4520 Group

SINGLE-CHIP 4-BIT CMOS MICROCOMPUTER

DESCRIPTION

The 4520 Group is a 4-bit single chip microcomputer designed with CMOS technology. Its CPU is that of 4500 series using a simple, high-speed instruction set. The computer is equipped with serial I/O, A-D converter and an LCD control circuit, and is suitable for household appliances and consumer equipment.

The various microcomputers in the 4520 Group include variations of the built-in memory type and package as shown the table below.

APPLICATION

VCRs, televisions, audio-visual equipment, microwave ovens, rice cookers, telephones, office automation, toys

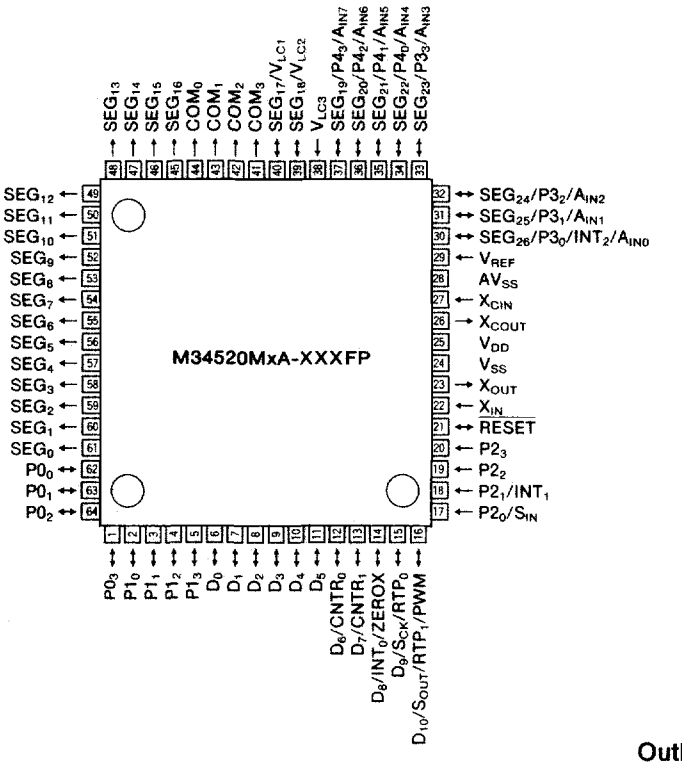
FEATURES

- Number of basic instructions 129
- Minimum instruction execution time 0.75μs
  (at 4MHz system clock frequency)
- Supply voltage
  at 1.5MHz system clock frequency ..... 2.2V to 5.5V
  (One Time PROM and EPROM version ..... 2.5V to 5.5V)
  at 4.0MHz system clock frequency ..... 4.5V to 5.5V
- Timers
  Timer 1 ..... 8-bit timer with a reload register
  Timer 2 ..... 8-bit timer with a reload register
  Timer 3 ..... 4 bits×2 (fixed dividing frequency)
  Timer 4 ..... 8-bit timer with a reload register
- Interrupt function ..... 8 sources
- A-D converter ..... 8-bit successive comparison method×8ch
- Serial I/O ..... 8-bit wide
- Clock generating circui
  Main clock (X_{IN}) : a ceramic resonator or external clock input
  Sub-clock (X_{CIN}) : a quartz-crystal oscillator (32kHz)
- Built-in LCD controller/driver
  Segment output 27
  Common output 4
- Zero cross detection circuit 1
- Real time output 2
- LED drive directly enabled (port D)

Product	ROM (PROM) size (×10 bits)	RAM size (×4 bits)	Package	ROM type
M34520M6A-XXXSP/FP	6144 words	384 words	SP : 64P4B FP : 64P6N-A	Mask ROM
M34520M8A-XXXSP/FP	8192 words			
M34520E8-XXXSP/FP	8192 words	384 words	SP : 64P4B FP : 64P6N-A	One Time PROM
M34520E8SP/FP *				
M34520E8SS/FS **			SS : 64S1B-E FS : 64D0	EPROM

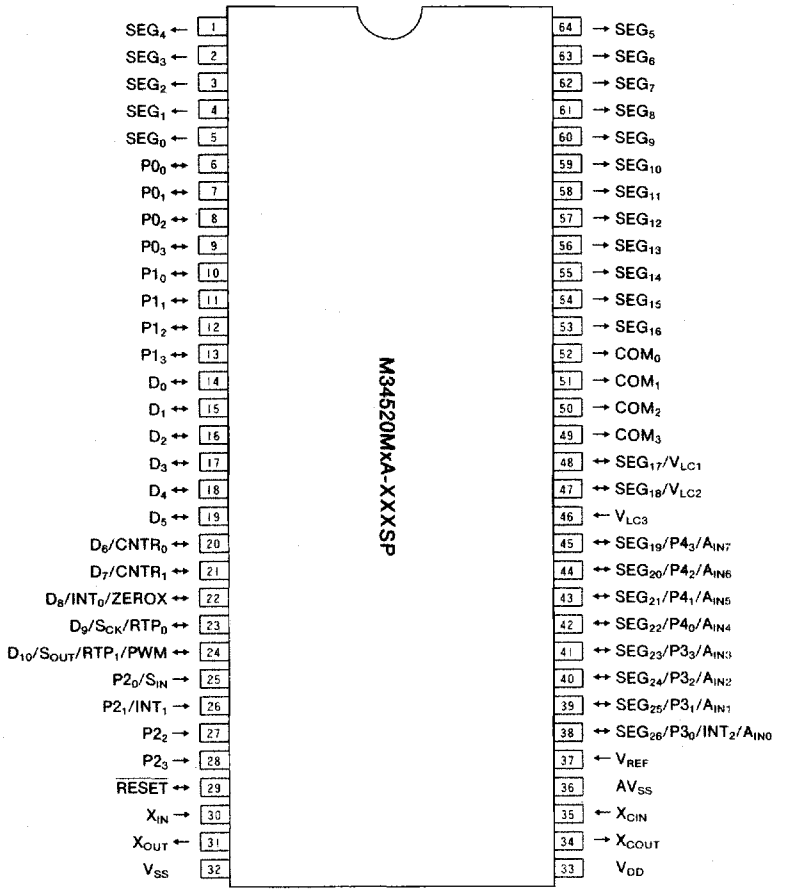
* : Shipped in blank ** : For program development only

PIN CONFIGURATION (TOP VIEW)



Outline 64P6N-A

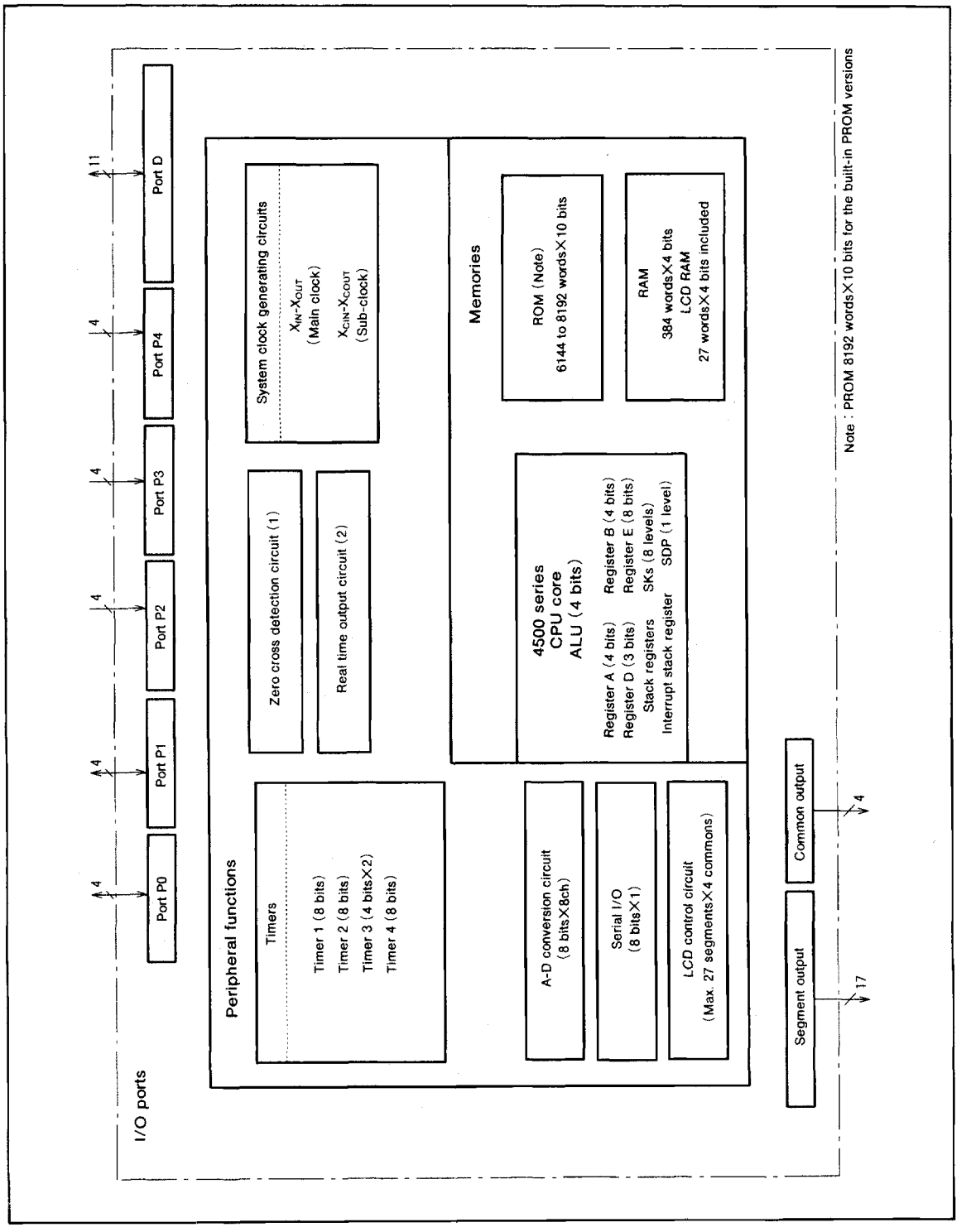
PIN CONFIGURATION (TOP VIEW)



Outline 64P4B

MITSUBISHI ELECTRIC

BLOCK DIAGRAM (M34520MxA-XXXSP/FP)



PERFORMANCE OVERVIEW

Parameter			Function
Number of basic instructions			129
Minimum instruction execution time			0.75 μs (at 4MHz system clock frequency)
System clock frequencies	Main clock		4MHz
	Sub-clock		32kHz
Memory sizes	ROM	M34520M6A	6144 words×10 bits
		M34520M8A	8192 words×10 bits
	RAM	M34520M6A	384 words×4 bits (LCD RAM 27 words×4 bits included)
		M34520M8A	
Input/Output pins	D_0–D_{10}	I/O	Eleven independent I/O ports; D_9 pin is also used as real time output. D_{10} pin is also used as real time output or PWM output. D_0–D_7, D_9 and D_{10} (10 pins) are the middle withstand voltage N-channel open-drain I/O pins and can drive directly.
	$P0_0$–$P0_3$	I/O	4-bit I/O port; each pin is equipped with a pull-up function and a key-on wakeup function (both programmable).
	$P1_0$–$P1_3$	I/O	4-bit I/O port; each pin is equipped with a pull-up function and a key-on wakeup function (both programmable).
	$P2_0$–$P2_3$	Input	4-bit input port
	$P3_0$–$P3_3$	Input	4-bit input port; each pin is also used as an analog input pin or a segment output pin.
	$P4_0$–$P4_3$	Input	4-bit input port; each pin is also used as an analog input pin or a segment output pin.
	SEG_0–SEG_{16}	LCD output	Seventeen LCD output pins; 10-bit expansion is enabled with ports P3, P4, V_{LC1} and V_{LC2}.
	COM_0–COM_3	LCD output	Four LCD output pins
	$CNTR_0$	Output	Timer output; $CNTR_0$ pin is also used as port D_6.
	$CNTR_1$	I/O	Timer I/O; $CNTR_1$ pin is also used as port D_7.
	INT_0	Input	Interrupt input; INT_0 pin is also used as port D_8 or zero cross input pin, and is equipped with a key-on wakeup function.
	INT_1	Input	Interrupt input; INT_1 pin is also used as port $P2_1$.
	INT_2	Input	Interrupt input; INT_2 pin is also used as port $P3_0$.
Timers	Timer 1		8-bit programmable timer with a reload register
	Timer 2		8-bit programmable timer with a reload register is also used as an event counter
	Timer 3		4-bit fixed dividing frequency ×2
	Timer 4		8-bit programmable timer with a reload register
A-D converter			8-bit successive comparison method×8ch
Serial I/O			8-bit wide
Interrupt	Source		8 (two for external, four for timer, A-D, and serial I/O)
	Nesting		1 level
Subroutine nesting			8 levels
LCD	Selective bias value		1/2, 1/3 bias
	Selective duty value		2, 3, 4 duty
	Common output		4
	Segment output		27
	Internal resistor for power supply		200kΩ (typical) ×3
Device structure			CMOS silicon gate
Packages	M34520MxA-XXXSP		64-pin plastic molded SDIP
	M34520MxA-XXXFP		64-pin plastic molded QFP
Operating temperature range			−20℃ to 85℃ (−20℃ to 70℃ at M34520E8SS/M34520E8FS)
Supply voltage	$f(X_{IN})$=1.5MHz		2.2V to 5.5V (2.5V to 5.5V at all built-in PROM versions)
	$f(X_{IN})$=4.0MHz		4.5V to 5.5V

PIN DESCRIPTIONS

Pin	Name	Input/Output	Function
V_{DD}	Power supply	—	Connected to a plus power supply
V_{SS}	Ground	—	Connected to a 0V power supply
AV_{SS}	Analog power supply input	Input	Connected to a 0V power supply for A-D converter.
V_{REF}	Reference voltage input	Input	Reference voltage input pin for A-D converter.
$\overline{RESET}$	Reset I/O	I/O	An N-channel open-drain I/O pin for a system reset. When the watchdog timer causes the system to be reset, the $\overline{RESET}$ pin outputs "L" level.
X_{IN}	Main clock input	Input	I/O pins of the main clock generating circuit. X_{IN} and X_{OUT} can be connected to a ceramic resonator. A feedback resistor is built-in between them.
X_{OUT}	Main clock output	Output	
X_{CIN}	Sub-clock input	Input	I/O pins of the sub-clock generating circuit. X_{CIN} and X_{COUT} are connected to a quarts-crystal oscillator. A feedback resistor is built-in between them.
X_{COUT}	Sub-clock output	Output	
D_0-D_{10}	I/O port D	I/O	Each pin has an independent 1-bit wide I/O function for instructions SZD, SD, and RD. Each bit is designated for independent use by register Y of the data pointer. Each pin has an output latch. For input use, set the latch of the specified bit to "1". All latches on port D can be set to "1" with the CLD instruction.
$P0_0-P0_3$	I/O port P0	I/O	Each of ports P0 and P1 serves as a 4-bit I/O port, and it can be used as inputs when the output latch is set to "1". Every pin of the ports has a key-on wakeup function and a pull-up function.
$P1_0-P1_3$	I/O port P1	I/O	
$P2_0-P2_3$	Input port P2	Input	4-bit input port
$P3_0-P3_3$	Input ports P3	Input	4-bit input port
$P4_0-P4_3$	Input ports P4	Input	4-bit input port
SEG_0-SEG_{26}	Segment output	Output	LCD segment output pins
COM_0-COM_3	Common output	Output	LCD common output pins. Pins COM_0 and COM_1 are used at 1/2 duty, pins COM_0-COM_2 are used at 1/3 duty, and pins COM_0-COM_3 are used at 1/4 duty.
$V_{LC1}-V_{LC3}$	LCD power input	Input	LCD power supply input pins. Connect V_{LC3} pin to V_{DD} pin when an internal resistor is used (connect to V_{DD} through a resistor if brightness control is necessary). Apply voltage such that $0 \leq V_{LC1} \leq V_{LC2} \leq V_{LC3} \leq V_{DD}$ when external power is used.
INT_0	Interrupt input	Input	INT_0 pin accepts an external interrupt. It also accepts the input signal which releases the system from the power down state (key-on wakeup function).
INT_1	Interrupt input	Input	INT_1 pin accepts an external interrupt.
INT_2	Interrupt input	Input	INT_2 pin accepts an external interrupt.
ZEROX	Zero cross input	Input	ZEROX/D_8/INT_0 pin is used as the zero cross input pin with software.
$CNTR_0$	Timer output	Output	$CNTR_0$ pin is used to output timer 1 underflow signals.
$CNTR_1$	Timer input/output	I/O	$CNTR_1$ pin is used to output timer 2 underflow signals, and to input clock signals to the timer 3 event counter.
RTP_0, RTP_1	Real time output RTP	Output	Pins RTP_0 and RTP_1 are used as the real time output pins with software.
PWM	PWM output	Output	$D_{10}/S_{OUT}/RTP_1$/PWM pin is also used as the PWM output pin with software.

PIN DESCRIPTIONS (continued)

Pin	Name	Input/Output	Function
S_{IN}	Serial data input	Input	$P2_0/S_{IN}$ pin is used to input serial data signals with software.
S_{OUT}	Serial data output	Output	$D_{10}/S_{OUT}/RTP_1/PWM$ pin is used to output serial data signals with software.
S_{CK}	Serial I/O clock input/output	I/O	$D_9/S_{CK}/RTP_0$ pin is used to input and output synchronous clock signals for serial data transfer with software.
A_{IN0}—A_{IN7}	Analog input A_{IN}	Input	Eight analog input pins

MULTIFUNCTION

Pin	Multifunction	Pin	Multifunction	Pin	Multifunction
D_6	$CNTR_0$	SEG_{17}	V_{LC1}	$CNTR_0$	D_6
D_7	$CNTR_1$	SEG_{18}	V_{LC2}	$CNTR_1$	D_7
D_8	INT_0/ZEROX	SEG_{19}	$P4_3/A_{IN7}$	RTP_0	D_9/S_{CK}
D_9	S_{CK}/RTP_0	SEG_{20}	$P4_2/A_{IN6}$	RTP_1	D_{10}/S_{OUT}/PWM
D_{10}	S_{OUT}/RTP_1/PWM	SEG_{21}	$P4_1/A_{IN5}$	PWM	$D_{10}/S_{OUT}/RTP_1$
$P2_0$	S_{IN}	SEG_{22}	$P4_0/A_{IN4}$	S_{IN}	$P2_0$
$P2_1$	INT_1	SEG_{23}	$P3_3/A_{IN3}$	S_{OUT}	D_{10}/RTP_1/PWM
$P3_0$	$SEG_{26}/INT_2/A_{IN0}$	SEG_{24}	$P3_2/A_{IN2}$	S_{CK}	D_9/RTP_0
$P3_1$	SEG_{25}/A_{IN1}	SEG_{25}	$P3_1/A_{IN1}$	A_{IN0}	$SEG_{26}/P3_0/INT_2$
$P3_2$	SEG_{24}/A_{IN2}	SEG_{26}	$P3_0/INT_2/A_{IN0}$	A_{IN1}	$SEG_{25}/P3_1$
$P3_3$	SEG_{23}/A_{IN3}	V_{LC1}	SEG_{17}	A_{IN2}	$SEG_{24}/P3_2$
$P4_0$	SEG_{22}/A_{IN4}	V_{LC2}	SEG_{18}	A_{IN3}	$SEG_{23}/P3_3$
$P4_1$	SEG_{21}/A_{IN5}	INT_0	D_8/ZEROX	A_{IN4}	$SEG_{22}/P4_0$
$P4_2$	SEG_{20}/A_{IN6}	INT_1	$P2_1$	A_{IN5}	$SEG_{21}/P4_1$
$P4_3$	SEG_{19}/A_{IN7}	INT_2	$SEG_{26}/P3_0/A_{IN0}$	A_{IN6}	$SEG_{20}/P4_2$
		ZEROX	D_8/INT_0	A_{IN7}	$SEG_{19}/P4_3$

Note : Pins except above have just single function.

CONNECTIONS OF UNUSED PINS

Pin	Connection	Pin	Connection
X_{OUT}	Open (when using external clock)	$P2_0/S_{IN}$, $P2_1/INT_1$, $P2_2$, $P2_3$	Connect to V_{SS} (Note 3)
X_{CIN}	Connect to V_{SS}		
X_{COUT}	Open		
AV_{SS}	Connect to V_{SS}	$SEG_{26}/P3_0/INT_2/A_{IN0}$, $SEG_{25}/P3_1/A_{IN1}$—$SEG_{23}/P3_3/A_{IN3}$	Select the SEG pin function and open these pins. (Note 4)
V_{REF}	Connect to V_{SS} (Note 1)		
D_0—D_5, $D_6/CNTR_0$, $D_7/CNTR_1$, $D_8/INT_0/ZEROX$, $D_9/S_{CK}/RTP_0$, $D_{10}/S_{OUT}/RTP_1/PWM$	Connect to V_{SS}, or set the output latch to "0" and open.	$SEG_{22}/P4_0/A_{IN4}$—$SEG_{19}/P4_3/A_{IN7}$	Select the SEG pin function and open these pins. (Note 4)
		COM_0—COM_3	Open
		SEG_0—SEG_{16}, SEG_{17}/V_{LC1}, SEG_{18}/V_{LC2}	Open (Note 5)
$P0_0$—$P0_3$	Open or connect to V_{SS} (Note 2)	V_{LC3}	When not using LCD, connect to V_{DD}
$P1_0$—$P1_3$	Open or connect to V_{SS} (Note 2)		

Notes 1. The 4520 Group has current flowing from the V_{REF} pin even when not using the A-D conversion. Accordingly, on systems that require low power consumption such as battery drive system, the power to the V_{REF} pin should be cut off when not using the A-D conversion. An example of a circuit to turn the power to the V_{REF} pin off by controlling the general purpose port is shown below.

2. When the $P0_0$—$P0_3$ and $P1_0$—$P1_3$ are connected to V_{SS}, turn off their pull-up transistors with software (register PU0i="0") and also invalidate the key-on wakeup functions (register K0i="0"). If the key-on wakeup functions are left valid, the system fails to return from power down. When these pins are disconnected, turn on their pull-up transistors (register PU0i="1") with software. Be sure to invalidate the key-on wakeup functions and the pull-up functions with every two bits. If only one of the two bits key-on wakeup functions is used, turn on their pull-up transistors with software and also disconnect the other pin. (i represents 0, 1, 2, or 3.)

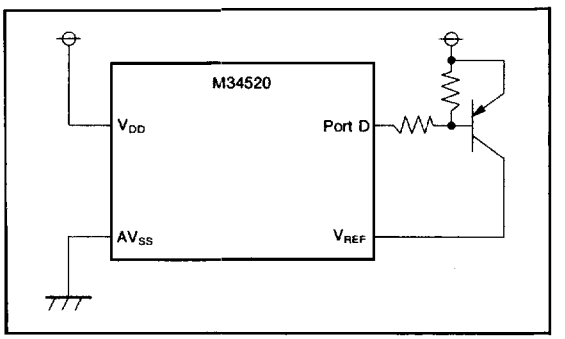
3. When not using the $P2_3$ pin of M34520E8, connect to V_{SS} through a 5kΩ resistor at the shortest distance.

4. When setting some of pins $SEG_{26}/P3_0/INT_2/A_{IN0}$—$SEG_{19}/P4_3/A_{IN7}$ to be unused, note the following.
Select the SEG pin function with register L2 and open the unused pins.
When selecting pins $P3_2/A_{IN2}$ and $P3_3/A_{IN3}$ function with the bit 2 of register L2, fix all pins ($P3_2/A_{IN2}$ and $P3_3/A_{IN3}$) to be used or unused.
When selecting pins $P4_0/A_{IN4}$ and $P4_3/A_{IN7}$ function with the bit 3 of register L2, fix all pins ($P4_0/A_{IN4}$ and $P4_3/A_{IN7}$) to be used or unused.

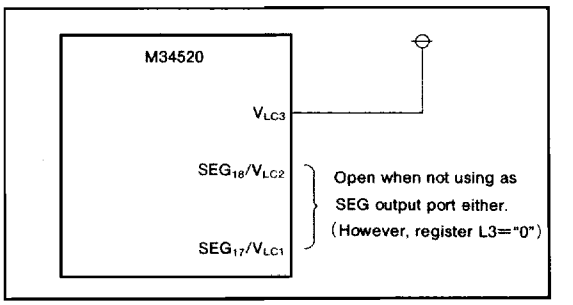
5. SEG_{18} and SEG_{17} are also used as V_{LC2} and V_{LC1}, respectively. Clear the LCD control register (L3) to "0_2" and cut them off from the internal LCD power supply and pins open when they are not used (register L3="0_2" at reset).

(Note when the output latch is set to "0" and pins are open)
- After reset is released, port is in a high-impedance state until it is switched to an output enabled state. Accordingly, the voltage level of pins is undefined and the excess of the supply current may occur while the port is in a high-impedance state.
- To set the output latch periodically by a program is recommended because the value of output latch may change by noise or program run-away (caused by noise).

(Note when connecting to V_{DD} and V_{SS})
- Connect the unused pins to V_{DD} or V_{SS} using the thickest wire at the shortest distance.



Example of handling V_{REF} pin at A-D conversion



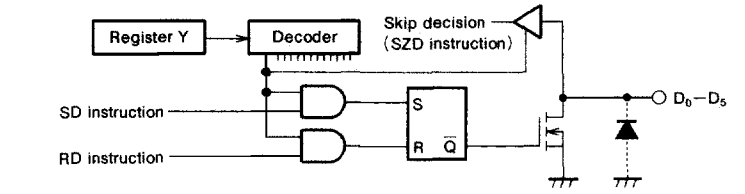
Handling of unused LCD power supply input pins

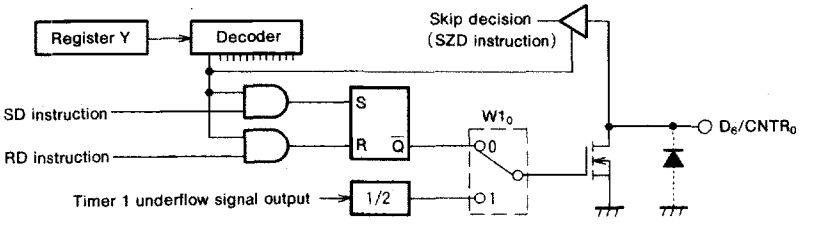
PORT FUNCTION

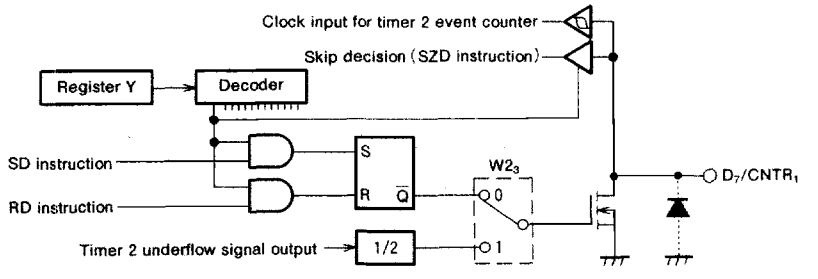
Port	Pin	Input/ Output	Output structure	Control bits	Control instructions	Control registers	Remarks
Port D	D_0—D_5	I/O	N-channel open-drain	1	SD		
	D_6/$CNTR_0$	(11)			RD	W1	
	D_7/$CNTR_1$				SZD	W2	
	D_8/INT_0/ZEROX				CLD	I1	Key-on wakeup functions
	D_9/S_{CK}/RTP_0				SNZI0	J1	
	D_{10}/S_{OUT}/RTP_1/PWM				(Note 1)	J2	
Port P0	$P0_0$—$P0_3$	I/O (4)	N-channel open-drain	4	OP0A IAP0	PU0 K0	Built-in programmable pull-up functions Key-on wakeup functions (programmable)
Port P1	$P1_0$—$P1_3$	I/O (4)	N-channel open-drain	4	OP1A IAP1	PU0 K0	Built-in programmable pull-up functions Key-on wakeup functions (programmable)
Port P2	$P2_0$/S_{IN} $P2_1$/INT_1 $P2_2$ $P2_3$	Input (4)		4	IAP2 SNZI1 (Note 2)		
Port P3	SEG_{26}/$P3_0$/INT_2/A_{IN0} SEG_{25}/$P3_1$/A_{IN1} SEG_{24}/$P3_2$/A_{IN2} SEG_{23}/$P3_3$/A_{IN3}	Input (4)		4	IAP3	L2	
Port P4	SEG_{22}/$P4_0$/A_{IN4} SEG_{21}/$P4_1$/A_{IN5} SEG_{20}/$P4_2$/A_{IN6} SEG_{19}/$P4_3$/A_{IN7}	Input (4)		4	IAP4	L2	
Power input for LCD	SEG_{17}/V_{LC1} SEG_{18}/V_{LC2} V_{LC3}	Input				L1 L3	

Notes 1. Level of D_8/INT_0 pin can be examined with the SNZI0 instruction.
2. Level of $P2_1$/INT_1 pin can be examined with the SNZI1 instruction.
However, level of OR operation between $P2_1$/INT_1 pin and SEG_{26}/$P3_0$/INT_2/A_{IN0} pin is examined when register $I3_0$ is "1".

PORT BLOCK DIAGRAMS



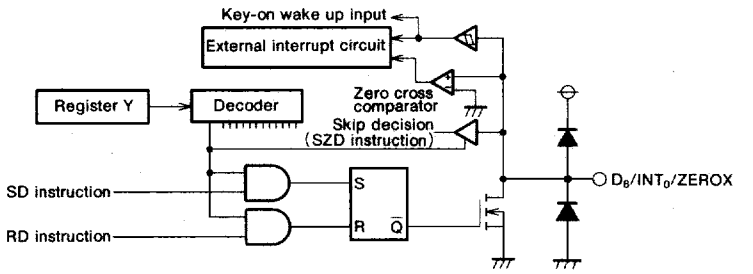


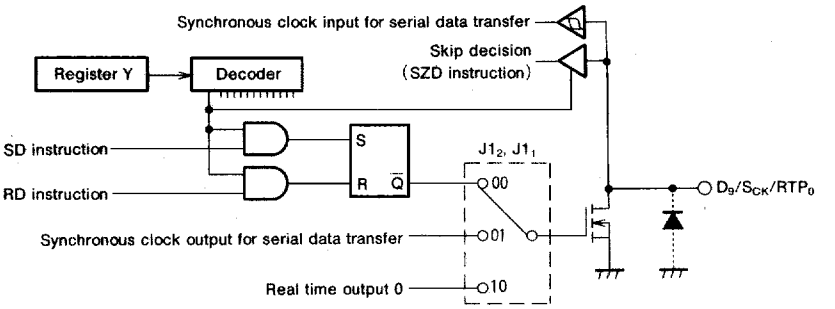


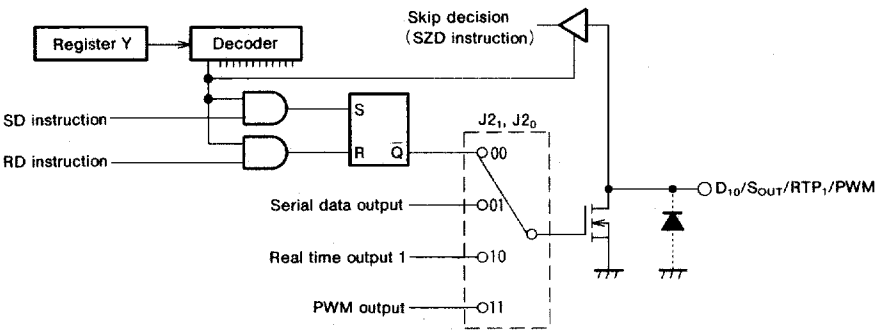
This symbol represents a parasitic diode on the port.

Applied potential to ports D_0—D_7 must be 12V or less.

PORT BLOCK DIAGRAMS (continued)



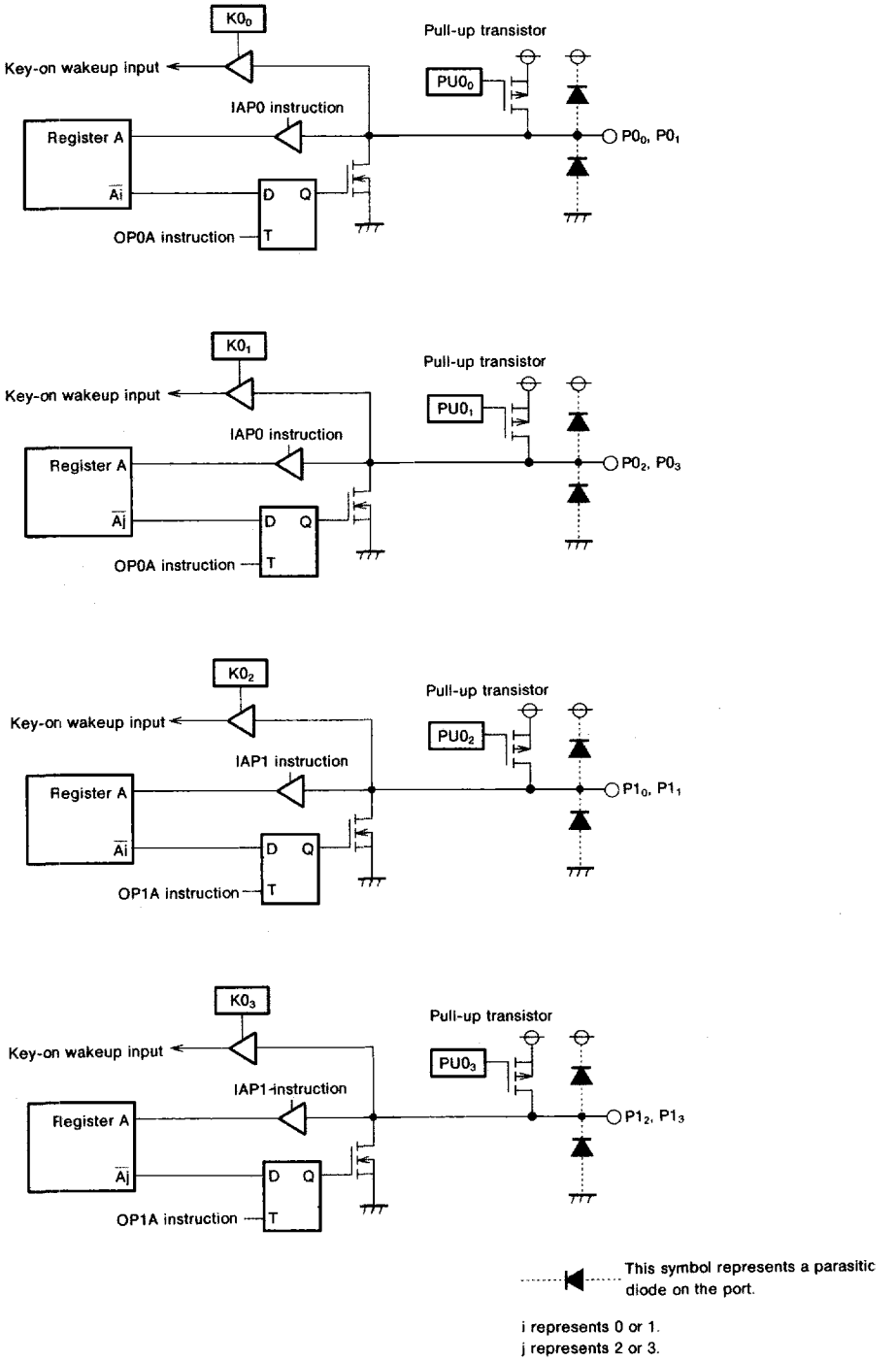




This symbol represents a parasitic diode on the port.

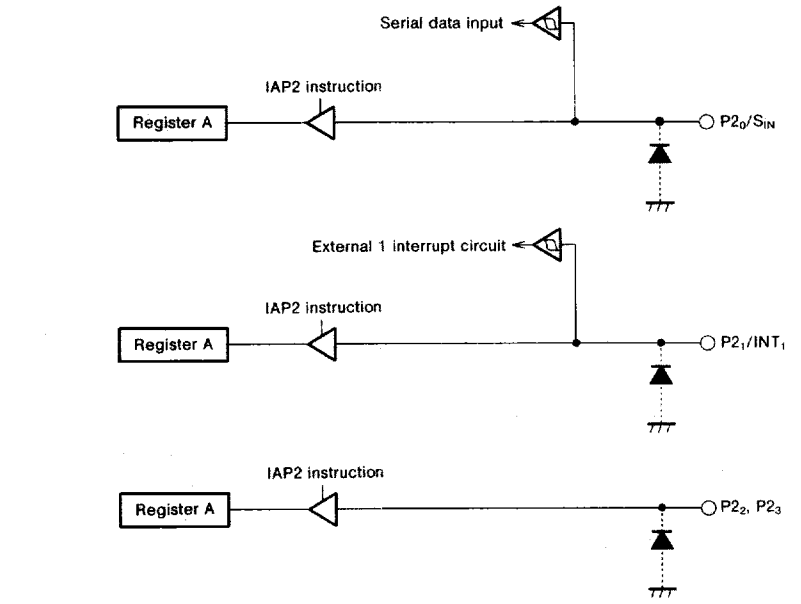
Applied potential to ports D_9 and D_{10} must be 12V or less.

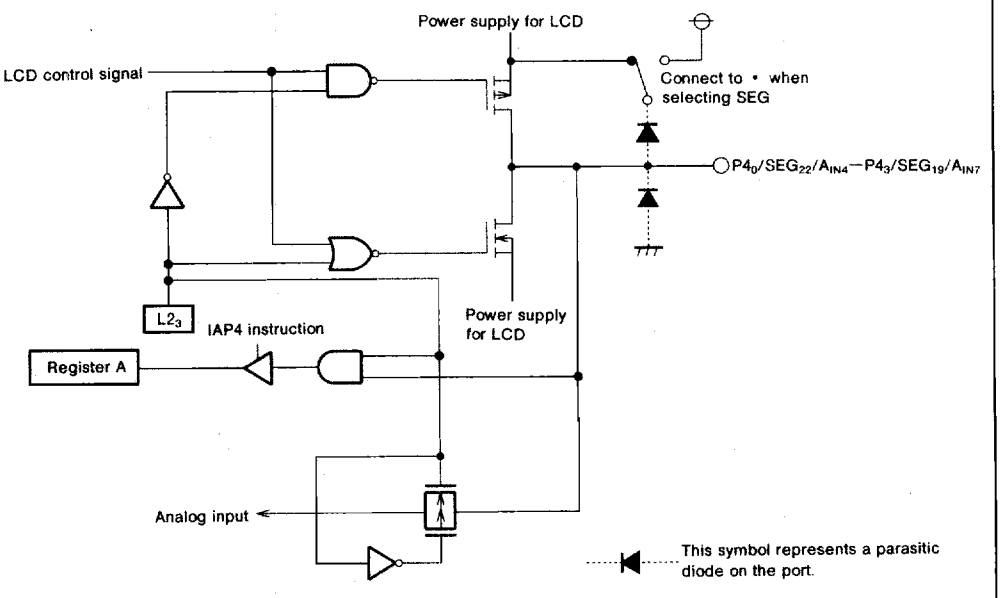
PORT BLOCK DIAGRAMS (continued)



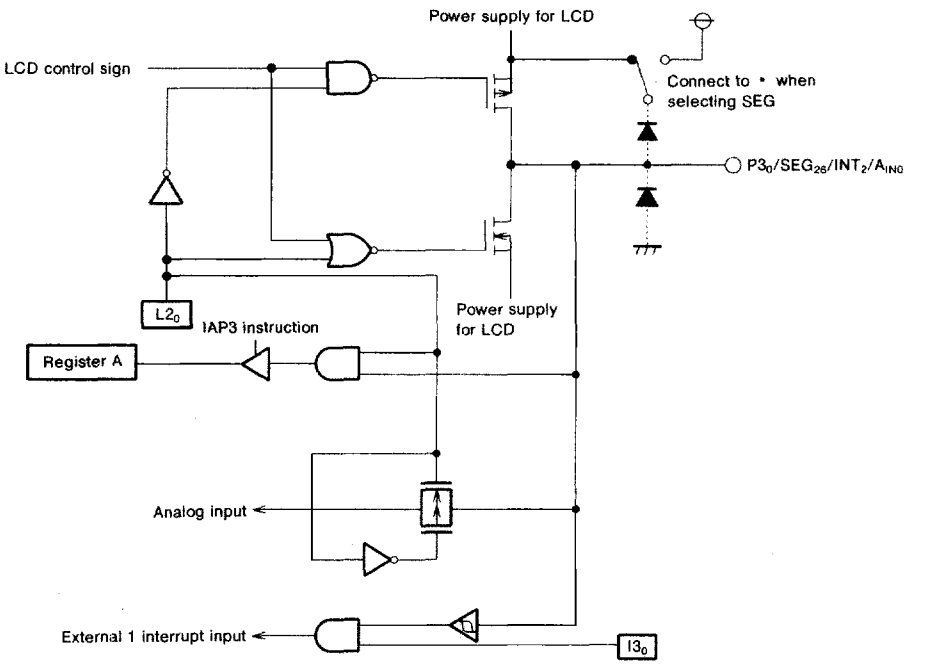
i represents 0 or 1.
j represents 2 or 3.

