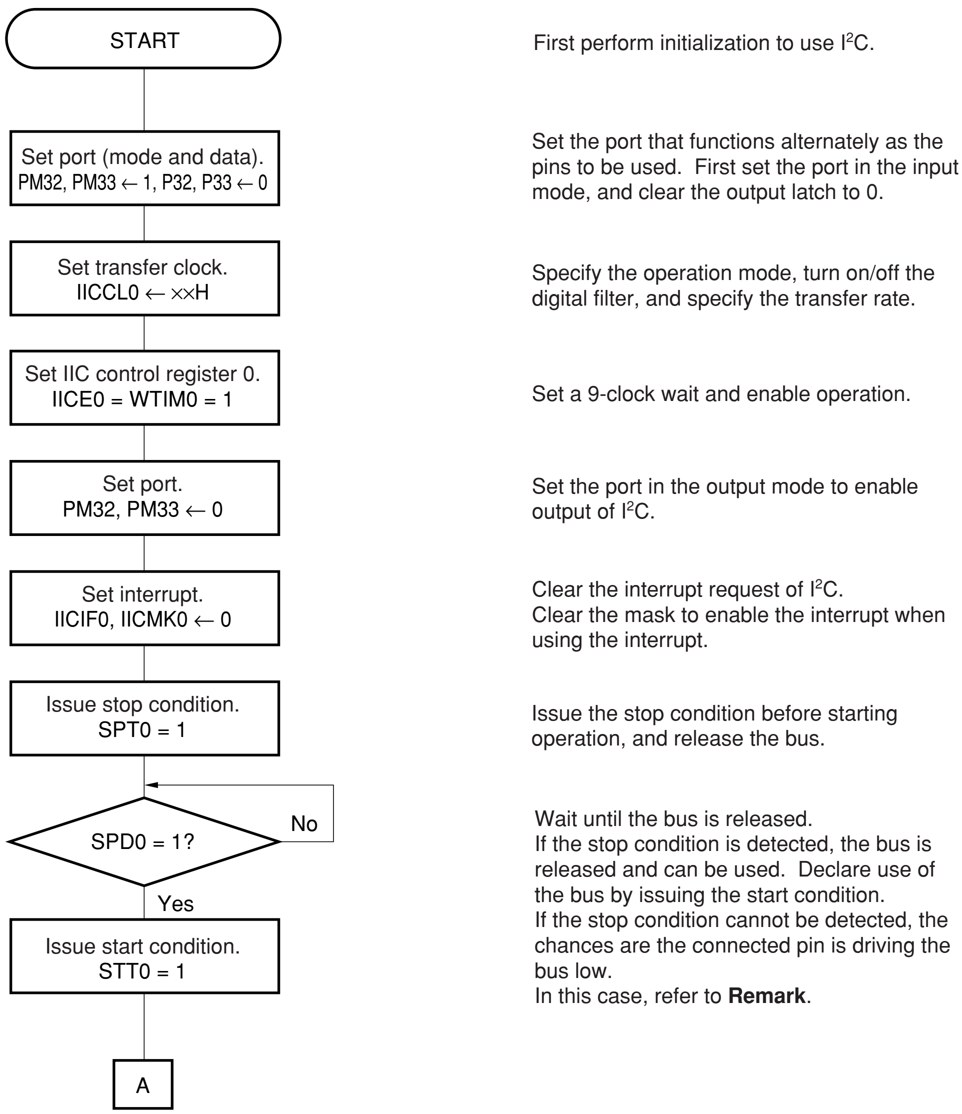
(a) Set IIC transfer clock select register 0 (IICCL0).
(b) Set (1) bit 7 (IICE0) of IIC control register 0 (IICC0).
(c) Set (1) bit 0 (SPT0) of IICC0.

18.5.15 Communication operations

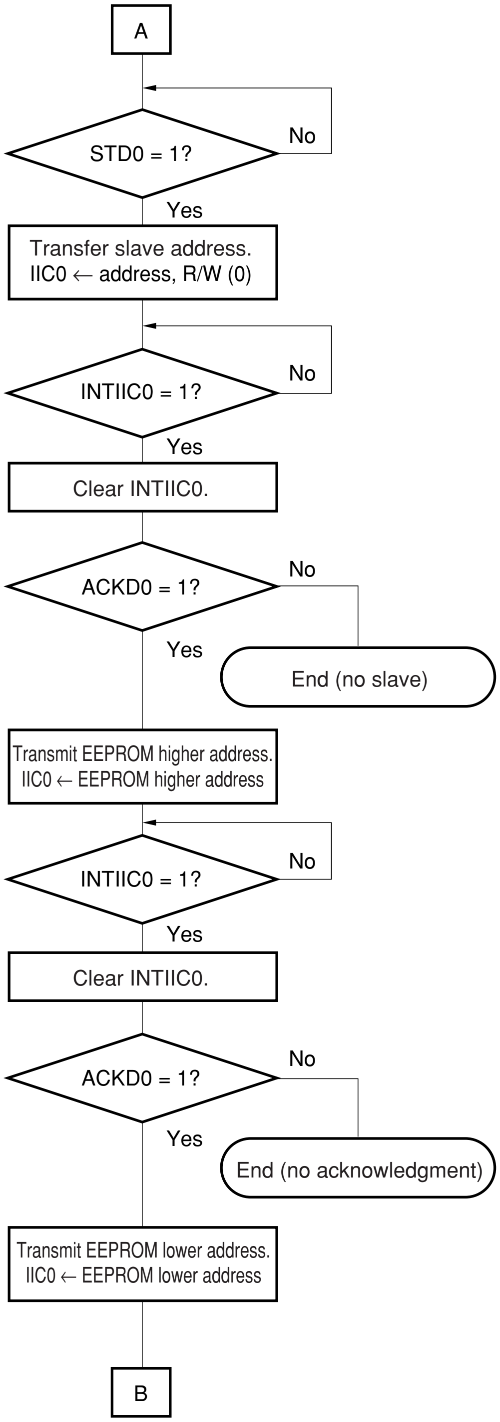
(1) Master operations

The procedure of controlling slave EEPROM™ using the μPD780024AY and 780034AY Subseries as the master of the I^2C bus is as follows.

★ **Figure 18-21. Master Operation Flowchart (1/5)**



★ **Figure 18-21. Master Operation Flowchart (2/5)**



Wait until the start condition is detected and the bus is ready.

Specify writing and transfer the address of the slave (EEPROM).

Wait until transfer is completed.

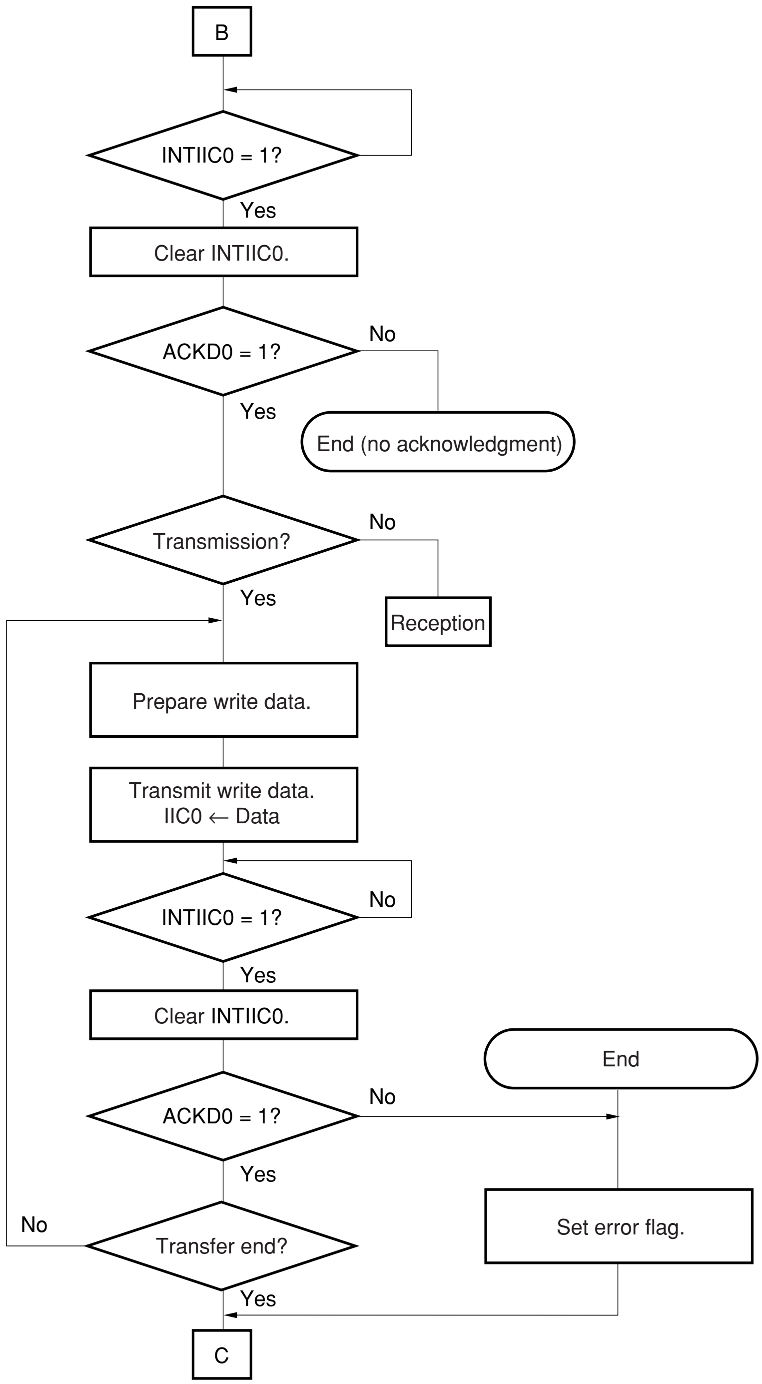
Clear INTIIC0 to poll INTIIC0 without using an interrupt.

If ACK is not sent, it means that the specified slave does not exist. End processing.

If a slave does exist, divide the address of EEPROM (2 bytes) into two, and start transmitting the address from the higher byte.
Each time transmission is completed, check ACK.

Transmit the lower address.

★ **Figure 18-21. Master Operation Flowchart (3/5)**



When writing data to EEPROM, continue writing data.

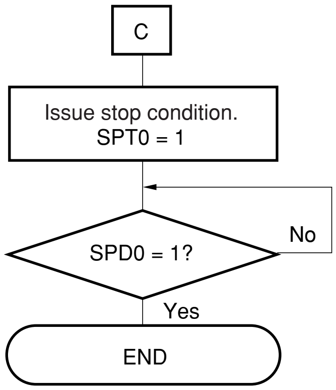
When reading data from EEPROM, start reception processing.

Prepare data to be written to EEPROM, and transmit it to EEPROM.

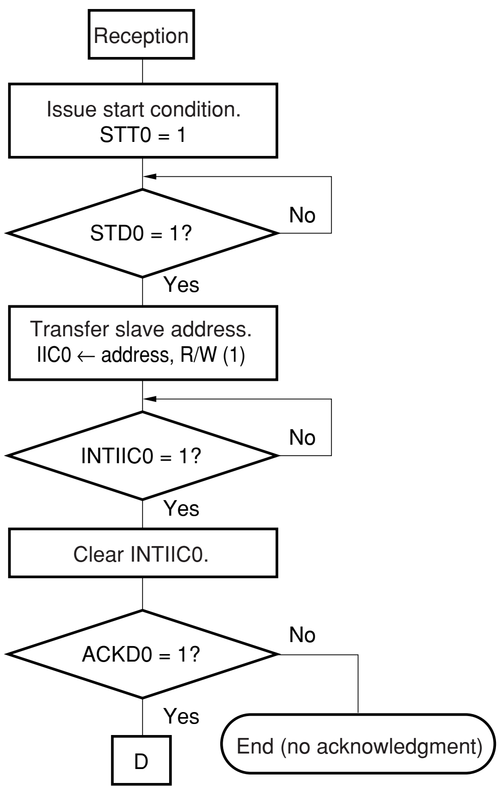
Each time data has been transmitted, the slave returns ACK. If any error occurs before transmission of the necessary data is completed, ACK may not be returned. In this case, end transfer.

In the case of an error, set the error flag as shown on the left, and release the bus.

★ **Figure 18-21. Master Operation Flowchart (4/5)**



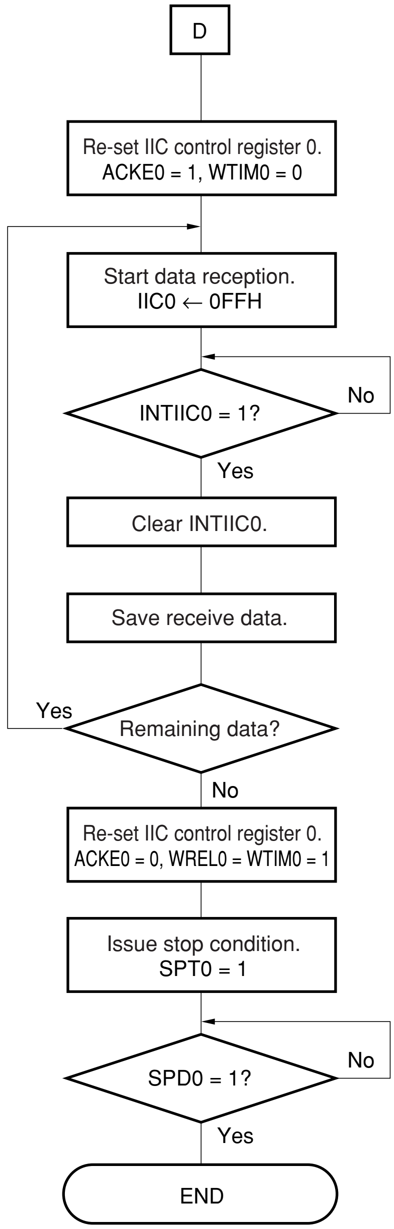
When transmission is completed, issue the stop condition to notify the slave of completion of transmission.



For reception, the data transfer direction must be changed. Issue the start condition again and redo (restart) communication.

Because the master receives data this time, set the R/W bit to 1 and transmit an address.

★ **Figure 18-21. Master Operation Flowchart (5/5)**



Set so that ACK is automatically returned after an 8-clock wait (set ACKE0 so that ACK is returned except when the last data is received. Specify an 8-clock wait so that automatic returning of ACK can be cleared when the last data is received).

Write dummy data to IIC0 and start reception (reception can also be started when WREL0 = 1).

Reception is completed when INTIIC0 occurs.

Save the received data to a buffer.

When reception of data is completed, disable automatic returning of ACK, set a 9-clock wait, cancel wait in the ACK cycle, and stop at the 9th clock. As a result, ACK is not returned to the slave. This indicates the completion of reception. Issue the stop condition and end communication.

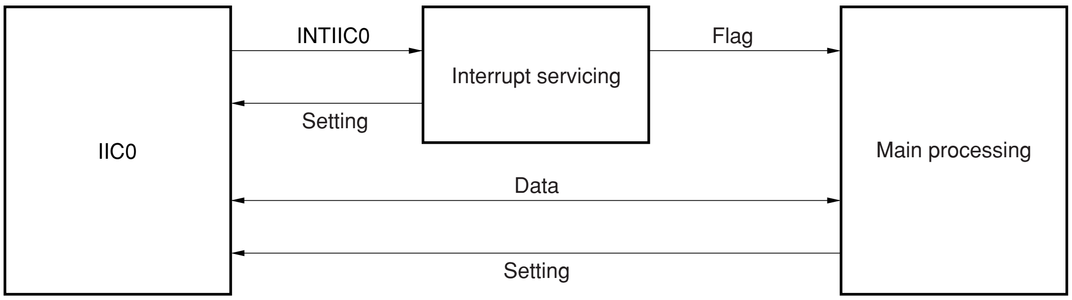
Remark While the slave is outputting a low level to the data line, the master cannot issue the stop condition. This happens if EEPROM is not reset, even though the microcontroller is reset, because of supply voltage fluctuation during communication (reading from EEPROM). In this case, the EEPROM continues sending data, and may output a low level to the data line. Because the structure of I^2C does not allow the master to forcibly make the data line high, the master cannot issue the stop condition.
To avoid this phenomenon, it is possible to use a clock line as a port, output a dummy clock from the port, continue reading data from EEPROM by inputting the dummy clock, and complete reading with some EEPROMs (because the data line goes high when reading is completed, the master can issue the stop condition. After that, the status of EEPROM can be controlled). At this time, the port corresponding to the data line must always be in the high-impedance state (high-level output).

★ **(2) Slave operation**

The processing procedure of the slave operation is as follows.

Basically, the slave operation is event-driven. Therefore, processing by the INTIIC0 interrupt (processing that must substantially change the operation status such as detection of a stop condition during communication) is necessary.

In the following explanation, it is assumed that the extension code is not supported for data communication. It is also assumed that the INTIIC0 interrupt servicing only performs status transition processing, and that actual data communication is performed by the main processing.



Therefore, data communication processing is performed by preparing the following three flags and passing them to the main processing instead of INTIIC0.

<1> Communication mode flag

This flag indicates the following two communication statuses.

- Clear mode: Status in which data communication is not performed
- Communication mode: Status in which data communication is performed (from valid address detection to stop condition detection, no detection of ACK from master, address mismatch)

<2> Ready flag

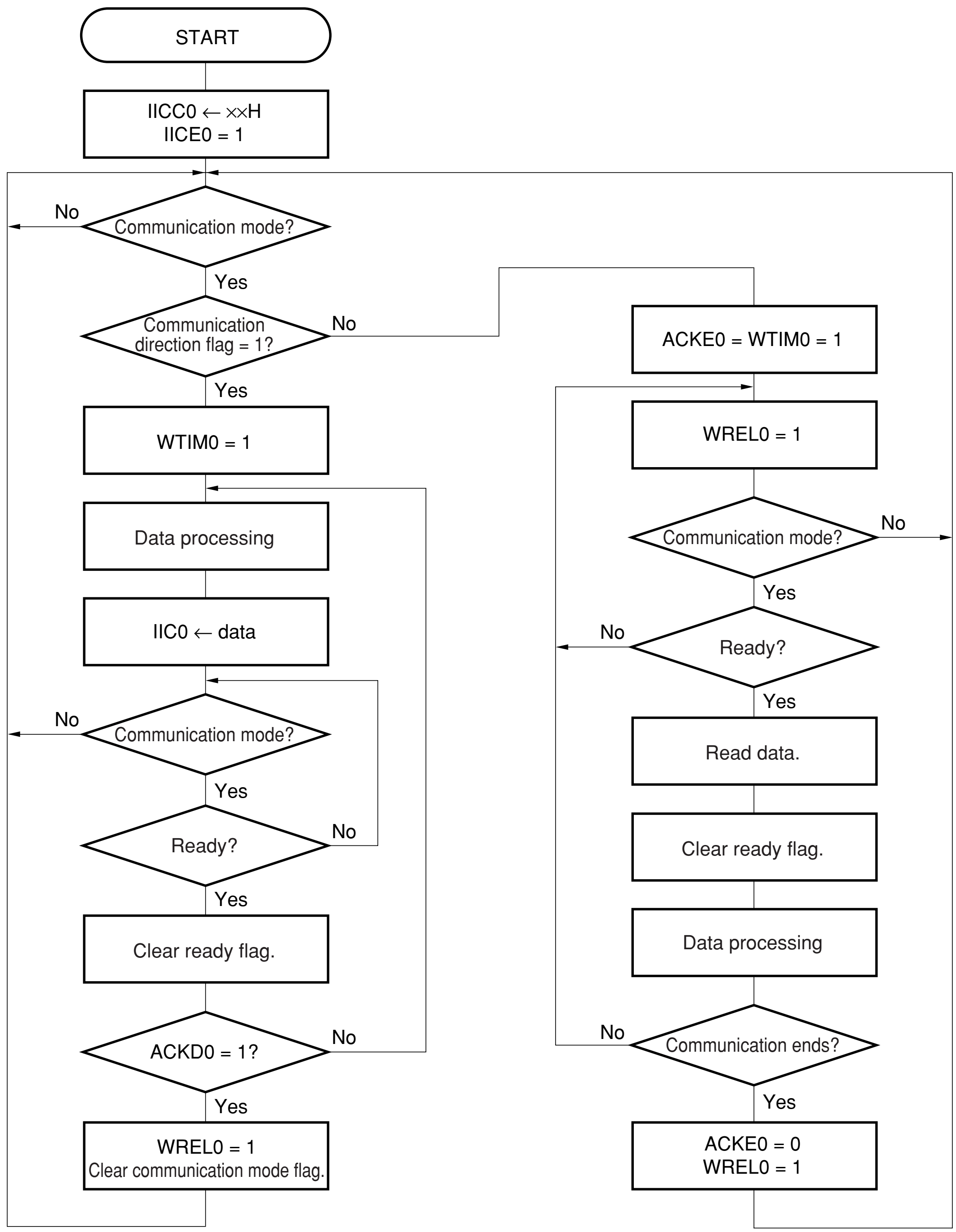
This flag indicates that data communication is enabled. Its function is the same as the INTIIC0 interrupt for ordinary data communication. This flag is set by interrupt servicing and cleared by the main processing. Clear this flag by interrupt servicing when communication is started. However, the ready flag is not set by interrupt servicing when the first data is transmitted. Therefore, the first data is transmitted without the flag being cleared (an address match is interpreted as a request for the next data).

<3> Communication direction flag

This flag indicates the direction of communication. Its value is the same as TRC0.

The main processing of the slave operation is explained next.
Start serial interface IIC0 and wait until communication is enabled. When communication is enabled, execute communication by using the communication mode flag and ready flag (processing of the stop condition and start condition is performed by an interrupt. Here, check the status by using the flags).
The transmission operation is repeated until the master no longer returns ACK. If ACK is not returned from the master, communication is completed.
For reception, the necessary amount of data is received. When communication is completed, ACK is not returned as the next data. After that, the master issues a stop condition or restart condition. Exit from the communication status occurs in this way.

Figure 18-22. Slave Operation Flowchart (1/2)



An example of the processing procedure of the slave with the INTIIC0 interrupt is explained below (processing is performed assuming that no extension code is used).

The INTIIC0 interrupt checks the status, and the following operations are performed.

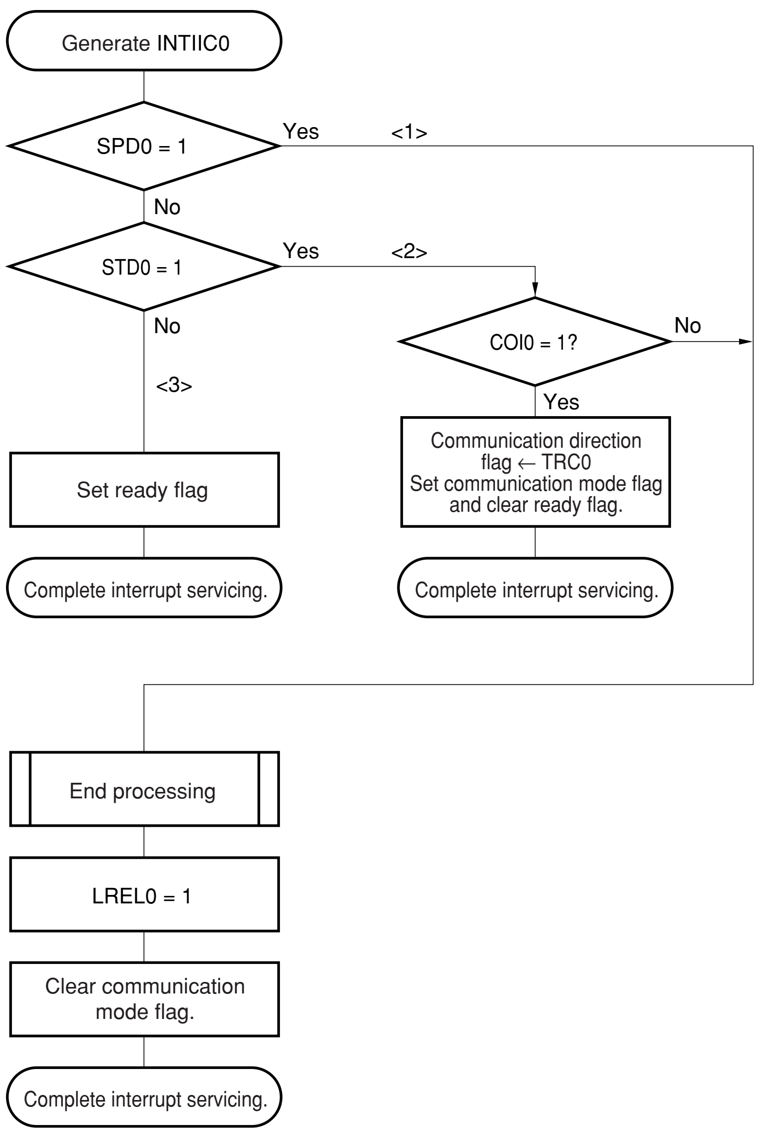
<1> Communication is stopped if the stop condition is issued.

<2> If the start condition is issued, the address is checked and communication is completed if the address does not match. If the address matches, the communication mode is set, wait is cancelled, and processing returns from the interrupt (the ready flag is cleared).

<3> For data transmit/receive, only the ready flag is set. Processing returns from the interrupt with the IIC0 bus remaining in the wait status.

Remark <1> to <3> above correspond to <1> to <3> in **Figure 18-22 Slave Operation Flowchart (2/2)**.

Figure 18-22. Slave Operation Flowchart (2/2)



18.5.16 Timing of I^{2}C interrupt request (INTIIC0) occurrence

The INTIIC0 interrupt request timing and IIC status register 0 (IICS0) settings corresponding to that timing are described below.

(1) Master device operation

(a) Start ~ Address ~ Data ~ Data ~ Stop (normal transmission/reception)

(i) When WTIM0 = 1

SPT0 = 1 ↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 ▲4 △5

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000×000B
▲3: IICS0 = 1000×000B (Sets WTIM0)
▲4: IICS0 = 1000××00B (Sets SPT0)
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

SPT0 = 1 ↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 △4

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000×100B
▲3: IICS0 = 1000××00B (Sets SPT0)
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop (restart)

(i) When WTIM0 = 0

					STT0 = 1 ↓						SPT0 = 1 ↓	
ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1	▲2	▲3				▲4	▲5	▲6	△7

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000×000B (Sets WTIM0)
▲3: IICS0 = 1000××00B (Clears WTIM0, sets STT0)
▲4: IICS0 = 1000×110B
▲5: IICS0 = 1000×000B (Sets WTIM0)
▲6: IICS0 = 1000××00B (Sets SPT0)
△7: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

					STT0 = 1 ↓						SPT0 = 1 ↓	
ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1		▲2				▲3		▲4	△5

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000××00B (Sets STT0)
▲3: IICS0 = 1000×110B
▲4: IICS0 = 1000××00B (Sets SPT0)
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(c) Start ~ Code ~ Data ~ Data ~ Stop (extension code transmission)

(i) When WTIM0 = 0

SPT0 = 1 ↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1		▲2		▲3	▲4 △5

▲1: IICS0 = 1010×110B
▲2: IICS0 = 1010×000B
▲3: IICS0 = 1010×000B (Sets WTIM0)
▲4: IICS0 = 1010××00B (Sets SPT0)
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

SPT0 = 1 ↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1		▲2			▲3 △4

▲1: IICS0 = 1010×110B
▲2: IICS0 = 1010×100B
▲3: IICS0 = 1010××00B (Sets SPT0)
△4: IICS0 = 00001001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(2) Slave device operation (slave address data reception time (matches with SVA0))

(a) Start ~ Address ~ Data ~ Data ~ Stop

(i) When WTIM0 = 0

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1		▲2		▲3	△4

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001×000B
▲3: IICS0 = 0001×000B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1		▲2		▲3	△4

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001×100B
▲3: IICS0 = 0001××00B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, matches with SVA0)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1		▲2				▲3		▲4	△5

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001×000B
▲3: IICS0 = 0001×110B
▲4: IICS0 = 0001×000B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, matches with SVA0)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1			▲2			▲3			▲4 △5

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001××00B
▲3: IICS0 = 0001×110B
▲4: IICS0 = 0001××00B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(c) Start ~ Address ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, extension code reception)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
				▲1	▲2				▲3		▲4	△5

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001×000B
▲3: IICS0 = 0010×010B
▲4: IICS0 = 0010×000B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, extension code reception)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
				▲1		▲2			▲3	▲4		▲5 △6

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001××00B
▲3: IICS0 = 0010×010B
▲4: IICS0 = 0010×110B
▲5: IICS0 = 0010××00B
△6: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(d) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, does not match with address (= not extension code))

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
				▲1		▲2					▲3		△4

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001×000B
▲3: IICS0 = 00000×10B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, does not match with address (= not extension code))

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
				▲1			▲2				▲3		△4

▲1: IICS0 = 0001×110B
▲2: IICS0 = 0001××00B
▲3: IICS0 = 00000×10B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(3) Slave device operation (when receiving extension code)

(a) Start ~ Code ~ Data ~ Data ~ Stop

(i) When WTIM0 = 0

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP	
			▲1		▲2		▲3		△4

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×000B
▲3: IICS0 = 0010×000B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP	
			▲1	▲2		▲3		▲4	△5

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×110B
▲3: IICS0 = 0010×100B
▲4: IICS0 = 0010××00B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(b) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, matches with SVA0)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1		▲2					▲3	▲4	△5

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×000B
▲3: IICS0 = 0001×110B
▲4: IICS0 = 0001×000B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, matches with SVA0)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1	▲2		▲3				▲4		▲5 △6

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×110B
▲3: IICS0 = 0010××00B
▲4: IICS0 = 0001×110B
▲5: IICS0 = 0001××00B
△6: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(c) Start ~ Code ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, extension code reception)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1		▲2				▲3		▲4	△5

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×000B
▲3: IICS0 = 0010×010B
▲4: IICS0 = 0010×000B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, extension code reception)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1	▲2		▲3			▲4	▲5		▲6 △7

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×110B
▲3: IICS0 = 0010××00B
▲4: IICS0 = 0010×010B
▲5: IICS0 = 0010×110B
▲6: IICS0 = 0010××00B
△7: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(d) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, does not match with address (= not extension code))

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1		▲2					▲3		△4

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×000B
▲3: IICS0 = 00000×10B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1 (after restart, does not match with address (= not extension code))

ST	AD6 to AD0	RW	AK	D7 to D0	AK	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1	▲2		▲3				▲4		△5

▲1: IICS0 = 0010×010B
▲2: IICS0 = 0010×110B
▲3: IICS0 = 0010××00B
▲4: IICS0 = 00000×10B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(4) Operation without communication

(a) Start ~ Code ~ Data ~ Data ~ Stop

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
								△1

△1: IICS0 = 00000001B

Remark △: Generated only when SPIE0 = 1

(5) Arbitration loss operation (operation as slave after arbitration loss)

(a) When arbitration loss occurs during transmission of slave address data

(i) When WTIM0 = 0

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 △4

▲1: IICS0 = 0101×110B (**Example** When ALD0 is read during interrupt servicing)
▲2: IICS0 = 0001×000B
▲3: IICS0 = 0001×000B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 △4

▲1: IICS0 = 0101×110B (**Example** When ALD0 is read during interrupt servicing)
▲2: IICS0 = 0001×100B
▲3: IICS0 = 0001××00B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(b) When arbitration loss occurs during transmission of extension code

(i) When WTIM0 = 0

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 △4

▲1: IICS0 = 0110×010B (**Example** When ALD0 is read during interrupt servicing)
▲2: IICS0 = 0010×000B
▲3: IICS0 = 0010×000B
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 ▲4 △5

▲1: IICS0 = 0110×010B (**Example** When ALD0 is read during interrupt servicing)
▲2: IICS0 = 0010×110B
▲3: IICS0 = 0010×100B
▲4: IICS0 = 0010××00B
△5: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(6) Operation when arbitration loss occurs (no communication after arbitration loss)

(a) When arbitration loss occurs during transmission of slave address data (when WTIM0 = 1)

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1					△2

▲1: IICS0 = 01000110B (**Example** When ALD0 is read during interrupt servicing)
△2: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1

(b) When arbitration loss occurs during transmission of extension code

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
		▲1						△2

▲1: IICS0 = 0110×010B (**Example** When ALD0 is read during interrupt servicing)
LREL0 is set to “1” by software
△2: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don’t care

(c) When arbitration loss occurs during transmission of data

(i) When WTIM0 = 0

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1	▲2				△3

▲1: IICS0 = 10001110B

▲2: IICS0 = 01000000B (**Example** When ALD0 is read during interrupt servicing)

△3: IICS0 = 00000001B

Remark ▲: Always generated

△: Generated only when SPIE0 = 1

(ii) When WTIM0 = 1

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1		▲2			△3

▲1: IICS0 = 10001110B

▲2: IICS0 = 01000100B (**Example** When ALD0 is read during interrupt servicing)

△3: IICS0 = 00000001B

Remark ▲: Always generated

△: Generated only when SPIE0 = 1

(d) When loss occurs due to restart condition during data transfer

(i) Not extension code (Example: unmatches with SVA0, WTIM0 = 1)

ST	AD6 to AD0	RW	AK	D7 to Dn	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
				▲1					▲2		△3

▲1: IICS0 = 1000×110B
▲2: IICS0 = 01000110B (**Example** When ALD0 is read during interrupt servicing)
△3: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care
n = 6 to 0

(ii) Extension code

ST	AD6 to AD0	RW	AK	D7 to Dn	ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
				▲1				▲2			△3

▲1: IICS0 = 1000×110B
▲2: IICS0 = 0110×010B (**Example** When ALD0 is read during interrupt servicing)
Sets LREL0 = 1 by software
△3: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care
n = 6 to 0

(e) When loss occurs due to stop condition during data transfer

ST	AD6 to AD0	RW	AK	D7 to Dn	SP
			▲1		△2

▲1: IICS0 = 1000×110B
△2: IICS0 = 01000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care
n = 6 to 0

(f) When arbitration loss occurs due to low-level data when attempting to generate a restart condition

(i) When WTIM0 = 1

STT0 = 1
↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	D7 to D0	AK	SP
			▲1		▲2		▲3			△4

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000×100B (Sets STT0)
▲3: IICS0 = 01000100B (**Example** When ALD0 is read during interrupt servicing)
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(g) When arbitration loss occurs due to a stop condition when attempting to generate a restart condition

(i) When WTIM0 = 1

STT0 = 1
↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	SP
			▲1		▲2	△3

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000××00B (Sets STT0)
△3: IICS0 = 01000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(h) When arbitration loss occurs due to low-level data when attempting to generate a stop condition

(i) When WTIM0 = 1

SPT0 = 1
↓

ST	AD6 to AD0	RW	AK	D7 to D0	AK	D7 to D0	AK	D7 to D0	AK	SP

▲1 ▲2 ▲3 △4

▲1: IICS0 = 1000×110B
▲2: IICS0 = 1000××00B (Sets SPT0)
▲3: IICS0 = 01000000B (**Example** When ALD0 is read during interrupt servicing)
△4: IICS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

18.6 Timing Charts

When using the I^2C bus mode, the master device outputs an address via the serial bus to select one of several slave devices as its communication partner.

After outputting the slave address, the master device transmits the TRC0 bit (bit 3 of IIC status register 0 (IICS0)), which specifies the data transfer direction, and then starts serial communication with the slave device.

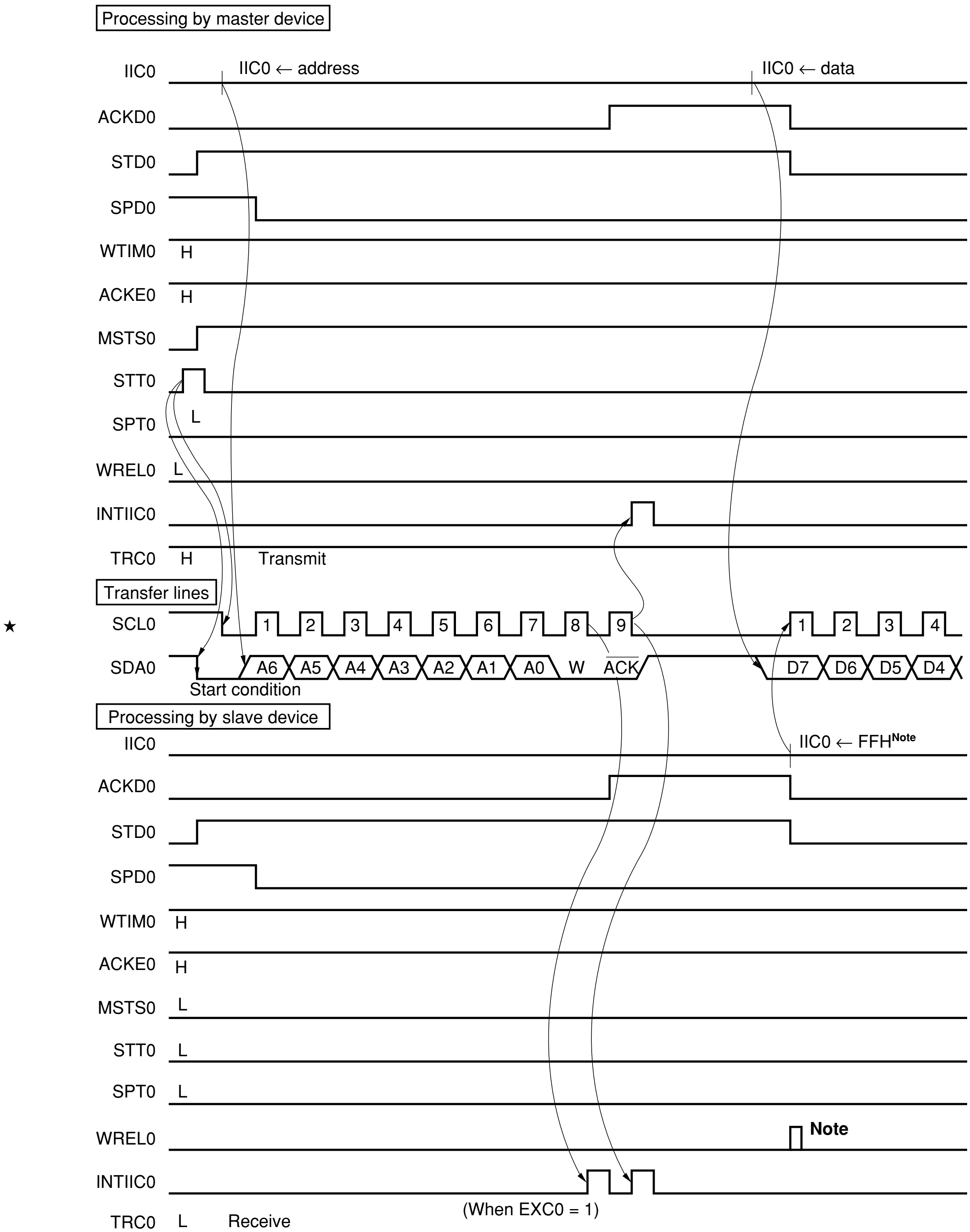
Figures 18-23 and 18-24 show timing charts of the data communication.

IIC shift register 0 (IIC0)'s shift operation is synchronized with the falling edge of the serial clock (SCL0). The transmit data is transferred to the SO0 latch and is output (MSB first) via the SDA0 pin.

Data input via the SDA0 pin is captured into IIC0 at the rising edge of SCL0.

Figure 18-23. Example of Master to Slave Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (1/3)

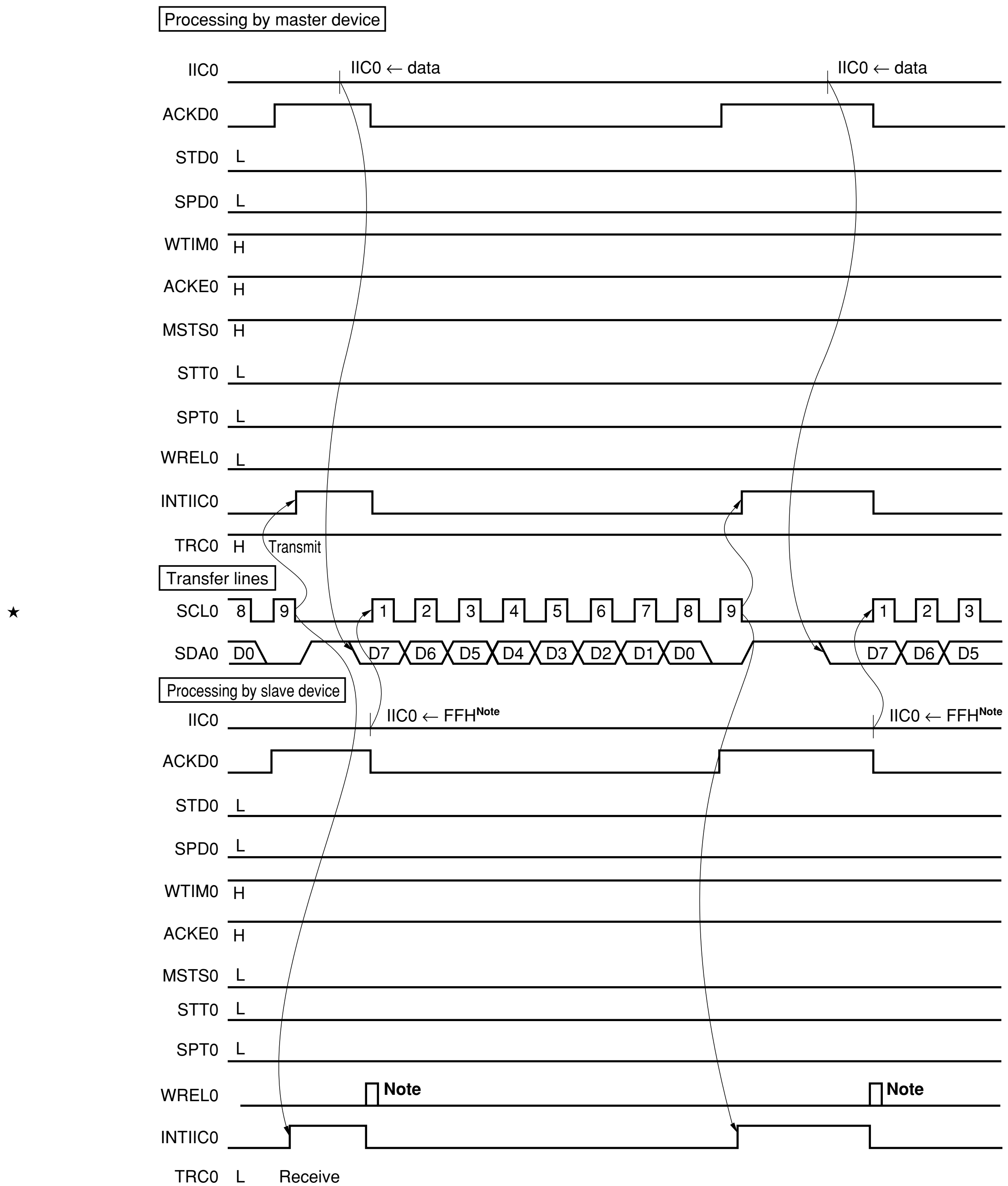
(1) Start condition ~ address



Note To cancel slave wait, write “FFH” to IIC0 or set WREL0.

Figure 18-23. Example of Master to Slave Communication
(When 9-Clock Wait Is Selected for Both Master and Slave) (2/3)

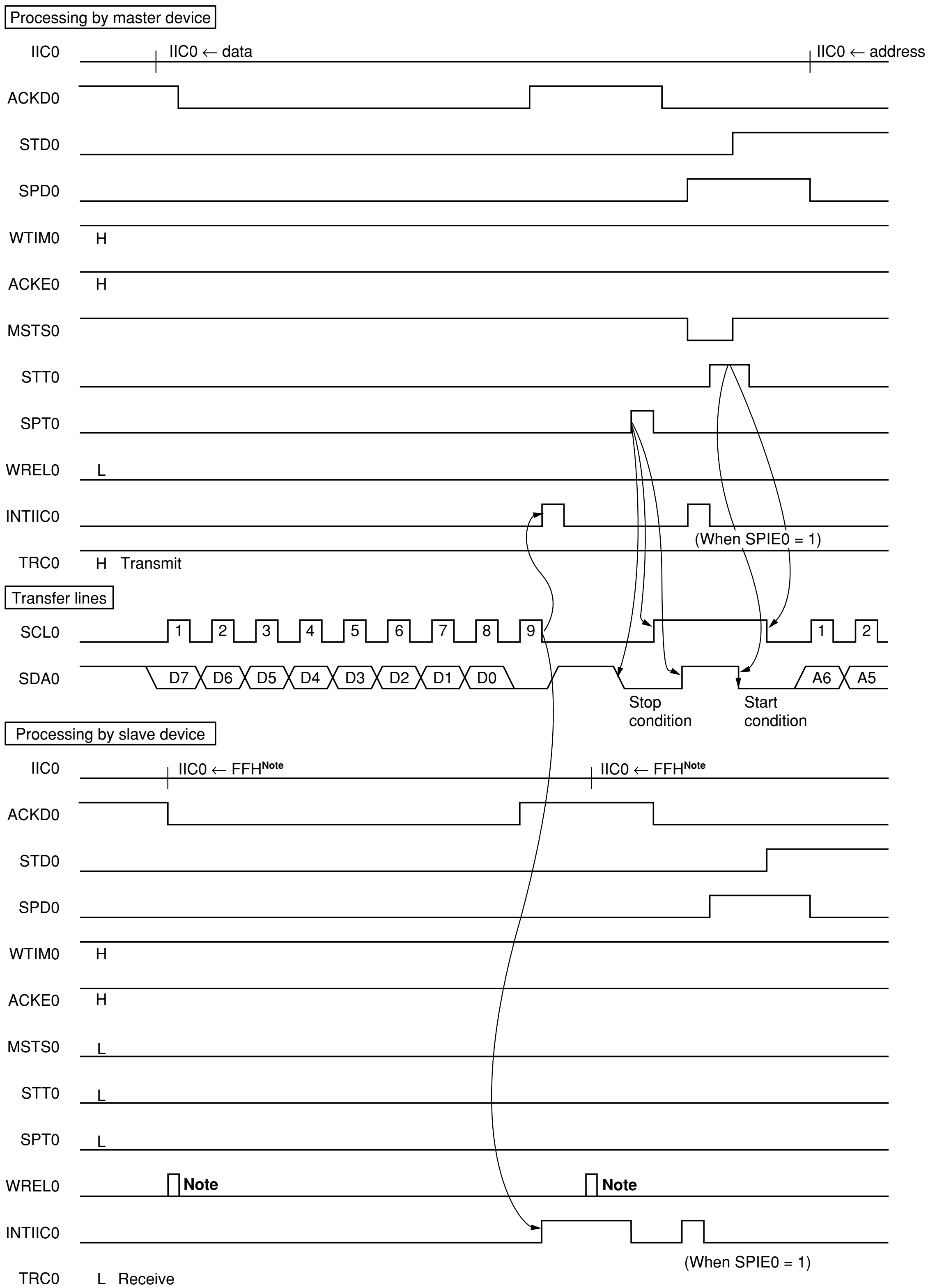
(2) Data



Note To cancel slave wait, write "FFH" to IIC0 or set WREL0.

Figure 18-23. Example of Master to Slave Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (3/3)

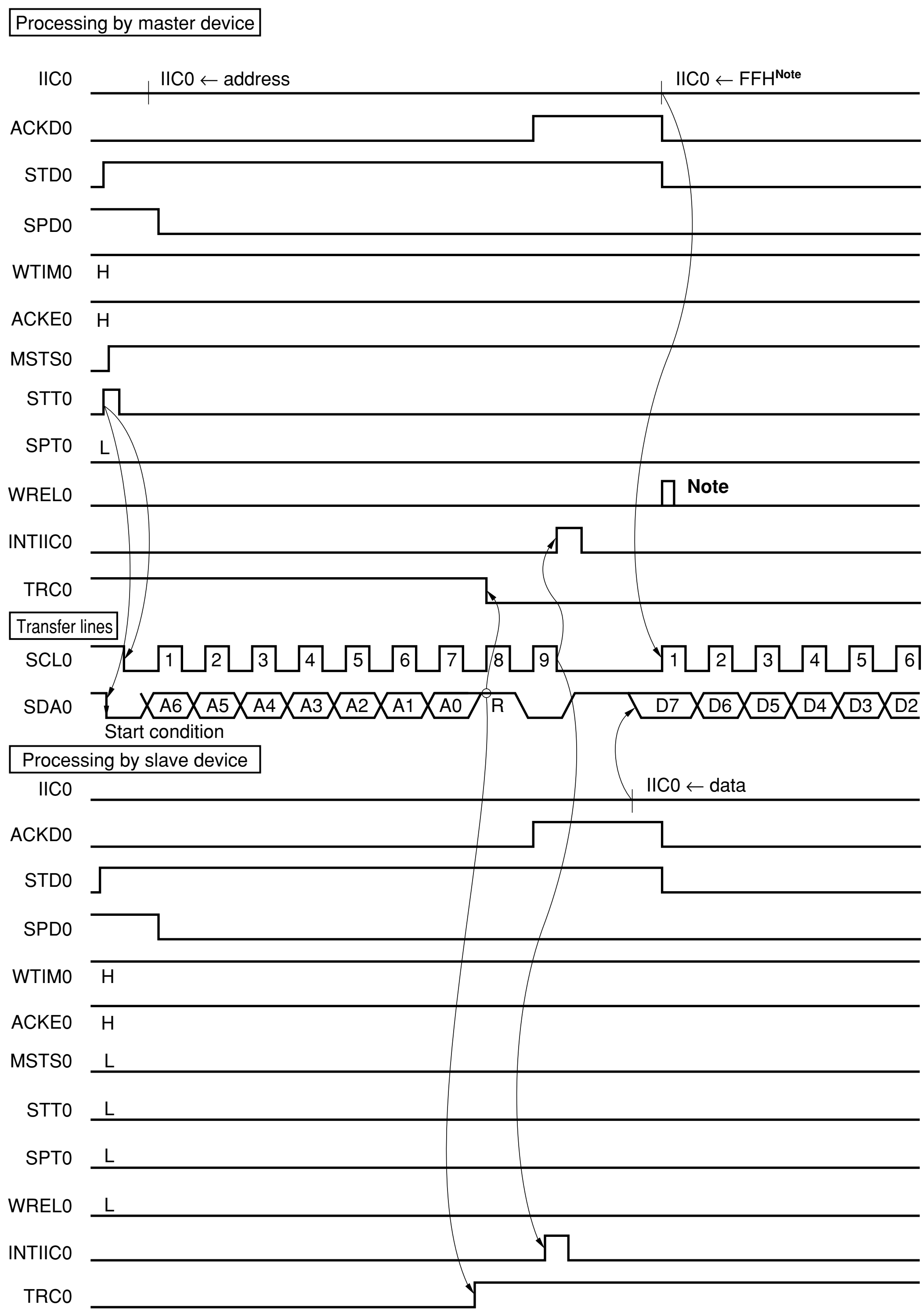
(3) Stop condition



Note To cancel slave wait, write "FFH" to IIC0 or set WREL0.

Figure 18-24. Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (1/3)

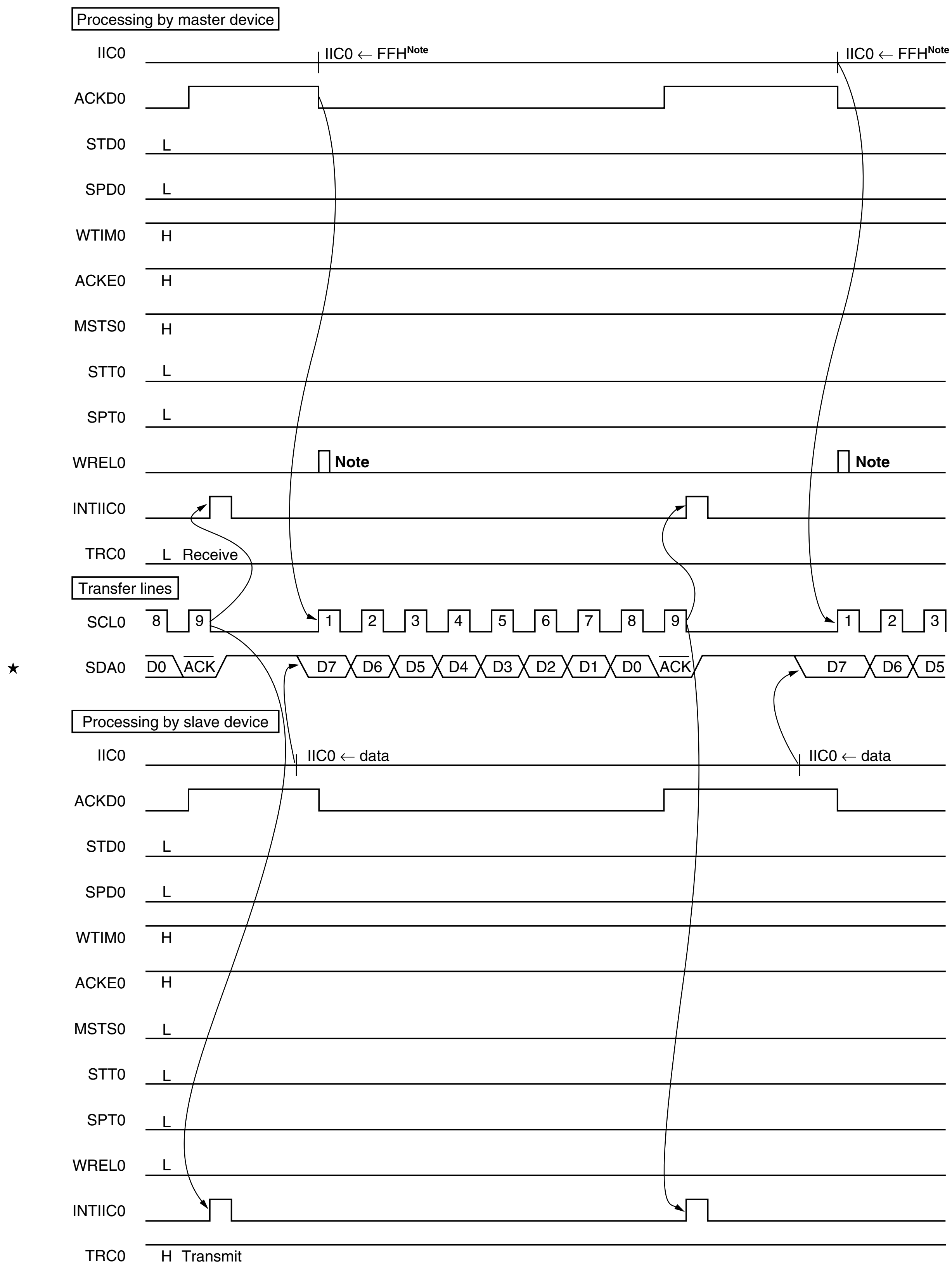
(1) Start condition ~ address



★ **Note** To cancel master wait, write "FFH" to IIC0 or set WREL0.

Figure 18-24. Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (2/3)

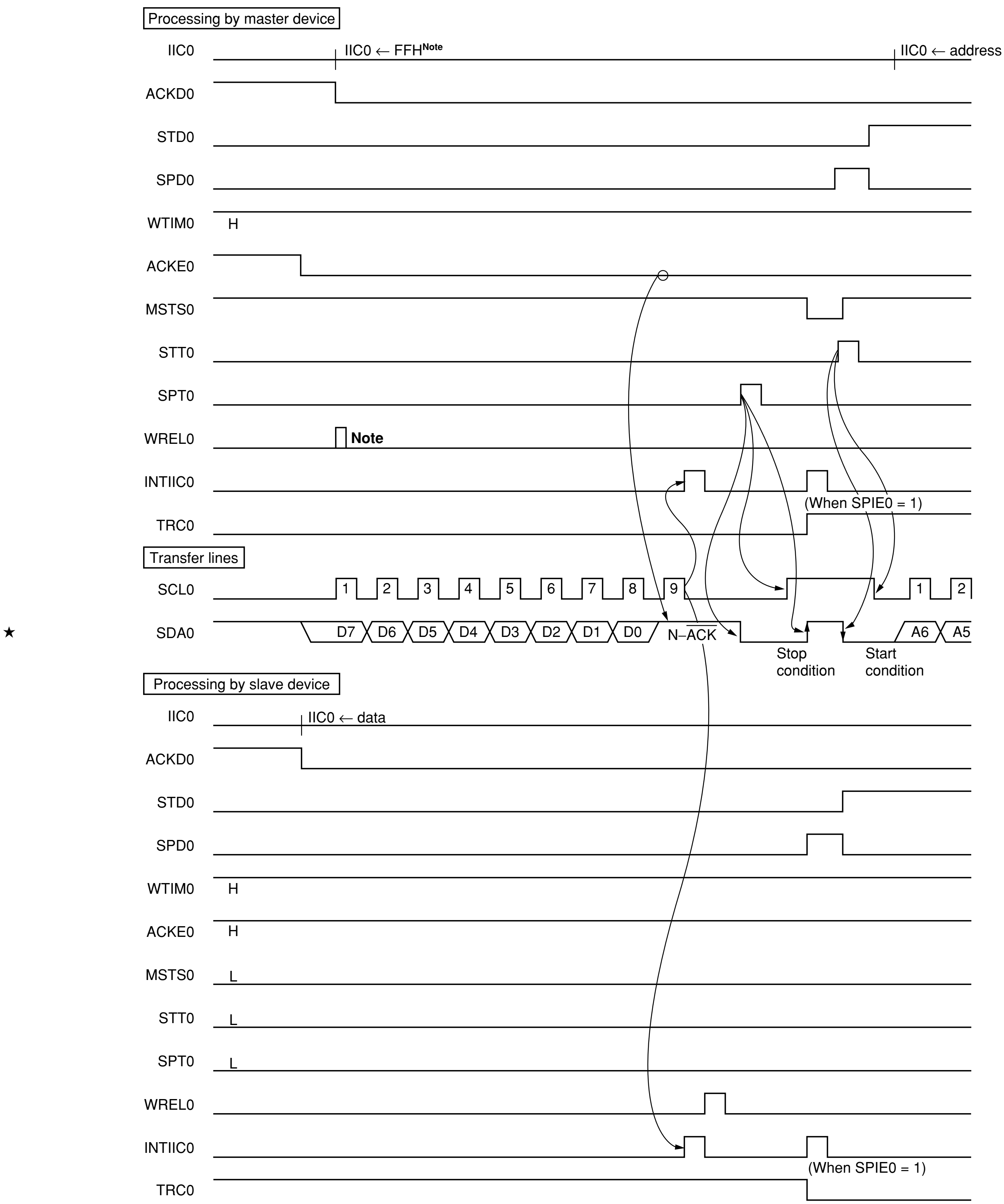
(2) Data



★ **Note** To cancel master wait, write “FFH” to IIC0 or set WREL0.

Figure 18-24. Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave) (3/3)

(3) Stop condition



★ **Note** To cancel master wait, write "FFH" to IIC0 or set WREL0.

CHAPTER 19 INTERRUPT FUNCTIONS

19.1 Interrupt Function Types

The following three types of interrupt functions are used.

(1) Non-maskable interrupt

This interrupt is acknowledged even in an interrupt disabled state. It does not undergo priority control and is given top priority over all other interrupt requests. The other interrupt requests are held pending while the non-maskable interrupt is serviced.

The non-maskable interrupt generates a standby release signal and releases the HALT mode during main system clock operation.

The non-maskable interrupt has only interrupt request from the watchdog timer.

(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 (PR0L, PR0H, PR1L).

Multiple high priority interrupts can be applied to low priority interrupts. If two or more interrupts with the same priority are simultaneously generated, each interrupt has a predetermined priority (see **Table 19-1**).

The maskable interrupt generates a standby release signal and releases the STOP and HALT modes.

Five external interrupt requests and 13 internal interrupt requests are incorporated as maskable interrupts.

(3) Software interrupt

This is a vectored interrupt to be generated by executing the BRK instruction. It is acknowledged even in an interrupt disabled state. The software interrupt does not undergo interrupt priority control.

19.2 Interrupt Sources and Configuration

A total of 20 interrupt sources exist among non-maskable, maskable, and software interrupts (see **Table 19-1**).

Remark A non-maskable interrupt or a maskable interrupt (internal) can be selected as the watchdog timer interrupt (INTWDT).

Table 19-1. Interrupt Source List

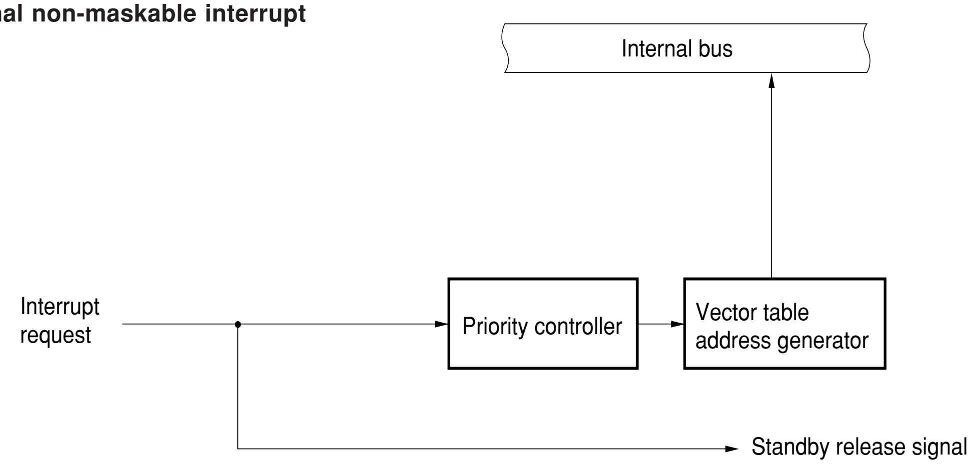
Interrupt Type	Default Priority[Note 1]	Interrupt Source		Internal/ External	Vector Table Address	Basic Configuration Type[Note 2]
		Name	Trigger			
Non-maskable	–	INTWDT	Watchdog timer overflow (with watchdog timer mode 1 selected)	Internal	0004H	(A)
Maskable	0	INTWDT	Watchdog timer overflow (with interval timer mode selected)			(B)
	1	INTP0	Pin input edge detection	External	0006H	(C)
	2	INTP1			0008H	
	3	INTP2			000AH	
	4	INTP3			000CH	
	5	INTSER0	Serial interface UART0 reception error generation	Internal	000EH	(B)
	6	INTSR0	End of serial interface UART0 reception		0010H	
	7	INTST0	End of serial interface UART0 transmission		0012H	
	8	INTCSI30	End of serial interface SIO30 transfer		0014H	
	9	INTCSI31	End of serial interface SIO31 transfer **[only for μPD780024A, 780034A Subseries]**		0016H	
	10	INTIIC0	End of serial interface IIC0 transfer **[only for μPD780024AY, 780034AY Subseries]**		0018H	
	11	INTWTI	Reference time interval signal from watch timer		001AH	
	12	INTTM00	Match between TM0 and CR00 (when CR00 is specified as compare register) Detection of TI01 valid edge (when CR00 is specified as capture register)		001CH	
	13	INTTM01	Match between TM0 and CR01 (when CR01 is specified as compare register) Detection of TI00 valid edge (when CR01 is specified as capture register)		001EH	
	14	INTTM50	Match between TM50 and CR50		0020H	
	15	INTTM51	Match between TM51 and CR51		0022H	
	16	INTAD0	End of A/D converter conversion		0024H	
	17	INTWT	Watch timer overflow		0026H	
	18	INTKR	Port 4 falling edge detection	External	0028H	(D)
Software	–	BRK	BRK instruction execution	–	003EH	(E)

Notes 1. The default priority is the priority applicable when two or more maskable interrupts are generated simultaneously. 0 is the highest priority, and 18 is the lowest.

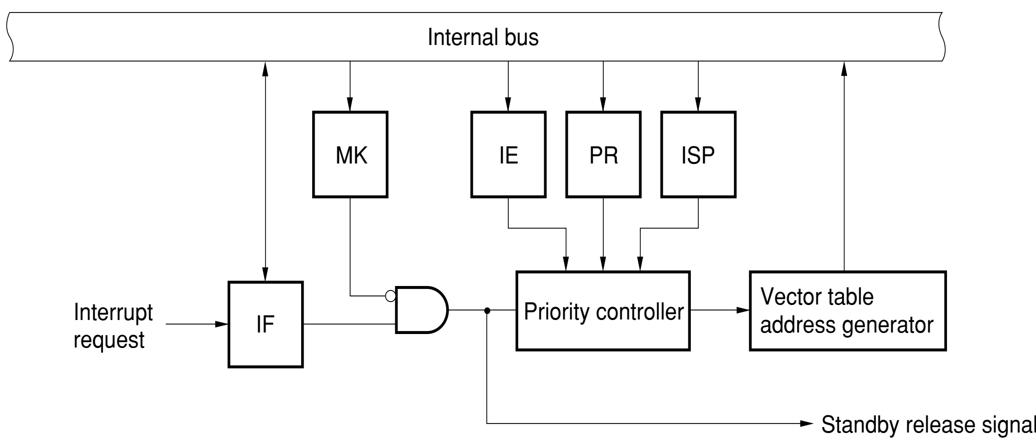
2. Basic configuration types (A) to (E) correspond to (A) to (E) in Figure 19-1.

Figure 19-1. Basic Configuration of Interrupt Function (1/2)

(A) Internal non-maskable interrupt



(B) Internal maskable interrupt



(C) External maskable interrupt (INTP0 to INTP3)

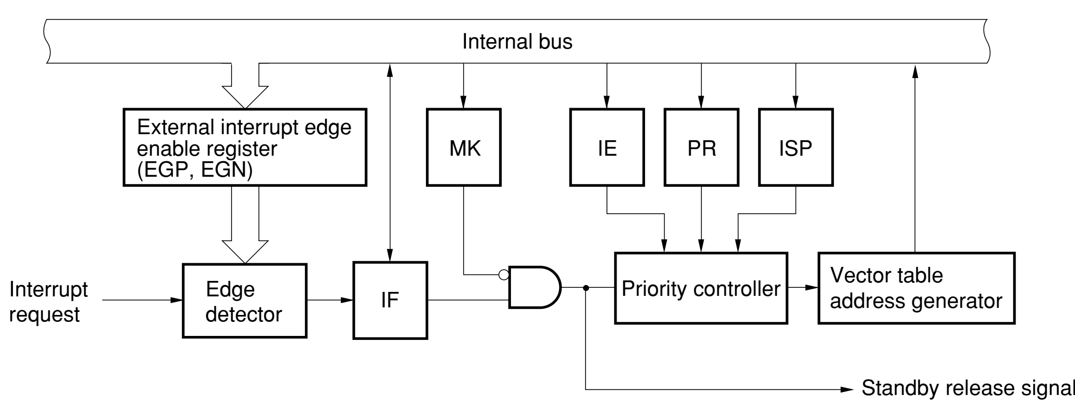
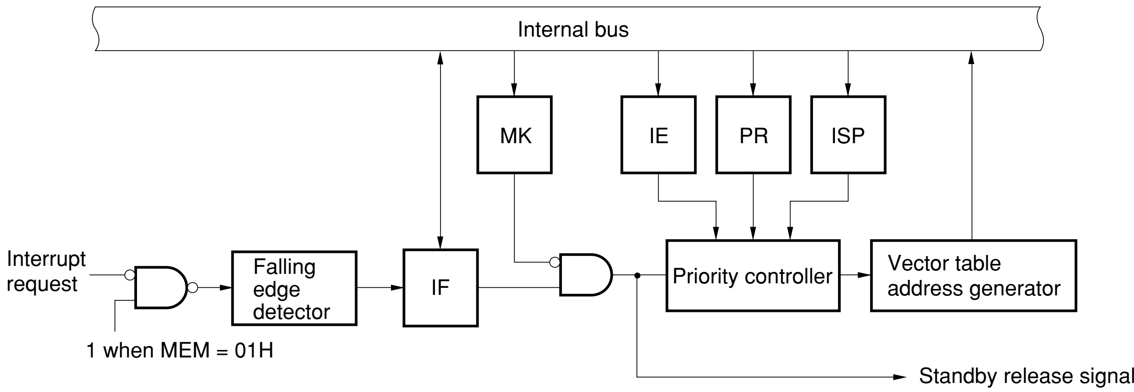
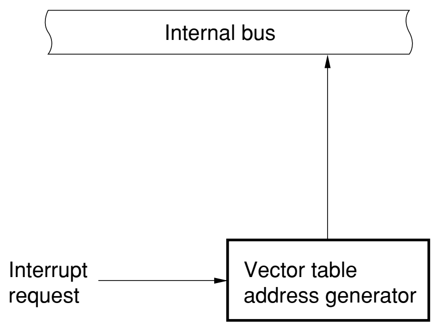


Figure 19-1. Basic Configuration of Interrupt Function (2/2)

(D) External maskable interrupt (INTKR)



★ **(E) Software interrupt**



IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specification flag
MEM: Memory expansion mode register

19.3 Interrupt Function Control Registers

The following 6 types of registers are used to control the interrupt functions.

