To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

Notice

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

“Standard”:	Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
“High Quality”:	Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
“Specific”:	Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.

8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

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

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

DATA SHEET

MOS INTEGRATED CIRCUIT

μPD17225, 17226, 17227, 17228

4-BIT SINGLE-CHIP MICROCONTROLLER FOR SMALL GENERAL-PURPOSE INFRARED REMOTE CONTROL TRANSMITTER

DESCRIPTION

μPD17225, 17226, 17227, 17228 (hereafter called μPD17225 subseries) are 4-bit single-chip microcontrollers for small general-purpose infrared remote control transmitters.

It employs a 17K architecture of general-purpose register type devices for the CPU, and can directly execute operations between memories instead of the conventional method of executing operations through the accumulator. Moreover, all the instructions are 16-bit 1-word instructions which can be programmed efficiently.

In addition, a one-time PROM model, μPD17P218, to which data can be written only once, is also available. It is convenient either for evaluating the μPD17225 subseries programs or small-scale production of application systems.

Detailed functions are described in the following manual. Be sure to read this manual when designing your system.

μPD172×× Subseries User's Manual: U12795E

FEATURES

- Infrared remote controller carrier generator circuit (REM output)
- 17K architecture: General-purpose register system
- Program memory (ROM), Data memory (RAM)

	μPD17225	μPD17226	μPD17227	μPD17228
Program memory (ROM)	4 K bytes (2048 × 16)	8 K bytes (4096 × 16)	12 K bytes (6144 × 16)	16 K bytes (8192 × 16)
Data memory (RAM)	111 × 4 bits		223 × 4 bits	

- 8-bit timer : 1 channel
- Basic internal timer/Watchdog timer : 1 channel ($\overline{\text{WDOUT}}$ output)
- Instruction execution time (can be changed in two steps)
  - at f_X = 4 MHz : 4 μs (high-speed mode)/8 μs (ordinary mode)
  - at f_X = 8 MHz : 2 μs (high-speed mode)/4 μs (ordinary mode)
- External interrupt pin (INT) : 1
- I/O pins : 20
- Supply voltage : V_{DD} = 2.2 to 3.6 V (at f_X = 8 MHz (high-speed mode))
  V_{DD} = 2.0 to 3.6 V (at f_X = 4 MHz (high-speed mode))
- Low-voltage detector circuit (mask option)

Unless otherwise specified, the μPD17225 is treated as the representative model throughout this document.

**The information in this document is subject to change without notice. Before using this document, please confirm that this is the latest version.
Not all devices/types available in every country. Please check with local NEC representative for availability and additional information.**

Document No. U12643EJ2V0DS00 (2nd edition)
Date Published April 1999 N CP (K)
Printed in Japan

The mark ★ shows major revised points.

© NEC Corporation 1997,1999

APPLICATION

Preset remote controllers, toys, portable systems, etc.

ORDERING INFORMATION

Part Number	Package
μPD17225CT-×××	28-pin plastic shrink DIP (400 mil)
μPD17225GT-×××	28-pin plastic SOP (375 mil)
★ μPD17225MC-×××-5A4	30-pin plastic shrink SOP (300 mil)
μPD17226CT-×××	28-pin plastic shrink DIP (400 mil)
μPD17226GT-×××	28-pin plastic SOP (375 mil)
★ μPD17226MC-×××-5A4	30-pin plastic shrink SOP (300 mil)
μPD17227CT-×××	28-pin plastic shrink DIP (400 mil)
μPD17227GT-×××	28-pin plastic SOP (375 mil)
★ μPD17227MC-×××-5A4	30-pin plastic shrink SOP (300 mil)
μPD17228CT-×××	28-pin plastic shrink DIP (400 mil)
μPD17228GT-×××	28-pin plastic SOP (375 mil)
★ μPD17228MC-×××-5A4	30-pin plastic shrink SOP (300 mil)

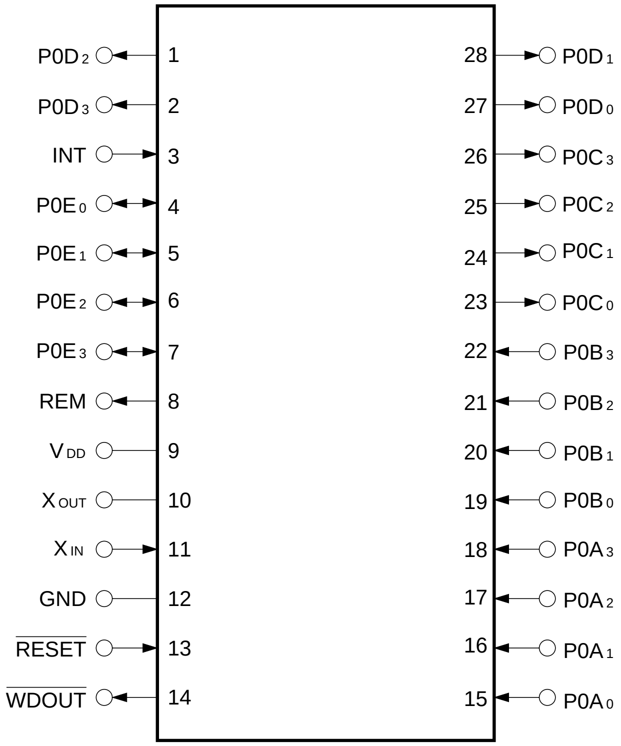
Remark ××× indicates ROM code suffix.

DIFFERENCE BETWEEN μPD17225 SUBSERIES AND μPD17215 SUBSERIES

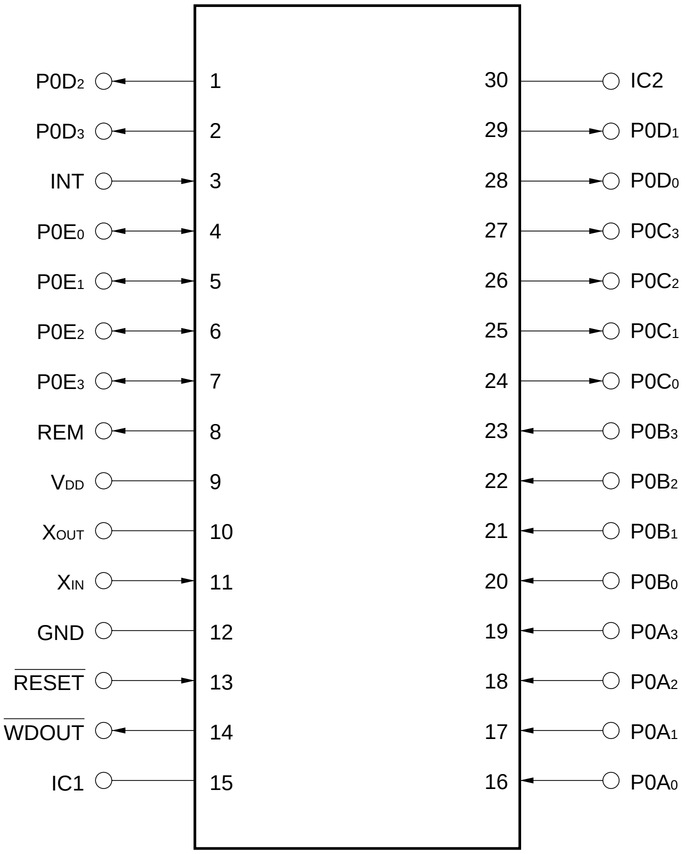
Item	μPD17225 Subseries	μPD17215 Subseries
Supply Voltage	V_{DD} = 2.0 to 3.6 V	V_{DD} = 2.0 to 5.5 V
Instruction Execution Time (t_{CY})	2 μs (V_{DD} = 2.2 to 3.6 V) 4 μs (V_{DD} = 2.0 to 3.6 V)	2 μs (V_{DD} = 3.5 to 5.5 V) 4 μs (V_{DD} = 2.2 to 5.5 V) 8 μs (V_{DD} = 2.0 to 5.5 V)
$\overline{\text{WDOUT}}$ Pin	Connected to $\overline{\text{RESET}}$ pin or V_{DD} via resistor (when not used)	Connected to $\overline{\text{RESET}}$ pin or GND (when not used)
	Connection of $\overline{\text{WDOUT}}$ pin differs between μPD17P218 and μPD17225, 17226, 17227, and 17228 when OTP is evaluated on μPD17225 subseries board and when $\overline{\text{WDOUT}}$ pin is not used. When μPD17P218 is used, malfunctioning does not occur even when $\overline{\text{WDOUT}}$ pin is pulled up, though connection can be changed by using a jumper switch on the external board.	
Others	Supply voltage, low-voltage detection voltage, oscillator characteristics, and noise characteristics differ. Although μPD17P218 is used as one-time PROM for evaluation for both subseries, μPD17P218 cannot be used to evaluate the low-voltage high-speed operation of the μPD17225 subseries.	

PIN CONFIGURATION (TOP VIEW)

- 28-pin plastic SOP (375 mil)
  μPD17225GT-×××, 17226GT-×××, 17227GT-×××, 17228GT-×××
- 28-pin plastic shrink DIP (400 mil)
  μPD17225CT-×××, 17226CT-×××, 17227CT-×××, 17228CT-×××

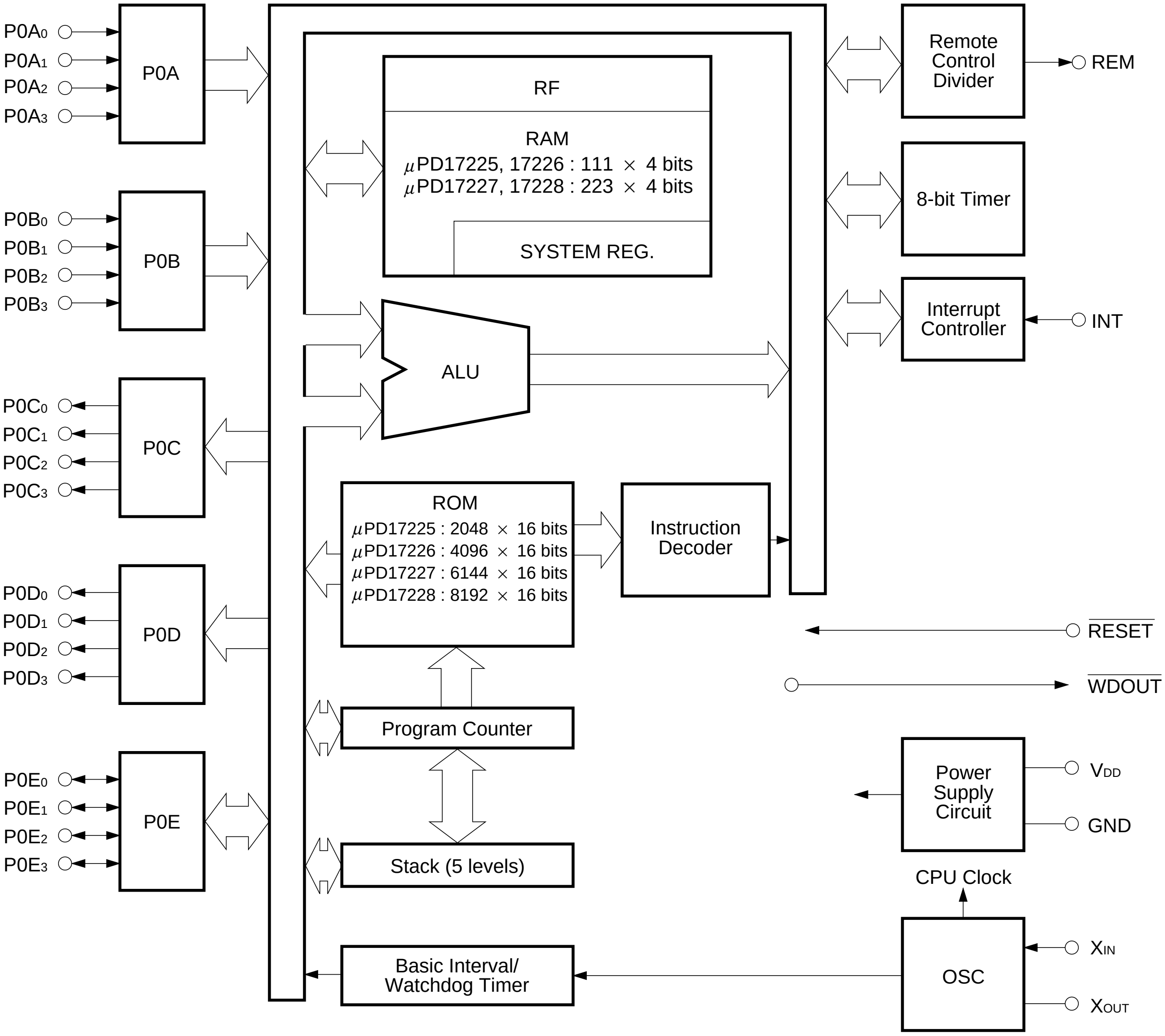


★ • 30-pin plastic shrink SOP (300 mil)
μPD17225MC-×××-5A4, μPD17226MC-×××-5A4, μPD17227MC-×××-5A4, μPD17228MC-×××-5A4



GND	:	Ground
★ IC1, IC2	:	Internally connected
INT	:	External interrupt request signal input
$P0A_0$-$P0A_3$	:	Input port (CMOS input)
$P0B_0$-$P0B_3$	:	Input port (CMOS input)
$P0C_0$-$P0C_3$	:	Output port (N-ch open-drain output)
$P0D_0$-$P0D_3$	:	Output port (N-ch open-drain output)
$P0E_0$-$P0E_3$	:	I/O port (CMOS push-pull output)
REM	:	Remote controller output (CMOS push-pull output)
$\overline{\text{RESET}}$	:	Reset input
V_{DD}	:	Power supply
$\overline{\text{WDOUT}}$	:	Hang-up/low voltage detection output (N-ch open-drain output)
X_{IN}, X_{OUT}	:	Resonator connection

BLOCK DIAGRAM



CONTENTS

1. PIN FUNCTIONS

1.1 Pin Function List

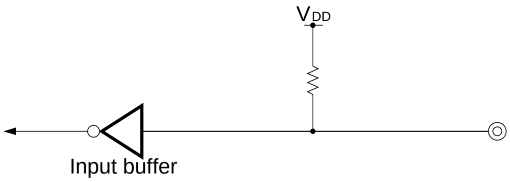
Pin No.	Symbol	Function	Output Form	On Reset
15 (16) 16 (17) 17 (18) 18 (19)	P0A$_0$ P0A$_1$ P0A$_2$ P0A$_3$	4-bit CMOS input port with pull-up resistor. Can be used for key return input of key matrix. When at least one of these pins goes low, standby function is released.	–	Input
19 (20) 20 (21) 21 (22) 22 (23)	P0B$_0$ P0B$_1$ P0B$_2$ P0B$_3$	4-bit CMOS input port with pull-up resistor. Can be used for key return input of key matrix. When at least one of these pins goes low, standby function is released.	–	Input
23 (24) 24 (25) 25 (26) 26 (27)	P0C$_0$ P0C$_1$ P0C$_2$ P0C$_3$	4-bit N-ch open-drain output port. Can be used for key source output of key matrix.	N-ch open-drain	Low-level output
27 (28) 28 (29) 1 (1) 2 (2)	P0D$_0$ P0D$_1$ P0D$_2$ P0D$_3$	4-bit N-ch open-drain output port. Can be used for key source output of key matrix.	N-ch open-drain	Low-level output
4 (4) 5 (5) 6 (6) 7 (7)	P0E$_0$ P0E$_1$ P0E$_2$ P0E$_3$	4-bit input/output port. Can be set in inputset in input or output mode in 1-bit units. In output mode, this port functions as a high current CMOS output port. In input mode, function as CMOS input and can be specified to connect pull-up resistor by program.	CMOS push-pull	Input
8 (8)	REM	Outputs transfer signal for infrared remote controller. Active-high output.	CMOS push-pull	Low-level output
13 (13)	$\overline{\text{RESET}}$	System reset input. CPU can be reset when low-level signal is input to this pin. While low-level signal is input, oscillator is stopped. Can be connected to pull-up resistor by mask option.	–	Input
9 (9)	V$_{DD}$	Power supply	–	–
12 (12)	GND	Ground	–	–
3 (3)	INT	External interrupt request signal input	–	Input
14 (14)	$\overline{\text{WDOUT}}$	Output detecting hang-up and drop in supply voltage. This pin outputs at low level either when an overflow occurs in the watch-dog timer, when an overflow/underflow occurs in the stack, or when the supply voltage drops below a specified level (mask option). Connect this pin to the $\overline{\text{RESET}}$ pin.	N-ch open-drain	High-impedance (Low-level output at low voltage detection)
11 (11) 10 (10)	X$_{IN}$ X$_{OUT}$	Connects ceramic resonator for system clock oscillation	–	(Oscillation stops)
★ (15) (30)	IC1 IC2	These pins cannot be used. Leave open.	–	–

Remark The number in parenthesis in the Pin No. column indicates the pin numbers of the 30-pin plastic SSOP.

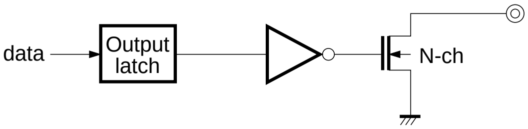
1.2 Input/Output Circuits

The equivalent input/output circuit for each μPD17225 pin is shown below.

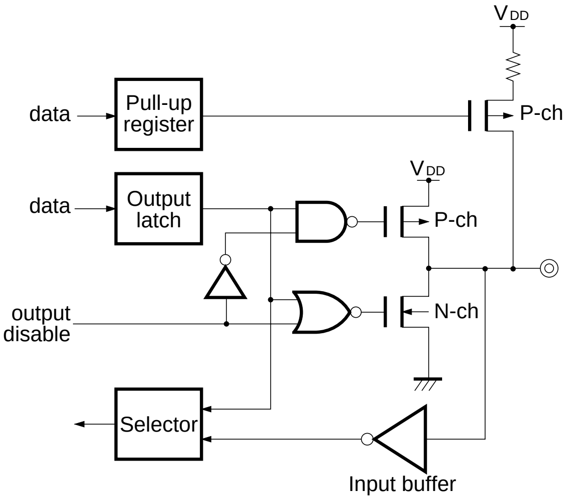
(1) P0A, P0B



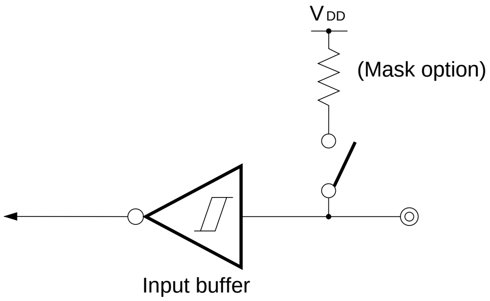
(2) P0C, P0D



(3) P0E

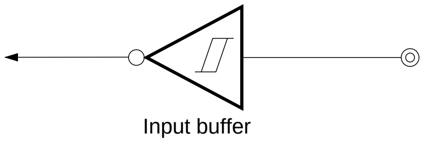


(4) $\overline{\text{RESET}}$



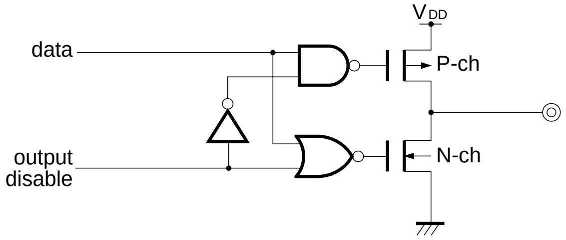
Schmitt trigger input with hysteresis characteristics

(5) INT

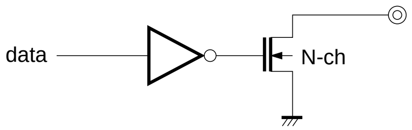


Schmitt trigger input with hysteresis characteristics

(6) REM



(7) $\overline{\text{WDOUT}}$



1.3 Processing of Unused Pins

Process the unused pins as follows:

Table 1-1. Processing of Unused Pins

Pin	Recommended Connection
$P0A_0$-$P0A_3$	Connect to V_{DD}.
$P0B_0$-$P0B_3$	Connect to V_{DD}.
$P0C_0$-$P0C_3$	Connect to GND.
$P0D_0$-$P0D_3$	Connect to GND.
$P0E_0$-$P0E_3$	Input : Individually connect to V_{DD} or GND via resistor. Output : Leave open.
REM	Leave open.
INT	Connect to GND.
$\overline{\text{WDOUT}}$	Connect to V_{DD} via resistor.
★ IC1, IC2	These pins cannot be used. Leave open.

2. MEMORY SPACE

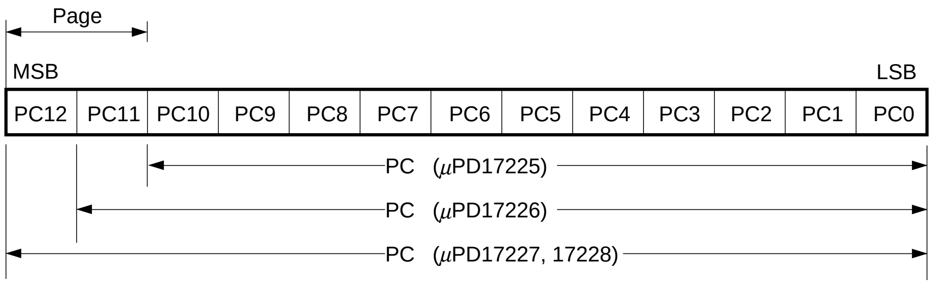
2.1 Program Counter (PC)

The program counter (PC) specifies an address of the program memory (ROM).