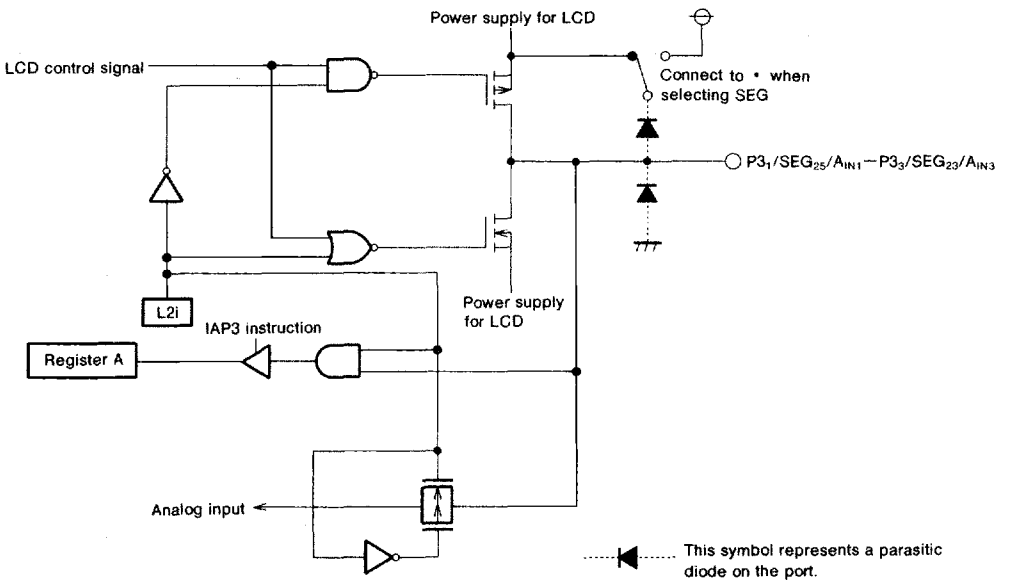
PORT BLOCK DIAGRAMS (continued)



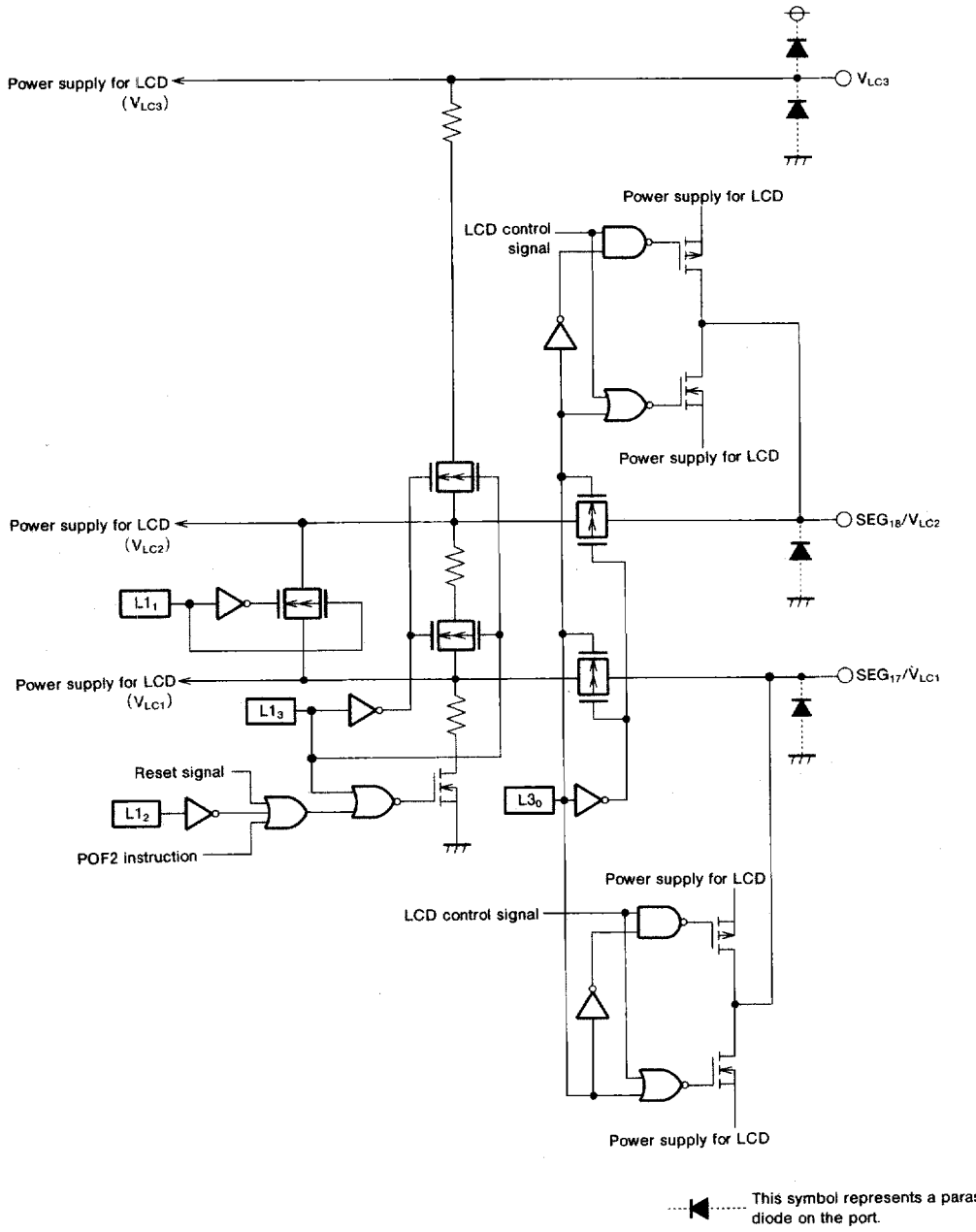


PORT BLOCK DIAGRAMS (continued)



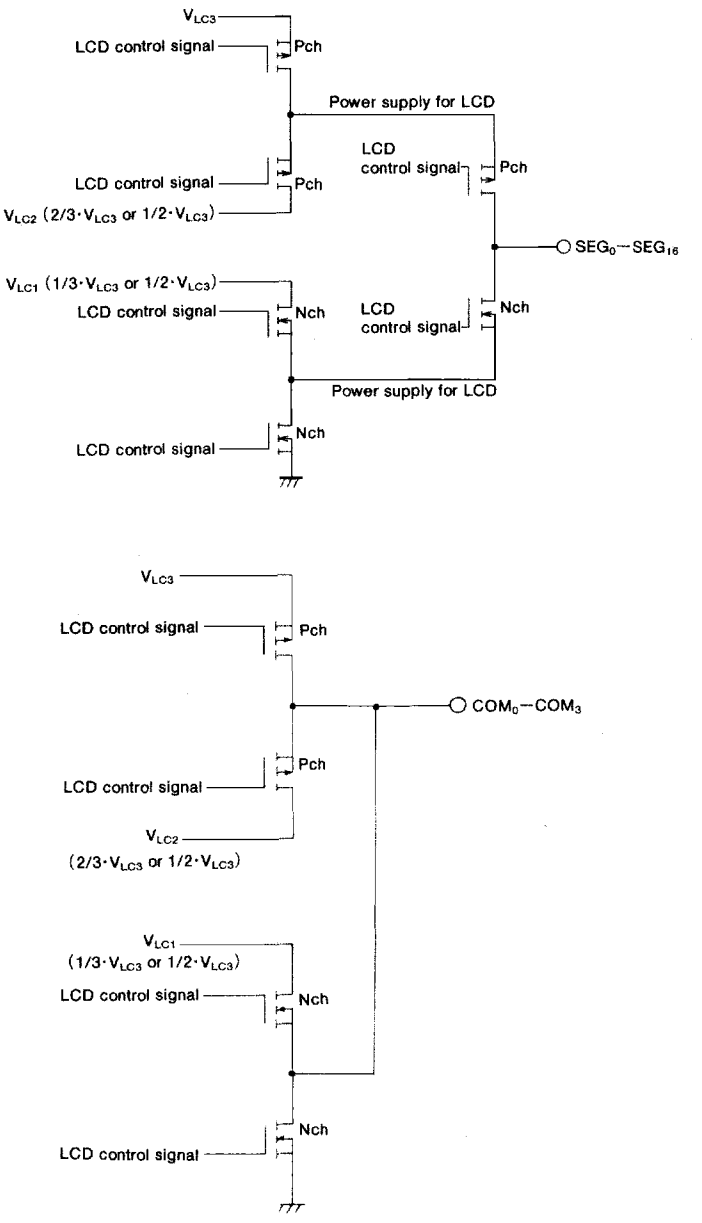


PORT BLOCK DIAGRAMS (continued)



MITSUBISHI ELECTRIC

PORT BLOCK DIAGRAMS (continued)



FUNCTION BLOCK OPERATIONS

ARITHMETIC LOGIC UNIT (ALU)

The arithmetic logic unit ALU performs 4-bit arithmetic such as 4-bit data addition, comparison, AND operation, OR operation, and bit manipulation.

REGISTER A AND CARRY FLAG

Register A is a 4-bit register used for arithmetic, transfer, exchange, and I/O operation.

Carry flag CY is a 1-bit flag that is set to "1" when there is a carry with the AMC instruction (it is unchanged with both A n instruction and AM instruction). The value of A_0 is stored in the carry flag CY with the RAR instruction.

Carry flag CY can be set to "1" with the SC instruction and cleared to "0" with the RC instruction.

REGISTERS B and E

Register B is a 4-bit register used for temporary storage of 4-bit data, and for 8-bit data transfer together with register A.

Register E is an 8-bit register. It can be used for 8-bit data transfer with register B used as the high-order 4 bits and register A as the low-order 4 bits.

REGISTER D

Register D is a 3-bit register. It is used to store a 7-bit ROM address together with register A and is used as a pointer within the specified page when the TABP p, BLA p, or BMLA p instruction is executed.

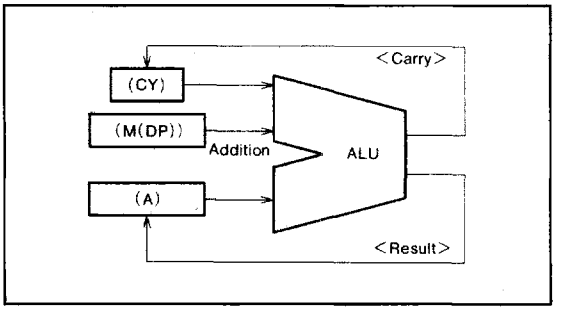


Fig. 1 AMC instruction execution example

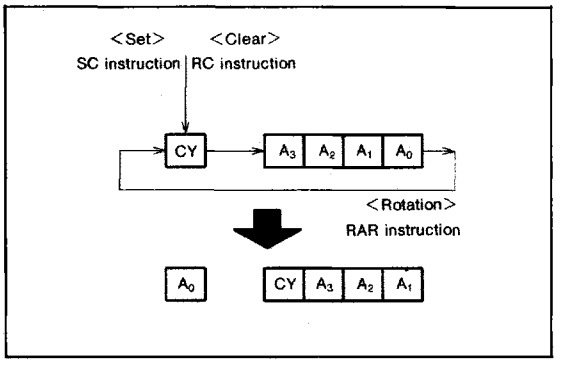


Fig. 2 RAR instruction execution example

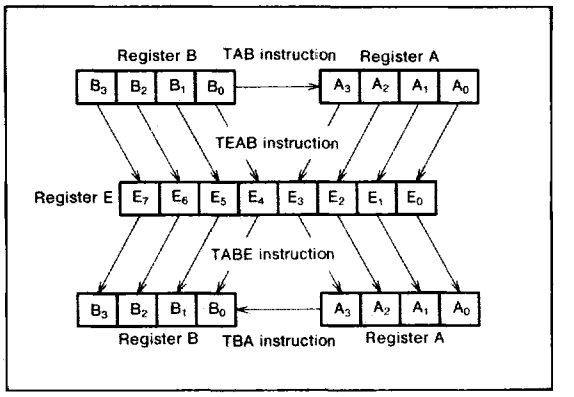


Fig. 3 Registers A, B and register E

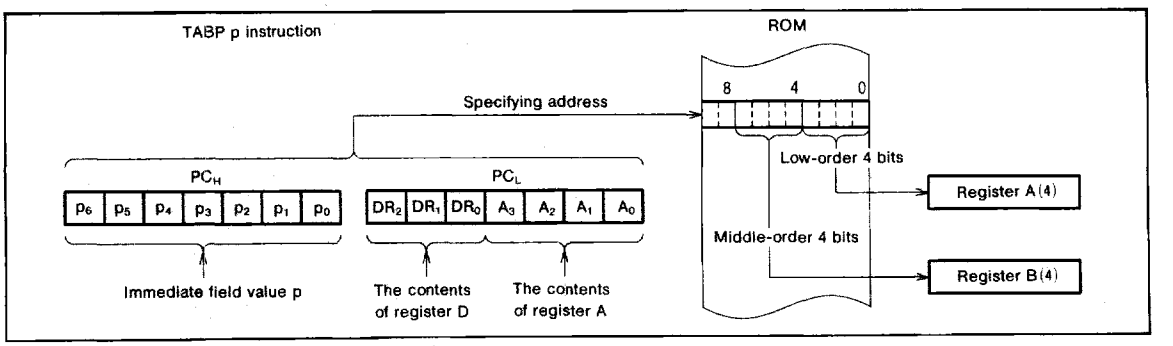


Fig. 4 TABP p instruction execution example

MITSUBISHI ELECTRIC

STACK REGISTERS (SKs)

Stack registers SKs are used to temporarily store the contents of program counter (PC) just before branching until returning to the original routine when;

- branching to an interrupt service routine (referred to as an interrupt service routine),
- performing a subroutine call, or
- executing the table reference instruction (TABP p).

Stack registers (SKs) are eight identical registers, so that subroutines can be nested up to 8 levels. However, one of stack registers is used respectively when using an interrupt service routine and when executing a table reference instruction. Accordingly, be careful not to over the stack when performing these operations together. The contents of registers SKs are destroyed when 8 levels are exceeded.

The register SK nesting level is pointed automatically by 3-bit stack pointer (SP). The contents of the stack pointer (SP) can be transferred to register A with the TASP instruction.

INTERRUPT STACK REGISTER (SDP)

Interrupt stack register (SDP) is a 1-stage register. When an interrupt occurs, this register (SDP) is used to temporarily store the contents of data pointer, carry flag, skip flag, register A, and register B just before an interrupt until returning to the original routine.

Unlike the stack registers (SKs), this register SDP is not used when executing the subroutine call instruction and the table reference instruction.

SKIP FLAG

Skip flag controls skip decision for the conditional skip instructions and continuous described skip instructions. When an interrupt occurs, the contents of skip flag is stored automatically in the interrupt stack register (SDP) and the skip condition is retained.

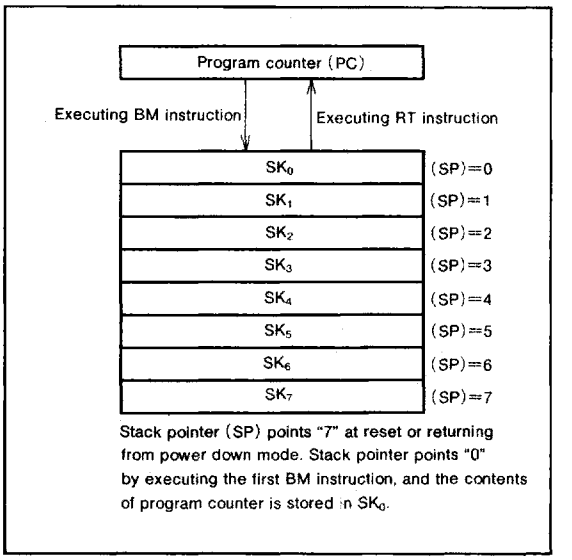


Fig. 5 Stack register (SK) structure

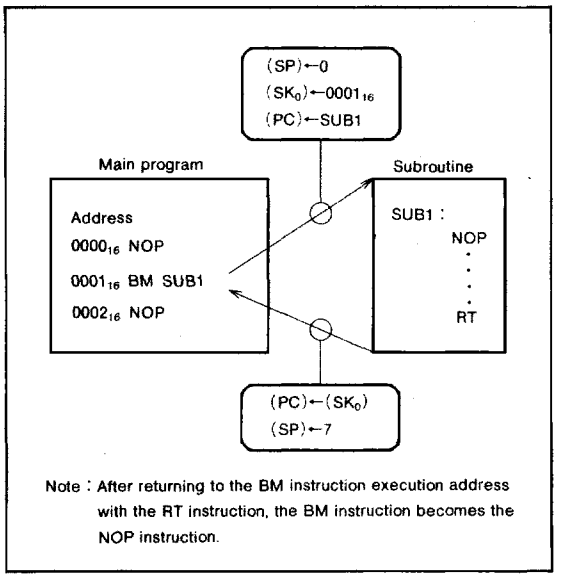


Fig. 6 Example of operation at subroutine call

PROGRAM MEMORY (ROM)

The program memory is the mask ROM. 1 word of ROM is composed of 10 bits. ROM is separated every 128 words by the unit of page (addresses 0 to 127).

Table 1 ROM size and pages

Product	ROM size (×10 bits)	Pages
M34520M6A	6144 words	48(0 to 47)
M34520M8A/E8	8192 words	64(0 to 63)

A part of page 1 (addresses 0080_{16} to $00FF_{16}$) is reserved for interrupt addresses. When an interrupt occurs, the address (interrupt address) corresponding to each interrupt is set in the program counter, and the instruction at the interrupt address is executed. When using an interrupt service routine, write the instruction generating the branch to that routine at an interrupt address.

Page 2 (addresses 0100_{16} to $017F_{16}$) is the special page for subroutine calls. Subroutines written in this page can be called from any page with the 1-word instruction (BM). Subroutines extending from page 2 to another page can also be called with the BM instruction when it starts on page 2.

All pages can be used as data areas with the TABP p instruction.

PROGRAM COUNTER (PC)

Program counter (PC) is used to specify a ROM address (page and address). It determines a sequence in which the instructions stored in ROM are read. It is a binary counter that increments the number of instruction bytes each time an instruction is executed. However, the value changes to the specified address when branch instructions, subroutine call instructions, return instructions, or the table reference instruction (TABP p) is executed.

The program counter consists of PC_H (most significant bit to bit 7) which specifies a ROM page and PC_L (bits 6 to 0) which specifies an address within a page. After it reaches the last address (address 127) of a page, it specifies address 0 of the next page.

Make sure that the PC_H does not specify after the last page of the built-in ROM.

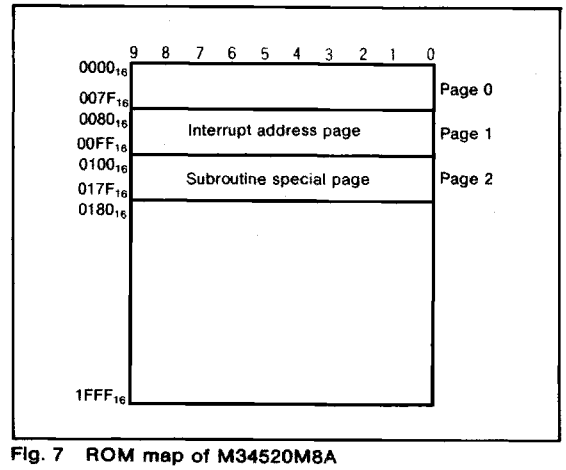


Fig. 7 ROM map of M34520M8A

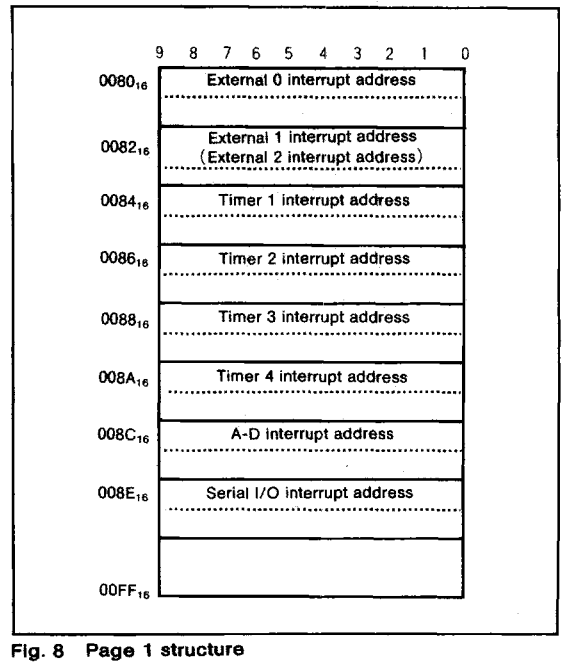


Fig. 8 Page 1 structure

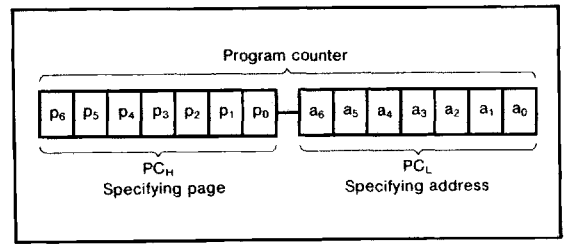


Fig. 9 Program counter structure

DATA MEMORY (RAM)

1 word of RAM is composed of 4 bits, but 1-bit manipulation (with the SB j, RB j, and SZB j instructions) is enabled for the entire memory area. A RAM address is specified by a data pointer. The data pointer consists of registers Z, X, and Y. Set a value to the data pointer certainly when executing an instruction to access RAM.

Table 2 RAM size

Product	RAM size
M34520M6A	384 words×4 bits
M34520M8A/E8	(1536 bits)

The RAM includes the area corresponding to the LCD. A segment is turned on automatically when "1" is written in the bit corresponding to the segment.

DATA POINTER (DP)

Data pointer (DP) is used to specify a RAM address and consists of registers Z, X, and Y. Register Z specifies a RAM file group, register X specifies a file, and register Y specifies a RAM digit.

Register Y is also used to specify the port D bit position. Set the value of register Y when using port D.

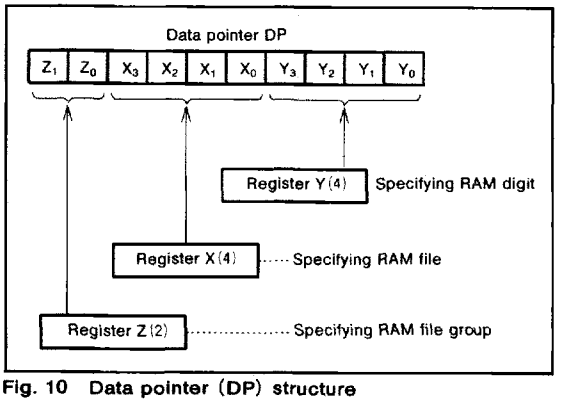


Fig. 10 Data pointer (DP) structure

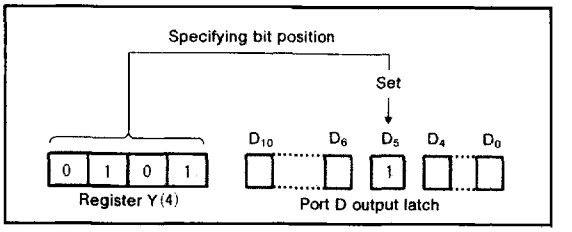


Fig. 11 SD instruction execution example

RAM 384 words×4 bits (1536 bits)

Register Y \ Register Z	0				1						
Register X	0	1	···	15	0	···	5	6	7	8	9
0								—	—	—	—
1								—	—	—	—
2								—	—	—	—
3								—	—	—	—
4								—	—	—	—
5								—	—	—	—
6								—	—	—	—
7								—	—	—	—
8								0	8	16	24
9								1	9	17	25
10								2	10	18	26
11								3	11	19	
12								4	12	20	
13								5	13	21	
14								6	14	22	
15								7	15	23	

Notes 1. The area marked "—" (Z=1, X=6 to 9, Y=0 to 7) is not a memory area.
2. The numbers in the shaded area represent the corresponding segment output pin numbers.

Fig. 12 RAM map

INTERRUPT FUNCTION

The interrupt type is a vectored interrupt branching to an individual address (interrupt address) according to each interrupt source. An interrupt occurs when the following 3 conditions are satisfied.

- An interrupt activated condition is satisfied (interrupt request flag="1")
- Interrupt enable bit is enabled ("1")
- Interrupt enable flag is enabled (INTE="1")

Table 3 shows the interrupt sources. (Refer to each interrupt request flag for details of activated conditions)

(1) Interrupt enable flag (INTE)

The interrupt enable flag (INTE) controls whether the every interrupt enable/disable. Interrupts are enabled when INTE flag is set to "1" with the EI instruction and disabled when INTE flag is cleared to "0" with the DI instruction. When any interrupt occurs, the INTE flag is automatically cleared to "0", so that other interrupts are disabled until the EI instruction is executed.

(2) Interrupt enable bit

Occurrence of each interrupt can be controlled with software. When an interrupt is not used, its corresponding skip instruction examines whether the interrupt activated condition is satisfied (whether the interrupt request flag="1") or not. Use an interrupt enable bit to select the corresponding interrupt or skip instruction.

(3) Interrupt request flag

When the activated condition for each interrupt is satisfied, the corresponding interrupt request flag is set to "1". Each interrupt request flag is cleared to "0" when either;

- an interrupt occurs, or
- the next instruction is skipped with a skip instruction.

Each interrupt request flag is set when the activated condition is satistied even if the interrupt is disabled by the interrupt enable flag (INTE) or its interrupt enable bit. Once set, the interrupt request flag retains set until a reset condition is satisfied.

Accordingly, an interrupt occurs when the interrupt disable state is released while the interrupt request flag is set.

If more than one interrupt request flag is set when interrupt disable state is released, the interrupt priority level is as follows shown in Table 3.

Table 3 Interrupt sources

Priority level	Interrupt name	Activated condition	Interrupt address
1	External 0 interrupt	Level change of INT_0 pin	Address 0 in page 1
2	External 1 interrupt	Level change of INT_1 pin	Address 2 in page 1
	External 2 interrupt	Level changes of pins INT_1 or INT_2 (OR of pins INT_1 and INT_2)	
3	Timer 1 interrupt	Timer 1 underflow	Address 4 in page 1
4	Timer 2 interrupt	Timer 2 underflow	Address 6 in page 1
5	Timer 3 interrupt	Timer 3 underflow	Address 8 in page 1
6	Timer 4 interrupt	Timer 4 underflow	Address A in page 1
7	A-D interrupt	Completion of A-D conversion	Address C in page 1
8	Serial I/O interrupt	Completion of serial I/O transfer	Address E in page 1

Table 4 Interrupt enable bit and skip instruction

Interrupt name	Request flag	Enable bit	Skip instruction
External 0 interrupt	EXF0	$V1_0$	SNZ0
External 1, 2 interrupts	EXF1	$V1_1$	SNZ1
Timer 1 interrupt	T1F	$V1_2$	SNZT1
Timer 2 interrupt	T2F	$V1_3$	SNZT2
Timer 3 interrupt	T3F	$V2_0$	SNZT3
Timer 4 interrupt	T4F	$V2_1$	SNZT4
A-D interrupt	ADF	$V2_2$	SNZAD
Serial I/O interrupt	SIOF	$V2_3$	SNZSI

Table 5 Interrupt enable bit function

Interrupt enable bit	Occurrence of interrupt	Skip instruction
1	Enabled	Invalid
0	Disabled	Valid

(4) Internal state during an interrupt

The internal state of the microcomputer during an interrupt is as follows.

- Program counter (PC)
  An interrupt address is set in program counter. The address to be executed when returning to the main routine is automatically stored in the stack register (SK).
- Interrupt enable flag (INTE)
  INTE flag is cleared to "0" so that interrupts are disabled.
- Interrupt request flag
  Only the request flag for the current interrupt source is cleared to "0".
- Data pointer, carry flag, skip flag, registers A and B
  The contents of these registers and flags are stored in the interrupt stack register (SDP).

(5) Interrupt processing

When an interrupt occurs, a program at an interrupt address is executed after branching a data storing sequence to stack register. Write the branch instruction to an interrupt service routine at an interrupt address. Use the RTI instruction to return from an interrupt service routine.

Interrupt enabled by executing the EI instruction is performed after executing 1 instruction (just after the next instruction is executed). Accordingly, when the EI instruction is executed just before the RTI instruction, interrupts are enabled after returning the main routine. (Refer to Fig.16)

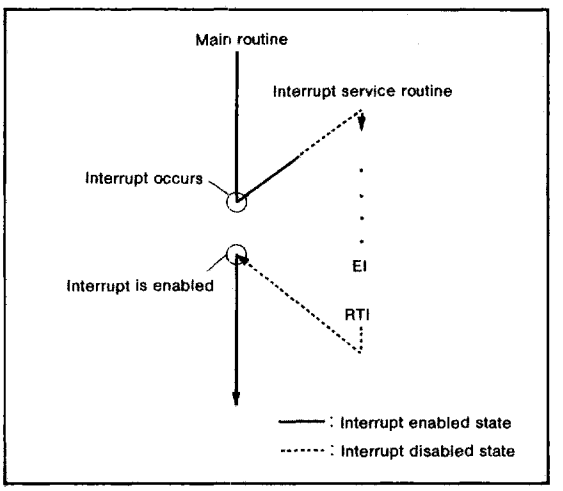


Fig. 13 Program example of interrupt processing

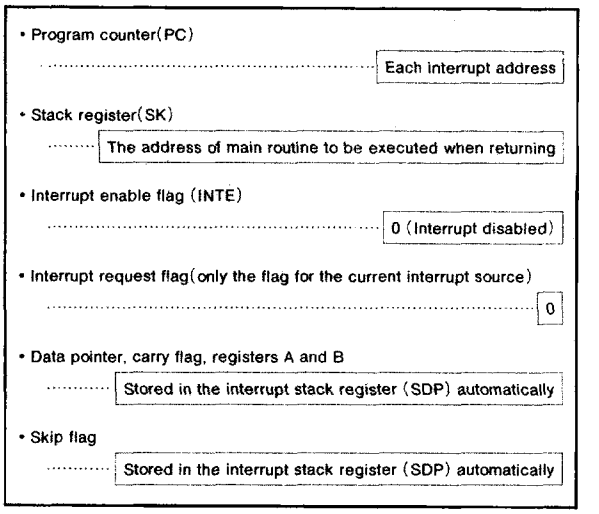


Fig. 14 Internal state when interrupt occurs

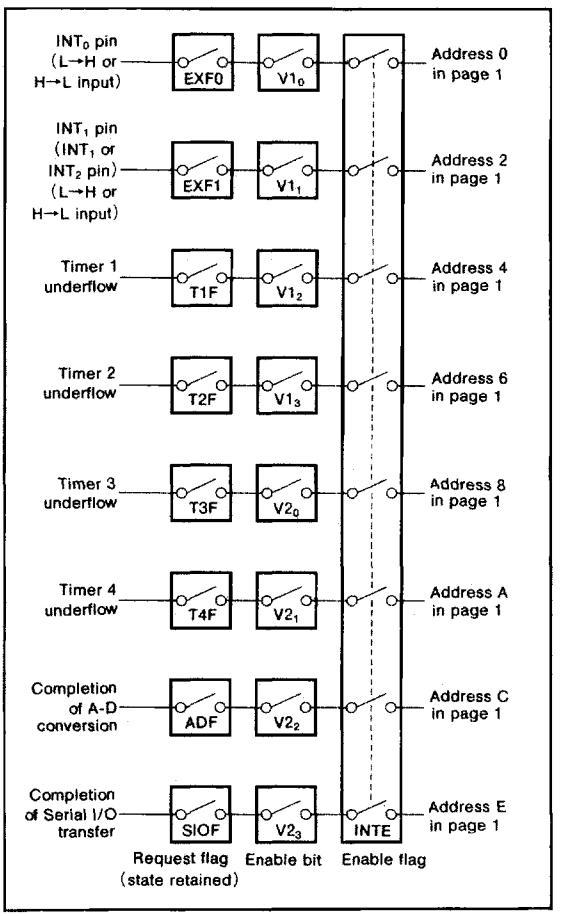
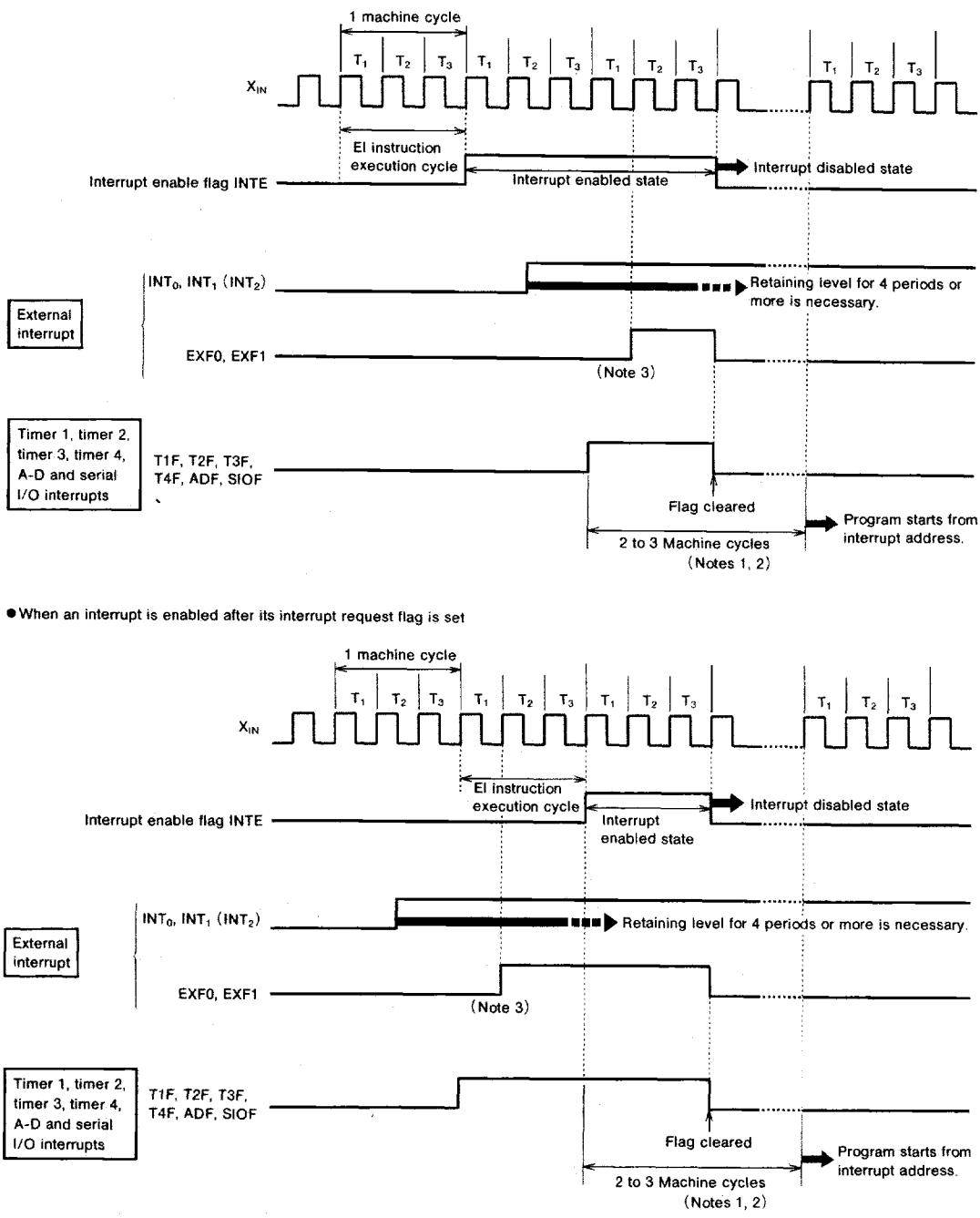


Fig. 15 Interrupt system diagram



Notes 1. This interval of cycles depends on the instruction being executed when EXF0, EXF1, T1F, T2F, T3F, T4F, ADF and SIOF flags are set to "1".

2. Address is stacked to the last cycle.

3. EXF0 and EXF1 flags are set at the second period.

Fig. 16 Interrupt sequence

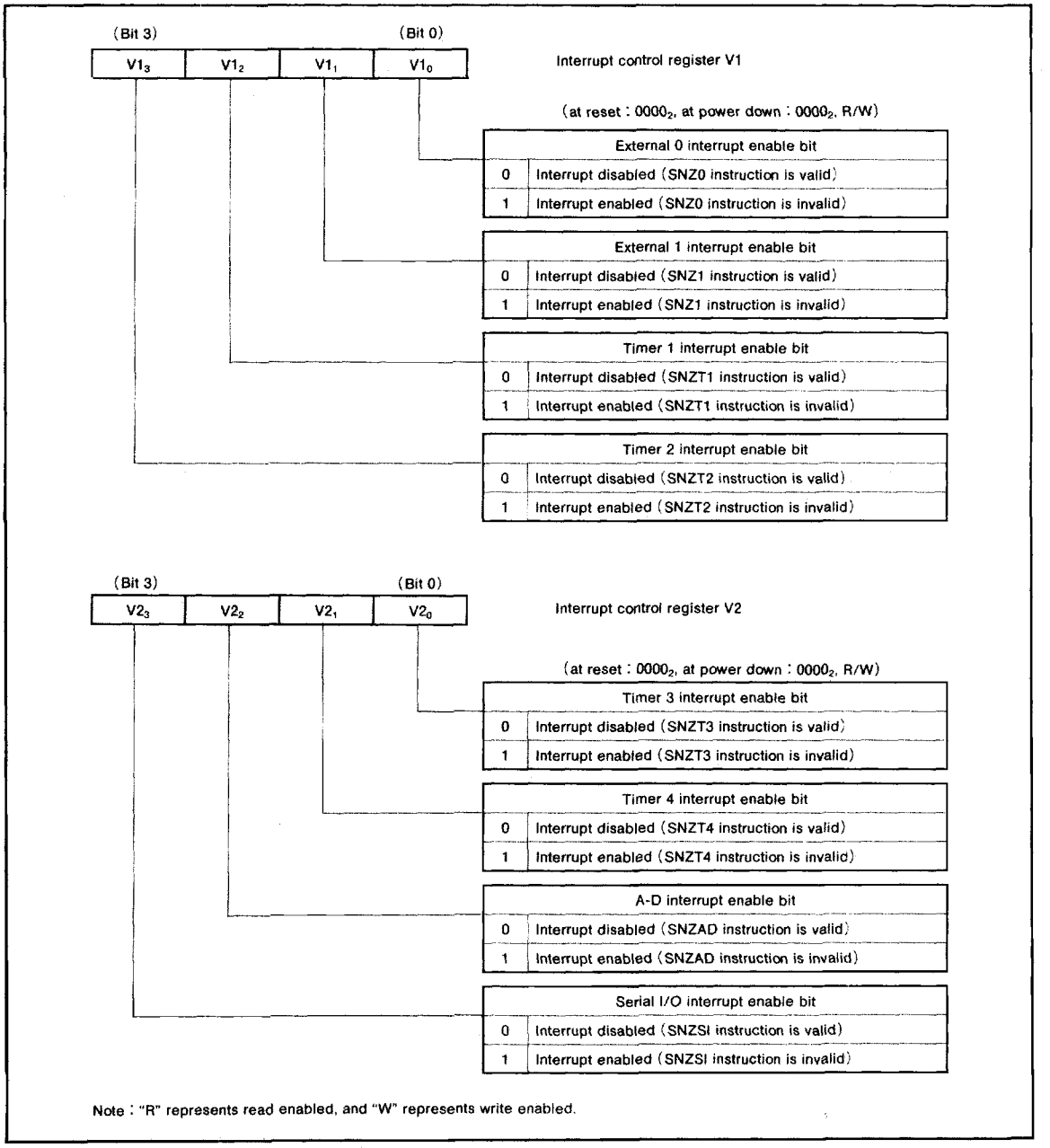
(6) Interrupt control register

• Interrupt control register (V1)

Interrupt enable bits of external 0, external 1, timer 1 and timer 2 are assigned to register V1. Set the contents of this register through register A with the TV1A instruction. The TAV1 instruction can be used to transfer the contents of register V1 to register A.

• Interrupt control register (V2)

Interrupt enable bits of timer 3, timer 4, A-D and serial I/O are assigned to register V2. Set the contents of this register through register A with the TV2A instruction. The TAV2 instruction can be used to transfer the contents of register V2 to register A.



Note : "R" represents read enabled, and "W" represents write enabled.

Fig. 17 Interrupt control register

EXTERNAL INTERRUPTS

An external interrupt request occurs (edge detect) when an valid waveform is input to the interrupt input pin. The 4520 Group has 3 external interrupt functions (external 0, external 1, and external 2).

External 0 interrupt is equipped with a noise detection circuit and a zero cross detection circuit. The noise detection circuit invalidates the first edge as noise when the second valid edge is detected within a certain interval after a valid edge is detected. The zero cross detection circuit detects the point when the voltage level of an alternating waveform passes 0V. The external interrupts can be controlled with the interrupt control registers (I1, I2, and I3).

Table 6 Interrupt activated condition

Interrupt name	Input pin	Activated condition	Valid edge selection bit
External 0 interrupt	$D_8/INT_0/ZEROX$	When the next waveform is input to INT_0 pin • Falling edge ("H"→"L") • Rising edge ("L"→"H") • Both rising edge and falling edge	$I1_1$ $I1_2$
External 1 interrupt	$P2_1/INT_1$	When the next waveform is input to INT_1 pin • Falling edge ("H"→"L") • Rising edge ("L"→"H")	$I2_2$
External 2 interrupt	$P2_1/INT_1$, $SEG_{26}/P3_0/INT_2/A_{IN0}$	When the next waveform is input to INT_1 pin or INT_2 pin (OR operation between INT_1 pin and INT_2 pin) • Falling edge ("H"→"L") • Rising edge ("L"→"H")	

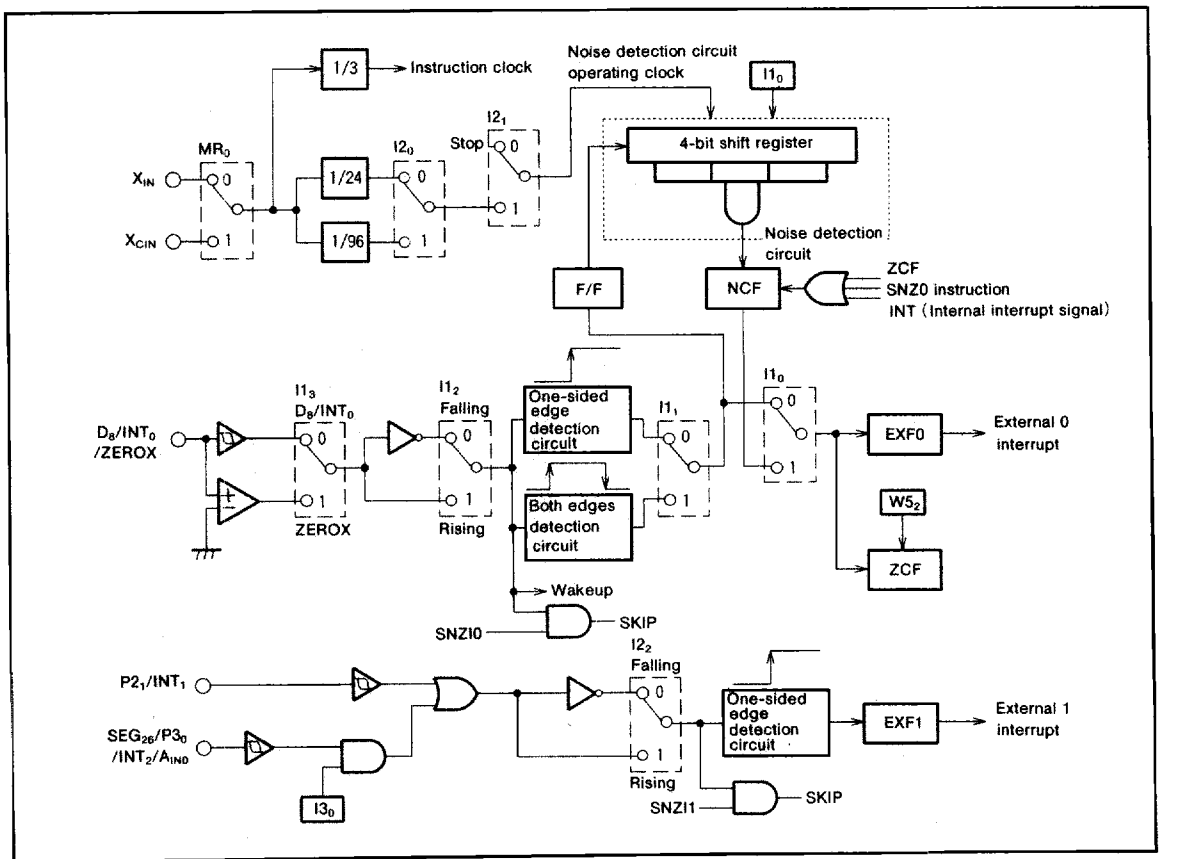
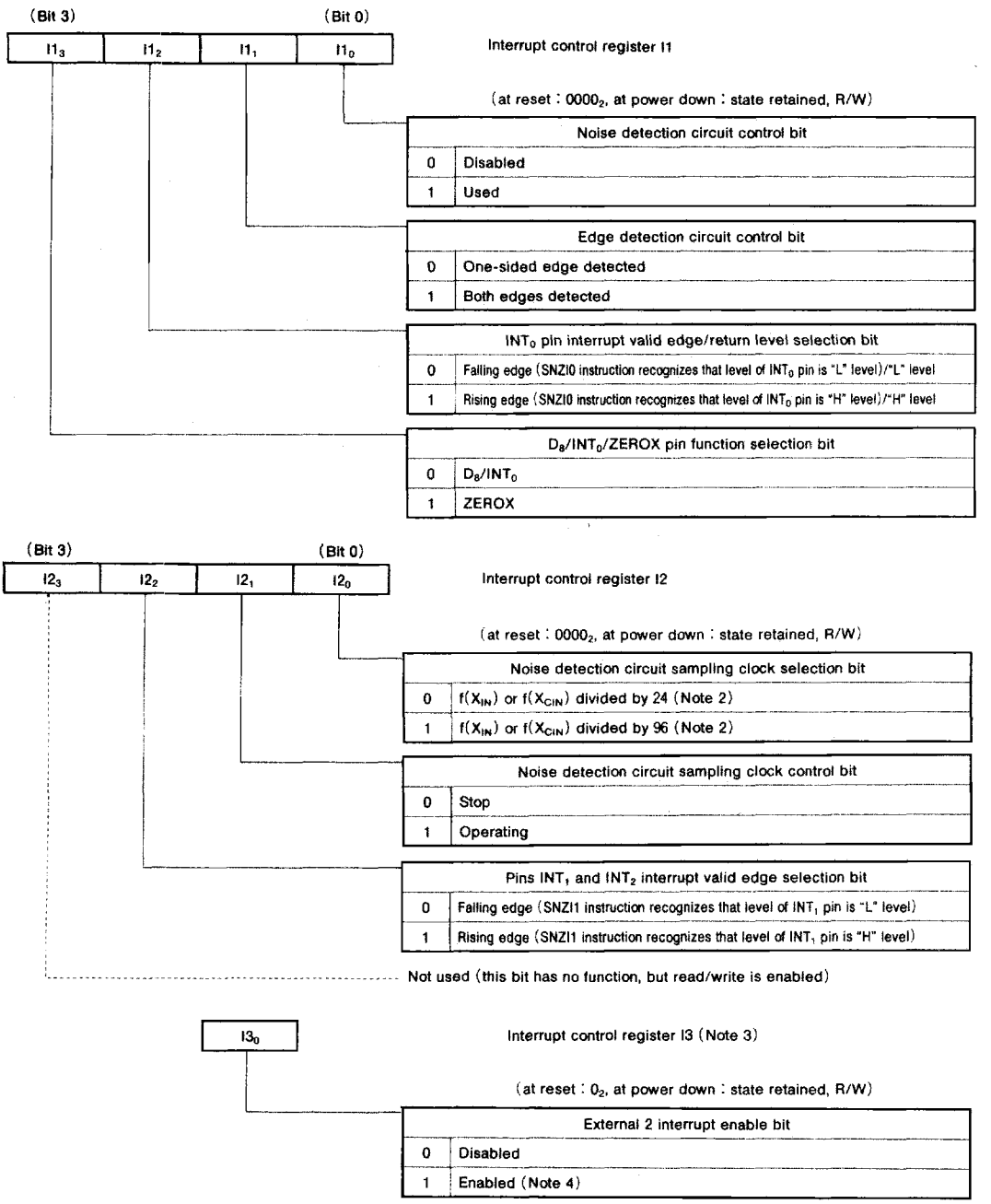


Fig. 18 External interrupt circuit structure



Notes 1. "R" represents read enabled, and "W" represents write enabled.

2. The signal in $f(X_{IN})$ or $f(X_{CIN})$ selected as system clock is used.

3. After executing the TAI3 instruction, "0" is stored in bits 3, 2 and 1 of register A.

4. External 2 interrupt activated condition is OR operation between INT_1 pin and INT_2 pin.

Fig. 19 External interrupt control registers

(1) External 0 interrupt request flag (EXF0)

The external 0 interrupt request flag (EXF0) is set to "1" when a valid waveform is input to the INT_0 pin. The valid edge polarity can be selected from rising edge , falling edge, or rising and falling edges. An example of how to use the external 0 interrupt is shown below.

① Set bits 0 and 3 of register I1 to "0" and select the valid edge polarity with bits 1 and 2.
② Clear EXF0 flag to "0" with the SNZ0 instruction.
③ Set the external 0 interrupt enable bit ($V1_0$) and the interrupt enable flag (INTE) to "1".