- Interrupt request flag registers (IF0L, IF0H, IF1L)
- Interrupt mask flag registers (MK0L, MK0H, MK1L)
- Priority specification flag registers (PR0L, PR0H, PR1L)
- External interrupt rising edge enable register (EGP)
- External interrupt falling edge enable register (EGN)
- Program status word (PSW)

Table 19-2 gives a list of interrupt request flags, interrupt mask flags, and priority specification flags corresponding to interrupt request sources.

Table 19-2. Flags Corresponding to Interrupt Request Sources

Interrupt Source	Interrupt Request Flag		Interrupt Mask Flag		Priority Specification Flag	
		Register		Register		Register
INTWDT	WDTIF Note 1	IF0L	WDTMK	MK0L	WDTPR	PR0L
INTP0	PIF0		PMK0		PPR0	
INTP1	PIF1		PMK1		PPR1	
INTP2	PIF2		PMK2		PPR2	
INTP3	PIF3		PMK3		PPR3	
INTSER0	SERIF0		SERMK0		SERPR0	
INTSR0	SRIF0		SRMK0		SRPR0	
INTST0	STIF0		STMK0		STPR0	
INTCSI30	CSIIF30	IF0H	CSIMK30	MK0H	CSIPR30	PR0H
INTCSI31 Note 2	CSIIF31 Note 2		CSIMK31 Note 2		CSIPR31 Note 2	
INTIIC0 Note 3	IICIF0 Note 3		IICMK0 Note 3		IICPR0 Note 3	
INTWTI	WTIIF		WTIMK		WTIPR	
INTTM00	TMIF00		TMMK00		TMPR00	
INTTM01	TMIF01		TMMK01		TMPR01	
INTTM50	TMIF50		TMMK50		TMPR50	
INTTM51	TMIF51		TMMK51		TMPR51	
INTAD0	ADIF0	IF1L	ADMK0	MK1L	ADPR0	PR1L
INTWT	WTIF		WTMK		WTPR	
INTKR	KRIF		KRMK		KRPR	

Notes 1. Interrupt control flag when watchdog timer is used as interval timer

2. μPD780024A, 780034A Subseries only

3. μPD780024AY, 780034AY Subseries only

(1) Interrupt request flag registers (IF0L, IF0H, IF1L)

The interrupt request flags are set to 1 when the corresponding interrupt request is generated or an instruction is executed. They are cleared to 0 when an instruction is executed upon acknowledgment of an interrupt request or upon application of $\overline{\text{RESET}}$ input.

IF0L, IF0H, and IF1L are set by a 1-bit or 8-bit memory manipulation instruction. When IF0L and IF0H are combined to form 16-bit register IF0, they are set by a 16-bit memory manipulation instruction.

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

Figure 19-2. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L)

Address: FFE0H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0L	STIF0	SRIF0	SERIF0	PIF3	PIF2	PIF1	PIF0	WDTIF

Address: FFE1H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0H	TMIF51	TMIF50	TMIF01	TMIF00	WTIIF	IICIF0[Note 1]	CSIIF31[Note 2]	CSIIF30

Address: FFE2H After reset: 00H R/W

Symbol	7	6	5	4	3	<2>	<1>	<0>
IF1L	0	0	0	0	0	KRIF	WTIF	ADIF0

XXIFX	Interrupt request flag
0	No interrupt request signal is generated
1	Interrupt request signal is generated, interrupt request status

Notes 1. Incorporated only in the μPD780024AY, 780034AY Subseries. Be sure to clear 0 for the μPD780024A, 780034A Subseries.

2. Incorporated only in the μPD780024A, 780034A Subseries. Be sure to clear 0 for the μPD780024AY, 780034AY Subseries.

Cautions 1. The WDTIF flag is R/W enabled only when the watchdog timer is used as the interval timer. If watchdog timer mode 1 is used, set the WDTIF flag to 0.

2. Be sure to clear bits 3 to 7 of IF1L to 0.

3. When operating a timer, serial interface, or A/D converter after standby release, run it once after clearing an interrupt request flag. An interrupt request flag may be set by noise.

4. When an interrupt is acknowledged, the interrupt request flag is automatically cleared, and then processing of the interrupt routine is started.

★ **5. When the interrupt request flag register is being manipulated (including 1-bit memory manipulation), if an interrupt request corresponding to other flags in the same register is generated, the flag corresponding to the interrupt request may not be set (1).**

(2) Interrupt mask flag registers (MK0L, MK0H, MK1L)

The interrupt mask flags are used to enable/disable the corresponding maskable interrupt service.

MK0L, MK0H, and MK1L are set by a 1-bit or 8-bit memory manipulation instruction. When MK0L and MK0H are combined to form a 16-bit register MK0, they are set by a 16-bit memory manipulation instruction.

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

Figure 19-3. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L)

Address: FFE4H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK0L	STMK0	SRMK0	SERMK0	PMK3	PMK2	PMK1	PMK0	WDTMK

Address: FFE5H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK0H	TMMK51	TMMK50	TMMK01	TMMK00	WTIMK	IICMK0[Note 1]	CSIMK31[Note 2]	CSIMK30

Address: FFE6H After reset: FFH R/W

Symbol	7	6	5	4	3	<2>	<1>	<0>
MK1L	1	1	1	1	1	KRMK	WTMK	ADMK0

XXMKX	Interrupt servicing control
0	Interrupt servicing enabled
1	Interrupt servicing disabled

Notes 1. Incorporated only in the μPD780024AY, 780034AY Subseries. Be sure to set 1 for the μPD780024A, 780034A Subseries.

2. Incorporated only in the μPD780024A, 780034A Subseries. Be sure to set 1 for the μPD780024AY, 780034AY Subseries.

Cautions 1. If the watchdog timer is used in watchdog timer mode 1, the contents of the WDTMK flag become undefined when read.

2. Because port 0 pins have an alternate function as 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, 1 should be set in the interrupt mask flag before using the output mode.

3. Be sure to set bits 3 to 7 of MK1L to 1.

(3) Priority specification flag registers (PR0L, PR0H, PR1L)

The priority specification flags are used to set the corresponding maskable interrupt priority orders.

PR0L, PR0H, and PR1L are set by a 1-bit or 8-bit memory manipulation instruction. If PR0L and PR0H are combined to form 16-bit register PR0, they are set by a 16-bit memory manipulation instruction.

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

Figure 19-4. Format of Priority Specification Flag Registers (PR0L, PR0H, PR1L)

Address: FFE8H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR0L	STPR0	SRPR0	SERPR0	PPR3	PPR2	PPR1	PPR0	WDTPR

Address: FFE9H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR0H	TMPR51	TMPR50	TMPR01	TMPR00	WTIPR	IICPR0**Note 1**	CSIPR31**Note 2**	CSIPR30

Address: FFEAH After reset: FFH R/W

Symbol	7	6	5	4	3	<2>	<1>	<0>
PR1L	1	1	1	1	1	KRPR	WTPR	ADPR0

XXPRX	Priority level selection
0	High priority level
1	Low priority level

Notes 1. Incorporated only in the μPD780024AY, 780034AY Subseries. Be sure to set 1 for the μPD780024A, 780034A Subseries.

2. Incorporated only in the μPD780024A, 780034A Subseries. Be sure to set 1 for the μPD780024AY, 780034AY Subseries.

Cautions 1. If the watchdog timer is used in watchdog timer mode 1, set the WDTPR flag to 1.

2. Be sure to set bits 3 to 7 of PR1L to 1.

(4) External interrupt rising edge enable register (EGP), external interrupt falling edge enable register (EGN)

These registers specify the valid edge for INTP0 to INTP3.

EGP and EGN are set by a 1-bit or 8-bit memory manipulation instruction.

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

Figure 19-5. Format of External Interrupt Rising Edge Enable Register (EGP), External Interrupt Falling Edge Enable Register (EGN)

Address: FF48H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGP	0	0	0	0	EGP3	EGP2	EGP1	EGP0

Address: FF49H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGN	0	0	0	0	EGN3	EGN2	EGN1	EGN0

EGPn	EGNn	INTPn pin valid edge selection (n = 0 to 3)
0	0	Interrupt disable
0	1	Falling edge
1	0	Rising edge
1	1	Both rising and falling edges

★ **Caution When the function is switched from external interrupt request to port, edge detection may be performed. Therefore, clear EGPn and EGNn to 0 before switching to the port mode.**

(5) Program status word (PSW)

The program status word is a register to hold the instruction execution result and the current status for an interrupt request. The IE flag to set maskable interrupt enable/disable and the ISP flag to control nesting processing are mapped.

Besides 8-bit read/write, this register can carry out operations with a bit manipulation instruction and dedicated instructions (EI and DI). When a vectored interrupt request is acknowledged, if the BRK instruction is executed, the contents of PSW are automatically saved into a stack and the IE flag is reset to 0. If a maskable interrupt request is acknowledged, the contents of the priority specification flag of the acknowledged interrupt are transferred to the ISP flag. The PSW contents are also saved into the stack with the PUSH PSW instruction. They are restored from the stack with the RETI, RETB, and POP PSW instructions.

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

Figure 19-6. Program Status Word Format

	7	6	5	4	3	2	1	0	After reset
PSW	IE	Z	RBS1	AC	RBS0	0	ISP	CY	02H

Used when normal instruction is executed

ISP	Priority of interrupt currently being serviced
0	High-priority interrupt servicing (Low-priority interrupt disable)
1	Interrupt request not acknowledged, or low-priority interrupt servicing (All maskable interrupts enable)

IE	Interrupt request acknowledge enable/disable
0	Disable
1	Enable

19.4 Interrupt Servicing Operations

19.4.1 Non-maskable interrupt request acknowledgment operation

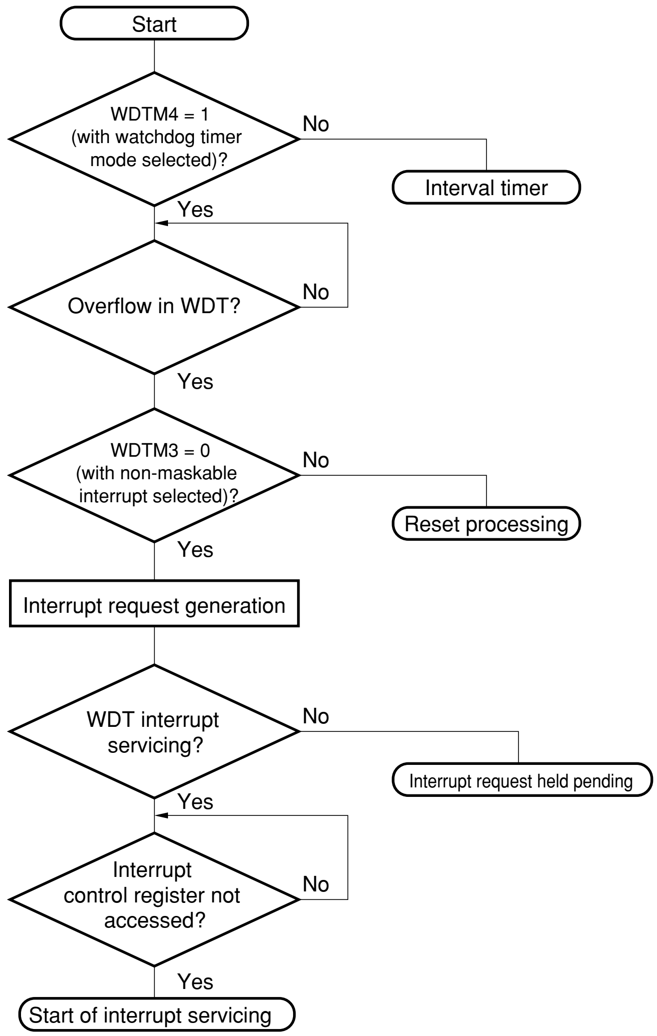
A non-maskable interrupt request is unconditionally acknowledged even in an interrupt acknowledgment disabled state. It does not undergo interrupt priority control and has the highest priority of all interrupts.

If a non-maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag and ISP flag are reset (0), and the contents of the vector table are loaded into the PC and branched.
★ At this time, the NMIS flag is set (1) to disable acknowledgment of multiple interrupts.

A new non-maskable interrupt request generated during execution of a non-maskable interrupt servicing program is acknowledged after the current non-maskable interrupt servicing program is terminated (following RETI instruction execution) and one main routine instruction has been executed. However, if a new non-maskable interrupt request is generated twice or more during non-maskable interrupt servicing program execution, only one non-maskable interrupt request is acknowledged after termination of the non-maskable interrupt servicing program. Figures 19-7, 19-8, and 19-9 show the flowchart of non-maskable interrupt request generation through acknowledgment, the acknowledgment timing of a non-maskable interrupt request, and the acknowledgment operation when multiple non-maskable interrupt requests are generated, respectively.

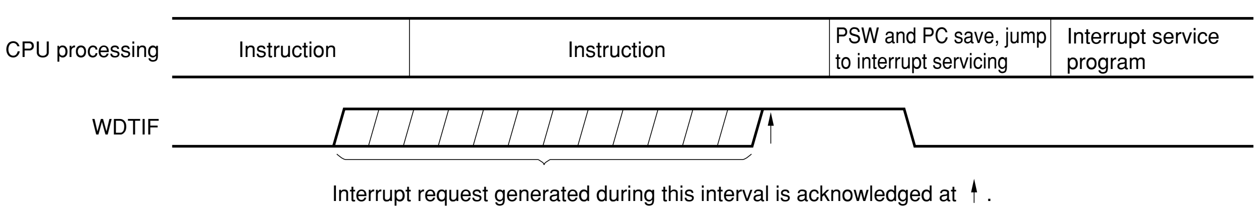
★ **Remark** When a non-maskable interrupt is acknowledged, the NMIS flag is set (1). It is cleared (0) by the RET or RETI instruction. When a non-maskable interrupt is generated, execute the RETI instruction to restore processing from the interrupt.

Figure 19-7. Non-Maskable Interrupt Request Generation to Acknowledge Flowchart



WDTM: Watchdog timer mode register
WDT: Watchdog timer

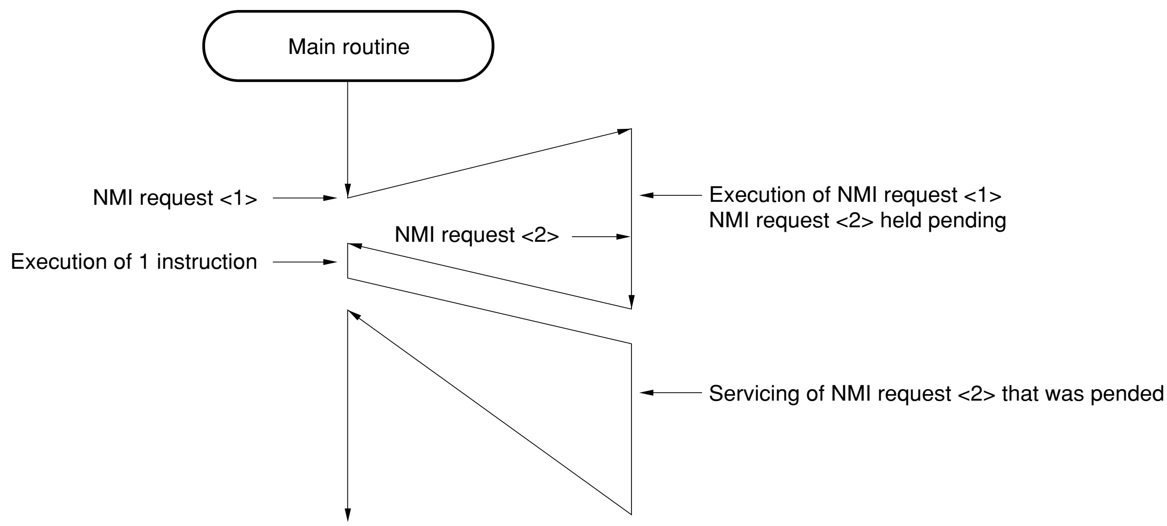
Figure 19-8. Non-Maskable Interrupt Request Acknowledge Timing



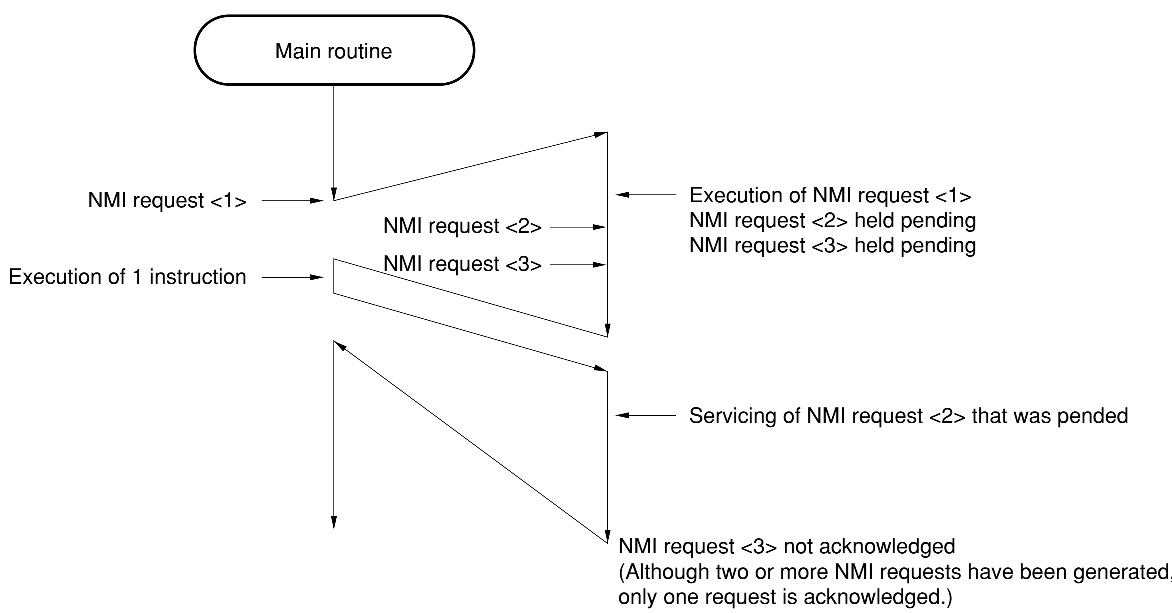
WDTIF: Watchdog timer interrupt request flag

Figure 19-9. Non-Maskable Interrupt Request Acknowledge Operation

(a) If a 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



19.4.2 Maskable interrupt request acknowledgment operation

A maskable interrupt request becomes acknowledgeable when an interrupt request flag is set to 1 and the mask (MK) flag corresponding to that interrupt request is cleared to 0. A vectored interrupt request is acknowledged if interrupts are enabled (when the IE flag is set to 1). However, a low-priority interrupt request is not acknowledged during servicing of a higher priority interrupt request (when the ISP flag is reset to 0).

★ Moreover, even if the EI instruction is executed during execution of a non-maskable interrupt servicing program (NMIS = 1), neither non-maskable interrupt requests nor maskable interrupt requests are acknowledged.

The times from generation of a maskable interrupt request until interrupt servicing is performed are listed in Table 19-3 below.

For the interrupt request acknowledgment timing, see **Figures 19-11** and **19-12**.

Table 19-3. Times from Generation of Maskable Interrupt Until Servicing

	Minimum Time	Maximum Time[Note]
When ××PR = 0	7 clocks	32 clocks
When ××PR = 1	8 clocks	33 clocks

Note If an interrupt request is generated just before a divide instruction, the wait time becomes longer.

Remark 1 clock: $1/f_{CPU}$ (f_{CPU}: CPU clock)

If two or more interrupt requests are generated simultaneously, the request with a higher priority level specified by the priority specification flag is acknowledged first. If two or more interrupt requests have the same priority level, the request with the highest default priority is acknowledged first.

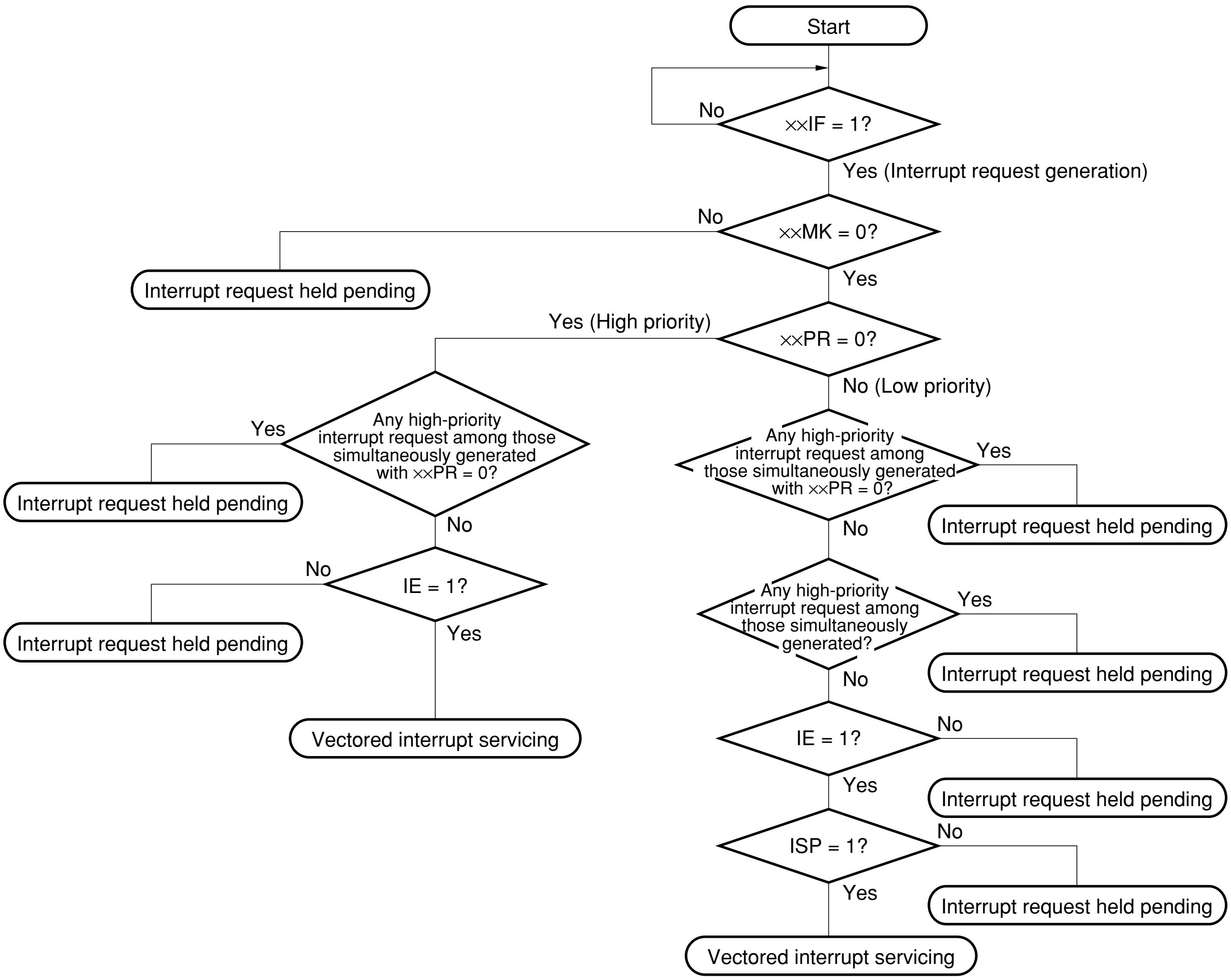
An interrupt request that is held pending is acknowledged when it becomes acknowledgeable.

Figure 19-10 shows the interrupt request acknowledgment algorithm.

If a maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag is reset (0), and the contents of the priority specification flag corresponding to the acknowledged interrupt are transferred to the ISP flag. Further, the vector table data determined for each interrupt request is loaded into the PC and branched.

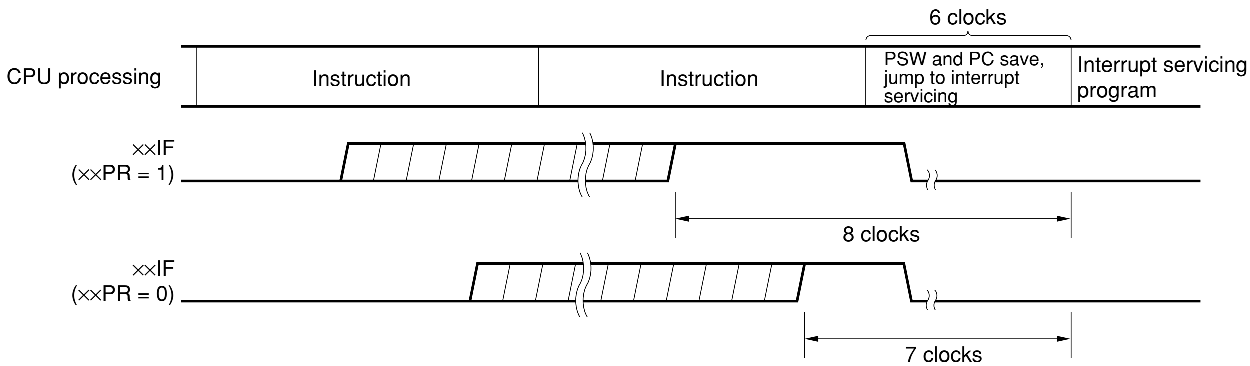
Return from an interrupt is possible using the RETI instruction.

Figure 19-10. Interrupt Request Acknowledge Processing Algorithm



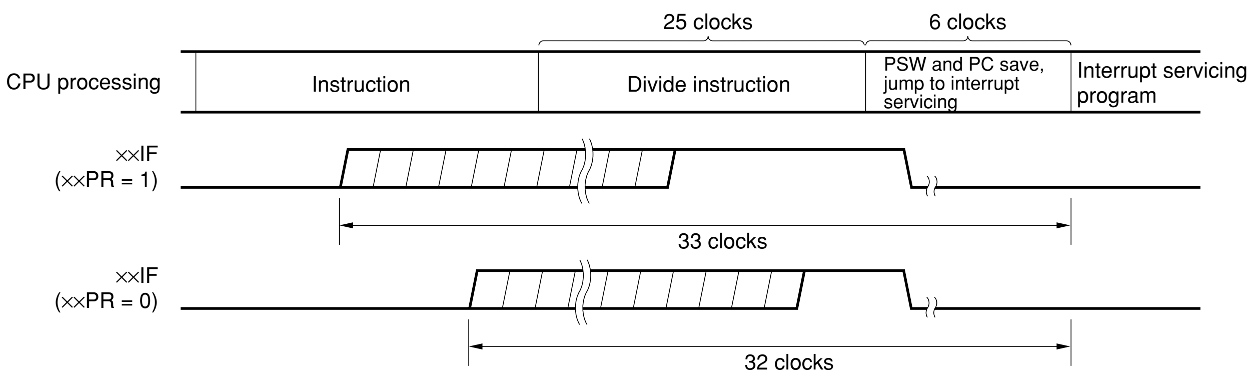
××IF: Interrupt request flag
××MK: Interrupt mask flag
××PR: Priority specification flag
IE: Flag that controls acknowledge of maskable interrupt request (1 = enable, 0 = disable)
ISP: Flag that indicates the priority level of the interrupt currently being serviced (0 = high-priority interrupt servicing, 1 = no interrupt request acknowledged, or low-priority interrupt servicing)

Figure 19-11. Interrupt Request Acknowledge Timing (Minimum Time)



Remark 1 clock: 1/f_{CPU} (f_{CPU}: CPU clock)

Figure 19-12. Interrupt Request Acknowledge Timing (Maximum Time)



Remark 1 clock: 1/f_{CPU} (f_{CPU}: CPU clock)

19.4.3 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 are saved into the stacks in the order of the program status word (PSW), then program counter (PC), the IE flag is reset (0), and the contents of the vector table (003EH, 003FH) are loaded into PC and branched.

Return from a software interrupt is possible with the RETB instruction.

Caution Do not use the RETI instruction for returning from the software interrupt.

19.4.4 Nesting interrupt servicing

Nesting occurs when another interrupt request is acknowledged during execution of an interrupt.

Nesting does not occur unless the interrupt request acknowledge enable state is selected (IE = 1) (except non-maskable interrupts). Also, when an interrupt request is acknowledged, interrupt request acknowledge becomes disabled (IE = 0). Therefore, to enable nesting, it is necessary to set (1) the IE flag with the EI instruction during interrupt servicing to enable interrupt acknowledge.

Moreover, even if interrupts are enabled, nesting may not be enabled, this being subject to interrupt priority control. Two types of priority control are available: default priority control and programmable priority control. Programmable priority control is used for nesting.

In the interrupt enable state, if an interrupt request with a priority equal to or higher than that of the interrupt currently being serviced is generated, it is acknowledged for nesting. If an interrupt with a priority lower than that of the interrupt currently being serviced is generated during interrupt servicing, it is not acknowledged for nesting.

Interrupt requests that are not enabled because of the interrupt disable state or they have a lower priority are held pending. When servicing of the current interrupt ends, the pended interrupt request is acknowledged following execution of at least one main processing instruction execution.

Nesting is not possible during non-maskable interrupt servicing.

Table 19-4 shows interrupt requests enabled for nesting and Figure 19-13 shows nesting examples.

★ **Table 19-4. Interrupt Requests Enabled for Nesting During Interrupt Servicing**

Nesting Request / Interrupt Being Serviced		Non-Maskable Interrupt Request	Maskable Interrupt Request				Software Interrupt Request
			PR = 0		PR = 1		
			IE = 1	IE = 0	IE = 1	IE = 0	
Non-maskable interrupt		×	×	×	×	×	○
Maskable interrupt	ISP = 0	○	○	×	×	×	○
	ISP = 1	○	○	×	○	×	○
Software interrupt		○	○	×	○	×	○

Remarks 1. ○: Nesting enabled

2. ×: Nesting disabled

3. ISP and IE are flags contained in the PSW.

ISP = 0: An interrupt with higher priority is being serviced.

ISP = 1: No interrupt request has been acknowledged, or an interrupt with a lower priority is being serviced.

IE = 0: Interrupt request acknowledgment is disabled.

IE = 1: Interrupt request acknowledgment is enabled.

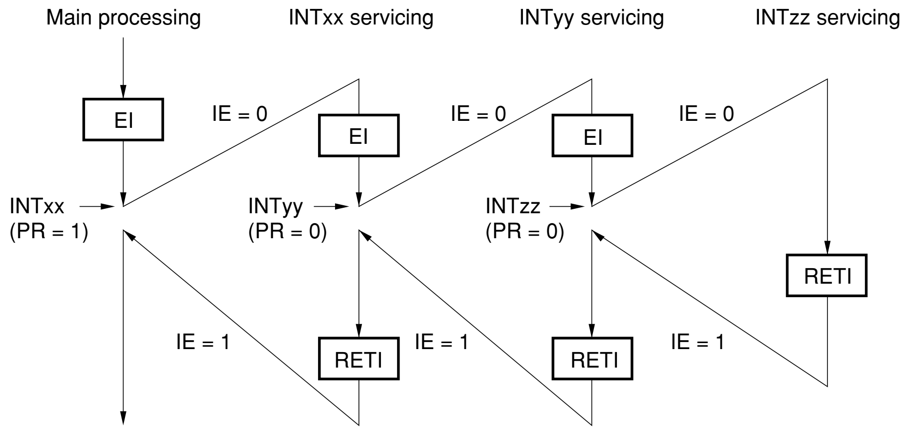
4. PR is a flag contained in PR0L, PR0H, and PR1L.

PR = 0: Higher priority level

PR = 1: Lower priority level

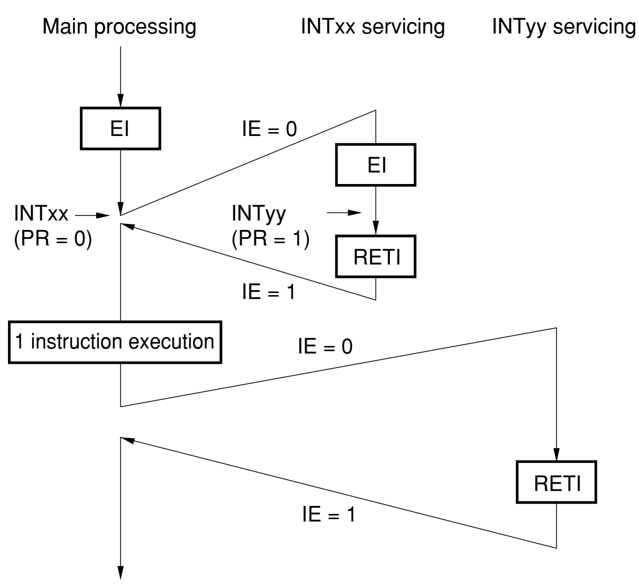
Figure 19-13. Nesting Examples (1/2)

Example 1. Nesting occurs twice



During servicing of interrupt INTxx, two interrupt requests, INTyy and INTzz, are acknowledged, and nesting takes place. Before each interrupt request is acknowledged, the EI instruction must always be issued to enable interrupt request acknowledgment.

Example 2. Nesting does not occur due to priority control

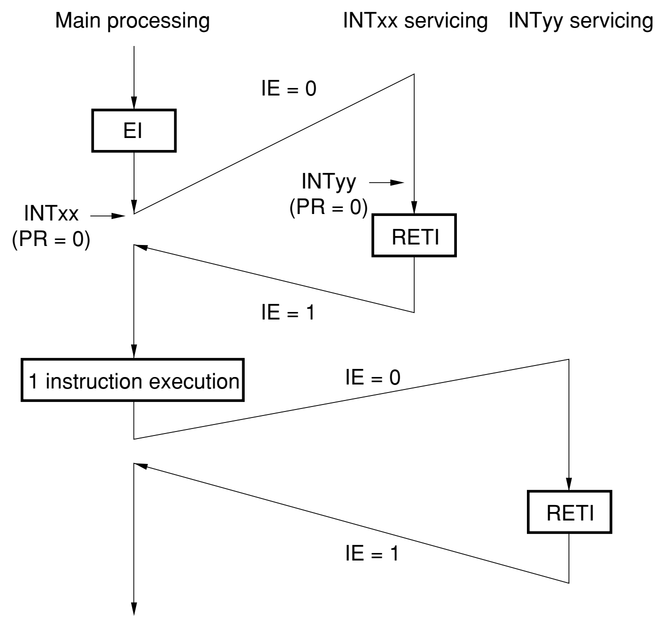


Interrupt request INTyy issued during servicing of interrupt INTxx is not acknowledged because its priority is lower than that of INTxx, and nesting does not take place. The INTyy interrupt request is held pending, and is acknowledged following execution of one main processing instruction.

PR = 0: Higher priority level
PR = 1: Lower priority level
IE = 0: Interrupt request acknowledgment disabled

Figure 19-13. Nesting Examples (2/2)

Example 3. Nesting does not occur because interrupts are not enabled



Interrupts are not enabled during servicing of interrupt INTxx (EI instruction is not issued), so interrupt request INTyy is not acknowledged and nesting does not take place. The INTyy interrupt request is held pending, and is acknowledged following execution of one main processing instruction.

PR = 0: Higher priority level
IE = 0: Interrupt request acknowledgment disabled

19.4.5 Interrupt request hold

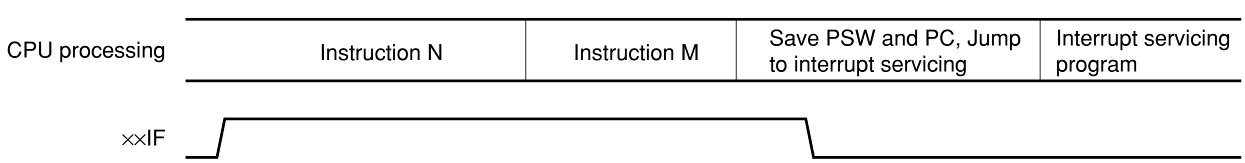
There are instructions where, even if an interrupt request is issued for them while another instruction is executed, request acknowledge is held pending until the end of execution of the next instruction. These instructions (interrupt request hold instructions) are listed below.

- MOV PSW, #byte
- MOV A, PSW
- MOV PSW, A
- MOV1 PSW.bit, CY
- MOV1 CY, PSW.bit
- AND1 CY, PSW.bit
- OR1 CY, PSW.bit
- XOR1 CY, PSW.bit
- SET1 PSW.bit
- CLR1 PSW.bit
- RETB
- RETI
- PUSH PSW
- POP PSW
- BT PSW.bit, $addr16
- BF PSW.bit, $addr16
- BTCLR PSW.bit, $addr16
- EI
- DI
- Manipulate instructions for the IF0L, IF0H, IF1L, MK0L, MK0H, MK1L, PR0L, PR0H, and PR1L registers

Caution The BRK instruction is not one of the above-listed interrupt request hold instructions. However, the software interrupt activated by executing the BRK instruction causes the IE flag to be cleared to 0. Therefore, even if a maskable interrupt request is generated during execution of the BRK instruction, the interrupt request is not acknowledged. However, a non-maskable interrupt request is acknowledged.

Figure 19-14 shows the timing with which interrupt requests are held pending.

Figure 19-14. Interrupt Request Hold



Remarks 1. Instruction N: Interrupt request hold instruction

2. Instruction M: Instruction other than interrupt request hold instruction

3. The ××PR (priority level) values do not affect the operation of ××IF (interrupt request).

★ Use the expanded-specification products of the μPD780024A and 780034A Subseries, under the conventional-specification conditions (f_X = 8.38 MHz: V_{DD} = 4.0 to 5.5 V, f_X = 5 MHz: V_{DD} = 2.7 to 5.5 V, f_X = 1.25 MHz: V_{DD} = 1.8 to 5.5 V).

The external device expansion function cannot be used under the expanded-specification conditions (high-speed operation).

20.1 External Device Expansion Function

The external device expansion function connects external devices to areas other than the internal ROM, RAM, and SFR. Connection of external devices uses ports 4 to 6. Ports 4 to 6 control address/data, read/write strobe, wait, address strobe, etc.

Table 20-1. Pin Functions in External Memory Expansion Mode

Pin Function When External Device Is Connected: Name	Pin Function When External Device Is Connected: Function	Alternate Function
AD0 to AD7	Multiplexed address/data bus	P40 to P47
A8 to A15	Address bus	P50 to P57
$\overline{RD}$	Read strobe signal	P64
$\overline{WR}$	Write strobe signal	P65
$\overline{WAIT}$	Wait signal	P66
ASTB	Address strobe signal	P67

Table 20-2. State of Port 4 to 6 Pins in External Memory Expansion Mode

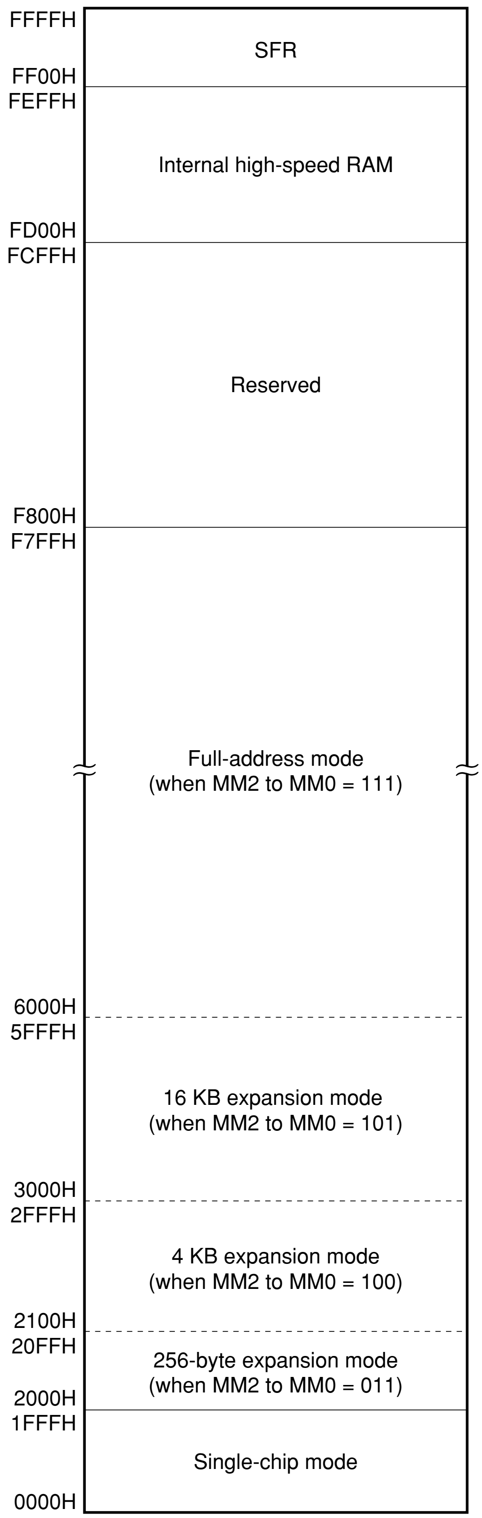
<table>
<tr><th rowspan="2">External Expansion Mode \ Port</th><th>Port 4</th><th colspan="8">Port 5</th><th colspan="4">Port 6</th></tr>
<tr><th>0 to 7</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>4</th><th>5</th><th>6</th><th>7</th></tr>
<tr><td>Single-chip mode</td><td>Port</td><td colspan="8">Port</td><td colspan="4">Port</td></tr>
<tr><td>256-byte expansion mode</td><td>Address/data</td><td colspan="8">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
<tr><td>4 KB expansion mode</td><td>Address/data</td><td colspan="4">Address</td><td colspan="4">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
<tr><td>16 KB expansion mode</td><td>Address/data</td><td colspan="6">Address</td><td colspan="2">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
<tr><td>Full-address mode</td><td>Address/data</td><td colspan="8">Address</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
</table>

Caution When the external wait function is not used, the $\overline{WAIT}$ pin can be used as a port in all modes.

The memory maps when the external device expansion function is used are as follows.

Figure 20-1. Memory Map When Using External Device Expansion Function (1/2)

(a) Memory map of μPD780021A, 780031A, 780021AY, 780031AY and of μPD78F0034A, 78F0034AY when internal ROM (flash memory) size is 8 KB



(b) Memory map of μPD780022A, 780032A, 780022AY, 780032AY and of μPD78F0034A, 78F0034AY when internal ROM (flash memory) size is 16 KB

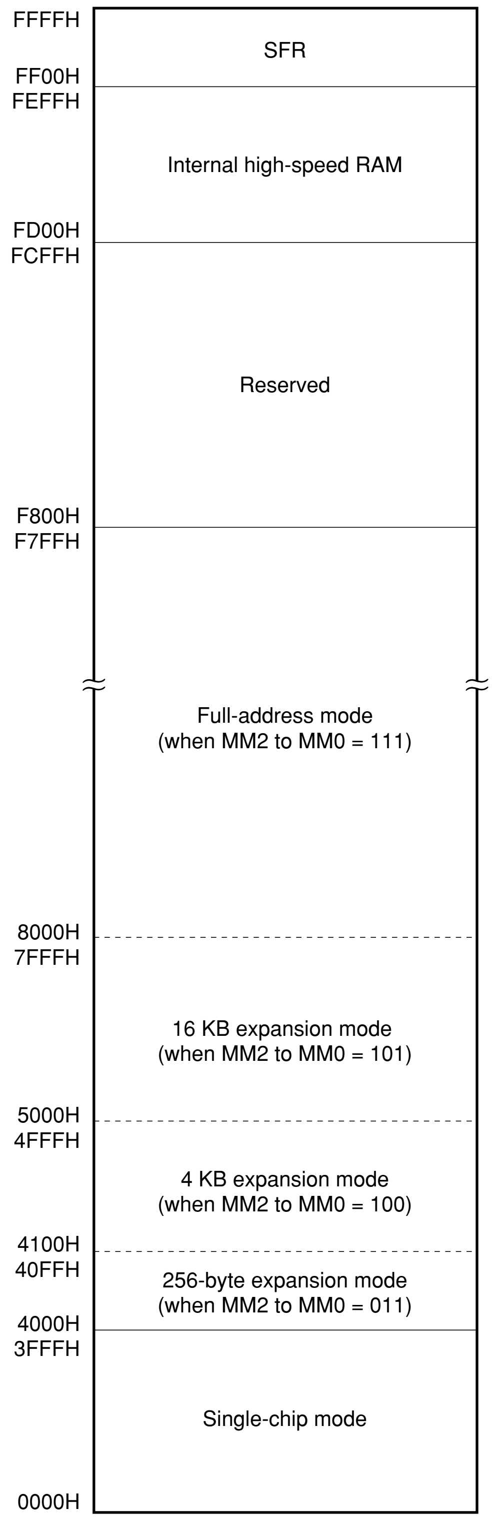
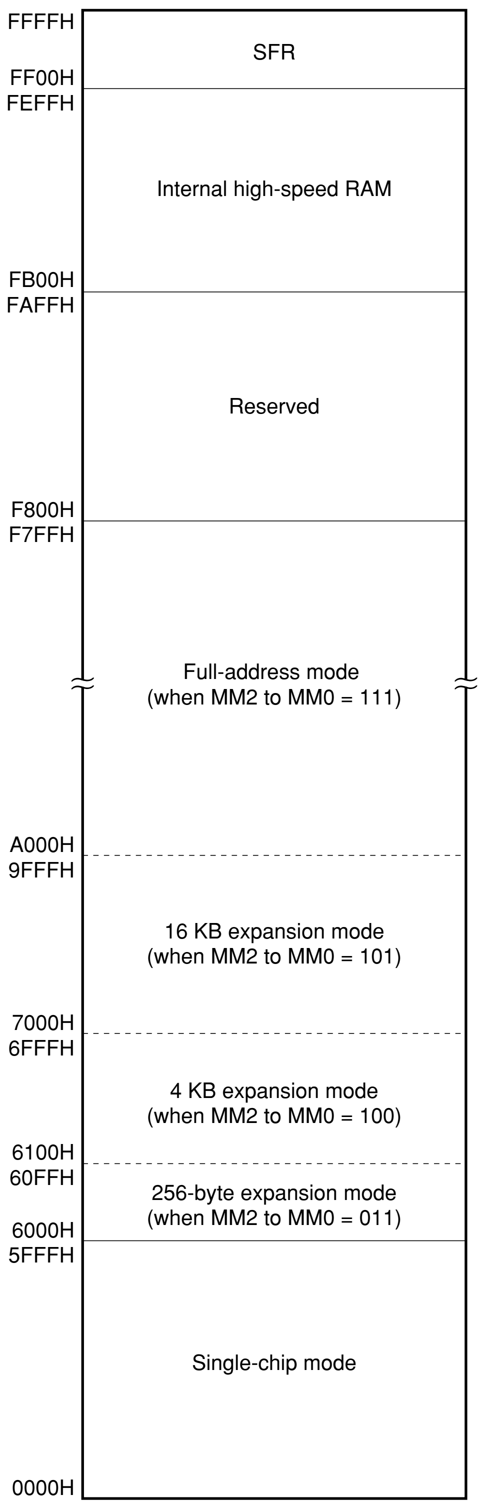
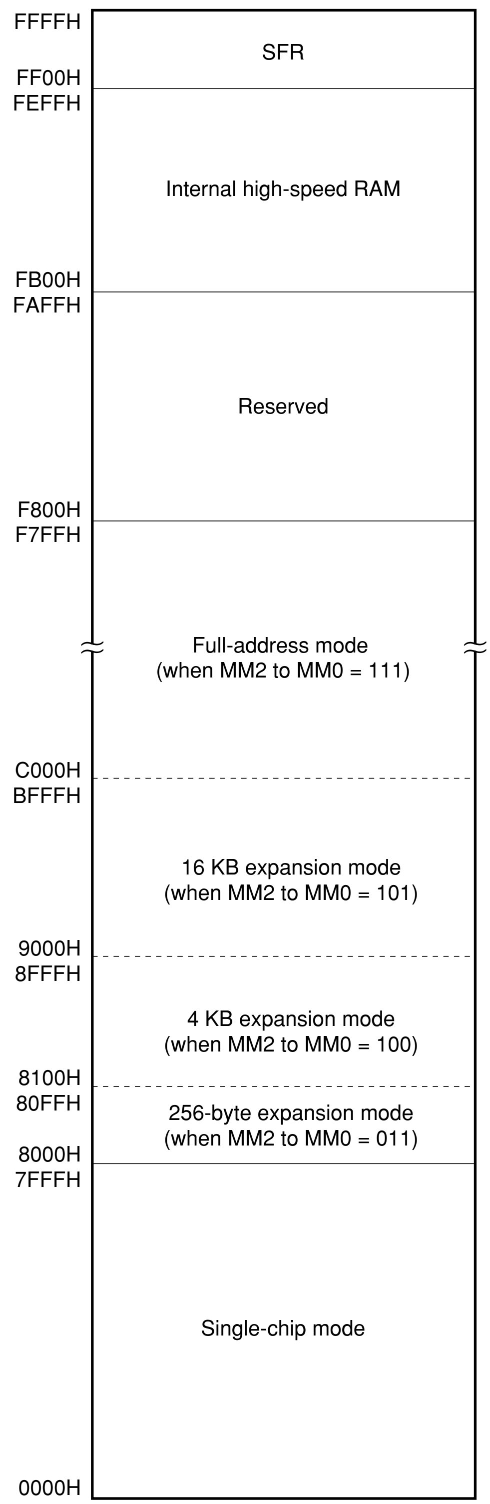


Figure 20-1. Memory Map When Using External Device Expansion Function (2/2)

(c) Memory map of μPD780023A, 780033A, 780023AY, 780033AY and of μPD78F0034A, 78F0034AY when internal ROM (flash memory) size is 24 KB



(d) Memory map of μPD780024A, 780034A, 780024AY, 780034AY and of μPD78F0034A, 78F0034AY when internal ROM (flash memory) size is 32 KB



20.2 External Device Expansion Function Control Registers

The external device expansion function is controlled by the following two registers.

- Memory expansion mode register (MEM)
- Memory expansion wait setting register (MM)

(1) Memory expansion mode register (MEM)

MEM sets the external expansion area.
MEM is set by a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears MEM to 00H.

Figure 20-2. Format of Memory Expansion Mode Register (MEM)

Address: FF47H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
MEM	0	0	0	0	0	MM2	MM1	MM0

<table>
<tr><th rowspan="2">MM2</th><th rowspan="2">MM1</th><th rowspan="2">MM0</th><th colspan="2" rowspan="2">Single-chip/memory expansion mode selection</th><th colspan="5">P40 to P47, P50 to P57, P64 to P67 pin state</th></tr>
<tr><th>P40 to P47</th><th>P50 to P53</th><th>P54, P55</th><th>P56, P57</th><th>P64 to P67</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td colspan="2">Single-chip mode</td><td colspan="5" rowspan="2">Port mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td colspan="2">Port 4 falling edge detection mode</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td rowspan="4">Memory expansion mode[Note 1]</td><td>256-byte mode</td><td rowspan="4">AD0 to AD7</td><td colspan="3">Port mode</td><td rowspan="4">P64 = $\overline{RD}$
P65 = $\overline{WR}$
P66 = $\overline{WAIT}$
P67 = ASTB</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>4 KB mode</td><td rowspan="3">A8 to A11</td><td colspan="2">Port mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>16 KB mode</td><td rowspan="2">A12, A13</td><td>Port mode</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Full-address mode[Note 2]</td><td>A14, A15</td></tr>
<tr><td colspan="3">Other than above</td><td colspan="7">Setting prohibited</td></tr>
</table>

★ **Notes 1.** When the memory expansion mode is set, if an area other than the external expansion area is accessed, the read value is undefined.

2. 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.

Caution When using the falling edge detection function of port 4, be sure to set MEM to 01H.

(2) Memory expansion wait setting register (MM)

MM sets the number of waits.

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

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

Figure 20-3. Format of Memory Expansion Wait Setting Register (MM)

Address: FFF8H After reset: 10H R/W

Symbol	7	6	5	4	3	2	1	0
MM	0	0	PW1	PW0	0	0	0	0

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

Cautions 1. When wait is controlled by the external wait pin, be sure to set the $\overline{\text{WAIT}}$/P66 pin to input mode (set bit 6 (PM66) of port mode register 6 (PM6) to 1).

★ **2. When wait is not controlled by the external wait pin, the $\overline{\text{WAIT}}$/P66 pin can be used as an I/O port pin.**

20.3 External Device Expansion Function Timing

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 in data read and instruction fetch from external memory.

During internal memory read, 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 in data write to external memory.

During internal memory write, 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) ASTB pin (Alternate function: P67)

Address strobe signal output pin. The address strobe signal is output regardless of data access and instruction fetch from external memory.

During internal memory access, the address strobe signal is output.

(5) AD0 to AD7, A8 to A15 pins (Alternate function: P40 to P47, P50 to P57)

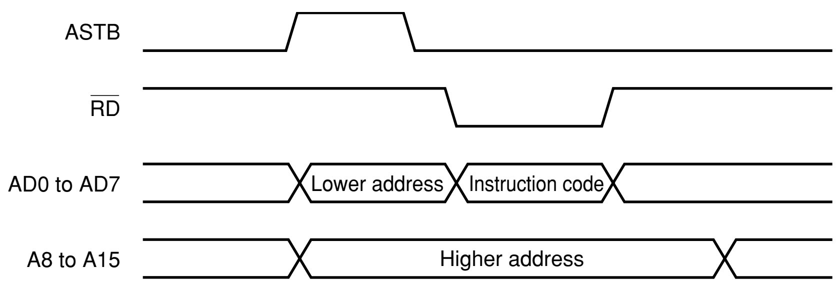
Address/data signal output pins. Valid signal is output or input during data accesses and instruction fetches from external memory.

These signals change even during internal memory access (output values are undefined).

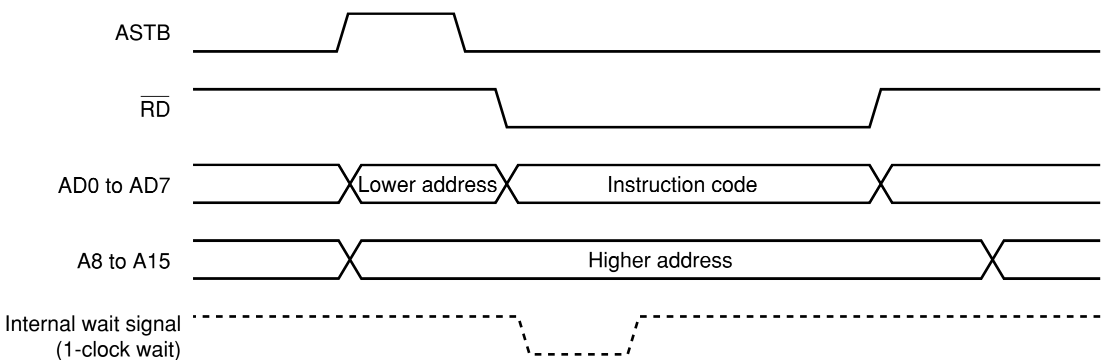
The timing charts are shown in Figures 20-4 to 20-7.

Figure 20-4. Instruction Fetch from External Memory

(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



(c) External wait (PW1, PW0 = 1, 1) setting

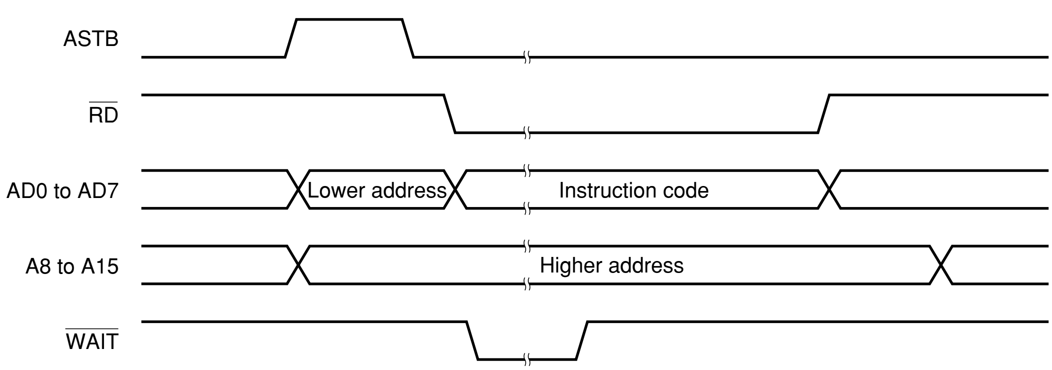
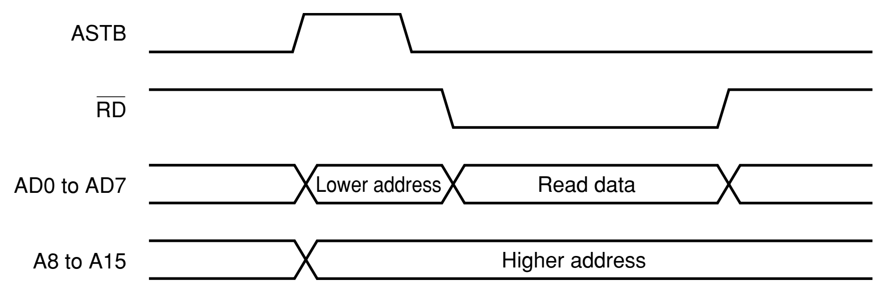
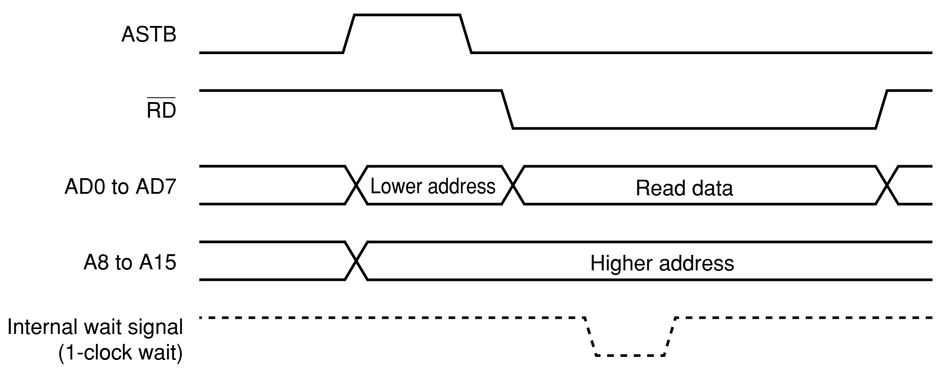


Figure 20-5. External Memory Read Timing

(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



(c) External wait (PW1, PW0 = 1, 1) setting

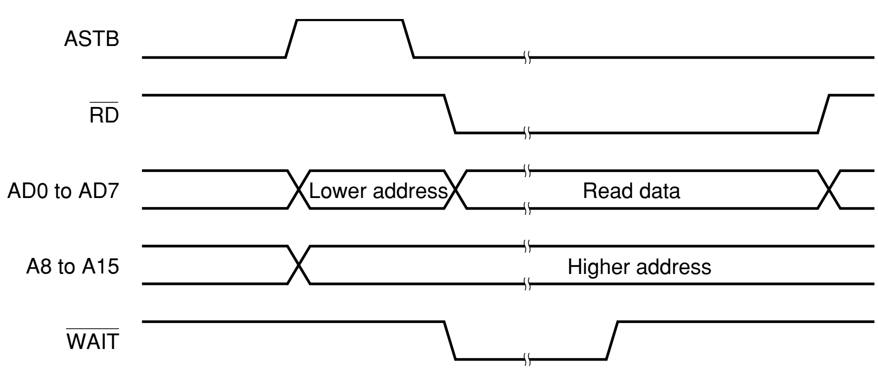
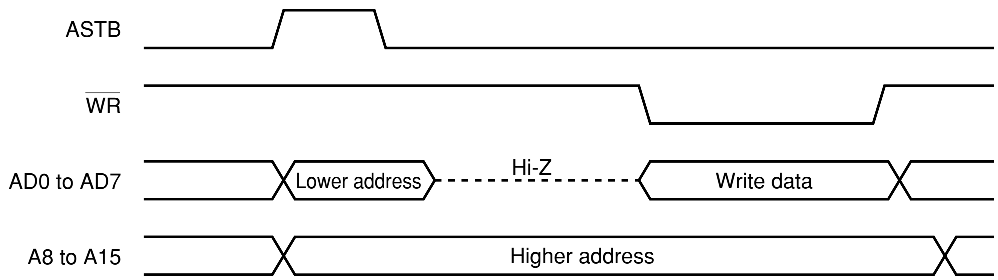
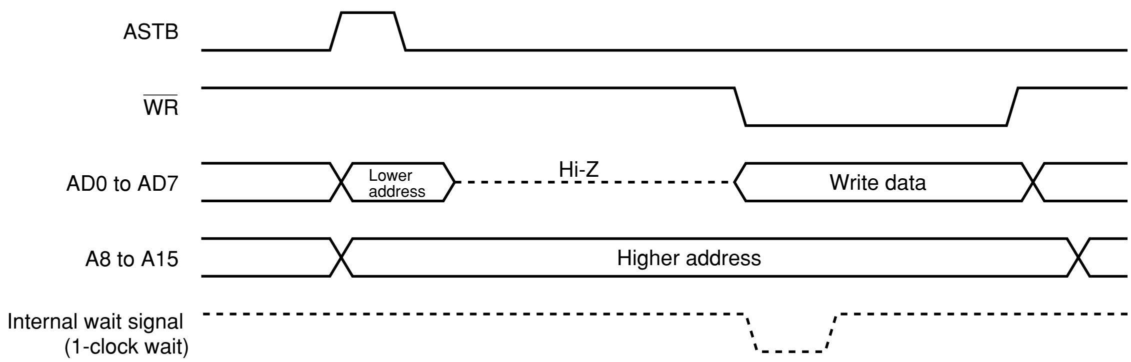


Figure 20-6. External Memory Write Timing

(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



(c) External wait (PW1, PW0 = 1, 1) setting

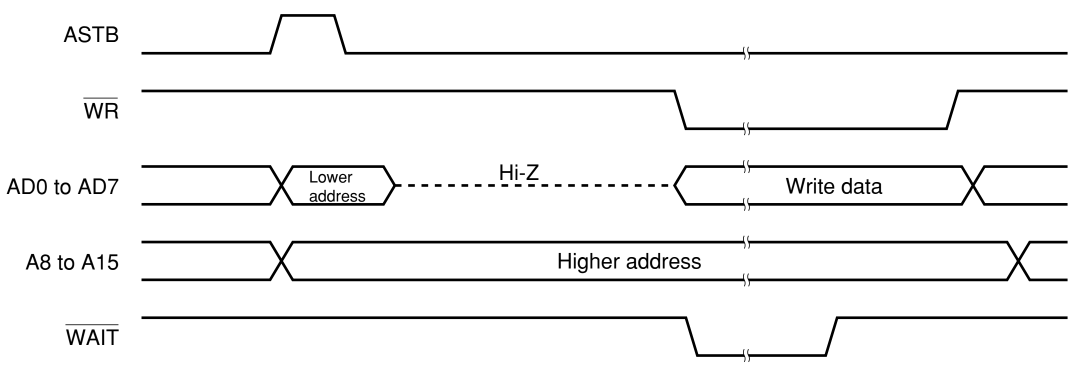
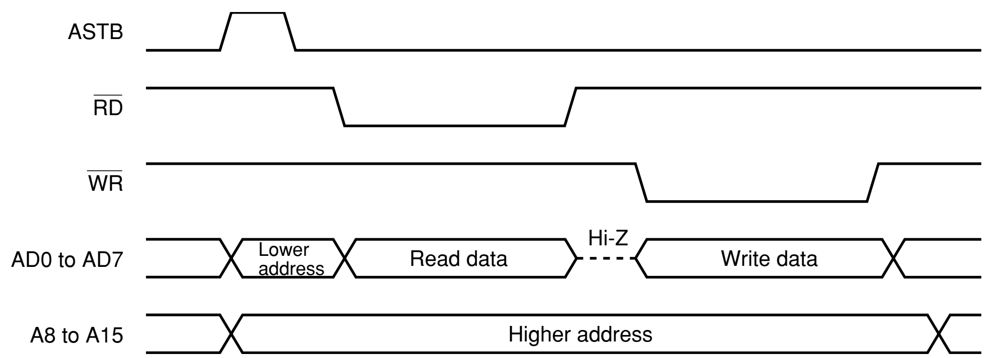
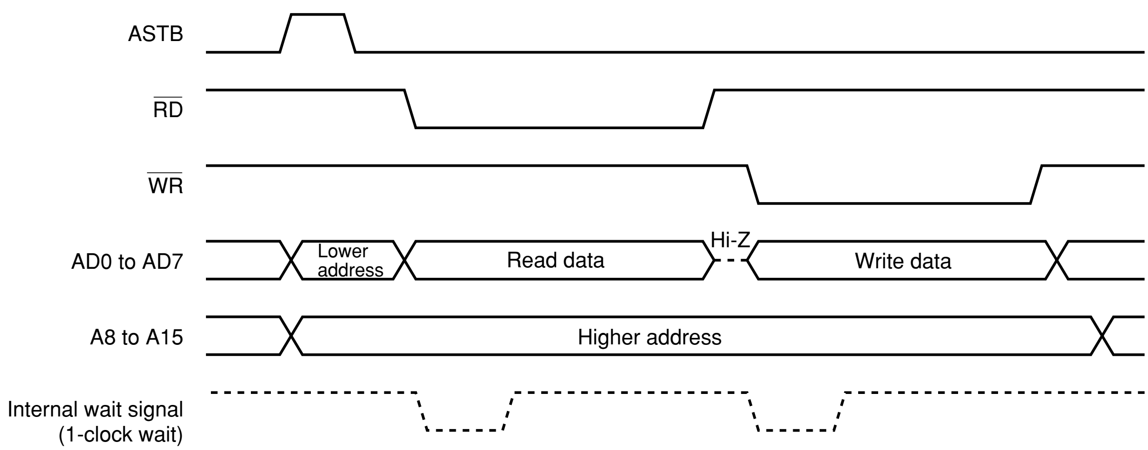


Figure 20-7. External Memory Read Modify Write Timing

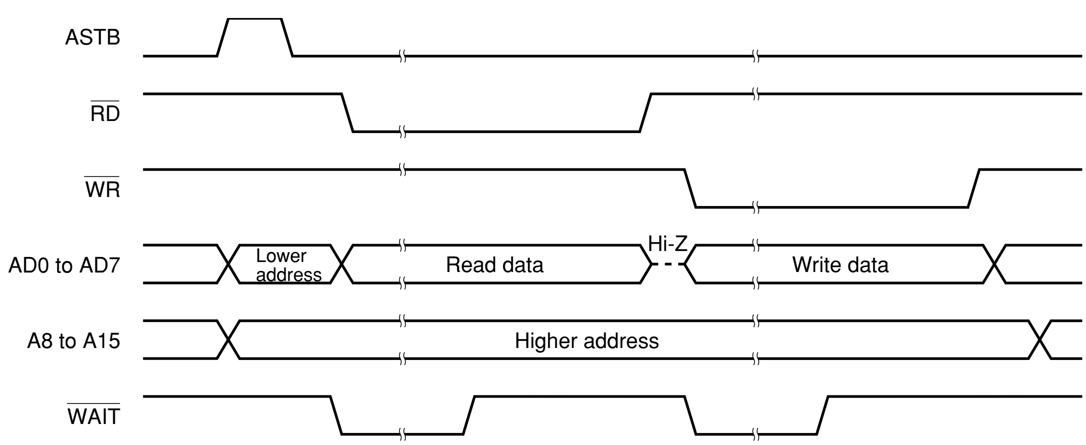
(a) No wait (PW1, PW0 = 0, 0) setting



(b) Wait (PW1, PW0 = 0, 1) setting



(c) External wait (PW1, PW0 = 1, 1) setting

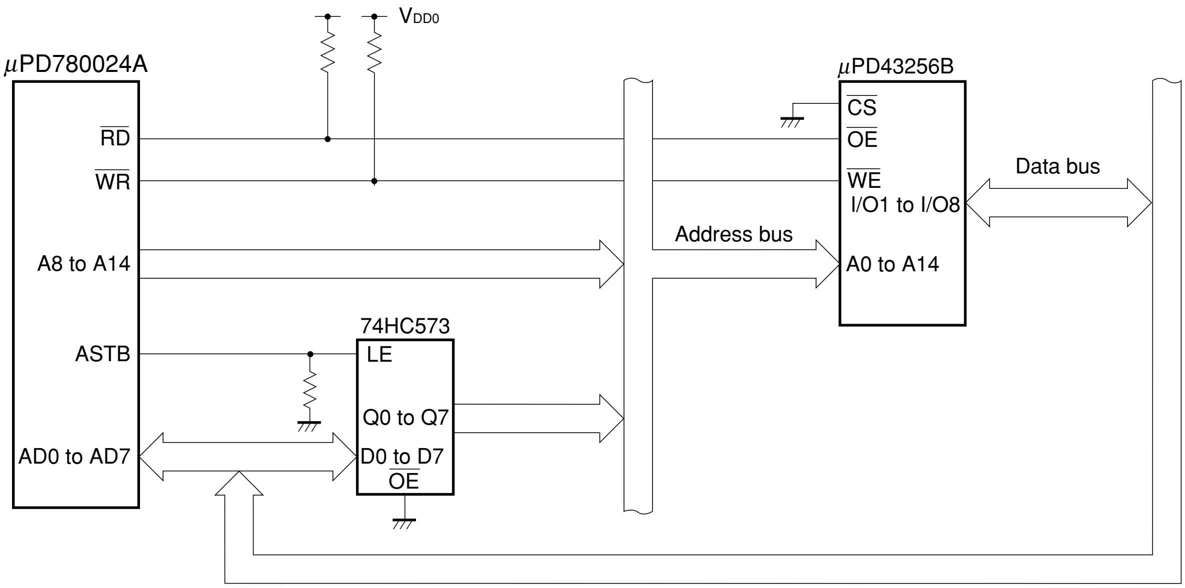


★ **Remark** The read-modify-write timing is that of an operation when a bit manipulation instruction is executed.