The program counter is an 11/12/13-bit binary counter as shown in Figure 2-1.

Its contents are initialized to address 0000H at reset.

Figure 2-1. Configuration of Program Counter



2.2 Program Memory (ROM)

The configuration of the program memory is as follows:

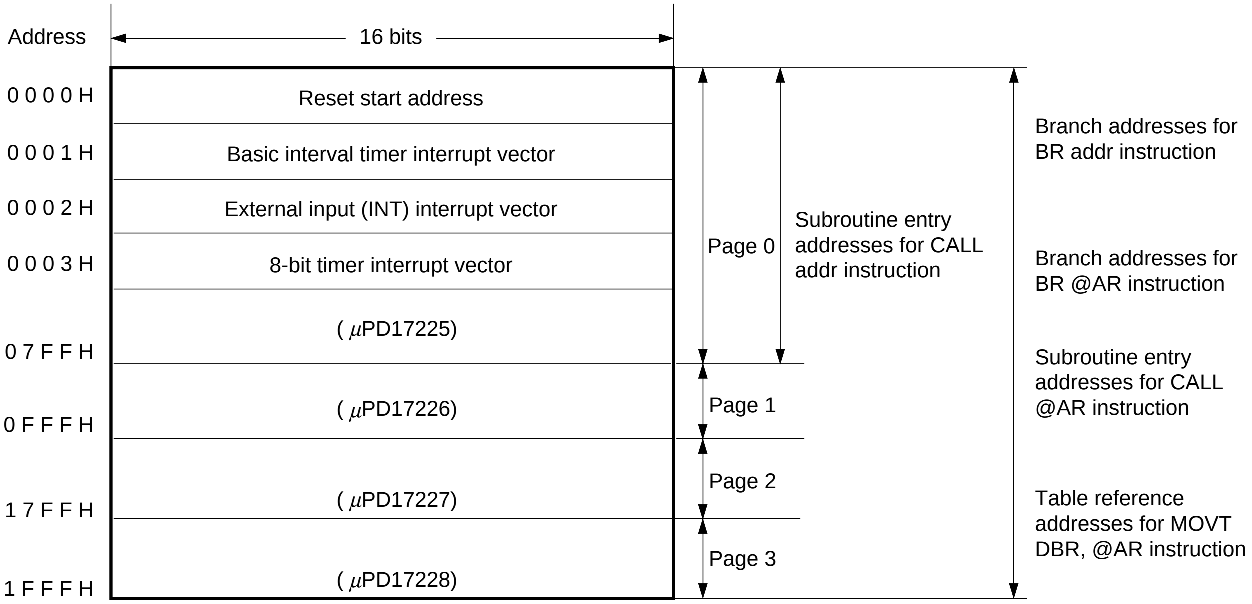
Part Number	Capacity	Address
μPD17225	2048 × 16 bits	0000H-07FFH
μPD17226	4096 × 16 bits	0000H-0FFFH
μPD17227	6144 × 16 bits	0000H-17FFH
μPD17228	8192 × 16 bits	0000H-1FFFH

The program memory stores a program, interrupt vector table, and fixed data table.

The program memory is addressed by the program counter.

Figure 2-2 shows the program memory map. The entire range of the program memory can be addressed by the BD addr, BR @AR, CALL @AR, MOVT DBF, and @AR instructions. Note, however, that the subroutine entry addresses that can be specified by the CALL addr instruction are from 0000H to 07FFH.

Figure 2-2. Program Memory Map



2.3 Stack

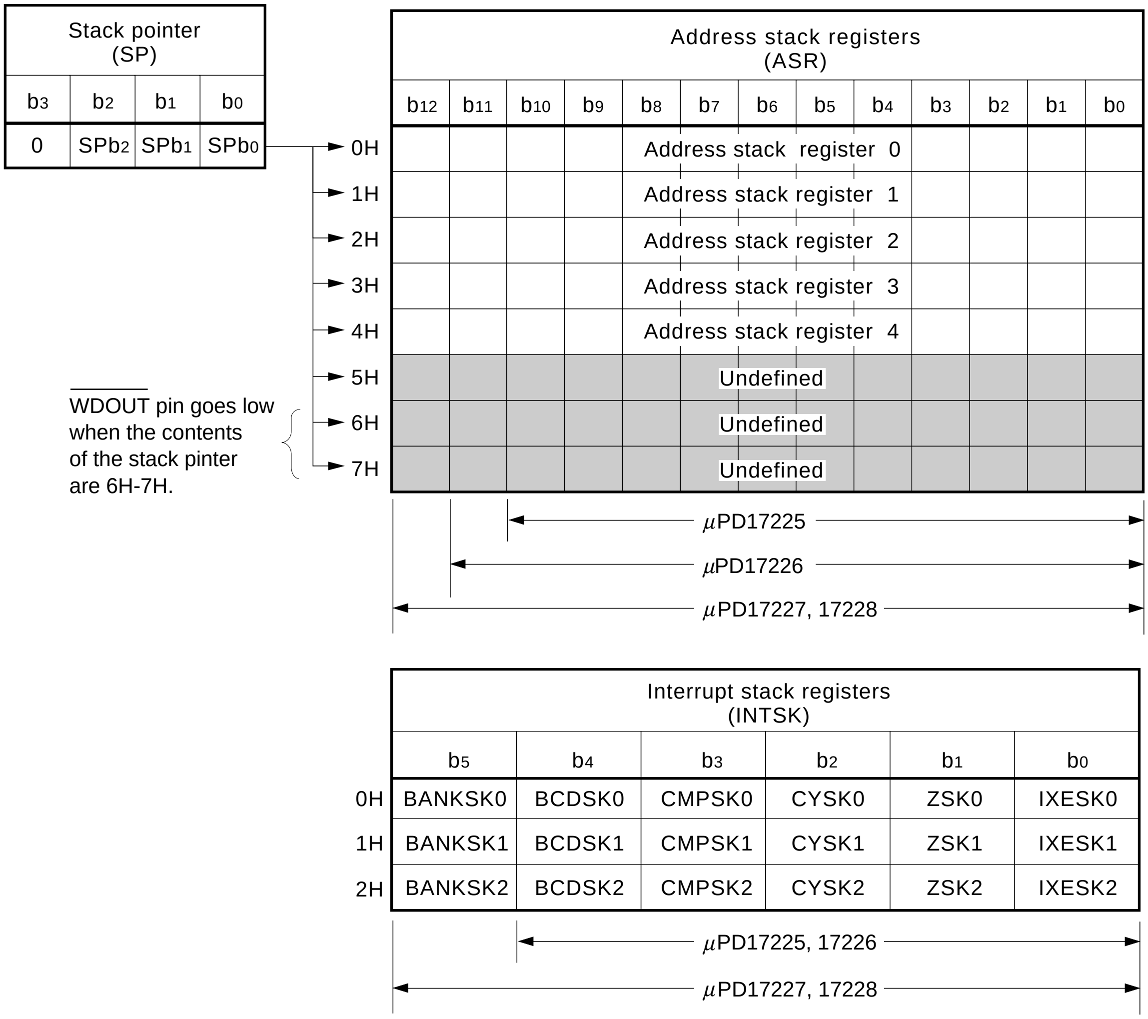
A stack is a register to save a program return address and the contents of system registers (to be described later) when a subroutine is called or when an interrupt is accepted.

2.3.1 Stack configuration

Figure 2-3 shows the stack configurarion.

A stack consists of a stack pointer (a 4-bit binary counter, the high-order 1 bit fixed to 0), five 11-bit (μPD17225)/12-bit (μPD17226)/13-bit (μPD17227, 17228) address stack registers, and three 5-bit (μPD17225, 17226)/6-bit (μPD17227, 17228) interrupt stack registers.

Figure 2-3. Stack Configuration



2.3.2 Function of stack

The address stack register stores a return address when the subroutine call instruction or table reference instruction (first instruction cycle) is executed or when an interrupt is accepted. It also stores the contents of the address registers (ARs) when a stack manipulation instruction (PUSH AR) is executed.

The $\overline{\text{WDOUT}}$ pin goes low if a subroutine call or interrupt exceeding 5 levels is executed.

The interrupt stack register (INTSK) saves the contents of the bank register (BANK) and program status word (PSWORD) when an interrupt is accepted. The saved contents are restored when an interrupt return (RETI) instruction is executed.

INTSK saves data each time an interrupt is accepted, but **the data stored first is lost if more than 3 levels of interrupts occur**.

2.3.3 Stack Pointer (SP) and Interrupt Stack Pointer

Table 2-1 shows the operations of the stack pointer (SP).

The stack pointer can take eight values, 0H-07. Because there are only five stack registers available, however, **the $\overline{\text{WDOUT}}$ pin goes low if the value of SP is 6 or greater**.

Table 2-1. Operations of Stack Pointer

Instruction	Value of Stack Pointer (SP)	Counter of Interrupt Stack Register
CALL addr CALL @AR MOVT DBF, @AR (1st Instruction Cycle) PUSH AR	–1	0
When Interrupt Is Accepted	–1	–1
RET RETSK MOVT DBF, @AR (2nd Instruction Cycle) POP AR	+1	0
RETI	+1	+1

2.4 Data Memory (RAM)

Data memory (random access memory) stores data for operations and control. It can be read-/write-accessed by instructions.

2.4.1 Memory configuration

Figure 2-4 shows the configuration of the data memory (RAM).

The data memory consists of two “banks”: BANK0 and BANK1.

In each bank, every 4 bits of data is assigned an address. The high-order 3 bits of the address indicate a “row address” and the low-order 4 bits of the address indicate a “column address”. For example, a data memory location indicated by row address 1H and column address 0AH is termed a data memory location at address 1AH. Each address stores data of 4 bits (= a “nibble”).

In addition, the data memory is divided into following six functional blocks:

(1) System register (SYSREG)

A system register (SYSREG) is resident on addresses 74H to 7FH (12 nibbles long) of each bank. In other nibbles, each bank has a system register at its addresses 74H to 7FH.

(2) Data buffer (DBF)

A data buffer is resident on addresses 0CH to 0FH (4 nibbles long) of bank 0 of data memory.
The reset value is 0320H.

(3) General register (GR)

A general register is resident on any row (16 nibbles long) of any bank of data memory.
The row address of the general register is pointed by the general pointer (RP) in the system register (SYSREG).

(4) Port register

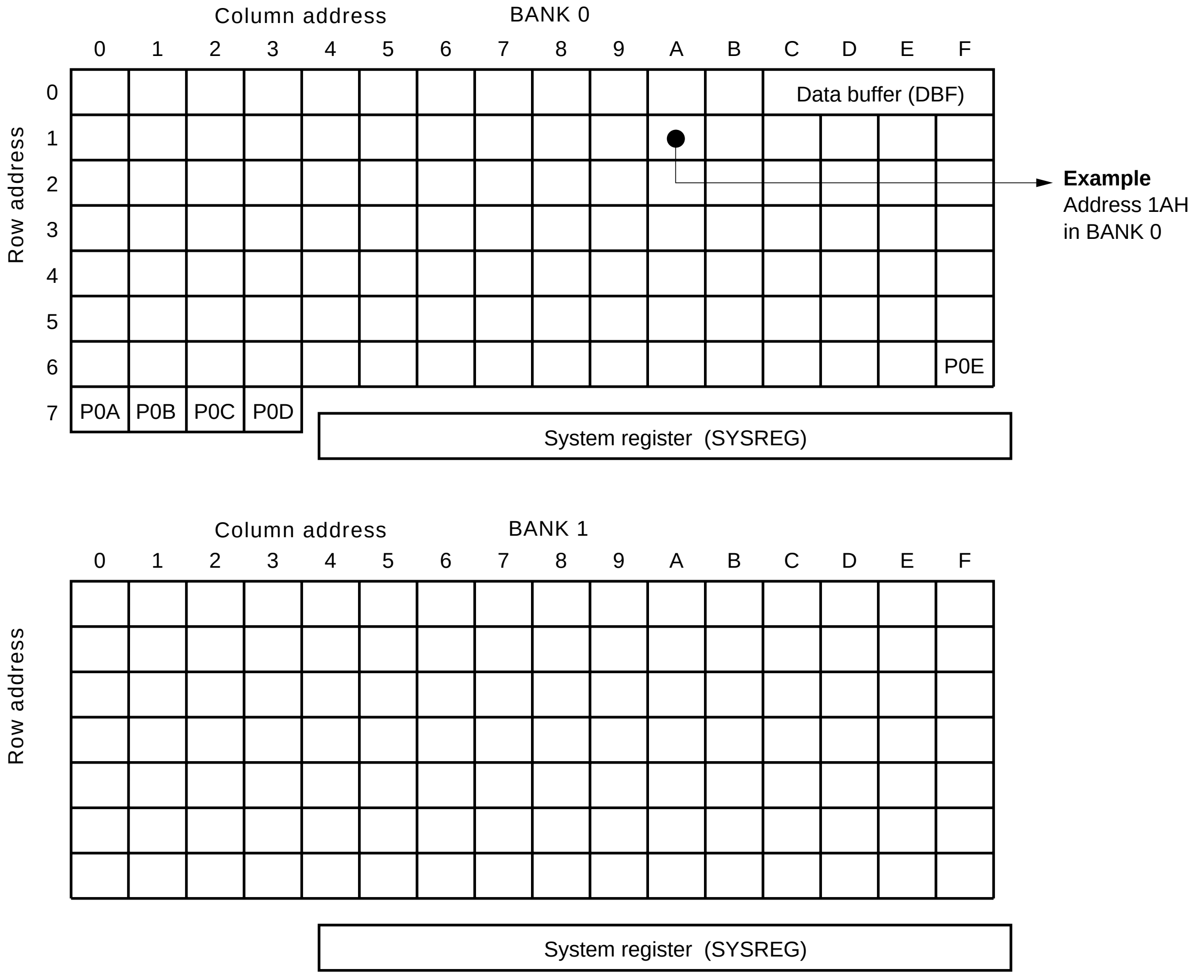
A port data register is resident on addresses 6FH, and 70H to 73H (5 nibbles) of BANK0 of data memory.
No data can be written to the addresses 70H to 73H of BANK1 (the values of addresses 70H to 73H of BANK0 are read in this case).
μPD17225 and 17226 are not provided with BANK1.

(5) General-purpose data memory

The general-purpose data memory area is an area of the data memory excluding the system register area, and the port register area. This memory area has a total of 223 nibbles (111 nibbles in BANK0 and 112 nibbles in BANK1).

μPD17225 and 17226 are not provided with BANK1.

Figure 2-4. Configuration of Data Memory



Caution **No data can be written to the addresses 70H to 73H of BANK1 (the value of P0A to P0D are read in this case).**

2.4.2 System registers (SYSREG)

The system registers are registers that are directly related to control of the CPU. These registers are mapped to addresses 74H-7FH on the data memory and can be referenced regardless of bank specification.

The system registers include the following registers:

- Address registers (AR0-AR3)**Note**
- Window register (WR)
- Bank register (BANK)**Note**
- Memory pointer enable flag (MPE)
- Memory pointers (MPH, MPL)
- Index registers (IXH, IXM, IXL)
- General register pointers (RPH, RPL)
- Program status word (PSWORD)

Note The address register (AR3) and the bank register (BANK) are fixed to 0 in the μPD17225 and 17226.

Figure 2-5. Configuration of System Register

Address	74H	75H	76H	77H	78H	79H	7AH	7BH	7CH	7DH	7EH	7FH
Name	Address register (AR)				Window register (WR)	Bank register (BANK)	Index register (IX) Data memory row address pointer (MP)			General register pointer (RP)		Program status word (PSWORD)
Symbol	AR 3	AR 2	AR 1	AR 0	WR	BANK	IXH MPH	IXM MPL	IXL	RPH	RPL	PSW
Bit	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0	b3 b2 b1 b0
Data	0 0 0 ← 0 0 0 0 0 0 0 0	(AR) (μPD17227, 17228) ← (AR) (μPD17226) 0 ←	(AR) (μPD17227, 17228) (AR) (μPD17226) (AR) (μPD17225)	→ → →	← (WR) →	(BANK) 0 0 0 ↔ *	MPE 0 0 0 * ← (IX) * ← (MP)	(IX) (MP) →	(IX) →	0 0 0 ← *	(RP) → BCD	CMP CY Z IXE
Initial Value At Reset	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	Undefined	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0

Note *: This bit is fixed to 0 in the μPD17225 and 17226.

2.4.3 General register (GR)

A general register is a 16-word register on the data memory and used for arithmetic operations and transfer of data to and from the data memory.

(1) Configuration of general register

Figure 2-6 shows the configuration of the general register.

A general register occupies 16 nibbles (16 × 4 bits) on a selected row address of the data memory.

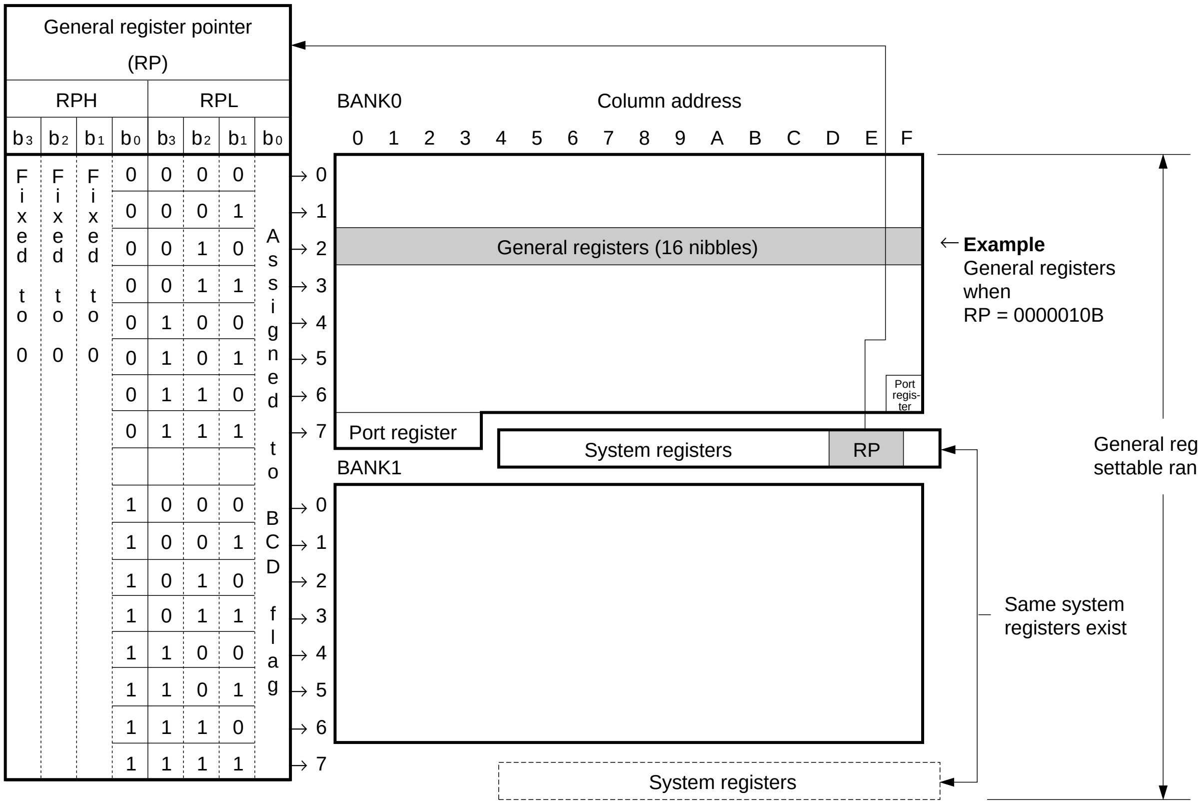
The row address is selected by the general register pointer (RP) of the system register. The RP having four significant bits in the μPD17227 and 17228 can point to any row address in the range of 0H to 7H of each bank (BANK0 and BANK1).

In the μPD17225 and 17226, 3 bits are available in the RP. These bits can point to any row address in the range of 0H to 7H of BANK0.

(2) Functions of the general register

The general register enables an arithmetic operation and data transfer between the data memory and a selected general register by a single instruction. As a general register is a part of the data memory, you can say that the general register enables arithmetic operation and data transfer between two locations of the data memory. Similarly, the general register can be accessed by a data memory manipulation instruction as it is a part of the data memory.

Figure 2-6. Configuration of General Registers



2.4.4 Data buffer (DBF)

The data buffer on the addresses 0CH to 0FH of data memory is used for data transfer to and from peripheral hardware and for storage of data during table reference.

(1) Functions of the data buffer

The data buffer has two major functions: a function to transfer to and from hardware and a function to read constant data from the program memory (for table reference). Figure 2-7 shows the relationship between the data buffer and peripheral hardware.