External 0 interrupt is now enabled. Now when a valid waveform is input to the INT_0 pin, the EXF0 flag is set to "1" and an external 0 interrupt occurs.

The external 0 interrupt circuit has a noise detection function and a zero cross detection function. The above usage does not involve these functions, and in this case the waveform causing the interrupt must be retained at their level for 4 periods or more of the signal used as the system clock. (Refer to Fig. 16)

The state of the EXF0 flag can be examined with the skip instruction (SNZ0). Use the interrupt control register (V1) to select the interrupt or the skip instruction.

EXF0 flag is cleared to "0" when the interrupt occurs or when the next instruction is skipped with the skip instruction.

The $D_8/INT_0/ZEROX$ pin need not be selected the external interrupt input INT_0 function or the normal I/O port D_8 function. However, the EXF0 flag is set to "1" when a valid waveform is input even if it is used as an I/O port D_8.

(2) External 1 interrupt request flag (EXF1)

The external 1 interrupt request flag (EXF1) is set to "1" when an activated condition of either an external 1 interrupt or external 2 interrupt is satisfied. The waveforms causing external 1 interrupt and external 2 interrupt must be retained at their level for 4 periods or more of the signal used as the system clock. (Refer to Fig. 16)

The state of the EXF1 flag can be examined with the skip instruction (SNZ1). Use the interrupt control register (V1) to select the interrupt or the skip instruction.

EXF1 flag is cleared to "0" when the interrupt occurs or when the next instruction is skipped with the skip instruction.

The $P2_1/INT_1$ pin need not be selected the external interrupt input INT_1 function or the input port $P2_1$ function. However, the EXF1 flag is set to "1" when a valid waveform is input even if it is used as an input port $P2_1$.

• External 1 interrupt activated condition

An external 1 interrupt activated condition is satisfied when a valid waveform is input to the INT_1 pin.

The valid edge polarity can be selected from falling edge or rising edge. An example of how to use the external 1 interrupt is shown below.

① Clear register I3 to "0"
② Select a valid edge polarity with the bit 2 of register I2.
③ Clear EXF1 flag to "0" with the SNZ1 instruction.
④ Set both the external 1 interrupt enable bit ($V1_1$) and the interrupt enable flag (INTE) to "1".

External 1 interrupt is now enabled. Now when a valid waveform is input to the INT_1 pin, the EXF1 flag is set to "1" and an external 1 interrupt occurs.

• External 2 interrupt activated condition

An external 2 interrupt activated condition is satisfied when a valid waveform is input to INT_1 pin or INT_2 pin (OR operation between INT_1 and INT_2). An example of how to use the external 2 interrupt is shown below.

① Set the bit 0 of LCD control register (L2) to "1" and register I3 to "1".

Perform steps ② to ④ for external 1 interrupt. Then the external 2 interrupt is enabled. Now when a valid waveform is input to INT_1 pin or INT_2 pin, the EXF1 flag is set to "1" and the external 2 interrupt occurs.

(3) Noise detection circuit

The external 0 interrupt is equipped with a noise detection circuit which invalidates the first edge as noise when the second valid edge is detected within a certain interval after a valid edge is detected. (Refer to Fig. 21) This noise detection circuit consists of 4-bit shift register and a noise canceller flag (NCF).

The clock used at the noise detection circuit is $f(X_{IN})$ or $f(X_{CIN})$ selected as the system clock by register MR. The signal obtained by dividing the frequency of this clock by 24 or 96 is used as the sampling clock of the noise detection circuit.

The interrupt activated conditions (valid edge interval) for external 0 interrupt using the noise detection circuit are as follows:

- When the signal of frequency divided by 24 is selected as sampling clock:
  Valid edge interval is 34 machine cycles or greater.
- When the signal of frequency divided by 96 is selected as sampling clock:
  Valid edge interval is 120 machine cycles or greater.

(1 machine cycle (sec) = $3/f(X_{IN})$ or $3/f(X_{CIN})$)

When the valid edge interval is less than the above, the previously detected edge is assumed to be noise and invalidated. However, the allowed values for the valid edge interval changes according to the internal state when the valid edge is input. The actual allowed values for the valid edge interval are 27 to 34 machine cycles or greater when the signal of frequency divided by 24 is selected and 99 to 120 machine cycles or greater when the signal of frequency divided by 96 is selected.

An example of how to use the external 0 interrupt using the noise detection circuit is shown below.

① Set the bit 0 of register I1 to "1", select the valid waveform (=valid edge) with bits 1 and 2, and the pin function with bit 3.
② Clear the external 0 interrupt request flag (EXF0) to "0" with the SNZ0 instruction.
③ Set both the external 0 interrupt enable bit ($V1_0$) and interrupt enable flag (INTE) to "1".
④ Select the sampling clock with the bit 0 of register I2 and set the bit 1 to "1".

External 0 interrupt is now enabled. Now when a valid waveform is input to the INT_0 pin, the NCF flag is set to "1" which in turn sets the EXF0 flag to "1" and an external 0 interrupt occurs.

The NCF flag is cleared to "0" when the external 0 interrupt occurs, when the SNZ0 instruction is executed, or when the zero cross input flag is set to "1".

(4) NOTES ON USING THE NOISE DETECTION CIRCUIT

Note the SNZ0 instruction execution timing when using the noise detection circuit. If the SNZ0 instruction is executed at the timing when the noise canceller flag (NCF) is being set, the conditions for setting the external 0 interrupt request flag (EXF0) and zero cross input flag (ZCF) become invalid.

Even when the noise detection circuit is used, an interrupt occurs when the waveform shown in Fig. 20 is input.

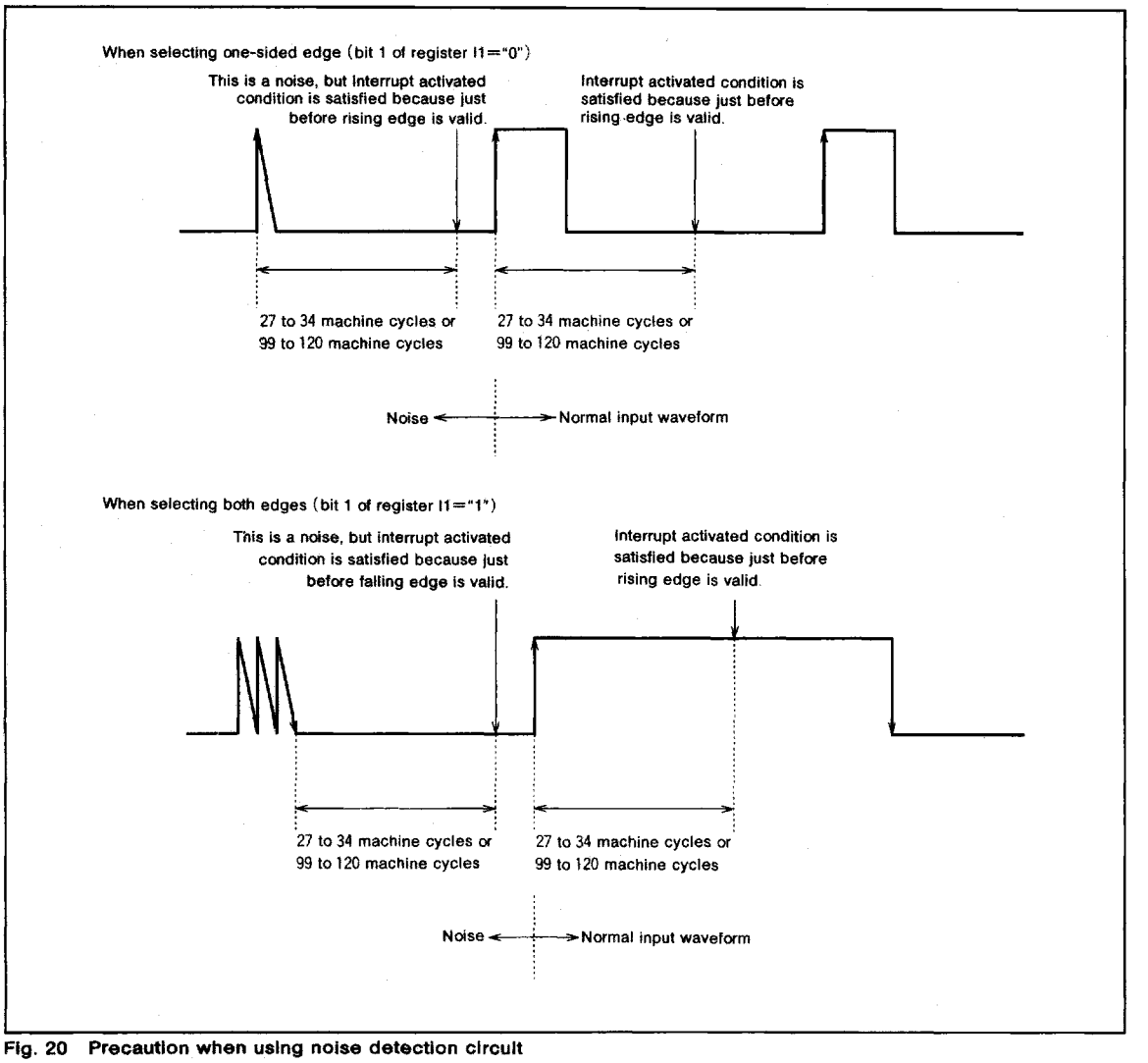


Fig. 20 Precaution when using noise detection circuit

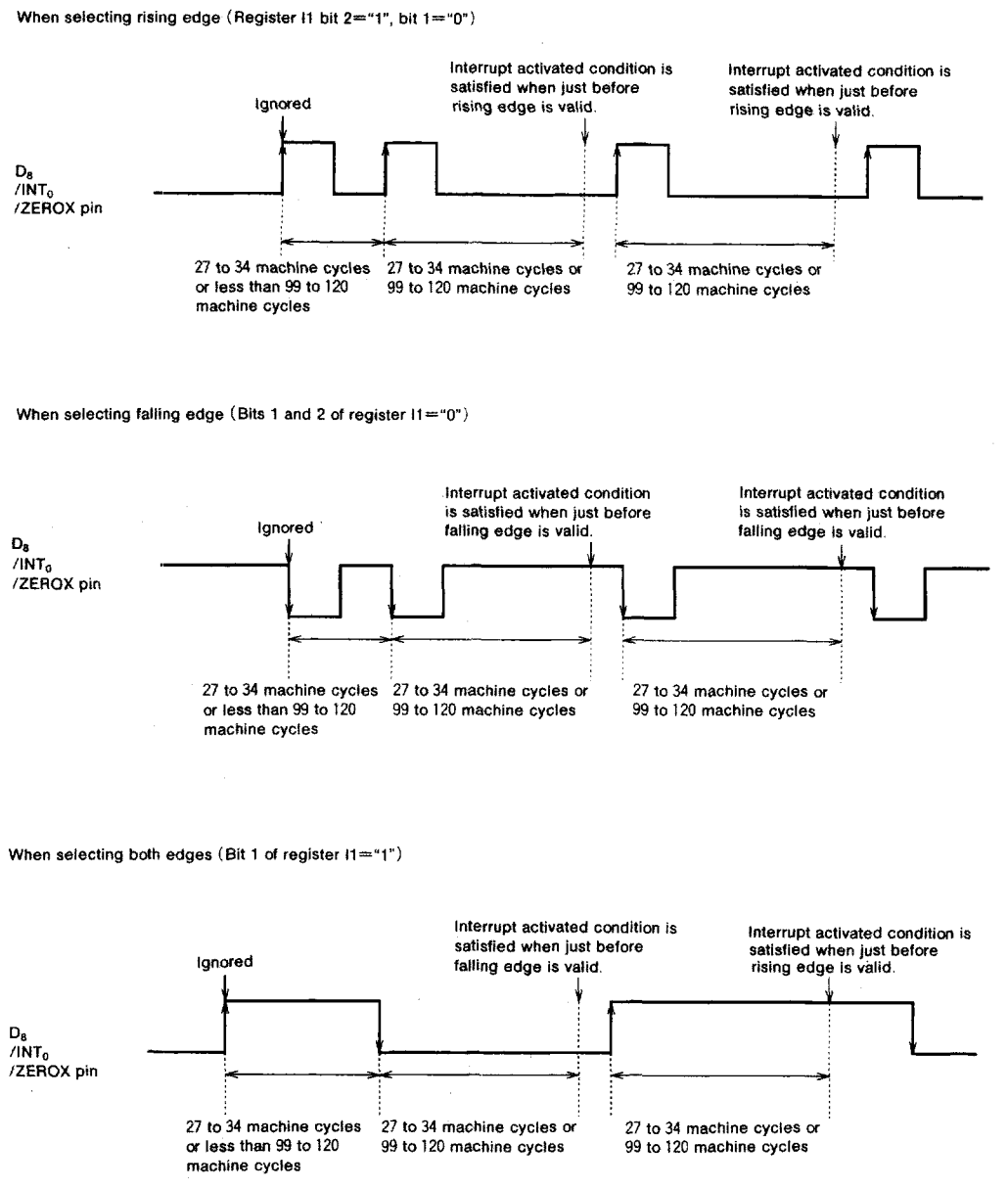


Fig. 21 Noise detection circuit operation

(5) Zero cross detection circuit

The external 0 interrupt is equipped with a zero cross detection circuit which detects the point when the voltage level of the alternating waveform passes 0V. Zero cross point to be detected can be selected when changing from minus to plus, plus to minus, or both.

The zero cross detection circuit can be used together with the noise detection circuit. When using together with the noise detection circuit, the zero cross interval of the input waveform must be equal to or greater than the allowed values for the valid edge interval (refer to description of the noise detection circuit). In addition, the zero cross point detected with the zero cross detection circuit can be used as the trigger to start timer 2 count operation (refer to description of zero cross input flag).

An example of how to use the external 0 interrupt using the zero cross detection circuit is shown below.

① Set bit 3 of register I1 to "1" and select the valid edge (=zero cross point to be detected) with bits 1 and 2.

② Clear the external 0 interrupt request flag (EXF0) to "0" with the SNZ0 instruction.

③ Set both external 0 interrupt enable bit ($V1_0$) and interrupt enable flag (INTE) to "1".

External 0 interrupt is now enabled. Now when an alternating waveform is input to the ZEROX pin, zero cross is detected, EXF0 flag is set to "1", and an external 0 interrupt occurs. The interval between the time from zero cross is detected until the time program at the interrupt address is executed is 3 to 4 machine cycles (the allowed value of edge interval is added when the noise detection circuit is used).

(6) Zero cross input flag (ZCF)

The ZCF flag is used to start timer 2 count operation when a zero cross point is detected with the zero cross detection circuit.

The ZCF flag is set to "1" when the voltage level of an alternating waveform input to the ZEROX pin passes 0V. When the bit 2 of timer control register (W2) is set to "1", timer 2 starts counting when the ZCF flag is set and stops counting when the ZCF flag is cleared.

The ZCF flag can be controlled by the bit 2 ($W5_2$) of timer control register (W5). It can be set (operation state) when $W5_2$ is set to "1". It cannot be set (stop state) because it is fixed to "0" when $W5_2$ is cleared to "0". Clear $W5_2$ to "0" when the ZCF flag is cleared. However, when the ZCF flag is to be used after reset, set $W5_2$ to "1" and return it to the operation state.

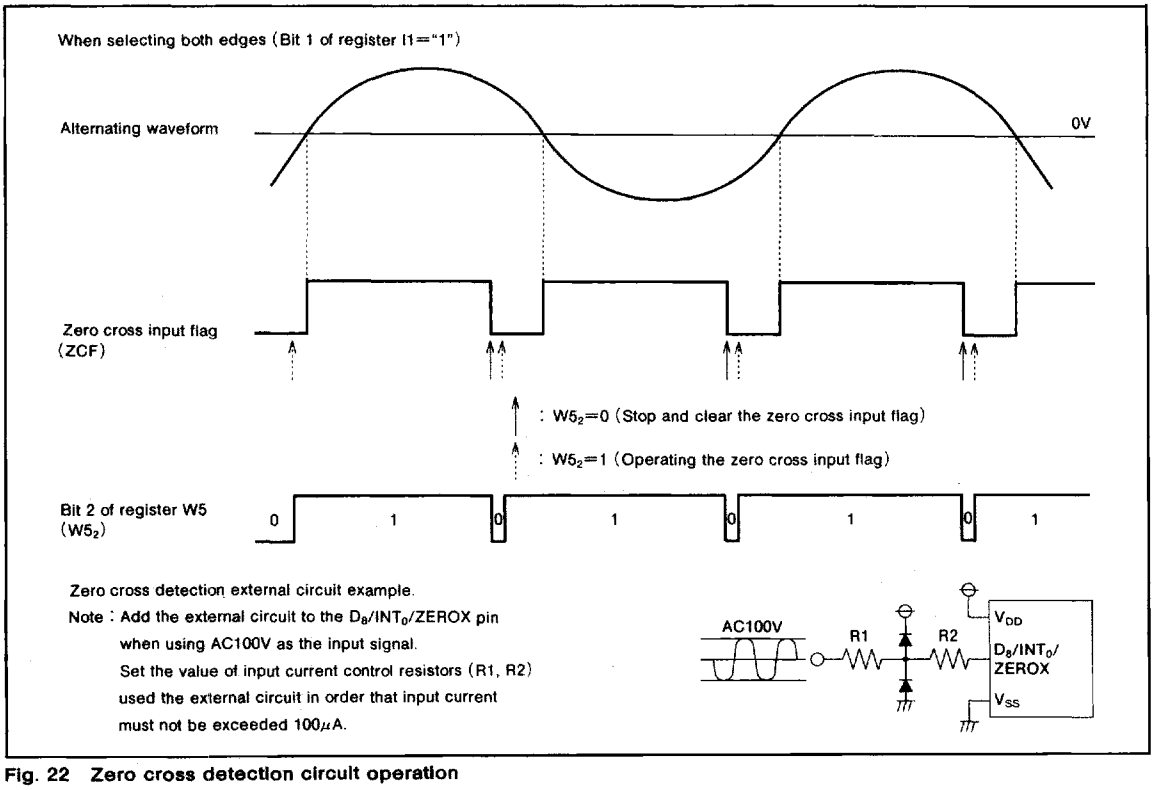


Fig. 22 Zero cross detection circuit operation

(7) External interrupt control register

- Interrupt control register (I1)
  Register I1 controls the operation of noise detection circuit, external 0 interrupt valid edge and return level from power down (valid level of wakeup signal) and D_8/INT_0/ZEROX pin function. Set the contents of this register through register A with the TI1A instruction. The TAI1 instruction can be used to transfer the contents of register I1 to register A.
- Interrupt control register (I2)
  Register I2 controls the sampling clock of noise detection circuit and external 1 and 2 interrupt valid edges. Set the contents of this register through register A with the TI2A instruction. The TAI2 instruction can be used to transfer the contents of register I2 to register A.
- Interrupt control register (I3)
  Register I3 controls the occurrence of external 2 interrupt. Set the contents of this register through register A with the TI3A instruction. The TAI3 instruction can be used to transfer the contents of register I3 to register A.

TIMERS

The 4520 Group has a programmable timer and a fixed dividing frequency timer.

- Programmable timer

  The programmable timer has a reload register and enables the frequency dividing ratio to be set. It is decremented from a setting value n. When it underflows (count to n+1), a timer interrupt request flag is set to "1", new data is loaded from the reload register, and count continues (auto-reload function).

- Fixed dividing frequency timer

  A fixed dividing frequency timer has a fixed frequency dividing ratio (n). The timer 3 interrupt request flag is set to "1" after every n count of the count pulse.

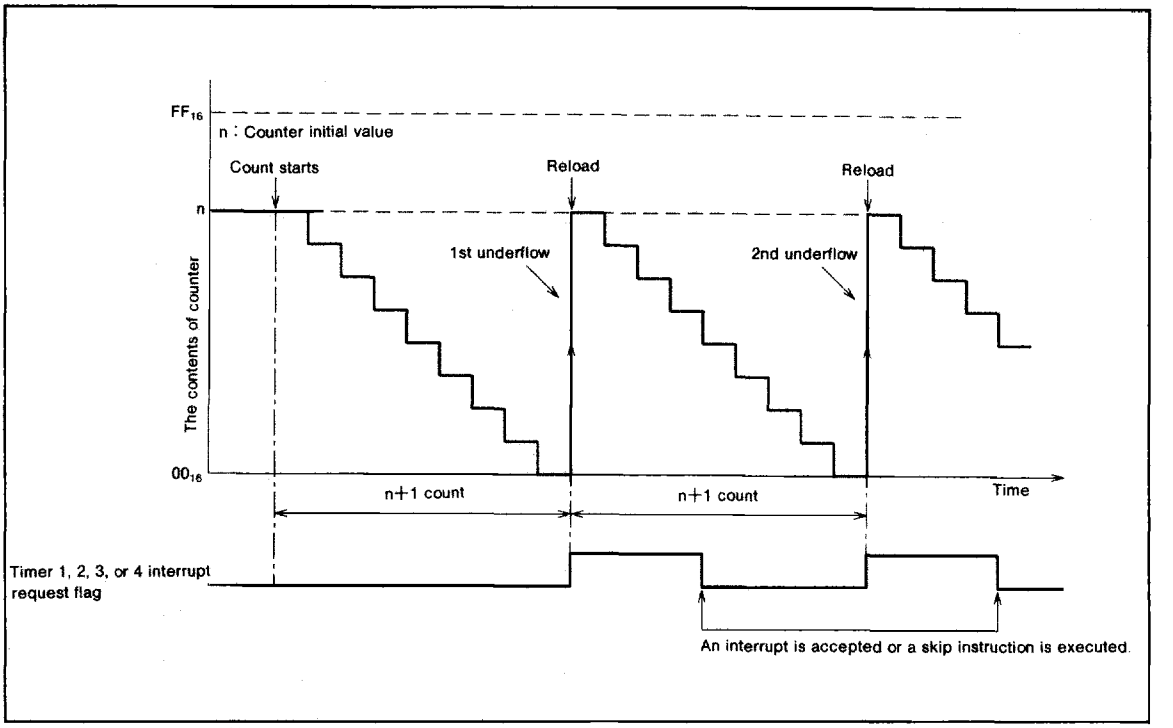


Fig. 23 Auto-reload function

The 4520 Group timer consists of the following circuits.
- Prescaler : frequency divider
- Timer 1 : 8-bit programmable timer
- Timer 2 : 8-bit programmable timer
- Timer 3 : 8-bit fixed dividing frequency timer
- Timer 4 : 8-bit programmable timer

(Timers 1 to 4 have the interrupt function)
- Watchdog timer
- Frequency divider for LCD
- PWM output
- Real time output

These timers can be controlled with the timer control registers (W1 to W5) and serial I/O mode registers (J1, J2). Each function is described below.

Table 7 Function related timers

Circuit	Structure	Count source	Frequency dividing ratio	Use of output signal	Control
Prescaler	Frequency divider	• $f(X_{IN})$ or $f(X_{CIN})$	6, 12	• Timer 1, 2 and 4 count sources	W1
Timer 1	8-bit programmable binary down counter	• Prescaler output (ORCLK)	1 to 256	• Timer 2 count source • $CNTR_0$ output • Timer 1 interrupt	W1
Timer 2	8-bit programmable binary down counter (Link ZCF flag)	• $f(X_{IN})$ • Timer 1 underflow • Prescaler output (ORCLK) • $CNTR_1$ input	1 to 256	• Timer 3, 4 count source. • PWM output • $CNTR_1$ output • Timer 2 interrupt	W2 W5
Timer 3	8-bit fixed dividing frequency binary down counter	• $f(X_{CIN})$ • Timer 2 underflow	256	• Timer 4 count source • Timer 3 interrupt	W3
	(Frequency divider (divide by 16))		(16)	• Frequency divider for LCD	
Timer 4	8-bit programmable binary down counter	• $f(X_{IN})$ • Timer 3 underflow • Timer 2 underflow • Prescaler output (ORCLK)	1 to 256	• Watchdog timer • PWM output • Real time output • Timer 4 interrupt • Power down 1 return	W4 W5
Watchdog timer	1-bit flag	• Timer 4 underflow		• System reset	W4
Frequency divider for LCD	4-bit counter +frequency divider (divide by 2)	• Timer 3 intermediate underflow • Timer 3 underflow	2(n+1) [n=0 to 15]	• LCD controller/driver	W3
PWM output	Flip flop	• Timer 2 underflow • Timer 4 underflow		• PWM output	W4 J2
Real time output	Output latch + output register	• Timer 4 underflow		• Real time output	J1 J2

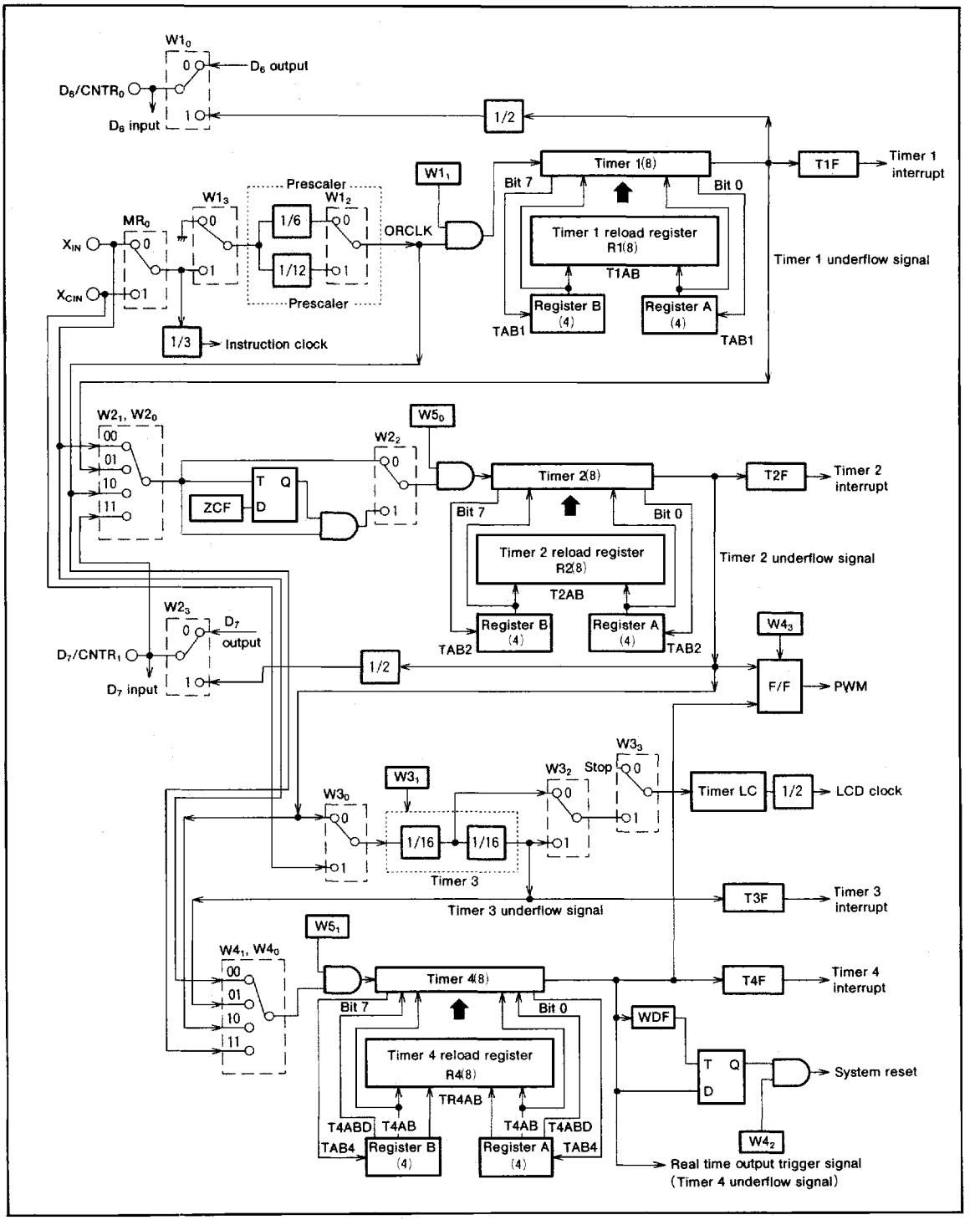
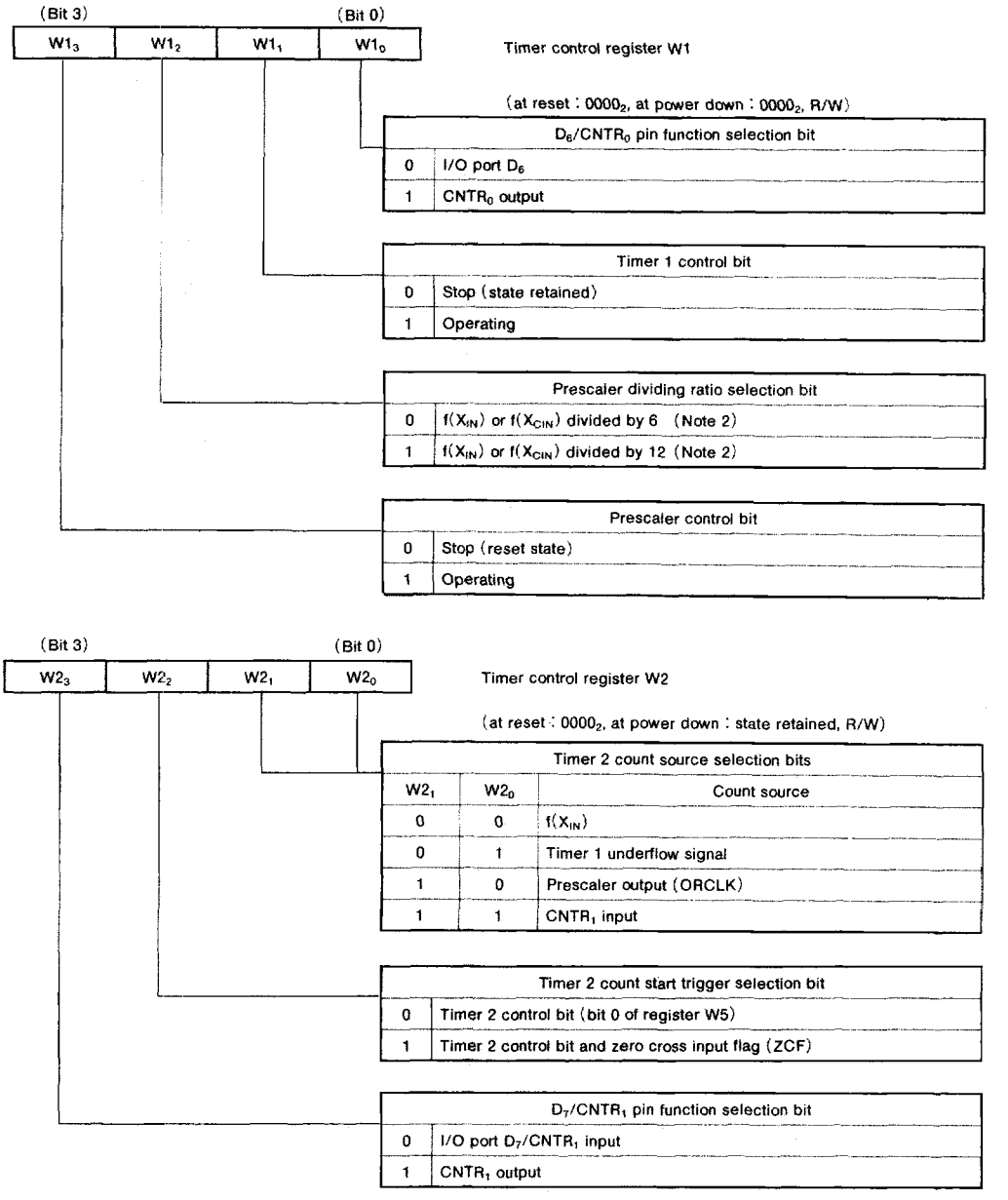


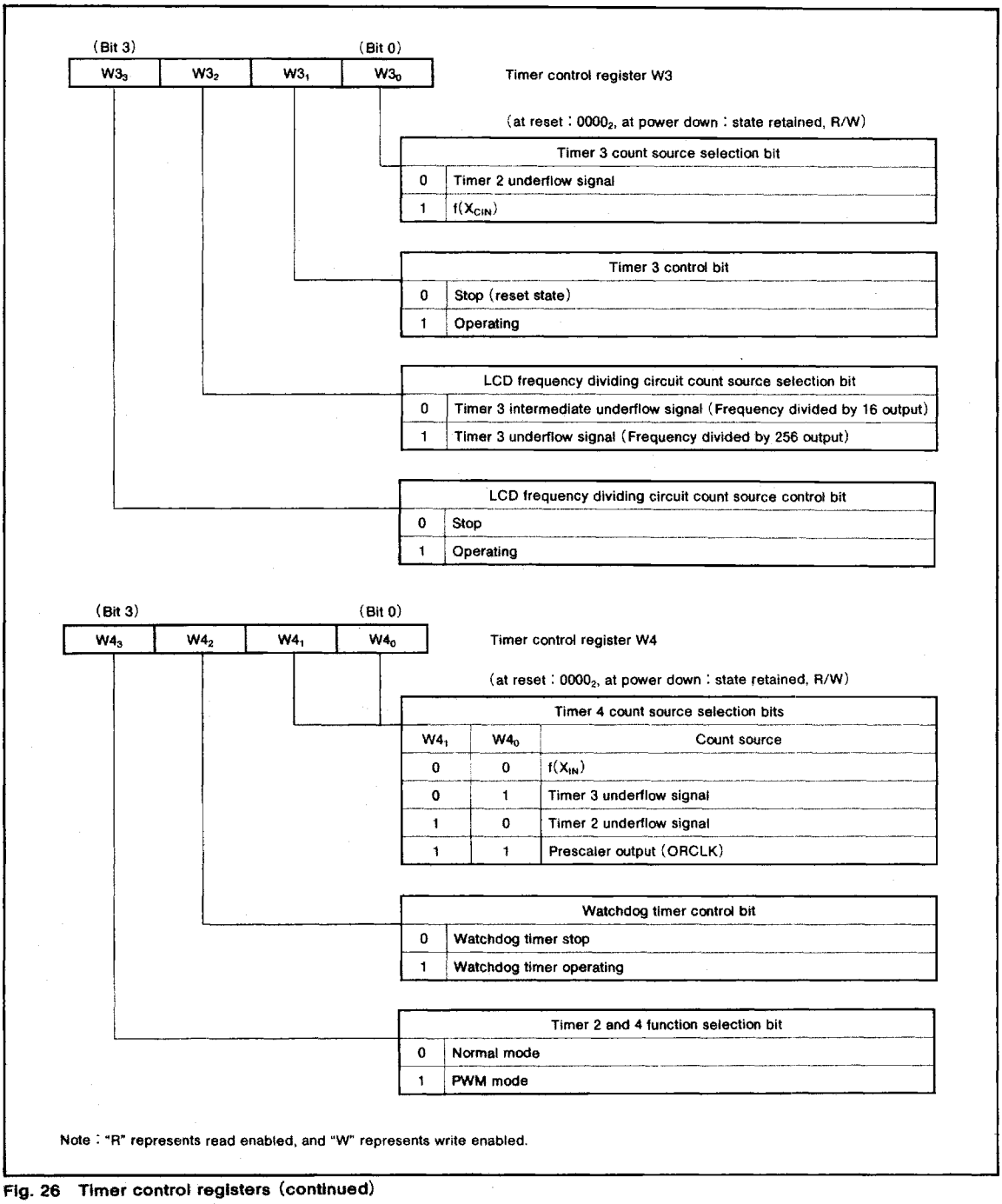
Fig. 24 Timers structure



Notes 1. "R" represents read enabled, and "W" represents write enabled.

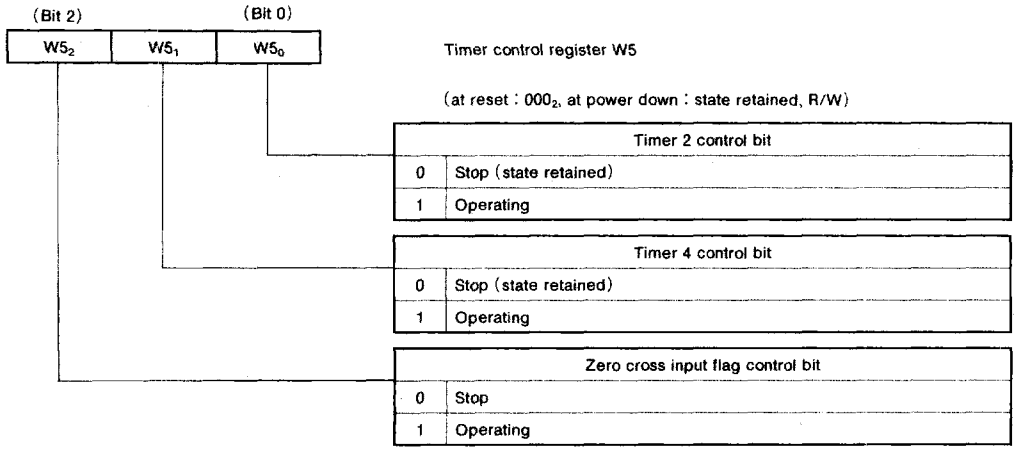
2. The signal in $f(X_{IN})$ or $f(X_{CIN})$ selected as a system clock is used.

Fig. 25 Timer control registers



Note : "R" represents read enabled, and "W" represents write enabled.

Fig. 26 Timer control registers (continued)



Notes 1. "R" represents read enabled, and "W" represents write enabled.

2. After executing the TAW5 instruction, "0" is stored in the bit 3 of register A.

Fig. 27 Timer control registers (continued)

(1) Prescaler

Prescaler is a frequency divider. Its frequency dividing ratio can be selected. Prescaler count source is $f(X_{IN})$ or $f(X_{CIN})$ which is a signal selected as a system clock with the register MR.

Use bit 2 of the register W1 to select the prescaler dividing ratio and bit 3 to start and stop its operation. Prescaler is reset state, and the output signal (ORCLK) stops when bit 3 of the register W1 are cleared to "0".

(2) Timer 1

Timer 1 is an 8-bit binary down counter with timer 1 reload register (R1). Data can be set simultaneously in timer 1 and reload register R1 with the T1AB instruction. Timer 1 starts counting after data is set in timer 1, and bit 1 of register W1 is set to "1".

When timer 1 underflows (the next count pulse is input after the contents of timer 1 becomes "0"), the timer 1 interrupt request flag (T1F) is set to "1", new data is loaded from the reload register R1, and count continues (auto-reload function). When a value set in reload register R1 is n, timer 1 divides the count source signal by n+1 (n=0 to 255).

Data can be read from timer 1 with the TAB1 instruction. When reading the data, stop the counter and then execute the TAB1 instruction. In addition, a signal obtained by dividing the frequency of the timer 1 underflow signal by 2 can be output from the D_6/$CNTR_0$ pin. Select the function of the D_6/$CNTR_0$ pin with the bit 0 of register W1.

(3) Timer 2

Timer 2 is an 8-bit binary down counter with timer 2 reload register (R2). Data can be set simultaneously in timer 2 and the reload register (R2) with the T2AB instruction. Timer 2 starts counting after setting data in timer 2, when the count source is selected with bits 0 and 1 of register W2 and the bit 0 of register W5 is set to "1". However, if the bit 2 of register W2 is set to "1", the zero cross input flag (ZCF) can be used as the timer 2 count start trigger (refer to description of external interrupts).

Once count is started, when timer 2 underflows (the next count pulse is input after the contents of timer 2 becomes "0"), the timer 2 interrupt request flag (T2F) is set to "1", new data is loaded from the reload register R2, and count continues (auto-reload function). When the value set in reload register R2 is n, timer 2 divides the count source signal by n+1 (n=0 to 255).

Data can be read from timer 2 with the TAB2 instruction. When reading the data, stop the counter and then execute the TAB2 instruction. In addition, a signal obtained by dividing the frequency of the timer 2 underflow signal by 2 can be output from the D_7/$CNTR_1$ pin. Select the function of the D_7/$CNTR_1$ pin with the bit 3 of register W2.

(4) Timer 3

Timer 3 is an 8-bit binary down counter. Timer 3 starts counting when a count source is selected with the bit 0 of register W3 and bit 1 is set to "1".

Once count is started, the timer 3 underflow flag (T3F) is set to "1" every 256 counts.

Timer 3 outputs both count source frequency by 16 (intermediate underflow) and by 256 (underflow). Timer 3 is cleared and both count source divided by 16 and 256 outputs are stopped when the bit 1 of register W3 is cleared to "0".

Timer 3 can be used as the clock counter during power down 1 state (executing the POF instruction).

(5) Timer 4

Timer 4 is an 8-bit binary down counter with timer 4 reload register (R4). Data can be set simultaneously in timer 4 and the reload register (R4) with the T4AB instruction. In addition, data can be written individually in timer 4 with the T4ABD instruction and in the reload register R4 with the TR4AB instruction. When data is written individually, count down after writing starts from the value set in timer 4, and count down after underflow starts from the value set in reload register R4.

Timer 4 starts counting after setting data in timer 4 when the count source is selected with bits 0 and 1 of register W4 and the bit 1 of register W5 is set to "1".

Once count is started, when timer 4 underflows (the next count pulse is input after the contents of timer 4 become "0"), the timer 4 interrupt request flag (T4F) is set to "1", new data is loaded from the reload register R4, and count continues (auto-reload function). When the value set is n, timer 4 divides the count source signal by n+1 (n=0 to 255).

Data can be read from timer 4 with the TAB4 instruction. When reading the data, stop the counter and then execute the TAB4 instruction. In addition, timer 4 can be used as the clock counter during power down 1 state (executing the POF instruction).

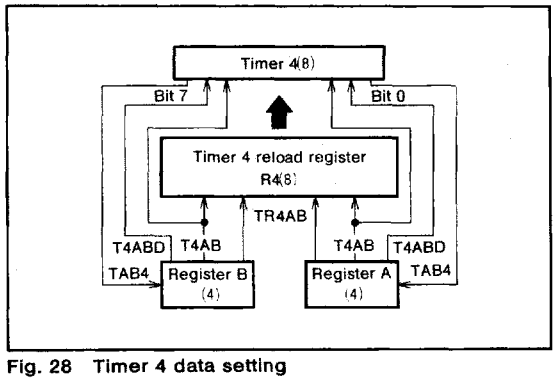


Fig. 28 Timer 4 data setting

(6) Watchdog timer

Watchdog timer consists of timer 4 and watchdog timer flag (WDF). When a timer 4 underflow signal occurs, the WDF flag is set to "1". When the timer 4 underflows once more while the WDF flag is set, the watchdog timer forces a system reset (operationally the same that power-on reset).

Whether to use watchdog timer can be set with the bit 2 of the register W4. When using watchdog timer, be sure to clear the WDF flag to "0" with the WRST instruction with a program before timer 4 underflows again. In order to effectively use watchdog timer, do not execute the WRST instruction during timer 4 interrupt.

(7) Timer interrupt request flags (T1F, T2F, T3F, and T4F)

Each timer interrupt request flag is set to "1" when each timer underflows. The state of these flags can be examined with skip instructions, respectively (SNZT2, SNZT3, and SNZT4). Use the interrupt or control registers (V1, V2) to select an interrupt or a skip instruction. The interrupt request flag is cleared to "0" when an interrupt occurs or when the next instruction is skipped with a skip instruction.

(8) Frequency divider for LCD

The frequency divider for the LCD consists of timer LC and frequency divider (divide by 2). Timer LC is a 4-bit programmable timer with reload latch. Data can be set simultaneously in the reload latch and timer LC with the TLCA instruction.

Timer LC starts counting when data is set in timer LC and the count source is selected with bit 2 of the register W3 and "1" is set to bit 3. When it underflows, data is loaded from the reload latch and count continues. When n is set in timer LC, the count source is divided by n+1 (n=0 to 15).

The timer LC underflow signal divided by 2 becomes the standard clock of the LCD.

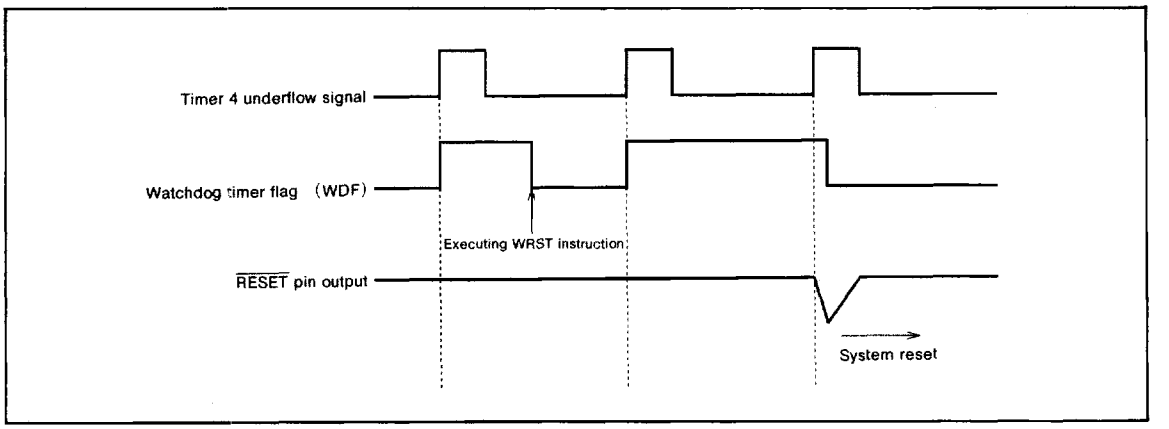


Fig. 29 Watchdog timer function

(9) PWM output

The PWM output function uses timer 2 and timer 4 to form a waveform having arbitrary frequency and duty, and outputs it from the $D_{10}/S_{OUT}/RTP_1$/PWM pin.