20.4 Example of Connection with Memory

This section provide an example of connecting the μPD780024A with external memory (in this example, SRAM) in Figure 20-8. 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 for internal ROM, and the addresses after 8000H for SRAM.

Figure 20-8. Connection Example of μPD780024A and Memory



CHAPTER 21 STANDBY FUNCTION

21.1 Standby Function and Configuration

21.1.1 Standby function

The standby function is designed to decrease power consumption of the system. The following two modes are available.

(1) HALT mode

HALT instruction execution sets the HALT mode. The HALT mode stops the CPU operation clock. The system clock oscillator continues oscillating. In this mode, power consumption is not decreased as much as in the STOP mode. However, the HALT mode is effective to restart operation immediately upon an interrupt request and to carry out intermittent operations.

(2) STOP mode

STOP instruction execution sets the STOP mode. In the STOP mode, the main system clock oscillator stops, stopping the whole system, thereby considerably reducing the CPU power consumption.

Data memory low-voltage hold (down to V_{DD} = 1.6 V) is possible. Thus, the STOP mode is effective to hold data memory contents with ultra-low power consumption. Because this mode can be released upon an interrupt request, it enables intermittent operations to be carried out.

However, because a wait time is required to stabilize oscillation after the STOP mode is released, select the HALT mode if it is necessary to start processing immediately upon an interrupt request.

In either of these two modes, all the contents of registers, flags and data memory just before the standby mode is set are held. The I/O port output latches and output buffer statuses are also held.

Cautions 1. The STOP mode can be used only when the system operates with 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 operation is transferred to the STOP mode, be sure to stop operation of the peripheral hardware operating with the main system clock before executing the STOP instruction.

3. The following sequence is recommended for reducing the power consumption of the A/D converter when the standby function is used: First clear bit 7 (ADCS0) of A/D converter mode register 0 (ADM0) to 0 to stop the A/D conversion operation, and then execute the HALT or STOP instruction.

21.1.2 Standby function control register

The wait time after the STOP mode is released upon an interrupt request is controlled by the oscillation stabilization time select register (OSTS).

OSTS is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets OSTS to 04H. Therefore, when the STOP mode is released by inputting $\overline{\text{RESET}}$, it takes 2^{17}/fx until release.

Figure 21-1. Format of Oscillation Stabilization Time Select Register (OSTS)

Address: FFFAH After reset: 04H R/W

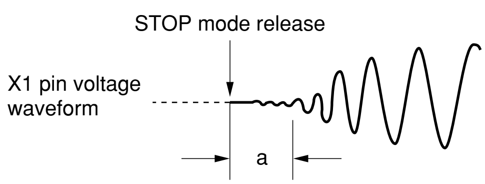
Symbol	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

★

OSTS2	OSTS1	OSTS0	Selection of oscillation stabilization time		
				fx = 8.38 MHz	fx = 12 MHz[Note]
0	0	0	2^{12}/fx	488 μs	341 μs
0	0	1	2^{14}/fx	1.95 ms	1.36 ms
0	1	0	2^{15}/fx	3.91 ms	2.73 ms
0	1	1	2^{16}/fx	7.82 ms	5.46 ms
1	0	0	2^{17}/fx	15.6 ms	10.9 ms
Other than above			Setting prohibited		

Note Expanded-specification products of μPD780024A, 780034A Subseries only.

Caution The wait time after the STOP mode is released does not include the time (see "a" in the illustration below) from STOP mode release to clock oscillation start. This applies regardless of whether STOP mode is released by $\overline{\text{RESET}}$ input or by interrupt request generation.



Remark fx: Main system clock oscillation frequency

21.2 Standby Function Operations

21.2.1 HALT mode

(1) HALT mode setting and operating statuses

The HALT mode is set by executing the HALT instruction. It can be set with the main system clock or the subsystem clock.

The operating statuses in the HALT mode are described below.

Table 21-1. HALT Mode Operating Statuses

<table>
<tr><th colspan="2" rowspan="2">HALT Mode Setting / Item</th><th colspan="2">HALT Instruction Execution When Using Main System Clock</th><th colspan="2">HALT Instruction Execution When Using Subsystem Clock</th></tr>
<tr><th>Without Subsystem Clock[Note 1]</th><th>With Subsystem Clock[Note 2]</th><th>With Main System Clock Oscillation</th><th>With Main System Clock Oscillation Stopped</th></tr>
<tr><td colspan="2">Clock generator</td><td colspan="4">Both main system clock and subsystem clock can be oscillated. Clock supply to CPU stops.</td></tr>
<tr><td colspan="2">CPU</td><td colspan="4">Operation stops.</td></tr>
<tr><td colspan="2">Ports (output latches)</td><td colspan="4">Status before HALT mode setting is held.</td></tr>
<tr><td colspan="2">16-bit timer/event counter 0</td><td colspan="3">Operable</td><td>Stop</td></tr>
<tr><td colspan="2">8-bit timer/event counters 50, 51</td><td colspan="3">Operable</td><td>Operable when TI50, TI51 are selected as count clock.</td></tr>
<tr><td colspan="2">Watch timer</td><td>Operable when $f_X/2^7$ is selected as count clock.</td><td colspan="2">Operable</td><td>Operable when f_{XT} is selected as count clock.</td></tr>
<tr><td colspan="2">Watchdog timer</td><td colspan="2">Operable</td><td colspan="2">Operation stops.</td></tr>
<tr><td colspan="2">★ Clock output</td><td>Operable when f_X to $f_X/2^7$ is selected as output clock.</td><td colspan="2">Operable</td><td>Operable when f_{XT} is selected as output clock.</td></tr>
<tr><td colspan="2">★ Buzzer output</td><td colspan="3">Operable</td><td>BUZ is at low level.</td></tr>
<tr><td colspan="2">A/D converter</td><td colspan="4">Stop</td></tr>
<tr><td colspan="2">Serial interface</td><td colspan="3">Operable</td><td>Operable during external clock input.</td></tr>
<tr><td colspan="2">External interrupt</td><td colspan="4">Operable</td></tr>
<tr><td rowspan="5">Bus line during external expansion</td><td>AD0 to AD7</td><td colspan="4">High impedance</td></tr>
<tr><td>A8 to A15</td><td colspan="4">Status before HALT mode setting is held.</td></tr>
<tr><td>ASTB</td><td colspan="4">Low level</td></tr>
<tr><td>$\overline{WR}$, $\overline{RD}$</td><td colspan="4">High level</td></tr>
<tr><td>$\overline{WAIT}$</td><td colspan="4">High impedance</td></tr>
</table>

Notes 1. Including case when external clock is not supplied.

2. Including case when external clock is supplied.

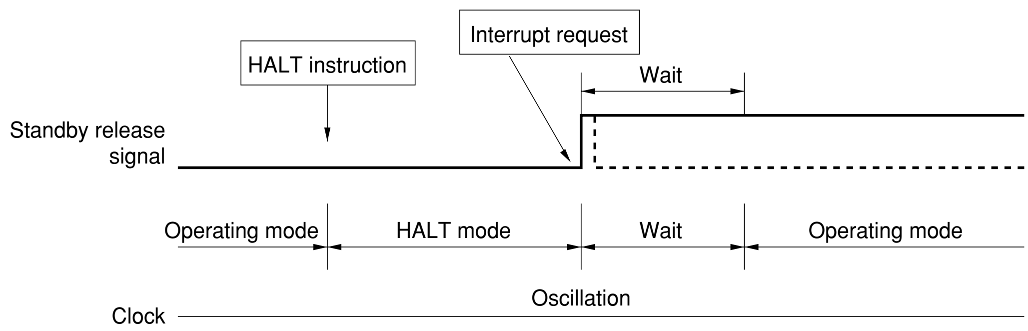
(2) HALT mode release

The HALT mode can be released with the following three types of sources.

(a) Release by unmasked interrupt request

When an unmasked interrupt request is generated, the HALT mode is released. If interrupt acknowledge is enabled, vectored interrupt service is carried out. If interrupt acknowledge is disabled, the next address instruction is executed.

Figure 21-2. HALT Mode Release by Interrupt Request Generation



Remarks 1. The broken line indicates the case when the interrupt request which has released the standby mode is acknowledged.

2. Wait times are as follows:

- When vectored interrupt service is carried out: 8 or 9 clocks
- When vectored interrupt service is not carried out: 2 or 3 clocks

(b) Release by non-maskable interrupt request

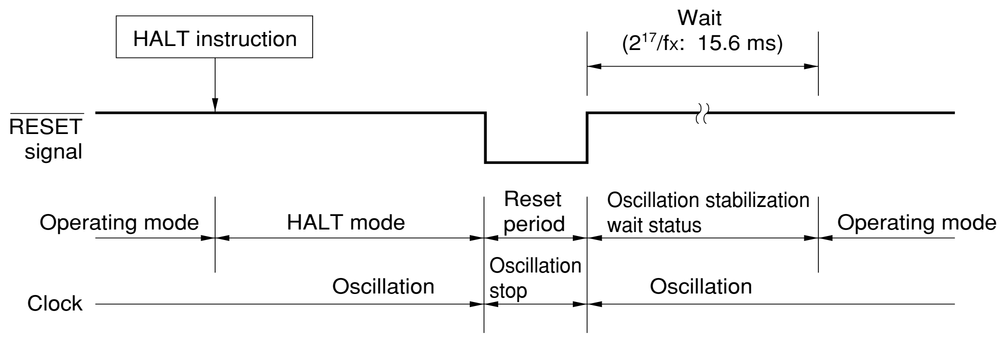
When a non-maskable interrupt request is generated, the HALT mode is released and vectored interrupt service is carried out whether interrupt acknowledge is enabled or disabled.

★ However, a non-maskable interrupt request is not generated during subsystem clock operation.

(c) Release by $\overline{\text{RESET}}$ input

When $\overline{\text{RESET}}$ signal is input, HALT mode is released. And, as in the case with normal reset operation, a program is executed after branch to the reset vector address.

Figure 21-3. HALT Mode Release by $\overline{\text{RESET}}$ Input



Remarks 1. fx: Main system clock oscillation frequency

2. Values in parentheses are for operation with fx = 8.38 MHz.

Table 21-2. Operation After HALT Mode Release

Release Source	MK××	PR××	IE	ISP	Operation
Maskable interrupt request	0	0	0	×	Next address instruction execution
	0	0	1	×	Interrupt service execution
	0	1	0	1	Next address instruction execution
	0	1	×	0	
	0	1	1	1	Interrupt service execution
	1	×	×	×	HALT mode hold
Non-maskable interrupt request	–	–	×	×	Interrupt service execution
$\overline{\text{RESET}}$ input	–	–	×	×	Reset processing

×: don't care

21.2.2 STOP mode

(1) STOP mode setting and operating statuses

The STOP mode is set by executing the STOP instruction. It can be set only with the main system clock.

Cautions 1. When the STOP mode is set, the X2 pin is internally connected to V_{DD1} 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. The operating mode is set after the wait set using the oscillation stabilization time select register (OSTS).

The operating statuses in the STOP mode are described below.

Table 21-3. STOP Mode Operating Statuses

Item (STOP Mode Setting)		With Subsystem Clock	Without Subsystem Clock
Clock generator		Only main system clock oscillation is stopped.	
CPU		Operation stops.	
Ports (output latches)		Status before STOP mode setting is held.	
16-bit timer/event counter 0		Operation stops.	
8-bit timer/event counters 50, 51		Operable only when TI50, TI51 are selected as count clock.	
Watch timer		Operable only when f_{XT} is selected as count clock.	Operation stops.
Watchdog timer		Operation stops.	
★ Clock output		Operable when f_{XT} is selected as output clock.	PCL is at low level.
Buzzer output		BUZ is at low level.	
A/D converter		Operation stops.	
Serial interface	Other than UART0	Operable only when externally supplied clock is specified as the serial clock.	
	UART0	Operation stops. (Transmit shift register 0 (TXS0), receive shift register 0 (RX0), and receive buffer register 0 (RXB0) hold the value just before the clock stopped.)	
External interrupt		Operable	
Bus line during external expansion	AD0 to AD7	High impedance	
	A8 to A15	Status before STOP mode setting is held.	
	ASTB	Low level	
	$\overline{WR}$, $\overline{RD}$	High level	
	$\overline{WAIT}$	High impedance	

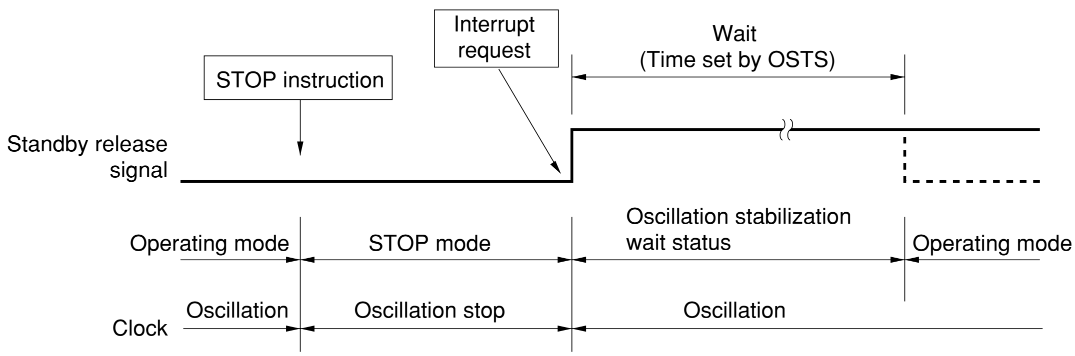
(2) STOP mode release

The STOP mode can be released by the following two types of sources.

(a) Release by unmasked interrupt request

When an unmasked interrupt request is generated, the STOP mode is released. If interrupt acknowledge is enabled after the lapse of oscillation stabilization time, vectored interrupt service is carried out. If interrupt acknowledge is disabled, the next address instruction is executed.

Figure 21-4. STOP Mode Release by Interrupt Request Generation

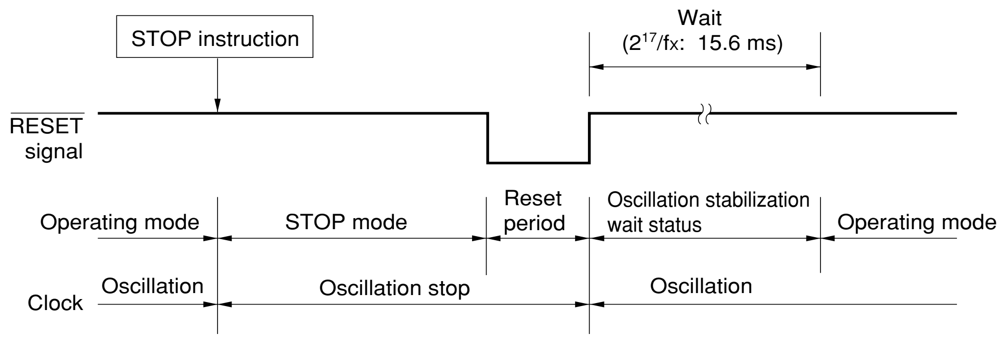


Remark The broken line indicates the case when the interrupt request which has released the standby mode is acknowledged.

(b) Release by $\overline{\text{RESET}}$ input

The STOP mode is released when $\overline{\text{RESET}}$ signal is input, and after the lapse of oscillation stabilization time, reset operation is carried out.

Figure 21-5. STOP Mode Release by $\overline{\text{RESET}}$ Input



Remarks 1. fx: Main system clock oscillation frequency

2. Values in parentheses are for operation with fx = 8.38 MHz.

Table 21-4. Operation After STOP Mode Release

Release Source	MK××	PR××	IE	ISP	Operation
Maskable interrupt request	0	0	0	×	Next address instruction execution
	0	0	1	×	Interrupt service execution
	0	1	0	1	Next address instruction execution
	0	1	×	0	
	0	1	1	1	Interrupt service execution
	1	×	×	×	STOP mode hold
$\overline{\text{RESET}}$ input	–	–	×	×	Reset processing

×: don't care

CHAPTER 22 RESET FUNCTION

The following two operations are available to generate the reset signal.

(1) External reset input via $\overline{\text{RESET}}$ pin
(2) Internal reset by watchdog timer program loop time detection

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 is set to the status shown in Table 22-1. Each pin has high impedance during reset input or during oscillation stabilization time just after reset release.

When a high level is input to the $\overline{\text{RESET}}$ pin, the reset is released and program execution starts after the lapse of oscillation stabilization time (2^{17}/fx). The reset applied by watchdog timer overflow is automatically released after a reset and program execution starts after the lapse of oscillation stabilization time (2^{17}/fx) (see **Figures 22-2** to **22-4**).

Cautions 1. For an external reset, input a low level for 10 μs or more to the $\overline{\text{RESET}}$ pin.

2. During reset input, main system clock oscillation remains stopped but subsystem clock oscillation continues.

3. When the STOP mode is released by reset, the STOP mode contents are held during reset input. However, the port pin becomes high impedance.

Figure 22-1. Reset Function Block Diagram

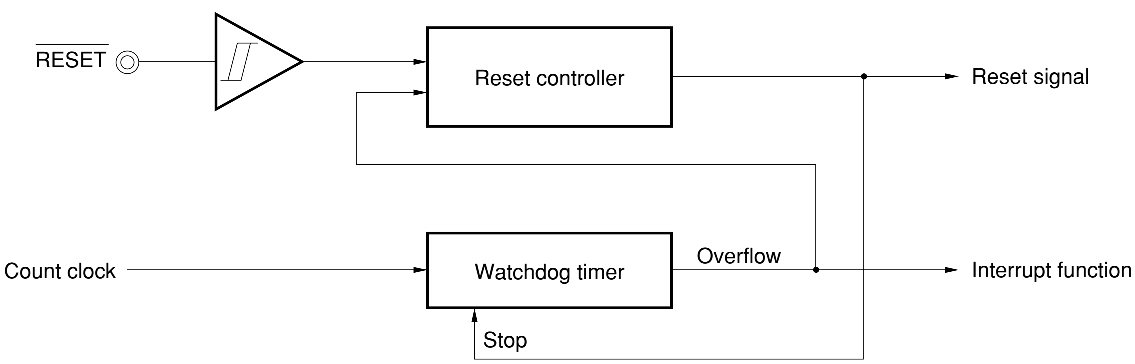


Figure 22-2. Timing of Reset by $\overline{\text{RESET}}$ Input

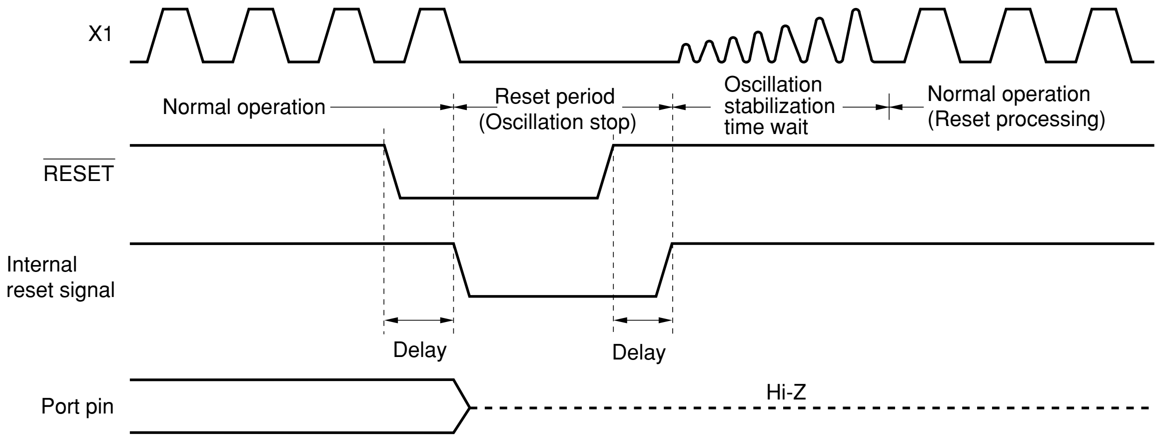


Figure 22-3. Timing of Reset Due to Watchdog Timer Overflow

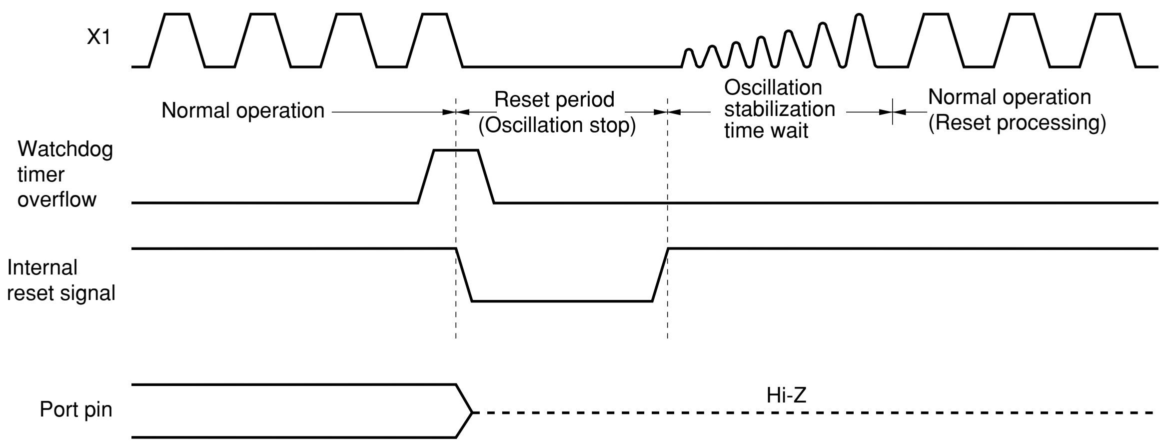


Figure 22-4. Timing of Reset in STOP Mode by $\overline{\text{RESET}}$ Input

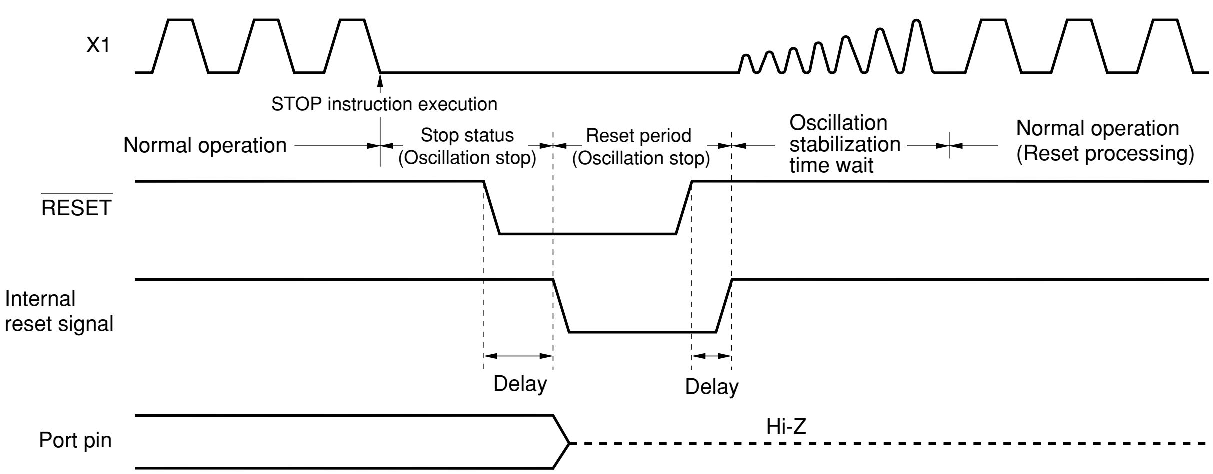


Table 22-1. Hardware Statuses After Reset (1/2)

Hardware		Status After Reset
Program counter (PC)[Note 1]		Contents of reset vector table (0000H, 0001H) are set.
Stack pointer (SP)		Undefined
Program status word (PSW)		02H
RAM	Data memory	Undefined[Note 2]
	General-purpose register	Undefined[Note 2]
Port (output latch)		00H
Port mode registers 0, 2 to 7 (PM0, PM2 to PM7)		FFH
Pull-up resistor option registers 0, 2 to 7 (PU0, PU2 to PU7)		00H
Processor clock control register (PCC)		04H
Memory size switching register (IMS)		CFH[Note 3]
Memory expansion mode register (MEM)		00H
Memory expansion wait setting register (MM)		10H
Oscillation stabilization time select register (OSTS)		04H
16-bit timer/event counter 0	Timer counter 0 (TM0)	0000H
	Capture/compare registers 00, 01 (CR00, CR01)	Undefined
	Prescaler mode register 0 (PRM0)	00H
	Capture/compare control register 0 (CRC0)	00H
	Mode control register 0 (TMC0)	00H
	Output control register 0 (TOC0)	00H
8-bit timer/event counters 50, 51	Timer counters 50, 51 (TM50, TM51)	00H
	Compare registers 50, 51 (CR50, CR51)	Undefined
	Clock select registers 50, 51 (TCL50, TCL51)	00H
	Mode control registers 50, 51 (TMC50, TMC51)	00H
Watch timer	Operation mode register (WTM)	00H
Watchdog timer	Clock select register (WDCS)	00H
	Mode register (WDTM)	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. When a reset is executed in the standby mode, the pre-reset status is held even after reset.

3. Although the initial value is CFH, use the following value to be set for each version.

μPD780021A, 780021AY, 780031A, 780031AY: 42H
μPD780022A, 780022AY, 780032A, 780032AY: 44H
μPD780023A, 780023AY, 780033A, 780033AY: C6H
μPD780024A, 780024AY, 780034A, 780034AY: C8H
μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY: Value for mask ROM versions

Table 22-1. Hardware Statuses After Reset (2/2)

Hardware		Status After Reset
Clock output/buzzer output controller	Clock output select register (CKS)	00H
A/D converter	Conversion result register 0 (ADCR0)	00H
	Mode register 0 (ADM0)	00H
	Analog input channel specification register 0 (ADS0)	00H
Serial interface UART0	Asynchronous serial interface mode register 0 (ASIM0)	00H
	Asynchronous serial interface status register 0 (ASIS0)	00H
	Baud rate generator control register 0 (BRGC0)	00H
	Transmit shift register 0 (TXS0)	FFH
	Receive buffer register 0 (RXB0)	
Serial interfaces SIO30, SIO31 Note 1	Shift registers 30, 31 (SIO30, SIO31)	Undefined
	Operation mode registers 30, 31 (CSIM30, CSIM31)	00H
Serial interface IIC0 Note 2	Transfer clock select register 0 (IICCL0)	00H
	Shift register 0 (IIC0)	00H
	Control register 0 (IICC0)	00H
	Status register 0 (IICS0)	00H
	Slave address register 0 (SVA0)	00H
Interrupt	Request flag registers 0L, 0H, 1L (IF0L, IF0H, IF1L)	00H
	Mask flag registers 0L, 0H, 1L (MK0L, MK0H, MK1L)	FFH
	Priority specification flag registers 0L, 0H, 1L (PR0L, PR0H, PR1L)	FFH
	External interrupt rising edge enable register (EGP)	00H
	External interrupt falling edge enable register (EGN)	00H

Notes 1. Serial interface SIO31 is provided only in the μPD780024A, 780034A Subseries.

2. Serial interface IIC0 is provided only in the μPD780024AY, 780034AY Subseries.

CHAPTER 23 μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY

The μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY are provided as the flash memory versions of the μPD780024A, 780034A, 780024AY, 780034AY Subseries.

The μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY are products that incorporate flash memory in which the program can be written, erased, and rewritten while it is mounted on the board.

Writing to flash memory can be performed with the memory mounted on the target system (on board). A dedicated flash programmer is connected to the target system to perform writing.

The following can be considered as the development environment and the applications using flash memory.

- Software can be altered after the μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY are solder-mounted on the target system.
- Small scale production of various models is made easier by differentiating software.
- Data adjustment in starting mass production is made easier.

Table 23-1 shows the correspondence between μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY and the mask ROM versions.

Table 23-1. Correspondence Between μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY, and Mask ROM Versions

Mask ROM Version / Flash Memory Version	μPD780021A/22A/23A/24A/31A/32A/33A/34A		μPD780021A(A)/22A(A)/23A(A)/24A(A)/31A(A)/32A(A)/33A(A)/34A(A)		μPD780021AY/22AY/23AY/24AY/31AY/32AY/33AY/34AY	μPD780021AY(A)/22AY(A)/23AY(A)/24AY(A)/31AY(A)/32AY(A)/33AY(A)/34AY(A)
	Conventional Products	Expanded-Specification Products	Conventional Products	Expanded-Specification Products		
μPD78F0034A	√	–	–	–	–	–
μPD78F0034B	–	√	–	–	–	–
μPD78F0034B(A)	–	–	√Note	√	–	–
μPD78F0034AY	–	–	–	–	√	–
μPD78F0034BY	–	–	–	–	√	–
μPD78F0034BY(A)	–	–	–	–	–	√

Note The μPD78F0034B(A) and the conventional products of the μPD780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), and 780034A(A) differ in the operating frequency. When replacing a flash memory version with a mask ROM version, note the supply voltage and operating frequency.

Remarks 1. √: Supported, –: Not supported

2. Expanded-specification products and conventional products of the μPD780024A and 780034A Subseries differ in operating frequency ratings. For details, refer to the description of electrical specifications.

3. Expanded-specification products of the μPD780024AY, 780034AY Subseries are not available. Only conventional products are available.

4. A special grade product of the μPD78F0034A, 78F0034AY is not available. Only a standard grade product is available.

23.1 Differences Between μPD78F0034A, 78F0034AY and μPD78F0034B, 78F0034BY

Table 23-2 shows the differences between the μPD78F0034A and μPD78F0034B, and Table 23-3 shows differences between the μPD78F0034AY and μPD78F0034BY.

Table 23-2. Differences Between μPD78F0034A and μPD78F0034B

Item		μPD78F0034A	μPD78F0034B
Guaranteed operating speed (operating frequency)	4.5 to 5.5 V	8.38 MHz (0.238 μs)	12 MHz (0.166 μs)
	4.0 to 5.5 V	8.38 MHz (0.238 μs)	8.38 MHz (0.238 μs)
	3.0 to 5.5 V	5 MHz (0.4 μs)	8.38 MHz (0.238 μs)
	2.7 to 5.5 V	5 MHz (0.4 μs)	5 MHz (0.4 μs)
	1.8 to 5.5 V	1.25 MHz (1.6 μs)	1.25 MHz (1.6 μs)
Minimum instruction execution time		Minimum instruction execution time variable function incorporated	
	When main system clock is selected	0.238 μs/0.477 μs/0.954 μs/1.90 μs/3.81 μs (@ 8.38 MHz operation, V_{DD} = 4.0 to 5.5 V)	0.166 μs/0.333 μs/0.666 μs/1.33 μs/2.66 μs (@ 12 MHz operation, V_{DD} = 4.5 to 5.5 V)
	When subsystem clock is selected	122 μs (32.768 kHz)	
Clock output		• 65.5 kHz, 131 kHz, 262 kHz, 524 kHz, 1.05 MHz, 2.10 MHz, 4.19 MHz, 8.38 MHz (@ 8.38 MHz operation with main system clock) • 32.768 kHz (@ 32.768 kHz operation with subsystem clock)	• 93.7 kHz, 187 kHz, 375 kHz, 750 kHz, 1.5 MHz, 3 MHz, 6 MHz, 12 MHz (@ 12 MHz operation with main system clock) • 32.768 kHz (@ 32.768 kHz operation with subsystem clock)
Buzzer output		1.02 kHz, 2.05 kHz, 4.10 kHz, 8.19 kHz (@ 8.38 MHz operation with main system clock)	1.46 kHz, 2.92 kHz, 5.85 kHz, 11.7 kHz (@ 12 MHz operation with main system clock)
Communication mode of flash memory programming		• 3-wire serial I/O: 2 channels Note • UART: 1 channel • Pseudo 3-wire serial I/O: 1 channel	• 3-wire serial I/O: 2 channels Note • UART: 1 channel • Pseudo 3-wire serial I/O: 1 channel
Electrical specifications, recommended soldering conditions		Refer to the description of electrical specifications and recommended soldering conditions.	

Note The μPD78F0034A cannot use a handshake mode.
The μPD78F0034B can use one channel (serial interface SIO30) as a handshake mode.

Remark The operating frequency ratings of the μPD78F0034A and the conventional products of the mask ROM versions of the μPD780024A, 780034A Subseries are the same. The operating frequency ratings of the μPD78F0034B and the expanded-specification products of the mask ROM versions of the μPD780024A, 780034A Subseries are the same.

Table 23-3. Differences Between μPD78F0034AY and μPD78F0034BY

Item		μPD78F0034AY	μPD78F0034BY
Guaranteed operating speed (operating frequency)	4.5 to 5.5 V	8.38 MHz (0.238 μs)	
	4.0 to 5.5 V	8.38 MHz (0.238 μs)	
	3.0 to 5.5 V	5 MHz (0.4 μs)	
	2.7 to 5.5 V	5 MHz (0.4 μs)	
	1.8 to 5.5 V	1.25 MHz (1.6 μs)	
Minimum instruction execution time		Minimum instruction execution time variable function incorporated	
	When main system clock is selected	0.238 μs/0.477 μs/0.954 μs/1.90 μs/3.81 μs (@ 8.38 MHz operation, V_{DD} = 4.0 to 5.5 V)	
	When subsystem clock is selected	122 μs (32.768 kHz)	
Clock output		• 65.5 kHz, 131 kHz, 262 kHz, 524 kHz, 1.05 MHz, 2.10 MHz, 4.19 MHz, 8.38 MHz (@ 8.38 MHz operation with main system clock) • 32.768 kHz (@ 32.768 kHz operation with subsystem clock)	
Buzzer output		1.02 kHz, 2.05 kHz, 4.10 kHz, 8.19 kHz (@ 8.38 MHz operation with main system clock)	
Communication mode of flash memory programming		• 3-wire serial I/O: 2 channels[Note] • UART: 1 channel • Pseudo 3-wire serial I/O: 1 channel	• 3-wire serial I/O: 2 channels[Note] • UART: 1 channel • Pseudo 3-wire serial I/O: 1 channel
Electrical specifications, recommended soldering conditions		Refer to the description of electrical specifications and recommended soldering conditions.	

Note The μPD78F0034AY cannot use a handshake mode.
The μPD78F0034BY can use one channel (serial interface SIO30) as a handshake mode.

Remark The operating frequency ratings of the μPD78F0034AY, 78F0034BY and the mask ROM versions of the μPD780024AY, 780034AY Subseries are the same.

23.2 Differences Between μPD78F0034B, 78F0034BY and μPD78F0034B(A), 78F0034BY(A)

The μPD78F0034B(A) and 78F0034BY(A) are products to which a quality assurance program more stringent than that used for the μPD78F0034B and 78F0034BY (standard products) is applied (NEC Electronics classifies these products as “special” quality grade products).

The μPD78F0034B, 78F0034BY and μPD78F0034B(A), 78F0034BY(A) only differ in the quality grade; there are no differences in functions and electrical specifications.

Table 23-4. Differences Between μPD78F0034B, 78F0034BY and μPD78F0034B(A), 78F0034BY(A)

Item	μPD78F0034B, 78F0034BY	μPD78F0034B(A), 78F0034BY(A)
Quality grade	Standard	Special
Functions and electrical specifications	No differences.	

This chapter explains the μPD78F0034B as the representative product of the μPD78F0034B and 78F0034B(A), and the μPD78F0034BY as the representative product of the μPD78F0034BY and 78F0034BY(A).

23.3 Differences Between µPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY and Mask ROM Versions

Tables 23-5 and 23-6 show the differences between the µPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY and the mask ROM versions.

Table 23-5. Differences Between µPD78F0034A, 78F0034B and Mask ROM Versions

Item	µPD78F0034A, 78F0034B	Mask ROM Versions	
		µPD780034A Subseries	µPD780024A Subseries[Note]
Internal ROM structure	Flash memory	Mask ROM	
Internal ROM capacity	32 KB	µPD780031A: 8 KB µPD780032A: 16 KB µPD780033A: 24 KB µPD780034A: 32 KB	µPD780021A: 8 KB µPD780022A: 16 KB µPD780023A: 24 KB µPD780024A: 32 KB
Internal high-speed RAM capacity	1,024 bytes	µPD780031A: 512 bytes µPD780032A: 512 bytes µPD780033A: 1,024 bytes µPD780034A: 1,024 bytes	µPD780021A: 512 bytes µPD780022A: 512 bytes µPD780023A: 1,024 bytes µPD780024A: 1,024 bytes
Minimum instruction execution time	Minimum instruction execution time variable function incorporated		
When main system clock is selected	• 0.166 µs/0.333 µs/0.666 µs/1.33 µs/2.66 µs (@ 12 MHz operation, µPD78F0034B and expanded-specification products of the mask ROM versions only) • 0.238 µs/0.477 µs/0.954 µs/1.90 µs/3.81 µs (@ 8.38 MHz operation)		
When subsystem clock is selected	122 µs (@32.768 kHz operation)		
Clock output	• 93.7 kHz, 187 kHz, 375 kHz, 750 kHz, 1.5 MHz, 3 MHz, 6 MHz, 12 MHz (@ 12 MHz operation with main system clock, µPD78F0034B and expanded-specification products of the mask ROM versions only) • 65.5 kHz, 131 kHz, 262 kHz, 524 kHz, 1.05 MHz, 2.10 MHz, 4.19 MHz, 8.38 MHz (@ 8.38 MHz operation with main system clock) • 32.768 kHz (@ 32.768 kHz operation with subsystem clock)		
Buzzer output	• 1.46 kHz, 2.92 kHz, 5.85 kHz, 11.7 kHz (@ 12 MHz operation with main system clock, µPD78F0034B and expanded-specification products of the mask ROM versions only) • 1.02 kHz, 2.05 kHz, 4.10 kHz, 8.19 kHz (@ 8.38 MHz operation with main system clock)		
A/D converter resolution	10 bits		8 bits
Mask option specification of on-chip pull-up resistor for pins P30 to P33	Not available	Available	
IC pin	Not provided	Provided	
V_{PP} pin	Provided	Not provided	
Electrical specifications, recommended soldering conditions	Refer to the description of electrical specifications and recommended soldering conditions.		

Note The µPD78F0034A and 78F0034B can be used as the flash memory version of the µPD780024A Subseries.

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 on the commercial samples (CS) (not engineering samples (ES)) of the mask ROM versions.

Table 23-6. Differences Between μPD78F0034AY, 78F0034BY and Mask ROM Versions

Item	μPD78F0034AY, 78F0034BY	Mask ROM Versions	
		μPD780034AY Subseries	μPD780024AY Subseries[Note]
Internal ROM structure	Flash memory	Mask ROM	
Internal ROM capacity	32 KB	μPD780031AY: 8 KB μPD780032AY: 16 KB μPD780033AY: 24 KB μPD780034AY: 32 KB	μPD780021AY: 8 KB μPD780022AY: 16 KB μPD780023AY: 24 KB μPD780024AY: 32 KB
Internal high-speed RAM capacity	1,024 bytes	μPD780031AY: 512 bytes μPD780032AY: 512 bytes μPD780033AY: 1,024 bytes μPD780034AY: 1,024 bytes	μPD780021AY: 512 bytes μPD780022AY: 512 bytes μPD780023AY: 1,024 bytes μPD780024AY: 1,024 bytes
Minimum instruction execution time	Minimum instruction execution time variable function incorporated		
When main system clock is selected	0.238 μs/0.477 μs/0.954 μs/1.90 μs/3.81 μs (@8.38 MHz operation)		
When subsystem clock is selected	122 μs (@32.768 kHz operation)		
Clock output	• 65.5 kHz, 131 kHz, 262 kHz, 524 kHz, 1.05 MHz, 2.10 MHz, 4.19 MHz, 8.38 MHz (@ 8.38 MHz operation with main system clock) • 32.768 kHz (@ 32.768 kHz operation with subsystem clock)		
Buzzer output	1.02 kHz, 2.05 kHz, 4.10 kHz, 8.19 kHz (@ 8.38 MHz operation with main system clock)		
A/D converter resolution	10 bits		8 bits
Mask option specification of on-chip pull-up resistor for pins P30 and P31	Not available	Available	
IC pin	Not provided	Provided	
V_{PP} pin	Provided	Not provided	
Electrical specifications, recommended soldering conditions	Refer to the description of electrical specifications and recommended soldering conditions.		

Note The μPD78F0034AY and 78F0034BY can be used as the flash memory version of the μPD780024AY Subseries.

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 on the commercial samples (CS) (not engineering samples (ES)) of the mask ROM versions.

23.4 Memory Size Switching Register

The μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY allow users to select the internal memory capacity using the memory size switching register (IMS) so that the same memory map as that of the mask ROM versions with a different size of internal memory capacity can be achieved.

IMS is set by an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets IMS to CFH.

Caution Be sure to set the value of the target mask ROM version to IMS as an initialization setting of the program. IMS is set to CFH by reset, so be sure to set the value of the target mask ROM version after reset.

Figure 23-1. Format of Memory Size Switching Register (IMS)

Address: FFF0H After reset: CFH R/W

Symbol	7	6	5	4	3	2	1	0
IMS	RAM2	RAM1	RAM0	0	ROM3	ROM2	ROM1	ROM0

RAM2	RAM1	RAM0	Internal high-speed RAM capacity selection
0	1	0	512 bytes
1	1	0	1024 bytes
Other than above			Setting prohibited

ROM3	ROM2	ROM1	ROM0	Internal ROM capacity selection
0	0	1	0	8 KB
0	1	0	0	16 KB
0	1	1	0	24 KB
1	0	0	0	32 KB
1	1	1	1	60 KB (setting prohibited)
Other than above				Setting prohibited

The IMS settings to obtain the same memory map as mask ROM versions are shown in Table 23-7.

Table 23-7. Memory Size Switching Register Settings

Target Mask ROM Versions	IMS Setting
μPD780021A, 780031A, 780021AY, 780031AY	42H
μPD780022A, 780032A, 780022AY, 780032AY	44H
μPD780023A, 780033A, 780023AY, 780033AY	C6H
μPD780024A, 780034A, 780024AY, 780034AY	C8H

Caution When using the mask ROM versions, be sure to set the value indicated in Table 23-7 to IMS.

23.5 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 made by Naito Densei 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

23.5.1 Programming environment

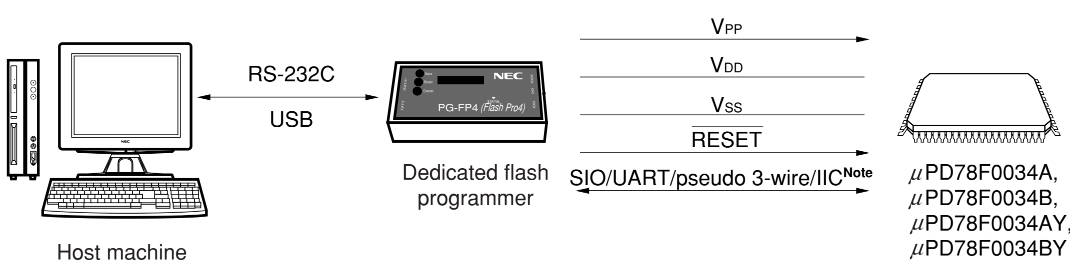
The following shows the environment required for μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY flash memory programming.

When Flashpro III or Flashpro IV 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 of Flashpro III/Flashpro IV.

Remark USB is supported by Flashpro IV only.

Figure 23-2. Environment for Writing Program to Flash Memory



Note IIC is supported by the μPD78F0034AY, 78F0034BY only.

23.5.2 Communication mode

Use the communication mode shown in Table 23-8 to perform communication between the dedicated flash programmer and the μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY.

Table 23-8. Communication Mode List (1/2)

(1) μPD78F0034A, 78F0034B

Communication Mode	Standard (TYPE) Setting[Note 1]					Pins Used	Number of V_{PP} Pulses
	Port (COMM PORT)	Speed (SIO CLOCK)	On Target (CPU CLOCK)	Frequency (Flashpro Clock)	Multiply Rate (Multiple Rate)		
3-wire serial I/O (SIO30)	SIO-ch0 (SIO ch-0)	2.4 kHz to 625 kHz[Note 2] (100 Hz to 1.25 MHz)[Note 2]	Optional	1 to 10 MHz[Note 2]	1.0	SI30/P20 SO30/P21 $\overline{SCK30}$/P22	0
3-wire serial I/O (SIO31)	SIO-ch1 (SIO ch-1)					SI31/P34 SO31/P35 $\overline{SCK31}$/P36	1
3-wire serial I/O (SIO30) with handshake Note 3	SIO-H/S (SIO ch-3 + handshake)					SI30/P20 SO30/P21 $\overline{SCK30}$/P22 HS/P25	3
UART (UART0)	UART-ch0 (UART ch-0)	4800 to 76800 Baud[Notes 2, 4] (4800 to 76800 bps)[Notes 2, 4]				RxD0/P23 TxD0/P24	8
Pseudo 3-wire serial I/O	Port-ch0 (Port A)	100 Hz to 1500 Hz[Note 2] (100 Hz to 1.25 MHz)[Note 2]				P70/TI00/TO0 (serial data input) P71/TI01 (serial data output) P72/TI50/TO50 (serial clock input)	12

Notes 1. Selection items for Standard settings on Flashpro IV (TYPE settings on Flashpro III).

2. The possible setting range differs depending on the voltage. For details, refer to the description of electrical specifications.

3. μPD78F0034B only

4. 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.

Remark Items enclosed in parentheses in the setting item column are the set value and set item of Flashpro III when they differ from those of Flashpro IV.

Table 23-8. Communication Mode List (2/2)

(2) μPD78F0034AY, 78F0034BY

Communication Mode	Standard (TYPE) Setting Note 1					Pins Used	Number of V_{PP} Pulses
	Port (COMM PORT)	Speed (SIO CLOCK)	On Target (CPU CLOCK)	Frequency (Flashpro Clock)	Multiply Rate (Multiple Rate)		
3-wire serial I/O (SIO30)	SIO-ch0 (SIO ch-0)	2.4 kHz to 625 kHz Note 2 (100 Hz to 1.25 MHz) Note 2	Optional	1 to 10 MHz Note 2	1.0	SI30/P20 SO30/P21 $\overline{SCK30}$/P22	0
3-wire serial I/O (SIO30) with handshake Note 3	SIO-H/S (SIO ch-3 + handshake)					SI30/P20 SO30/P21 $\overline{SCK30}$/P22 HS/P25	3
I^2C bus (IIC0)	IIC-ch0 (I^2C ch-0)	10 k to 100 k Band Note 2 (50 kHz) Note 2				SDA0/P32 SCL0/P33	4
UART (UART0)	UART-ch0 (UART ch-0)	4800 to 76800 Baud Notes 2, 4 (4800 to 76800 bps) Notes 2, 4				RxD0/P23 TxD0/P24	8
Pseudo 3-wire serial I/O	Port-ch0 (Port A)	100 Hz to 1500 Hz Note 2 (100 Hz to 1.25 MHz) Note 2				P70/TI00/TO0 (serial data input) P71/TI01 (serial data output) P72/TI50/TO50 (serial clock input)	12

Notes 1. Selection items for Standard settings on Flashpro IV (TYPE settings on Flashpro III).

2. The possible setting range differs depending on the voltage. For details, refer to the description of electrical specifications.

3. μPD78F0034BY only

4. 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.

Remark Items enclosed in parentheses in the setting item column are the set value and set item of Flashpro III when they differ from those of Flashpro IV.

Figure 23-3. Communication Mode Selection Format

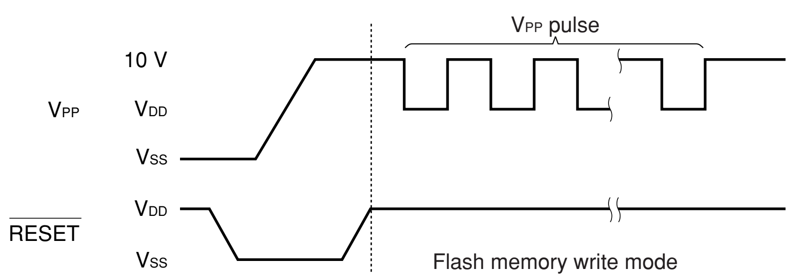
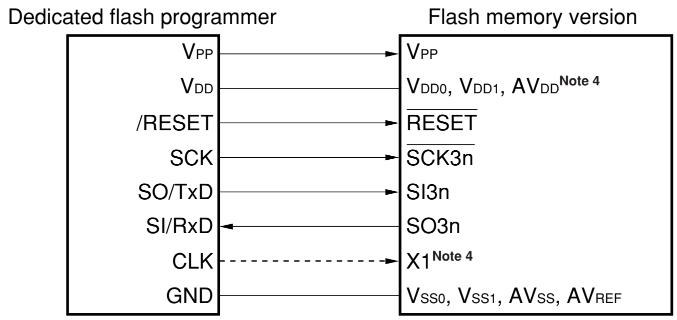
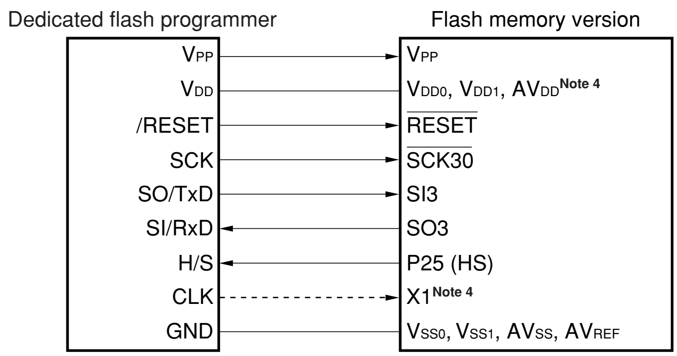


Figure 23-4. Example of Connection with Dedicated Flash Programmer (1/2)

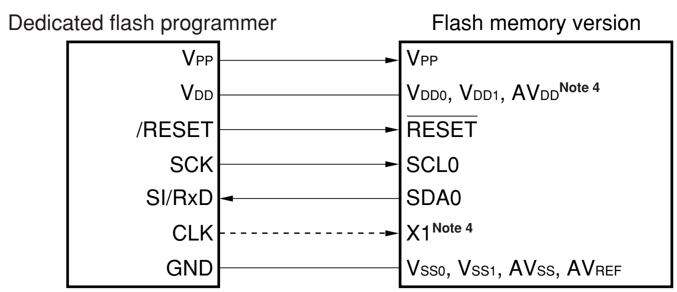
(a) 3-wire serial I/O (SIO3n[Note 1])



(b) 3-wire serial I/O (SIO30) with handshake[Note 2]



(c) I^{2}C bus (IIC0)[Note 3]



Notes 1. n = 0, 1: μPD78F0034A, 78F0034B
n = 0: μPD78F0034AY, 78F0034BY

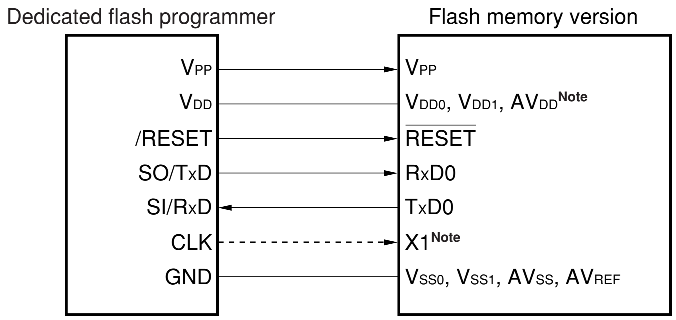
2. μPD78F0034B, 78F0034BY only

3. μPD78F0034AY, 78F0034BY only

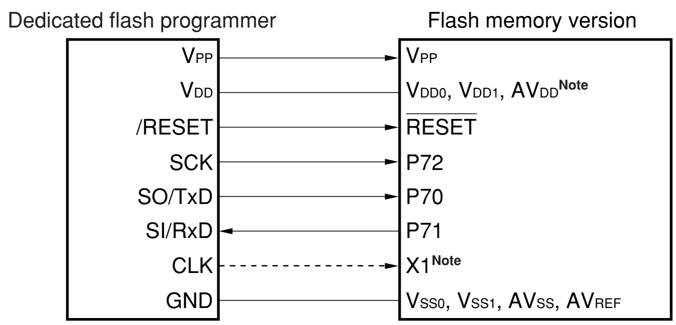
4. The V_{DD0}, V_{DD1}, AV_{DD}, and X1 pins can be supplied on board. In this case, these pins do not need to be connected to the dedicated flash programmer, but V_{DD} voltage must be supplied to these pins (except the X1 pin) before programming is started.

Figure 23-4. Example of Connection with Dedicated Flash Programmer (2/2)

(d) UART (UART0)



(e) Pseudo 3-wire serial I/O



Note The V_{DD0}, V_{DD1}, AV_{DD}, and X1 pins can be supplied on board. In this case, these pins do not need to be connected to the dedicated flash programmer, but V_{DD} voltage must be supplied to these pins (except the X1 pin) before programming is started.

If Flashpro III/Flashpro IV is used as the dedicated flash programmer, the following signals are generated for the μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY. For details, refer to the manual of Flashpro III/Flashpro IV.

Table 23-9. Pin Connection List

Signal Name	I/O	Pin Function	Pin Name	SIO30	SIO31 Note 1	SIO30 (HS) Note 2	UART0	IIC0 Note 3	Pseudo 3-wire
V_{PP}	Output	Write voltage	V_{PP}	◎	◎	◎	◎	◎	◎
V_{DD}	I/O	V_{DD} voltage generation/voltage monitoring	V_{DD0}, V_{DD1}, AV_{DD}	Note 4 ◎	Note 4 ◎	Note 4 ◎	Note 4 ◎	Note 4 ◎	Note 4 ◎
GND	–	Ground	V_{SS0}, V_{SS1}, AV_{SS}, AV_{REF}	◎	◎	◎	◎	◎	◎
CLK	Output	Clock output	X1	○	○	○	○	○	○
/RESET	Output	Reset signal	$\overline{\text{RESET}}$	◎	◎	◎	◎	◎	◎
SI/RxD	Input	Reception signal	SO30/SO31 Note 1 TxD0/SDA0 Note 3/P71	◎	◎	◎	◎	◎	◎
SO/TxD	Output	Transmission signal	SI30/SI31 Note 1/RxD0/P70	◎	◎	◎	◎	×	◎
SCK	Output	Transfer clock	$\overline{\text{SCK30}}$/$\overline{\text{SCK31}}$ Note 1/SCL0 Note 3/P72	◎	◎	◎	×	◎	◎
H/S	Input	Handshake signal	P25 (HS) Note 2	×	×	◎	×	×	×

Notes 1. μPD78F0034A, 78F0034B only

2. μPD78F0034B, 78F0034BY only

3. μPD78F0034AY, 78F0034BY only

4. V_{DD} 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.

23.5.3 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.

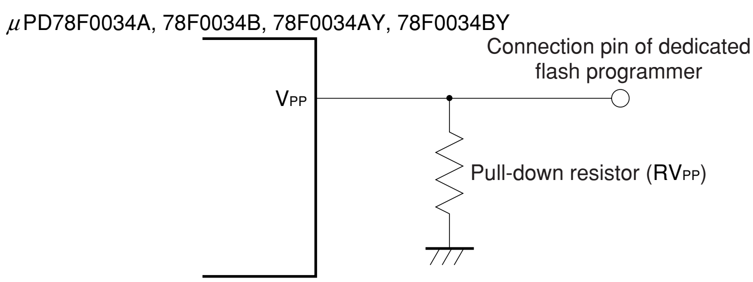
<V_{PP} pin>

In normal operation mode, input 0 V to the V_{PP} pin. In flash memory programming mode, a write voltage of 10.0 V (TYP.) is supplied to the V_{PP} pin, so perform the following.

(1) Connect a pull-down resistor (RV_{PP} = 10 kΩ) to the V_{PP} pin.
(2) Use the jumper on the board to switch the V_{PP} pin input to either the programmer or directly to GND.

A V_{PP} pin connection example is shown below.

Figure 23-5. V_{PP} Pin Connection Example



<Serial interface pin>

The following shows the pins used by the serial interface.

Serial Interface	Pins Used
3-wire serial I/O (SIO30)	SI30/P20, SO30/P21, $\overline{SCK30}$/P22
3-wire serial I/O (SIO31)**Note 1**	SI31/P34, SO31/P35, $\overline{SCK31}$/P36
3-wire serial I/O (SIO30) with handshake**Note 2**	SI30/P20, SO30/P21, $\overline{SCK30}$/P22, HS/P25
I^2C bus (IIC0)**Note 3**	SDA0/P32, SCL0/P33
UART (UART0)	RxD0/P23, TxD0/P24
Pseudo 3-wire serial I/O	P70/TI00/TO0 (serial data input), P71/TI01 (serial data output), P72/TI50/TO50 (serial clock input)

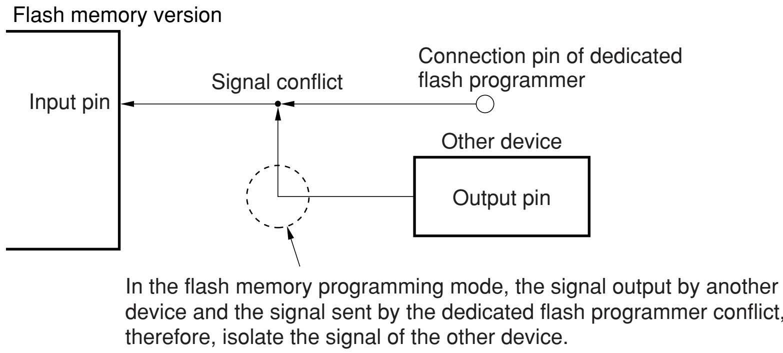
Notes 1. μPD78F0034A, 78F0034B only
2. μPD78F0034B, 78F0034BY only
3. μPD78F0034AY, 78F0034BY only

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.

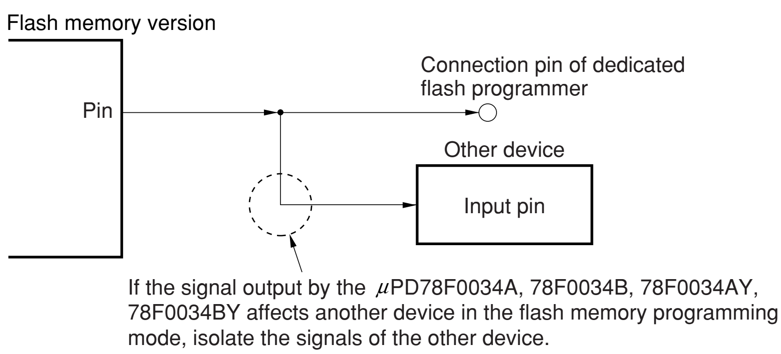
Figure 23-6. Signal Conflict (Input Pin of Serial Interface)

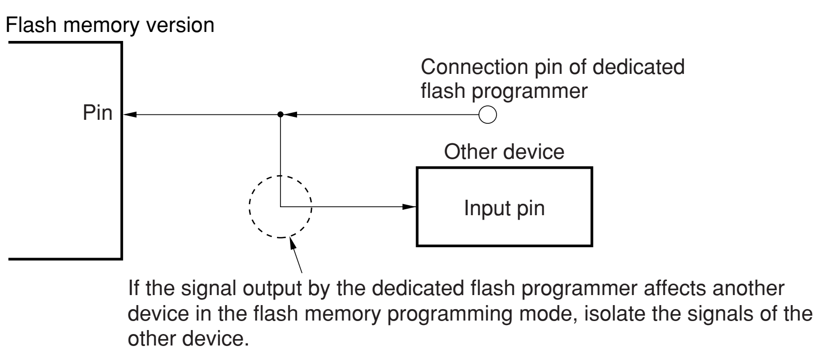


(2) Abnormal operation of other device

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, which 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 23-7. Abnormal Operation of Other Device



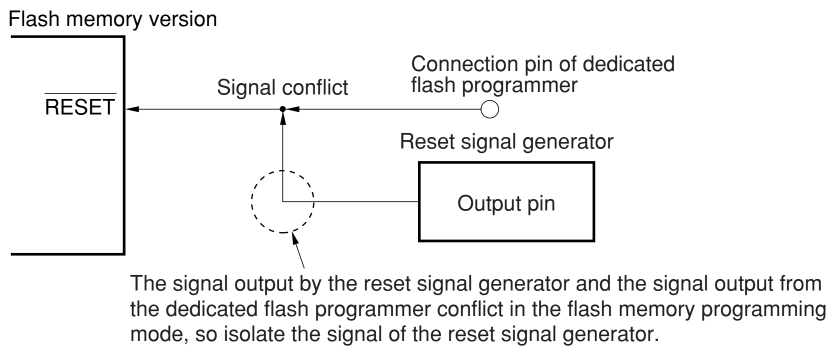


<$\overline{\text{RESET}}$ pin>

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 23-8. Signal Conflict ($\overline{\text{RESET}}$ Pin)



<Port pins>

When the μPD78F0034A, 78F0034B, 78F0034AY, and 78F0034BY 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 V_{DD} 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 V_{DD} of the flash programmer.

Supply the same power as in the normal operation mode to the other power supply pins (AV_{DD}, AV_{REF}, and AV_{SS}).

23.5.4 Connection on adapter for flash memory writing

Examples of the recommended connection when using the adapter for flash memory writing are shown below.

