DATA SHEET

NEC

MOS INTEGRATED CIRCUIT

μPD78320,78322

16/8-BIT SINGLE-CHIP MICROCONTROLLER

DESCRIPTION

The μPD78322 is a 16-/8-bit single-chip microcontroller that incorporates a high-performance 16-bit CPU. The μPD78322 is one of 78K/III series.

A realtime pulse unit for realtime pulse control required in motor control, an A/D converter, a ROM, and a RAM have been integrated into one chip.

The μPD78322 incorporates 16K-byte mask ROM and 640-byte RAM.

The μPD78320 is provided as a ROM-less product of the μPD78322. Also, the μPD78P322 is provided as an on-chip PROM product.

Detailed information about product features and specifications can be found in the following document.

μPD78322 User's Manual : IEU-1248

FEATURES

- Internal 16-bit architecture and external 8-bit data bus
- High-speed processing by pipeline control and instruction prefetch
  - Minimum instruction execution time: 250 ns (with 16 MHz external clock in operation)
- Instruction set suitable for control operations (μPD78312 upward compatible)
  - Multiplication/division instruction (16 bits × 16 bits, 32 bits ÷ 16 bits)
  - Bit manipulation instruction
  - String instruction, etc.
- On-chip high-function interrupt controller
  - 3-level priority specifiable
  - 3-type interrupt servicing mode selectable
    (Vectored interrupt function, context switching function, and macro service function)
- Variety of peripheral hardware
  - Realtime pulse unit
  - 8-channel, 10-bit A/D converter
  - Watchdog timer
- Powerful serial interface (with an on-chip dedicated baud rate generator)
  - UART ····· 1 channel
  - SBI (NEC Standard Serial Bus Interface) } ····· 1 channel
  - 3-wire serial I/O }

APPLICATIONS

- Motor control devices

Unless there are any particular notices, the μPD78322 is described as the representative model in this document.

Document No. U10455EJ7V0DS00 (7th edition)
(Previous No. IC-2354)
Date Published November 1995 P
Printed in Japan

The mark ★ shows major revised points.

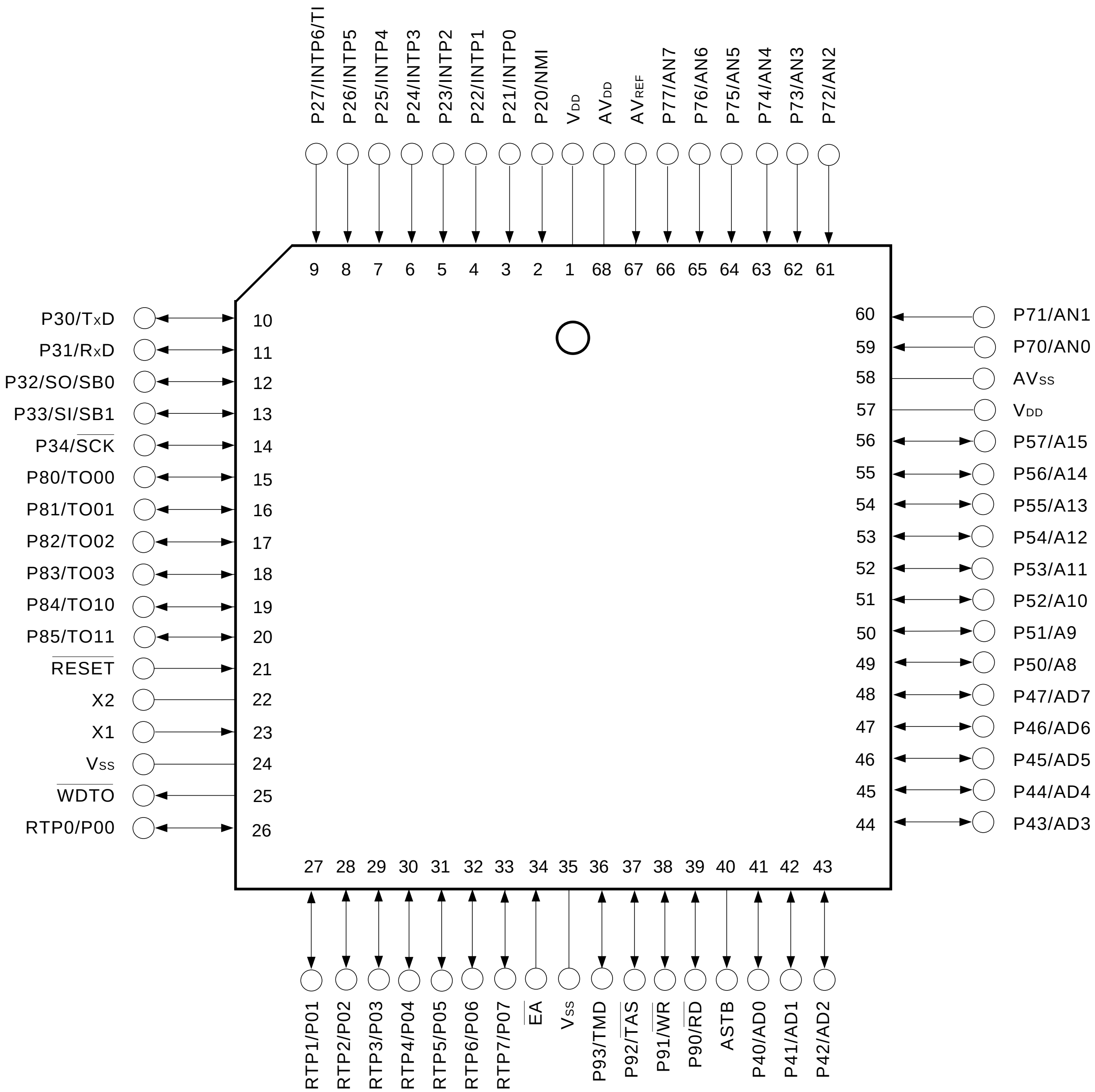
ORDERING INFORMATION

Part Number	Package	Internal ROM
μPD78320GF-3B9	80-pin plastic QFP (14 × 20 mm)	None
μPD78320GJ-5BJ	74-pin plastic QFP (20 × 20 mm)	None
μPD78320L	68-pin plastic QFJ (□950 mil)	None
μPD78322GF-××× -3B9	80-pin plastic QFP (14 × 20 mm)	Mask ROM
μPD78322GJ-××× -5BJ	74-pin plastic QFP (20 × 20 mm)	Mask ROM
μPD78322L-×××	68-pin plastic QFJ (□950 mil)	Mask ROM

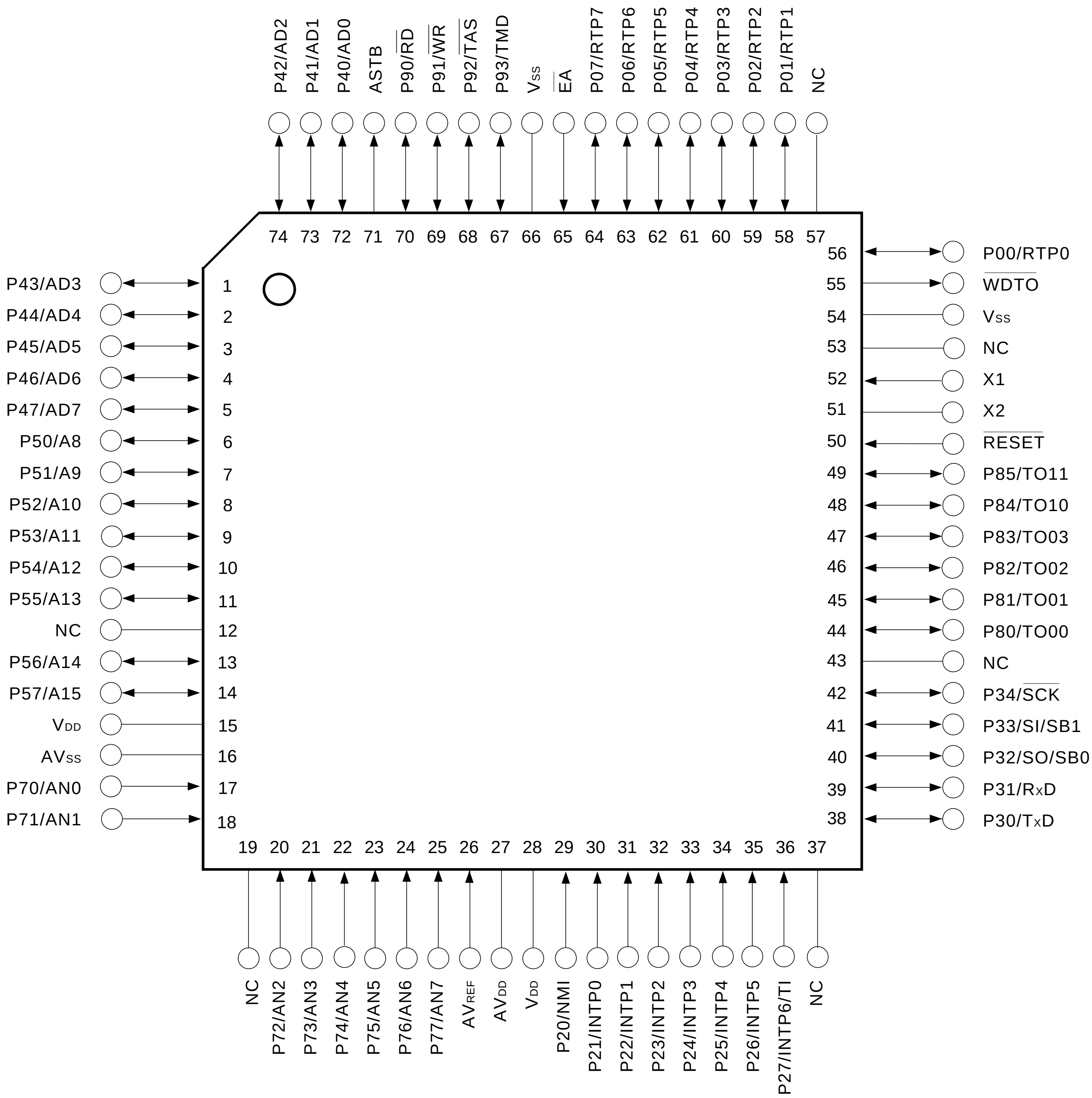
Remark ××× indicates ROM code number.

PIN CONFIGURATION

- **68-pin plastic QFJ (□950 mil)**
  μPD78320L
  μPD78322L-×××

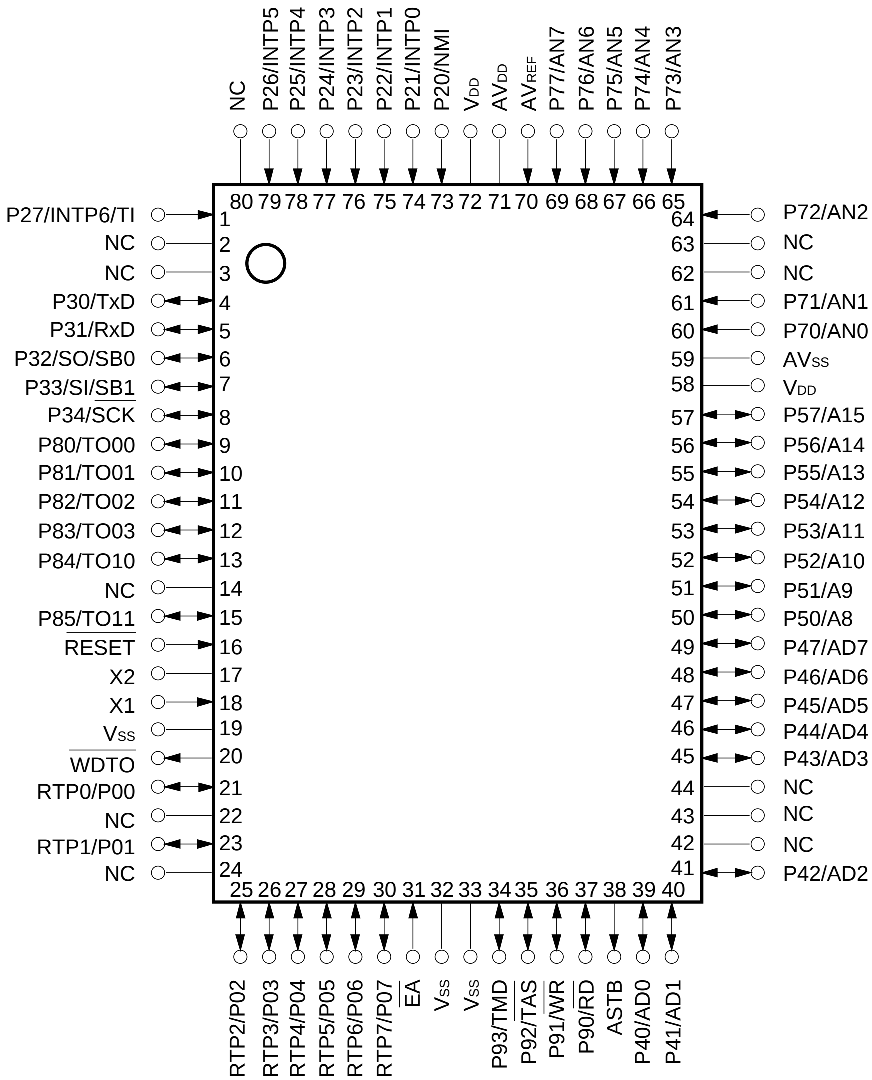


- **74-pin plastic QFP (20 × 20 mm)**
  μPD78320GJ-5BJ
  μPD78322GJ-×××-5BJ



Caution The NC pin should be connected to VSS for noise control (can also be left open).

- **80-pin plastic QFP (14 × 20 mm)**
  μPD78320GF-3B9
  μPD78322GF-×××-3B9



Caution The NC pin should be connected to V_{SS} for noise control (can also be left open).

P00 to P07 : Port0
P20 to P27 : Port2
P30 to P34 : Port3
P40 to P47 : Port4
P50 to P57 : Port5
P70 to P77 : Port7
P80 to P85 : Port8
P90 to P93 : Port9
NMI : Nonmaskable Interrupt
INTP0 to INTP6 : Interrupt From Peripherals
RTP0 to RTP7 : Realtime Port
TI : Timer Input
TxD : Transmit Data
RxD : Receive Data
SB0/SO : Serial Bus/Serial Output
SB1/SI : Serial Bus/Serial Input
SCK : Serial Clock
TO00 to TO03, TO10, TO11 : Timer Output

RESET : Reset
X1, X2 : Crystal
WDTO : Watchdog Timer Output
EA : External Access
TMD : Turbo Mode
TAS : Turbo Access Strobe
WR : Write Strobe
RD : Read Strobe
ASTB : Address Strobe
AD0 to AD7 : Address/Data Bus
A8 to A15 : Address Bus
AN0 to AN7 : Analog Input
AV_{REF} : Analog Reference Voltage
AV_{SS} : Analog V_{SS}
AV_{DD} : Analog V_{DD}
V_{DD} : Power Supply
V_{SS} : Ground
NC : Non-connection

GENERAL DESCRIPTION OF FUNCTIONS

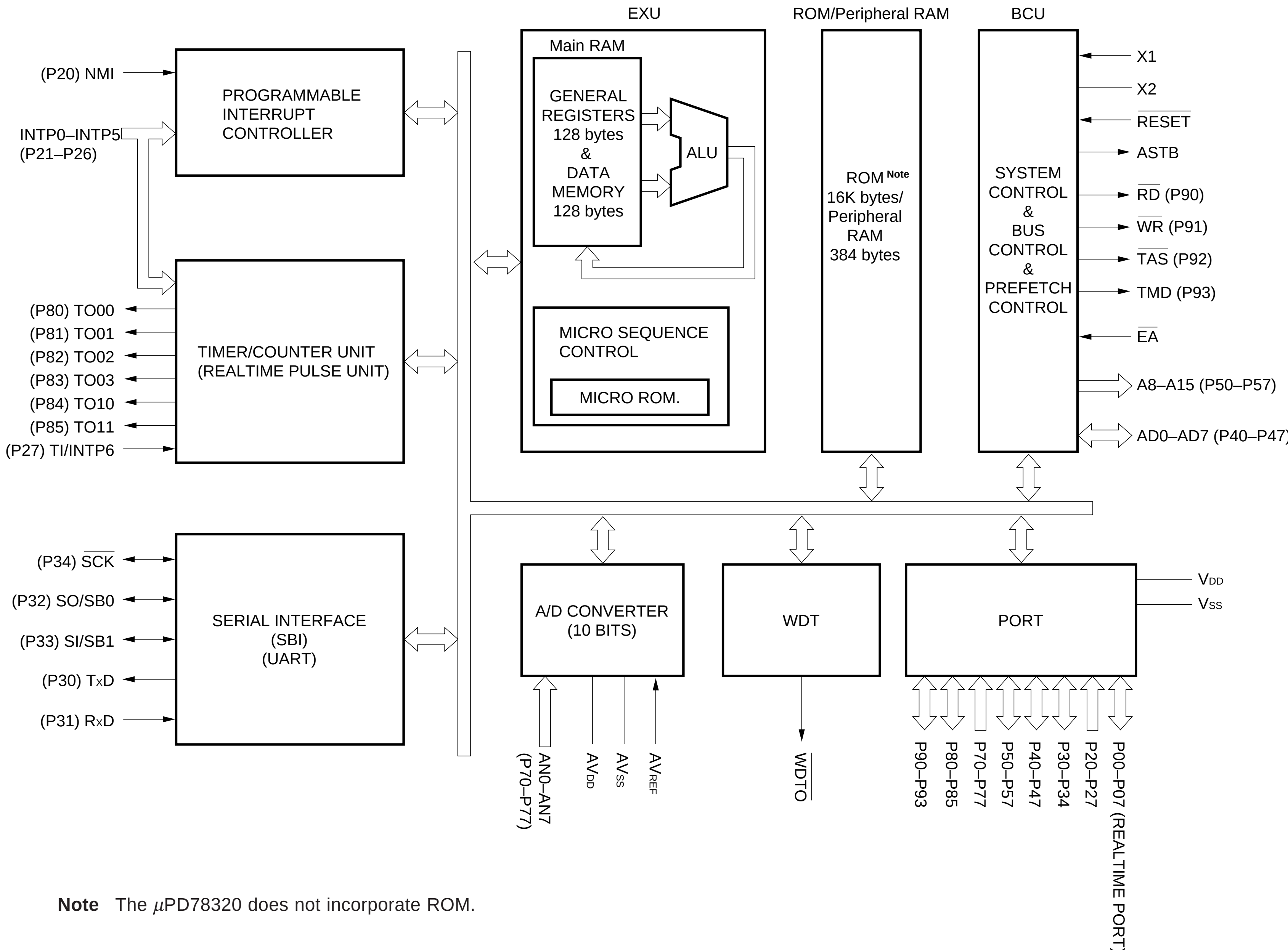
Basic instructions	111
Minimum instruction execution time	250 ns (with 16 MHz external clock in operation)
Internal memory	• ROM : 16384 × 8 bits (μPD78322) None (μPD78320) • RAM : 640 × 8 bits
Memory space	64K bytes
General registers	8 bits × 16 × 8 banks (memory mapping)
I/O line	• Input port : 16 (dual-function as analog input: 8) • Input/output port : 39 (μPD78322) 21 (μPD78320)
Realtime pulse unit	• 18-/16-bit free running timer × 1 • 16-bit timer/event counter × 1 • 16-bit compare register × 6 • 18-bit capture register × 4 • 18-bit capture/compare register × 2 • Realtime output port × 8
Serial communication interface	Serial interface with a dedicated baud rate generator • UART : 1 channel • SBI (NEC Serial Bus Interface) : 1 channel
A/D converter	10-bit resolution (8 analog inputs)
Interrupt	• External : 8, internal : 14 (dual-function as external : 2) • 3 servicing modes (vectored interrupt function, context switching function, and macro service function)
Test factor	Internal : 1
Standby	STOP mode/HALT mode
Instruction set	16-bit transfer/operation instruction, multiplication/division instruction (16 × 16, 32 ÷ 16), bit manipulation instruction, string instruction, etc.
Others	On-chip watchdog timer
Package	• 68-pin plastic QFJ (□ 950 mil) • 74-pin plastic QFP (20 × 20 mm) • 80-pin plastic QFP (14 × 20 mm)

DIFFERENCES BETWEEN μPD78322 AND 78320

Item \ Product Name		μPD78322	μPD78320
Internal ROM		16K bytes	None
I/O line	Input	16 (dual-function as analog input: 8)	
	Input /output	39	21
Port 4 (P40 to P47)		Specifiable as I/O as an 8-bit unit. Functions as multiplexed address/data buses (AD0 to AD7) in the external memory expansion mode.	Functions always as multiplexed address/data buses.
Port 5 (P50 to P57)		Specifiable as I/O bit-wise. Functions as address bus (A8 to A15) in the external memory expansion mode.	Functions always as address bus.
Port 9 (P90 to P93)		Specifiable as I/O bit-wise. In the external memory expansion mode, P90 and P91 function as $\overline{RD}$ strobe signal output and $\overline{WR}$ strobe signal output, respectively. In the external memory high-speed fetch mode, P92 and P93 function as $\overline{TAS}$ output and TMD output, respectively.	Always P90 and P91 function as $\overline{RD}$ strobe and $\overline{WR}$ strobe signal output, respectively.
Memory expansion mode register (MM)		Port 4 I/O mode is set as an 8-bit unit .	In the μPD78322 emulation mode, turbo access manager (μPD71P301)Note PA and PB pins are controlled as port 4 and port 5 emulation pins.
Port 5 mode register (PM5)		Port 5 I/O mode is set bit-wise.	

★ **Note** Maintenance product

BLOCK DIAGRAM



Note The μPD78320 does not incorporate ROM.

