HD404508, HD4074509

Description

The HD404508, HD4074509 is a CMOS 4-bit single-chip HMCS400 series microcomputer providing high program productivity. It is suitable for a digital tuning system by virtue of its PLL frequency synthesizer with 160 MHz prescaler, IF counter, A/D converter, and LCD driver incorporated in addition to ROM, RAM, I/O, timer, and serial interface. The HD4074509 is a ZTAT microcomputer incorporating PROM. It can dramatically shorten system development period and provide a smooth transition debugging to mass production.

Features

- 4-bit CPU (HMCS400)
- 8192 words × 10 bits ROM (HD404508)
- 16384 words × 10 bits programmable ROM (HD4074509) (Program specification is compatible with 27256 type.)
- 512 digits × 4 bits RAM
- I/O ports
  —I/O ports: 29
  —Input ports: 2
  —Output ports: 16 (multiplexed with the LCD segment pins)
- PLL with prescaler (max. 160 MHz)
  —12 programmable reference frequency
  —2-modulus prescaler (1/32, 1/33)
  —Phase comparator
  —PLL lock detection circuit
- LCD driver
  —Display duty: static, 1/2, 1/3
  —Segment signal output pins: 28 (including 16 multiplexed pins with R ports)
  —Common signal output pins: 3
- IF counter
  —Gate time: 1 ms, 4 ms, 8 ms, ∞ ms
  —Maximum input frequency: 15 MHz
- Two clock synchronous SCIs
- A/D converter (8 bits × 2 channels)
- Three timer/counters
  —8-bit timer (multiplexed with reload, event, PWM)
  —8-bit timer (multiplexed with reload, event)
  —20-bit timer (free-running, 125 ms interrupt)
- Instruction cycle time: 1.8 μs
- Subroutine stack: 16 levels including interrupts
- Five external and six internal interrupts
- Package: 80-pin flat plastic package (FP-80B)

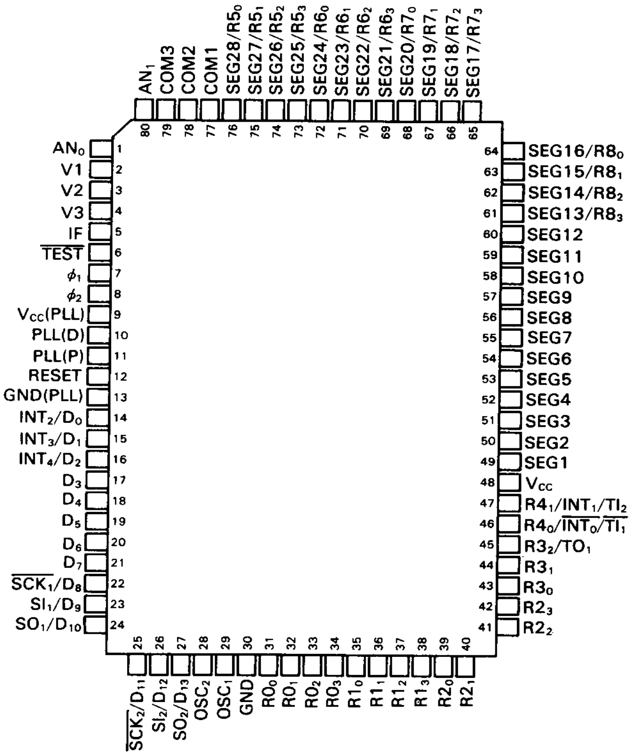
Program Development Support Tools

- Cross assembler and simulator software for use with IBM PCs and compatibles
- In circuit emulator for use with IBM PC
- Programming socket adapter for programming the EPROM-on-chip device

Ordering Information

Part No.	ROM (Words)	Package
HD404508FS	8192 (Mask ROM)	FP-80B
HD4074509FS	16384 (PROM)	

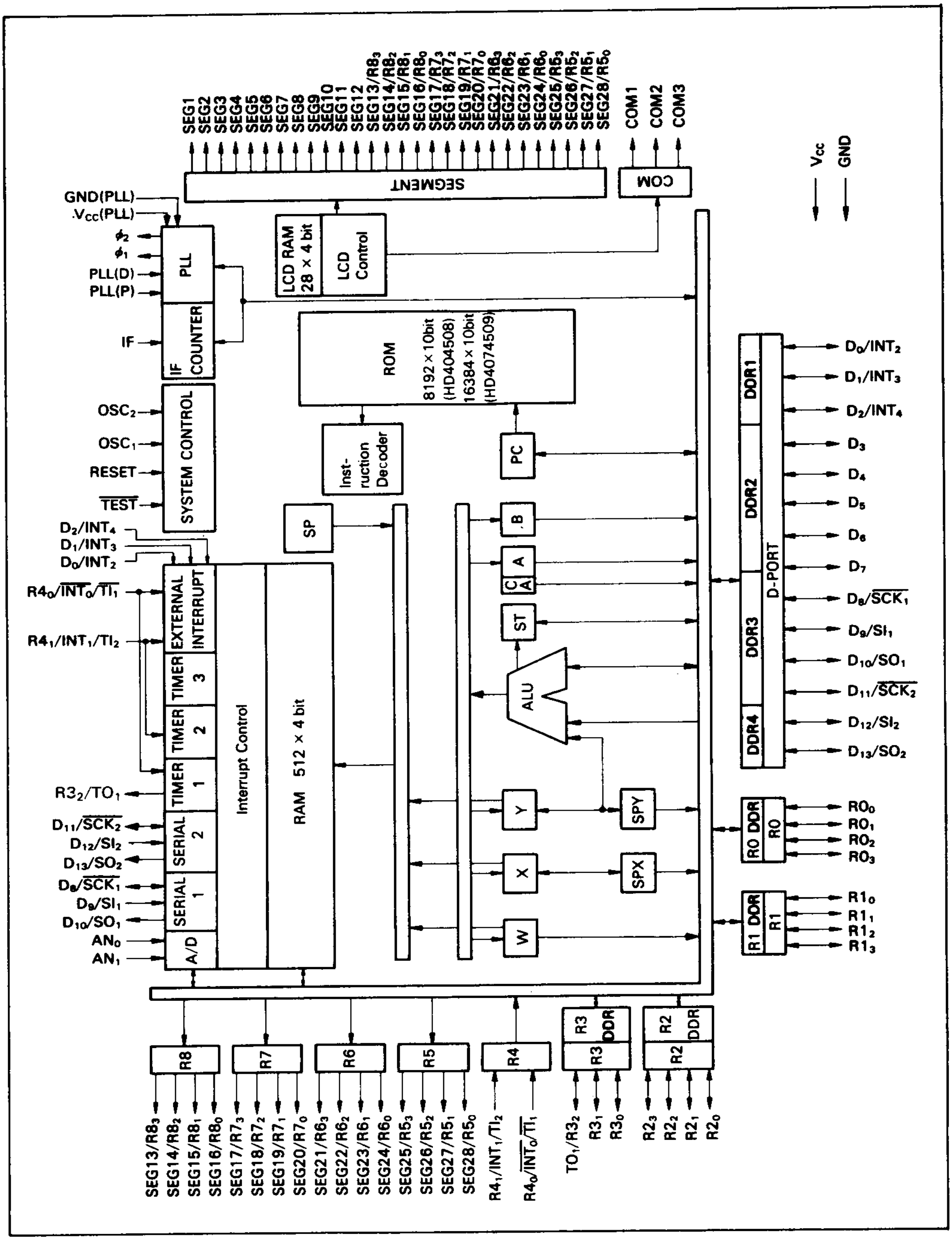
Pin Arrangement



(FP-80B)

(Top View)

Block Diagram



Pin Function

Power supply

V_{CC}: Power supply voltage (+5 V) is applied to V_{CC}.
GND: Connected to the ground (0 V).
V_{CC}(PLL): Connected to power supply for PLL (+5 V).
GND(PLL): Connected to the ground for PLL (0V).

Test

$\overline{\text{TEST}}$: Input for a factory test mode. Connected to the V_{CC} for normal operation.

Reset

RESET: RESET resets the MCU.

Oscillation

OSC1, OSC2: Input pins for the crystal oscillator circuits. Connected to the 4.5 MHz crystal resonator.

Ports

D port (D_0-D_{13}): The D port is an input/output port accessed on a bit basis. All D_0-D_{13} are input/output pins. D_0, D_1 and D_2 are multiplexed with INT_2, INT_3 and INT_4, respectively. D_8 to D_{13} are multiplexed with $\overline{SCK}_1$, SI_1, SO_1, $\overline{SCK}_2$, SI_2, and SO_2, respectively.

R ports (R0-R8): The R ports are input/output ports accessed in 4-bit units.
However, R3 is a 3-bit port and R4 a 2-bit port. R0 to R3 are I/O ports; R5 to R8 output ports; and R4 an input port. The pins $R3_2$, $R5_0$,-$R8_3$, $R4_0$, and $R4_1$ are multiplexed with TO_1, SEG28-SEG13, $\overline{INT}_0/\overline{TI}_1$, and INT_1/TI_2, respectively.

Interrupt

$\overline{INT}_0$, INT_1, INT_2, INT_3, INT_4: External interrupts for the MCU. These five pins are multiplexed with $R4_0/\overline{TI}_1$, $R4_1/TI_2$, D_0, D_1 and D_2, respectively.

Serial communication interface

$\overline{SCK}_1$, SI_1, SO_1, $\overline{SCK}_2$, SI_2, SO_2: These are transmit clock input/output pins ($\overline{SCK}_1$, $\overline{SCK}_2$), serial data input pins (SI_1, SI_2), and serial data output pins (SO_1, SO_2) used for serial interface. $\overline{SCK}_1$, $\overline{SCK}_2$, SI_1, SI_2, SO_1, and SO_2 are multiplexed with D_8, D_{11}, D_9, D_{12}, D_{10}, and D_{13}, respectively.

Timer

$\overline{TI}_1$, TI_2, TO_1: These are external clock input pins ($\overline{TI}_1$, TI_2) and timer output pin (TO_1) used for the timer. $\overline{TI}_1$, TI_2, and TO_1 are multiplexed with $R4_0/\overline{INT}_0$, $R4_1/INT_1$, and $R3_2$, respectively.

Liquid crystal display

COM1, COM2, COM3, SEG1-SEG28: Thes are the common signal output pins (COM1-COM3) and segment signal output pins (SEG1-SEG28) used for the LCD driver. The sixteen pins from SEG13 to SEG28 are multiplexed with $R8_3$-$R8_0$, $R7_3$-$R7_0$, $R6_3$-$R6_0$, and $R5_3$-$R5_0$, respectively.
V1, V2, V3: Power supply pins for the LCD.

PLL

PLL (P): An input pin for 160 MHz max. of local oscillation output (VCO output). This pin becomes active when pulse swallow mode is selected during PLL enable. Since an alternating current amplifier is incorporated, local oscillation output should be latched after being filtered through a capacitor.

PLL (D): An input pin for 20 MHz max. of local oscillation output (VCO output). This pin becomes active when direct dividing mode is selected during PLL enable. Since an alternating current amplifier is incorporated, local oscillation output should be latched after being filtered through a capacitor.

ϕ1, ϕ2: These pins are outputs of a phase comparator used by the PLL function. The same signal is output from ϕ1 and ϕ2.

IF counter

IF: An input pin for intermediate frequency measurement. Since an alternating current amplifier is incorporated, local oscillation output must be latched after being filtered through a capacitor.

AD converter

AN_0, AN_1: These pins are the AD converter input pins.

Memory Map

ROM Memory MAP

ROM memory map is shown in figure 1 and is explained below.

Vector address area ($0000 to $000F): Locations $0000 through $000F are reserved for JMPL instruction to branch to the start addresses of the reset routine and the interrupt service routine. After reset or interrupt routine is serviced, the program is executed from the vector address.

Zero page subroutine area ($0000 to $003F): Locations $0000 through $003F are reserved for subroutines. The program branches to the subroutine in $0000-$003F by the CAL instruction.

Pattern area ($0000 to $0FFF): The ROM data in locations $0000 through $0FFF can be referred to by the P instruction.

Program area
($0000 to $1FFF: HD404508,)
($0000 to $3FFF: HD4074509):
Locations $0000 through $1FFF, $3FFF can be used for program code.

