NEC

MOS INTEGRATED CIRCUIT
μPD17120

SMALL GENERAL-PURPOSE
4 BIT SINGLE-CHIP MICROCONTROLLER

The μPD17120 is a 4-bit single-chip microcontroller containing timer, a power-on/power-down reset circuit, and a serial interface.

For the CPU, the μPD17120 employs a 17K architecture using general registers. The new architecture allows operations to be performed directly on data memory, without involving accumulators as conventionally done. In addition, each instruction is 16 bits (one word) long, allowing programming to be done efficiently.

The μPD17P132, a one-time PROM product, is available for evaluation of the μPD17120 and for small-scale production.

The following user's manual completely describes the functions of the μPD17120. Be sure to read it before designing an application system.

μPD17120 Sub-Series User's Manual: IEU-1367

FEATURES

- 17K architecture: General registers, 16-bit instructions
- Program memory (ROM): 1.5K bytes (768 × 16 bits)
- Data memory (RAM): 64 × 4 bits
- Instruction execution time: 8 μs (when the microcontroller operates at 2 MHz with RC oscillation[Note])
- External interrupt: 1 line (INT pin, with sensor input)
- Timer function: 1 channel
- 3-wire serial interface: 1 channel
- Input/output pins: 19 pins (including one sensor input pin)
- Power-on/power-down reset function
- Supply voltage: V_{DD} = 2.7 to 5.5 V

Note The capacitor for RC oscillation is contained in the μPD17120.

APPLICATIONS

Controlling electric appliances such as electric fans

Document No. IC-2972A
(O. D. No. IC-8407C)
Date Published June 1995 P
Printed in Japan

Major changes in this revision are indicated by stars (★) in the margins.

ORDERING INFORMATION

Part number	Package
μPD17120CS-×××	24-pin plastic shrink DIP (300 mil)
μPD17120GT-×××	24-pin plastic SOP (375 mil)

Remark ××× indicates the ROM code.

CHARACTERISTICS

Item	Description
ROM capacity	1.5K bytes (768 × 16 bits)
RAM capacity	64 × 4 bits (The stack is separated from memory.)
Stack	5 address stacks, 1 interrupt stack
Number of input/output ports	19 { • 18 input/output ports • 1 input port for sensing an interrupt (INT pin[Note])
	1 channel (8-bit timer)
Timer	1 channel (3-wire type)
Serial interface Interrupt	• 1 external interrupt (INT) { • Detection of the rising edge, falling edge, or both edges can be selected. • 2 internal interrupts { • Timer (TM) • Serial interface (SIO)
Execution time of an instruction	8 μs (when operating at 2 MHz with RC oscillation)
Standby function	STOP, HALT
Power-on/power-down reset circuit	Built-in (Can be used in an application circuit where V_{DD} is 5 V ±10 %)
Operating power voltage	• V_{DD} = 2.7 to 5.5 V • V_{DD} = 4.5 to 5.5 V (when the power-on/power-down reset functions are used)
Package	• 24-pin plastic shrink DIP (300 mil) • 24-pin plastic SOP (375 mil)
★ One-time PROM product	μPD17P132

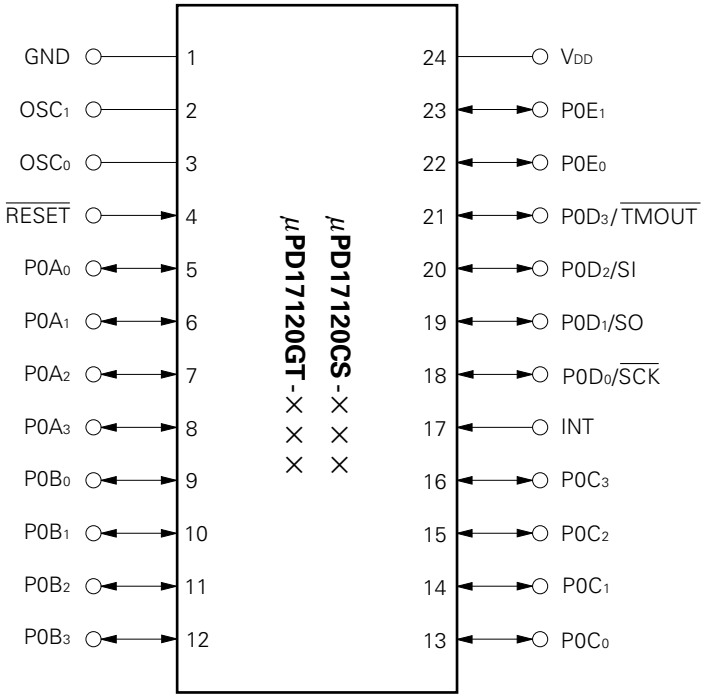
★ **Note** The INT pin can be used as an input pin (sensor input) when the external interrupt function is not used. The status of the pin is read with the INT flag of the control register, not with the port register.

★ **Caution Although a PROM product is highly compatible with a masked ROM product in respect of functions, they differ in internal ROM circuits and part of electrical characteristics. Before changing the PROM product to the masked ROM product in an application system, evaluate the system carefully using the masked ROM product.**

CONTENTS

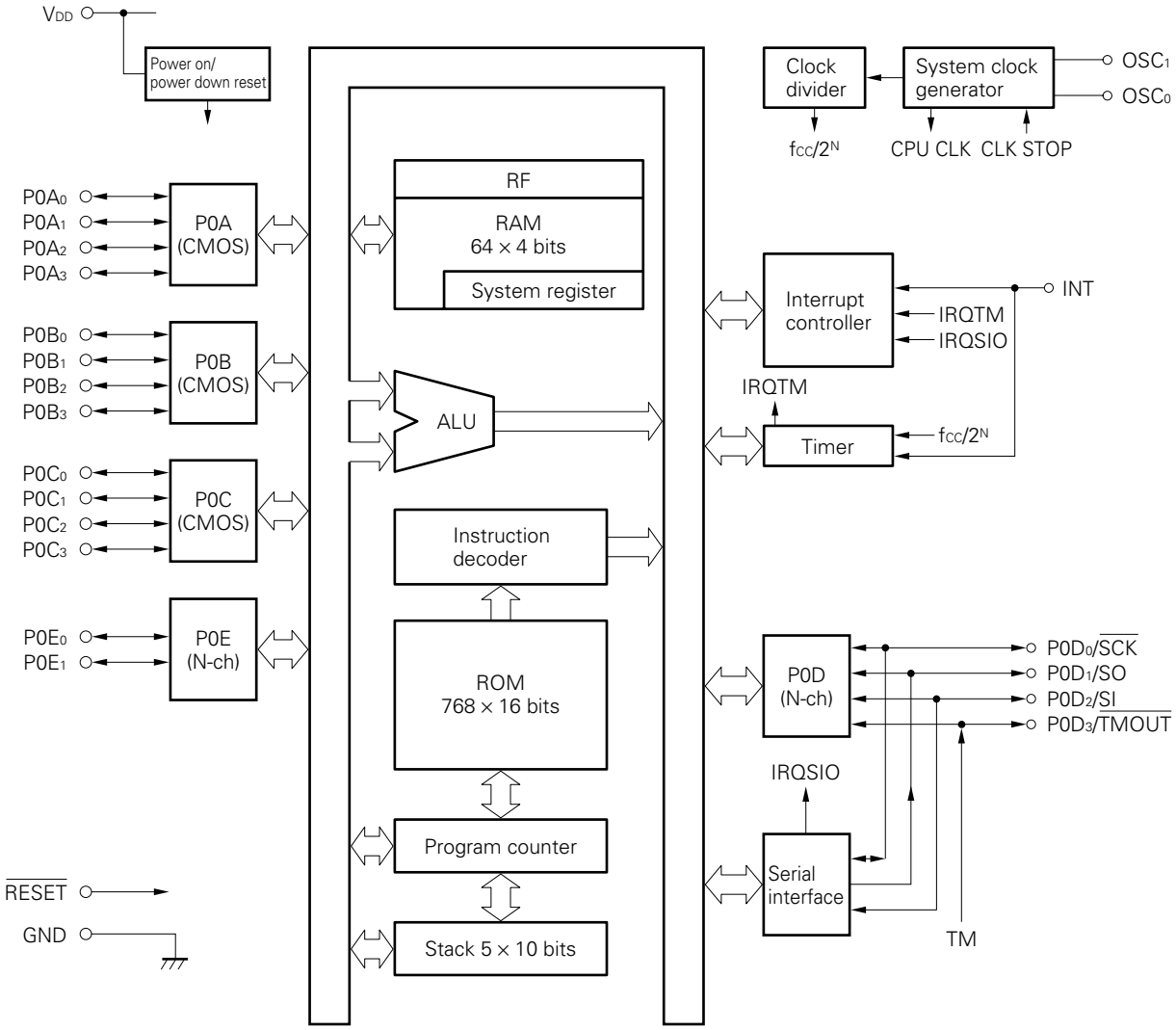
1. PIN CONFIGURATION (TOP VIEW)

24-pin plastic shrink DIP
24-pin plastic SOP



$\overline{\text{RESET}}$:	Reset input	$P0A_0$-$P0A_3$:	Port 0A
$\overline{\text{TMOUT}}$:	Timer output	$P0B_0$-$P0B_3$:	Port 0B
INT:	External interrupt input	$P0C_0$-$P0C_3$:	Port 0C
SI:	Serial data input	$P0D_0$-$P0D_3$:	Port 0D
SO:	Serial data output	$P0E_0$,$P0E_1$:	Port 0E
$\overline{\text{SCK}}$:	Serial clock input/output	V_{DD}:	Power supply
OSC_0,OSC_1:	System clock oscillation	GND:	Ground

2. BLOCK DIAGRAM



Remark The terms CMOS and N-ch in parentheses indicate the output form of the port.

CMOS: CMOS push-pull output

N-ch: N-channel open-drain output (Each pin can contain pull-up resistor as specified using a mask option.)

3. PINS

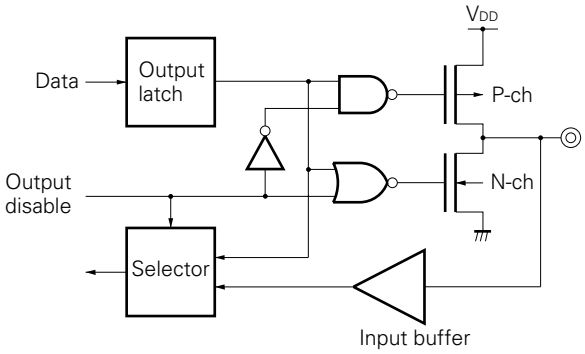
3.1 PIN FUNCTIONS

Pin No.	Pin name	Function	Output	After reset
1	GND	Ground	–	–
2 3	OSC_1 OSC_0	For system clock oscillation Resistor is connected from OSC_0 to OSC_1.	–	–
4	$\overline{RESET}$	Reset input pin • Pull-up resistor incorporation specifiable by mask option	–	Input
5 - 8	$P0A_0$ - $P0A_3$	Port 0A • 4-bit input/output port • Input/output setting allowed in units of 1 bit	CMOS push-pull	Input
9 - 12	$P0B_0$ - $P0B_3$	Port 0B • 4-bit input/output port • Input/output setting allowed in units of 4 bits	CMOS push-pull	Input
13 - 16	$P0C_0$ - $P0C_3$	Port 0C • 4-bit input/output port • Input/output setting allowed in units of 1 bit	CMOS push-pull	Input
17	INT	External interrupt request or sensor signal	–	Input
18	$P0D_0/\overline{SCK}$	Pin for port 0D, timer output, serial data input, serial data output, and serial clock input/output • $P0D_0$ - $P0D_3$ • 4-bit input/output port • Input/output setting allowed in units of 1 bit • Pull-up resistor incorporation specifiable by mask option in units of 1 bit • $\overline{SCK}$ • Serial clock input/output	N-ch open drain	Input (P0D)
19	$P0D_1$/SO	• SO • Serial data output		
20	$P0D_2$/SI	• SI • Serial data input		
21	$P0D_3/\overline{TMOUT}$	• $\overline{TMOUT}$ • Timer output		
22 23	$P0E_0$ $P0E_1$	Port 0E • 2-bit input/output port • Input/output setting allowed in units of 1 bit • Pull-up resistor incorporation specifiable by mask option in units of 1 bit	N-ch open drain	Input
24	V_{DD}	Power supply	–	–

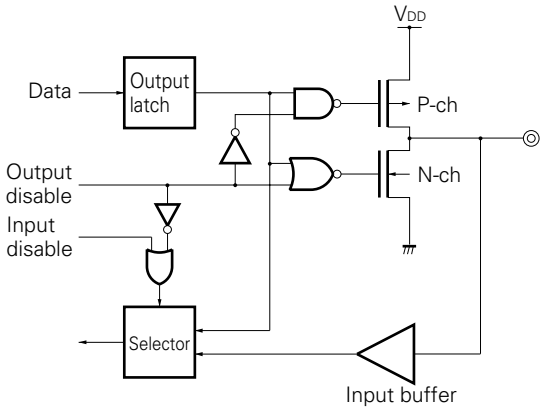
3.2 PIN EQUIVALENT CIRCUIT

Below are simplified diagrams of the input/output circuits for each pin.

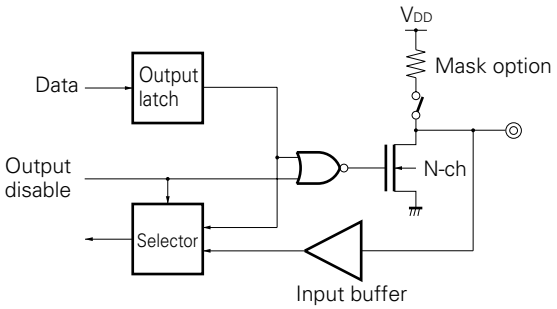
(1) P0A0 - P0A3, P0B0 - P0B3



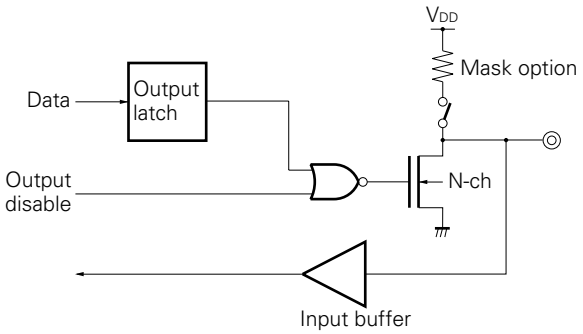
(2) P0C0 - P0C3



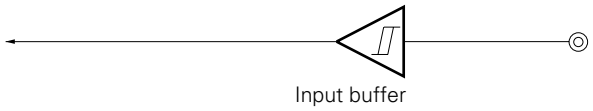
(3) P0D0 - P0D3



(4) P0E0, P0E1

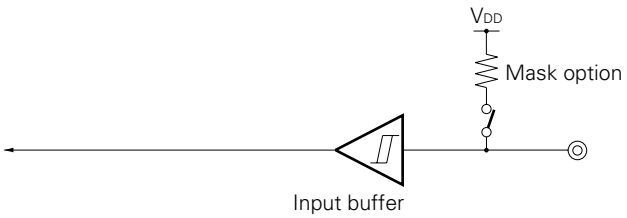


(5) INT



Schmit trigger input with hysteresis characteristics

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



Schmit trigger input with hysteresis characteristics

★ 3.3 HANDLING UNUSED PINS

Connect unused pins as follows:

Table 3-1 Handling Unused Pins

Pin			Recommended conditions and handling	
			Internal	External
Port	Input mode	P0A, P0B, P0C	—	Connect to V_{DD} or ground through resistors for each pin.Note 1
		P0D, P0E	Pull-up resistors that can be specified with the mask option are not incorporated.	
			Pull-up resistors that can be specified with the mask option are incorporated.	Leave open.
	Output mode	P0A, P0B, P0C (CMOS ports)	—	Leave open.
		P0D, P0E (N-ch open-drain port)	Outputs low level without pull-up resistors that can be specified with the mask option.	
			Outputs low level with pull-up resistors that can be specified with the mask option.	
External interrupt (INT)Note 2			—	Connect directly to ground.
$\overline{\text{RESET}}$Note 3 (when only the built-in power-on/power-down reset function is used)			Pull-up resistors that can be specified with the mask option are not incorporated.	Connect directly to V_{DD}.
			Pull-up resistors that can be specified with the mask option are incorporated.	

Notes 1. When a pin is pulled up to V_{DD} (connected to V_{DD} through a resistor) or pulled down to ground (connected to ground through a resistor) outside the chip, take the driving capacity and maximum current consumption of a port into consideration. When using high-resistance pull-up or pull-down resistors, apply appropriate countermeasures to ensure that noise is not attracted by the resistors. Although the optimum pull-up or pull-down resistor varies with the application circuit, in general, a resistor of 10 to 100 kilohms is suitable.

2. Since the INT pin is also used for setting the test mode, connect it directly to ground when the pin is not used.

3. When designing an application circuit which requires high reliability, be sure to design a circuit to which an external $\overline{\text{RESET}}$ signal can be input. Since the $\overline{\text{RESET}}$ pin is also used for setting the test mode, connect it to V_{DD} directly when not used.

Caution To fix the I/O mode and output level of a pin, it is recommended that they should be specified repeatedly within a loop in a program.

3.4 NOTES ON USE OF THE $\overline{\text{RESET}}$ AND INT PINS

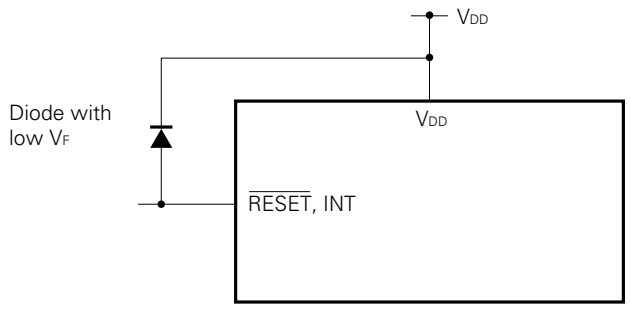
The $\overline{\text{RESET}}$ and INT pins have the test mode selecting function for testing the internal operation of the μPD17120 (IC test), besides the functions shown in **Section 3.1**.

Applying a voltage exceeding V_{DD} to the $\overline{\text{RESET}}$ or INT pin causes the μPD17120 to enter the test mode. When noise exceeding V_{DD} comes in during normal operation, the device is switched to the test mode.

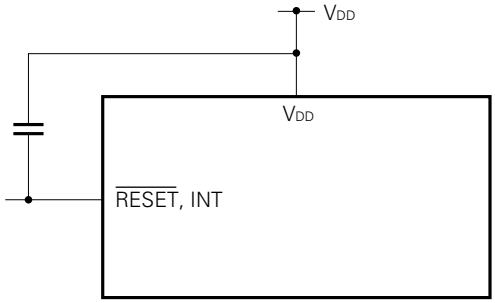
For example, if the wiring from the $\overline{\text{RESET}}$ or INT pin is too long, noise may be induced on the wiring, causing this mode switching.

When installing the wiring, lay the wiring in such a way that noise is suppressed as much as possible. If noise yet arises, use an external part to suppress it as shown below.

- **Connect a diode with low V_F between the pin and V_{DD}.**



- **Connect a capacitor between the pin and V_{DD}.**



4. PROGRAM MEMORY (ROM)

The μPD17120 is loaded with a 1.5K-byte (768 × 16 bit) mask ROM as program memory.

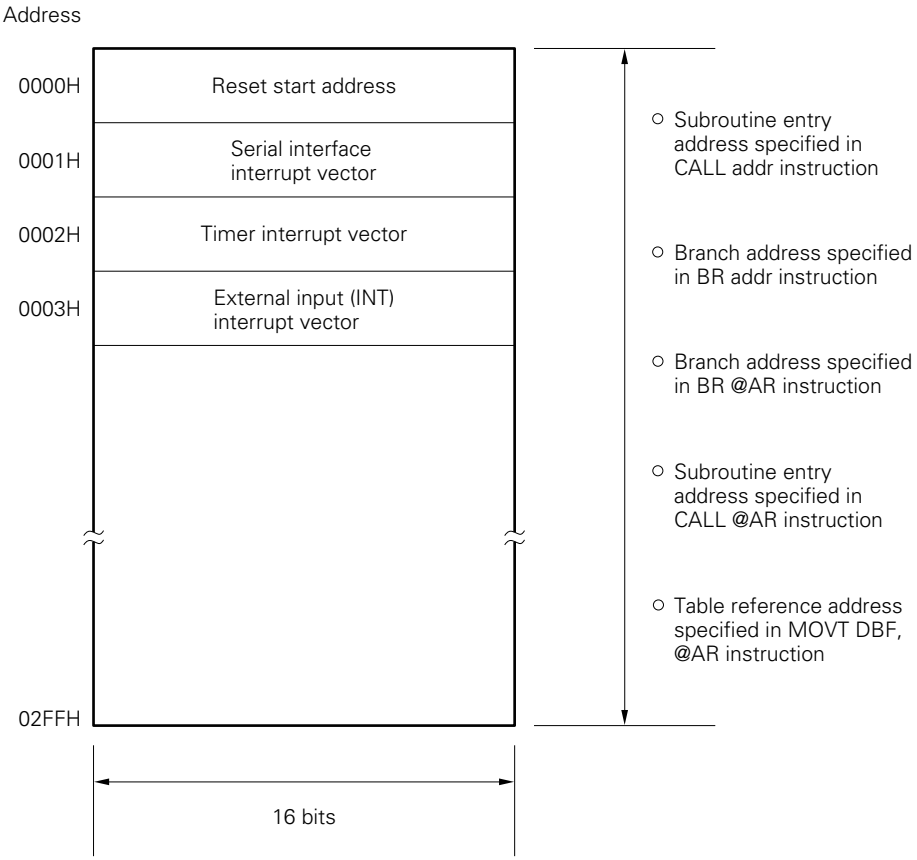
The program memory address is specified by the program counter.