Figure 2-7. Data Buffer and Peripheral Hardware

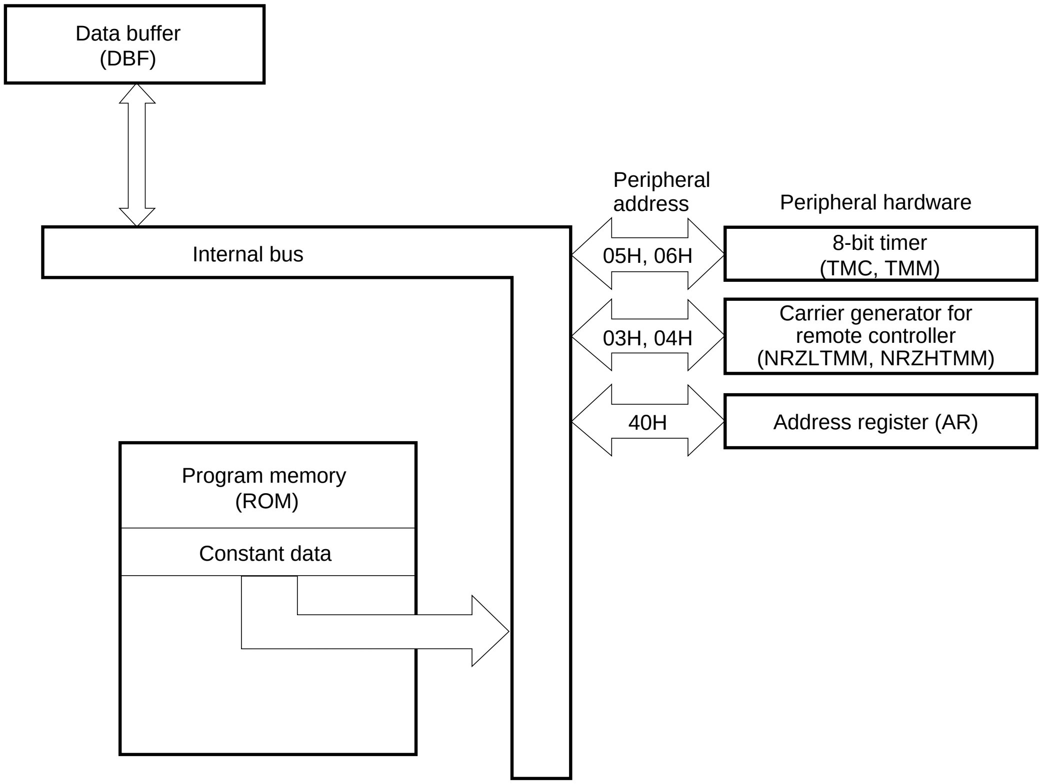


Table 2-2. Relations between Hardware Peripherals and Data Buffer

Hardware Peripherals	Peripheral Register Transferring Data with Data Buffer				
	Name	Symbol	Peripheral Address	Data Buffer Used	PUT/GET
8-Bit Timer	8-bit counter	TMC	05H	DBF0, DBF1	GET only
	8-bit modulo register	TMM	06H	DBF0, DBF1	PUT only
Remote Controller Carrier Generator	NRZ low-level timer modulo register	NRZLTMM	03H	DBF0, DBF1	PUT GET
	NRZ high-level timer modulo register	NRZHTMM	04H	DBF0, DBF1	PUT (clear bit 3 of DBF1 to 0) GET (bits 3 of DBF1 is always 0)
Address Register	Address register	AR	40H	DBF0-DBF3	PUT (bits 0-3 of AR3 and bit 3 of AR2 are any)[Note 1] GET (bits 0-3 of AR3 and bit 3 of AR2 are always 0)[Note 2]

Notes **1.** In the µPD17226: bits 0 to 3 of AR3 are any, in the µPD17227, 17228: bits 1 to 3 of AR3 are any

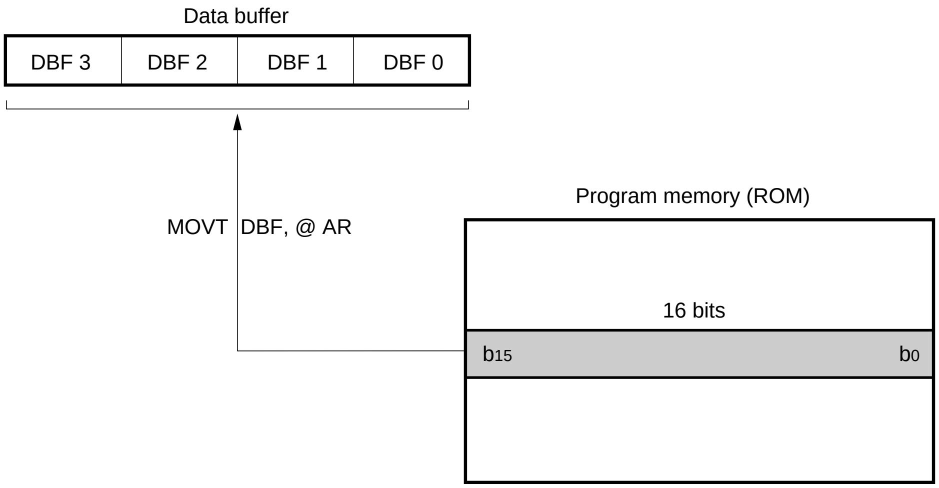
2. In the µPD17226: bits 0 to 3 of AR3 are always 0, in the µPD17227, 17228: bits 1 to 3 of AR3 are always 0

(2) Table reference

A MOVT instruction reads constant data from a specified location of the program memory (ROM) and sets it in the data buffer.

The function of the MOVT instruction is explained below.

MOVT DBF, @AR: Reads data from a program memory location pointed to by the address register (AR) and sets it in the data buffer (DBF).



(3) Note on using data buffer

When transferring data to/from the peripheral hardware via the data buffer, the unused peripheral addresses, write-only peripheral registers (only when executing PUT), and read-only peripheral registers (only when executing GET) must be handled as follows:

- **When device operates**
  Nothing changes even if data is written to the read-only register.
  If the unused address is read, an undefined value is read. Nothing changes even if data is written to that address.

- **Using assembler**
  An error occurs if an instruction is executed to read a write-only register.
  Again, an error occurs if an instruction is executed to write data to a read-only register.
  An error also occurs if an instruction is executed to read or write an unused address.

- **If an in-circuit emulator (IE-17K or IE-17K-ET) is used** (when instruction is executed for patch processing)
  An undefined value is read if an attempt is made to read the data of a write-only register, but an error does not occur.
  Nothing changes even if data is written to a read-only register, and an error does not occur.
  An undefined value is read if an unused address is read; nothing changes even if data is written to this address. An error does not occur.

2.5 Register File (RF)

The register file mainly consists of registers that set the conditions of the peripheral hardware.

These registers can be controlled by dedicated instructions PEEK and POKE, and the embedded macro instructions of RA17K, SETn, CLRn, and INITFLG.

2.5.1 Configuration of register file

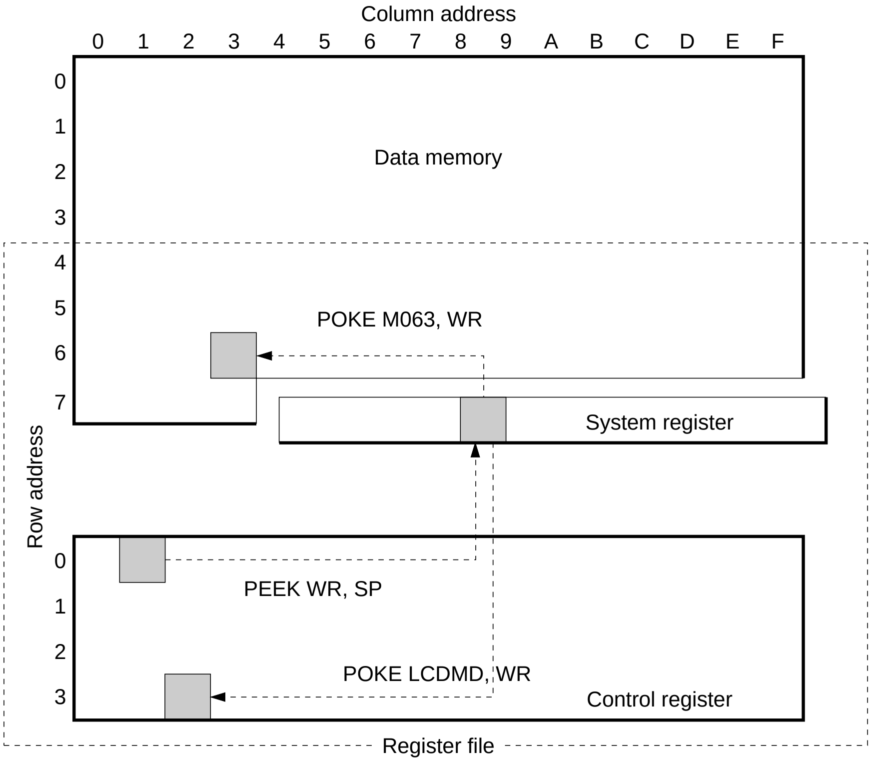
Figure 2-8 shows the configuration of the register file and how the register file is accessed by the PEEK and POKE instructions.

The control registers are controlled by using dedicated instructions PEEK and POKE. Since the control registers are assigned to addresses 00H-3FH regardless of the bank, the addresses 00H-3FH of the general-purpose data memory cannot be accessed when the PEEK or POKE instruction is used.

The addresses that can be accessed by the PEEK and POKE instructions are the addresses 00H-3FH of the control registers and 40H-7FH of the general-purpose data memory. The register file consists of these addresses.

The control registers are assigned to addresses 80H-BFH on the IE-17K to facilitate debugging.

Figure 2-8. Register File Access with PEEK or POKE Instructions



2.5.2 Control registers

The control registers consists of a total of 64 nibbles (64 x 4 bits) of the addresses 00H-3FH of the register file.

Of these, however, only 14 nibbles are actually used. The remaining 50 nibbles are unused registers that are inhibited from being read or written.

When the "PEEK WR, rf" instruction is executed, the contents of the register file addressed by "rf" are read to the window register.

When the "POKE rf, WR" instruction is executed, the contents of the window register are written to the register file addressed by "rf".

When using the assembler (RA17K), the macro instructions listed below, which are embedded as flag type symbol manipulation instructions, can be used. The macro instructions allow the contents of the register file to be manipulated in bit units.

For the configuration of the control register, refer to **Figure 11-1 Register File List**.

SETn : Sets flag to "1"
CLRn : Sets flag to "0"
SKTn : Skips if all flags are "1"
SKFn : Skips if all flags are "0"
NOTn : Complements flag
INITFLG : Initializes flag
INITFLGX : Initalizes flag

2.5.3 Notes on using register files

When using the register files, bear in mind the points described below. For details, refer to **μPD172xx subseries User's Manual (U12795E)**.

(1) When manipulating control registers (read-only and unused registers)

When manipulating the write-only (W), the read-only (R) and unused control registers by using the assembler or in-circuit emulator, keep in mind the following points:

- **When device operates**

  Nothing changes even if data is written to the read-only register.
  If the unused register is read, an undefined value is read; nothing is changed even if data is written to this register.

- **Using assembler**

  An error occurs if instruction is excecuted to read data to the write-only register.
  An error occurs if an instruction is executed to write data to the read-only register.
  An error also occurs if an instruction is executed to read or write the unused address.

- **When an in-circuit emulator (IE-17K or IE-17K-ET) is used** (when instruction is executed for patch processing)

  An undefined value is read if the write-only register is read, and an error does not occur.
  Nothing changes even if data is written to the read-only register, and an error does not occur.
  An undefined value is read if the unused address is read; nothing changes even if data is written to this address. An error does not occur.

(2) Symbol definition of register file

An error occurs if a register file address is directly specified as a numeral by the operand “rf” of the “PEEK WR, rf” or “POKE rf, WR” instruction if the 17K Series Assembler (RA17K) is being used.

Therefore, the addresses of the register file must be defined in advance as symbols.

To define the addresses of the control registers as symbols, define them as the addresses 80H-BFH of BANK0.

The portion of the register file overlapping the data memory (40H-7FH), however, can be defined as symbols as is.

3. PORTS

3.1 Port 0A (P0A$_0$-P0A$_3$)

This is a 4-bit input port. Data is read through port register P0A (address 70H). This port is a CMOS input port with a pull-up resistor, and can be used for key return input for a key matrix.

When a low-level signal is input to at least one of the pins in this port in the standby mode, the standby mode is released.

3.2 Port 0B (P0B$_0$-P0B$_3$)

This is a 4-bit input port. Data is read through port register P0B (address 71H). This port is a CMOS input port with a pull-up resistor, and can be used for key return input for a key matrix.

When a low-level signal is input to at least one of the pins in this port in the standby mode, the standby mode is released.

3.3 Port 0C (P0C$_0$-P0C$_3$)

This is a 4-bit output port. The contents of the output latch are read and output data is set through port register P0C (address 72H). This port is an N-ch open-drain output port, and can be used as the key source of a key matrix.

In the standby mode, this port outputs low-level signals.

3.4 Port 0D (P0D$_0$-P0D$_3$)

This is a 4-bit output port. The contents of the output latch are read and output data is set through port register P0D (address 73H). This port is an N-ch open-drain output port, and can be used as the key source for a key matrix.

In the standby mode, this port outputs low-level signals.

3.5 Port 0E (P0E$_0$-P0E$_3$)

This is a 4-bit I/O port which can be set in either the input or output mode in 1-bit units by the P0EBIO (address 27H) of the register file.

To read the input data or to set the output data, use the P0E register (address 6F). When data is read in the output mode, the contents of the output latch are read.

Connection of a pull-up resistor can be specified in 1-bit units by the P0EBPU (address 17H) of the register file. (When the pull-up resistor is connected, note that the pull-up resistor is not disconnected even when the output mode is set.)

On reset, this port functions as an input port.

3.6 INT Pin

This pin inputs an external interrupt request signal. At either the rising or falling edge of the signal input to this pin, the IRQ flag (RF: address 3EH, bit 0) is set.

The status of this pin can be read by using the INT flag (RF: address 0FH, bit 0). When the high level is input to the pin, the INT flag is set to “1”; when the low level is input, the flag is reset to “0” (refer to **7.2.1 INT**).

Figure 3-1. Relations between Port Register and Each Pin

Bank	Address	Port	Bit		Output Form	Contents to Be Read		Contents to Be Written		On Reset
						Input Mode	Output Mode	Input Mode	Output Mode	
0	70H	Port 0A	b_3	P0A$_3$	Input only	Pin status	–	–	–	Input mode (w/pull-up resistor)
			b_2	P0A$_2$						
			b_1	P0A$_1$						
			b_0	P0A$_0$						
	71H	Port 0B	b_3	P0B$_3$						
			b_2	P0B$_2$						
			b_1	P0B$_1$						
			b_0	P0B$_0$						
	72H	Port 0C	b_3	P0C$_3$	N-ch open-drain (Output only)	–	Output latch	–	Output latch	Output mode (Low level output)
			b_2	P0C$_2$						
			b_1	P0C$_1$						
			b_0	P0C$_0$						
	73H	Port 0D	b_3	P0D$_3$						
			b_2	P0D$_2$						
			b_1	P0D$_1$						
			b_0	P0D$_0$						
	6FH	Port 0E	b_3	P0E$_3$	COMS push-pull	Pin status		Output latch		Input mode (w/pull-up resistor)
			b_2	P0E$_2$						
			b_1	P0E$_1$						
			b_0	P0E$_0$						

3.7 Switching Bit I/O

The I/O which can be set in the input or output mode in bit units is called a bit I/O. P0E is a bit I/O port, which can be set in the input or output mode in bit units by the register file shown below. When the mode is changed from input to output, the P0E output latch contents are output to the port lines, as soon as the mode has been changed.

3	2	1	0	Address:	On reset:	R/W:
P0EBIO3	P0EBIO2	P0EBIO1	P0EBIO0	RF : 27H	0H	R/W

P0EBIO0	Sets P0E$_0$ Input/Output Mode
0	Sets P0E$_0$ in input mode
1	Sets P0E$_0$ in output mode

P0EBIO1	Sets P0E$_1$ Input/Output Mode
0	Sets P0E$_1$ in input mode
1	Sets P0E$_1$ in output mode

P0EBIO2	Sets P0E$_2$ Input/Output Mode
0	Sets P0E$_2$ in input mode
1	Sets P0E$_2$ in output mode

P0EBIO3	Sets P0E$_3$ Input/Output Mode
0	Sets P0E$_3$ in input mode
1	Sets P0E$_3$ in output mode

3.8 Specifying Pull-up Resistor Connection

Whether or not a pull-up resistor is connected to port P0E can be specified by the following registers of the register file in 1-bit units when the port is in the input mode[Note].

3	2	1	0	Address:	On reset:	R/W:
P0EBPU3	P0EBPU2	P0EBPU1	P0EBPU0	RF : 17H	0H	R/W

P0EBPU0	Connects Pull-Up Resistor to $P0E_0$
0	Not connected
1	Connected

P0EBPU1	Connects Pull-Up Resistor to $P0E_1$
0	Not connected
1	Connected

P0EBPU2	Connects Pull-Up Resistor to $P0E_2$
0	Not connected
1	Connected

P0EBPU3	Connects Pull-Up Resistor to $P0E_3$
0	Not connected
1	Connected

Note To disconnect the pull-up resistor in the output mode, clear the corresponding bit of the P0EBPU register.

4. CLOCK GENERATOR CIRCUIT

4.1 Instruction Execution Time (CPU Clock) Selection

The μPD17225 is equipped with a clock oscillator that supplies clocks to the CPU and hardware peripherals. Instruction execution time can be changed in two steps (ordinary mode and high-speed mode) without changing the oscillation frequency.

To change the instruction execution time, change the mode of SYSCK (RF: address 02H) of the register file by using the POKE instruction.

Note, that the mode is actually only changed when the instruction next to the POKE instruction has been executed.

When using the high-speed mode, pay attention to the supply voltage. (Refer to **13. ELECTRICAL SPECIFICATIONS**.)

At reset, the ordinary mode is set.

3	2	1	0	Address:	On reset:	R/W:
0	0	0	SYSCK	RF : 02H	0H	R/W

SYSCK	Selects Instruction Execution Time
0	Ordinary mode 32/f_X (8μs)
1	High-speed mode 16/f_X (4μs)

Figures in ( ): indicate figures when system clock f_X = 4 MHz.

5. 8-BIT TIMER AND REMOTE CONTROLLER CARRIER GENERATOR CIRCUIT

The μPD17225 is equipped with the 8-bit timer which is mainly used to generate the leader pulse of the remote controller signal, and to output codes.

5.1 Configuration of 8-bit Timer (with modulo function)

Figure 5-1 shows the configuration of the 8-bit timer.

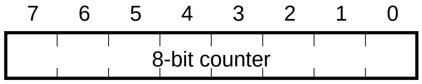
As shown in this figure, the 8-bit timer consists of an 8-bit counter (TMC), an 8-bit modulo register (TMM), a comparator that compares the value of the timer with the value of the modulo register, and a selector that selects the operation clock of the 8-bit timer.

To start/stop the 8-bit timer, and to reset the 8-bit counter, TMEN (address 33H, bit 3) and TMRES (address 33H, bit 2) of the register file are used. To select the operation clock of the 8-bit timer, use TMCK1 (address 33H, bit 1) and TMCK0 (address 33H, bit 0) of the register file.

The value of the 8-bit counter is read by using the GET instruction through DBF (data buffer). No value can be set to the 8-bit counter. A value is set to the modulo register by using the PUT instruction through DBF. The value of the modulo register cannot be read.

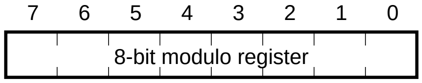
When the value of the counter coincides with that of the modulo register, an interrupt flag (IRQTM: address 3FH, bit 0) of the register file is set.

TMC



Address	On reset	R/W
Peripheral register: 05H	00H	R

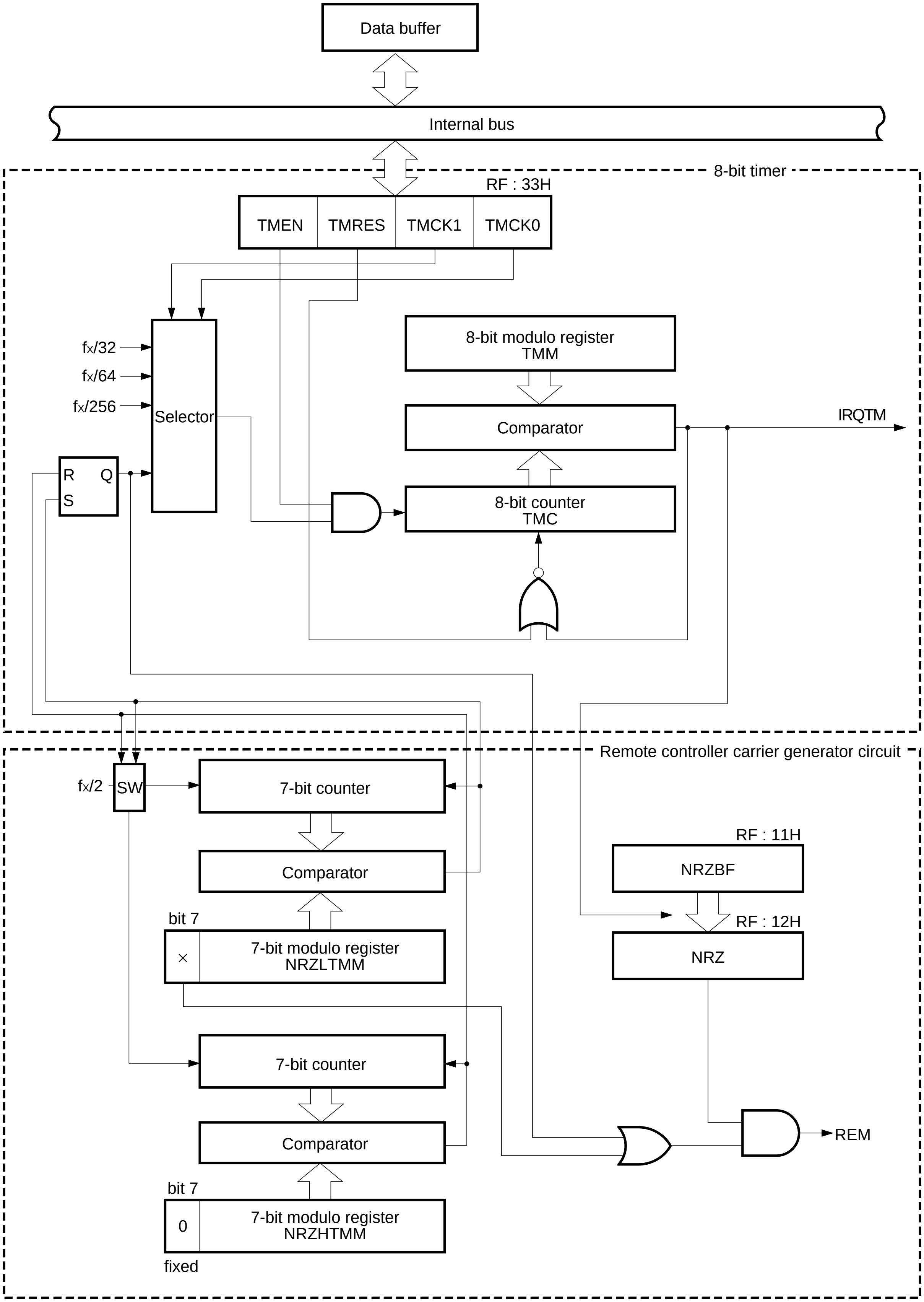
TMM



Address	On reset	R/W
Peripheral register: 06H	FFH	W

Caution Do not clear TMM to 0 (IRQTM is not set).