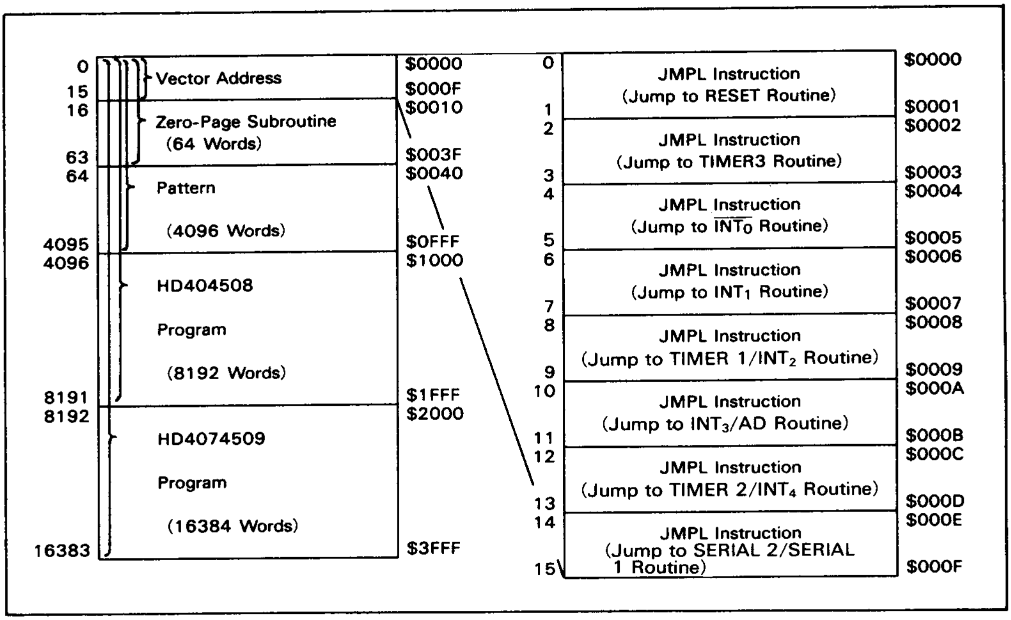

0
15
16
63
64
4095
4096
8191
8192
16383
Vector Address
Zero-Page Subroutine (64 Words)
Pattern (4096 Words)
HD404508 Program (8192 Words)
HD4074509 Program (16384 Words)
$0000
$000F
$0010
$003F
$0040
$0FFF
$1000
$1FFF
$2000
$3FFF
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
JMPL Instruction (Jump to RESET Routine)
JMPL Instruction (Jump to TIMER3 Routine)
JMPL Instruction (Jump to $\overline{INT_0}$ Routine)
JMPL Instruction (Jump to INT_1 Routine)
JMPL Instruction (Jump to TIMER 1/INT_2 Routine)
JMPL Instruction (Jump to INT_3/AD Routine)
JMPL Instruction (Jump to TIMER 2/INT_4 Routine)
JMPL Instruction (Jump to SERIAL 2/SERIAL 1 Routine)
$0000
$0001
$0002
$0003
$0004
$0005
$0006
$0007
$0008
$0009
$000A
$000B
$000C
$000D
$000E
$000F

Figure 1. ROM Memory Map

RAM Memory MAP

The MCU contains 512 digits × 4 bits RAM comprising data area and stack area. In addition to these areas, interrupt control bits, special registers, and display data RAM are mapped to the same RAM memory space. RAM memory map (figure 2) is described in the following paragraphs.

Interrupt control bit area ($000 to $003, $020 to $023): The interrupt control bit area (figure3) is used to control interrupt. It can be accessed only by the RAM bit manipulation instruction. However, the interrupt request flag cannot be set by software. The RSP bit is used only to reset the stack pointer.

Special registers area ($004 to $01F, $024 to $03F): The special registers are the mode registers for external interrupt, serial interface, and timer, data direction and data registers for I/O ports. As shown in figure 2, these registers can be classified into three types: write-only, read-only, and read/write registers.
Note that some of these registers cannot be accessed by the RAM bit manipulation instruction.

Data area ($040 to $01FF): The memory register (MR), 16 addresses deep ($040-$04F), can be accessed by the LAMR and XMRA instructions (figure 4).

Stack area ($03C0 to $03FF): The stack area is used for saving the contents of the program counter (PC), status (ST), and carry (CA). This area can be used as a 16-nesting-level subroutine stack in which one level requires 4 digits. The data to be saved and save conditions are shown in figure 4. The program counter is restored by the RTN and RTNI instructions. Status and carry are restored by the RTNI instruction only. Any space not used is available data storage.

0	RAM-mapped Register	$000
63		$03F
64	Memory Register (MR) (16 digits)	$040
79		$04F
80	DATA (448 Digits)	$050
511		$1FF
512	▨	$200
927		$39F
928	LCD Data (28 Digits)	$3A0
955		$3BB
956	▨	$3BC
959		$3BF
960	Stack (64 Digits)	$3C0
1023		$3FF

0	Interrupt Control Bits (1)		B	$000
3				$003
4	Miscelaneous Reg.	(MIS)	R/W	$004 • 3
5	PLL Control Reg. A	(PCRA)	W	$005
6	PLL Control Reg.B.C	(PCRB/PCRC)	R/W	$006 • 1 • 3
7	PLL Data Reg. A	(PDRA)	W	$007
8	PLL Data Reg. B	(PDRB)	W	$008
9	PLL Data Reg. C	(PDRC)	W	$009
10	PLL Data Reg. D	(PDRD)	W	$00A
11	▨			$00B
12	A/D Mode Reg.	(AMR)	W	$00C
13	A/D Data Reg. Lower	(ADRL)	R	$00D • 3
14	A/D Data Reg. Upper	(ADRU)	R	$00E • 3
15	▨			$00F
16	Serial Mode Reg. 1	(SMR1)	R/W	$010 • 2 • 3
17	Serial Clock Reg. 1	(SCR1)	W	$011 • 2
18	Serial Data Reg.1 Lower	(SR1L)	R/W	$012 • 3
19	Serial Data Reg.1 Upper	(SR1U)	R/W	$013 • 3
20	Serial Mode Reg. 2	(SMR2)	R/W	$014 • 2 • 3
21	Serial Clock Reg. 2	(SCR2)	W	$015 • 2
22	Serial Data Reg.2 Lower	(SR2L)	R/W	$016 • 3
23	Serial Data Reg.2 Upper	(SR2U)	R/W	$017 • 3
24	Timer Mode Reg. 1	(TMR1)	W	$018 • 2
25	Timer Output Reg. 1	(TOR1)	R/W	$019 • 2 • 3
26	Timer 1	(TC1L/TL1L)	R/W	$01A • 1 • 3
27	Timer 1	(TC1U/TL1U)	R/W	$01B • 1 • 3
28	Timer Mode Reg. 2	(TMR2)	W	$01C • 2
29	Timer Output Reg. 2	(TOR2)	R/W	$01D • 2 • 3
30	Timer 2	(TC2L/TL2L)	R/W	$01E • 1 • 3
31	Timer 2	(TC2U/TL2U)	R/W	$01F • 1 • 3
32	Interrupt Control Bits (2)		B	$020
35				$023
36	LCD Output Reg.	(LOR)	W	$024
37	LCD Control Reg.	(LCR)	W	$025
38	LCD Mode Reg.	(LMR)	W	$026
39	▨			$027
40	IF Control Reg.1.2	(IFC1/IFC2)	R/W	$028 • 1 • 3
41	IF DATA Reg. A	(IFDA)	R	$029 • 3
42	IF DATA Reg. B	(IFDB)	R	$02A • 3
43	IF DATA Reg. C	(IFDC)	R	$02B • 3
44	IF DATA Reg. D	(IFDD)	R	$02C • 3
45	▨			$02D
46	Port Mode Reg. A	(PMRA)	W	$02E
47	▨			$02F
48	Port R0 DDR	(DDR0)	W	$030
49	Port R1 DDR	(DDR1)	W	$031
50	Port R2 DDR	(DDR2)	W	$032
51	Port R3 DDR	(DDR3)	W	$033
52	▨			$034
58				$03A
59	Port D DDR 0	(DDRD0)	W	$03B
60	Port D DDR 1	(DDRD1)	W	$03C
61	Port D DDR 2	(DDRD2)	W	$03D
62	Port D DDR 3	(DDRD3)	W	$03E
63	▨			$03F

B : Bit Set/Reset
R : Read Only
W : Write Only
R/W : Read/Write

• 1 : Two registers are mapped on the same address.

$006	PCRC	R	PCRB	W
$01A	TC1L	R	TL1L	W
$01B	TC1U	R	TL1U	W
$01E	TC2L	R	TL2L	W
$01F	TC2U	R	TL2U	W
$028	IFC2	R	IFC1	W

• 2: SEM/SEMD and REM/REMD instructions can be used for these registers.

• 3: TM/TMD instruction can be used for these registers.

▨: Reserved. Expandable.

Figure 2. RAM Memory Map

	bit 3	bit 2	bit 1	bit 0	
0	IMT3 (IM of TIMER 3)	IFT3 (IF of TIMER 3)	RSP (Reset SP Bit)	I/E (Interrupt Enable Flag)	\$000
1	IM1 (IM of INT_1)	IF1 (IF of INT_1)	IM0 (IM of $\overline{INT_0}$)	IF0 (IF of $\overline{INT_0}$)	\$001
2	IM3 (IM of INT_3)	IF3 (IF of INT_3)	IMT1 (IM of TIMER 1)	IFT1 (IF of TIMER 1)	\$002
3	IMS2 (IM of SERIAL 2)	IFS2 (IF of SERIAL 2)	IMT2 (IM of TIMER 2)	IFT2 (IF of TIMER 2)	\$003

	bit 3	bit 2	bit 1	bit 0	
32	▨	▨	▨	ADSF (A/D Start Bit, Status Flag)	\$020
33	▨	▨	▨	▨	\$021
34	IMAD (IM of A/D)	IFAD (IF of A/D)	IM2 (IM of INT_2)	IF2 (IF of INT_2)	\$022
35	IMS1 (IM of SERIAL 1)	IFS1 (IF of SERIAL 1)	IM4 (IM of INT_4)	IF4 (IF of INT_4)	\$023

IF : Interrupt Request Flag
IM : Interrupt Mask
I/E : Interrupt Enable Flag
SP : Stack Pointer

(Note) Each bit in the interrupt control bits area is set by the SEM/SEMD instruction, reset by the REM/REMD instruction, and is tested by the TM/TMD instruction. It is not affected by other instructions. Furthermore the interrupt request flag is not affected by the SEM/SEMD instruction. The content of status becomes invalid when "Reserved" bit and RSP bit are tested by a TM or TMD instruction. ▨ : Reserved

Figure 3. Configuration of Interrupt Control Bit Area (RAM Space)

Memory Register

64	MR (0)	\$040
65	MR (1)	\$041
66	MR (2)	\$042
67	MR (3)	\$043
68	MR (4)	\$044
69	MR (5)	\$045
70	MR (6)	\$046
71	MR (7)	\$047
72	MR (8)	\$048
73	MR (9)	\$049
74	MR (10)	\$04A
75	MR (11)	\$04B
76	MR (12)	\$04C
77	MR (13)	\$04D
78	MR (14)	\$04E
79	MR (15)	\$04F

Stack Area

960	level 16	\$3C0
	level 15	
	level 14	
	level 13	
	level 12	
	level 11	
	level 10	
	level 9	
	level 8	
	level 7	
	level 6	
	level 5	
	level 4	
	level 3	
	level 2	
1023	level 1	\$3FF

	bit 3	bit 2	bit 1	bit 0	
1020	ST	$\overline{PC_{13}}$	$\overline{PC_{12}}$	$\overline{PC_{11}}$	\$3FC
1021	$\overline{PC_{10}}$	$\overline{PC_9}$	$\overline{PC_8}$	$\overline{PC_7}$	\$3FD
1022	CA	$\overline{PC_6}$	$\overline{PC_5}$	$\overline{PC_4}$	\$3FE
1023	$\overline{PC_3}$	$\overline{PC_2}$	$\overline{PC_1}$	$\overline{PC_0}$	\$3FF

$PC_{13}-PC_0$; Program Counter
ST; Status
CA; Carry

Figure 4. Configuration of Memory Register, Stack Area and Stack Position

Functional Description

Registers and Flags

The MCU has nine registers and two flags for the CPU operations. They are illustrated in figure 5 and described in the following paragraphs.

Accumulator (A), B register (B): The 4-bit register accumulator and B resgister are used to hold the results from the arithmetic logic unit (ALU), and to transfer data to/from memories, I/O, and other registers.

W register (W), X register (X), Y register (Y): W register is a 2-bit, and X and Y registers are 4-bit registers used for register indirect RAM addressing. Y register is also used for D-port addressing.

SPX register (SPX), SPY register (SPY): The 4-bit registers SPX and SPY are used to assist the X and Y registers, respectively.

Carry (CA): Carry (CA) stores ALU overflow generated by the arithmetic operation. It is affected by the SEC, REC, ROTL, and ROTR instructions. Carry is pushed onto the stack during interrupt servicing, and popped from the stack by the RTNI instruction, but not by the RTN instruction.

Status (ST): Status (ST) latches ALU overflow generated by the arithmetic and compare instructions, Not Zero from ALU, and results of bit tests. It is a branch condition of the BR, BRL, CAL, and CALL instructions. The contents of the status remain unchanged until the next arithmetic, compare, or bit test instruction is executed. ST becomes 1 after the BR, BRL, CAL, or CALL instruction is executed irrespective of whether it is executed or skipped. The contents of the status are pushed onto the stack during interrupt servicing, and popped from the stack by the RTNI instruction, but not by the RTN instruction.

Program counter (PC): The 14-bit program counter is a binary counter which holds a ROM address.

Stack pointer (SP): The 10-bit stack pointer contains the address of the next stack area. SP is initialized to $3FF by MCU reset. It is decremented by 4 when data is pushed onto the stack, and is incremented by 4 when data is popped from the stack.
Since the upper 4 bits of SP are fixed to 1111, the stack can be used for up to 16 levels.
SP is initialized to $3FF in two ways; MCU reset and RSP bit reset with the REM or REMD instruction.