Set the "L" period of PWM waveform with timer 2, and the "H" period with timer 4. The function of the $D_{10}/S_{OUT}/RTP_1$/PWM pin can be selected with the serial I/O mode register (J2).

During the "L" period of the PWM waveform, timer 4 is stopped and timer 2 counts the count source. When timer 2 underflows, the PWM waveform changes to "H". During the "H" period, timer 2 is stopped and timer 4 counts the count source. When timer 4 underflows, the PWM waveform changes to "L". This sequence is repeated.

An example of how to use the PWM output is shown below.

① Set bits 0 and 1 of serial I/O mode register (J2) to "1".
② Set the bit 3 of register W4 to "1".
③ Set data in timer 2 and timer 4.
④ Set bits 0 and 1 of register W5 to "1".

Now a PWM waveform is output from $D_{10}/S_{OUT}/RTP_1$/PWM pin. However, do not select timer 2 underflow as the count source for timer 4 when using PWM output function.

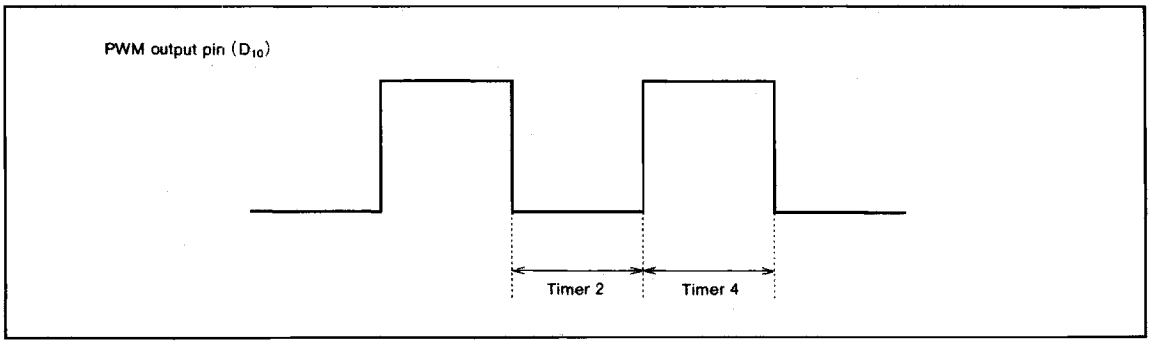


Fig. 30 PWM waveform

(10) Realtime output

The realtime output function uses timer 4 underflow as the trigger to output the contents of the realtime output register (RTP) from the realtime output pin.

Each time a timer 4 underflow occurs, the data set in the realtime output register is transferred to the realtime output latch and at the same time outputs from the realtime output pin. RTP0 and RTP1 can be used for realtime output.

An example of how to use the realtime output is shown below.

Table 8 Real time output pins and real time output registers

Real time output pin	Corresponding bits of real time output register
$D_9/S_{CK}/RTP_0$	Bit 0 (RTP_0)
$D_{10}/S_{OUT}/RTP_1/PWM$	Bit 1 (RTP_1)

① Set the function of the $D_9/S_{CK}/RTP_0$ or $D_{10}/S_{OUT}/RTP_1/PWM$ pin to realtime output with the serial I/O mode registers (J1 and J2).

② Set the initial output value in realtime output latch (RTPL).
(The realtime output latch can be set to "1" with the RTPS instruction or cleared to "0" with the RTPR instruction. However, the timer 4 interrupt request flag (T4F) must be cleared to "0" when executing the RTPS or RTPR instruction.)

③ Set data in the realtime output register (RTP).
(Set the data in the realtime output register through register A with the TRTPA instruction.)

④ Select the timer 4 count source with bits 0 and 1 of register W4.

⑤ Set data in timer 4 and reload register R4, and then start timer 4 count operation with the bit 2 of register W5.

Now the contents of the realtime output register is output from the realtime output pin each time a timer 4 underflow occurs.

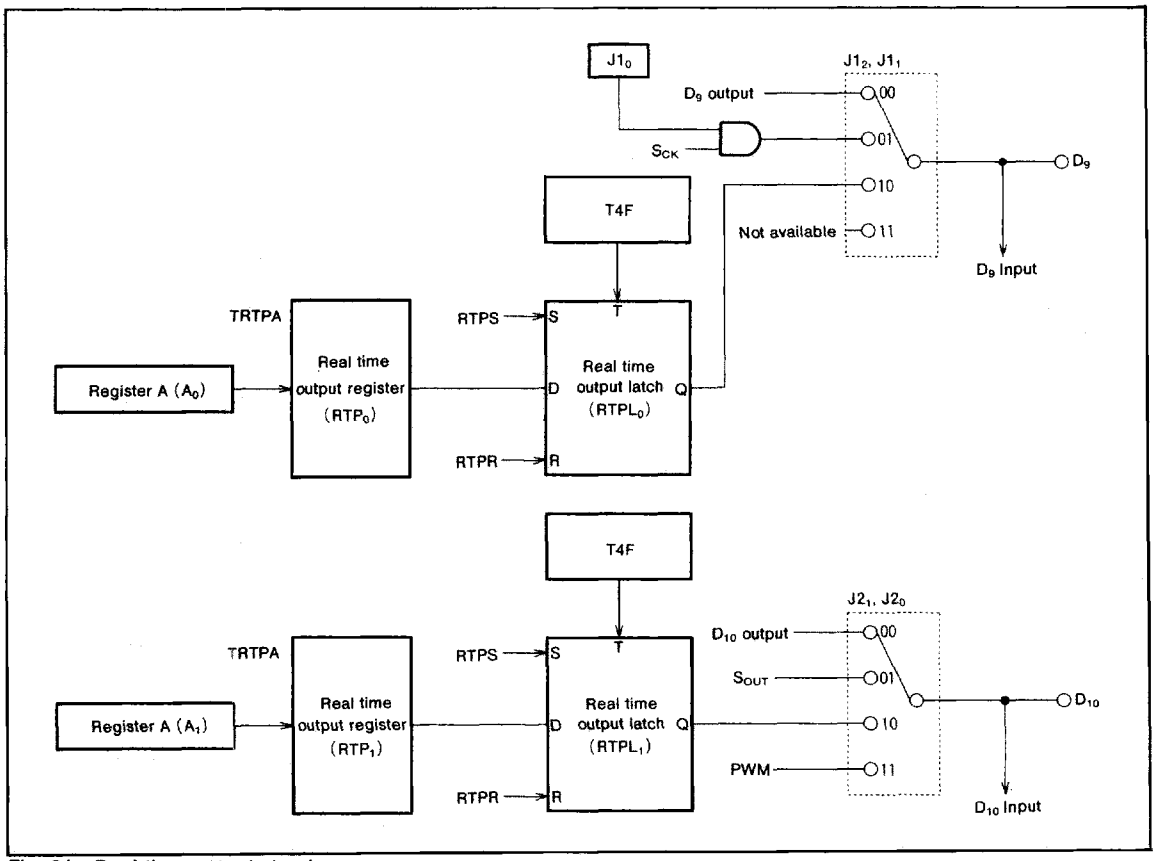


Fig. 31 Real time output structure

Realtime output can be used together with the zero cross detection circuit. An example of how to use the realtime output using the zero cross detection circuit is shown below.

① Repeat steps ① to ③ described above.
② Set timer 4 count source to timer 2 underflow signal with bits 0 and 1 of register W4.
③ Clear the bit 2 of register W5 to "0" and the bit 2 of register W2 to "1".
④ Set bits 0 and 3 of register I1 to "1" and select the valid edge (=zero cross point to be detected) with bits 1 and 2.
⑤ Set data in timer 2 and reload register R2, timer 4 and reload register R4, and then start timer 2 and timer 4 count operations with register W5.
⑥ Set the bit 2 of register W5 to "1".

Now the zero cross of alternating waveform input to the $D_8/INT_0/ZEROX$ pin is detected and timer 2 count operation starts. Timer 4 counts the timer 2 underflow signal and the contents of the realtime output latch are output from the realtime output pin each time timer 4 underflows.

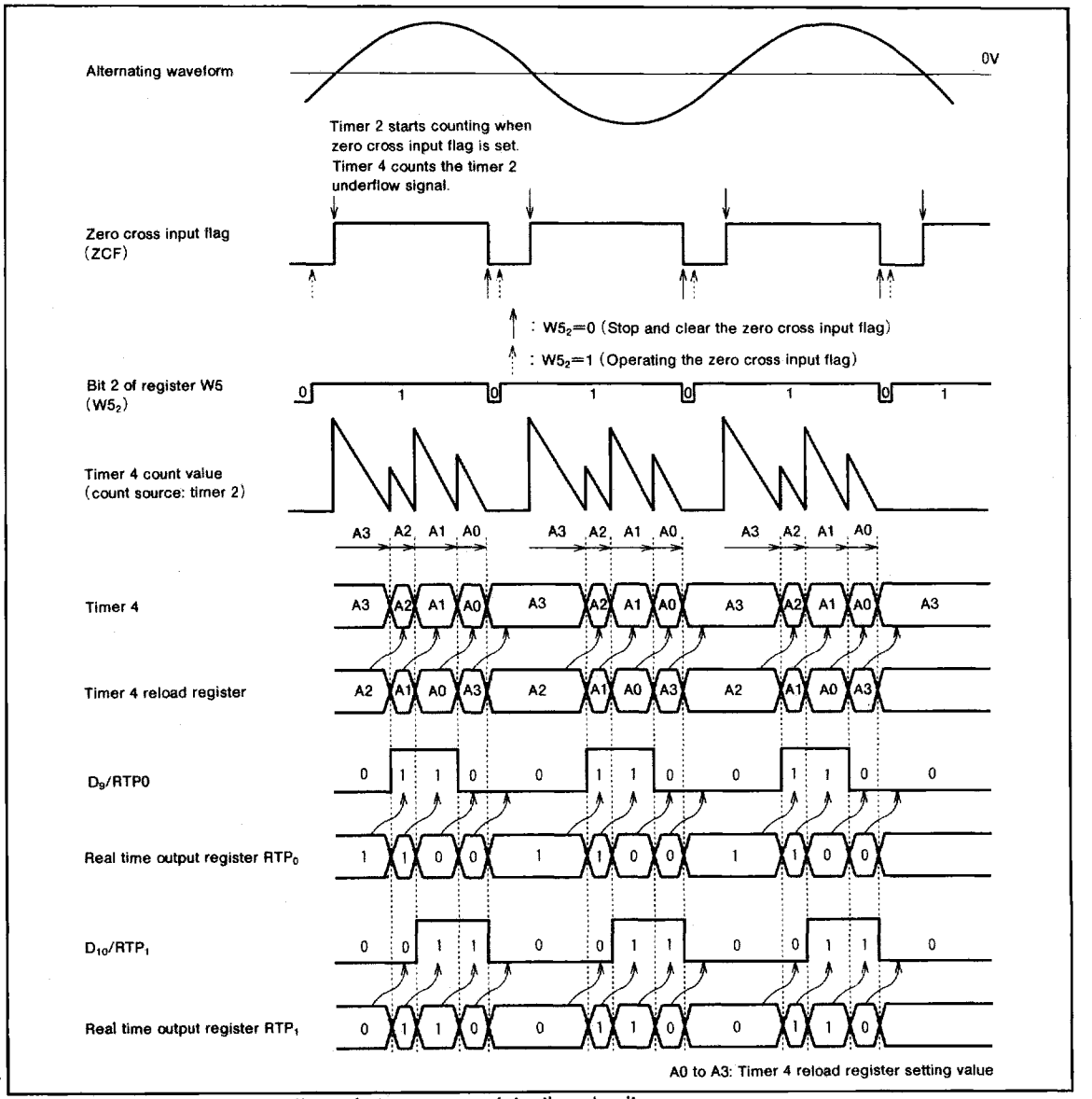


Fig. 32 Real time output operation using zero cross detection circuit

(11) Timer control register

- Timer control register (W1)
  Register W1 controls the $D_6/CNTR_0$ pin function, timer 1 count operation, prescaler dividing ratio and prescaler count operation. Set the contents of this register with the TW1A instruction through register A. The TAW1 instruction can be used to transfer the contents of register W1 to register A.
- Timer control register (W2)
  Register W2 controls the timer 2 count source, count start trigger and $D_7/CNTR_1$ pin function. Set the contents of this register with the TW2A instruction through register A. The TAW2 instruction can be used to transfer the contents of register W2 to register A.
- Timer control register (W3)
  Register W3 controls the count operation and count source for timer 3 and LCD frequency dividing circuit. Set the contents of this register with the TW3A instruction through register A. The TAW3 instruction can be used to transfer the contents of register W3 to register A.
- Timer control register (W4)
  Register W4 controls the timer 4 count source, the operation of watchdog timer and timer 2 and 4 function. Set the contents of this register through register A with the TW4A instruction. The TAW4 instruction can be used to transfer the contents of register W4 to register A.
- Timer control register (W5)
  Register W5 controls the count operation of timer 2 and 4, and the operation of zero cross input flag. Set the contents of this register through register A with the TW5A instruction. The TAW5 instruction can be used to transfer the contents of register W5 to register A. After executing the TAW5 instruction, "0" is stored in bit 3 of register A.
- Serial I/O mode register (J1)
  In addition to control serial I/O, register J1 controls the $D_9/S_{CK}/RTP_0$ pin function. Set the contents of this register through register A with the TJ1A instruction. The TAJ1 instruction can be used to transfer the contents of register J1 to register A. After executing the TAJ1 instruction, "0" is stored in bit 3 of register A.
- Serial I/O mode register (J2)
  Register J2 controls the $D_{10}/S_{OUT}/RTP_1/PWM$ pin function. Set the contents of this register through register A with the TJ2A instruction. The TAJ2 instruction can be used to transfer the contents of register J2 to register A. After executing the TAJ2 instruction, "0" is stored in bits 3 and 2 of register A.

(12) Precautions

Note the following for the use of timers.

- Prescaler precautions
  Stop the prescaler to change its frequency dividing ratio.
- Timer precautions
  Stop timer 1, 2, 3, or 4 counting to change its count source, as well as to execute the TAB1, TAB2, or TAB4 instruction for reading the data (from timer 1, 2, or 4).
  When the timer 4 write instruction (T4ABD) is executed, timer 4 count value before writing is invalid. Do not execute the timer 4 write instruction (T4ABD) just before and just after timer 4 underflow occurs.
  Fully stabilize the $f(X_{CIN})$ oscillation so as to select it as a timer 3 count source.

SERIAL I/O

The 4520 Group has a built-in clock synchronous serial I/O which can serially transmit or receive 8-bit data. Serial I/O consists of :

- Serial I/O register (H, L)
- Serial I/O mode registers (J1 and J2)
- Serial I/O transmission/reception completion flag (SIOF)
- Serial I/O counter

Register A is used to perform data transfer with internal CPU, and the serial I/O pins are used for external data transfer.

The pin functions of the serial I/O pins can be set with the serial I/O mode registers (J1 and J2).

Table 9 Serial I/O pins

Pin	Pin function when selecting serial I/O
$D_9/S_{CK}/RTP_0$	Clock I/O (S_{CK})
$D_{10}/S_{OUT}/RTP_1/PWM$	Serial data output (S_{OUT})
$P2_0/S_{IN}$	Serial data input (S_{IN})

Note : $P2_0/S_{IN}$ is no need to select the function.

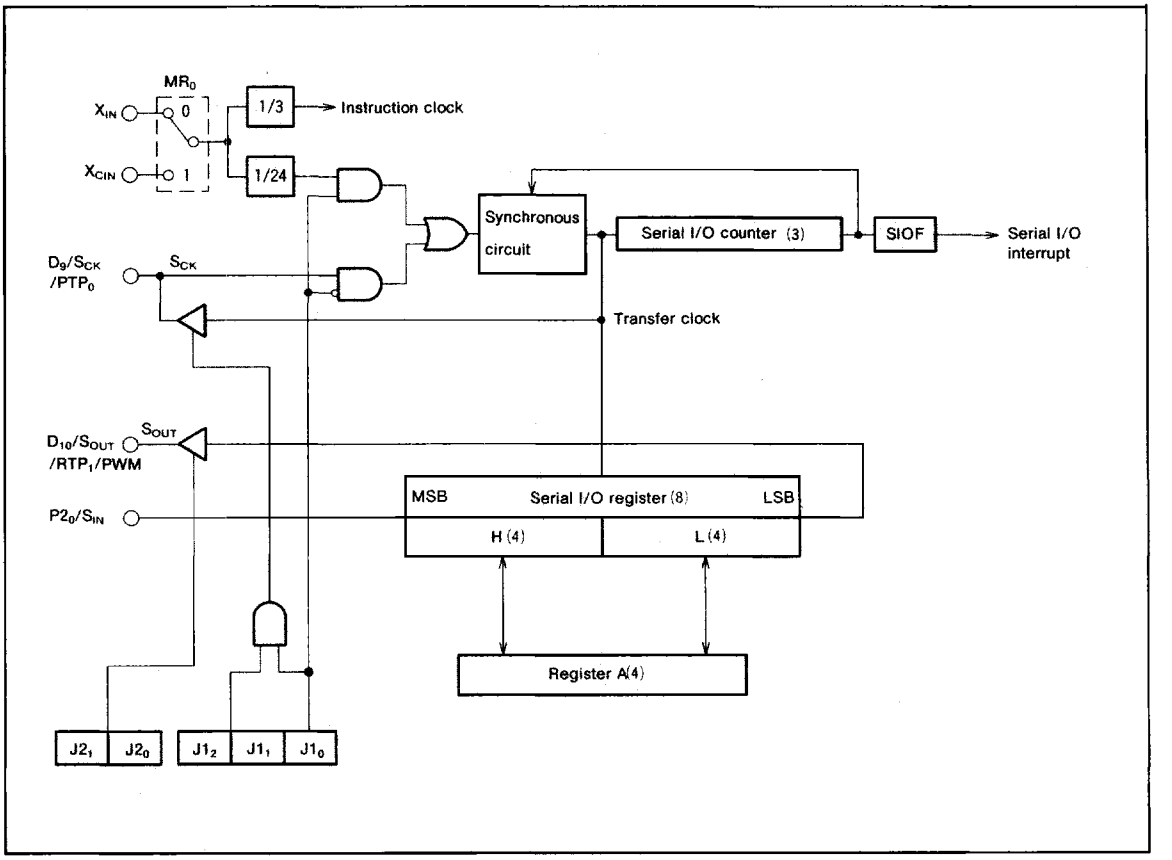
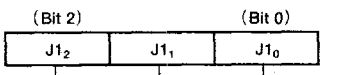


Fig. 33 Serial I/O structure

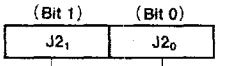


Serial I/O mode register J1 (Note 2)

(at reset : 000_2, at power down : state retained, R/W)

Serial I/O synchronous clock selection bit	
0	External clock
1	$f(X_{IN})$ or $f(X_{CIN})$ divided by 24

$D_9/S_{CK}/RTP_0$ pin function selection bits		
$J1_2$	$J1_1$	Function
0	0	I/O port D_9
0	1	Serial I/O clock input/output S_{CK}
1	0	Real time output RTP_0
1	1	Not available



Serial I/O mode register J2 (Note 3)

(at reset : 00_2, at power down : state retained, R/W)

$D_{10}/S_{OUT}/RTP_1$/PWM pin function selection bits		
$J2_1$	$J2_0$	Function
0	0	I/O port D_{10}
0	1	Serial data output S_{OUT}
1	0	Real time output RTP_1
1	1	PWM output PWM

Notes 1. "R" represents read enabled, and "W" represents write enabled.

2. After executing the TAJ1 instruction, "0" is stored in bit 3 of register A.

3. After executing the TAJ2 instruction, "0" is stored in bits 2 and 3 of register A.

Fig. 34 Serial I/O mode register

(1) Serial I/O register (registers H and L)

The serial I/O register is an 8-bit data transfer serial/parallel conversion shift register which consists of register H and register L. Register H and register L are 4-bit registers. Store the high-order 4 bits of the transmission data in register H and the low-order 4 bits in register L. Data can be set in registers H and L through register A with the THA and TLA instructions, respectively.

During transmission, each bit data is transmitted LSB first from the lowermost bit (bit 0) of register L, and during reception, each bit data is received LSB first to register H starting from the topmost bit (bit 3). When registers H and L are used as work registers without using serial I/O, pull up the S_{CK} pin or set its function to other than S_{CK}.

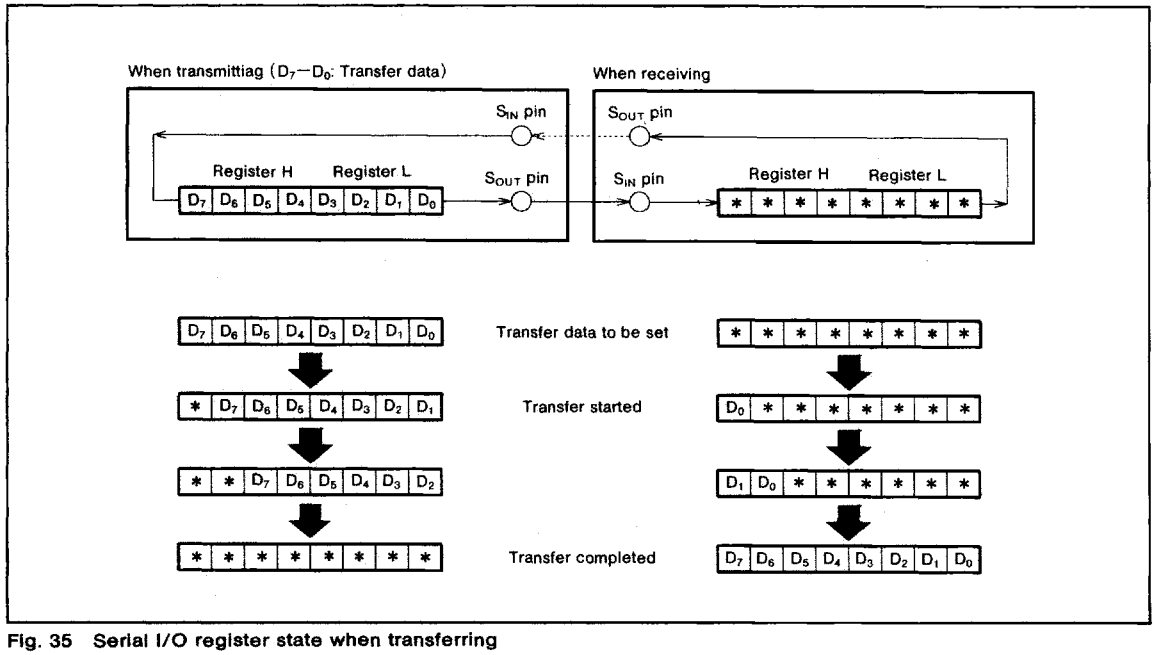


Fig. 35 Serial I/O register state when transferring

(2) Serial I/O transmission/reception completion flag (SIOF)

SIOF flag is set to "1" when serial data transmission or reception completes. The state of this flag can be examined with the skip instruction (SNZSI). Use the interrupt control register (V2) to select the interrupt or the skip instruction.

SIOF flag is cleared to "0" when the interrupt occurs or when the next instruction is skipped with the skip instruction.

(3) Serial I/O start instruction (SST instruction)

When the SST instruction is executed, the SIOF flag is cleared and then serial I/O transmission/reception is started.

(4) Serial I/O mode register

• Serial I/O mode register (J1)

Register J1 controls the synchronous clock and D_9/S_{CK}/RTP_0 pin function. Set the contents of this register through register A with the TJ1A instruction. The TAJ1 instruction can be used to transfer the contents of register J1 to register A. After executing the TAJ1 instruction, "0" is stored in the bit 3 of register A.

• Serial I/O mode register (J2)

Register J2 controls the D_{10}/S_{OUT}/RTP_1/PWM pin function. Set the contents of this register through register A with the TJ2A instruction. The TAJ2 instruction can be used to transfer the contents of register J2 to register A. After executing the TAJ2 instruction, "0" is stored in bits 3 and 2 of register A.

(5) How to use serial I/O

The connection example in Fig. 36 is used to show the data transfer timing and data transfer sequence. Serial I/O interrupt is not used in this example. In the actual wiring, pull up the wiring between each pin with a resistor.

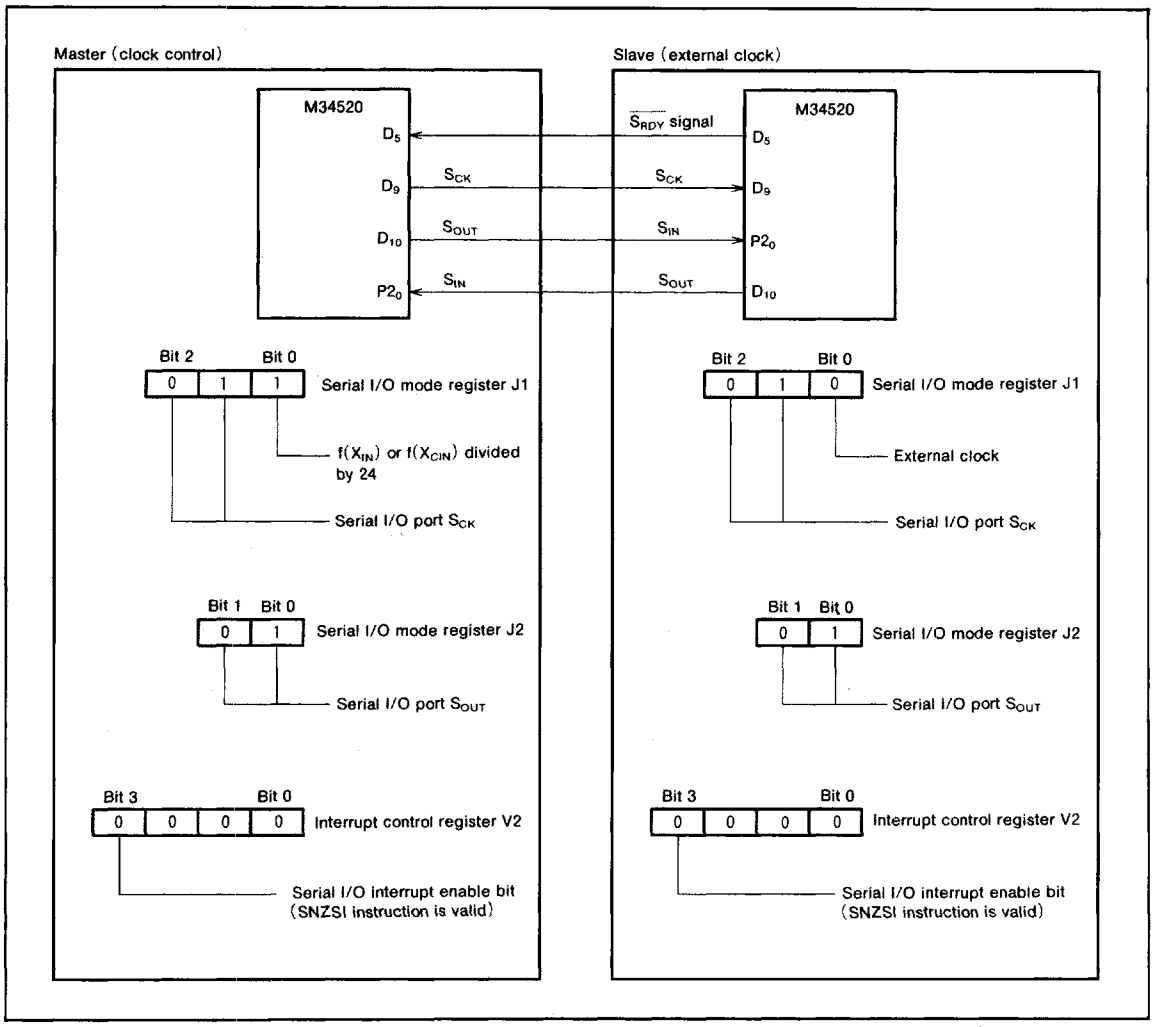


Fig. 36 Serial I/O connection example

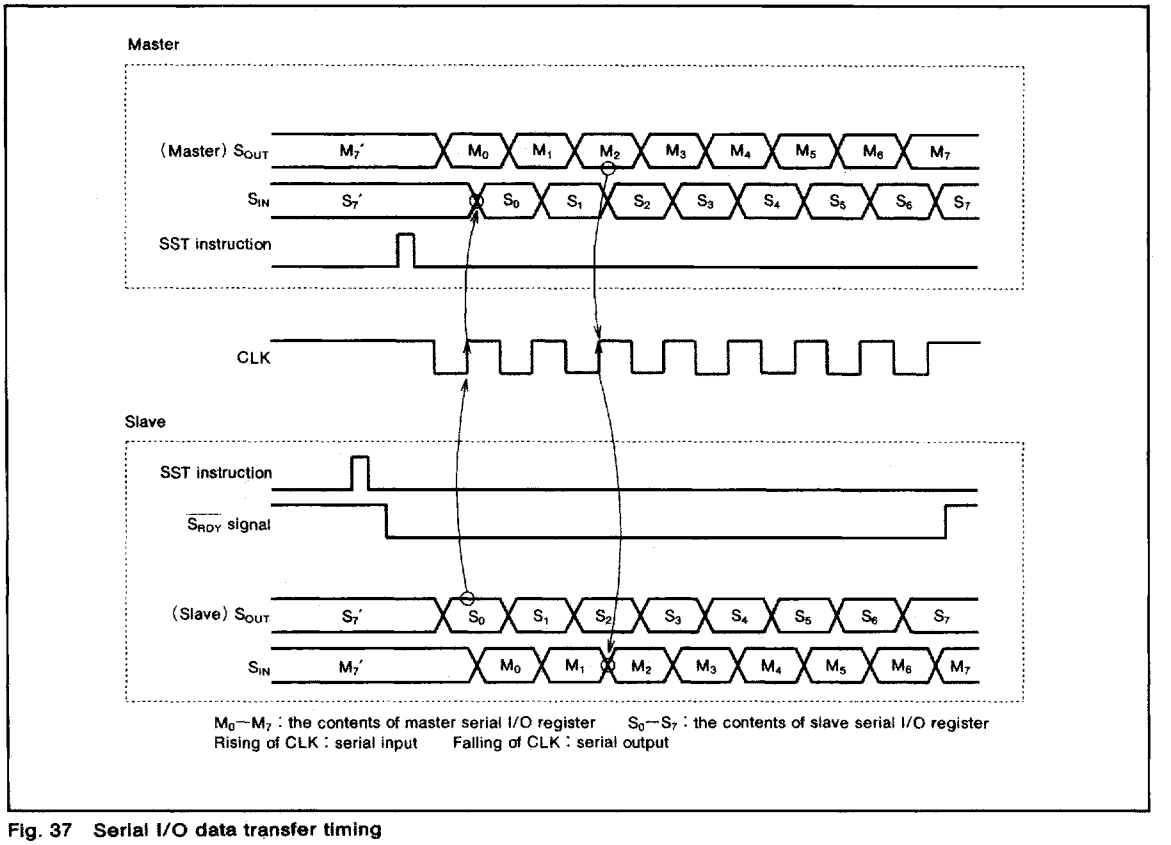


Fig. 37 Serial I/O data transfer timing

Table 10 Processing sequence of data transfer from master to slave

Master (transmission)	Slave (reception)
[Initial setting] • Setting serial I/O mode registers J1 and J2, and interrupt control register V2 shown in Fig. 36.	[Initial setting] • Setting serial I/O mode registers J1 and J2, and interrupt control register V2 shown in Fig. 36.
TJ1A, TJ2A and TV2A instructions	TJ1A, TJ2A and TV2A instructions
• Setting the port received enable signal ($\overline{S_{RDY}}$) to input mode. (Port D_5 is used in this example)	• Setting the port transmitted enable signal ($\overline{S_{RDY}}$) and outputting "H" level. (reception impossible) (Port D_5 is used in this example)
SD instruction	SD instruction
*[Transmission enable state] • Storing transmission data in serial I/O registers H and L.	*[Reception enable state] • SIOF flag is cleared to "0"
THA, TLA instructions	SST instruction
	• "L" level is output from port D_5. (Reception possible)
	RD instruction
[Transmission] • Check port D_5 is "L" level.	[Reception]
SZD instruction	
• Starting the serial transfer	
SST instruction	
• Check transmission completes	• Check reception completes
SNZSI instruction	SNZSI instruction
• Wait (timing when continuously transferring)	• "H" level is output from port D_5.
	SD instruction
	[Data processing]

1-byte data is serially transferred on this process. Subsequently, data can be transferred continuously by repeating the process from *. When an external clock is selected as synchronous clock, control the clock externally because serial transfer is performed as long as clock is externally input (Unlike an internal clock, an external clock is not stopped when serial transfer is completed.) However, the SIOF flag is set to "1" when the clock is counted 8 times after executing the SST instruction. Be sure to set the initial level of the external clock to "H".

LCD FUNCTION

The 4520 Group has a built-in LCD (Liquid Crystal Display) controller/driver. When proper voltage is applied to the LCD power supply input pins, and data is set in timer control registers (W2, W3), timer LC, LCD control registers (L1 to L3), and LCD RAM, the controller/driver automatically reads the display data, controls the LCD display by setting duty and bias.

4 common signal output pins and 27 segment signal output pins can be used to drive the LCD to control the display of up to 108 segments (when 1/4 duty and 1/3 bias are selected). When the required number of segment pins is 27 or less, SEG_{17}–SEG_{26} can be used as I/O ports.

(1) Duty and bias

There are three combinations of duty and bias for displaying data on the LCD. Use bits 0 and 1 of the LCD control register (L1) to select the proper display method for the LCD panel being used.

- 1/2 duty, 1/2 bias
- 1/3 duty, 1/3 bias
- 1/4 duty, 1/3 bias

Table 11 Duty and maximum number of displayed pixels

Duty	Maximum number of displayed pixels	Used COM pins
1/2	54 segments	COM_0, COM_1 (Note)
1/3	81 segments	COM_0–COM_2 (Note)
1/4	108 segments	COM_0–COM_3

Note : Leave unused COM pins open.

(2) LCD clock control

The LCD clock is determined by the setting values of the timer 2 count source selection bits ($W2_0$ and $W2_1$), timer 3 count source selection bit ($W3_0$), LCD frequency dividing circuit count source selection bit ($W3_2$), and timer LC. Accordingly, the frequency (F) of the LCD clock is obtained by the following formula. The number (① to ⑤) under formula corresponds to Fig. 38.

- When using the timer 2 underflow output as timer 3 count source ($W3_0$=0)

$$F=\underbrace{\text{Frequency of timer 2 count source}}_{①}\times\underbrace{\frac{1}{T2+1}}_{②}\times\underbrace{\frac{1}{T3}}_{③}\times\underbrace{\frac{1}{LC+1}}_{④}\times\underbrace{\frac{1}{2}}_{⑤}$$

- When using $f(X_{CIN})$ as timer 3 count source ($W3_0$=1)

$$F=f(X_{CIN})\times\underbrace{\frac{1}{T3}}_{③}\times\underbrace{\frac{1}{LC+1}}_{④}\times\underbrace{\frac{1}{2}}_{⑤}$$

T2 : Timer 2 setting value (0 to 255)
T3 : Timer 3 frequency dividing ratio (16 or 256)
LC : Timer LC setting value (0 to 15)

The frame frequency for each display method can be obtained by the following formula :

$$\text{Frame frequency}=\frac{F}{n}\text{(Hz)}\quad\text{Frame cycle}=\frac{n}{F}\text{(S)}$$

F : LCD clock frequency
1/n : Duty

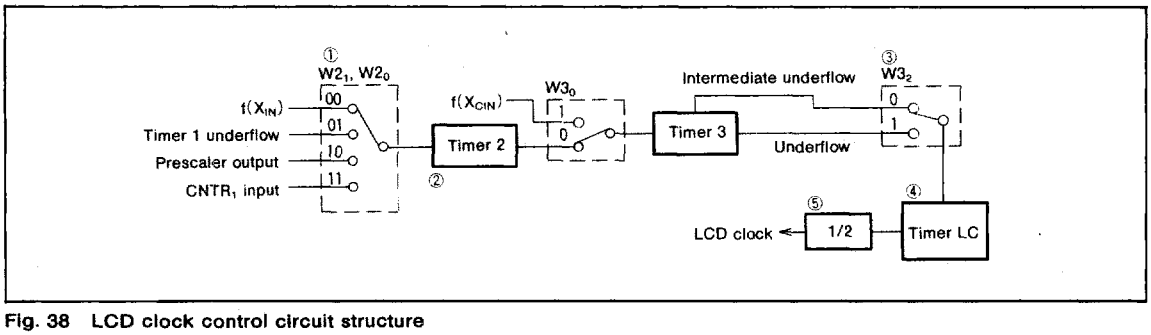


Fig. 38 LCD clock control circuit structure

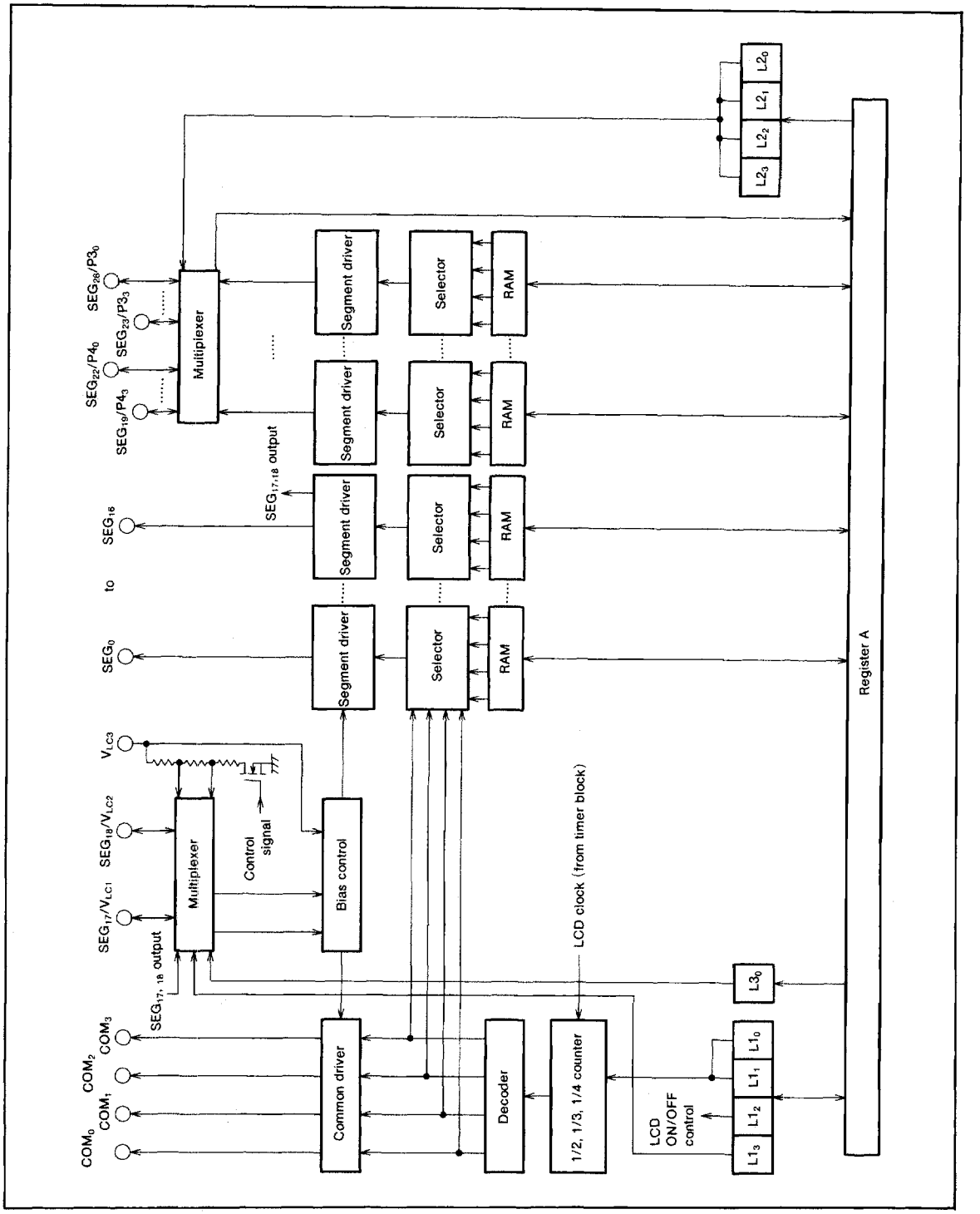


Fig. 39 LCD controller/driver structure

(Bit 3) (Bit 0)

$L1_3$	$L1_2$	$L1_1$	$L1_0$

LCD control register L1

(at reset : 0000_2, at power down : state retained, R/W)

LCD duty and bias selection bits			
$L1_1$	$L1_0$	Duty	Bias
0	0	Not available	
0	1	1/2	1/2
1	0	1/3	1/3
1	1	1/4	1/3

LCD ON/OFF bit	
0	OFF
1	ON

LCD power supply dividing resistor control bit	
0	Connecting internal dividing resistor to LCD power circuit
1	Disconnecting internal dividing resisor from LCD power circuit

(Bit 3) (Bit 0)

$L2_3$	$L2_2$	$L2_1$	$L2_0$

LCD control register L2 (Note 2)

(at reset : 0000_2, at power down : state retained, W)

$SEG_{26}/P3_0/\overline{INT_2}/A_{IN0}$ pin function switch bit	
0	SEG_{26}
1	$P3_0/\overline{INT_2}/A_{IN0}$

$SEG_{25}/P3_1/A_{IN1}$ pin function switch bit	
0	SEG_{25}
1	$P3_1/A_{IN1}$

$SEG_{23}/P3_3/A_{IN3}$, $SEG_{24}/P3_2/A_{IN2}$ pin function switch bit	
0	SEG_{23}, SEG_{24}
1	$P3_3/A_{IN3}$, $P3_2/A_{IN2}$

$SEG_{19}/P4_3/A_{IN7}$—$SEG_{22}/P4_0/A_{IN4}$ pin function switch bit	
0	SEG_{19}—SEG_{22}
1	$P4_3/A_{IN7}$—$P4_0/A_{IN4}$

$L3_0$

LCD control register L3

(at reset : 0_2, at power down : state retained, W)

SEG_{17}/V_{LC1}, SEG_{18}/V_{LC2} pin function switch bit	
0	SEG_{17}, SEG_{18}
1	V_{LC1}, V_{LC2}

Notes 1. "R" represents read enabled, and "W" represents write enabled.

2. After setting the register L2, select A_{IN0}—A_{IN7} with register Q1.

Fig. 40 LCD control registers

(3) LCD RAM

The RAM contains areas corresponding to the liquid crystal display. When "1" is written to this LCD RAM, the display pixel corresponding to the bit is displayed automatically.

Z	1											
X	6				7				8			
Y \ Bit	3	2	1	0	3	2	1	0	3	2	1	0
8	SEG_0	SEG_0	SEG_0	SEG_0	SEG_8	SEG_8	SEG_8	SEG_8	SEG_{16}	SEG_{16}	SEG_{16}	SEG_{16}
9	SEG_1	SEG_1	SEG_1	SEG_1	SEG_9	SEG_9	SEG_9	SEG_9	SEG_{17}	SEG_{17}	SEG_{17}	SEG_{17}
10	SEG_2	SEG_2	SEG_2	SEG_2	SEG_{10}	SEG_{10}	SEG_{10}	SEG_{10}	SEG_{18}	SEG_{18}	SEG_{18}	SEG_{18}
11	SEG_3	SEG_3	SEG_3	SEG_3	SEG_{11}	SEG_{11}	SEG_{11}	SEG_{11}	SEG_{19}	SEG_{19}	SEG_{19}	SEG_{19}
12	SEG_4	SEG_4	SEG_4	SEG_4	SEG_{12}	SEG_{12}	SEG_{12}	SEG_{12}	SEG_{20}	SEG_{20}	SEG_{20}	SEG_{20}
13	SEG_5	SEG_5	SEG_5	SEG_5	SEG_{13}	SEG_{13}	SEG_{13}	SEG_{13}	SEG_{21}	SEG_{21}	SEG_{21}	SEG_{21}
14	SEG_6	SEG_6	SEG_6	SEG_6	SEG_{14}	SEG_{14}	SEG_{14}	SEG_{14}	SEG_{22}	SEG_{22}	SEG_{22}	SEG_{22}
15	SEG_7	SEG_7	SEG_7	SEG_7	SEG_{15}	SEG_{15}	SEG_{15}	SEG_{15}	SEG_{23}	SEG_{23}	SEG_{23}	SEG_{23}
COM	COM_3	COM_2	COM_1	COM_0	COM_3	COM_2	COM_1	COM_0	COM_3	COM_2	COM_1	COM_0

Z	1			
X	9			
Y \ Bit	3	2	1	0
8	SEG_{24}	SEG_{24}	SEG_{24}	SEG_{24}
9	SEG_{25}	SEG_{25}	SEG_{25}	SEG_{25}
10	SEG_{26}	SEG_{26}	SEG_{26}	SEG_{26}
11				
12				
13				
14				
15				
COM	COM_3	COM_2	COM_1	COM_0

Fig. 41 LCD RAM map

(4) LCD control register

- LCD control register (L1)

Register L1 controls the combinations of duty and bias, LCD on/off, and internal dividing resistor connection. Set the contents of this register with the TL1A instruction through register A. The TAL1 instruction can also be used to transfer the contents of register L1 to register A.