Figure 5-1. Configuration of 8-bit Timer and Remote Controller Carrier Generator Circuit



Remark TMM, TMC, NRZLTMM, and NRZHTMM are peripheral registers.

5.2 Function of 8-bit Timer (with modulo function)

3	2	1	0	Address	On reset	R/W
TMEN	TMRES	TMCK1	TMCK0	RF : 33H	8H[Note 1]	R/W[Note 2]

TMCK1	TMCK0	8-Bit Timer Clock Source Selection
0	0	Count clock: f_X/32 (Measurement time range: 8 μs to 2.048 ms)
0	1	Count clock: f_X/64 (Measurement time range: 16 μs to 4.096 ms)
1	0	Count clock : f_X/256 (Measurement time range: 64 μs to 16.384 ms)
1	1	Remote control carrier generation circuit output

Value indicated by parentheses is for when
( ): f_{SYS} (system clock) = f_X = 4MHz

TMRES	8-Bit Timer Reset Flag
0	Data read out is always "0"
1	Resets 8-bit counter and IRQTM

TMEN	8-Bit Timer Count Enable Flag
0	Stops 8-bit timer count operation
1	Enable 8-bit timer count operation (falling edge)

Notes 1. When the STOP mode is released, bit 3 must be set.

2. Bit 2 is a write-only bit.

NRZLTMM

7	6	5	4	3	2	1	0	Address	On reset	R/W
×	7-bit modulo register							Peripheral register: 03H	Undefined	R/W

Bit 7	Output Control of REM Pin
0	When NRZ = 1, carrier output to REM pin
1	When NRZ = 1, high-level output to REM pin

NRZHTMM

7	6	5	4	3	2	1	0	Address	On reset	R/W
0	7-bit modulo register							Peripheral register: 04H	Undefined	R/W

Bit 7
Fixed to 0

5.3 Carrier Generator Circuit for Remote Controller

μPD17225 is provided with a carrier generator circuit for the remote controller.

The remote controller carrier generator circuit consists of a 7-bit counter, NRZ high-level timer modulo register (NRZHTMM), and NRZ low-level timer modulo register (NRZLTMM). The high-level and low-level periods are set in the corresponding modulo registers through the DBF to determine the carrier duty factor and carrier frequency.

The system clock (f_X) is divided by two and is input to the 7-bit counter. Therefore, when a 4-MHz resonator is used, 2 MHz (0.5 μs) is input to the counter as the clock; when a 32-kHz oscillator (f_{XT}) is used, 16 kHz is input.

The NRZ high-level output timer modulo register is called NRZHTMM, and the NRZ low-level timer modulo register is called NRZLTMM. Data is written to these registers by the PUT instruction. The contents for these register are read by the GET instruction.

Bit 7 of NRZLTMM specifies whether the carrier or high level is output to the REM pin. To output the carrier, be sure to clear bit 7 to 0.

5.3.1 Remote controller signal output control

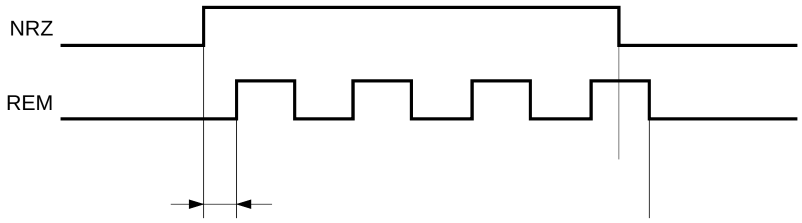
The REM pin, which outputs the carrier, is controlled by bits NRZ and NRZBF for the register file and timer 0. While the NRZ content is “1”, the clock generated by the remote controller carrier generator circuit is output to the REM pin; while the NRZ content is “0”, the REM pin outputs a low level. The NRZBF content is automatically transferred to NRZ by the interrupt signal generated by timer 0. If data is set in NRZBF in advance, the REM pin status changes in synchronization with the timer 0 counting operation.

If the interrupt signal is generated from timer 0 with the REM pin at the high level, NRZ being “1”, and the carrier clock at the high level, the REM pin output is not in accordance with the updated content of NRZ, until the carrier clock goes low. This processing is useful for holding the high level pulse width from the output carrier constant (refer to the figure below).

When the content of NRZ is “0”, the remote controller carrier generator circuit stops. However, if the clock for timer 0 is output from the remote controller carrier generator circuit, the clock continues to operate, even when the NRZ content becomes “0”.

An actual example showing a remote controller signal output to the REM pin is presented below.

When bit 7 of NRZLTMM is 0 (carrier output)



MAX. 500 ns (delay)[Note]
(fx = 4 MHz)

REM pin does not go low
until carrier goes low
even if NRZ becomes 0

Note Value when (TMCK1, TMCK0) ≠ (1, 1).
When (TMCK1, TMCK0) = (1, 1), the value differs depending on how NRZ is manipulated. If NRZ is set by an instruction, the width of the first high-level pulse may be shortened. If NRZ is set by data transferred from NRZBF, the high-level pulse is delayed by the low-level pulse of the carrier clock.

When bit 7 of NRZLTMM is 0 (carrier not output)

3	2	1	0	Address	On reset	R/W
0	0	0	NRZ	RF : 12H	0H	R/W

NRZ	NRZ Data
0	Outputs low level to REM pin
1	Outputs a carrier to REM pin or high level output

3	2	1	0	Address	On reset	R/W
0	0	0	NRZBF	RF : 11H	0H	R/W

NRZBF	NRZ Data Output Next
0	NRZ buffer bit. Transfered to NRZ by interrupt signal for 8-bit timer.
1	

Setting carrier frequency and duty factor

Where the system clock frequency is f_X and carrier frequency is f_C:

ℓ (division ratio) = $f_X/(2 \times f_C)$

ℓ is divided into m:n and is set in the modulo registers as follows:

High-level period set value = $\{\ell \times m/(m + n)\} - 1$
Low-level period set value = $\{\ell \times n/(m + n)\} - 1$

Example Where f_C = 38 kHz, duty factor (high-level period) = 1/3, and f_X = 4 MHz,

ℓ = 4 MHz/(2 × 38 kHz) = 52.6
m:n = 1:2

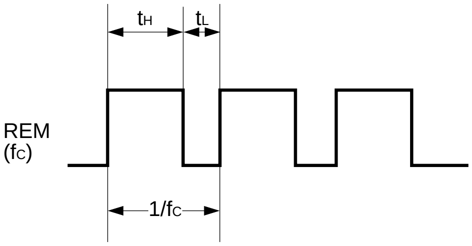
From the above, the value of the modulo register is:

High-level period ≒ 17
Low-level period ≒ 34

Therefore, the carrier frequency is 37.74 kHz.

Table 5-1. Carrier Frequency List (f_X = 4 MHz)

Set value		t_H (μs)	t_L (μs)	$1/f_C$ (μs)	f_C (kHz)	Duty
NRZHTMM	NRZLTMM					
00H	00H	0.5	0.5	1.0	1000	1/2
01H	02H	1.0	1.5	2.5	400	2/5
04H	04H	2.5	2.5	5.0	200	1/2
09H	09H	5.0	5.0	10.0	100	1/2
0FH	10H	8.0	8.0	16.5	60.6	1/2
0FH	21H	8.0	17.0	25.0	40.0	1/3
11H	21H	9.0	17.0	26.0	38.5	1/3
11H	22H	9.0	17.5	26.5	37.7	1/3
19H	35H	13.0	27.0	40.0	25.0	1/3
3FH	3FH	32.0	32.0	64.0	15.6	1/2
7FH	7FH	64.0	64.0	120.0	7.8	1/2



5.3.2 Countermeasures against noise during transmission (carrier output)

When a signal is transmitted from the transmitter of a remote controller, a peak current of 0.5 to 1 A may flow through the infrared LED. Since two batteries are usually used as the power source of the transmitter, several Ω of equivalent resistance (r) exists in the power source as shown in Figure 5-2. This resistance increases to 10 to 20 Ω if the supply voltage drops to 2 V. While the carrier is output from the REM pin (while the infrared LED lights), therefore, a high-frequency noise may be generated on the power lines due to the voltage fluctuation that may take place especially during switching.

To minimize the influence on the microcontroller of this high-frequency noise, take the following measures:

<1> Separate the power lines of the microcontroller from the power lines of the infrared LED with the terminals of the batteries at the center. Use thick power lines and keep the wiring short.

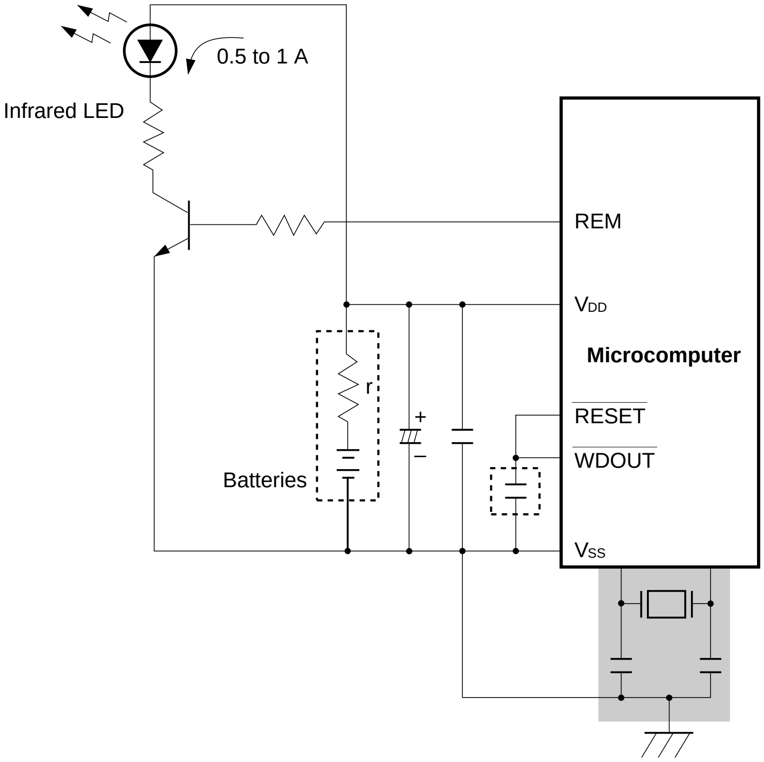
<2> Locate the resonator as close as possible to the microcontroller and shield it with GND lines (as indicated by the shaded portion in the figure below).

<3> Locate the capacitor for stabilization of the power supply closely to the power lines of the microcontroller. Also, use a capacitor to eliminate high-frequency noise.

<4> To prevent data from changing, do not execute an interrupt that requires read/write processing and stack, such as key scan interrupt, and the CALL/RET instruction, while the carrier is output.

<5> To improve the reliability in case of program hang-up, use the watchdog timer (connect the $\overline{\text{WDOUT}}$ and $\overline{\text{RESET}}$ pins).

Figure 5-2. Example of Countermeasures against Noise



Remark In this figure, the $\overline{\text{RESET}}$ pin is connected to a pull-up resistor by mask option.

6. BASIC INTERVAL TIMER/WATCHDOG TIMER

The basic interval timer has a function to generate the interval timer interrupt signal and watchdog timer reset signal.

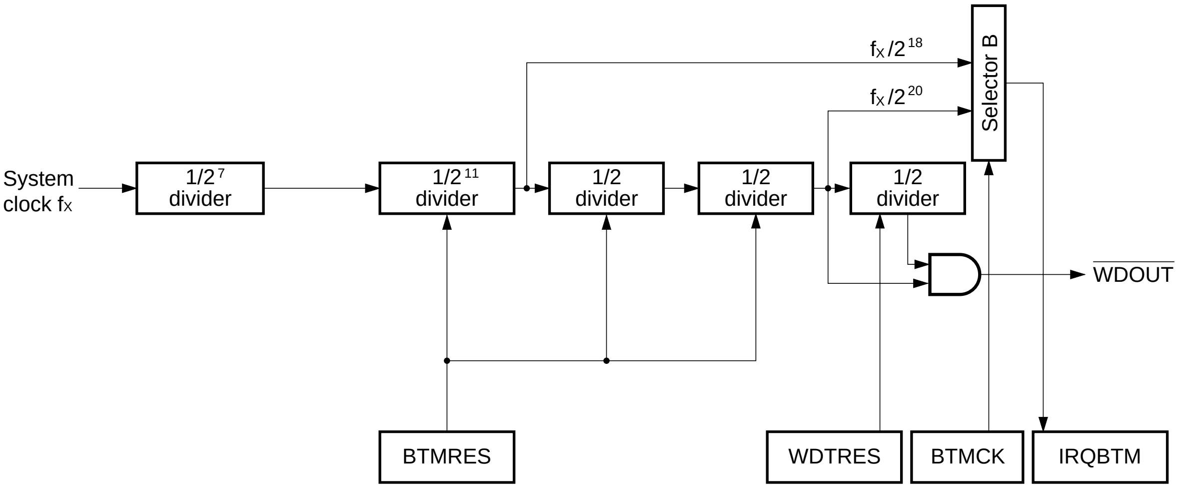
6.1 Source Clock for Basic Interval Timer

The system clock (f_X) is divided, to generate the source clock for the basic interval timer. The input clock frequency for the basic interval timer is $f_X/2^7$. When the CPU is set in the STOP mode, the basic interval timer also stops.

6.2 Controlling Basic Interval Timer

The basic interval timer is controlled by the bits on the register file. That is, the basic interval timer is reset by BTMRES. The frequency for the interrupt signal, output by the basic interval timer, is selected by BTMMD, and the watchdog timer is reset by WDTRES.

Figure 6-1. Basic Interval Timer Configuration



3	2	1	0	Address	On reset	R/W
WDTRES	BTMCK	BTMRES	0	RF : 03H	0H	R/W[Note]

BTMRES	Basic Interval Timer Reset
0	Data read out is always "0"
1	Writing "1" resets basic interval timer

BTMCK	Basic Interval Timer Mode Selection
0	Generates interrupt signal IRQBTM every $f_X/2^{20}$
1	Generates interrupt signal IRQBTM every $f_X/2^{18}$

WDTRES	Watchdog Timer Reset
0	Data read out is always "0"
1	Writing "1" resets watchdog timer ($f_X/2^{21}$ counter)

Note Bits 1 and 3 are write-only bits.

6.3 Operation Timing for Watchdog Timer

The basic interval timer can be used as a watchdog timer.

Unless the watchdog timer is reset within a fixed time[Note], it judges that "the program has hung up", and the μPD17225 is reset. It is therefore necessary to reset through programming the watchdog timer with in a fixed time.

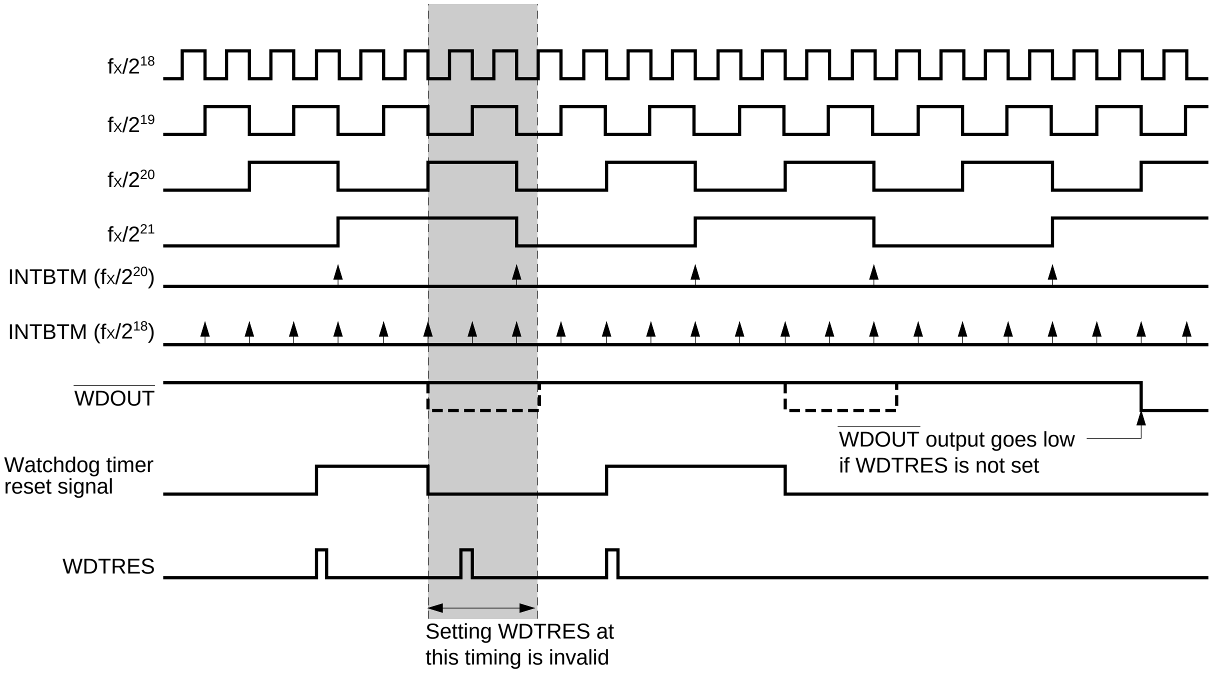
The watchdog timer can be reset by setting WDTRES to 1.

Note Fixed time: approx. 340 ms (at 4 MHz)

Cautions 1. The watchdog timer cannot be reset in the shaded range in Figure 6-2. Therefore, set WDTRES before both the fx/2^{21} and fx/2^{20} signals go high.

2. Refer to 9. RESET for the $\overline{\text{WDOUT}}$ pin function.

Figure 6-2. Watchdog Timer Operation Timing



7. INTERRUPT FUNCTIONS

7.1 Interrupt Sources

μPD17225 is provided with three interrupt sources.

When an interrupt has been accepted, the program execution automatically branches to a predetermined address, which is called a vector address. A vector address is assigned to each interrupt source, as shown in Table 7-1.

Table 7-1. Vector Address

Priority	Interrupt Source	Ext/Int	Vector Address
1	8-bit timer	Internal	0003H
2	INT pin rising and falling edges	External	0002H
3	Basic interval timer	Internal	0001H

When more than one interrupt request is issued at the same time, the interrupts are accepted in sequence, starting from the one with the highest priority.

Whether an interrupt is enabled or disabled is specified by the EI or DI instruction. The basic condition under which an interrupt is accepted is that the interrupt is enabled by the EI instruction. While the DI instruction is executed, or while an interrupt is accepted, the interrupt is disabled.

To enable accepting an interrupt after the interrupt has been processed, the EI instruction must be executed before the RETI instruction. Accepting the interrupt is enabled by the EI instruction after the instruction next to the EI instruction has been executed. Therefore, no interrupt can be accepted between the EI and RETI instructions.

Caution In interrupt processing, only the BCD, CMP, CY, Z, IXE flags are automatically saved to the stack by the hardware, to a maximum of three levels. Also, within the interrupt processing contents, when peripheral hardware (timer, A/D converter, etc.) is accessed, the DBF and WR contents are not saved by the hardware. Accordingly, it is recommended that at the beginning of interrupt processing DBF and WR be saved by software to RAM, and immediately before finishing interrupt processing the saved contents be returned to thier original location.

7.2 Hardware of Interrupt Control Circuit

This section describes the flags of the interrupt control circuit.

(1) Interrupt request flag and interrupt enable flag

The interrupt request flag (IRQ×××) is set to 1 when an interrupt request is generated, and is automatically cleared to 0 when the interrupt processing is excuted.
An interrupt enable flag (IP×××) is provided to each interrupt request flag. When the IPxxx flag is 1, the interrupt is enabled; when it is 0, the interrupt is disabled.

(2) EI/DI instruction

Whether an accepted interrupt is executed or not is specified by the EI or DI instruction.
When the EI instruction is executed, INTE (interrupt enable flag), which enables the interrupt, is set to 1. The INTE flag is not registered on the register file. Consequently, the status of this flag cannot be checked by an instruction.
The DI flag clears the INTE flag to 0 to disable all the interrupts.
The INTE flag is also cleared to 0 at reset, disabling all the interrupts.

Table 7-2. Interrupt Request Flags and Interrupt Enable Flag

Interrupt Request Flag	Signal Setting Interrupt Request Flag	Interrupt Enable Flag
IRQTM	Reset by 8-bit timer.	IPTM
IRQ	Set when edge of INT pin input signal is detected	IP
IRQBTM	Reset by basic interval timer.	IPBTM

7.2.1 INT

This flag reads the INT pin status.
When a high level is input to the INT pin, this flag is set to “1”; when a low level is input, the flag is reset to “0”.