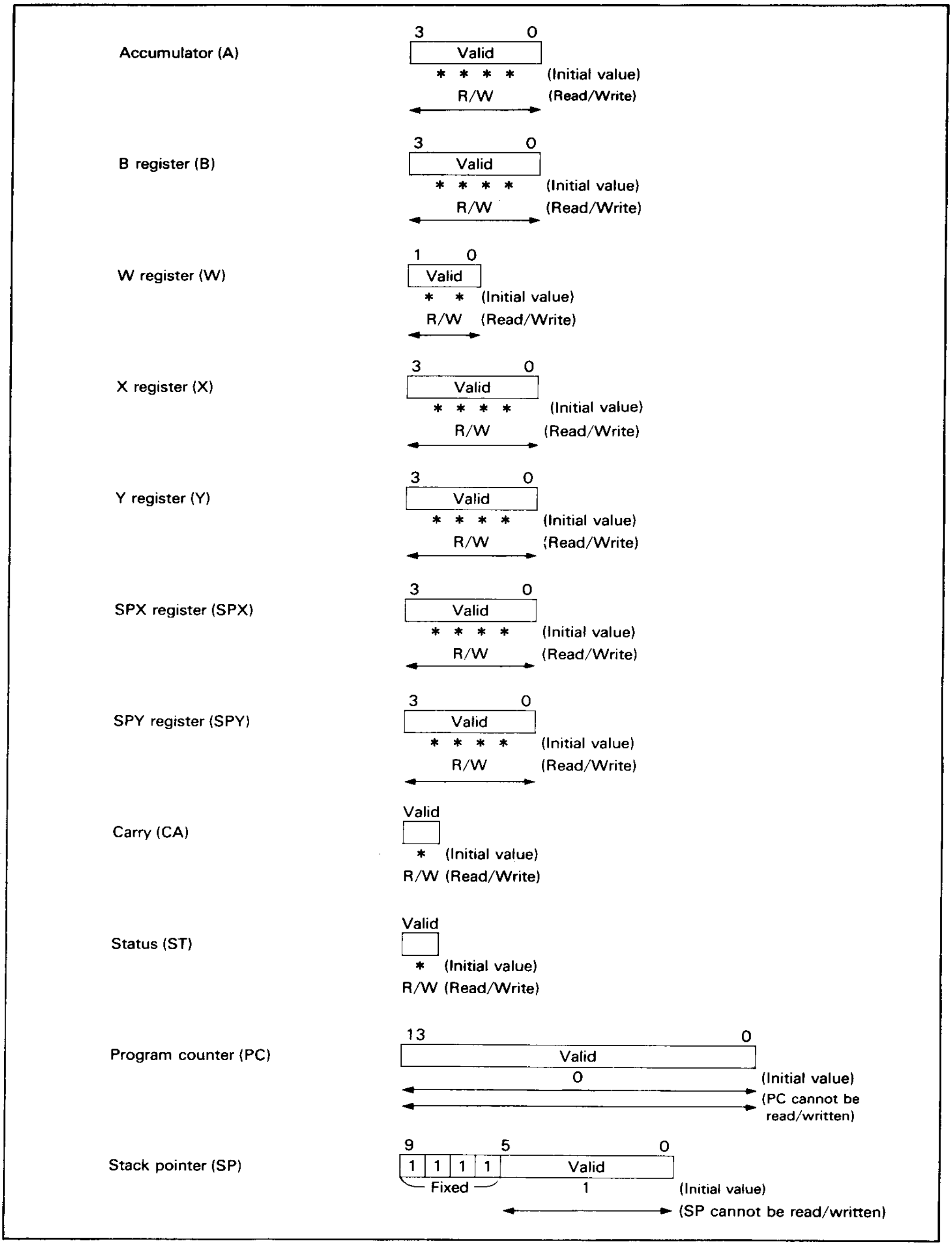


Figure 5. Registers and Flags

Reset

Power-on reset:

At power-on, the MCU automatically enters reset state for approximately 60 to 65 ms to assure oscillation stabilization time, and then the MCU recovers from that state.

Reset by RESET pin (reset from active mode, standby mode):

The MCU is reset at the rising edge of the reset signal. This signal should be asserted high for at least 3.6 μs. The MCU is then reset for one instruction cycle time, after which reset is cancelled.

Reset by RESET pin (reset from stop mode):

The MCU is reset at the rising edge of the reset signal. This signal should be asserted high for at least 20 ms. At the beginning of oscillation, the MCU automatically enters reset state for 60 to 65 ms to assure oscillation stabilization time, and then reset is cancelled.

Read function:

The accessible RESET pin is mapped to the bit 0 ($004, 0) of the miscellaneous register (MLS).
Note that from power-on to the begining of normal oscillation, the I/O pins are undefined since the I/O data direction registers are not initialized.Values initialized by MCU reset are shown in table 1 and table 2.

Table 1. Initial Value by MCU Reset

Items		Initial Value by MCU Reset	Contents
Program Counter (PC)		$0000	Execute program from the top of ROM address
Status (ST)		1	Enable to branch with conditional branch instruction
Stack Pointer (SP)		$3FF	Stack level is 0
Interrupt Flag/Mask	Interrupt Enable Flag (I/E)	0	
	Interrupt Request Flag (IF)	0	
	Interrupt Mask (IM)	1	
PLL	PLL Control Register A, B, C (PCRA, PCRB, PCRC)	0000	
A/D	A/D Mode Register (AMR)	0000	
Serial Interface	Serial Mode Register 1, 2 (SMR1, SMR2)	0000	
	Serial Clock Register 1, 2 (SCR1, SCR2)	0000	
Timer/ Counter	Timer Mode Register 1, 2 (TMR1, TMR2)	0000	
	Timer Output Register 1, 2 (TOR1, TOR2)	0000	
	Timer/Event Counter 1, 2 Lower, Upper (TC1L, TC1U, TC2L, TC2U)	0000	
	Timer 3	Note 1	
LCD	LCD Output Register (LOR)	Note 2	
	LCD Control Register (LCR)	0000	
	LCD Mode Register (LMR)	0000	After reset is cancelled, duty must be set by program
IF	IF Control Register 1, 2 (IFC1, IFC2)	0000	
	IF Counter Data Register A, B, C, D (IFDA, IFDB, IFDC, IFDD)	0000	
I/O	Port D Data Register	1	
	Port R0-R3 Data Register	Note 3	
	Port R0-R3 DDR (DDR0, DDR1, DDR2, DDR3)	Note 4	
	Port D DDR (DDRD0, DDRD1, DDRD2, DDRD3)	0000	
	Port Mode Register A (PMRA)	0000	

Note 1: Initial value of timer 3 by MCU reset
The initial value of depends on the value of the data retention bit (MIS$_3$: $004, 3).

Item	Reset when power-on reset or data retention bit is 0	Reset when data retention bit is 1
Timer 3	All bits become 0 and are initialized	Not initialized

Note 2: Initial value of the LCD output register by MCU reset
The initial value of the LCD output register depends on the value of the data retention bit (MIS_3: $004, 3).

Item	Reset when power-on reset or data retention bit is 0	Reset when data retention bit is 1
LCD output register	LCD output register becomes 0000 and SEG/R-port pins become R port	Retained (SEG/R port is not changed)

Note 3: Initial value of the port R0-R3 data register by MCU reset
The initial value of the port R0-R3 data register depends on the value of the data retention bit (MIS_3: $004, 3).

Item	Reset when power-on reset or data retention bit is 0	Reset when data retention bit is 1
Port R0-R3 data register	1	Retained

Note 4: Initial value of the R port data direction register by MCU reset
Initial value of the R port data direction register depends on the value of the data retention bit (MIS_3: $004, 3).

Item	Reset when power-on reset or data retention bit is 0	Reset when data retention bit is 1
R port data direction register	The R port data direction register becomes 0000 and R port becomes input port	Retained (I/O of the R port is not changed)

Table 2. Values of the Registers except for Table 1 after the Reset

Items		Initial Value by MCU Reset from Active Mode	Initial Value by MCU Reset from Standby Mode	Initial Value by MCU Reset from Oscillation Stop Mode	Initial Value by MCU Reset from Power-on
Carry (CA)		The contents of the items just before MCU reset are not assured. It is necessary to initialize them by software again.			
Accumulator (A)					
B Register (B)					
W Register (W)					
X/SPX Register (X/SPX)					
Y/SPY Register (Y/SPY)					
PLL Data Register (PDRA, PDRB, PDRC, PDRD)					
A/D Data Register Lower, Upper (ADRL, ADRU)					
Serial Data Register 1, 2 Lower, Upper (SR1L, SR1U, SR2L, SR2U)					
RAM		The contents of RAM just before MCU reset are retained			Same as above
Miscellaneous Register (MIS)	MIS_3	The contents of MIS_3 just before MCU reset are retained			0
	MIS_0	MIS_0 reads RESET pin level at read operation			

Interrupt

Eleven interrupt sources are available on the MCU: external requests ($\overline{\mathrm{INT}_0}$, INT_1, INT_2, INT_3, INT_4), timers (TIMER 1, TIMER 2, TIMER 3), A/D converter, and serial interfaces (SERIAL 1, SERIAL 2). For each source, an interrupt request flag (IF) and interrupt mask (IM) are provided to control and maintain the interrupt requests. To control the entire interrupt process, the interrupt enable flag (I/E) is provided.

Since the vector addresses are shared between timer 1 and INT_2, between A/D and INT_3, between timer 2 and INT_4, and between serial 1 and serial 2, determining which request occurs must be done by software.

Interrupt control bits and interrupt processing: The interrupt control bits are mapped to addresses \$000 through \$003, and \$020 through \$023 of RAM space and are accessed by the RAM bit manipulation instruction.

However, interrupt request flag IF cannot be set by software.

The interrupt enable flag (I/E) and interrupt request flag (IF) are set to 0, and the interrupt mask (IM) is initialized to 1 by MCU reset.

An interrupt control circuit block diagram is shown in figure 6. Interrupt priority and vector addresses are shown in table 3, and the interrupt processing conditions for 11 different interrupt types in table 4.

When interrupt request flag is 1 and interrupt mask is 0, an interrupt request is generated. If the interrupt enable flag is 1 at that time, interrupt processing activates. Then a vector address corresponding to the interrupt request is generated from priority PLA.

An interrupt processing sequence and flowchart are shown in figures 7 and 8 respectively. When an interrupt is received, the current instruction execution finishes at the first cycle, I/E is reset at the second cycle, then the contents of the carry, status, and program counter are pushed onto the stack in the second and third cycles, and the program jumps to the vector address to restart instruction execution in the third cycle. For each vector address area, the JMPL instruction must be programmed to branch the starting address of interrupt routine. The interrupt request flag which causes interrupt processing must be reset by software in an interrupt routine.

Interrupt enable flag (I/E: \$000, 0): The interrupt enable flag controls enable/disable of all interrupt requests. It is reset by interrupt processing and set by the RTNI instruction. (See table 5.)

External interrupts ($\overline{\mathrm{INT}_0}$, INT_1, INT_2, INT_3, INT_4): Five external interrupt pins are provided for the MCU. The external interrupt request flag is set at the falling edge of the $\overline{\mathrm{INT}_0}$ input. It is set at the rising edge of the INT_1, INT_2, INT_3, and INT_4 inputs. When using $\overline{\mathrm{INT}_0}$ and INT_1, interrupt input should be enabled by setting the external interrupt enable bit of the timer output register (TOR1: \$019, TOR2: \$01D). If the interrupt enable bit is reset, the external interrupt request flag is not set and interrupt processing is not performed since input signal itself is maskes. (See section "Timer".)

When using INT_2, INT_3, and INT_4, the corresponding bits of port mode register A (PMRA: \$02E) select external interrupt input. If port mode register A is set, the corresponding data direction register bit is reset automatically to input. If port mode register A is reset, external interrupt request flag is not set in spite of external interrupt signal input and interrupt processing is not performed since the external interrupt input signal is masked.

External interrupt requrst flags (IF0: \$001, 0 IF1: \$001, 2 IF2: \$022, 0 IF3: 002, 2 IF4: \$023, 0): External interrupt request flag IF0 is set at the falling edge of the $\overline{\mathrm{INT}_0}$ input. IF1-IF4 are set at the rising edge of INT_1-INT_4. (See table 6.)

External interrupt masks (IM0: \$001, 1 IM1: \$001, 3 IM2: \$022, 1 IM3: \$022, 3 IM4: \$023,1): The external interrupt request masks mask the interrupt request from the external interrupt request flag. (See table 7.)

Port mode register A (PMRA: \$02E): Port mode register A is a 3-bit write-only register which controls the D_0/INT_2, D_1/INT_3, and D_2/INT_4 pins as shown in figure 9.

Port mode register A is initialized to \$0 by MCU reset, so that these pins are all set to D port I/O pins after reset.