★ Program memory stores the program and the constant data table. The reset start address and interrupt vector addresses are assigned to 0000H to 0003H in program memory.

4.1 PROGRAM MEMORY ORGANIZATION

Fig. 4-1 shows a program memory map. Branch instructions, subroutine calls, and table references can specify any address in program memory.

Fig. 4-1 Program Memory Map for μPD17120



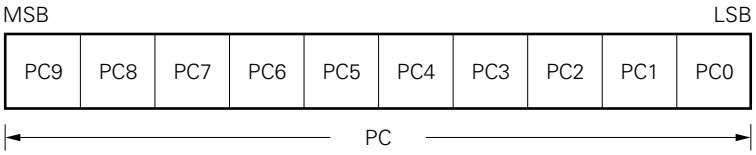
5. PROGRAM COUNTER (PC)

The program counter is used to specify an address in program memory.

5.1 PROGRAM COUNTER CONFIGURATION

As shown in Fig. 5-1, the program counter is a 10-bit binary counter.

Fig. 5-1 Program Counter



5.2 PROGRAM COUNTER OPERATION

Normally, the program counter is automatically incremented each time a command is executed. The memory address at which the next instruction to be executed is stored is assigned to the program counter under the following conditions: At reset; when a branch, subroutine call, return, or table referencing instruction is executed; or when an interrupt is received.

Table 5-1 Value of the Program Counter after an Instruction Is Executed

Instruction \ Program counter bit	Program counter value									
	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
During reset	0	0	0	0	0	0	0	0	0	0
BR addr CALL addr	Value set by addr									
BR @AR CALL @AR MOVT DBF, @AR	Value in the address register (AR)									
RET RETSK RETI	Value in the address stack location pointed to by the stack pointer (return address)									
During interrupt	Vector address for the interrupt									

5.3 NOTES ON USING THE PROGRAM COUNTER

The program counter (PC) of the μPD17120 is a 10-bit counter which can specify a program as large as 1024 steps. However, the ROM can contain only 768 steps (at addresses 0000H to 02FFH). If the program counter specifies address 300H or higher, the counter reads the contents of address FFFFH, and enters into a status equivalent to when instruction "SKF PSW, #0FH" is executed.

Therefore, the following guidelines should be observed.

(1) When a program consists of up to 768 steps (up to address 02FFH), use a branch instruction (BR) or a return instruction (RET) at the end of the program. If neither a branch nor a return instruction is used, the program counter will not automatically be set to address 0000H, but will instead specify an address for which no ROM exists.

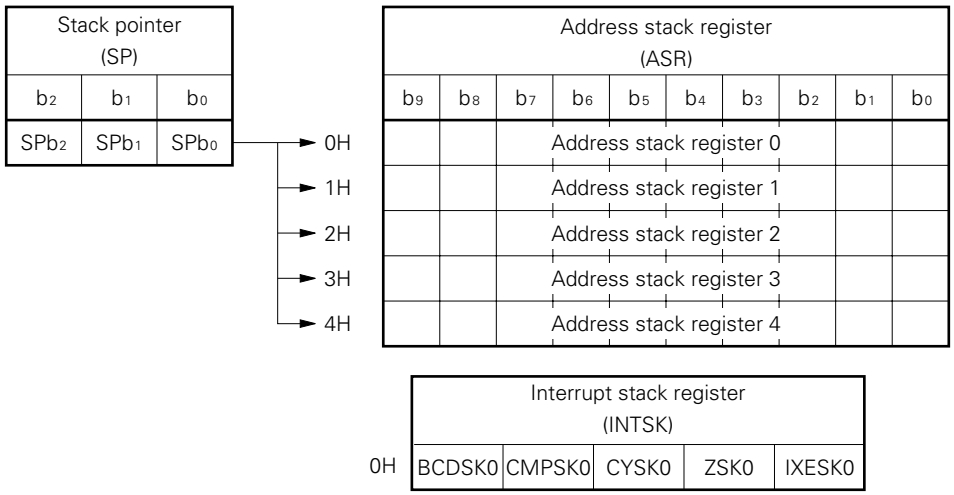
(2) Do not use instructions which branch to addresses after the 768th step (address 02FFH).

6. STACK

Fig. 6-1 shows the stack configuration. The stack consists of five address stack registers and one interrupt stack register.

The stack is used to save the return address during execution of subroutine calls and table reference instructions. When an interrupt occurs, the program return address and the program status work (PSWORD) are automatically saved in the stack. Then, all bits of the bank and PSWORD are cleared to 0.

Fig. 6-1 Stack Configuration



7. DATA MEMORY (RAM)

Data memory (RAM) stores data such as operation and control data. Data can be read from or written to data memory with an instruction during normal operation.

7.1 DATA MEMORY ORGANIZATION

Data memory locations have 7-bit addresses. The three high-order bits of each address are called the row address, and the four low-order bits are called the column address.

For example, the row address of address 1AH is 1H. The column address is 0AH.

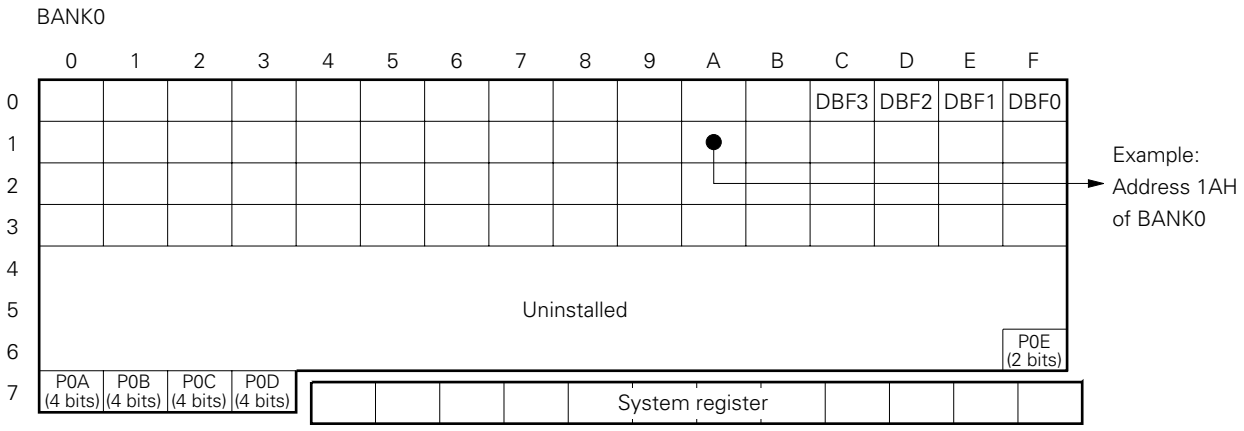
Each addressed memory location is 4-bits (one nibble) long.

Data memory contains an area to which the user is allowed to store data freely, as well as areas which are reserved for the use of specific functions.

The areas reserved for specific functions are as follows:

- System register (SYSREG) (See **Chapter 9**.)
- Data buffer (DBF) (See **Chapter 11**.)
- Port registers (See **Chapter 13**.)

Fig. 7-1 Organization of Data Memory



7.2 UNINSTALLED DATA MEMORY

There is no hardware installed for addresses 40H to 6EH. Any attempt to read from this area will yield an unpredictable value. Any instructions to write data to this area are invalid and must not be used.

If this uninstalled data memory area is used by a 17K-series assembly (AS17K) program or by an in-circuit emulator (IE-17K or IE-17K-ET), the following occurs:

AS17K: An error is indicated.

IE-17K or IE-17K-ET: Write instructions are invalidated and read instructions read 0.

8. GENERAL REGISTER (GR)

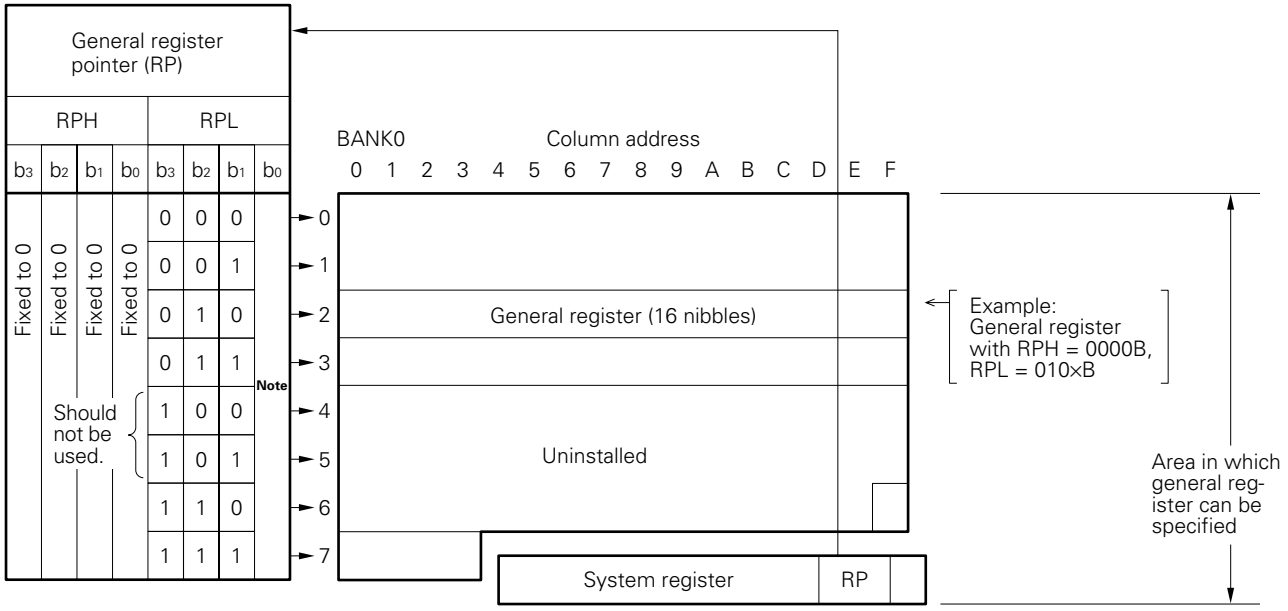
The general register, as the name implies, is a general register used for data transfer and manipulation. In the 17K series, the location of the general register is not fixed. The area used for the general register is in data memory, as specified by the general register pointer (RP). Thus, part of the data memory area can be specified as the general register as required, allowing data transfer in data memory and data memory manipulation to be performed with a single instruction.

8.1 GENERAL REGISTER POINTER (RP)

RP is a pointer used to specify part of data memory as the general register. In RP, specify a desired data memory bank and row address for the general register. RP consists of seven bits: 7DH (RPH), and the three high-order bits of 7EH (RPL) in the system register (see **Chapter 9**).

Set a bank in RPH, and a data memory row address in RPL.

Fig. 8-1 General Register Pointer Configuration



Note Allocated to the flag BCD

9. SYSTEM REGISTER (SYSREG)

The system register (SYSREG), located in data memory, is used for direct control of the CPU.

9.1 SYSTEM REGISTER CONFIGURATION

Fig. 9-1 shows the allocation address of the system register in data memory. As shown in Fig. 9-1, the system register is allocated in addresses 74H to 7FH of data memory.

Since the system register is allocated in data memory, it can be manipulated using any of the instructions available for manipulating data memory. Therefore, it is also possible to put the system register in the general register.

Fig. 9-1 Allocation of System Register in Data Memory

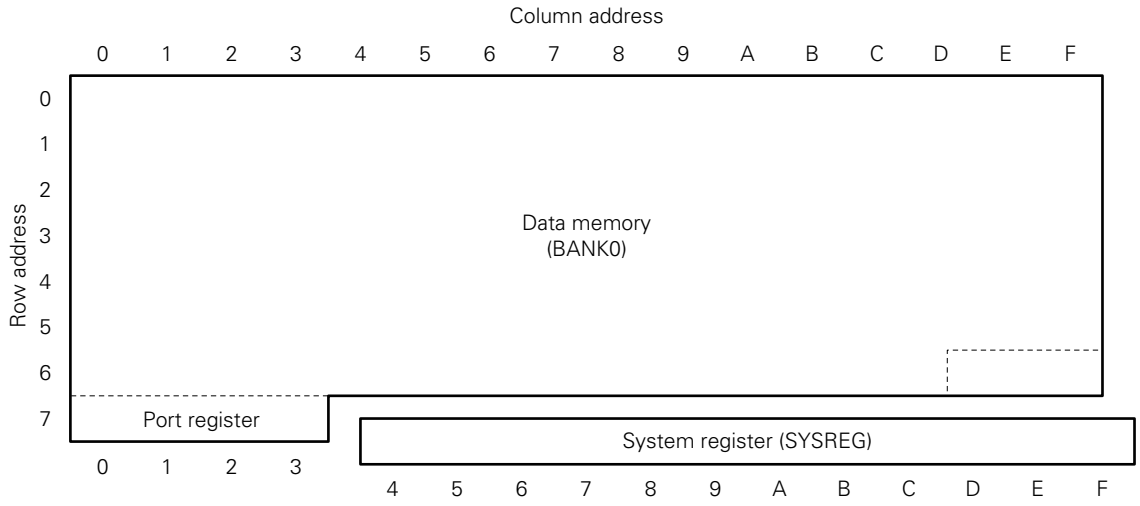


Fig. 9-2 shows the configuration of the system register. As shown in Fig. 9-2, the system register consists of the following seven registers.

- 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)

Fig. 9-2 System Register Configuration

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	AR3	AR2	AR1	AR0	WR	BANK	IXH / MPH	IXM / MPL	IXL	RPH	RPL	PSW
Bit	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0	b_3 b_2 b_1 b_0
Data[Note]	0 0 0 0 (AR)	0 0				0 0 0 0 (BANK)	MPE 0 0 0 (MP)	0 (IX)		0 0 0 0 (RP)		BCD CMP CY Z IXE
Initial value when reset	0 0 0 0	0 0	0 0 0 0	0 0 0 0	Not defined	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 0

Note A bit for which 0 is written is fixed at 0.

★ **Remark** Once the contents of PSWORD are saved in the interrupt stack register, all the five bits of PSWORD are cleared to 0.

10. REGISTER FILE (RF)

The register file is a register used mainly for specifying conditions for peripheral hardware.

The register file can be controlled using dedicated instructions PEEK and POKE or AS17K macro instructions SETn, CLRn, and INITFLG.

10.1 REGISTER FILE CONFIGURATION

10.1.1 Configuration of the Register File

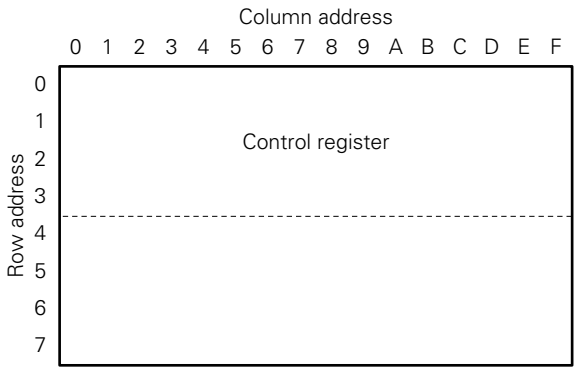
Fig. 10-1 shows the configuration of the register file.

As shown in Fig. 10-1, the register file is a register consisting of 128 nibbles (128 × 4 bits).

In the same way as with data memory, the register file is divided into addresses in units of four bits. It has a total of 128 nibbles specified in row addresses from 0H to 7H and column addresses from 0H to 0FH.

Address locations 00H to 3FH define an area called the control register.

Fig. 10-1 Register File Configuration



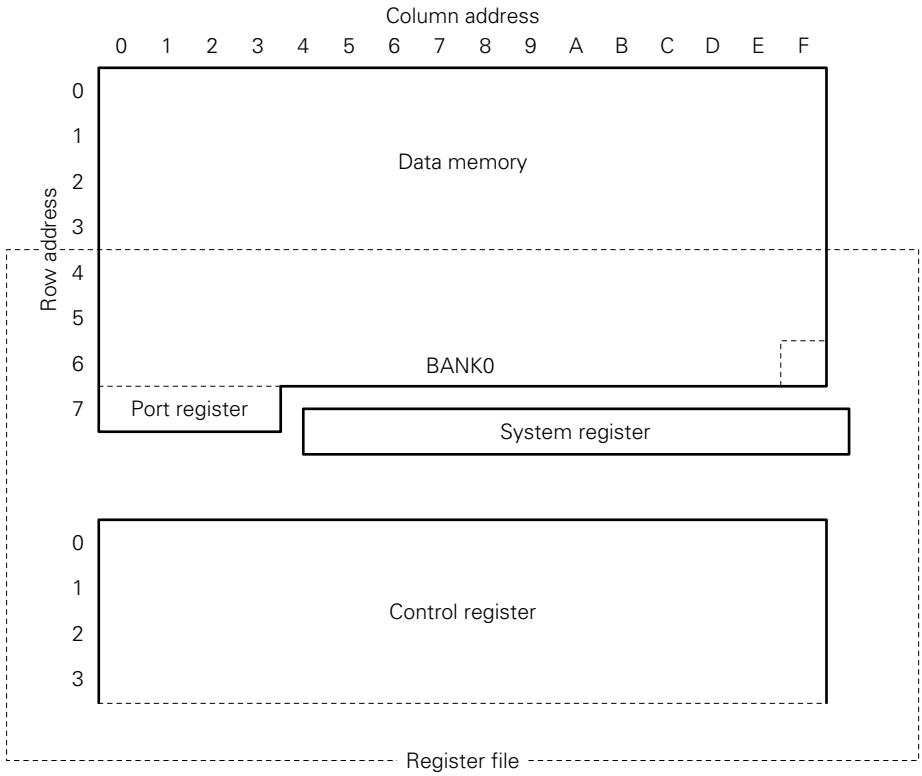
10.1.2 Relationship between the Register File and Data Memory

Fig. 10-2 shows the relationship between the register file and data memory.

As shown in Fig. 10-2, the register file overlaps with data memory at addresses 40H to 7FH.

This means that, on a program, it seems that the same memory exists in the register file at addresses 40H to 7FH and in the data memory at addresses 40H to 7FH.

Fig. 10-2 Relationship Between the Register File and Data Memory



10.2 FUNCTIONS OF THE REGISTER FILE

10.2.1 Functions of the Register File

The register file is a collection of registers in which peripheral hardware conditions are set with the PEEK instruction or POKE instruction.

The register used to control the peripheral hardware is located at addresses 00H to 3FH. This area is called the control register.

Addresses 40H to 7FH of the register file constitute normal data memory. Thus, not only the MOV instruction, but also the PEEK and POKE instructions, can be used to enable this part to perform read and write operations.

10.2.2 Control Register Functions

The peripheral hardware whose conditions can be controlled by control registers is listed below.

For details concerning peripheral hardware and the control register, see the section for the peripheral hardware concerned.

- Ports
- 8-bit timer counter (TM)
- Serial interface (SIO)
- Interrupt function
- Stack pointer (SP)

11. DATA BUFFER (DBF)

The data buffer consists of four nibbles allocated in addresses 0CH to 0FH in BANK0.

The data buffer acts as a data storage area for the CPU peripheral hardware (address register, serial interface, and timer) through use of the GET and PUT instructions. It also acts as data storage used for receiving and transferring data. By using the MOVT DBF, and @AR instructions, fixed data in program memory can be read into the data buffer.

11.1 DATA BUFFER CONFIGURATION

Fig. 11-1 shows the allocation of the data buffer in data memory.

As shown in Fig. 11-1, the data buffer is allocated in address locations 0CH to 0FH in data memory and consists of 4 nibbles (4 × 4 bits), totaling 16 bits.

Fig. 11-1 Allocation of the Data Buffer

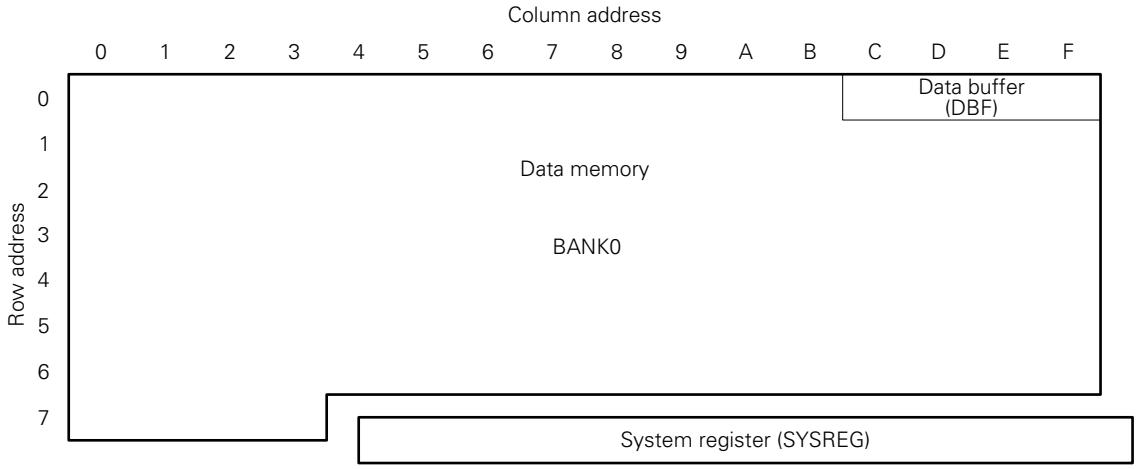


Fig. 11-2 shows the configuration of the data buffer. As shown in Fig. 11-2, the data buffer is made up of sixteen bits with its least significant bit in bit 0 of address 0FH and its most significant bit in bit 3 of address 0CH.