- LCD control register (L2)

Register L2 controls pins SEG_{19}—SEG_{26} function. After set this register, select analog input A_{IN} by register Q1. Set the contents of this register with the TL2A instruction through register A.

- LCD control register (L3)

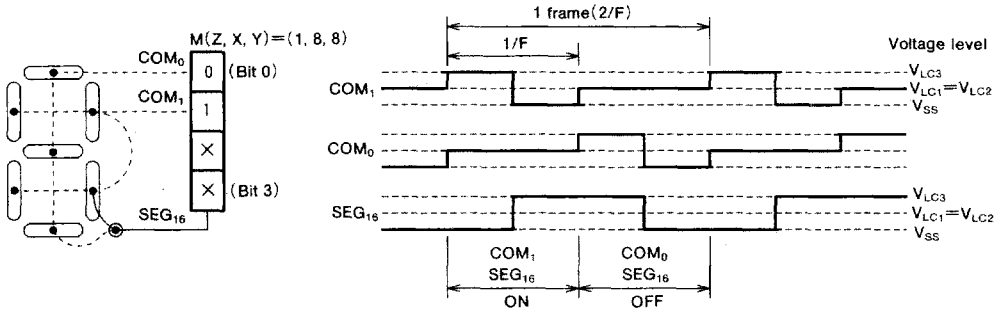
Register L3 controls pins SEG_{17}/V_{LC1} and D_{18}/V_{LC2} function. Set the contents of this register with the TL3A instruction through register A.

(5) LCD drive waveform

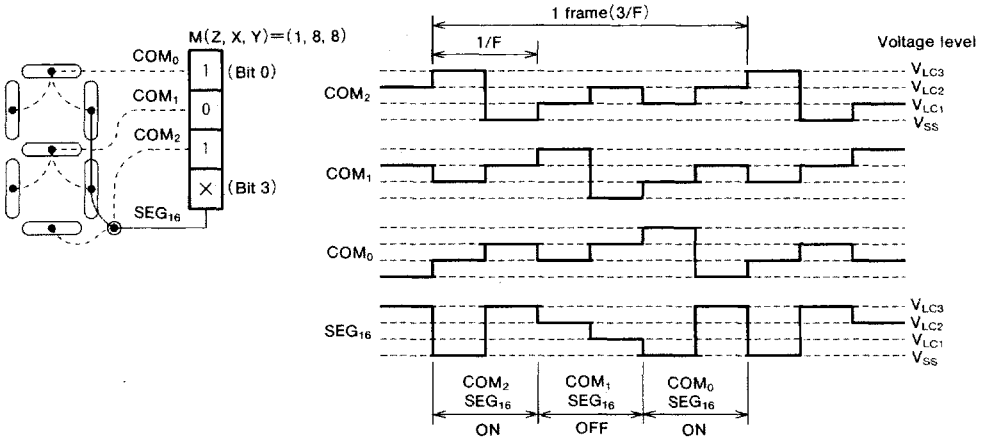
Fig. 42 shows the drive waveform example for each display method. When "1" is written in the LCD RAM data, the voltage difference between the corresponding common pin and segment pin becomes $|V_{LC3}|$ and the display pixel at the cross section turns on.

When returning from reset and being the power down 2 state, display pixel turns off because every segment output pin and common output pin becomes V_{LC3} level.

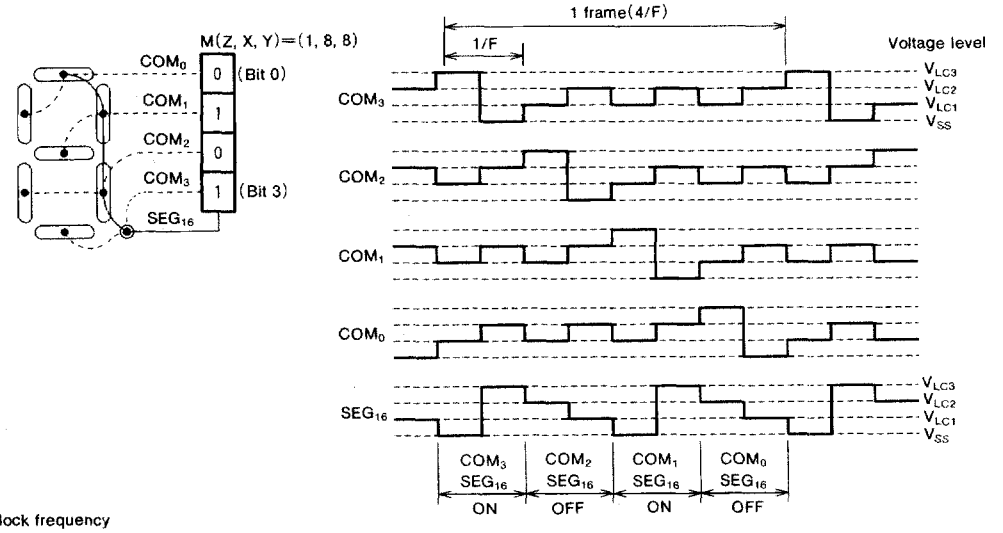
1/2 duty, 1/2 bias : When writing $(XX10)_2$ to address M(Z, X, Y)=(1, 8, 8) in RAM



1/3 duty, 1/3 bias : When writing $(X101)_2$ to address M(Z, X, Y)=(1, 8, 8) in RAM



1/4 duty, 1/3 bias : When writing $(1010)_2$ to address M(Z, X, Y)=(1, 8, 8) in RAM



F : LCD clock frequency

X : Set an arbitrary value. (These bits are not related to drive waveform setting at each duty.)

Fig. 42 Drive wave example

MITSUBISHI ELECTRIC

(6) LCD power supply

The 4520 Group has the built-in LCD dividing resistor that can be disconnected with software. Select whether to connect this internal dividing resistor or not, and select the LCD power supply circuit appropriate for the LCD panel being used according to the combination of 3 items in the following Table 12. LCD power supply control.

Table 12 LCD power supply control

Control Item	Control Bit	
Connect/disconnect internal dividing resistor to/from LCD power supply.	$L1_3$	
	Connecting	0
	Disconnecting	1
Connect/disconnect pins SEG_{17}/V_{LC1} and SEG_{18}/V_{LC2} to/from LCD power supply.	$L3_0$	
	Disconnecting	0
	Connecting	1
Use 1/2 or 1/3 bias.	$L1_1$	
	1/2 bias	0
	1/3 bias	1

- When connecting the internal dividing resistor and disconnecting pins SEG_{17}/V_{LC1} and SEG_{18}/V_{LC2}
  $[L1_3=0, L3_0=0]$
  In this case, 0 to V_{LC3} (V) voltage is applied to the LCD panel. Apply voltage between 2.2V and V_{DD} to the V_{LC3} pin. (circuit example a)
- When connecting the internal dividing resistor and connecting pins SEG_{17}/V_{LC1} and SEG_{18}/V_{LC2}
  $[L1_3=0, L3_0=1]$
  In this case, internally generated divided voltage is output from pins SEG_{17}/V_{LC1} and SEG_{18}/V_{LC2}. Accordingly, the impedance of the LCD power can be reduced by externally connecting a capacitor to the pins SEG_{17}/V_{LC1} and SEG_{18}/V_{LC2}. Apply voltage between 2.2V and V_{DD} to the V_{LC3} pin.
  (1/3 bias : circuit example b, 1/2 bias : circuit example c)
- When disconnecting the internal dividing resistor and connecting pins SEG_{17}/V_{LC1} and SEG_{18}/V_{LC2}
  $[L1_3=1, L3_0=1]$
  This is the external power input mode. Apply the following voltage to each LCD power supply input pins.
  When using 1/3 bias : $(2.2V \leq V_{LC3} \leq V_{DD})$

  $V_{LC2}=\frac{2}{3}V_{LC3}$, $V_{LC1}=\frac{1}{3}V_{LC3}$

  When using 1/2 bias : $(2.2V \leq V_{LC3} \leq V_{DD})$

  $V_{LC2}=V_{LC1}=\frac{1}{2}V_{LC3}$

  (1/3 bias : circuit example d, 1/2 bias : circuit example e)

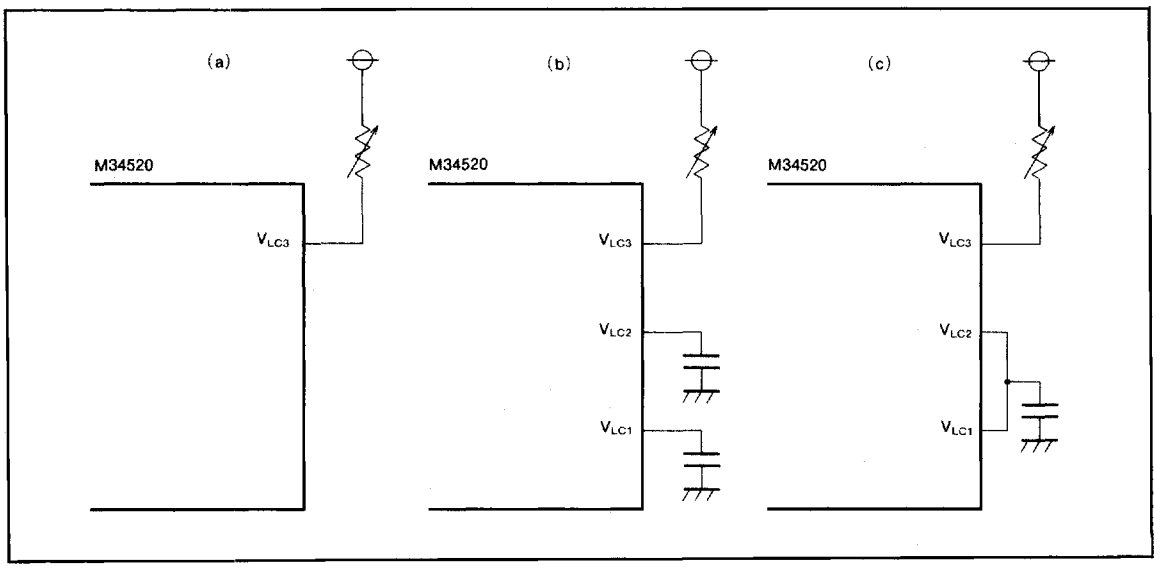


Fig. 43 LCD power circuit example

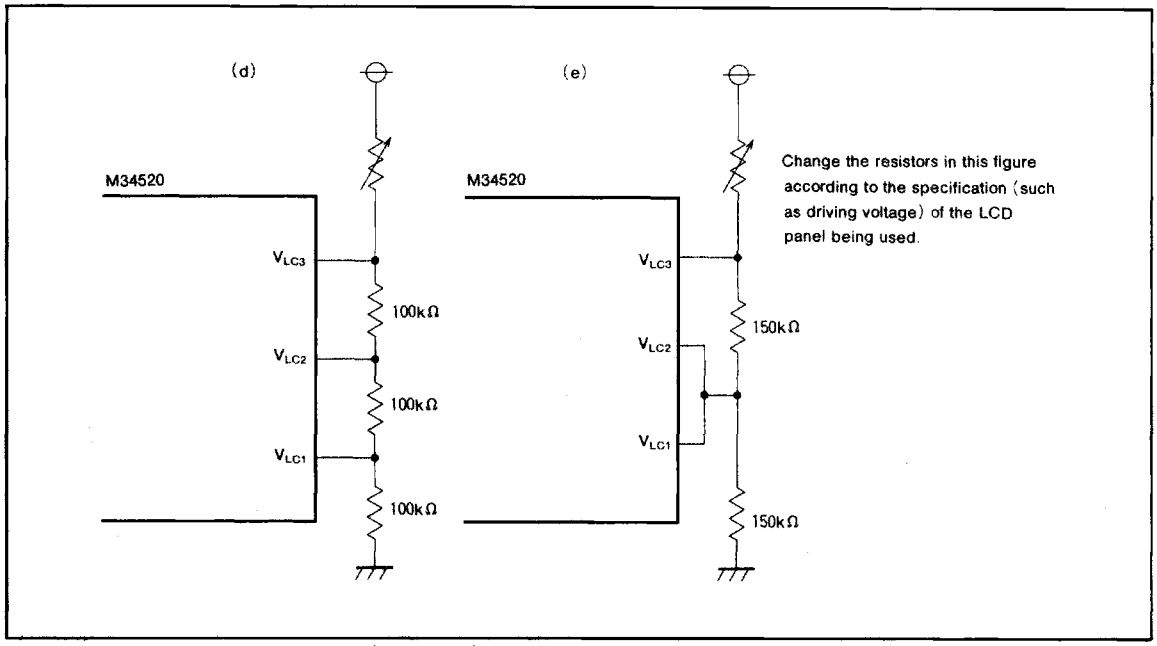


Fig. 44 LCD power circuit example (continued)

(7) LCD display method

The connection example in Fig. 45 shows the display method, how to set the LCD control register, and the drive waveform when displaying the character "9".

1. Select the duty and bias combinations with bits 0 and 1 of the register L1.
2. Set the internal resistor with bit 3 of the register L1, and register L3.
3. Switch pins $SEG_{26}/P3_0/INT_2/A_{IN0}$ and $SEG_{25}/P3_1/A_{IN1}$ to segment output ports with bits 0 and 1 of the register L2.
4. Write $(1011)_2$ and $(0111)_2$ to addresses M (Z, X, Y) = (1, 9, 9) and M (Z, X, Y) = (1, 9, 10) in RAM as shown in Fig. 47.
5. Character "9" is displayed by setting bit 2 of the register L1 to "1".

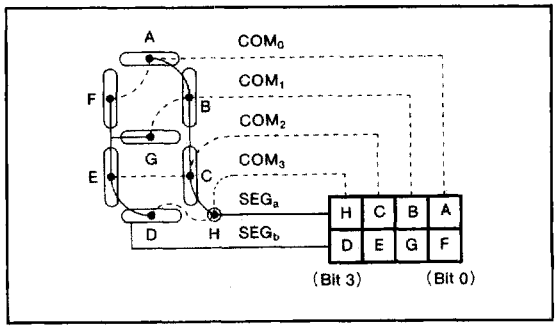


Fig. 45 LCD connection example

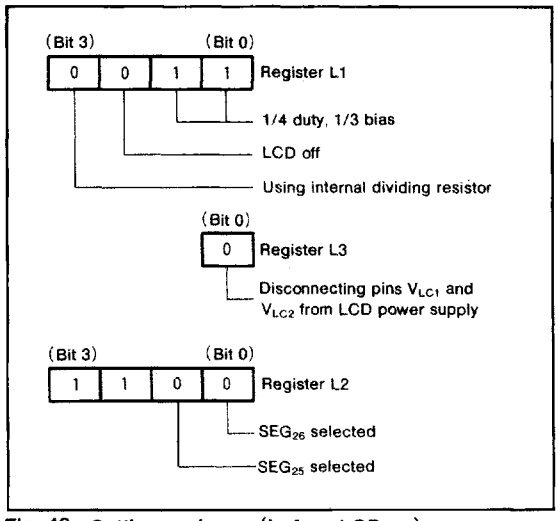


Fig. 46 Setting registers (before LCD on)

MITSUBISHI ELECTRIC

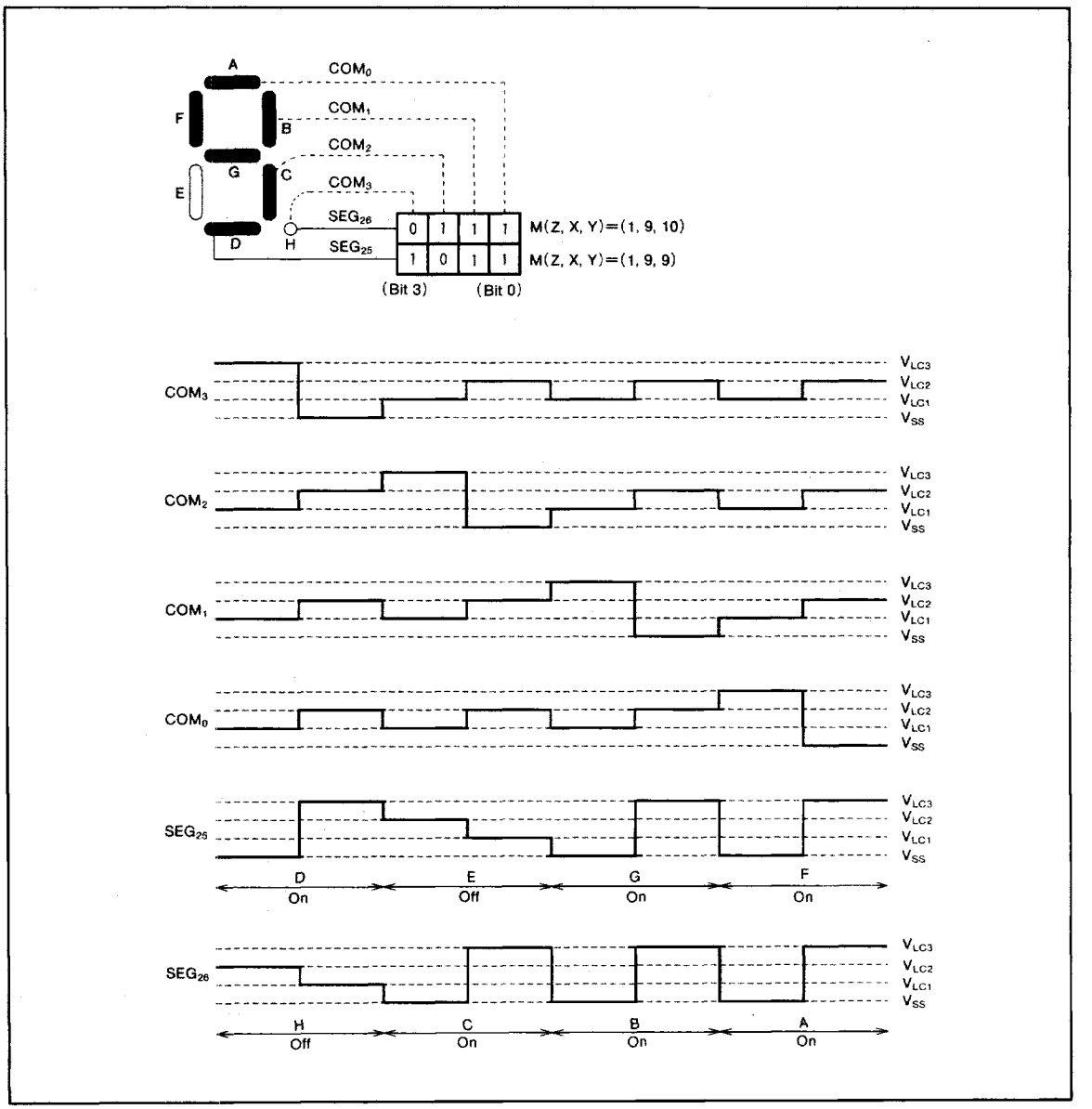


Fig. 47 Display pattern example and drive waveform example

A-D CONVERSION CIRCUIT

The 4520 Group has a built-in A-D conversion circuit that performs conversion by 8-bit successive comparison method. Table 13 shows the characteristics of this A-D conversion circuit.

Table 13 A-D conversion circuit function

Parameter	Characteristics
Conversion format	Successive comparison method
Resolution	8 bits
Absolute accuracy	±2LSB
Conversion speed	25.5μs (at 4MHz system clock frequency)
Analog input pin	8 (selecting from A_{IN0}—A_{IN7})

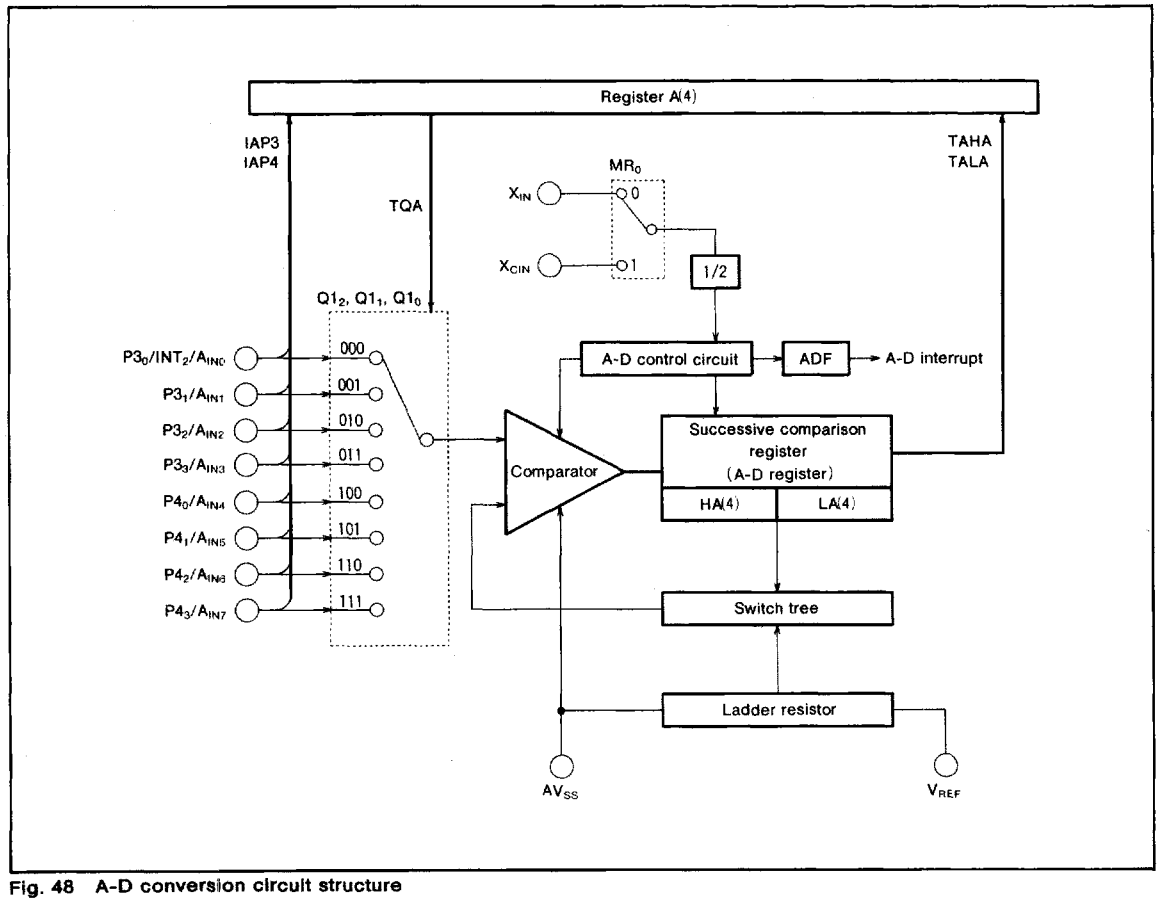
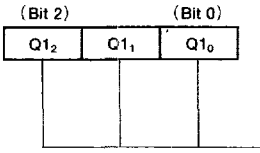


Fig. 48 A-D conversion circuit structure



A-D control register Q1

(at reset : 000_2, at power down : state retained, R/W)

Analog input pin selection bits			
$Q1_2$	$Q1_1$	$Q1_0$	Selective pin
0	0	0	A_{IN0}
0	0	1	A_{IN1}
0	1	0	A_{IN2}
0	1	1	A_{IN3}
1	0	0	A_{IN4}
1	0	1	A_{IN5}
1	1	0	A_{IN6}
1	1	1	A_{IN7}

Notes 1. "R" represents read enabled, and "W" represents write enabled.
2. After executing the TAQ1 instruction, "0" is stored in the bit 3 of register A.

Fig. 49 A-D control register

(1) Successive comparison register (HA, LA)
Successive comparison register consists of register HA and register LA. Register HA and register LA are 4-bit registers. The high-order 4 bits of the 8-bit digital data which is the A-D conversion result of analog input is stored in register HA, and the low-order 4 bits are stored in register LA. The contents of these registers can be transferred to register A with the TAHA instruction and TALA instruction, respectively. However, do not execute these instructions during A-D conversion.
When the contents of successive approximation register is n, the logic value of the reference voltage V_{REF} and comparison voltage V_{ref} can be obtained by the following formula.

Logic value of comparison voltage V_{ref}

- When n=0, $V_{ref}=0$
- When n=1 to 255

$$V_{ref}=\frac{V_{REF}}{256}\times(n-0.5)$$

n : The value of A-D register
(Decimal expression)

(2) A-D conversion completion flag (ADF)
The ADF flag is set to "1" when A-D conversion completes. The state of this flag can be examined with the skip instruction (SNZAD). Use the interrupt control register (V2) to select the interrupt or the skip instruction.
The ADF flag is cleared to "0" when the interrupt occurs or when the next instruction is skipped with the skip instruction.

(3) A-D conversion start instruction (ADST instruction)
A-D conversion starts when the ADST instruction is executed. However, be sure to execute the SNZAD instruction and clear the ADF flag to "0", and then execute A-D conversion with the ADST instruction. The conversion result is automatically stored in the successive approximation register.

(4) A-D control register (Q1)
Register Q1 is used to select one of the 8 analog input pins. After set the pin function with the LCD control register (L2), select the analog input with this register.

(5) Operating description

A-D conversion is started with the A-D conversion start instruction (ADST). The internal operation during A-D conversion is as follows:

① When A-D conversion starts, the successive comparison register is cleared to "00_{16}".

② Next, the topmost bit of the successive comparison register is set to "1", and the comparision voltage V_{ref} is compared with the analog input voltage V_{IN}.

③ When the comparison result is $V_{ref} < V_{IN}$, the topmost bit of the successive comparison register remains set to "1". When $V_{ref} > V_{IN}$, it is cleared to "0".

The M34520 repeats this operation to the lowermost bit of the successive comparison register to convert an analog value to a digital value. A-D conversion stops after 102 clock cycles (25.5μs when $f(X_{IN}) = 4MHz$) from the start and the conversion result is stored in the successive comparison register. An A-D interrupt activated condition is satisfied and the A-D interrupt request flag (ADF) is set to "1" as soon as A-D conversion completes.

Table 14 Change of successive comparison register during A-D conversion

	Change of successive comparison register	Comparison voltage (V_{ref}) value
At starting conversion	0 0 0 0 0 0 0 0	0
First comparison	1 0 0 0 0 0 0 0	$\frac{V_{REF}}{2} - \frac{V_{REF}}{512}$
Second comparison	*1 1 0 0 0 0 0 0	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} - \frac{V_{REF}}{512}$
Third comparison	*1 *2 1 0 0 0 0 0	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \frac{V_{REF}}{8} - \frac{V_{REF}}{512}$
After eighth comparison completes	A-D conversion result: *1 *2 *3 *4 *5 *6 *7 *8	

*1 : First comparison result
*2 : Second comparison result
*3 : Third comparison result
*4 : Fourth comparison result
*5 : Fifth comparison result
*6 : Sixth comparison result
*7 : Seventh comparison result
*8 : Eighth comparison result

(6) A-D converter equivalent connection diagram
Fig. 50 shows the A-D converter equivalent connection diagram. Fig. 51 shows the A-D conversion timing diagram.

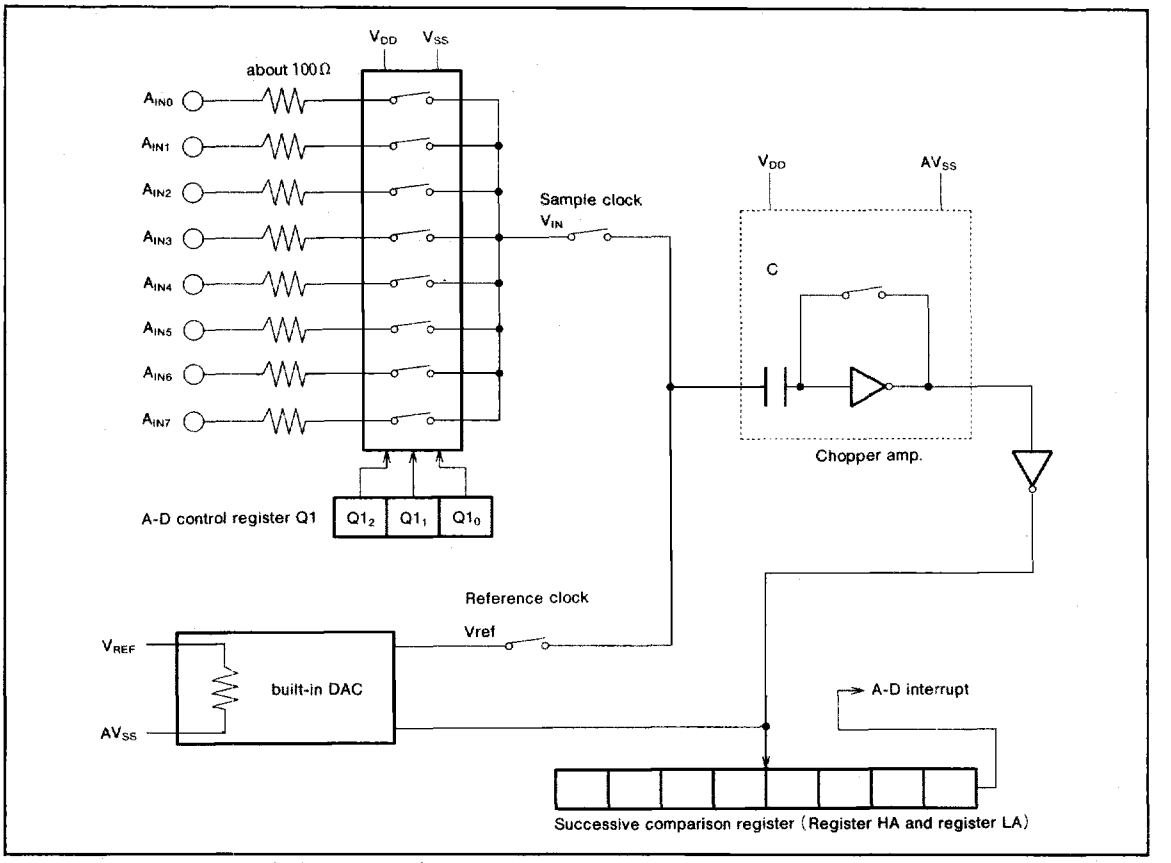


Fig. 50 A-D converter equivalent connection

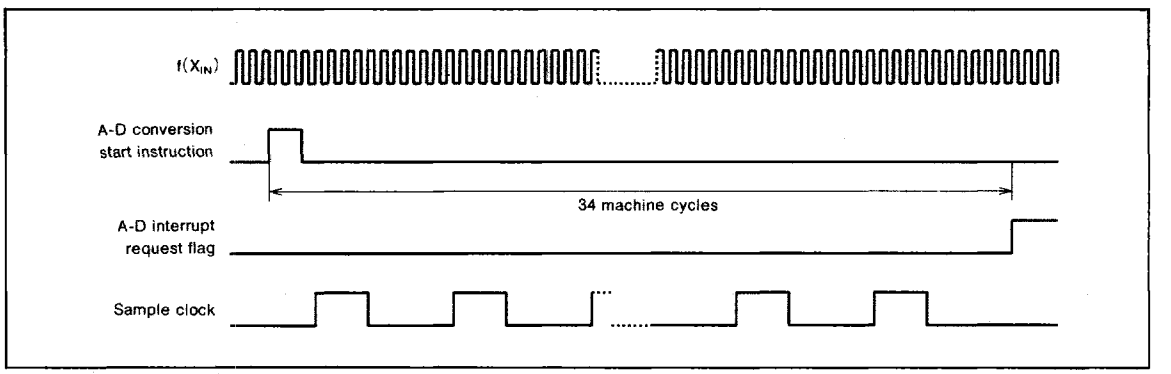


Fig. 51 A-D conversion timing chart

(7) How to use A-D conversion

How to use A-D conversion is explained using as example in which the analog input from the $SEG_{26}/P3_0/INT_2/A_{IN0}$ pin is A-D converted and the high-order 4 bits of the converted data are stored in address M (0, 0, 0) in RAM and the low-order 4-bits are stored in address M (0, 0, 1). The A-D interrupt is not used in this example.

① Select the $P3_0/INT_2/A_{IN0}$ function with the bit 0 of LCD control register (L2) and then select the A_{IN0} pin with the A-D control register (Q1) (refer to Fig. 52).

② Execute the SNZAD instruction and clear the ADF flag to "0".

③ Execute the ADST instruction and start A-D conversion.

④ Examine the state of the ADF flag with the SNZAD instruction to determine the end of A-D conversion.

⑤ Transfer the high-order 4 bits of the converted data to M (0, 0, 0) from register HA through register A (TAHA instruction).

⑥ Similarly, transfer the low-order 4 bits of the converted data to M (0, 0, 1) from register LA through register A (TALA instruction).

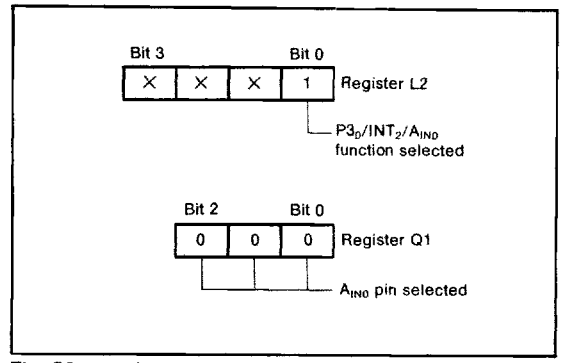


Fig. 52 Register setting example

RESET FUNCTION

System reset is performed by applying "L" level to the $\overline{\text{RESET}}$ pin for 1 machine cycle or more when the following condition is satisfied;

• the value of supply voltage is the minimum value or more of the recommended operating conditions

Then when "H" level is applied to the $\overline{\text{RESET}}$ pin, the program starts from address 0 in page 0.

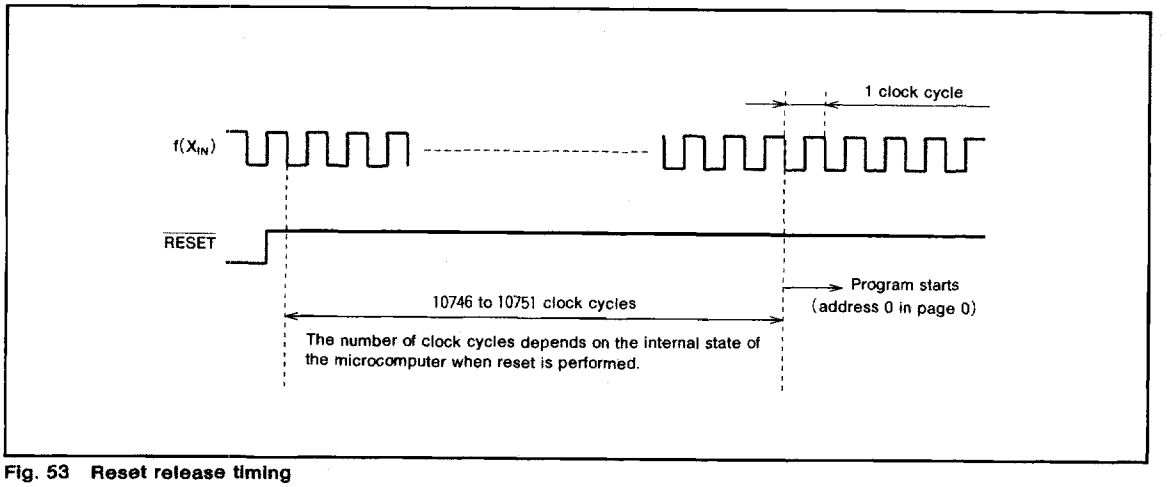


Fig. 53 Reset release timing

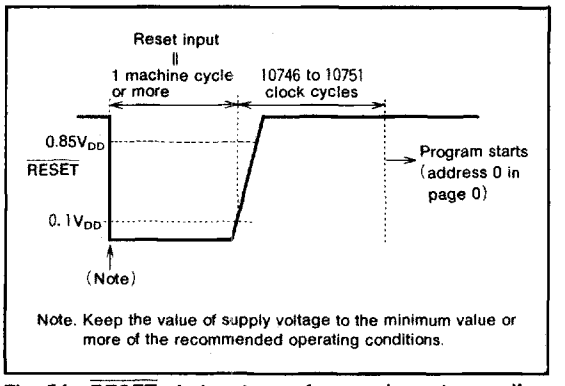


Fig. 54 $\overline{\text{RESET}}$ pin input waveform and reset operation

(1) Power-on reset

Reset can be performed automatically at power on (power-on reset) by connecting resistors, a diode, and a capacitor to the $\overline{\text{RESET}}$ pin. Connect the $\overline{\text{RESET}}$ pin and the external circuit at the shortest distance.

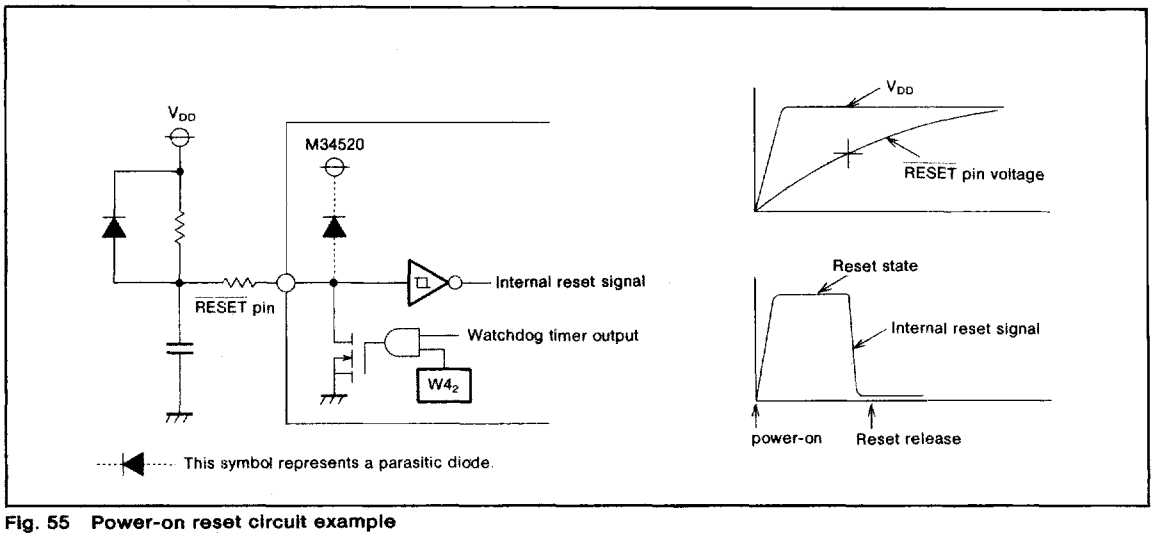


Fig. 55 Power-on reset circuit example

(2) Internal state at reset

Table 15 shows port state at reset, and Fig. 56 and Fig. 57 show internal state at reset (they are the same after reset is released).

The contents of timers, registers, flags and RAM except shown in Fig. 56 and 57 are undefined, so set the initial value to them.

Table 15 Port state at reset state

Name	Function at reset	State at reset
D_0—D_5	D_0—D_5	High impedance (Note)
D_6/$CNTR_0$, D_7/$CNTR_1$, D_8/INT_0/ZEROX, D_9/S_{CK}/RTP_0, D_{10}/S_{OUT}/RTP_1/PWM	D_6—D_{10}	
$P0_0$—$P0_3$	$P0_0$—$P0_3$	
$P1_0$—$P1_3$	$P1_0$—$P1_3$	
$P2_0$/S_{IN}, $P2_1$/INT_1	$P2_0$, $P2_1$	High impedance
$P2_2$, $P2_3$	$P2_2$, $P2_3$	
SEG_0—SEG_{16}	SEG_0—SEG_{16}	V_{LC3} Level
SEG_{17}/V_{LC1}, SEG_{18}/V_{LC2}	SEG_{17}, SEG_{18}	
SEG_{19}/$P4_3$/A_{IN7}— SEG_{22}/$P4_0$/A_{IN4}	SEG_{19}—SEG_{22}	
SEG_{23}/$P3_3$/A_{IN3}— SEG_{25}/$P3_1$/A_{IN1}	SEG_{23}—SEG_{25}	
SEG_{26}/$P3_0$/INT_2/A_{IN0}	SEG_{26}	
COM_0—COM_3	COM_0—COM_3	

Note : Output latch is set to "1".

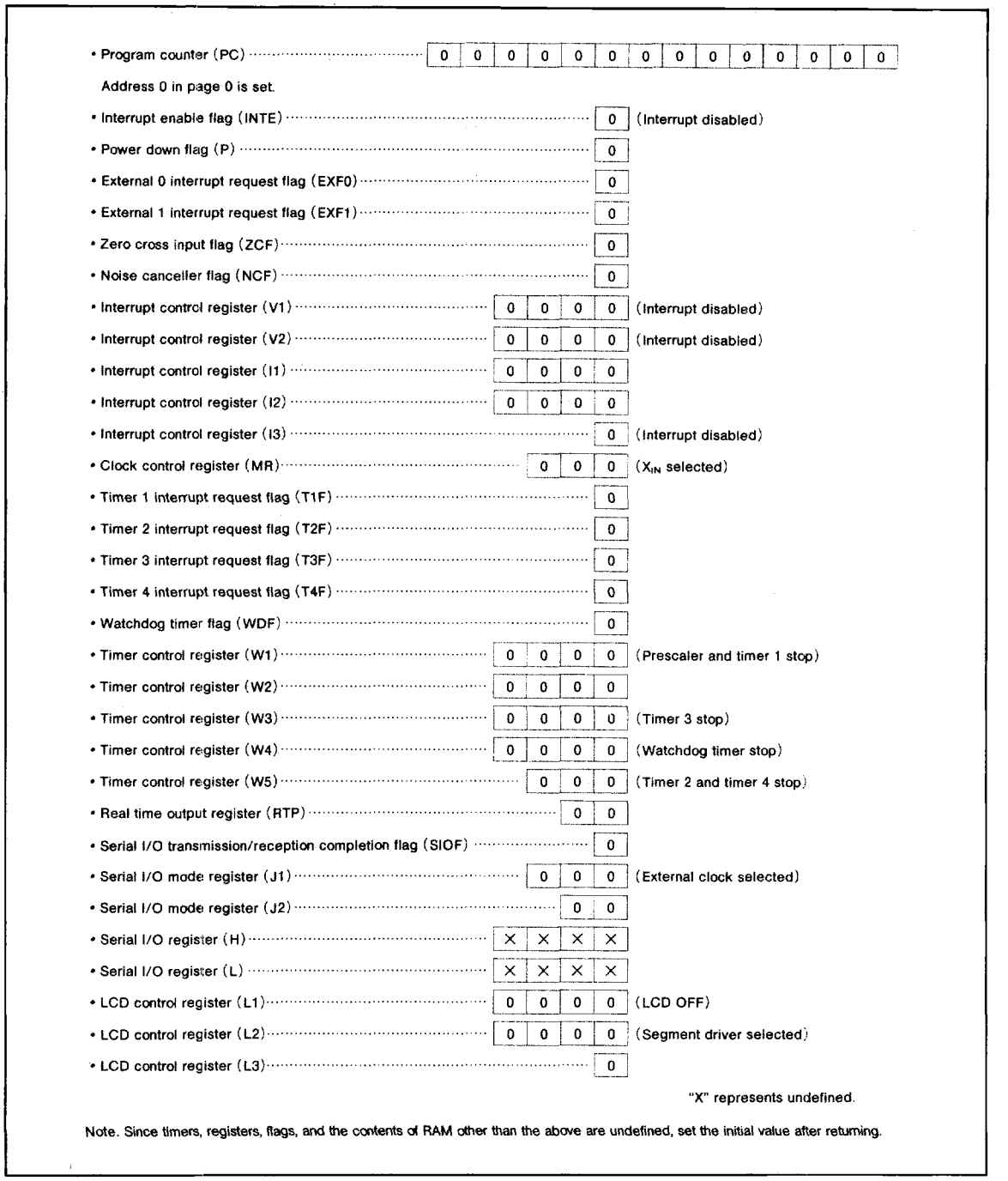


Fig. 56 Internal state at reset

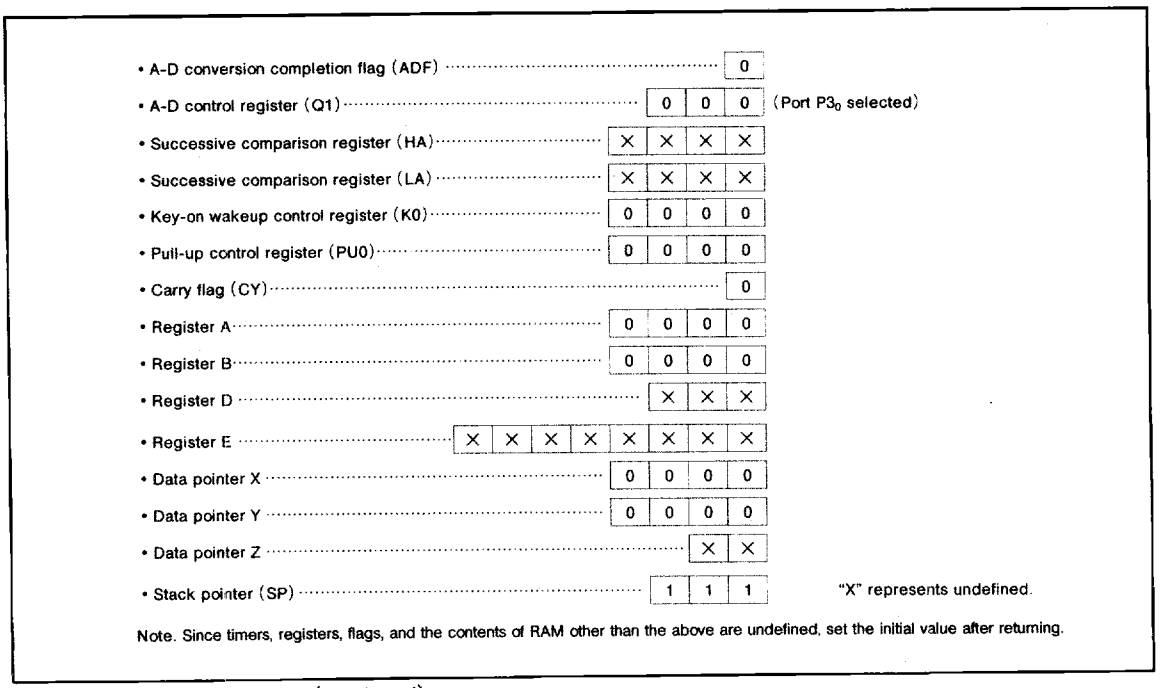


Fig. 57 Internal state at reset (continued)

POWER DOWN FUNCTION

The 4520 Group has two power down functions.