Figure 23-9. Example of Wiring Adapter for Flash Memory Writing in 3-Wire Serial I/O (SIO30) Mode (1/2)

(1) 64-pin plastic SDIP (19.05 mm (750))

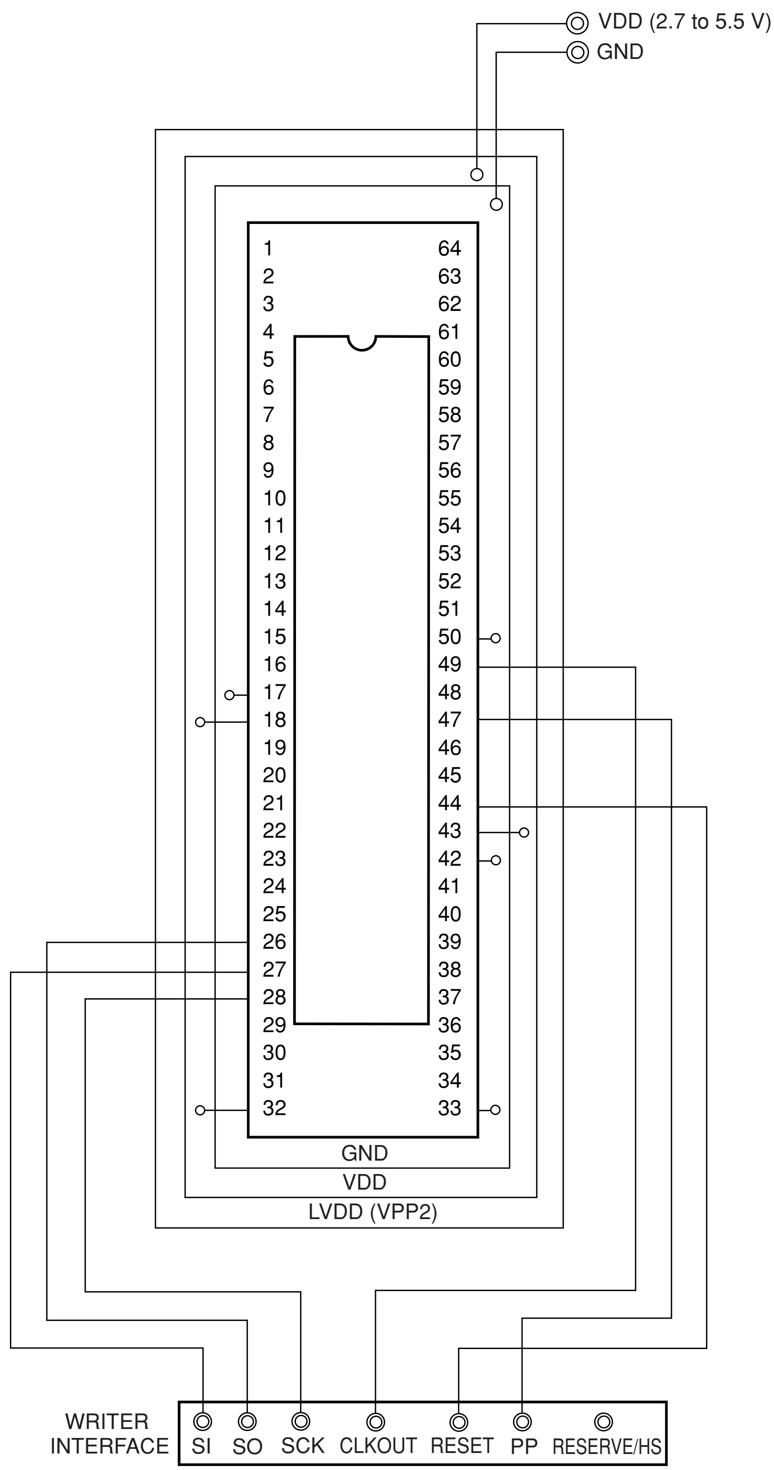


Figure 23-9. Example of Wiring Adapter for Flash Memory Writing in 3-Wire Serial I/O (SIO30) Mode (2/2)

(2) 64-pin plastic QFP (14 × 14), 64-pin plastic LQFP (14 × 14), 64-pin plastic TQFP (12 × 12), 64-pin plastic LQFP (10 × 10)

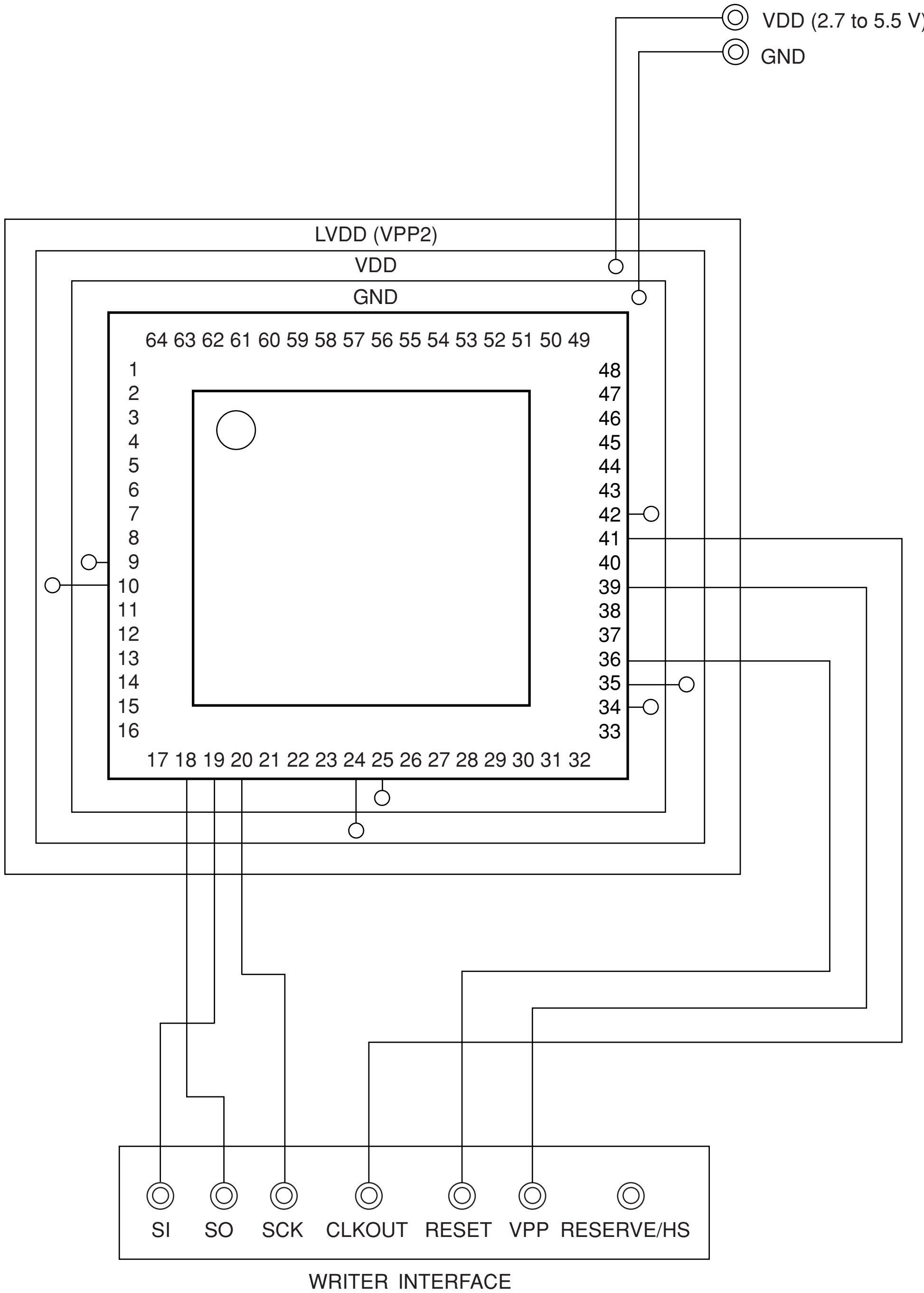


Figure 23-10. Example of Wiring Adapter for Flash Memory Writing in 3-Wire Serial I/O (SIO31) Mode (μPD78F0034A, 78F0034B only) (1/2)

(1) 64-pin plastic SDIP (19.05 mm (750))

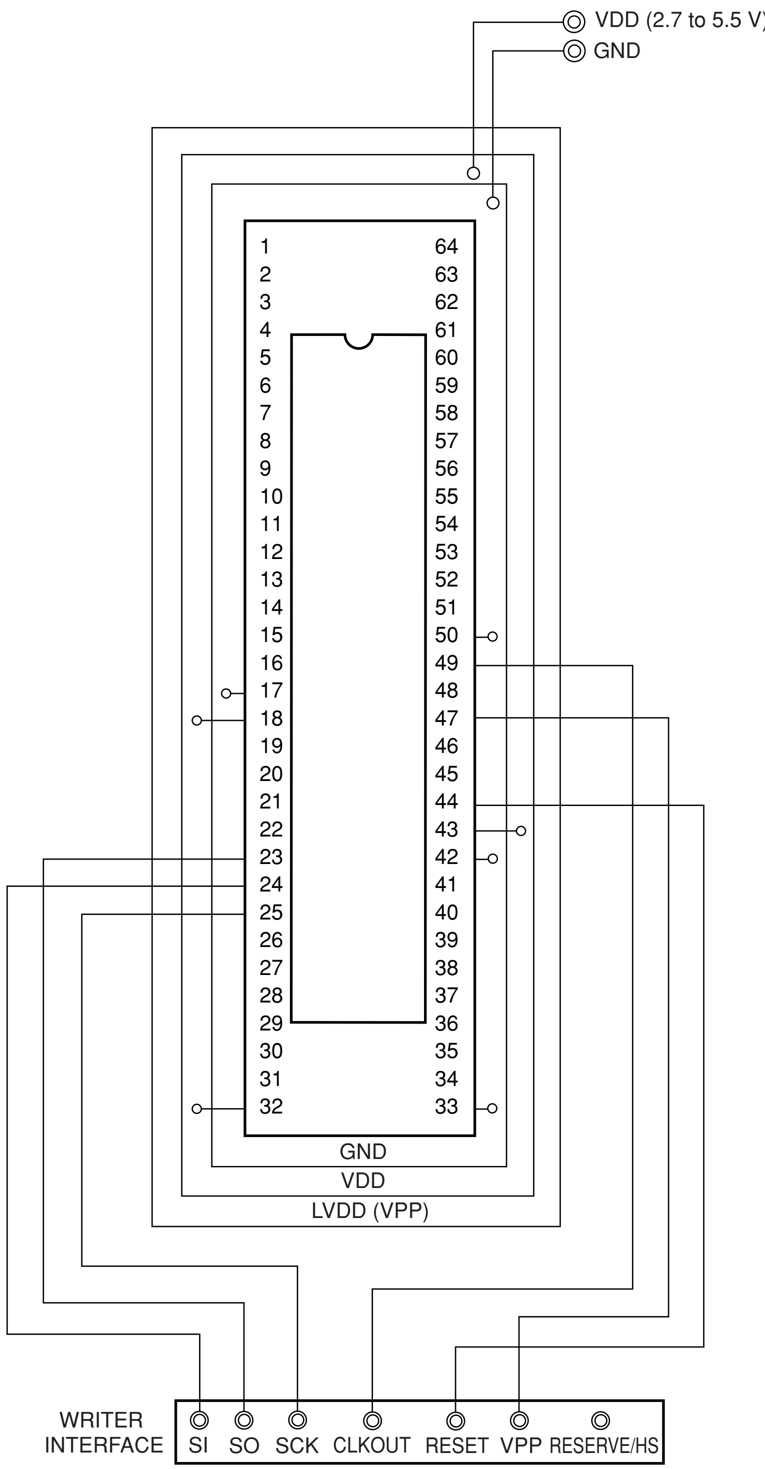


Figure 23-10. Example of Wiring Adapter for Flash Memory Writing in 3-Wire Serial I/O (SIO31) Mode (μPD78F0034A, 78F0034B only) (2/2)

(2) 64-pin plastic QFP (14 × 14), 64-pin plastic LQFP (14 × 14), 64-pin plastic TQFP (12 × 12), 64-pin plastic LQFP (10 × 10)

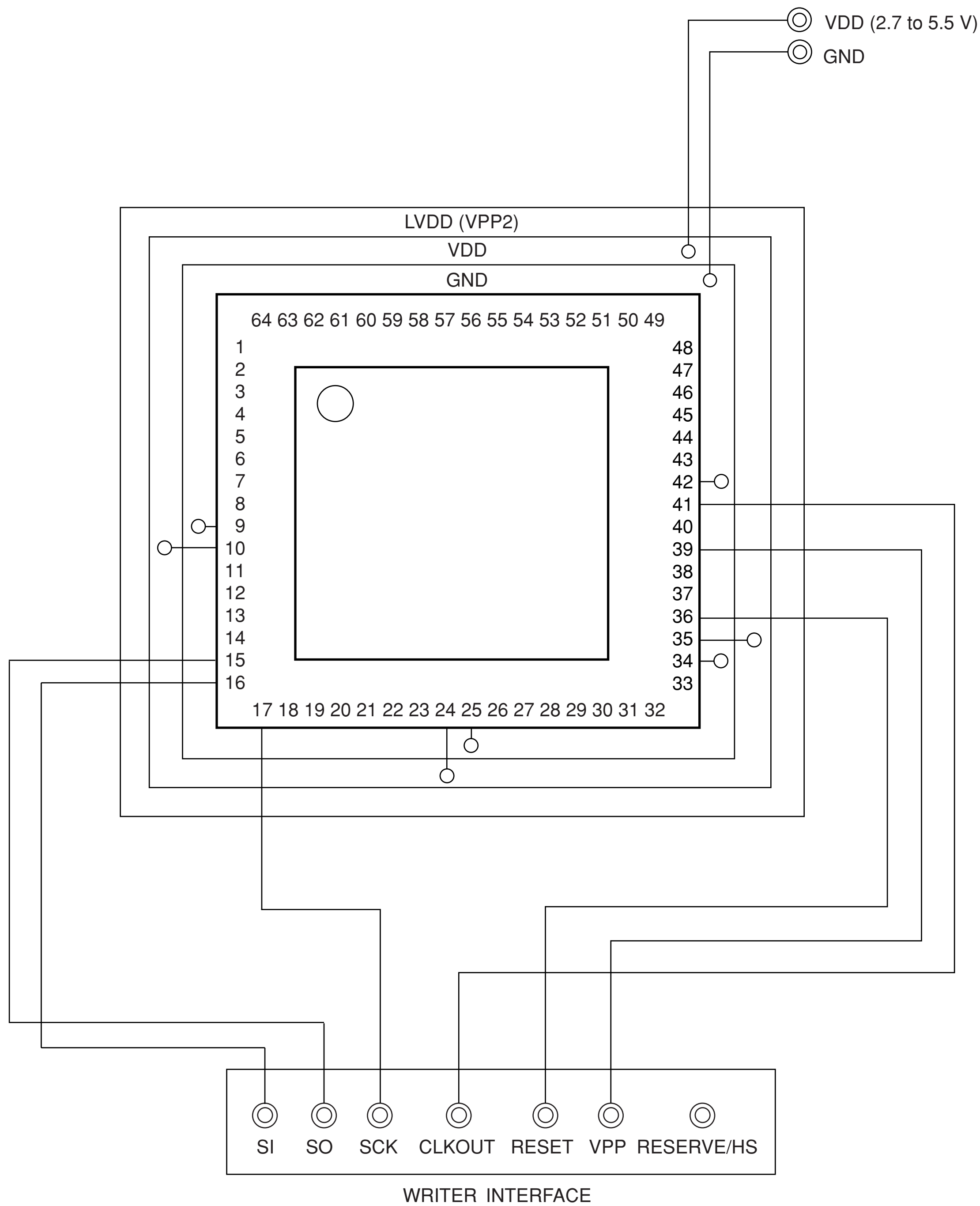


Figure 23-11. Example of Wiring Adapter for Flash Memory Writing in 3-Wire Serial I/O (SIO30 + HS) Mode (μPD78F0034B, 78F0034BY only) (1/2)

(1) 64-pin plastic SDIP (19.05 mm (750))

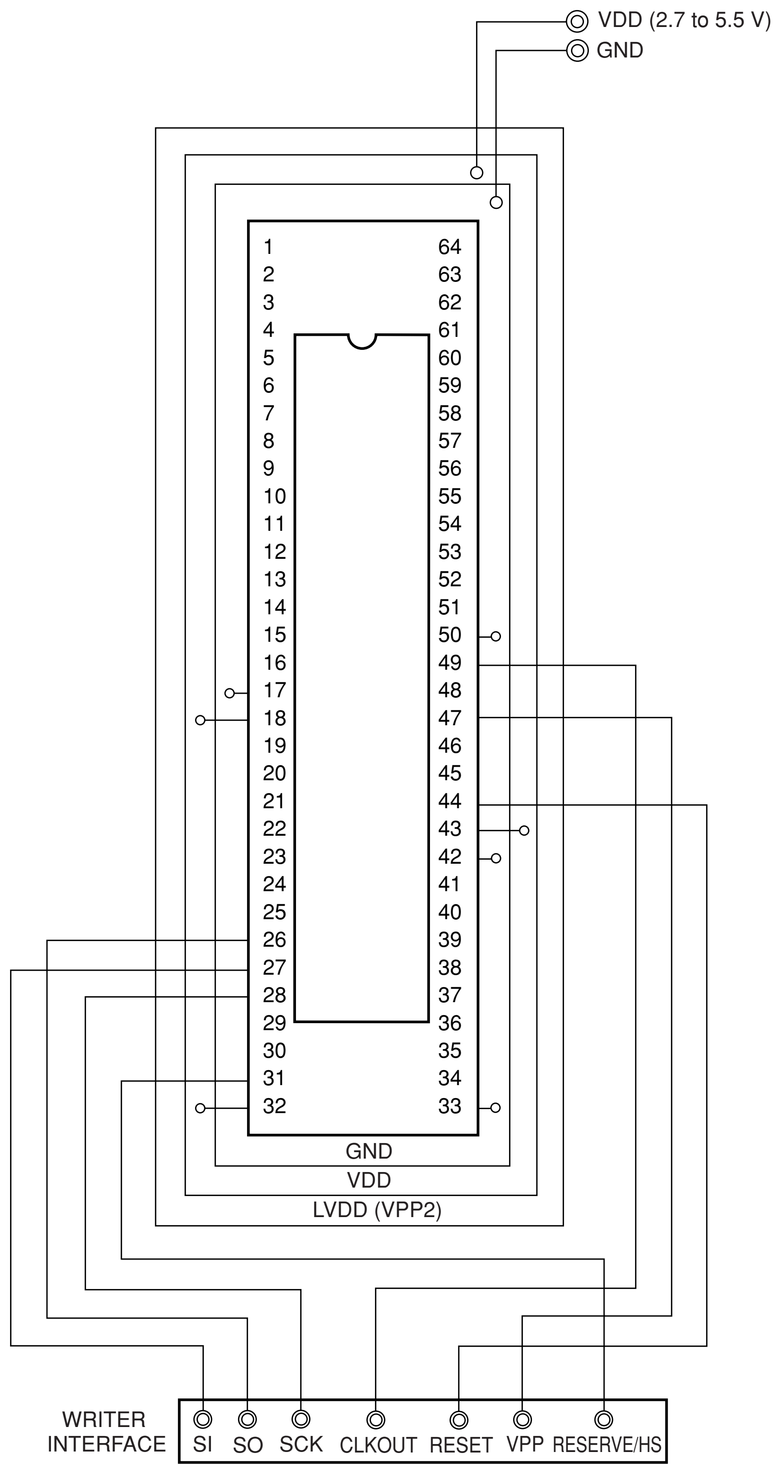


Figure 23-11. Example of Wiring Adapter for Flash Memory Writing in 3-Wire Serial I/O (SIO30 + HS) Mode (μPD78F0034B, 78F0034BY only) (2/2)

(2) 64-pin plastic QFP (14 × 14), 64-pin plastic LQFP (14 × 14), 64-pin plastic TQFP (12 × 12), 64-pin plastic LQFP (10 × 10)

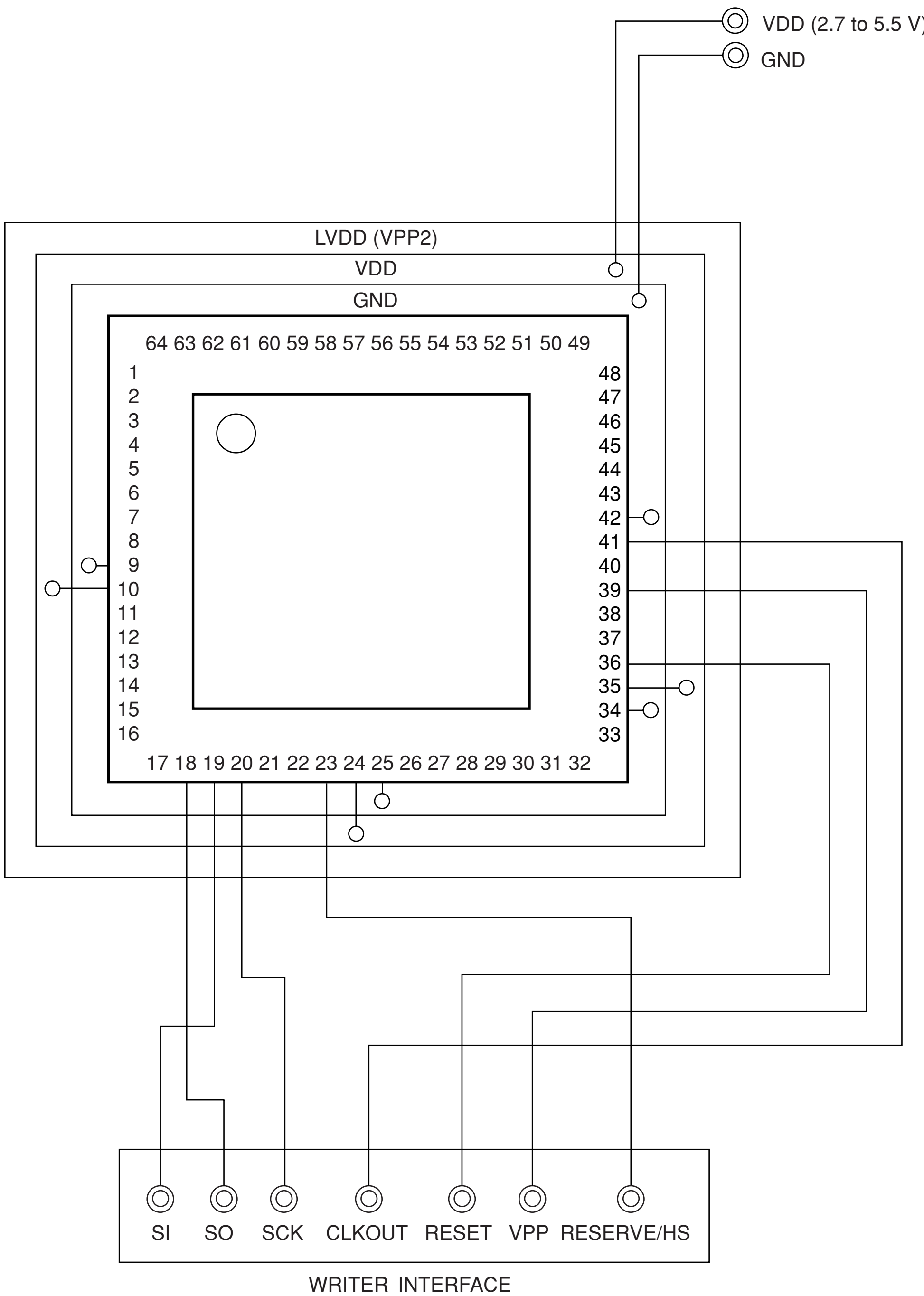


Figure 23-12. Example of Wiring Adapter for Flash Memory Writing in I²C Bus (IIC0) Mode (μPD78F0034AY, 78F0034BY only) (1/2)

(1) 64-pin plastic SDIP (19.05 mm (750))

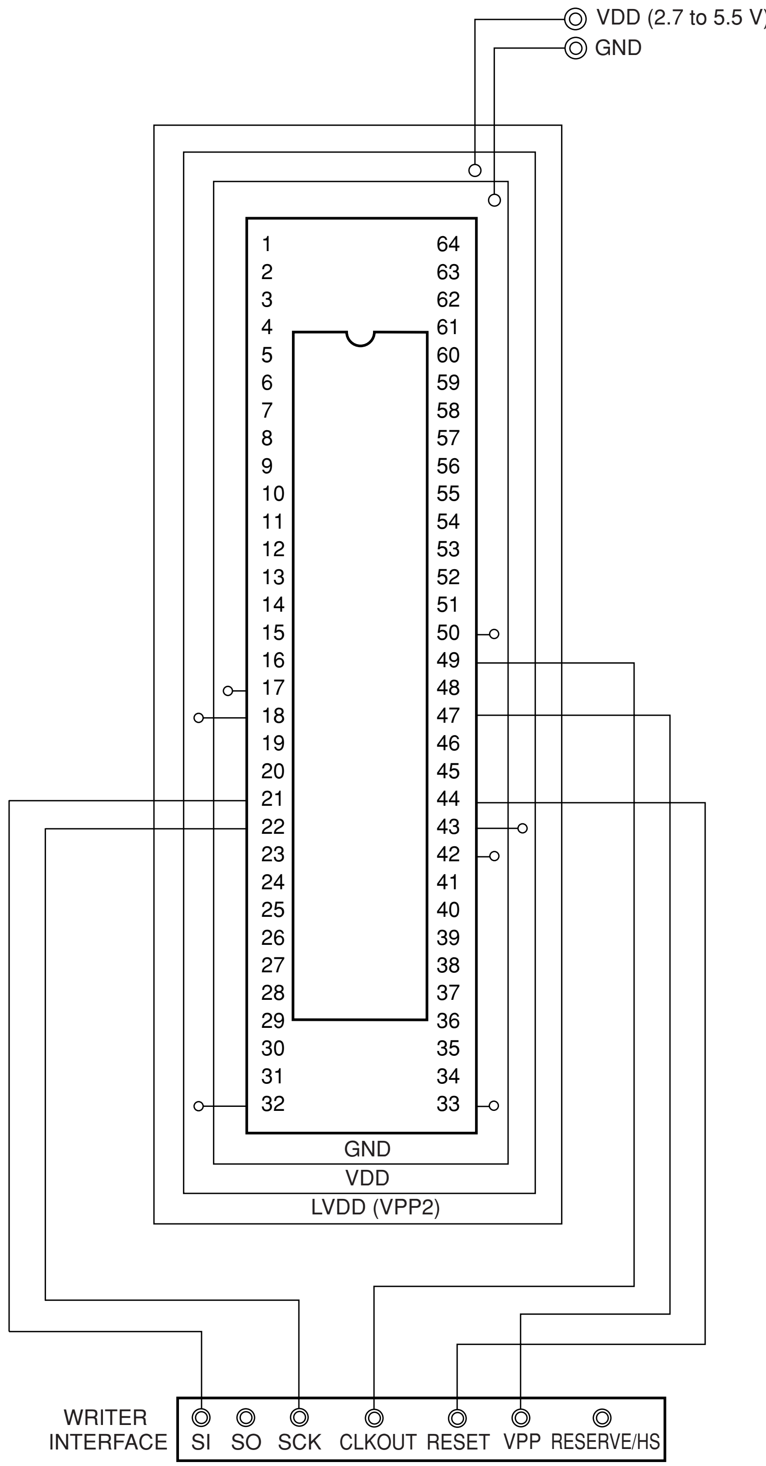


Figure 23-12. Example of Wiring Adapter for Flash Memory Writing in I²C Bus (IIC0) Mode (μPD78F0034AY, 78F0034BY only) (2/2)

(2) 64-pin plastic QFP (14 × 14), 64-pin plastic LQFP (14 × 14), 64-pin plastic TQFP (12 × 12), 64-pin plastic LQFP (10 × 10)

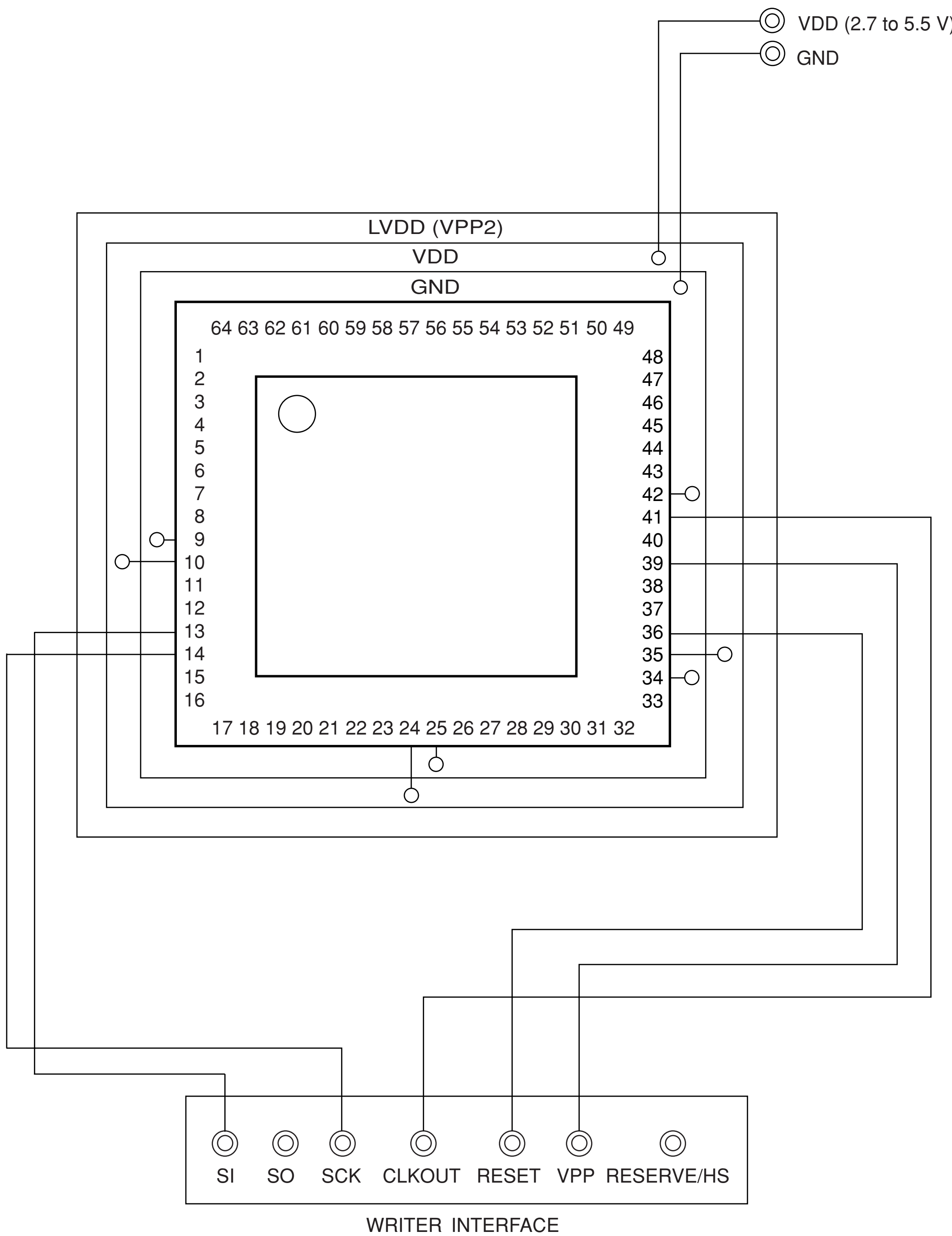


Figure 23-13. Example of Wiring Adapter for Flash Memory Writing in UART (UART0) Mode (1/2)

(1) 64-pin plastic SDIP (19.05 mm (750))

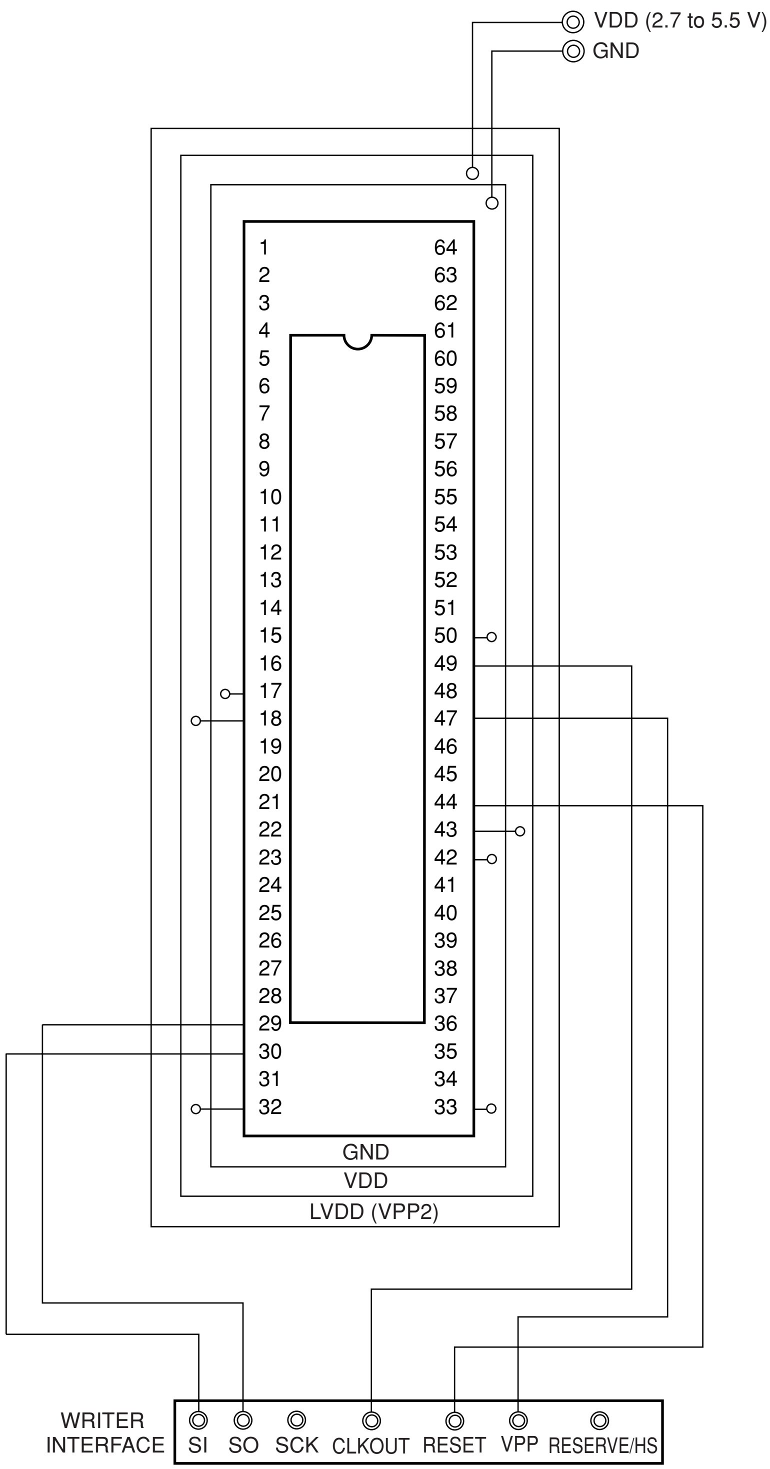


Figure 23-13. Example of Wiring Adapter for Flash Memory Writing in UART (UART0) Mode (2/2)

(2) 64-pin plastic QFP (14 × 14), 64-pin plastic LQFP (14 × 14), 64-pin plastic TQFP (12 × 12), 64-pin plastic LQFP (10 × 10)

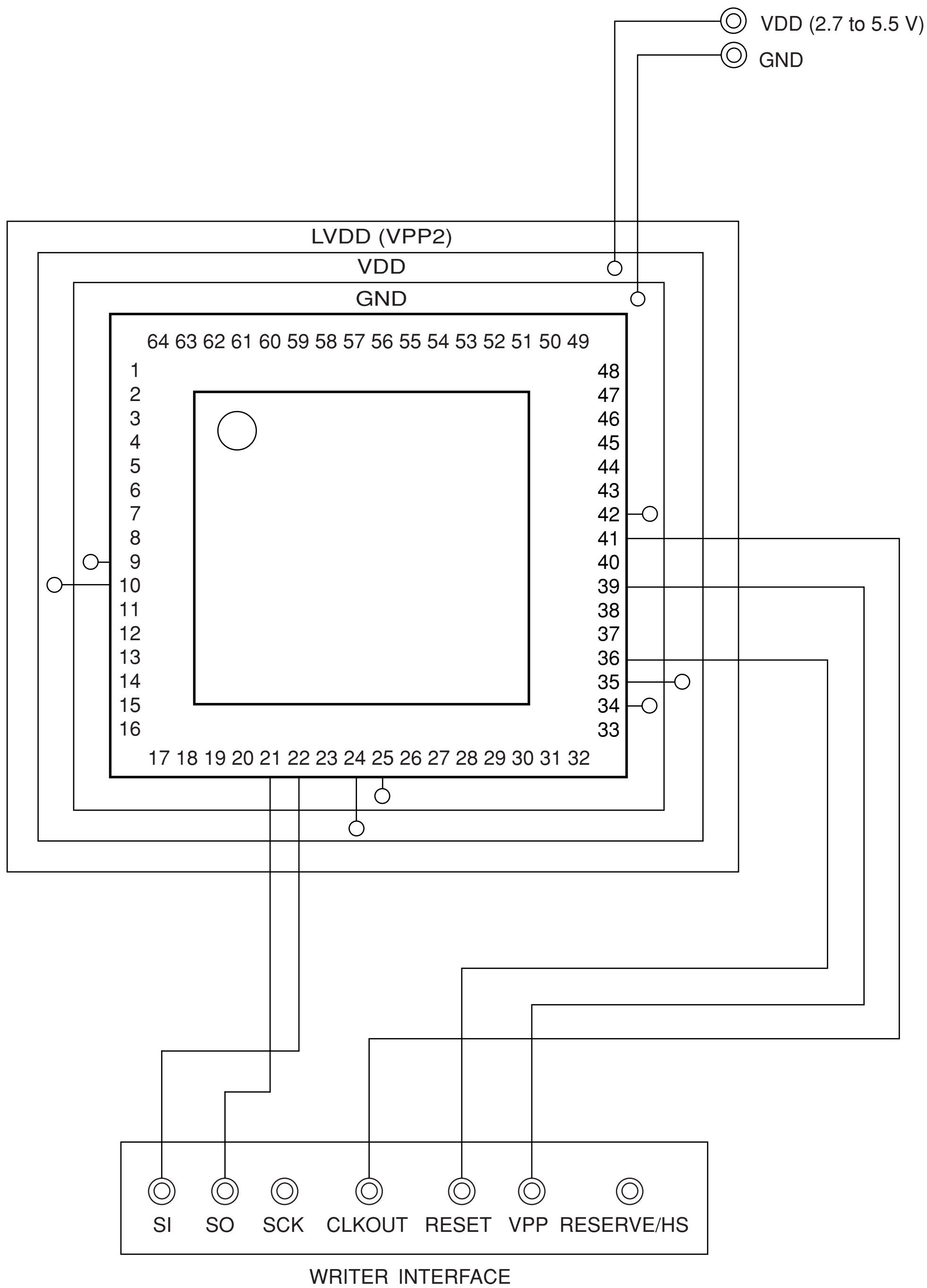


Figure 23-14. Example of Wiring Adapter for Flash Memory Writing in Pseudo 3-Wire Serial I/O Mode (1/2)

(1) 64-pin plastic SDIP (19.05 mm (750))

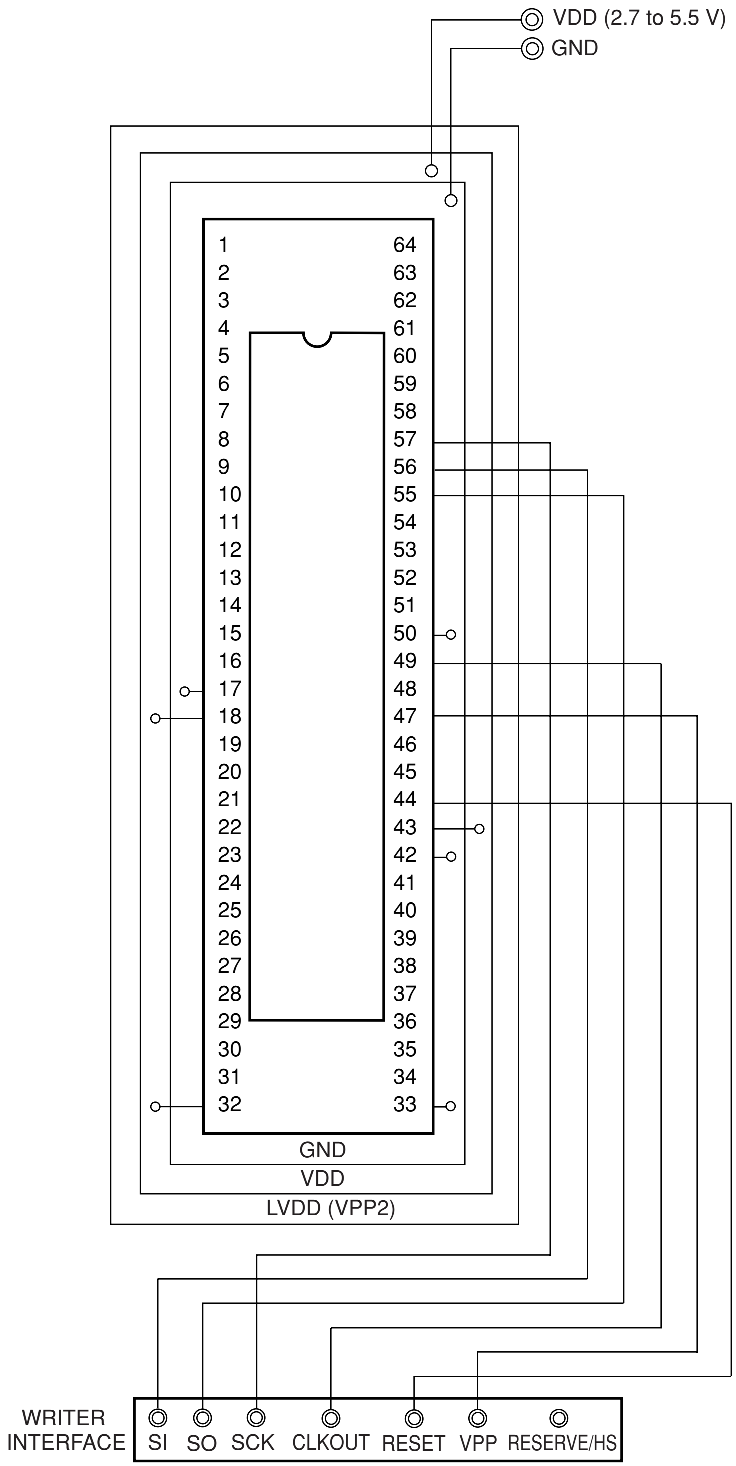
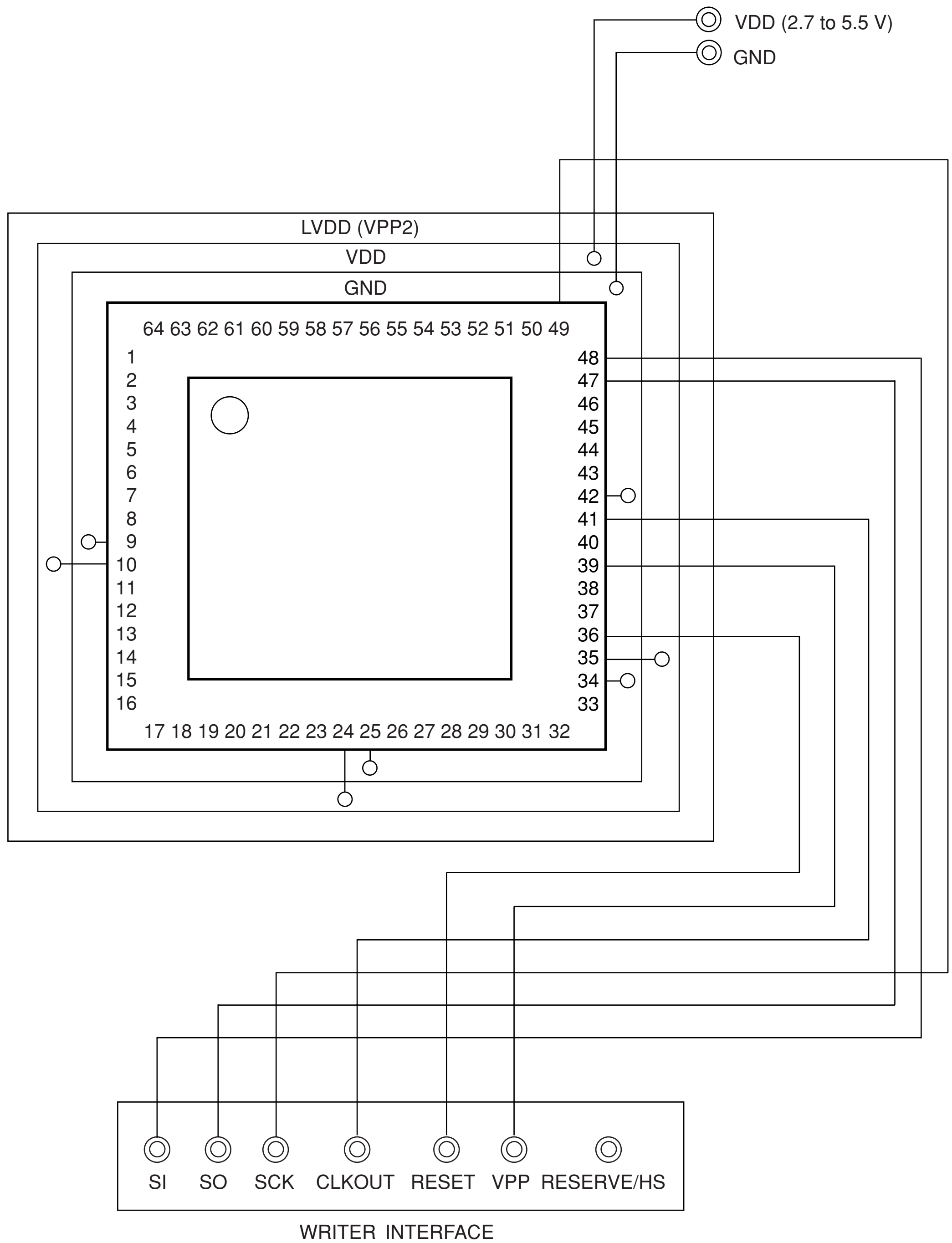


Figure 23-14. Example of Wiring Adapter for Flash Memory Writing in Pseudo 3-Wire Serial I/O Mode (2/2)

(2) 64-pin plastic QFP (14 × 14), 64-pin plastic LQFP (14 × 14), 64-pin plastic TQFP (12 × 12), 64-pin plastic LQFP (10 × 10)



CHAPTER 24 INSTRUCTION SET

This chapter lists each instruction set of the μPD780024A, 780034A, 780024AY, 780034AY Subseries in table form. For details of its operation and operation code, refer to the separate document **78K/0 Series Instructions User's Manual (U12326E)**.

24.1 Conventions

24.1.1 Operand identifiers and specification methods

Operands are written in “Operand” column of each instruction in accordance with the specification method of the instruction operand identifier (refer to the assembler specifications for detail). When there are two or more methods, select one of them. Alphabetic letters in capitals and symbols, #, !, $, and [ ] are key words and must be written as they are. Each symbol has the following meaning.

- #: Immediate data specification
- !: Absolute address specification
- $: Relative address specification
- [ ]: Indirect address specification

In the case of immediate data, describe an appropriate numeric value or a label. When using a label, be sure to write the #, !, $, and [ ] symbols.

For operand register identifiers, r and rp, either function names (X, A, C, etc.) or absolute names (names in parentheses in the table below, R0, R1, R2, etc.) can be used for specification.

Table 24-1. Operand Identifiers and Specification Methods

Identifier	Specification Method
r rp sfr sfrp	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7) AX (RP0), BC (RP1), DE (RP2), HL (RP3) Special function register symbol[Note] Special function register symbol (16-bit manipulatable register even addresses only)[Note]
saddr saddrp	FE20H to FF1FH Immediate data or labels FE20H to FF1FH Immediate data or labels (even address only)
addr16 addr11 addr5	0000H to FFFFH Immediate data or labels (only even addresses for 16-bit data transfer instructions) 0800H to 0FFFH Immediate data or labels 0040H to 007FH Immediate data or labels (even address only)
word byte bit	16-bit immediate data or label 8-bit immediate data or label 3-bit immediate data or label
RBn	RB0 to RB3

Note Addresses from FFD0H to FFDFH cannot be accessed with these operands.

Remark For special function register symbols, see **Table 5-5 Special Function Register List**.

24.1.2 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
X_H, X_L: Higher 8 bits and lower 8 bits of 16-bit register
$\wedge$: Logical product (AND)
$\vee$: Logical sum (OR)
$\veebar$: Exclusive logical sum (exclusive OR)
$\overline{\quad}$: Inverted data
addr16: 16-bit immediate data or label
jdisp8: Signed 8-bit data (displacement value)

24.1.3 Description of “flag operation” column

(Blank): Not affected
0: Cleared to 0
1: Set to 1
×: Set/cleared according to the result
R: Previously saved value is restored

24.2 Operation List

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag
				Note 1	Note 2		Z AC CY
8-bit data transfer	**MOV**	r, #byte	2	4	–	r ← byte	
		saddr, #byte	3	6	7	(saddr) ← byte	
		sfr, #byte	3	–	7	sfr ← byte	
		A, r Note 3	1	2	–	A ← r	
		r, A Note 3	1	2	–	r ← A	
		A, saddr	2	4	5	A ← (saddr)	
		saddr, A	2	4	5	(saddr) ← A	
		A, sfr	2	–	5	A ← sfr	
		sfr, A	2	–	5	sfr ← A	
		A, !addr16	3	8	9 + n	A ← (addr16)	
		!addr16, A	3	8	9 + m	(addr16) ← A	
		PSW, #byte	3	–	7	PSW ← byte	× × ×
		A, PSW	2	–	5	A ← PSW	
		PSW, A	2	–	5	PSW ← A	× × ×
		A, [DE]	1	4	5 + 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	
	XCH	A, r Note 3	1	2	–	A ↔ r	
		A, saddr	2	4	6	A ↔ (saddr)	
		A, sfr	2	–	6	A ↔ (sfr)	
		A, !addr16	3	8	10 + 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 with no data access
2. When an area except the internal high-speed RAM area is accessed
3. Except r = A

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to 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	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit data transfer	**MOVW**	rp, #word	3	6	–	rp ← word			
		saddrp, #word	4	8	10	(saddrp) ← word			
		sfrp, #word	4	–	10	sfrp ← word			
		AX, saddrp	2	6	8	AX ← (saddrp)			
		saddrp, AX	2	6	8	(saddrp) ← AX			
		AX, sfrp	2	–	8	AX ← sfrp			
		sfrp, AX	2	–	8	sfrp ← AX			
		AX, rp **Note 3**	1	4	–	AX ← rp			
		rp, AX **Note 3**	1	4	–	rp ← AX			
		AX, !addr16	3	10	12 + 2n	AX ← (addr16)			
		!addr16, AX	3	10	12 + 2m	(addr16) ← AX			
	XCHW	AX, rp **Note 3**	1	4	–	AX ↔ rp			
8-bit operation	**ADD**	A, #byte	2	4	–	A, CY ← A + byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte	×	×	×
		A, r **Note 4**	2	4	–	A, CY ← A + r	×	×	×
		r, A	2	4	–	r, CY ← r + A	×	×	×
		A, saddr	2	4	5	A, CY ← A + (saddr)	×	×	×
		A, !addr16	3	8	9 + 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)	×	×	×
	ADDC	A, #byte	2	4	–	A, CY ← A + byte + CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte + CY	×	×	×
		A, r **Note 4**	2	4	–	A, CY ← A + r + CY	×	×	×
		r, A	2	4	–	r, CY ← r + A + CY	×	×	×
		A, saddr	2	4	5	A, CY ← A + (saddr) + CY	×	×	×
		A, !addr16	3	8	9 + 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	×	×	×

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access
2. When an area except the internal high-speed RAM area is accessed
3. Only when rp = BC, DE or HL
4. Except r = A

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).
2. This clock cycle applies to 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	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	**SUB**	A, #byte	2	4	–	A, CY ← A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte	×	×	×
		A, r Note 3	2	4	–	A, CY ← A – r	×	×	×
		r, A	2	4	–	r, CY ← r – A	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr)	×	×	×
		A, !addr16	3	8	9 + 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)	×	×	×
	SUBC	A, #byte	2	4	–	A, CY ← A – byte – CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte – CY	×	×	×
		A, r Note 3	2	4	–	A, CY ← A – r – CY	×	×	×
		r, A	2	4	–	r, CY ← r – A – CY	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr) – CY	×	×	×
		A, !addr16	3	8	9 + 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	×	×	×
	AND	A, #byte	2	4	–	A ← A∧byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr)∧byte	×		
		A, r Note 3	2	4	–	A ← A∧r	×		
		r, A	2	4	–	r ← r∧A	×		
		A, saddr	2	4	5	A ← A∧(saddr)	×		
		A, !addr16	3	8	9 + 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 with no data access

2. When an area except the internal high-speed RAM area is accessed

3. Except r = A

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to internal ROM program.

3. n is the number of waits when external memory expansion area is read from.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	**OR**	A, #byte	2	4	–	A ← A$\vee$byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr)$\vee$byte	×		
		A, r Note 3	2	4	–	A ← A$\vee$r	×		
		r, A	2	4	–	r ← r$\vee$A	×		
		A, saddr	2	4	5	A ← A$\vee$(saddr)	×		
		A, !addr16	3	8	9 + n	A ← A$\vee$(addr16)	×		
		A, [HL]	1	4	5 + n	A ← A$\vee$(HL)	×		
		A, [HL + byte]	2	8	9 + n	A ← A$\vee$(HL + byte)	×		
		A, [HL + B]	2	8	9 + n	A ← A$\vee$(HL + B)	×		
		A, [HL + C]	2	8	9 + n	A ← A$\vee$(HL + C)	×		
	XOR	A, #byte	2	4	–	A ← A$\forall$byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr)$\forall$byte	×		
		A, r Note 3	2	4	–	A ← A$\forall$r	×		
		r, A	2	4	–	r ← r$\forall$A	×		
		A, saddr	2	4	5	A ← A$\forall$(saddr)	×		
		A, !addr16	3	8	9 + n	A ← A$\forall$(addr16)	×		
		A, [HL]	1	4	5 + n	A ← A$\forall$(HL)	×		
		A, [HL + byte]	2	8	9 + n	A ← A$\forall$(HL + byte)	×		
		A, [HL + B]	2	8	9 + n	A ← A$\forall$(HL + B)	×		
		A, [HL + C]	2	8	9 + n	A ← A$\forall$(HL + C)	×		
	CMP	A, #byte	2	4	–	A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr) – byte	×	×	×
		A, r Note 3	2	4	–	A – r	×	×	×
		r, A	2	4	–	r – A	×	×	×
		A, saddr	2	4	5	A – (saddr)	×	×	×
		A, !addr16	3	8	9 + n	A – (addr16)	×	×	×
		A, [HL]	1	4	5 + n	A – (HL)	×	×	×
		A, [HL + byte]	2	8	9 + n	A – (HL + byte)	×	×	×
		A, [HL + B]	2	8	9 + n	A – (HL + B)	×	×	×
		A, [HL + C]	2	8	9 + n	A – (HL + C)	×	×	×

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

3. Except r = A

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to internal ROM program.

3. n is the number of waits when external memory expansion area is read from.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit operation	**ADDW**	AX, #word	3	6	–	AX, CY ← AX + word	×	×	×
	SUBW	AX, #word	3	6	–	AX, CY ← AX – word	×	×	×
	CMPW	AX, #word	3	6	–	AX – word	×	×	×
Multiply/ divide	**MULU**	X	2	16	–	AX ← A × X			
	DIVUW	C	2	25	–	AX (Quotient), C (Remainder) ← AX ÷ C			
Increment/ decrement	**INC**	r	1	2	–	r ← r + 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) + 1	×	×	
	DEC	r	1	2	–	r ← r – 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) – 1	×	×	
	INCW	rp	1	4	–	rp ← rp + 1			
	DECW	rp	1	4	–	rp ← rp – 1			
Rotate	**ROR**	A, 1	1	2	–	(CY, $A_7 \leftarrow A_0$, $A_{m-1} \leftarrow A_m$) × 1 time			×
	ROL	A, 1	1	2	–	(CY, $A_0 \leftarrow A_7$, $A_{m+1} \leftarrow A_m$) × 1 time			×
	RORC	A, 1	1	2	–	(CY ← A_0, A_7 ← CY, $A_{m-1} \leftarrow A_m$) × 1 time			×
	ROLC	A, 1	1	2	–	(CY ← A_7, A_0 ← CY, $A_{m+1} \leftarrow A_m$) × 1 time			×
	ROR4	[HL]	2	10	12 + n + m	$A_{3-0} \leftarrow (HL)_{3-0}$, $(HL)_{7-4} \leftarrow A_{3-0}$, $(HL)_{3-0} \leftarrow (HL)_{7-4}$			
	ROL4	[HL]	2	10	12 + n + m	$A_{3-0} \leftarrow (HL)_{7-4}$, $(HL)_{3-0} \leftarrow A_{3-0}$, $(HL)_{7-4} \leftarrow (HL)_{3-0}$			
BCD adjust	**ADJBA**		2	4	–	Decimal Adjust Accumulator after Addition	×	×	×
	ADJBS		2	4	–	Decimal Adjust Accumulator after Subtract	×	×	×
Bit manipu- late	**MOV1**	CY, saddr.bit	3	6	7	CY ← (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← sfr.bit			×
		CY, A.bit	2	4	–	CY ← A.bit			×
		CY, PSW.bit	3	–	7	CY ← PSW.bit			×
		CY, [HL].bit	2	6	7 + 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 with no data access

2. When an area except the internal high-speed RAM area is accessed

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to 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	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Bit manipulate	**AND1**	CY, saddr.bit	3	6	7	CY ← CY∧(saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY∧sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY∧A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY∧PSW.bit			×
		CY, [HL].bit	2	6	7 + n	CY ← CY∧(HL).bit			×
	OR1	CY, saddr.bit	3	6	7	CY ← CY∨(saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY∨sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY∨A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY∨PSW.bit			×
		CY, [HL].bit	2	6	7 + n	CY ← CY∨(HL).bit			×
	XOR1	CY, saddr.bit	3	6	7	CY ← CY⊻(saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY⊻sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY⊻A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY⊻PSW.bit			×
		CY, [HL].bit	2	6	7 + n	CY ← CY⊻(HL).bit			×
	SET1	saddr.bit	2	4	6	(saddr.bit) ← 1			
		sfr.bit	3	–	8	sfr.bit ← 1			
		A.bit	2	4	–	A.bit ← 1			
		PSW.bit	2	–	6	PSW.bit ← 1	×	×	×
		[HL].bit	2	6	8 + n + m	(HL).bit ← 1			
	CLR1	saddr.bit	2	4	6	(saddr.bit) ← 0			
		sfr.bit	3	–	8	sfr.bit ← 0			
		A.bit	2	4	–	A.bit ← 0			
		PSW.bit	2	–	6	PSW.bit ← 0	×	×	×
		[HL].bit	2	6	8 + n + m	(HL).bit ← 0			
	SET1	CY	1	2	–	CY ← 1			1
	CLR1	CY	1	2	–	CY ← 0			0
	NOT1	CY	1	2	–	CY ← $\overline{\text{CY}}$			×

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to 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	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Call/return	**CALL**	!addr16	3	7	–	$(SP - 1) \leftarrow (PC + 3)_H$, $(SP - 2) \leftarrow (PC + 3)_L$, $PC \leftarrow addr16$, $SP \leftarrow SP - 2$			
	CALLF	!addr11	2	5	–	$(SP - 1) \leftarrow (PC + 2)_H$, $(SP - 2) \leftarrow (PC + 2)_L$, $PC_{15-11} \leftarrow 00001$, $PC_{10-0} \leftarrow addr11$, $SP \leftarrow SP - 2$			
	CALLT	[addr5]	1	6	–	$(SP - 1) \leftarrow (PC + 1)_H$, $(SP - 2) \leftarrow (PC + 1)_L$, $PC_H \leftarrow (00000000, addr5 + 1)$, $PC_L \leftarrow (00000000, addr5)$, $SP \leftarrow SP - 2$			
	BRK		1	6	–	$(SP - 1) \leftarrow PSW$, $(SP - 2) \leftarrow (PC + 1)_H$, $(SP - 3) \leftarrow (PC + 1)_L$, $PC_H \leftarrow (003FH)$, $PC_L \leftarrow (003EH)$, $SP \leftarrow SP - 3$, $IE \leftarrow 0$			
	RET		1	6	–	$PC_H \leftarrow (SP + 1)$, $PC_L \leftarrow (SP)$, $SP \leftarrow SP + 2$			
	RETI		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
	RETB		1	6	–	$PC_H \leftarrow (SP + 1)$, $PC_L \leftarrow (SP)$, $PSW \leftarrow (SP + 2)$, $SP \leftarrow SP + 3$	R	R	R
Stack manipu-late	**PUSH**	PSW	1	2	–	$(SP - 1) \leftarrow PSW$, $SP \leftarrow SP - 1$			
		rp	1	4	–	$(SP - 1) \leftarrow rp_H$, $(SP - 2) \leftarrow rp_L$, $SP \leftarrow SP - 2$			
	POP	PSW	1	2	–	$PSW \leftarrow (SP)$, $SP \leftarrow SP + 1$	R	R	R
		rp	1	4	–	$rp_H \leftarrow (SP + 1)$, $rp_L \leftarrow (SP)$, $SP \leftarrow SP + 2$			
	MOVW	SP, #word	4	–	10	$SP \leftarrow word$			
		SP, AX	2	–	8	$SP \leftarrow AX$			
		AX, SP	2	–	8	$AX \leftarrow SP$			
Uncondi-tional branch	**BR**	!addr16	3	6	–	$PC \leftarrow addr16$			
		$addr16	2	6	–	$PC \leftarrow PC + 2 + jdisp8$			
		AX	2	8	–	$PC_H \leftarrow A$, $PC_L \leftarrow X$			
Conditional branch	**BC**	$addr16	2	6	–	$PC \leftarrow PC + 2 + jdisp8$ if $CY = 1$			
	BNC	$addr16	2	6	–	$PC \leftarrow PC + 2 + jdisp8$ if $CY = 0$			
	BZ	$addr16	2	6	–	$PC \leftarrow PC + 2 + jdisp8$ if $Z = 1$			
	BNZ	$addr16	2	6	–	$PC \leftarrow PC + 2 + jdisp8$ if $Z = 0$			

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to internal ROM program.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag
				Note 1	Note 2		Z AC CY
Conditional branch	**BT**	saddr.bit, $addr16	3	8	9	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1	
		sfr.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.bit = 1	
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 1	
		PSW.bit, $addr16	3	–	9	PC ← PC + 3 + jdisp8 if PSW.bit = 1	
		[HL].bit, $addr16	3	10	11 + n	PC ← PC + 3 + jdisp8 if (HL).bit = 1	
	BF	saddr.bit, $addr16	4	10	11	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0	
		sfr.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.bit = 0	
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 0	
		PSW.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if PSW.bit = 0	
		[HL].bit, $addr16	3	10	11 + n	PC ← PC + 3 + jdisp8 if (HL).bit = 0	
	BTCLR	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	
	DBNZ	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	**SEL**	RBn	2	4	–	RBS1, 0 ← n	
	NOP		1	2	–	No Operation	
	EI		2	–	6	IE ← 1 (Enable Interrupt)	
	DI		2	–	6	IE ← 0 (Disable Interrupt)	
	HALT		2	6	–	Set HALT Mode	
	STOP		2	6	–	Set STOP Mode	

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to 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.

24.3 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 ROLC	
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 / First Operand	#word	AX	rp Note	sfrp	saddrp	!addr16	SP	None
AX	ADDW SUBW CMPW		MOVW XCHW	MOVW	MOVW	MOVW	MOVW	
rp	MOVW	MOVW Note						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

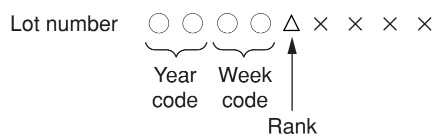
★

CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS: fx = 1.0 TO 12 MHz)

Target products

- μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A, 780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), 780034A(A) for which orders were received after December 1, 2001 (Products with a rank[Note] other than K, E, P, X)
- μPD78F0034B, 78F0034B(A)

Note The rank is indicated by the 5th digit from the left in the lot number marked on the package.



Absolute Maximum Ratings (T_A = 25°C)

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V_{DD}			–0.3 to +6.5	V
	V_{PP}	Flash memory version only, **Note 2**		–0.3 to +10.5	V
	AV_{DD}			–0.3 to V_{DD} + 0.3**Note 1**	V
	AV_{REF}			–0.3 to V_{DD} + 0.3**Note 1**	V
	AV_{SS}			–0.3 to +0.3	V
Input voltage	V_{I1}	P00 to P03, P10 to P17, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75, X1, X2, XT1, XT2, $\overline{\text{RESET}}$		–0.3 to V_{DD} + 0.3**Note 1**	V
	V_{I2}	P30 to P33	N-ch open drain	–0.3 to +6.5	V
			On-chip pull-up resistor	–0.3 to V_{DD} + 0.3**Note 1**	V
Output voltage	V_O			–0.3 to V_{DD} + 0.3**Note 1**	V
Analog input voltage	V_{AN}	P10 to P17	Analog input pin	AV_{SS} – 0.3 to AV_{REF} + 0.3**Note 1** and –0.3 to V_{DD} + 0.3**Note 1**	V
Output current, high	I_{OH}	Per pin		–10	mA
		Total for P00 to P03, P40 to P47, P50 to P57, P64 to P67, P70 to P75		–15	mA
		Total for P20 to P25, P30 to P36		–15	mA
Output current, low	I_{OL}	Per pin for P00 to P03, P20 to P25, P34 to P36, P40 to P47, P64 to P67, P70 to P75		20	mA
		Per pin for P30 to P33, P50 to P57		30	mA
		Total for P00 to P03, P40 to P47, P64 to P67, P70 to P75		50	mA
		Total for P20 to P25		20	mA
		Total for P30 to P36		100	mA
		Total for P50 to P57		100	mA
Operating ambient temperature	T_A	During normal operation		–40 to +85	°C
		During flash memory programming		+10 to +80	°C
Storage temperature	T_{stg}	Mask ROM version		–65 to +150	°C
		Flash memory version		–40 to +125	°C

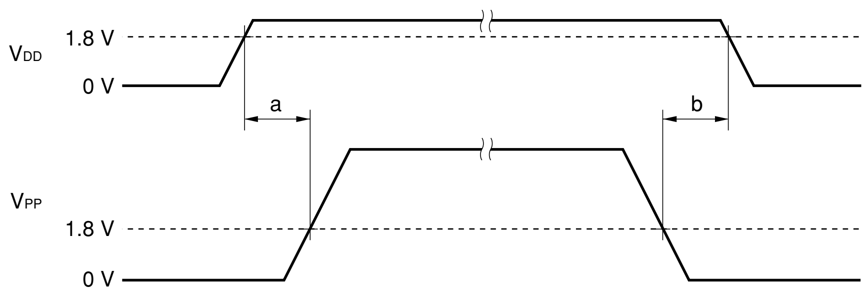
Notes 1. 6.5 V or below
(**Note 2** is explained on the following page.)

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.

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

Notes 2. 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 μs or more after V_{DD} has reached the lower-limit value (1.8 V) of the operating voltage range (see a in the figure below).
- **When supply voltage drops**
  V_{DD} must be lowered 10 μs or more after V_{PP} falls below the lower-limit value (1.8 V) of the operating voltage range of V_{DD} (see b in the figure below).

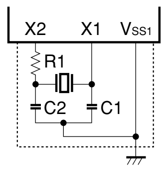
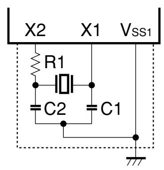
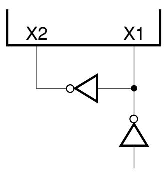


Capacitance (T_A = 25°C, V_{DD} = V_{SS} = 0 V)

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.	P00 to P03, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75			15	pF
			P30 to P33			20	pF

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

Main System Clock Oscillator Characteristics (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f_X)Note 1	4.5 V ≤ V_{DD} ≤ 5.5 V	1.0		12.0	MHz
			3.0 V ≤ V_{DD} < 4.5 V	1.0		8.38	
			1.8 V ≤ V_{DD} < 3.0 V	1.0		5.0	
		Oscillation stabilization timeNote 2	After V_{DD} reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency (f_X)Note 1	4.5 V ≤ V_{DD} ≤ 5.5 V	1.0		12.0	MHz
			3.0 V ≤ V_{DD} < 4.5 V	1.0		8.38	
			1.8 V ≤ V_{DD} < 3.0 V	1.0		5.0	
		Oscillation stabilization timeNote 2	4.0 V ≤ V_{DD} ≤ 5.5 V			10	ms
			1.8 V ≤ V_{DD} < 4.0 V			30	
External clock		X1 input frequency (f_X)Note 1	4.5 V ≤ V_{DD} ≤ 5.5 V	1.0		12.0	MHz
			3.0 V ≤ V_{DD} < 4.5 V	1.0		8.38	
			1.8 V ≤ V_{DD} < 3.0 V	1.0		5.0	
		X1 input high-/low-level width (t_{XH}, t_{XL})	4.5 V ≤ V_{DD} ≤ 5.5 V	38		500	ns
			3.0 V ≤ V_{DD} < 4.5 V	50		500	
			1.8 V ≤ V_{DD} < 3.0 V	85		500	

Notes 1. Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

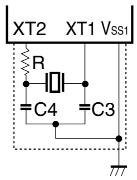
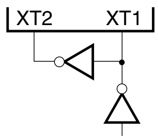
2. Time required to stabilize oscillation after reset or STOP mode release.

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.

Subsystem Clock Oscillator Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Crystal resonator		Oscillation frequency (f_{XT})Note 1		32	32.768	35	kHz
		Oscillation stabilization timeNote 2	4.0 V ≤ V_{DD} ≤ 5.5 V		1.2	2	s
			1.8 V ≤ V_{DD} < 4.0 V			10	
External clock		XT1 input frequency (f_{XT})Note 1		32		38.5	kHz
		XT1 input high-/low-level width (t_{XTH}, t_{XTL})		12		15	μs

Notes 1. Indicates only oscillator characteristics. Refer to **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 figure to avoid an adverse effect from wiring capacitance.