Fig. 11-2 Data Buffer Configuration

Data memory BANK0	Address	0CH				0DH				0EH				0FH			
	Bit	b_3	b_2	b_1	b_0	b_3	b_2	b_1	b_0	b_3	b_2	b_1	b_0	b_3	b_2	b_1	b_0
Data buffer	Bit	b_{15}	b_{14}	b_{13}	b_{12}	b_{11}	b_{10}	b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
	Symbol	DBF3				DBF2				DBF1				DBF0			
	Data	^ MSB v				Data											^ LSB v

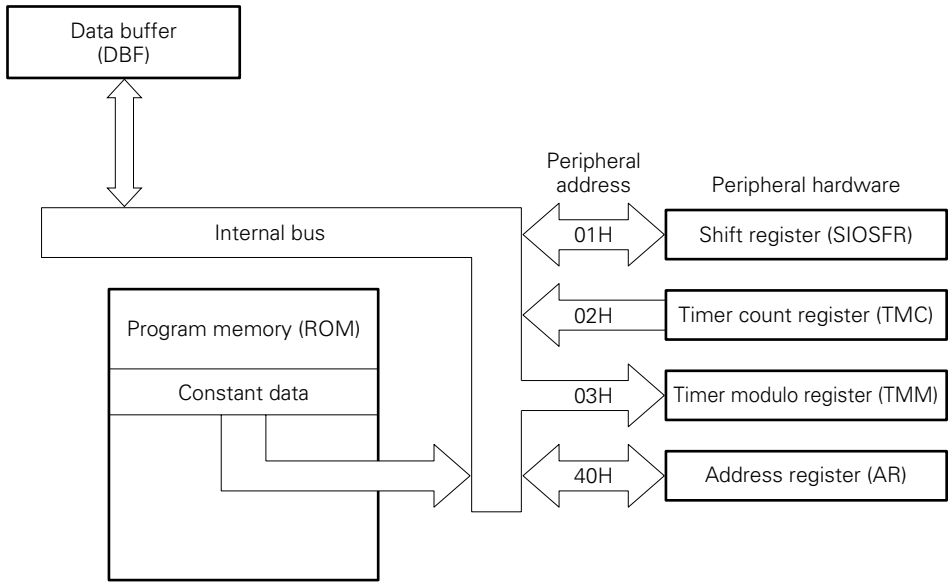
Because the data buffer is allocated in data memory, it can be used in any of the data memory manipulation instructions.

11.2 FUNCTIONS OF THE DATA BUFFER

The data buffer has two separate functions.

The data buffer is used for data transfer with peripheral hardware. The data buffer is also used for reading constant data in program memory. Fig. 11-3 shows the relationship between the data buffer and peripheral hardware.

Fig. 11-3 Relationship Between the Data Buffer and Peripheral Hardware



12. ALU BLOCK

The ALU is used for performing arithmetic operations, logical operations, bit evaluations, comparison evaluations, and rotations on 4-bit data.

12.1 ALU BLOCK CONFIGURATION

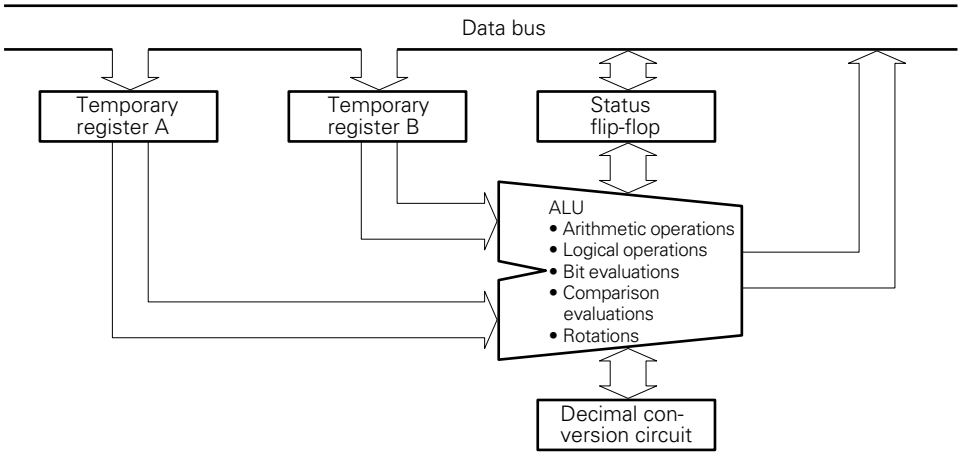
Fig. 12-1 shows the configuration of the ALU block.

As shown in Fig. 12-1, the ALU block consists of the main 4-bit data processor, temporary registers A and B, the status flip-flop for controlling the status of the ALU, and the decimal conversion circuit for use during arithmetic operations in BCD.

As shown in Fig. 12-1, the status flip-flop consists of the following flags: Zero flag flip-flop, carry flag flip-flop, compare flag flip-flop, and the BCD flag flip-flop.

Each flag in the status flip-flop corresponds directly to a flag in the program status word (PSWORD: addresses 7EH, 7FH) located in the system register. The flags in the program status word are the following: Zero flag (Z), carry flag (CY), compare flag (CMP), and the BCD flag (BCD).

Fig. 12-1 Configuration of the ALU



Address	7EH	7FH			
Name	Program status word (PSWORD)				
Bit	b_0	b_3	b_2	b_1	b_0
Flag	BCD	CMP	CY	Z	IXE

Status flip-flop			
BCD flag flip-flop	CMP flag flip-flop	CY flag flip-flop	Z flag flip-flop

Function outline
Indicates when the result of an arithmetic operation is 0.
Stores the borrow or carry from an arithmetic operation.
Used to indicate whether to store the result of an arithmetic operation.
Used to indicate whether to perform decimal correction for arithmetic operations.

13. PORTS

13.1 PORT 0A ($P0A_0$, $P0A_1$, $P0A_2$, $P0A_3$)

Port 0A is a 4-bit input/output port with an output latch. It is mapped into address 70H in data memory. The output format is CMOS push-pull output.

Input or output can be specified bit-by-bit. Input/output can be specified by P0ABIO0 to P0ABIO3 (address 35H) in the register file.

At reset, P0ABIOn is 0 (n = 0 to 3) and all P0A pins are input ports. The contents of the port output latch are 0.

Table 13-1 Writing into and Reading from the Port Register (0.70H)

(n = 0 to 3)

P0ABIOn RF: 35H	Pin input/output	BANK0 70H	
		Write	Read
0	Input	Possible Write to the P0A latch	P0A pin status
1	Output		P0A latch contents

13.2 PORT 0B ($P0B_0$, $P0B_1$, $P0B_2$, $P0B_3$)

Port 0B is a 4-bit input/output port with an output latch. It is mapped into address 71H of BANK0 in data memory. The output format is CMOS push-pull output.

Input or output can be specified in 4-bit units. Input/ output is specified by P0BGIO (bit 0 in address 24H) in the register file.

At reset, P0BGIO is 0 and all P0B pins are input ports. The value of the port 0B output latch is 0.

Table 13-2 Writing into and Reading from the Port Register (0.71H)

P0BGIO RF: 24H, bit 0	Pin input/output	BANK0 71H	
		Write	Read
0	Input	Possible Write to the P0B latch	P0B pin status
1	Output		P0B latch contents

13.3 PORT 0C ($P0C_0$, $P0C_1$, $P0C_2$, $P0C_3$)

Port 0C is a 4-bit input/output port with an output latch. It is mapped into address 72H of BANK0 in data memory. The output format is CMOS push-pull output.

Input or output can be specified bit-by-bit. Input/output can be specified by P0CBIO0 to P0CBIO3 (address 34H) in the register file.

At reset, P0CBIOn is 0 (n = 0 to 3) and all P0C pins are input ports. The contents of the port output latch are 0.

Table 13-3 Writing into and Reading from the Port Register (0.72H)

(n = 0 to 3)

P0CBIOn RF: 34H	Pin input/output	BANK0 72H	
		Write	Read
0	Input	Possible Write to the P0C latch	P0C pin status
1	Output		P0C latch contents

13.4 PORT 0D (P0D0/$\overline{SCK}$, P0D1/SO, P0D2/SI, P0D3/$\overline{TMOUT}$)

Port 0D is a 4-bit input/output port with an output latch. It is mapped into address 73H in data memory. The output format is N-ch open-drain output. By mask option, the port can contain pull-up resistors bit-by-bit.

Input or output can be specified bit-by-bit. Input/output is specified with P0DBIO0 to P0DBIO3 (address 33H) in the register file.

At reset, P0DBIOn is set to 0 (n = 0 to 3) and all P0D pins become input ports. The contents of the port output latch become 0. The output latch contents remain unchanged even if P0DBIOn changes from 1 to 0.

Port 0D can also be used for serial interface input/output or timer output. SIOEN (bit 0 in address 0AH) in the register file is used to switch ports (P0D0 to P0D2) to serial interface input/output ($\overline{SCK}$, SI, SO) and vice versa. TMOSEL (bit 0 in address 12H) in the register file is used to switch a port (P0D3) to timer output ($\overline{TMOUT}$) and vice versa. If TMOSEL = 1 is selected, 1 is output at timer reset. This output is inverted every time a timer count value matches the modulo register contents.

Table 13-4 Register File Contents and Pin Functions

(n = 0 to 3)

Register file value			Pin function			
TMOSEL RF: 12H Bit 0	SIOEN RF: 0AH Bit 0	P0DBIOn RF: 33H	P0D0/$\overline{SCK}$	P0D1/SO	P0D2/SI	P0D3/$\overline{TMOUT}$
0	0	0	Input port			
		1	Output port			
	1	0	$\overline{SCK}$	SO	SI	Input port
		1				Output port
1	0	0	Input port			$\overline{TMOUT}$
		1	Output port			
	1	0	$\overline{SCK}$	SO	SI	
		1				

Table 13-5 Contents Read from the Port Register (0.73H)

Port mode		Contents read from the port register (0.73H)
Input port		Pin status
Output port		Output latch contents
$\overline{SCK}$	An internal clock is selected as a serial clock.	Output latch contents
	An external clock is selected as a serial clock.	Pin status
SO		Undefined**Note**
SI		Pin status
$\overline{TMOUT}$		Output latch contents

Note See **Chapter 15** for details.

13.5 PORT 0E ($P0E_0$, $P0E_1$)

Port 0E is a 2-bit input/output port with an output latch. It is mapped into bits 0 and 1 in address 6FH in data memory. The output format is N-ch open-drain output. By mask option, the port can contain pull-up resistors bit-by-bit.

Input or output can be specified bit-by-bit. Input/output is specified by P0EBIO0 and P0EBIO1 (bits 0 and 1 in address 32H) in the register file.

When a read instruction is executed, not the output latch data but the pin status is read regardless of the input or output mode.

At reset, P0EBIOn is set to 0 (n = 0 and 1) and each P0E pin becomes input port. The contents of the port output latch are 0.

The write instruction specified for bits 2 and 3 of address 6FH is invalidated. If it is executed, 0 is read out.

Table 13-6 Writing into and Reading from the Port Register (0.6FH.0 and 0.6FH.1)

(n = 0 and 1)

P0EBIOn RF: 32H	Pin input/output	BANK0 6FH	
		Write	Read
0	Input	Possible	P0E pin status
1	Output	Write to the P0E output latch	

★ 13.6 NOTES ON MANIPULATING PORT REGISTERS

The states of only the port 0E pins of the μPD17120 can be read even when the port pins have been set to output mode.

When a port register is manipulated with a built-in macro instruction (such as SETn or CLRn) or an AND, OR, or XOR instruction, the states of those pins for which the state should remain unchanged may change unexpectedly.

Especially when the port 0E pins are set to low externally, always take the possibility of this change in the states of the pins into consideration.

When a CLR1 P0E1 instruction (identical to an AND 6FH, #1101B instruction) is applied to the port 0E pins, the corresponding port register and internal states are changed, as shown in Fig. 13-1.

Assume that the states of port 0E are those shown in Fig. 13-1 #. Pins $P0E_1$ and $P0E_0$, both used as output pins, output high level, while pin $P0E_0$ forcibly set to low externally.

(Although the μPD17120 does not support pins $P0E_3$ and $P0E_2$, they are virtually assumed to exist within a program.)

When a CLR1 P0E1 instruction is executed to set pin $P0E_1$ to low, the states of the port 0E pins change as shown in Fig. 13-1 $. The port register changes such that pin $P0E_1$ output low level and pin $P0E_0$, required to output high level, actually output low level. This is because the CLR1 P0E1 instruction has been applied to the states of the port 0E pins, but not to the states of the port register.

To prevent this problem, use another instruction, such as a MOV instruction, to specify the states of all port 0E pins, not merely the states of those pins whose states are to be changed. In this example, it is recommended that a MOV 6FH, #1101B instruction be used to set only pin $P0E_1$ to low.

Fig. 13-1 Changes in the Port Register According to the Execution of a CLR1 P0E1 Instruction

Before the instruction is executed

	$P0E_3$	$P0E_2$	$P0E_1$	$P0E_0$
Port register	Does not exist.		1	1
Internal state	—	—	H output	H output
Pin state	—	—	H	L (forcible)

Executing a CLR1 P0E1 instruction [AND 6FH, #1101B]

$ **After the instruction is executed**

	$P0E_3$	$P0E_2$	$P0E_1$	$P0E_0$
Port register	Does not exist.		0	0
Internal state	—	—	L output	L output
Pin state	—	—	L	L

H: High level, L: Low level

14. 8-BIT TIMER COUNTER (TM)

An 8-bit timer counter is incorporated in μPD17120.

The timer is controlled by hardware operation with the PUT/GET instruction or by register operation in the register file with the PEEK/POKE instruction.

14.1 CONFIGURATION OF 8-BIT TIMER COUNTER

Fig. 14-1 shows the configuration of the 8-bit timer counters. An 8-bit timer counter consists of an 8-bit counter register, 8-bit modulo register, comparator (compares counter register values and modulo register values), and selector (for count pulse selection).

Caution The modulo register is a write-only register.
The counter register is a read-only register.

Fig. 14-1 Configuration of the 8-Bit Timer Counter ★

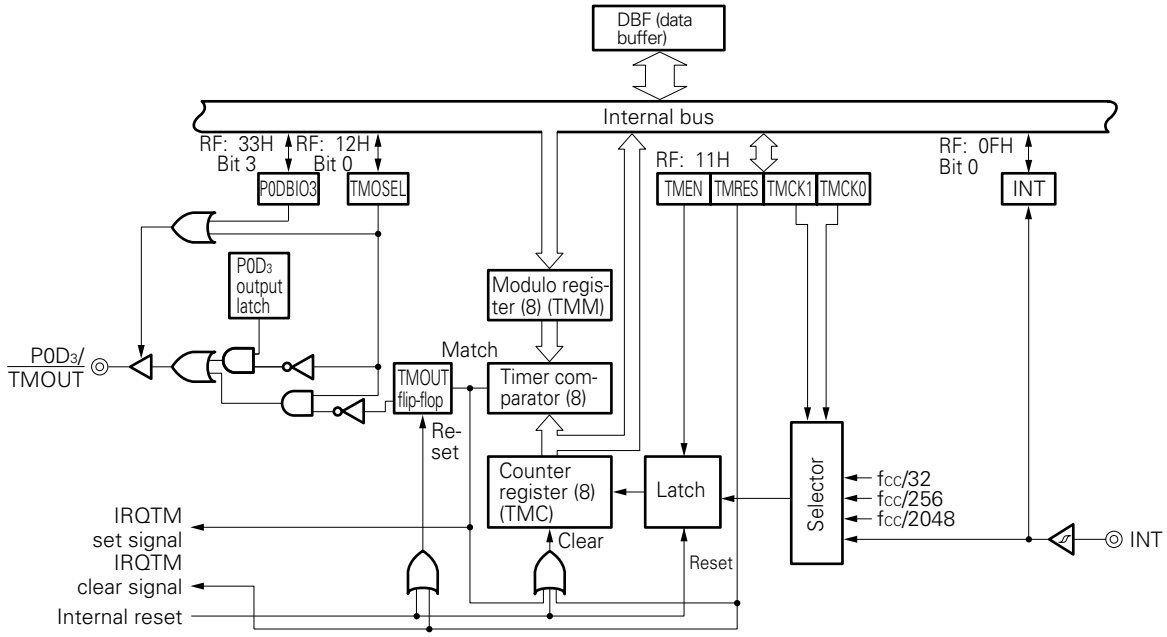


Table 14-1 Source Clock

Register file value		Source clock to be selected
TMCK1	TMCK0	
0	0	$f_{CC}/256$
0	1	$f_{CC}/32$
1	0	$f_{CC}/2048$
1	1	External clock input to the INT pin

14.2 OUTPUTTING A TIMER SIGNAL

The P0D3/$\overline{\text{TMOUT}}$ pin functions as a timer match signal output pin when the TMOSEL flag is set to 1. The P0DBIO3 value has nothing to do with this setting.

The timer contains a match signal output flip-flop. It reverses the output each time the comparator of the 8-bit timer outputs a match signal. When the TMOSEL flag is set to 1, the contents of this flip-flop are output to the P0D3/$\overline{\text{TMOUT}}$ pin.

The P0D3/$\overline{\text{TMOUT}}$ pin is an N-ch open-drain output pin. The mask option enables this pin to contain a pull-up resistor. If this pin does not contain a pull-up resistor, its initial status is high impedance.

An internal timer output flip-flop starts operating when TMEN is set to 1. To make the flip-flop start output beginning at an initial value, set 1 in TMRES and reset the flip-flop.

15. SERIAL INTERFACE (SIO)

The serial interface consists of an 8-bit shift register (SIOSFR), serial mode register, and serial clock counter. It is used for serial data input/output.

15.1 FUNCTIONS OF THE SERIAL INTERFACE

This serial interface provides three signal lines: serial clock input pin ($\overline{SCK}$), serial data output pin (SO), and serial data input pin (SI). It allows 8 bits to be sent or received in synchronization with clocks. It can be connected to peripheral input/output devices using any method with a mode compatible to that used by the μPD7500 or 75X series.

(1) Serial clock

Three types of internal clocks and one type of external clock are able to be selected. If an internal clock is selected as a serial clock, it is automatically output to the $P0D_0/\overline{SCK}$ pin.

Table 15-1 Shift Clock

Register file value		Shift clock to be selected
SIOCK1	SIOCK0	
0	0	External clock input to the $\overline{SCK}$ pin
0	1	$f_{CC}/16$
1	0	$f_{CC}/128$
1	1	$f_{CC}/1024$

(2) Transmission

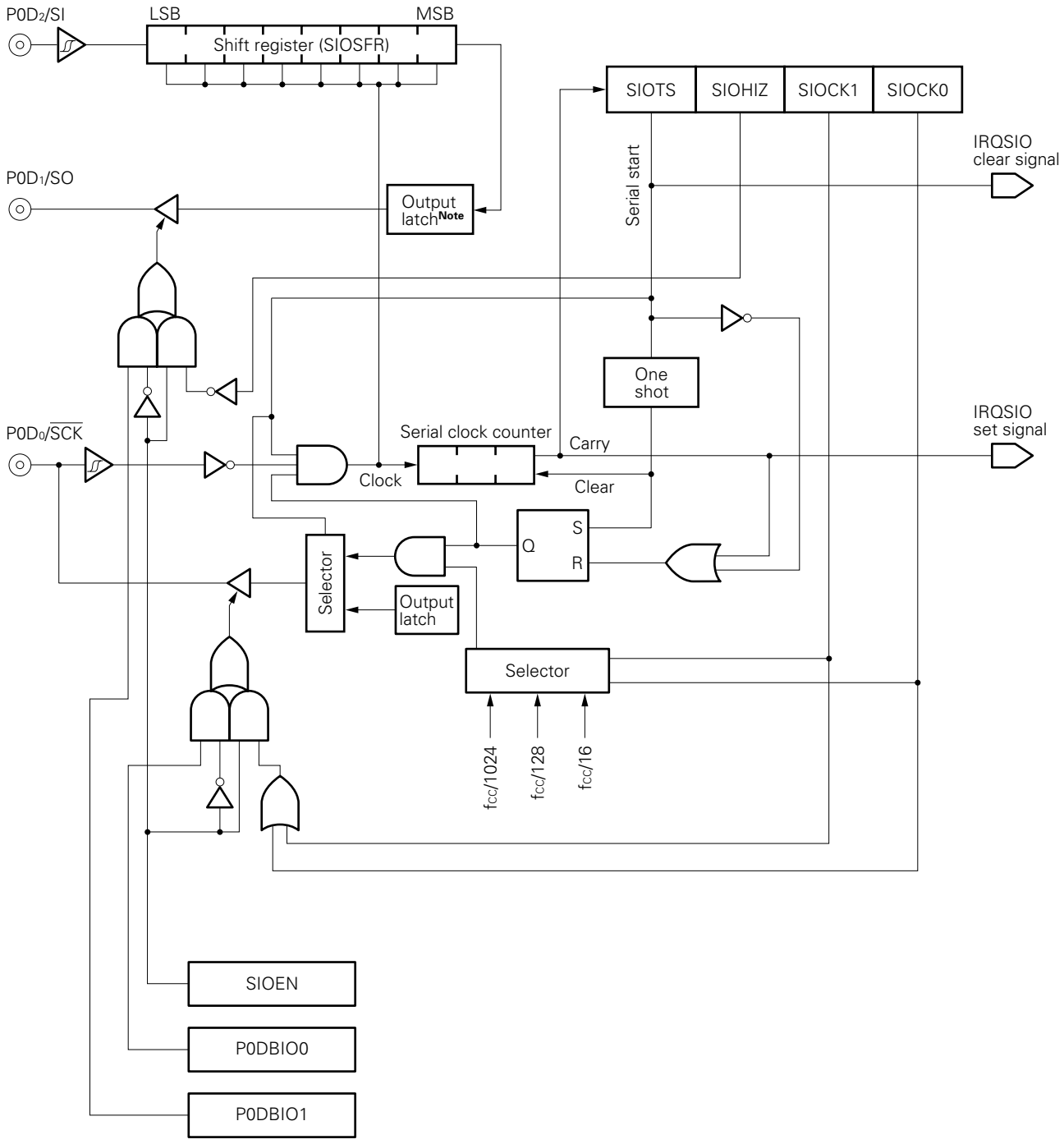
When SIOEN is set to 1, the pins of port 0D ($P0D_0/\overline{SCK}$, $P0D_1$/SO, $P0D_2$/SI) function as the pins of the serial interface. The serial interface operates in synchronization with the falling edge of the external or internal clock by setting SIOTS to 1. When SIOTS is set to 1, IRQSIO is automatically cleared.