- Power down 1 (clock operating mode) ..... POF instruction
- Power down 2 (RAM back-up mode) ..... POF2 instruction

Note : Be sure to disable interrupts by executing the DI instruction before the POF (power down 1) or POF2 (power down 2) instruction is executed.

Power down is performed by executing each instruction. The start condition is different between these power downs and normal reset.

- Return from power-down state ..... Warm start condition
- Return from reset state ..... Cold start condition

(1) Power down 1 (clock operating mode)
The following functions and states are retained at a power down with the POF instruction.
- RAM
- Reset circuit
- X_{CIN}-X_{COUT} oscillation
- LCD display
- Timer 3, timer 4

(2) Power down 2 (RAM back-up mode)
The following functions and states are retained at a power down with the POF2 instruction.
- RAM
- Reset circuit

Unlike power down 1, all oscillations stop with power down 2.

(3) Warm start condition
The system returns from the power-down state when :
- external wakeup signal is input, or
- timer 4 interrupt request flag is set in power down 1 state, or when :
- external wakeup signal is input in power down 2 state. In either case, the CPU starts executing the program from address 0 in page 0 after returning. In this case, the P flag is set to "1".

(4) Cold start condition
The CPU starts executing the program from address 0 in page 0 when :
- reset pulse is input, or
- reset by watchdog timer.

In this case, the P flag is cleared to "0".

Table 16 Functions and states retained at power down

Function	Power down	
	Mode 1	Mode 2
Program counter (PC) Registers A, B Carry flag (CY) Stack pointer (SP)(Note 2)	×	×
Contents of RAM	○	○
Port level	○	○
Clock control register (MR)	○	○
Timer control register (W1)	×	×
Timer control registers (W2 to W5)	○	○
Interrupt control registers (V1, V2)	×	×
Interrupt control registers (I1 to I3)	○	○
LCD display function	○	(Note 3)
LCD control registers (L1 to L3)	○	○
Timer LC	○	(Note 4)
Timer 1 function	×	×
Timer 2 function	(Note 4)	(Note 4)
Timer 3 function	○	(Note 4)
Timer 4 function	○	(Note 4)
A-D function	×	×
Serial I/O function	×	×
Serial I/O mode registers (J1, J2)	○	○
A-D control register (Q1)	○	○
Pull-up control register (PU0)	○	○
Key-on wakeup control register (K0)	○	○
Real time output register (RTP)	○	○
External 0 interrupt request flag (EXF0)	×	×
External 1 interrupt request flag (EXF1)	×	×
Noise canceller flag (NCF)	×	×
Zero cross input flag (ZCF)	×	×
Timer 1 interrupt request flag (T1F)	×	×
Timer 2 interrupt request flag (T2F)	(Note 4)	(Note 4)
Timer 3 interrupt request flag (T3F)	○	(Note 4)
Timer 4 interrupt request flag (T4F)	○	(Note 4)
Watchdog timer flag (WDF)	○	(Note 5)
A-D conversion completion flag (ADF)	×	×
Serial I/O transmission/reception completion flag (SIOF)	×	×
Interrupt enable flag (INTE)	×	×

Notes 1. "O" represents that the function can be retained, and "X" represents that the function is initialized.
Registers and flags other than the above are undefined at power down, and set an initial value after returning.
2. The stack pointer (SP) points the level of stack register and is initialized to "7" at power down.
3. The LCD is turned off.
4. The state of the timer is undefined.
5. Stop the watchdog timer with software, and then execute the POF2 instruction.

(5) Identification of the start condition

The start condition (warm start or cold start) can be identified by examining the state of P flag with the SNZP instruction. The warm start condition (timer 4 or external wakeup signal) can be identified by examining the state of the T4F flag.

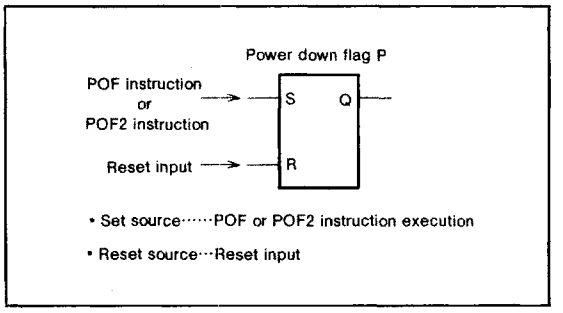


Fig. 58 Set source and reset source of P flag

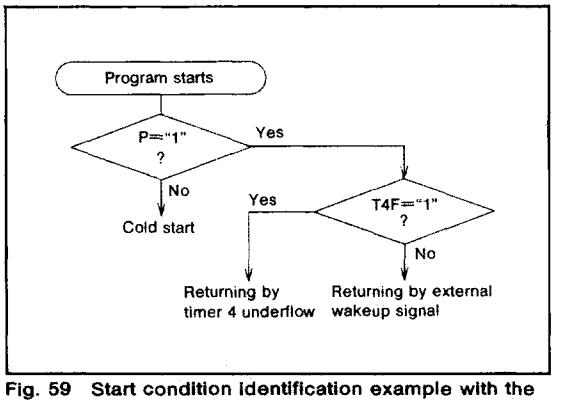


Fig. 59 Start condition identification example with the SNZP instruction

(6) State transition

State transition is described using Fig. 60.

① Cold start (Return from the reset state)

The state is A after a cold start from the reset state. In state A, bit 0 (MR_0) and bit 1 (MR_1) of the clock control register are both "0", and $f(X_{IN})$ is selected as the system clock.

② Transition from state A to low-speed mode state C via state B

First set MR_0 to "1" (state B) to switch the system clock, and then set MR_1 to "1" (state C) to stop $f(X_{IN})$ oscillation.

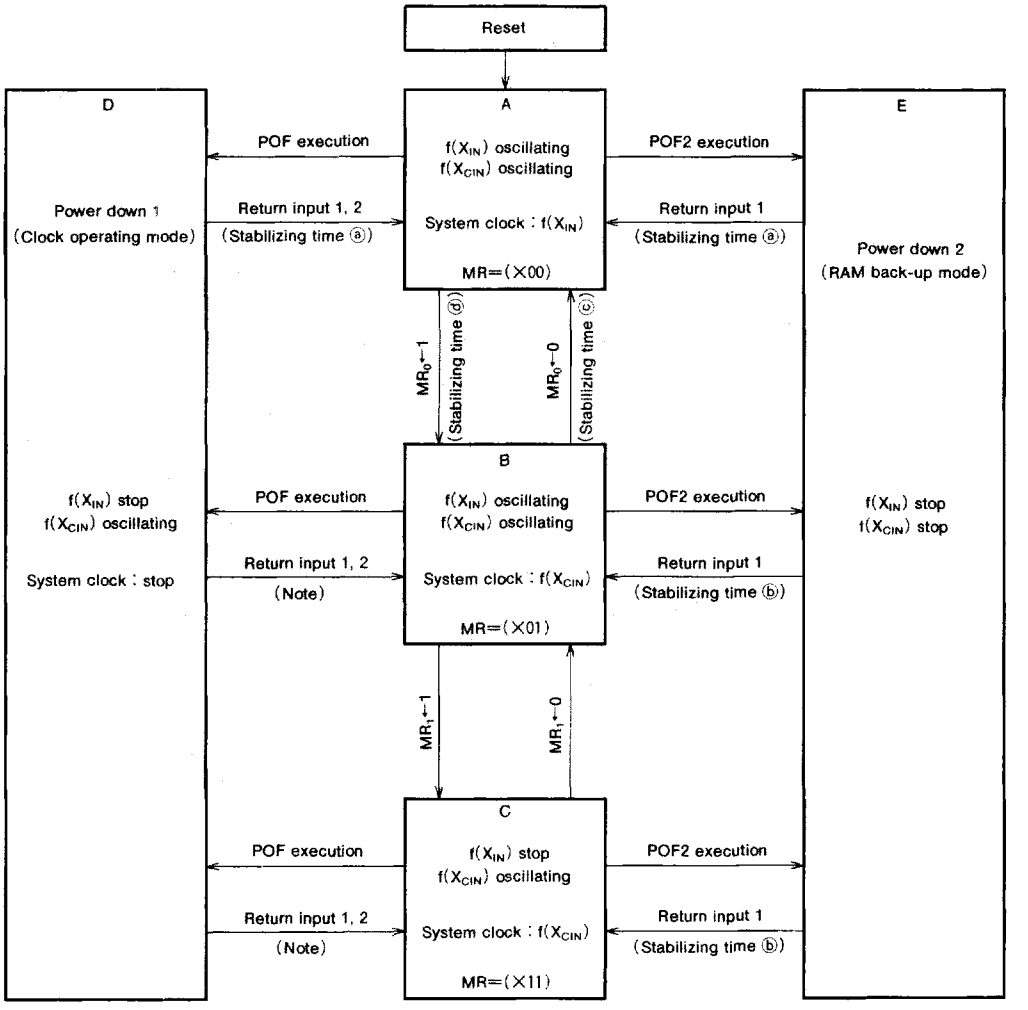
However, after a cold start, do not use $f(X_{CIN})$ as system clock and count source until $f(X_{CIN})$ oscillation sufficiently stabilizes (same as when returning from state E to state A).

③ Transition from state D (power down 1) or state E (power down 2)

The power down 1 (state D) or power down 2 (state E) state can be entered from state A, B, or C with the POF or POF2 instruction. When returning, the state returns to the state before executing the POF or POF2 instruction, but stabilizing time is generated automatically according to the state as shown in the Fig. 60 because the oscillation stabilizing time depends on the state of $f(X_{IN})$ or $f(X_{CIN})$.

④ Transition from state C to state A

First clear MR_1 to "0" to go to state B, generate sufficient time for $f(X_{IN})$ oscillation to stabilize with software, and then clear MR_0 to "0" to go to state A. Also generate sufficient time for $f(X_{IN})$ oscillation to stabilize with software, and then clear MR_0 to "0" to go to state A from state B after the transition to state D from state B with the POF instruction (State transition : B→D→B→A).



Stabilizing time ⓐ : Microcomputer starts its operation after counting the $f(X_{IN})$ clock signal 10746 to 10751 times with hardware to stabilize the $f(X_{IN})$ oscillation.

Stabilizing time ⓑ : Microcomputer starts its operation after counting the $f(X_{CIN})$ clock signal 10746 to 10751 times with hardware to stabilize the $f(X_{CIN})$ oscillation.

Stabilizing time ⓒ : Generate an interval required to stabilize the $f(X_{IN})$ oscillation in state B with software at the transition C→B→A or D→B→A.

Stabilizing time ⓓ : Generate an interval required to stabilize the $f(X_{CIN})$ oscillation in state A with software since the $f(X_{CIN})$ oscillation may be unstable immediately after cold starting or the return from power down 2.

Return input 1 : External wakeup signal ($P0_0$—$P0_3$, $P1_0$—$P1_3$, INT_0)

Return input 2 : Timer 4 interrupt request flag

X : Not used (this bit has no function)

Note : The microcomputer starts its operation after counting $f(X_{CIN})$ 42 to 48 times. (automatic generating)

Fig. 60 State transition diagram

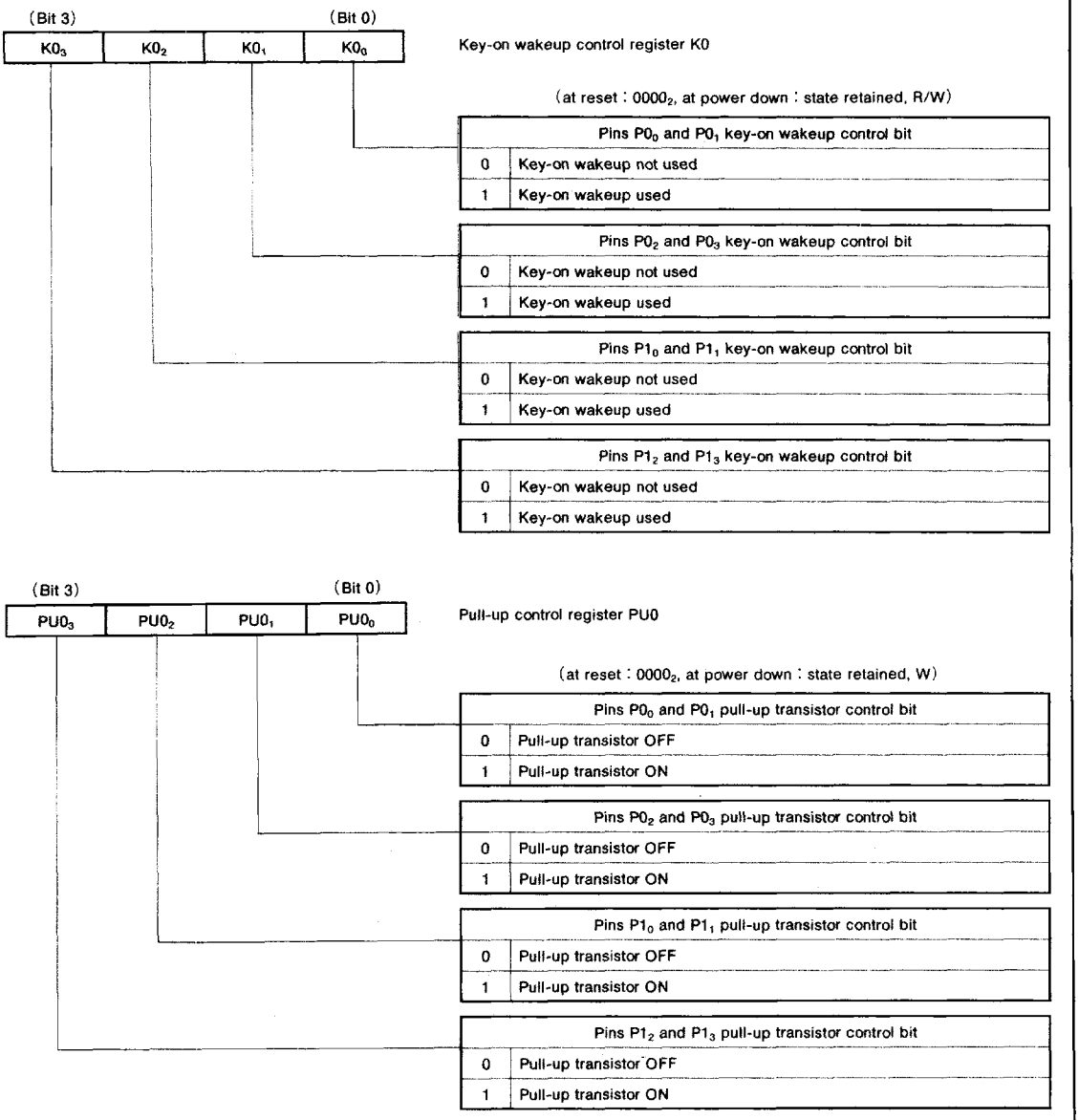
(7) Return signal
An external wakeup signal or timer 4 interrupt request flag is used to return from power down 1. External wakeup signal is used to return from power down 2 because the oscillation is stopped. Table 17 shows the return condition for each return source.

(8) Ports P0, P1 control register
- Key-on wakeup control register (K0)
Register K0 controls the ports P0 and P1 key-on wakeup functions. Set the contents of this register through register A with the TK0A instruction. In addition, the TAK0 instruction can be used to transfer the contents of register K0 to register A.
- Pull-up control register (PU0)
Register PU0 is used to control the ON/OFF of the ports P0 and P1 pull-up transistors. Set the contents of this register through register A with the TPU0A instruction.

Table 17 Return source and return condition

Return Source		Return Condition	Remarks
External wakeup signal	Ports P0, P1	Return by an external falling edge input ("H"→"L").	Set the port using the key-on wakeup function selected with register K0 to "H" level before going into power down state because the falling edge detection circuit is also used as ports P0 and P1.
	Port D_8/INT_0	Return by an external "H" level or "L" level input. The EXF0 flag is not set.	Select the return level ("L" level or "H" level) with bit 2 of the interrupt control register (I1) before going into the power down state according to the external state.
Timer 4 interrupt request flag		Return when the timer 4 underflows and the T4F flag is set to "1".	Allowed only when returning from power down 1 (executing the POF instruction). However, when the POF or POF2 instruction is executed with T4F="1", return condition is recognized and return is performed.

Note : When the POF or POF2 instruction is executed with T4F=1, return condition is recognized and return is immediately performed after going into power down state.



Note : "R" represents read enabled, and "W" represents write enabled.

Fig. 61 Ports P0 and P1 control registers

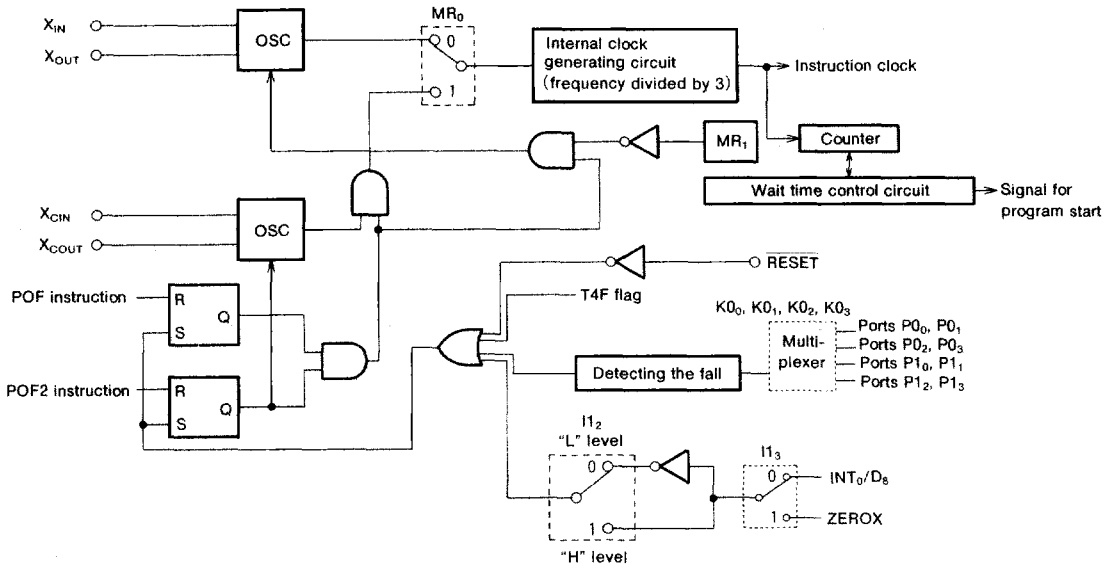
CLOCK CONTROL

The clock control circuit consists of the following circuits.

- $f(X_{IN})$ clock generating circuit
- $f(X_{CIN})$ clock generating circuit
- Control circuit to stop the clock oscillation
- Control circuit to return from power down state

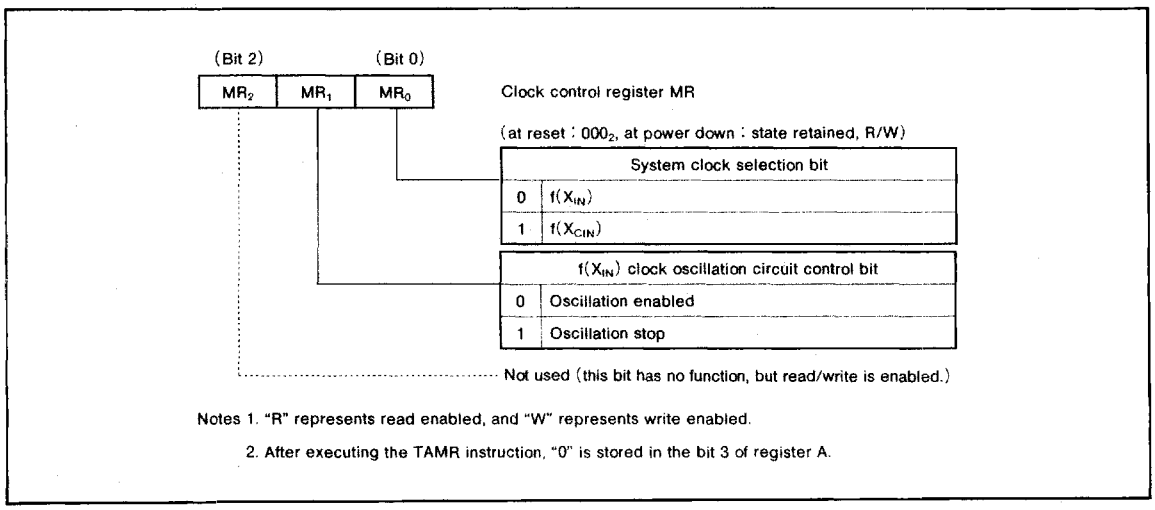
System clock selection and clock oscillation start/stop are controlled with the clock control register (MR).

The $f(X_{IN})$ clock is selected just after a cold start. The instruction clock can be switched to the $f(X_{CIN})$ clock with the system clock selection bit (MR_0). At a warm start, the clock selected just before power down is used. The instruction clock is the signal divided by 3 of the selected clock ($f(X_{IN})/3$ or $f(X_{CIN})/3$).



Note : Refer to Fig. 59 State transition diagram for time set by wait time control circuit.

Fig. 62 Clock control circuit structure



Notes 1. "R" represents read enabled, and "W" represents write enabled.

2. After executing the TAMR instruction, "0" is stored in the bit 3 of register A.

Fig. 63 Registers related clock control

MITSUBISHI ELECTRIC

(1) $f(X_{IN})$ clock generating circuit
Clock oscillation ($f(X_{IN})$) is obtained by externally connecting a ceramic resonator. Connect this external circuit to pins X_{IN} and X_{OUT} at the shortest distance. A feedback resistor is built in between pins X_{IN} and X_{OUT}. When external clock signal is input, connect the clock source to X_{IN} and leave X_{OUT} open. When using an external clock, the maximum value of external clock oscillation frequency is shown in Table 18.

Table 18 Maximum value of external clock oscillation frequency

Supply voltage	Oscillation frequency (duty ratio)
4.5V to 5.5V	3.0MHz (40% to 60%)
2.2V to 5.5V	800kHz (30% to 70%)

Note : 2.5V to 5.5V for One Time and EPROM version

(2) $f(X_{CIN})$ clock generating circuit
Clock oscillation ($f(X_{CIN})$) is obtained by externally connecting a quartz-crystal oscillator. Connect this external circuit to pins X_{CIN} and X_{COUT} at the shortest distance. A feedback resistor is built-in between pins X_{CIN} and X_{COUT}.
Unlike the $f(X_{IN})$ clock generating circuit, external clock signal cannot be used for this circuit.

ROM ordering method
Please submit the information described below when ordering Mask ROM.
(1) M34520M6A-XXXSP/FP ROM Order Confirmation Form or M34520M8A-XXXSP/FP ROM Order Confirmation Form 1
(2) Data to be written into mask ROM ..... EPROM (three sets containing the identical data)
(3) Mark Specification Form 1

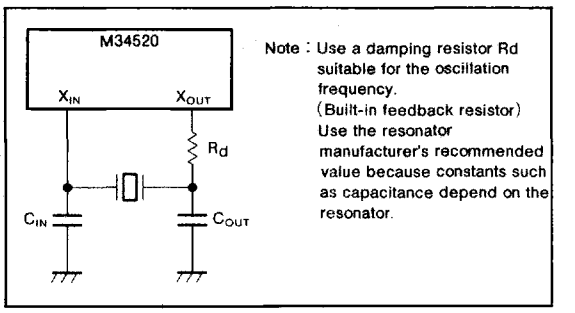


Fig. 64 Ceramic resonator external circuit

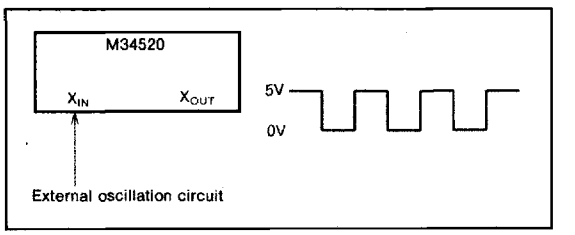


Fig. 65 External clock input circuit

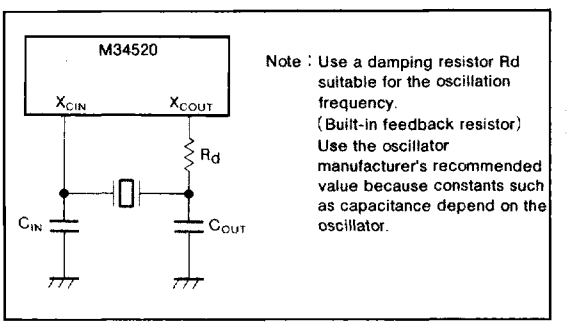


Fig. 66 Quartz-crystal oscillator external circuit

LIST OF PRECAUTIONS

(1) Noise and latch-up prevention

Connect a capacitor (approx. 0.1μF) between pins V_{DD} and V_{SS} at the shortest distance using the thickest wire and equalizing its width and length to prevent noise and latch-up.

In the built-in PROM versions, $P2_3$ pin is also used as V_{PP} pin. Accordingly, when using this pin, use this pin through a resistor about 5kΩ in series at the shortest distance. When not using the $P2_3$ pin of M34520E8, connect to V_{SS} through a 5kΩ resistor at the shortest distance.

The V_{PP} pin functions as the power supply input pin for the internal EPROM, and the impedance is lowered to enable current for writing to flow when writing a program to the EPROM. Accordingly, unless it is connected as shown in Fig. 67, noise may be collected into the internal EPROM from the V_{PP} pin and this may cause program run-away because instruction or data from the EPROM cannot be read correctly.

The location of the bypass capacitor is significant in guarding against the strong noise from the V_{SS} and V_{DD} lines. As shown in Fig. 68, after connecting the bypass capacitor to V_{DD} and V_{SS} lines, connect at equal distance to the pins V_{SS} and V_{DD} of the microcomputer.

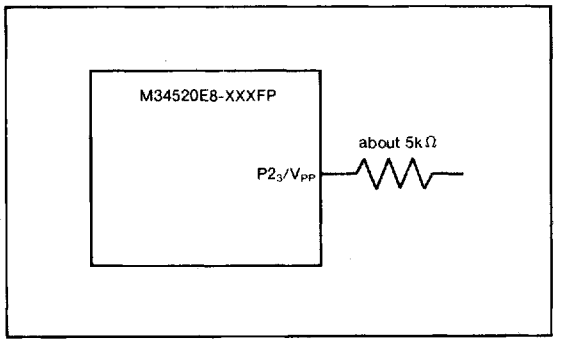


Fig 67 Wiring example when using $P2_3/V_{PP}$ pin

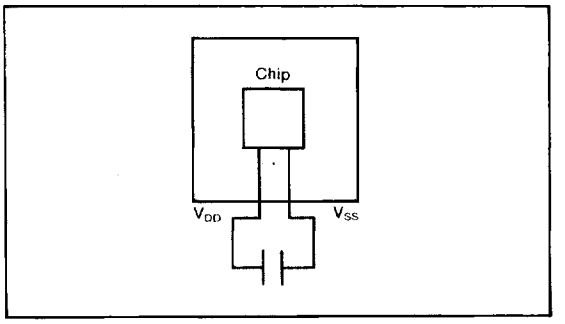


Fig. 68 Noise prevention for V_{DD} and V_{SS} lines

This will significantly reduce the effect of noise.

(2) Prescaler

Stop the prescaler to change its frequency dividing ratio.

(3) Timer 1, 2, 3, 4

Stop timer 1, 2, 3, or 4 counting to change its count source, as well as to execute the TAB1, TAB2, or TAB4 instruction for reading the data (from timer 1, 2, or 4)

When timer 4 write instruction (T4ABD) is executed, count value of timer 4 is invalid. Do not execute the timer 4 write instruction (T4ABD) just before or after timer 4 underflow signal occurs.

Fully stabilize the f(X_{CIN}) oscillation so as to select it as a timer 3 count source.

(4) Notes on unused pins

When the $P0_0 - P0_3$ and $P1_0 - P1_3$ are connected to V_{SS}, turn off their pull-up transistors with software (PU0$_i$ = "0"), and also invalidate key-on wakeup functions (K0$_i$="0").

If the key-on wakeup functions are left valid, the system fails to return from power down. When these pins are disconnected, turn on their pull-up transistors (register PU0$_i$ = "1") with software. Be sure to invalidate the key-on wakeup functions and the pull-up functions with every two bits. If only one of the two bits key-on wakeup functions is used, turn on their pull-up transistors with software and also disconnect the other pin. (i represents 0, 1, 2, or 3.)

(5) Built-in PROM version precautions

The operating supply voltage of the built-in EPROM version (M34520E8SS, M34520E8FS) and the One Time PROM version (M34520E8-XXXSP/FP, M34520E8SP/FP) is 2.5V to 5.5V.

The operating temperature range of the built-in EPROM version is −20℃ to 70℃.

Built-in EPROM version is the microcomputer for program development. Use this microcomputer only for program development and prototype test.

(6) Notes on INT_0 pin

When the input edge polarity of the INT_0 pin is changed with the bit 2 of the register I1 in a program, be careful about the following notes.

- Clear the bit 0 of the register V1 to "0" before the input edge polarity of the interrupt input pin INT_0 is changed with the bit 2 of the register I1 (refer to Fig. 69①).
- Depending on the input state of the INT_0 pin, the external 0 interrupt request flag EXF0 may be set when the input edge polarity is changed. Accordingly, set a value to register I1, and execute the SNZ0 instruction after executing at least one instruction (refer to Fig. 69②), and then clear the EXF0 flag.

```
    :
LA   4          ;(XXX0₂)
TV1A            ; The SNZ0 instruction is valid·····①
LA   4
TI1A            ; Change of the input polarity
                ·····②
NOP
SNZ0            ; The SNZ0 instruction is executed
NOP
    :
```

Fig. 69 External 0 interrupt program example

(7) Notes on pins INT_1 and INT_2

When the input edge polarity of pins INT_1 and INT_2 are changed with the bit 2 of the register I2 in a program, be careful about the following notes.

- Clear the bit 1 of the register V1 to "0" before the input edge polarity of the interrupt input pins INT_1 and INT_2 are changed with the bit 2 of the register I2 (refer to Fig. 70③).
- Depending on the input state of pins INT_1 and INT_2, the external 1 interrupt request flag EXF1 may be set when the input edge polarity is changed.
  Accordingly, set a value to register I2, and execute the SNZ1 instruction after executing at least one instruction (refer to Fig. 70④), and then clear the EXF1 flag.

```
    :
LA   8          ;(XX0X₂)
TV1A            ; The SNZ1 instruction is valid·····③
LA   4
TI2A            ; Change of the input polarity
                ·····④
NOP
SNZ1            ; The SNZ1 instruction is executed
NOP
    :
```

Fig. 70 External 1, 2 interrupt program example

(8) Note on noise detection circuit

Be careful about the following note when using the noise detection circuit without using the zero cross detection function.

- Clear the bit 2($W5_2$) of timer control register (W5) to "0" and fix the ZCF flag to reset state because the noise canceller flag (NCF) is cleared to "0" when the zero cross input flag (ZCF) is set to "1".

(9) Note on f(X_{IN}) clock generating circuit

When using a ceramic resonator for f(X_{IN}) clock, connect this ceramic resonator external circuit to pins X_{IN} and X_{OUT} at the shortest distance.

(10) Note on f(X_{CIN}) clock generating circuit

When using f(X_{CIN}) clock, connect the quarts-crystal oscillator external circuit to pins X_{CIN} and X_{COUT} at the shortest distance.

Unlike the f(X_{IN}) clock generating circuit, external clock signal cannot be used for this circuit.

(11) Note on A-D conversion

A-D conversion circuit is equipped with a capacitor which is used to compare the analog voltage. Accordingly, when the analog voltage is input from the circuit with high-impedance and, charge/discharge noise is generated and the sufficient A-D accuracy may not be obtained. Therefore, reduce the impedance or, connect a capacitor (0.01μF to 1μF) to analog input pins.

When the overvoltage applied to the A-D conversion circuit may occur, connect an external circuit in order to keep the voltage within the specifications as shown the Fig. 72. In addition, test the application products sufficiently.

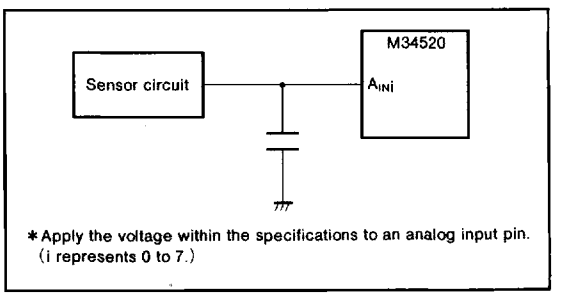


Fig. 71 Analog input external circuit example-1

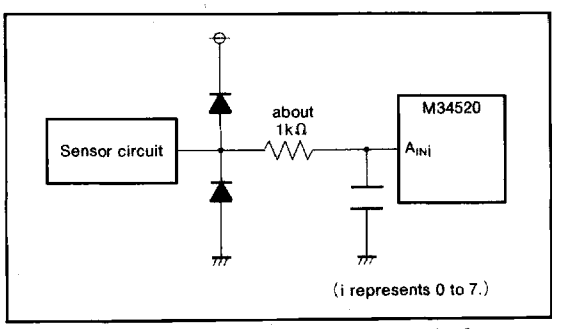


Fig. 72 Analog input external circuit example-2

(12) Note on interrupt
Be sure to disable interrupts by executing the DI instruction before the POF (power down 1) or POF2 (power down 2) instruction is executed.

(13) Note on program counter
Make sure that the PC_H does not specify after the last page of the built-in ROM.

(14) Note on LCD
The 4520 Group has the LCD dividing resistor that can be disconnected by software. Select whether to connect this internal dividing resistor or not, the LCD power supply circuit appropriate for the LCD panel being used, and test sufficiently.

LIST OF INSTRUCTION FUNCTION

Grouping	Mnemonic	Function
Register to register transfer	TAB	(A)←(B)
	TBA	(B)←(A)
	TAY	(A)←(Y)
	TYA	(Y)←(A)
	TEAB	(E_7—E_4)←(B) (E_3—E_0)←(A)
	TABE	(B)←(E_7—E_4) (A)←(E_3—E_0)
	TDA	(DR_2—DR_0)←(A_2—A_0)
	TAD	(A_2—A_0)←(DR_2—DR_0) (A_3)←0
	TAZ	(A_1, A_0)←(Z_1, Z_0) (A_3, A_2)←0
	TAX	(A)←(X)
	TASP	(A_2—A_0)←(SP_2—SP_0) (A_3)←0
RAM addresses	LXY x, y	(X)←x, x=0 to 15 (Y)←y, y=0 to 15
	LZ z	(Z)←z, z=0 to 3
	INY	(Y)←(Y)+1
	DEY	(Y)←(Y)−1
RAM to register transfer	TAM j	(A)←(M(DP)) (X)←(X)EXOR(j) j=0 to 15
	XAM j	(A)←→(M(DP)) (X)←(X)EXOR(j) j=0 to 15
	XAMD j	(A)←→(M(DP)) (X)←(X)EXOR(j) j=0 to 15 (Y)←(Y)−1
RAM to register transfer	XAMI j	(A)←→(M(DP)) (X)←(X)EXOR(j) j=0 to 15 (Y)←(Y)+1
	TMA j	(M(DP))←(A) (X)←(X)EXOR(j) j=0 to 15
Arithmetic operation	LA n	(A)←n However, n=0 to 15
	TABP p	(SP)←(SP)+1 (SK(SP))←(PC) (PC_H)←p (PC_L)←(DR_2—DR_0, A_3—A_0) (B)←(ROM(PC))$_{7\ to\ 4}$ (A)←(ROM(PC))$_{3\ to\ 0}$ (PC)←(SK(SP)) (SP)←(SP)−1
	AM	(A)←(A)+(M(DP))
	AMC	(A)←(A)+(M(DP))+(CY) (CY)←Carry
	A n	(A)←(A)+n However, n=0 to 15
	AND	(A)←(A)AND(M(DP))
	OR	(A)←(A)OR(M(DP))
	SC	(CY)←1
	RC	(CY)←0
	SZC	(CY)=0?
	CMA	(A)←($\overline{A}$)
	RAR	CY → $A_3A_2A_1A_0$
Bit operation	SB j	(Mj(DP))←1 j=0 to 3
	RB j	(Mj(DP))←0 j=0 to 3
Bit operation	SZB j	(Mj(DP))←0? j=0 to 3
Comparison operation	SEAM	(A)=(M(DP))?
	SEA n	(A)=n? However, n=0 to 15
Branch operation	B a	(PC_L)←a_6—a_0
	BL p, a	(PC_H)←p (PC_L)←a_6—a_0
	BLA p	(PC_H)←p (PC_L)←(DR_2—DR_0, A_3—A_0)
Subroutine operation	BM a	(SP)←(SP)+1 (SK(SP))←(PC) (PC_H)←2 (PC_L)←a_6—a_0
	BML p, a	(SP)←(SP)+1 (SK(SP))←(PC) (PC_H)←p (PC_L)←a_6—a_0
	BMLA p	(SP)←(SP)+1 (SK(SP))←(PC) (PC_H)←p (PC_L)←(DR_2—DR_0, A_3—A_0)
Return operation	RTI	(PC)←(SK(SP)) (SP)←(SP)−1
	RT	(PC)←(SK(SP)) (SP)←(SP)−1
	RTS	(PC)←(SK(SP)) (SP)←(SP)−1

Grouping	Mnemonic	Function
Interrupt operation	DI	(INTE)←0
	EI	(INTE)←1
	SNZ0	(EXF0)=1? After skipping (EXF0)←0
	SNZ1	(EXF1)=1? After skipping (EXF1)←0
	SNZI0	$I1_2$=1 : (INT_0)="H" $I1_2$=0 : (INT_0)="L"
	SNZI1	$I2_2$=1 : (INT_1)="H" $I2_2$=0 : (INT_1)="L"
	TAV1	(A)←(V1)
	TV1A	(V1)←(A)
	TAV2	(A)←(V2)
	TV2A	(V2)←(A)
	TAI1	(A)←(I1)
	TI1A	(I1)←(A)
	TAI2	(A)←(I2)
	TI2A	(I2)←(A)
	TAI3	(A_0)←(I3) (A_3—A_1)←0
	TI3A	(I3)←(A_0)
Timer operation	TAW1	(A)←(W1)
	TW1A	(W1)←(A)
	TAW2	(A)←(W2)
	TW2A	(W2)←(A)
	TAW3	(A)←(W3)
	TW3A	(W3)←(A)
	TAW4	(A)←(W4)
	TW4A	(W4)←(A)
	TAW5	(A_2—A_0)←($W5_2$—$W5_0$) (A_3)←0

Grouping	Mnemonic	Function
Timer operation	TW5A	($W5_2$—$W5_0$)←(A_2—A_0)
	TAB1	(B)←($T1_7$—$T1_4$) (A)←($T1_3$—$T1_0$)
	T1AB	($R1_7$—$R1_4$)←(B) ($T1_7$—$T1_4$)←(B) ($R1_3$—$R1_0$)←(A) ($T1_3$—$T1_0$)←(A)
	TAB2	(B)←($T2_7$—$T2_4$) (A)←($T2_3$—$T2_0$)
	T2AB	($R2_7$—$R2_4$)←(B) ($T2_7$—$T2_4$)←(B) ($R2_3$—$R2_0$)←(A) ($T2_3$—$T2_0$)←(A)
	TAB4	(B)←($T4_7$—$T4_4$) (A)←($T4_3$—$T4_0$)
	T4AB	($R4_7$—$R4_4$)←(B) ($T4_7$—$T4_4$)←(B) ($R4_3$—$R4_0$)←(A) ($T4_3$—$T4_0$)←(A)
	T4ABD	($T4_7$—$T4_4$)←(B) ($T4_3$—$T4_0$)←(A)
	TR4AB	($R4_7$—$R4_4$)←(B) ($R4_3$—$R4_0$)←(A)
	SNZT1	(T1F)=1? After skipping (T1F)←0
	SNZT2	(T2F)=1? After skipping (T2F)←0
	SNZT3	(T3F)=1? After skipping (T3F)←0
	SNZT4	(T4F)=1? After skipping (T4F)←0
Input/Output operation	IAP0	(A)←(P0)
	OP0A	(P0)←(A)
	IAP1	(A)←(P1)
	OP1A	(P1)←(A)
	IAP2	(A)←(P2)

Grouping	Mnemonic	Function
Input/Output operation	IAP3	(A)←(P3)
	IAP4	(A)←(P4)
	CLD	(D)←1
	RD	(D(Y))←0 (Y)=0 to 10
	SD	(D(Y))←1 (Y)=0 to 10
	SZD	(D(Y))=0? (Y)=0 to 10
	TK0A	(K0)←(A)
	TAK0	A←(K0)
	TPU0A	(PU0)←(A)
LCD control operation	TL1A	(L1)←(A)
	TAL1	(A)←(L1)
	TL2A	(L2)←(A)
	TL3A	(L3)←(A_0)
	TLCA	(LC)←(A)
Serial I/O control operation	TAH	(A)←(H)
	THA	(H)←(A)
	TAL	(A)←(L)
	TLA	(L)←(A)
	TAJ1	(A_2—A_0)←($J1_2$—$J1_0$) (A_3)←0
	TJ1A	($J1_2$—$J1_0$)←(A_2—A_0)
	TAJ2	(A_1, A_0)←($J2_1$, $J2_0$) (A_3, A_2)←0
	TJ2A	($J2_1$, $J2_0$)←(A_1, A_0)
	SST	(SIOF)←0, Serial I/O starts
	SNZSI	(SIOF)=1? After skipping, (SIOF)←0

MITSUBISHI ELECTRIC

Grouping	Mnemonic	Function
A-D conversion operation	TAHA	(A)←(HA)
	TALA	(A)←(LA)
	TAQ1	$(A_2-A_0)\leftarrow(Q1_2-Q1_0)$ $(A_3)\leftarrow 0$
	TQ1A	$(Q1_2-Q1_0)\leftarrow(A_2-A_0)$
	ADST	A-D conversion starts
	SNZAD	(ADF)=1? After skipping, (ADF)←0
Other operation	NOP	(PC)←(PC)+1
	POF	Power down 1
	POF2	Power down 2
	SNZP	(P)=1?
	TAMR	$(A_2-A_0)\leftarrow(MR_2-MR_0)$ $(A_3)\leftarrow 0$
	TMRA	$(MR_2-MR_0)\leftarrow(A_2-A_0)$
	WRST	(WDF)←0
	RTPS	$(RTPL_1)\leftarrow 1$ $(RTPL_0)\leftarrow 1$
	RTPR	$(RTPL_1)\leftarrow 0$ $(RTPL_0)\leftarrow 0$
	TRTPA	(RTP)←(A)

INSTRUCTION CODE TABLE

D_3–D_0 \ D_9–D_4	Hexadecimal notation	000000	000001	000010	000011	000100	000101	000110	000111	001000	001001	001010	001011	001100	001101	001110	001111	010000 to 010111	011000 to 011111
		00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10 to 17	18 to 1F
0000	0	NOP	BLA	SZB 0	BMLA	—	TASP	A 0	LA 0	TABP 0	TABP 16	TABP 32	TABP 48*	BML	BML	BL	BL	BM	B
0001	1	—	CLD	SZB 1	—	—	TAD	A 1	LA 1	TABP 1	TABP 17	TABP 33	TABP 49*	BML	BML	BL	BL	BM	B
0010	2	POF	—	SZB 2	—	—	TAX	A 2	LA 2	TABP 2	TABP 18	TABP 34	TABP 50*	BML	BML	BL	BL	BM	B
0011	3	SNZP	INY	SZB 3	—	—	TAZ	A 3	LA 3	TABP 3	TABP 19	TABP 35	TABP 51*	BML	BML	BL	BL	BM	B
0100	4	DI	RD	SZD	—	RT	TAV1	A 4	LA 4	TABP 4	TABP 20	TABP 36	TABP 52*	BML	BML	BL	BL	BM	B
0101	5	EI	SD	SEAn	—	RTS	TAV2	A 5	LA 5	TABP 5	TABP 21	TABP 37	TABP 53*	BML	BML	BL	BL	BM	B
0110	6	RC	—	SEAM	—	RTI	—	A 6	LA 6	TABP 6	TABP 22	TABP 38	TABP 54*	BML	BML	BL	BL	BM	B
0111	7	SC	DEY	—	—	—	—	A 7	LA 7	TABP 7	TABP 23	TABP 39	TABP 55*	BML	BML	BL	BL	BM	B
1000	8	POF2	AND	—	SNZ0	LZ 0	—	A 8	LA 8	TABP 8	TABP 24	TABP 40	TABP 56*	BML	BML	BL	BL	BM	B
1001	9	—	OR	TDA	SNZ1	LZ 1	—	A 9	LA 9	TABP 9	TABP 25	TABP 41	TABP 57*	BML	BML	BL	BL	BM	B
1010	A	AM	TEAB	TABE	SNZI0	LZ 2	—	A 10	LA 10	TABP 10	TABP 26	TABP 42	TABP 58*	BML	BML	BL	BL	BM	B
1011	B	AMC	—	—	SNZI1	LZ 3	—	A 11	LA 11	TABP 11	TABP 27	TABP 43	TABP 59*	BML	BML	BL	BL	BM	B
1100	C	TYA	CMA	—	—	RB 0	SB 0	A 12	LA 12	TABP 12	TABP 28	TABP 44	TABP 60*	BML	BML	BL	BL	BM	B
1101	D	—	RAR	—	—	RB 1	SB 1	A 13	LA 13	TABP 13	TABP 29	TABP 45	TABP 61*	BML	BML	BL	BL	BM	B
1110	E	TBA	TAB	—	TV2A	RB 2	SB 2	A 14	LA 14	TABP 14	TABP 30	TABP 46	TABP 62*	BML	BML	BL	BL	BM	B
1111	F	—	TAY	SZC	TV1A	RB 3	SB 3	A 15	LA 15	TABP 15	TABP 31	TABP 47	TABP 63*	BML	BML	BL	BL	BM	B

The above table shows the relationship between machine language codes and machine language instructions. D_3–D_0 show the low-order 4 bits of the machine language code, and D_9–D_4 show the high-order 6 bits of the machine language code. The hexadecimal representation of the code is also provided. There are one-word instructions and two-word instructions, but only the first word of each instruction is shown. Do not use the code marked "—".