3	2	1	0	Address	On reset	R/W
0	0	0	INT	RF : 0FH	Undefined	R

INT	INT Pin Level Detection
0	INT pin : Low level
1	INT pin : High level

7.2.2 IEG

This pin selects the interrupt edge to be detected on the INT pin.

When this flag is “0”, the interrupt is detected at the rising edge; when it is “1”, the interrupt is detected at the falling edge.

3	2	1	0	Address	On reset	R/W
0	0	0	IEG	RF : 1FH	0H	R/W

IEG	INT Pin Interrupt Detection Edge Selection
0	Rising edge of INT pin
1	Falling edge of INT pin

7.2.3 Interrupt enable flag

This flag enables each interrupt source. When this flag is “1”, the corresponding interrupt is enabled; when it is “0”, the interrupt is disabled.

3	2	1	0	Address	On reset	R/W
0	IPBTM	IP	IPTM	RF : 2FH	0H	R/W

IPTM	8-Bit Timer Interrupt Enable Flag
0	Disables interrupt acceptance by 8-bit timer
1	Enables interrupt acceptance by 8-bit timer

IP	INT Pin Interrupt Enable Flag
0	Disables interrupt acceptance by INT pin input
1	Enables interrupt acceptance by INT pin input

IPBTM	Basic Interval Timer Interrupt Enable Flag
0	Disables interrupt acceptance by basic interval timer
1	Enables interrupt acceptance by basic interval timer

7.2.4 IRQ

This is an interrupt request flag that indicates the interrupt request status.

When an interrupt request is generated, this flag is set to “1”. When the interrupt has been accepted, the interrupt request flag is reset to “0”.

The interrupt request flag can be read or written by the program. Therefore, when it is set to “1”, an interrupt can be generated by the software. By writing “0” to the flag, the interrupt pending status can be canceled.

3	2	1	0	Address	On reset	R/W
0	0	0	IRQBTM	RF : 3DH	0H	R/W

IRQBTM	Basic Interval Timer Interrupt Request Flag
0	Interrupt request has not been made.
1	Basic interval timer interrupt request has been made.

3	2	1	0	Address	On reset	R/W
0	0	0	IRQ	RF : 3EH	0H	R/W

IRQ	INT Pin Interrupt Request Flag
0	Interrupt request has not been made.
1	Interrupt request has been made at rising edge or falling edge of INT input.

3	2	1	0	Address	On reset	R/W
0	0	0	IRQTM	RF : 3FH	1H[Note]	R/W

IRQTM	8-Bit Timer Interrupt Request Flag
0	Interrupt request has not been made.
1	8-bit timer interrupt request has been made.

Note 1H is also set after releasing STOP mode.

7.3 Interrupt Sequence

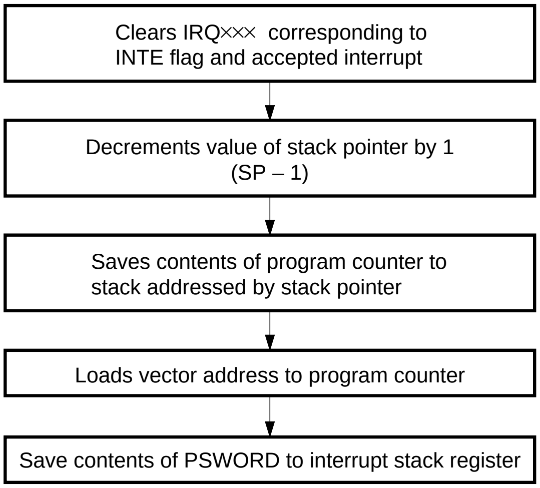
If IRQ×× flag is set to “1” when IP×× flag is “1”, interrupt processing is started after the instruction cycle of the instruction executed when IRQ×× flag was set has ended. Since the MOVT instruction, EI instruction, and the instruction which matches the condition to skip use two instruction cycles, the interrupt enabled while this instruction is executed is processed after the second instruction cycle is over.

If IP×× flag is “0”, the interrupt processing is not performed even if IRQ×× flag is set, until IP×× flag is set.

If two or more interrupts are enabled simultaneously, the interrupts are processed starting from the one with the highest priority. The interrupt with the lower priority is kept pending until the processing of the interrupt with the higher priority is finished.

7.3.1 Operations when interrupt is accepted

When an interrupt has been accepted, the CPU performs processing in the following sequence:

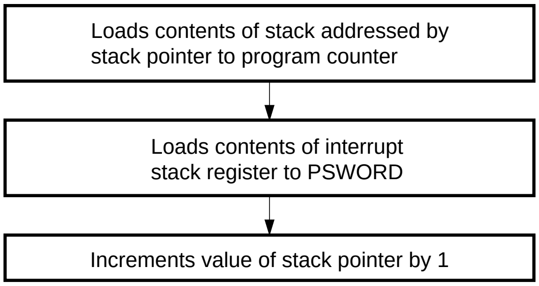


One instruction cycle is required to perform the above processing.

7.3.2 Returning from interrupt processing routine

To return from an interrupt processing routine, use the RETI instruction.

Then the following processing is executed within an instruction cycle.



To enable an interrupt after the processing of an interrupt has been finished, the EI instruction must be executed immediately before the RETI instruction.

Accepting the interrupt is enabled by the EI instruction after the instruction next to the EI instruction has been executed. Therefore, the interrupt is not accepted between the EI and RETI instructions.

8. STANDBY FUNCTIONS

μPD17225 is provided with HALT and STOP modes as standby functions. By using the standby function, current consumption can be reduced.

In the HALT mode, the program is not executed, but the system clock f_X is not stopped. This mode is maintained, until the HALT mode release condition is satisfied.

In the STOP mode, the system clock is stopped and program execution is stopped. This mode is maintained, until the STOP mode release condition is satisfied.

The HALT mode is set, when the HALT instruction has been executed. The STOP mode is set, when the STOP instruction has been executed.

8.1 HALT Mode

In this mode, program execution is temporarily stopped, with the main clock continuing oscillating, to reduce current consumption. Use the HALT instruction to set the HALT mode.

The HALT mode releasing condition can be specified by the operand for the HALT instruction, as shown in Table 8-1.

After the HALT mode has been released, the operation is performed as shown in Table 8-1 and Figure 8-2.

Caution Do not execute an instruction that clears the interrupt request flag (IRQ×××) for which the interrupt enable flag (IP×××) is set immediately before the HALT 8H instruction; otherwise, the HALT mode may not be set.

Table 8-1. HALT Mode Releasing Conditions

Operand Value	Releasing Conditions
0010B (02H)	When interrupt request (IRQTM) occurs for 8-bit timer
1000B (08H)	<1> When interrupt request (IRQTM, IRQWTM, or IRQ), whose interrupt enable flag (IPTM, IPBTM, or IP) is set, occurs <2> When any of $P0A_0$-$P0A_3$ and $P0B_0$-$P0B_3$ pins goes low
Other Than Above	Setting prohibited

Table 8-2. Operations After HALT Mode Release (1/2)

(a) HALT 08H

HALT Mode Released by:	Interrupt Status	Interrupt Enable Flag	Operations after HALT Mode Release
Low-Level Input of $P0A_0$-$P0A_3$, $P0B_0$-$P0B_3$	Don’t care	Don’t care	Instruction next to HALT is executed
When Release Condition Is Satisfied by Interrupt	DI	Disabled	Standby mode is not released
		Enabled	Instruction next to HALT is executed
	EI	Disabled	Standby mode is not released
		Enabled	Branches to interrupt vector address

Table 8-2. Operations After HALT Mode Release (2/2)

(b) HALT 02H

HALT Mode Released by:	Interrupt Status	Interrupt Enable Flag	Operations after HALT Mode Release
8-Bit Timer	DI	Disabled	Instructions are executed from the instruction next to the HALT instruction.
		Enabled	
	EI	Disabled	
		Enabled	Branches to interrupt vector address

8.2 HALT Instruction Execution Conditions

The HALT instruction can be executed, only under special conditions, as shown in Table 8-3, to prevent the program from hangup.

If the conditions in Table 8-3 are not satisfied, the HALT instruction is treated as an NOP instruction.

Table 8-3. HALT Instruction Execution Conditions

Operand Value	Execution Conditions
0010B (02H)	When all interrupt request flags (IRQTM) of 8-bit timer are reset
1000B (08H)	<1> When interrupt request flag is reset, corresponding to interrupt whose interrupt enable flag (IPTM, IPBTM, or IP) is set <2> When high level is input to all $P0A_0$-$P0A_3$ and $P0B_0$-$P0B_3$ pins
Other Than Above	Setting prohibited

8.3 STOP Mode

In the STOP mode, the system clock (f_X) oscillation is stopped and the program execution is stopped to minimize current consumption.

To set the STOP mode, use the STOP instruction.

The STOP mode releasing condition can be specified by the STOP instruction operand, as shown in Table 8-4.

After the STOP mode has released, the operation is performed as follows:

<1> Resets IRQTM.
<2> Starts the basic interval timer and watchdog timer (does not reset).
<3> Resets and starts the 8-bit timer.
<4> Executes the instruction next to [STOP 8H] when the current value of the 8-bit counter coincides with the value of the modulo register (IRQTM is set).

The μPD17225 oscillator is stopped, when the STOP instruction has been executed (i.e., in the STOP mode). Oscillation is not resumed, until the STOP mode is released. After the STOP mode has been released, the HALT mode is set. Set the time required to release the HALT mode by using the timer with modulo function.

The time that elapses, after the STOP mode has been released by occurrence of an interrupt, until an operation mode is set, is shown in the following table.

8-Bit Modulo Register Set Value (TMM)	Time Required to Set Operation Mode after STOP Mode Release
	At 4 MHz
40H	4.160 ms (64 μs × 65)
FFH	16.384 ms (64 μs × 256)

Remark Set the 8-bit modulo timer before executing STOP instruction.

Caution Do not execute an instruction that clears the interrupt request flag (IRQ×××) for which the interrupt enable flag (IP×××) is set immediately before the STOP 8H instruction; otherwise, the STOP mode may not be set.

Table 8-4. STOP Mode Releasing Conditions

Operand Value	Releasing Conditions
1000B (08H)	When any of $P0A_0$-$P0A_3$ and $P0B_0$-$P0B_3$ pins goes low
Other Than Above	Setting prohibited

8.4 STOP Instruction Execution Conditions

The STOP instruction can be executed, only under special conditions, as shown in Table 8-5, to prevent the program from hang-up.

If the conditions in Table 8-5 are not satisfied, the STOP instruction is treated as an NOP instruction.

Table 8-5. STOP Instruction Execution Conditions

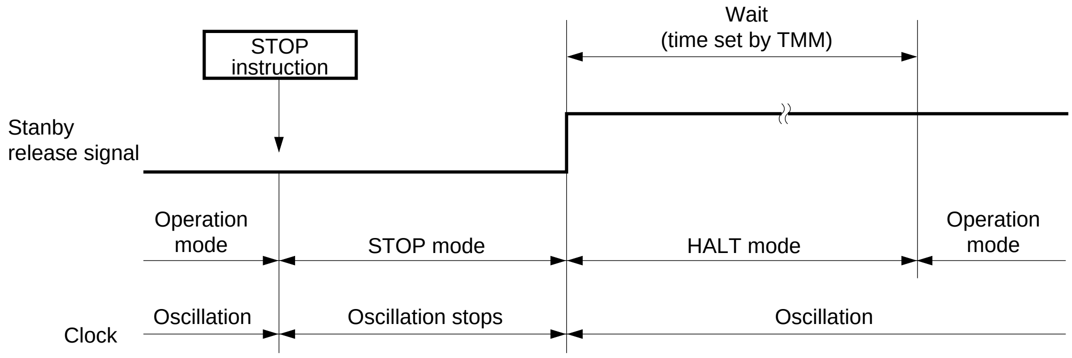
Operand Value	Execution Conditions
1000B (08H)	High level input for all $P0A_0$-$P0A_3$ and $P0B_0$-$P0B_3$ pins
Other Than Above	Setting prohibited

8.5 Releasing Standby Mode

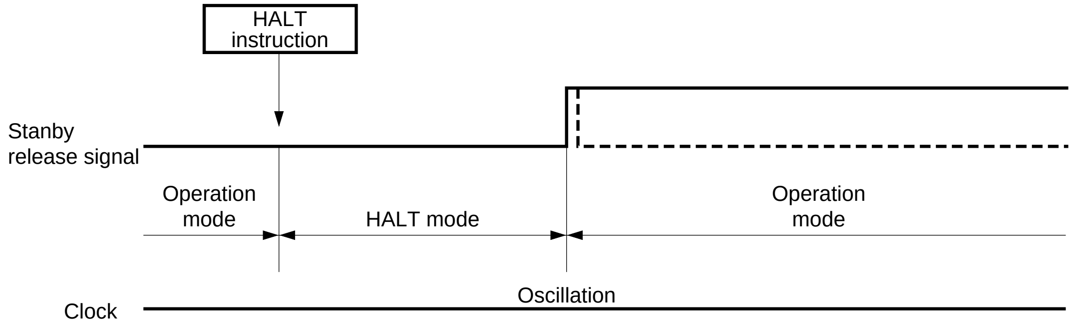
Operations for releasing the STOP and HALT modes will be as shown in Figure 8-1.

Figure 8-1. Operations After Standby Mode Release

(a) Releasing STOP mode by interrupt



(b) Releasing HALT mode by interrupt



Remark The dotted line indicates the operation to be performed when the interrupt request, releasing the standby mode, has been accepted.

9. RESET

9.1 Reset by Reset Signal Input

When a low-level signal more than 50 μs is input to the $\overline{\text{RESET}}$ pin, μPD17225 is reset.

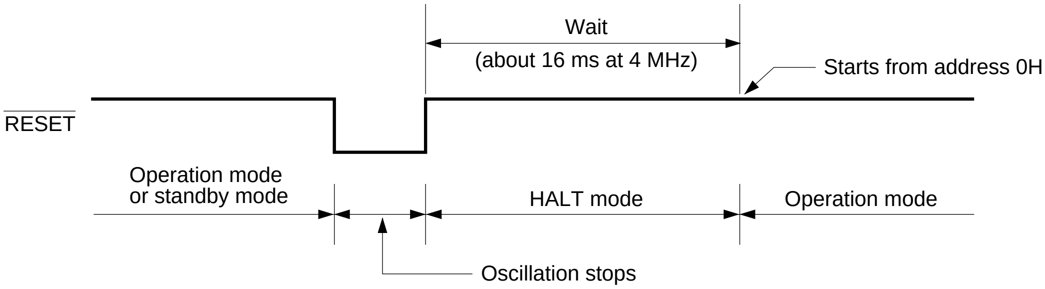
When the system is reset, the oscillator circuit remains in the HALT mode and then enters an operation mode, like when the STOP mode has been released. The wait time, after the reset signal has been removed, is 16.384 ms (f_X = 4 MHz).

On power application, input the reset signal at least once because the internal circuitry operations are not stable. When μPD17225 is reset, the following initialization takes place:

(1) Program counter is reset to 0.
(2) Flags in the register file are initialized to their default values (for the default values, refer to **Figure 11-1 Register Files**).
(3) The default value (0320H) is written to the data buffer (DBF).
(4) The hardware peripherals are initialized.
(5) The system clock (f_X) stops oscillation.

When the $\overline{\text{RESET}}$ pin is made high, the system clock starts oscillating, and the program execution starts from address 0 about 16 ms (at 4 MHz) later.

Figure 9-1. Reset Operation by $\overline{\text{RESET}}$ Input



9.2 Reset by Watchdog Timer (Connect $\overline{\text{RESET}}$ and $\overline{\text{WDOUT}}$ pins)

When the watchdog timer operates during program execution, a low level is output to the $\overline{\text{WDOUT}}$ pin, and the program counter is reset to 0.

If the watchdog timer is not reset for a fixed period of time, the program can be restarted from address 0H.

Program so that the watchdog timer is reset at intervals of within 340 ms (at f_X = 4 MHz) (set the WDTRES flag).

9.3 Reset by Stack Pointer (Connect $\overline{\text{RESET}}$ and $\overline{\text{WDOUT}}$ pins)

When the value of the stack pointer reaches 6H or 7H during program execution, a low level is output to the $\overline{\text{WDOUT}}$ pin, and the program counter is reset to 0.

If the nesting level of the interrupt or subroutine call exceeds 5 (stack over flow), or if the return instruction is executed without correspondence between CALL and return (RET) instructions established, then regardless of a stack level of 0 (stack underflow), the program can be restarted from address 0H.

Table 9-1. Status of Each Hardware After Reset

Hardware		$\overline{\text{RESET}}$ Input During Standby Mode	$\overline{\text{RESET}}$ Input During Operation
Program Counter (PC)		0000H	0000H
Port	Input/output	Input	Input
	Output latch	0	0
Data Memory (RAM)	General-purpose data memory (Except DBF, port register)	Retains previous status	Undefined
	DBF	0320H	0320H
	System register (SYSREG)	0	0
	WR	Retains previous status	Undefined
Control Register		Refer to **Figure 11-1 Register Files**	
8-bit Timer	Counter (TMC)	00H	00H
	Modulo register (TMM)	FFH	FFH
Remote Controller Carrier Generator	NRZ high-level timer modulo register (NRZHTMM)	Retains previous status	Undefined
	NRZ low-level timer modulo register (NRZLTMM)		
Basic Interval Timer/Watchdog Timer Counter		00H	00H

10. LOW-VOLTAGE DETECTOR CIRCUIT (CONNECT $\overline{\text{RESET}}$ AND $\overline{\text{WDOUT}}$ PINS)

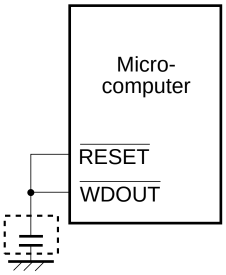
The low-voltage detector circuit outputs a low level from the $\overline{\text{WDOUT}}$ pin for initialization (reset) to prevent program hang-up that may take place when the batteries are replaced, if the circuit detects a low voltage.

A drop in the supply voltage is detected if the status of V_{DD} = 1.7 to 2.0 V lasts for 1 ms or longer. Note, however, that 1 ms is the guaranteed value and that the microcontroller may be reset even if the above low-voltage condition lasts for less than 1 ms.

Although the voltage at which the the reset function is effected ranges from 1.7 to 2.0 V, the program counter is prevented from hang-up even if the supply voltage drops until the reset function is effected, if the instruction execution time is from 4 to 32 μs. Note that some oscillators stop oscillating before the reset function is effected.

The low-voltage detector circuit can be set arbitrarily by the mask option.

Caution Connect a capacitor to the $\overline{\text{RESET}}$ pin as shown below to stabilize the operation.



Remark In this figure, the $\overline{\text{RESET}}$ pin is connected to a pull-up resistor by the mask option.

11. ASSEMBLER RESERVED WORDS

11.1 Mask Option Directives

When developing the μPD17225 program, mask options must be specified by using mask option directives in the program.

The $\overline{\text{RESET}}$ pin for μPD17225 requires a mask option to be specified.

11.1.1 OPTION and ENDOP directives

That portion of the program enclosed by the OPTION and ENDOP directives is called a mask option definition block. This block is described in the following format:

Description format:

Symbol	Mnemonic	Operand	Comment
[Label:]	OPTION		[;Comment]
	⋮		
	ENDOP		

11.1.2 Mask option definition directives

Table 11-1 lists the directives that can be used in the mask option definition block.

Here is an example of mask option definition:

Description example:

Symbol	Mnemonic	Operand	Comment
	OPTION		
	OPTRES	PULLUP	; RESET pin has pull-up resistors.
	OPTPOC	USEPOC	; Internal low-voltage detector circuit
	ENDOP		

Table 11-1. Mask Option Definition Directives

Name	Directive	Operands	1st Operand	2nd Operand	3rd Operand	4th Operand
RESET	OPTRES	1	Mask option of RESET			
			PULLUP (w/pull-up resistor) OPEN (w/o pull-up resistor)			
POC	OPTPOC	1	USEPOC (low-voltage detector circuit provided)			
			NOUSEPOC (low-voltage detector circuit not provided)			

11.2 Reserved Symbols

The symbols defined by the μPD17225 device file are listed in Table 11-2.

The defined symbols are the following register file names, port names, and peripheral hardware names.

11.2.1 Register file

The names of the symbols assigned to the register file are defined. These registers are accessed by the PEEK and POKE instructions through the window register (WR). Figure 11-1 shows the register file.

11.2.2 Registers and ports on data memory

The names of the registers assigned at addresses 00H through 7FH on the data memory and the names of ports assigned to address 70H and those that follow, and system register names are defined. Figure 11-2 shows the data memory configuration.

11.2.3 Peripheral hardware

The names of peripheral hardware accessed by the GET and PUT instructions are defined. Table 11-3 shows the peripheral hardware.