Transmission starts from the most significant bit of the shift register in synchronization with the falling edge of the serial clock. SI pin information is stored in the shift register starting at the most significant bit in synchronization with the rising edge of the serial clock.

When the transfer of 8-bit data is completed, SIOTS is automatically cleared to 0 and IRQSIO is set to 1.

Remark Serial transmission starts only from the most significant bit of the shift register contents. It is not possible to start transmission from the least significant bit. SI pin status is always stored in the shift register in synchronization with the rising edge of the serial clock.

Fig. 15-1 Block Diagram of the Serial Interface



★ **Note** The output latch of the shift register is also used as that of the P0D1 pin. Therefore, executing an output instruction for the P0D1 pin changes the output latch status of the shift register.

15.2 3-WIRE SERIAL INTERFACE OPERATION MODES

Two modes can be used for the serial interface. If the serial interface function is selected, the P0D2/SI pin always takes in data in synchronization with the serial clock.

- 8-bit transmission and reception mode (simultaneous transmission and reception)
- 8-bit reception mode (with the SO pin set to the high impedance status)

Table 15-2 Serial Interface Operation Mode

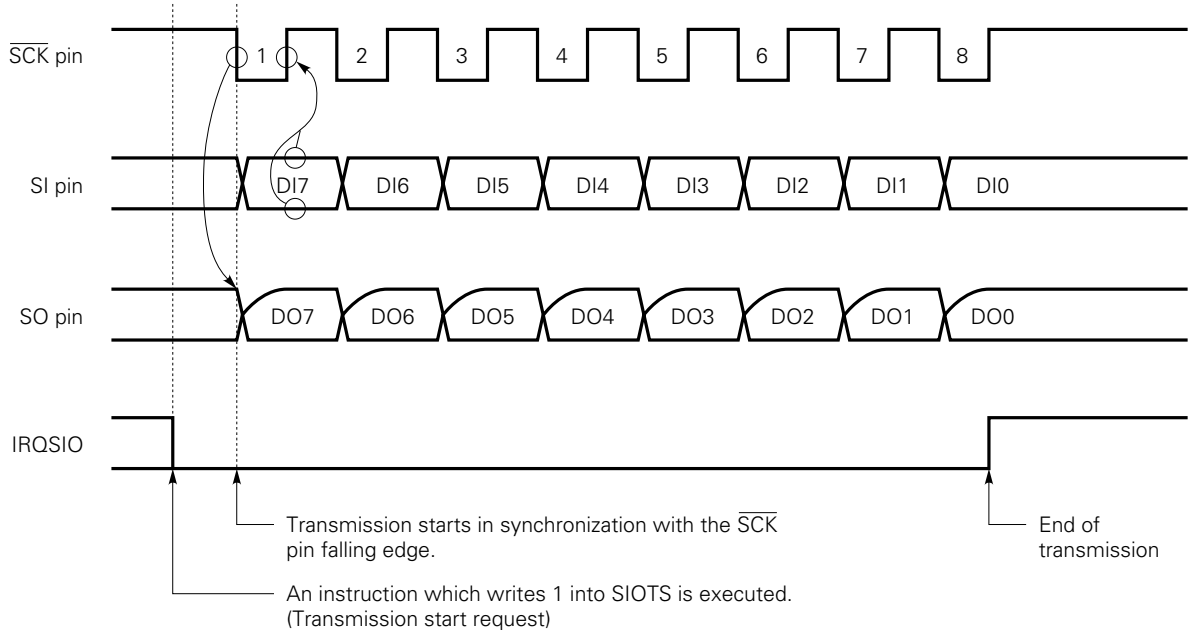
SIOEN	SIOHIZ	P0D2/SI pin	P0D1/SO pin	Serial interface operation mode
1	0	SI	SO	8-bit transmission and reception mode
1	1	SI	P0D1 (input)	8-bit reception mode
0	×	P0D2 (I/O)	P0D1 (I/O)	General port mode

×: Don't care

(1) 8-bit transmission and reception mode (simultaneous transmission and reception)

Serial data input/output is controlled by a serial clock. The most significant bit of the shift register is output from the SO line with a falling edge of the serial clock ($\overline{SCK}$). The contents of the shift register is shifted one bit and at the same time, data on the SI line is loaded into the least significant bit of the shift register. The serial clock counter counts serial clock pulses. Every time it counts eight clocks, the internal interrupt request flag is set to 1 (IRQSIO ← 1).

Fig. 15-2 Timing of 8-Bit Transmission and Reception Mode (Simultaneous Transmission and Reception)



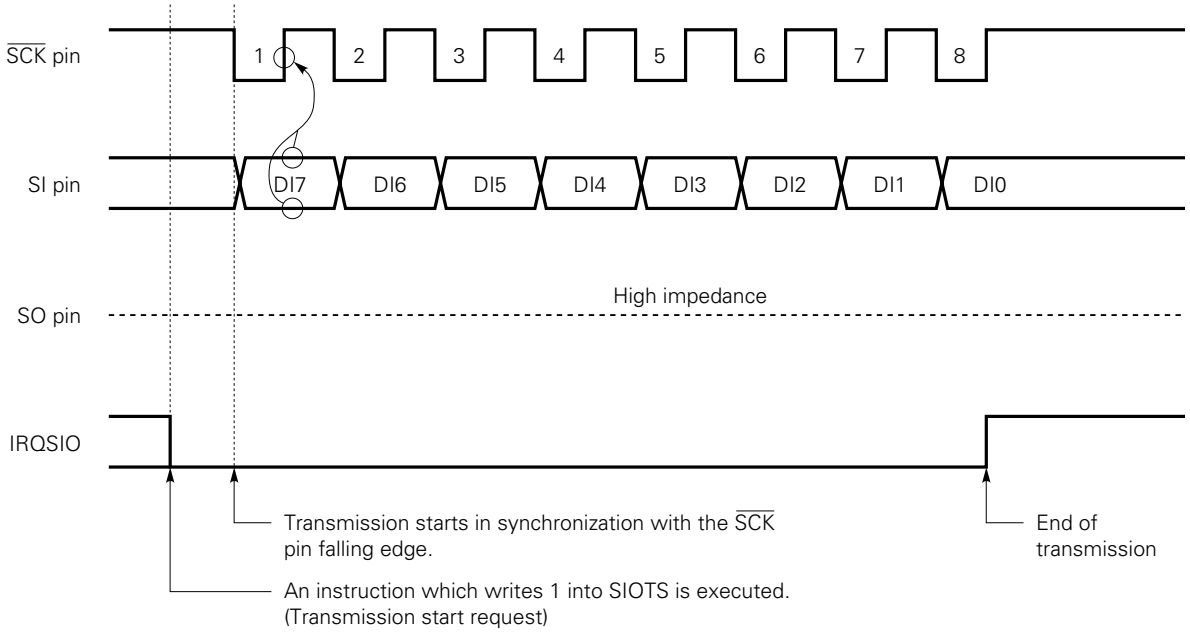
Remark DIn : Input serial data
DOn: Output serial data

(2) 8-bit transmission and reception mode (SO pin in the high impedance status)

When SIOHIZ is 1, the P0D1/SO pin is in the high impedance status. If serial clock supply starts by writing 1 in SIOTS, only the reception function of the serial interface operates.

The P0D1/SO pin is in the high impedance status and can be used for input port (P0D1).

Fig. 15-3 Timing of the 8-Bit Reception Mode



Remark DIn: Input serial data

(3) Operation stop mode

If the value in SIOTS (RF: 1AH, bit 3) is 0, the serial interface enters operation stop mode. In this mode, no serial transfer occurs.

In this mode, the shift register does not perform shifting and can be used as an ordinary 8-bit register.

16. INTERRUPT FUNCTIONS

The μPD17120 has two internal interrupt functions and one external interrupt function. It can be used in various applications.

The interrupt control circuit of the μPD17120 has the features listed below. This circuit enables very high-speed interrupt handling.

(a) Used to determine whether an interrupt can be accepted with the interrupt mask enable flag (INTE), which is controlled by the EI or DI instruction, and interrupt enable flag (IP×××).
(b) The interrupt request flag (IRQ×××) can be tested or cleared. (Interrupt generation can be checked by software.)
(c) Standby mode (STOP, HALT) can be released by an interrupt request. (Release source can be selected by the interrupt enable flag.)

Cautions 1. In interrupt handling, the BCD, CMP, CY, Z, and IXE flags are saved in the stack automatically by the hardware for up to three levels of multiple interrupts. The DBF and WR are not saved by the hardware when peripheral hardware such as the timer or serial interface is accessed in interrupt handling. It is recommended that the DBF and WR be saved in RAM by the software at the beginning of interrupt handling. Saved data can be loaded back into the DBF and WR immediately before the end of interrupt handling.

2. Since the interrupt stack has only one level, multiple interrupts cannot be performed by hardware. When more than one interrupt is received, the data from the first interrupt is lost.

16.1 INTERRUPT SOURCE TYPES AND VECTOR ADDRESSES

For every interrupt in the μPD17120, when the interrupt is accepted, a branch occurs to the vector address associated with the interrupt source. This method is called the vectored interrupt method. Table 16-1 lists the interrupt source types and vector addresses.

If two or more interrupt requests occur or multiple suspended interrupt requests are enabled at the same time, they are handled according to priorities shown in Table 16-1. ★

Table 16-1 Interrupt Source Types

Interrupt source	Priority	Vector address	IRQ flag	IP flag	IEG flag	Internal/ external	Remarks
INT pin (RF:0FH, bit 0)	1	0003H	IRQ RF:3FH, bit 0	IP RF:2FH, bit 0	IEGMD0,1 RF:1FH bit 0, 1	External	Rising edge, falling edge or rising/falling edge (both) can be selected.
Timer	2	0002H	IRQTM RF:3EH, bit 0	IPTM RF:2FH, bit 1	–	Internal	
Serial interface	3	0001H	IRQSIO RF:3DH, bit 0	IPSIO RF:2FH, bit 2	–	Internal	

16.2 HARDWARE COMPONENTS OF THE INTERRUPT CONTROL CIRCUIT

The flags of the interrupt control circuit are explained below.

(1) Interrupt request flag and the interrupt enable flag

The interrupt request flag (IRQ×××) is set to 1 when an interrupt request occurs. When interrupt handling is executed, the flag is automatically cleared to 0.

An interrupt enable flag (IP×××) is provided for each interrupt request flag. If the flag is 1, an interrupt is enabled. If it is 0, the interrupt is disabled.

(2) EI/DI instruction

The EI/DI instruction is used to determine whether an accepted interrupt is to be executed.

If the EI instruction is executed, the interrupt enable flag (INTE) for enabling interrupt reception is set. Since the INTE flag is not registered in the register file, flag status cannot be checked by instructions.

The DI instruction clears the INTE flag to 0 and disables all interrupts.

At reset the INTE flag is cleared to 0 and all interrupts are disabled.

Table 16-2 Interrupt Request Flag and Interrupt Enable Flag

Interrupt request flag	Signal for setting the interrupt request flag	Interrupt enable flag
IRQ	Set by edge detection of an INT pin input signal. A detection edge is selected by IEGMD0 or IEGMD1.	IP
IRQTM	Set by a match signal from timer.	IPTM
IRQSIO	Set by a serial data transmission end signal from the serial interface.	IPSIO

17. STANDBY FUNCTION ★

17.1 OVERVIEW OF THE STANDBY FUNCTION

The μPD17120 can reduce its current by using the standby function. The standby function supports STOP and HALT modes.

In the STOP mode, the system clock is stopped and the CPU current is reduced to almost only a leak current. This mode is useful in retaining data memory contents without operating the CPU.

In the HALT mode, the oscillation of the system clock continues. However, the system clock is not supplied to the CPU, stopping CPU operation. In this mode, current reduction is less than that in the STOP mode. However, since the system clock is oscillating, operation can be started immediately after the HALT mode is released. In both STOP and HALT modes, the statuses of the data memory, registers, and output latches of the output port used immediately before the standby mode is set are maintained (except STOP 0000B). Therefore, in order to lower consumption current for the entire system, input/output port statuses should be set beforehand.

Table 17-1 Standby Mode Status

		STOP mode	HALT mode
Programmed instruction		STOP instruction	HALT instruction
Clock oscillator		Oscillation stopped	Oscillation continued
Operation status	CPU	• Operation stopped	
	RAM	• The contents held immediately before setting standby mode are retained.	
	Port	• The status existing immediately before setting standby mode is retained.**Note**	
	TM	• Operation stopped. (The count is reset to 0.) (Count-up is also inhibited.)	• Operable
	SIO	• Operable only when the external clock is selected as the shift clock.**Note**	• Operable
	INT	• Operable	

Note When STOP 0000B is executed, all pins are set to input port mode even if the pins are used in dual-function mode.

Cautions 1. Always specify a NOP instruction immediately before STOP and HALT instructions.

2. When an interrupt request flag and the corresponding interrupt enable flag are both set, and the associated interrupt is specified as the standby mode release condition, the system does not enter the standby mode.

17.2 HALT MODE

17.2.1 Setting HALT Mode

Executing a HALT instruction sets HALT mode.

Operand $b_3b_2b_1b_0$ of the HALT instruction indicates the HALT mode release conditions.

Table 17-2 HALT Mode Release Conditions

Format: HALT $b_3b_2b_1b_0$B

Bit	HALT mode release conditions[Note 1]
b_3	When this bit is 1, release by IRQ×××is permitted.[Notes 2, 4]
b_2	Fixed at 0
b_1	When this bit is 1, forced release by IRQTM1 is permitted.[Notes 3, 4]
b_0	Fixed at 0

Notes 1. When HALT 0000B is specified, HALT mode can be released only by reset ($\overline{\text{RESET}}$ input or power-on/power-down reset).

2. IP××× must be 1.

3. HALT mode is released regardless of the IPTM status.

4. If a HALT instruction is executed when IRQ××× = 1, the HALT instruction is ignored (treated as a NOP instruction), and HALT mode is not set.

17.2.2 Starting Address After HALT Mode is Released

The starting address depends on the release conditions and interrupt enable conditions.

Table 17-3 Starting Address After HALT Mode is Released

Release condition	Starting address after release
Reset[Note 1]	Address 0
IRQ×××[Note 2]	For DI, address subsequent to the HALT instruction
	For EI, interrupt vector (When more than one IRQ××× is set, the interrupt vector having the highest priority)

Notes 1. $\overline{\text{RESET}}$ input and power-on/power-down reset are valid.

2. Except when forced release is made with IRQTM, IP××× must be 1.

17.3 STOP MODE

17.3.1 Setting STOP Mode

Executing a STOP instruction results in STOP mode being set.

Operand $b_3b_2b_1b_0$ of the STOP instruction indicates the STOP mode release conditions.

Table 17-4 STOP Mode Release Conditions

Format: STOP $b_3b_2b_1b_0$B

Bit	STOP mode release condition[Note 1]
b_3	When this bit is 1, release by IRQ××× is permitted.[Notes 2, 3]
b_2	Fixed at 0
b_1	Fixed at 0
b_0	Fixed at 0

Notes 1. When STOP 0000B is specified, STOP mode can be released only with reset ($\overline{\text{RESET}}$ input or power-on/power-down reset). When STOP 0000B is executed, the microcomputer is initialized to the state existing immediately after the reset.

2. IP××× must be 1. STOP mode cannot be released with IRQTM.

3. If the STOP instruction is executed when IRQ××× = 1, the STOP instruction is ignored (treated as a NOP instruction), and STOP mode is not set.

17.3.2 Starting Address After STOP Mode is Released

The starting address depends on the release conditions and interrupt enable conditions.

Table 17-5 Starting Address After STOP Mode is Released

Release condition	Starting address after release
Reset[Note 1]	Address 0
IRQ×××[Note 2]	For DI, address subsequent to the STOP instruction
	For EI, interrupt vector (When more than one IRQ××× is set, the interrupt vector having the highest priority)

Notes 1. $\overline{\text{RESET}}$ input and power-on/power-down reset are valid.

2. IP××× must be 1. STOP mode cannot be released with IRQTM.

18. RESET

This product provides three reset functions:

Reset by $\overline{\text{RESET}}$ input
$ Power-on/power-down reset at power-on or power voltage drop
% Address stack overflow or underflow reset

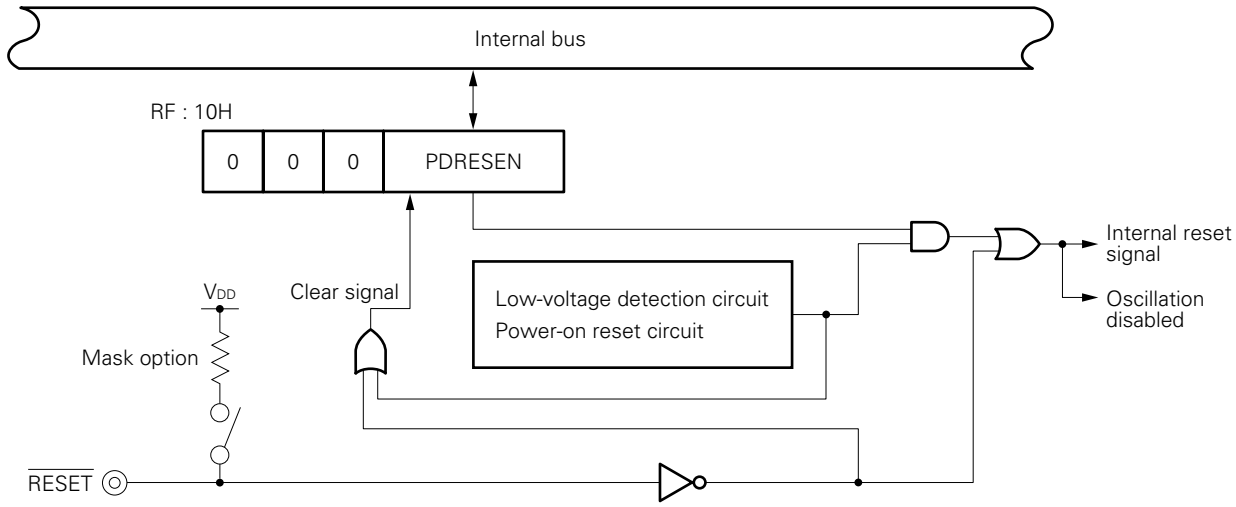
18.1 RESET FUNCTIONS

The reset functions are used to initialize device operations. The initialized hardware depends on the reset type. See **Table 18-1** for reset functions.

★ **Table 18-1 Hardware Statuses after Reset**

Hardware \ Reset type		• $\overline{\text{RESET}}$ input during operation • Built-in power-on/ power-down reset during operation	• $\overline{\text{RESET}}$ input in the standby mode • Built-in power-on/ power-down reset in the standby mode	• Stack overflow or underflow
Program counter		0000H	0000H	0000H
Port	Input/output	Input	Input	Input
	Output latch	0	0	Undefined
General-purpose data memory	Other than DBF	Undefined	Statuses before reset are retained	Undefined
	DBF	Undefined	Undefined	Undefined
System register	Other than WR	0	0	0
	WR	Undefined	Statuses before reset are retained	Undefined
Control register		SP = 5H, IRQTM1 = 1, TMEN = 1, and INT indicate the current status of the INT pin. The others are 0. See **Chapter 10**.		SP = 5H and INT indicate the current status of the INT pin. The others retain their statuses before reset.
Timer	Count register	00H	00H	Undefined
	Modulo register	FFH	FFH	FFH
Serial interface shift register (SIOSFR)		Undefined	Statuses before reset are retained	Undefined

Fig. 18-1 Reset Block Configuration



18.2 RESETTING

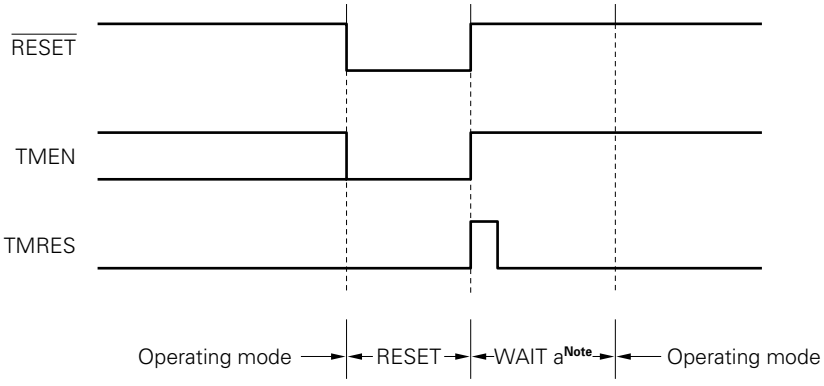
Operation when reset is caused by $\overline{\text{RESET}}$ input is shown in Fig. 18-2.