The codes for the second word of a two-word instruction are described below.

	The second word
BL	10 paaa aaaa
BML	10 paaa aaaa
BLA	10 pp00 pppp
BMLA	10 pp00 pppp
SEA	00 0111 nnnn
SZD	00 0010 1011

* cannot be used at M34520M6A-XXXSP/FP.

D_3—D_0 (Hexadecimal notation) \ D_9—D_4	100000	100001	100010	100011	100100	100101	100110	100111	101000	101001	101010	101011	101100	101101	101110	101111	110000 111111
0000 (0)	THA	TW3A	OP0A	T1AB	TAH	—	IAP0	TAB1	SNZT1	—	WRST	TMA 0	TAM 0	XAM 0	XAMI 0	XAMD 0	LXY
0001 (1)	TLA	TW4A	OP1A	T2AB	TAL	—	IAP1	TAB2	SNZT2	—	RTPS	TMA 1	TAM 1	XAM 1	XAMI 1	XAMD 1	LXY
0010 (2)	TJ1A	TW5A	—	—	TAJ1	TAMR	IAP2	—	SNZT3	—	RTPR	TMA 2	TAM 2	XAM 2	XAMI 2	XAMD 2	LXY
0011 (3)	TJ2A	—	—	T4AB	TAJ2	TAI1	IAP3	TAB4	SNZT4	—	—	TMA 3	TAM 3	XAM 3	XAMI 3	XAMD 3	LXY
0100 (4)	TQ1A	—	—	—	TAQ1	TAI2	IAP4	—	—	—	—	TMA 4	TAM 4	XAM 4	XAMI 4	XAMD 4	LXY
0101 (5)	—	—	—	—	—	TAI3	—	—	—	—	—	TMA 5	TAM 5	XAM 5	XAMI 5	XAMD 5	LXY
0110 (6)	—	TMRA	—	T4ABD	—	TAK0	—	—	—	—	—	TMA 6	TAM 6	XAM 6	XAMI 6	XAMD 6	LXY
0111 (7)	—	TI1A	—	—	—	—	—	—	SNZAD	—	—	TMA 7	TAM 7	XAM 7	XAMI 7	XAMD 7	LXY
1000 (8)	—	TI2A	—	—	TAHA	—	—	—	SNZSI	—	—	TMA 8	TAM 8	XAM 8	XAMI 8	XAMD 8	LXY
1001 (9)	—	TRTPA	—	—	TALA	—	—	—	—	—	—	TMA 9	TAM 9	XAM 9	XAMI 9	XAMD 9	LXY
1010 (A)	TL1A	TI3A	—	—	TAL1	—	—	—	—	—	—	TMA 10	TAM 10	XAM 10	XAMI 10	XAMD 10	LXY
1011 (B)	TL2A	TK0A	—	—	TAW1	—	—	—	—	—	—	TMA 11	TAM 11	XAM 11	XAMI 11	XAMD 11	LXY
1100 (C)	TL3A	—	—	TR4AB	TAW2	—	—	—	—	—	—	TMA 12	TAM 12	XAM 12	XAMI 12	XAMD 12	LXY
1101 (D)	TLCA	—	TPU0A	—	TAW3	—	—	—	—	—	—	TMA 13	TAM 13	XAM 13	XAMI 13	XAMD 13	LXY
1110 (E)	TW1A	—	—	—	TAW4	—	—	—	—	SST	—	TMA 14	TAM 14	XAM 14	XAMI 14	XAMD 14	LXY
1111 (F)	TW2A	—	—	—	TAW5	—	—	—	—	ADST	—	TMA 15	TAM 15	XAM 15	XAMI 15	XAMD 15	LXY

The above table shows the relationship between machine language codes and machine language instructions. D_3—D_0 show the low-order 4 bits of the machine language code, and D_9—D_4 show the high-order 6 bits of the machine language code. The hexadecimal representation of the code is also provided. There are one-word instructions and two-word instructions, but only the first word of each instruction is shown. Do not use the code marked "—".

The codes for the second word of a two-word instruction are described below.

	The second word
BL	1 0 p a a a a a a a a
BML	1 0 p a a a a a a a a
BLA	1 0 p p 0 0 p p p p
BMLA	1 0 p p 0 0 p p p p
SEA	0 0 0 1 1 1 n n n n
SZD	0 0 0 0 1 0 1 0 1 1

MACHINE INSTRUCTIONS

Type of instructions	Mnemonic	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation	Number of words	Number of cycles	Functions
Register to register transfer	TAB	0	0	0	0	0	1	1	1	1	0	0 1 E	1	1	(A)←(B)
	TBA	0	0	0	0	0	0	1	1	1	0	0 0 E	1	1	(B)←(A)
	TAY	0	0	0	0	0	1	1	1	1	1	0 1 F	1	1	(A)←(Y)
	TYA	0	0	0	0	0	0	1	1	0	0	0 0 C	1	1	(Y)←(A)
	TEAB	0	0	0	0	0	1	1	0	1	0	0 1 A	1	1	(E_7—E_4)←(B), (E_3—E_0)←(A)
	TABE	0	0	0	0	1	0	1	0	1	0	0 2 A	1	1	(B)←(E_7—E_4), (A)←(E_3—E_0)
	TDA	0	0	0	0	1	0	1	0	0	1	0 2 9	1	1	(DR_2—DR_0)←(A_2—A_0)
	TAD	0	0	0	1	0	1	0	0	0	1	0 5 1	1	1	(A_2—A_0)←(DR_2—DR_0), (A_3)←0
	TAZ	0	0	0	1	0	1	0	0	1	1	0 5 3	1	1	(A_1, A_0)←(Z_1, Z_0) (A_3, A_2)←0
	TAX	0	0	0	1	0	1	0	0	1	0	0 5 2	1	1	(A)←(X)
	TASP	0	0	0	1	0	1	0	0	0	0	0 5 0	1	1	(A_2—A_0)←(SP_2—SP_0) (A_3)←0
RAM addresses	LXY x, y	1	1	x_3	x_2	x_1	x_0	y_3	y_2	y_1	y_0	3 x y	1	1	(X)←x, x= 0 to 15 (Y)←y, y= 0 to 15
	LZ z	0	0	0	1	0	0	1	0	z_1	z_0	0 4 8+z	1	1	(Z)←z, z= 0 to 3
	INY	0	0	0	0	0	1	0	0	1	1	0 1 3	1	1	(Y)←(Y)+ 1
	DEY	0	0	0	0	0	1	0	1	1	1	0 1 7	1	1	(Y)←(Y)− 1
RAM to register transfer	TAM j	1	0	1	1	0	0	j	j	j	j	2 C j	1	1	(A)←(M(DP)) (X)←(X)EXOR(j), j= 0 to 15
	XAM j	1	0	1	1	0	1	j	j	j	j	2 D j	1	1	(A)←→(M(DP)) (X)←(X)EXOR(j), j= 0 to 15
	XAMD j	1	0	1	1	1	1	j	j	j	j	2 F j	1	1	(A)←→(M(DP)) (X)←(X)EXOR(j), j= 0 to 15 (Y)←(Y)− 1
	XAMI j	1	0	1	1	1	0	j	j	j	j	2 E j	1	1	(A)←→(M(DP)) (X)←(X)EXOR(j), j= 0 to 15 (Y)←(Y)+ 1
	TMA j	1	0	1	0	1	1	j	j	j	j	2 B j	1	1	(M(DP))←(A) (X)←(X)EXOR(j), j= 0 to 15
Arithmetic operation	LA n	0	0	0	1	1	1	n	n	n	n	0 7 n	1	1	(A)←n, However, n= 0 to 15
	TABP p	0	0	1	0	p_5	p_4	p_3	p_2	p_1	p_0	0 8 p+p	1	3	(SP)←(SP)+ 1 (SK(SP))←(PC) (PC_H)←p (PC_L)←(DR_2—DR_0, A_3—A_0) (B)←(ROM(PC))$_{7\ to\ 4}$ (A)←(ROM(PC))$_{3\ to\ 0}$ (PC)←(SK(SP)) (SP)←(SP)− 1 (Note)
	AM	0	0	0	0	0	0	1	0	1	0	0 0 A	1	1	(A)←(A)+(M(DP))

Note : p is 0 to 63 for M34520M8A.
p is 0 to 47 for M34520M6A.

MITSUBISHI ELECTRIC

Skip condition	Carry flag CY	Detailed description
—	—	Transfers the contents of register B to register A.
—	—	Transfers the contents of register A to register B.
—	—	Transfers the contents of register Y to register A.
—	—	Transfers the contents of register A to register Y.
—	—	Transfers the contents of registers A and B to register E.
—	—	Transfers the contents of register E to registers A and B.
—	—	Transfers the contents of register A to register D.
—	—	Transfers the contents of register D to register A.
—	—	Transfers the contents of register Z to register A.
—	—	Transfers the contents of register X to register A.
—	—	Transfers the contents of stack pointer (SP) to register A.
Continuous description	—	Loads the value x in the immediate field to register X, and the value y in the immediate field to register Y. When the LXY instructions are continuously coded and executed, only the first LXY instruction is executed and other LXY instructions coded continuously are skipped.
—	—	Loads the value z in the immediate field to register Z.
(Y) = 0	—	Adds 1 to the contents of register Y. As a result of addition, when the contents of register Y is 0, the next instruction is skipped.
(Y) = 15	—	Subtracts 1 from the contents of register Y. As a result of subtraction, when the contents of register Y is 15, the next instruction is skipped.
—	—	After transferring the contents of M(DP) to register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
—	—	After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
(Y) = 15	—	After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X. Subtracts 1 from the contents of register Y. As a result of subtraction, when the contents of register Y is 15, the next instruction is skipped.
(Y) = 0	—	After exchanging the contents of M(DP) with the contents of register A, an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X. Adds 1 to the contents of register Y. As a result of addition, when the contents of register Y is 0, the next instruction is skipped.
—	—	After transferring the contents of register A to M(DP), an exclusive OR operation is performed between register X and the value j in the immediate field, and stores the result in register X.
Continuous description	—	Loads the value n in the immediate field to the register A. When the LA instructions are continuously coded and executed, only the first LA instruction is executed and other LA instructions coded continuously are skipped.
—	—	Transfers bits 7 to 4 to register B and bits 3 to 0 to register A. These bits 7 to 0 are the ROM pattern in address (DR_2 DR_1 DR_0 A_3 A_2 A_1 A_0) specified by registers A and D in page p. When this instruction is executed, 1 stage of stack register is used.
—	—	Adds the contents of M(DP) to register A. Stores the result in register A. The contents of carry flag CY remains unchanged.

Type of instructions	Mnemonic	Instruction code D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation			Number of words	Number of cycles	Functions
Arithmetic operation	AMC	0	0	0	0	0	0	1	0	1	1	0	0	B	1	1	(A)←(A)+(M(DP))+(CY) (CY)←Carry
	An	0	0	0	1	1	0	n	n	n	n	0	6	n	1	1	(A)←(A)+n However, n= 0 to 15
	AND	0	0	0	0	0	1	1	0	0	0	0	1	8	1	1	(A)←(A) AND (M(DP))
	OR	0	0	0	0	0	1	1	0	0	1	0	1	9	1	1	(A)←(A) OR (M(DP))
	SC	0	0	0	0	0	0	0	1	1	1	0	0	7	1	1	(CY)←1
	RC	0	0	0	0	0	0	0	1	1	0	0	0	6	1	1	(CY)←0
	SZC	0	0	0	0	1	0	1	1	1	1	0	2	F	1	1	(CY)= 0 ?
	CMA	0	0	0	0	0	1	1	1	0	0	0	1	C	1	1	(A)←($\overline{A}$)
	RAR	0	0	0	0	0	1	1	1	0	1	0	1	D	1	1	→ CY → $A_3A_2A_1A_0$
Bit operation	SB j	0	0	0	1	0	1	1	1	j	j	0	5	C+j	1	1	(Mj(DP))←1 j= 0 to 3
	RB j	0	0	0	1	0	0	1	1	j	j	0	4	C+j	1	1	(Mj(DP))←0 j= 0 to 3
	SZB j	0	0	0	0	1	0	0	0	j	j	0	2	j	1	1	(Mj(DP))= 0 ? j= 0 to 3
Comparison operation	SEAM	0	0	0	0	1	0	0	1	1	0	0	2	6	1	1	(A)=(M(DP)) ?
	SEA n	0	0	0	0	1	0	0	1	0	1	0	2	5	2	2	(A)=n ?
		0	0	0	1	1	1	n	n	n	n	0	7	n			However, n= 0 to 15
Branch operation	Ba	0	1	1	a_6	a_5	a_4	a_3	a_2	a_1	a_0	1	8	a+a	1	1	(PC_L)←a_6–a_0
	BL p, a	0	0	1	1	1	p_4	p_3	p_2	p_1	p_0	0	E	p+p	2	2	(PC_H)←p (PC_L)←a_6–a_0 (Note)
		1	0	p_5	a_6	a_5	a_4	a_3	a_2	a_1	a_0	2	p	a+a			
	BLA p	0	0	0	0	0	1	0	0	0	0	0	1	0	2	2	(PC_H)←p (PC_L)←(DR_2–DR_0, A_3–A_0) (Note)
		1	0	p_5	p_4	0	0	p_3	p_2	p_1	p_0	2	p	p			
Subroutine operation	BM a	0	1	0	a_6	a_5	a_4	a_3	a_2	a_1	a_0	1	a	a	1	1	(SP)←(SP)+1 (SK(SP))←(PC) (PC_H)←2, (PC_L)←a_6–a_0
	BML p, a	0	0	1	1	0	p_4	p_3	p_2	p_1	p_0	0	C	p+p	2	2	(SP)←(SP)+1 (SK(SP))←(PC) (PC_H)←p, (PC_L)←a_6–a_0 (Note)
		1	0	p_5	a_6	a_5	a_4	a_3	a_2	a_1	a_0	2	p	a+a			
	BMLA p	0	0	0	0	1	1	0	0	0	0	0	3	0	2	2	(SP)←(SP)+1 (SK(SP))←(PC)
		1	0	p_5	p_4	0	0	p_3	p_2	p_1	p_0	2	p	p			(PC_H)←p (PC_L)←(DR_2–DR_0, A_3–A_0) (Note)

Note : p is 0 to 63 for M34520M8A.
p is 0 to 47 for M34520M6A.

Skip condition	Carry flag CY	Detailed description
—	0/1	Adds the contents of M(DP) and carry flag CY to register A. Stores the result in register A and carry flag CY.
Overflow= 0	—	Adds the value n in the immediate field to register A. The contents of carry flag CY remains unchanged. Skips the next instruction when there is no overflow as the result of operation.
—	—	Takes the AND operation between the contents of register A and the contents of M(DP), and stores the result in register A.
—	—	Takes the OR operation between the contents of register A and the contents of M(DP), and stores the result in register A.
—	1	Sets (1) to carry flag CY.
—	0	Clears (0) to carry flag CY.
(CY)= 0	—	Skips the next instruction when the contents of carry flag CY is 0.
—	—	Stores the one's complement for register A's contents in register A.
—	0/1	Rotates 1 bit of the contents of register A including the contents of carry flag CY to the right.
—	—	Sets (1) to the contents of bit j (bit specified by the value j in the immediate field) of M(DP).
—	—	Clears (0) the contents of bit j (bit specified by the value j in the immediate field) of M(DP).
(Mj(DP))= 0 , j= 0 to 3	—	Skips the next instruction when the contents of bit j (bit specified by the value j in the immediate field) of M(DP) is "0".
(A)=(M(DP))	—	Skips the next instruction when the contents of register A is equal to the contents of M(DP).
(A)=n	—	Skips the next instruction when the contents of register A is equal to the value n in the immediate field.
—	—	Branch within a page : Branches to address a in the identical page.
—	—	Branch out of a page : Branches to address a in page p.
—	—	Branch out of a page : Branches to address (address DR_2 DR_1 DR_0 A_3 A_2 A_1 $A_0)_2$ specified by registers D and A in page p.
—	—	Call the subroutine in page 2 : Calls the subroutine at address a in page 2.
—	—	Call the subroutine : Calls the subroutine at address a in page p.
—	—	Call the subroutine : Calls the subroutine at address (DR_2 DR_1 DR_0 A_3 A_2 A_1 $A_0)_2$ specified by registers D and A in page p.

Type of instructions	Mnemonic	Instruction code D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation	Number of words	Number of cycles	Functions
Return operation	RTI	0	0	0	1	0	0	0	1	1	0	046	1	1	(PC)←(SK(SP)) (SP)←(SP)−1
	RT	0	0	0	1	0	0	0	1	0	0	044	1	2	(PC)←(SK(SP)) (SP)←(SP)−1
	RTS	0	0	0	1	0	0	0	1	0	1	045	1	2	(PC)←(SK(SP)) (SP)←(SP)−1
Interrupt operation	DI	0	0	0	0	0	0	0	1	0	0	004	1	1	(INTE)←0
	EI	0	0	0	0	0	0	0	1	0	1	005	1	1	(INTE)←1
	SNZ0	0	0	0	0	1	1	1	0	0	0	038	1	1	(EXF0)=1 ?, After skipping (EXF0)←0
	SNZ1	0	0	0	0	1	1	1	0	0	1	039	1	1	(EXF1)=1 ?, After skipping (EXF1)←0
	SNZI0	0	0	0	0	1	1	1	0	1	0	03A	1	1	$I1_2$=1 : (INT_0)="H" ? $I1_2$=0 : (INT_0)="L" ?
	SNZI1 (Note)	0	0	0	0	1	1	1	0	1	1	03B	1	1	$I2_2$=1 : (INT_1)="H" ? $I2_2$=0 : (INT_1)="L" ?
	TAV1	0	0	0	1	0	1	0	1	0	0	054	1	1	(A)←(V1)
	TV1A	0	0	0	0	1	1	1	1	1	1	03F	1	1	(V1)←(A)
	TAV2	0	0	0	1	0	1	0	1	0	1	055	1	1	(A)←(V2)
	TV2A	0	0	0	0	1	1	1	1	1	0	03E	1	1	(V2)←(A)
	TAI1	1	0	0	1	0	1	0	0	1	1	253	1	1	(A)←(I1)
	TI1A	1	0	0	0	0	1	0	1	1	1	217	1	1	(I1)←(A)
	TAI2	1	0	0	1	0	1	0	1	0	0	254	1	1	(A)←(I2)
	TI2A	1	0	0	0	0	1	1	0	0	0	218	1	1	(I2)←(A)
	TAI3	1	0	0	1	0	1	0	1	0	1	255	1	1	(A_0)←(I3), (A_3−A_1)←0
	TI3A	1	0	0	0	0	1	1	0	1	0	21A	1	1	(I3)←(A_0)

Note : OR operation between INT_1 pin and INT_2 pin when "1" is set to register I3.

Skip condition	Carry flag CY	Detailed description
—	—	Returns from interrupt service routine to main routine. Returns each value of data pointer (X, Y, Z), carry flag, skip status, NOP mode status by the continuous description of the LA/LXY instruction, registers A and B to the states just before interrupt.
—	—	Returns from subroutine to the routine called the subroutine.
Skip at uncondition	—	Returns from subroutine to the routine called the subroutine, and skips the next instruction at uncondition.
—	—	Clears (0) to the interrupt enable flag INTE, and disables the interrupt.
—	—	Sets (1) to the interrupt enable flag INTE, and enables the interrupt.
(EXF0)= 1	—	Skips the next instruction when the contents of EXF0 flag is "1". After skipping, clears the EXF0 flag.
(EXF1)=1	—	Skips the next instruction when the contents of EXF1 flag is "1". After skipping, clears EXF1 flag.
(INT_0)="H" $I1_2$=1	—	When bit 2 of register I1 ($I1_2$) is "1" : Skips the next instruction when the level of INT_0 pin is "H".
(INT_0)="L" $I1_2$=0	—	When bit 2 of register I1 ($I1_2$) is "0" : Skips the next instruction when the level of INT_0 pin is "L".
(INT_1)="H" $I2_2$=1	—	When bit 2 of register I2 ($I2_2$) is "1" : Skips the next instruction when the level of INT_1 pin is "H".
(INT_1)="L" $I2_2$=0	—	When bit 2 of register I2 ($I2_2$) is "0" : Skips the next instruction when the level of INT_1 pin is "L".
—	—	Transfers the contents of interrupt control register V1 to register A.
—	—	Transfers the contents of register A to interrupt control register V1.
—	—	Transfers the contents of interrupt control register V2 to register A.
—	—	Transfers the contents of register A to interrupt control register V2.
—	—	Transfers the contents of interrupt control register I1 to register A.
—	—	Transfers the contents of register A to interrupt control register I1.
—	—	Transfers the contents of interrupt control register I2 to register A.
—	—	Transfers the contents of register A to interrupt control register I2.
—	—	Transfers the contents of interrupt control register I3 to register A.
—	—	Transfers the contents of register A to interrupt control register I3.

Parameter / Type of instructions	Mnemonic	Instruction code										Hexadecimal notation			Number of words	Number of cycles	Functions
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0						
Timer operation	TAW1	1	0	0	1	0	0	1	0	1	1	2	4	B	1	1	(A)←(W1)
	TW1A	1	0	0	0	0	0	1	1	1	0	2	0	E	1	1	(W1)←(A)
	TAW2	1	0	0	1	0	0	1	1	0	0	2	4	C	1	1	(A)←(W2)
	TW2A	1	0	0	0	0	0	1	1	1	1	2	0	F	1	1	(W2)←(A)
	TAW3	1	0	0	1	0	0	1	1	0	1	2	4	D	1	1	(A)←(W3)
	TW3A	1	0	0	0	0	1	0	0	0	0	2	1	0	1	1	(W3)←(A)
	TAW4	1	0	0	1	0	0	1	1	1	0	2	4	E	1	1	(A)←(W4)
	TW4A	1	0	0	0	0	1	0	0	0	1	2	1	1	1	1	(W4)←(A)
	TAW5	1	0	0	1	0	0	1	1	1	1	2	4	F	1	1	(A_2-A_0)←$(W5_2-W5_0)$, (A_3)←0
	TW5A	1	0	0	0	0	1	0	0	1	0	2	1	2	1	1	$(W5_2-W5_0)$←(A_2-A_0)
	TAB1	1	0	0	1	1	1	0	0	0	0	2	7	0	1	1	(B)←$(T1_7-T1_4)$, (A)←$(T1_3-T1_0)$
	T1AB	1	0	0	0	1	1	0	0	0	0	2	3	0	1	1	$(R1_7-R1_4)$←(B), $(T1_7-T1_4)$←(B) $(R1_3-R1_0)$←(A), $(T1_3-T1_0)$←(A)
	TAB2	1	0	0	1	1	1	0	0	0	1	2	7	1	1	1	(B)←$(T2_7-T2_4)$, (A)←$(T2_3-T2_0)$
	T2AB	1	0	0	0	1	1	0	0	0	1	2	3	1	1	1	$(R2_7-R2_4)$←(B), $(T2_7-T2_4)$←(B) $(R2_3-R2_0)$←(A), $(T2_3-T2_0)$←(A)
	TAB4	1	0	0	1	1	1	0	0	1	1	2	7	3	1	1	(B)←$(T4_7-T4_4)$, (A)←$(T4_3-T4_0)$
	T4AB	1	0	0	0	1	1	0	0	1	1	2	3	3	1	1	$(R4_7-R4_4)$←(B), $(T4_7-T4_4)$←(B) $(R4_3-R4_0)$←(A), $(T4_3-T4_0)$←(A)
	T4ABD	1	0	0	0	1	1	0	1	1	0	2	3	6	1	1	$(T4_7-T4_4)$←(B), $(T4_3-T4_0)$←(A)
	TR4AB	1	0	0	0	1	1	1	1	0	0	2	3	C	1	1	$(R4_7-R4_4)$←(B), $(R4_3-R4_0)$←(A)
	SNZT1	1	0	1	0	0	0	0	0	0	0	2	8	0	1	1	(T1F)=1 ? After skipping, (T1F)←0
	SNZT2	1	0	1	0	0	0	0	0	0	1	2	8	1	1	1	(T2F)=1 ? After skipping, (T2F)←0
	SNZT3	1	0	1	0	0	0	0	0	1	0	2	8	2	1	1	(T3F)=1 ? After skipping, (T3F)←0
	SNZT4	1	0	1	0	0	0	0	0	1	1	2	8	3	1	1	(T4F)=1 ? After skipping, (T4F)←0

Skip condition	Carry flag CY	Detailed description
—	—	Transfers the contents of timer control register W1 to register A.
—	—	Transfers the contents of register A to timer control register W1.
—	—	Transfers the contents of timer control register W2 to register A.
—	—	Transfers the contents of register A to timer control register W2.
—	—	Transfers the contents of timer control register W3 to register A.
—	—	Transfers the contents of register A to timer control register W3.
—	—	Transfers the contents of timer control register W4 to register A.
—	—	Transfers the contents of register A to timer control register W4.
—	—	Transfers the contents of timer control register W5 to register A.
—	—	Transfers the contents of register A to timer control register W5.
—	—	Transfers the contents of timer 1 to registers A and B.
—	—	Transfers the contents of registers A and B to timer 1 and timer 1 reload register.
—	—	Transfers the contents of timer 2 to registers A and B.
—	—	Transfers the contents of registers A and B to timer 2 and timer 2 reload register.
—	—	Transfers the contents of timer 4 to registers A and B.
—	—	Transfers the contents of registers A and B to timer 4 and timer 4 reload register.
—	—	Transfers the contents of registers A and B to timer 4.
—	—	Transfers the contents of registers A and B to timer 4 reload register.
(T1F)=1	—	Skips the next instruction when the contents of T1F flag is "1". After skipping, clears the T1F flag.
(T2F)=1	—	Skips the next instruction when the contents of T2F flag is "1". After skipping, clears T2F flag.
(T3F)=1	—	Skips the next instruction when the contents of T3F flag is "1". After skipping, clears the T3F flag.
(T4F)=1	—	Skips the next instruction when the contents of T4F flag is "1". After skipping, clears T4F flag.

Type of instructions \ Parameter	Mnemonic	Instruction code										Hexadecimal notation			Number of words	Number of cycles	Functions
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0						
Input/Output operation	IAP0	1	0	0	1	1	0	0	0	0	0	2	6	0	1	1	(A)←(P0)
	OP0A	1	0	0	0	1	0	0	0	0	0	2	2	0	1	1	(P0)←(A)
	IAP1	1	0	0	1	1	0	0	0	0	1	2	6	1	1	1	(A)←(P1)
	OP1A	1	0	0	0	1	0	0	0	0	1	2	2	1	1	1	(P1)←(A)
	IAP2	1	0	0	1	1	0	0	0	1	0	2	6	2	1	1	(A)←(P2)
	IAP3	1	0	0	1	1	0	0	0	1	1	2	6	3	1	1	(A)←(P3)
	IAP4	1	0	0	1	1	0	0	1	0	0	2	6	4	1	1	(A)←(P4)
	CLD	0	0	0	0	0	1	0	0	0	1	0	1	1	1	1	(D)←1
	RD	0	0	0	0	0	1	0	1	0	0	0	1	4	1	1	(D(Y))←0 (Y)=0 to 10
	SD	0	0	0	0	0	1	0	1	0	1	0	1	5	1	1	(D(Y))←1 (Y)=0 to 10
	SZD	0	0	0	0	1	0	0	1	0	0	0	2	4	2	2	(D(Y))=0 ? (Y)=0 to 10
		0	0	0	0	1	0	1	0	1	1	0	2	B			
	TK0A	1	0	0	0	0	1	1	0	1	1	2	1	B	1	1	(K0)←(A)
	TAK0	1	0	0	1	0	1	0	1	1	0	2	5	6	1	1	(A)←(K0)
	TPU0A	1	0	0	0	1	0	1	1	0	1	2	2	D	1	1	(PU0)←(A)
LCD control operation	TL1A	1	0	0	0	0	0	1	0	1	0	2	0	A	1	1	(L1)←(A)
	TAL1	1	0	0	1	0	0	1	0	1	0	2	4	A	1	1	(A)←(L1)
	TL2A	1	0	0	0	0	0	1	0	1	1	2	0	B	1	1	(L2)←(A)
	TL3A	1	0	0	0	0	0	1	1	0	0	2	0	C	1	1	(L3)←(A_0)
	TLCA	1	0	0	0	0	0	1	1	0	1	2	0	D	1	1	(LC)←(A)
Serial I/O operation	TAH	1	0	0	1	0	0	0	0	0	0	2	4	0	1	1	(A)←(H)
	THA	1	0	0	0	0	0	0	0	0	0	2	0	0	1	1	(H)←(A)
	TAL	1	0	0	1	0	0	0	0	0	1	2	4	1	1	1	(A)←(L)
	TLA	1	0	0	0	0	0	0	0	0	1	2	0	1	1	1	(L)←(A)
	TAJ1	1	0	0	1	0	0	0	0	1	0	2	4	2	1	1	(A_2—A_0)←($J1_2$—$J1_0$), (A_3)←0
	TJ1A	1	0	0	0	0	0	0	0	1	0	2	0	2	1	1	($J1_2$—$J1_0$)←(A_2—A_0)
	TAJ2	1	0	0	1	0	0	0	0	1	1	2	4	3	1	1	(A_1, A_0)←($J2_1$, $J2_0$), (A_3, A_2)←0
	TJ2A	1	0	0	0	0	0	0	0	1	1	2	0	3	1	1	($J2_1$, $J2_0$)←(A_1, A_0)
	SST	1	0	1	0	0	1	1	1	1	0	2	9	E	1	1	(SIOF)←0, serial I/O starts
	SNZSI	1	0	1	0	0	0	1	0	0	0	2	8	8	1	1	(SIOF)=1 ?, After skipping, (SIOF)←0

Skip condition	Carry flag CY	Detailed description
—	—	Transfers input of port P0 to register A.
—	—	Outputs the contents of register A to port P0.
—	—	Transfers input of port P1 to register A.
—	—	Outputs the contents of register A to port P1.
—	—	Transfers input of port P2 to register A.
—	—	Transfers input of port P3 to register A.
—	—	Transfers input of port P4 to register A.
—	—	Sets (1) to port D.
—	—	Clears (0) to a bit of port D specified by register Y.
—	—	Sets (1) to a bit of port D specified by register Y.
(D(Y))= 0 (Y)= 0 to 10	—	Skips the next instruction when a bit of port D specified by register Y is "0".
—	—	Transfers the contents of register A to key-on wakeup control register K0.
—	—	Transfers the contents of key-on wakeup control register K0 to register A.
—	—	Transfers the contents of register A to pull-up control register PU0.
—	—	Transfers the contents of register A to LCD control register L1.
—	—	Transfers the contents of LCD control register L1 to register A.
—	—	Transfers the contents of register A to LCD control register L2.
—	—	Transfers the contents of register A to LCD control register L3.
—	—	Transfers the contents of register A to timer LC and reload register.
—	—	Transfers the contents of register H to register A.
—	—	Transfers the contents of register A to register H.
—	—	Transfers the contents of register L to register A.
—	—	Transfers the contents of register A to register L.
—	—	Transfers the contents of serial I/O mode register J1 to register A.
—	—	Transfers the contents of register A to serial I/O mode register J1.
—	—	Transfers the contents of serial I/O mode register J2 to register A.
—	—	Transfers the contents of register A to serial I/O mode register J2.
—	—	Clears (0) to SIOF flag and starts serial I/O.
(SIOF)= 1	—	Skips the next instruction when SIOF flag is "1". After skipping, clears (0) to SIOF flag.

Parameter / Type of instructions	Mnemonic	Instruction code										Hexadecimal notation			Number of words	Number of cycles	Functions
		D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0						
A-D conversion operation	TAHA	1	0	0	1	0	0	1	0	0	0	2	4	8	1	1	(A)←(HA)
	TALA	1	0	0	1	0	0	1	0	0	1	2	4	9	1	1	(A)←(LA)
	TAQ1	1	0	0	1	0	0	0	1	0	0	2	4	4	1	1	$(A_2-A_0)\leftarrow(Q1_2-Q1_0)$, $(A_3)\leftarrow 0$
	TQ1A	1	0	0	0	0	0	0	1	0	0	2	0	4	1	1	$(Q1_2-Q1_0)\leftarrow(A_2-A_0)$
	ADST	1	0	1	0	0	1	1	1	1	1	2	9	F	1	1	A-D conversion starts
	SNZAD	1	0	1	0	0	0	0	1	1	1	2	8	7	1	1	(ADF)=1 ?, After skipping, (ADF)←0
Other operation	NOP	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	(PC)←(PC)+1
	POF	0	0	0	0	0	0	0	0	1	0	0	0	2	1	1	Power down 1
	POF2	0	0	0	0	0	0	1	0	0	0	0	0	8	1	1	Power down 2
	SNZP	0	0	0	0	0	0	0	0	1	1	0	0	3	1	1	(P)=1 ?
	TAMR	1	0	0	1	0	1	0	0	1	0	2	5	2	1	1	$(A_2-A_0)\leftarrow(MR_2-MR_0)$, $(A_3)\leftarrow 0$
	TMRA	1	0	0	0	0	1	0	1	1	0	2	1	6	1	1	$(MR_2-MR_0)\leftarrow(A_2-A_0)$
	WRST	1	0	1	0	1	0	0	0	0	0	2	A	0	1	1	(WDF)←0
	RTPS	1	0	1	0	1	0	0	0	0	1	2	A	1	1	1	$(RTPL_1)\leftarrow 1$, $(RTPL_0)\leftarrow 1$
	RTPR	1	0	1	0	1	0	0	0	1	0	2	A	2	1	1	$(RTPL_1)\leftarrow 0$, $(RTPL_0)\leftarrow 0$
	TRTPA	1	0	0	0	0	1	1	0	0	1	2	1	9	1	1	(RTP)←(A)

Skip condition	Carry flag CY	Detailed description
—	—	Transfers the contents of register HA to register A.
—	—	Transfers the contents of register LA to register A.
—	—	Transfers the contents of A-D control register Q1 to register A.
—	—	Transfers the contents of register A to A-D control register Q1.
—	—	Starts A-D conversion.
(ADF)= 1	—	Skips the next instruction when ADF flag is "1". After skipping, clears (0) to ADF flag.
—	—	No operation
—	—	Puts the system in power down 1 state (clock operating mode). f(X_{CIN}) oscillation, LCD, timer 3 and timer 4 can be used.
—	—	Puts the system in power down 2 state (RAM back-up mode). Oscillations are all stopped.
(P)= 1	—	Skips the next instruction when the contents of P flag is "1". After skipping, the contents of P flag remains unchanged.
—	—	Transfers the contents of clock control register MR to register A.
—	—	Transfers the contents of register A to clock control register MR.
—	—	Clears watchdog timer flag WDF.
—	—	Sets (1) to both real time output latch $RTPL_1$ and $RTPL_0$.
—	—	Clears (0) to both real time output latch $RTPL_1$ and $RTPL_0$.
—	—	Transfers the contents of register A to real time output register RTP.

SYMBOL

The symbols shown are used in the following instruction function table and instruction list.

Symbol	Contents	Symbol	Contents
A	Register A (4 bits)	T1	Timer 1
B	Register B (4 bits)	T2	Timer 2
DR	Register D (3 bits)	T3	Timer 3
E	Register E (8 bits)	T4	Timer 4
Q1	A-D control register Q1 (3 bits)	LC	Timer LC
HA	Successive comparison register HA (4 bits)	T1F	Timer 1 interrupt request flag
LA	Successive comparison register LA (4 bits)	T2F	Timer 2 interrupt request flag
J1	Serial I/O mode register J1 (3 bits)	T3F	Timer 3 interrupt request flag
J2	Serial I/O mode register J2 (2 bits)	T4F	Timer 4 interrupt request flag
H	Serial I/O register H (4 bits)	WDF	Watchdog timer flag
L	Serial I/O register L (4 bits)	INTE	Interrupt enable flag
L1	LCD control register L1 (4 bits)	EXF0	External 0 interrupt request flag
L2	LCD control register L2 (4 bits)	EXF1	External 1 interrupt request flag
L3	LCD control register L3 (1 bit)	ZCF	Zero cross input flag
MR	Clock control register MR (3 bits)	NCF	Noise canceller flag
V1	Interrupt control register V1 (4 bits)	P	Power down flag
V2	Interrupt control register V2 (4 bits)	ADF	A-D conversion completion flag
I1	Interrupt control register I1 (4 bits)	SIOF	Serial I/O transmission/reception completion flag
I2	Interrupt control register I2 (4 bits)		
I3	Interrupt control register I3 (1 bit)	D	Port D (11 bits)
W1	Timer control register W1 (4 bits)	P0	Port P0 (4 bits)
W2	Timer control register W2 (4 bits)	P1	Port P1 (4 bits)
W3	Timer control register W3 (4 bits)	P2	Port P2 (4 bits)
W4	Timer control register W4 (4 bits)	P3	Port P3 (4 bits)
W5	Timer control register W5 (3 bits)	P4	Port P4 (4 bits)
RTP	Real time output register RTP (2 bits)		
K0	Key-on wakeup control register (4 bits)	x	Hexadecimal variable
PU0	Pull-up control register PU0 (4 bits)	y	Hexadecimal variable
		z	Hexadecimal variable
X	Register X (4 bits)	p	Hexadecimal variable
Y	Register Y (4 bits)	n	Hexadecimal constant
Z	Register Z (2 bits)	i	Hexadecimal constant
DP	Data pointer (10 bits)	j	Hexadecimal constant
	(It consists of registers X, Y and Z)	$A_3A_2A_1A_0$	Binary notation of hexadecimal variable A
PC	Program counter (14 bits)		(same for others)
PC_H	High-order 7 bits of program counter		
PC_L	Low-order 7 bits of program counter	←	Direction of data movement
SK	Stack register (14 bits×8)	( )	Contents of registers and memories
SP	Stack pointer (3 bits)	−	Negate, Flag unchanged after executing instruction
CY	Carry flag	M(DP)	RAM address specified by the data pointer
R1	Timer 1 reload register	a	Label indicating address $a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0$
R2	Timer 2 reload register	p, a	Label indicating address $a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0$ in page $p_5\ p_4\ p_3\ p_2\ p_1\ p_0$
R4	Timer 4 reload register		
		C	Hex. C + Hex. number X (also same for others)
		+	
		×	

Note : The 4520 Group just invalidates the next instruction when a skip is performed. The contents of program counter is not increased by 2. Accordingly, the number of cycles does not change even if a skip is not performed. However, the cycle count becomes "1" when the TABP $_p$, RT, or RTS instruction is skipped.

CONTROL REGISTERS

Interrupt control register V1		at reset : 0000_2		at power down : 0000_2	R/W
$V1_3$	Timer 2 interrupt enable bit	0	Interrupt disabled (SNZT2 instruction is valid)		
		1	Interrupt enabled (SNZT2 instruction is invalid)		
$V1_2$	Timer 1 interrupt enable bit	0	Interrupt disabled (SNZT1 instruction is valid)		
		1	Interrupt enabled (SNZT1 instruction is invalid)		
$V1_1$	External 1 interrupt enable bit	0	Interrupt disabled (SNZ1 instruction is valid)		
		1	Interrupt enabled (SNZ1 instruction is invalid)		
$V1_0$	External 0 interrupt enable bit	0	Interrupt disabled (SNZ0 instruction is valid)		
		1	Interrupt enabled (SNZ0 instruction is invalid)		
Interrupt control register V2		at reset : 0000_2		at power down : 0000_2	R/W
$V2_3$	Serial I/O interrupt enable bit	0	Interrupt disabled (SNZSI instruction is valid)		
		1	Interrupt enabled (SNZSI instruction is invalid)		
$V2_2$	A-D interrupt enable bit	0	Interrupt disabled (SNZAD instruction is valid)		
		1	Interrupt enabled (SNZAD instruction is invalid)		
$V2_1$	Timer 4 interrupt enable bit	0	Interrupt disabled (SNZT4 instruction is valid)		
		1	Interrupt enabled (SNZT4 instruction is invalid)		
$V2_0$	Timer 3 interrupt enable bit	0	Interrupt disabled (SNZT3 instruction is valid)		
		1	Interrupt enabled (SNZT3 instruction is invalid)		
Clock control register MR (Note 2)		at reset : 000_2		at power down : state retained	R/W
MR_2	Not used	0	This bit has no function, but read/write is enabled		
		1			
MR_1	$f(X_{IN})$ clock oscillation circuit control bit	0	Oscillation enabled		
		1	Oscillation stop		
MR_0	System clock selection bit	0	$f(X_{IN})$		
		1	$f(X_{CIN})$		
Interrupt control register I1		at reset : 0000_2		at power down : state retained	R/W
$I1_3$	D_8/INT_0/ZEROX pin function selection bit	0	D_8/INT_0		
		1	ZEROX		
$I1_2$	INT_0 pin interrupt valid edge/ return level selection bit	0	Falling edge (SNZI0 instruction recognizes that level of INT_0 pin is "L" level)/"L" level		
		1	Rising edge (SNZI0 instruction recognizes that level of INT_0 pin is "H" level)/"H" level		
$I1_1$	Edge detection circuit control bit	0	One-sided edge detected		
		1	Both edges detected		
$I1_0$	Noise detection circuit control bit	0	Disabled		
		1	Used		
Interrupt control register I2		at reset : 0000_2		at power down : state retained	R/W
$I2_3$	Not used	0	This bit has no function, but read/write is enabled.		
		1			
$I2_2$	Pins INT_1 and INT_2 interrupt valid edge selection bit	0	Falling edge (SNZI1 instruction recognizes that level of INT_1 pin is "L" level)		
		1	Rising edge (SNZI1 instruction recognizes that level of INT_1 pin is "H" level)		
$I2_1$	Noise detection circuit sampling clock control bit	0	Stop		
		1	Operating		
$I2_0$	Noise detection circuit sampling clock selection bit	0	$f(X_{IN})$ or $f(X_{CIN})$ divided by 24 (Note 2)		
		1	$f(X_{IN})$ or $f(X_{CIN})$ divided by 96 (Note 2)		
Interrupt control register I3 (Note 3)		at reset : 0_2		at power down : state retained	R/W
$I3_0$	External 2 interrupt enable bit	0	Disabled		
		1	Enabled (Note 4)		

Notes 1. "R" represents read enabled, and "W" represents write enabled.
2. After executing the TAMR instruction, "0" is stored in the bit 3 of register A.
3. After executing the TAI3 instruction, "0" is stored in bits 3, 2 and 1 of register A.
4. External interrupt activated condition is OR operation between INT_1 and INT_2.