Table 11-2. Reserved Symbols (1/2)

Symbol Name	Attribute	Value	R/W	Description
DBF3	MEM	0.0CH	R/W	Bits 15-12 of data buffer
DBF2	MEM	0.0DH	R/W	Bits 11-8 of data buffer
DBF1	MEM	0.0EH	R/W	Bits 7-4 of data buffer
DBF0	MEM	0.0FH	R/W	Bits 3-0 of data buffer
AR3	MEM	0.74H	**Note**	Bits 15-12 of address register
AR2	MEM	0.75H	R/W	Bits 11-8 of address register
AR1	MEM	0.76H	R/W	Bits 7-4 of address register
AR0	MEM	0.77H	R/W	Bits 3-0 of address register
WR	MEM	0.78H	R/W	Window register
BANK	MEM	0.79H	**Note**	Bank register
IXH	MEM	0.7AH	**Note**	Index register, high
MPH	MEM	0.7AH	**Note**	Data memory row address pointer, high
MPE	FLG	0.7AH.3	R/W	Memory pointer enable flag
IXM	MEM	0.7BH	R/W	Index register, middle
MPL	MEM	0.7BH	R/W	Data memory row address pointer, low
IXL	MEM	0.7CH	R/W	Index register, low
RPH	MEM	0.7DH	**Note**	General register pointer, high
RPL	MEM	0.7EH	R/W	General register pointer, low
PSW	MEM	0.7FH	R/W	Program status word
BCD	FLG	0.7EH.0	R/W	BCD flag
CMP	FLG	0.7FH.3	R/W	Compare flag
CY	FLG	0.7FH.2	R/W	Carry flag
Z	FLG	0.7FH.1	R/W	Zero flag
IXE	FLG	0.7FH.0	R/W	Index enable flag
P0A0	FLG	0.70H.0	R/W	Bit 0 of port 0A
P0A1	FLG	0.70H.1	R/W	Bit 1 of port 0A
P0A2	FLG	0.70H.2	R/W	Bit 2 of port 0A
P0A3	FLG	0.70H.3	R/W	Bit 3 of port 0A
P0B0	FLG	0.71H.0	R/W	Bit 0 of port 0B
P0B1	FLG	0.71H.1	R/W	Bit 1 of port 0B
P0B2	FLG	0.71H.2	R/W	Bit 2 of port 0B
P0B3	FLG	0.71H.3	R/W	Bit 3 of port 0B
P0C0	FLG	0.72H.0	R/W	Bit 0 of port 0C
P0C1	FLG	0.72H.1	R/W	Bit 1 of port 0C
P0C2	FLG	0.72H.2	R/W	Bit 2 of port 0C
P0C3	FLG	0.72H.3	R/W	Bit 3 of port 0C
P0D0	FLG	0.73H.0	R/W	Bit 0 of port 0D
P0D1	FLG	0.73H.1	R/W	Bit 1 of port 0D
P0D2	FLG	0.73H.2	R/W	Bit 2 of port 0D
P0D3	FLG	0.73H.3	R/W	Bit 3 of port 0D

Note R: μPD17225, 17226 R/W: μPD17227, 17228

Table 11-2. Reserved Symbols (2/2)

Symbol Name	Attribute	Value	R/W	Description
P0E0	FLG	0.6FH.0	R/W	Bit 0 of port 0E
P0E1	FLG	0.6FH.1	R/W	Bit 1 of port 0E
P0E2	FLG	0.6FH.2	R/W	Bit 2 of port 0E
P0E3	FLG	0.6FH.3	R/W	Bit 3 of port 0E
SP	MEM	0.81H	R/W	Stack pointer
SYSCK	FLG	0.82H.0	R/W	System clock select flag
WDTRES	FLG	0.83H.3	R/W	Watchdog timer reset flag
BTMCK	FLG	0.83H.2	R/W	Basic interval timer mode select flag
BTMRES	FLG	0.83H.1	R/W	Basic interval timer mode reset flag
INT	FLG	0.8FH.0	R	INT pin status flag
NRZBF	FLG	0.91H.0	R/W	NRZ buffer data flag
NRZ	FLG	0.92H.0	R/W	NRZ data flag
P0EBPU0	FLG	0.97H.0	R/W	$P0E_0$ pull-up setting flag
P0EBPU1	FLG	0.97H.1	R/W	$P0E_1$ pull-up setting flag
P0EBPU2	FLG	0.97H.2	R/W	$P0E_2$ pull-up setting flag
P0EBPU3	FLG	0.97H.3	R/W	$P0E_3$ pull-up setting flag
IEG	FLG	0.9FH.0	R/W	INT pin interrupt edge flag
P0EBIO0	FLG	0.0A7H.0	R/W	$P0E_0$ I/O setting flag
P0EBIO1	FLG	0.0A7H.1	R/W	$P0E_1$ I/O setting flag
P0EBIO2	FLG	0.0A7H.2	R/W	$P0E_2$ I/O setting flag
P0EBIO3	FLG	0.0A7H.3	R/W	$P0E_3$ I/O setting flag
IPBTM	FLG	0.0AFH.2	R/W	Basic interval timer interrupt enable flag
IP	FLG	0.0AFH.1	R/W	INT pin interrupt enable flag
IPTM	FLG	0.0AFH.0	R/W	Timer interrupt enable flag
TMEN	FLG	0.0B3H.3	R/W	Timer enable flag
TMRES	FLG	0.0B3H.2	R/W	Timer reset flag
TMCK1	FLG	0.0B3H.1	R/W	Timer clock flag
TMCK0	FLG	0.0B3H.0	R/W	Timer clock flag
IRQBTM	FLG	0.0BDH.0	R/W	Basic interval timer interrupt request flag
IRQ	FLG	0.0BEH.0	R/W	INT pin interrupt request flag
IRQTM	FLG	0.0BFH.0	R/W	Timer interrupt request flag
TMC	DAT	05H	R	Timer count register
TMM	DAT	06H	W	Timer modulo register
NRZLTMM	DAT	03H	R/W	NRZ low-level timer modulo register
NRZHTMM	DAT	04H	R/W	NRZ high-level timer modulo register
AR	DAT	40H	R/W	Address register
DBF	DAT	0FH	—	Fixed operand value for PUT, GET, MOVT instruction
IX	DAT	01H	—	Fixed operand value for INC instruction

[MEMO]

Figure 11-1. Register Files (1/2)

Row Address \ Column Address		0	Note	1	Note	2	Note	3	Note	4	Note	5	Note	6	Note	7	Note
0	Bit 3			SP	0	0	0	WDTRES	0								
	Bit 2				1	0	0	BTMCK	0								
	Bit 1				0	0	0	BTMRES	0								
	Bit 0				1	SYSCK	0	0	0								
1	Bit 3			0	0	0	0									P0EBPU3	0
	Bit 2			0	0	0	0									P0EBPU2	0
	Bit 1			0	0	0	0									P0EBPU1	0
	Bit 0			NRZBF	0	NRZ	0									P0EBPU0	0
2	Bit 3															P0EBIO3	0
	Bit 2															P0EBIO2	0
	Bit 1															P0EBIO1	0
	Bit 0															P0EBIO0	0
3	Bit 3							TMEN	1								
	Bit 2							TMRES	0								
	Bit 1							TMCK1	0								
	Bit 0							TMCK0	0								

Note On reset

Figure 11-2. Data Memory Configuration

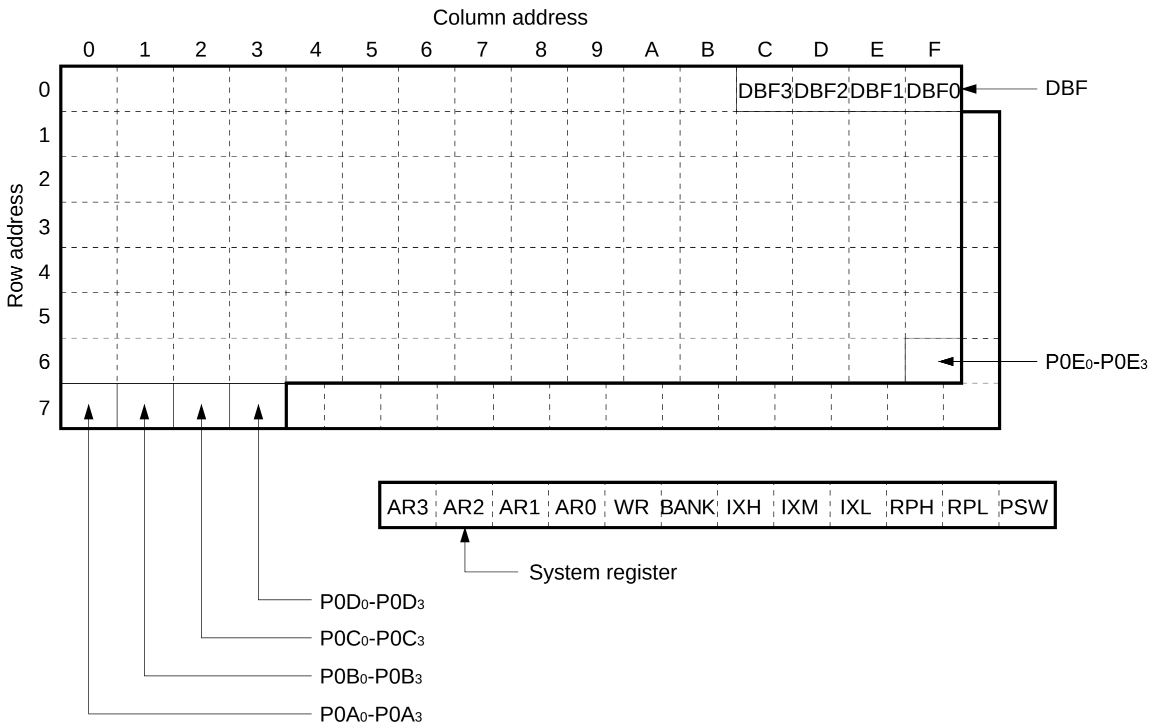


Figure 11-1. Register Files (2/2)

Column Address / Row Address		8	Note	9	Note	A	Note	B	Note	C	Note	D	Note	E	Note	F	Note
0	Bit 3															0	0
	Bit 2															0	0
	Bit 1															0	0
	Bit 0															INT	P
1	Bit 3															0	0
	Bit 2															0	0
	Bit 1															0	0
	Bit 0															IEG	0
2	Bit 3															0	0
	Bit 2															IPBTM	0
	Bit 1															IP	0
	Bit 0															IPTM	0
3	Bit 3											0	0	0	0	0	0
	Bit 2											0	0	0	0	0	0
	Bit 1											0	0	0	0	0	0
	Bit 0											IRQBTM	0	IRQ	0	IRQTM	1

Note On reset

P: When INT pin is high level, 1 or when INT pin is low level, 0.

Table 11-3. Peripheral Hardware

Name	Address	Valid Bit	Description
TMC	05H	8	Timer count register
TMM	06H	8	Timer modulo register
NRZLTMM	03H	8	Low-level timer modulo register for NRZ
NRZHTMM	04H	8	High-level timer modulo register for NRZ
AR	40H	16	Address register

12. INSTRUCTION SET

12.1 Instruction Set Outline

b_{15} b_{14}-b_{11}		0		1	
BIN.	HEX.				
0 0 0 0	0	ADD	r, m	ADD	m, #n4
0 0 0 1	1	SUB	r, m	SUB	m, #n4
0 0 1 0	2	ADDC	r, m	ADDC	m, #n4
0 0 1 1	3	SUBC	r, m	SUBC	m, #n4
0 1 0 0	4	AND	r, m	AND	m, #n4
0 1 0 1	5	XOR	r, m	XOR	m, #n4
0 1 1 0	6	OR	r, m	OR	m, #n4
0 1 1 1	7	INC	AR		
		INC	IX		
		MOVT	DBF, @AR		
		BR	@AR		
		CALL	@AR		
		RET			
		RETSK			
		EI			
		DI			
		RETI			
		PUSH	AR		
		POP	AR		
		GET	DBF, p		
		PUT	p, DBF		
		PEEK	WR, rf		
		POKE	rf, WR		
		RORC	r		
		STOP	s		
		HALT	h		
		NOP			
1 0 0 0	8	LD	r, m	ST	m, r
1 0 0 1	9	SKE	m, #n4	SKGE	m, #n4
1 0 1 0	A	MOV	@r, m	MOV	m, @r
1 0 1 1	B	SKNE	m, #n4	SKLT	m, #n4
1 1 0 0	C	BR	addr (Page 0)	CALL	addr
1 1 0 1	D	BR	addr (Page 1)	MOV	m, #n4
1 1 1 0	E	BR	addr (Page 2)	SKT	m, #n
1 1 1 1	F	BR	addr (Page 3)	SKF	m, #n

12.2 Legend

AR : Address register
ASR : Address stack register specified by stack pointer
addr : Program memory address (low-order 11 bits)
BANK : Bank register
CMP : Compare register
CY : Carry flag
DBF : Data buffer
h : Halt releasing condition
INTEF : Interrupt enable flag
INTR : Register automatically saved to stack in case of interrupt
INTSK : Interrupt stack register
IX : Index register
MP : Data memory row address pointer
  MPE : Memory pointer enable flag
m : Data memory address specified by m_R, m_C
  m_R : Data memory row address (high)
  m_C : Data memory column address (low)
n : Bit position (4 bits)
n4 : Immediate data (4 bits)
PAGE : Page (bit 11 and 12 of program counter)
PC : Program counter
p : Peripheral address
  p_H : Peripheral address (high-order 3 bits)
  p_L : Peripheral address (low-order 4 bits)
r : General register column address
rf : Register file address
  rf_R : Register file row address (high-order 3 bits)
  rf_C : Register file column address (low-order 4 bits)
SP : Stack pointer
s : Stop releasing condition
WR : Window register
(×) : Contents addressed by ×

12.3 List of Instruction Sets

Group	Mnemonic	Operand	Operation	Instruction Code			
				OP Code	Operand		
Addition	ADD	r, m	(r) ← (r) + (m)	00000	m_R	m_C	r
		m, #n4	(m) ← (m) + n4	10000	m_R	m_C	n4
	ADDC	r, m	(r) ← (r) + (m) + CY	00010	m_R	m_C	r
		m, #n4	(m) ← (m) + n4 + CY	10010	m_R	m_C	n4
	INC	AR	AR ← AR + 1	00111	000	1001	0000
		IX	IX ← IX + 1	00111	000	1000	0000
Subtraction	SUB	r, m	(r) ← (r) – (m)	00001	m_R	m_C	r
		m, #n4	(m) ← (m) – n4	10001	m_R	m_C	n4
	SUBC	r, m	(r) ← (r) – (m) – CY	00011	m_R	m_C	r
		m, #n4	(m) ← (m) – n4 – CY	10011	m_R	m_C	n4
Logical	OR	r, m	(r) ← (r) ∨ (m)	00110	m_R	m_C	r
		m, #n4	(m) ← (m) ∨ n4	10110	m_R	m_C	n4
	AND	r, m	(r) ← (r) ∧ (m)	00100	m_R	m_C	r
		m, #n4	(m) ← (m) ∧ n4	10100	m_R	m_C	n4
	XOR	r, m	(r) ← (r) ∀ (m)	00101	m_R	m_C	r
		m, #n4	(m) ← (m) ∀ n4	10101	m_R	m_C	n4
Judge	SKT	m, #n	CMP ← 0, if (m) ∧ n = n, then skip	11110	m_R	m_C	n
	SKF	m, #n	CMP ← 0, if (m) ∧ n = 0, then skip	11111	m_R	m_C	n
Compare	SKE	m, #n4	(m) – n4, skip if zero	01001	m_R	m_C	n4
	SKNE	m, #n4	(m) – n4, skip if not zero	01011	m_R	m_C	n4
	SKGE	m, #n4	(m) – n4, skip if not borrow	11001	m_R	m_C	n4
	SKLT	m, #n4	(m) – n4, skip if borrow	11011	m_R	m_C	n4
Rotate	RORC	r	$\begin{tikzcd} CY \arrow[r] & (r)_{b3} \arrow[r] & (r)_{b2} \arrow[r] & (r)_{b1} \arrow[r] & (r)_{b0} \arrow[llll, bend left] \end{tikzcd}$	00111	000	0111	r
Transfer	LD	r, m	(r) ← (m)	01000	m_R	m_C	r
	ST	m, r	(m) ← (r)	11000	m_R	m_C	r
	MOV	@r, m	if MPE = 1 : (MP, (r)) ← (m) if MPE = 0 : (BANK, m_R, (r)) ← (m)	01010	m_R	m_C	r
		m, @r	if MPE = 1 : (m) ← (MP, (r)) if MPE = 0 : (m) ← (BANK, m_R, (r))	11010	m_R	m_C	r
		m, #n4	(m) ← n4	11101	m_R	m_C	n4
	MOVT	DBF, @AR	SP ← SP – 1, ASR ← PC, PC ← AR DBF ← (PC), PC ← ASR, SP ← SP + 1	00111	000	0001	0000

Group	Mnemonic	Operand	Operation	Instruction Code			
				OP Code	Operand		
Transfer	PUSH	AR	SP ← SP – 1, ASR ← AR	00111	000	1101	0000
	POP	AR	AR ← ASR, SP ← SP + 1	00111	000	1100	0000
	PEEK	WR, rf	WR ← (rf)	00111	rf$_R$	0011	rf$_C$
	POKE	rf, WR	(rf) ← WR	00111	rf$_R$	0010	rf$_C$
	GET	DBF, p	(DBF) ← (p)	00111	p$_H$	1011	p$_L$
	PUT	p, DBF	(p) ← (DBF)	00111	p$_H$	1010	p$_L$
Branch	BR	addr	**Note**	01100	addr		
		@AR	PC ← AR	00111	000	0100	0000
Subroutine	CALL	addr	SP ← SP – 1, ASR ← PC, PC$_{10-0}$ ← addr, PAGE ← 0	11100	addr		
		@AR	SP ← SP – 1, ASR ← PC, PC ← AR	00111	000	0101	0000
	RET		PC ← ASR, SP ← SP + 1	00111	000	1110	0000
	RETSK		PC ← ASR, SP ← SP + 1 and skip	00111	001	1110	0000
	RETI		PC ← ASR, INTR ← INTSK, SP ← SP + 1	00111	100	1110	0000
Interrupt	EI		INTEF ← 1	00111	000	1111	0000
	DI		INTEF ← 0	00111	001	1111	0000
Other	STOP	s	STOP	00111	010	1111	s
	HALT	h	HALT	00111	011	1111	h
	NOP		No operation	00111	100	1111	0000

Note The operation and OP code of “BR addr” of the μPD17225, 17226, 17227, and 17228 are as follows, respectively.

(a) μPD17225

Operand	Operation	OP Code
addr	PC$_{10-0}$ ← addr	01100

(b) μPD17226

Operand	Operation	OP Code
addr	PC$_{10-0}$ ← addr, PAGE ← 0	01100
	PC$_{10-0}$ ← addr, PAGE ← 1	01101

(c) μPD17227

Operand	Operation	OP Code
addr	PC$_{10-0}$ ← addr, PAGE ← 0	01100
	PC$_{10-0}$ ← addr, PAGE ← 1	01101
	PC$_{10-0}$ ← addr, PAGE ← 2	01110

(d) μPD17228

Operand	Operation	OP Code
addr	$PC_{10-0} \leftarrow$ addr, PAGE $\leftarrow$ 0	01100
	$PC_{10-0} \leftarrow$ addr, PAGE $\leftarrow$ 1	01101
	$PC_{10-0} \leftarrow$ addr, PAGE $\leftarrow$ 2	01110
	$PC_{10-0} \leftarrow$ addr, PAGE $\leftarrow$ 3	01111

12.4 Assembler (RA17K) Built-In Macro Instruction

Legend

flag n : FLG type symbol

n : Bit number

< > : Contents in < > can be omitted

	Mnemonic	Operand	Operation	n
Built-in macro	SKTn	flag 1, ...flag n	if (flag 1) to (flag n) = all “1”, then skip	$1 \leq n \leq 4$
	SKFn	flag 1, ...flag n	if (flag 1) to (flag n) = all “0”, then skip	$1 \leq n \leq 4$
	SETn	flag 1, ...flag n	(flag 1) to (flag n) ← 1	$1 \leq n \leq 4$
	CLRn	flag 1, ...flag n	(flag 1) to (flag n) ← 0	$1 \leq n \leq 4$
	NOTn	flag 1, ...flag n	if (flag n) = “0”, then (flag n) ← 1 if (flag n) = “1”, then (flag n) ← 0	$1 \leq n \leq 4$
	INITFLG	<NOT> flag 1, ⋯<<NOT> flag n>	if description = NOT flag n, then (flag n) ← 0 if description = flag n, then (flag n) ← 1	$1 \leq n \leq 4$
	BANKn		(BANK) ← n	n = 0^{Note}
Expantion instruction	BRX	Label	Jump Label	—
	CALLX	function-name	CALL sub-routine	—
	INITFLGX	<NOT/INV> flag 1, ...<NOT/INV> flag n	if description = NOT (or INV) flag, (flag) ← 0 if description = flag, (flag) ← 1	$n \leq 4$

Note μPD17227, 17228: n = 0, 1

13. ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (T_A = 25°C)

Item	Symbol	Conditions		Ratings	Unit
Supply Voltage	V_{DD}			–0.3 to +3.8	V
Input Voltage	V_I			–0.3 to V_{DD} + 0.3	V
Output Voltage	V_O			–0.3 to V_{DD} + 0.3	V
High-Level Output Current[Note]	I_{OH}	REM pin	Peak value	–36.0	mA
			rms value	–24.0	mA
		1 pin (P0E pin)	Peak value	–7.5	mA
			rms value	–5.0	mA
		Total of P0E pins	Peak value	–22.5	mA
			rms value	–15.0	mA
Low-Level Output Current[Note]	I_{OL}	1 pin (P0C, P0D, P0E, REM or $\overline{\text{WDOUT}}$ pin)	Peak value	7.5	mA
			rms value	5.0	mA
		Total of P0C, P0D, $\overline{\text{WDOUT}}$ pins	Peak value	22.5	mA
			rms value	15.0	mA
		Total of P0E pins	Peak value	30.0	mA
			rms value	20.0	mA
Operating Temperature	T_A			–40 to +85	°C
Storage Temperature	T_{stg}			–65 to +150	°C
Power Dissipation	P_d	T_A = 85 °C		180	mW

Note Calculate rms value by this expression: [rms value] = [Peak value] $\times \sqrt{\text{Duty}}$

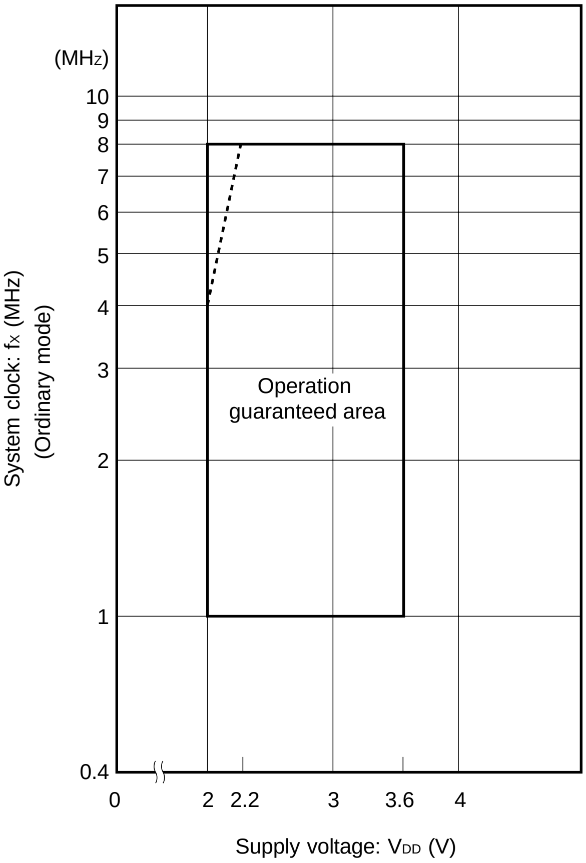
Caution Even if one of the parameters exceeds its absolute maximum rating even momentarily, the quality of the product may be degraded. The absolute maximum rating therefore specifies the upper or lower limit of the value at which the product can be used without physical damages. Be sure not to exceed or fall below this value when using the product.