CONTENTS

1. LIST OF PIN FUNCTIONS

1.1 PORT PINS

Pin Name	I/O	Function	Dual-Function Pin
P00 to P07	Input/ output	Port 0 8-bit input/output port Input/output can be specified bit-wise Also serves as a realtime output port.	RTP0 to RTP7
P20	Input	Port 2 8-bit dedicated input port	NMI
P21			INTP0
P22			INTP1
P23			INTP2
P24			INTP3
P25			INTP4
P26			INTP5
P27			INTP6/TI
P30	Input/ output	Port 3 5-bit input/output port Input/output can be specified bit-wise	TxD
P31			RxD
P32			SO/SB0
P33			SI/SB1
P34			$\overline{\text{SCK}}$
P40 to P47	Input/ output	Port 4 8-bit input/output port Input/output can be specified in 8-bit unit.	AD0 to AD7
P50 to P57	Input/ output	Port 5 8-bit input/output port Input/output can be specified bit-wise	A8 to A15
P70 to P77	Input	Port 7 8-bit dedicated input port	AN0 to AN7
P80	Input/ output	Port 8 6-bit input/output port Input/output can be specified bit-wise	TO00
P81			TO01
P82			TO02
P83			TO03
P84			TO10
P85			TO11
P90	Input/ output	Port 9 4-bit input/output port Input/output can be specified bit-wise	$\overline{\text{RD}}$
P91			$\overline{\text{WR}}$
P92			$\overline{\text{TAS}}$
P93			TMD

1.2 NON-PORT PINS (1/2)

Pin Name	I/O	Function	Dual-Function Pin
RTP0 to RTP7	Output	Realtime output port which generates pulses in synchronization with the trigger signal transmitted from the realtime pulse unit (RPU).	P00 to P07
NMI	Input	Nonmaskable interrupot request input capable of specifying the effective at the rising or falling edge by a mode register.	P20
INTP0	Input	External interrupt request input capable of specifying the effective edgy by a mode register.	P21
INTP1			P22
INTP2			P23
INTP3			P24
INTP4			P25
INTP5			P26
INTP6			P27/TI
TI	Input	External count clock input to timer 1 (TM1)	P27/INTP6
TxD	Output	Serial data output of asynchronous serial interface (UART)	P30
RxD	Input	Serial data input of asynchronous serial interface (UART)	P31
SO	Output	Serial data output of clocked serial interface in 3-wire mode	P32/SB0
SI	Input	Serial data input of clocked serial interface in 3-wire mode	P33/SB1
SB0	Input /output	Serial data input/output of clocked serial interface in SBI mode	P32/SO
SB1			P33/SI
$\overline{\text{SCK}}$	Input /output	Serial clock input/output of clocked serial interface	P34
AD0 to AD7	Input /output	Multiplexed address/data bus for external memory expansion	P40 to P47
A8 to A15	Output	Address bus for external memory expansion	P50 to P57
TO00	Output	Pulse output from the realtime pulse unit	P80
TO01			P81
TO02			P82
TO03			P83
TO10			P84
TO11			P85
$\overline{\text{RD}}$	Output	Strobe signal output generated for external memory read operation	P90
$\overline{\text{WR}}$		Strobe signal output generated for external memory write operation	P91
$\overline{\text{TAS}}$		Control signal output generated for access to turbo access manager μPD71P301[Note]	P92
TMD			P93
$\overline{\text{WDTO}}$	Output	Signal output indicating that the watchdog timer has generated a nonmascable interrupt.	—
ASTB	Output	Timing signal output generated for externally latching the lower address information output from pins AD0 to AD7 in order to access the external memory.	—

Note Maintenance product

1.2 NON- PORT PINS (2/2)

Pin Name	I/O	Function	Dual-Function Pin
$\overline{\text{EA}}$	Input	In the μPD78322, $\overline{\text{EA}}$ pin is normally connected to V_{DD}. Connecting $\overline{\text{EA}}$ pin to V_{SS} sets the ROM-less mode and accesses the external memory. In the μPD78320, this pin should be fixed to "0" (low level). The $\overline{\text{EA}}$ pin level cannot be changed during operation.	—
AN0 to AN7	Input	A/D converter analog input	—
AV_{REF}	Input	A/D converter reference voltage input	—
AV_{DD}	—	A/D converter analog power supply	—
AV_{SS}	—	A/D converter GND	—
$\overline{\text{RESET}}$	Input	System reset input	—
X1	Input	Crystal connect pin for sysem clock oscillation. When an external clock is supplied, the clock is input to X1 and the inverted clock is input to X2. (X2 can also be left open.)	—
X2	—		—
V_{DD}	—	Positive power supply	—
V_{SS}	—	GND pin	—
NC	—	Not internally connected. Connected to V_{SS} (GND) (can also be left open).	—

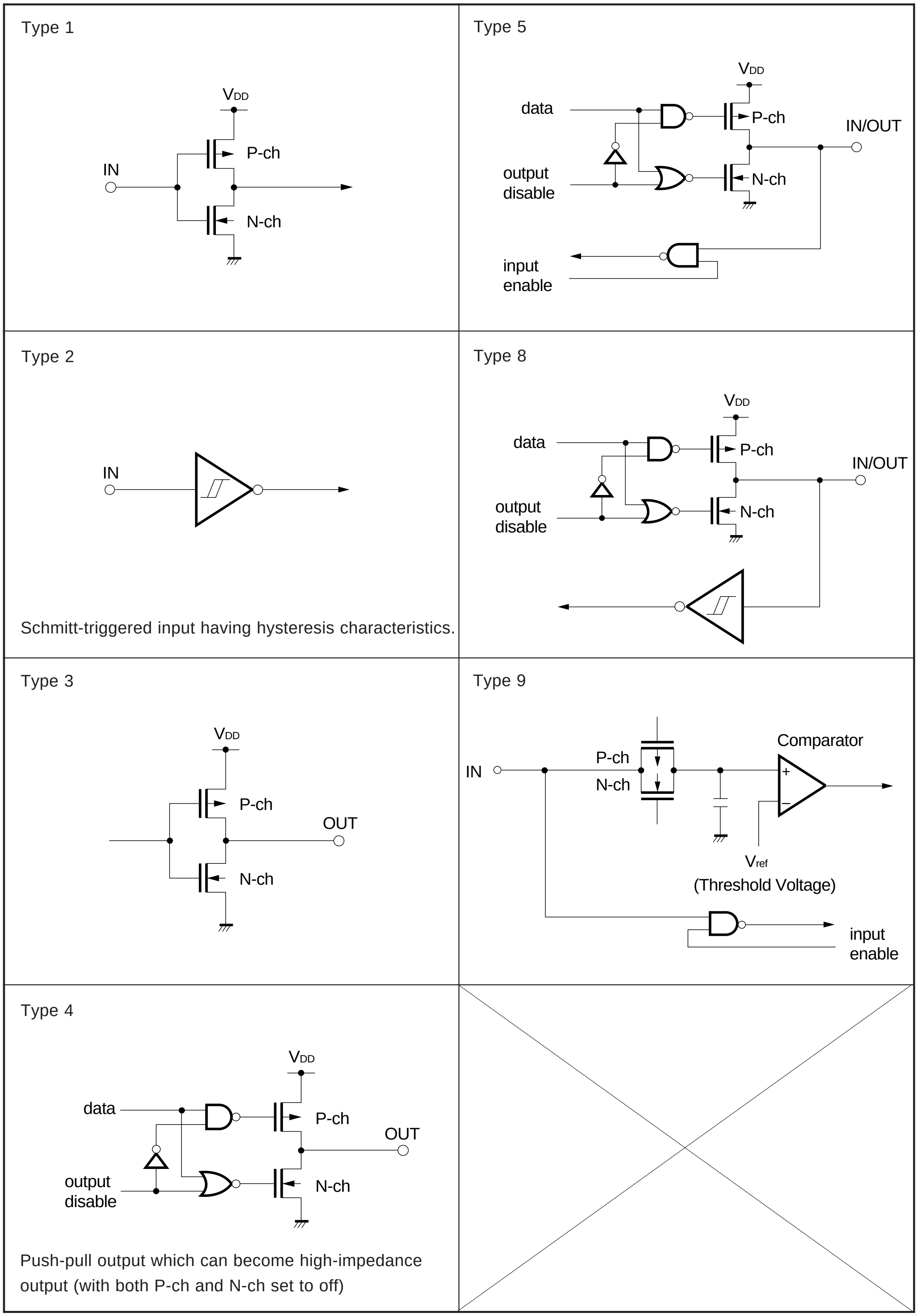
1.3 PIN INPUT/OUTPUT CIRCUITS AND RECOMMENDED CONNECTION OF UNUSED PINS

The pin input/output circuits, partly simplified, are shown in Table 1-1 and Figure 1-1.

Table 1-1. I/O Circuit Types of Pins and their Recommended Connection Methods when Unused

Pin	Input/Output Circuit Type	Recommended Connection Method
P00 to P07/RTP0 to RTP7	5	Input mode : Individually connected to V_{DD} or V_{SS} via resistor Output mode : Leave open
P20/NMI P21 to P26/INTP0 to INTP5 P27/INTP6/TI	2	Connected to V_{SS}
P30/TxD P31/RxD	5	Input mode : Individually connected to V_{DD} or V_{SS} via resistor Output mode : Leave open
P32/SO/SB0 P33/SI/SB1 P34/$\overline{SCK}$	8	
P40 to P47/AD0 to AD7 P50 to P57/A8 to A15	5	
P70 to P77/AN0 to AN7	9	Connected to V_{SS}
P80 to P83/TO00 to TO05 P84, P85/TO10, TO11	5	Input mode : Individually connected to V_{DD} or V_{SS} via resistor Output mode : Leave open
P90/$\overline{RD}$ P91/$\overline{WR}$ P92/$\overline{TAS}$ P93/TMD	5	
$\overline{WDTO}$	3	Leave open
ASTB	4	
$\overline{EA}$	1	—
$\overline{RESET}$	2	—
AV_{REF}, AV_{SS}	—	Connected to V_{SS}
AV_{DD}	—	Connected to V_{DD}
NC	—	Connected to V_{SS} (can also be left open)

Figure 1-1. Pin Input/Output Circuits



2. CPU ARCHITECTURE

2.1 MEMORY SPACE

In the μPD78322 a maximum of 64K bytes of memory can be addressed (see **Figure 2-1**).

Program fetches can be performed within the area from 0000H to FDFFH. However, when external memory expansion is implemented in the area from FE00H to FFFFH (main RAM and special function register area), program fetches can also be performed on this area. In this case, a program fetch is performed on the external memory, not on the main RAM or special function registers.

(1) Vector table area

Interrupt request from the peripheral hardware, reset input, external interrupt request and interrupt branch address by break instruction are stored in the 0000H to 003FH 64-byte area. Generation of an interrupt request sets the even address content of each table in the lower 8 bits of the program counter (PC) and the odd address content in the higher 8 bits, and a branch is made.

Interrupt Source		Vector Table Address
$\overline{\text{RESET}}$	($\overline{\text{RESET}}$ pin input)	0000H
NMI	(NMI pin input)	0002H
WDT	(Watchdog timer)	0004H
TMF0	(Realtime pulse unit)	0006H
EXF0	(INTP0 pin input)	0008H
EXF1	(INTP1 pin input)	000AH
EXF2	(INTP2 pin input)	000CH
EXF3	(INTP3 pin input)	000EH
EXF4/CCFX0	(INTP4 pin input/realtime pulse unit)	0010H
EXF5/CCFX1	(INTP5 pin input/realtime pulse unit)	0012H
EXF6/TI	(INTP6/TI pin input)	0014H
CMF00	(Realtime pulse unit)	0016H
CMF01	(Realtime pulse unit)	0018H
CMF02	(Realtime pulse unit)	001AH
CMF03	(Realtime pulse unit)	001CH
CMF10	(Realtime pulse unit)	001EH
CMF11	(Realtime pulse unit)	0020H
SRF	(Serial receive complete)	0024H
STF	(Serial send complete)	0026H
CSIIF	(Clocked serial interface)	0028H
ADF	(A/D converter)	002AH
Operation code trap		003CH
BRK	(Break instruction)	003EH

If bit 1 (TPF) of CPU control word (CCW) is set to 1, the 8002H to 803FH external memory area is used as an interrupt vector table in place of 0002H to 003FH.

(2) CALLT table area

32 tables of call addresses of 1-byte call instruction (CALLT) can be stored in the 0040H to 007FH 64-byte area.

If bit 1 (TPF) of CPU control word (CCW) is set to 1, the 8040H to 807FH external memory area is used as a CALLT instruction table in place of 0040H to 007FH.

(3) CALLF entry area

The 0800H to 0FFFH area can be directly subroutine-called by 2-byte call instruction (CALLF).

(4) Internal RAM area

A 640-byte RAM is built in FC80H to FEFFH area.

This area is composed of the following 2 RAMs.

- Peripheral RAM : FC80H to FDFFH (384 bytes)
- Main RAM : FE00H to FEFFH (256 bytes)

The main RAM can be accessed at high speed.

In the main RAM area, the macro service control word and general register group composed of 8 register banks are mapped onto the 36 bytes from FE06H to FE2BH and the 128 bytes from FE80H to FEFFH, respectively.

(5) Special function register (SFR) area

Registers having specially assigned functions, such as on-chip peripheral hardware mode registers and control registers, are mapped in the FF00H to FFFFH area. Addresses without mapped registers cannot be accessed.

(6) External memory area

The μPD78322 can add external memories (ROM, RAM) to the 48K-byte (4000H to FFFFH) area gradually.

The μPD78320 can connect external memories (ROM, RAM) to the 64K-byte (0000H to FFFFH) area.

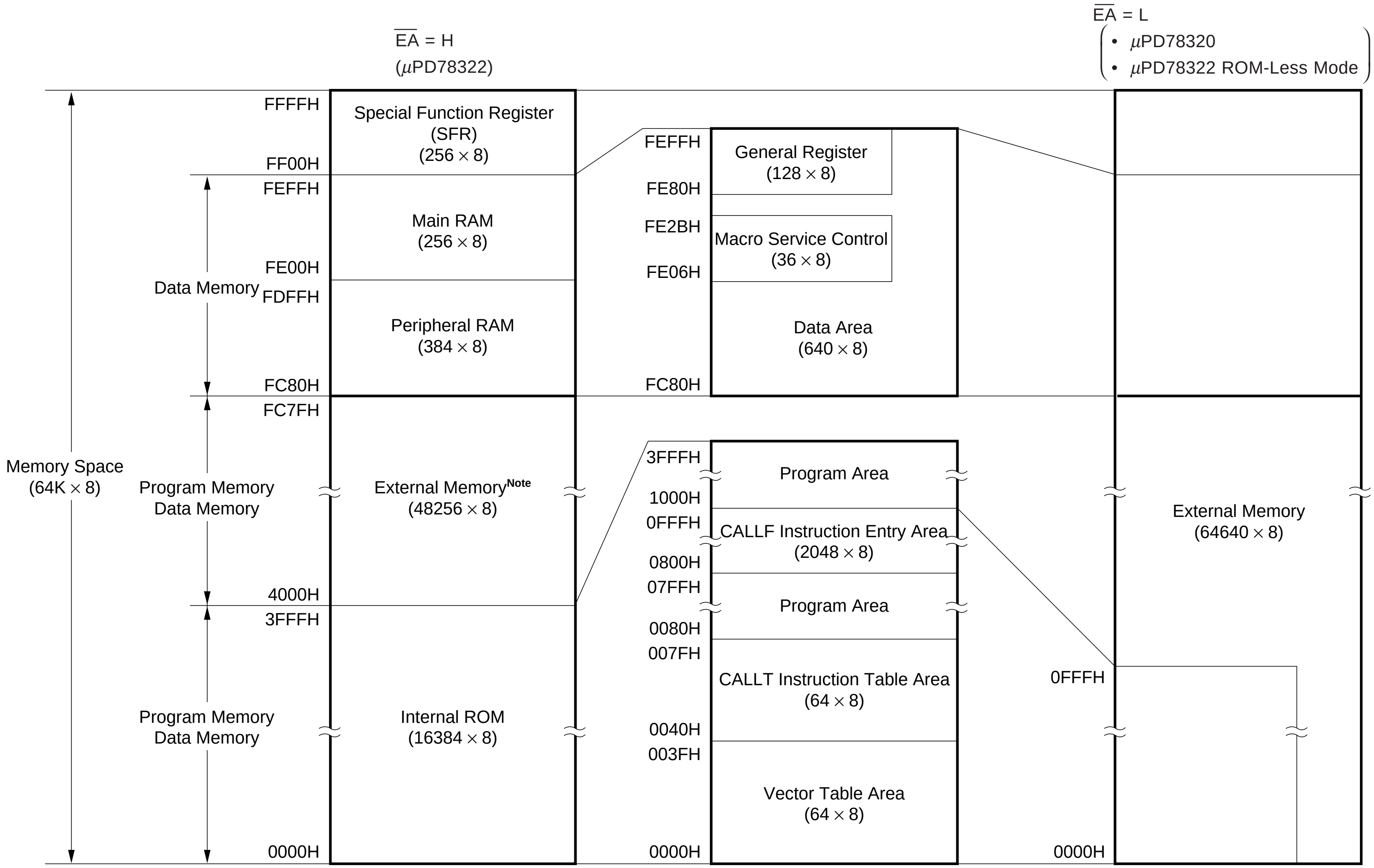
Each external memory can be accessed using P40/AD0 to P47/AD7 (multiplexed address/data bus), P50/A8 to P57/A15 (address bus) and $\overline{\text{RD}}$, $\overline{\text{WR}}$ and ASTB signals.

The external access area is mapped in the FFD0H to FFDFH 16-byte area of the special function register (SFR). In this way, the external memory can be accessed by SFR addressing.

★ Dedicated pins ($\overline{\text{TAS}}$ and TMD pins) are provided to connect turbo access manager (μPD71P301)[Note]. If the μPD71P301 is used, the program processing speed equal to that of the internal ROM can be obtained.

Note Maintenance product

Figure 2-1. Memory Map



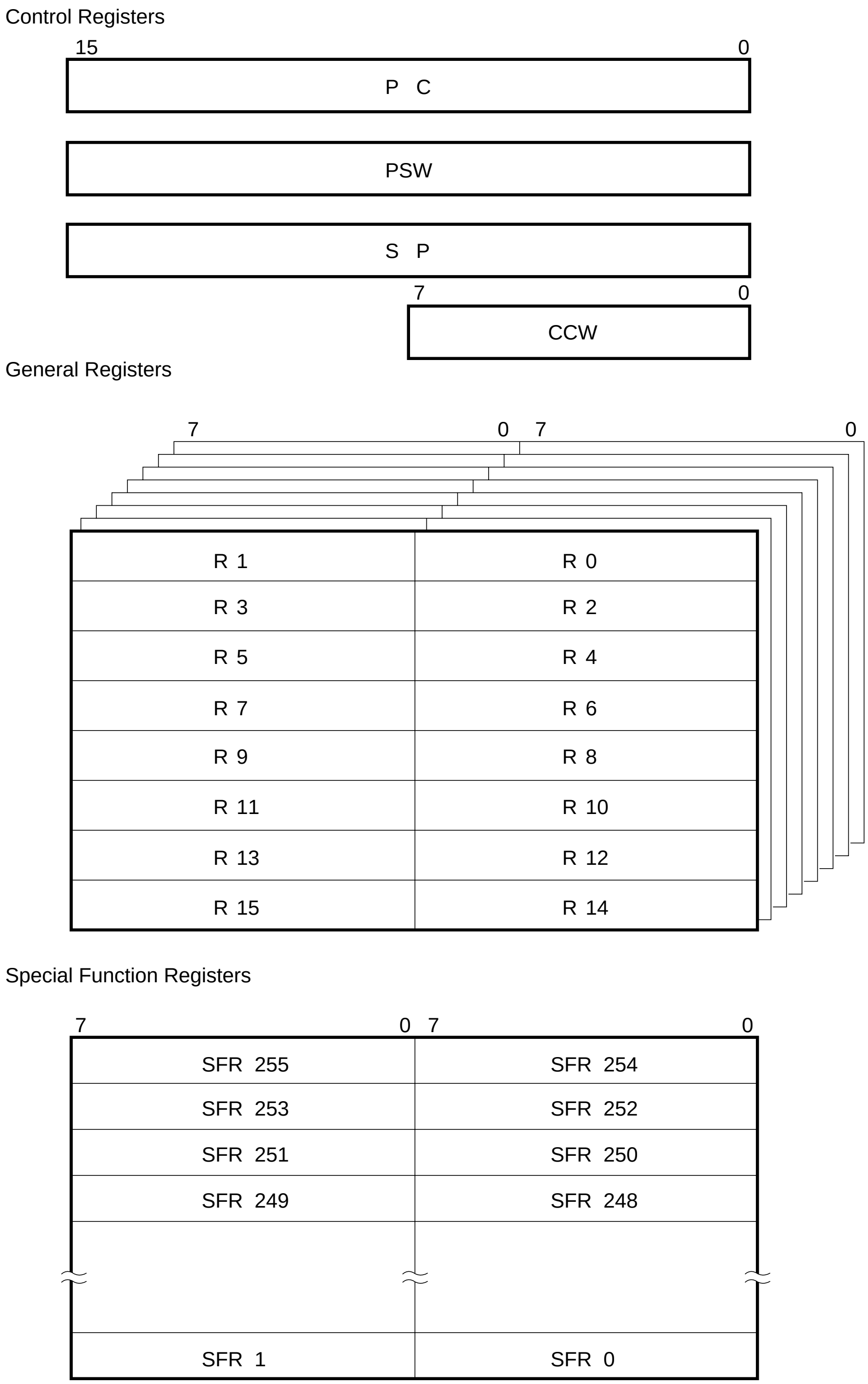
Note Accessed in external memory expansion mode.

★ **Caution For word access (including stack operations) to the main RAM area (FE00H-FEFFH), the address that specifies the operand must be an even value.**

2.2 PROCESSOR REGISTERS

The processor registers consist mainly of three groups. They are general registers consisting of 8 banks of sixteen 8-bit registers, control registers consisting of one 8-bit register and three 16-bit registers, and special function registers such as peripheral hardware I/O mode registers.

Figure 2-2. Register Configuration



Remark The CCWs of the control registers are mapped in the special function register (SFR) area.

2.2.1 Control Register

The control registers carry out dedicated functions such as control of the program sequence, status and stack memory, and modification of operand addressing. They consist of three 16-bit registers and one 8-bit register.

(1) Program counter (PC)

This is a 16-bit register which holds the address information of the next program to be executed. It is normally incremented according to the number of bytes of the instruction to be fetched. If an instruction with data branch is executed, immediate data and the register content are set. $\overline{\text{RESET}}$ input sets and branches the data of 0000H and 0001H reset vector tables in the PC.

(2) Program status word (PSW)

This is a 16-bit register consisting of various flags which are set or reset by the result of instruction execution. Read/ write access is carried out in units of the higher 8 bits (PSWH) or lower 8 bits (PSWL). Each flag can be manipulated using the bit manipulation instruction. If an interrupt request is made or BRK instruction is executed, data is automatically saved in the stack and is recovered by RETI or RETB instruction.

All bits are reset to 0 by $\overline{\text{RESET}}$ input.

Figure 2-3. PSW Format

	7	6	5	4	3	2	1	0
PSWH	UF	RBS2	RBS1	RBS0	0	0	0	0

	7	6	5	4	3	2	1	0
PSWL	S	Z	RSS	AC	IE	P/V	LT	CY

(a) Interrupt priority level transition flag (LT)

This flag is used to control the interrupt priority. For normal operation of the interrupt control circuit, this bit must not be manipulated by a program.

(b) Carry flag (CY)

If a carry is generated out of bit 7 or 15 as a result of the execution of an operation instruction or a borrow is generated into bit 7 or 15, this flag is set to 1. In all other cases, this flag is reset to 0. This flag can be tested by the conditional branch instruction.

When a bit manipulation instruction is executed, this flag functions as a bit accumulator.

(c) Zero flag (Z)

When the operation result is zero, this flag is set to 1. In all other cases, this flag is reset to 0. This flag can be tested by the conditional branch instruction.

(d) Sign flag (S)

When MSB of the operation result is “1”, this flag is set to 1. When the MSB is “0”, this flag is reset to 0. This flag can be tested by the conditional branch instruction.

(e) Parity/overflow flag (P/V)

Only when an overflow or underflow occurs as two’s complement during execution of an arithmetic operation instruction, this flag is set to 1. In all other cases, it is reset to 0 (overflow flag operation).

If the bit number of the operation result set to 1 is even during execution of an logic operation instruction, this flag is set to 1. If the bit number is odd, this flag is reset to 0 (parity flag operation).

This flag can be tested by the conditional branch instruction.

(f) Auxiliary carry flag (AC)

If a carry is generated out of bit 3 as a result of operation or a borrow is generated into bit 3, this flag is set to 1. In all other cases, this flag is reset to 0. This flag can be tested by the conditional branch instruction.

(g) Register set select flag (RSS)

This flag is used to specify general registers which function as X, A, C and B. As shown in Table 2-1, the RSS value determines the relationship between the functional register and the absolute register.

Thus, another register set (X, A, C, B) can be used by switching the RSS flag.

(h) Interrupt request enable flag (IE)

This flag is used to indicate interrupt request enable/disable. This flag is set to 1 by execution of EI instruction and is reset to 0 by execution of DI instruction or acceptance of an interrupt.

(i) Register bank select flag (RBS0 to RBS2)

This is a 3-bit flag to select one of eight register banks (RBANK0 to RBANK7).

(j) User flag (UF)

This flag is set or reset in the user program and can be used for program control.

(3) Stack pointer (SP)

This is a 16-bit register which holds the first address of the stack area (LIFO format) of the memory. It is manipulated by a dedicated instruction.

SP is decremented before write (save) operation into the stack memory and is incremented after read (restore) operation from the stack memory.

Since SP becomes indeterminate by $\overline{\text{RESET}}$ input, it must be set before subroutine call, etc..

(4) CPU control word (CCW)

This is an 8-bit register consisting of CPU control related flags. It is mapped in the special function register area and can be controlled by the software. All bits are reset to 0 by $\overline{\text{RESET}}$ input.

Figure 2-4. CCW Format

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	TPF	0	CCW

- **Table position flag (TPF)**

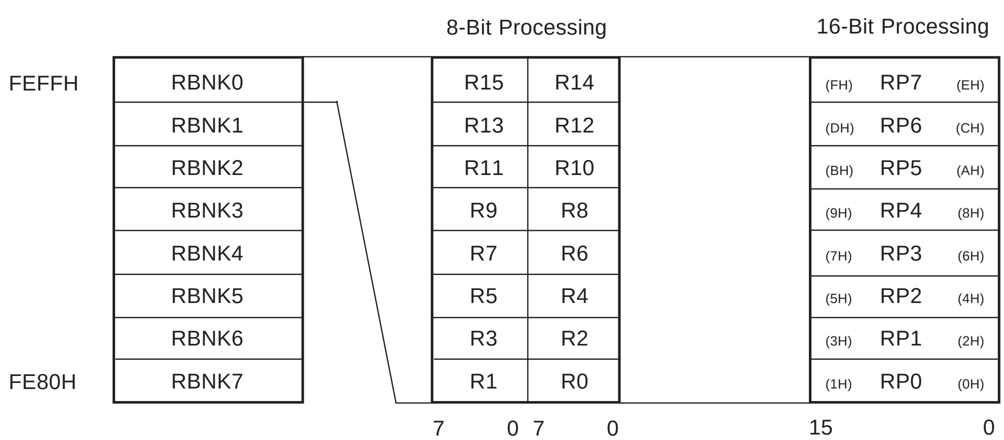
This flag is used to specify the interrupt vector table area and the memory area used as CALLT instruction table area.

As TPF has been reset to 0 after application of $\overline{\text{RESET}}$ input, the 0000H to 007FH address is used as each table area. The 8002H to 807FH address of the external memory area in place of 0002H to 007FH address can be used as each table area by setting TPF to 1 using the software. The vector tables of the BRK instruction, operation code trap interrupt and reset input are fixed to 003EH, 003CH and 0000H, respectively, and they are not affected by TPF.

2.2.2 General Registers

These are 128-byte registers mapped in the special area (FE80H to FEFFH) of the internal RAM space. They consist of eight register banks. The general register in the bank consists of sixteen 8-bit registers.

Figure 2-5. General Register Memory Location



The sixteen 8-bit registers can function as eight 16-bit register pairs (RP0 to RP7) as well.

As shown in Table 2-1, the sixteen 8-bit registers are characterized by functional names. The X register functions as the lower half of the 16-bit accumulator, the A register functions as the upper half of the 8-bit or 16-bit accumulator, the B and C registers function as counters, and DE, HL, VP and UP function as address register pairs. In particular the VP register functions as a base register and the UP register functions as a user stack pointer.

The unique function register changes as shown in Table 2-1 according to the value of the register set select flag (RSS) in the PSW.

Thus, if the program is described by the functional name, another register set of X, A, C and B can be used by means of the RSS flag.

The μPD78322 can carry out processed data addressing operations, implied addressing by functional names with importance attached to the unique function of each register and register addressing by absolute names with a view to fast processing with a small number of data transfers or creating highly descriptive programs.