Table 3. Vector Addresses and Interrupt Priority

Reset, interrupt	Priority	Vector address
RESET	–	\$0000
TIMER3	1	\$0002
$\overline{INT_0}$	2	\$0004
INT_1	3	\$0006
$TIMER1/INT_2$	4	\$0008
$INT_3/A/D$	5	\$000A
$TIMER2/INT_4$	6	\$000C
SERIAL2/SERIAL1	7	\$000E

Table 4. Interrupt Servicing Conditions

	Interrupt Source						
Interrupt Control Bit	TIMER3	$\overline{INT_0}$	INT_1	$TIMER1/INT_2$	$INT_3/A/D$	$TIMER2/INT_4$	SERIAL1 /SERIAL2
I/E	1	1	1	1	1	1	1
$IFT3 \cdot \overline{IMT3}$	1	0	0	0	0	0	0
$IF0 \cdot \overline{IM0}$	*	1	0	0	0	0	0
$IF1 \cdot \overline{IM1}$	*	*	1	0	0	0	0
$IFT1 \cdot \overline{IMT1} + IF2 \cdot \overline{IM2}$	*	*	*	1	0	0	0
$IF3 \cdot \overline{IM3} + IFAD \cdot \overline{IMAD}$	*	*	*	*	1	0	0
$IFT2 \cdot \overline{IMT2} + IF4 \cdot \overline{IM4}$	*	*	*	*	*	1	0
$IFS2 \cdot \overline{IMS2} + IFS1 \cdot \overline{IMS1}$	*	*	*	*	*	*	1

* : Both 0 and 1 do not affect operation

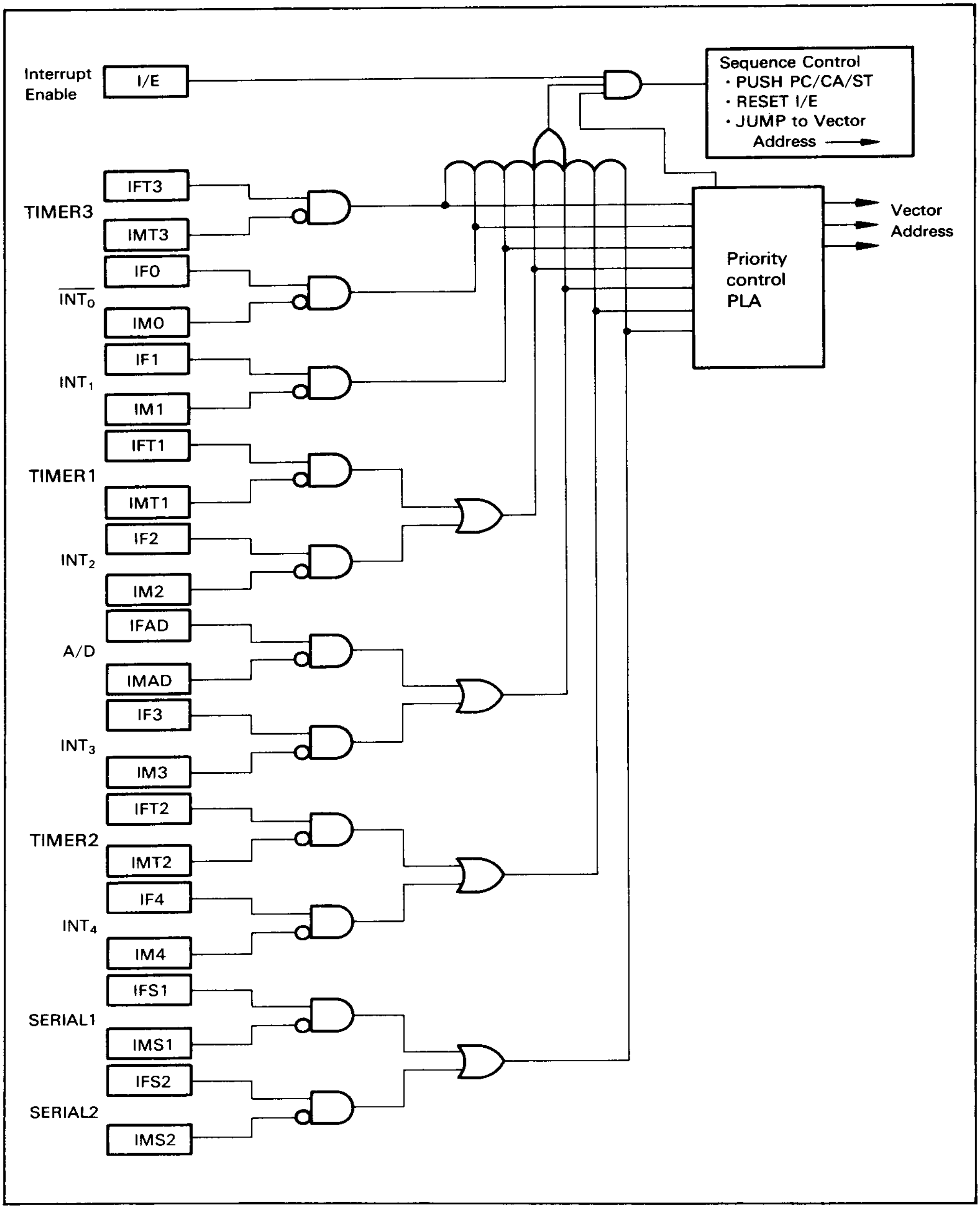


Figure 6. Interrupt Control Circuit Block Diagram

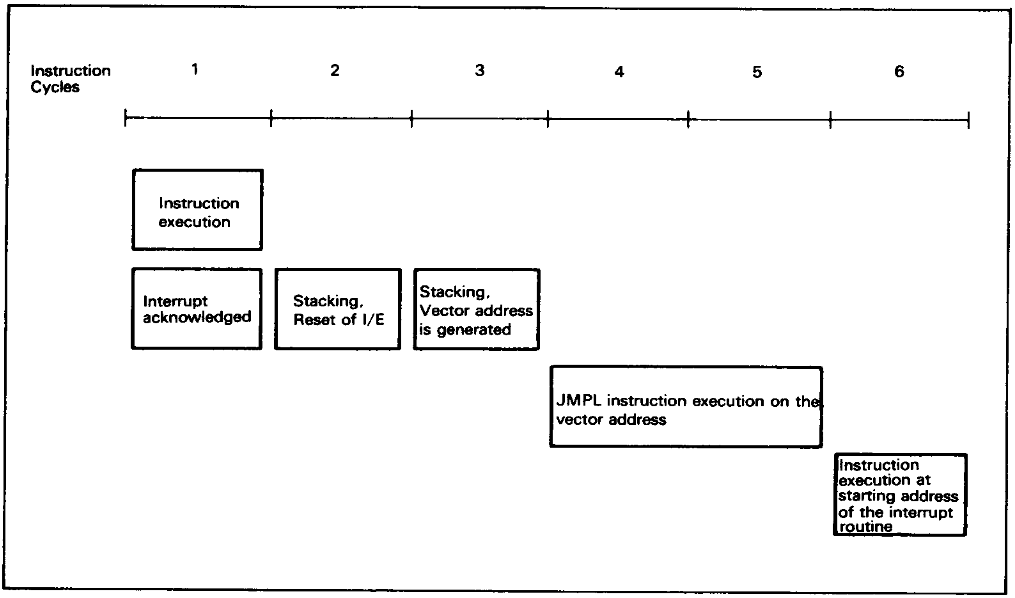

Instruction Cycles
1
2
3
4
5
6
Instruction execution
Interrupt acknowledged
Stacking, Reset of I/E
Stacking, Vector address is generated
JMPL instruction execution on the vector address
Instruction execution at starting address of the interrupt routine

Figure 7. Interrupt Servicing Sequence

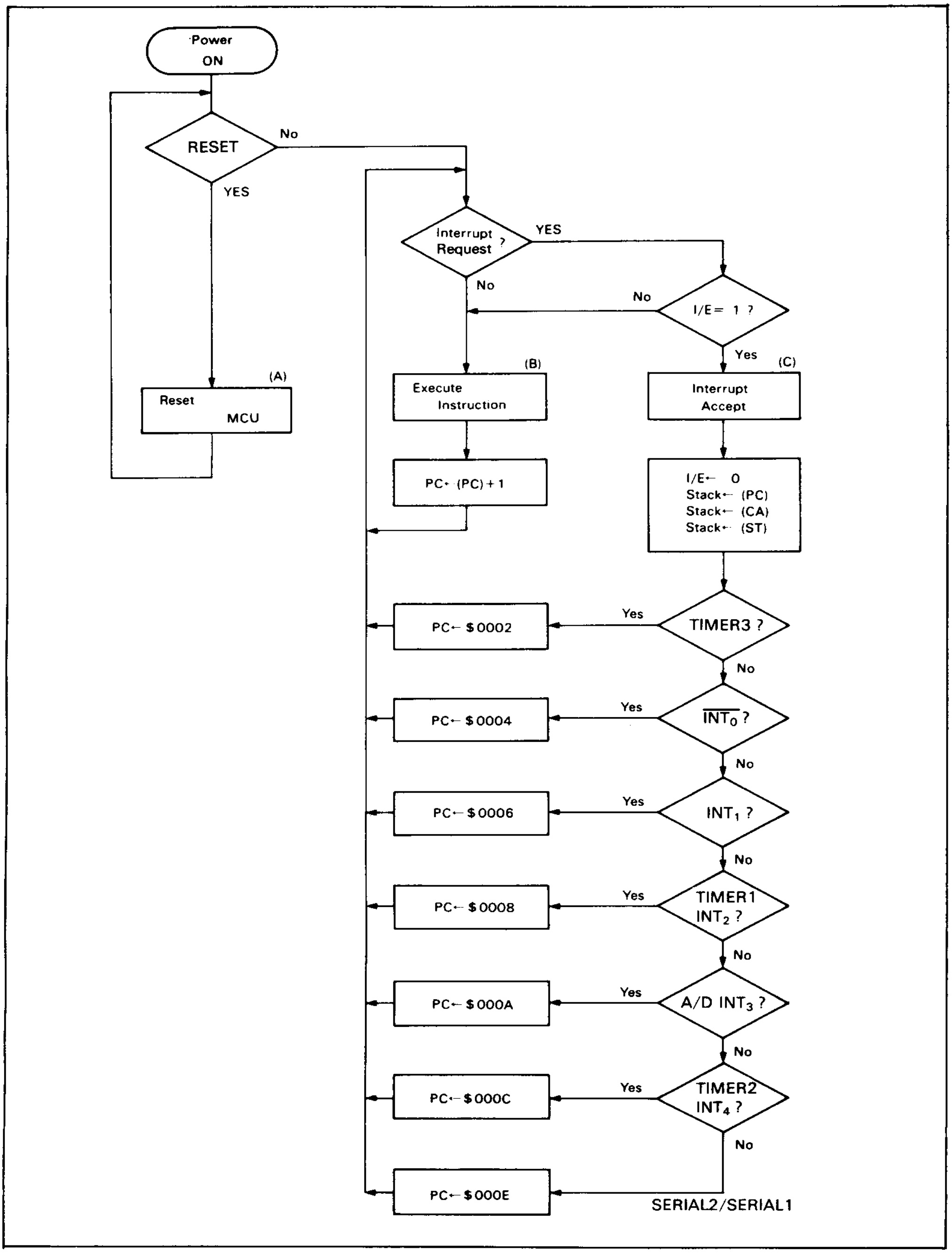


Figure 8. Interrupt Servicing Flowchart

Table 5. Interrupt Enable Flag ($000,0)

Interrupt Enable Flag (I/E)	Interrupt Enable /Disable
0	Disable
1	Enable

Initial value 0, R/W

Table 6. External Interrupt Request Flag ($001,0, $001,2, $002,2, $022,0, $023,0)

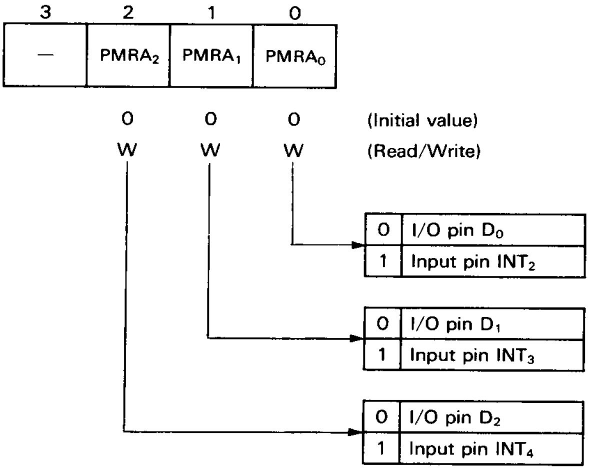
External Interrupt Request Flag (IF0, IF1, IF2, IF3, IF4)	Interrupt Requests
0	No
1	Yes

Initial value 0, R/W (cannot be set to 1)

Table 7. External Interrupt Mask ($001,1, $001,3, $002,3, $022,1, $023,1)

External Interrupt Mask (IM0, IM1, IM2, IM3, IM4)	Interrupt Requests
0	Enable
1	Disable (masks)

Initial value 1, R/W



Notes: • '—': reserved bit

• SEM/SEMD and REM/REMD instructions cannot be used.

Figure 9. Port Mode Register A (PMRA: $02E)

Low Power Dissipation Mode

The MCU has two low power dissipation modes, standby mode, stop mode. The function of the low power dissipation mode is shown in table 8. CPU state transition between low power dissipation modes is shown in figure 10.