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines.**
- **Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as V_{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 power 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 of the subsystem clock, customers are required to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

Recommended Oscillator Constant

To use the μPD78F0034B or 78F0034B(A), for the resonator selection and oscillator constant, customers are required to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

- **Mask ROM versions of μPD780024A, 780034A Subseries (expanded-specification product)**
  Main system clock: Ceramic resonator (T_A = −40 to +85°C)

Manufacturer	Part Number	Frequency (MHz)	Recommended Circuit Constant			Oscillation Voltage Range	
			C1 (pF)	C2 (pF)	R1 (kΩ)	MIN. (V)	MAX. (V)
Murata Mfg. Co., Ltd.	CSBFB1M00J58	1.00	100	100	2.2	1.8	5.5
	CSBLA1M00J58	1.00	100	100	2.2	1.8	5.5
	CSTCC2M00G56	2.00	On-chip	On-chip	0	1.8	5.5
	CSTLS2M00G56	2.00	On-chip	On-chip	0	1.8	5.5
	CSTCC3M58G53	3.58	On-chip	On-chip	0	1.8	5.5
	CSTLS3M58G53	3.58	On-chip	On-chip	0	1.8	5.5
	CSTCR4M00G53	4.00	On-chip	On-chip	0	1.8	5.5
	CSTLS4M00G53	4.00	On-chip	On-chip	0	1.8	5.5
	CSTCR4M19G53	4.19	On-chip	On-chip	0	1.8	5.5
	CSTLS4M19G53	4.19	On-chip	On-chip	0	1.8	5.5
	CSTCR4M91G53	4.91	On-chip	On-chip	0	1.8	5.5
	CSTLS4M91G53	4.91	On-chip	On-chip	0	1.8	5.5
	CSTCR5M00G53	5.00	On-chip	On-chip	0	1.8	5.5
	CSTLS5M00G53	5.00	On-chip	On-chip	0	1.8	5.5
	CSTCE8M00G52	8.00	On-chip	On-chip	0	2.7	5.5
	CSTLS8M00G53	8.00	On-chip	On-chip	0	2.7	5.5
	CSTCE8M38G52	8.38	On-chip	On-chip	0	3.0	5.5
	CSTLS8M38G53	8.38	On-chip	On-chip	0	3.0	5.5
	CSTCE10M0G52	10.00	On-chip	On-chip	0	3.0	5.5
	CSTLS10M0G53	10.00	On-chip	On-chip	0	3.0	5.5
	CSTCE12M0G52	12.00	On-chip	On-chip	0	4.5	5.5
	CSTLA12M0T55	12.00	On-chip	On-chip	0	4.5	5.5
TDK	CCR3.58MC3	3.58	On-chip	On-chip	0	1.8	5.5
	CCR4.19MC3	4.19	On-chip	On-chip	0	1.8	5.5
	CCR5.0MC3	5.00	On-chip	On-chip	0	1.8	5.5
	CCR8.0MC5	8.00	On-chip	On-chip	0	2.7	5.5
	CCR8.38MC5	8.38	On-chip	On-chip	0	3.0	5.5

Caution The oscillator constant is a reference value based on evaluation in specific environments by the resonator manufacturer. If the oscillator characteristics need to be optimized in the actual application, request the resonator manufacturer for evaluation on the implementation circuit. Note that the oscillation voltage and oscillation frequency merely indicate the characteristics of the oscillator. Use the internal operation conditions of the μPD780024A, 780034A Subseries within the specifications of the DC and AC characteristics.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (1/4)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output current, high	I_{OH}	Per pin				–1	mA
		All pins				–15	mA
Output current, low	I_{OL}	Per pin for P00 to P03, P20 to P25, P34 to P36, P40 to P47, P64 to P67, P70 to P75				10	mA
		Per pin for P30 to P33, P50 to P57				15	mA
		Total for P00 to P03, P40 to P47, P64 to P67, P70 to P75				20	mA
		Total for P20 to P25				10	mA
		Total for P30 to P36				70	mA
		Total for P50 to P57				70	mA
Input voltage, high	V_{IH1}	P10 to P17, P21, P24, P35, P40 to P47, P50 to P57, P64 to P67, P74, P75	2.7 V ≤ V_{DD} ≤ 5.5 V	$0.7V_{DD}$		V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	$0.8V_{DD}$		V_{DD}	V
	V_{IH2}	P00 to P03, P20, P22, P23, P25, P34, P36, P70 to P73, $\overline{\text{RESET}}$	2.7 V ≤ V_{DD} ≤ 5.5 V	$0.8V_{DD}$		V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	$0.85V_{DD}$		V_{DD}	V
	V_{IH3}	P30 to P33 (N-ch open-drain)	2.7 V ≤ V_{DD} ≤ 5.5 V	$0.7V_{DD}$		5.5	V
			1.8 V ≤ V_{DD} < 2.7 V	$0.8V_{DD}$		5.5	V
	V_{IH4}	X1, X2	2.7 V ≤ V_{DD} ≤ 5.5 V	V_{DD} – 0.5		V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	V_{DD} – 0.2		V_{DD}	V
	V_{IH5}	XT1, XT2	4.0 V ≤ V_{DD} ≤ 5.5 V	$0.8V_{DD}$		V_{DD}	V
			1.8 V ≤ V_{DD} < 4.0 V	$0.9V_{DD}$		V_{DD}	V
Input voltage, low	V_{IL1}	P10 to P17, P21, P24, P35, P40 to P47, P50 to P57, P64 to P67, P74, P75	2.7 V ≤ V_{DD} ≤ 5.5 V	0		$0.3V_{DD}$	V
			1.8 V ≤ V_{DD} < 2.7 V	0		$0.2V_{DD}$	V
	V_{IL2}	P00 to P03, P20, P22, P23, P25, P34, P36, P70 to P73, $\overline{\text{RESET}}$	2.7 V ≤ V_{DD} ≤ 5.5 V	0		$0.2V_{DD}$	V
			1.8 V ≤ V_{DD} < 2.7 V	0		$0.15V_{DD}$	V
	V_{IL3}	P30 to P33	4.0 V ≤ V_{DD} ≤ 5.5 V	0		$0.3V_{DD}$	V
			2.7 V ≤ V_{DD} < 4.0 V	0		$0.2V_{DD}$	V
			1.8 V ≤ V_{DD} < 2.7 V	0		$0.1V_{DD}$	V
	V_{IL4}	X1, X2	2.7 V ≤ V_{DD} ≤ 5.5 V	0		0.4	V
			1.8 V ≤ V_{DD} < 2.7 V	0		0.2	V
	V_{IL5}	XT1, XT2	4.0 V ≤ V_{DD} ≤ 5.5 V	0		$0.2V_{DD}$	V
			1.8 V ≤ V_{DD} < 4.0 V	0		$0.1V_{DD}$	V
Output voltage, high	V_{OH1}	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OH} = –1 mA		V_{DD} – 1.0		V_{DD}	V
		1.8 V ≤ V_{DD} < 4.0 V, I_{OH} = –100 μA		V_{DD} – 0.5		V_{DD}	V
Output voltage, low	V_{OL1}	P30 to P33	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OL} = 15 mA			2.0	V
	V_{OL2}	P50 to P57			0.4	2.0	V
	V_{OL3}	P00 to P03, P20 to P25, P34 to P36, P40 to P47, P64 to P67, P70 to P75	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OL} = 1.6 mA			0.4	V
	V_{OL4}	I_{OL} = 400 μA				0.5	V

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (2/4)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	I_{LIH1}	$V_{IN} = V_{DD}$	P00 to P03, P10 to P17, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75, $\overline{\text{RESET}}$			3	μA
	I_{LIH2}		X1, X2, XT1, XT2			20	μA
	I_{LIH3}	V_{IN} = 5.5 V	P30 to P33			3	μA
Input leakage current, low	I_{LIL1}	V_{IN} = 0 V	P00 to P03, P10 to P17, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75, $\overline{\text{RESET}}$			–3	μA
	I_{LIL2}		X1, X2, XT1, XT2			–20	μA
	I_{LIL3}		P30 to P33			–3	μA
Output leakage current, high	I_{LOH}	$V_{OUT} = V_{DD}$				3	μA
Output leakage current, low	I_{LOL}	V_{OUT} = 0 V				–3	μA
Mask option pull-up resistance (mask ROM version only)	R_1	V_{IN} = 0 V, P30, P31, P32, P33		15	30	90	kΩ
Software pull-up resistance	R_2	V_{IN} = 0 V, P00 to P03, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75		15	30	90	kΩ

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V) (3/4)

(1) Mask ROM versions of μPD780024A, 780034A Subseries (expanded-specification product)

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Power supply current Note 1	I_{DD1} Note 2	12.0 MHz crystal oscillation operating mode	V_{DD} = 5.0 V ±10% Note 3	When A/D converter is stopped		8.5	17	mA
				When A/D converter is operating Note 7		9.5	19	mA
		8.38 MHz crystal oscillation operating mode	V_{DD} = 5.0 V ±10% Note 3	When A/D converter is stopped		5.5	11	mA
				When A/D converter is operating Note 7		6.5	13	mA
			V_{DD} = 3.0 V + 10% Notes 3, 6	When A/D converter is stopped		3	6	mA
				When A/D converter is operating Note 7		4	8	mA
		5.00 MHz crystal oscillation operating mode	V_{DD} = 3.0 V ±10% Note 3	When A/D converter is stopped		2	4	mA
				When A/D converter is operating Note 7		3	6	mA
			V_{DD} = 2.0 V ±10% Note 4	When A/D converter is stopped		0.4	1.5	mA
				When A/D converter is operating Note 7		1.4	4.2	mA
	I_{DD2}	12.0 MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ±10% Note 3	When peripheral functions are stopped		2	4	mA
				When peripheral functions are operating			10	mA
		8.38 MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ±10% Note 3	When peripheral functions are stopped		1.1	2.2	mA
				When peripheral functions are operating			4.7	mA
			V_{DD} = 3.0 V + 10% Notes 3, 6	When peripheral functions are stopped		0.5	1	mA
				When peripheral functions are operating			4	mA
		5.00 MHz crystal oscillation HALT mode	V_{DD} = 3.0 V ±10% Note 3	When peripheral functions are stopped		0.35	0.7	mA
				When peripheral functions are operating			1.7	mA
			V_{DD} = 2.0 V ±10% Note 4	When peripheral functions are stopped		0.15	0.4	mA
				When peripheral functions are operating			1.1	mA
	I_{DD3}	32.768 kHz crystal oscillation operating mode Note 5		V_{DD} = 5.0 V ±10%		40	80	μA
				V_{DD} = 3.0 V ±10%		20	40	μA
				V_{DD} = 2.0 V ±10%		10	20	μA
	I_{DD4}	32.768 kHz crystal oscillation HALT mode Note 5		V_{DD} = 5.0 V ±10%		30	60	μA
				V_{DD} = 3.0 V ±10%		6	18	μA
				V_{DD} = 2.0 V ±10%		2	10	μA
	I_{DD5}	XT1 = V_{DD}, STOP mode When feedback resistor is not used		V_{DD} = 5.0 V ±10%		0.1	30	μA
				V_{DD} = 3.0 V ±10%		0.05	10	μA
				V_{DD} = 2.0 V ±10%		0.05	10	μA

Notes 1. Total current through the internal power supply (V_{DD0}, V_{DD1}) (except the current through pull-up resistors of ports).

2. I_{DD1} includes the peripheral operation current.

3. When the processor clock control register (PCC) is cleared to 00H.

4. When PCC is set to 02H.

5. When main system clock operation is stopped.

6. The values show the specifications when V_{DD} = 3.0 to 3.3 V. The value in the TYP. column show the specifications when V_{DD} = 3.0 V.

7. Includes the current through the AV_{DD} pin.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (4/4)

(2) μPD78F0034B, 78F0034B(A)

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Power supply current Note 1	I_{DD1} Note 2	12.0 MHz crystal oscillation operating mode	V_{DD} = 5.0 V ±10% Note 3	When A/D converter is stopped		16	32	mA
				When A/D converter is operating Note 7		17	34	mA
		8.38 MHz crystal oscillation operating mode	V_{DD} = 5.0 V ±10% Note 3	When A/D converter is stopped		10.5	21	mA
				When A/D converter is operating Note 7		11.5	23	mA
			V_{DD} = 3.0 V + 10% Notes 3, 6	When A/D converter is stopped		7	14	mA
				When A/D converter is operating Note 7		8	16	mA
		5.00 MHz crystal oscillation operating mode	V_{DD} = 3.0 V ±10% Note 3	When A/D converter is stopped		4.5	9	mA
				When A/D converter is operating Note 7		5.5	11	mA
			V_{DD} = 2.0 V ±10% Note 4	When A/D converter is stopped		1	2	mA
				When A/D converter is operating Note 7		2	6	mA
	I_{DD2}	12.0 MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ±10% Note 3	When peripheral functions are stopped		2	4	mA
				When peripheral functions are operating			8	mA
		8.38 MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ±10% Note 3	When peripheral functions are stopped		1.2	2.4	mA
				When peripheral functions are operating			5	mA
			V_{DD} = 3.0 V + 10% Notes 3, 6	When peripheral functions are stopped		0.6	1.2	mA
				When peripheral functions are operating			2.4	mA
		5.00 MHz crystal oscillation HALT mode	V_{DD} = 3.0 V ±10% Note 3	When peripheral functions are stopped		0.4	0.8	mA
				When peripheral functions are operating			1.7	mA
			V_{DD} = 2.0 V ±10% Note 4	When peripheral functions are stopped		0.2	0.4	mA
				When peripheral functions are operating			1.1	mA
	I_{DD3}	32.768 kHz crystal oscillation operating mode Note 5		V_{DD} = 5.0 V ±10%		115	230	μA
				V_{DD} = 3.0 V ±10%		95	190	μA
				V_{DD} = 2.0 V ±10%		75	150	μA
	I_{DD4}	32.768 kHz crystal oscillation HALT mode Note 5		V_{DD} = 5.0 V ±10%		30	60	μA
				V_{DD} = 3.0 V ±10%		6	18	μA
				V_{DD} = 2.0 V ±10%		2	10	μA
	I_{DD5}	XT1 = V_{DD}, STOP mode When feedback resistor is not used		V_{DD} = 5.0 V ±10%		0.1	30	μA
				V_{DD} = 3.0 V ±10%		0.05	10	μA
				V_{DD} = 2.0 V ±10%		0.05	10	μA

Notes 1. Total current through the internal power supply (V_{DD0}, V_{DD1}) (except the current through pull-up resistors of ports).

2. I_{DD1} includes the peripheral operation current.

3. When the processor clock control register (PCC) is cleared to 00H.

4. When PCC is set to 02H.

5. When main system clock operation is stopped.

6. The values show the specifications when V_{DD} = 3.0 to 3.3 V. The value in the TYP. column show the specifications when V_{DD} = 3.0 V.

7. Includes the current through the AV_{DD} pin.

AC Characteristics

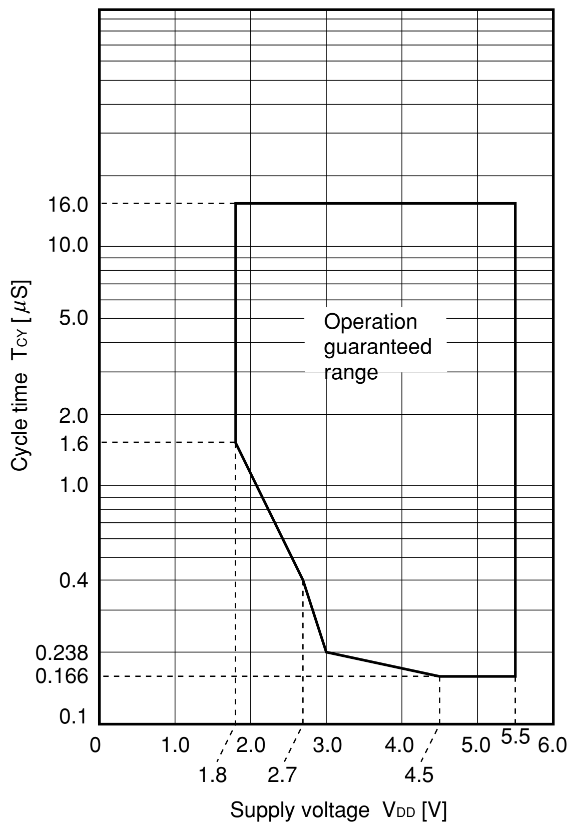
(1) Basic operation (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Cycle time (Min. instruction execution time)	T_{CY}	Operating with main system clock	4.5 V ≤ V_{DD} ≤ 5.5 V	0.166		16	μs
			3.0 V ≤ V_{DD} < 4.5 V	0.238		16	μs
			2.7 V ≤ V_{DD} < 3.0 V	0.4		16	μs
			1.8 V ≤ V_{DD} < 2.7 V	1.6		16	μs
		Operating with subsystem clock		103.9 Note 1	122	125	μs
TI00, TI01 input high-/low-level width	t_{TIH0}, t_{TIL0}	3.0 V ≤ V_{DD} ≤ 5.5 V		$2/f_{sam}$ + 0.1 Note 2			μs
		2.7 V ≤ V_{DD} < 3.0 V		$2/f_{sam}$ + 0.2 Note 2			μs
		1.8 V ≤ V_{DD} < 2.7 V		$2/f_{sam}$ + 0.5 Note 2			μs
TI50, TI51 input frequency	f_{TI5}	2.7 V ≤ V_{DD} ≤ 5.5 V		0		4	MHz
		1.8 V ≤ V_{DD} < 2.7 V		0		275	kHz
TI50, TI51 input high-/low-level width	t_{TIH5}, t_{TIL5}	2.7 V ≤ V_{DD} ≤ 5.5 V		100			ns
		1.8 V ≤ V_{DD} < 2.7 V		1.8			μs
Interrupt request input high-/low-level width	t_{INTH}, t_{INTL}	INTP0 to INTP3, P40 to P47	2.7 V ≤ V_{DD} ≤ 5.5 V	1			μs
			1.8 V ≤ V_{DD} < 2.7 V	2			μs
$\overline{\text{RESET}}$ low-level width	t_{RSL}	2.7 V ≤ V_{DD} ≤ 5.5 V		10			μs
		1.8 V ≤ V_{DD} < 2.7 V		20			μs

Notes 1. Value when the external clock is used. When a crystal resonator is used, it is 114 μs (MIN.).

2. Selection of f_{sam} = f_X, $f_X/4$, $f_X/64$ is possible using bits 0 and 1 (PRM00, PRM01) of prescaler mode register 0 (PRM0). However, if the TI00 valid edge is selected as the count clock, the value becomes f_{sam} = $f_X/8$.

T_{CY} vs. V_{DD} (main system clock operation)



(2) Read/write operation (T_A = –40 to +85°C, V_{DD} = 4.0 to 5.5 V) (1/3)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t_{ASTH}		$0.3t_{CY}$		ns
Address setup time	t_{ADS}		20		ns
Address hold time	t_{ADH}		6		ns
Data input time from address	t_{ADD1}			$(2 + 2n)t_{CY} - 54$	ns
	t_{ADD2}			$(3 + 2n)t_{CY} - 60$	ns
Address output time from $\overline{RD}$↓	t_{RDAD}		0	100	ns
Data input time from $\overline{RD}$↓	t_{RDD1}			$(2 + 2n)t_{CY} - 87$	ns
	t_{RDD2}			$(3 + 2n)t_{CY} - 93$	ns
Read data hold time	t_{RDH}		0		ns
$\overline{RD}$ low-level width	t_{RDL1}		$(1.5 + 2n)t_{CY} - 33$		ns
	t_{RDL2}		$(2.5 + 2n)t_{CY} - 33$		ns
Input time from $\overline{RD}$↓ to $\overline{WAIT}$↓	t_{RDWT1}			$t_{CY} - 43$	ns
	t_{RDWT2}			$t_{CY} - 43$	ns
Input time from $\overline{WR}$↓ to $\overline{WAIT}$↓	t_{WRWT}			$t_{CY} - 25$	ns
$\overline{WAIT}$ low-level width	t_{WTL}		$(0.5 + 2n)t_{CY} + 10$	$(2 + 2n)t_{CY}$	ns
Write data setup time	t_{WDS}		60		ns
Write data hold time	t_{WDH}		6		ns
$\overline{WR}$ low-level width	t_{WRL1}		$(1.5 + 2n)t_{CY} - 15$		ns
Delay time from ASTB↓ to $\overline{RD}$↓	t_{ASTRD}		6		ns
Delay time from ASTB↓ to $\overline{WR}$↓	t_{ASTWR}		$2t_{CY} - 15$		ns
Delay time from $\overline{RD}$↑ to ASTB↑ at external fetch	t_{RDAST}		$0.8t_{CY} - 15$	$1.2t_{CY}$	ns
Address hold time from $\overline{RD}$↑ at external fetch	t_{RDADH}		$0.8t_{CY} - 15$	$1.2t_{CY} + 30$	ns
Write data output time from $\overline{RD}$↑	t_{RDWD}		40		ns
Write data output time from $\overline{WR}$↓	t_{WRWD}		10	60	ns
Address hold time from $\overline{WR}$↑	t_{WRADH}		$0.8t_{CY} - 15$	$1.2t_{CY} + 30$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{RD}$↑	t_{WTRD}		$0.8t_{CY}$	$2.5t_{CY} + 25$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{WR}$↑	t_{WTWR}		$0.8t_{CY}$	$2.5t_{CY} + 25$	ns

Caution **T_{CY} can only be used when the MIN. value is 0.238 μs.**

Remarks **1.** $t_{CY} = T_{CY}/4$

2. n indicates the number of waits.

3. C_L = 100 pF (C_L indicates the load capacitance of the AD0 to AD7, A8 to A15, $\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, and ASTB pins.)

(2) Read/write operation (T_A = –40 to +85°C, V_{DD} = 2.7 to 4.0 V) (2/3)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t_{ASTH}		$0.3t_{CY}$		ns
Address setup time	t_{ADS}		30		ns
Address hold time	t_{ADH}		10		ns
Input time from address to data	t_{ADD1}			$(2 + 2n)t_{CY} - 108$	ns
	t_{ADD2}			$(3 + 2n)t_{CY} - 120$	ns
Output time from $\overline{\text{RD}}$↓ to address	t_{RDAD}		0	200	ns
Input time from $\overline{\text{RD}}$↓ to data	t_{RDD1}			$(2 + 2n)t_{CY} - 148$	ns
	t_{RDD2}			$(3 + 2n)t_{CY} - 162$	ns
Read data hold time	t_{RDH}		0		ns
$\overline{\text{RD}}$ low-level width	t_{RDL1}		$(1.5 + 2n)t_{CY} - 40$		ns
	t_{RDL2}		$(2.5 + 2n)t_{CY} - 40$		ns
Input time from $\overline{\text{RD}}$↓ to $\overline{\text{WAIT}}$↓	t_{RDWT1}			$t_{CY} - 75$	ns
	t_{RDWT2}			$t_{CY} - 60$	ns
Input time from $\overline{\text{WR}}$↓ to $\overline{\text{WAIT}}$↓	t_{WRWT}			$t_{CY} - 50$	ns
$\overline{\text{WAIT}}$ low-level width	t_{WTL}		$(0.5 + 2n)t_{CY} + 10$	$(2 + 2n)t_{CY}$	ns
Write data setup time	t_{WDS}		60		ns
Write data hold time	t_{WDH}		10		ns
$\overline{\text{WR}}$ low-level width	t_{WRL1}		$(1.5 + 2n)t_{CY} - 30$		ns
Delay time from ASTB↓ to $\overline{\text{RD}}$↓	t_{ASTRD}		10		ns
Delay time from ASTB↓ to $\overline{\text{WR}}$↓	t_{ASTWR}		$2t_{CY} - 30$		ns
Delay time from $\overline{\text{RD}}$↑ to ASTB↑ at external fetch	t_{RDAST}		$0.8t_{CY} - 30$	$1.2t_{CY}$	ns
Hold time from $\overline{\text{RD}}$↑ to address at external fetch	t_{RDADH}		$0.8t_{CY} - 30$	$1.2t_{CY} + 60$	ns
Write data output time from $\overline{\text{RD}}$↑	t_{RDWD}		40		ns
Write data output time from $\overline{\text{WR}}$↓	t_{WRWD}		20	120	ns
Hold time from $\overline{\text{WR}}$↑ to address	t_{WRADH}		$0.8t_{CY} - 30$	$1.2t_{CY} + 60$	ns
Delay time from $\overline{\text{WAIT}}$↑ to $\overline{\text{RD}}$↑	t_{WTRD}		$0.5t_{CY}$	$2.5t_{CY} + 50$	ns
Delay time from $\overline{\text{WAIT}}$↑ to $\overline{\text{WR}}$↑	t_{WTWR}		$0.5t_{CY}$	$2.5t_{CY} + 50$	ns

Caution **T_{CY} can only be used when the MIN. value is 0.4 μs.**

Remarks **1.** $t_{CY} = T_{CY}/4$

2. n indicates the number of waits.

3. C_L = 100 pF (C_L indicates the load capacitance of the AD0 to AD7, A8 to A15, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WAIT}}$, and ASTB pins.)

(2) Read/write operation (T_A = −40 to +85°C, V_{DD} = 1.8 to 2.7 V) (3/3)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t_{ASTH}		$0.3t_{CY}$		ns
Address setup time	t_{ADS}		120		ns
Address hold time	t_{ADH}		20		ns
Input time from address to data	t_{ADD1}			$(2 + 2n)t_{CY} - 233$	ns
	t_{ADD2}			$(3 + 2n)t_{CY} - 240$	ns
Output time from $\overline{RD}$↓ to address	t_{RDAD}		0	400	ns
Input time from $\overline{RD}$↓ to data	t_{RDD1}			$(2 + 2n)t_{CY} - 325$	ns
	t_{RDD2}			$(3 + 2n)t_{CY} - 332$	ns
Read data hold time	t_{RDH}		0		ns
$\overline{RD}$ low-level width	t_{RDL1}		$(1.5 + 2n)t_{CY} - 92$		ns
	t_{RDL2}		$(2.5 + 2n)t_{CY} - 92$		ns
Input time from $\overline{RD}$↓ to $\overline{WAIT}$↓	t_{RDWT1}			$t_{CY} - 350$	ns
	t_{RDWT2}			$t_{CY} - 132$	ns
Input time from $\overline{WR}$↓ to $\overline{WAIT}$↓	t_{WRWT}			$t_{CY} - 100$	ns
$\overline{WAIT}$ low-level width	t_{WTL}		$(0.5 + 2n)t_{CY} + 10$	$(2 + 2n)t_{CY}$	ns
Write data setup time	t_{WDS}		60		ns
Write data hold time	t_{WDH}		20		ns
$\overline{WR}$ low-level width	t_{WRL1}		$(1.5 + 2n)t_{CY} - 60$		ns
Delay time from ASTB↓ to $\overline{RD}$↓	t_{ASTRD}		20		ns
Delay time from ASTB↓ to $\overline{WR}$↓	t_{ASTWR}		$2t_{CY} - 60$		ns
Delay time from $\overline{RD}$↑ to ASTB↑ at external fetch	t_{RDAST}		$0.8t_{CY} - 60$	$1.2t_{CY}$	ns
Hold time from $\overline{RD}$↑ to address at external fetch	t_{RDADH}		$0.8t_{CY} - 60$	$1.2t_{CY} + 120$	ns
Write data output time from $\overline{RD}$↑	t_{RDWD}		40		ns
Write data output time from $\overline{WR}$↓	t_{WRWD}		40	240	ns
Hold time from $\overline{WR}$↑ to address	t_{WRADH}		$0.8t_{CY} - 60$	$1.2t_{CY} + 120$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{RD}$↑	t_{WTRD}		$0.5t_{CY}$	$2.5t_{CY} + 100$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{WR}$↑	t_{WTWR}		$0.5t_{CY}$	$2.5t_{CY} + 100$	ns

Caution **T_{CY} can only be used when the MIN. value is 1.6 μs.**

Remarks
1. $t_{CY} = T_{CY}/4$
2. n indicates the number of waits.
3. C_L = 100 pF (C_L indicates the load capacitance of the AD0 to AD7, A8 to A15, $\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, and ASTB pins.)

(3) Serial interface (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (1/2)

(a) 3-wire serial I/O mode ($\overline{\text{SCK3n}}$... Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK3n}}$ cycle time	t_{KCY1}	4.5 V ≤ V_{DD} ≤ 5.5 V	666			ns
		3.0 V ≤ V_{DD} < 4.5 V	954			ns
		2.7 V ≤ V_{DD} < 3.0 V	1600			ns
		1.8 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK3n}}$ high-/low-level width	t_{KH1}, t_{KL1}	3.0 V ≤ V_{DD} ≤ 5.5 V	t_{KCY1}/2 – 50			ns
		1.8 V ≤ V_{DD} < 3.0 V	t_{KCY1}/2 – 100			ns
SI3n setup time (to $\overline{\text{SCK3n}}$↑)	t_{SIK1}	3.0 V ≤ V_{DD} ≤ 5.5 V	100			ns
		2.7 V ≤ V_{DD} < 3.0 V	150			ns
		1.8 V ≤ V_{DD} < 2.7 V	300			ns
SI3n hold time (from $\overline{\text{SCK3n}}$↑)	t_{KSI1}	4.5 V ≤ V_{DD} ≤ 5.5 V	300			ns
		1.8 V ≤ V_{DD} < 4.5 V	400			ns
Delay time from $\overline{\text{SCK3n}}$↓ to SO3n output	t_{KSO1}	C = 100 pF[Note] 4.5 V ≤ V_{DD} ≤ 5.5 V			200	ns
		C = 100 pF[Note] 1.8 V ≤ V_{DD} < 4.5 V			300	ns

Note C is the load capacitance of the $\overline{\text{SCK3n}}$ and SO3n output lines.

(b) 3-wire serial I/O mode ($\overline{\text{SCK3n}}$... External clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK3n}}$ cycle time	t_{KCY2}	4.5 V ≤ V_{DD} ≤ 5.5 V	666			ns
		3.0 V ≤ V_{DD} < 4.5 V	800			ns
		2.7 V ≤ V_{DD} < 3.0 V	1600			ns
		1.8 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK3n}}$ high-/low-level width	t_{KH2}, t_{KL2}	4.5 V ≤ V_{DD} ≤ 5.5 V	333			ns
		3.0 V ≤ V_{DD} < 4.5 V	400			ns
		2.7 V ≤ V_{DD} < 3.0 V	800			ns
		1.8 V ≤ V_{DD} < 2.7 V	1600			ns
SI3n setup time (to $\overline{\text{SCK3n}}$↑)	t_{SIK2}		100			ns
SI3n hold time (from $\overline{\text{SCK3n}}$↑)	t_{KSI2}	4.5 V ≤ V_{DD} ≤ 5.5 V	300			ns
		1.8 V ≤ V_{DD} < 4.5 V	400			ns
Delay time from $\overline{\text{SCK3n}}$↓ to SO3n output	t_{KSO2}	C = 100 pF[Note] 4.5 V ≤ V_{DD} ≤ 5.5 V			200	ns
		C = 100 pF[Note] 1.8 V ≤ V_{DD} < 4.5 V			300	ns

Note C is the load capacitance of the SO3n output line.

Remark n = 0, 1

(3) Serial interface (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V) (2/2)

(c) UART mode (dedicated baud-rate generator output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		4.5 V ≤ V_{DD} ≤ 5.5 V			187500	bps
		3.0 V ≤ V_{DD} < 4.5 V			131031	bps
		2.7 V ≤ V_{DD} < 3.0 V			78125	bps
		1.8 V ≤ V_{DD} < 2.7 V			39063	bps

(d) UART mode (external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK0 cycle time	t_{KCY3}	4.0 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.0 V	1600			ns
		1.8 V ≤ V_{DD} < 2.7 V	3200			ns
ASCK0 high-/low-level width	t_{KH3}, t_{KL3}	4.0 V ≤ V_{DD} ≤ 5.5 V	400			ns
		2.7 V ≤ V_{DD} < 4.0 V	800			ns
		1.8 V ≤ V_{DD} < 2.7 V	1600			ns
Transfer rate		4.0 V ≤ V_{DD} ≤ 5.5 V			39063	bps
		2.7 V ≤ V_{DD} < 4.0 V			19531	bps
		1.8 V ≤ V_{DD} < 2.7 V			9766	bps

(e) UART mode (infrared data transfer mode)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Transfer rate		4.0 V ≤ V_{DD} ≤ 5.5 V		131031	bps
Allowable bit rate error		4.0 V ≤ V_{DD} ≤ 5.5 V		±0.87	%
Output pulse width		4.0 V ≤ V_{DD} ≤ 5.5 V	1.2	0.24/fbr[Note]	μs
Input pulse width		4.0 V ≤ V_{DD} ≤ 5.5 V	4/f_X		μs

Note fbr: Specified baud rate

A/D Converter Characteristics (T_A = –40 to +85°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, AV_{SS} = V_{SS} = 0 V) (1/2)

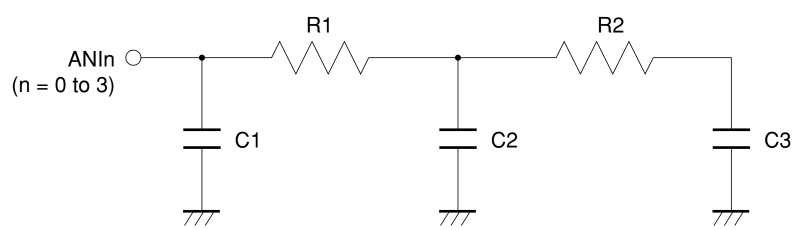
(1) 8-bit A/D converter: μPD780024A Subseries

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error[Note]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
Conversion time	t_{CONV}	4.5 V ≤ AV_{DD} ≤ 5.5 V	12		96	μs
		4.0 V ≤ AV_{DD} < 4.5 V	14		96	μs
		2.7 V ≤ AV_{DD} < 4.0 V	17		96	μs
		1.8 V ≤ AV_{DD} < 2.7 V	28		96	μs
Analog input voltage	V_{AIN}		0		AV_{REF}	V
Reference voltage	AV_{REF}		1.8		AV_{DD}	V
Resistance between AV_{REF} and AV_{SS}	R_{REF}	During A/D converter operation	20	40		kΩ

Note Excludes quantization error (±1/2 LSB). This value is indicated as a ratio (%FSR) to the full-scale value.

Remark The impedance of the analog input pins is shown below.

[Equivalent circuit]



[Parameter value]

(TYP.)

AV_{DD}	R1	R2	C1	C2	C3
2.7 V	12 kΩ	8.0 kΩ	3.0 pF	3.0 pF	2.0 pF
4.5 V	4 kΩ	2.7 kΩ	3.0 pF	1.4 pF	2.0 pF

A/D Converter Characteristics (T_A = –40 to +85°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, AV_{SS} = V_{SS} = 0 V) (2/2)

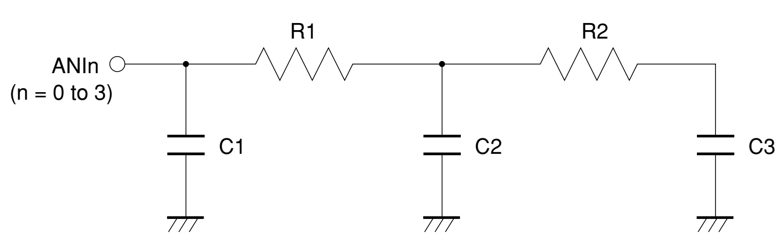
(2) 10-bit A/D converter: μPD780034A Subseries

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			10	10	10	bit
Overall error[Notes 1, 2]		4.0 V ≤ AV_{REF} ≤ 5.5 V		±0.2	±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V		±0.3	±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V		±0.6	±1.2	%FSR
Conversion time	t_{CONV}	4.5 V ≤ AV_{DD} ≤ 5.5 V	12		96	μs
		4.0 V ≤ AV_{DD} < 4.5 V	14		96	μs
		2.7 V ≤ AV_{DD} < 4.0 V	17		96	μs
		1.8 V ≤ AV_{DD} < 2.7 V	28		96	μs
Zero-scale error[Notes 1, 2]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
Full-scale error[Notes 1, 2]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
Integral linearity error[Note 1]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±2.5	LSB
		2.7 V ≤ AV_{REF} < 4.0 V			±4.5	LSB
		1.8 V ≤ AV_{REF} < 2.7 V			±8.5	LSB
Differential linearity error		4.0 V ≤ AV_{REF} ≤ 5.5 V			±1.5	LSB
		2.7 V ≤ AV_{REF} < 4.0 V			±2.0	LSB
		1.8 V ≤ AV_{REF} < 2.7 V			±3.5	LSB
Analog input voltage	V_{AIN}		0		AV_{REF}	V
Reference voltage	AV_{REF}		1.8		AV_{DD}	V
Resistance between AV_{REF} and AV_{SS}	R_{REF}	During A/D converter operation	20	40		kΩ

Notes 1. Excludes quantization error (±1/2 LSB).
2. This value is indicated as a ratio to the full-scale value.

Remark The impedance of the analog input pins is shown below.

[Equivalent circuit]



[Parameter value]

(TYP.)

AV_{DD}	R1	R2	C1	C2	C3
2.7 V	12 kΩ	8.0 kΩ	3.0 pF	3.0 pF	2.0 pF
4.5 V	4 kΩ	2.7 kΩ	3.0 pF	1.4 pF	2.0 pF

Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (T_A = –40 to +85°C)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention power supply voltage	V_{DDDR}		1.6		5.5	V
Data retention power supply current	I_{DDDR}	V_{DDDR} = 1.6 V (Subsystem clock unused (XT1 = V_{DD}) and feedback resistor disconnected)		0.1	30	μA
Release signal set time	t_{SREL}		0			μs
Oscillation stabilization wait time	t_{WAIT}	Release by $\overline{\text{RESET}}$		$2^{17}/f_X$		s
		Release by interrupt request		**Note**		s

Note Selection of $2^{12}/f_X$ and $2^{14}/f_X$ to $2^{17}/f_X$ is possible using bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register (OSTS).

Flash Memory Programming Characteristics (1/2)
(T_A = +10 to +40°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, V_{SS} = AV_{SS} = 0 V)

- **μPD78F0034B, 78F0034B(A)**

(a) Write erase characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Operating frequency	f_X	4.5 V ≤ V_{DD} ≤ 5.5 V	1.0		10.0	MHz
		3.0 V ≤ V_{DD} < 4.5 V	1.0		8.38	MHz
		2.7 V ≤ V_{DD} < 3.0 V	1.0		5.00	MHz
		1.8 V ≤ V_{DD} < 2.7 V	1.0		1.25	MHz
V_{PP} supply voltage	V_{PP2}	During flash memory programming	9.7	10.0	10.3	V
V_{DD} supply current[Note 1]	I_{DD}	When V_{PP} = V_{PP2}, 10 MHz crystal oscillation operating mode, V_{DD} = 5.0 V±10%			30	mA
		When V_{PP} = V_{PP2}, 8.38 MHz crystal oscillation operating mode, V_{DD} = 5.0 V±10%			24	mA
		When V_{PP} = V_{PP2}, 8.38 MHz crystal oscillation operating mode, V_{DD} = 3.0 V±10%			17	mA
V_{PP} supply current	I_{PP}	When V_{PP} = V_{PP2}			100	mA
Step erase time[Note 2]	T_{er}		0.199	0.2	0.201	s
Overall erase time[Note 3]	T_{era}	When step erase time = 0.2 s			20	s/chip
Writeback time[Note 4]	T_{wb}		49.4	50	50.6	ms
Number of writebacks per writeback command[Note 5]	C_{wb}	When writeback time = 50 ms			60	Times
Number of erases/writebacks	C_{erwb}				16	Times
Step write time[Note 6]	T_{wr}		48	50	52	μs
Overall write time per word[Note 7]	T_{wrw}	When step write time = 50 μs (1 word = 1 byte)	48		520	μs
Number of rewrites per chip[Note 8]	C_{erwr}	1 erase + 1 write after erase = 1 rewrite			20	Times/area

Notes 1. AV_{DD} current and port current (current that flows through the internal pull-up resistor) are not included.

2. The recommended setting value of the step erase time is 0.2 s.

3. The prewrite time before erasure and the erase verify time (writeback time) are not included.

4. The recommended setting value of the writeback time is 50 ms.

5. Writeback is executed once by the issuance of the writeback command. Therefore, the number of retries must be the maximum value minus the number of commands issued.

6. The recommended setting value of the step write time is 50 μs.

7. The actual write time per word is 100 μs longer. The internal verify time during or after a write is not included.

8. When a product is first written after shipment, "erase → write" and "write only" are both taken as one rewrite.

Example: P: Write, E: Erase

Shipped product →P→E→P→E→P: 3 rewrites

Shipped product →E→P→E→P→E→P: 3 rewrites

Flash Memory Programming Characteristics (2/2)
(T_A = +10 to +40°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, V_{SS} = AV_{SS} = 0 V)

• μPD78F0034B, 78F0034B(A)

(b) Serial write operation characteristics

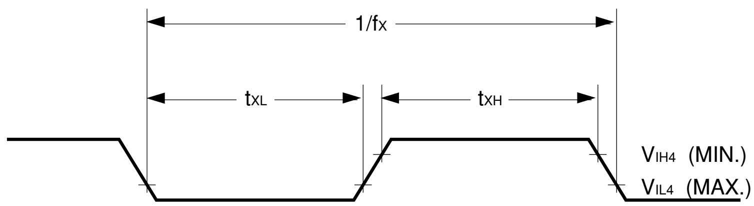
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Set time from V_{DD}↑ to V_{PP}↑	t_{DP}		10			μs
Release time from V_{PP}↑ to $\overline{RESET}$↑	t_{PR}		1.0			μs
V_{PP} pulse input start time from $\overline{RESET}$↑	t_{RP}		1.0			μs
V_{PP} pulse high-/low-level width	t_{PW}		8.0			μs
V_{PP} pulse input end time from $\overline{RESET}$↑	t_{RPE}				20	ms
V_{PP} pulse low-level input voltage	V_{PPL}		$0.8V_{DD}$		$1.2V_{DD}$	V
V_{PP} pulse high-level input voltage	V_{PPH}		9.7	10.0	10.3	V

Timing Charts

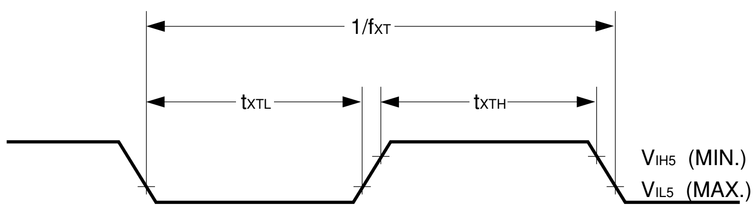
AC Timing Test Points (excluding X1, XT1 input)

Clock Timing

X1 input

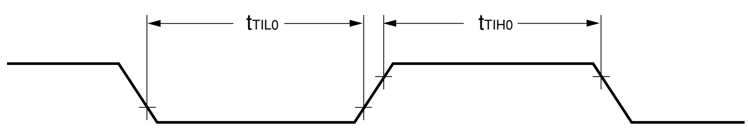


XT1 input

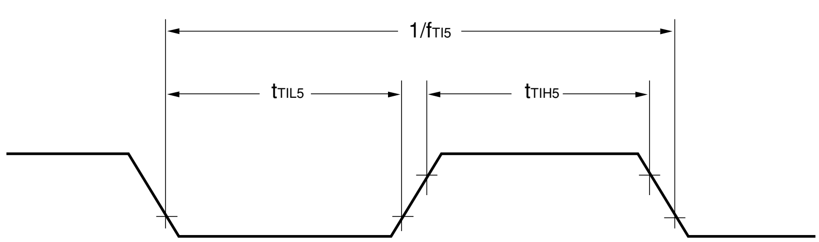


TI Timing

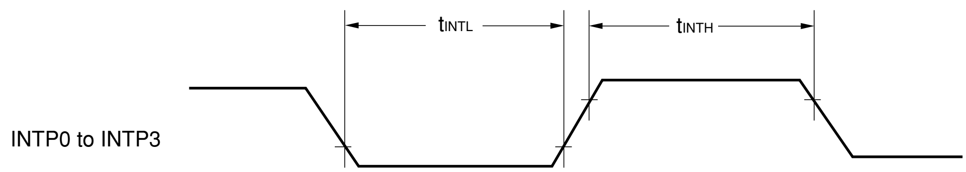
TI00, TI01



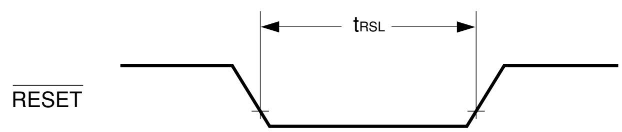
TI50, TI51



Interrupt Request Input Timing

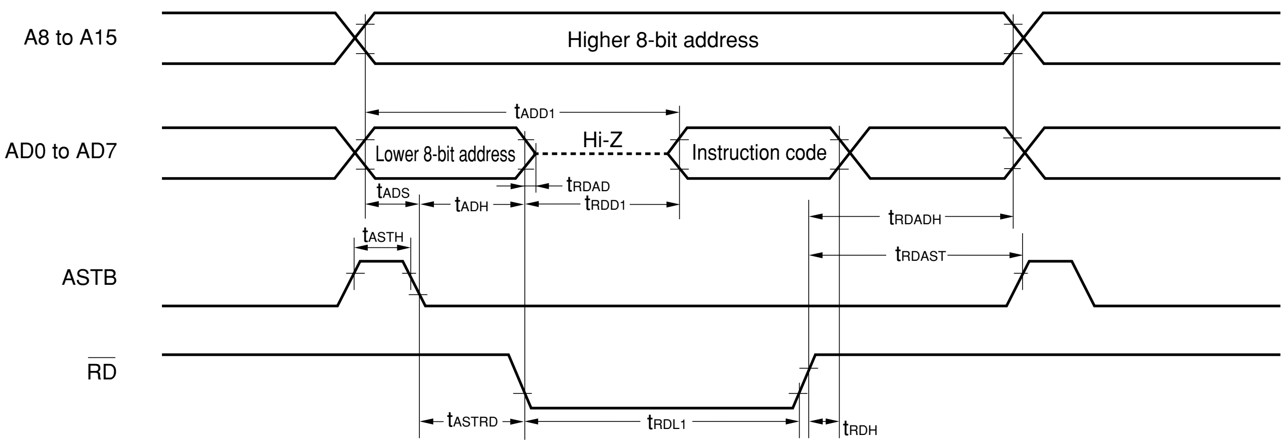


$\overline{\text{RESET}}$ Input Timing

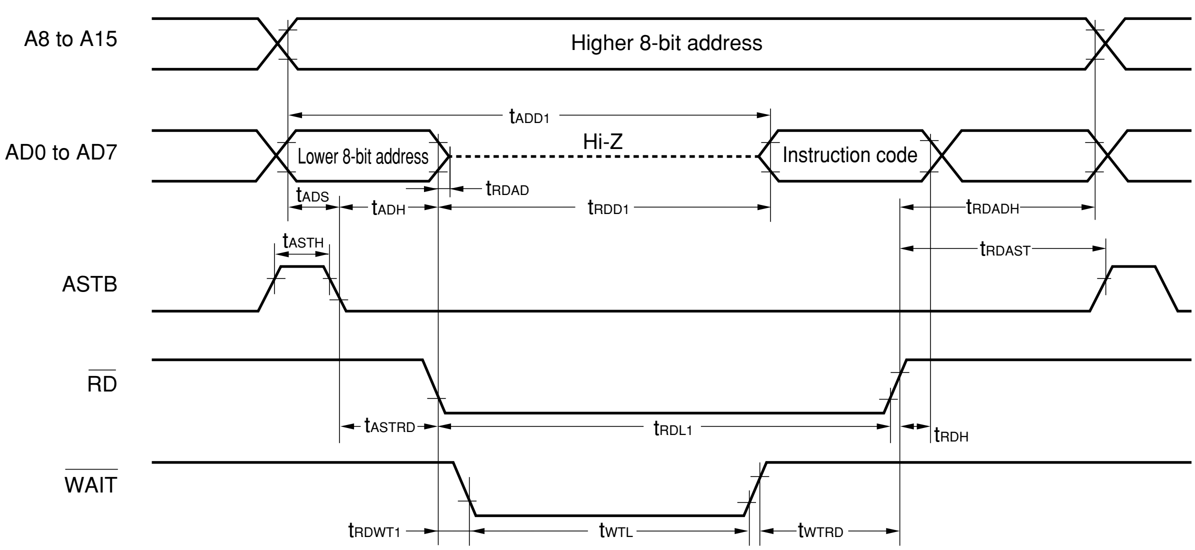


Read/Write Operation

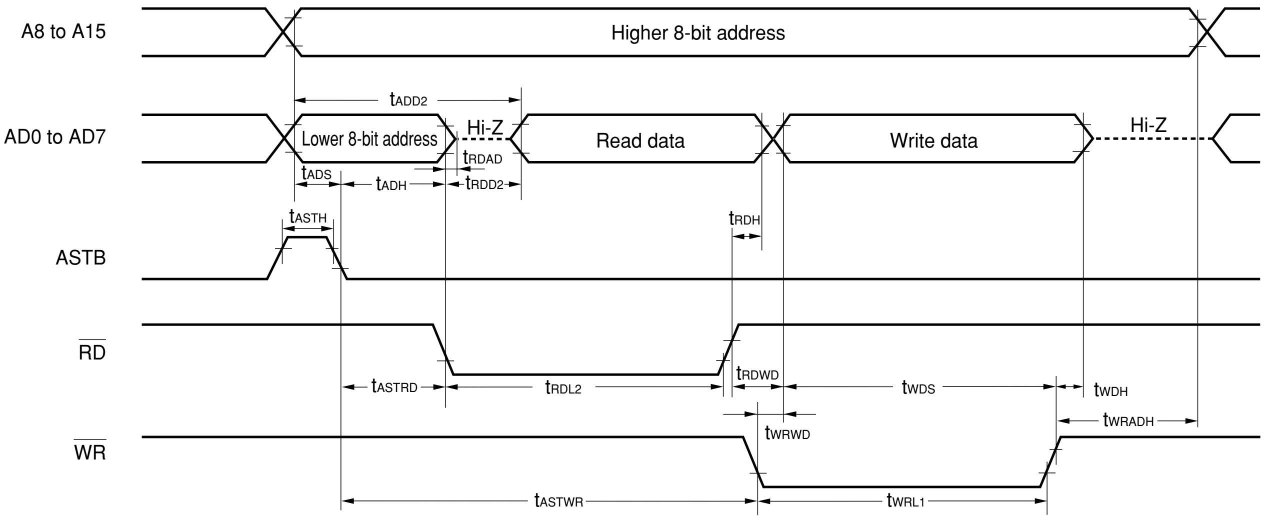
External fetch (no wait):



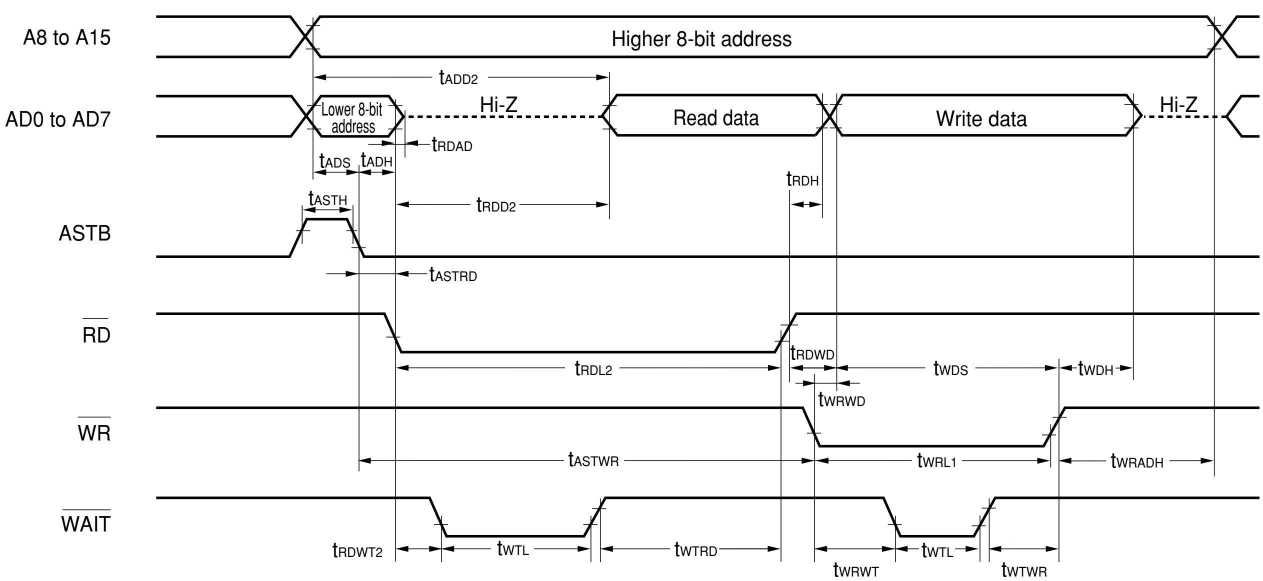
External fetch (wait insertion):



External data access (no wait):

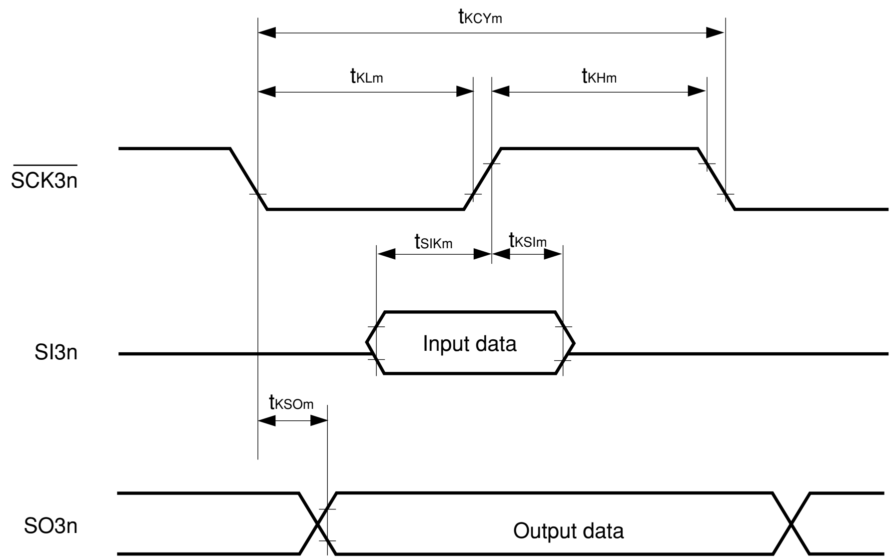


External data access (wait insertion):



Serial Transfer Timing

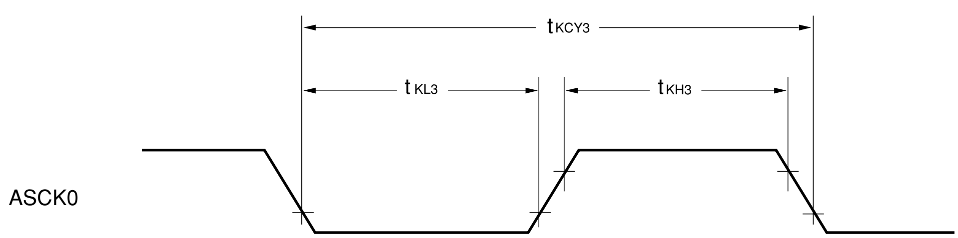
3-wire serial I/O mode:



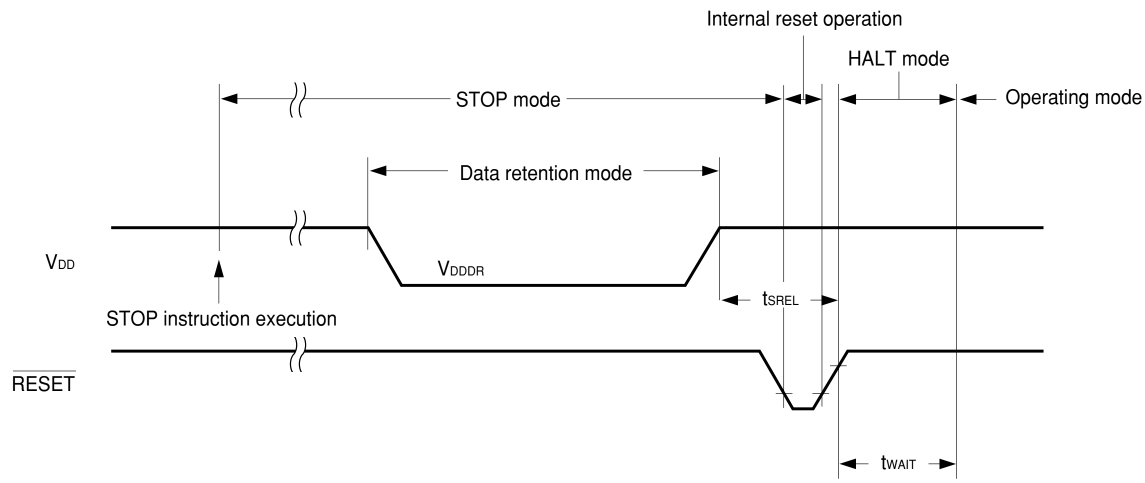
Remarks 1. m = 1, 2

2. n = 0, 1

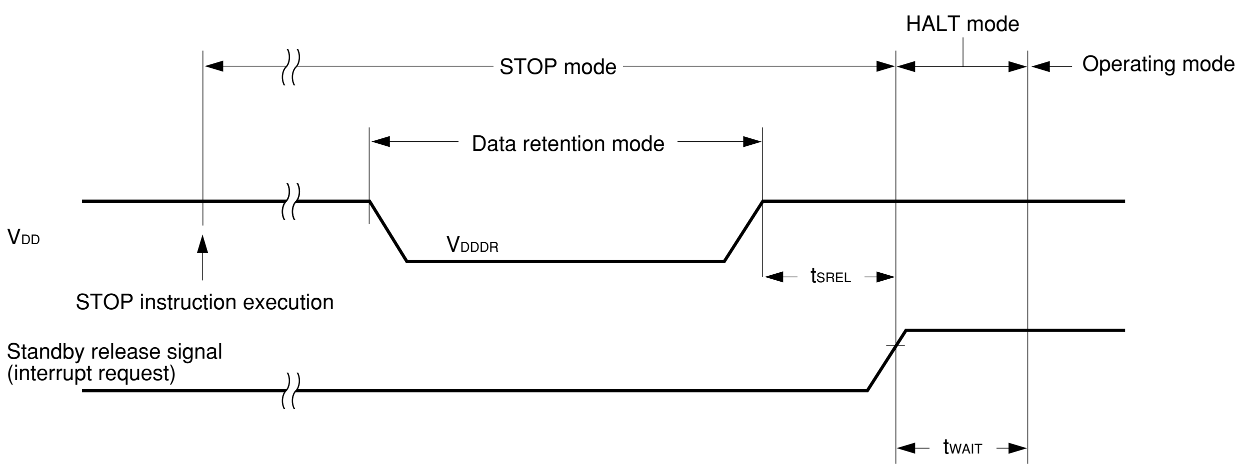
UART mode (external clock input):



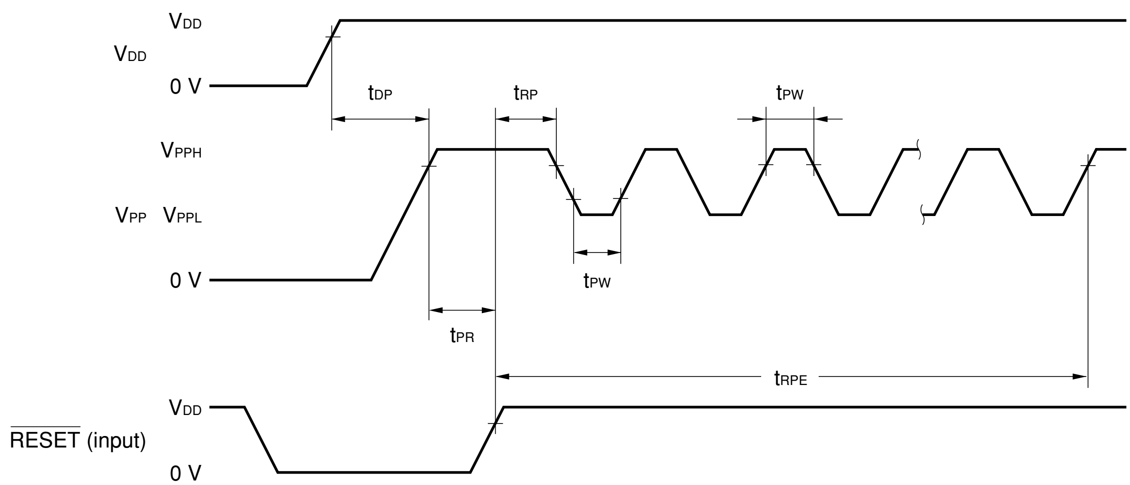
Data Retention Timing (STOP Mode Release by $\overline{\text{RESET}}$)



Data Retention Timing (Standby Release Signal: STOP Mode Release by Interrupt Request Signal)



Flash Memory Write Mode Set Timing



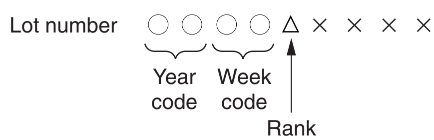
★

CHAPTER 26 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS: f_X = 1.0 TO 8.38 MHz)

Target products

- μPD780021A, 780022A, 780023A, 780024A, 780031A, 780032A, 780033A, 780034A, 780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), 780034A(A) for which orders were received before November 30, 2001 (Products with rank[Note] K, E, P, X)
- μPD780021AY, 780022AY, 780023AY, 780024AY, 780031AY, 780032AY, 780033AY, 780034AY, 780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A), 780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A)
- μPD78F0034A, 78F0034AY, 78F0034BY, 78F0034BY(A)

Note The rank is indicated by the 5th digit from the left in the lot number marked on the package.



Caution The μPD780021AY(A), 780023AY(A), 780024AY(A), 780031AY(A), 780032AY(A), 780033AY(A), and 780034AY(A) are under development.

The electrical specifications of the above products are target values (reference values only); mass-produced products do not necessarily satisfy these ratings.

Absolute Maximum Ratings (T_A = 25°C) (1/2)

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V_{DD}			–0.3 to +6.5	V
	V_{PP}	Flash memory version only, **Note 2**		–0.3 to +10.5	V
	AV_{DD}			–0.3 to V_{DD} + 0.3 Note 1	V
	AV_{REF}			–0.3 to V_{DD} + 0.3 Note 1	V
	AV_{SS}			–0.3 to +0.3	V
Input voltage	V_{I1}	P00 to P03, P10 to P17, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75, X1, X2, XT1, XT2, $\overline{\text{RESET}}$		–0.3 to V_{DD} + 0.3 Note 1	V
	V_{I2}	P30 to P33	N-ch open drain	–0.3 to +6.5	V
			On-chip pull-up resistor	–0.3 to V_{DD} + 0.3 Note 1	V
Output voltage	V_O			–0.3 to V_{DD} + 0.3 Note 1	V
Analog input voltage	V_{AN}	P10 to P17	Analog input pin	AV_{SS} – 0.3 to AV_{REF} + 0.3 Note 1 and –0.3 to V_{DD} + 0.3 Note 1	V

Notes 1. 6.5 V or below

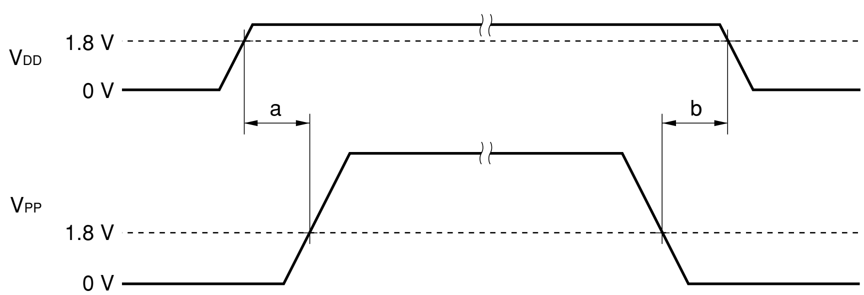
2. Make sure that the following conditions of the V_{PP} voltage application timing are satisfied when the flash memory is written.

- **When supply voltage rises**

  V_{PP} must exceed V_{DD} 10 μs or more after V_{DD} has reached the lower-limit value (1.8 V) of the operating voltage range (see a in the figure below).

- **When supply voltage drops**

  V_{DD} must be lowered 10 μs or more after V_{PP} falls below the lower-limit value (1.8 V) of the operating voltage range of V_{DD} (see b in the figure below).



Caution Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

Absolute Maximum Ratings (T_A = 25°C) (2/2)

Parameter	Symbol	Conditions	Ratings	Unit
Output current, high	I_{OH}	Per pin	–10	mA
		Total for P00 to P03, P40 to P47, P50 to P57, P64 to P67, P70 to P75	–15	mA
		Total for P20 to P25, P30 to P36	–15	mA
Output current, low	I_{OL}	Per pin for P00 to P03, P20 to P25, P34 to P36, P40 to P47, P64 to P67, P70 to P75	20	mA
		Per pin for P30 to P33, P50 to P57	30	mA
		Total for P00 to P03, P40 to P47, P64 to P67, P70 to P75	50	mA
		Total for P20 to P25	20	mA
		Total for P30 to P36	100	mA
		Total for P50 to P57	100	mA
Operating ambient temperature	T_A	During normal operation	–40 to +85	°C
		During flash memory programming	–10 to +80	°C
Storage temperature	T_{stg}	Mask ROM version	–65 to +150	°C
		Flash memory version	–40 to +125	°C

Caution Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

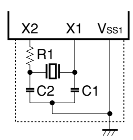
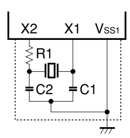
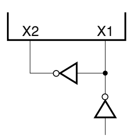
Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

Capacitance (T_A = 25°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.	P00 to P03, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75			15	pF
			P30 to P33			20	pF

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

Main System Clock Oscillator Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f_X)Note 1	4.0 V ≤ V_{DD} ≤ 5.5 V	1.0		8.38	MHz
			1.8 V ≤ V_{DD} < 4.0 V	1.0		5.0	
		Oscillation stabilization timeNote 2	After V_{DD} reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency (f_X)Note 1	4.0 V ≤ V_{DD} ≤ 5.5 V	1.0		8.38	MHz
			1.8 V ≤ V_{DD} < 4.0 V	1.0		5.0	
		Oscillation stabilization timeNote 2	4.0 V ≤ V_{DD} ≤ 5.5 V			10	ms
			1.8 V ≤ V_{DD} < 4.0 V			30	
External clock		X1 input frequency (f_X)Note 1	4.0 V ≤ V_{DD} ≤ 5.5 V	1.0		8.38	MHz
			1.8 V ≤ V_{DD} < 4.0 V	1.0		5.0	
		X1 input high-/low-level width (t_{XH}, t_{XL})	4.0 V ≤ V_{DD} ≤ 5.5 V	50		500	ns
			1.8 V ≤ V_{DD} < 4.0 V	85		500	

Notes 1. Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

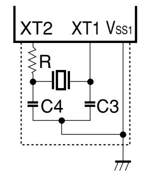
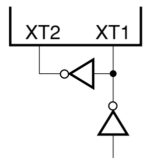
2. Time required to stabilize oscillation after reset or STOP mode release.

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.

Subsystem Clock Oscillator Characteristics (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Crystal resonator		Oscillation frequency (f_{XT})Note 1		32	32.768	35	kHz
		Oscillation stabilization timeNote 2	4.0 V ≤ V_{DD} ≤ 5.5 V		1.2	2	s
			1.8 V ≤ V_{DD} < 4.0 V			10	
External clock		XT1 input frequency (f_{XT})Note 1		32		38.5	kHz
		XT1 input high-/low-level width (t_{XTH}, t_{XTL})		12		15	μs

Notes 1. Indicates only oscillator characteristics. Refer to **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 figure to avoid an adverse effect from wiring capacitance.

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines.**
- **Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as V_{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 power 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 of the subsystem clock, customers are required to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

Recommended Oscillator Constant (1/2)

To use the μPD78F0034BY or 78F0034BY(A), for the resonator selection and oscillator constant, customers are required to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

(1) Mask ROM versions of μPD780024A, 780034A Subseries (conventional product) and μPD780024AY, 780034AY Subseries
Main system clock: Ceramic resonator (T_A = −40 to +85°C)

Manufacturer	Part Number	Frequency (MHz)	Recommended Circuit Constant			Oscillation Voltage Range	
			C1 (pF)	C2 (pF)	R1 (kΩ)	MIN. (V)	MAX. (V)
Murata Mfg. Co., Ltd.	CSBFB1M00J58	1.00	100	100	2.2	1.8	5.5
	CSBLA1M00J58	1.00	100	100	2.2	1.8	5.5
	CSTCC2M00G56	2.00	On-chip	On-chip	0	1.8	5.5
	CSTLS2M00G56	2.00	On-chip	On-chip	0	1.8	5.5
	CSTCC3M58G53	3.58	On-chip	On-chip	0	1.8	5.5
	CSTLS3M58G53	3.58	On-chip	On-chip	0	1.8	5.5
	CSTCR4M00G53	4.00	On-chip	On-chip	0	1.8	5.5
	CSTLS4M00G53	4.00	On-chip	On-chip	0	1.8	5.5
	CSTCR4M19G53	4.19	On-chip	On-chip	0	1.8	5.5
	CSTLS4M19G53	4.19	On-chip	On-chip	0	1.8	5.5
	CSTCR4M91G53	4.91	On-chip	On-chip	0	1.8	5.5
	CSTLS4M91G53	4.91	On-chip	On-chip	0	1.8	5.5
	CSTCR5M00G53	5.00	On-chip	On-chip	0	1.8	5.5
	CSTLS5M00G53	5.00	On-chip	On-chip	0	1.8	5.5
	CSTCE8M00G52	8.00	On-chip	On-chip	0	2.7	5.5
	CSTLS8M00G53	8.00	On-chip	On-chip	0	2.7	5.5
	CSTCE8M38G52	8.38	On-chip	On-chip	0	4.0	5.5
	CSTLS8M38G53	8.38	On-chip	On-chip	0	4.0	5.5
TDK	CCR3.58MC3	3.58	On-chip	On-chip	0	1.8	5.5
	CCR4.19MC3	4.19	On-chip	On-chip	0	1.8	5.5
	CCR5.0MC3	5.00	On-chip	On-chip	0	1.8	5.5
	CCR8.0MC5	8.00	On-chip	On-chip	0	2.7	5.5
	CCR8.38MC5	8.38	On-chip	On-chip	0	4.0	5.5

Caution The oscillator constant is a reference value based on evaluation in specific environments by the resonator manufacturer. If the oscillator characteristics need to be optimized in the actual application, request the resonator manufacturer for evaluation on the implementation circuit. Note that the oscillation voltage and oscillation frequency merely indicate the characteristics of the oscillator. Use the internal operation conditions of the μPD780024A, 780024AY, 780034A, 780034AY Subseries within the specifications of the DC and AC characteristics.