Table 2-1. General Register Configuration

Absolute Name	Functional Name	
	RSS = 0	RSS = 1
R0	X	
R1	A	
R2	C	
R3	B	
R4		X
R5		A
R6		C
R7		B
R8	VP_L	VP_L
R9	VP_H	VP_H
R10	UP_L	UP_L
R11	UP_H	UP_H
R12	E	E
R13	D	D
R14	L	L
R15	H	H

Absolute Name	Functional Name	
	RSS = 0	RSS = 1
RP0	AX	
RP1	BC	
RP2		AX
RP3		BC
RP4	VP	VP
RP5	UP	UP
RP6	DE	DE
RP7	HL	HL

2.2.3 Special Function Registers (SFR)

These registers are provided with special functions. They include various peripheral hardware mode registers and control registers (CCW).

The special function registers are assigned in the FF00H to FFFFH 256-byte space. Short direct memory addressing is applied to the FF00H to FF1FH 32-byte area for processing with a short word length.

The bit manipulation, arithmetic and transfer instructions can be executed in all areas. The FFD0H to FFDFH 16-byte area is externally accessible by SFR addressing. Thus, the external memory can be accessed and the external device bit manipulation can be carried out by an instruction having a short word length.

Table 2-2 lists the special function registers (SFR). The items in the table have the following meanings.

- Symbol................. Indicates the address of the built-in special function register.
  Can be described in the instruction operand column.
- R/W....................... Indicates if the corresponding special function register can read or write.
  R/W : Read/write enable
  R : Read only enable (register bit test enable)
  W : Write only enable
- Manipulable bit unit
  .... Indicates the applicable manipulation bit unit for the corresponding special function register.
  16-bit manipulable SFR can be described in operand sfrp. When specified by an address, an even address is described.
  1-bit manipulable SFR can be described by the bit manipulation instruction.
- On reset .... Indicates the state of each register when $\overline{\text{RESET}}$ is input.

Cautions 1. Addresses for which no special function registers have been assigned cannot be accessed in the FF00H to FFFFH area.

2. Do not write to the read only register. If data is written, the internal circuit may malfunction.

Table 2-2. List of Special Function Registers (1/4)

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Unit			On Reset
				1 bit	8 bits	16 bits	
FF00H	Port 0	P0	R/W	○	○	—	Undefined
FF02H	Port 2	P2	R	—	○	—	
FF03H	Port 3	P3	R/W	○	○		
FF04H	Port 4	P4		○	○	—	
FF05H	Port 5	P5		○	○		
FF07H	Port 7	P7	R	—	○	—	
FF08H	Port 8	P8	R/W	○	○	—	
FF09H	Port 9	P9		○	○		
FF0AH FF0BH	Free running counter (lower 16 bits) Note	TM0LW	R	—	—	○	0000H
FF10H FF11H	Capture register X0 (lower 16 bits) Note	CTX0LW		— —	— —	○	Undefined
FF12H FF13H	Capture register 01 (lower 16 bits) Note	CT01LW		— —	— —	○	
FF14H FF15H	Capture register 02 (lower 16 bits) Note	CT02LW		— —	— —	○	
FF16H FF17H	Capture register 03 (lower 16 bits) Note	CT03LW		— —	— —	○	
FF18H FF19H	Capture/compoare register X0 (lower 16 bits) Note	CCX0LW	R/W	— —	— —	○	
FF1AH FF1BH	Capture/compoare register 01 (lower 16 bits) Note	CC01LW		— —	— —	○	
FF20H	Port 0 mode register	PM0	W	—	○	—	FFH
FF23H	Port 3 mode register	PM3		—	○	—	×××1 1111B
FF25H	Port 5 mode register	PM5		—	○	—	FFH
FF28H	Port 8 mode register	PM8		—	○	—	××11 1111B
FF29H	Port 9 mode register	PM9		—	○	—	×××× 1111B
FF2AH FF2BH	Free runnting counter (higher 16 bits) Note	TM0UW	R	— —	— —	○	0000H
FF2CH FF2DH	Timer register 1	TM1		— —	— —	○	
FF30H FF31H	Capture register X0 (higher 16 bits) Note	CTX0UW		— —	— —	○	Undefined
FF32H FF33H	Capture register 01 (higher 16 bits) Note	CT01UW		— —	— —	○	
FF34H FF35H	Capture register 02 (higher 16 bits) Note	CT02UW		— —	— —	○	

Note Upper or lower half of 18-bit register.

Table 2-2. List of Special Function Registers (2/4)

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Unit			On Reset
				1 bit	8 bits	16 bits	
FF36H	Capture register 03	CT03UW	R	—	—	○	Undefined
FF37H	(higher 16 bits) Note			—	—		
FF38H	Capture/compoare register X0	CCX0UW	R/W	—	—	○	
FF39H	(higher 16 bits) Note			—	—		
FF3AH	Capture/compoare register 01	CC01UW		—	—	○	
FF3BH	(higher 16 bits) Note			—	—		
FF40H	Port 0 mode control register	PMC0	W	—	○	—	00H
FF41H	Realtime output port set register	RTPS	R/W	○	○	—	
FF43H	Port 3 mode control register	PMC3	W	—	○	—	××× 0 0000B
FF48H	Port 8 mode control register	PMC8		—	○	—	×× 00 0000B
FF4CH	Baud rate generator	BRG	R/W	—	—	○	Undefined
FF4DH				—	—		
FF60H	Realtime output port register	RTP		○	○	—	
FF61H	Realtime output port reset register	RTPR		○	○	—	00H
FF62H	Port read control register	PRDC		○	○	—	
FF68H	A/D converter mode register	ADM		○	○	—	
FF6AH	A/D conversion result register (for 16-bit access)	ADCR	R	—	—	○	Undefined
FF6BH	A/D conversion result register (for upper 8-bit access)	ADCRH		—	○	—	
FF70H	Compare register 00	CM00	R/W	—	—	○	
FF71H				—	—		
FF72H	Compare register 01	CM01	R/W	—	—	○	
FF73H				—	—		
FF74H	Compare register 02	CM02		—	—	○	
FF75H				—	—		
FF76H	Compare register 03	CM03		—	—	○	
FF77H				—	—		
FF7CH	Compare register 10	CM10		—	—	○	
FF7DH				—	—		
FF7EH	Compare register 11	CM11		—	—	○	
FF7FH				—	—		
FF80H	Clocked serial interface mode register	CSIM		○	○	—	00H
FF82H	Serial bus interface control register	SBIC		○	○	—	
FF86H	Serial I/O shift register	SIO		○	○	—	Undefined

Note Upper or lower half of 18-bit register.

Table 2-2. List of Special Function Registers (3/4)

Address	Special Function Register (SFR) Name	Symbol		R/W	Manipulable Bit Unit 1 bit	Manipulable Bit Unit 8 bits	Manipulable Bit Unit 16 bits	On Reset
FF88H	Asynchronous serial interface mode register	ASIM		R/W	○	○	—	80H
FF8AH	Asynchronous serial interface status register	ASIS		R	○	○	—	00H
FF8CH	Serial receive buffer :UART	RXB			—	○	—	Undefined
FF8EH	Serial send shift register :UART	TXS		W	—	○	—	
FFB0H	Timer control register	TMC		R/W	○	○	—	00H
FFB1H	Baud rate generator mode register	BRGM			○	○	—	
FFB2H	Prescalar mode register	PRM			○	○	—	
FFB8H	Timer output control register 0	TOC0			○	○	—	
FFB9H	Timer output control register 1	TOC1			○	○	—	
FFBFH	RPU mode register	RPUM			○	○	—	
FFC0H	Standby control register	STBC		R/W Note	○	○	—	0000 ×000B
FFC1H	CPU control word	CCW		R/W	○	○	—	00H
FFC2H	Watchdog timer mode register	WDM		R/W Note	○	○	—	
FFC4H	Memory expansion mode register	MM		R/W	○	○	—	
FFC6H	Programmable wait control register	PWC			○	○	—	22H
FFC9H	Fetch cycle control register	FCC			○	○	—	00H
FFD0H to FFDFH	External acces area				○	○	—	Undefined
FFE0H	Interrupt request flag rgister 0L	IF0L	IF0		○	○	○	00H
FFE1H	Interrupt request flag rgister 0H	IF0H			○	○		
FFE2H	Interrupt request flag rgister 1L	IF1L	IF1		○	○	○	
FFE3H	—	—			—	—		—
FFE4H	Interrupt mask flag rgister 0L	MK0L	MK0		○	○	○	FFH
FFE5H	Interrupt mask flag rgister 0H	MK0H			○	○		
FFE6H	Interrupt mask flag rgister 1L	MK1L	MK1		○	○	○	×××× ×111B
FFE7H	—	—			—	—		—
FFE8H	Priority specify bufer register 0L	PB0L	PB0		○	○	○	00H
FFE9H	Priority specify bufer register 0H	PB0H			○	○		
FFEAH	Priority specify bufer register 1L	PB1L	PB1		○	○	○	
FFEBH	—	—			—	—		—
FFECH	Interrupt servicing mode specify register 0L	ISM0L	ISM0		○	○	○	00H
FFEDH	Interrupt servicing mode specify register 0H	ISM0H			○	○		
FFEEH	Interrupt servicing mode specify register 1L	ISM1L	ISM1		○	○	○	
FFEFH	—	—			—	—		—

Note Write enable in case of special instructions.

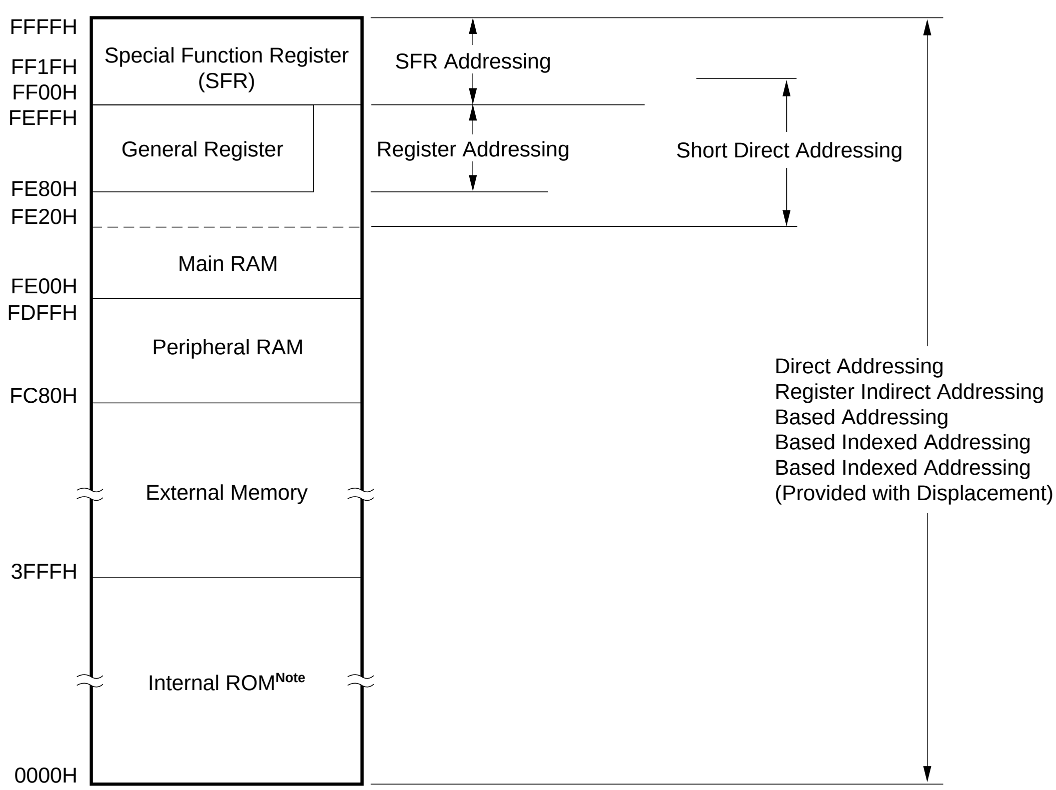
Table 2-2. List of Special Function Registers (4/4)

Address	Special Function Register (SFR) Name	Symbol		R/W	Manipulable Bit Unit			On Reset
					1 bit	8 bits	16 bits	
FFF0H	Context switching enable register 0L	CSE0L	CSE0	R/W	○	○	○	00H
FFF1H	Context switching enable register 0H	CSE0H			○	○		
FFF2H	Context switching enable register 1L	CSE1L	CSE1		○	○	○	
FFF3H	—	—			—	—		—
FFF4H	External interupt mode register 0	INTM0			○	○	—	00H
FFF5H	External interupt mode register 1	INTM1			○	○	—	
FFF8H	In-service priority register	ISPR		R	—	○	—	
FFF9H	Priority specify register	PRSL		R/W	○	○	—	

2.3 DATA MEMORY ADDRESSING

In the μPD78322, the internal RAM space (FC80H to FEFFH) and the special function register area (FF00H to FFFFH) are mapped in the FC80H to FFFFH area. In the FE20H to FF1FH space of the data memory, short direct addressing enables direct addressing by 1-byte data in an instruction word.

Figure 2-6. Data Memory Addressing



Note When $\overline{EA}$ = L, or with the μPD78320, this is external memory.

Caution For word access (including stack operations) to the main RAM area (FE00H-FEFFH), the address that specifies the operand must be an even value. ★

2.3.1 General Register Addressing

The general registers consist of eight register banks, each consisting of sixteen 8-bit registers or eight 16-bit registers.

General register addressing is carried out using the register specify field of 3 or 4 bits supplied from an instruction word, the register bank select flag (RBS0 to RBS2) and the register set select flag (RSS) in the PSW.

2.3.2 Short Direct Addressing

Short direct addressing which enables direct address specification by 1-byte data in an instruction word is applied to the FE20H to FF1FH space. The short direct memory is accessed as 8-bit or 16-bit data. When accessing the memory as 16-bit data, specification of even data for 1-byte address specify data will cause 2-byte data specified by continuous addresses of even and odd addresses to be accessed. (Do not specify odd number for address specify data.)

2.3.3 Special Function Register (SFR) Addressing

This addressing is applied to operations for the special function register (SFR) mapped in the SFR area of FF00H to FFFFH. Addressing is performed by 1-byte data in the instruction word corresponding to the lower 8 bits of the special function register address. For 16-bit access of 16-bit manipulable SFR, 2-byte data specified by continuous even and odd addresses is accessed as is the case with short direct addressing.

3. BLOCK FUNCTIONS

3.1 BUS CONTROL UNIT (BCU)

In the BCU, the necessary bus cycle is started according to the physical address obtained by the execution unit (EXU). If no bus cycle startup request is made from the EXU, a prefetch address is generated and instruction prefetch is carried out. The prefetched operation code is fetched into the instruction queue.

3.2 EXECUTION UNIT (EXU)

In the EXU, address calculation, arithmetic logical operation and data transfer are controlled by microprograms. A 256-byte RAM is built in the EXU.

The 256-byte main RAM in the EXU is accessible by the relevant instruction faster than peripheral RAM (384 bytes).

3.3 ROM/RAM

This block consists of a 16K-byte ROM and a 384-byte peripheral RAM. However, the μPD78320 does not incorporate ROM.

ROM access can be disabled by $\overline{\text{EA}}$ pin.

3.4 INTERRUPT CONTROLLER

Various interrupt requests (NMI, INTP0 to INTP6) generated either externally or from the peripheral hardware are serviced by the context switch, vectored interrupt or macro service function.

The 3-level interrupt priority is also specified.

3.5 PORT FUNCTIONS

Table 3-1 lists the digital input/output ports.
Each port can carry out many control operations including 8 and other bit data input/output manipulations.

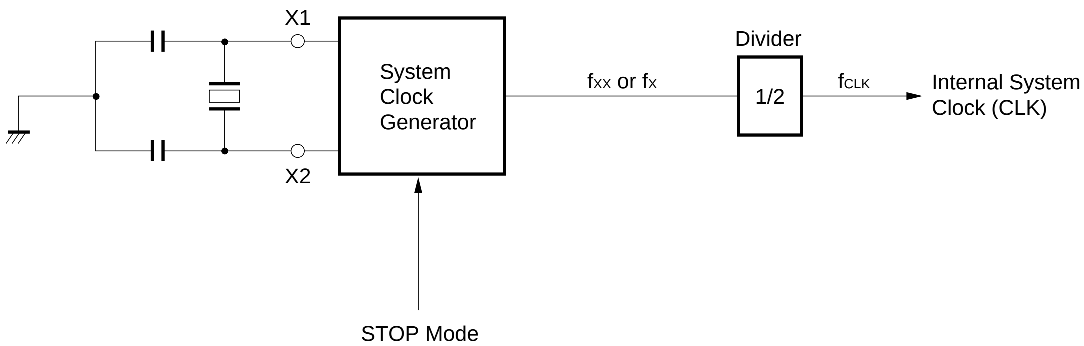
Table 3-1. Port Functions and Features

Port Name	Function	Feature	Remarks
Port 0	8-bit input/outpput	Specifiable bit-wise for input/output. Also specifiable for realtime output port..	Dual-function as pins RTP0 to RTP7
Port 2	8-bit input	Input port pin. Functions as an external interrupt input.	Dual-function as pins NMI, INTP0 to INTP5, INTP6/TI
Port 3	5-bit input/output	Specifiable bit-wise for port pins or control pins.	Dual-function as pins TxD, RxD, SO/SB0, SI/SB1, $\overline{\text{SCK}}$
Port 4	8-bit input/output	Specifiable in 8-bit units for input or output. Functions as the multiplexed address/data bus (AD0 to AD7) in the external memory expansion mode.	————
Port 5	8-bit input/output	Specifiable bit-wise for input or output. Functions as the address bus (A8 to A15) in the external memory expansion mode. Pins which are not used as the address bus can be used as a port.	————
Port 7	8-bit input	Input port pin. Also functions as analog input to the A/D converter.	Dual-function as pins AN0 to AN7
Port 8	6-bit input/output	Specifiable bit-wise for the port pin or control pin.	Dual-function as pins TO00 to TO03, TO10 to TO11
Port 9	4-bit input/output	Specifiable bit-wise for input/output. P90 and P91 function as $\overline{\text{RD}}$ output and $\overline{\text{WR}}$ output, respectively, in the external memory expansion mode. P92 and P93 function as $\overline{\text{TAS}}$ output and TMD output, respectively, in the external memory high-speed fetch mode.	————

3.6 CLOCK GENERATOR

The clock generator generates and controls internal system clocks (CLK) supplied to the CPU.
It is configured as shown in Figure 3-1.

Figure 3-1. Clock Generator Block Diagram



Remarks **1.** f_{XX} : Crystal oscillator frequency
2. f_X : External clock frequency
3. f_{CLK} : Internal system clock frequency

The system clock oscillator oscillates by a crystal resonator connected to X1 and X2 pins. It stops oscillating when set to the standby mode (STOP).

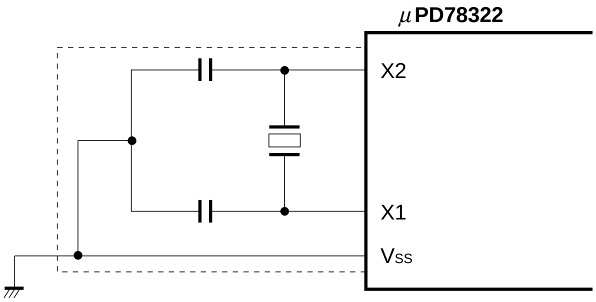
External clocks can be input to the system clock oscillator. In such cases, input a clock signal to the X1 pin and input the inverted clock signal to the X2 pin. The X2 pin can also be left open.

Caution When using external clocks, do not set the STBC STP bit.

The divider generates internal system clocks (f_{CLK}) by dividing a system clock oscillator output (fxx for crystal oscillation and fx for external clocks) into two parts.

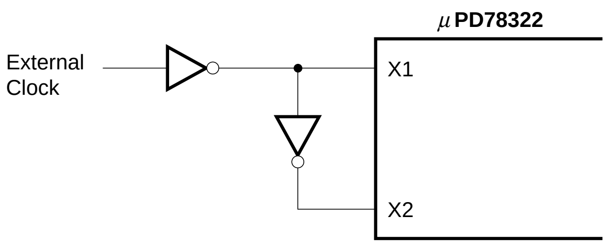
Figure 3-2. Externally-Mounted System Clock Oscillator

(a) Crystal oscillator

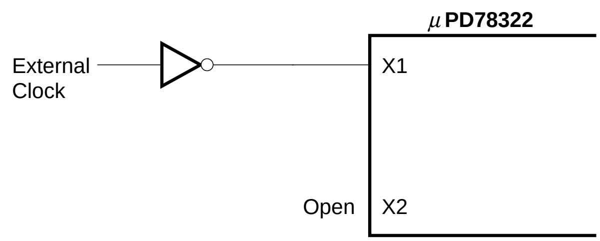


(b) External clock

(i) When the inverted phase of an external clock to be input to the X1 pin is input to the X2 pin



(ii) When X2 pin is left open



Cautions 1. When the system clock oscillator is used, the following points should be noted concerning wiring within broken lines shown in Figure 3-2, in order to prevent the effects of wiring capacitance, etc.

- **Keep the wiring as short as possible.**
- **Do not cross any other signal lines, and keep clear of lines in which a high fluctuating current flows.**
- **Ensure that oscillator capacitor connection points are always at the same potential as V_{SS}. Do not ground in a ground pattern in which a high current flows.**
- **Do not take a signal from the oscillator.**

2. When an external clock is input to the X1 pin and the X2 pin is left open, ensure that no loads such as wiring capacitance are connected to the X2 pin.

3.7 REALTIME PULSE UNIT (RPU)

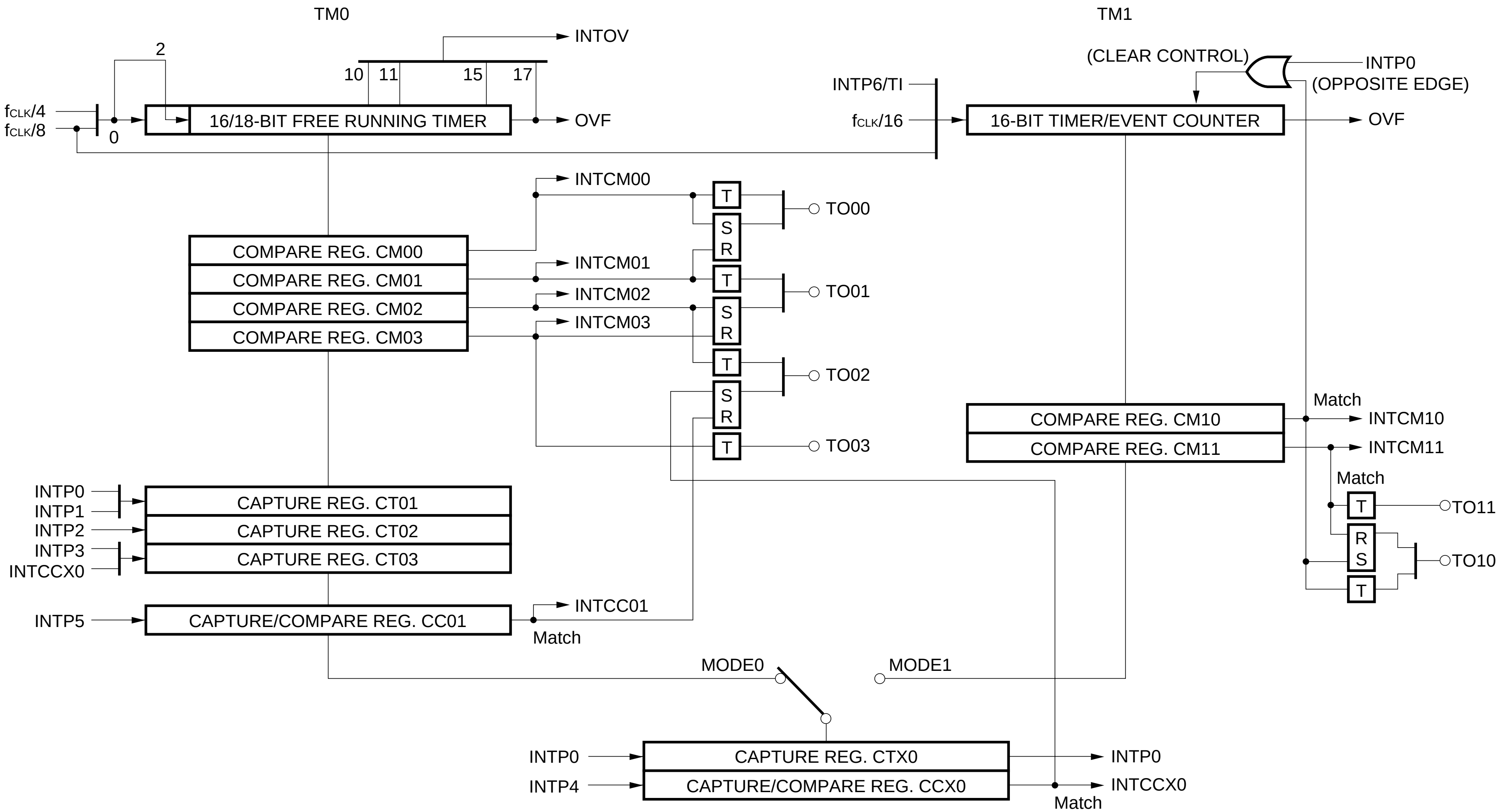
This unit can measure pulse intervals and frequencies, and generate programmable pulse outputs.

It consists mainly of two timers. To flexibly cope with many applications, the configuration of registers connected to the timers can be changed using programs. To meet various applications, toggle output (6 max.) or set/reset output (4 max.) can be selected as timer output.