Table 8. Low Power Dissipation Mode

		Conditions						
Low Power Dissipation Mode	**Instruction**	**Oscillator circuit**	**Instruction Execution (ϕCPU*2)**	**Interrupt Function (ϕPER*3)**	**RAM**	**Register, Flag**	**Input/ Output Pin**	**Timer 1,2,3 Serial Interface 1, 2, A/D, IF counter, LCD,PLL (ϕPER*3)**
Standby Mode	SBY instruction	Active	Stop	Active	Retained	Retained	Retained	Active
Stop Mode	Stop instruction	Stop	Stop	Stop	Retained	Reset *1	R port: retained D port: high impedance	Stop

*1) The MCU recovers from STOP mode by Reset input. Refer to table 1 of section "Reset" for the contents of the flags and registers.
*2) ϕCPU: System clock
*3) ϕPER: Interrupt, peripheral function clock

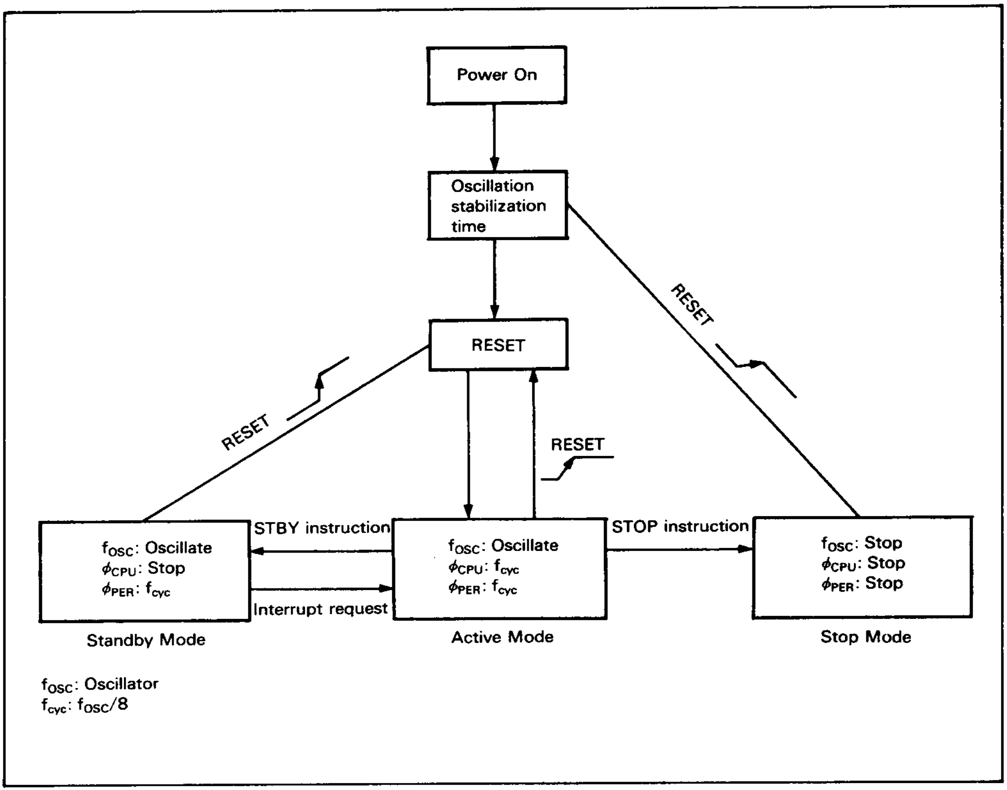


Figure 10. CPU State Transition

Active Mode: In active mode, the MCU operates depending on the clock generated from the OSC1 and OSC2 oscillator circuits.

Standby Mode: The MCU enters standby mode when a SBY instruction is excuted. In this mode, an oscillator, interrupt, timer, serial interface, A/D, LCD, PLL, and IF counter continue to operate, but all instruction-related clocks stop.
This in turn stops the CPU, retains all RAM and register contents, and maintains current I/O pin status.

The standby mode is terminated by a RESET or interrupt request. After an interrupt request, the MCU resumes by executing the next instruction following the SBY instruction. Then, if the interrupt enable flag is 1, the interrupt is processed. If the interrupt enable flag is 0, the interrupt request is left pending and normal instruction execution continues.

MCU operating flowchart in standby mode is shown in figure 11.

Stop Mode: The MCU enters stop mode when a stop instruction is executed. In this mode, the oscillator stops, causing all MCU functions to also stop.

The stop mode is terminated by a RESET input as shown in figure 12. Reset must be High for at least 20 ms to stabilize oscillation. During stop mode, all RAM contents are retained.

When the MCU resumes after stop mode, the accuracy of the contents of the accumulator, B register, W register, X/SPX register, Y/SPY register, carry, and serial data register cannot be guaranteed.

MCU Operation Sequence: The MCU operates according to the flowcharts shown in figures 13 and 14. The RESET is an asynchronous input, which resets the MCU regardless of the MCU state.

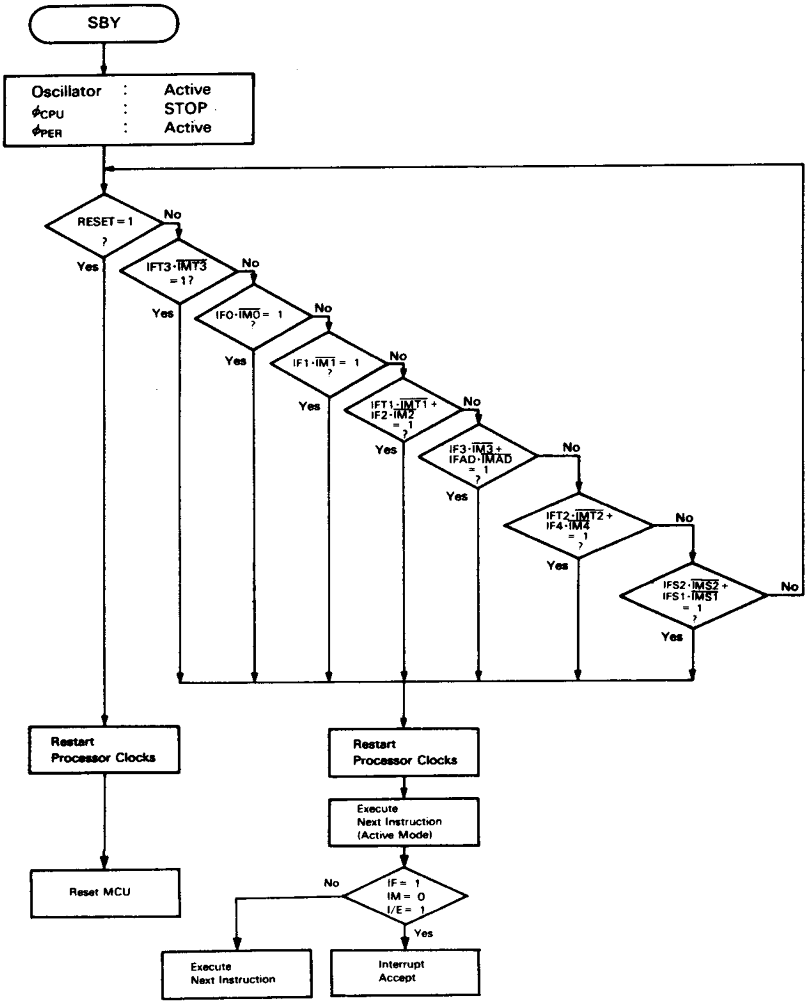

SBY
Oscillator : Active
ϕ_{CPU} : STOP
ϕ_{PER} : Active
RESET = 1 ?
No
Yes
$IFT3 \cdot \overline{IMT3} = 1$?
No
Yes
$IF0 \cdot \overline{IM0} = 1$?
No
Yes
$IF1 \cdot \overline{IM1} = 1$?
No
Yes
$IFT1 \cdot \overline{IMT1} + IF2 \cdot \overline{IM2} = 1$?
No
Yes
$IF3 \cdot \overline{IM3} + IFAD \cdot \overline{IMAD} = 1$?
No
Yes
$IFT2 \cdot \overline{IMT2} + IF4 \cdot \overline{IM4} = 1$?
No
Yes
$IFS2 \cdot \overline{IMS2} + IFS1 \cdot \overline{IMS1} = 1$?
No
Yes
Restart Processor Clocks
Reset MCU
Restart Processor Clocks
Execute Next Instruction (Active Mode)
IF = 1
IM = 0
I/E = 1
No
Yes
Execute Next Instruction
Interrupt Accept

Figure 11. A Standby Mode Flowchart

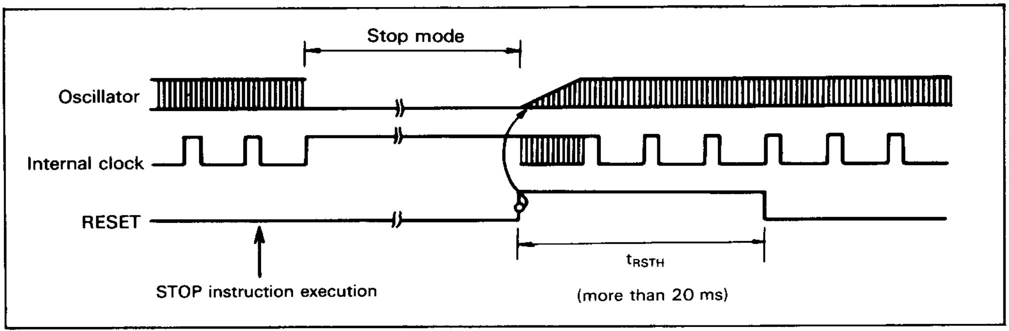


Figure 12. Timing Diagram When Canceling Stop Mode

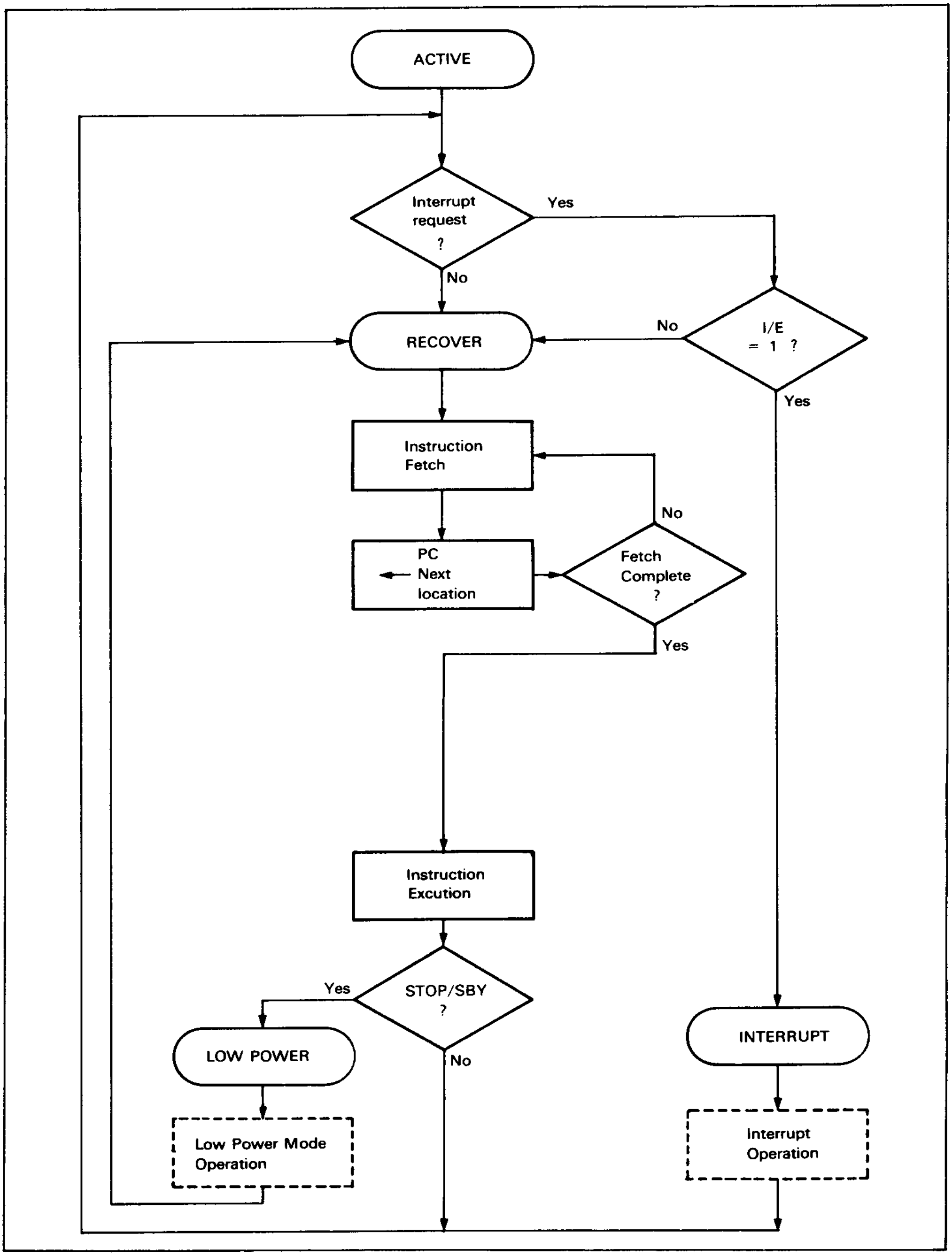


Figure 13. MCU Operation Sequence (MCU Operation Cycle)