Recommended Oscillator Constant (2/2)

(2) μPD78F0034A, 78F0034AY
Main system clock: Ceramic resonator (T_A = −40 to +85°C)

Manufacturer	Part Number	Frequency (MHz)	Recommended Circuit Constant			Oscillation Voltage Range	
			C1 (pF)	C2 (pF)	R1 (kΩ)	MIN. (V)	MAX. (V)
Murata Mfg. Co., Ltd.	CSBFB1M00J58	1.00	100	100	2.2	1.9	5.5
	CSBLA1M00J58	1.00	100	100	2.2	1.9	5.5
	CSTCC2M00G56	2.00	On-chip	On-chip	0	1.8	5.5
	CSTLS2M00G56	2.00	On-chip	On-chip	0	1.8	5.5
	CSTCC3M58G53	3.58	On-chip	On-chip	0	1.8	5.5
	CSTLS3M58G53	3.58	On-chip	On-chip	0	1.8	5.5
	CSTCR4M00G53	4.00	On-chip	On-chip	0	1.8	5.5
	CSTLS4M00G53	4.00	On-chip	On-chip	0	1.8	5.5
	CSTCR4M19G53	4.19	On-chip	On-chip	0	1.8	5.5
	CSTLS4M19G53	4.19	On-chip	On-chip	0	1.8	5.5
	CSTCR4M91G53	4.91	On-chip	On-chip	0	1.8	5.5
	CSTLS4M91G53	4.91	On-chip	On-chip	0	1.8	5.5
	CSTCR5M00G53	5.00	On-chip	On-chip	0	1.8	5.5
	CSTLS5M00G53	5.00	On-chip	On-chip	0	1.8	5.5
	CSTCE8M00G52	8.00	On-chip	On-chip	0	2.7	5.5
	CSTLS8M00G53	8.00	On-chip	On-chip	0	2.7	5.5
	CSTLS8M00G53093	8.00	On-chip	On-chip	0	2.7	5.5
	CSTCE8M38G52	8.38	On-chip	On-chip	0	4.0	5.5
	CSTLS8M38G53	8.38	On-chip	On-chip	0	4.0	5.5
	CSTLS8M38G53093	8.38	On-chip	On-chip	0	4.0	5.5
TDK	CCR3.58MC3	3.58	On-chip	On-chip	0	1.8	5.5
	CCR4.19MC3	4.19	On-chip	On-chip	0	1.8	5.5
	CCR5.0MC3	5.00	On-chip	On-chip	0	1.8	5.5
	CCR8.0MC5	8.00	On-chip	On-chip	0	2.7	5.5
	CCR8.38MC5	8.38	On-chip	On-chip	0	4.0	5.5

Caution The oscillator constant is a reference value based on evaluation in specific environments by the resonator manufacturer. If the oscillator characteristics need to be optimized in the actual application, request the resonator manufacturer for evaluation on the implementation circuit. Note that the oscillation voltage and oscillation frequency merely indicate the characteristics of the oscillator. Use the internal operation conditions of the μPD780024A, 780024AY, 780034A, 780034AY Subseries within the specifications of the DC and AC characteristics.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (1/4)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output current, high	I_{OH}	Per pin				–1	mA
		All pins				–15	mA
Output current, low	I_{OL}	Per pin for P00 to P03, P20 to P25, P34 to P36, P40 to P47, P64 to P67, P70 to P75				10	mA
		Per pin for P30 to P33, P50 to P57				15	mA
		Total for P00 to P03, P40 to P47, P64 to P67, P70 to P75				20	mA
		Total for P20 to P25				10	mA
		Total for P30 to P36				70	mA
		Total for P50 to P57				70	mA
Input voltage, high	V_{IH1}	P10 to P17, P21, P24, P35, P40 to P47, P50 to P57, P64 to P67, P74, P75	2.7 V ≤ V_{DD} ≤ 5.5 V	0.7V_{DD}		V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	0.8V_{DD}		V_{DD}	V
	V_{IH2}	P00 to P03, P20, P22, P23, P25, P34, P36, P70 to P73, $\overline{\text{RESET}}$	2.7 V ≤ V_{DD} ≤ 5.5 V	0.8V_{DD}		V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	0.85V_{DD}		V_{DD}	V
	V_{IH3}	P30 to P33 (N-ch open-drain)	2.7 V ≤ V_{DD} ≤ 5.5 V	0.7V_{DD}		5.5	V
			1.8 V ≤ V_{DD} < 2.7 V	0.8V_{DD}		5.5	V
	V_{IH4}	X1, X2	2.7 V ≤ V_{DD} ≤ 5.5 V	V_{DD} – 0.5		V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	V_{DD} – 0.2		V_{DD}	V
	V_{IH5}	XT1, XT2	4.0 V ≤ V_{DD} ≤ 5.5 V	0.8V_{DD}		V_{DD}	V
			1.8 V ≤ V_{DD} < 4.0 V	0.9V_{DD}		V_{DD}	V
Input voltage, low	V_{IL1}	P10 to P17, P21, P24, P35, P40 to P47, P50 to P57, P64 to P67, P74, P75	2.7 V ≤ V_{DD} ≤ 5.5 V	0		0.3V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	0		0.2V_{DD}	V
	V_{IL2}	P00 to P03, P20, P22, P23, P25, P34, P36, P70 to P73, $\overline{\text{RESET}}$	2.7 V ≤ V_{DD} ≤ 5.5 V	0		0.2V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	0		0.15V_{DD}	V
	V_{IL3}	P30 to P33	4.0 V ≤ V_{DD} ≤ 5.5 V	0		0.3V_{DD}	V
			2.7 V ≤ V_{DD} < 4.0 V	0		0.2V_{DD}	V
			1.8 V ≤ V_{DD} < 2.7 V	0		0.1V_{DD}	V
	V_{IL4}	X1, X2	2.7 V ≤ V_{DD} ≤ 5.5 V	0		0.4	V
			1.8 V ≤ V_{DD} < 2.7 V	0		0.2	V
	V_{IL5}	XT1, XT2	4.0 V ≤ V_{DD} ≤ 5.5 V	0		0.2V_{DD}	V
			1.8 V ≤ V_{DD} < 4.0 V	0		0.1V_{DD}	V
Output voltage, high	V_{OH1}	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OH} = –1 mA		V_{DD} – 1.0		V_{DD}	V
		1.8 V ≤ V_{DD} < 4.0 V, I_{OH} = –100 μA		V_{DD} – 0.5		V_{DD}	V
Output voltage, low	V_{OL1}	P30 to P33	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OL} = 15 mA			2.0	V
	V_{OL2}	P50 to P57			0.4	2.0	V
	V_{OL3}	P00 to P03, P20 to P25, P34 to P36, P40 to P47, P64 to P67, P70 to P75	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OL} = 1.6 mA			0.4	V
	V_{OL4}	I_{OL} = 400 μA				0.5	V

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (2/4)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	I_{LIH1}	$V_{IN} = V_{DD}$	P00 to P03, P10 to P17, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75, $\overline{RESET}$			3	μA
	I_{LIH2}		X1, X2, XT1, XT2			20	μA
	I_{LIH3}	V_{IN} = 5.5 V	P30 to P33			3	μA
Input leakage current, low	I_{LIL1}	V_{IN} = 0 V	P00 to P03, P10 to P17, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75, $\overline{RESET}$			–3	μA
	I_{LIL2}		X1, X2, XT1, XT2			–20	μA
	I_{LIL3}		P30 to P33			–3	μA
Output leakage current, high	I_{LOH}	$V_{OUT} = V_{DD}$				3	μA
Output leakage current, low	I_{LOL}	V_{OUT} = 0 V				–3	μA
Mask option pull-up resistance (mask ROM version only)	R_1	V_{IN} = 0 V, P30, P31, P32**Note**, P33**Note**		15	30	90	kΩ
Software pull-up resistance	R_2	V_{IN} = 0 V, P00 to P03, P20 to P25, P34 to P36, P40 to P47, P50 to P57, P64 to P67, P70 to P75		15	30	90	kΩ

Note μPD780024A, 780034A Subseries only

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V) (3/4)

(1) Mask ROM versions of μPD780024A, 780034A Subseries (conventional product) and μPD780024AY, 780034AY Subseries

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Power supply current[Note 1]	I_{DD1}[Note 2]	8.38 MHz crystal oscillation operating mode	V_{DD} = 5.0 V ±10%[Note 3]	When A/D converter is stopped		5.5	11	mA
				When A/D converter is operating[Note 6]		6.5	13	mA
		5.00 MHz crystal oscillation operating mode	V_{DD} = 3.0 V ±10%[Note 3]	When A/D converter is stopped		2	4	mA
				When A/D converter is operating[Note 6]		3	6	mA
			V_{DD} = 2.0 V ±10%[Note 4]	When A/D converter is stopped		0.4	1.5	mA
				When A/D converter is operating[Note 6]		1.4	4.2	mA
	I_{DD2}	8.38 MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ±10%[Note 3]	When peripheral functions are stopped		1.1	2.2	mA
				When peripheral functions are operating			4.7	mA
		5.00 MHz crystal oscillation HALT mode	V_{DD} = 3.0 V ±10%[Note 3]	When peripheral functions are stopped		0.35	0.7	mA
				When peripheral functions are operating			1.7	mA
			V_{DD} = 2.0 V ±10%[Note 4]	When peripheral functions are stopped		0.15	0.4	mA
				When peripheral functions are operating			1.1	mA
	I_{DD3}	32.768 kHz crystal oscillation operating mode[Note 5]		V_{DD} = 5.0 V ±10%		40	80	μA
				V_{DD} = 3.0 V ±10%		20	40	μA
				V_{DD} = 2.0 V ±10%		10	20	μA
	I_{DD4}	32.768 kHz crystal oscillation HALT mode[Note 5]		V_{DD} = 5.0 V ±10%		30	60	μA
				V_{DD} = 3.0 V ±10%		6	18	μA
				V_{DD} = 2.0 V ±10%		2	10	μA
	I_{DD5}	XT1 = V_{DD}, STOP mode When feedback resistor is not used		V_{DD} = 5.0 V ±10%		0.1	30	μA
				V_{DD} = 3.0 V ±10%		0.05	10	μA
				V_{DD} = 2.0 V ±10%		0.05	10	μA

Notes 1. Total current through the internal power supply (V_{DD0}, V_{DD1}) (except the current through pull-up resistors of ports).

2. I_{DD1} includes the peripheral operation current.

3. When the processor clock control register (PCC) is cleared to 00H.

4. When PCC is set to 02H.

5. When main system clock operation is stopped.

6. Includes the current through the AV_{DD} pin.

DC Characteristics (T_A = −40 to +85°C, V_{DD} = 1.8 to 5.5 V) (4/4)

(2) μPD78F0034A, 78F0034AY, 78F0034BY, 78F0034BY(A)

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Power supply current Note 1	I_{DD1} Note 2	8.38 MHz crystal oscillation operating mode	V_{DD} = 5.0 V ±10% Note 3	When A/D converter is stopped		10.5	21	mA
				When A/D converter is operating Note 6		11.5	23	mA
		5.00 MHz crystal oscillation operating mode	V_{DD} = 3.0 V ±10% Note 3	When A/D converter is stopped		4.5	9	mA
				When A/D converter is operating Note 6		5.5	11	mA
			V_{DD} = 2.0 V ±10% Note 4	When A/D converter is stopped		1	2	mA
				When A/D converter is operating Note 6		2	6	mA
	I_{DD2}	8.38 MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ±10% Note 3	When peripheral functions are stopped		1.2	2.4	mA
				When peripheral functions are operating			5	mA
		5.00 MHz crystal oscillation HALT mode	V_{DD} = 3.0 V ±10% Note 3	When peripheral functions are stopped		0.4	0.8	mA
				When peripheral functions are operating			1.7	mA
			V_{DD} = 2.0 V ±10% Note 4	When peripheral functions are stopped		0.2	0.4	mA
				When peripheral functions are operating			1.1	mA
	I_{DD3}	32.768 kHz crystal oscillation operating mode Note 5		V_{DD} = 5.0 V ±10% Note 2		115	230	μA
				V_{DD} = 3.0 V ±10% Note 2		95	190	μA
				V_{DD} = 2.0 V ±10% Note 3		75	150	μA
	I_{DD4}	32.768 kHz crystal oscillation HALT mode Note 5		V_{DD} = 5.0 V ±10% Note 2		30	60	μA
				V_{DD} = 3.0 V ±10% Note 2		6	18	μA
				V_{DD} = 2.0 V ±10% Note 3		2	10	μA
	I_{DD5}	XT1 = V_{DD}, STOP mode When feedback resistor is not used		V_{DD} = 5.0 V ±10% Note 2		0.1	30	μA
				V_{DD} = 3.0 V ±10% Note 2		0.05	10	μA
				V_{DD} = 2.0 V ±10% Note 3		0.05	10	μA

Notes 1. Total current through the internal power supply (V_{DD0}, V_{DD1}) (except the current through pull-up resistors of ports).

2. I_{DD1} includes the peripheral operation current.

3. When the processor clock control register (PCC) is cleared to 00H.

4. When PCC is set to 02H.

5. When main system clock operation is stopped.

6. Includes the current through the AV_{DD} pin.

AC Characteristics

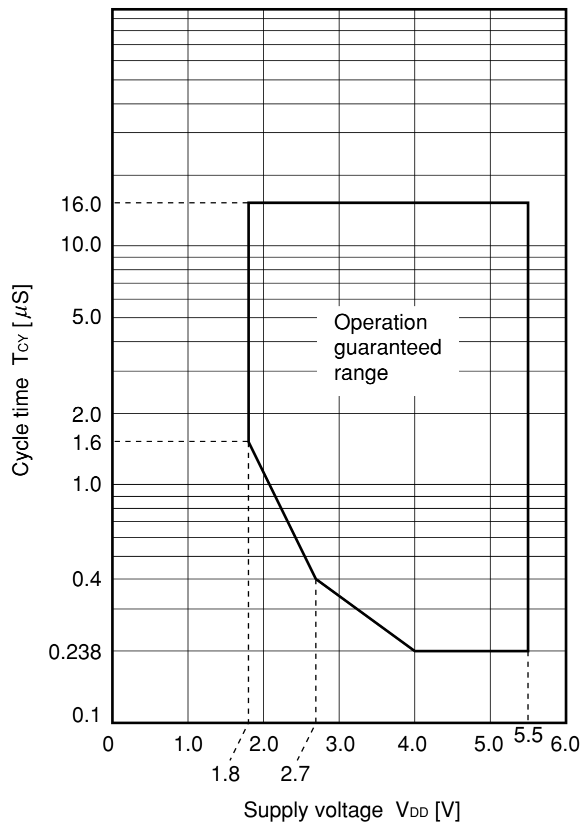
(1) Basic operation (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Cycle time (Min. instruction execution time)	T_{CY}	Operating with main system clock	4.0 V ≤ V_{DD} ≤ 5.5 V	0.238		16	μs
			2.7 V ≤ V_{DD} < 4.0 V	0.4		16	μs
			1.8 V ≤ V_{DD} < 2.7 V	1.6		16	μs
		Operating with subsystem clock		103.9 Note 1	122	125	μs
TI00, TI01 input high-/low-level width	t_{TIH0}, t_{TIL0}	4.0 V ≤ V_{DD} ≤ 5.5 V		$2/f_{sam}$ + 0.1 Note 2			μs
		2.7 V ≤ V_{DD} < 4.0 V		$2/f_{sam}$ + 0.2 Note 2			μs
		1.8 V ≤ V_{DD} < 2.7 V		$2/f_{sam}$ + 0.5 Note 2			μs
TI50, TI51 input frequency	f_{TI5}	2.7 V ≤ V_{DD} ≤ 5.5 V		0		4	MHz
		1.8 V ≤ V_{DD} < 2.7 V		0		275	kHz
TI50, TI51 input high-/low-level width	t_{TIH5}, t_{TIL5}	2.7 V ≤ V_{DD} ≤ 5.5 V		100			ns
		1.8 V ≤ V_{DD} < 2.7 V		1.8			μs
Interrupt request input high-/low-level width	t_{INTH}, t_{INTL}	INTP0 to INTP3, P40 to P47	2.7 V ≤ V_{DD} ≤ 5.5 V	1			μs
			1.8 V ≤ V_{DD} < 2.7 V	2			μs
$\overline{\text{RESET}}$ low-level width	t_{RSL}	2.7 V ≤ V_{DD} ≤ 5.5 V		10			μs
		1.8 V ≤ V_{DD} < 2.7 V		20			μs

Notes **1.** Value when the external clock is used. When a crystal resonator is used, it is 114 μs (MIN.).

2. Selection of f_{sam} = f_X, $f_X/4$, $f_X/64$ is possible using bits 0 and 1 (PRM00, PRM01) of prescaler mode register 0 (PRM0). However, if the TI00 valid edge is selected as the count clock, the value becomes f_{sam} = $f_X/8$.

T_{CY} vs. V_{DD} (main system clock operation)



(2) Read/write operation (T_A = –40 to +85°C, V_{DD} = 4.0 to 5.5 V) (1/3)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t_{ASTH}		$0.3t_{CY}$		ns
Address setup time	t_{ADS}		20		ns
Address hold time	t_{ADH}		6		ns
Data input time from address	t_{ADD1}			$(2 + 2n)t_{CY} - 54$	ns
	t_{ADD2}			$(3 + 2n)t_{CY} - 60$	ns
Address output time from $\overline{RD}\downarrow$	t_{RDAD}		0	100	ns
Data input time from $\overline{RD}\downarrow$	t_{RDD1}			$(2 + 2n)t_{CY} - 87$	ns
	t_{RDD2}			$(3 + 2n)t_{CY} - 93$	ns
Read data hold time	t_{RDH}		0		ns
$\overline{RD}$ low-level width	t_{RDL1}		$(1.5 + 2n)t_{CY} - 33$		ns
	t_{RDL2}		$(2.5 + 2n)t_{CY} - 33$		ns
Input time from $\overline{RD}\downarrow$ to $\overline{WAIT}\downarrow$	t_{RDWT1}			$t_{CY} - 43$	ns
	t_{RDWT2}			$t_{CY} - 43$	ns
Input time from $\overline{WR}\downarrow$ to $\overline{WAIT}\downarrow$	t_{WRWT}			$t_{CY} - 25$	ns
$\overline{WAIT}$ low-level width	t_{WTL}		$(0.5 + 2n)t_{CY} + 10$	$(2 + 2n)t_{CY}$	ns
Write data setup time	t_{WDS}		60		ns
Write data hold time	t_{WDH}		6		ns
$\overline{WR}$ low-level width	t_{WRL1}		$(1.5 + 2n)t_{CY} - 15$		ns
Delay time from ASTB$\downarrow$ to $\overline{RD}\downarrow$	t_{ASTRD}		6		ns
Delay time from ASTB$\downarrow$ to $\overline{WR}\downarrow$	t_{ASTWR}		$2t_{CY} - 15$		ns
Delay time from $\overline{RD}\uparrow$ to ASTB$\uparrow$ at external fetch	t_{RDAST}		$0.8t_{CY} - 15$	$1.2t_{CY}$	ns
Address hold time from $\overline{RD}\uparrow$ at external fetch	t_{RDADH}		$0.8t_{CY} - 15$	$1.2t_{CY} + 30$	ns
Write data output time from $\overline{RD}\uparrow$	t_{RDWD}		40		ns
Write data output time from $\overline{WR}\downarrow$	t_{WRWD}		10	60	ns
Address hold time from $\overline{WR}\uparrow$	t_{WRADH}		$0.8t_{CY} - 15$	$1.2t_{CY} + 30$	ns
Delay time from $\overline{WAIT}\uparrow$ to $\overline{RD}\uparrow$	t_{WTRD}		$0.8t_{CY}$	$2.5t_{CY} + 25$	ns
Delay time from $\overline{WAIT}\uparrow$ to $\overline{WR}\uparrow$	t_{WTWR}		$0.8t_{CY}$	$2.5t_{CY} + 25$	ns

Remarks 1. $t_{CY} = T_{CY}/4$

2. n indicates the number of waits.

3. C_L = 100 pF (C_L indicates the load capacitance of the AD0 to AD7, A8 to A15, $\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, and ASTB pins.)

(2) Read/write operation (T_A = –40 to +85°C, V_{DD} = 2.7 to 4.0 V) (2/3)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t_{ASTH}		$0.3t_{CY}$		ns
Address setup time	t_{ADS}		30		ns
Address hold time	t_{ADH}		10		ns
Input time from address to data	t_{ADD1}			$(2 + 2n)t_{CY} – 108$	ns
	t_{ADD2}			$(3 + 2n)t_{CY} – 120$	ns
Output time from $\overline{RD}$↓ to address	t_{RDAD}		0	200	ns
Input time from $\overline{RD}$↓ to data	t_{RDD1}			$(2 + 2n)t_{CY} – 148$	ns
	t_{RDD2}			$(3 + 2n)t_{CY} – 162$	ns
Read data hold time	t_{RDH}		0		ns
$\overline{RD}$ low-level width	t_{RDL1}		$(1.5 + 2n)t_{CY} – 40$		ns
	t_{RDL2}		$(2.5 + 2n)t_{CY} – 40$		ns
Input time from $\overline{RD}$↓ to $\overline{WAIT}$↓	t_{RDWT1}			$t_{CY} – 75$	ns
	t_{RDWT2}			$t_{CY} – 60$	ns
Input time from $\overline{WR}$↓ to $\overline{WAIT}$↓	t_{WRWT}			$t_{CY} – 50$	ns
$\overline{WAIT}$ low-level width	t_{WTL}		$(0.5 + 2n)t_{CY} + 10$	$(2 + 2n)t_{CY}$	ns
Write data setup time	t_{WDS}		60		ns
Write data hold time	t_{WDH}		10		ns
$\overline{WR}$ low-level width	t_{WRL1}		$(1.5 + 2n)t_{CY} – 30$		ns
Delay time from ASTB↓ to $\overline{RD}$↓	t_{ASTRD}		10		ns
Delay time from ASTB↓ to $\overline{WR}$↓	t_{ASTWR}		$2t_{CY} – 30$		ns
Delay time from $\overline{RD}$↑ to ASTB↑ at external fetch	t_{RDAST}		$0.8t_{CY} – 30$	$1.2t_{CY}$	ns
Hold time from $\overline{RD}$↑ to address at external fetch	t_{RDADH}		$0.8t_{CY} – 30$	$1.2t_{CY} + 60$	ns
Write data output time from $\overline{RD}$↑	t_{RDWD}		40		ns
Write data output time from $\overline{WR}$↓	t_{WRWD}		20	120	ns
Hold time from $\overline{WR}$↑ to address	t_{WRADH}		$0.8t_{CY} – 30$	$1.2t_{CY} + 60$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{RD}$↑	t_{WTRD}		$0.5t_{CY}$	$2.5t_{CY} + 50$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{WR}$↑	t_{WTWR}		$0.5t_{CY}$	$2.5t_{CY} + 50$	ns

Remarks **1.** $t_{CY} = T_{CY}/4$

2. n indicates the number of waits.

3. C_L = 100 pF (C_L indicates the load capacitance of the AD0 to AD7, A8 to A15, $\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, and ASTB pins.)

(2) Read/write operation (T_A = –40 to +85°C, V_{DD} = 1.8 to 2.7 V) (3/3)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
ASTB high-level width	t_{ASTH}		$0.3t_{CY}$		ns
Address setup time	t_{ADS}		120		ns
Address hold time	t_{ADH}		20		ns
Input time from address to data	t_{ADD1}			$(2 + 2n)t_{CY} – 233$	ns
	t_{ADD2}			$(3 + 2n)t_{CY} – 240$	ns
Output time from $\overline{RD}$↓ to address	t_{RDAD}		0	400	ns
Input time from $\overline{RD}$↓ to data	t_{RDD1}			$(2 + 2n)t_{CY} – 325$	ns
	t_{RDD2}			$(3 + 2n)t_{CY} – 332$	ns
Read data hold time	t_{RDH}		0		ns
$\overline{RD}$ low-level width	t_{RDL1}		$(1.5 + 2n)t_{CY} – 92$		ns
	t_{RDL2}		$(2.5 + 2n)t_{CY} – 92$		ns
Input time from $\overline{RD}$↓ to $\overline{WAIT}$↓	t_{RDWT1}			$t_{CY} – 350$	ns
	t_{RDWT2}			$t_{CY} – 132$	ns
Input time from $\overline{WR}$↓ to $\overline{WAIT}$↓	t_{WRWT}			$t_{CY} – 100$	ns
$\overline{WAIT}$ low-level width	t_{WTL}		$(0.5 + 2n)t_{CY} + 10$	$(2 + 2n)t_{CY}$	ns
Write data setup time	t_{WDS}		60		ns
Write data hold time	t_{WDH}		20		ns
$\overline{WR}$ low-level width	t_{WRL1}		$(1.5 + 2n)t_{CY} – 60$		ns
Delay time from ASTB↓ to $\overline{RD}$↓	t_{ASTRD}		20		ns
Delay time from ASTB↓ to $\overline{WR}$↓	t_{ASTWR}		$2t_{CY} – 60$		ns
Delay time from $\overline{RD}$↑ to ASTB↑ at external fetch	t_{RDAST}		$0.8t_{CY} – 60$	$1.2t_{CY}$	ns
Hold time from $\overline{RD}$↑ to address at external fetch	t_{RDADH}		$0.8t_{CY} – 60$	$1.2t_{CY} + 120$	ns
Write data output time from $\overline{RD}$↑	t_{RDWD}		40		ns
Write data output time from $\overline{WR}$↓	t_{WRWD}		40	240	ns
Hold time from $\overline{WR}$↑ to address	t_{WRADH}		$0.8t_{CY} – 60$	$1.2t_{CY} + 120$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{RD}$↑	t_{WTRD}		$0.5t_{CY}$	$2.5t_{CY} + 100$	ns
Delay time from $\overline{WAIT}$↑ to $\overline{WR}$↑	t_{WTWR}		$0.5t_{CY}$	$2.5t_{CY} + 100$	ns

Remarks **1.** $t_{CY} = T_{CY}/4$

2. n indicates the number of waits.

3. C_L = 100 pF (C_L indicates the load capacitance of the AD0 to AD7, A8 to A15, $\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, and ASTB pins.)

(3) Serial interface (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (1/3)

(a) 3-wire serial I/O mode ($\overline{SCK3n}$... Internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{SCK3n}$ cycle time	t_{KCY1}	4.0 V ≤ V_{DD} ≤ 5.5 V	954			ns
		2.7 V ≤ V_{DD} < 4.0 V	1600			ns
		1.8 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{SCK3n}$ high-/low-level width	t_{KH1}, t_{KL1}	4.0 V ≤ V_{DD} ≤ 5.5 V	t_{KCY1}/2 – 50			ns
		1.8 V ≤ V_{DD} < 4.0 V	t_{KCY1}/2 – 100			ns
SI3n setup time (to $\overline{SCK3n}$↑)	t_{SIK1}	4.0 V ≤ V_{DD} ≤ 5.5 V	100			ns
		2.7 V ≤ V_{DD} < 4.0 V	150			ns
		1.8 V ≤ V_{DD} < 2.7 V	300			ns
SI3n hold time (from $\overline{SCK3n}$↑)	t_{KSI1}		400			ns
Delay time from $\overline{SCK3n}$↓ to SO3n output	t_{KSO1}	C = 100 pF**Note**			300	ns

Note C is the load capacitance of the $\overline{SCK3n}$ and SO3n output lines.

(b) 3-wire serial I/O mode ($\overline{SCK3n}$... External clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{SCK3n}$ cycle time	t_{KCY2}	4.0 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.0 V	1600			ns
		1.8 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{SCK3n}$ high-/low-level width	t_{KH2}, t_{KL2}	4.0 V ≤ V_{DD} ≤ 5.5 V	400			ns
		2.7 V ≤ V_{DD} < 4.0 V	800			ns
		1.8 V ≤ V_{DD} < 2.7 V	1600			ns
SI3n setup time (to $\overline{SCK3n}$↑)	t_{SIK2}		100			ns
SI3n hold time (from $\overline{SCK3n}$↑)	t_{KSI2}		400			ns
Delay time from $\overline{SCK3n}$↓ to SO3n output	t_{KSO2}	C = 100 pF**Note**			300	ns

Note C is the load capacitance of the SO3n output line.

Remark μPD780024A, 780034A Subseries: n = 0, 1
μPD780024AY, 780034AY Subseries: n = 0

(3) Serial interface (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (2/3)

(c) UART mode (dedicated baud-rate generator output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		4.0 V ≤ V_{DD} ≤ 5.5 V			131031	bps
		2.7 V ≤ V_{DD} < 4.0 V			78125	bps
		1.8 V ≤ V_{DD} < 2.7 V			39063	bps

(d) UART mode (external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK0 cycle time	t_{KCY3}	4.0 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.0 V	1600			ns
		1.8 V ≤ V_{DD} < 2.7 V	3200			ns
ASCK0 high-/low-level width	t_{KH3}, t_{KL3}	4.0 V ≤ V_{DD} ≤ 5.5 V	400			ns
		2.7 V ≤ V_{DD} < 4.0 V	800			ns
		1.8 V ≤ V_{DD} < 2.7 V	1600			ns
Transfer rate		4.0 V ≤ V_{DD} ≤ 5.5 V			39063	bps
		2.7 V ≤ V_{DD} < 4.0 V			19531	bps
		1.8 V ≤ V_{DD} < 2.7 V			9766	bps

(e) UART mode (infrared data transfer mode)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Transfer rate		4.0 V ≤ V_{DD} ≤ 5.5 V		131031	bps
Allowable bit rate error		4.0 V ≤ V_{DD} ≤ 5.5 V		±0.87	%
Output pulse width		4.0 V ≤ V_{DD} ≤ 5.5 V	1.2	0.24/fbr **Note**	μs
Input pulse width		4.0 V ≤ V_{DD} ≤ 5.5 V	4/f_X		μs

Note fbr: Specified baud rate

(3) Serial interface (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (3/3)

(f) I²C bus mode (μPD780024AY, 780034AY Subseries only)

Parameter		Symbol	Standard Mode		High-Speed Mode		Unit
			MIN.	MAX.	MIN.	MAX.	
SCL0 clock frequency		f_{CLK}	0	100	0	400	kHz
Bus free time (between stop and start conditions)		t_{BUF}	4.7	–	1.3	–	μs
Hold time[Note 1]		$t_{HD:STA}$	4.0	–	0.6	–	μs
SCL0 clock low-level width		t_{LOW}	4.7	–	1.3	–	μs
SCL0 clock high-level width		t_{HIGH}	4.0	–	0.6	–	μs
Start/restart condition setup time		$t_{SU:STA}$	4.7	–	0.6	–	μs
Data hold time	CBUS-compatible master	$t_{HD:DAT}$	5.0	–	–	–	μs
	I²C bus		0[Note 2]	–	0[Note 2]	0.9[Note 3]	μs
Data setup time		$t_{SU:DAT}$	250	–	100[Note 4]	–	ns
SDA0 and SCL0 signal rise time		t_R	–	1000	20 + 0.1Cb[Note 5]	300	ns
SDA0 and SCL0 signal fall time		t_F	–	300	20 + 0.1Cb[Note 5]	300	ns
Stop condition setup time		$t_{SU:STO}$	4.0	–	0.6	–	μs
Spike pulse width controlled by input filter		t_{SP}	–	–	0	50	ns
Capacitive load per bus line		Cb	–	400	–	400	pF

Notes 1. In the start condition, the first clock pulse is generated after this hold time.

2. To fill in the undefined area of the SCL0 falling edge, it is necessary for the device to internally provide at least 300 ns of hold time for the SDA0 signal (which is $V_{IHmin.}$ of the SCL0 signal).

3. If the device does not extend the SCL0 signal low hold time (t_{LOW}), only the maximum data hold time $t_{HD:DAT}$ needs to be fulfilled.

4. The high-speed mode I²C bus is available in a standard mode I²C bus system. At this time, the conditions described below must be satisfied.

- If the device does not extend the SCL0 signal low state hold time
  $t_{SU:DAT} \geq 250$ ns
- If the device extends the SCL0 signal low state hold time
  Be sure to transmit the next data bit to the SDA0 line before the SCL0 line is released ($t_{Rmax.} + t_{SU:DAT}$ = 1000 + 250 = 1250 ns by standard mode I²C bus specification).

5. Cb: Total capacitance per bus line (unit: pF)

A/D Converter Characteristics (T_A = –40 to +85°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, AV_{SS} = V_{SS} = 0 V) (1/2)

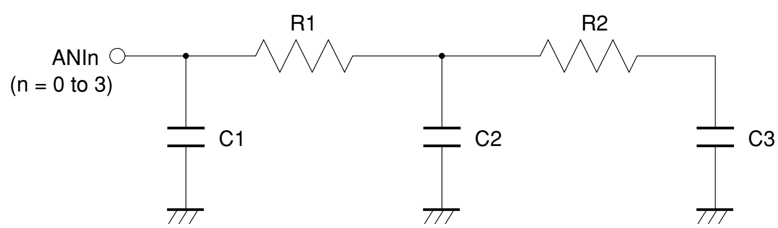
(1) 8-bit A/D converter: μPD780024A, 780024AY Subseries

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error[Note]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
Conversion time	t_{CONV}	4.0 V ≤ AV_{DD} ≤ 5.5 V	14		96	μs
		2.7 V ≤ AV_{DD} < 4.0 V	19		96	μs
		1.8 V ≤ AV_{DD} < 2.7 V	28		96	μs
Analog input voltage	V_{AIN}		0		AV_{REF}	V
Reference voltage	AV_{REF}		1.8		AV_{DD}	V
Resistance between AV_{REF} and AV_{SS}	R_{REF}	During A/D converter operation	20	40		kΩ

Note Excludes quantization error (±1/2 LSB). This value is indicated as a ratio (%FSR) to the full-scale value.

Remark The impedance of the analog input pins is shown below.

[Equivalent circuit]



[Parameter value]

(TYP.)

AV_{DD}	R1	R2	C1	C2	C3
2.7 V	12 kΩ	8.0 kΩ	3.0 pF	3.0 pF	2.0 pF
4.5 V	4 kΩ	2.7 kΩ	3.0 pF	1.4 pF	2.0 pF

A/D Converter Characteristics (T_A = –40 to +85°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, AV_{SS} = V_{SS} = 0 V) (2/2)

(2) 10-bit A/D converter: μPD780034A, 780034AY Subseries

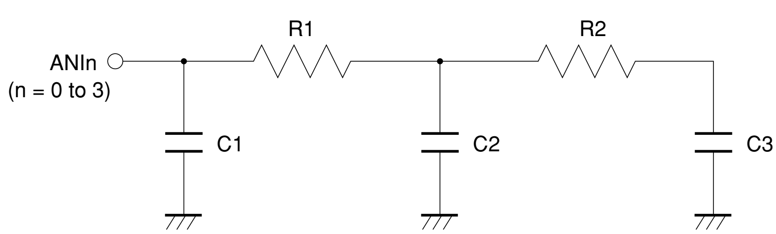
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			10	10	10	bit
Overall error[Notes 1, 2]		4.0 V ≤ AV_{REF} ≤ 5.5 V		±0.2	±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V		±0.3	±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V		±0.6	±1.2	%FSR
Conversion time	t_{CONV}	4.0 V ≤ AV_{DD} ≤ 5.5 V	14		96	μs
		2.7 V ≤ AV_{DD} < 4.0 V	19		96	μs
		1.8 V ≤ AV_{DD} < 2.7 V	28		96	μs
Zero-scale error[Notes 1, 2]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
Full-scale error[Notes 1, 2]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		1.8 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
Integral linearity error[Note 1]		4.0 V ≤ AV_{REF} ≤ 5.5 V			±2.5	LSB
		2.7 V ≤ AV_{REF} < 4.0 V			±4.5	LSB
		1.8 V ≤ AV_{REF} < 2.7 V			±8.5	LSB
Differential linearity error		4.0 V ≤ AV_{REF} ≤ 5.5 V			±1.5	LSB
		2.7 V ≤ AV_{REF} < 4.0 V			±2.0	LSB
		1.8 V ≤ AV_{REF} < 2.7 V			±3.5	LSB
Analog input voltage	V_{AIN}		0		AV_{REF}	V
Reference voltage	AV_{REF}		1.8		AV_{DD}	V
Resistance between AV_{REF} and AV_{SS}	R_{REF}	During A/D converter operation	20	40		kΩ

Notes 1. Excludes quantization error (±1/2 LSB).

2. This value is indicated as a ratio to the full-scale value.

Remark The impedance of the analog input pins is shown below.

[Equivalent circuit]



[Parameter value]

(TYP.)

AV_{DD}	R1	R2	C1	C2	C3
2.7 V	12 kΩ	8.0 kΩ	3.0 pF	3.0 pF	2.0 pF
4.5 V	4 kΩ	2.7 kΩ	3.0 pF	1.4 pF	2.0 pF

Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (T_A = –40 to +85°C)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention power supply voltage	V_{DDDR}		1.6		5.5	V
Data retention power supply current	I_{DDDR}	V_{DDDR} = 1.6 V (Subsystem clock unused (XT1 = V_{DD}) and feedback resistor disconnected)		0.1	30	μA
Release signal set time	t_{SREL}		0			μs
Oscillation stabilization wait time	t_{WAIT}	Release by $\overline{\text{RESET}}$		$2^{17}/f_X$		s
		Release by interrupt request		**Note**		s

Note Selection of $2^{12}/f_X$ and $2^{14}/f_X$ to $2^{17}/f_X$ is possible using bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register (OSTS).

Flash Memory Programming Characteristics (1/3)
(T_A = +10 to +40°C, V_{DD} = AV_{DD} = 2.7 to 5.5 V, V_{SS} = AV_{SS} = 0 V)

(1) μPD78F0034A, 78F0034AY

(a) Write erase characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Operating frequency	f_X	4.0 V ≤ V_{DD} ≤ 5.5 V	1.0		8.38	MHz
		2.7 V ≤ V_{DD} < 4.0 V	1.0		5.0	MHz
V_{PP} supply voltage	V_{PP2}	During flash memory programming[Note 1]	9.7	10.0	10.3	V
V_{DD} supply current[Note 2]	I_{DD}	When V_{PP} = V_{PP2}, 8.38 MHz crystal oscillation operating mode, V_{DD} = 5.0 V±10%			24	mA
		When V_{PP} = V_{PP2}, 5.0 MHz crystal oscillation operating mode, V_{DD} = 3.0 V±10%			12	mA
V_{PP} supply current	I_{PP}	When V_{PP} = V_{PP2}			100	mA
Step erase time[Note 3]	T_{er}		0.99	1.0	1.01	s
Overall erase time[Note 4]	T_{era}	When step erase time = 1 s			20	s/chip
Step write time[Note 5]	T_{wr}		50		100	μs
Overall write time per word[Note 6]	T_{wrw}	When step write time = 100 μs (1 word = 1 byte)			1000	μs
Number of rewrites per chip[Note 7]	C_{erwr}	1 erase + 1 write after erase = 1 rewrite			20[Note 8]	Times/area

Notes 1. Product rank “K, E, P” indicates 10.2 V (MIN.), 10.3 V (TYP.), and 10.4 V (MAX.).

2. AV_{DD} current and port current (current that flows through the internal pull-up resistor) are not included.

3. The recommended setting value of the step erase time is 1 s.

4. The prewrite time before erasure and the erase verify time are not included.

5. The recommended setting value of the step write time is 50 μs.

6. The actual write time per word is 100 μs longer. The internal verify time during or after a write is not included.

7. When a product is first written after shipment, “erase → write” and “write only” are both taken as one rewrite.

Example: P: Write, E: Erase
Shipped product →P→E→P→E→P: 3 rewrites
Shipped product →E→P→E→P→E→P: 3 rewrites

8. Product rank “K, E” indicates one time (MAX.).

(b) Serial write operation characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Set time from V_{DD}↑ to V_{PP}↑	t_{DP}		10			μs
Release time from V_{PP}↑ to $\overline{RESET}$↑	t_{PR}		1.0			μs
V_{PP} pulse input start time from $\overline{RESET}$↑	t_{RP}		1.0			μs
V_{PP} pulse high-/low-level width	t_{PW}		8.0			μs
V_{PP} pulse input end time from $\overline{RESET}$↑	t_{RPE}				20	ms
V_{PP} pulse low-level input voltage	V_{PPL}		0.8V_{DD}	V_{DD}	1.2V_{DD}	V
V_{PP} pulse high-level input voltage	V_{PPH}		9.7	10.0	10.3	V

Flash Memory Programming Characteristics (2/3)
(T_A = +10 to +40°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, V_{SS} = AV_{SS} = 0 V)

(2) μPD78F0034BY, 78F0034BY(A)

(a) Write erase characteristics

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Operating frequency	f_X	$4.0\ V \le V_{DD} \le 5.5\ V$			1.0		8.38	MHz
		$2.7\ V \le V_{DD} < 4.0\ V$			1.0		5.0	MHz
		$1.8\ V \le V_{DD} < 2.7\ V$			1.0		1.25	MHz
V_{PP} supply voltage	V_{PP2}	During flash memory programming			9.7	10.0	10.3	V
V_{DD} supply current[Note 1]	I_{DD}	When $V_{PP} = V_{PP2}$	8.38 MHz crystal oscillation operating mode	V_{DD} = 5.0 V±10%			24	mA
				V_{DD} = 3.0 V±10%			17	mA
V_{PP} supply current	I_{PP}	When $V_{PP} = V_{PP2}$					100	mA
Step erase time[Note 2]	T_{er}				0.199	0.2	0.201	s
Overall erase time[Note 3]	T_{era}	When step erase time = 0.2 s					20	s/chip
Writeback time[Note 4]	T_{wb}				49.4	50	50.6	ms
Number of writebacks per writeback command[Note 5]	C_{wb}	When writeback time = 50 ms					60	Times
Number of erases/writebacks	C_{erwb}						16	Times
Step write time[Note 6]	T_{wr}				48	50	52	μs
Overall write time per word[Note 7]	T_{wrw}	When step write time = 50 μs (1 word = 1 byte)			48		520	μs
Number of rewrites per chip[Note 8]	C_{erwr}	1 erase + 1 write after erase = 1 rewrite					20	Times/area

Notes 1. AV_{DD} current and port current (current that flows through the internal pull-up resistor) are not included.

2. The recommended setting value of the step erase time is 0.2 s.

3. The prewrite time before erasure and the erase verify time (writeback time) are not included.

4. The recommended setting value of the writeback time is 50 ms.

5. Writeback is executed once by the issuance of the writeback command. Therefore, the number of retries must be the maximum value minus the number of commands issued.

6. The recommended setting value of the step write time is 50 μs.

7. The actual write time per word is 100 μs longer. The internal verify time during or after a write is not included.

8. When a product is first written after shipment, "erase → write" and "write only" are both taken as one rewrite.

Example: P: Write, E: Erase

Shipped product →P→E→P→E→P: 3 rewrites

Shipped product →E→P→E→P→E→P: 3 rewrites

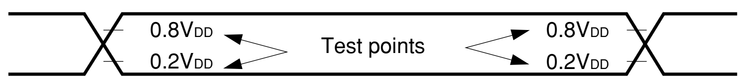
Flash Memory Programming Characteristics (3/3)
(T_A = +10 to +40°C, V_{DD} = AV_{DD} = 1.8 to 5.5 V, V_{SS} = AV_{SS} = 0 V)

(b) Serial write operation characteristics

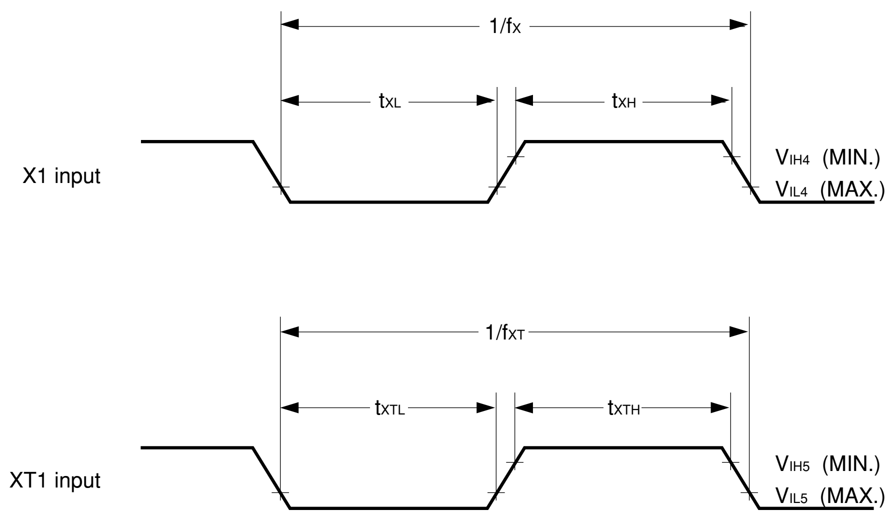
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Set time from V_{DD}↑ to V_{PP}↑	t_{DP}		10			μs
Release time from V_{PP}↑ to $\overline{RESET}$↑	t_{PR}		1.0			μs
V_{PP} pulse input start time from $\overline{RESET}$↑	t_{RP}		1.0			μs
V_{PP} pulse high-/low-level width	t_{PW}		8.0			μs
V_{PP} pulse input end time from $\overline{RESET}$↑	t_{RPE}				20	ms
V_{PP} pulse low-level input voltage	V_{PPL}		0.8V_{DD}		1.2V_{DD}	V
V_{PP} pulse high-level input voltage	V_{PPH}		9.7	10.0	10.3	V

Timing Charts

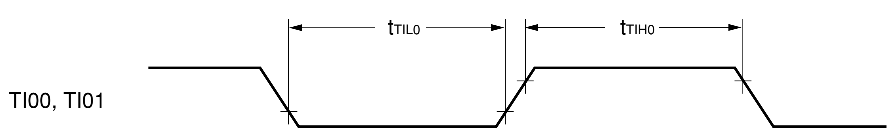
AC Timing Test Points (excluding X1, XT1 input)

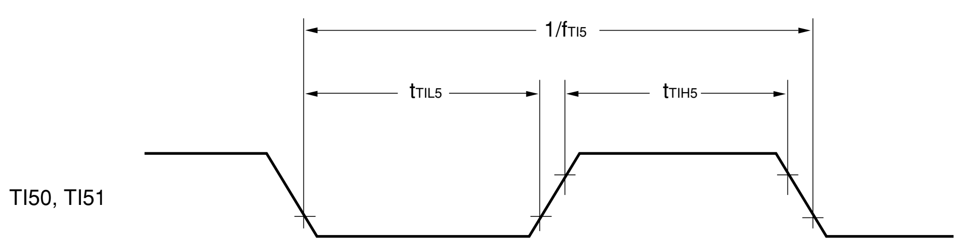


Clock Timing

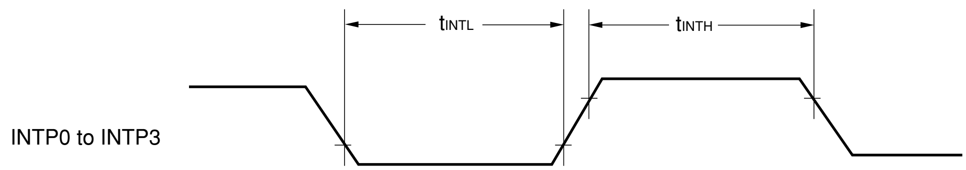


TI Timing

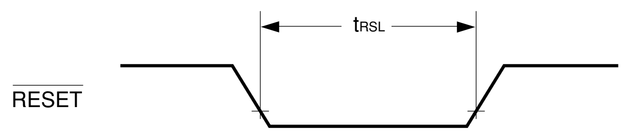




Interrupt Request Input Timing

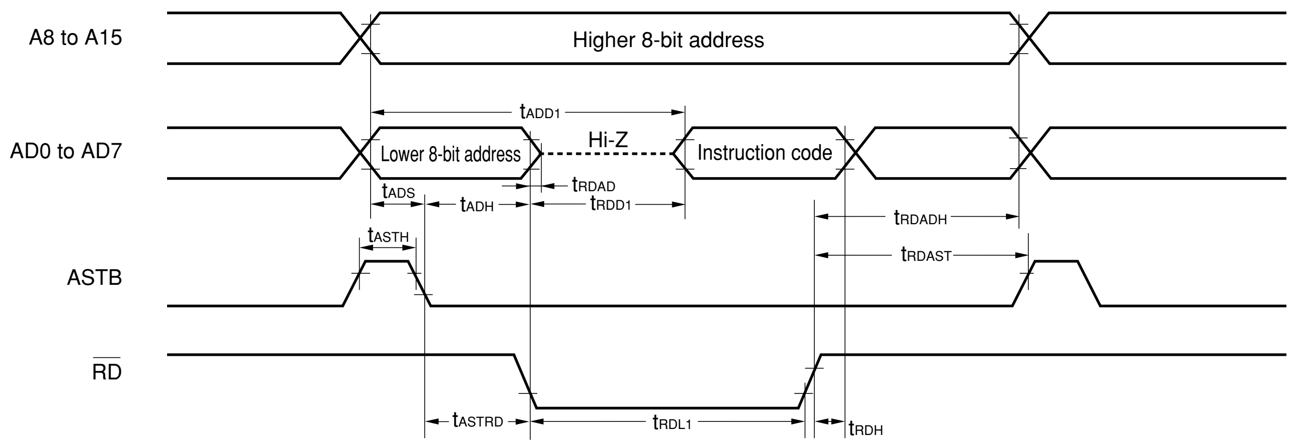


$\overline{\text{RESET}}$ Input Timing

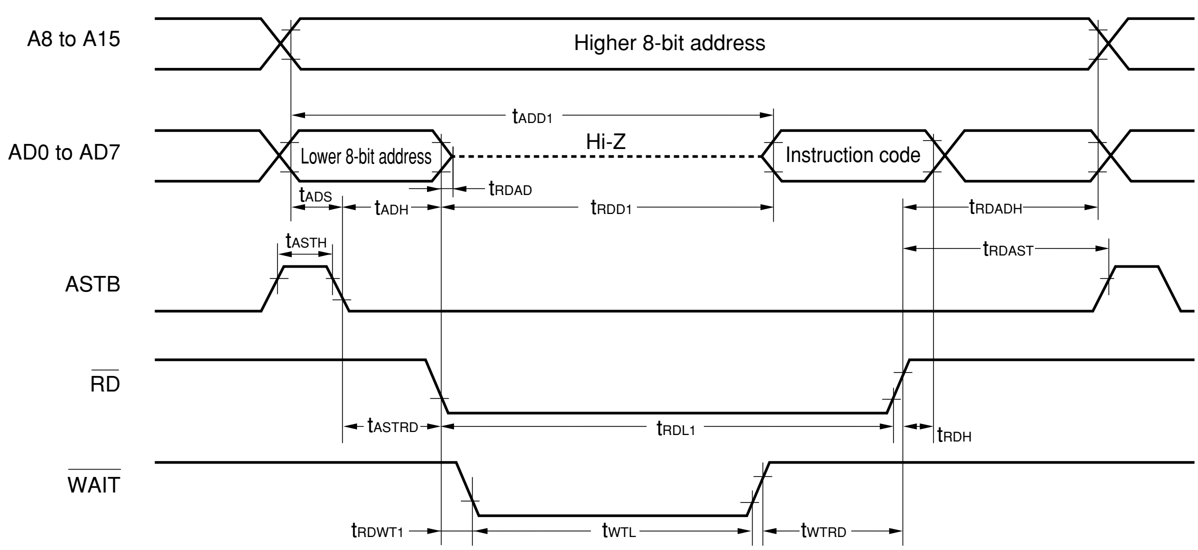


Read/Write Operation

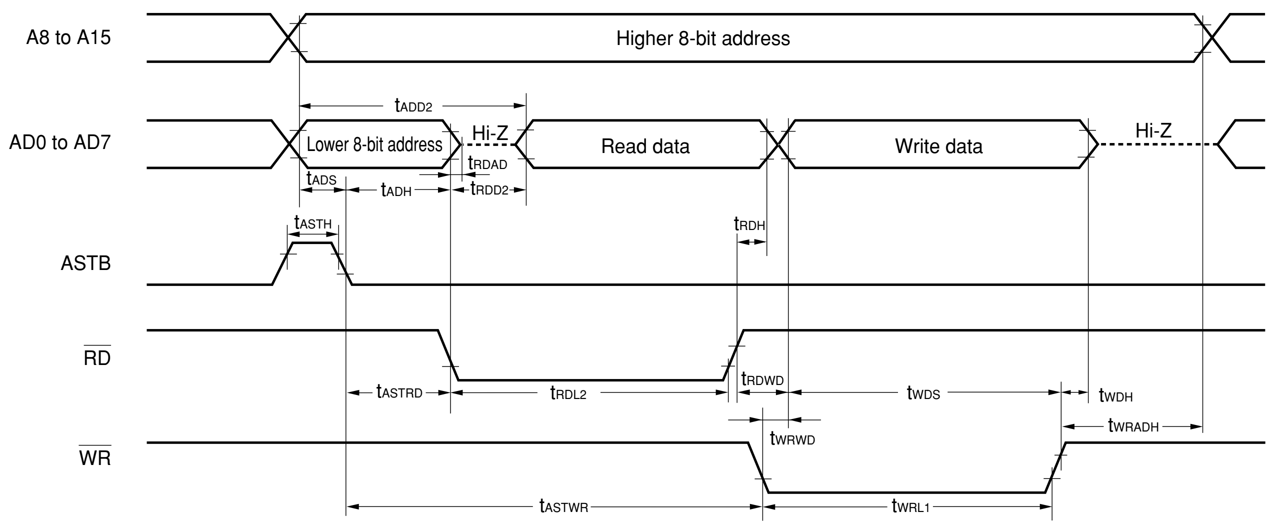
External fetch (no wait):



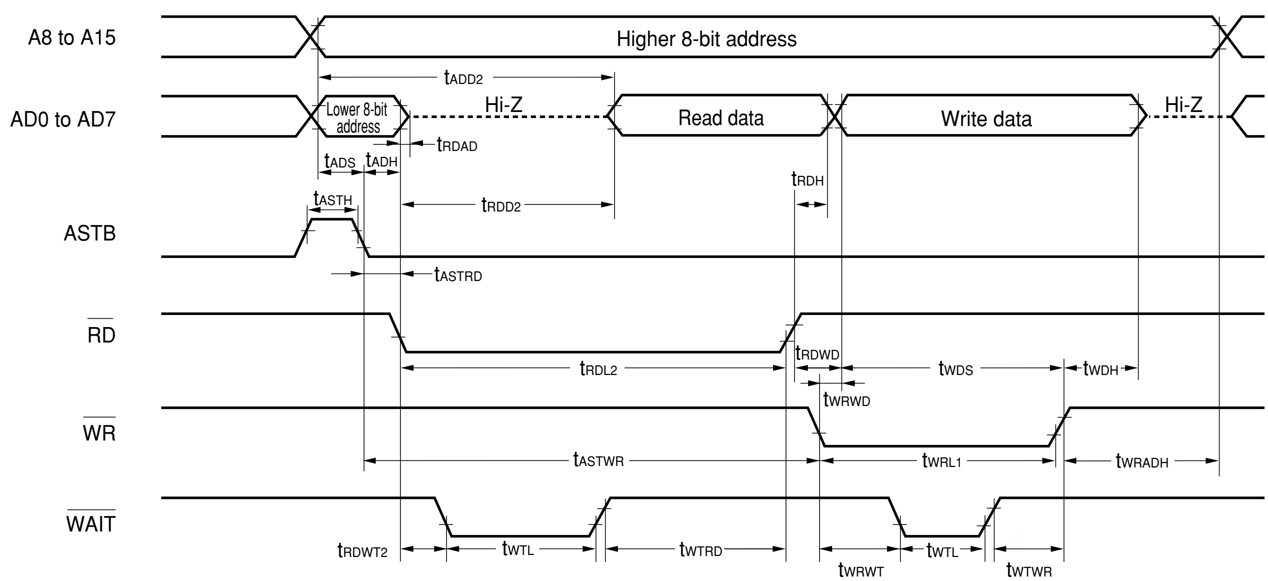
External fetch (wait insertion):



External data access (no wait):

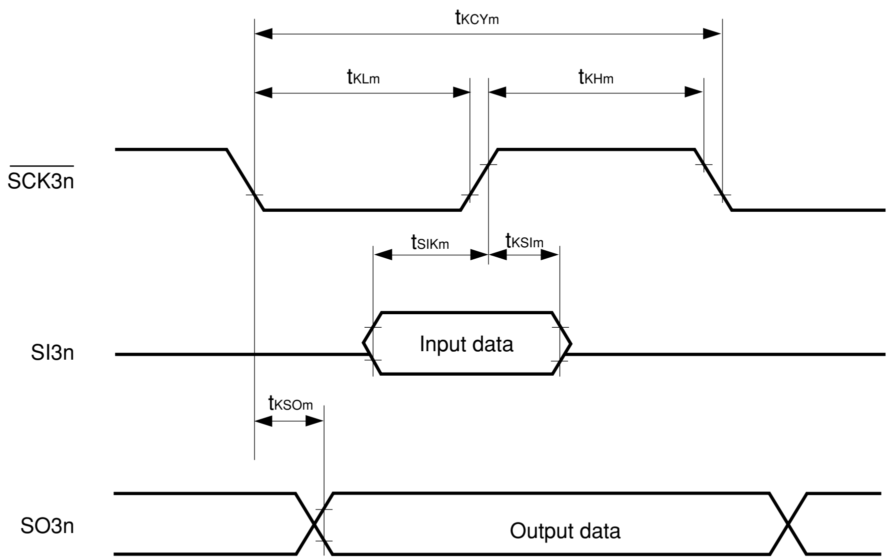


External data access (wait insertion):



Serial Transfer Timing

3-wire serial I/O mode:

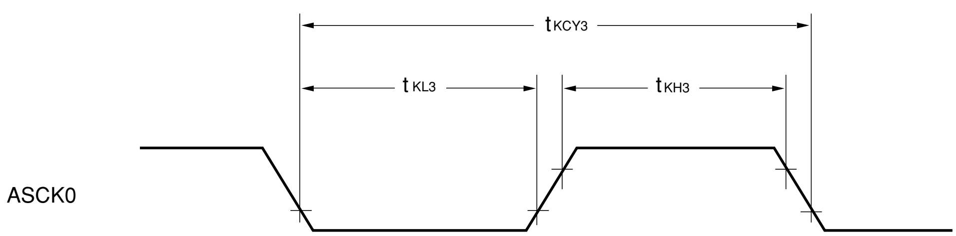


Remarks 1. m = 1, 2

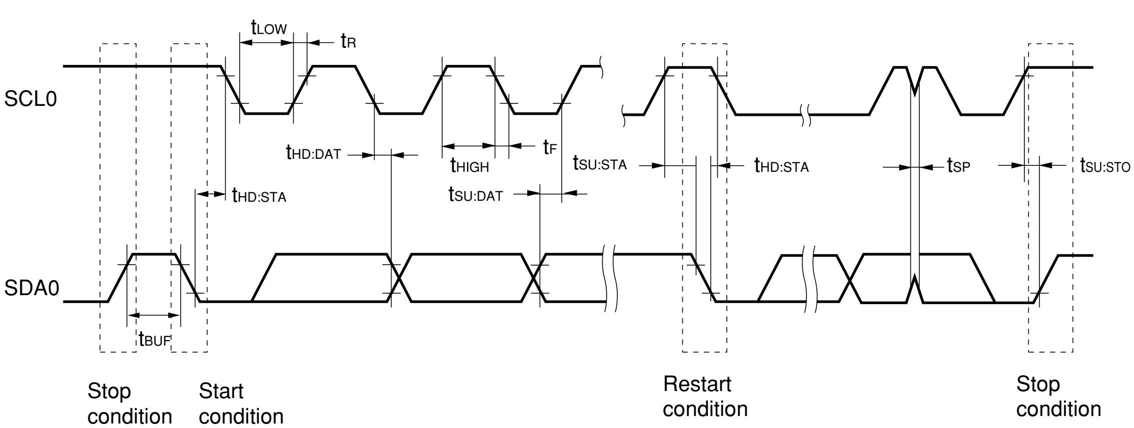
2. μPD780024A, 780034A Subseries: n = 0, 1

μPD780024AY, 780034AY Subseries: n = 0

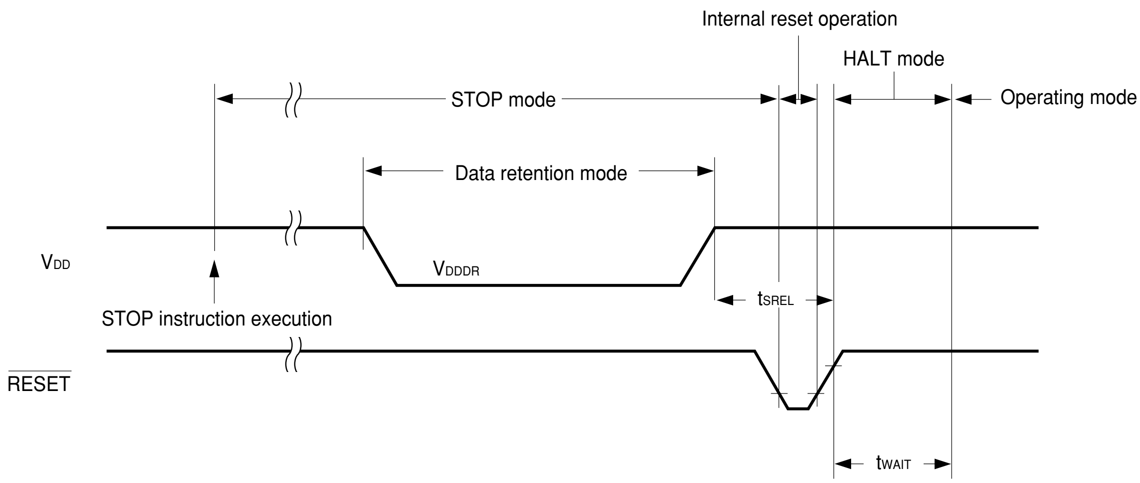
UART mode (external clock input):



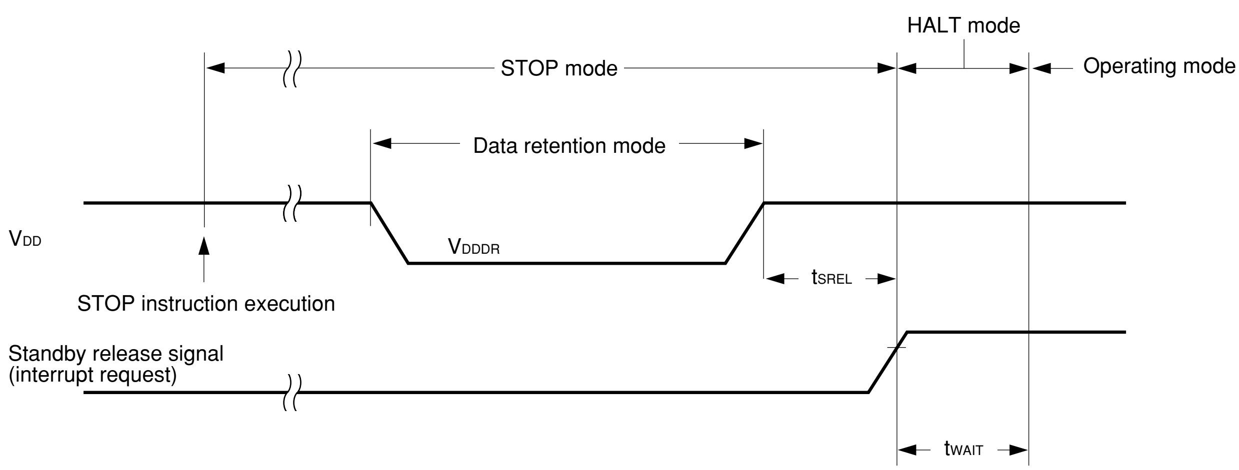
I²C bus mode (μPD780024AY, 780034AY Subseries only):



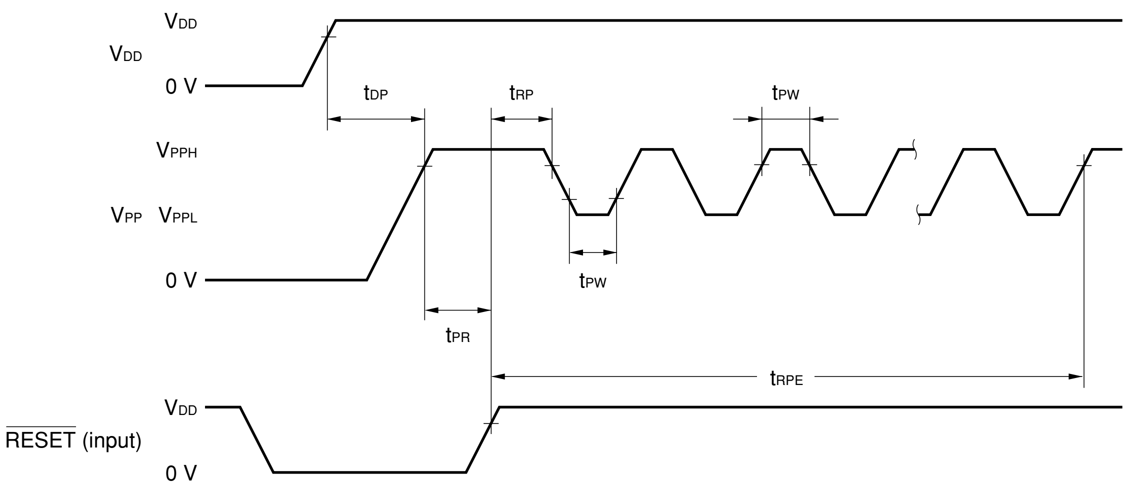
Data Retention Timing (STOP Mode Release by $\overline{\text{RESET}}$)



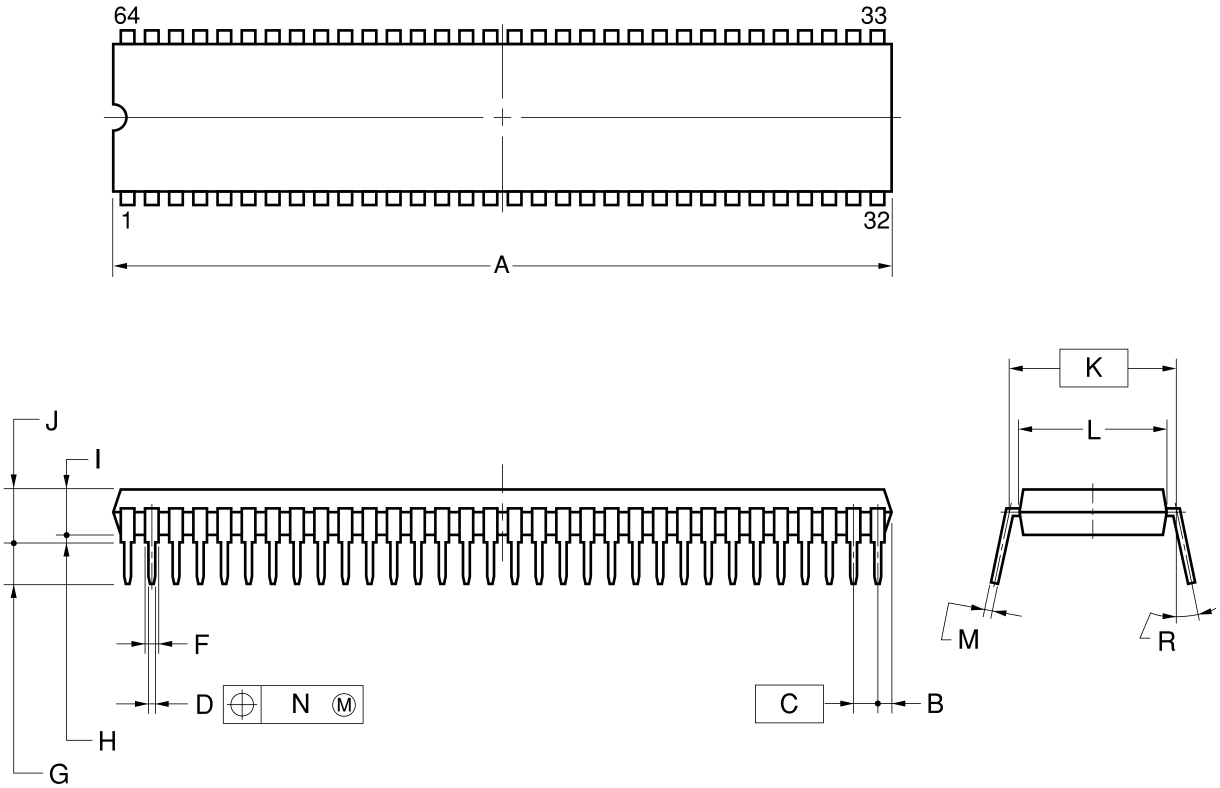
Data Retention Timing (Standby Release Signal: STOP Mode Release by Interrupt Request Signal)



Flash Memory Write Mode Set Timing



64-PIN PLASTIC SDIP (19.05mm(750))



NOTES

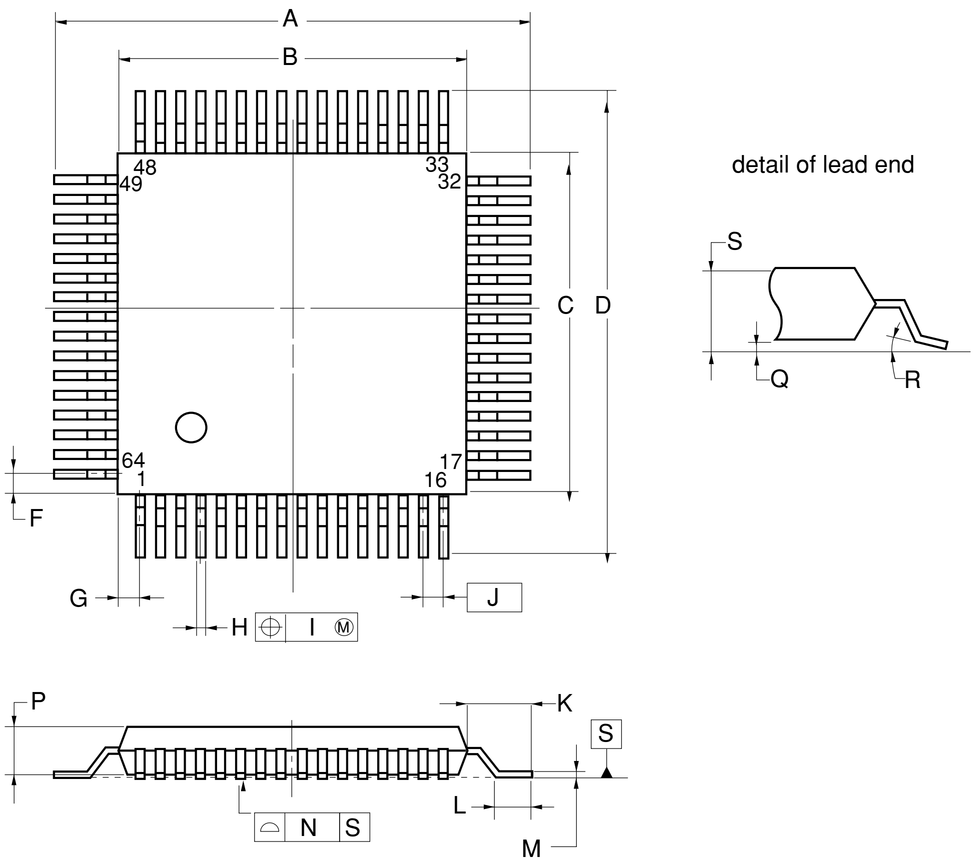
1. Each lead centerline is located within 0.17 mm of its true position (T.P.) at maximum material condition.
2. Item "K" to center of leads when formed parallel.