3.7.1 Configuration

The realtime pulse unit is configured mainly of timer 0 (TM0) which functions as a 16-bit or 18-bit free running timer and timer 1 (TM1) which functions as a 16-bit timer/event counter shown in Figure 3-3.

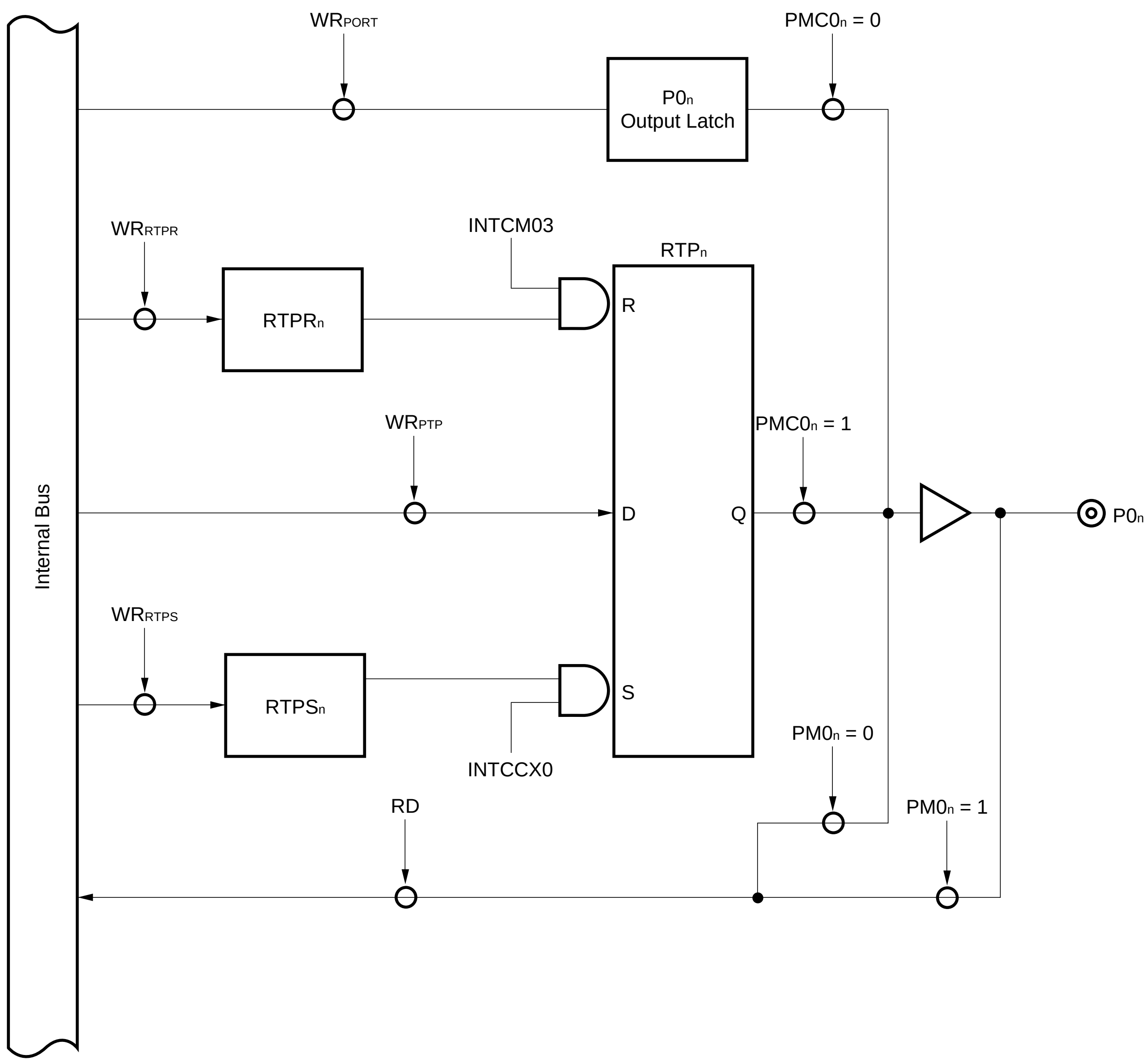
Figure 3-3. Realtime Pulse Unit Configuration



3.7.2 Realtime Output Function

The realtime output port can set/reset port outputs bit-wise in synchronization with the trigger signal transmitted from the RPU (Realtime Pulse Unit). It enables to generate multi-channel synchronous pulses easily.

Figure 3-4. Realtime Output Port

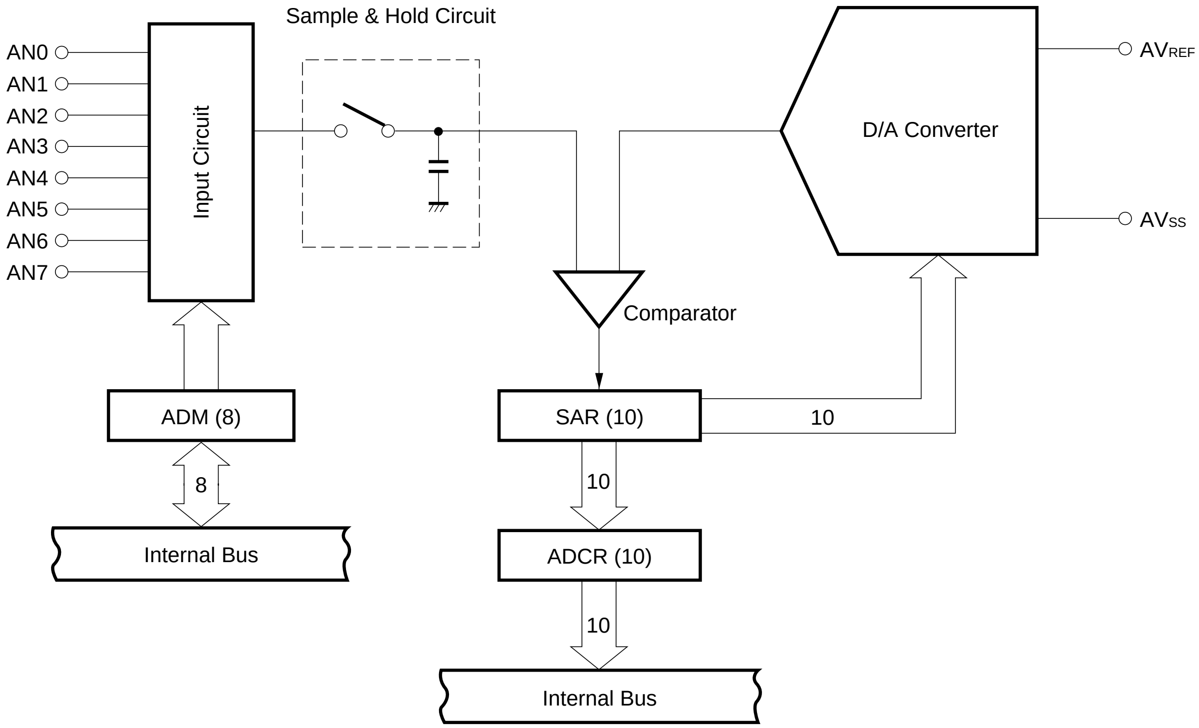


3.8 A/D CONVERTER

The μPD78322 incorporates a high-speed, high-resolution 10-bit analog/digital (A/D) converter. This A/D converter is equipped with eight analog inputs (AN0 to AN7) and A/D conversion result register (ADCR) which holds the conversion results.

Upon termination of conversion, the interrupt which can start the macro service is generated.

Figure 3-5. A/D Converter Block Diagram



3.9 SERIAL INTERFACE

The μPD78322 is equipped with the following two independent channels for the serial interface function.

- Asynchronous serial interface
- Clocked serial interface
  - 3-wire serial I/O mode
  - Serial bus interface mode (SBI mode)

Since the μPD78322 incorporates a baud rate generator, it can set any serial transfer rate irrespective of the operating frequency. The baud rate generator functions for the 2-channel serial interface in common.

The serial transfer rate can be selected from 75 bps to 19.2 Kbps by setting the mode register.

Figure 3-6. Asynchronous Serial Interface Block Diagram

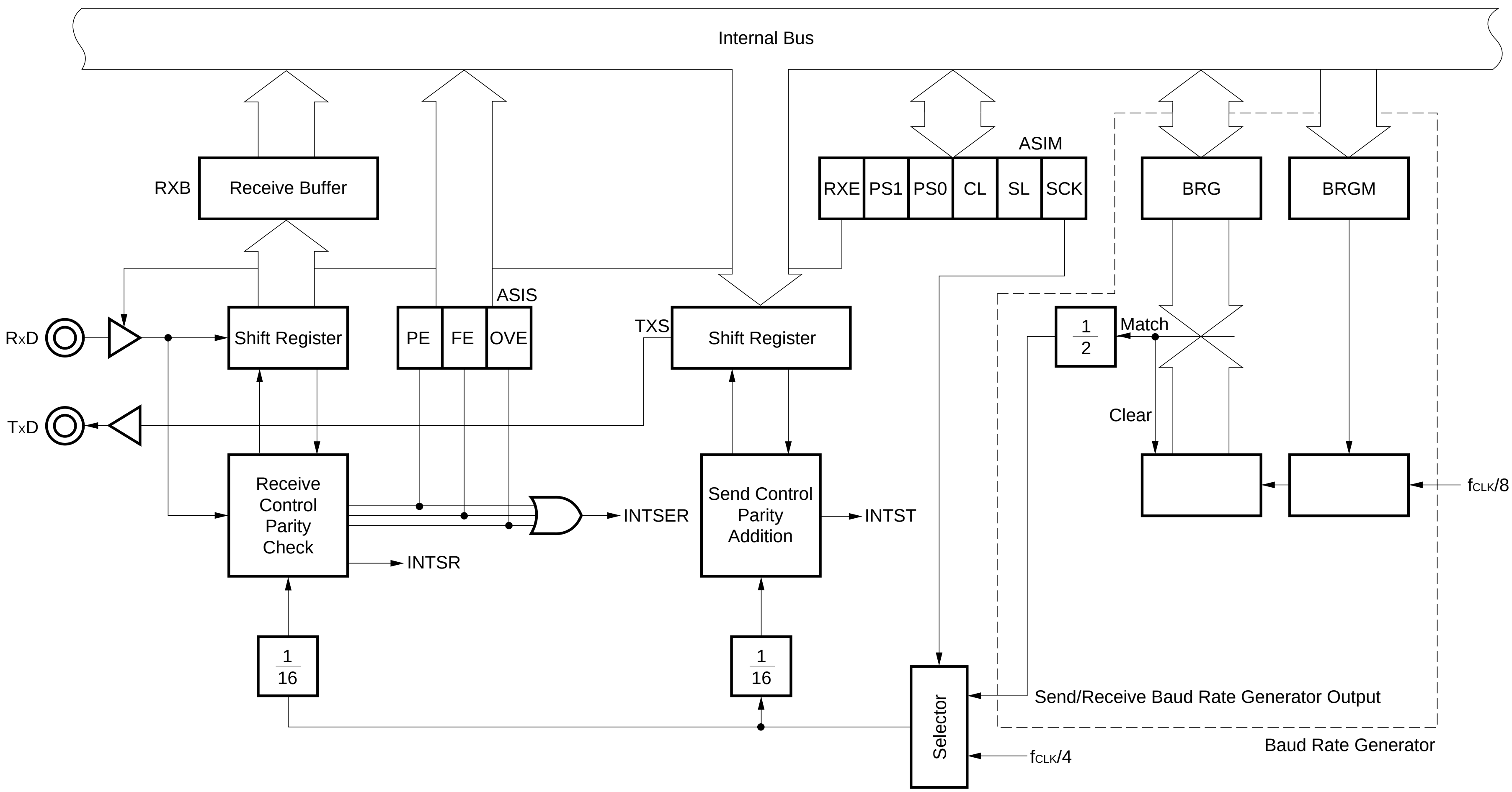
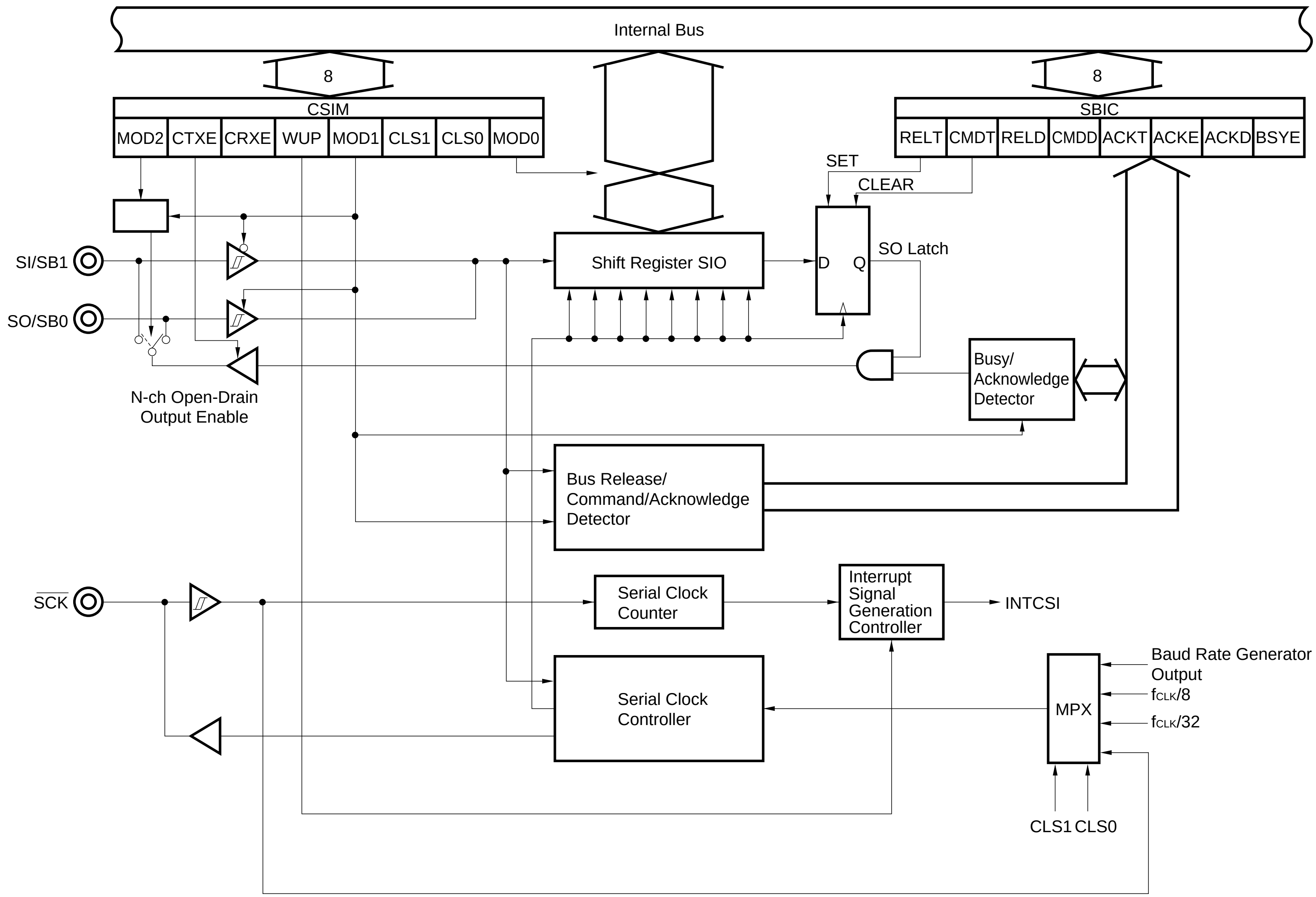


Figure 3-7. Block Diagram of Clocked Serial Interface



3.10 WATCHDOG TIMER

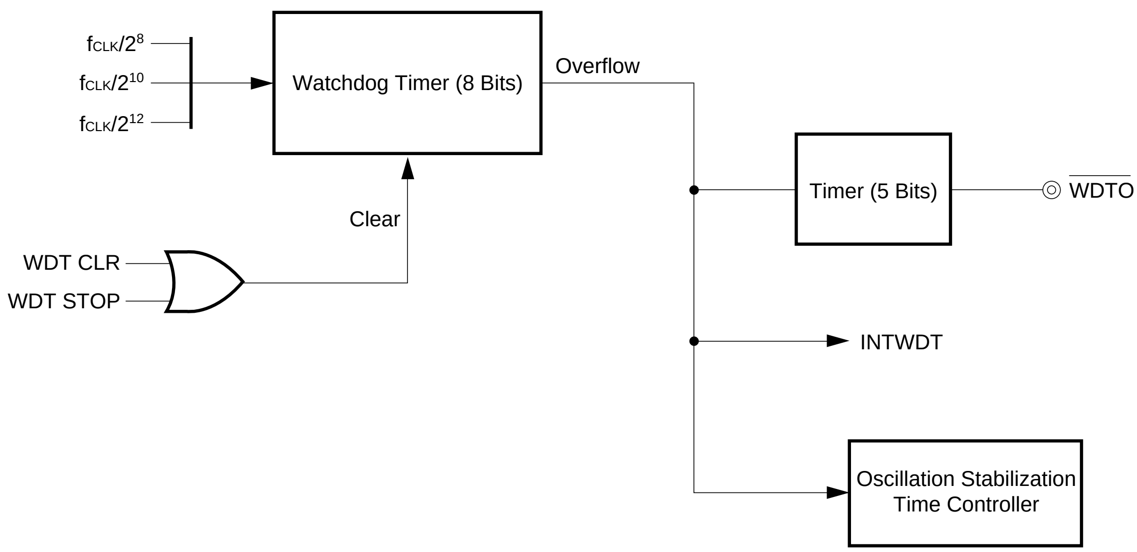
The watchdog timer is used to prevent program overrun and deadlock. Normal operation of the program or system can be confirmed by checking that no watchdog timer interrupt has been generated. Thus, an instruction to clear the watchdog timer (timer start) is set into each program module.

If the watchdog timer clear instruction is not cleared within the time period set into the watchdog timer and the watchdog timer overflows, a watchdog timer interrupt is generated, and a low level is generated to $\overline{\text{WDTO}}$ pin, thereby notifying of an error in the program.

The watchdog timer can also be used to maintain the oscillation stabilization time of the oscillator after the stop mode has been released.

Figure 3-8 shows the watchdog timer configuration.

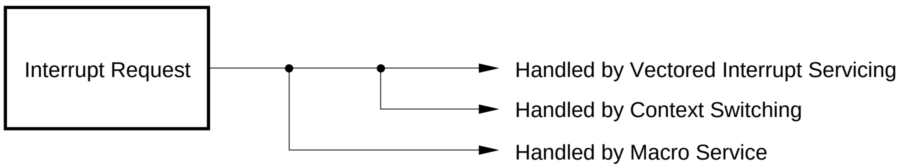
Figure 3-8. Watchdog Timer Configuration



4. INTERRUPT FUNCTIONS

4.1 OVERVIEW

In the μPD78322, various interrupt requests generated externally or from the on-chip peripheral hardware are handled in the following three servicing modes.



Interrupt requests are classified into the following three groups.

- Nonmaskable interrupt requests
- Maskable interrupt requests
- Interrupt requests by software

Figure 4-1 shows the maskable interrupt request servicing modes. Table 4-1 gives a listing of interrupt factors which can be serviced.

Figure 4-1. Interrupt Request Servicing Modes

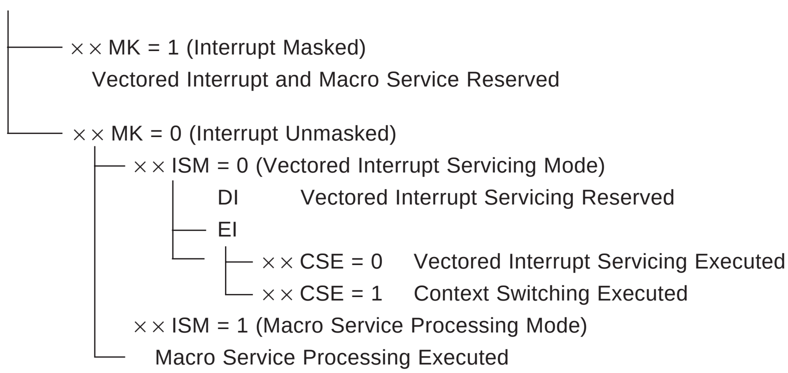


Table 4-1. List of Interrupt Factors

Interrupt Request Type	Default Priority	Interrupt Factor		Generator Unit	Macro Servicez	Vector Table Address
		Request Signal	Function			
Software	——	——	BRK instruction	——	——	003EH
	——	——	Operation code trap	——	——	003CH
Non-maskable	——	NMI	NMI pin input	(External interrupt)	——	0002H
	——	INTWDT	Watchdog timer	(WDT)	——	0004H
Maskable	0	INTOV	Timer 0 overflow	(RPU)	Available	0006H
	1	INTP0	INTP0 pin input	(External)		0008H
	2	INTP1	INTP1 pin input	(External)		000AH
	3	INTP2	INTP2 pin input	(External)		000CH
	4	INTP3	INTP3 pin input	(Exteranl)		000EH
	5	INTP4/INTCCX0	INTP4 pin input/CCX0 match signal	(RPU/exteranl)		0010H
	6	INTP5/INTCC01	INTP5 pin input/CC01 match signal	(RPU/exteranl)		0012H
	7	INTP6/TI	INTP6 pin input/TI input	(Exteranl)		0014H
	8	INTCM00	CM00 match signal	(RPU)		0016H
	9	INTCM01	CM01 match signal	(RPU)		0018H
	10	INTCM02	CM02 match signal	(RPU)		001AH
	11	INTCM03	CM03 match signal	(RPU)		001CH
	12	INTCM10	CM10 match signal	(RPU)		001EH
	13	INTCM11	CM11 match signal	(RPU)		0020H
	14	INTSR	Serial receive terminate interrupt	(UART)		0024H
	15	INTST	Serial send terminate interrupt	(UART)		0026H
	16	INTCSI	Serial send/receive interrupt	(CSI)		0028H
	17	INTAD	A/D conversion terminate interrupt	(A/D)		002AH
——	——	INTSER[Note]	Serial receive error signal	(UART)	——	——[Note]
Reset	——	RESET	Reset input	——	——	0000H

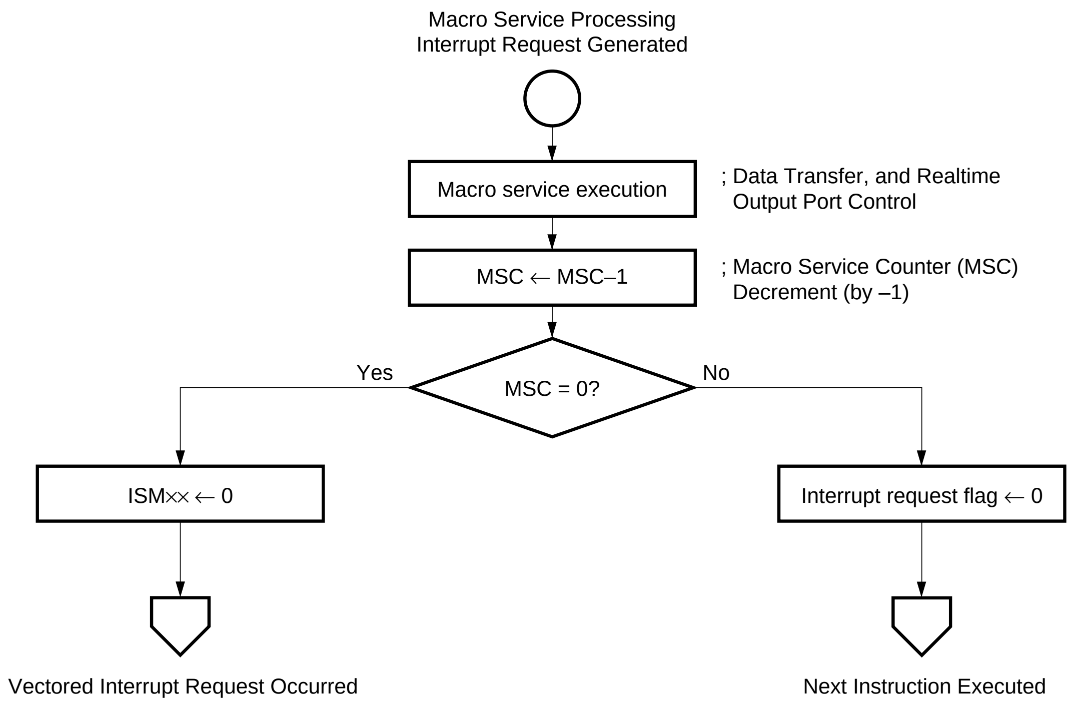
Note This is a test factor. A vectored interrupt is not generated.

4.2 MACRO SERVICE

The macro service function is executed at the interrupt request to carry out data operation and data transfer in hardware terms between the special function register area and the memory space.

Upon startup of the macro service, the CPU stops program execution temporarily. 1-byte/2-byte data operation, transfer, etc. are automatically carried out between the special function register (SFR) and the memory. Upon termination of the macro service, the interrupt request flag is reset to 0 and the CPU restarts program execution. When the CPU carries out the macro service operations as many as set into the macro service counter (MSC), a vectored interrupt request is generated after completion.

★ **Figure 4-2. Macro Service Processing Sequence Example**



4.3 CONTEXT SWITCHING FUNCTION

This is the function to first select the specified register bank in hardware terms by generating an interrupt request or executing BRKCS instruction, to branch the selected register bank to the vector address prestored in the register bank, and also to stack the current PC and PSW contents into the register bank.