Recommended Operating Ranges (T$_A$ = –40 to +85°C, V$_{DD}$ = 2.0 to 3.6 V)

Item	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Supply Voltage	V$_{DD1}$	f$_X$ = 1 MHz	High-speed mode (Instruction execution time: 16 μs)	2.0		3.6	V
	V$_{DD2}$	f$_X$ = 4 MHz	Ordinary mode (Instruction execution time: 4 μs)				
	V$_{DD3}$	f$_X$ = 8 MHz	High-speed mode (Instruction execution time: 4 μs)				
	V$_{DD4}$		High-speed mode (Instruction execution time: 2 μs)	2.2		3.6	V
Oscillation Frequency	f$_X$			1.0	4.0	8.0	MHz
Operating Temperature	T$_A$			–40	+25	+85	°C
Low-Voltage Detector Circuit[Note] (Mask Option)	T$_{CY}$			4		32	μs

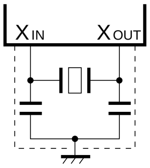
Note Reset if the status of V$_{DD}$ = 1.7 to 2.0 V lasts for 1 ms or longer. Program hang-up does not occur even if the voltage drops, until the reset function is effected (when the $\overline{\text{RESET}}$ pin and $\overline{\text{WDOUT}}$ pin are connected). Some oscillators stop oscillating before the reset function is effected.

f$_X$ vs V$_{DD}$



Remark The region indicated by the broken line in the above figure is the guaranteed operating range in the high-speed mode.

System Clock Oscillator Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 3.6 V)

Resonator	Recommended Constants	Item	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f_X)[Note 1]		1.0	4.0	8.0	MHz
		Oscillation stabilization time[Note 2]	After V_{DD} reached MIN. in oscillation voltage range			4	ms

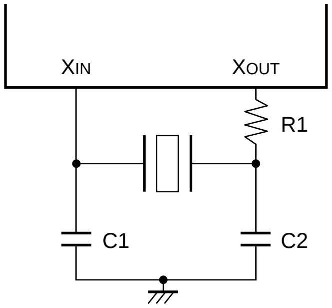
Notes 1. The oscillation frequency only indicates the oscillator characteristics.

2. The oscillation stabilization time is necessary for oscillation to be stabilized, after V_{DD} application or STOP mode release.

Caution **To use a system clock oscillator circuit, perform the wiring in the area enclosed by the dotted line in the above figure as follows, to avoid adverse wiring capacitance influences:**

- **Keep wiring length as short as possible.**
- **Do not cross a signal line with some other signal lines. Do not route the wiring in the vicinity of lines through which a large current flows.**
- **Always keep the oscillator circuit capacitor ground at the same potential as GND. Do not ground the capacitor to a ground pattern, through which a large current flows.**
- **Do not extract signals from the oscillator circuit.**

External circuit example



★ **Main System Clock: Ceramic Resonator (T_A = –40 to +85 °C)**

Manufacturer	Part Number	Recommended Circuit Constants		Oscillation Voltage Range		Remark
		C1 (pF)	C2 (pF)	MIN. (V)	MAX. (V)	
Murata Mfg. Co., Ltd.	CSA2.00MG	30	30	2.0	3.6	
	CSA3.00MG	30	30	2.0	3.6	
	CSA4.00MG	30	30	2.0	3.6	
	CSA6.00MG	30	30	2.0	3.6	
	CSA8.00MTZ	30	30	2.2	3.6	
TDK Corp.	CCR1000K2	100	100	2.0	3.6	
	CCR4.0MC3	—	—	2.0	3.6	Built-in capacitor
	CCR6.0MC3	—	—	2.0	3.6	Built-in capacitor
	CCR8.0MC5	—	—	2.0	3.6	Built-in capacitor
	FCR4.0MC5	—	—	2.0	3.6	Built-in capacitor
	FCR6.0MC5	—	—	2.0	3.6	Built-in capacitor
Matsushita Electronic Components Co., Ltd.	EF0EC2004A4	—	—	2.0	3.6	Built-in capacitor
	EF0EC3004A4	—	—	2.0	3.6	Built-in capacitor
	EF0EC4004A4	—	—	2.0	3.6	Built-in capacitor
	EF0EC6004A4	—	—	2.0	3.6	Built-in capacitor
	EF0EC8004A4	—	—	2.0	3.6	Built-in capacitor
Toko Ceramic Co., Ltd.	CRHF2.50	30	30	2.2	3.6	
	CRHF4.00	30	30	2.2	3.6	
	CRHF6.00	30	30	2.2	3.6	

Caution The oscillation circuit constants and oscillation voltage range indicate conditions for stable oscillation but do not guarantee accuracy of the oscillation frequency. If the application circuit requires accuracy of the oscillation frequency, it is necessary to set the oscillation frequency of the resonator in the application circuit. For this, it is necessary to directly contact the manufacturer of the resonator being used.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 3.6 V)

Item	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
High-Level Input Voltage	V_{IH1}	P0A, P0B			0.7V_{DD}		V_{DD}	V
	V_{IH2}	P0E, $\overline{\text{RESET}}$, INT			0.8V_{DD}		V_{DD}	V
Low-Level Input Voltage	V_{IL1}	P0A, P0B			0		0.3V_{DD}	V
	V_{IL2}	P0E			0		0.35V_{DD}	V
	V_{IL3}	$\overline{\text{RESET}}$, INT pin			0		0.2V_{DD}	V
High-Level Input Leakage Current	I_{LIH}	P0A, P0B, P0E, $\overline{\text{RESET}}$, INT	$V_I = V_{DD}$				3	μA
Low-Level Input Leakage Current	I_{LIL1}	INT	V_I = 0 V				–3	μA
	I_{LIL2}	P0E, $\overline{\text{RESET}}$	V_I = 0 V w/o pull-up resistor				–3	μA
High-Level Output Leakage Current	I_{LOH}	P0C, P0D, P0E, $\overline{\text{WDOUT}}$	$V_O = V_{DD}$				3	μA
Low-Level Output Leakage Current	I_{LOL}	P0E, $\overline{\text{WDOUT}}$	V_O = 0 V w/o pull-up resistor				–3	μA
Internal Pull-Up Resistor	R_{U1}	P0E, $\overline{\text{RESET}}$			25	50	100	kΩ
	R_{U2}	P0A, P0B			100	200	400	kΩ
High-Level Output Current	I_{OH1}	REM	V_{OH} = 1.0 V, V_{DD} = 3 V		–6	–13	–24	mA
High-Level Output Voltage	V_{OH}	P0E, REM	I_{OH} = –0.5 mA		V_{DD}–0.3		V_{DD}	V
Low-Level Output Voltage	V_{OL1}	P0C, P0D, REM, $\overline{\text{WDOUT}}$	I_{OL} = 0.5 mA				0.3	V
	V_{OL2}	P0E	I_{OL} = 1.5 mA				0.3	V
Low-Voltage Detector Circuit (Mask Option)	V_{DT}	$\overline{\text{WDOUT}}$ = low level	$V_{DT} = V_{DD}$		1.7	1.85	2.0	V
Data Retention Voltage	V_{DDDR}	$\overline{\text{RESET}}$ = low level or STOP mode			1.3			V
Supply Current	I_{DD1}	Operating mode (high-speed)	V_{DD} = 3 V ± 10 %	f_X = 1 MHz		0.6	1.2	mA
				f_X = 4 MHz		0.75	1.3	mA
				f_X = 8 MHz		0.9	1.8	mA
	I_{DD2}	Operating mode (low-speed)	V_{DD} = 3 V ± 10 %	f_X = 1 MHz		0.475	0.95	mA
				f_X = 4 MHz		0.6	1.1	mA
				f_X = 8 MHz		0.8	1.6	mA
	I_{DD3}	HALT mode	V_{DD} = 3 V ± 10 %	f_X = 1 MHz		0.4	0.8	mA
				f_X = 4 MHz		0.45	0.85	mA
				f_X = 8 MHz		0.5	1.0	mA
	I_{DD4}	STOP mode	V_{DD} = 3 V ± 10 %			2.0	20.0	μA
			built-in POC	T_A = 25 °C		2.0	5.0	μA

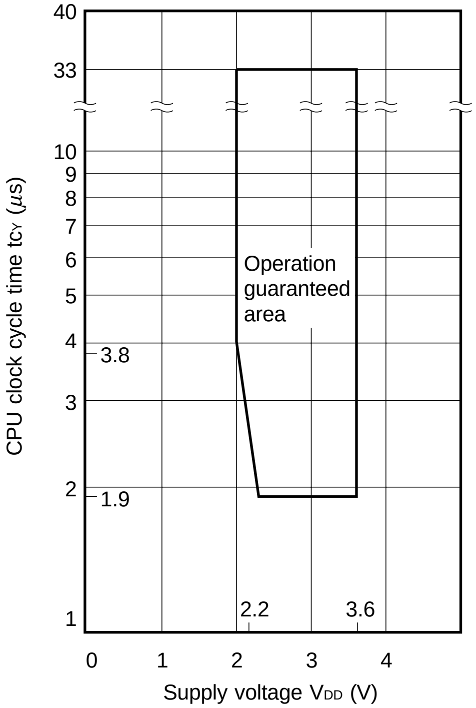
AC Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 3.6 V)

Item	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
CPU Clock Cycle Time[Note] (Instruction Execution Time)	t_{CY1}		3.8		33	μs
	t_{CY2}	V_{DD} = 2.2 to 3.6 V	1.9		33	μs
INT High/Low Level Width	t_{INTH}, t_{INTL}		20			μs
$\overline{\text{RESET}}$ Low Level Width	t_{RSL}		10			μs

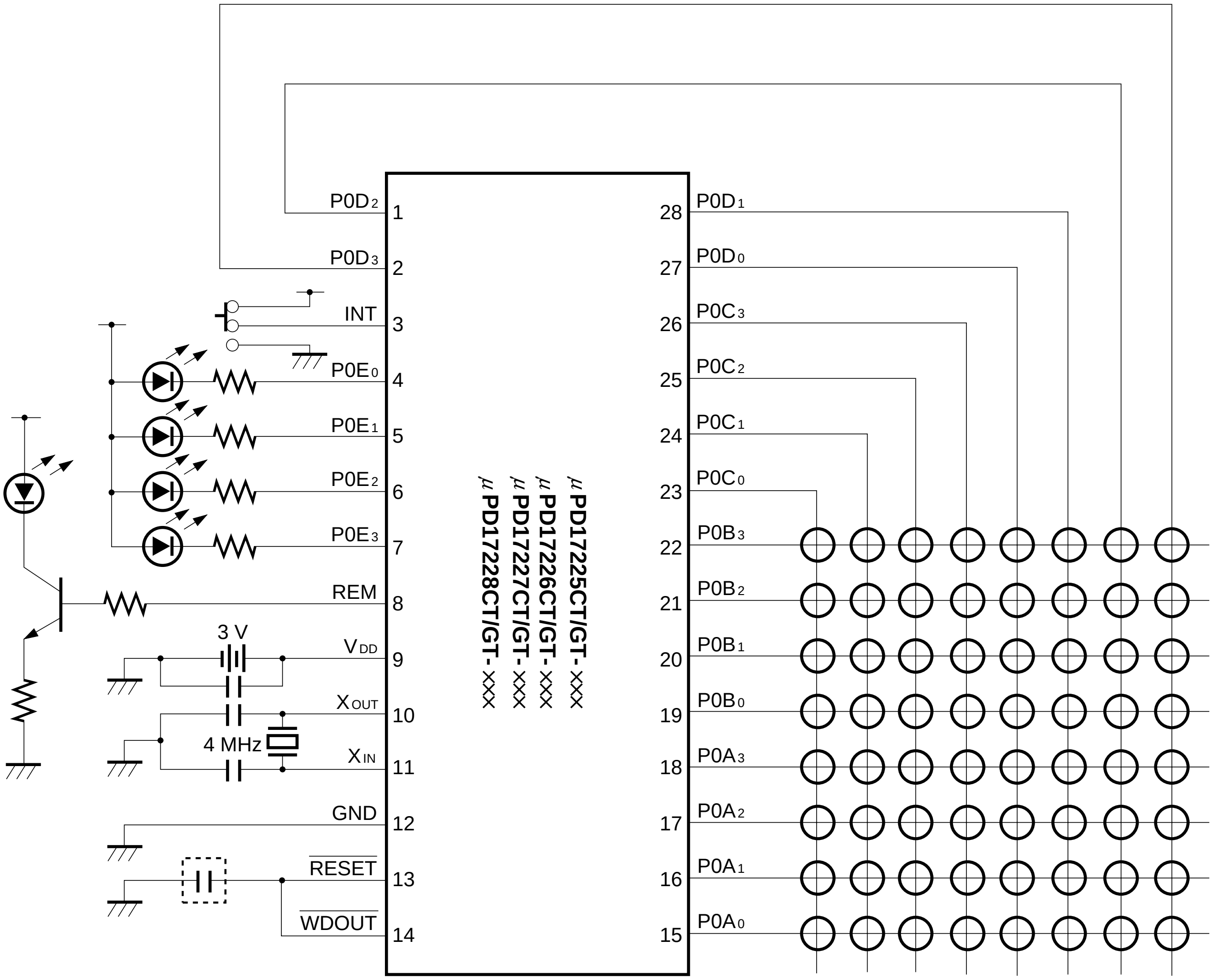
Note The CPU clock cycle time (instruction execution time) is determined by the oscillation frequency of the resonator connected and SYSCK (RF: address 02H) of the register file.

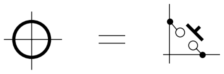
The figure on the right shows the CPU clock cycle time t_{CY} vs. supply voltage V_{DD} characteristics (refer to **4. CLOCK GENERATOR CIRCUIT**).

t_{CY} vs V_{DD}



14. APPLICATION CIRCUIT EXAMPLE

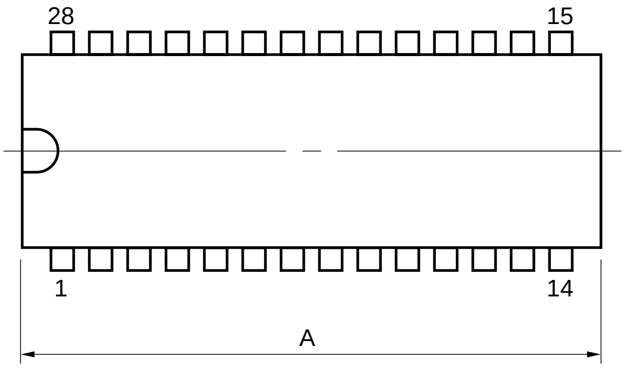


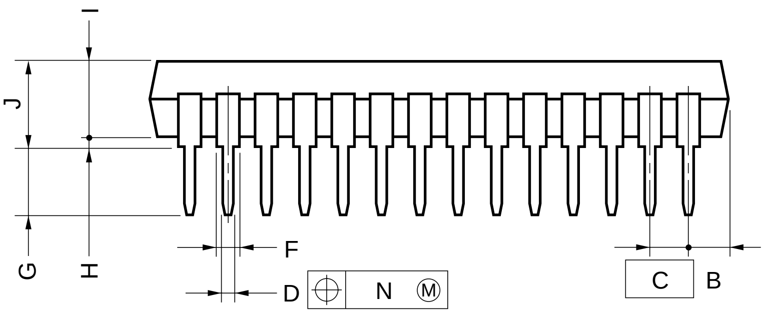


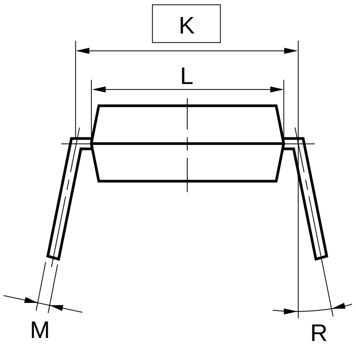
Remark The RESET pin can be connected to a pull-up resistor by the mask option.

15. PACKAGE DRAWINGS

28 PIN PLASTIC SHRINK DIP (400 mil)







NOTES

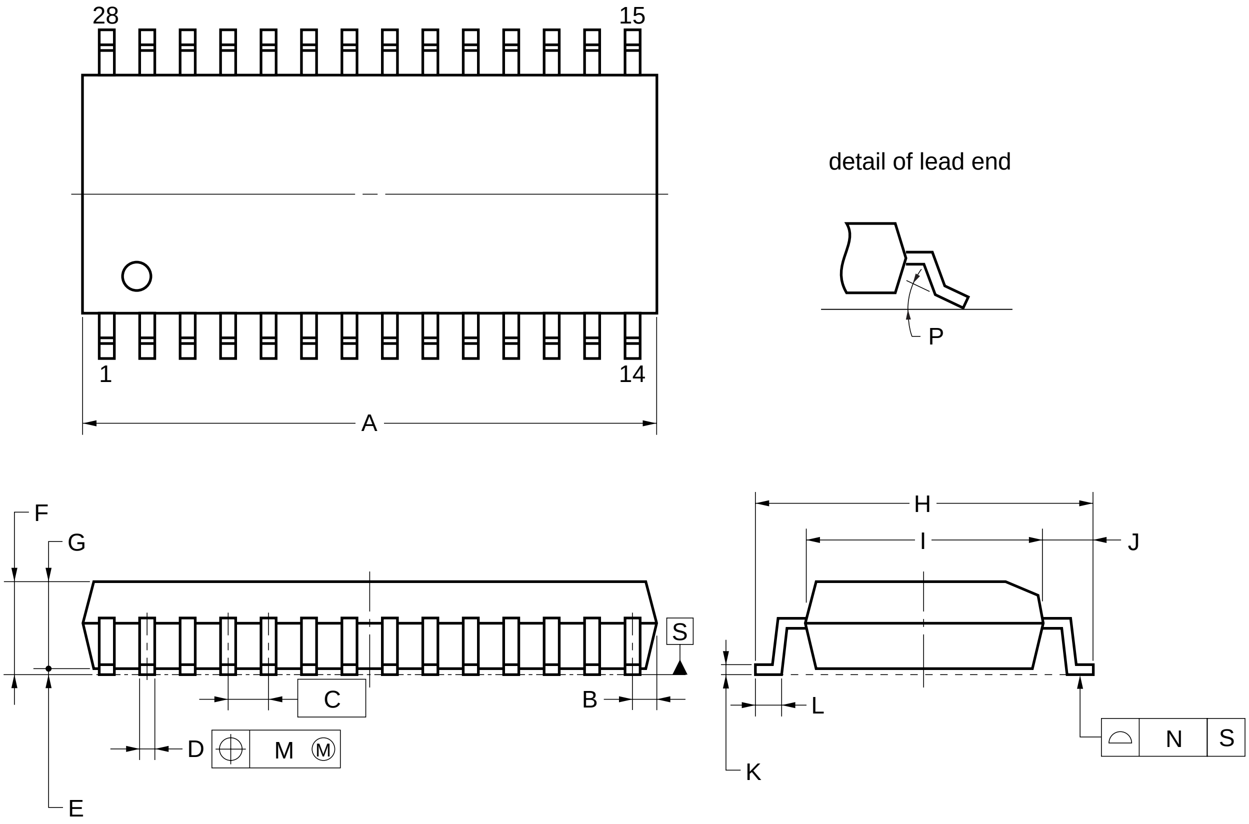
1) Each lead centerline is located within 0.17 mm (0.007 inch) of its true position (T.P.) at maximum material condition.

2) Item "K" to center of leads when formed parallel.