ITEM	MILLIMETERS
A	$58.0^{+0.68}_{-0.20}$
B	1.78 MAX.
C	1.778 (T.P.)
D	0.50±0.10
F	0.9 MIN.
G	3.2±0.3
H	0.51 MIN.
I	$4.05^{+0.26}_{-0.20}$
J	5.08 MAX.
K	19.05 (T.P.)
L	17.0±0.2
M	$0.25^{+0.10}_{-0.05}$
N	0.17
R	0 ~ 15°

P64C-70-750A,C-4

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

64-PIN PLASTIC QFP (14x14)



NOTE

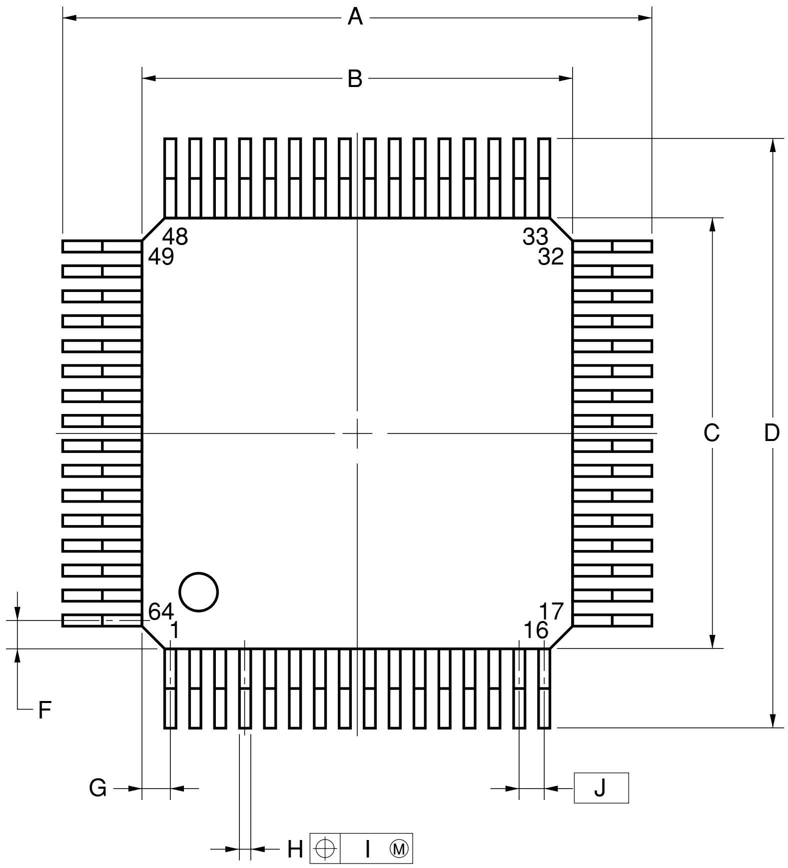
Each lead centerline is located within 0.15 mm of its true position (T.P.) at maximum material condition.

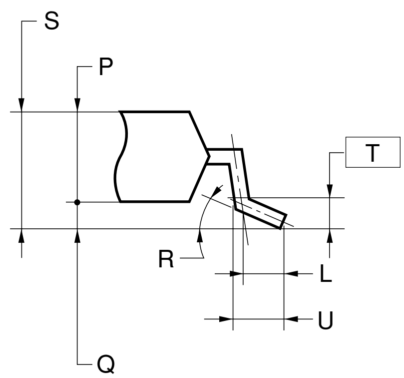
ITEM	MILLIMETERS
A	17.6±0.4
B	14.0±0.2
C	14.0±0.2
D	17.6±0.4
F	1.0
G	1.0
H	$0.37^{+0.08}_{-0.07}$
I	0.15
J	0.8 (T.P.)
K	1.8±0.2
L	0.8±0.2
M	$0.17^{+0.08}_{-0.07}$
N	0.10
P	2.55±0.1
Q	0.1±0.1
R	5°± 5°
S	2.85 MAX.

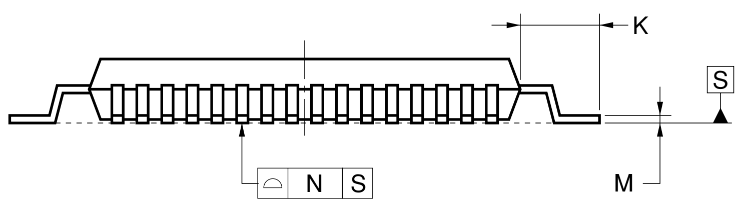
P64GC-80-AB8-5

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

64-PIN PLASTIC LQFP (14x14)







NOTE

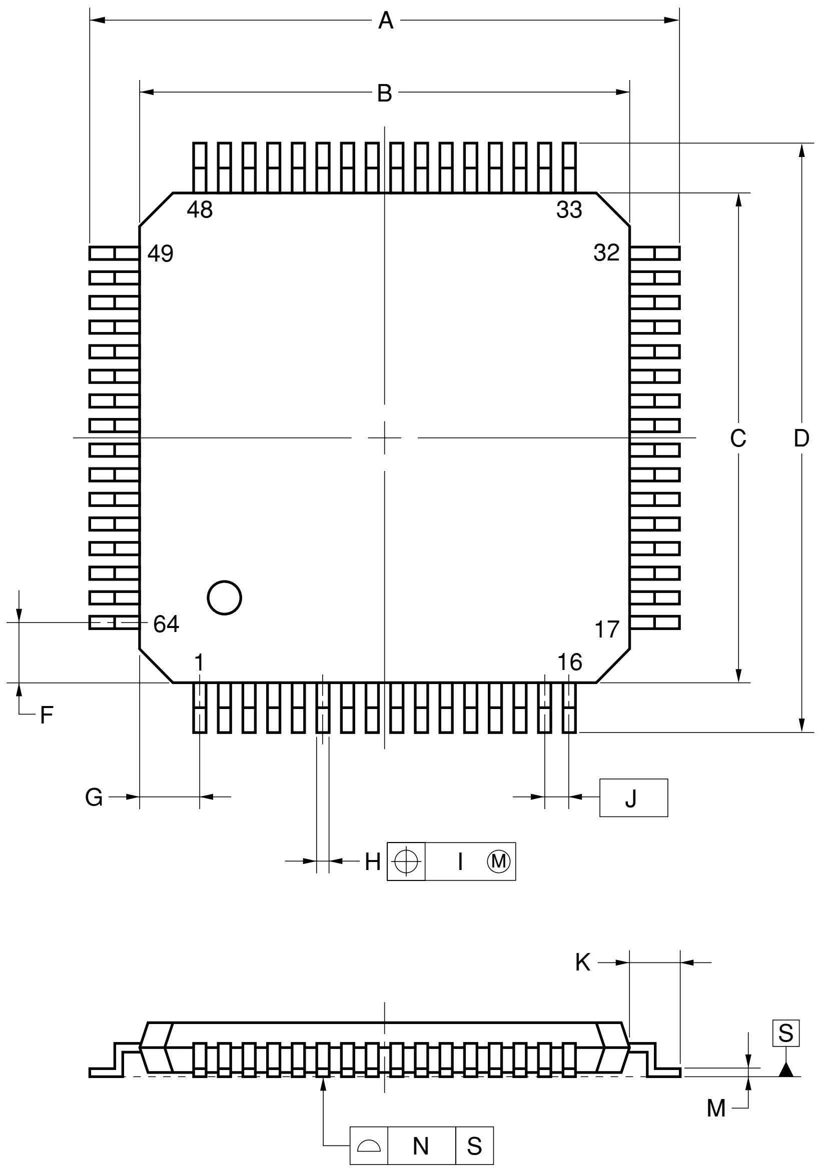
Each lead centerline is located within 0.20 mm of its true position (T.P.) at maximum material condition.

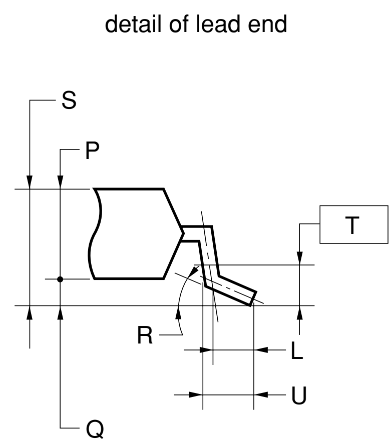
ITEM	MILLIMETERS
A	17.2±0.2
B	14.0±0.2
C	14.0±0.2
D	17.2±0.2
F	1.0
G	1.0
H	$0.37^{+0.08}_{-0.07}$
I	0.20
J	0.8 (T.P.)
K	1.6±0.2
L	0.8
M	$0.17^{+0.03}_{-0.06}$
N	0.10
P	1.4±0.1
Q	0.127±0.075
R	$3°^{+4°}_{-3°}$
S	1.7 MAX.
T	0.25
U	0.886±0.15

P64GC-80-8BS

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

64-PIN PLASTIC TQFP (12x12)





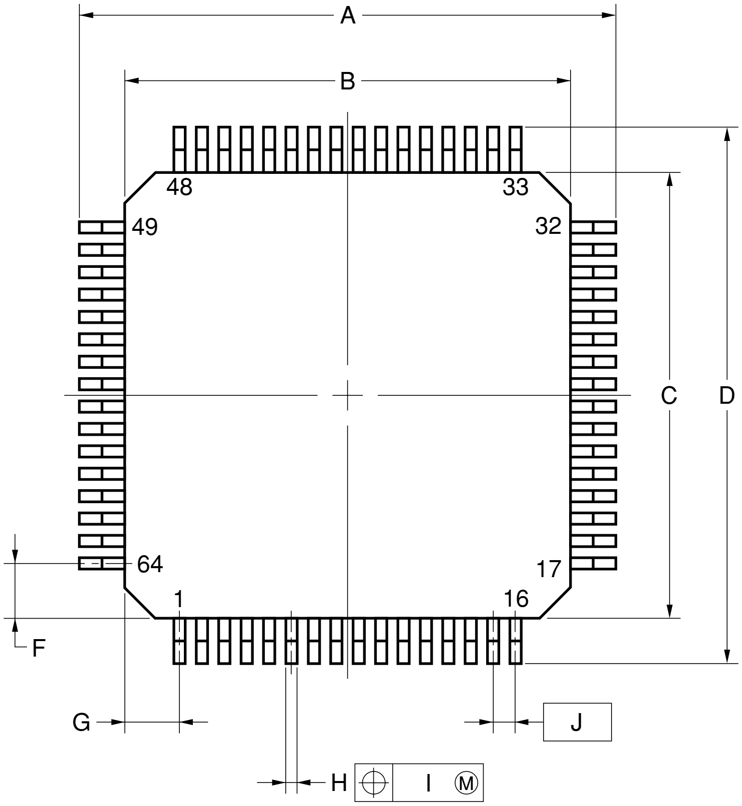
NOTE
Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	14.0±0.2
B	12.0±0.2
C	12.0±0.2
D	14.0±0.2
F	1.125
G	1.125
H	$0.32^{+0.06}_{-0.10}$
I	0.13
J	0.65 (T.P.)
K	1.0±0.2
L	0.5
M	$0.17^{+0.03}_{-0.07}$
N	0.10
P	1.0
Q	0.1±0.05
R	$3^{\circ}{}^{+4^{\circ}}_{-3^{\circ}}$
S	1.1±0.1
T	0.25
U	0.6±0.15

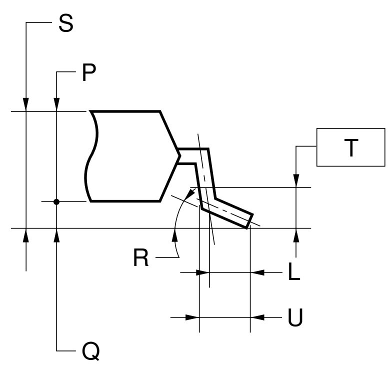
P64GK-65-9ET-3

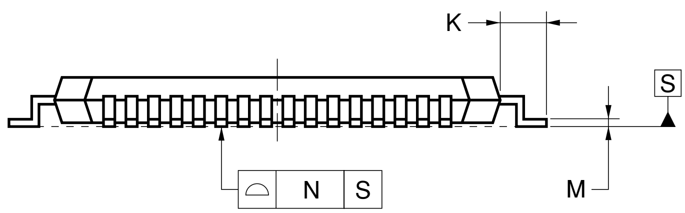
Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

64-PIN PLASTIC LQFP (10x10)



detail of lead end



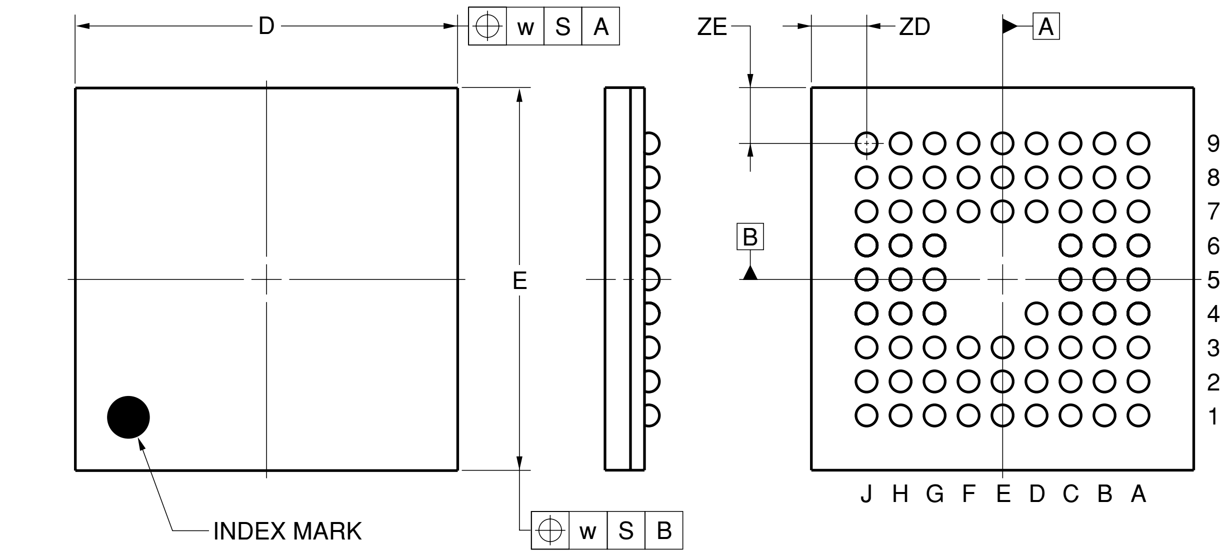


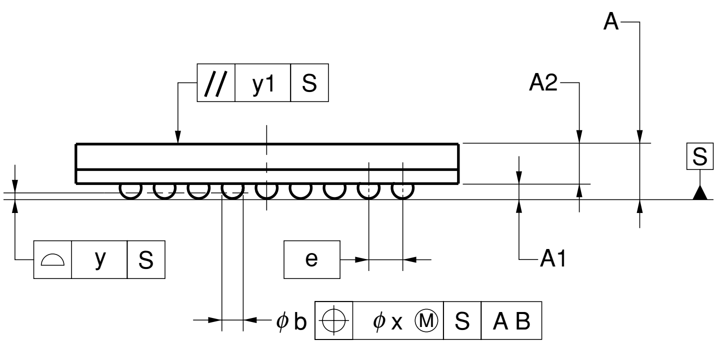
ITEM	MILLIMETERS
A	12.0±0.2
B	10.0±0.2
C	10.0±0.2
D	12.0±0.2
F	1.25
G	1.25
H	0.22±0.05
I	0.08
J	0.5 (T.P.)
K	1.0±0.2
L	0.5
M	$0.17^{+0.03}_{-0.07}$
N	0.08
P	1.4
Q	0.1±0.05
R	$3°^{+4°}_{-3°}$
S	1.5±0.10
T	0.25
U	0.6±0.15

S64GB-50-8EU-1

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

73-PIN PLASTIC FBGA (9x9)





(UNIT:mm)

ITEM	DIMENSIONS
D	9.00±0.10
E	9.00±0.10
w	0.20
A	1.28±0.10
A1	0.35±0.06
A2	0.93
e	0.80
b	$0.50^{+0.05}_{-0.10}$
x	0.08
y	0.10
y1	0.20
ZD	1.30
ZE	1.30

P73F1-80-CN3

Remark The external dimensions and materials of the ES version are the same as those of the mass-produced version.

★
CHAPTER 28 RECOMMENDED SOLDERING CONDITIONS

This product should be soldered and mounted under the following recommended conditions.

For soldering methods and conditions other than those recommended below, contact an NEC Electronics sales representative.

For technical information, see the following website.

Semiconductor Device Mount Manual (http://www.necel.com/pkg/en/mount/index.html)

Caution Evaluation of the soldering conditions for the following products is incomplete because these products are under development.

- **64-pin plastic LQFP (GB-8EU type) of μPD780021A(A), 780022A(A), 780023A(A), 780024A(A), 780031A(A), 780032A(A), 780033A(A), 780034A(A)**
- **μPD780021AY(A), 780022AY(A) (except 64-pin plastic QFP (GC-AB8 type)), 780023AY(A), 780024AY(A), 780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A)**

Table 28-1. Surface Mounting Type Soldering Conditions (1/5)

(1) 64-pin plastic QFP (14 × 14)

μPD780021AGC-×××-AB8, 780022AGC-×××-AB8, 780023AGC-×××-AB8,
μPD780024AGC-×××-AB8, 780021AYGC-×××-AB8, 780022AYGC-×××-AB8,
μPD780023AYGC-×××-AB8, 780024AYGC-×××-AB8, 780031AGC-×××-AB8,
μPD780032AGC-×××-AB8, 780033AGC-×××-AB8, 780034AGC-×××-AB8,
μPD780031AYGC-×××-AB8, 780032AYGC-×××-AB8, 780033AYGC-×××-AB8,
μPD780034AYGC-×××-AB8, 780021AGC(A)-×××-AB8, 780022AGC(A)-×××-AB8,
μPD780023AGC(A)-×××-AB8, 780024AGC(A)-×××-AB8, 780022AYGC(A)-×××-AB8,
μPD780031AGC(A)-×××-AB8, 780032AGC(A)-×××-AB8, 780033AGC(A)-×××-AB8,
μPD780034AGC(A)-×××-AB8

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Three times or less	IR35-00-3
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Three times or less	VP15-00-3
Wave soldering	Solder bath temperature: 260°C max., Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature)	WS60-00-1
Partial heating	Pin temperature: 300°C max., Time: 3 seconds max. (per pin row)	–

Caution Do not use different soldering methods together (except for partial heating).

Table 28-1. Surface Mounting Type Soldering Conditions (2/5)

(2) 64-pin plastic LQFP (14 × 14)
μPD780021AGC-×××-8BS, 780022AGC-×××-8BS, 780023AGC-×××-8BS,
μPD780024AGC-×××-8BS, 780021AYGC-×××-8BS, 780022AYGC-×××-8BS,
μPD780023AYGC-×××-8BS, 780024AYGC-×××-8BS, 780031AGC-×××-8BS,
μPD780032AGC-×××-8BS, 780033AGC-×××-8BS, 780034AGC-×××-8BS,
μPD780031AYGC-×××-8BS, 780032AYGC-×××-8BS, 780033AYGC-×××-8BS,
μPD780034AYGC-×××-8BS, 780021AGC(A)-×××-8BS, 780022AGC(A)-×××-8BS,
μPD780023AGC(A)-×××-8BS, 780024AGC(A)-×××-8BS, 780031AGC(A)-×××-8BS,
μPD780032AGC(A)-×××-8BS, 780033AGC(A)-×××-8BS, 780034AGC(A)-×××-8BS,
μPD78F0034AGC-8BS, 78F0034AYGC-8BS, 78F0034BGC-8BS,
μPD78F0034BYGC-8BS, 78F0034BGC(A)-8BS, 78F0034BYGC(A)-8BS
64-pin plastic QFP (14 × 14)
μPD78F0034AGC-AB8[Note], 78F0034AYGC-AB8

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Two times or less	IR35-00-2
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Two times or less	VP15-00-2
Wave soldering	Solder bath temperature: 260°C max., Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature)	WS60-00-1
Partial heating	Pin temperature: 300°C max., Time: 3 seconds max. (per pin row)	–

Note Maintenance product

Caution Do not use different soldering methods together (except for partial heating).

Table 28-1. Surface Mounting Type Soldering Conditions (3/5)

(3) 64-pin plastic TQFP (12 × 12)
μPD780021AGK-×××-9ET, 780022AGK-×××-9ET, 780023AGK-×××-9ET,
μPD780024AGK-×××-9ET, 780021AYGK-×××-9ET, 780022AYGK-×××-9ET,
μPD780023AYGK-×××-9ET, 780024AYGK-×××-9ET, 780031AGK-×××-9ET,
μPD780032AGK-×××-9ET, 780033AGK-×××-9ET, 780034AGK-×××-9ET,
μPD780031AYGK-×××-9ET, 780032AYGK-×××-9ET, 780033AYGK-×××-9ET,
μPD780034AYGK-×××-9ET, 780021AGK(A)-×××-9ET, 780022AGK(A)-×××-9ET,
μPD780023AGK(A)-×××-9ET, 780024AGK(A)-×××-9ET, 780031AGK(A)-×××-9ET,
μPD780032AGK(A)-×××-9ET, 780033AGK(A)-×××-9ET, 780034AGK(A)-×××-9ET,
μPD78F0034AGK-9ET, 78F0034AYGK-9ET, 78F0034BGK-9ET, 78F0034BYGK-9ET,
μPD78F0034BGK(A)-9ET, 78F0034BYGK(A)-9ET

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Two times or less, Exposure limit: 7 days[Note] (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: Two times or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 hours)	VP15-107-2
Wave soldering	Solder bath temperature: 260°C max., Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature), Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 hours)	WS60-107-1
Partial heating	Pin temperature: 300°C max., Time: 3 seconds max. (per pin row)	–

Note After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

Caution Do not use different soldering methods together (except for partial heating).

Table 28-1. Surface Mounting Type Soldering Conditions (4/5)

(4) 64-pin plastic LQFP (10 × 10)
μPD780021AGB-×××-8EU, 780022AGB-×××-8EU, 780023AGB-×××-8EU,
μPD780024AGB-×××-8EU, 780021AYGB-×××-8EU, 780022AYGB-×××-8EU,
μPD780023AYGB-×××-8EU, 780024AYGB-×××-8EU, 780031AGB-×××-8EU,
μPD780032AGB-×××-8EU, 780033AGB-×××-8EU, 780034AGB-×××-8EU,
μPD780031AYGB-×××-8EU, 780032AYGB-×××-8EU, 780033AYGB-×××-8EU,
μPD780034AYGB-×××-8EU

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Two times or less	IR35-00-2
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Two times or less	VP15-00-2
Partial heating	Pin temperature: 300°C max., Time: 3 seconds max. (per pin row)	–

Caution Do not use different soldering methods together (except for partial heating).

(5) 64-pin plastic LQFP (10 × 10)
μPD78F0034AGB-8EU, 78F0034AYGB-8EU, 78F0034BGB-8EU, 78F0034BYGB-8EU,
μPD78F0034BGB(A)-8EU, 78F0034BYGB(A)-8EU

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Two times or less, Exposure limit: 7 days[Note] (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: Two times or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 hours)	VP15-107-2
Partial heating	Pin temperature: 300°C max., Time: 3 seconds max. (per pin row)	–

Note After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

Caution Do not use different soldering methods together (except for partial heating).

Table 28-1. Surface Mounting Type Soldering Conditions (5/5)

(6) 73-pin plastic FBGA (9 × 9)
μPD780021AF1-×××-CN3, 780022AF1-×××-CN3, 780023AF1-×××-CN3,
μPD780024AF1-×××-CN3, 780021AYF1-×××-CN3, 780022AYF1-×××-CN3,
μPD780023AYF1-×××-CN3, 780024AYF1-×××-CN3, 780031AF1-×××-CN3,
μPD780032AF1-×××-CN3, 780033AF1-×××-CN3, 780034AF1-×××-CN3,
μPD780031AYF1-×××-CN3, 780032AYF1-×××-CN3, 780033AYF1-×××-CN3,
μPD780034AF1-×××-CN3, 78F0034BF1-CN3, 78F0034BYF1-CN3

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 260°C, Time: 60 seconds max. (at 220°C or higher), Count: Three times or less, Exposure limit: 3 days[Note] (after that, prebake at 125°C for 20 hours)	IR60-203-3
VPS	Package peak temperature: 215°C, Time: 40 seconds max. (at 200°C or higher), Count: Three times or less, Exposure limit: 3 days[Note] (after that, prebake at 125°C for 20 hours)	VP15-203-3

Note After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

Caution Do not use different soldering methods together.

Table 28-2. Insertion Type Soldering Conditions

64-pin plastic SDIP (19.05 mm (750))
μPD780021ACW-×××, 780022ACW-×××, 780023ACW-×××, 780024ACW-×××,
μPD780021AYCW-×××, 780022AYCW-×××, 780023AYCW-×××, 780024AYCW-×××,
μPD780031ACW-×××, 780032ACW-×××, 780033ACW-×××, 780034ACW-×××,
μPD780031AYCW-×××, 780032AYCW-×××, 780033AYCW-×××, 780034AYCW-×××,
μPD780021ACW(A)-×××, 780022ACW(A)-×××, 780023ACW(A)-×××, 780024ACW(A)-×××,
μPD780031ACW(A)-×××, 780032ACW(A)-×××, 780033ACW(A)-×××, 780034ACW(A)-×××,
μPD78F0034ACW, 78F0034AYCW

Soldering Method	Soldering Conditions
Wave soldering (only for pins)	Solder bath temperature: 260°C max., Time: 10 seconds max.
Partial heating	Pin temperature: 300°C max., Time: 3 seconds max. (per pin row)

Caution Apply wave soldering only to the pins and be careful not to bring solder into direct contact with the package.

APPENDIX A DIFFERENCES BETWEEN μPD78018F, 780024A, 780034A, AND 780078 SUBSERIES

Tables A-1 and A-2 show the major differences between the μPD78018F, 780024A, 780034A, and 780078 Subseries.

Table A-1. Major Differences Between μPD78018F, 780024A, 780034A, and 780078 Subseries (Hardware)

Item \ Name		μPD78018F Subseries[Note]	μPD780024A, 780034A Subseries	μPD780078 Subseries
EMI noise reduction		Not provided	Provided	
Internal I^2C bus version (Y subseries)		Provided	Provided (multi-task supported)	
Flash memory version		μPD78F018F	μPD78F0034A, 78F0034B	μPD78F0078
ROM		8 KB to 60 KB	8 KB to 32 KB	48 KB, 60 KB
Internal high-speed RAM		512, 1024 bytes	512, 1024 bytes	1024 bytes
Internal expansion RAM		512, 1024 bytes	Not provided	1024 bytes
Minimum instruction execution time		0.4 μs (10 MHz)	0.24 μs (8.38 MHz), 0.16 μs (12 MHz, expanded-specification products only)	
Number of I/O ports		53	51	52
Timer		16 bits: 1, 8 bits: 2, Watch timer: 1, Watchdog timer: 1	16 bits: 1, 8 bits: 2, Watch timer: 1, Watchdog timer: 1	16 bits: 2, 8 bits: 2, Watch timer: 1, Watchdog timer: 1
A/D converter		8 bits × 8	• 8 bits × 8 (μPD780024A Subseries) • 10 bits × 8 (μPD780034A Subseries)	10 bits × 8
Serial interface	Subseries without suffix Y	3-wire/2-wire/SBI: 1, 3-wire (automatic transmission/reception): 1	3-wire: 2, UART: 1	3-wire: 1, UART: 1, 3-wire/UART: 1
	Subseries with suffix Y	3-wire/2-wire/I^2C: 1, 3-wire (automatic transmission/reception): 1	3-wire: 2, UART: 1, Multi-master I^2C: 1	3-wire: 1, UART: 1, 3-wire/UART: 1, Multi-master I^2C: 1
Timer output		3 (14-bit PWM output possible: 2)	3 (8-bit PWM output possible: 2)	4 (8-bit PWM output possible: 2)
Package		• 64-pin SDIP (19.05 mm (750)) • 64-pin QFP (14 × 14) • 64-pin LQFP (12 × 12)	• 64-pin SDIP (19.05 mm (750)) • 64-pin QFP (14 × 14) • 64-pin TQFP (12 × 12) • 64-pin LQFP (14 × 14) • 64-pin LQFP (10 × 10) • 73-pin FBGA (9 × 9)	• 64-pin QFP (14 × 14) • 64-pin TQFP (12 × 12) • 64-pin LQFP (14 × 14)
Device file		DF78014	DF780034	DF780078
Emulation board		IE-78014-R-EM-A, IE-78018-NS-EM1	IE-780034-NS-EM1	IE-780078-NS-EM1
Electrical specifications Recommended soldering conditions		Refer to the data sheet or user's manual (with electrical specifications) of each product.		

Note Maintenance product

Table A-2. Major Differences Between μPD78018F, 780024A, 780034A, and 780078 Subseries (Software) (1/2)

<table>
<tr><th colspan="2">Item \ Name</th><th>μPD78018F Subseries[Note 1]</th><th>μPD780024A, 780034A Subseries</th><th colspan="2">μPD780078 Subseries</th></tr>
<tr><td colspan="2" rowspan="2">A/D converter</td><td rowspan="2">–</td><td colspan="3">Take the appropriate measures for the first A/D conversion result immediately after the A/D conversion operation is started (ADCS0 is set to 1), such as discarding it, because it may not satisfy the rating.</td></tr>
<tr><td></td><td colspan="2">However, if a wait time of 14 μs (MIN.) has been secured after ADCE0 was set to 1 before starting operation (ADCS0 is set to 1), the first data can be used.</td></tr>
<tr><td colspan="2" rowspan="2">16-bit timer/event counter</td><td>1 ch</td><td>1 ch</td><td colspan="2">2 ch</td></tr>
<tr><td>TM0</td><td>TM0</td><td>TM00</td><td>TM01</td></tr>
<tr><td></td><td>Interval timer
PWM output
PPG output
Pulse width measurement
External event counter
Square wave output</td><td>√
√
–
√
√
√</td><td>√
–
√
√
√
√</td><td colspan="2">√
–
√
√
√
√</td></tr>
<tr><td></td><td>Count clock</td><td>fx/2, fx/2², fx/2³, TI0</td><td>fx, fx/2², fx/2⁶, TI00</td><td>fx, fx/2², fx/2⁶, TI000</td><td>fx/2, fx/2³ fx/2⁹, TI001</td></tr>
<tr><td></td><td>Control register
Output control register
Compare/capture register
Prescaler mode register
Capture/compare control register</td><td>TMC0
TOC0
CR00, CR01 (Capture only)
TCL0[Note 2]
–</td><td>TMC0
TOC0
CR00, CR01
PRM0
CRC0</td><td>TMC00
TOC00
CR000, CR010
PRM00
CRC00</td><td>TMC01
TOC01
CR001, CR011
PRM01
CRC01</td></tr>
<tr><td></td><td>Interrupt</td><td>INTTM0</td><td>INTTM00, INTTM01</td><td>INTTM000, INTTM010</td><td>INTTM001, INTTM011</td></tr>
</table>

Notes 1. Maintenance product

2. TCL0: Timer clock select register 0

Table A-2. Major Differences Between μPD78018F, 780024A, 780034A, and 780078 Subseries (Software) (2/2)

Item \ Name	μPD78018F Subseries[Note]		μPD780024A, 780034A Subseries	μPD780078 Subseries
8-bit timer/event counter	2 ch		2 ch	
	TM1	TM2	TM50	TM51
Unit mode				
Interval timer	√		√	√
External event counter	√		√	√
Square wave output	√		√	√
PWM output	–		√	√
Cascade connection mode				
Interval timer	√		√	√
External event counter	√		√	√
Square wave output	√		√	√
Count clock	fx/2^2, fx/2^3, fx/2^4, fx/2^5, fx/2^6, fx/2^7, fx/2^8, fx/2^9, fx/2^{10},fx2^{12}, TI1	fx/2^2, fx/2^3, fx/2^4, fx/2^5, fx/2^6, fx/2^7, fx/2^8, fx/2^9, fx/2^{10}, fx/2^{12}, TI2	fx, fx/2^2, fx/2^4, fx/2^6, fx/2^8, fx/2^{10}, TI50	fx/2, fx/2^3, fx/2^5, fx/2^7, fx/2^9, fx/2^{11}, TI51
Control register	TMC1		TMC50	TMC51
Output control register	TOC1		TMC50	TMC51
Clock select register	TCL1		TCL50	TCL51
Interrupt	INTTM1	INTTM2	INTTM50	INTTM51

Note Maintenance product

APPENDIX B DEVELOPMENT TOOLS

The following development tools are available for the development of systems that employ the μPD780024A, 780034A, 780024AY, and 780034AY Subseries.

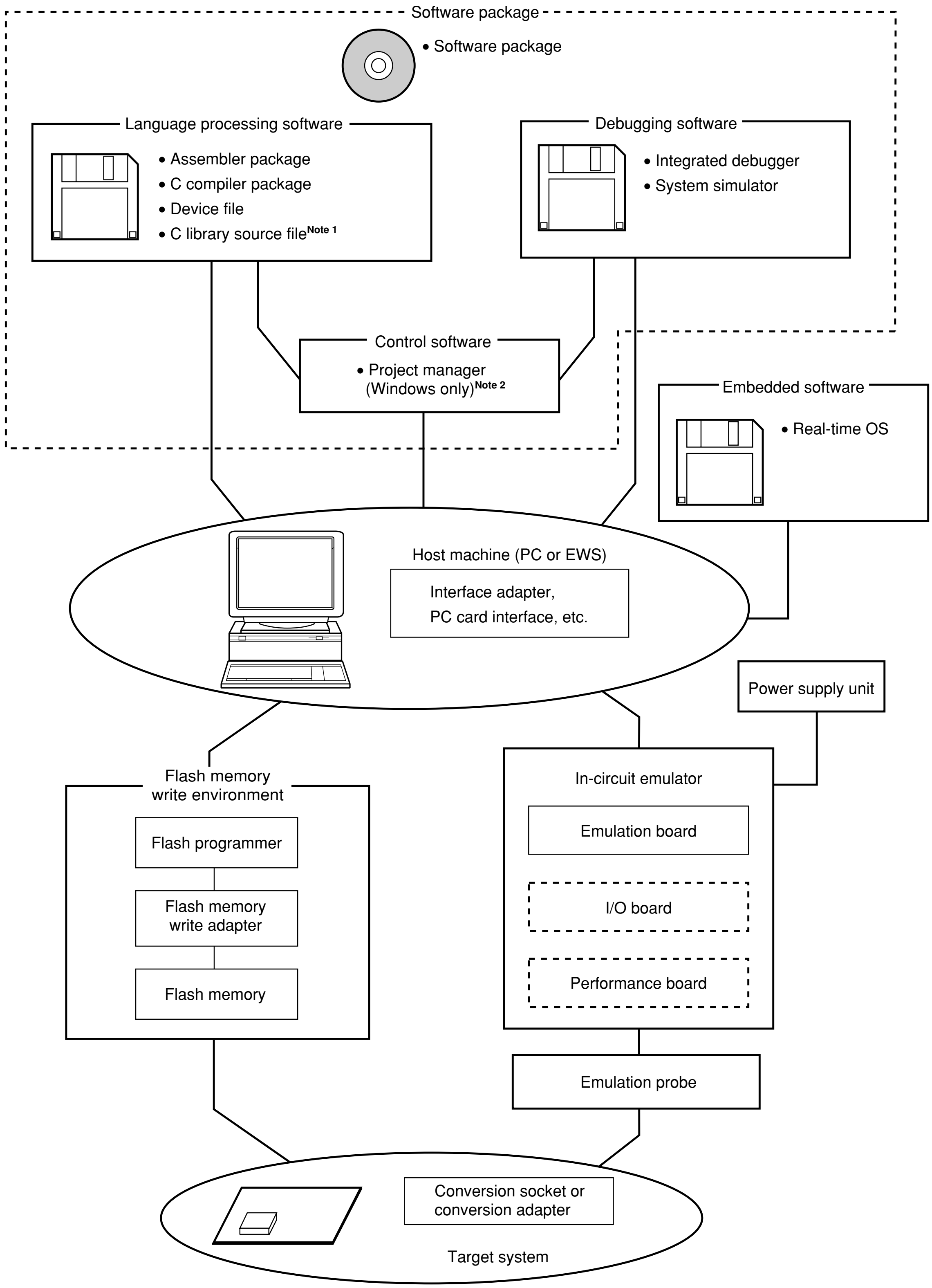
Figure B-1 shows the development tool configuration.

- **Support for PC98-NX series**
  Unless otherwise specified, products compatible with IBM PC/AT™ computers are compatible with PC98-NX series computers. When using PC98-NX series computers, refer to the explanation for IBM PC/AT computers.

- **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. Development Tool Configuration (1/2)**

(1) When using the in-circuit emulator IE-78K0-NS, IE-78K0-NS-A

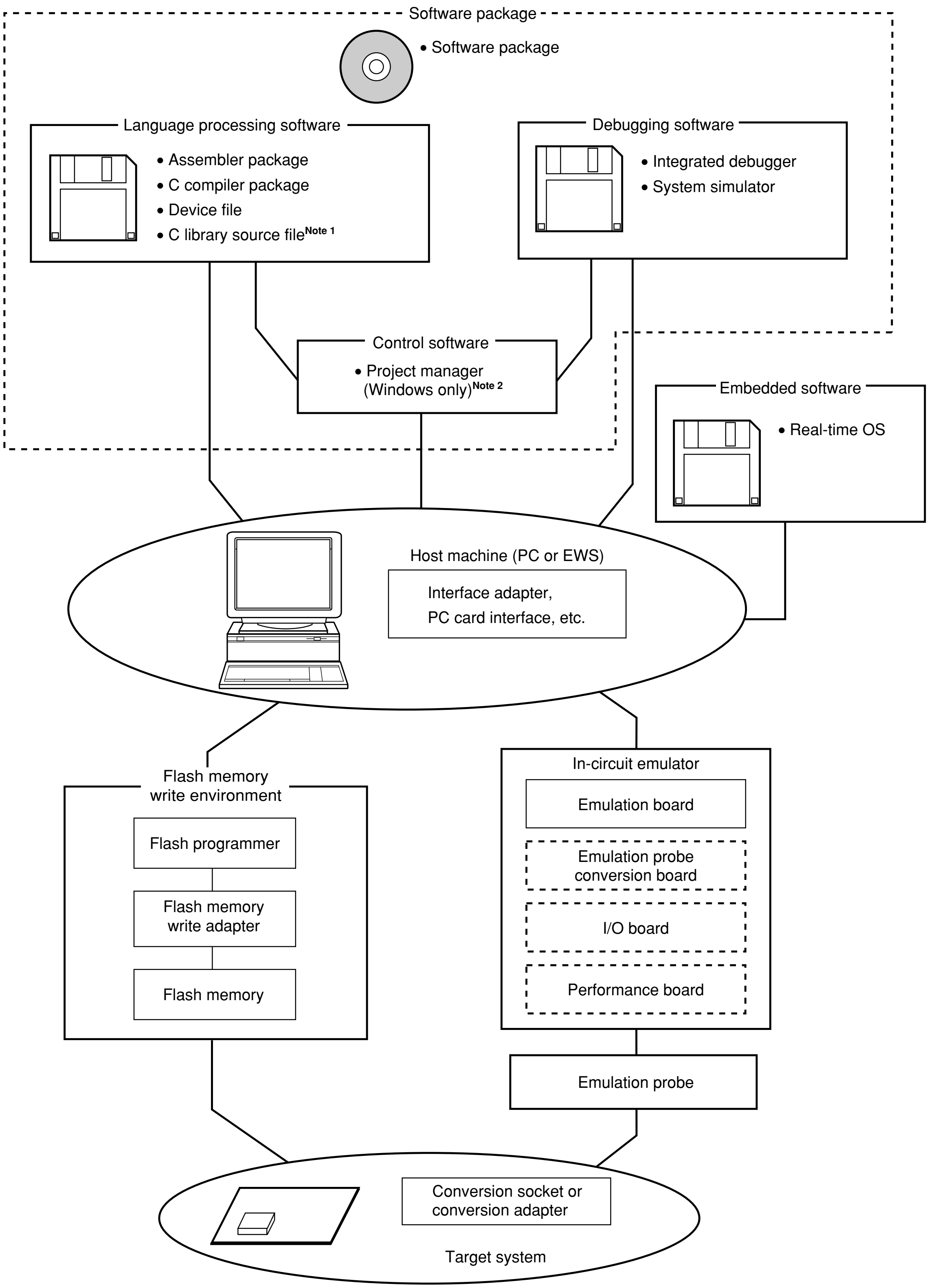


Notes 1. The C library source file is not included in the software package.

2. The project manager is included in the assembler package.
The project manager is only used for Windows.

★ **Figure B-1. Development Tool Configuration (2/2)**

(2) When using the in-circuit emulator IE-78001-R-A



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: μS××××SP78K0

Remark ×××× in the part number differs depending on the OS used.

μS××××SP78K0

××××	Host Machine	OS	Supply Medium
AB17	PC-9800 series, IBM PC/AT compatibles	Windows (Japanese version)	CD-ROM
BB17		Windows (English version)	

B.2 Language Processing Software

RA78K0 Assembler package	This assembler converts programs written in mnemonics into object codes executable with a microcontroller. Further, this assembler is provided with functions capable of automatically creating symbol tables and branch instruction optimization. This assembler should be used in combination with device file (DF780024 or DF780034) (sold separately). **<Precaution when using RA78K0 in PC environment>** This assembler package is a DOS-based application. It can also be used in Windows, however, by using the project manager (included in the assembler package) on Windows.
	Part Number: μS××××RA78K0
CC78K0 C compiler package	This compiler converts programs written in C language into object codes executable with a microcontroller. This compiler should be used in combination with an assembler package and device file (both sold separately). **<Precaution when using CC78K0 in PC environment>** This C compiler package is a DOS-based application. It can also be used in Windows, however, by using the project manager (included in the assembler package) on Windows.
	Part Number: μS××××CC78K0
DF780024**Note 1** DF780034**Note 1** Device file	This file contains information peculiar to the device. This device file should be used in combination with tool (RA78K0, CC78K0, SM78K0, ID78K0-NS, and RX78K0) (all sold separately). The corresponding OS and host machine differ depending on the tool used. • DF780024: μPD780024A, 780024AY, 780024AS Subseries • DF780034: μPD780034A, 780034AY, 780034AS Subseries
	Part Number: μS××××DF780024, μS××××DF780034
CC78K0-L**Note 2** C library source file	This is a source file of functions configuring the object library included in the C compiler package. This file is required to match the object library included in C compiler package to the user's specifications. It does not depend on the operating environment because it is a source file.
	Part Number: μS××××CC78K0-L

Notes 1. The DF780024 and DF780034 can be used in common with the RA78K0, CC78K0, SM78K0, ID78K0-NS, and RX78K0.

2. CC78K0-L is not included in the software package (SP78K0).

★ **Remark** ×××× in the part number differs depending on the host machine and OS used.

μS××××RA78K0
μS××××CC78K0

××××	Host Machine	OS	Supply Medium
AB13	PC-9800 series, IBM PC/AT compatibles	Windows (Japanese version)	3.5-inch 2HD FD
BB13		Windows (English version)	
AB17		Windows (Japanese version)	CD-ROM
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)	

μS××××DF780024
μS××××DF780034
μS××××CC78K0-L

××××	Host Machine	OS	Supply Medium
AB13	PC-9800 series, IBM PC/AT compatibles	Windows (Japanese version)	3.5-inch 2HD FD
BB13		Windows (English version)	
3P16	HP9000 series 700	HP-UX (Rel. 10.10)	DAT
3K13	SPARCstation	SunOS (Rel. 4.1.4), Solaris (Rel. 2.5.1)	3.5-inch 2HD FD
3K15			1/4-inch CGMT

B.3 Control Software

Project manager	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. **<Caution>** The project manager is included in the assembler package (RA78K0). It can only be used in Windows.

★ ## B.4 Flash Memory Writing Tools

Flashpro III (part number: FL-PR3, PG-FP3) Flashpro IV (part number: FL-PR4, PG-FP4) Flash programmer	Flash programmer dedicated to microcontrollers with on-chip flash memory.
FA-64CW FA-64GC-8BS-A FA-64GC FA-64GK-9ET FA-64GB-8EU-A FA-73F1-CN3-A Flash memory writing adapter	Flash memory writing adapter used connected to the Flashpro III and Flashpro IV. • FA-64CW: 64-pin plastic SDIP (CW type) • FA-64GC-8BS-A: 64-pin plastic LQFP (GC-8BS type) • FA-64GC: 64-pin plastic QFP (GC-AB8 type) • FA-64GK-9ET: 64-pin plastic TQFP (GK-9ET type) • FA-64GB-8EU-A: 64-pin plastic LQFP (GB-8EU type) • FA-73F1-CN3-A: 73-pin plastic FBGA (F1-CN3 type)

Remark FL-PR3, FL-PR4, FA-64CW, FA-64GC-8BS-A, FA-64GC, FA-64GK-9ET, FA-64GB-8EU-A, and FA-73F1-CN3-A are products of Naito Densei Machida Mfg. Co., Ltd.
Contact: +81-45-475-4191 Naito Densei Machida Mfg. Co., Ltd.

B.5 Debugging Tools (Hardware)

B.5.1 When using the in-circuit emulator IE-78K0-NS, IE-78K0-NS-A

(1/2)

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 the 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 connected to the IE-78K0-NS to expand its functions. Adding this board adds a coverage function and enhances debugging functions such as tracer and timer functions.
IE-78K0-NS-A In-circuit emulator	A combination of 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 receptacle of 100 V to 240 V AC.
IE-70000-98-IF-C Interface adapter	This adapter is required when using a PC-9800 series computer (except notebook type) as the 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 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 host machine (ISA bus compatible).
IE-70000-PCI-IF-A Interface adapter	This adapter is required when using a computer with a PCI bus as the host machine.
IE-780034-NS-EM1 Emulation board	This board emulates the operations of the peripheral hardware peculiar to a device. It should be used in combination with an in-circuit emulator.
NP-64CW NP-H64CW Emulation probe	This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic SDIP (CW type).

(2/2)

NP-64GC Emulation probe		This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic QFP (GC-AB8 type) and 64-pin plastic LQFP (GC-8BS type).
	EV-9200GC-64 Conversion socket (see **Figures B-2** and **B-3**)	This conversion socket connects the NP-64GC to a target system board designed for a 64-pin plastic QFP (GC-AB8 type) and 64-pin plastic LQFP (GC-8BS type).
NP-64GC-TQ ★ NP-H64GC-TQ Emulation probe		This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic QFP (GC-AB8 type) and 64-pin plastic LQFP (GC-8BS type).
	TGC-064SAP Conversion adapter (see **Figure B-4**)	This conversion adapter connects the NP-64GC-TQ or NP-H64GC-TQ to a target system board designed for a 64-pin plastic QFP (GC-AB8 type) and 64-pin plastic LQFP (GC-8BS type).
NP-64GK ★ NP-H64GK-TQ Emulation probe		This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic TQFP (GK-9ET type).
★	TGK-064SBW Conversion adapter (see **Figure B-5**)	This conversion adapter connects the NP-64GK or NP-H64GK-TQ to a target system board designed for a 64-pin plastic TQFP (GK-9ET type).
NP-H64GB-TQ Emulation probe		This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic LQFP (GB-8EU type).
	TGB-064SDP Conversion adapter (see **Figure B-6**)	This conversion adapter connects the NP-H64GB-TQ to a target system board designed for a 64-pin plastic LQFP (GB-8EU type).
NP-73F1-CN3 Emulation probe		This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 73-pin plastic FBGA (F1-CN3 type).
	CSICE73A0909N01, LSPACK73A0909N01, CSSOCKET73A0909N01 Conversion socket	This conversion socket connects the NP-73F1-CN3 to a target system board designed for a 73-pin plastic FBGA (F1-CN3 type). • CSICE73A0909N01: YQSOCKET/LSPACK conversion adapter • LSPACK73A0909N01: Socket for target connection • CSSOCKET73A0909N01: Socket for emulator connection

Remarks 1. NP-64CW, NP-H64CW, NP-64GC, NP-64GC-TQ, NP-H64GC-TQ, NP-64GK, NP-H64GK-TQ, NP-H64GB-TQ, and NP-73F1-CN3 are products of Naito Densei Machida Mfg. Co., Ltd.
Contact: +81-45-475-4191 Naito Densei Machida Mfg. Co., Ltd.

2. TGC-064SAP, TGK-064SBW, TGB-064SDP, CSICE73A0909N01, LSPACK73A0909N01, and CSSOCKET73A0909N01 are products of TOKYO ELETECH CORPORATION.
Contact: Daimaru Kogyo, Ltd. Phone: Tokyo +81-3-3820-7112 Electronics Dept.
Osaka +81-6-6244-6672 Electronics 2nd Dept.

3. EV-9200GC-64 is sold in five-unit sets.

4. TGK-064SBW and TGC-064SAP are sold in single units.

5. The emulation probe (NP-73F1-CN3) is supplied with a conversion socket (CSICE73A0909N01, LSPACK73A0909N01, CSSOCKET73A0909N01).

B.5.2 When using the in-circuit emulator IE-78001-R-A

IE-78001-R-A 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 the integrated debugger (ID78K0). This emulator should be used in combination with an emulation probe and interface adapter, which is required to connect this emulator to the host machine.
IE-70000-98-IF-C Interface adapter	This adapter is required when using a PC-9800 series computer (except notebook type) as the host machine (C bus compatible).
IE-70000-PC-IF-C Interface adapter	This adapter is required when using an IBM PC/AT compatible computer as the host machine (ISA bus compatible).
IE-70000-PCI-IF-A Interface adapter	This adapter is required when using a computer with a PCI bus as the host machine.
IE-780034-NS-EM1 Emulation board	This board emulates the operations of the peripheral hardware peculiar to a device. It should be used in combination with an in-circuit emulator and emulation probe conversion board.
IE-78K0-R-EX1 Emulation probe conversion board	This board is required when using the IE-780034-NS-EM1 on the IE-78001-R-A.
EP-78240CW-R[Note] Emulation probe	This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic SDIP (CW type).
EP-78240GC-R[Note] Emulation probe	This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic QFP (GC-AB8 type) and 64-pin plastic LQFP (GC-8BS type).
EV-9200GC-64 Conversion socket (see **Figures B-2** and **B-3**)	This conversion socket connects the EP-78240GC-R to a target system board designed for a 64-pin plastic QFP (GC-AB8 type) and 64-pin plastic LQFP (GC-8BS type).
EP-78012GK-R Emulation probe	This probe is used to connect the in-circuit emulator to a target system and is designed for use with a 64-pin plastic TQFP (GK-9ET type).
★ TGK-064SBW Conversion adapter (see **Figure B-5**)	This conversion adapter connects the EP-78012GK-R to a target system board designed for a 64-pin plastic TQFP (GK-9ET type).

Note Maintenance product

Caution The IE-78001-R-A is not supported for the 64-pin plastic LQFP (GB-8EU type) and 73-pin plastic FBGA (F1-CN3 type).

Remarks 1. TGK-064SBW is a product of TOKYO ELETECH CORPORATION.
Contact: Daimaru Kogyo, Ltd. Phone: Tokyo +81-3-3820-7112 Electronics Dept.
Osaka +81-6-6244-6672 Electronics 2nd Dept.

2. EV-9200GC-64 is sold in five-unit sets.

3. TGK-064SBW 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 the device file (DF780024 or DF780034) (sold separately).
	Part Number: μS××××SM78K0
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 the device file (sold separately).
ID78K0 Integrated debugger (supporting in-circuit emulator IE-78001-R-A)	Part Number: μS××××ID78K0-NS, μS××××ID78K0

★ **Remark** ×××× in the part number differs depending on the host machine and OS used.

μS××××SM78K0
μS××××ID78K0-NS
μS××××ID78K0

××××	Host Machine	OS	Supply Medium
AB13	IBM PC/AT 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	RX78K0 is a real-time OS conforming to the μITRON specifications. A tool (configurator) for generating the nucleus of RX78K0 and multiple information tables is supplied. It is used in combination with an assembler package (RA78K0) and device file (DF780024 or DF780034) (both sold separately). **<Precaution when using RX78K0 in PC environment>** The real-time OS is a DOS-based application. It should be used from the DOS prompt when using in Windows.
	Part number: μS××××RX78013-ΔΔΔΔ

Caution When purchasing the RX78K0, fill in the purchase application form in advance and sign the user agreement.

Remark ×××× and ΔΔΔΔ in the part number differ depending on the host machine and OS used.

μS××××RX78013-ΔΔΔΔ

ΔΔΔΔ	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

★

××××	Host Machine	OS	Supply Medium
AA13	PC-9800 series	Windows (Japanese version)	3.5-inch 2HD FD
AB13	IBM PC/AT compatibles	Windows (Japanese version)	
BB13		Windows (English version)	

B.8 System Upgrade 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.

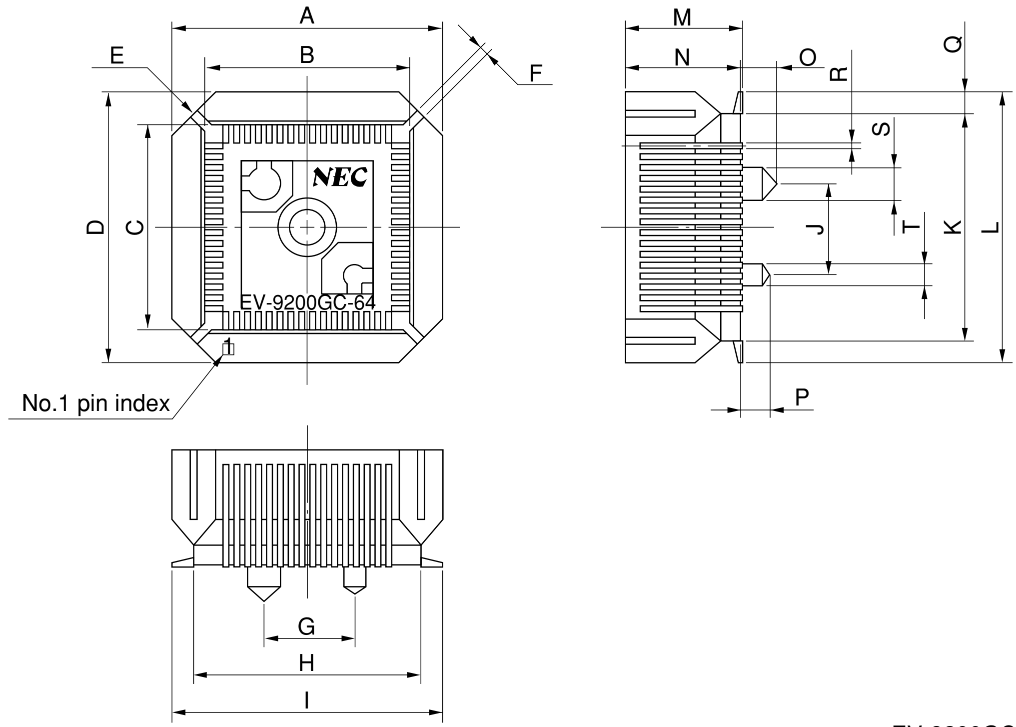
Table B-1. System Upgrade Method from Former In-Circuit Emulator for 78K/0 Series to IE-78001-R-A

In-Circuit Emulator Owned	In-Circuit Emulator Cabinet Upgrade[Note]	Board to Be Purchased
IE-78000-R	Required	IE-78001-R-BK
IE-78000-R-A	Not required	

Note For upgrading of a cabinet, send your in-circuit emulator to NEC Electronics.

B.9 Package Drawings of Conversion Socket and Conversion Adapter

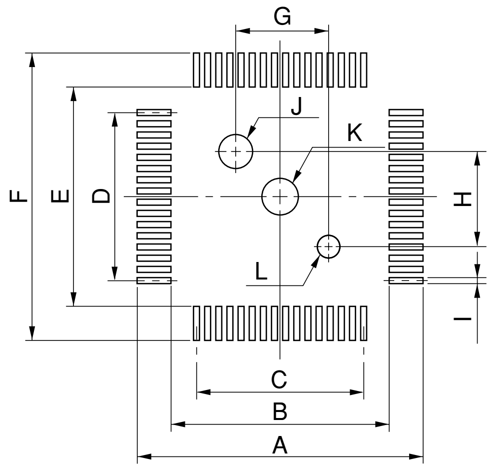
Figure B-2. EV-9200GC-64 Package Drawing (for Reference Only)



EV-9200GC-64-G0

ITEM	MILLIMETERS	INCHES
A	18.8	0.74
B	14.1	0.555
C	14.1	0.555
D	18.8	0.74
E	4-C 3.0	4-C 0.118
F	0.8	0.031
G	6.0	0.236
H	15.8	0.622
I	18.5	0.728
J	6.0	0.236
K	15.8	0.622
L	18.5	0.728
M	8.0	0.315
N	7.8	0.307
O	2.5	0.098
P	2.0	0.079
Q	1.35	0.053
R	0.35±0.1	$0.014^{+0.004}_{-0.005}$
S	ϕ2.3	ϕ0.091
T	ϕ1.5	ϕ0.059

Figure B-3. EV-9200GC-64 Recommended Board Mounting Pattern (for Reference Only)

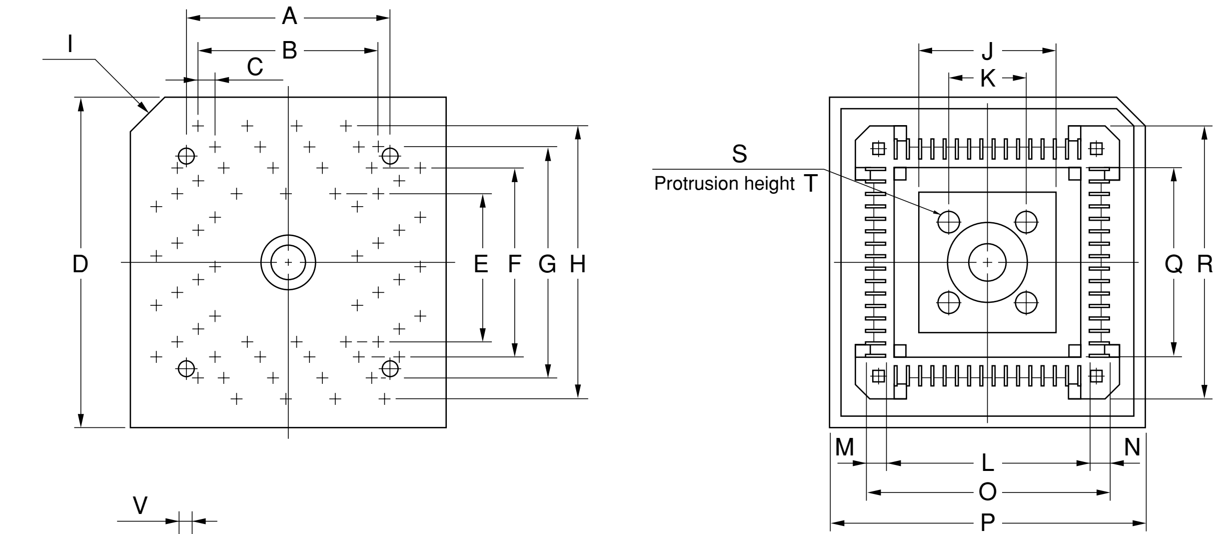


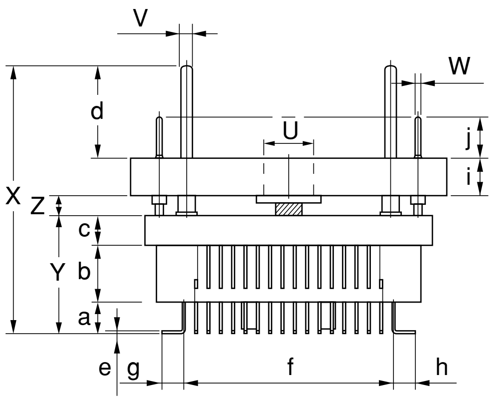
EV-9200GC-64-P1E

ITEM	MILLIMETERS	INCHES
A	19.5	0.768
B	14.8	0.583
C	0.8–0.02 × 15=12.0–0.05	$0.031^{+0.002}_{-0.001}$ × 0.591=$0.472^{+0.003}_{-0.002}$
D	0.8–0.02 × 15=12.0–0.05	$0.031^{+0.002}_{-0.001}$ × 0.591=$0.472^{+0.003}_{-0.002}$
E	14.8	0.583
F	19.5	0.768
G	6.00–0.08	$0.236^{+0.004}_{-0.003}$
H	6.00–0.08	$0.236^{+0.004}_{-0.003}$
I	0.5–0.02	$0.197^{+0.001}_{-0.002}$
J	ϕ2.36–0.03	$\phi 0.093^{+0.001}_{-0.002}$
K	ϕ2.2–0.1	$\phi 0.087^{+0.004}_{-0.005}$
L	ϕ1.57–0.03	$\phi 0.062^{+0.001}_{-0.002}$

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 Mount Manual" website (http://www.necel.com/pkg/en/mount/index.html).

★ **Figure B-4. TGC-064SAP Package Drawing (for Reference Only)**





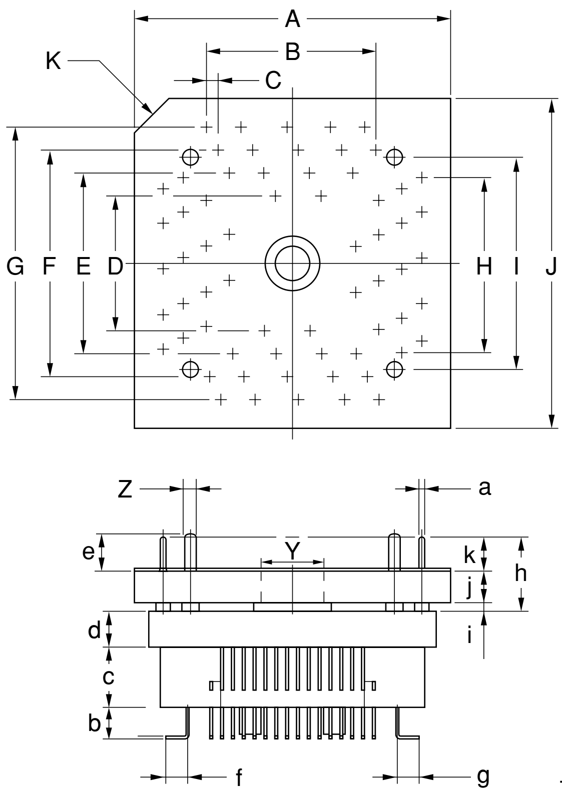
ITEM	MILLIMETERS	INCHES
A	14.12	0.556
B	0.8x15=12.0	0.031x0.591=0.472
C	0.8	0.031
D	20.65	0.813
E	10.0	0.394
F	12.4	0.488
G	14.8	0.583
H	17.2	0.677
I	C 2.0	C 0.079
J	9.05	0.356
K	5.0	0.197
L	13.35	0.526
M	1.325	0.052
N	1.325	0.052
O	16.0	0.630
P	20.65	0.813
Q	12.5	0.492
R	17.5	0.689
S	4-ϕ1.3	4-ϕ0.051
T	1.8	0.071
U	ϕ3.55	ϕ0.140
V	ϕ0.9	ϕ0.035
W	ϕ0.3	ϕ0.012
X	(19.65)	(0.667)
Y	7.35	0.289
Z	1.2	0.047

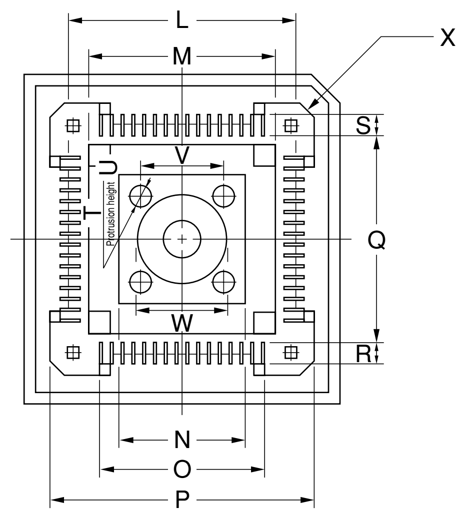
ITEM	MILLIMETERS	INCHES
a	1.85	0.073
b	3.5	0.138
c	2.0	0.079
d	6.0	0.236
e	0.25	0.010
f	13.6	0.535
g	1.2	0.047
h	1.2	0.047
i	2.4	0.094
j	2.7	0.106

TGC-064SAP-G0E

note: Product by TOKYO ELETECH CORPORATION.