If the $\overline{\text{RESET}}$ pin is set from low to high, system clock generation starts and an oscillation stability wait occurs with the timer. Program execution starts from address 0000H.

If power-on reset is used, the reset signals shown in Fig. 18-2 are internally generated. Operation is the same as that when reset is caused by $\overline{\text{RESET}}$ input.

At stack overflow and underflow reset, oscillation stability wait time (WAIT a) does not occur. Operation starts from address 0000H after initial statuses are internally set.

Fig. 18-2 Resetting



Note This is oscillation stability wait time. Operating mode is set when the timer counts system clocks 256 × 256 times (approx. 32 ms at fcc = 2 MHz).

18.3 POWER-ON/POWER-DOWN RESET FUNCTION

The μPD17120 is provided with two reset functions to prevent malfunctions from occurring in the microcontroller. They are the power-on reset function and power-down reset function. The power-on reset function resets the microcontroller when it detects that power was turned on. The power-down reset function resets the microcontroller when it detects drops in the power voltage.

These functions are implemented by the power-voltage monitoring circuit whose operating voltage has a different range than the logic circuits in the microcontroller and the oscillation circuit (which stops oscillation at reset to put the microcontroller in a temporary stop state). Conditions required to enable these functions and their operations will be described next.

★ **Caution When designing an application circuit which requires high reliability, do not design a reset function which depends only on a built-in power-on/power-down reset function. Be sure to design a circuit to which an external $\overline{\text{RESET}}$ signal can be input.**

18.3.1 Conditions Required to Enable the Power-On Reset Function

This function is effective when used together with the power-down reset function.

The following conditions are required to validate the power-on reset function:

The power voltage must be 4.5 to 5.5 V during normal operation, including the standby state.
$ The power-down reset function must be enabled during normal operation, including the standby state.
% The power voltage must rise from 0 V to the specified voltage.
& The time it takes for the power voltage to rise from 0 to 2.7 V must be long enough for stable oscillation to be counted in the timer. This takes about 32 ms with f_{CC} being 2 MHz, which is equivalent to 256×256 pulses of the system clock.

Cautions 1. If the above conditions are not satisfied, the power-on reset function will not operate effectively. In this case, an external reset circuit needs to be added.

2. In the standby state, even if the power-down reset function operates normally, general-purpose data memory (except for DBF) retains data up to V_{DD} = 2.7 V. If, however, data is changed due to an external error, the data in memory is not guaranteed.

18.3.2 Description and Operation of the Power-On Reset Function

The power-on reset function resets the microcontroller when it detects that power was turned on in the hardware, regardless of the software state.

The power-on reset circuit operates under a lower voltage than the other internal circuits in the μPD17120. It initializes the microcontroller regardless whether the oscillation circuit is operating. When the reset operation is terminated, the timer counts the number of oscillation pulses sent from the oscillator until it reaches the specified value. Within this period, oscillation becomes stable and the power voltage applied to the microcontroller enters the range (V_{DD} = 2.7 to 5.5 V) in which the microcontroller is guaranteed to operate.

When this period elapses, the microcontroller enters normal operation mode. Fig. 18-3 shows an example of the power-on reset operation.

Operation of the power-on reset circuit

This circuit always monitors the voltage applied to the V_{DD} pin.

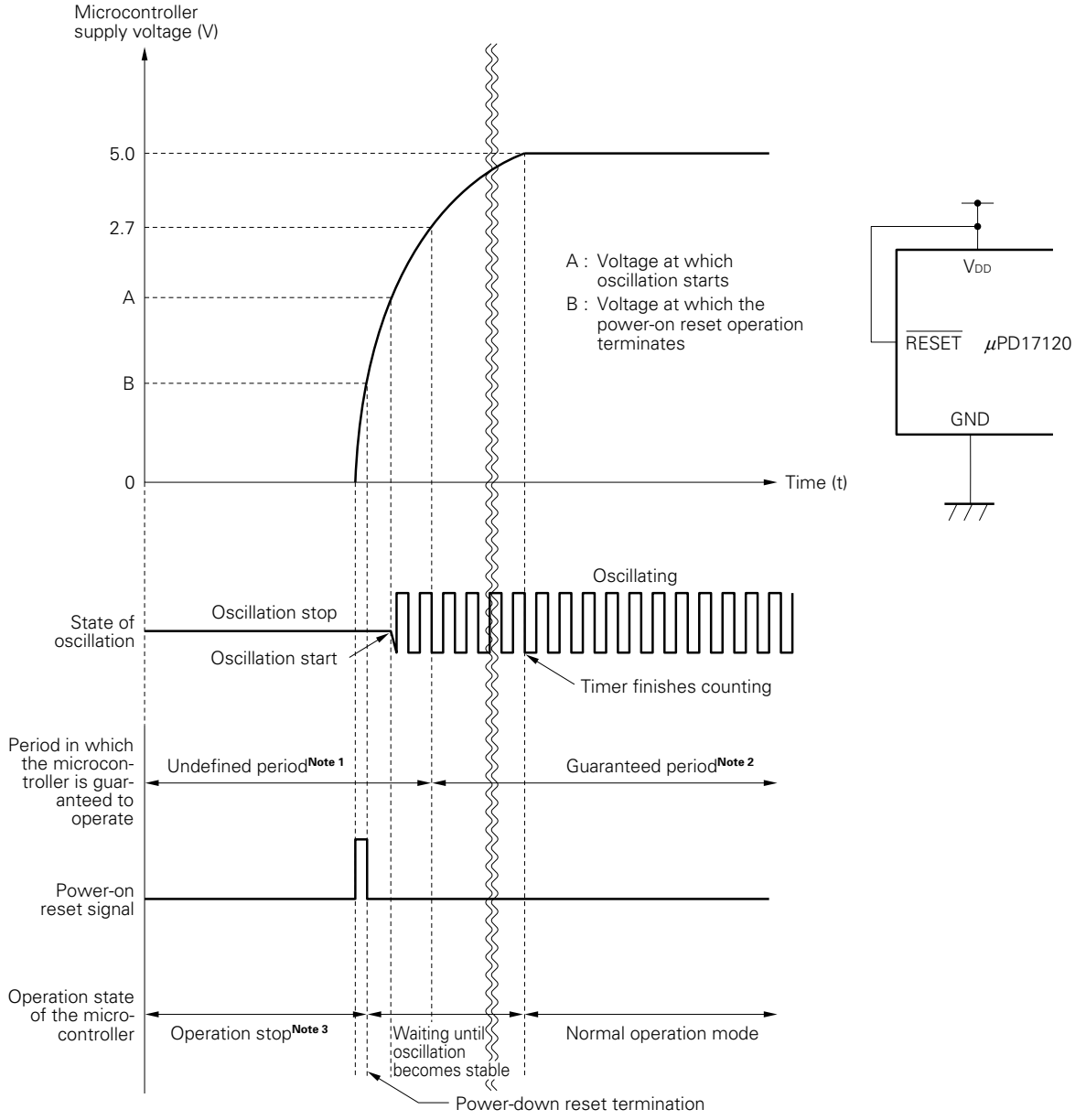
$ This circuit resets[Note] the microcontroller until power reaches a particular voltage (typically 1.5 V), regardless whether the oscillation circuit is operating.

% This circuit stops oscillation during the reset operation.

& When reset is terminated, the timer counts oscillation pulses. The microcontroller waits until oscillation becomes stable and the power voltage becomes V_{DD} = 2.7 V or higher.

Note The power-on reset circuit resets the microcontroller when the power voltage reaches the voltage at which the internal circuit can operate, namely an internal reset signal can be accepted. ★

Fig. 18-3 Example of the Power-On Reset Operation



Notes 1. During the operation-undefined period, not all of the operations specified for the μPD17120 are guaranteed. The power-on reset functions even in this period.

2. The operation-guaranteed period refers to the time in which all the operations specified for the μPD17120 are guaranteed.

3. An operation stop state refers to the state in which all of the functions of the microcontroller are stopped.

18.3.3 Condition Required for Use of the Power-Down Reset Function

The power-down reset function can be enabled or disabled using software. The following condition is required to use this function:

- The power voltage must be 4.5 to 5.5 V during normal operation, including the standby state.

Caution When the microcontroller is used with a power voltage of 2.7 to 4.5 V, add an external reset circuit instead of using the internal power-down reset circuit. If the internal power-down reset circuit is used with a power voltage of 2.7 to 4.5 V, reset operation may not terminate.

18.3.4 Description and Operation of the Power-Down Reset Function

This function is enabled by setting the power-down reset enable flag (PDRESEN) using software.

When this function detects a power voltage drop, it issues the reset signal to the microcontroller. It then initializes the microcontroller. Stopping oscillation during reset prevents the power voltage in the microcontroller from fluctuating out of control. When the specified power voltage recovers and the power-down reset operation is terminated, the microcontroller waits the time required for stable oscillation using the timer. The microcontroller then enters normal operation (starts from the top of memory).

Fig. 18-4 shows an example of the power-down operation. Fig. 18-5 shows an example of reset operation during the period from power-down reset to power recovery.

Operation of the power-down reset circuit

This circuit always monitors the voltage applied to the V_{DD} pin.

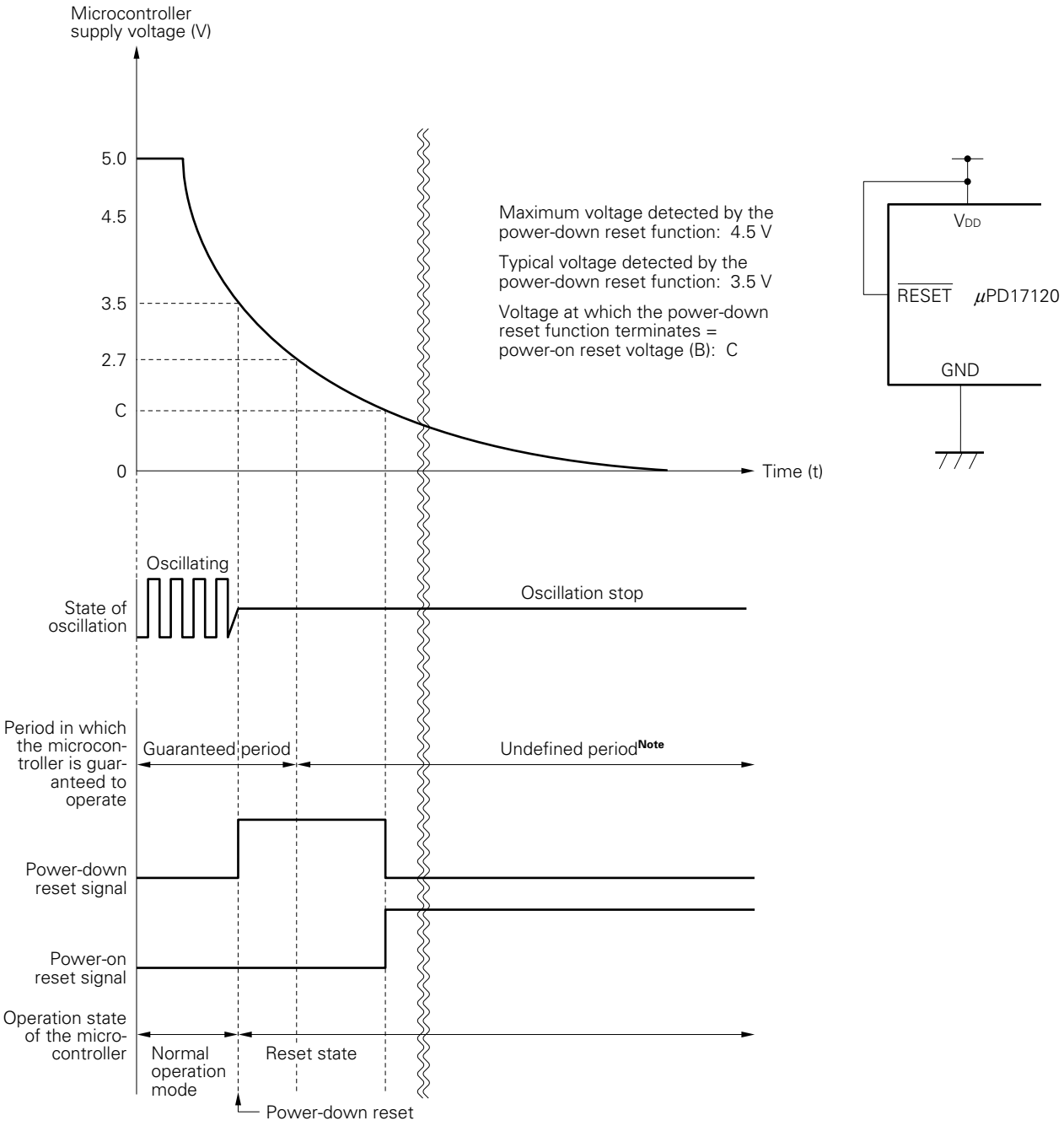
$ When this circuit detects a power voltage drop, it issues a reset signal to the other parts of the microcontroller. It continues to send this reset signal until the power voltage recovers or all the functions in the microcontroller stop.

% This circuit stops oscillation during the reset operation to prevent software crashes.
When the power voltage recovers to the low-voltage detection level (typically 3.5 V, 4.5 V maximum) before the power-down reset function stops, the microcontroller waits the time required for stable oscillation using the timer, then enters normal operation mode.

& When the power voltage recovers from 0 V, the power-on reset function has priority.

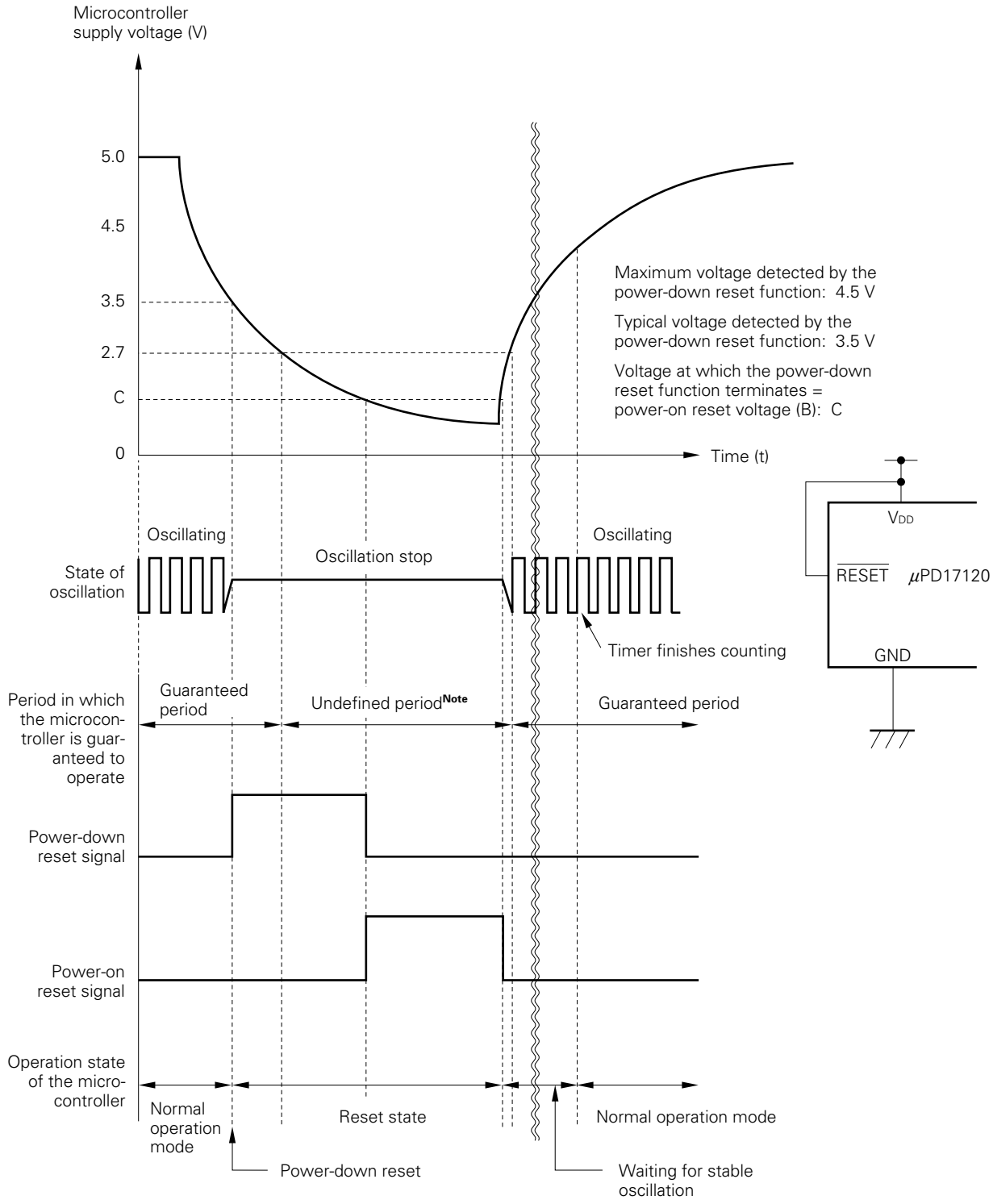
(After the power-down reset function stops and the power voltage recovers before it reaches 0 V, the microcontroller waits using the timer until oscillation becomes stable and the power voltage (V_{DD}) reaches 2.7 V. The microcontroller then enters normal operation mode.

Fig. 18-4 Example of the Power-Down Reset Operation



Note During the operation-undefined period, not all the operations specified for the μPD17120 are not guaranteed. Even in this period, however, the power-down reset functions and continues to issue a reset signal until all the functions in the microcontroller stop .

Fig. 18-5 Example of Reset Operation during the Period from Power-Down Reset to Power Recovery



Note During the operation-undefined period, not all the operations specified for the μPD17120 are not guaranteed. Even in this period, however, the power-down reset functions and continues to issue a reset signal until all the functions in the microcontroller stop .

19. INSTRUCTION SET

19.1 LEGEND

AR	:	Address register
ASR	:	Address stack register pointed to by the stack pointer
addr	:	Program memory address (11 bits, one high-order bit is always 0.)
BANK	:	Bank register
CMP	:	Compare flag
CY	:	Carry flag
DBF	:	Data buffer
h	:	HALT release condition
INTEF	:	Interrupt enable flag
INTR	:	Register automatically saved in the stack when an interrupt occurs
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 and m_C
m_R	:	Data memory row address (high-order)
m_C	:	Data memory column address (low-order)
n	:	Bit position (four bits)
n4	:	Immediate data (four bits)
PC	:	Program counter
p	:	Peripheral address
p_H	:	Peripheral address (three high-order bits)
p_L	:	Peripheral address (four low-order bits)
r	:	General register column address
rf	:	Register file address
rf_R	:	Register file row address (three high-order bits)
rf_C	:	Register file column address (four low-order bits)
SP	:	Stack pointer
s	:	STOP release condition
WR	:	Window register
(×)	:	Contents of ×

19.2 LIST OF THE INSTRUCTION SET

Instruction set	Mnemonic	Operand	Operation	Instruction code			
				Op code	Operand		
Add	ADD	r, m	(r) ← (r) + (m)	00000	m_R	mc	r
		m, #n4	(m) ← (m) + n4	10000	m_R	mc	n4
	ADDC	r, m	(r) ← (r) + (m) + CY	00010	m_R	mc	r
		m, #n4	(m) ← (m) + n4 + CY	10010	m_R	mc	n4
	INC	AR	AR ← AR + 1	00111	000	1001	0000
		IX	IX ← IX + 1	00111	000	1000	0000
Subtract	SUB	r, m	(r) ← (r) – (m)	00001	m_R	mc	r
		m, #n4	(m) ← (m) – n4	10001	m_R	mc	n4
	SUBC	r, m	(r) ← (r) – (m) – CY	00011	m_R	mc	r
		m, #n4	(m) ← (m) – n4 – CY	10011	m_R	mc	n4
Logical operation	OR	r, m	(r) ← (r) ∨ (m)	00110	m_R	mc	r
		m, #n4	(m) ← (m) ∨ n4	10110	m_R	mc	n4
	AND	r, m	(r) ← (r) ∧ (m)	00100	m_R	mc	r
		m, #n4	(m) ← (m) ∧ n4	10100	m_R	mc	n4
	XOR	r, m	(r) ← (r) ∀ (m)	00101	m_R	mc	r
		m, #n4	(m) ← (m) ∀ n4	10101	m_R	mc	n4
Test	SKT	m, #n	CMP ← 0, if (m) ∧ n = n, then skip	11110	m_R	mc	n
	SKF	m, #n	CMP ← 0, if (m) ∧ n = 0, then skip	11111	m_R	mc	n
Compare	SKE	m, #n4	(m) – n4, skip if zero	01001	m_R	mc	n4
	SKNE	m, #n4	(m) – n4, skip if not zero	01011	m_R	mc	n4
	SKGE	m, #n4	(m) – n4, skip if not borrow	11001	m_R	mc	n4
	SKLT	m, #n4	(m) – n4, skip if borrow	11011	m_R	mc	n4
Rotation	RORC	r	→ CY → $(r)_{b3}$ → $(r)_{b2}$ → $(r)_{b1}$ → $(r)_{b0}$	00111	000	0111	r
Transfer	LD	r, m	(r) ← (m)	01000	m_R	mc	r
	ST	m, r	(m) ← (r)	11000	m_R	mc	r
	MOV	@r, m	if MPE = 1: (MP, (r)) ← (m) if MPE = 0: (BANK, m_R, (r)) ← (m)	01010	m_R	mc	r
		m, @r	if MPE = 1: (m) ← (MP, (r)) if MPE = 0: (m) ← (BANK, m_R, (r))	11010	m_R	mc	r
		m, #n4	(m) ← n4	11101	m_R	mc	n4
★	MOVT[Note]	DBF, @AR	SP ← SP – 1, ASR ← PC, PC ← AR, DBF ← (PC), PC ← ASR, SP ← SP + 1	00111	000	0001	0000

★ **Note** Exceptionally, two instruction cycles are required to execute the MOVT instruction.