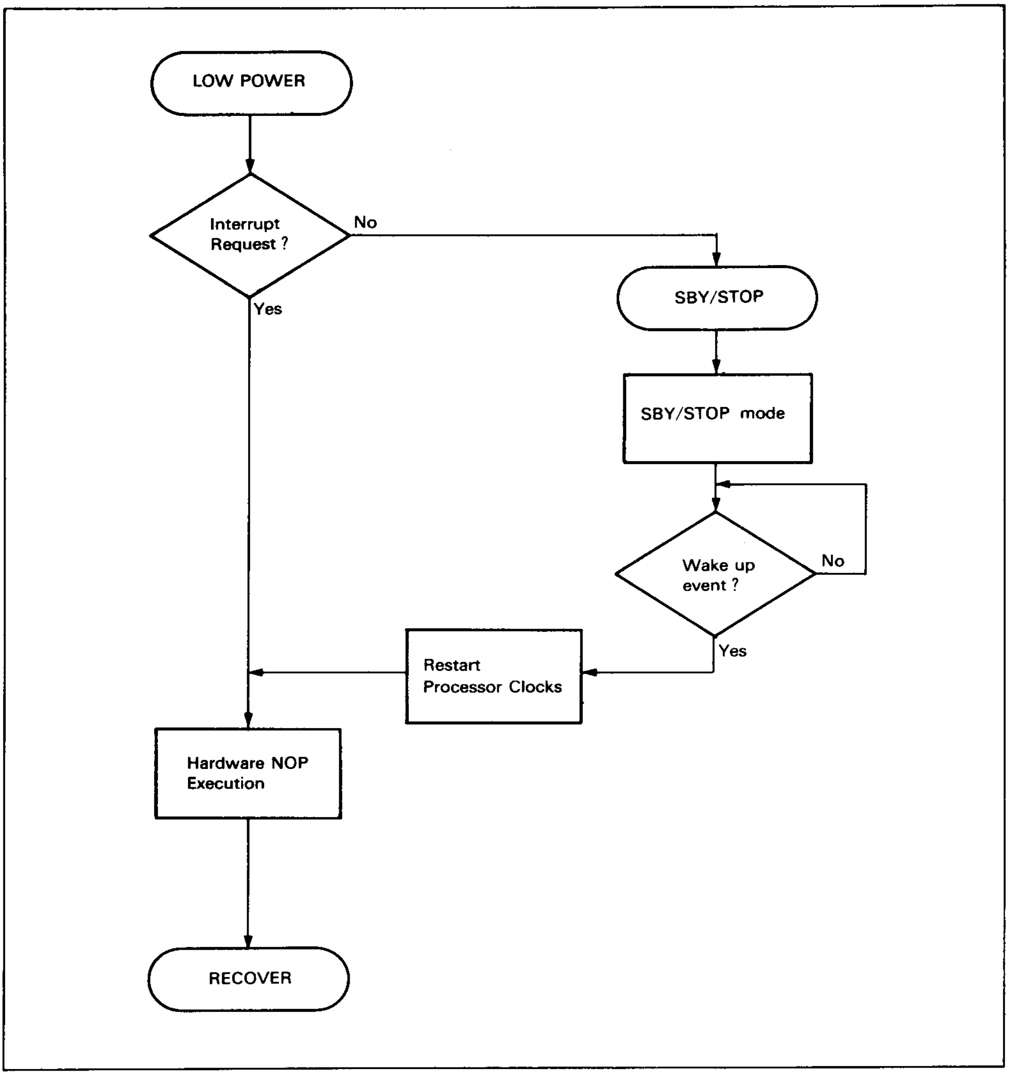


Figure 14. MCU Operation Sequence (Low Power Mode Operation)

PLL Function

The PLL function performs phase comparison between an external input signal and an internal reference signal, and outputs a comparison result signal. Block diagram is shown in figure 15.

PLL Features:

- 12 programmable reference frequencies (1, 2, 5, 6.25, 9, 10, 12.5, 18, 20, 25, 50, 100 kHz)
- Two dividing modes
  - —Pulse swallow mode
    divide ratio: 1024 to (2^{15} − 1)
  - —Direct mode
    divide ratio: 4 to (2^{15} − 1)
- Two independent comparison output signal pins
- PLL lock detection function

Reference frequency generator: The reference frequency generator divides source oscillation (4.5 MHz) to generate twelve different reference frequencies f_r: 1, 2, 5, 6.25, 9, 10, 12.5, 18, 20, 25, 50, and 100 kHz. Any of them can be selected by programming bit 0 to bit 3 of PLL control register A ($PCRA_0$-$PCRA_3$).

Variable divider: The variable divider is a down-counter composed of a 5-bit M counter and a 10-bit A counter. In direct dividing mode, the variable divider functions as a 15-bit down-counter with connected M and A counters to perform the auto-reload function. In pulse swallow dividing mode, both M and A counters function as down-counters which input 2-modulus prescaler (mode 1/33) output. When the M counter reaches 0, 2-modulus prescaler mode changes from 1/33 to 1/32, and the A counter continues to count. When the A counter reaches 0, the M and A counters are auto-reloaded and start counting again. The dividing mode (direct or pulse swallow) can be selected by programming bit 2 of PLL control register B ($PCRB_2$). Divide value should be set in the PLL data register . The value is set in the PLL reload register via the PLL data register. The contents of the PLL data register (15 bits) are loaded in the PLL reload register automatically by the write instruction to the lower four bits of the PLL data register (PDRA register). Therefore, data must be written in the registers in the following order: PDRD, PDRC, PDRB, and PDRA.

Phase comparator: The phase comparator detects phase difference between the reference frequency (f_r) and the output signal (f_v) from the variable divider. The result is output from the $\phi 1$ and $\phi 2$ pins, as shown in figure 16, figure 17.

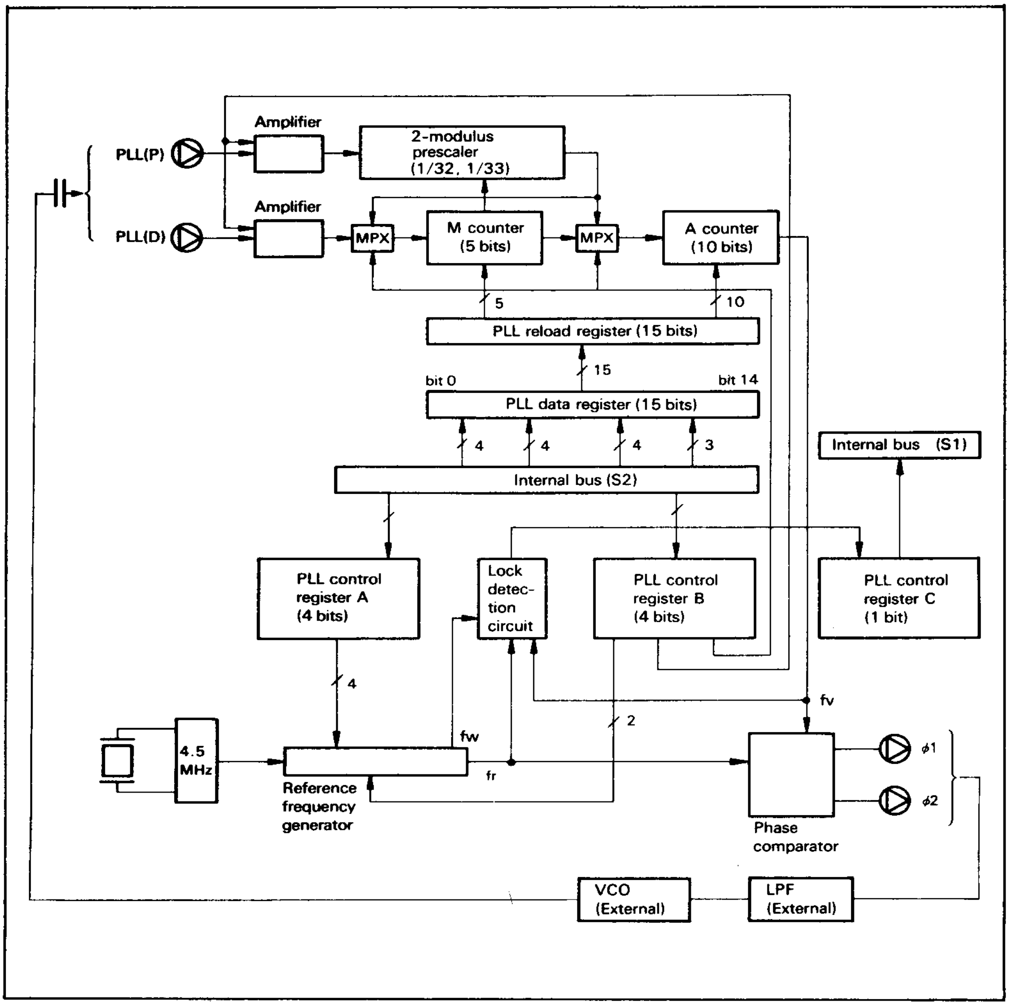


Figure 15. PLL Function Block Diagram

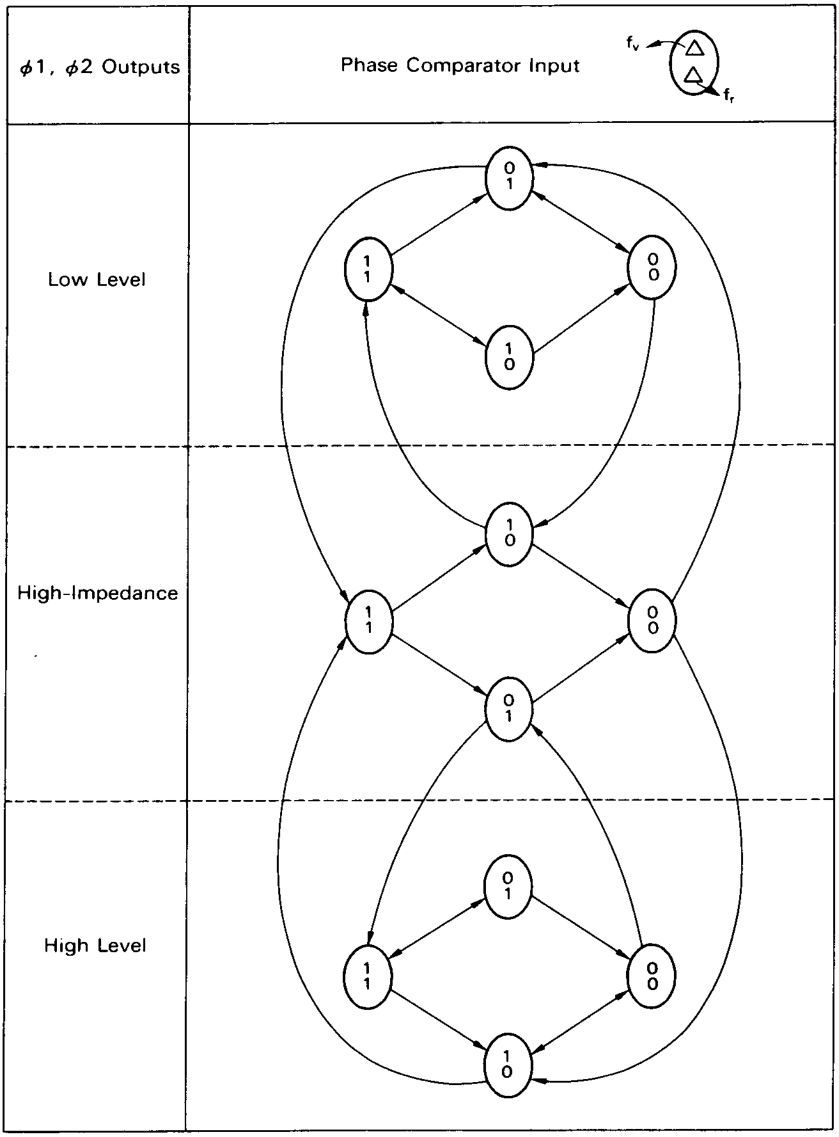


Figure 16. State Diagram (Phase Comparator)

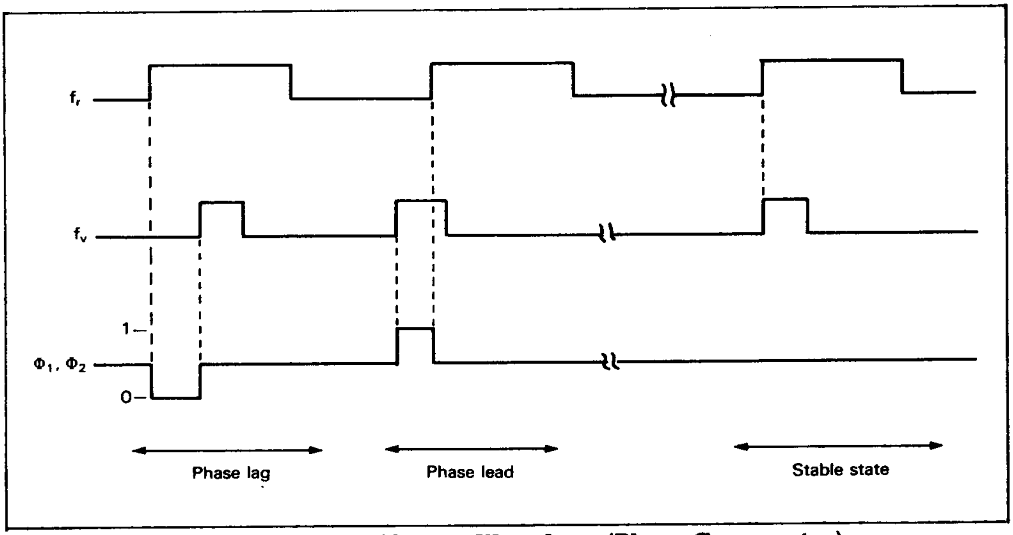


Figure 17. Input/Output Waveform (Phase Comparator)

Lock detection circuit: If the phase difference between the reference frequency (f_r) and a signal (f_v) generated by dividing an external input signal through the variable divider is less than a specified value $\pm\alpha$, PLL is assumed to be locked and bit 3 of the PLL control register C (PCRC$_3$) is set. The set condition of PCRC$_3$ is shown in figure 18. PCRC$_3$ is reset automatically after being read. Note: Value α varies with programming bit 0 and bit 1 of PLL control register B (PCRB$_0$, PCRB$_1$).
When entering the stop mode, set PCRB$_3$ to 0 and PLL is disabled. PLL (P) and PLL (D) are pulled down to the GND so that current is not consumed by the alternate amplifier.