★ **Figure B-5. TGK-064SBW Package Drawing (for Reference Only)**





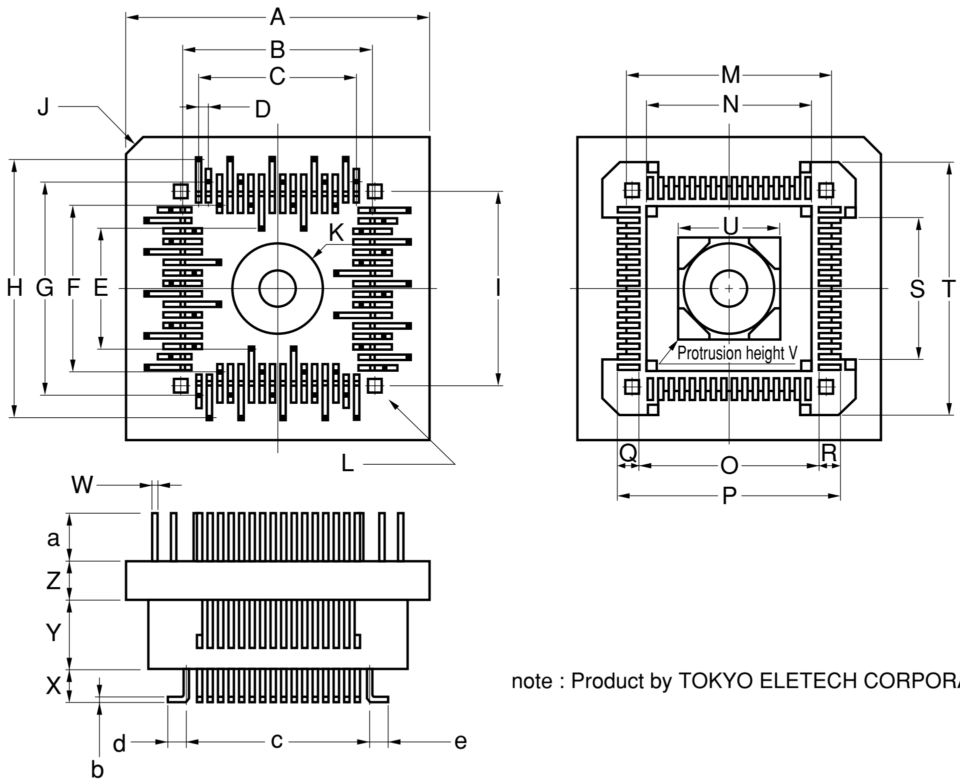
ITEM	MILLIMETERS	INCHES
A	18.4	0.724
B	0.65x15=9.75	0.026x0.591=0.384
C	0.65	0.026
D	7.75	0.305
E	10.15	0.400
F	12.55	0.494
G	14.95	0.589
H	0.65x15=9.75	0.026x0.591=0.384
I	11.85	0.467
J	18.4	0.724
K	C 2.0	C 0.079
L	12.45	0.490
M	10.25	0.404
N	7.7	0.303
O	10.02	0.394
P	14.92	0.587
Q	11.1	0.437
R	1.45	0.057
S	1.45	0.057
T	4-ϕ1.3	4-ϕ0.051
U	1.8	0.071
V	5.0	0.197
W	ϕ5.3	ϕ0.209
X	4-C 1.0	4-C 0.039
Y	ϕ3.55	ϕ0.140
Z	ϕ0.9	ϕ0.035

ITEM	MILLIMETERS	INCHES
a	ϕ0.3	ϕ0.012
b	1.85	0.073
c	3.5	0.138
d	2.0	0.079
e	3.9	0.154
f	1.325	0.052
g	1.325	0.052
h	5.9	0.232
i	0.8	0.031
j	2.4	0.094
k	2.7	0.106

TGK-064SBW-G1E

note: Product by TOKYO ELETECH CORPORATION.

★ **Figure B-6. TGB-064SDP Package Drawing (for Reference Only) (1/2)**



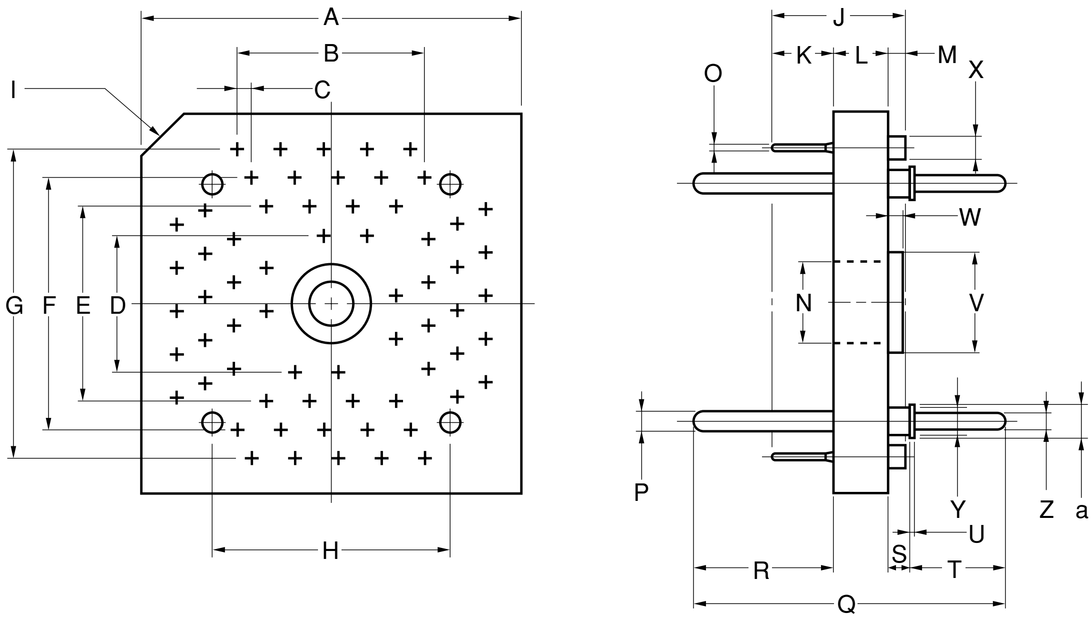
note : Product by TOKYO ELETECH CORPORATION

ITEM	MILLIMETERS	INCHES
A	14.0	0.551
B	9.24	0.364
C	0.5×15=7.5	0.020×0.591=0.295
D	0.5	0.020
E	5.64	0.222
F	8.04	0.317
G	10.44	0.411
H	12.84	0.506
I	9.77	0.385
J	C 1.0	C 0.039
K	ϕ4.3	ϕ0.169
L	4- 0.7	4- 0.028
M	10.1	0.398
N	8.0	0.315
O	8.99	0.354
P	12.0	0.472
Q	1.505	0.059
R	1.505	0.059
S	6.75	0.266
T	12.9	0.508
U	5.0	0.197
V	1.8	0.071
W	0.25	0.010
X	1.85	0.073
Y	3.5	0.138
Z	2.0	0.079

ITEM	MILLIMETERS	INCHES
a	2.6	0.102
b	0.25	0.010
c	9.24	0.364
d	1.38	0.054
e	1.38	0.054

TGB-064SDP-G1E-1

★ **Figure B-6. TGB-064SDP Package Drawing (for Reference Only) (2/2)**



note : Product by TOKYO ELETECH CORPORATION

ITEM	MILLIMETERS	INCHES
A	16.0	0.630
B	0.5×15=7.5	0.020×0.591=0.295
C	0.5	0.020
D	5.64	0.222
E	8.04	0.317
F	10.44	0.411
G	12.84	0.506
H	9.77	0.385
I	C 2.0	C 0.079
J	5.9	0.232
K	2.7	0.106
L	2.4	0.094
M	0.8	0.031
N	ϕ3.5	ϕ0.138
O	ϕ0.3	ϕ0.012
P	ϕ0.9	ϕ0.035
Q	13.6	0.535
R	6.0	0.236
S	1.0	0.039
T	4.2	0.165
U	0.2	0.008
V	ϕ4.0	ϕ0.157
W	0.7	0.028
X	ϕ1.03	ϕ0.041
Y	ϕ1.1	ϕ0.043
Z	ϕ0.7	ϕ0.028

ITEM	MILLIMETERS	INCHES
a	ϕ1.4	ϕ0.055

TGB-064SDP-G1E-2

★ # APPENDIX C NOTES ON TARGET SYSTEM DESIGN

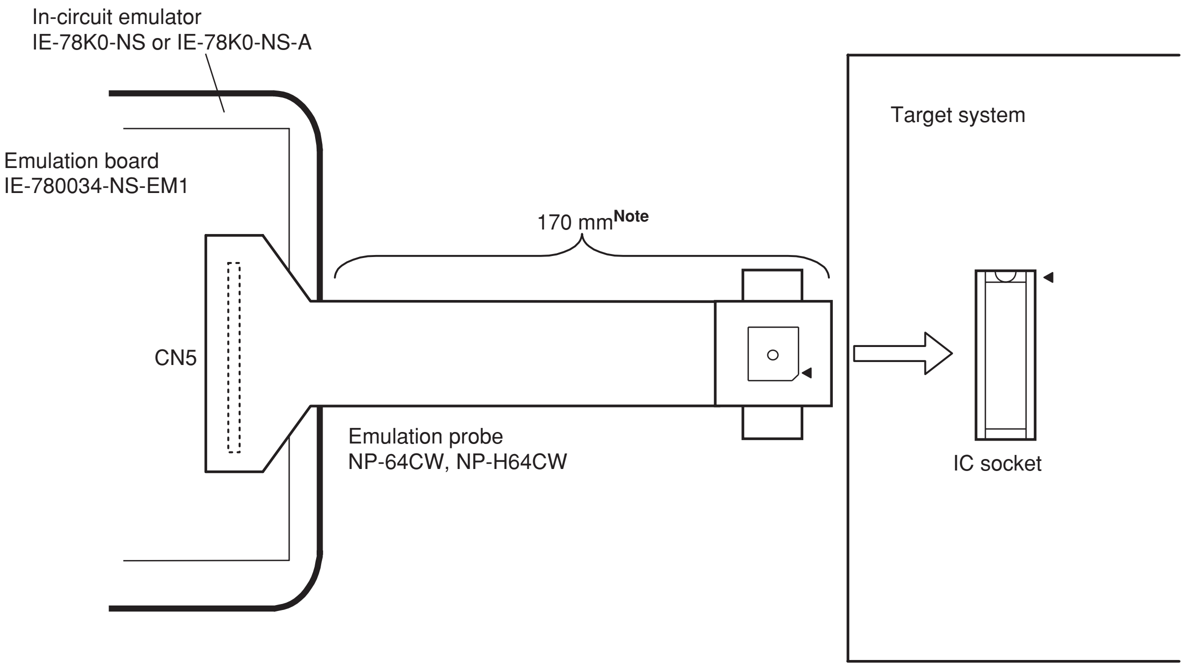
The following shows a diagram of the connection conditions between the emulation probe and conversion adapter. Design your system making allowances for conditions such as the shape of parts mounted on the target system, as shown below.

Of the products described in this chapter, all the emulation probes are products of Naito Densei Machida Mfg. Co., Ltd., and all the conversion adapters are products of TOKYO ELETECH CORPORATION.

Table C-1. Distance Between IE System and Conversion Adapter

Emulation Probe	Conversion Adapter	Distance Between IE System and Conversion Adapter
NP-64CW	–	170 mm
NP-H64CW		370 mm
NP-64GC-TQ	TGC-064SAP	155 mm
NP-H64GC-TQ		355 mm
NP-64GK	TGK-064SBW	155 mm
NP-H64GK-TQ		355 mm
NP-64GB-TQ	TGB-064SDP	155 mm
NP-H64GB-TQ		355 mm
NP-73F1-CN3	CSICE73A0909N01, LSPACK73A0909N01, CSSOCKET73A0909N01	213 mm

Figure C-1. Distance Between In-Circuit Emulator and Conversion Adapter (When Using 64CW)



Note Distance when using NP-64CW. This is 370 mm when using NP-H64CW.

Figure C-2. Connection Conditions of Target System (When Using NP-64CW)

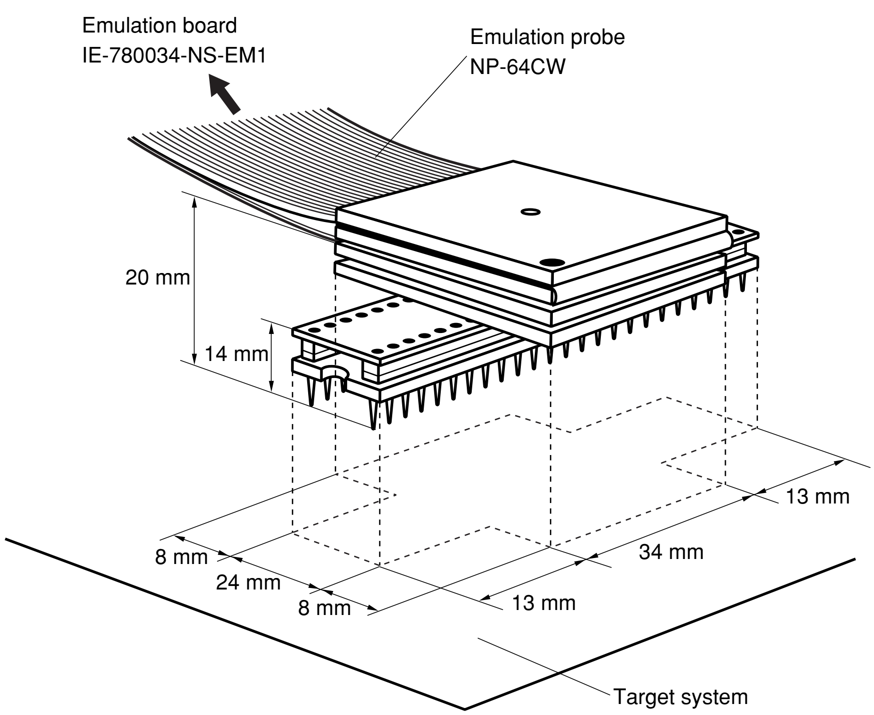


Figure C-3. Connection Conditions of Target System (When Using NP-H64CW)

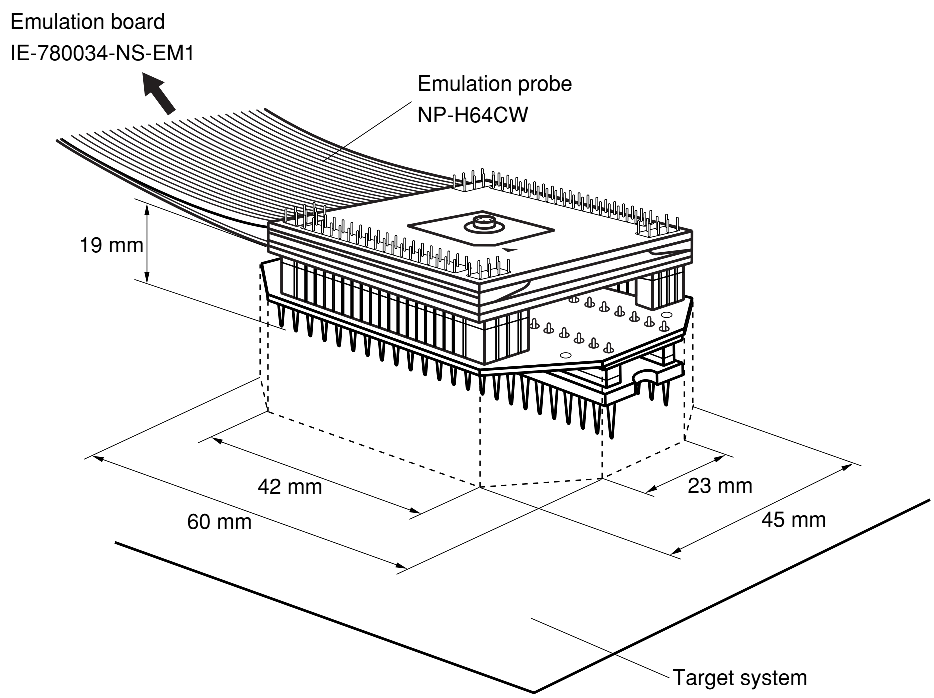
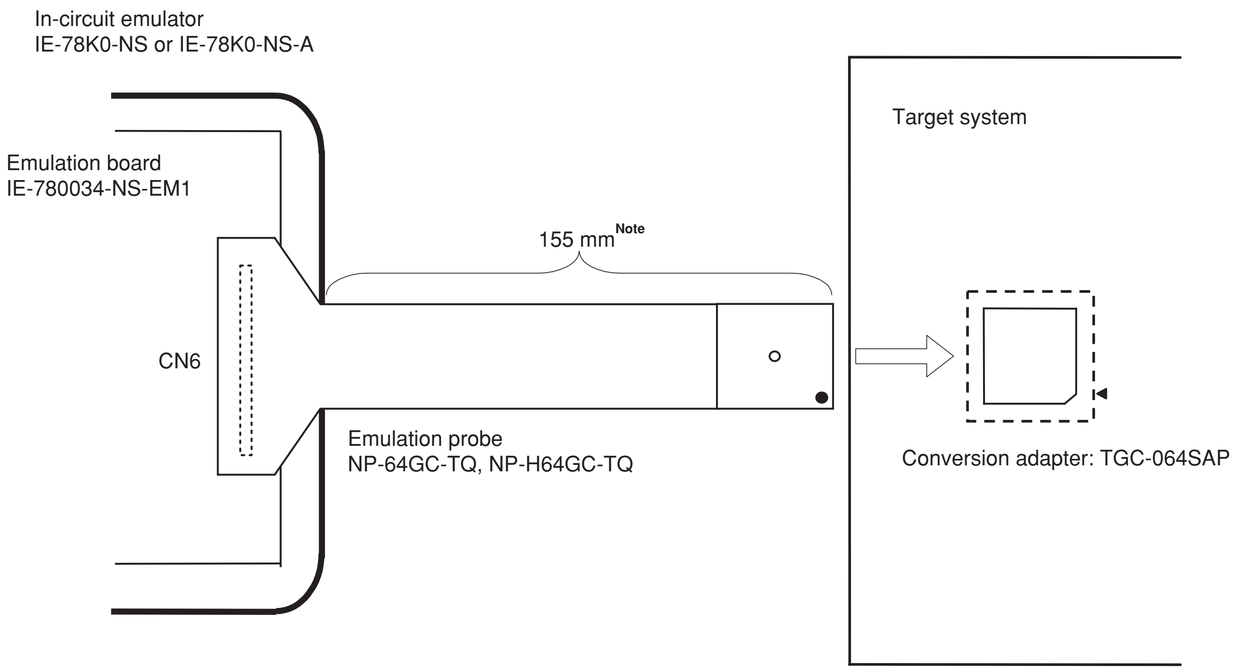


Figure C-4. Distance Between In-Circuit Emulator and Conversion Adapter (When Using 64GC)



Note Distance when using NP-64GC-TQ. This is 355 mm when using NP-H64GC-TQ.

Figure C-5. Connection Conditions of Target System (When Using NP-64GC-TQ)

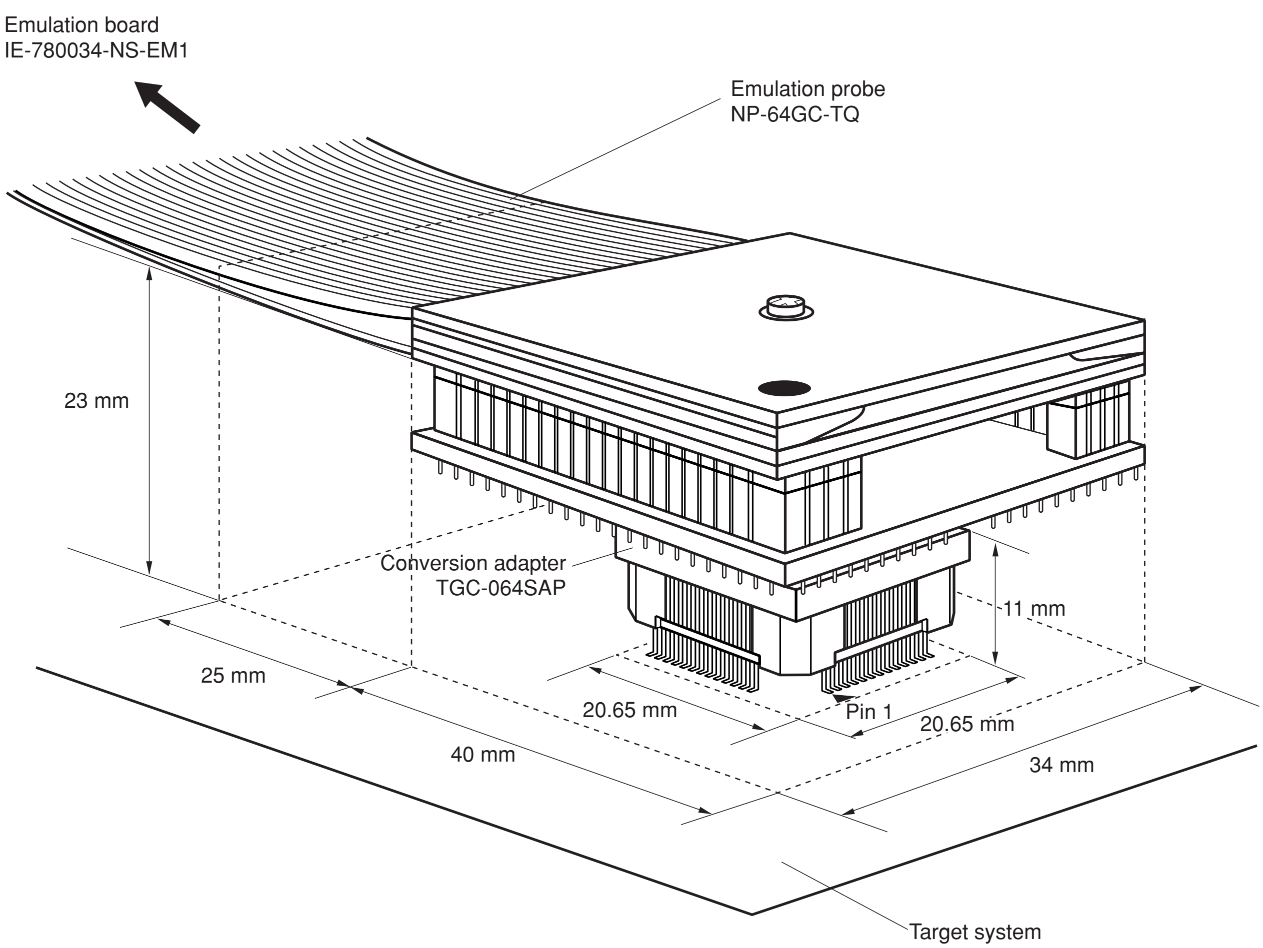


Figure C-6. Connection Conditions of Target System (When Using NP-H64GC-TQ)

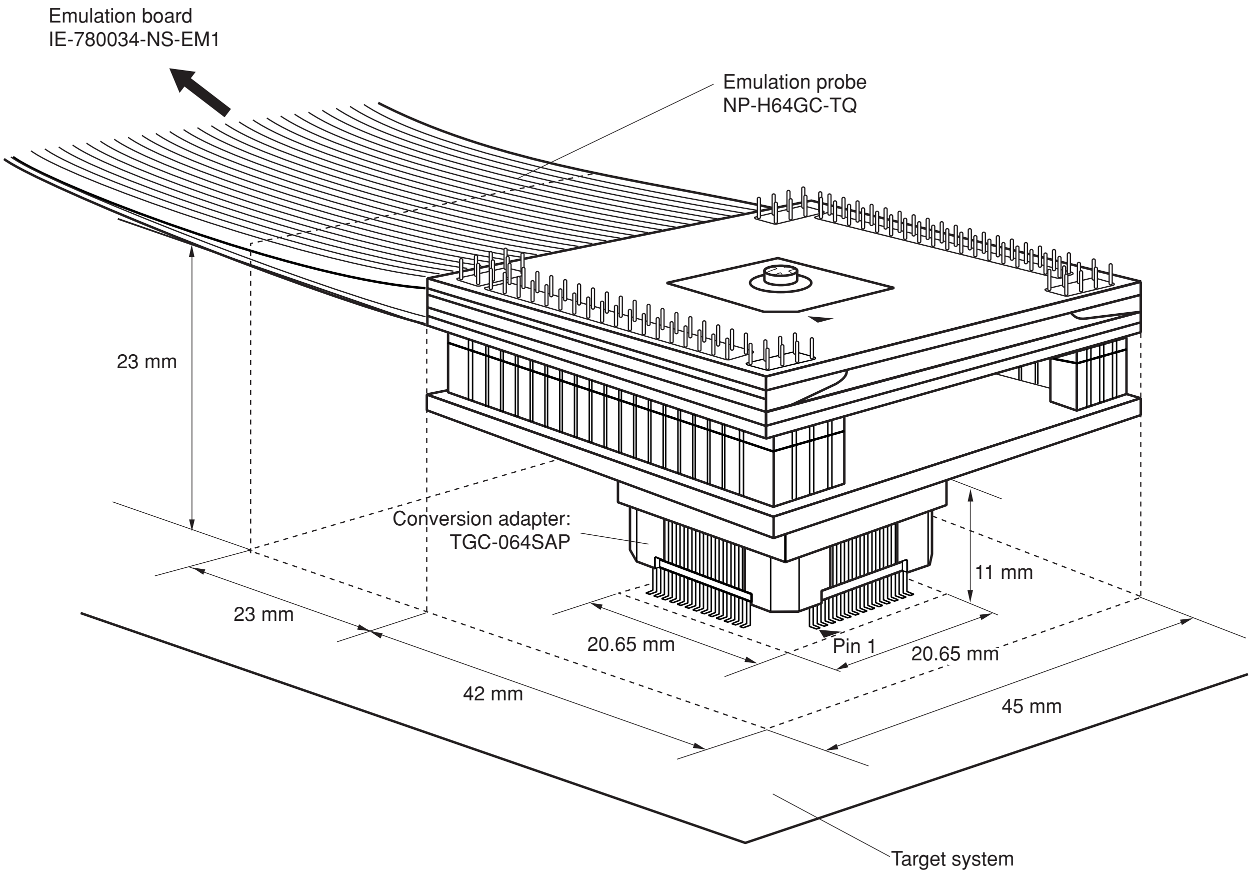
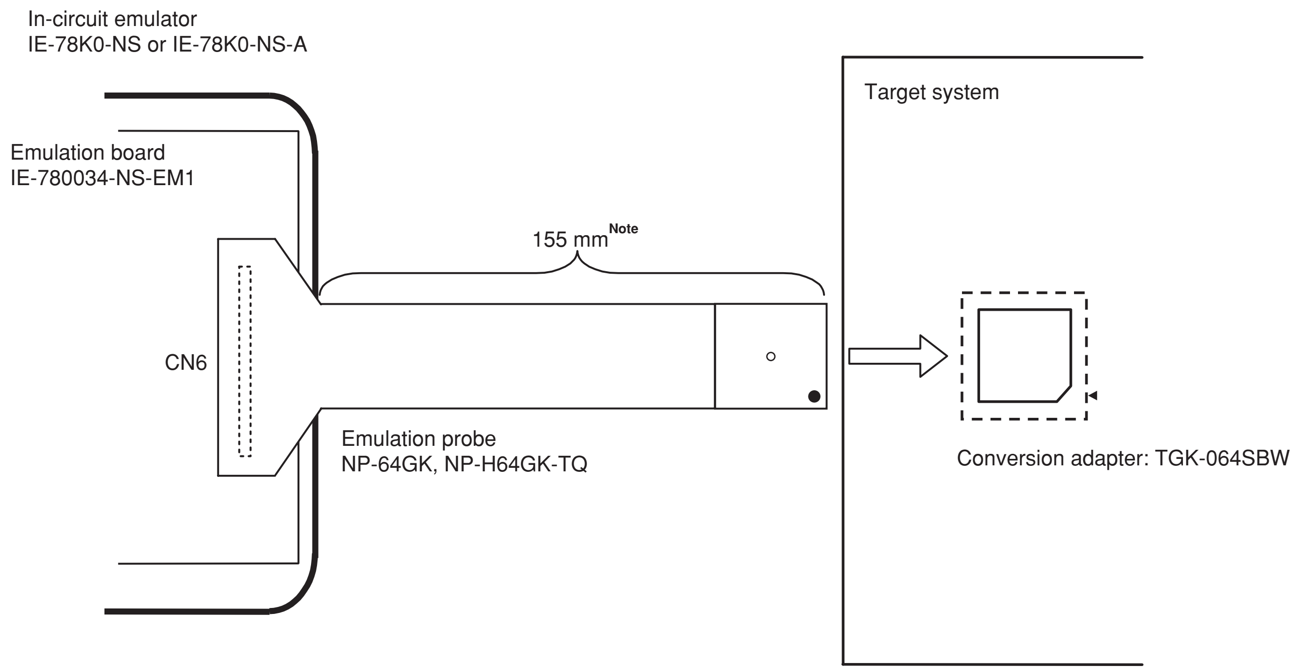


Figure C-7. Distance Between In-Circuit Emulator and Conversion Adapter (When Using 64GK)



Note Distance when using NP-64GK. This is 355 mm when using NP-H64GK-TQ.

Figure C-8. Connection Conditions of Target System (When Using NP-64GK)

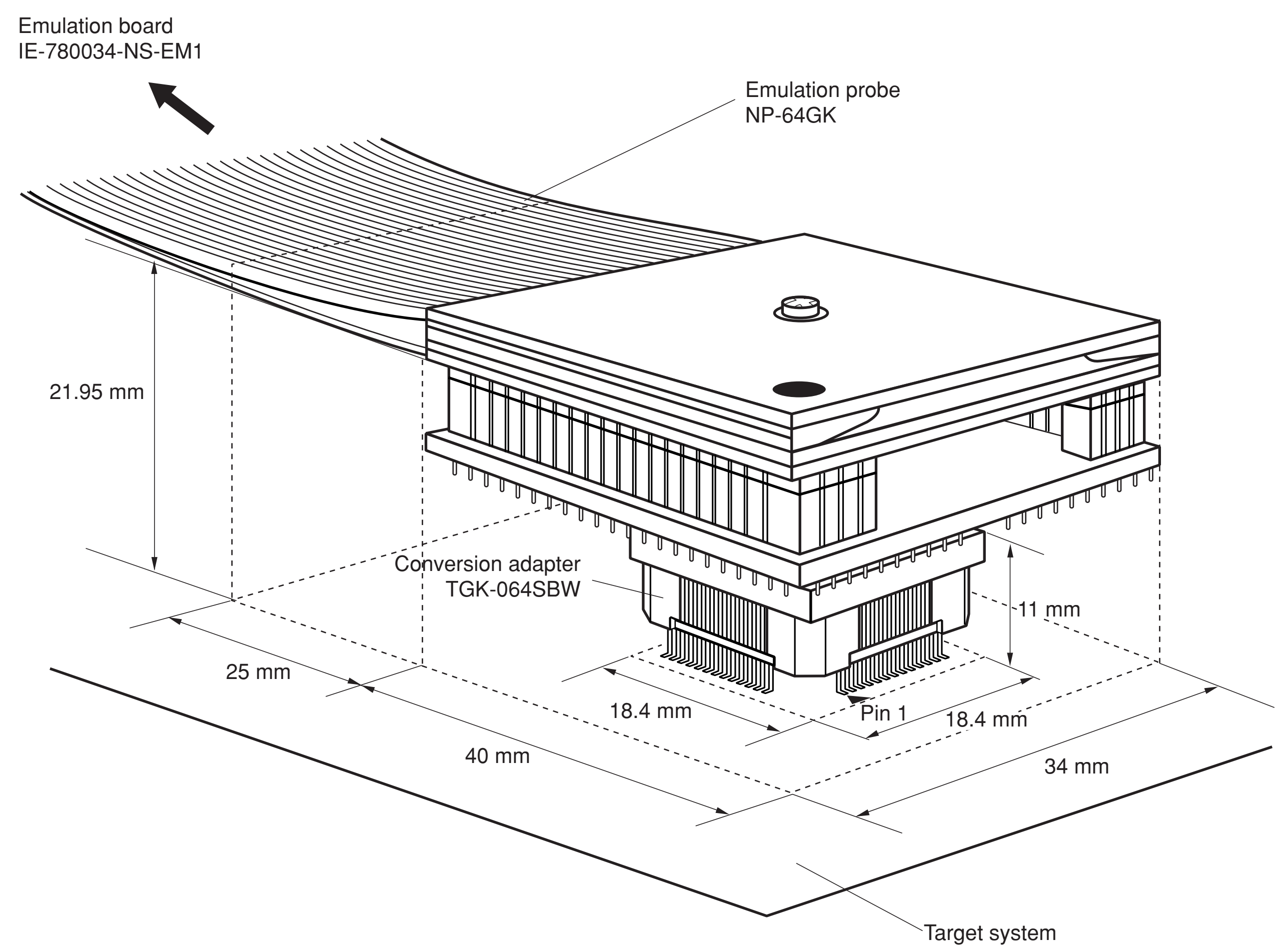


Figure C-9. Connection Conditions of Target System (When Using NP-H64GK-TQ)

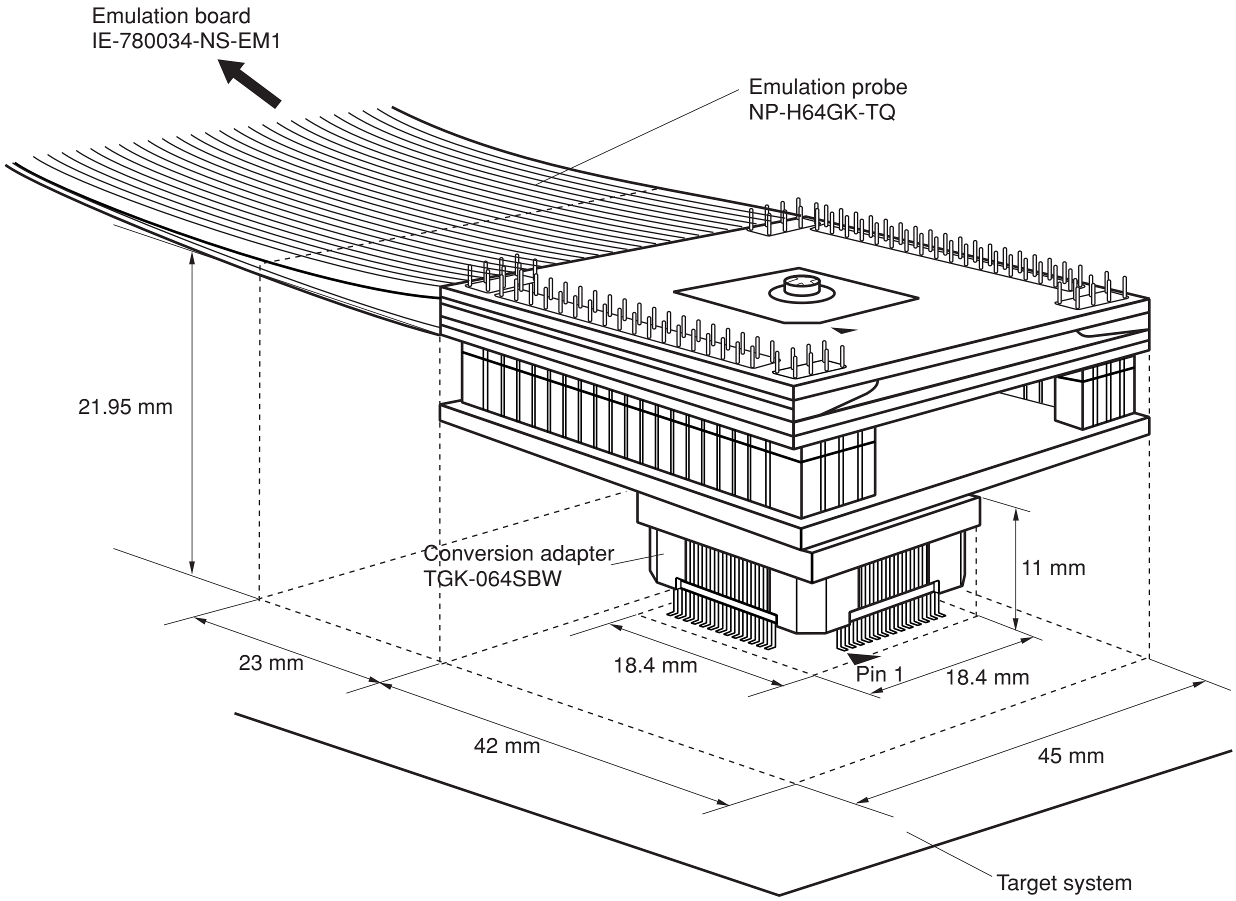
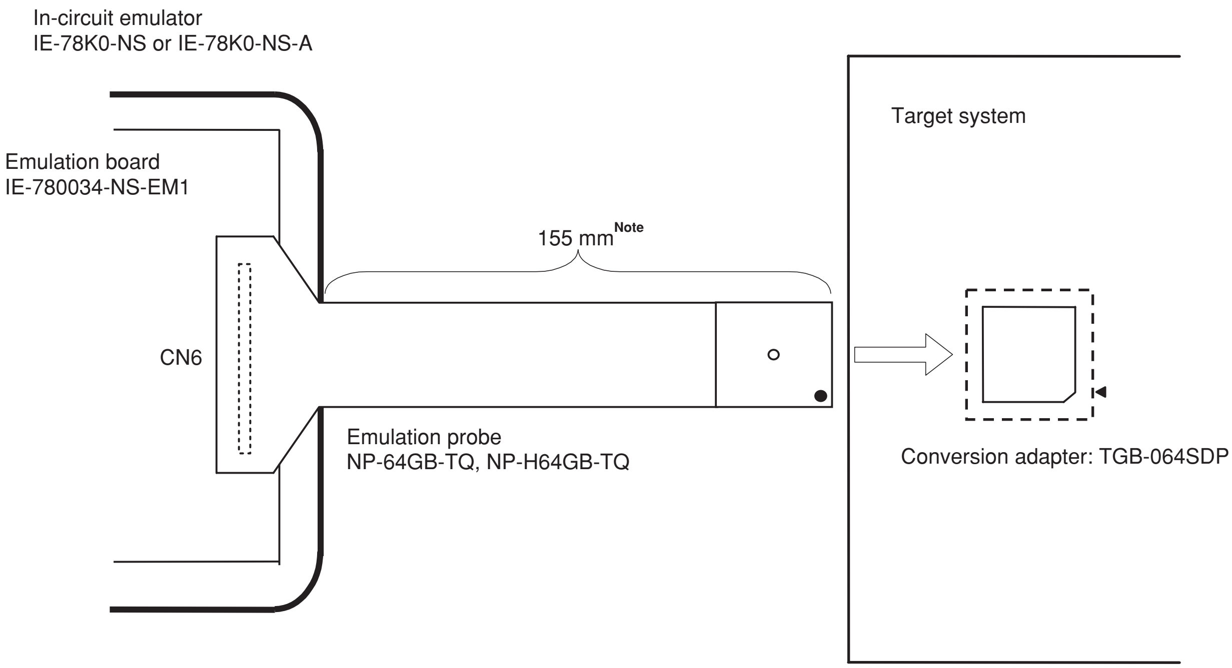


Figure C-10. Distance Between In-Circuit Emulator and Conversion Adapter (When Using 64GB)



Note Distance when using NP-64GB-TQ. This is 355 mm when using NP-H64GB-TQ.

Figure C-11. Connection Conditions of Target System (When Using NP-64GB-TQ)

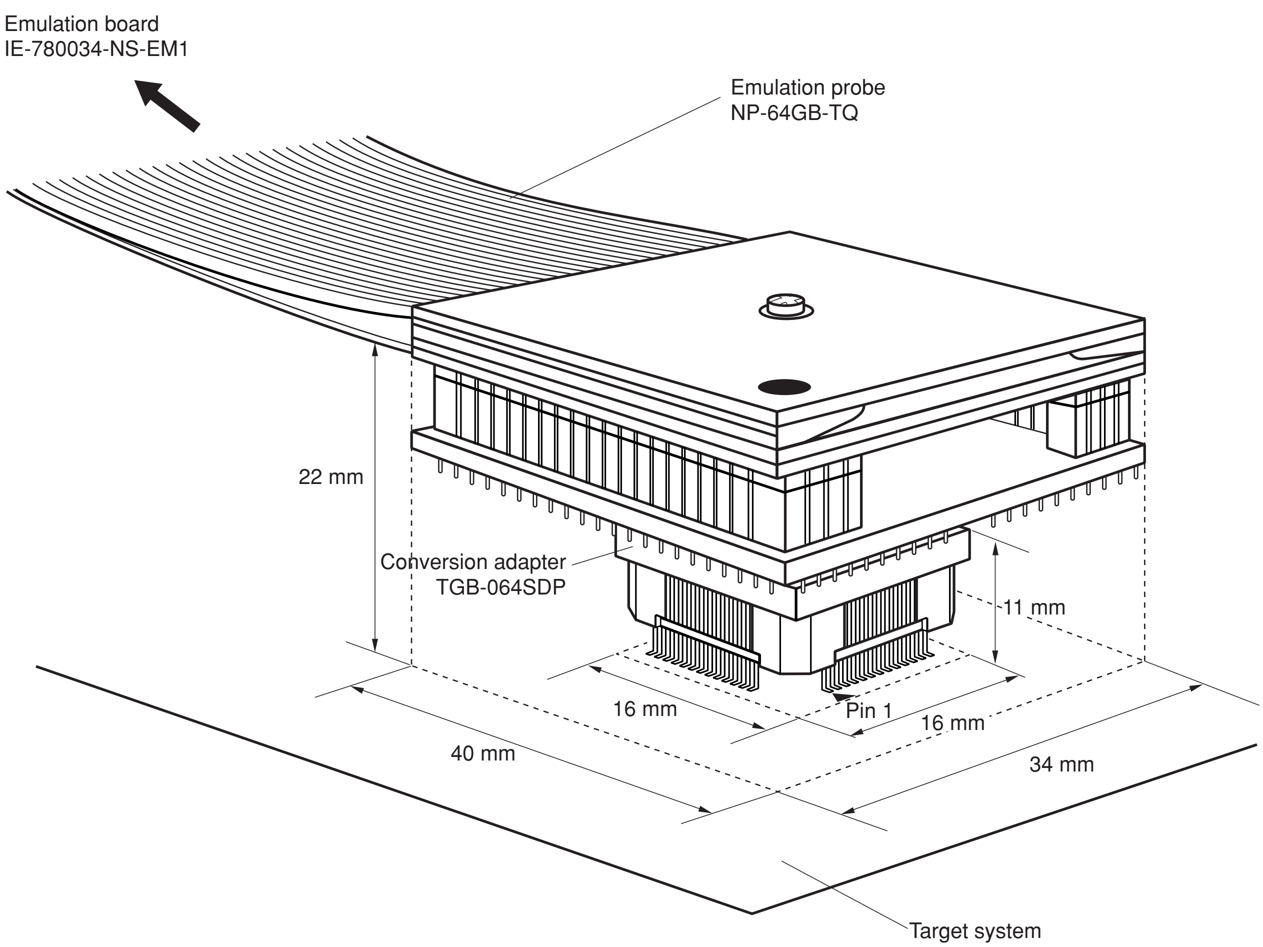


Figure C-12. Connection Conditions of Target System (When Using NP-H64GB-TQ)

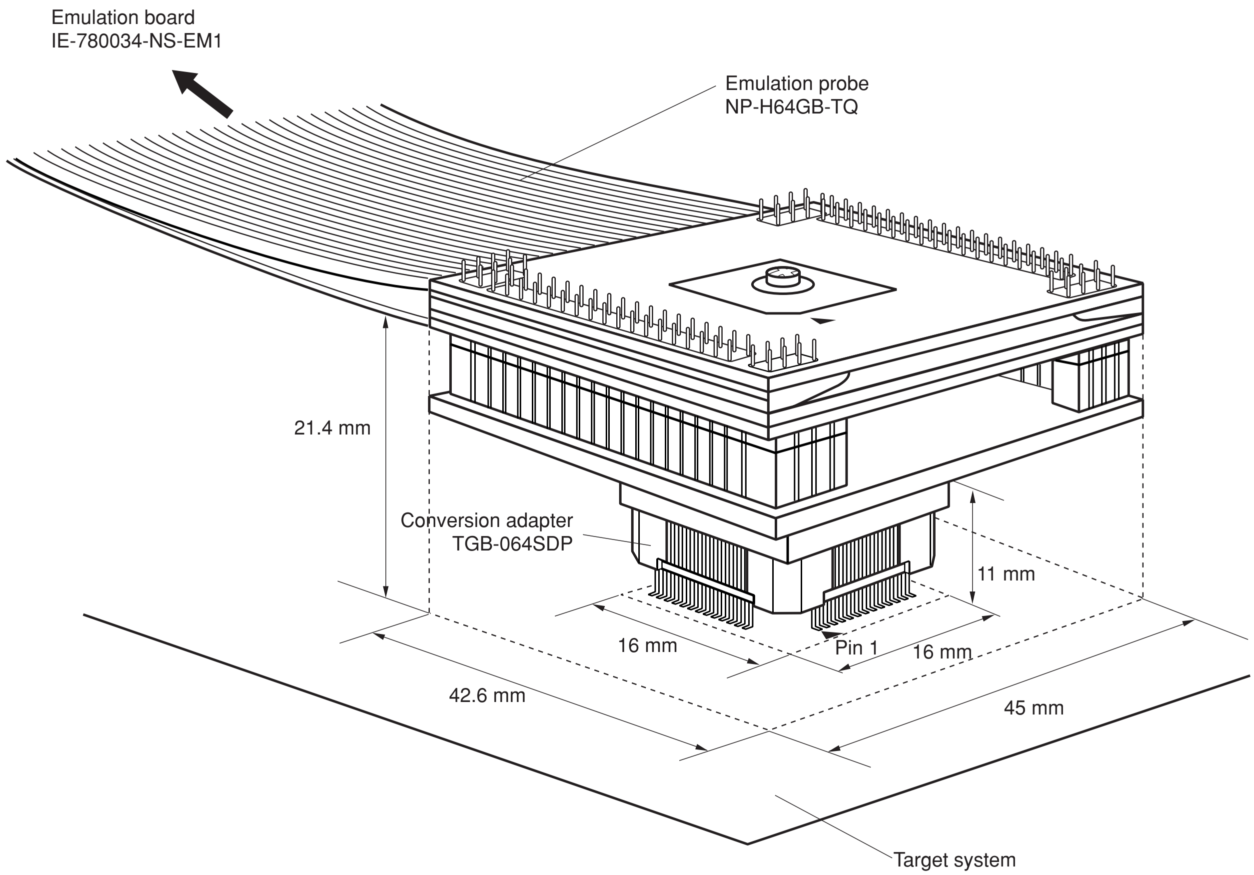


Figure C-13. Distance Between In-Circuit Emulator and Conversion Socket (When Using NP-73F1-CN3)

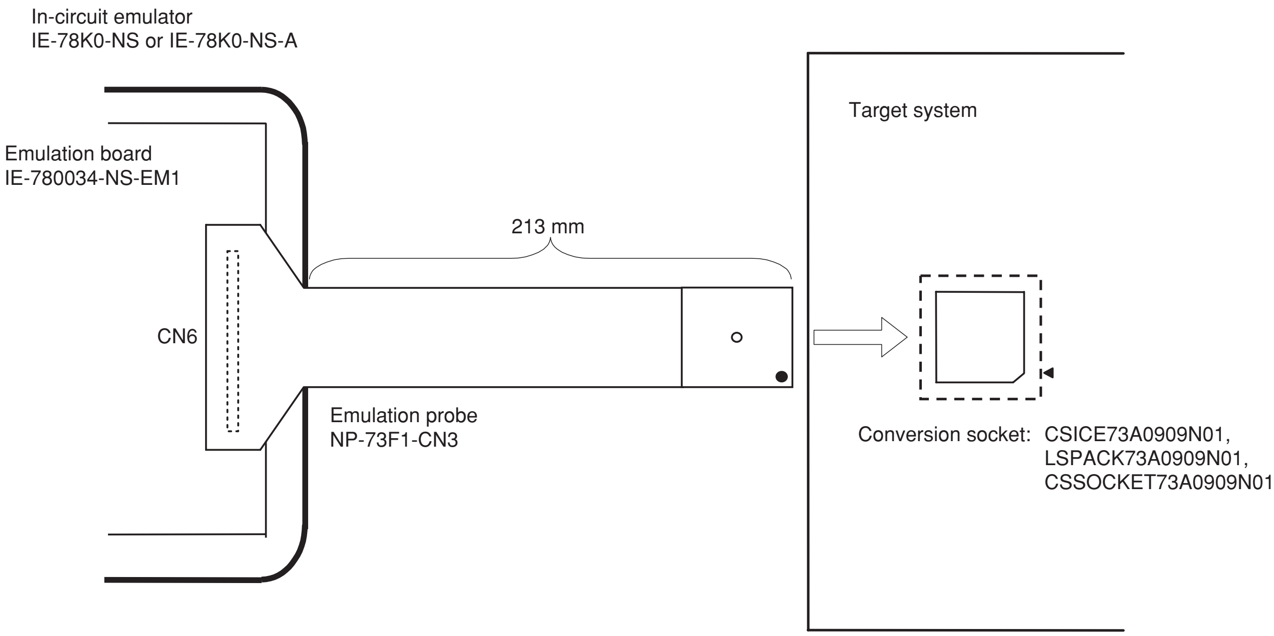
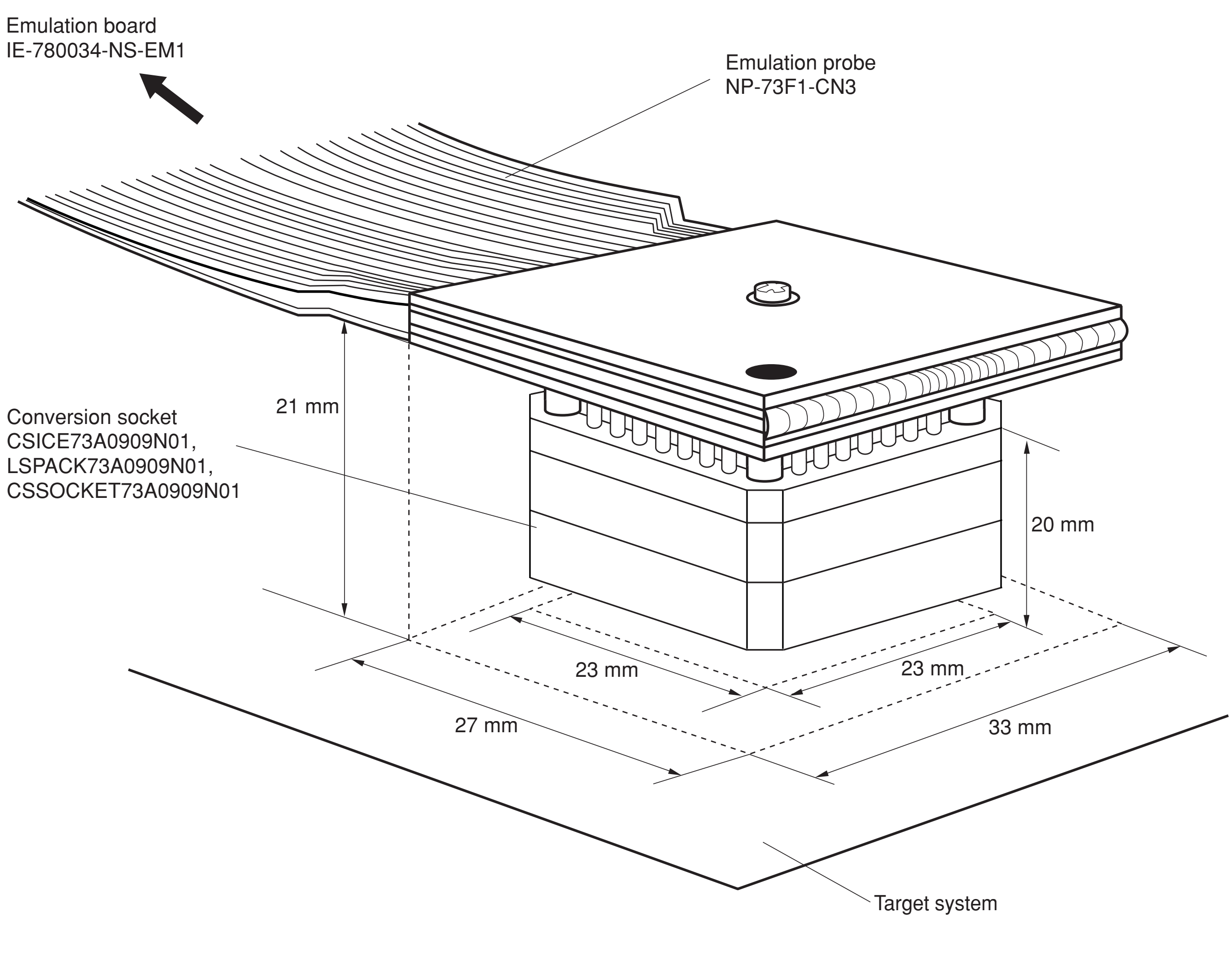


Figure C-14. Connection Conditions of Target System (When Using NP-73F1-CN3)



APPENDIX D REGISTER INDEX

D.1 Register Name Index

[P]

[R]

[S]

[T]

[W]

D.2 Register Symbol Index

[P]

[R]

[S]

[T]

APPENDIX E REVISION HISTORY

The revision history for this manual is detailed below. “Chapter” indicates the chapter of each edition. (1/5)

Edition	Revision from Previous Edition	Chapter
Second edition	Deletion of the following products • μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A), 780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A)	Throughout
	Deletion of the following package • 64-pin plastic LQFP (GK-8A8 type)	
	Addition of the following packages • 64-pin plastic TQFP (GK-9ET type) • 64-pin plastic LQFP (GB-8EU type)	
	Modification of recommended connection of unused pins in **Table 3-1 Pin I/O Circuit Types**	**CHAPTER 3 PIN FUNCTION (μPD780024A, 780034A SUBSERIES)**
	Modification of recommended connection of unused pins in **Table 4-1 Pin I/O Circuit Types**	**CHAPTER 4 PIN FUNCTION (μPD780024AY, 780034AY SUBSERIES)**
	Modification of **Figure 6-2 P00 to P03 Block Diagram**	**CHAPTER 6 PORT FUNCTIONS**
	Modification of **Figure 6-4 P20, P22, P23, P25 Block Diagram**	
	Modification of **Figure 6-7 P34 and P36 Block Diagram (μPD780024A, 780034A Subseries)**	
	Modification of **Figure 6-8 P35 Block Diagram (μPD780024A, 780034A Subseries)**	
	Modification of **Figure 6-10 P32 and P33 Block Diagram (μPD780024AY, 780034AY Subseries)**	
	Modification of **Figure 6-12 P40 to P47 Block Diagram**	
	Modification of **Figure 6-14 P50 to P57 Block Diagram**	
	Modification of **Figure 6-15 P64 to P67 Block Diagram**	
	Modification of **Figure 6-16 P70 to P73 Block Diagram**	
	Modification of **Figure 6-17 P74 and P75 Block Diagram**	
	Addition of note for feedback resistor in **Figure 7-3 Processor Clock Control Register (PCC) Format**	**CHAPTER 7 CLOCK GENERATOR**
	Deletion of one-shot pulse output function	**CHAPTER 8 16-BIT TIMER/EVENT COUNTER 0**
	Addition of caution for INTWT in **Figure 10-3 Operation Timing of Watch Timer/Interval Timer**	**CHAPTER 10 WATCH TIMER**
	Addition of **13.5 How to Read A/D Converter Characteristics Table**	**CHAPTER 13 8-BIT A/D CONVERTER (μPD780024A, 780024AY SUBSERIES)**
	13.6 A/D Converter Cautions Addition of (10) Timing at which A/D conversion result is undefined Addition of (11) Notes on board design Addition of (13) AV_{REF} pin Addition of (14) Internal equivalent circuit of ANI0 to ANI7 pins and permissible signal source impedance	
	Addition of **14.5 How to Read A/D Converter Characteristics Table**	**CHAPTER 14 10-BIT A/D CONVERTER (μPD780034A, 780034AY SUBSERIES)**
	14.6 A/D Converter Cautions Addition of (10) Timing at which A/D conversion result is undefined Addition of (11) Notes on board design Addition of (13) AV_{REF} pin Addition of (14) Internal equivalent circuit of ANI0 to ANI7 pins and permissible signal source impedance	

(2/5)

Edition	Revision from Previous Edition	Chapter
Second edition	Modification of **Figure 18-3 IIC Control Register 0 (IICC0) Format**	**CHAPTER 18 SERIAL INTERFACE (IIC0) (μPD780024AY, 780034AY SUBSERIES ONLY)**
	Deletion of Flashpro II	**CHAPTER 23 μPD78F0034A, 78F0034AY**
	Revision of development tools	**APPENDIX B DEVELOPMENT TOOLS**
Third edition	Addition of the following products μPD780021AY(A), 780022AY(A), 780023AY(A), 780024AY(A), μPD780031AY(A), 780032AY(A), 780033AY(A), 780034AY(A), μPD78F0034B, 78F0034B(A), 78F0034BY, 78F0034BY(A)	Throughout
	Addition of the following packages • 64-pin plastic LQFP (GC-8BS type) • 73-pin plastic FBGA (F1-CN3 type)	
	Addition of expanded-specification products to μPD780024A, 780034A Subseries	
	Addition of **1.1 Expanded-Specification Products and Conventional Products**	**CHAPTER 1 OUTLINE (μPD780024A, 780034A SUBSERIES)**
	Addition of **1.10 Correspondence Between Mask ROM Versions and Flash Memory Versions**	
	Modification of **1.11 Differences Between Standard Grade Products and Special Grade Products**	
	Addition of **1.12 Correspondence Between Products and Packages**	
	Addition of **2.9 Correspondence Between Mask ROM Versions and Flash Memory Versions**	**CHAPTER 2 OUTLINE (μPD780024AY, 780034AY SUBSERIES)**
	Modification of **2.10 Differences Between Standard Grade Products and Special Grade Products**	
	Addition of **2.11 Correspondence Between Products and Packages**	
	Addition of description of pin processing in **3.2.18 V_{PP} (flash memory versions only)**	**CHAPTER 3 PIN FUNCTION (μPD780024A, 780034A SUBSERIES)**
	Modification of **Table 3-1 Pin I/O Circuit Types**	
	Addition of description of pin processing in **4.2.18 V_{PP} (flash memory versions only)**	**CHAPTER 4 PIN FUNCTION (μPD780024AY, 780034AY SUBSERIES)**
	Modification of **Table 4-1 Pin I/O Circuit Types**	
	Addition of description of program area in **5.1.2 Internal data memory space**	**CHAPTER 5 CPU ARCHITECTURE**
	Modification of **Figure 5-14 Data to Be Saved to Stack Memory** and **Figure 5-15 Data to Be Restored from Stack Memory**	
	Modification of **[Description example]** in **5.4.4 Short direct addressing**	
	Addition of **[Illustration]** in **5.4.7 Based addressing**, **5.4.8 Based indexed addressing**, and **5.4.9 Stack addressing**	
	Modification of port block diagram (**Figures 6-2 Block Diagram of P00 to P03** to **6-23 Block Diagram of P74 and P75**)	**CHAPTER 6 PORT FUNCTIONS**
	Addition of **Table 6-6 Port Mode Registers and Output Latch Settings When Alternate Function Is Used**	

(3/5)

Edition	Revision from Previous Edition	Chapter
Third edition	Addition of description of internal feedback resistor and oscillation stabilization time select register (OSTS) in **7.3 Clock Generator Control Registers**	**CHAPTER 7 CLOCK GENERATOR**
	Modification of **Figure 8-1 Block Diagram of 16-Bit Timer/Event Counter 0**	**CHAPTER 8 16-BIT TIMER/EVENT COUNTER 0**
	Modification of **Tables 8-2 TI00/TO0/P70 Pin Valid Edge and CR00, CR01 Capture Trigger** and **8-3 TI01/P71 Pin Valid Edge and CR00 Capture Trigger** in 2nd edition to **Table 8-2 CR00 Capture Trigger and Valid Edges of TI00 and TI01 Pins** and **Table 8-3 CR01 Capture Trigger and Valid Edge of TI00 Pin (CRC02 = 1)**	
	Modification of description procedure of each function in **8.4 Operation of 16-Bit Timer/Event Counter 0**	
	Addition of **Figure 8-26 PPG Output Configuration Diagram** and **Figure 8-27 PPG Output Operation Timing**	
	Addition of **8.5 Program List**	
	Modification of **8.6 (3) Capture register data retention timing** and addition of **(11) STOP mode or main system clock stop mode setting**	
	Modification of **Figures 9-1 Block Diagram of 8-Bit Timer/Event Counter 50** and **9-2 Block Diagram of 8-Bit Timer/Event Counter 51**	**CHAPTER 9 8-BIT TIMER/EVENT COUNTERS 50, 51**
	Deletion of **Caution** in **Figures 9-5 Format of 8-Bit Timer Mode Control Register 50 (TMC50)** and **9-6 Format of 8-Bit Timer Mode Control Register 51 (TMC51)**	
	Addition of **[Setting]** in **9.4.2 External event counter operation**	
	Addition of description of frequency to **[Setting]** in **9.4.3 Square-wave output (8-bit resolution) operation**	
	Addition of description of cycle and duty ratio to **[Setting]** in **9.4.4 8-bit PWM output operation**	
	Addition of **9.5 Program List**	
	Deletion of **9.5 (2) Operation after compare register change during timer count operation** in 2nd edition	
	Deletion of oscillation stabilization time select register (OSTS) from **11.3 Registers to Control Watchdog Timer** in 2nd edition	**CHAPTER 11 WATCHDOG TIMER**
	Modification of **Figure 12-1 Block Diagram of Clock Output/Buzzer Output Controller**	**CHAPTER 12 CLOCK OUTPUT/ BUZZER OUTPUT CONTROLLER**
	Modification of description in **13.2 (3) Sample & hold circuit**, **(4) Voltage comparator**, and addition of **(10) ADTRG pin**	**CHAPTER 13 8-BIT A/D CONVERTER (μPD780024A, 780024AY SUBSERIES)**
	Addition of **Table 13-2 Sampling Time and A/D Conversion Start Delay Time of A/D Converter**	
	Deletion of **13.6 (4) Noise countermeasures** (contents of deletion are added to **Figure 13-18 Example of Connecting Capacitor to AV$_{REF}$ Pin** and **Figure 13-20 Example of Connection If Signal Source Impedance Is High**), and addition of **(14) Input impedance of ANI0 to ANI7 pins**	
	Modification of **Table 13-3 Resistances and Capacitances of Equivalent Circuit (Reference Values)**	

(4/5)

Edition	Revision from Previous Edition	Chapter
Third edition	Addition of **Figure 14-2 Format of A/D Conversion Result Register 0 (ADCR0)**	**CHAPTER 14 10-BIT A/D CONVERTER (μPD780034A, 780034AY SUBSERIES)**
	Modification of description in **14.2 (3) Sample & hold circuit**, **(4) Voltage comparator**, and addition of **(10) ADTRG pin**	
	Addition of **Table 14-2 Sampling Time and A/D Conversion Start Delay Time of A/D Converter**	
	Deletion of **14.6 (4) Noise countermeasures** (contents of deletion are added to **Figure 14-19 Example of Connecting Capacitor to AV_{REF} Pin** and **Figure 14-21 Example of Connection If Signal Source Impedance Is High**), and addition of **(14) Input impedance of ANI0 to ANI7 pins**	
	Modification of **Table 14-3 Resistances and Capacitances of Equivalent Circuit (Reference Values)**	
	Modification of **Figure 16-1 Block Diagram of Serial Interface UART0**	**CHAPTER 16 SERIAL INTERFACE UART0**
	Move of description of asynchronous serial interface status register 0 (ASIS0) in **16.3 Registers to Control Serial Interface UART0** to **16.2 Configuration of Serial Interface UART0**	
	Addition of **Caution** in **Figure 16-7 Error Tolerance (When k = 0), Including Sampling Errors**	
	Modification of **Caution** in **Figure 16-10 Timing of Asynchronous Serial Interface Receive Completion Interrupt Request**	
	Addition of **(1) Registers to be used** and **(3) Relationship between main system clock and baud rate** in **16.4.3 Infrared data transfer mode**	
	Addition of **Table 16-6 Register Settings**	
	Modification of **Figure 17-1 Block Diagram of Serial Interface SIO3n**	**CHAPTER 17 SERIAL INTERFACES SIO30 AND SIO31**
	Addition of **Note 3** and **Caution** in **Figures 17-2 Format of Serial Operation Mode Register 30 (CSIM30)** and **17-3 Format of Serial Operation Mode Register 31 (CSIM31)**	
	Addition of **Table 17-2 Register Settings**	
	Modification of **Figure 18-1 Block Diagram of Serial Interface IIC0**	**CHAPTER 18 SERIAL INTERFACE IIC0 (μPD780024AY, 780034AY SUBSERIES ONLY)**
	Unification of **18.2 (1) IIC shift register 0 (IIC0)** and **(4) IIC shift register 0 (IIC0)** in 2nd edition, and **(2) Slave address register 0 (SVA0)** and **(3) Slave address register 0 (SVA0)** in 2nd edition	
	Addition of description to "Transfer lines" in **Figure 18-16 Wait Signal**	
	Addition of description to **Notes 1** and **2** in **Table 18-2 INTIIC0 Timing and Wait Control**	
	Modification of **Figure 18-21 Master Operation Flowchart**	
	Modification of **18.5.15 (2) Slave operation**	
	Modification of **(1) Start condition ~ address** and **(2) Data** in **Figure 18-23 Example of Master to Slave Communication (When 9-Clock Wait Is Selected for Both Master and Slave)**	
	Modification of **Figure 18-24 Example of Slave to Master Communication (When 9-Clock Wait Is Selected for Both Master and Slave)**	

(5/5)

Edition	Revision from Previous Edition	Chapter
Third edition	Modification of **(E) Software interrupt** in **Figure 19-1 Basic Configuration of Interrupt Function**	**CHAPTER 19 INTERRUPT FUNCTIONS**
	Addition of **Caution 5** in **Figure 19-2 Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L)**	
	Addition of **Caution** in **Figure 19-5 Format of External Interrupt Rising Edge Enable Register (EGP), External Interrupt Falling Edge Enable Register (EGN)**	
	Addition of description and **Remark** in **19.4.1 Non-maskable interrupt request acknowledgment operation**	
	Addition of description in **19.4.2 Maskable interrupt request acknowledgment operation**	
	Addition of an item in **Table 19-4 Interrupt Requests Enabled for Nesting During Interrupt Servicing**	
	Addition of description of using expanded-specification products	**CHAPTER 20 EXTERNAL DEVICE EXPANSION FUNCTION**
	Addition of clock output and buzzer output in **Table 21-1 HALT Mode Operating Statuses**	**CHAPTER 21 STANDBY FUNCTION**
	Modification of clock output in **Table 21-3 STOP Mode Operating Statuses**	
	Revision of description	**CHAPTER 23 μPD78F0034A, 78F0034B, 78F0034AY, 78F0034BY**
	Addition of description	**CHAPTER 25 ELECTRICAL SPECIFICATIONS (EXPANDED-SPECIFICATION PRODUCTS: f_X = 1.0 TO 12 MHz)**
		CHAPTER 26 ELECTRICAL SPECIFICATIONS (CONVENTIONAL PRODUCTS: f_X = 1.0 TO 8.38 MHz)
		CHAPTER 27 PACKAGE DRAWINGS
		CHAPTER 28 RECOMMENDED SOLDERING CONDITIONS
	Revision of description	**APPENDIX A DIFFERENCES BETWEEN μPD78018F, 780024A, 780034A, AND 780078 SUBSERIES**
	Revision of description	**APPENDIX B DEVELOPMENT TOOLS**
	Addition of description	**APPENDIX C NOTES ON TARGET SYSTEM DESIGN**