Instruction set	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	PC ← addr	01100	addr		
		@AR	PC ← AR	00111	000	0100	0000
Sub-routine	CALL	addr	SP ← SP – 1, ASR ← PC, PC ← addr	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
Others	STOP	s	STOP	00111	010	1111	s
	HALT	h	HALT	00111	011	1111	h
	NOP		No operation	00111	100	1111	0000

19.3 ASSEMBLER (AS17K) BUILT-IN MACRO INSTRUCTIONS ★

Legend

flag n: FLG symbol

< > : Characters enclosed in < > can be omitted.

	Mnemonic	Operand	Operation	n
Built-in macro	SKTn	flag 1, ···flag n	if (flag 1) - (flag n) = all "1", then skip	1 ≤ n ≤ 4
	SKFn	flag 1, ···flag n	if (flag 1) - (flag n) = all "0", then skip	1 ≤ n ≤ 4
	SETn	flag 1, ···flag n	(flag 1) - (flag n) ← 1	1 ≤ n ≤ 4
	CLRn	flag 1, ···flag n	(flag 1) - (flag n) ← 0	1 ≤ n ≤ 4
	NOTn	flag 1, ···flag n	if (flag n) = "0", then (flag n) ← 1 if (flag n) = "1", then (flag n) ← 0	1 ≤ n ≤ 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 ≤ n ≤ 4
	BANKn		(BANK) ← n	n = 0

20. ASSEMBLER RESERVED WORDS

20.1 MASK OPTION PSEUDO INSTRUCTIONS

To create μPD17120 programs, it is necessary to specify whether pins that can have pull-up resistors have pull-up resistors. This is done in the assembler source program using mask option pseudo instructions. To set the mask option, note that D17120.OPT file in the AS17120 (μPD17120 device file) must be in the current directory at assembly time.

Specify mask options for the following pins:

- $\overline{\text{RESET}}$ pin
- Port 0D ($P0D_3$, $P0D_2$, $P0D_1$, $P0D_0$)
- Port 0E ($P0E_1$, $P0E_0$)

20.1.1 OPTION and ENDOP Pseudo Instructions

The block from the OPTION pseudo instruction to the ENDOP pseudo instruction is defined as the option definition block.

The format for the mask option definition block is shown below. Only the three pseudo instructions listed in Table 20-1 can be described in this block.

Format:

Symbol	Mnemonic	Operand	Comment
[label:]	OPTION		[;comment]
	•		
	•		
	•		
	•		
	ENDOP		

20.1.2 Mask Option Definition Pseudo Instructions

Table 20-1 lists the pseudo instructions which define the mask options for each pin.

Table 20-1 Mask Option Definition Pseudo Instructions

Pin	Mask option pseudo instruction	Number of operands	Parameter name
$\overline{\text{RESET}}$	OPTRES	1	OPEN (without pull-up resistor) PULLUP (with pull-up resistor)
$P0D_3$-$P0D_0$	OPTP0D	4	OPEN (without pull-up resistor) PULLUP (with pull-up resistor)
$P0E_1$, $P0E_0$	OPTP0E	2	OPEN (without pull-up resistor) PULLUP (with pull-up resistor)

The OPTRES format is shown below. Specify the $\overline{\text{RESET}}$ mask option in the operand field.

Symbol	Mnemonic	Operand	Comment
[label:]	OPTRES	($\overline{\text{RESET}}$)	[;comment]

The OPTP0D format is shown below. Specify mask options for all pins of port 0D. Specify the pins in the operand field starting at the first operand in the order $P0D_3$, $P0D_2$, $P0D_1$, then $P0D_0$.

Symbol	Mnemonic	Operand	Comment
[label:]	OPTP0D	($P0D_3$),($P0D_2$),($P0D_1$),($P0D_0$)	[;comment]

The OPTP0E format is shown below. Specify mask options for all pins of port 0E. Specify the pins in the operand field starting at the first operand in the order $P0E_1$, then $P0E_0$.

Symbol	Mnemonic	Operand	Comment
[label:]	OPTP0E	($P0E_1$),($P0E_0$)	[;comment]

Example of describing mask options

$\overline{\text{RESET}}$ pin: Pull-up

$P0D_3$: Open, $P0D_2$: Open, $P0D_1$: Pull-up, $P0D_0$: Pull-up,

$P0E_1$: Pull-up, $P0E_0$: Open

```
Symbol              Mnemonic    Operand                    Comment
; μPD17120
Setting mask options:  OPTION
                    ;
                       OPTRES      PULLUP
                       OPTP0D      OPEN,OPEN,PULLUP,PULLUP
                       OPTP0E      PULLUP,OPEN
                    ;
                       ENDOP
```

20.2 RESERVED SYMBOLS

The reserved symbols defined in the μPD17120 device file (AS17120) are listed below.

System register (SYSREG)

Symbolic name	Attribute	Value	Read/ write	Description
AR3	MEM	0.74H	R	Bits b15 to b12 of the address register
AR2	MEM	0.75H	R/W	Bits b11 to b8 of the address register
AR1	MEM	0.76H	R/W	Bits b7 to b4 of the address register
AR0	MEM	0.77H	R/W	Bits b3 to b0 of the address register
WR	MEM	0.78H	R/W	Window register
BANK	MEM	0.79H	R/W	Bank register
IXH	MEM	0.7AH	R/W	Index register high
MPH	MEM	0.7AH	R/W	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	R/W	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

Data buffer (DBF)

Symbolic name	Attribute	Value	Read/ write	Description
DBF3	MEM	0.0CH	R/W	DBF bits b15 to b12
DBF2	MEM	0.0DH	R/W	DBF bits b11 to b8
DBF1	MEM	0.0EH	R/W	DBF bits b7 to b4
DBF0	MEM	0.0FH	R/W	DBF bits b3 to b0

Port register

Symbolic name	Attribute	Value	Read/ write	Description
P0E1	FLG	0.6FH.1	R/W	Port 0E bit b1
P0E0	FLG	0.6FH.0	R/W	Port 0E bit b0
P0A3	FLG	0.70H.3	R/W	Port 0A bit b3
P0A2	FLG	0.70H.2	R/W	Port 0A bit b2
P0A1	FLG	0.70H.1	R/W	Port 0A bit b1
P0A0	FLG	0.70H.0	R/W	Port 0A bit b0
P0B3	FLG	0.71H.3	R/W	Port 0B bit b3
P0B2	FLG	0.71H.2	R/W	Port 0B bit b2
P0B1	FLG	0.71H.1	R/W	Port 0B bit b1
P0B0	FLG	0.71H.0	R/W	Port 0B bit b0
P0C3	FLG	0.72H.3	R/W	Port 0C bit b3
P0C2	FLG	0.72H.2	R/W	Port 0C bit b2
P0C1	FLG	0.72H.1	R/W	Port 0C bit b1
P0C0	FLG	0.72H.0	R/W	Port 0C bit b0
P0D3	FLG	0.73H.3	R/W	Port 0D bit b3
P0D2	FLG	0.73H.2	R/W	Port 0D bit b2
P0D1	FLG	0.73H.1	R/W	Port 0D bit b1
P0D0	FLG	0.73H.0	R/W	Port 0D bit b0

Register file (control register)

(1/2)

Symbolic name	Attribute	Value	Read/ write	Description
SP	MEM	0.81H	R/W	Stack pointer
SIOEN	FLG	0.8AH.0	R/W	SIO enable flag
INT	FLG	0.8FH.0	R	INT pin status flag
PDRESEN	FLG	0.90H.0	R/W	Power-down reset enable flag
TMEN	FLG	0.91H.3	R/W	Timer enable flag
TMRES	FLG	0.91H.2	R/W	Timer reset flag
TMCK1	FLG	0.91H.1	R/W	Timer source count pulse flag bit 1
TMCK0	FLG	0.91H.0	R/W	Timer source count pulse flag bit 0
TMOSEL	FLG	0.92H.0	R/W	$P0D_3/\overline{TMOUT}$ selection flag
SIOTS	FLG	0.9AH.3	R/W	SIO start flag
SIOHIZ	FLG	0.9AH.2	R/W	SO pin state
SIOCK1	FLG	0.9AH.1	R/W	Serial clock selection flag bit 1
SIOCK0	FLG	0.9AH.0	R/W	Serial clock selection flag bit 0
IEGMD1	FLG	0.9FH.1	R/W	INT pin edge detection selection flag bit 1
IEGMD0	FLG	0.9FH.0	R/W	INT pin edge detection selection flag bit 0
P0BGIO	FLG	0.A4H.0	R/W	P0B group input/output selection flag (1 = all P0Bs are output ports.)

Register file (control register)

(2/2)

Symbolic name	Attribute	Value	Read/ write	Description
IPSIO	FLG	0.AFH.2	R/W	SIO interrupt enable flag
IPTM	FLG	0.AFH.1	R/W	Timer interrupt enable flag
IP	FLG	0.AFH.0	R/W	INT pin interrupt enable flag
P0EBIO1	FLG	0.B2H.1	R/W	$P0E_1$ input/output selection flag (1 = output port)
P0EBIO0	FLG	0.B2H.0	R/W	$P0E_0$ input/output selection flag (1 = output port)
P0DBIO3	FLG	0.B3H.3	R/W	$P0D_3$ input/output selection flag (1 = output port)
P0DBIO2	FLG	0.B3H.2	R/W	$P0D_2$ input/output selection flag (1 = output port)
P0DBIO1	FLG	0.B3H.1	R/W	$P0D_1$ input/output selection flag (1 = output port)
P0DBIO0	FLG	0.B3H.0	R/W	$P0D_0$ input/output selection flag (1 = output port)
P0CBIO3	FLG	0.B4H.3	R/W	$P0C_3$ input/output selection flag (1 = output port)
P0CBIO2	FLG	0.B4H.2	R/W	$P0C_2$ input/output selection flag (1 = output port)
P0CBIO1	FLG	0.B4H.1	R/W	$P0C_1$ input/output selection flag (1 = output port)
P0CBIO0	FLG	0.B4H.0	R/W	$P0C_0$ input/output selection flag (1 = output port)
P0ABIO3	FLG	0.B5H.3	R/W	$P0A_3$ input/output selection flag (1 = output port)
P0ABIO2	FLG	0.B5H.2	R/W	$P0A_2$ input/output selection flag (1 = output port)
P0ABIO1	FLG	0.B5H.1	R/W	$P0A_1$ input/output selection flag (1 = output port)
P0ABIO0	FLG	0.B5H.0	R/W	$P0A_0$ input/output selection flag (1 = output port)
IRQSIO	FLG	0.BDH.0	R/W	SIO interrupt request flag
IRQTM	FLG	0.BEH.0	R/W	Timer interrupt request flag
IRQ	FLG	0.BFH.0	R/W	INT pin interrupt request flag

Peripheral hardware register

Symbolic name	Attribute	Value	Read/ write	Description
SIOSFR	DAT	01H	R/W	Peripheral address of the shift register
TMC	DAT	02H	R	Peripheral address of the timer counter register
TMM	DAT	03H	W	Peripheral address of the timer modulo register
AR	DAT	40H	R/W	Peripheral address of the address register for GET, PUT, PUSH, CALL, BR, MOVT, and INC instructions

★ **Others**

Symbolic name	Attribute	Value	Description
DBF	DAT	0FH	Fixed operand value for a PUT/GET/MOVT instruction
IX	DAT	01H	Fixed operand value for an INC instruction

[MEMO]

Fig. 20-1 Control Register Configuration (1/2)

Row address	Item	Column address 0	1	2	3	4	5	6	7
0 (8)	Symbol		0 SP						
	When reset		0 1 0 1						
	Read/ Write		R/W						
1 (9)	Symbol	0 0 0 PDRESEN	TMEN TMRES TMCK1 TMCK0	0 0 0 TMOSEL					
	When reset	0 0 0 0	1 0 0 0	0 0 0 0					
	Read/ Write	R/W	R/W	R/W					
2 (A)	Symbol					0 0 0 P0BGIO			
	When reset					0 0 0 0			
	Read/ Write					R/W			
3 (B)	Symbol			0 0 P0EBIO1 P0EBIO0	P0DBIO3 P0DBIO2 P0DBIO1 P0DBIO0	P0CBIO3 P0CBIO2 P0CBIO1 P0CBIO0	P0ABIO3 P0ABIO2 P0ABIO1 P0ABIO0		
	When reset			0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0		
	Read/ Write			R/W	R/W	R/W	R/W		

Remark The address enclosed in parentheses apply when the AS17K assembler is used.
The names of all the flags in the control registers are assembler reserved words saved in the device file. Using these reserved words is useful in programming.

Fig. 20-1 Control Register Configuration (2/2)

8	9	A	B	C	D	E	F
		0 0 0 SIOEN					0 0 0 INT
		0 0 0 0					0 0 0 Note
		R/W					R
		SIOTS SIOHIZ SIOCK1 SIOCK0					0 0 IEGMD1 IEGMD0
		0 0 0 0					0 0 0 0
		R/W					R/W
							0 IPSIO IPTM IP
							0 0 0 0
							R/W
					0 0 0 IRQSIO	0 0 0 IRQTM	0 0 0 IRQ
					0 0 0 0	0 0 0 1	0 0 0 0
					R/W	R/W	R/W

Note The INT flag depends on the status of the INT pin. ★

21. ELECTRICAL CHARACTERISTICS

Absolute maximum ratings (T_A = 25 °C)

Parameter	Symbol	Conditions		Rated value	Unit
Supply voltage	V_{DD}			–0.3 to +7.0	V
Input voltage	V_I	P0A, P0B, P0C, INT, $\overline{RESET}$		–0.3 to V_{DD} + 0.3	V
		P0D, P0E	When a built-in pull-up resistor is connected.	–0.3 to V_{DD} + 0.3	V
			When a built-in pull-up resistor is not connected.	–0.3 to +10.0	
Output voltage	V_O	P0A, P0B, P0C		–0.3 to V_{DD} + 0.3	V
		P0D, P0E	When a built-in pull-up resistor is connected.	–0.3 to V_{DD} + 0.3	V
			When a built-in pull-up resistor is not connected.	–0.3 to +10.0	
High-level output current	I_{OH}	Each of P0A, P0B, and P0C		-5	mA
		Total of all output pins		-20	mA
Low-level output current	I_{OL}	Each of P0A, P0B, and P0C		5	mA
		Each of P0D and P0E		30	mA
		Total of P0A, P0B, and P0C output pins		20	mA
		Total of P0D and P0E output pins		60	mA
		Total of all output pins		80	mA
Operating ambient temperature	T_A			–40 to +85	°C
Storage temperature	T_{stg}			–65 to +150	°C
★ Allowable dissipation	P_d	T_A = 85 °C	Plastic shrink DIP	155	mW
			Plastic SOP	95	mW

Caution Absolute maximum ratings are rated values beyond which some physical damages may be caused to the product; if any of the parameters in the table above exceeds its rated value even for a moment, the quality of the product may deteriorate. Be sure to use the product within the rated values.

Recommended power voltage range (T_A = –40 to +85 °C)

Parameter	Conditions	Min.	Typ.	Max.	Unit
CPU**Note**		2.7		5.5	V
Power-on/power-down reset circuit	Rising time of the power voltage (from 0 to 2.7 V): 4096 × t_{CY} or less	4.5		5.5	V

Note Excluding the power-on/power-down reset circuit

Remark t_{CY} = 16/f_{CC} (f_{CC}: frequency of system clock oscillator)

System clock oscillator characteristics (T_A = –40 to +85 °C)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
System clock oscillation frequency	f_{CC}	V_{DD} = 4.5 to 5.5 V, R_{OSC} = 10 kΩ	1.6	2	2.4	MHz
		V_{DD} = 4.5 to 5.5 V, R_{OSC} = 24 kΩ	0.8	1	1.2	MHz
		V_{DD} = 2.7 to 5.5 V, R_{OSC} = 24 kΩ	0.6	1	1.2	MHz
		V_{DD} = 2.7 to 3.3 V, R_{OSC} = 51 kΩ	400	500	600	kHz

DC characteristics (V_{DD} = 2.7 to 5.5 V, T_A = –40 to +85 °C)

Parameter	Symbol	Conditions			Min.	Typ.	Max.	Unit
High-level input voltage	V_{IH1}	P0A, P0B, P0C, P0D, P0E			0.7V_{DD}		V_{DD}	V
	V_{IH2}	$\overline{RESET}$, $\overline{SCK}$, SI, INT			0.8V_{DD}		V_{DD}	V
Low-level input voltage	V_{IL1}	P0A, P0B, P0C			0		0.3V_{DD}	V
	V_{IL2}	P0D, P0E, $\overline{RESET}$, $\overline{SCK}$, SI, INT			0		0.2V_{DD}	V
High-level output voltage	V_{OH}	P0A, P0B, P0C	V_{DD} = 4.5 to 5.5 V I_{OH} = –1.0 mA		V_{DD} - 0.3			V
			V_{DD} = 2.7 to 4.5 V I_{OH} = –0.5 mA		V_{DD} - 0.3			V
Low-level output voltage	V_{OL1}	P0A, P0B, P0C, P0D, P0E	V_{DD} = 4.5 to 5.5 V I_{OL} = 1.0 mA				0.3	V
			V_{DD} = 2.7 to 4.5 V I_{OL} = 0.5 mA				0.3	V
	V_{OL2}	P0D, P0E	V_{DD} = 4.5 to 5.5 V I_{OL} = 15 mA				1.0	V
			V_{DD} = 2.7 to 4.5 V I_{OL} = 15 mA				2.0	V
High-level input leakage current	I_{LIH}	P0A, P0B, P0C, P0D, P0E V_{IN} = V_{DD}					3	μA
Low-level input leakage current	I_{LIL}	P0A, P0B, P0C, P0D, P0E V_{IN} = 0 V					–3	μA
High-level output leakage current	I_{LOH}	P0A, P0B, P0C, P0D, P0E V_{OUT} = V_{DD}					3	μA
Low-level output leakage current	I_{LOL}	P0A, P0B, P0C, P0D, P0E V_{OUT} = 0 V					–3	μA
Built-in pull-up resistor	R_{PULL}	P0D, P0E, $\overline{RESET}$			50	100	200	kΩ
Power supply current[Note]	I_{DD1}	Operation mode	f_{CC} = 2.0 MHz	V_{DD} = 5 V ±10 %		0.8	2.0	mA
				V_{DD} = 3 V ±10 %		0.5	1.5	mA
			f_{CC} = 1.0 MHz	V_{DD} = 5 V ±10 %		0.4	1.0	mA
				V_{DD} = 3 V ±10 %		0.25	0.75	mA
			f_{CC} = 500 kHz	V_{DD} = 5 V ±10 %		250	500	μA
				V_{DD} = 3 V ±10 %		125	375	μA
	I_{DD2}	HALT mode	f_{CC} = 2.0 MHz	V_{DD} = 5 V ±10 %		0.6	1.5	mA
				V_{DD} = 3 V ±10 %		0.3	1.0	mA
			f_{CC} = 1.0 MHz	V_{DD} = 5 V ±10 %		0.3	0.8	mA
				V_{DD} = 3 V ±10 %		0.15	0.5	mA
			f_{CC} = 500 kHz	V_{DD} = 5 V ±10 %		150	300	μA
				V_{DD} = 3 V ±10 %		100	200	μA
	I_{DD3}	STOP mode	V_{DD} = 5 V ±10 %			3.0	10	μA
			V_{DD} = 3 V ±10 %			2.0	10	μA

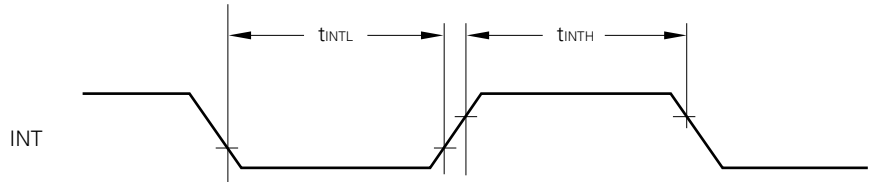
Note This current excludes the current which flows through the built-in pull-up resistor.