PLL control register A (PCRA: $005): PLL control register A is a 4-bit write-only register (figure 19).

PLL control registers B, C (PCRB: $006, PCRC: $006): PLL control register B is a 4-bit write-only register. PLL control register C is a 1-bit read-only register(figure 20).

PLL data register A (PDRA: $007)
PLL data register B (PDRB: $008)
PLL data register C (PDRC: $009)
PLL data register D (PDRD: $00A): PLL data registers A, B and C are 4-bit write-only registers, and PLL data register D is a 3-bit write-only register (figure 21).

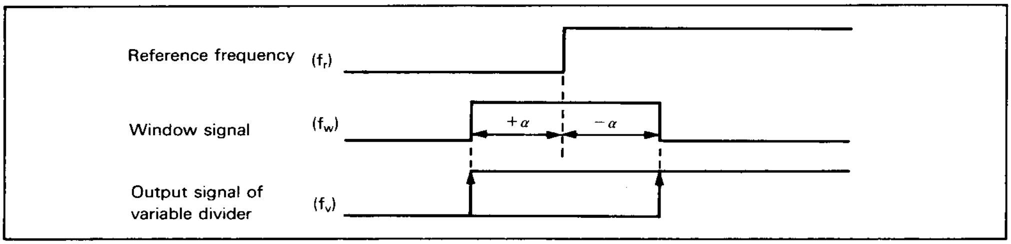
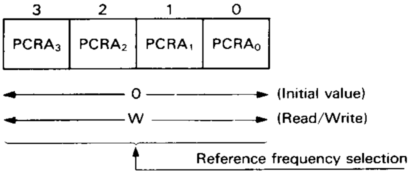


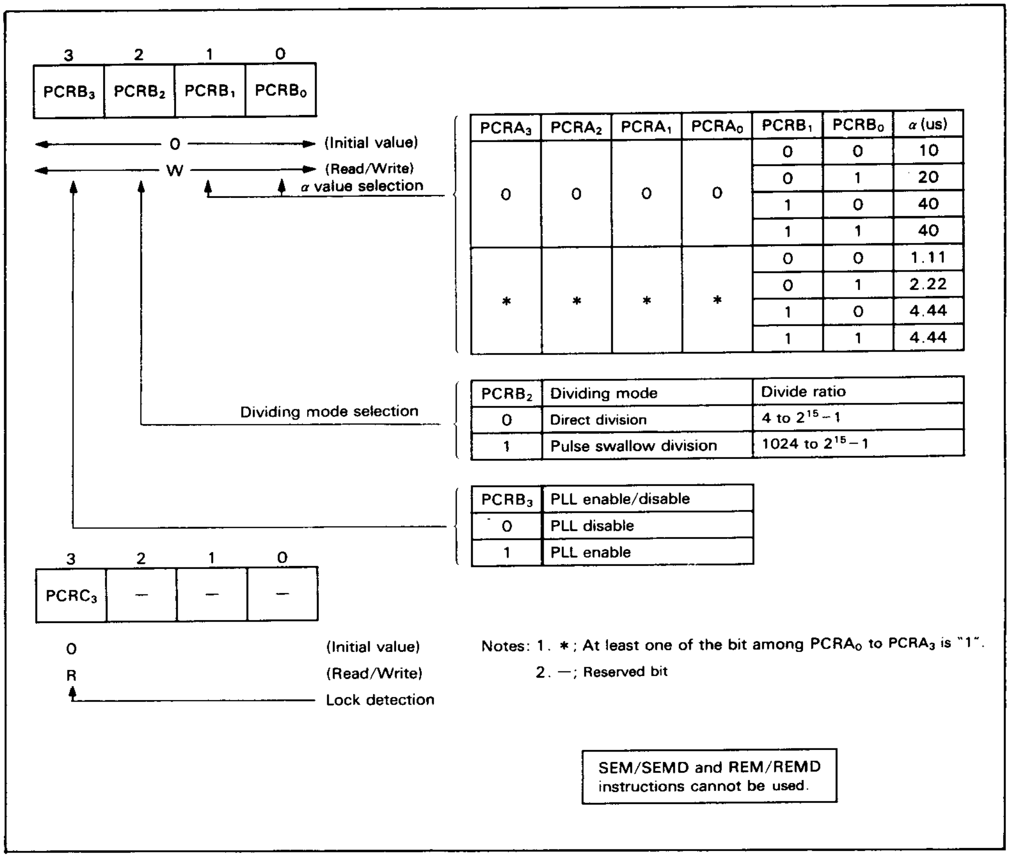
Figure 18. PCRC$_3$ SET Condition



PCRA$_3$	PCRA$_2$	PCRA$_1$	PCRA$_0$	Reference frequency (kHz)
0	0	0	0	1
0	0	0	1	5
0	0	1	0	6.25
0	0	1	1	9
0	1	0	0	10
0	1	0	1	12.5
0	1	1	0	25
0	1	1	1	50
1	0	0	0	2
1	0	0	1	10
1	0	1	0	12.5
1	0	1	1	18
1	1	0	0	20
1	1	0	1	25
1	1	1	0	50
1	1	1	1	100

SEM/SEMD and REM/REMD instructions cannot be used.

Figure 19. PLL Control Register A (PCRA: $005)



PCRA3	PCRA2	PCRA1	PCRA0	PCRB1	PCRB0	α (us)
0	0	0	0	0	0	10
				0	1	20
				1	0	40
				1	1	40
*	*	*	*	0	0	1.11
				0	1	2.22
				1	0	4.44
				1	1	4.44

PCRB2	Dividing mode	Divide ratio
0	Direct division	4 to $2^{15}-1$
1	Pulse swallow division	1024 to $2^{15}-1$

PCRB3	PLL enable/disable
0	PLL disable
1	PLL enable

Notes: 1. *; At least one of the bit among PCRA0 to PCRA3 is "1".

2. —; Reserved bit

SEM/SEMD and REM/REMD instructions cannot be used.

Figure 20. PLL Control Register B, C (PCRB: $006, PCRC: $006)

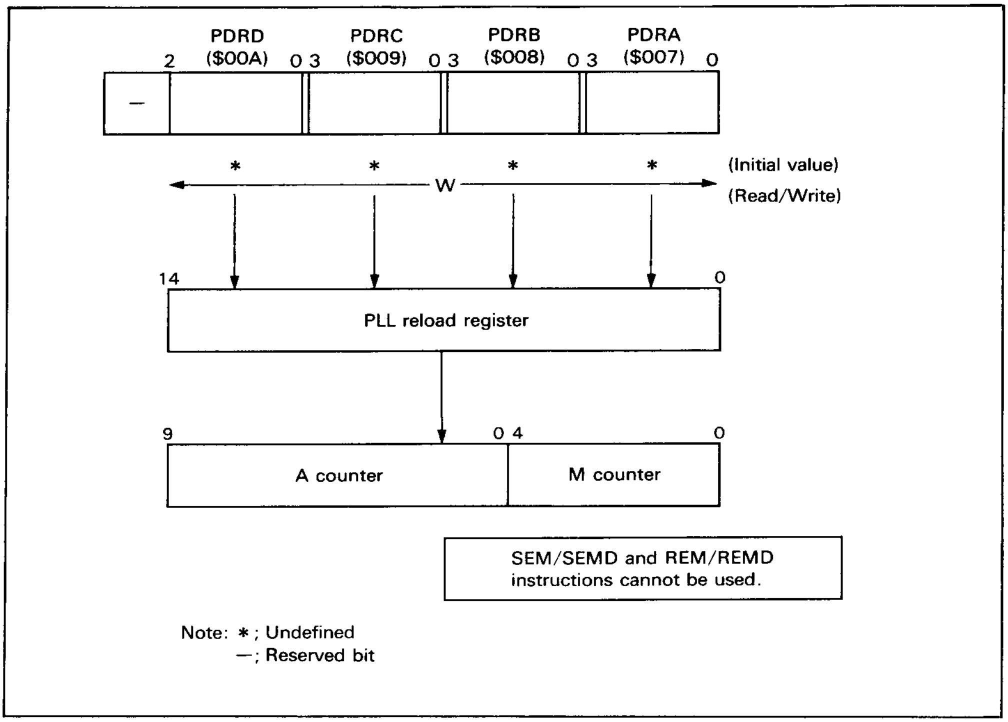


Figure 21. PLL Data Register and PLL Reload Register

IF Counter

IF Counter Features:

- Resolution: 16 bits
- Gate time: 1, 4, 8, ∞ ms
- Input signal: 15 MHz max.

IF Counter Function: The IF counter is a 16-bit binary counter whose data can be read. It is used to detect a stop signal during auto-search tuning; if a desired IF frequency range is found by counting the input frequency of the IF pin during auto-search tuning, it can be assumed that a radio station exists on the received frequency. The block diagram of the IF counter is shown in figure 22.

Count time (gate signal) of the IF counter can be selected from among 1, 4, 8 and ∞ ms by the IF counter control register. After all, the frequency input to the IF pin can be measured by counting the number of pulses input to the IF counter within the selected count time.

The IF counter is reset by clearing the IF clear start bit of the IF counter control register, and initiates counting by setting the IF clear start bit. When the IF clear start bit is set, the busy bit is set, and when counting is terminated, it is cleared. Accordingly, whether or not counting is being performed can be acknowledged by software. Note that when entering into the stop mode, $IFC1_3$ must be set to 0 and the IF counter must be disabled. The IF pin is pulled down to the GND, so that current is not consumed by the alternate amplifier.

IF conter control register 1 (IFC1: $028): IF counter control register 1 is a 4-bit write-only register (figure 23).

IF counter data register A (IFDA: $029)
IF counter data register B (IFDB: $02A)
IF counter data register C (IFDC: $02B)
IF counter data register D (IFDD: $02C): IF counter data registers A to D are 4-bit read-only registers. The most significant digit is IFDD, the other bits are, in order: IFDC, IFDB, and IFDA (figure 24).

IF counter control register 2 (IFC2: $028): IF counter control register 2 is a 1-bit read-only register (figure 25).

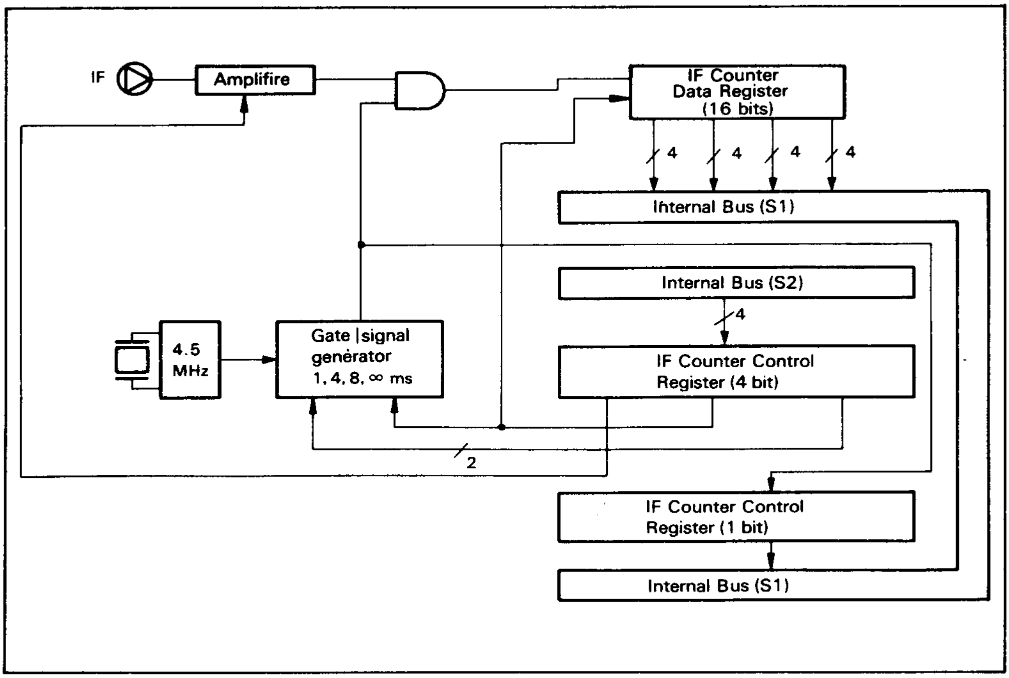

IF
Amplifire
IF Counter Data Register (16 bits)
4
4
4
4
Internal Bus (S1)
Internal Bus (S2)
4
IF Counter Control Register (4 bit)
4.5 MHz
Gate signal generator 1, 4, 8, ∞ ms
2
IF Counter Control Register (1 bit)
Internal Bus (S1)

Figure 22. IF Counter Block Diagram

Gate Signal: The IF counter's gate signal time is specified by the IF counter control register 1. The basic clock timing is a 1 ms pulse generated asynchronously with an instruction. Dividing this clock produces a 1, 4, or 8 ms gate signal. (Gate signal is selected among 1, 4, 8, ∞ ms.) Accordingly, even if the IF clear start bit of the IF counter control register 1 is set to initiate counting, the IF counter does not start counting until the first basic clock is generated after executing the current instruction. Time difference between the busy bit and the actual gate signal is max. 1 ms, independent of the specified gate time, since basic clock timing is 1 ms.

Note: Precautions when gate signal ∞ is selected.