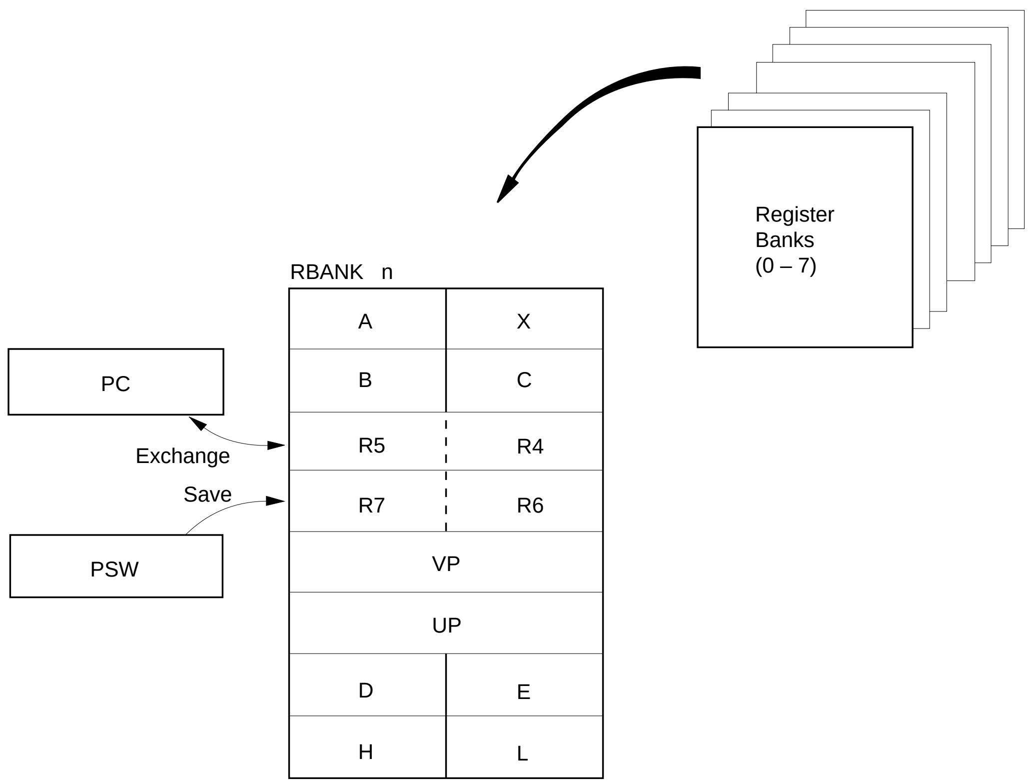
4.3.1 Context Switching Function by Interrupt Request

The context switching function start is enabled by setting the ××CSE bit preset at each interrupt request to 1.

If an unmasked interrupt request for which the context switching function has been enabled is generated in the EI state, the register bank which is specified by the lower 3 bits of the low address (even address) of the corresponding interrupt vector table address is selected. The vector address prestored in the selected register bank is transferred to the PC, the PC and PSW contents are saved into the register bank, and the operation is branched to the interrupt service routine.

Return is by means of executing the RETCS instruction.

Figure 4-3. Context Switching by Interrupt Request Generation



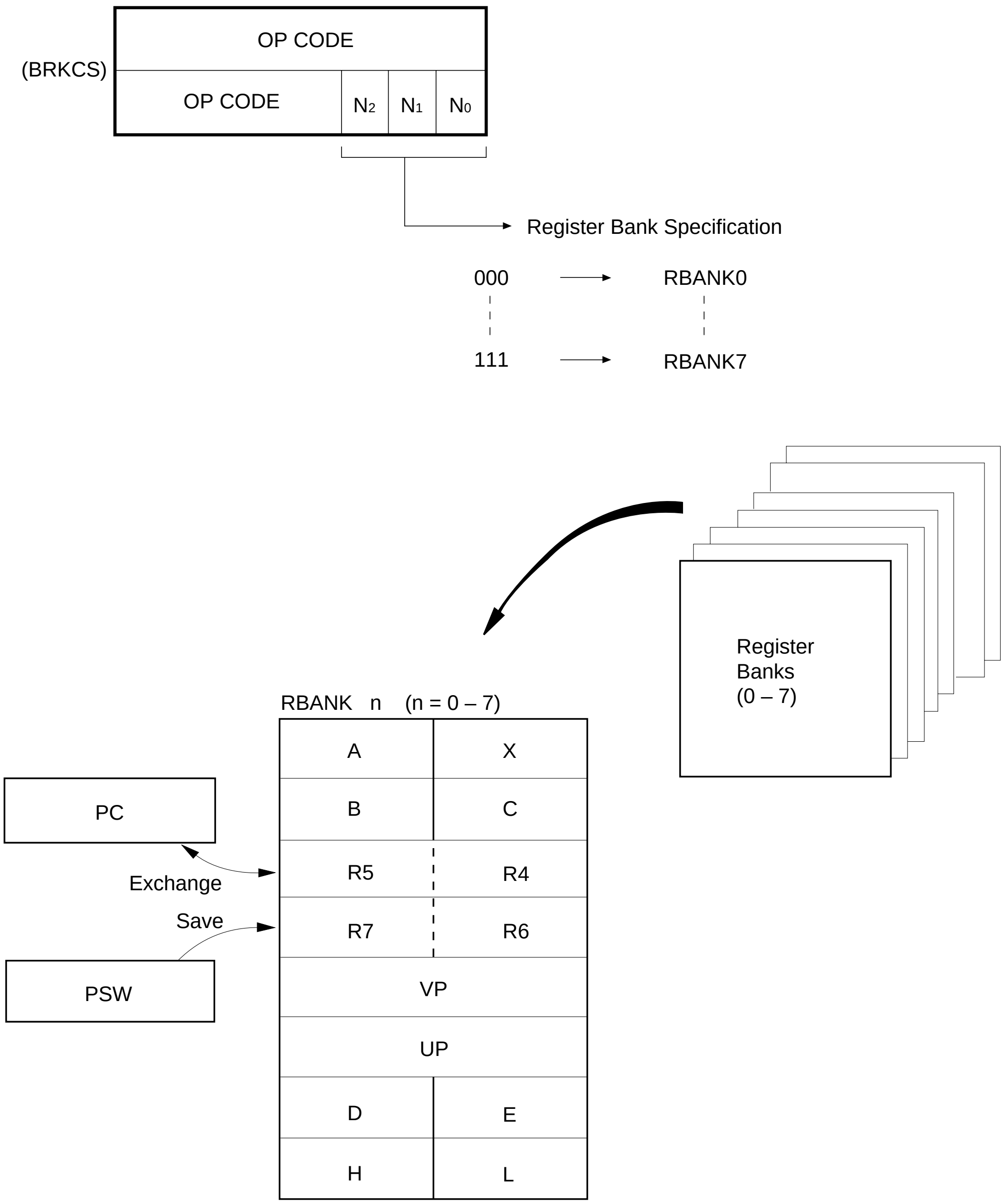
4.3.2 Context Switching Function by BRKCS Instruction

The context switching function can be started by executing BRKCS instruction.

The context switched register bank is specified by the lower 3-bit immediate data of the 2nd operation code of BRKCS instruction. When BRKCS instruction is executed, the register bank specified by the 3-bit immediate data is selected, the vector address prestored in the register bank is set and branched to the PC, and the PC and PSW contents are saved into the register bank.

Return is by means of executing the RETCSB instruction.

Figure 4-4. Context Switching by Execution of BRKCS Instruction



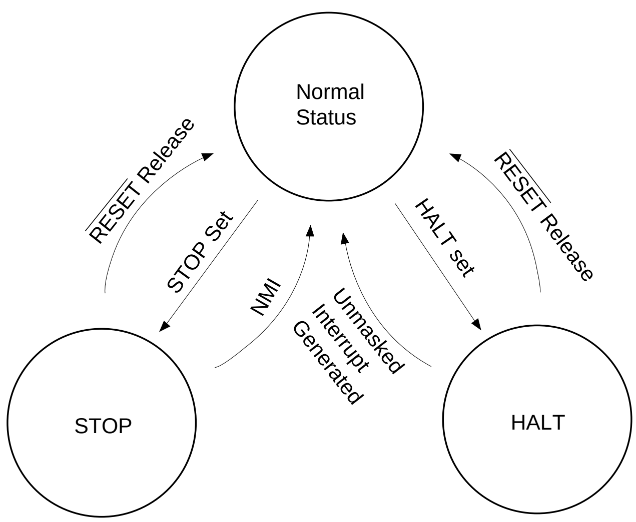
5. STANDBY FUNCTIONS

The μPD78322 has the standby function to decrease the power consumption of the system. The following two modes are available for execution of the standby function.

- HALT mode .... Mode for halting the CPU operation clock. The total power consumption of the system can be decreased by intermittent operation in combination with the normal operating mode.
- STOP mode .... Mode for stopping the whole system by stopping the oscillator. Considerably low power consumption with leak current only can be set.

Each mode is set by the software. Figure 5-1 shows standby mode (STOP/HALT mode) transition.

Figure 5-1. Standby Status Transition



6. EXTERNAL DEVICE EXPANSION FUNCTION

The μPD78322 can expand external devices (data memory, program memory or peripheral device) for areas (4000H to FFFFH) except the internal ROM and RAM areas. Tables 6-1 through 6-3 shows the pin used for external device access and the pin function setting procedure.

Table 6-1. Pin Function Setting (μPD78322)

$\overline{EA}$ Pin	Memory Expansion Mode Register		Fetch Cycle Control Register	Pin Function						Remarks
	MM0 to MM2	MM7		P40 to P47	P50 to P57	P90	P91	P92	P93	
1	Port mode	0	00H	General port						——
		1	Setting prohibited							
	Expansion mode	0	00H	AD0 to AD7	Set to A8 to A15 in steps	$\overline{RD}$	$\overline{WR}$	General-purpose port		External device connection mode
		1	Except 00H					$\overline{TAS}$	TMD	μPD71P301 connection mode

For P50 to P57 pins, the number of bits which serve as address buses can be changed according to the externally expanded memory size. The memory can be expanded in steps from 256 bytes to about 48K bytes. The pins which are not used as the address bus can be used as the general-purpose input/output port.

Table 6-2. Port and Address Setting for Port 5 (μPD78322)

P57	P56	P55	P54	P53	P52	P51	P50	External Address Space
Port	Port	Port	Port	Port	Port	Port	Port	256 bytes or less
Port	Port	Port	Port	A11	A10	A9	A8	4K bytes or less
Port	Port	A13	A12	A11	A10	A9	A8	16K bytes or less
A15	A14	A13	A12	A11	A10	A9	A8	About 48K bytes or less

Table 6-3. Setting Pin Function (μPD78320)

$\overline{EA}$ Pin	Memory Expansion Mode Register	Fetch Cycle Control Register	Pin Function						Remarks
	MM7		AD0 to AD7	A8 to A15	$\overline{RD}$	$\overline{WR}$	P92	P93	
ASTB	——	——	AD to AD7	A8 to A15	$\overline{RD}$	$\overline{WR}$	$\overline{TAS}$	TMD	μPD78322 emulation mode
0	0	00H					General-purpose port		External device connection mode
	1	Except 00H					$\overline{TAS}$	TMD	μPD71P301 connection mode

7. OPERATION AFTER RESET

If the $\overline{\text{RESET}}$ input pin is set to the low level, the system reset is applied and each hardware becomes as initialized status (reset status). If $\overline{\text{RESET}}$ input becomes high level, the reset state is released and program execution is started. Initialize the contents of various registers in the program as required.

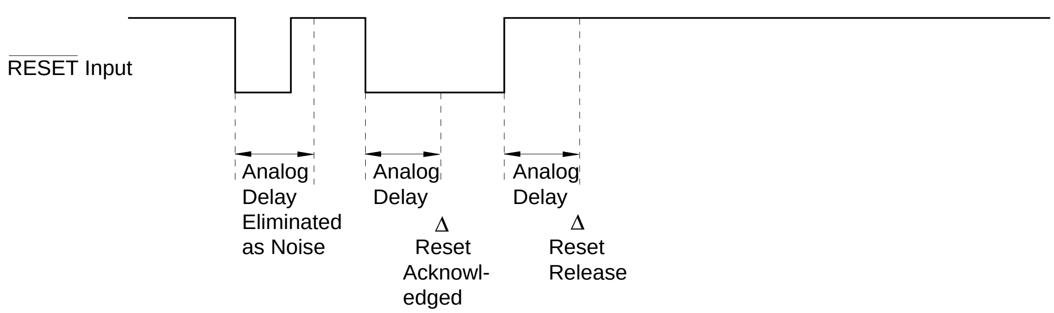
Change the number of cycles for the programmable wait control register and the fetch cycle control register as required in particular.

The $\overline{\text{RESET}}$ input pin is equipped with an analog delay noise eliminator to prevent malfunctioning due to noise.

Cautions 1. While $\overline{\text{RESET}}$ is active (low level), all pins remain high impedance (except $\overline{\text{WDTO}}$, AV_{REF}, AV_{DD}, AV_{SS}, V_{DD}, V_{SS}, X1 and X2).

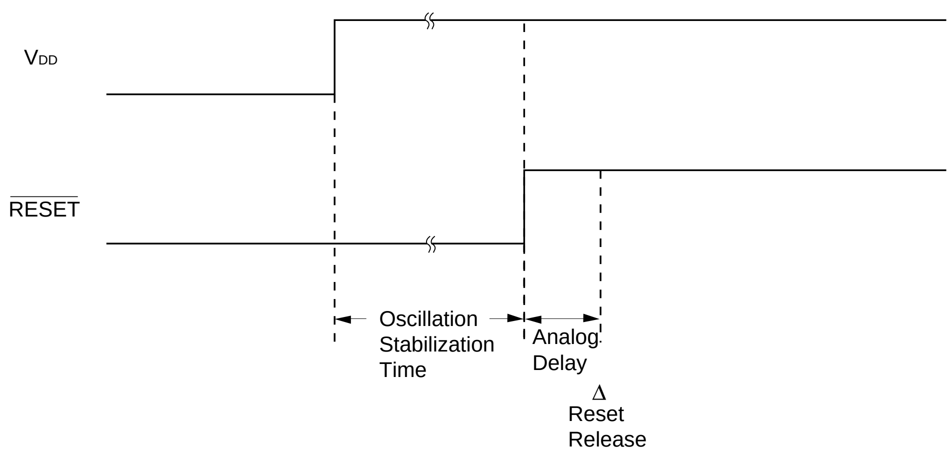
2. If RAM has been expanded externally, mount a pull-up resistor to the P90/$\overline{\text{RD}}$ and P91/$\overline{\text{WR}}$ pins. It is possible that the P90/$\overline{\text{RD}}$ and P91/$\overline{\text{WR}}$ pins become high impedance resulting in an external RAM contents corruption. In addition, signals may collide on the address/data bus, resulting in the destruction of the input/output circuit.

Figure 7-1. Reset Signal Acknowledge



For reset operation upon power-up, secure the oscillation stabilization time of about 40 msec from power-up to reset acknowledge as shown in Figure 7-2.

Figure 7-2. Reset Upon Power-Up



8. INSTRUCTION SET

This chapter covers instruction operations.
For the operation codes and the number of instruction execution clock cycles, see μ**PD78322 User's Manual (IEU-1248)**.

(1) Operand identifier and description method

In each instruction operand field, enter the operand using the description method for the instruction operand identifier (refer to the assembler specification for details). If two or more factors are included in the description method field, select one factor. The capital alphabetic letters and +, -, #, $, ! and [ ] symbols are keywords and should be described as they are.

In case of immediate data, describe appropriate numeric values or labels. When describing labels, make sure to describe #, $, ! and [ ] symbols.

Table 8-1. Operand Identifier and Description Method

Identifier	Description Method
r r1 r2	R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15 R0, R1, R2, R3, R4, R5, R6, R7 C, B
rp rp1 rp2	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7 RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7 DE, HL, VP, UP
sfr sfrp	Special function register code (see **Table 2-2**) Special function register code (16-bit manipulation enable register; see **Table 2-2**)
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7 (Two or more instructions can be described. Only PUSH and POP instructions can be described for RP5 and only PUSHU and POPU instructions can be described for PSW.)
mem	[DE], [HL], [DE+], [HL+], [DE-], [HL-], [VP], [UP] ; Register indirect mode [DE+A], [HL+A], [DE+B], [HL+B], [VP+DE], [VP+HL] ; Based indexed mode [DE+byte], [HL+byte], [VP+byte], [UP+byte], [SP+byte] ; Based mode word[A], word[B], word[DE], word[HL] ; Indexed mode
saddr saddrp	FE20H to FF1FH immediate data or label FE20H to FF1EH immediate data (bit0 = 0) or label (for 16-bit manipulation)
$addr16 !addr16 addr11 addr5	0000H to FDFFH immediate data or label; relative addressing 0000H to FDFFH immediate data or label; immediate addressing (Up to FFFFH describable by MOV instruction) 800H to FFFH immediate data or label 40H to 7EH immediate data (bit0 = 0)[Note] or label
word byte bit n	16-bit immediate data or label 8-bit immediate data or label 3-bit immediate data or label 3-bit immediate data (0 to 7)

Note Do not make word access to bit0 = 1 (odd address).

Remarks
1. Although rp and rp1 have the same describable register names, they generate different codes.
2. r, r1, rp, rp1 and post can be described with absolute names (R0 to R15, RP0 to RP7) as well as functional names (X, A, C, B, E, D, L, H, AX, BC, DE, HL, VP, UP (see **Table 2-1** for details of the relationships between the absolute and functional names).
3. Immediate addressing is enabled for all spaces. Relative addressing is only enabled from the first address of the subsequent instruction to the range of -128 to +127.

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags				
					S	Z	AC	P/V	CY
8-bit data transfer instruction	MOV	r1, #byte	2	r1 ← byte					
		saddr, #byte	3	(saddr) ← byte					
		sfr[Note], #byte	3	sfr ← byte					
		r, r1	2	r ← r1					
		A, r1	1	A ← r1					
		A, saddr	2	A ← (saddr)					
		saddr, A	2	(saddr) ← A					
		saddr, saddr	3	(saddr) ← (saddr)					
		A, sfr	2	A ← sfr					
		sfr, A	2	sfr ← A					
		A, mem	1-4	A ← (mem)					
		mem, A	1-4	(mem) ← A					
		A, [saddrp]	2	A ← ((saddrp))					
		[saddrp], A	2	((saddrp)) ← A					
		A, !addr16	4	A ← (addr16)					
		!addr16, A	4	(addr16) ← A					
		PSWL, #byte	3	PSW_L ← byte	×	×	×	×	×
		PSWH, #byte	3	PSW_H ← byte					
		PSWL, A	2	PSW_L ← A	×	×	×	×	×
		PSWH, A	2	PSW_H ← A					
		A, PSWL	2	A ← PSW_L					
		A, PSWH	2	A ← PSW_H					
	XCH	A, r1	1	A ↔ r1					
		r, r1	2	r ↔ r1					
		A, mem	2-4	A ↔ (mem)					
		A, saddr	2	A ↔ (saddr)					
		A, sfr	3	A ↔ sfr					
		A, [saddrp]	2	A ↔ ((saddrp))					
		saddr, saddr	3	(saddr) ↔ (saddr)					

Note If STBC and WDM are described for sft, a different dedicated instruction having a different number of bytes is used.

Remark For the symbols in the Flags column, refer to the table below.

Symbol	Description
(Blank)	No change
0	Clear to 0.
1	Set to 1.
×	Set/clear according to the result.
P	P/V flag operates as a parity flag
V	P/V flag operates as an overflow flag.
R	The previously saved value is restored.

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags				
					S	Z	AC	P/V	CY
16-bit data transfer instruction	MOVW	rp1, #word	3	rp1 ← word					
		saddrp, #word	4	(saddrp) ← word					
		sfrp, #word	4	sfrp ← word					
		rp, rp1	2	rp ← rp1					
		AX, saddrp	2	AX ← (saddrp)					
		saddrp, AX	2	(saddrp) ← AX					
		saddrp, saddrp	3	(saddrp) ← (saddrp)					
		AX, sfrp	2	AX ← sfrp					
		sfrp, AX	2	sfrp ← AX					
		rp1, !addr16	4	rp1 ← (addr16)					
		!addr16, rp1	4	(addr16) ← rp1					
		AX, mem	2-4	AX ← (mem)					
		mem, AX	2-4	(mem) ← AX					
	XCHW	AX, saddrp	2	AX ↔ (saddrp)					
		AX, sfrp	3	AX ↔ sfrp					
		saddrp, saddrp	3	(saddrp) ↔ (saddrp)					
		rp,rp1	2	rp ↔ rp1					
		AX, mem	2-4	AX ↔ (mem)					
8-bit operation instruction	ADD	A, #byte	2	A, CY ← A + byte	×	×	×	V	×
		saddr, #byte	3	(saddr), CY ← (saddr) + byte	×	×	×	V	×
		sfr, #byte	4	sfr, CY ← sfr + byte	×	×	×	V	×
		r, r1	2	r, CY ← r + r1	×	×	×	V	×
		A, saddr	2	A, CY ← A + (saddr)	×	×	×	V	×
		A, sfr	3	A, CY ← A + sfr	×	×	×	V	×
		saddr, saddr	3	(saddr), CY ← (saddr) + (saddr)	×	×	×	V	×
		A, mem	2-4	A, CY ← A + (mem)	×	×	×	V	×
		mem, A	2-4	(mem), CY ← (mem) + A	×	×	×	V	×
	ADDC	A, #byte	2	A, CY ← A + byte + CY	×	×	×	V	×
		saddr, #byte	3	(saddr), CY ← (saddr) + byte + CY	×	×	×	V	×
		sfr, #byte	4	sfr, CY ← sfr + byte + CY	×	×	×	V	×
		r, r1	2	r, CY ← r + r1 + CY	×	×	×	V	×
		A, saddr	2	A, CY ← A + (saddr) + CY	×	×	×	V	×
		A, sfr	3	A, CY ← A + sfr + CY	×	×	×	V	×
		saddr, saddr	3	(saddr), CY ← (saddr) + (saddr) + CY	×	×	×	V	×
		A, mem	2-4	A, CY ← A + (mem) + CY	×	×	×	V	×
		mem, A	2-4	(mem), CY ← (mem) + A + CY	×	×	×	V	×

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags S	Z	AC	P/V	CY
8-bit operation instruction	**SUB**	A, #byte	2	A, CY ← A – byte	×	×	×	V	×
		saddr, #byte	3	(saddr), CY ← (saddr) – byte	×	×	×	V	×
		sfr, #byte	4	sfr, CY ← sfr – byte	×	×	×	V	×
		r, r1	2	r, CY ← r – r1	×	×	×	V	×
		A, saddr	2	A, CY ← A – (saddr)	×	×	×	V	×
		A, sfr	3	A, CY ← A – sfr	×	×	×	V	×
		saddr, saddr	3	(saddr), CY ← (saddr) – (saddr)	×	×	×	V	×
		A, mem	2-4	A, CY ← A – (mem)	×	×	×	V	×
		mem, A	2-4	(mem), CY ← (mem) – A	×	×	×	V	×
	SUBC	A, #byte	2	A, CY ← A – byte – CY	×	×	×	V	×
		saddr, #byte	3	(saddr), CY ← (saddr) – byte – CY	×	×	×	V	×
		sfr, #byte	4	sfr, CY ← sfr – byte – CY	×	×	×	V	×
		r, r1	2	r, CY ← r – r1 – CY	×	×	×	V	×
		A, saddr	2	A, CY ← A – (saddr) – CY	×	×	×	V	×
		A, sfr	3	A, CY ← A – sfr – CY	×	×	×	V	×
		saddr, saddr	3	(saddr), CY ← (saddr) – (saddr) – CY	×	×	×	V	×
		A, mem	2-4	A, CY ← A – (mem) – CY	×	×	×	V	×
		mem, A	2-4	(mem), CY ← (mem) – A – CY	×	×	×	V	×
	AND	A, #byte	2	A ← A ∧ byte	×	×		P	
		saddr, #byte	3	(saddr) ← (saddr) ∧ byte	×	×		P	
		sfr, #byte	4	sfr ← sfr ∧ byte	×	×		P	
		r, r1	2	r ← r ∧ r1	×	×		P	
		A, saddr	2	A ← A ∧ (saddr)	×	×		P	
		A, sfr	3	A ← A ∧ sfr	×	×		P	
		saddr, saddr	3	(saddr) ← (saddr) ∧ (saddr)	×	×		P	
		A, mem	2-4	A ← A ∧ (mem)	×	×		P	
		mem, A	2-4	(mem) ← (mem) ∧ A	×	×		P	