ITEM	MILLIMETERS	INCHES
A	28.46 MAX.	1.121 MAX.
B	2.67 MAX.	0.106 MAX.
C	1.778 (T.P.)	0.070 (T.P.)
D	0.50±0.10	$0.020^{+0.004}_{-0.005}$
F	0.85 MIN.	0.033 MIN.
G	3.2±0.3	0.126±0.012
H	0.51 MIN.	0.020 MIN.
I	4.31 MAX.	0.170 MAX.
J	5.08 MAX.	0.200 MAX.
K	10.16 (T.P.)	0.400 (T.P.)
L	8.6	0.339
M	$0.25^{+0.10}_{-0.05}$	$0.010^{+0.004}_{-0.003}$
N	0.17	0.007
R	0~15°	0~15°

S28C-70-400B-1

28-PIN PLASTIC SOP (375 mil)



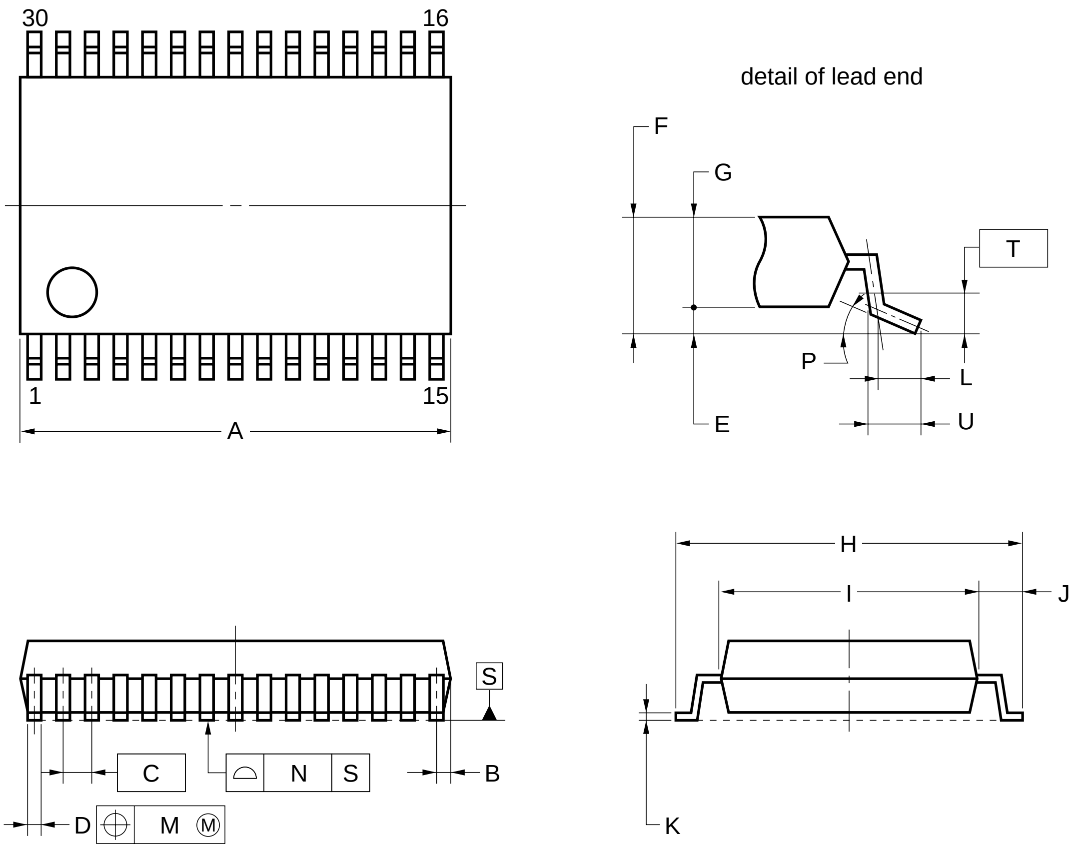
NOTE

Each lead centerline is located within 0.12 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	17.9±0.17
B	0.78 MAX.
C	1.27 (T.P.)
D	$0.42^{+0.08}_{-0.07}$
E	0.1±0.1
F	2.6±0.2
G	2.50
H	10.3±0.3
I	7.2±0.2
J	1.6±0.2
K	$0.17^{+0.08}_{-0.07}$
L	0.8±0.2
M	0.12
N	0.15
P	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$

P28GM-50-375B-4

★ 30 PIN PLASTIC SSOP (300 mil)



NOTE

Each lead centerline is located within 0.10 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	9.85±0.15
B	0.45 MAX.
C	0.65 (T.P.)
D	$0.24^{+0.08}_{-0.07}$
E	0.1±0.05
F	1.3±0.1
G	1.2
H	8.1±0.2
I	6.1±0.2
J	1.0±0.2
K	0.17±0.03
L	0.5
M	0.13
N	0.10
P	$3°^{+5°}_{-3°}$
T	0.25
U	0.6±0.15

S30MC-65-5A4-1

16. RECOMMENDED SOLDERING CONDITIONS

For the μPD17225 soldering must be performed under the following conditions.

For details of recommended conditions for surface mounting, refer to information document **"Semiconductor Device Mounting Technology Manual" (C10535E)**.

For other soldering methods, please consult with NEC personnel.

Table 16-1. Soldering Conditions of Surface Mount Type

(1) μPD17225GT-×××: 28-pin plastic SOP (375 mil)
μPD17226GT-×××: 28-pin plastic SOP (375 mil)
μPD17227GT-×××: 28-pin plastic SOP (375 mil)
μPD17228GT-×××: 28-pin plastic SOP (375 mil)

Soldering Method	Soldering Conditions	Symbol
Infrated Reflow	Package peak temperature: 235 °C, Time: 30 seconds max. (210 °C min.), Number of times: 2 max.	IR35-00-2
VPS	Package peak temperature: 215 °C, Time: 40 seconds max. (200 °C min.), Number of times: 2 max.	VP15-00-2
★ Wave Soldering	Solder bath temperature: 260 °C max, Time: 10 seconds max., Number of times: once, preheating temperature: 120 °C max. (package surface temperature)	WS66-00-1
Partial Heating	Pin temperature: 300 °C max., Time: 3 seconds max. (per side of device)	—

Caution Do not use two or more soldering methods in combination (except the partial heating method).

★ **(2) μPD17225MC-×××-5A4: 30-pin plastic shrink SOP (300 mil)**
μPD17226MC-×××-5A4: 30-pin plastic shrink SOP (300 mil)
μPD17227MC-×××-5A4: 30-pin plastic shrink SOP (300 mil)
μPD17228MC-×××-5A4: 30-pin plastic shrink SOP (300 mil)

Soldering Method	Soldering Conditions	Symbol
Infrated Reflow	Package peak temperature: 235 °C, Time: 30 seconds max. (210 °C min.), Number of times: 3 max.	IR35-00-3
VPS	Package peak temperature: 215 °C, Time: 40 seconds max. (200 °C min.), Number of times: 3 max.	VP15-00-3
Wave Soldering	Solder bath temperature: 260 °C max, Time: 10 seconds max., Number of times: once, preheating temperature: 120 °C max. (package surface temperature)	WS66-00-1
Partial Heating	Pin temperature: 300 °C max., Time: 3 seconds max. (per side of device)	—

Caution Do not use two or more soldering methods in combination (except the partial heating method).

Table 16-2. Soldering Conditions of Through-Hole Type

μPD17225CT-×××: 28-pin plastic shrink DIP (400 mil)
μPD17226CT-×××: 28-pin plastic shrink DIP (400 mil)
μPD17227CT-×××: 28-pin plastic shrink DIP (400 mil)
μPD17228CT-×××: 28-pin plastic shrink DIP (400 mil)

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)

Caution The wave solding must be performed at the lead part only. Note that the solder must not be directly contacted to the package body.

APPENDIX A. DIFFERENCES AMONG μPD17225, 17226, 17227, 17228 AND μPD17P218

μPD17P218 is equipped with PROM to which data can be written by the user instead of the internal mask ROM (program memory) of the μPD17228.

Table A-1 shows the differences between the μPD17225, 17226, 17227, 17228 and μPD17P218.

The differences among these five models are the program memory and mask option, and their CPU functions and internal hardware are identical. Therefore, the μPD17P218 can be used to evaluate the program developed for the μPD17225, 17226, 17227, and 17228 system. **Note, however, that some of the electrical specifications such as supply current and low-voltage detection voltage of the μPD17P218 are different from those of the μPD17225, 17226, 17227, and 17228.**

Table A-1. Differences among μPD17225, 17226, 17227, 17228 and μPD17P218

Product Name Item	μPD17P218	μPD17225	μPD17226	μPD17227	μPD17228
Program Memory	One-time PROM	Mask ROM			
	16 K bytes (8192 × 16) (0000H-1FFFH)	4 K bytes (2048 × 16) (0000H-07FFH)	8 K bytes (4096 × 16) (0000H-0FFFH)	12 K bytes (6144 × 16) (0000H-17FFH)	16 K bytes (8192 × 16) (0000H-1FFFH)
Data Memory	223 × 4 bits	111 × 4 bits		223 × 4 bits	
Pull-Up Resistor of RESET Pin	Provided	Any (mask option)			
Low-Voltage Detector Circuit^{Note}	Provided	Any (mask option)			
V_{PP} Pin, Operation Mode Select Pin	Provided	Not provided			
Handling of $\overline{\text{WDOUT}}$ Pin When Not Used	Connect to GND	Connect to V_{DD} via resistor			
Instruction Execution Time (T_{CY})	2 μs (V_{DD} = 3.5 to 5.5 V) 4 μs (V_{DD} = 2.2 to 5.5 V) 8 μs (V_{DD} = 2.0 to 5.5 V)	2 μs (V_{DD} = 2.2 to 3.6 V) 4 μs (V_{DD} = 2.0 to 3.6 V)			
Operation When P0C, P0D Are Standby	Retain output level immediately before standby mode				
Supply Voltage	V_{DD} = 2.0 to 5.5 V	V_{DD} = 2.0 to 3.6 V			
Package	28-pin plastic SOP (375 mil) 28-pin plastic shrink DIP (400 mil) ★ 30-pin plastic shrink SOP (300 mil)				

Note Although the circuit configuration is identical, its electrical characteristics differ depending on the product.

[MEMO]

APPENDIX B. FUNCTIONAL COMPARISON OF μPD17225 SUBSERIES RELATED PRODUCTS

<table>
<tr><th colspan="2">Product Name
Item</th><th>μPD17201A</th><th>μPD17207</th><th>μPD17202A</th><th>μPD17215</th><th>μPD17216</th><th>μPD17217</th><th>μPD17218</th></tr>
<tr><td colspan="2">ROM Capacity (Bit)</td><td>3072 × 16</td><td>4096 × 16</td><td colspan="2">2048 × 16</td><td>4096 × 16</td><td>6144 × 16</td><td>8192 × 16</td></tr>
<tr><td colspan="2">RAM Capacity (Bit)</td><td colspan="2">336 × 4</td><td>112 × 4</td><td colspan="2">111 × 4</td><td colspan="2">223 × 4</td></tr>
<tr><td colspan="2">LCD Controller/Driver</td><td colspan="2">136 segments max.</td><td>96 segments max.</td><td colspan="4">Not provided</td></tr>
<tr><td colspan="2">Infrared Remote Controller Carrier Generator (REM)</td><td colspan="2">LED output is high-active</td><td>LED output is low-active</td><td colspan="4">Provided (without LED output)</td></tr>
<tr><td colspan="2">I/O Ports</td><td colspan="2">19 lines</td><td>16 lines</td><td colspan="4">20 lines</td></tr>
<tr><td colspan="2">External Interrupt (INT)</td><td colspan="2">1 line
(rising-edge detection)</td><td colspan="5">1 line (rising-edge, falling-edge detection)</td></tr>
<tr><td colspan="2">Analog Input</td><td colspan="2">4 channels (8-bit A/D)</td><td colspan="5">Not provided</td></tr>
<tr><td colspan="2">Timer</td><td colspan="3">2 channels { 8-bit timer
Watch timer</td><td colspan="4">2 channels { 8-bit timer
Basic interval timer</td></tr>
<tr><td colspan="2">Watchdog Timer</td><td colspan="7">Provided ($\overline{\text{WDOUT}}$ output)</td></tr>
<tr><td colspan="2">Low-Voltage Detector Circuit^{Note}</td><td colspan="3">Not provided</td><td colspan="4">Provided ($\overline{\text{WDOUT}}$ output)</td></tr>
<tr><td colspan="2">Serial Interface</td><td colspan="2">1 channel</td><td colspan="5">Not provided</td></tr>
<tr><td colspan="2">Stack</td><td colspan="7">5 levels (3 levels for multiplexed interrupt)</td></tr>
<tr><td rowspan="2">Instruction Execution Time</td><td>Main System Clock</td><td colspan="3">4 μs (4 MHz: with ceramic or crystal resonator, V_{DD} = 2.2 to 5.5 V)</td><td colspan="4">• 2 μs (8 MHz ceramic resonator: in high-speed mode, V_{DD} = 3.5 to 5.5 V)
• 4 μs (4 MHz ceramic resonator: in high-speed mode, V_{DD} = 2.2 to 5.5 V)
• 8 μs (2 MHz ceramic resonator: in high-speed mode, V_{DD} = 2.0 to 5.5 V)</td></tr>
<tr><td>Subsystem Clock</td><td colspan="3">488 μs (32.768 kHz: with crystal resonator, V_{DD} = 2.0 to 5.5 V)</td><td colspan="4">Not provided</td></tr>
<tr><td colspan="2">Supply Voltage
(With Subsystem Clock)</td><td colspan="3">V_{DD} = 2.2 to 5.5 V (V_{DD} = 2.0 to 5.5 V)</td><td colspan="4">V_{DD} = 2.0 to 5.5 V</td></tr>
<tr><td colspan="2">Standby Function</td><td colspan="7">STOP, HALT</td></tr>
<tr><td colspan="2">Package</td><td colspan="2">80-pin plastic QFP</td><td>64-pin plastic QFP</td><td colspan="4">28-pin plastic SOP
28-pin plastic shrink DIP</td></tr>
<tr><td colspan="2">One-Time PROM Products</td><td colspan="2">μPD17P207</td><td>μPD17P202A</td><td colspan="4">μPD17P218</td></tr>
</table>

Note Note that although all the products have the same circuit construction, the electrical specifications differ dependant on each product.

μPD17225	μPD17226	μPD17227	μPD17228
2048 × 16	4096 × 16	6144 × 16	8192 × 16
111 × 4		223 × 4	
Not provided			
Provided (without LED output)			
20 pins			
1 pin (rising edge, falling edge detection)			
Not provided			
2 channels 8-bit timer Basic interval timer			
Provided ($\overline{\text{WDOUT}}$ output)			
Provided ($\overline{\text{WDOUT}}$ output)			
Not provided			
5 levels (3 nesting levels)			
• 2 μs (8-MHz ceramic resonator: in high-speed mode, V_{DD} = 2.2 to 3.6 V) • 4 μs (4-MHz ceramic resonator: in high-speed mode, V_{DD} = 2.0 to 3.6 V)			
Not provided			
V_{DD} = 2.0 to 3.6 V			
STOP, HALT			
28-pin plastic SOP 28-pin plastic shrink DIP ★ 30-pin plastic shrink SOP			
μPD17P218			

★ APPENDIX C. DEVELOPMENT TOOLS

To develop the programs for the μPD17225 subseries, the following development tools are available:

Hardware

Name	Remarks
In-Circuit Emulator (IE-17K, IE-17K-ET[Note 1])	IE-17K and IE-17K-ET are the in-circuit emulators used in common with the 17K series microcontroller. IE-17K and IE-17K-ET are connected to a PC-9800 series or IBM PC/AT™ compatible machines as the host machine with RS-232C. By using these in-circuit emulators with a system evaluation board corresponding to the microcomputer, the emulators can emulate the microcomputer. A higher level debugging environment can be provided by using man-machine interface *SIMPLEHOST*™.
SE Board (SE-17225)	This is an SE board for μPD17225 subseries. It can be used alone to evaluate a system or in combination with an in-circuit emulator for debugging.
Emulation Probe (EP-17K28CT)	EP-17K28CT is an emulation probe for 17K series 28-pin shrink DIP (400mil).
Emulation Probe (EP-17K28GT)	EP-17K28GT is an emulation probe for 17K series 28-pin SOP (375 mil). When used with EV-9500GT-28[Note 2], it connects an SE board to the target system.
Emulation Probe (EP-17K30GS)	EP-17K30GS is an emulation probe for 17K series 30-pin shrink SOP (300 mil) (under development).
Conversion Adapter (EV-9500GT-28[Note 2])	EV-9500GT-28 is a conversion adapter for 28-pin SOP (375 mil) and is used to connect EP-17K28GT to the target system.
PROM Programmer (AF-9703[Note 3], AF-9704[Note 3], AF-9705[Note 3], AF-9706[Note 3])	AF-9703, AF-9704, AF-9705, and AF-9706 are PROM programmers corresponding to μPD17P218. By connecting program adapter AF-9808J or AF-9808H to this PROM programmer, μPD17P218 can be programmed.
Program Adapter (AF-9808J[Note 3], AF-9808H[Note 3])	AF-9808J and AF-9808H are adapters that is used to program μPD17P218CT and μPD17P218GT respectively, and is used in combination with AF-9703, AF-9704, AF-9705, or AF-9706.

Notes 1. Low-cost model: External power supply type

2. Two EV-9500GT-28s are supplied with the EP-17K28GT. Five EV-9500GT-28s are optionally available as a set.

3. These are products from Ando Electric Co., Ltd. For details, consult Ando Electric Co., Ltd. (Tel: 03-3733-1163).

Software

Name	Outline	Host Machine	OS	Supply	Order Code
17K Assembler (RA17K)	The RA17K is an assembler common to the 17K series products. When developing the program of devices, RA17K is used in combination with a device file (AS17225).	PC-9800 series	Japanese Windows™	3.5" 2HD	μSAA13RA17K
		IBM PC/AT compatible machine	Japanese Windows	3.5" 2HC	μSAB13RA17K
			English Windows		μSBB13RA17K
17K Series C-like Compiler (*emIC-17K*™)	The *emIC-17K* is a C-like compiler common to the 17K series. Used in combination with the RA17K.	PC-9800 series	Japanese Windows	3.5” 2HD	μSAA13CC17K
		IBM PC/AT compatible machine	Japanese Windows	3.5” 2HC	μSAB13CC17K
			English Windows		μSBB13CC17K
Device File (AS17225)	The AS17225 is a device file for μPD17225, 17226, 17227, and 17228 respectively, and are used in combination with an assembler for the 17K series (RA17K).	PC-9800 series	Japanese Windows	3.5" 2HD	μSAA13AS17225
		IBM PC/AT compatible machine	Japanese Windows	3.5" 2HC	μSAB13AS17225
			English Windows		μSBB13AS17225
Support Software (*SIMPLEHOST*)	*SIMPLEHOST* is a software package that enables man-machine interface on the Windows when a program is developed by using an in-circuit emulator and a personal computer.	PC-9800 series	Japanese Windows	3.5" 2HD	μSAA13ID17K
		IBM PC/AT compatible machine	Japanese Windows	3.5" 2HC	μSAB13ID17K
			English Windows		μSBB13ID17K

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.

Regional Information

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

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

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

NEC Electronics Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics (Germany) GmbH
Duesseldorf, Germany
Tel: 0211-65 03 02
Fax: 0211-65 03 490

NEC Electronics (UK) Ltd.
Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Italiana s.r.l.
Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

NEC Electronics (Germany) GmbH
Benelux Office
Eindhoven, The Netherlands
Tel: 040-2445845
Fax: 040-2444580

NEC Electronics (France) S.A.
Velizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

NEC Electronics (France) S.A.
Spain Office
Madrid, Spain
Tel: 91-504-2787
Fax: 91-504-2860

NEC Electronics (Germany) GmbH
Scandinavia Office
Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Singapore Pte. Ltd.
United Square, Singapore 1130
Tel: 65-253-8311
Fax: 65-250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-2719-2377
Fax: 02-2719-5951

NEC do Brasil S.A.
Electron Devices Division
Rodovia Presidente Dutra, Km 214
07210-902-Guarulhos-SP Brasil
Tel: 55-11-6465-6810
Fax: 55-11-6465-6829

J99.1

***emIC-17K* and *SIMPLEHOST* are trademarks of NEC Corporation.**
Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.
PC/AT is a trademark of IBM Corporation.

The export of this product from Japan is prohibited without governmental license. To export or re-export this product from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

- **The information in this document is subject to change without notice. Before using this document, please confirm that this is the latest version.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Corporation. NEC Corporation assumes no responsibility for any errors which may appear in this document.
- NEC Corporation does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from use of a device described herein or any other liability arising from use of such device. No license, either express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Corporation or others.
- Descriptions of circuits, software, and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software, and information in the design of the customer's equipment shall be done under the full responsibility of the customer. NEC Corporation assumes no responsibility for any losses incurred by the customer or third parties arising from the use of these circuits, software, and information.
- While NEC Corporation has been making continuous effort to enhance the reliability of its semiconductor devices, the possibility of defects cannot be eliminated entirely. To minimize risks of damage or injury to persons or property arising from a defect in an NEC semiconductor device, customers must incorporate sufficient safety measures in its design, such as redundancy, fire-containment, and anti-failure features.
- NEC devices are classified into the following three quality grades:
  "Standard", "Special", and "Specific". The Specific quality grade applies only to devices developed based on a customer designated “quality assurance program“ for a specific application. The recommended applications of a device depend on its quality grade, as indicated below. Customers must check the quality grade of each device before using it in a particular application.
  - Standard: Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots
  - Special: Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support)
  - Specific: Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems or medical equipment for life support, etc.

  The quality grade of NEC devices is "Standard" unless otherwise specified in NEC's Data Sheets or Data Books. If customers intend to use NEC devices for applications other than those specified for Standard quality grade, they should contact an NEC sales representative in advance.

M7 98.8