AC characteristics (V_{DD} = 2.7 to 5.5 V, T_A = –40 to +85 °C)

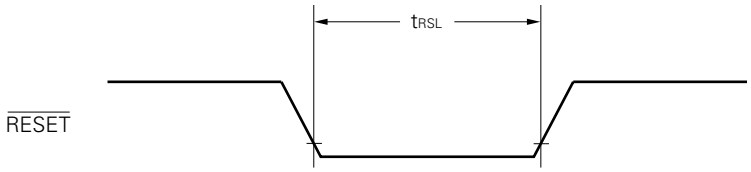
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
CPU clock cycle time (instruction execution time)	t_{CY}		6.6		41	μs
INT high/low level width (external interrupt input)	t_{INTH},	V_{DD} = 4.5 to 5.5 V	10			μs
	t_{INTL}		50			μs
$\overline{\text{RESET}}$ low level width	t_{RSL}	V_{DD} = 4.5 to 5.5 V	10			μs
			50			μs

Remark t_{CY} = 16/f_{CC} (f_{CC}: frequency of system clock oscillator)

Interrupt input timing



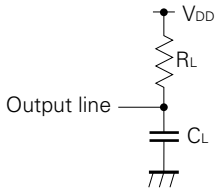
$\overline{\text{RESET}}$ input timing



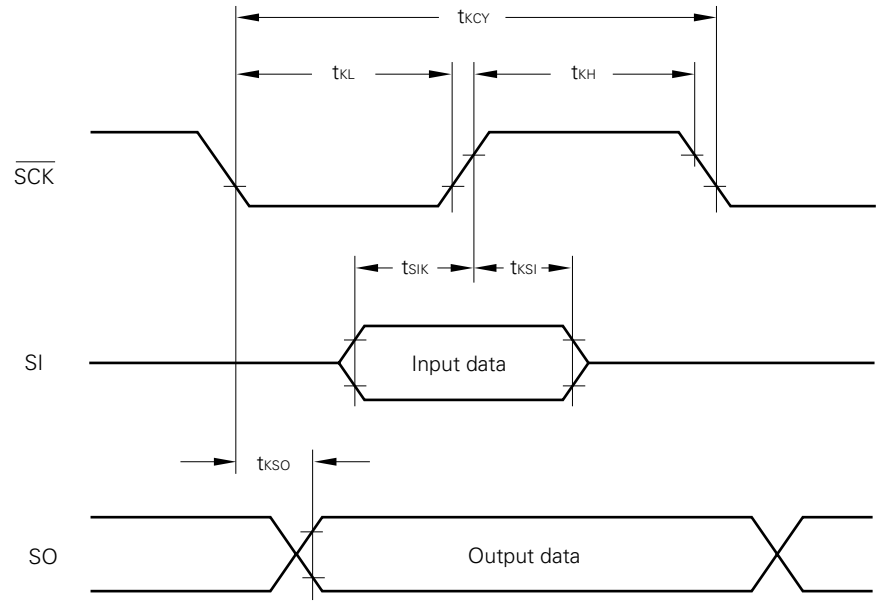
Serial transfer operation (V_{DD} = 2.7 to 5.5 V, T_A = –40 to +85 °C) ★

Parameter	Symbol		Conditions		Min.	Typ.	Max.	Unit
$\overline{SCK}$ cycle time	t_{KCY}	Input	V_{DD} = 4.5 to 5.5 V		2.0			μs
					10			μs
		Output	R_L = 1 kΩ, C_L = 100 pF	V_{DD} = 4.5 to 5.5 V	8.0			μs
					16			μs
			Built-in pull-up resistor, C_L = 100 pF	V_{DD} = 4.5 to 5.5 V	150			μs
					300			μs
$\overline{SCK}$ high/low level width	t_{KH}, t_{KL}	Input	V_{DD} = 4.5 to 5.5 V		1.0			μs
					5.0			μs
		Output	R_L = 1 kΩ, C_L = 100 pF	V_{DD} = 4.5 to 5.5 V	t_{KCY}/2-0.6			μs
					t_{KCY}/2-1.2			μs
			Built-in pull-up resistor, C_L = 100 pF	V_{DD} = 4.5 to 5.5 V	t_{KCY}/2-70			μs
					t_{KCY}/2-140			μs
SI setup time (with respect to $\overline{SCK}$↑)	t_{SIK}				100			ns
SI hold time (with respect to $\overline{SCK}$↑)	t_{KSI}				100			ns
Delay from $\overline{SCK}$↓ to SO	t_{KSO}	R_L = 1 kΩ, C_L = 100 pF		V_{DD} = 4.5 to 5.5 V			0.8	μs
							1.4	μs
		Built-in pull-up resistor, C_L = 100 pF		V_{DD} = 4.5 to 5.5 V			70	μs
							140	μs

Remark R_L and C_L are a resistive load and a capacitive load for the output line.



Serial transfer timing

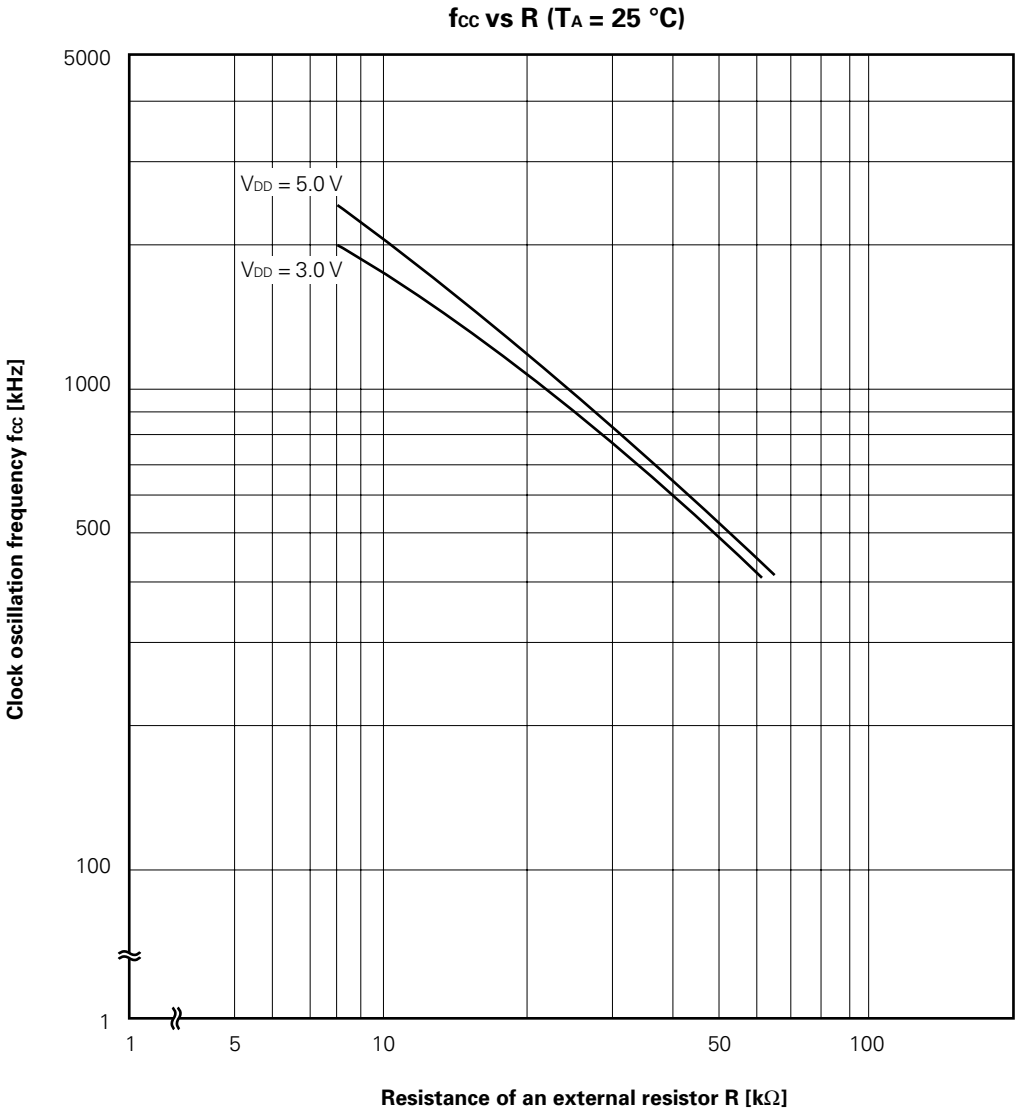


Power-on/power-down reset circuit characteristics (T_A = –40 to +85 °C)

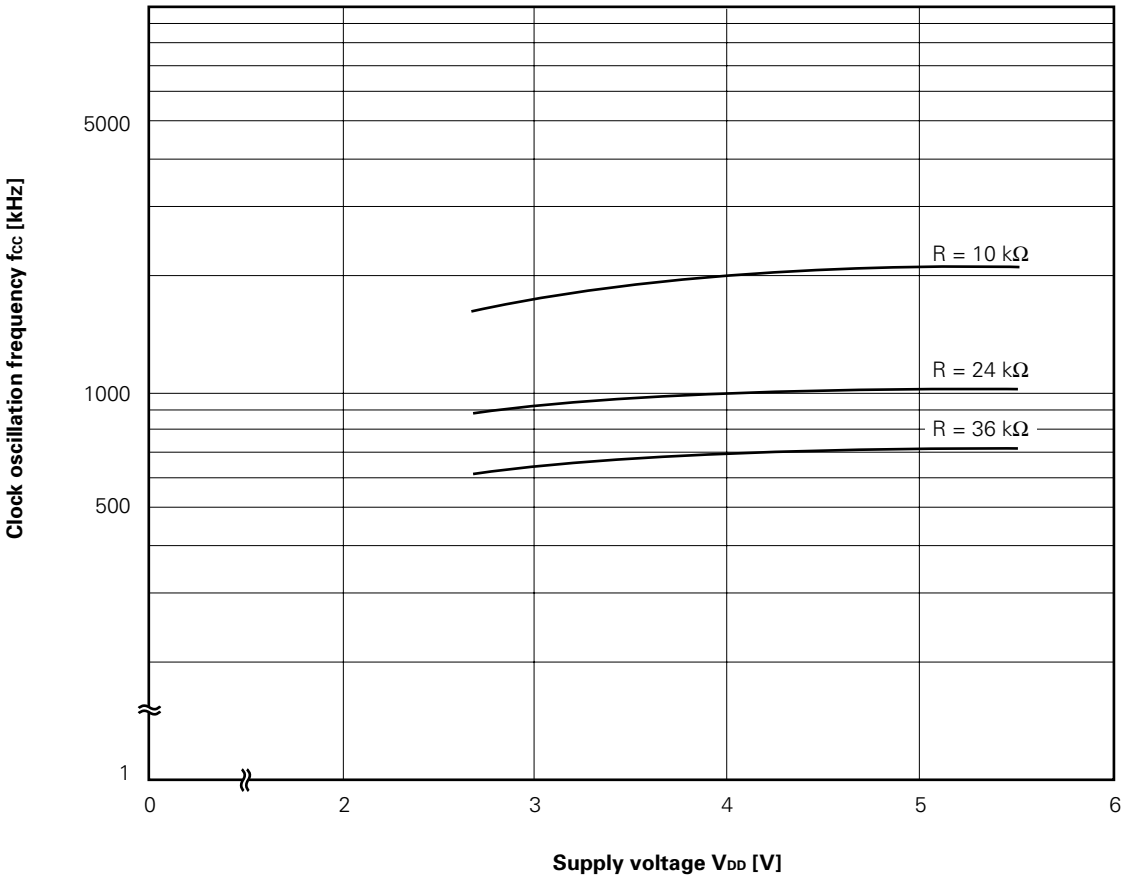
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Power voltage rise time when power-on reset is valid	t_{POR}	V_{DD} = 0 → 2.7 V Rising must start at 0 V.			4096t_{CY}	μs
Voltage for power-down reset circuit	V_{PDR}	When PDRESEN = 1		3.5	4.5	V

Remark t_{CY} = 16/f_{CC} (f_{CC}: frequency of system clock oscillator)

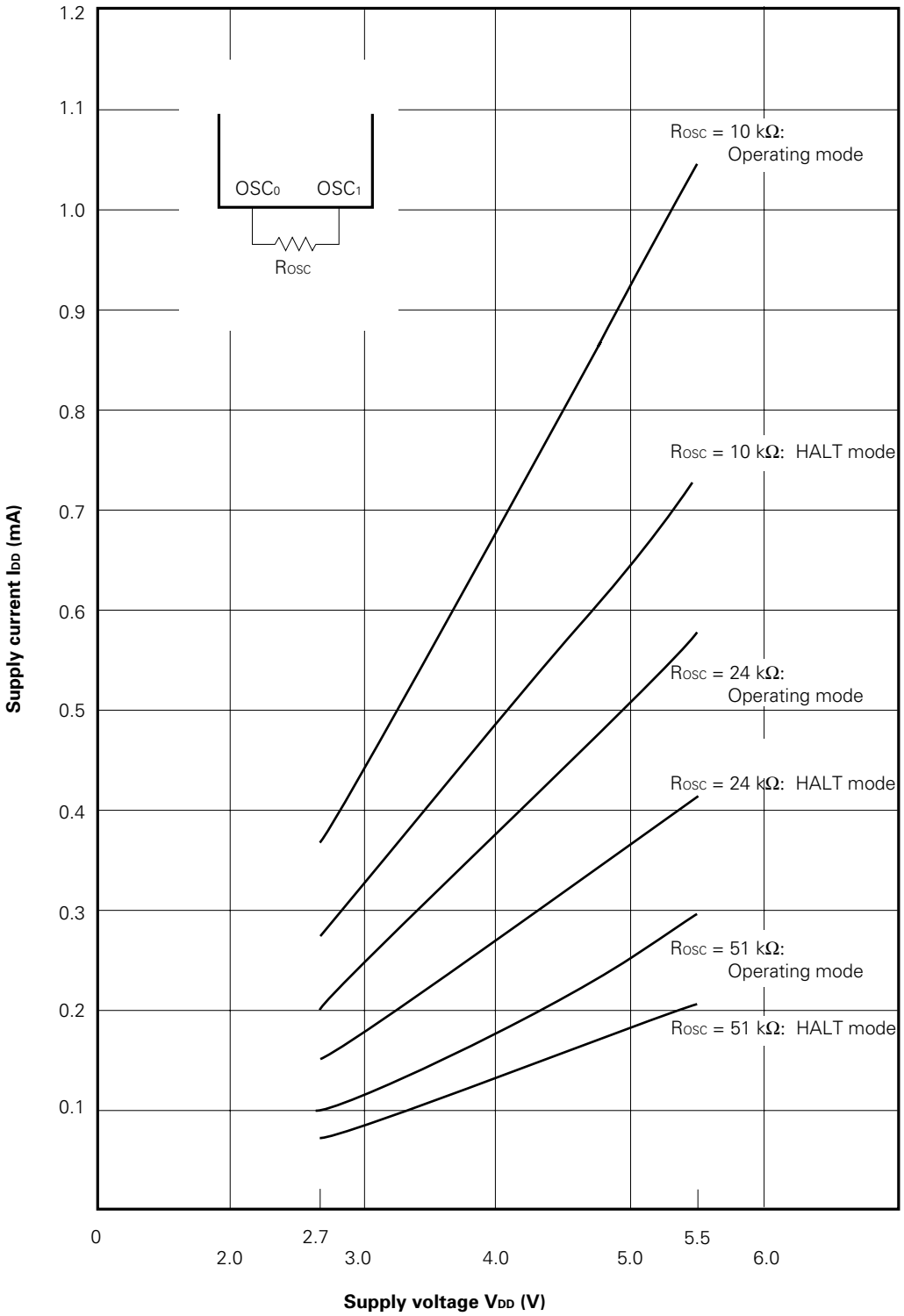
22. CHARACTERISTIC CURVES (FOR REFERENCE)

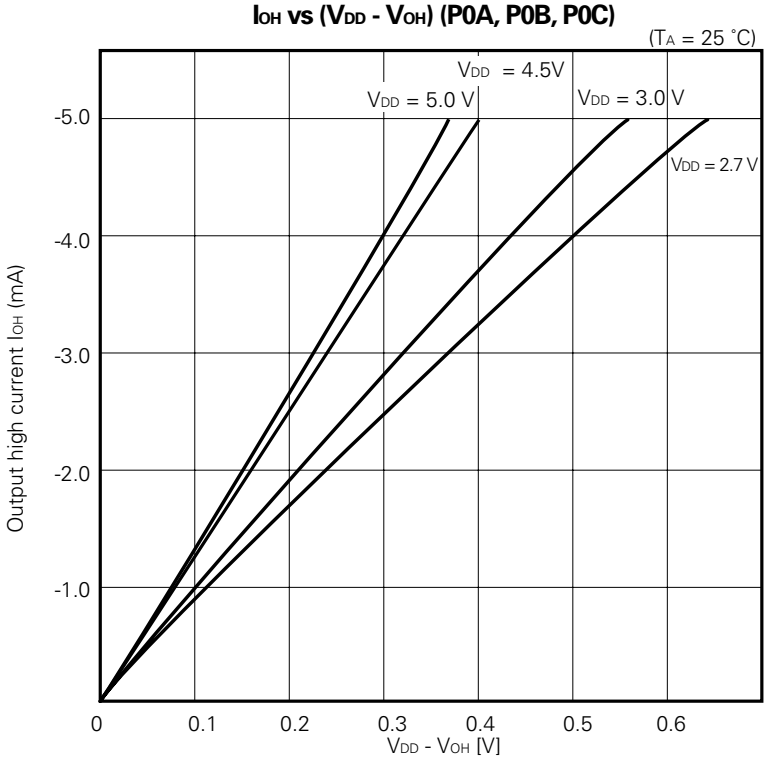


f_{CC} vs V_{DD} (T_A = –40 to +85 °C)

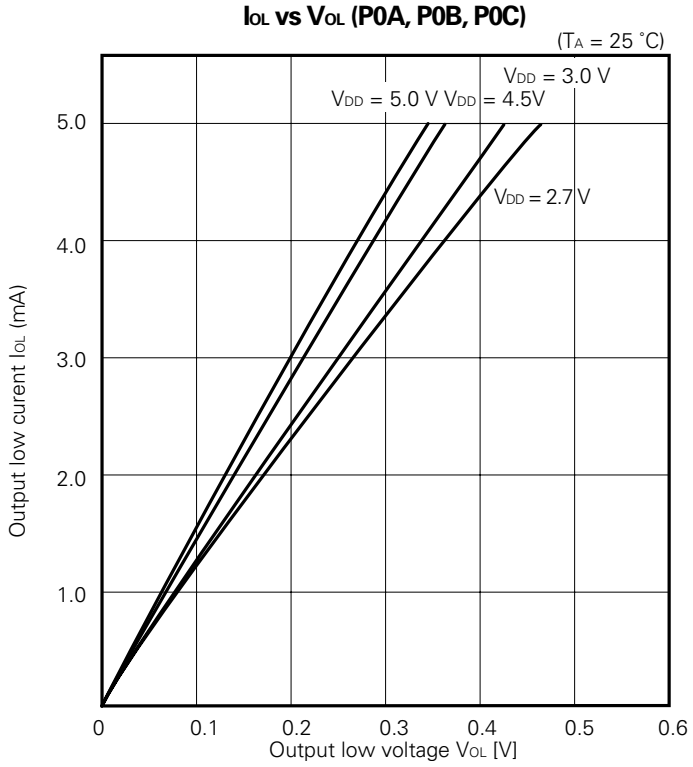


Supply Current I_{DD} (mA)

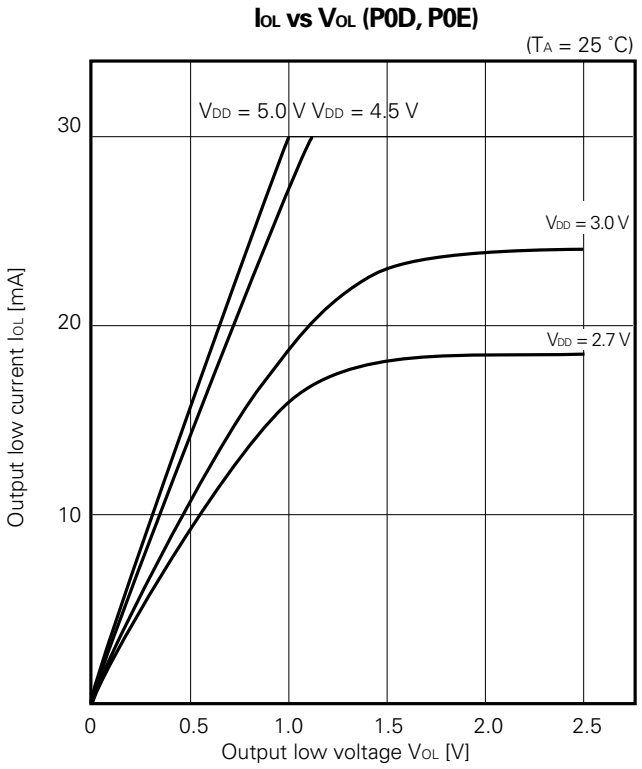




Caution Absolute maximum rating of the output current is –5 mA per pin.



Caution Absolute maximum rating of the output current is 5 mA per pin.

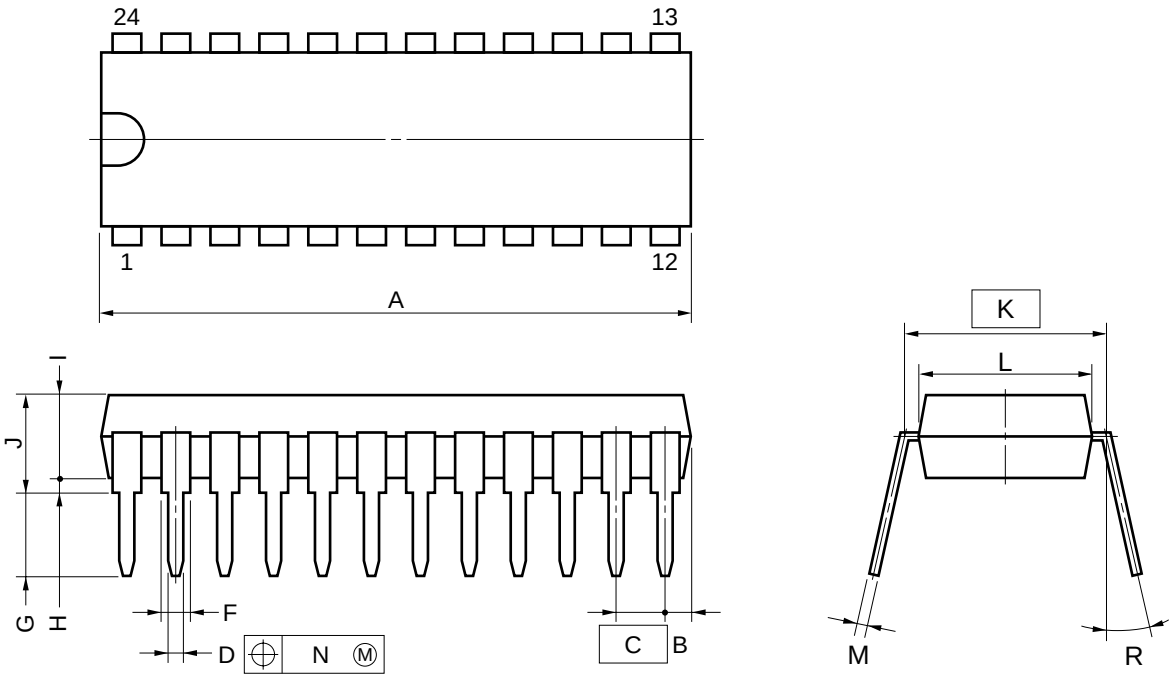


Caution Absolute maximum rating of the output current is 30 mA per pin.