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags S	Z	AC	P/V	CY
8-bit operation instruction	**OR**	A, #byte	2	A ← A ∨ byte	×	×		P	
		saddr, #byte	3	(saddr) ← (saddr) ∨ byte	×	×		P	
		sfr, #byte	4	sfr ← sfr ∨ byte	×	×		P	
		r, r1	2	r ← r ∨ r1	×	×		P	
		A, saddr	2	A ← A ∨ (saddr)	×	×		P	
		A, sfr	3	A ← A ∨ sfr	×	×		P	
		saddr, saddr	3	(saddr) ← (saddr) ∨ (saddr)	×	×		P	
		A, mem	2-4	A ← A ∨ (mem)	×	×		P	
		mem, A	2-4	(mem) ← (mem) ∨ A	×	×		P	
	XOR	A, #byte	2	A ← A ⊻ byte	×	×		P	
		saddr, #byte	3	(saddr) ← (saddr) ⊻ byte	×	×		P	
		sfr, #byte	4	sfr ← sfr ⊻ byte	×	×		P	
		r, r1	2	r ← r ⊻ r1	×	×		P	
		A, saddr	2	A ← A ⊻ (saddr)	×	×		P	
		A, sfr	3	A ← A ⊻ sfr	×	×		P	
		saddr, saddr	3	(saddr) ← (saddr) ⊻ (saddr)	×	×		P	
		A, mem	2-4	A ← A ⊻ (mem)	×	×		P	
		mem, A	2-4	(mem) ← (mem) ⊻ A	×	×		P	
	CMP	A, #byte	2	A – byte	×	×	×	V	×
		saddr, #byte	3	(saddr) – byte	×	×	×	V	×
		sfr, #byte	4	sfr – byte	×	×	×	V	×
		r, r1	2	r – r1	×	×	×	V	×
		A, saddr	2	A – (saddr)	×	×	×	V	×
		A, sfr	3	A – sfr	×	×	×	V	×
		saddr, saddr	3	(saddr) – (saddr)	×	×	×	V	×
		A, mem	2-4	A – (mem)	×	×	×	V	×
		mem, A	2-4	(mem) – A	×	×	×	V	×

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags S	Z	AC	P/V	CY
16-bit operation instruction	**ADDW**	AX, #word	3	AX, CY ← AX + word	×	×	×	V	×
		saddrp, #word	4	(saddrp), CY ← (saddrp) + word	×	×	×	V	×
		sfrp, #word	5	sfrp, CY ← sfrp + word	×	×	×	V	×
		rp, rp1	2	rp, CY ← rp + rp1	×	×	×	V	×
		AX, saddrp	2	AX, CY ← AX + (saddrp)	×	×	×	V	×
		AX, sfrp	3	AX, CY ← AX + sfrp	×	×	×	V	×
		saddrp, saddrp	3	(saddrp), CY ← (saddrp) + (saddrp)	×	×	×	V	×
	SUBW	AX, #word	3	AX, CY ← AX – word	×	×	×	V	×
		saddrp, #word	4	(saddrp), CY ← (saddrp) – word	×	×	×	V	×
		sfrp, #word	5	sfrp, CY ← sfrp – word	×	×	×	V	×
		rp, rp1	2	rp, CY ← rp – rp1	×	×	×	V	×
		AX, saddrp	2	AX, CY ← AX – (saddrp)	×	×	×	V	×
		AX, sfrp	3	AX, CY ← AX – sfrp	×	×	×	V	×
		saddrp, saddrp	3	(saddrp), CY ← (saddrp) – (saddrp)	×	×	×	V	×
	CMPW	AX, #word	3	AX – word	×	×	×	V	×
		saddrp, #word	4	(saddrp) – word	×	×	×	V	×
		sfrp, #word	5	sfrp – word	×	×	×	V	×
		rp, rp1	2	rp – rp1	×	×	×	V	×
		AX, saddrp	2	AX – (saddrp)	×	×	×	V	×
		AX, sfrp	3	AX – sfrp	×	×	×	V	×
		saddrp, saddrp	3	(saddrp) – (saddrp)	×	×	×	V	×
Multiplication/division instruction	**MULU**	r1	2	AX ← A × r1					
	DIVUW	r1	2	AX(quotient), r1(remainder) ← AX ÷ r1					
	MULUW	rp1	2	AX(higher 16 bits), rp1(lower 16 bits) ← AX × rp1					
	DIVUX	rp1	2	AXDE(quotient), rp1(remainder) ← AXDE ÷ rp1					
Signed multiplication instruction	**MULW**	rp1	2	AX(higher 16 bits), rp1(lower 16 bits) ← AX × rp1					

Instruction Group	Mnemonic	Operand	Bytes	Operation	S	Z	AC	P/V	CY
Increment/decrement instruction	**INC**	r1	1	r1 ← r1 + 1	×	×	×	V	
		saddr	2	(saddr) ← (saddr) + 1	×	×	×	V	
	DEC	r1	1	r1 ← r1 – 1	×	×	×	V	
		saddr	2	(saddr) ← (saddr) – 1	×	×	×	V	
	INCW	rp2	1	rp2 ← rp2 + 1					
		saddrp	3	(saddrp) ← (saddrp) + 1					
	DECW	rp2	1	rp2 ← rp2 – 1					
		saddrp	3	(saddrp) ← (saddrp) – 1					
Shift/rotate instruction	**ROR**	r1, n	2	(CY, $r1_7$ ← $r1_0$, $r1_{m-1}$ ← $r1_m$) × n times				P	×
	ROL	r1, n	2	(CY, $r1_0$ ← $r1_7$, $r1_{m+1}$ ← $r1_m$) × n times				P	×
	RORC	r1, n	2	(CY ← $r1_0$, $r1_7$ ← CY, $r1_{m-1}$ ← $r1_m$) × n times				P	×
	ROLC	r1, n	2	(CY ← $r1_7$, $r1_0$ ← CY, $r1_{m+1}$ ← $r1_m$) × n times				P	×
	SHR	r1, n	2	(CY ← $r1_0$, $r1_7$ ← 0, $r1_{m-1}$ ← $r1_m$) × n times	×	×	0	P	×
	SHL	r1, n	2	(CY ← $r1_7$, $r1_0$ ← 0, $r1_{m+1}$ ← $r1_m$) × n times	×	×	0	P	×
	SHRW	rp1, n	2	(CY← $rp1_0$, $rp1_{15}$← 0, $rp1_{m-1}$← $rp1_m$) × n times	×	×	0	P	×
	SHLW	rp1, n	2	(CY← $rp1_{15}$, $rp1_0$← 0, $rp1_{m+1}$← $rp1_m$) × n times	×	×	0	P	×
	ROR4	[rp1]	2	A_{3-0} ← $(rp1)_{3-0}$, $(rp1)_{7-4}$ ← A_{3-0}, $(rp1)_{3-0}$ ← $(rp1)_{7-4}$					
	ROL4	[rp1]	2	A_{3-0} ← $(rp1)_{7-4}$, $(rp1)_{3-0}$ ← A_{3-0}, $(rp1)_{7-4}$ ← $(rp1)_{3-0}$					
BCD adjustment instruction	**ADJBA**		2	Decimal Adjust Accumulator	×	×	×	P	×
	ADJBS								
Data conversion instruction	**CVTBW**		1	When A_7 = 0, X ← A, A ← 00H When A_7 = 1, X ← A, A ← FFH					

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags				
					S	Z	AC	P/V	CY
Bit manipulation instruction	**MOV1**	CY, saddr. bit	3	CY ← (saddr.bit)					×
		CY, sfr. bit	3	CY ← sfr.bit					×
		CY, A. bit	2	CY ← A.bit					×
		CY, X. bit	2	CY ← X.bit					×
		CY, PSWH. bit	2	CY ← PSW$_H$.bit					×
		CY, PSWL. bit	2	CY ← PSW$_L$.bit					×
		saddr. bit, CY	3	(saddr.bit) ← CY					
		sfr. bit, CY	3	sfr.bit ← CY					
		A. bit, CY	2	A.bit ← CY					
		X. bit, CY	2	X.bit ← CY					
		PSWH. bit, CY	2	PSW$_H$.bit ← CY					
		PSWL. bit, CY	2	PSW$_L$.bit ← CY					
	AND1	CY, saddr. bit	3	CY ← CY ∧ (saddr.bit)					×
		CY, /saddr. bit	3	CY ← CY ∧ $\overline{\text{(saddr.bit)}}$					×
		CY, sfr. bit	3	CY ← CY ∧ sfr.bit					×
		CY, /sfr. bit	3	CY ← CY ∧ $\overline{\text{sfr.bit}}$					×
		CY, A. bit	2	CY ← CY ∧ A.bit					×
		CY, /A. bit	2	CY ← CY ∧ $\overline{\text{A.bit}}$					×
		CY, X. bit	2	CY ← CY ∧ X.bit					×
		CY, /X. bit	2	CY ← CY ∧ $\overline{\text{X.bit}}$					×
		CY, PSWH. bit	2	CY ← CY ∧ PSW$_H$.bit					×
		CY, /PSWH. bit	2	CY ← CY ∧ $\overline{\text{PSW}_H\text{.bit}}$					×
		CY, PSWL. bit	2	CY ← CY ∧ PSW$_L$.bit					×
		CY, /PSWL. bit	2	CY ← CY ∧ $\overline{\text{PSW}_L\text{.bit}}$					×
	OR1	CY, saddr. bit	3	CY ← CY ∨ (saddr.bit)					×
		CY, /saddr. bit	3	CY ← CY ∨ $\overline{\text{(saddr.bit)}}$					×
		CY, sfr. bit	3	CY ← CY ∨ sfr.bit					×
		CY, /sfr. bit	3	CY ← CY ∨ $\overline{\text{sfr.bit}}$					×
		CY, A. bit	2	CY ← CY ∨ A.bit					×
		CY, /A. bit	2	CY ← CY ∨ $\overline{\text{A.bit}}$					×
		CY, X. bit	2	CY ← CY ∨ X.bit					×
		CY, /X. bit	2	CY ← CY ∨ $\overline{\text{X.bit}}$					×
		CY, PSWH. bit	2	CY ← CY ∨ PSW$_H$.bit					×
		CY, /PSWH. bit	2	CY ← CY ∨ $\overline{\text{PSW}_H\text{.bit}}$					×
		CY, PSWL. bit	2	CY ← CY ∨ PSW$_L$.bit					×
		CY, /PSWL. bit	2	CY ← CY ∨ $\overline{\text{PSW}_L\text{.bit}}$					×

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags S	Z	AC	P/V	CY
Bit manipulation instruction	**XOR1**	CY, saddr. bit	3	CY ← CY $\veebar$ (saddr.bit)					×
		CY, sfr. bit	3	CY ← CY $\veebar$ sfr.bit					×
		CY, A. bit	2	CY ← CY $\veebar$ A.bit					×
		CY, X. bit	2	CY ← CY $\veebar$ X.bit					×
		CY, PSWH. bit	2	CY ← CY $\veebar$ PSW_H.bit					×
		CY, PSWL. bit	2	CY ← CY $\veebar$ PSW_L.bit					×
	SET1	saddr. bit	2	(saddr.bit) ← 1					
		sfr. bit	3	sfr.bit ← 1					
		A. bit	2	A.bit ← 1					
		X. bit	2	X.bit ← 1					
		PSWH. bit	2	PSW_H.bit ← 1					
		PSWL. bit	2	PSW_L.bit ← 1	×	×	×	×	×
	CLR1	saddr. bit	2	(saddr.bit) ← 0					
		sfr. bit	3	sfr.bit ← 0					
		A. bit	2	A.bit ← 0					
		X. bit	2	X.bit ← 0					
		PSWH. bit	2	PSW_H.bit ← 0					
		PSWL. bit	2	PSW_L.bit ← 0	×	×	×	×	×
	NOT1	saddr. bit	3	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$					
		sfr. bit	3	sfr.bit ← $\overline{\text{sfr.bit}}$					
		A. bit	2	A.bit ← $\overline{\text{A.bit}}$					
		X. bit	2	X.bit ← $\overline{\text{X.bit}}$					
		PSWH. bit	2	PSW_H.bit ← $\overline{PSW_H\text{.bit}}$					
		PSWL. bit	2	PSW_L.bit ← $\overline{PSW_L\text{.bit}}$	×	×	×	×	×
	SET1	CY	1	CY ← 1					1
	CLR1	CY	1	CY ← 0					0
	NOT1	CY	1	CY ← $\overline{\text{CY}}$					×

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags				
					S	Z	AC	P/V	CY
Call/return instruction	**CALL**	!addr16	3	$(SP-1) \leftarrow (PC+3)_H$, $(SP-2) \leftarrow (PC+3)_L$, $PC \leftarrow addr16$, $SP \leftarrow SP-2$					
	CALLF	!addr11	2	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_{15-11} \leftarrow 00001$, $PC_{10-0} \leftarrow addr11$, $SP \leftarrow SP-2$					
	CALLT	[addr5]	1	$(SP-1) \leftarrow (PC+1)_H$, $(SP-2) \leftarrow (PC+1)_L$, $PC_H \leftarrow$ (TPF, 00000000, addr5+1), $PC_L \leftarrow$ (TPF, 00000000, addr5), $SP \leftarrow SP-2$					
	CALL	rp1	2	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_H \leftarrow rp1_H$, $PC_L \leftarrow rp1_L$, $SP \leftarrow SP-2$					
		[rp1]	2	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_H \leftarrow (rp1+1)$, $PC_L \leftarrow (rp1)$, $SP \leftarrow SP-2$					
	BRK		1	$(SP-1) \leftarrow PSW_H$, $(SP-2) \leftarrow PSW_L$ $(SP-3) \leftarrow (PC+1)_H$, $(SP-4) \leftarrow (PC+1)_L$, $PC_L \leftarrow (003EH)$, $PC_H \leftarrow (003FH)$, $SP \leftarrow SP-4$ $IE \leftarrow 0$					
	RET		1	$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$					
	RETB		1	$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$ $PSW_L \leftarrow (SP+2)$, $PSW_H \leftarrow (SP+3)$ $SP \leftarrow SP+4$	R	R	R	R	R
	RETI		1	$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$ $PSW_L \leftarrow (SP+2)$, $PSW_H \leftarrow (SP+3)$ $SP \leftarrow SP+4$	R	R	R	R	R
Stack manipulation instruction	**PUSH**	sfrp	3	$(SP-1) \leftarrow sfr_H$ $(SP-2) \leftarrow sfr_L$ $SP \leftarrow SP-2$					
		post	2	$\{(SP-1) \leftarrow post_H$, $(SP-2) \leftarrow post_L$, $SP \leftarrow SP-2\}$ × n times[Note]					
		PSW	1	$(SP-1) \leftarrow PSW_H$, $(SP-2) \leftarrow PSW_L$, $SP \leftarrow SP-2$					
	PUSHU	post	2	$\{(UP-1) \leftarrow post_H$, $(UP-2) \leftarrow post_L$, $UP \leftarrow UP-2\}$ × n times[Note]					
	POP	sfrp	3	$sfr_L \leftarrow (SP)$ $sfr_H \leftarrow (SP+1)$ $SP \leftarrow SP+2$					
		post	2	$\{post_L \leftarrow (SP)$, $post_H \leftarrow (SP+1)$, $SP \leftarrow SP+2\}$ × n times[Note]					
		PSW	1	$PSW_L \leftarrow (SP)$, $PSW_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$	R	R	R	R	R
	POPU	post	2	$\{post_L \leftarrow (UP)$, $post_H \leftarrow (UP+1)$, $UP \leftarrow UP+2\}$ × n times[Note]					
	MOVW	SP, #word	4	$SP \leftarrow word$					
		SP, AX	2	$SP \leftarrow AX$					
		AX, SP	2	$AX \leftarrow SP$					
	INCW	SP	2	$SP \leftarrow SP+1$					
	DECW	SP	2	$SP \leftarrow SP-1$					
Special instruction	**CHKL**	sfr	3	(pin level) $\forall$ (signal level before output buffer)	×	×		P	
	CHKLA	sfr	3	$A \leftarrow$ (pin level) $\forall$ (signal level before output buffer)	×	×		P	

Note n indicates the number of registers described as post.

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags				
					S	Z	AC	P/V	CY
Unconditional branch instruction	**BR**	!addr16	3	PC ← addr16					
		rp1	2	PC$_H$ ← rp1$_H$, PC$_L$ ← rp1$_L$					
		[rp1]	2	PC$_H$ ← (rp1+1), PC$_L$ ← (rp1)					
		$ addr16	2	PC ← PC+2+jdisp8					
Conditional branch instruction	**BC**	$ addr16	2	PC ← PC+2+jdisp8 if CY=1					
	BL								
	BNC	$ addr16	2	PC ← PC+2+jdisp8 if CY=0					
	BNL								
	BZ	$ addr16	2	PC ← PC+2+jdisp8 if Z=1					
	BE								
	BNZ	$ addr16	2	PC ← PC+2+jdisp8 if Z=0					
	BNE								
	BV	$ addr16	2	PC ← PC+2+jdisp8 if P/V=1					
	BPE								
	BNV	$ addr16	2	PC ← PC+2+jdisp8 if P/V=0					
	BPO								
	BN	$ addr16	2	PC ← PC+2+jdisp8 if S=1					
	BP	$ addr16	2	PC ← PC+2+jdisp8 if S=0					
	BGT	$ addr16	3	PC ← PC+3+jdisp8 if (P/V $\veebar$ S) $\vee$ Z=0					
	BGE	$ addr16	3	PC ← PC+3+jdisp8 if P/V $\veebar$ S=0					
	BLT	$ addr16	3	PC ← PC+3+jdisp8 if P/V $\veebar$ S=1					
	BLE	$ addr16	3	PC ← PC+3+jdisp8 if (P/V $\veebar$ S) $\vee$ Z=1					
	BH	$ addr16	3	PC ← PC+3+jdisp8 if Z $\vee$ CY=0					
	BNH	$ addr16	3	PC ← PC+3+jdisp8 if Z $\vee$ CY=1					
	BT	saddr. bit, $ addr16	3	PC ← PC+3+jdisp8 if (saddr.bit)=1					
		sfr. bit, $ addr16	4	PC ← PC+4+jdisp8 if sfr.bit=1					
		A. bit, $ addr16	3	PC ← PC+3+jdisp8 if A.bit=1					
		X. bit, $ addr16	3	PC ← PC+3+jdisp8 if X.bit=1					
		PSWH. bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW$_H$.bit=1					
		PSWL. bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW$_L$.bit=1					
	BF	saddr. bit, $ addr16	4	PC ← PC+4+jdisp8 if (saddr.bit)=0					
		sfr. bit, $ addr16	4	PC ← PC+4+jdisp8 if sfr.bit=0					
		A. bit, $ addr16	3	PC ← PC+3+jdisp8 if A.bit=0					
		X. bit, $ addr16	3	PC ← PC+3+jdisp8 if X.bit=0					
		PSWH. bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW$_H$.bit=0					
		PSWL. bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW$_L$.bit=0					

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags S	Z	AC	P/V	CY
Conditional branch instruction	BTCLR	saddr.bit, $ addr16	4	PC ← PC+4+jdisp8 if (saddr.bit)=1 then reset (saddr.bit)					
		sfr.bit, $ addr16	4	PC ← PC+4+jdisp8 if sfr.bit=1 then reset sfr.bit					
		A.bit, $ addr16	3	PC ← PC+3+jdisp8 if A.bit=1 then reset A.bit					
		X.bit, $ addr16	3	PC ← PC+3+jdisp8 if X.bit=1 then reset X.bit					
		PSWH.bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW_H.bit=1 then reset PSW_H.bit					
		PSWL.bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW_L.bit=1 then reset PSW_L.bit	×	×	×	×	×
	BFSET	saddr.bit, $ addr16	4	PC ← PC+4+jdisp8 if (saddr.bit)=0 then set (saddr.bit)					
		sfr.bit, $ addr16	4	PC ← PC+4+jdisp8 if sfr.bit=0 then set sfr.bit					
		A.bit, $ addr16	3	PC ← PC+3+jdisp8 if A.bit=0 then set A.bit					
		X.bit, $ addr16	3	PC ← PC+3+jdisp8 if X.bit=0 then set X.bit					
		PSWH.bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW_H.bit=0 then set PSW_H.bit					
		PSWL.bit, $ addr16	3	PC ← PC+3+jdisp8 if PSW_L.bit=0 then set PSW_L.bit	×	×	×	×	×
	DBNZ	r2, $ addr16	2	r2 ← r2–1, then PC ← PC+2+jdisp8 if r2≠0					
		saddr, $ addr16	3	(saddr) ← (saddr)–1, then PC ← PC+3+jdisp8 if (saddr) ≠0					
Context switching instruction	BRKCS	RBn	2	PC_H ↔ R5, PC_L ↔ R4, R7 ← PSW_H, R6←PSW_L, RBS2–0← n, RSS←0, IE←0					
	RETCS	!addr16	3	PC_H ← R5, PC_L ← R4, R5, R4 ← addr16, PSW_H ← R7, PSW_L ← R6	R	R	R	R	R
	RETCSB	!addr16	4	PC_H ← R5, PC_L ← R4, R5, R4 ← addr16, PSW_H ← R7, PSW_L ← R6	R	R	R	R	R

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags				
					S	Z	AC	P/V	CY
String instruction	**MOVM**	[DE +], A	2	(DE +) ← A, C ← C–1 End if C=0					
		[DE –], A	2	(DE –) ← A, C ← C–1 End if C=0					
	MOVBK	[DE +], [HL +]	2	(DE +) ← (HL +), C ← C–1 End if C=0					
		[DE –], [HL –]	2	(DE –) ← (HL –), C ← C–1 End if C=0					
	XCHM	[DE +], A	2	(DE +) ↔ A, C ← C–1 End if C=0					
		[DE –], A	2	(DE –) ↔ A, C ← C–1 End if C=0					
	XCHBK	[DE +], [HL +]	2	(DE +) ↔ (HL +), C ← C–1 End if C=0					
		[DE –], [HL –]	2	(DE –) ↔ (HL –), C ← C–1 End if C=0					
	CMPME	[DE +], A	2	(DE +) – A, C ← C–1 End if C=0 or Z=0	×	×	×	V	×
		[DE –], A	2	(DE –) – A, C ← C–1 End if C=0 or Z=0	×	×	×	V	×
	CMPBKE	[DE +], [HL +]	2	(DE +) – (HL +), C ← C–1 End if C=0 or Z=0	×	×	×	V	×
		[DE –], [HL –]	2	(DE –) – (HL –), C ← C–1 End if C=0 or Z=0	×	×	×	V	×
	CMPMNE	[DE +], A	2	(DE +) – A, C ← C–1 End if C=0 or Z=1	×	×	×	V	×
		[DE –], A	2	(DE –) – A, C ← C–1 End if C=0 or Z=1	×	×	×	V	×
	CMPBKNE	[DE +], [HL +]	2	(DE +) – (HL +), C ← C–1 End if C=0 or Z=1	×	×	×	V	×
		[DE –], [HL –]	2	(DE –) – (HL –), C ← C–1 End if C=0 or Z=1	×	×	×	V	×
	CMPMC	[DE +], A	2	(DE +) – A, C ← C–1 End if C=0 or CY=0	×	×	×	V	×
		[DE –], A	2	(DE –) – A, C ← C–1 End if C=0 or CY=0	×	×	×	V	×

Instruction Group	Mnemonic	Operand	Bytes	Operation	Flags S	Z	AC	P/V	CY
String instruction	CMPBKC	[DE +], [HL +]	2	(DE +) – (HL +), C ← C–1 End if C=0 or CY=0	×	×	×	V	×
		[DE –], [HL –]	2	(DE –) – (HL –), C ← C–1 End if C=0 or CY=0	×	×	×	V	×
	CMPMNC	[DE +], A	2	(DE +) – A, C ← C–1 End if C=0 or CY=1	×	×	×	V	×
		[DE –], A	2	(DE –) – A, C ← C–1 End if C=0 or CY=1	×	×	×	V	×
	CMPBKNC	[DE +], [HL +]	2	(DE +) – (HL +), C ← C–1 End if C=0 or CY=1	×	×	×	V	×
		[DE –], [HL –]	2	(DE –) – (HL –), C ← C–1 End if C=0 or CY=1	×	×	×	V	×
CPU control instruction	MOV	STBC, #byte	4	STBC ← byte[Note]					
		WDM, #byte	4	WDM ← byte[Note]					
	SWRS		1	RSS ← $\overline{\text{RSS}}$					
	SEL	RBn	2	RBS2–0 ← n, RSS ← 0					
		RBn, ALT	2	RBS2–0 ← n, RSS ← 1					
	NOP		1	No Operation					
	EI		1	IE ← 1 (Enable Interrupt)					
	DI		1	IE ← 0 (Disable Interrupt)					

Note If the operation code of STBC register and WDM register manipulation instructions is abnormal, an operation code trap interrupt is generated.

Operation in the event of trap:

$(SP-1) \leftarrow PSW_H$, $(SP-2) \leftarrow PSW_L$,
$(SP-3) \leftarrow (PC-4)_H$, $(SP-4) \leftarrow (PC-4)_L$,
$PC_L \leftarrow (003CH)$, $PC_H \leftarrow (003DH)$,
$SP \leftarrow SP-4$, $IE \leftarrow 0$

9. ELECTRICAL SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS (T_A = 25 °C)

Parameter	Symbol	Test Conditions		Rating	Unit
Supply voltage	V_{DD}			–0.5 to + 7.0	V
	AV_{DD}			–0.5 to V_{DD} + 0.5	V
	AV_{SS}			–0.5 to + 0.5	V
Input voltage	V_I	**Note 1**		–0.5 to V_{DD} + 0.5	V
Output voltage	V_O			–0.5 to V_{DD} + 0.5	V
Output current low	I_{OL}	All output pins		4.0	mA
		All output pins total		90	mA
Output current high	I_{OH}	All output pins		–1.0	mA
		All output pins total		–20	mA
Analog input voltage	V_{IAN}	**Note 2**	$AV_{DD} > V_{DD}$	–0.5 to V_{DD} +0.5	V
			$V_{DD} \geq AV_{DD}$	–0.5 to AV_{DD} +0.5	
A/D converter reference input voltage	AV_{REF}		$AV_{DD} > V_{DD}$	–0.5 to V_{DD} +0.3	V
			$V_{DD} \geq AV_{DD}$	–0.5 to AV_{DD} +0.3	
Operating ambient temperature	T_A			–10 to + 70	°C
Storage temperature	T_{stg}			–65 to + 150	°C

Notes **1.** Except the pin described in **Note 2**.
2. P70/ANI0 to P77/ANI7 pins.