Timer control register W1		at reset : 0000_2		at power down : 0000_2	R/W
$W1_3$	Prescaler control bit	0	Stop (reset state)		
		1	Operating		
$W1_2$	Prescaler dividing ratio selection bit	0	$f(X_{IN})$ or $f(X_{CIN})$ divided by 6 (Note 2)		
		1	$f(X_{IN})$ or $f(X_{CIN})$ divided by 12 (Note 2)		
$W1_1$	Timer 1 control bit	0	Stop (state retained)		
		1	Operating		
$W1_0$	$D_6/CNTR_0$ pin function selection bit	0	I/O port D_6		
		1	$CNTR_0$ output		

Timer control register W2		at reset : 0000_2		at power down : state retained	R/W
$W2_3$	$D_7/CNTR_1$ pin function selection bit	0	I/O port $D_7/CNTR_1$ input		
		1	$CNTR_1$ output		
$W2_2$	Timer 2 count start trigger selection bit	0	Timer 2 control bit (bit 0 of register W5)		
		1	Timer 2 control bit and zero cross input flag (ZCF)		
$W2_1$	Timer 2 count source selection bits	$W2_1$	$W2_0$	Count source	
		0	0	$f(X_{IN})$	
$W2_0$		0	1	Timer 1 underflow signal	
		1	0	Prescaler output (ORCLK)	
		1	1	$CNTR_1$ input	

Timer control register W3		at reset : 0000_2		at power down : state retained	R/W
$W3_3$	LCD frequency dividing circuit count source control bit	0	Stop		
		1	Operating		
$W3_2$	LCD frequency dividing circuit count source selection bit	0	Timer 3 intermediate underflow signal (Frequency divided by 16 output)		
		1	Timer 3 underflow signal (Frequency divided by 256 output)		
$W3_1$	Timer 3 control bit	0	Stop (reset state)		
		1	Operating		
$W3_0$	Timer 3 count source selection bit	0	Timer 2 underflow signal		
		1	$f(X_{CIN})$		

Timer control register W4		at reset : 0000_2		at power down : state retained	R/W
$W4_3$	Timer 2 and 4 function selection bit	0	Normal mode		
		1	PWM mode		
$W4_2$	Watchdog timer control bit	0	Watchdog timer stop		
		1	Watchdog timer operating		
$W4_1$	Timer 4 count source selection bits	$W4_1$	$W4_0$	Count source	
		0	0	$f(X_{IN})$	
$W4_0$		0	1	Timer 3 underflow signal	
		1	0	Timer 2 underflow signal	
		1	1	Prescaler output (ORCLK)	

Timer control register W5 (Note 3)		at reset : 000_2		at power down : state retained	R/W
$W5_2$	Zero cross input flag control bit	0	Stop		
		1	Operating		
$W5_1$	Timer 4 control bit	0	Stop (state retained)		
		1	Operating		
$W5_0$	Timer 2 control bit	0	Stop (state retained)		
		1	Operating		

Notes 1. "R" represents read enabled, and "W" represents write enabled.
2. The signal in $f(X_{IN})$ or $f(X_{CIN})$ selected as a system clock is used.
3. After executing the TAW5 instruction, "0" is stored in bit 3 of register A.

Serial I/O mode register J1 (Note 2)		at reset : 000_2		at power down : state retained	R/W
$J1_2$	$D_9/S_{CK}/RTP_0$ pin function selection bits	$J1_2$	$J1_1$	Function	
		0	0	I/O port D_9	
$J1_1$		0	1	Serial I/O clock input/output S_{CK}	
		1	0	Real time output RTP_0	
		1	1	Not available	
$J1_0$	Serial I/O synchronous clock selection bit	0	External clock		
		1	$f(X_{IN})$ or $f(X_{CIN})$ divided by 24		

Serial I/O mode register J2 (Note 3)		at reset : 00_2		at power down : state retained	R/W
$J2_1$	$D_{10}/S_{OUT}/RTP_1/PWM$ pin function selection bits	$J2_1$	$J2_0$	Function	
		0	0	I/O port D_{10}	
$J2_0$		0	1	Serial data output S_{OUT}	
		1	0	Real time output RTP_1	
		1	1	PWM output PWM	

LCD control register L1		at reset : 0000_2		at power down : state retained	R/W
$L1_3$	LCD power supply dividing resistor control bit	0	Connecting internal dividing resistor to LCD power circuit		
		1	Disconnecting internal dividing resistor from LCD power circuit		
$L1_2$	LCD ON/OFF bit	0	OFF		
		1	ON		
$L1_1$	LCD duty and bias selection bits	$L1_1$	$L1_0$	Duty	Bias
		0	0	Not available	
$L1_0$		0	1	1/2	1/2
		1	0	1/3	1/3
		1	1	1/4	1/3

LCD control register L2 (Note 4)		at reset : 0000_2		at power down : state retained	W
$L2_3$	$SEG_{19}/P4_3/A_{IN7}-SEG_{22}/P4_0/A_{IN4}$ pin function switch bit	0	$SEG_{19}-SEG_{22}$		
		1	$P4_3/A_{IN7}-P4_0/A_{IN4}$		
$L2_2$	$SEG_{23}/P3_3/A_{IN3}$, $SEG_{24}/P3_2/A_{IN2}$ pin function switch bit	0	SEG_{23}, SEG_{24}		
		1	$P3_3/A_{IN3}$, $P3_2/A_{IN2}$		
$L2_1$	$SEG_{25}/P3_1/A_{IN1}$ pin function switch bit	0	SEG_{25}		
		1	$P3_1/A_{IN1}$		
$L2_0$	$SEG_{26}/P3_0/INT_2/A_{IN0}$ pin function switch bit	0	SEG_{26}		
		1	$P3_0/INT_2/A_{IN0}$		

LCD control register L3		at reset : 0_2		at power down : state retained	W
$L3_0$	SEG_{17}/V_{LC1}, SEG_{18}/V_{LC2} pin function switch bit	0	SEG_{17}, SEG_{18}		
		1	V_{LC1}, V_{LC2}		

Notes 1. "R" represents read enabled, and "W" represents write enabled.
2. After executing the TAJ1 instruction, "0" is stored in bit 3 of register A.
3. After executing the TAJ2 instruction, "0" is stored in bits 3 and 2 of register A.
4. After setting the register L2, select $A_{IN0}-A_{IN7}$ with register Q1.

A-D control register Q1 (Note 4)		at reset : 000_2		at power down : state retained	R/W
$Q1_2$	Analog input pin selection bits	$Q1_2$ $Q1_1$ $Q1_0$	Selective pin		
		0 0 0	A_{IN0}		
$Q1_1$		0 0 1	A_{IN1}		
		0 1 0	A_{IN2}		
		0 1 1	A_{IN3}		
		1 0 0	A_{IN4}		
$Q1_0$		1 0 1	A_{IN5}		
		1 1 0	A_{IN6}		
		1 1 1	A_{IN7}		
Key-on wakeup control register K0		at reset : 0000_2		at power down : state retained	R/W
$K0_3$	Pins $P1_2$ and $P1_3$ key-on wakeup control bit	0	Key-on wakeup not used		
		1	Key-on wakeup used		
$K0_2$	Pins $P1_0$ and $P1_1$ key-on wakeup control bit	0	Key-on wakeup not used		
		1	Key-on wakeup used		
$K0_1$	Pins $P0_2$ and $P0_3$ key-on wakeup control bit	0	Key-on wakeup not used		
		1	Key-on wakeup used		
$K0_0$	Pins $P0_0$ and $P0_1$ key-on wakeup control bit	0	Key-on wakeup not used		
		1	Key-on wakeup used		
Pull-up control register PU0		at reset : 0000_2		at power down : state retained	W
$PU0_3$	Pins $P1_2$ and $P1_3$ pull-up transistor control bit	0	Pull-up transistor OFF		
		1	Pull-up transistor ON		
$PU0_2$	Pins $P1_0$ and $P1_1$ pull-up transistor control bit	0	Pull-up transistor OFF		
		1	Pull-up transistor ON		
$PU0_1$	Pins $P0_2$ and $P0_3$ pull-up transistor control bit	0	Pull-up transistor OFF		
		1	Pull-up transistor ON		
$PU0_0$	Pins $P0_0$ and $P0_1$ pull-up transistor control bit	0	Pull-up transistor OFF		
		1	Pull-up transistor ON		

Notes 1. "R" represents read enabled, and "W" represents write enabled.
2. After executing the TAQ1 instruction, "0" is stored in bit 3 of register A.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions		Rating	Unit
V_{DD}	Supply voltage			−0.3 to 7	V
V_I	Input voltage P0, P1, P2, P3, P4, $\overline{RESET}$, X_{IN}, X_{CIN}, V_{REF}			−0.3 to V_{DD}+0.3	V
V_I	Input voltage D_0−D_7, D_9, D_{10}			−0.3 to 13	V
V_I	Input voltage V_{LC1}, V_{LC2}, V_{LC3}			−0.3 to V_{DD}+0.3	V
V_I	Input voltage D_8/INT_0/ZEROX			−0.7 to V_{DD}+0.7	V
I_I	Input current D_8/INT_0/ZEROX			−100 to 100	μA
V_O	Output voltage P0, P1, $\overline{RESET}$	Output transistors in the cut-off state		−0.3 to V_{DD}+0.3	V
V_O	Output voltage D_8	Output transistors in the cut-off state		−0.7 to V_{DD}+0.7	V
V_O	Output voltage D_0−D_7, D_9, D_{10}	Output transistors in the cut-off state		−0.3 to 13	V
V_O	Output voltage X_{OUT}, X_{COUT}			−0.3 to V_{DD}+0.3	V
V_O	Output voltage SEG, COM			−0.3 to V_{LC3}+0.3	V
P_d	Power dissipation	T_a=25℃	M34520MxA-XXXSP	1100	mW
			M34520MxA-XXXFP	300	
T_{opr}	Operating temperature range (Note 1)			−20 to 85	℃
T_{stg}	Storage temperature range			−40 to 125	℃

Note1. Operating temperature range for built-in EPROM version (M34520E8FS, M34520E8SS) is −20℃ to 70℃.

RECOMMENDED OPERATING CONDITIONS (T_a=−20℃ to 85℃, V_{DD}=2.2V to 5.5V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{DD}	Supply voltage (Note 1)	f(X_{IN})=4.0MHz	4.5	5.0	5.5	V
		f(X_{IN})=1.5MHz	2.2		5.5	
V_{RAM}	RAM back-up voltage (at RAM back-up mode)		2.0		5.5	V
V_{SS}	Supply voltage			0		V
AV_{SS}	Analog supply voltage (Note 2)			V_{SS}		V
V_{LC3}	Supply voltage for LCD (Note 3)		2.2		5.5	V
V_{IH}	"H" level input voltage P0, P1, P2, P3, P4, D_8		0.7V_{DD}		V_{DD}	V
V_{IH}	"H" level input voltage D_0−D_7, D_9, D_{10}		0.7V_{DD}		12	V
V_{IH}	"H" level input voltage X_{IN}		0.7V_{DD}		V_{DD}	V
V_{IH}	"H" level input voltage $\overline{RESET}$		0.85V_{DD}		V_{DD}	V
V_{IH}	"H" level input voltage $CNTR_1$, S_{IN}, S_{CK}, INT_0, INT_1, INT_2		0.8V_{DD}		V_{DD}	V
V_{IL}	"L" level input voltage P0, P1, P2, P3, P4, D_0−D_{10}		0		0.3V_{DD}	V
V_{IL}	"L" level input voltage X_{IN}		0		0.3V_{DD}	V
V_{IL}	"L" level input voltage $\overline{RESET}$		0		0.1V_{DD}	V
V_{IL}	"L" level input voltage $CNTR_1$, S_{IN}, S_{CK}, INT_0, INT_1, INT_2		0		0.2V_{DD}	V
$I_{OL(peak)}$	"L" level peak output current P0, P1, $\overline{RESET}$	V_{DD}=4.5V to 5.5V			10	mA
$I_{OL(peak)}$	"L" level peak output current D_4, D_5				40	mA
$I_{OL(peak)}$	"L" level peak output current D_0−D_3, D_6−D_{10}				24	mA
$I_{OL(avg)}$	"L" level average output current P0, P1, $\overline{RESET}$ (Note 4)				5	mA
$I_{OL(avg)}$	"L" level average output current D_4, D_5 (Note 4)				20	mA
$I_{OL(avg)}$	"L" level average output current D_0−D_3, D_6−D_{10} (Note 4)				12	mA
f(X_{IN})	f(X_{IN}) clock frequency (with a ceramic resonator) (Note 5)	V_{DD}=4.5V to 5.5V			4.0	MHz
		V_{DD}=2.2V to 5.5V			1.5	
f(X_{IN})	f(X_{IN}) clock frequency (Note 5) (with external clock input) (Note 6)	V_{DD}=4.5V to 5.5V			3.0	MHz
		V_{DD}=2.2V to 5.5V			0.8	
f(X_{CIN})	f(X_{CIN}) clock frequency (with a quartz-crystal oscillator) (Note 7)	V_{DD}=2.2V to 5.5V			50	kHz
tw(S_{CK})	Serial I/O external clock cycle ("H" level or "L" level pulse width)	V_{DD}=4.5V to 5.5V	750			ns
		V_{DD}=2.2V to 5.5V	2.0			μs
tw(CNTR1)	Timer external input cycle ("H" level or "L" level pulse width)	V_{DD}=4.5V to 5.5V	750			ns
		V_{DD}=2.2V to 5.5V	2.0			μs

Notes 1. Supply voltage at Mask ROM version is 2.2V to 5.5V.
Supply voltage at built-in PROM version is 2.5V to 5.5V.
- When using zero cross detection circuit
  V_{DD}=3.0V to 5.5V (Mask ROM version)
  V_{DD}=4.0V to 5.5V (Built-in PROM version)
- When using A-D converter
  V_{DD}=2.7V to 5.5V (Mask ROM version, built-in PROM version)

2. Use AV_{SS} and V_{SS} at the same voltage level.
3. When using 1/2 bias : V_{LC1}=V_{LC2}=1/2·V_{LC3}
When using 1/3 bias : V_{LC1}=1/3·V_{LC3}, V_{LC2}=2/3·V_{LC3}
4. Keep the total currents of I_{OL} (avg) for ports P0, P1, D_0−D_{10}, and $\overline{RESET}$ to 80mA or less.
5. However, the minimum value for the system clock frequency when using an A-D converter is 400kHz.
6. Keep the duty ratio of the external clock within the range shown in Table 19.

Table 19 Supply voltage and duty ratio of external clock

Supply voltage	Duty ratio of external clock
4.5V to 5.5V	40% to 60%
2.2V to 5.5V	30% to 70%

7. External clock cannot be used as f(X_{CIN}) clock.

ELECTRICAL CHARACTERISTICS (T_a=−20℃ to 85℃, V_{DD}=3.0V, unless otherwise noted)

Symbol	Parameter		Test condition	Limits			Unit
				Min.	Typ.	Max.	
V_{OL}	"L" level output voltage P0, P1, $\overline{RESET}$		I_{OL}=2mA			0.9	V
V_{OL}	"L" level output voltage D_4, D_5		I_{OL}=8mA			1.5	V
V_{OL}	"L" level output voltage D_0—D_3, D_6, D_7—D_{10}		I_{OL}=2mA			0.9	V
I_{IH}	"H" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$		V_I=V_{DD} (Note 1)			1	μA
I_{IH}	"H" level input current D_0—D_7, D_9, D_{10}		V_I=12V			1	μA
I_{IH}	"H" level input current V_{REF}		V_I=3V			1	mA
I_{IL}	"L" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$		V_I=0V (Note 1) (Note 2)	−1			μA
I_{IL}	"L" level input current D_0—D_7, D_9, D_{10}		V_I=0V	−1			μA
I_{OZ}	Output current at off-state D_0—D_7, D_9, D_{10}		V_O=12V			1	μA
I_{OZ}	Output current at off-state P0, P1, D_8, $\overline{RESET}$		V_O=V_{DD}			1	μA
I_{DD}	Supply current	at active high-speed mode	f(X_{IN})=1.5MHz f(X_{CIN})=32kHz		0.7	2.1	mA
			f(X_{IN})=500kHz f(X_{CIN})=32kHz		0.3	0.9	mA
		at active low-speed mode	f(X_{IN})=Stop f(X_{CIN})=32kHz		20	60	μA
		at power down 1 mode (LCD operation)	f(X_{IN})=Stop f(X_{CIN})=32kHz		7	21	μA
		at power down 2 mode	f(X_{IN})=Stop f(X_{CIN})=Stop T_a=25℃		0.1	1.0	μA
			f(X_{IN})=Stop f(X_{CIN})=Stop			10	μA
IA_{DD}	A-D operation current		f(X_{IN})=1.5MHz		0.2	0.6	mA
IZ_{DD}	Zero cross comparator operation current		f(X_{IN})=1.5MHz		0.8	2.4	mA
R_{PU}	Pull-up resistor value P0 and P1		V_{DD}=3V V_I=0V	40	100	250	kΩ
V_{T+}−V_{T-}	Hysteresis INT_0, INT_1, INT_2				0.3		V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$				0.7		V
R_{COM}	COM output impedance		(Note 3)		2	10	kΩ
R_{SEG}	SEG output impedance		(Note 3)		3	15	kΩ
R_{VLC}	Internal resistor value for LCD power (impedance between V_{LC3} and V_{SS})		T_a=25℃	300	600	1200	kΩ
—	Zero cross comparator accuracy					±0.04V_{DD}	V

Notes 1. In this case, port P4 function is selected with software.
2. The pull-up resistors of ports P0 and P1 are disconnected.
3. Use an external power supply for the LCD power, and leave all pins open except the measured pin.

ELECTRICAL CHARACTERISTICS (T_a=−20℃ to 85℃, V_{DD}=4.5V to 5.5V, unless otherwise noted)

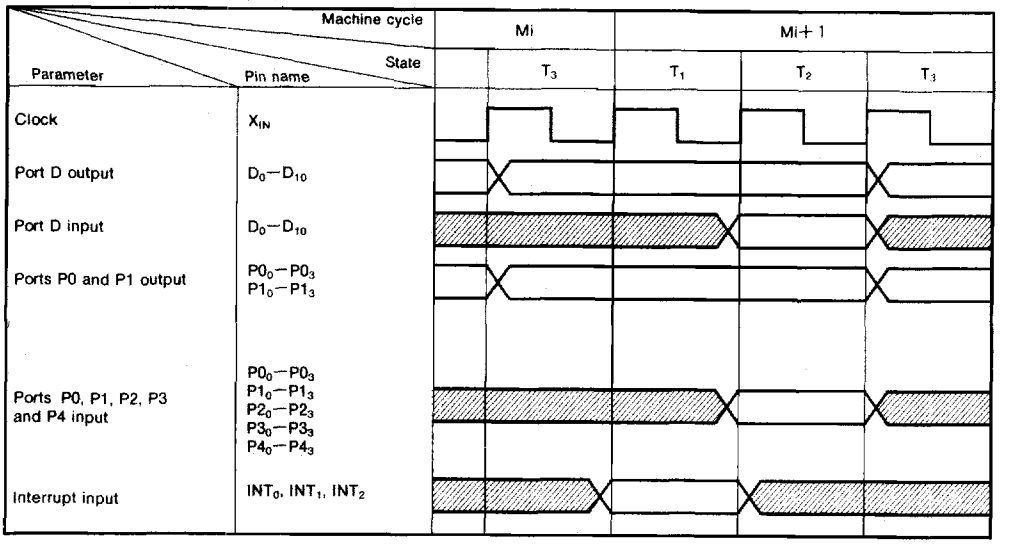
Symbol	Parameter		Test condition	Limits			Unit
				Min.	Typ.	Max.	
V_{OL}	"L" level output voltage P0, P1, $\overline{RESET}$		I_{OL}=5mA			0.5	V
V_{OL}	"L" level output voltage D_4, D_5		I_{OL}=20mA			2	V
V_{OL}	"L" level output voltage D_0−D_3, D_6, D_7−D_{10}		I_{OL}=12mA			2	V
I_{IH}	"H" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$		V_I=V_{DD} (Note 1)			1	μA
I_{IH}	"H" level input current D_0−D_7, D_9, D_{10}		V_I=12V			1	μA
I_{IH}	"H" level input current V_{REF}		V_I=5V			2	mA
I_{IL}	"L" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$		V_I=0V (Note 1) (Note 2)	−1			μA
I_{IL}	"L" level input current D_0−D_7, D_9, D_{10}		V_I=0V	−1			μA
I_{OZ}	Output current at off-state D_0−D_7, D_9, D_{10}		V_O=12V			1	μA
I_{OZ}	Output current at off-state P0, P1, D_8, $\overline{RESET}$		V_O=V_{DD}			1	μA
I_{DD}	Supply current	at active high-speed mode	f(X_{IN})=4MHz f(X_{CIN})=32kHz		3	9	mA
		at active low-speed mode	f(X_{IN})=Stop f(X_{CIN})=32kHz		50	150	μA
		at power down 1 mode (LCD operation)	f(X_{IN})=Stop f(X_{CIN})=32kHz		30	90	μA
		at power down 2 mode	f(X_{IN})=Stop f(X_{CIN})=Stop T_a=25℃		0.1	1.0	μA
			f(X_{IN})=Stop f(X_{CIN})=Stop			10	μA
IA_{DD}	A-D operation current		f(X_{IN})=4MHz		0.8	2.5	mA
IZ_{DD}	Zero cross comparator operation current		f(X_{IN})=4MHz		1.0	3.0	mA
R_{PU}	Pull-up resistor value P0 and P1		V_{DD}=5V V_I=0V	20	50	125	kΩ
V_{T+}−V_{T-}	Hysteresis INT_0, INT_1, INT_2				0.3		V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$				1.8		V
R_{COM}	COM output impedance		(Note 3)		2	10	kΩ
R_{SEG}	SEG output impedance		(Note 3)		3	15	kΩ
R_{VLC}	Internal resistor value for LCD power (impedance between V_{LC3} and V_{SS})		T_a=25℃	300	600	1200	kΩ
—	Zero cross comparator accuracy					±0.04V_{DD}	V

A-D CONVERTER CHARACTERISTICS (T_a=−20℃ to 85℃, V_{DD}=5V, V_{SS}=AV_{SS}=0V, f(X_{IN})=4MHz, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min.	Typ.	Max.	
—	Resolution					8	bit
—	Absolute accuracy	(Note 4)	V_{DD}=V_{REF}=5.12V			±2	LSB
R_{LADDER}	Ladder resistor			2.5	5	10	kΩ
t_{CONV}	Conversion time		f(X_{IN})=4MHz			25.5	μs
V_{REF}	Reference input voltage			2.7		V_{DD}	V
V_{IA}	Analog input voltage			0		V_{REF}	V

Notes 1. In this case, port P4 function is selected with software.
2. The pull-up resistors of ports P0 and P1 are disconnected.
3. Use an external power supply for the LCD power, and leave all pins open except the measured pin.
4. The absolute accuracy does not include quantification error.

BASIC TIMING DIAGRAM



Notes on zero cross input
Apply the voltage within the limits of −0.7 to 7.7V. Because the voltage is supplied to V_{DD} and V_{SS} through a parasitic diode in the microcomputer when the voltage is applied to port D_8 while V_{DD} and V_{SS} are open.
And keep the current to port D_8 to 100μA or less.

BUILT-IN PROM VERSION

In addition to the mask ROM versions, the 4520 Group has three programmable ROM versions software compatible with the mask ROM. One is the window-type EPROM version supplied with a built-in EPROM which can be written to and erased. Others are the One Time PROM versions whose PROMs can only be written to and not be erased. Since the functions of the built-in EPROM and One Time PROM versions are exactly the same, except erasure, all of them are referred to as built-in PROM versions in this explanation, unless otherwise noted.

The built-in PROM versions have functions similar to those of the mask ROM versions, but they have PROM mode that enables writing to built-in PROM.

Table 20 shows the product of built-in PROM version. Fig. 73 and Fig. 74 show the pin configulations of built-in PROM version. The One Time PROM versions have pin-compatibility with the mask ROM version. The built-in EPROM version has a different outline.

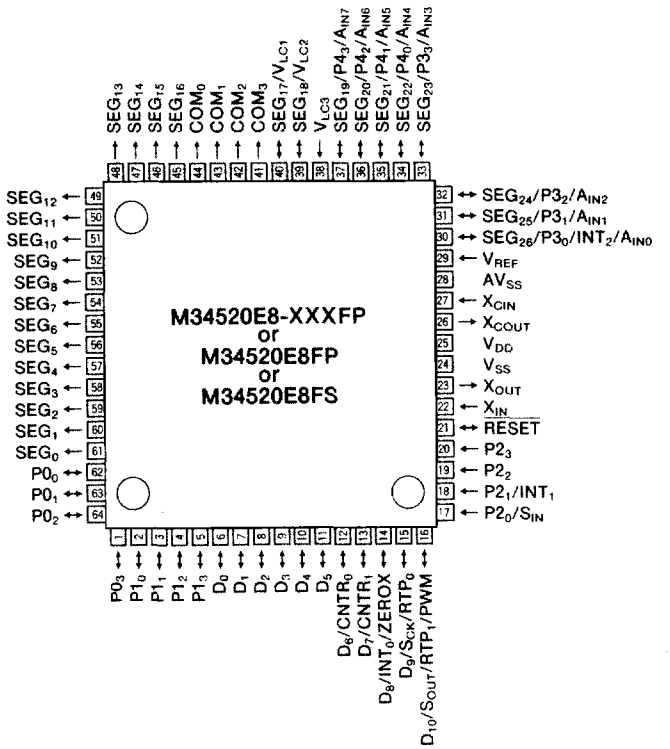
The built-in EPROM version is the microcomputer for program development. Use this microcomputer only for program development and prototype test.

Table 20 Product of built-in PROM version

Product	PROM size (×10 bits)	RAM size (×4 bits)	Package	Remarks
M34520E8-XXXSP/FP	8192 words	384 words	SP : 64P4B FP : 64P6N-A	One Time PROM version (shipped after writing) (Shipping after writing and testing in factory)
M34520E8SP/FP				One Time PROM version (shipped in blank)
M34520E8SS/FS *			SS : 64S1B-E FS : 64D0	Built-in EPROM version

* : For program development only

PIN CONFIGURATION (TOP VIEW)

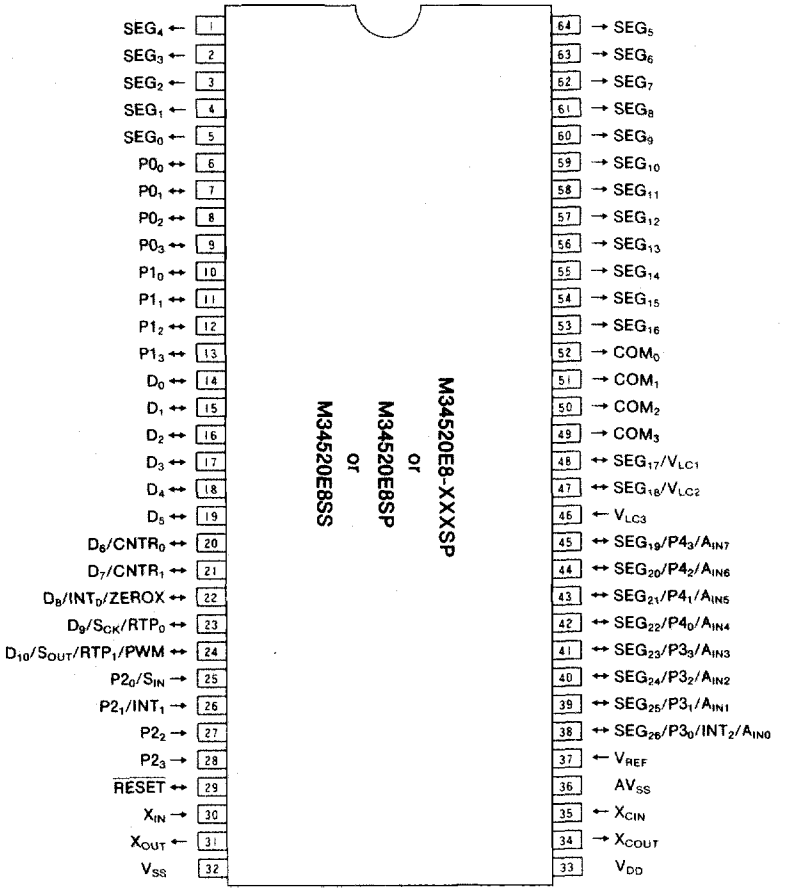


Outline 64P6N-A (One Time PROM version)
64D0 (Built-in EPROM version)

Fig. 73 Pin configuration of built-in PROM version

MITSUBISHI ELECTRIC

PIN CONFIGURATION (TOP VIEW)



Outline 64P4B (One Time PROM version)
64S1B-E (Built-in EPROM version)

Fig. 74 Pin configuration of built-in PROM version (continued)

(1) PROM mode

Each built-in PROM version has a PROM mode in addition to a normal operation mode. The PROM mode is used to write to and read from the built-in PROM.

In the PROM mode, the programming adapter can be used with a general-purpose PROM programmer to write to or read from the built-in PROM as if it were M5M27C256K. Programming adapters are listed in Table 21.

- Writing and reading of built-in PROM

  Programming voltage is 12.5V. Write the program in the PROM of the built-in PROM version as shown in Fig. 75.
- Erasing

  Only the built-in EPROM (M34520E8SS/FS) version has a transparent window for erasing on the top surface of the package. The EPROM is erased when it is exposed to ultraviolet light with a wavelength of 2537 Å to an integrated dose of 15 W·s/cm² or more through the window.

(2) Notes on handling

① Sunlight and fluorescent lamp contain light that can erase written information. Be sure to cover the transparent glass portion with a seal or other similar materials except when erasing.

② Mitsubishi Electric corp. provides the seal for covering the transparent glass. Take care that the seal does not touch the lead pins.

③ Clean the transparent glass before erasing. Fingers' fat and paste disturb the passage of ultraviolet light and may affect badly the erasure capability.

④ A high voltage is used for writing. Take care that overvoltage is not applied. Take care especially at turning on the power.

⑤ For the One Time PROM version shipped in blank, Mitsubishi Electric corp. does not perform PROM writing test and screening in the assembly process and following processes. In order to improve reliability after writing, performing writing and test according to the flow shown in Fig. 76 before using is recommended. (Products shipped in blank : PROM contents is not written in factory when shipped)

Table 21 Programming adapters

Microcomputer	Programming adapter
M34520E8-XXXSP, M34520E8SP, M34520E8SS	PCA4747
M34520E8-XXXFP, M34520E8FP	PCA4748
M34520E8FS	PCA4749

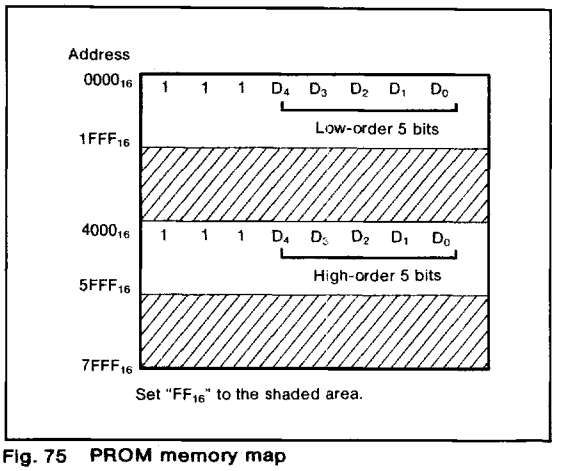


Fig. 75 PROM memory map

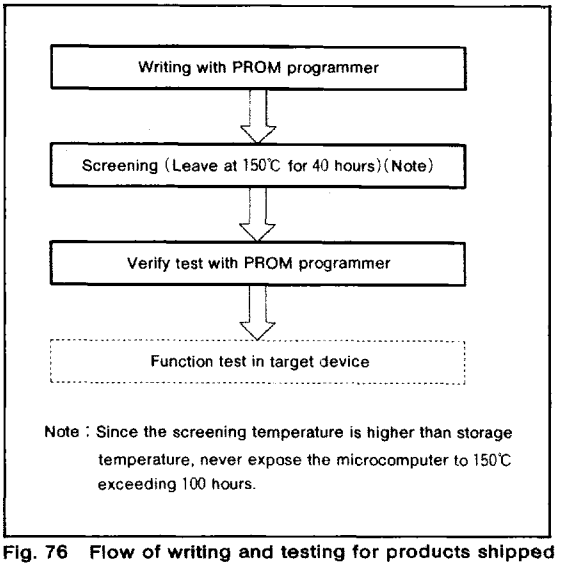


Fig. 76 Flow of writing and testing for products shipped in blank

ABSOLUTE MAXIMUM RATINGS FOR BUILT-IN PROM VERSION

Symbol	Parameter	Conditions		Rating	Unit
V_{DD}	Supply voltage			−0.3 to 7	V
V_I	Input voltage P0, P1, P2, P3, P4, $\overline{RESET}$, X_{IN}, X_{CIN}, V_{REF}			−0.3 to V_{DD}+0.3	V
V_I	Input voltage D_0−D_7, D_9, D_{10}			−0.3 to 13	V
V_I	Input voltage V_{LC1}, V_{LC2}, V_{LC3}			−0.3 to V_{DD}+0.3	V
V_I	Input voltage D_8/INT_0/ZEROX			−0.7 to V_{DD}+0.7	V
I_I	Input current D_8/INT_0/ZEROX			−100 to 100	μA
V_O	Output voltage P0, P1, $\overline{RESET}$	Output transistors in the cut-off state		−0.3 to V_{DD}+0.3	V
V_O	Output voltage D_8	Output transistors in the cut-off state		−0.7 to V_{DD}+0.7	V
V_O	Output voltage D_0−D_7, D_9, D_{10}	Output transistors in the cut-off state		−0.3 to 13	V
V_O	Output voltage X_{OUT}, X_{COUT}			−0.3 to V_{DD}+0.3	V
V_O	Output voltage SEG, COM			−0.3 to V_{LC3}+0.3	V
P_d	Power dissipation	T_a=25°C	DIP (Note 2)	1100	mW
			QFP, LCC (Note 2)	300	
T_{opr}	Operating temperature range (Note 1)			−20 to 85	°C
T_{stg}	Storage temperature range			−40 to 125	°C

Note1. Operating temperature range for built-in EPROM version (M34520E8FS, M34520E8SS) is −20°C to 70°C.
2. DIP : M34520E8-XXXSP, M34520E8SP, M34520E8SS
QFP, LCC : M34520E8-XXXFP, M34520E8FP, M34520E8FS

RECOMMENDED OPERATING CONDITIONS FOR BUILT-IN PROM VERSION

(T_a=−20℃ to 85℃ (Note 1), V_{DD}=2.5V to 5.5V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{DD}	Supply voltage (Note 2)	f(X_{IN})=4.0MHz	4.5	5.0	5.5	V
		f(X_{IN})=1.5MHz	2.5		5.5	
V_{RAM}	RAM back-up voltage (at RAM back-up mode)		2.0		5.5	V
V_{SS}	Supply voltage			0		V
AV_{SS}	Analog supply voltage (Note 3)			V_{SS}		V
V_{LC3}	Supply voltage for LCD (Note 4)		2.5		5.5	V
V_{IH}	"H" level input voltage P0, P1, P2, P3, P4, D_8		0.7V_{DD}		V_{DD}	V
V_{IH}	"H" level input voltage D_0–D_7, D_9, D_{10}		0.7V_{DD}		12	V
V_{IH}	"H" level input voltage X_{IN}		0.7V_{DD}		V_{DD}	V
V_{IH}	"H" level input voltage $\overline{RESET}$		0.85V_{DD}		V_{DD}	V
V_{IH}	"H" level input voltage $CNTR_1$, S_{IN}, S_{CK}, INT_0, INT_1, INT_2		0.8V_{DD}		V_{DD}	V
V_{IL}	"L" level input voltage P0, P1, P2, P3, P4, D_0–D_{10}		0		0.3V_{DD}	V
V_{IL}	"L" level input voltage X_{IN}		0		0.3V_{DD}	V
V_{IL}	"L" level input voltage $\overline{RESET}$		0		0.1V_{DD}	V
V_{IL}	"L" level input voltage $CNTR_1$, S_{IN}, S_{CK}, INT_0, INT_1, INT_2		0		0.2V_{DD}	V
I_{OL}(peak)	"L" level peak output current P0, P1, $\overline{RESET}$	V_{DD}=4.5V to 5.5V			10	mA
I_{OL}(peak)	"L" level peak output current D_4, D_5				40	mA
I_{OL}(peak)	"L" level peak output current D_0–D_3, D_6–D_{10}				24	mA
I_{OL}(avg)	"L" level average output current P0, P1, $\overline{RESET}$ (Note 5)				5	mA
I_{OL}(avg)	"L" level average output current D_4, D_5 (Note 5)				20	mA
I_{OL}(avg)	"L" level average output current D_0–D_3, D_6–D_{10} (Note 5)				12	mA
f(X_{IN})	f(X_{IN}) clock frequency (with a ceramic resonator) (Note 6)	V_{DD}=4.5V to 5.5V			4.0	MHz
		V_{DD}=2.5V to 5.5V			1.5	
f(X_{IN})	f(X_{IN}) clock frequency (Note 6) (with external clock input) (Note 7)	V_{DD}=4.5V to 5.5V			3.0	MHz
		V_{DD}=2.5V to 5.5V			0.8	
f(X_{CIN})	f(X_{CIN}) clock frequency (with a quartz-crystal oscillator) (Note 8)	V_{DD}=2.5V to 5.5V			50	kHz
tw(S_{CK})	Serial I/O external clock cycle ("H" level or "L" level pulse width)	V_{DD}=4.5V to 5.5V	750			ns
		V_{DD}=2.5V to 5.5V	2.0			μs
tw(CNTR1)	Timer external input cycle ("H" level or "L" level pulse width)	V_{DD}=4.5V to 5.5V	750			ns
		V_{DD}=2.5V to 5.5V	2.0			μs

Notes 1. Operating temperature range for built-in EPROM version (M34520E8FS, M34520E8SS) is −20℃ to 70℃.

2. Supply voltage at Mask ROM version is 2.2V to 5.5V.
   Supply voltage at built-in PROM version is 2.5V to 5.5V.
   - When using zero cross detection circuit
     V_{DD}=3.0V to 5.5V (Mask ROM version)
     V_{DD}=4.0V to 5.5V (Built-in PROM version)
   - When using A-D converter
     V_{DD}=2.7V to 5.5V (Mask ROM version, built-in PROM version)

3. Use AV_{SS} and V_{SS} at the same voltage level.

4. When using 1/2 bias : V_{LC1}=V_{LC2}=1/2·V_{LC3}
   When using 1/3 bias : V_{LC1}=1/3·V_{LC3}, V_{LC2}=2/3·V_{LC3}

5. Keep the total currents of I_{OL} (avg) for ports P0, P1, D_0–D_{10}, and $\overline{RESET}$ to 80mA or less.

6. However, the minimum value for the system clock frequency when using an A-D converter is 400kHz.

7. Keep the duty ratio of the external clock within the range shown in Table 22.

Table 22 Supply voltage and duty ratio of external clock

Supply voltage	Duty ratio of external clock
4.5V to 5.5V	40% to 60%
2.5V to 5.5V	30% to 70%

8. External clock cannot be used as f(X_{CIN}) clock.

ELECTRICAL CHARACTERISTICS FOR BUILT-IN PROM VERSION

(T_a=−20℃ to 85℃ (Note 1), V_{DD}=3.0V, unless otherwise noted)

Symbol	Parameter		Test condition	Limits			Unit
				Min.	Typ.	Max.	
V_{OL}	"L" level output voltage P0, P1, $\overline{RESET}$		I_{OL}=2mA			0.9	V
V_{OL}	"L" level output voltage D_4, D_5		I_{OL}=8mA			1.5	V
V_{OL}	"L" level output voltage D_0−D_3, D_6, D_7−D_{10}		I_{OL}=2mA			0.9	V
I_{IH}	"H" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$		V_I=V_{DD} (Note 2)			1	μA
I_{IH}	"H" level input current D_0−D_7, D_9, D_{10}		V_I=12V			1	μA
I_{IH}	"H" level input current V_{REF}		V_I=3V			1	mA
I_{IL}	"L" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$		V_I=0V (Note 2) (Note 3)	−1			μA
I_{IL}	"L" level input current D_0−D_7, D_9, D_{10}		V_I=0V	−1			μA
I_{OZ}	Output current at off-state D_0−D_7, D_9, D_{10}		V_O=12V			1	μA
I_{OZ}	Output current at off-state P0, P1, D_8, $\overline{RESET}$		V_O=V_{DD}			1	μA
I_{DD}	Supply current	at active high-speed mode	f(X_{IN})=1.5MHz f(X_{CIN})=32kHz		0.7	2.1	mA
			f(X_{IN})=500kHz f(X_{CIN})=32kHz		0.3	0.9	mA
		at active low-speed mode	f(X_{IN})=Stop f(X_{CIN})=32kHz		20	60	μA
		at power down 1 mode (LCD operation)	f(X_{IN})=Stop f(X_{CIN})=32kHz		7	21	μA
		at power down 2 mode	f(X_{IN})=Stop f(X_{CIN})=Stop T_a=25℃		0.1	1.0	μA
			f(X_{IN})=Stop f(X_{CIN})=Stop			10	μA
IA_{DD}	A-D operation current		f(X_{IN})=1.5MHz		0.2	0.6	mA
IZ_{DD}	Zero cross comparator operation current		f(X_{IN})=1.5MHz		0.8	2.4	mA
R_{PU}	Pull-up resistor value P0 and P1		V_{DD}=3V V_I=0V	40	100	250	kΩ
V_{T+}−V_{T-}	Hysteresis INT_0, INT_1, INT_2				0.3		V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$				0.7		V
R_{COM}	COM output impedance		(Note 4)		2	10	kΩ
R_{SEG}	SEG output impedance		(Note 4)		3	15	kΩ
R_{VLC}	Internal resistor value for LCD power (impedance between V_{LC3} and V_{SS})		T_a=25℃	300	600	1200	kΩ
—	Zero cross comparator accuracy					±0.04V_{DD}	V

Notes 1. Operating temperature range for built-in EPROM version (M34520E8FS, M34520E8SS) is −20℃ to 70℃.
2. In this case, port P4 function is selected with software.
3. The pull-up resistors of ports P0 and P1 are disconnected.
4. Use an external power supply for the LCD power, and leave all pins open except the measured pin.

ELECTRICAL CHARACTERISTICS FOR BUILT-IN PROM VERSION

(T_a=−20℃ to 85℃ (Note 1), V_{DD}=4.5V to 5.5V, unless otherwise noted)

Symbol	Parameter	Test condition	Limits			Unit
			Min.	Typ.	Max.	
V_{OL}	"L" level output voltage P0, P1, $\overline{RESET}$	I_{OL}=5mA			0.5	V
V_{OL}	"L" level output voltage D_4, D_5	I_{OL}=20mA			2	V
V_{OL}	"L" level output voltage D_0−D_3, D_6, D_7−D_{10}	I_{OL}=12mA			2	V
I_{IH}	"H" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$	V_I=V_{DD} (Note 2)			1	μA
I_{IH}	"H" level input current D_0−D_7, D_9, D_{10}	V_I=12V			1	μA
I_{IH}	"H" level input current V_{REF}	V_I=5V			2	mA
I_{IL}	"L" level input current P0, P1, P2, P3, P4, D_8, $\overline{RESET}$	V_I=0V (Note 2) (Note 3)	−1			μA
I_{IL}	"L" level input current D_0−D_7, D_9, D_{10}	V_I=0V	−1			μA
I_{OZ}	Output current at off-state D_0−D_7, D_9, D_{10}	V_O=12V			1	μA
I_{OZ}	Output current at off-state P0, P1, D_8, $\overline{RESET}$	V_O=V_{DD}			1	μA
I_{DD}	Supply current — at active high-speed mode	$f(X_{IN})$=4MHz, $f(X_{CIN})$=32kHz		3	9	mA
	at active low-speed mode	$f(X_{IN})$=Stop, $f(X_{CIN})$=32kHz		50	150	μA
	at power down 1 mode (LCD operation)	$f(X_{IN})$=Stop, $f(X_{CIN})$=32kHz		30	90	μA
	at power down 2 mode	$f(X_{IN})$=Stop, $f(X_{CIN})$=Stop, T_a=25℃		0.1	1.0	μA
		$f(X_{IN})$=Stop, $f(X_{CIN})$=Stop			10	μA
IA_{DD}	A-D operation current	$f(X_{IN})$=4MHz		0.8	2.5	mA
IZ_{DD}	Zero cross comparator operation current	$f(X_{IN})$=4MHz		1.0	3.0	mA
R_{PU}	Pull-up resistor value P0 and P1	V_{DD}=5V, V_I=0V	20	50	125	kΩ
V_{T+} − V_{T-}	Hysteresis INT_0, INT_1, INT_2			0.3		V
V_{T+} − V_{T-}	Hysteresis $\overline{RESET}$			1.8		V
R_{COM}	COM output impedance	(Note 4)		2	10	kΩ
R_{SEG}	SEG output impedance	(Note 4)		3	15	kΩ
R_{VLC}	Internal resistor value for LCD power (impedance between V_{LC3} and V_{SS})	T_a=25℃	300	600	1200	kΩ
—	Zero cross comparator accuracy				±0.04V_{DD}	V

A-D CONVERTER CHARACTERISTICS FOR BUILT-IN PROM VERSION

(T_a=−20℃ to 85℃ (Note 1), V_{DD}=5V, V_{SS}=AV_{SS}=0V, $f(X_{IN})$=4MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
—	Resolution				8	bit
—	Absolute accuracy (Note 5)	V_{DD}=V_{REF}=5.12V			±2	LSB
R_{LADDER}	Ladder resistor		2.5	5	10	kΩ
t_{CONV}	Conversion time	$f(X_{IN})$=4MHz			25.5	μs
V_{REF}	Reference input voltage		2.7		V_{DD}	V
V_{IA}	Analog input voltage		0		V_{REF}	V

Notes 1. Operating temperature range for built-in EPROM version (M34520E8FS, M34520E8SS) is −20℃ to 70℃.
2. In this case, port P4 function is selected with software.
3. The pull-up resistors of ports P0 and P1 are disconnected.
4. Use an external power supply for the LCD power, and leave all pins open except the measured pin.
5. The absolute accuracy does not include quantification error.

Notes on zero cross input
Apply the voltage within the limits of −0.7V to 7.7V. Because the voltage is supplied to V_{DD} and V_{SS} through a parasitic diode in the microcomputer when the voltage is applied to port D_8 while V_{DD} and V_{SS} are open.
And keep the current to port D_8 to 100μA or less.