23. PACKAGE DRAWINGS

PACKAGE DRAWINGS OF MASS-PRODUCED PRODUCTS (1/2)

24 PIN PLASTIC SHRINK DIP (300 mil)



NOTE

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.

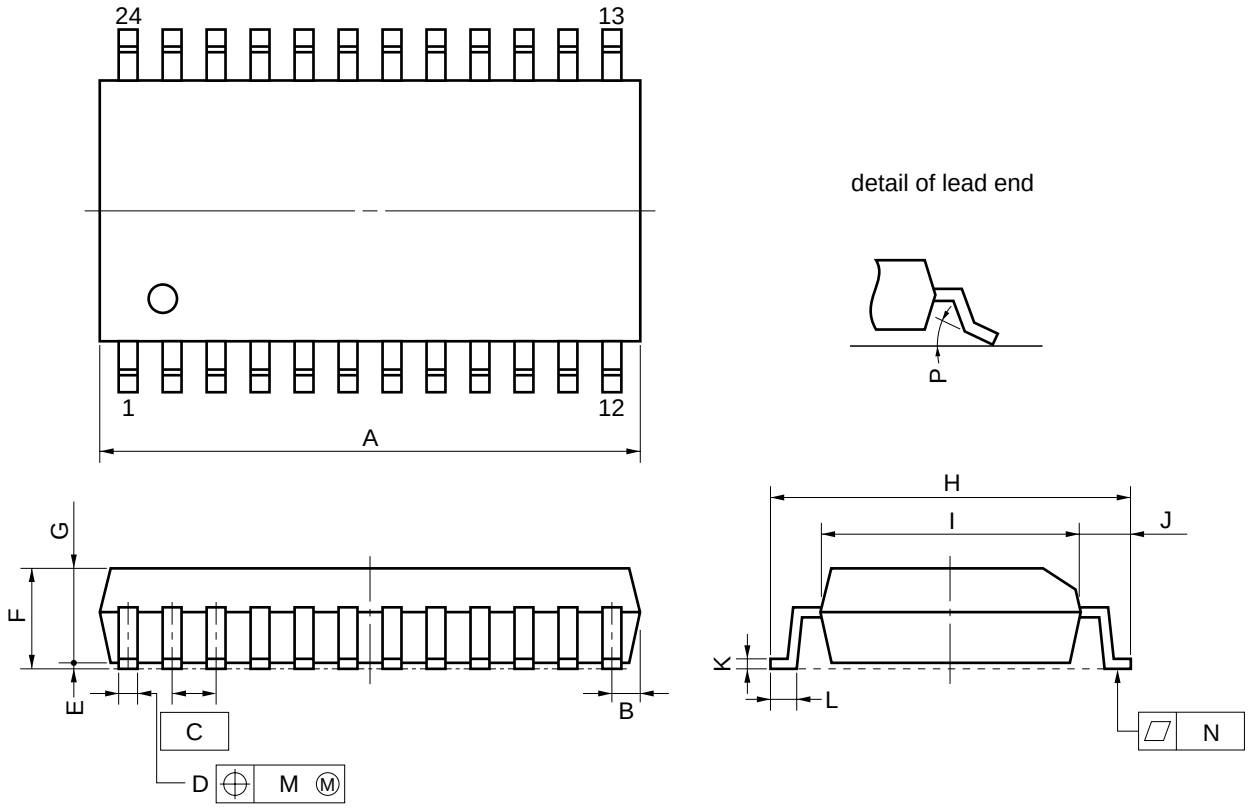
ITEM	MILLIMETERS	INCHES
A	23.12 MAX.	0.911 MAX.
B	1.78 MAX.	0.070 MAX.
C	1.778 (T.P.)	0.070 (T.P.)
D	0.50 ± 0.10	$0.020^{+0.004}_{-0.005}$
F	0.85 MIN.	0.033 MIN.
G	3.2 ± 0.3	0.126 ± 0.012
H	0.51 MIN.	0.020 MIN.
I	4.31 MAX.	0.170 MAX.
J	5.08 MAX.	0.200 MAX.
K	7.62 (T.P.)	0.300 (T.P.)
L	6.5	0.256
M	$0.25^{+0.10}_{-0.05}$	$0.010^{+0.004}_{-0.003}$
N	0.17	0.007
R	0~15°	0~15°

S24C-70-300B-1

Caution The ES is different from the corresponding mass-produced products in shape and material. See "ES PACKAGE DRAWINGS (1/2)."

PACKAGE DRAWINGS OF MASS-PRODUCED PRODUCTS (2/2)

24 PIN PLASTIC SOP (375 mil)



NOTE

Each lead centerline is located within 0.12 mm (0.005 inch) of its true position (T.P.) at maximum material condition.

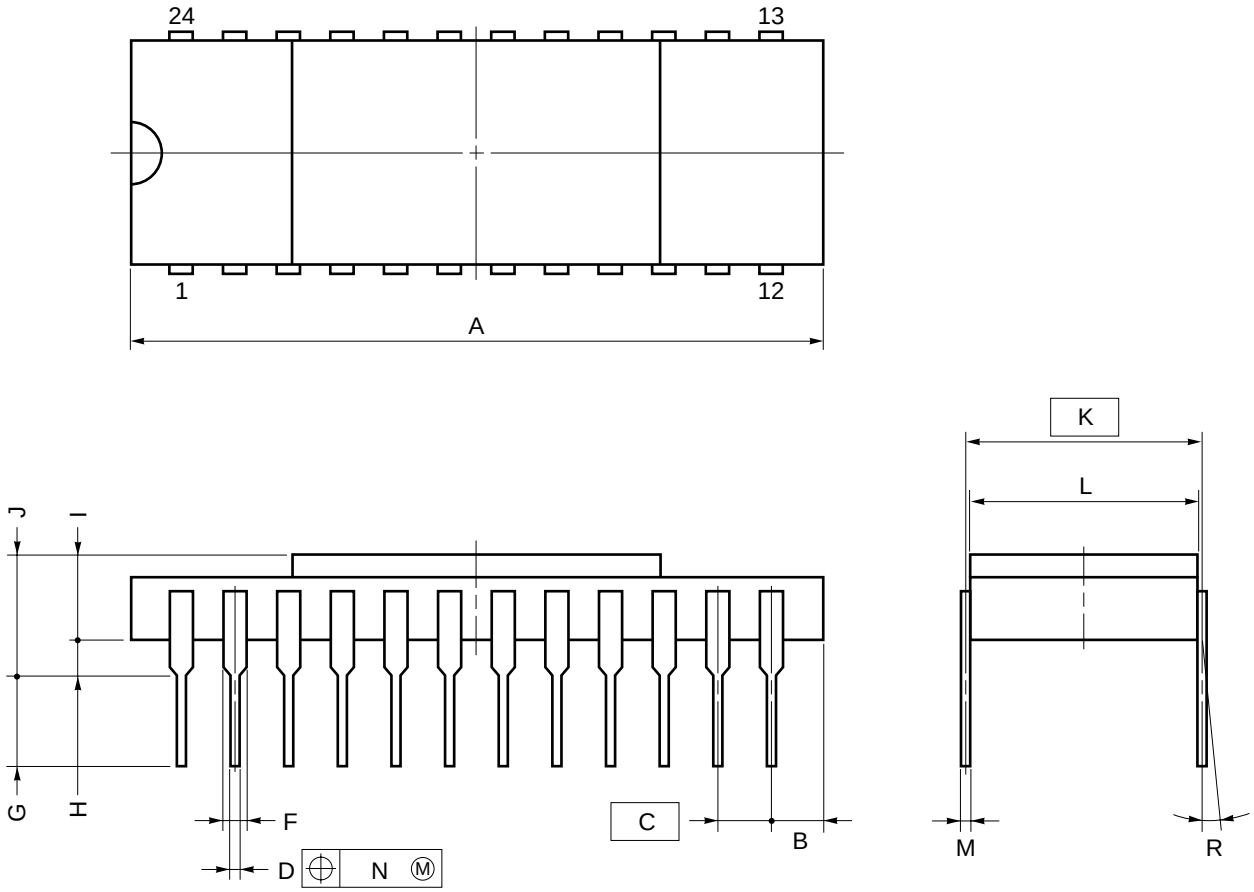
ITEM	MILLIMETERS	INCHES
A	15.54 MAX.	0.612 MAX.
B	0.78 MAX.	0.031 MAX.
C	1.27 (T.P.)	0.050 (T.P.)
D	$0.40^{+0.10}_{-0.05}$	$0.016^{+0.004}_{-0.003}$
E	0.1±0.1	0.004±0.004
F	2.9 MAX.	0.115 MAX.
G	2.50	0.098
H	10.3±0.3	$0.406^{+0.012}_{-0.013}$
I	7.2	0.283
J	1.6	0.063
K	$0.15^{+0.10}_{-0.05}$	$0.006^{+0.004}_{-0.002}$
L	0.8±0.2	$0.031^{+0.009}_{-0.008}$
M	0.12	0.005
N	0.15	0.006
P	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$

P24GM-50-375B-3

Caution The ES is different from the corresponding mass-produced products in shape and material. See "ES PACKAGE DRAWINGS (2/2)."

ES PACKAGE DRAWINGS (1/2)

24 PIN CERAMIC SHRINK DIP (300 mil) (FOR ES)



NOTES

1) Each lead centerline is located within 0.25 mm (0.01 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	24.0 MAX.	0.945 MAX.
B	2.3 MAX.	0.091 MAX.
C	1.778 (T.P.)	0.070 (T.P.)
D	0.46±0.05	0.018±0.002
F	0.8 MIN.	0.031 MIN.
G	3.0±1.0	0.118±0.04
H	1.0 MIN.	0.039 MIN.
I	2.7	0.106
J	4.3 MAX.	0.170 MAX.
K	7.62 (T.P.)	0.300 (T.P.)
L	7.5	0.295
M	0.25±0.05	$0.010^{+0.002}_{-0.003}$
N	0.25	0.01
R	0~15°	0~15°

P24D-70-300B1-1

MEMO

24. COMPARISON OF FUNCTIONS OF μPD17120 SUB-SERIES

Item \ Product	μPD17120	μPD17121	μPD17132	μPD17133
ROM	1.5K bits		2K bytes	
RAM	64 × 4 bits		111 × 4 bits	
Stack	Five levels of address stack One level of interrupt stack			
Instruction execution time (clock, operating voltage)	8 μs (2 MHz, 2.7 to 5.5 V)	2 μs (8 MHz, 4.5 to 5.5 V) 4 μs (4 MHz, 2.7 to 5.5 V)	8 μs (2 MHz, 2.7 to 5.5 V)	2 μs (8 MHz, 4.5 to 5.5 V) 4 μs (4 MHz, 2.7 to 5.5 V)
I/O CMOS I/O	12 (P0A, P0B, P0C)			
I/O Sense input	1 (INT)			
I/O N-ch open-drain I/O	6 (P0D, P0E Withstand voltage: 9 V) P0D pull-up resistor: Mask option P0E pull-up resistor: Mask option		6 (P0D, $P0E_0$ Withstand voltage: 9 V; $P0E_1$ Withstand voltage: V_{DD}) P0D pull-up resistor: Mask option P0E pull-up resistor: Mask option	
Built-in pull-up resistance	100 kΩ TYP.			
Comparator (operating voltage)	None		4 (V_{DD} = 2.7 to 5.5 V)	
Reference voltage pin	—		V_{ref} = (V_{ref} = 0 V to V_{DD})	
Timer (8-bit)	1 (Timer output: $\overline{TMOUT}$)			
Interrupt External	1			
Interrupt Internal	2 (TM, SIO)			
SIO	1 (Clock-synchronous three-wire)			
Stand-by function	HALT, STOP			
Oscillation settling time	256 × 256 count			
Power-on/power-down reset circuit	Built-in (Can be used in an application circuit where V_{DD} is 5 V ±10 %)	Built-in (Can be used in an application circuit where V_{DD} is 5 V ±10 %, f_X is 400 kHz to 4 MHz)	Built-in (Can be used in an application circuit where V_{DD} is 5 V ±10 %)	Built-in (Can be used in an application circuit where V_{DD} is 5 V ±10 %, f_X is 400 kHz to 4 MHz)
Package	24-pin plastic shrink DIP (300 mil) 24-pin plastic SOP (375 mil)			
One-time PROM	μPD17P132	μPD17P133	μPD17P132	μPD17P133

25. RECOMMENDED SOLDERING CONDITIONS

The conditions listed below shall be met when soldering the μPD17120.

For details of the recommended soldering conditions, refer to our document *SMD Surface Mount Technology Manual* (IEI-1207).

Please consult with our sales offices in case any other soldering process is used, or in case soldering is done under different conditions.

Table 25-1 Soldering Conditions for Surface-Mount Devices

μPD17120GT-×××: 24-pin plastic SOP (375 mil)

Soldering process	Soldering conditions	Symbol
Infrared ray reflow	Peak package's surface temperature: 235 °C Reflow time: 30 seconds or less (at 210 °C or more) Maximum allowable number of reflow processes: 2 Exposure limit: 7 days[Note] (20 hours of pre-baking is required at 125 °C afterward.) <Cautions> (1) Do not start reflow-soldering the device if its temperature is higher than the room temperature because of a previous reflow soldering. (2) Do not use water for flux cleaning before a second reflow soldering.	IR35-207-2
VPS	Peak package's surface temperature: 215 °C Reflow time: 40 seconds or less (at 200 °C or more) Maximum allowable number of reflow processes: 2 Exposure limit: 7 days[Note] (20 hours of pre-baking is required at 125 °C afterward.) <Cautions> (1) Do not start reflow-soldering the device if its temperature is higher than the room temperature because of a previous reflow soldering. (2) Do not use water for flux cleaning before a second reflow soldering.	VP15-207-2
Wave soldering	Temperature in the soldering vessel: 260 °C or less Soldering time: 10 seconds or less Number of soldering process: 1 Preheating temperature: 120 °C max. (measured on the package surface) Exposure limit: 7 days[Note] (20 hours of pre-baking is required at 125 °C afterward.)	WS60-207-1 ★
Partial heating method	Terminal temperature: 300 °C or less Flow time: 3 seconds or less (for each side of device)	—

Note Exposure limit before soldering after dry-pack package is opened.
Storage conditions: Temperature of 25 °C and maximum relative humidity at 65 % or less

Caution Do not apply more than a single process at once, except for "Partial heating method."

Table 25-2 Soldering Conditions for Through Hole Mount Devices

μPD17120CS-×××: 24-pin plastic shrink DIP (300 mil)

Soldering process	Soldering conditions
Wave soldering (only for terminals)	Solder temperature: 260 °C or less Flow time: 10 seconds or less
Partial heating method	Terminal temperature: 300 °C or less Flow time: 3 seconds or less (for each terminal) ★

Caution In wave soldering, apply solder only to the terminals. Care must be taken that jet solder does not come in contact with the main body of the package.

APPENDIX DEVELOPMENT TOOLS

The following support tools are available for developing programs for the μPD17120.

Hardware

Name	Description
In-circuit emulator [IE-17K IE-17K-ET Note 1 EMU-17K Note 2]	The IE-17K, IE-17K-ET, and EMU-17K are in-circuit emulators applicable to the 17K series. The IE-17K and IE-17K-ET are connected to the PC-9800 series (host machine) or IBM PC/AT™ through the RS-232-C interface. The EMU-17K is inserted into the extension slot of the PC-9800 series (host machine). Use the system evaluation board (SE board) corresponding to each product together with one of these in-circuit emulators. SIMPLEHOST™, a man machine interface, implements an advanced debug environment. The EMU-17K also enables user to check the contents of the data memory in real time.
SE board (SE-17120)	The SE-17120 is an SE board for the μPD17120 sub-series. It is used solely for evaluating the system. It is also used for debugging in combination with the in-circuit emulator.
Emulation probe (EP-17120CS)	The EP-17120CS is an emulation probe for the μPD17120 sub-series. Use this emulation probe to connect the SE board to target system.
PROM programmer [AF-9703 Note 3 ★ AF-9704 Note 3 AF-9705 Note 3 AF-9706 Note 3]	The AF-9703, AF-9704, AF-9705, and AF-9706 are PROM programmers for the μPD17P132. Use one of these PROM programmers with the program adapter, AF-9808, to write a program into the μPD17P132.
★ Program adapter (AF-9808M Note 3)	The AF-9808M is a socket unit for the μPD17P132CS or μPD17P132GT. It is used with the AF-9703, AF-9704, AF-9705, or AF-9706.

Notes 1. Low-end model, operating on an external power supply

2. The EMU-17K is a product of IC Co., Ltd. Contact IC Co., Ltd. (Tokyo, 03-3447-3793) for details.

3. The AF-9703, AF-9704, AF-9705, AF-9706, and AF-9808M are products of Ando Electric Co., Ltd. Contact Ando Electric Co., Ltd. (Tokyo, 03-3733-1151) for details.

Software

Name	Description	Host machine	OS		Distribution media	Part number
17K series assembler (AS17K)	AS17K is an assembler applicable to the 17K series. In developing μPD17120 programs, AS17K is used in combination with a device file (AS17120).	PC-9800 series	MS-DOS™		5.25-inch, 2HD	μS5A10AS17K
					3.5-inch, 2HD	μS5A13AS17K
		IBM PC/AT	PC DOS™		5.25-inch, 2HC	μS7B10AS17K
					3.5-inch, 2HC	μS7B13AS17K
Device file (AS17120)	AS17120 is a device file for the μPD17120 and μPD17P132 . It is used together with the assembler (AS17K), which is applicable to the 17K series.	PC-9800 series	MS-DOS		5.25-inch, 2HD	μS5A10AS17120 **Note**
					3.5-inch, 2HD	μS5A13AS17120 **Note**
		IBM PC/AT	PC DOS		5.25-inch, 2HC	μS7B10AS17120 **Note**
					3.5-inch, 2HC	μS7B13AS17120 **Note**
Support software (SIMPLEHOST)	SIMPLEHOST, running on the Windows™, provides man-machine-interface in developing programs by using a personal computer and the in-circuit emulator.	PC-9800 series	MS-DOS	Windows	5.25-inch, 2HD	μS5A10IE17K
					3.5-inch, 2HD	μS5A13IE17K
		IBM PC/AT	PC DOS		5.25-inch, 2HC	μS7B10IE17K
					3.5-inch, 2HC	μS7B13IE17K

Note μS××××AS17120 indicates the AS17120, AS17121, AS17132, and AS17133.

Remark The following table lists the versions of the operating systems described in the above table.

OS	Versions
MS-DOS	Ver. 3.30 to Ver. 5.00A**Note**
PC DOS	Ver. 3.1 to Ver. 5.0**Note**
Windows	Ver. 3.0 to Ver. 3.1

Note MS-DOS versions 5.00 and 5.00A and PC DOS Ver. 5.0 are provided with a task swap function. This function, however, cannot be used in these software packages.

[MEMO]

Cautions on CMOS Devices

Countermeasures against static electricity for all MOSs

Caution When handling MOS devices, take care so that they are not electrostatically charged.

Strong static electricity may cause dielectric breakdown in gates. When transporting or storing MOS devices, use conductive trays, magazine cases, shock absorbers, or metal cases that NEC uses for packaging and shipping. Be sure to ground MOS devices during assembling. Do not allow MOS devices to stand on plastic plates or do not touch pins. Also handle boards on which MOS devices are mounted in the same way.

$ **CMOS-specific handling of unused input pins**

Caution Hold CMOS devices at a fixed input level.

Unlike bipolar or NMOS devices, if a CMOS device is operated with no input, an intermediate-level input may be caused by noise. This allows current to flow in the CMOS device, resulting in a malfunction. Use a pull-up or pull-down resistor to hold a fixed input level. Since unused pins may function as output pins at unexpected times, each unused pin should be separately connected to the V_{DD} or GND pin through a resistor.

If handling of unused pins is documented, follow the instructions in the document.

% **Statuses of all MOS devices at initialization**

Caution The initial status of a MOS device is unpredictable when power is turned on.

Since characteristics of a MOS device are determined by the amount of ions implanted in molecules, the initial status cannot be determined in the manufacture process. NEC has no responsibility for the output statuses of pins, input and output settings, and the contents of registers at power on. However, NEC assures operation after reset and items for mode setting if they are defined.

When you turn on a device having a reset function, be sure to reset the device first.

SIMPLEHOST is a trademark of NEC Corporation.

MS-DOS and Windows are trademarks of Microsoft Corporation.

PC/AT and PC DOS are trademarks of IBM Corporation.

The export of this product from Japan is regulated by the Japanese government. To export this product may be prohibited without governmental license, the need for which must be judged by the customer. The export or re-export of this product from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

M4 94.11