★ **Caution** **If the absolute maximum rating of any one of the parameters is exceeded even momentarily, the quality of the product may be degraded. In other words, the product may be physically damaged if any of the absolute maximum ratings is exceeded. Be sure to use the product without exceeding these ratings.**

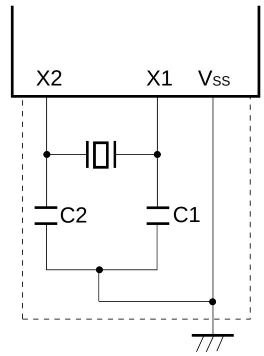
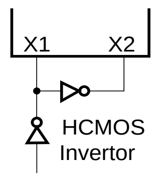
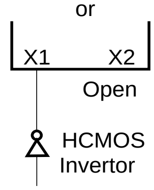
RECOMMENDED OPERATING CONDITION

Oscillation Frequency	T_A	V_{DD}
8 MHz ≤ f_{XX} ≤ 16 MHz	–10 to +70 °C	+5.0 V ±10 %

CAPACITANCE (T_A = 25 °C, V_{SS} = V_{DD} = 0 V)

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Input capacitance	C_I	f = 1 MHz Unmeasured pins returned to 0 V.			10	pF
Output capacitance	C_O				20	pF
I/O capacitance	C_{IO}				20	pF

OSCILLATOR CHARACTERISTICS (T_A = –10 to +70 °C, V_{DD} = +5 V ±10 %, V_{SS} = 0 V)

Resonator	Recommended Circuit	Parameter	MIN.	MAX.	Unit
Ceramic resonator or crystal resonator		Oscillation frequency (f_{XX})	8	16	MHz
External clock	 or 	X1 input frequency (f_X)	8	16	MHz
		X1 input rise/fall time (t_{XR}, t_{XF})	0	20	ns
		X1 input high/low level width (t_{WXH}, t_{WXL})	25	80	ns

Caution **When using the system clock oscillation circuit, wire the part encircled in the dotted line in the following manner to avoid the influence of the wiring capacity, etc.** ★

- **Make the wiring as short as possible.**
- **Avoid intersecting other signal conductors. Avoid approaching lines in which very high fluctuating currents run.**
- **Make sure that the grounding point of the oscillation circuit capacitor always has the same electrical potential as V_{SS}. Avoid grounding with a grand pattern in which very high currents run.**
- **Do not fetch signals from the oscillation circuit.**

RECOMMENDED OSCILLATOR CONSTANT

Ceramic Resonator

Manufacturer	Product Name	Frequency [MHz]	Recommended Constant	
			C1 [pF]	C2 [pF]
Murata Mfg. Co., Ltd.	CSA8.00MT CSA12.0MT	8.0 12.0	30	30
	CSA14.74MXZ040 CSA16.00MX040	14.74 16.0	15	15
	CST8.00MTW CST12.0MTW CST14.74MXW0C3 CST16.00MXW0C3	8.0 12.0 14.74 16.0	On-chip	On-chip

Crystal Resonator

Manufacturer	Product Name	Frequency [MHz]	Recommended Constant	
			C1 [pF]	C2 [pF]
Kinseki Co., Ltd.	HC49/U-S	8 to 16	10	10
	HC49/U			

DC CHARACTERISTICS (T_A = –10 to +70 °C, V_{DD} = +5 V ±10 %, V_{SS} = 0 V)

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Input voltage low	V_{IL}		0		0.8	V
Input voltage high	V_{IH1}	**Note 1**	2.2			V
	V_{IH2}	**Note 2**	$0.8V_{DD}$			V
Output voltage low	V_{OL}	I_{OL} = 2.0 mA			0.45	V
Output voltage high	V_{OH}	I_{OH} = –400 μA	V_{DD} – 1.0			V
Input leakage current	I_{LI}	0 V ≤ V_I ≤ V_{DD}			±10	μA
Output leakage current	I_{LO}	0 V ≤ V_O ≤ V_{DD}			±10	μA
V_{DD} supply current	I_{DD1}	Operating mode		40	65	mA
	I_{DD2}	HALT mode		20	35	mA
Data retention voltage	V_{DDDR}	STOP mode	2.5			V
Data retention current	I_{DDDR}	STOP mode, V_{DDDR} = 2.5 V		2	10	μA
		STOP mode, V_{DDDR} = 5.0 V ±10 %		10	50	μA

Notes **1.** Except the pin descried in **Note 2**.

2. RESET, X1, X2, P20/NMI, P21/INTP0, P22/INTP1, P23/INTP2, P24/INTP3, P25/INTP4, P26/INTP5, P27/INTP6/TI, P32/SO/SB0, P33/SI/SB1, P34/SCK pins.

★ **AC CHARACTERISTICS (T_A = –10 to +70 °C, V_{DD} = +5 V ±10 %, V_{SS} = 0 V)**

Non-consecutive read/write operation (with general-purpose memory connected)

Parameter	Symbol	Test Conditions	MIN.	MAX.	Unit
System clock cycle time	t_{CYK}		125	250	ns
Address setup time (vs. ASTB↓)	t_{SAST}		32		ns
Address hold time (vs. ASTB↓)	t_{HSTA}		32		ns
$\overline{RD}$↓ delay time from address	t_{DAR}		85		ns
Address float time from $\overline{RD}$↓	t_{FRA}			0	ns
Data input time from address	t_{DAID}			222	ns
Data input time from $\overline{RD}$↓	t_{DRID}			112	ns
$\overline{RD}$↓ delay time from ASTB↓	t_{DSTR}		42		ns
Data hold time (vs. $\overline{RD}$↑)	t_{HRID}		0		ns
Address active time from $\overline{RD}$↑	t_{DRA}		50		ns
$\overline{RD}$ low-level width	t_{WRL}		157		ns
ASTB high-level width	t_{WSTH}		37		ns
$\overline{WR}$↓ delay time from address	t_{DAW}		85		ns
Data output time from ASTB↓	t_{DSTOD}			102	ns
Data output time from $\overline{WR}$↓	t_{DWOD}			40	ns
$\overline{WR}$↓ delay time from ASTB↓	t_{DSTW}		42		ns
Data setup time (vs. $\overline{WR}$↑)	t_{SODW}		147		ns
Data hold time (vs. $\overline{WR}$↑)	t_{HWOD}		32		ns
ASTB↑ delay time from $\overline{WR}$↑	t_{DWST}		42		ns
$\overline{WR}$ low-level width	t_{WWL}		157		ns

t_{CYK} DEPENDENT BUS TIMING DEFINITION

Parameter	Expression	MIN./MAX.	Unit
t_{SAST}	0.5T – 30	MIN.	ns
t_{HSTA}	0.5T – 30	MIN.	ns
t_{DAR}	T – 40	MIN.	ns
t_{DAID}	(2.5 + n) T – 90	MAX.	ns
t_{DRID}	(1.5 + n) T – 75	MAX.	ns
t_{DSTR}	0.5T – 20	MIN.	ns
t_{DRA}	0.5T – 12	MIN.	ns
t_{WRL}	(1.5 + n) T – 30	MIN.	ns
t_{WSTH}	0.5T – 25	MIN.	ns
t_{DAW}	T – 40	MIN.	ns
t_{DSTOD}	0.5T + 40	MAX.	ns
t_{DSTW}	0.5T – 20	MIN.	ns
t_{SODW}	1.5T – 40	MIN.	ns
t_{HWOD}	0.5T – 30	MIN.	ns
t_{DWST}	0.5T – 20	MIN.	ns
t_{WWL}	(1.5 + n) T – 30	MIN.	ns

Remarks
1. T = t_{CYK} = 1/f_{CLK} (f_{CLK} is an internal system clock frequency)
2. n indicates the number of wait cycles defined by user software.
3. Depends on t_{CYK} for the bus timing shown in this table only.

SERIAL OPERATION (T_A = –10 to +70 °C, V_{DD} = +5 V ±10 %, V_{SS} = 0 V)

Parameter	Symbol	Test Conditions		MIN.	MAX.	Unit
Serial clock cycle time	t_{CYSK}	$\overline{SCK}$ output	Internal division by 8	1		μs
		$\overline{SCK}$ input	External clock	1		μs
Serial clock low-level width	t_{WSKL}	$\overline{SCK}$ output	Internal division by 8	420		ns
		$\overline{SCK}$ input	External clock	420		ns
Serial clock high-level width	t_{WSKH}	$\overline{SCK}$ output	Internal division by 8	420		ns
		$\overline{SCK}$ input	External clock	420		ns
SI setup time (vs. $\overline{SCK}$↑)	t_{SRXSK}			80		ns
SI hold time (vs. $\overline{SCK}$↑)	t_{HSKRX}			80		ns
SO delay time from $\overline{SCK}$↓	t_{DSKTX}	R = 1 kΩ, C = 100 pF			210	ns

OTHER OPERATION (T_A = –10 to +70 °C, V_{DD} = +5 V ±10 %, V_{SS} = 0 V)

Parameter	Symbol	Test Conditions	MIN.	MAX.	Unit
NMI high/low-level width	t_{WNIH}, t_{WNIL}		5		μs
INTP0 high/low-level width	t_{WI0H}, t_{WI0L}		8T		t_{CYK}
INTP1 high/low-level width	t_{WI1H}, t_{WI1L}		8T		t_{CYK}
INTP2 high/low-evel width	t_{WI2H}, t_{WI2L}		8T		t_{CYK}
NTP3 high/low-level width	t_{WI3H}, t_{WI3L}		8T		t_{CYK}
NTP4 high/low-level width	t_{WI4H}, t_{WI4L}		8T		t_{CYK}
INTP5 high/low-level width	t_{WI5H}, t_{WI5L}		8T		t_{CYK}
INTP6 high/low-level width	t_{WI6H}, t_{WI6L}		8T		t_{CYK}
$\overline{RESET}$ high/low-level width	t_{WRSH}, t_{WRSL}		5		μs
TI high/low-level width	t_{WTIH}, t_{WTIL}	In TM1 event counter mode	8T		t_{CYK}

A/D CONVERTER CHARACTERISTICS (T_A = –10 to +70 °C, V_{DD} = +5 V ±10 %, V_{SS} = AV_{SS} = 0 V, V_{DD} – 0.5 V ≤ AV_{DD} ≤ V_{DD})

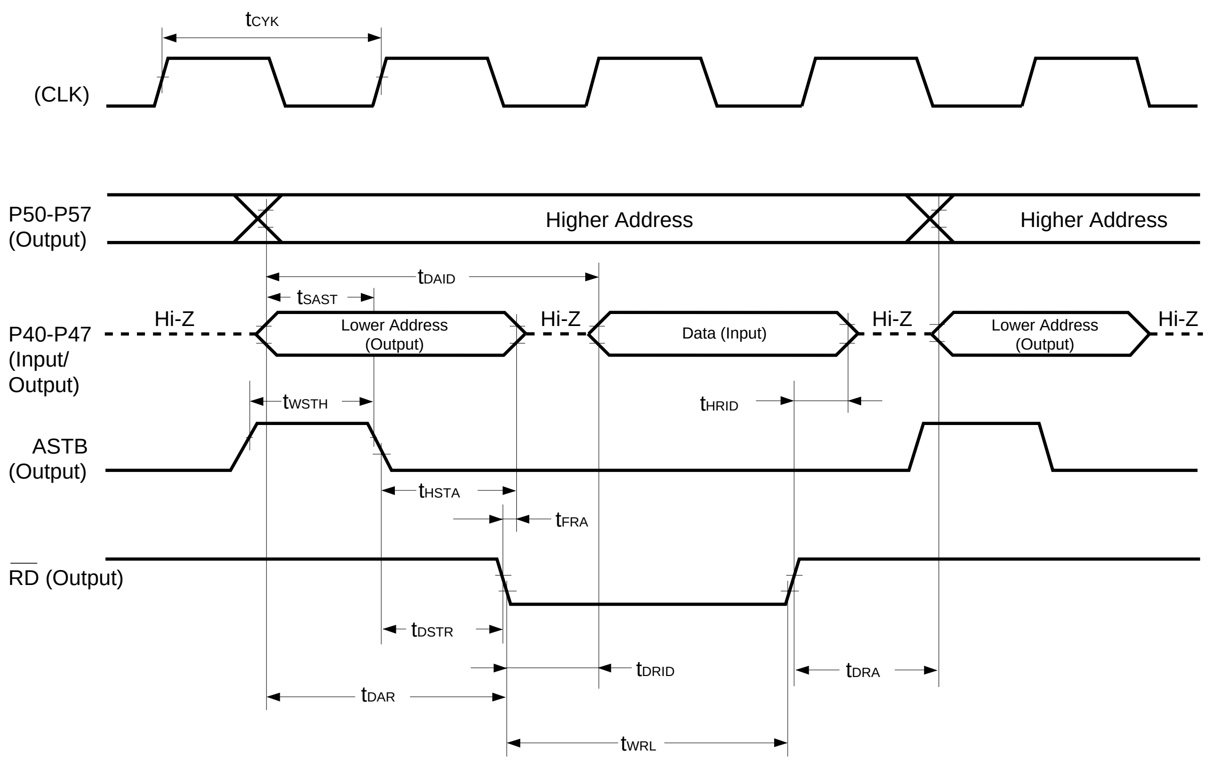
Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Resolution			10			bit
Total error Note 1		4.5 V ≤ AV_{REF} ≤ AV_{DD}			±0.4	%FSR
		3.4 V ≤ AV_{REF} ≤ AV_{DD}			±0.7	%FSR
Quantization error					±1/2	LSB
Conversion time	t_{CONV}		144			t_{CYK}
Sampling time	t_{SAMP}		24			t_{CYK}
Zero scale error Note 1		4.5 V ≤ AV_{REF} ≤ AV_{DD}		±1.5	±2.5	LSB
		3.4 V ≤ AV_{REF} ≤ AV_{DD}		±1.5	±4.5	LSB
Full scale error Note 1		4.5 V ≤ AV_{REF} ≤ AV_{DD}		±1.5	±2.5	LSB
		3.4 V ≤ AV_{REF} ≤ AV_{DD}		±1.5	±4.5	LSB
Non-linear error Note 1		4.5 V ≤ AV_{REF} ≤ AV_{DD}		±1.5	±2.5	LSB
		3.4 V ≤ AV_{REF} ≤ AV_{DD}		±1.5	±4.5	LSB
Analog input voltage Note 2	V_{IAN}		–0.3		AV_{DD}	V
Reference voltage	AV_{REF}		3.4		AV_{DD}	V
AV_{REF} current	AI_{REF}			1.0	3.0	mA
AV_{DD} supply current	AI_{DD}			2.0	6.0	mA
A/D converter data retention current	AI_{DDR}	STOP mode, AV_{DDR} = 2.5 V		2.0	10	μA
		STOP mode, AV_{DDR} = 5 V ±10 %		10	50	μA

★

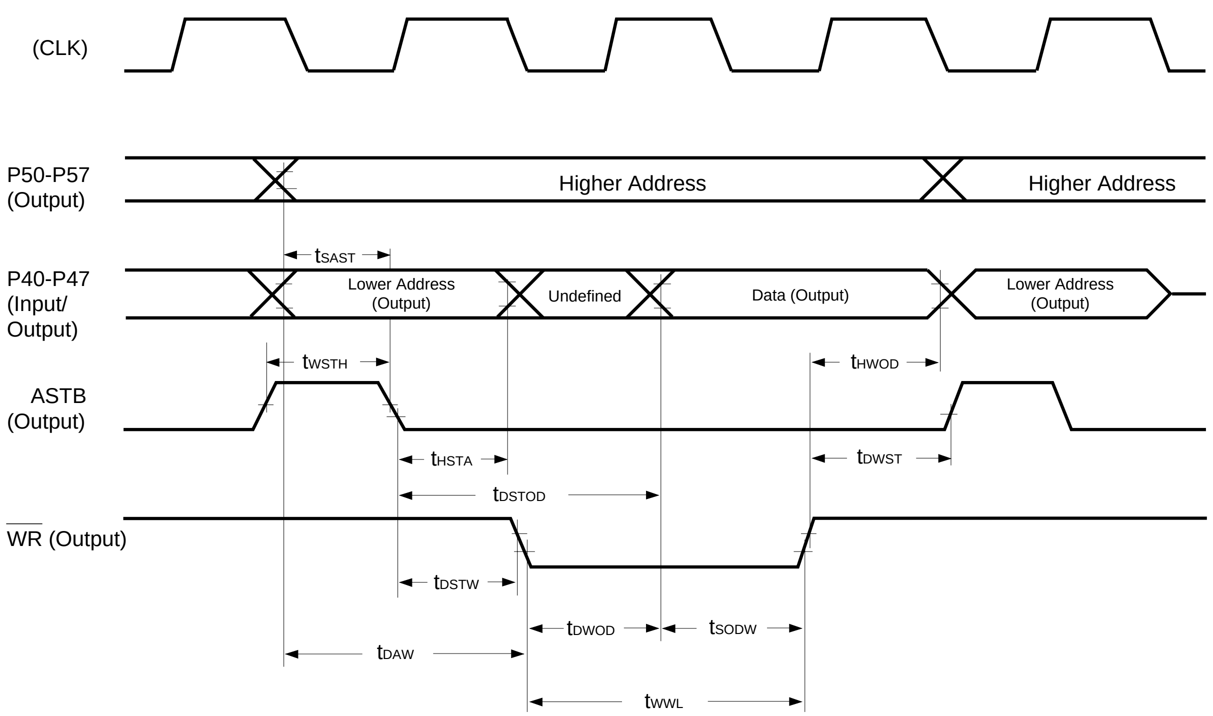
Notes **1.** Quantization error excluded.

2. When –0.3 V ≤ V_{IAN} ≤ 0 V, the conversion result becomes 000H.
When 0 V < V_{IAN} < AV_{REF}, the conversion is performed at a resolution of 10 bits.
When AV_{REF} ≤ V_{IAN} ≤ AV_{DD}, the conversion result is 3FFH.

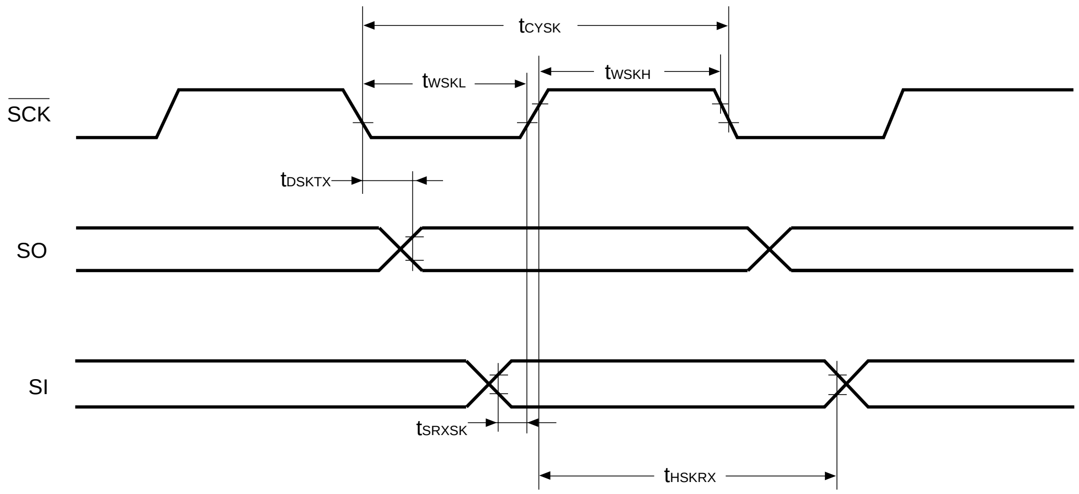
Non-Consecutive Read Operation



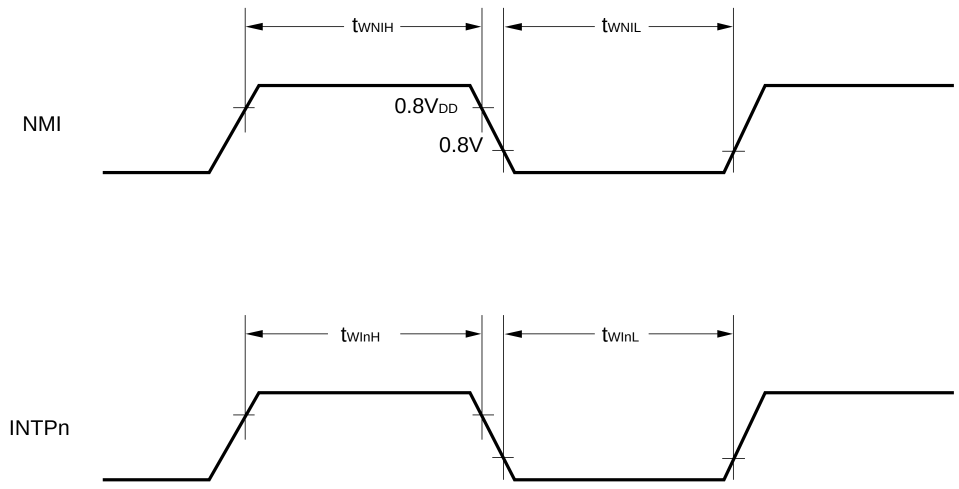
Non-Consecutive Write Operation



Serial Operation

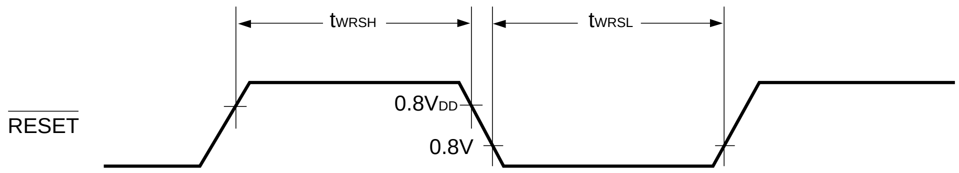


Interrupt Input Timing

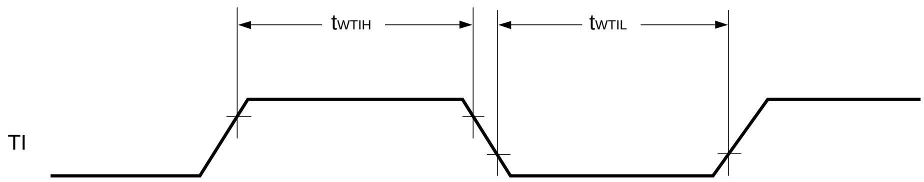


Remark n = 0 to 6

Reset Input Timing

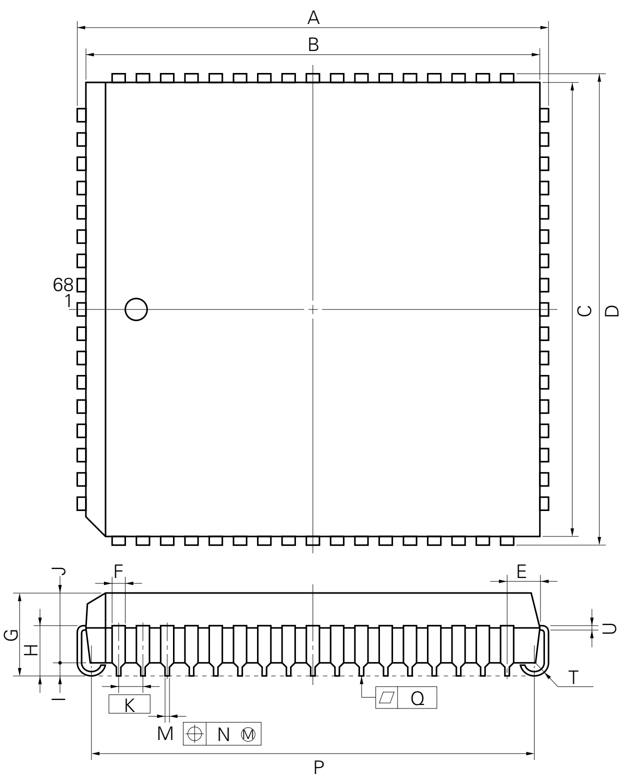


TI Pin Input Timing



10. PACKAGE DRAWINGS

68 PIN PLASTIC QFJ (□950 mil)



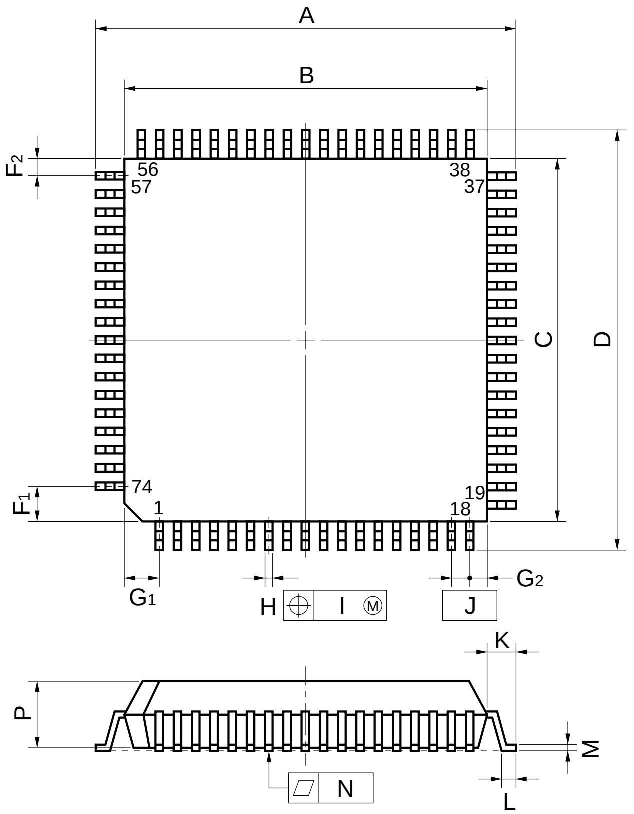
P68L-50A1-2

NOTE

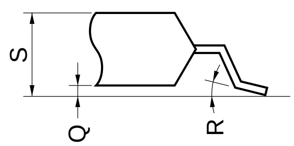
Each lead centerline is located within 0.12 mm (0.005 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	25.2±0.2	0.992±0.008
B	24.20	0.953
C	24.20	0.953
D	25.2±0.2	0.992±0.008
E	1.94±0.15	$0.076^{+0.007}_{-0.006}$
F	0.6	0.024
G	4.4±0.2	$0.173^{+0.009}_{-0.008}$
H	2.8±0.2	$0.110^{+0.009}_{-0.008}$
I	0.9 MIN.	0.035 MIN.
J	3.4	0.134
K	1.27 (T.P.)	0.050 (T.P.)
M	0.40±1.0	$0.016^{+0.004}_{-0.005}$
N	0.12	0.005
P	23.12±0.20	$0.910^{+0.009}_{-0.008}$
Q	0.15	0.006
T	R 0.8	R 0.031
U	$0.20^{+0.10}_{-0.05}$	$0.008^{+0.004}_{-0.002}$

74 PIN PLASTIC QFP (□20)



detail of lead end



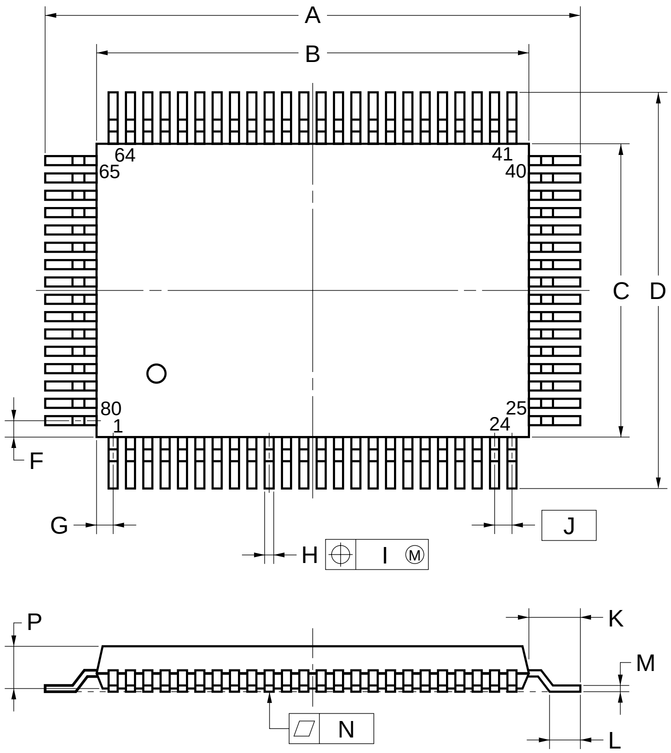
NOTE

Each lead centerline is located within 0.20 mm (0.008 inch) of its true position (T.P.) at maximum material condition.

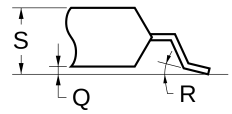
ITEM	MILLIMETERS	INCHES
A	23.2±0.4	$0.913^{+0.017}_{-0.016}$
B	20.0±0.2	$0.787^{+0.009}_{-0.008}$
C	20.0±0.2	$0.787^{+0.009}_{-0.008}$
D	23.2±0.4	$0.913^{+0.017}_{-0.016}$
F_1	2.0	0.079
F_2	1.0	0.039
G_1	2.0	0.079
G_2	1.0	0.039
H	0.40±0.10	$0.016^{+0.004}_{-0.005}$
I	0.20	0.008
J	1.0 (T.P.)	0.039 (T.P.)
K	1.6±0.2	0.063±0.008
L	0.8±0.2	$0.031^{+0.009}_{-0.008}$
M	$0.15^{+0.10}_{-0.05}$	$0.006^{+0.004}_{-0.003}$
N	0.10	0.004
P	3.7	0.146
Q	0.1±0.1	0.004±0.004
R	5°±5°	5°±5°
S	4.0 MAX.	0.158 MAX.

S74GJ-100-5BJ-3

80 PIN PLASTIC QFP (14×20)



detail of lead end



NOTE

Each lead centerline is located within 0.15 mm (0.006 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	23.6±0.4	0.929±0.016
B	20.0±0.2	$0.795^{+0.009}_{-0.008}$
C	14.0±0.2	$0.551^{+0.009}_{-0.008}$
D	17.6±0.4	0.693±0.016
F	1.0	0.039
G	0.8	0.031
H	0.35±0.10	$0.014^{+0.004}_{-0.005}$
I	0.15	0.006
J	0.8 (T.P.)	0.031 (T.P.)
K	1.8±0.2	$0.071^{+0.008}_{-0.009}$
L	0.8±0.2	$0.031^{+0.009}_{-0.008}$
M	$0.15^{+0.10}_{-0.05}$	$0.006^{+0.004}_{-0.003}$
N	0.10	0.004
P	2.7	0.106
Q	0.1±0.1	0.004±0.004
R	5°±5°	5°±5°
S	3.0 MAX.	0.119 MAX.

P80GF-80-3B9-3

11. RECOMMENDED SOLDERING CONDITIONS

The μPD78322 should be soldered and mounted under the conditions recommended in the table below.

For detail of recommended soldering conditions, refer to the information document **"Semiconductor Device Mounting Technology Manual" (IE-1207)**.

For soldering methods and conditions other than those recommended below, contact our salesman.

Table 11-1. Soldering Conditions for Surface Mount Type

★ **μPD78320GF-3B9 : 80-pin plastic QFP (14 × 20 mm)**
★ **μPD78322GF-×××-3B9 : 80-pin plastic QFP (14 × 20 mm)**
★ **μPD78320GJ-5BJ : 74-pin plastic QFP (20 × 20 mm)**
★ **μPD78322GJ-×××-5BJ : 74-pin plastic QFP (20 × 20 mm)**

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235 °C, Time: 30 sec. max. (at 210 °C or above), Number of times: twice or less <Caution> (1) The second reflow should be started after the temperature of the device which would have been changed by the first reflow has returned to normal. (2) Please avoid flux water washing after the first reflow.	IR35-00-2
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (at 200 °C or above), Number of times: twice or less <Caution> (1) The second reflow should be started after the temperature of the device which would have been changed by the first reflow has returned to normal. (2) Please avoid flux water washing after the first reflow.	VP15-00-2
Wave soldering	Solder bath temperature: 260 °C max. Time: 10 sec. max., Number of times: Once Preheating temperature: 120 °C max. (package surface temperature)	WS60-00-1
Pin part heating	Pin temperature: 300 °C max. Time: 3 sec. max. (Per device side)	—

μPD78320L : 68-pin plastic QFJ (□950 mil)
μPD78322L-××× : 68-pin plastic QFJ (□950 mil)

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 230 °C, Time: 30 sec. max. (at 210 °C or above), Number of times: Once	IR30-00-1
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (at 200 °C or above), Number of times: Once	VP15-00-1
Pin part heating	Pin part temperature: 300 °C max. Time: 3 sec. max. (Per device side)	—

Caution Use more than one soldering method should be avoided (except in the case of pin part heating).

APPENDIX A. LIST OF 78K/III SERIES PRODUCTS (1/2)

		μPD78322	μPD78320	μPD78312[Note]	μPD78310[Note]	μPD78312A	μPD78310A
Basic instruction		111		96			
Minimum instruction execution time		250 ns (at 16 MHz operation)		500 ns (at 12 MHz operation)			
Internal memory	ROM	16384 × 8 bits	—	8192 × 8 bits	—	8192 × 8 bits	—
	RAM	640 × 8 bits		256 × 8 bits			
Memory space		64K bytes					
I/O lines	Input	16 (including 8 analog inputs)		12 (including 4 analog inputs)			
	Output	—		1			
	I/O	39	21	40	24	40	24
Pulse unit		Real-time pulse unit • 18-/16-bit free running timer × 1 • 16-bit timer/event counter × 1 • 16-bit compare register × 6 • 18-bit capture register × 4 • 18-bit capture/compare register × 2 • Real-time output port × 8		Multi-function pulse I/O unit • 16-bit presettable up/down-counter × 2 • 16-bit free running counter capture function × 2 • 16-bit interval time × 2 • High-precision PWM output × 2 • Real-time output port : 4 bits × 2			
				Count unit mode 4 (4-multiplication mode) function not available		Count unit mode 4 (4-multiplication mode) function available	
				Count start function by interval time external trigger not available		Count start function by interval timer external trigger available	
Serial communication interface		• Dedicated on-chip baud rate generator • UART ... 1 channel • SBI, • 3-wire serial I/O } ... 1 channel		• 8 bits (full-duplex transmission/reception) • Dedicated on-chip baud rete generator • 2 transfer modes (asynchronous mode, I/O interface mode)			
A/D converter		Eight 10-bit resolution inputs		Four 8-bit resolution inputs			
Interrupt		• 8 external, 14 internal (shared with external : 2) • 3-level programmable priority		• 4 external, 13 internal • 8-level programmable priority			
		• 3 processing modes (vectored interrupt, context switching and macro service functions)					

Note Maintenance product

LIST OF 78K/III SERIES PRODUCTS (2/2)

	μPD78322	μPD78320	μPD78312[Note]	μPD78310[Note]	μPD78312A	μPD78310A
Test source	Internal : 1		—			
Instruction set	Instructions for μPD78312 and 78310 significantly added.		—		Following instructions added for μPD78312 and 78310 • MOVW rp1, !addr16 instruction • MOVW !addr16, rp1 instruction	
Pulse unit	• On-chip watchdog timer • Standby function (STOP/HALT)					
	—		• 20-bit time base counter • Pseudo static RAM refresh function			
Package	• 68-pin plastic QFJ (□ 950 mil) • 74-pin plastic QFP (20 × 20 mm) • 80-pin plastic QFP (14 × 20 mm)		• 64-pin plastic shrink DIP (750 mil) • 64-pin plastic QFP (14 × 20 mm) • 64-pin plastic QUIP • 68-pin plastic QFJ (□ 950 mil)			

Note Maintenance product

APPENDIX B. TOOLS ★

B.1 DEVELOPMENT TOOLS

The following development tools are available for system development using the μPD78322.

Language Processor

Tool	Host machine	OS	Supply medium	Part number
78K/III series relocatable assembler (RA78K/III)	Refers to the relocatable assembler which can be used commonly for the 78K/III series. Equipped with the macro function, the relocatable assembler is aimed at improved development efficiency. The assembler is also accompanied by the structured assembler which can describe the program control structure explicitly, thus making it possible to improve the productivity and the maintainability of the program.			
	PC-9800 series	MS-DOS™	3.5-inch 2HD	μS5A13RA78K3
			5-inch 2HD	μS5A10RA78K3
	IBM PC/AT™ and its compatible machine	PC DOS™	3.5-inch 2HC	μS7B13RA78K3
			5-inch 2HC	μS7B10RA78K3
	HP9000 series 700™	HP-UX™	DAT	μS3P16RA78K3
	SPARCstation™	SunOS™	Cartridge tape (QIC-24)	μS3K15RA78K3
	NEWS™	NEWS-OS™		μS3R15RA78K3
78K/III series C compiler (CC78K/III)	Refers to the C compiler which can be commonly used in the 78K/III series. This compiler is a program converting the programs written in the C language to those object codes which are executable by microcontrollers. When using this compiler, the 78K/III series relocatable assembler (RA78K/III) is required.			
	PC-9800 series	MS-DOS	3.5-inch 2HD	μS5A13CC78K3
			5-inch 2HD	μS5A10CC78K3
	IBM PC/AT and its compatible machine	PC DOS	3.5-inch 2HC	μS7B13CC78K3
			5-inch 2HC	μS7B10CC78K3
	HP9000 series 700	HP-UX	DAT	μS3P16CC78K3
	SPARCstation	SunOS	Cartridge tape (QIC-24)	μS3K15CC78K3
	NEWS	NEWS-OS		μS3R15CC78K3

Remark Relocatable assembler and C compiler operations are assured only on the host machine and the OS above.

PROM Writing Tools

<table>
<tr><td rowspan="3">Hardware</td><td>PG-1500</td><td colspan="4">This PROM programmer allows programming, in stand-alone mode or via operation from a host computer, of a single-chip microcontroller with on-chip PROM by connection of the board provided and a separately available programmer adapter.
It can program typical 256K-bit to 4M-bit PROMs.</td></tr>
<tr><td>UNISITE
2900</td><td colspan="4">PROM programmer made by Data I/O Japan Corporation.</td></tr>
<tr><td>PA-78P322GF
PA-78P322GJ
PA-78P322K
PA-78P322KC
PA-78P322KD
PA-78P322L</td><td colspan="4">PROM programmer adapters for writing programs to the μPD78P322 with a general PROM programmer such as the PG-1500.
PA-78P322GF ... For μPD78P322GF
PA-78P322GJ ... For μPD78P322GJ
PA-78P322K ... For μPD78P322K
PA-78P322KC ... For μPD78P322KC
PA-78P322KD ... For μPD78P322KD
PA-78P322L ... For μPD78P322L</td></tr>
<tr><td rowspan="6">Software</td><td rowspan="6">PG-1500 controller</td><td colspan="4">Connects PG-1500 and host machine via a serial and parallel interface, and controls the PG-1500 on the host machine.</td></tr>
<tr><td>Host Machine</td><td>OS</td><td>Supply medium</td><td>Part number</td></tr>
<tr><td rowspan="2">PC-9800 series</td><td rowspan="2">MS-DOS</td><td>3.5-inch 2HD</td><td>μS5A13PG1500</td></tr>
<tr><td>5-inch 2HD</td><td>μS5A10PG1500</td></tr>
<tr><td rowspan="2">IBM PC/AT and its compatible machine</td><td rowspan="2">PC DOS</td><td>3.5-inch 2HC</td><td>μS7B13PG1500</td></tr>
<tr><td>5-inch 2HC</td><td>μS7B10PG1500</td></tr>
</table>

Remark Operation of the PG-1500 controller is guaranteed only on the host machines and operating systems quoted above.

Debugging Tools

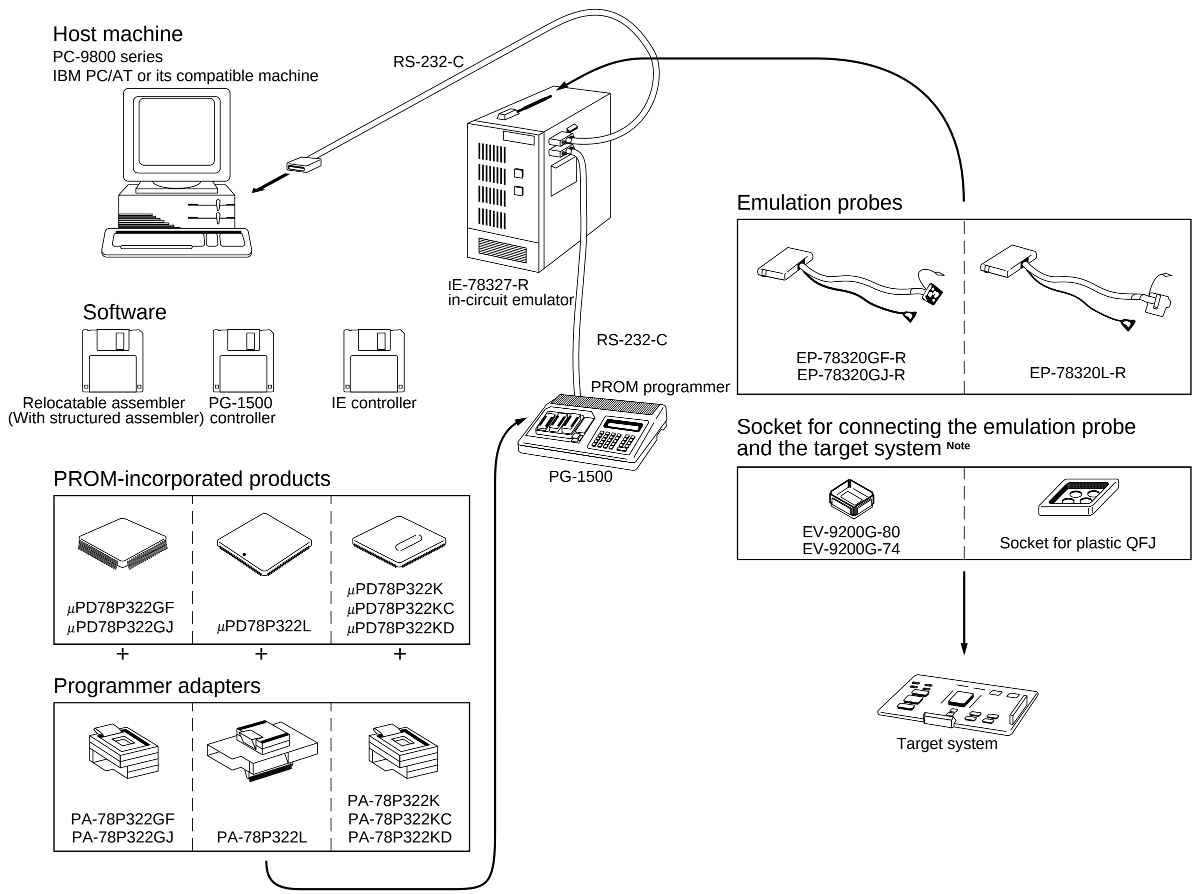
<table>
<tr><td rowspan="2">Hardware</td><td>IE-78327-R
IE-78320-R[Note]</td><td colspan="4">These are the in-circuit emulators which can be used for the development and debugging of application systems. Debugging is performed by connecting them to a host machine. The IE-78327-R can be used commonly for both the μPD78322 subseries and the μPD78328 subseries.
The IE-78320-R can be used for the μPD78322 subseries.</td></tr>
<tr><td>EP-78320GF-R
EP-78320GJ-R
EP-78320L-R</td><td colspan="4">These are the emulation probes for connecting the IE-78327-R or IE-78320-R to a target system.
EP-78320GF-R: for 80-pin plastic QFP
EP-78320GJ-R: for 74-pin plastic QFP
EP-78320L-R : for 68-pin plastic QFJ</td></tr>
<tr><td rowspan="13">Software</td><td rowspan="7">IE-78327-R
control program
(IE controller)</td><td colspan="4">This program is for controlling the IE-78327-R from a host machine. It can execute commands automatically, thus enabling more efficient debugging.</td></tr>
<tr><td rowspan="2">Host machine</td><td colspan="2"></td><td rowspan="2">Part number</td></tr>
<tr><td>OS</td><td>Supply medium</td></tr>
<tr><td rowspan="2">PC-9800 series</td><td rowspan="2">MS-DOS</td><td>3.5-inch 2HD</td><td>μS5A13IE78327</td></tr>
<tr><td>5-inch 2HD</td><td>μS5A10IE78327</td></tr>
<tr><td rowspan="2">IBM PC/AT and its compatible machine</td><td rowspan="2">PC DOS</td><td>3.5-inch 2HC</td><td>μS7B13IE78327</td></tr>
<tr><td>5-inch 2HC</td><td>μS7B10IE78327</td></tr>
<tr><td rowspan="6">IE-78320-R
control program[Note]
(IE controller)</td><td colspan="4">This program is for controlling the IE-78320-R from a host machine. It can execute commands automatically, thus enabling more efficient debugging.</td></tr>
<tr><td rowspan="2">Host machine</td><td colspan="2"></td><td rowspan="2">Part number</td></tr>
<tr><td>OS</td><td>Supply medium</td></tr>
<tr><td rowspan="2">PC-9800 series</td><td rowspan="2">MS-DOS</td><td>3.5-inch 2HD</td><td>μS5A13IE78320</td></tr>
<tr><td>5-inch 2HD</td><td>μS5A10IE78320</td></tr>
<tr><td>IBM PC/AT and its compatible machine</td><td>PC DOS</td><td>5-inch 2HC</td><td>μS7B10IE78320</td></tr>
</table>

Remarks **1.** The operation of each software is assured only on the host machine and the OS above.

2. μPD78322 subseries: μPD78320, 78322, 78P322, 78323, 78324, 78P324, 78320(A), 78320(A1), 78320(A2), 78322(A), 78322(A1), 78322(A2), 78323(A), 78323(A1), 78323(A2), 78324(A), 78324(A1), 78324(A2), 78P324(A), 78P324(A1), 78P324(A2)

μPD78328 subseries: μPD78327, 78328, 78P328, 78327(A), 78328(A)

Note The existing product IE-78320-R is a maintenance product. If you are going to newly purchase an in-circuit emulator, please use the alternative product IE-78327-R.



Note The socket is supplied with the emulation probe.

Remark It is also possible to use the host machine and the PG-1500 by connecting them directly by the RS-232C.

B.2 EVALUATION TOOLS

To evaluate the functions of the μPD78322, the following tools are made available.

Part Number	Host Machine	Function
EB-78320-98	PC-9800 series	By connecting to a host machine, it is possible to evaluate the functions equipped by the μPD78322 in a simple manner. The command system of this product basically conforms to that of IE-78327-R and IE-78320-R. Therefore, it is easy to move to the development work of application systems by IE-78327-R or IE-78320-R. In addition a turbo access manager (μPD71P301)[Note] can be mounted on the board.
EB-78320-PC	IBM PC/AT or its compatible machine	

Note The turbo access manager (μPD71P301) is a maintenance product.

Cautions 1. This product is not a development tool of μPD78322 application systems.

2. This product is not equipped with the emulation function for executing the ROM incorporated in the μPD78322.

B.3 EMBEDDED SOFTWARE

The following embedded software programs are available to perform program development and maintenance more efficiently.

Eeal-time OS

Real-time OS (RX78K/III)	The RX78K/III is designed to provide a multi-task environment in the field of control application where real-time operation is required. By using this real-time OS, the performance of the whole system can be improved by allocating CPU's idle time to other processings. The RX78K/III provides the system call based on the μITRON specifications. The RX78K/III package provides tools (configurators) for creating RX78K/III's nucleus and multiple information table.			
	Host machine	OS	Supply medium	Part number
	PC-9800 series	MS-DOS	3.5-inch 2HD	μS5A13RX78320
			5-inch 2HD	μS5A10RX78320
	IBM PC/AT and its compatible machine	PC DOS	3.5-inch 2HC	μS7B13RX78320
			5-inch 2HC	μS7B10RX78320

Caution To purchase the operating system above, you need to fill in a purchase application form beforehand and sign a contract allowing you to use the software.

Remark When using the real-time OS RX78K/III, you need the assembler package RA78K/III (optional) as well.

Fuzzy Inference Development Support System

Fuzzy knowledge data creation tools (FE9000, FE9200)	This program supports inputting/editing/evaluating (through simulation) of the fuzzy knowledge data (fuzzy rules and membership functions).			
	Host machine	OS	Supply medium	Part number
	PC-9800 series	MS-DOS	3.5-inch 2HD	μS5A13FE9000
			5-inch 2HD	μS5A10FE9000
	IBM PC/AT and its compatible machine	PC DOS / Winsows™	3.5-inch 2HC	μS7B13FE9200
			5-inch 2HC	μS7B10FE9200
Translator (FT78K3)Note	This program converts the fuzzy knowledge data obtained with fuzzy knowledge data creation tools to an assembler source program for RA78K/III.			
	Host machine	OS	Supply medium	Part number
	PC-9800 series	MS-DOS	3.5-inch 2HD	μS5A13FT78K3
			5-inch 2HD	μS5A10FT78K3
	IBM PC/AT and its compatible machine	PC DOS	3.5-inch 2HC	μS7B13FT78K3
			5-inch 2HC	μS7B10FT78K3
Fuzzy inference module (FI78K/III)Note	This program executes fuzzy inference. Fuzzy inference is executed by being linked to the fuzzy knowledge data converted by the translator.			
	Host machine	OS	Supply medium	Part number
	PC-9800 series	MS-DOS	3.5-inch 2HD	μS5A13FI78K3
			5-inch 2HD	μS5A10FI78K3
	IBM PC/AT and its compatible machine	PC DOS	3.5-inch 2HC	μS7B13FI78K3
			5-inch 2HC	μS7B10FI78K3
Fuzzy inference debugger (FD78K/III)	This is a support software program for evaluating and adjusting the fuzzy knowledge data at a hardware level by using the in-circuit emulator.			
	Host machine	OS	Supply medium	Part number
	PC-9800 series	MS-DOS	3.5-inch 2HD	μS5A13FD78K3
			5-inch 2HD	μS5A10FD78K3
	IBM PC/AT and its compatible machine	PC DOS	3.5-inch 2HC	μS7B13FD78K3
			5-inch 2HC	μS7B10FD78K3

Note Under development

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

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

② HANDLING OF UNUSED INPUT PINS FOR CMOS

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

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

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

For this product, the following User's Manual is available as a separate volume. Please refer to it in conjunction with manual.

- **μPD78322 User's Manual : IEU1248**

The export of these products from Japan is regulated by the Japanese government. The export of some or all of these products may be prohibited without governmental license. To export or re-export some or all of these products from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

License not needed	:	μPD78320
The customer must judge the need for license	:	μPD78322