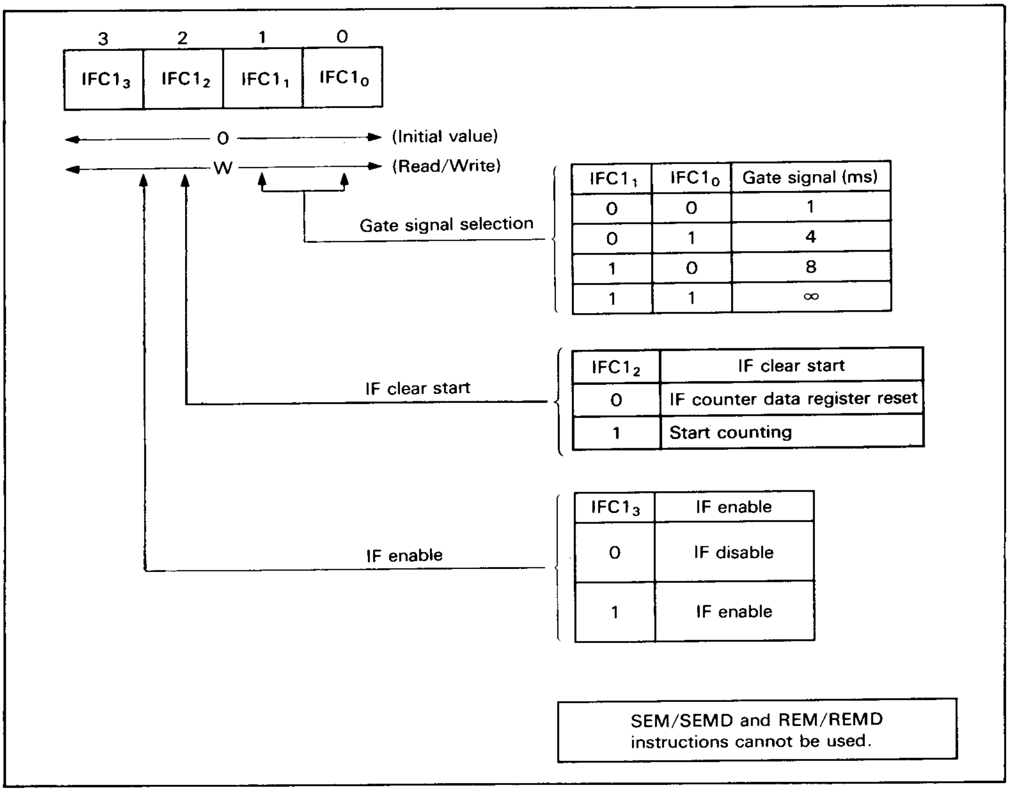
When gate signal ∞ is selected, the counter always operates by the IF input signal. Consequently, if data is read at counter update wrong data may be read. Thus, a gate must be closed once before data is read. To close a gate in the ∞ mode, a value other than ∞ should be set in the gate signal selection bits (bit 1, 0) during setting of the IF enable bit (bit 3) and IF clear start bit (bit 2) of the IF counter control register 1 to 1.

To continue counting after data read, the ∞ mode should be set again. Make certain that the counter does not operate while the gate is being closed.

Accordingly, gate closure time should be considered when counting continues after data read.

When gate signal ∞ is selected, the IF counter does not start counting until the first basic clock is generated after executing the current instruction (max. 1 ms) even if the IF clear start bit of the IF counter control register is set to initiate counting.

Busy signal is invalid for the gate signal ∞ mode.



$IFC1_1$	$IFC1_0$	Gate signal (ms)
0	0	1
0	1	4
1	0	8
1	1	∞

$IFC1_2$	IF clear start
0	IF counter data register reset
1	Start counting

$IFC1_3$	IF enable
0	IF disable
1	IF enable

SEM/SEMD and REM/REMD instructions cannot be used.

Figure 23. IF Counter Control Register 1 (IFC1: $028)

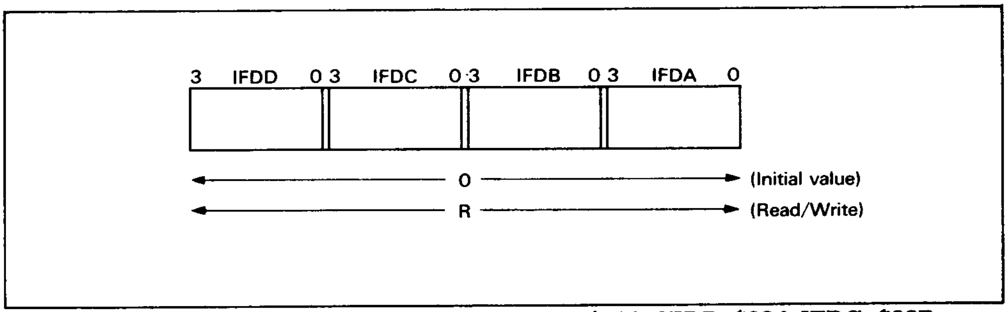


Figure 24. IF Counter Data Register (IFDA: $029, IFDB: $02A,IFDC: $02B, IFDD: $02C)

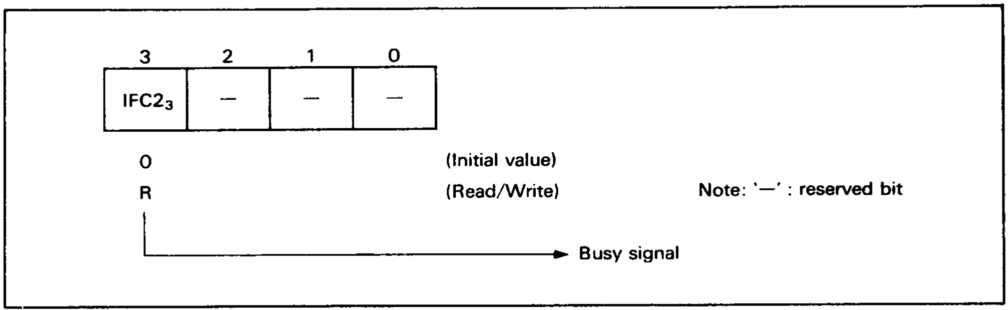


Figure 25. IF Counter Control Register 2 (IFC2: $028)

Serial Interface

The serial interface is used to transmit/receive 8-bit data serially. The MCU has two serial interfaces, serial 1 and serial 2. The serial interfaces consist of the serial data registers (SR1, SR2), the serial mode registers (SMR1, SMR2), the serial clock registers (SCR1, SCR2), the octal counter, and the multiplexer as illustrated in figure 26.

Pins $D_8/\overline{SCK_1}$, $D_{11}/\overline{SCK_2}$, D_9/SI_1, D_{12}/SI_2, D_{10}/SO_1, D_{13}/SO_2, and STS instruction are controlled by the serial mode register. The serial clock register controls the transfer clock. The contents of the serial data register can be written into or read out by software. The data in the serial data register can be shifted synchronously with the transfer clock signal.

Serial interface operation is initiated by the STS instruction while the enable bit of the target serial interface (bit 3 of the serial mode register of each serial interface) is set. The octal counter is reset to $0 by the STS instruction. The counter starts to count at the falling edge of the transfer clock signal ($\overline{SCK_1}$, $\overline{SCK_2}$) and increments by one at the rising edge of the signals.

When the octal counter is reset to $0 after eight transfer clock signals, or when a transmit/receive operation is discontinued by resetting the octal counter, the serial interrupt request flag will be set.

To control two serial interfaces by executing the STS instruction once, an enable bit (bit 3) of the serial mode register is provided for each serial interface. When the STS instruction is executed, the serial interface of which an enable bit is set starts to operate. If both enable bits are reset, the serial interface does not operate even if the STS instruction is executed. If both enable bits are set, two serial interfaces can be operated by executing the STS instruction once.

The enable bit is reset after the STS instruction execution automatically.

Consequently, to operate the serial interface, the STS instruction should be executed after setting an enable bit.

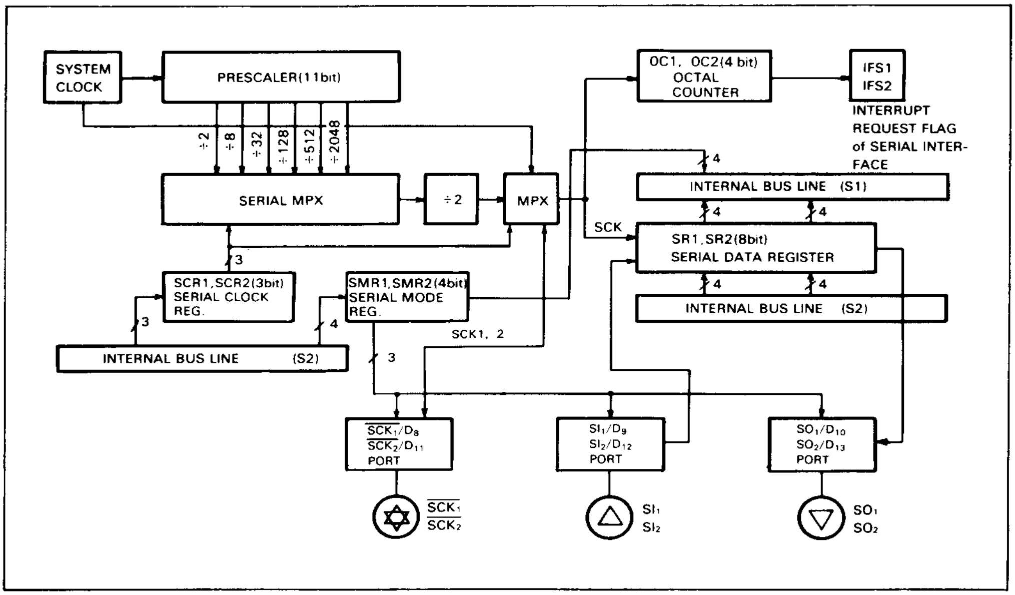


Figure 26. Serial Interface Block Diagram

Serial Mode Registers (SMR1: $010, SMR2: $014): The 4-bit write-only serial mode registers control the serial interface operation and the pins $\overline{SCK_1}$, $\overline{SCK_2}$, SI_1, SI_2, SO_1, and SO_2 as shown in figure 27.

The write signal to the serial mode register initializes the operating state of the serial interface.

The write signal to the serial mode register stops the serial data register and octal counter from applying transfer clock, and it also resets the octal counter to $0 simultaneously. Therefore, when the serial interface is in the "Transfer State", the write signal causes the serial mode register to cease the data transfer and to set the serial interrupt request flag.

The contents of the serial mode register is invalid until the second instruction is executed after a write instruction. Therefore, it will be necessary to execute the STS instruction after the data in the serial mode register has been changed completely. The serial mode register will be reset to $0 by MCU reset.

Bit 3 of the serial mode register is an enable bit for the serial interface. To operate the serial interface, this bit should be set before executing the STS instruction.

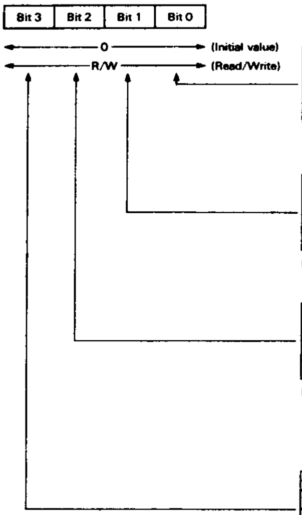
If the STS instruction is executed while the enable bit is set, the serial interface starts to operate. If the STS instruction is executed while the enable bits of both serial interfaces are set, two serial interfaces start to operate. The internal clocks of SERIAL 1 and SERIAL 2 are asynchronous each other.

The enable bit is automatically reset after the STS instruction execution.

Serial Clock Registers (SCR1: $011, SCR2: $015): The serial clock register is a 3-bit write-only register which controls the transfer clock source and prescaler divide ratio as shown in figure 28. A write signal to the serial clock register initializes internal state of the serial interface. A write signal to the serial clock register stops the serial data register and octal counter from applying transfer clock, and it resets the octal counter. Therefore, when the serial interface is in the transfer state, the write signal to the serial clock register stops the data transfer and sets the serial interrupt request flag.

Serial Data Registers (SR1L: $012, SR1U: $013, SR2L: $016, SR2U: $017): The 8-bit read/write serial data register consists of a lower digit (SR1L: $012, SR2L: $016) and a higher digit (SR1U: $013, SR2U: $017).

The data in the serial data register will be output from the SO_1 and SO_2 pins, from LSB to MSB, synchronously with the falling edge of the transfer clock. At the same time, external data will be input from the SI_1 and SI_2 pins to the serial data register, to LSB first, synchronously with the rising edge of the transfer clock. Figure 29 shows the I/O timing chart for the transfer clock signal and the data. The read/write operations of the serial data register should be performed after the completion of data transmit/receive. Otherwise the data may not be guaranteed.



SMR1,SMR2	D_{10}/SO_1, D_{13}/SO_2
Bit 0	
0	I/O pins D_{10}, D_{13}
1	Output pins SO_1, SO_2

Note: When bit 0 of SMR1 or SMR2 is set, D_{10} or D_{13} becomes output pin automatically.

SMR1,SMR2	D_9/SI_1, D_{12}/SI_2
Bit 1	
0	I/O pins D_9, D_{12}
1	Input pins SI_1, SI_2

Note: When bit 1 of SMR1 or SMR2 is set. D_9 or D_{12} becomes input pin automatically.

SMR1,SMR2	$D_8/\overline{SCK_1}$, $D_{11}/\overline{SCK_2}$
Bit 2	
0	I/O pins D_8, D_{11}
1	Input or output pins $\overline{SCK_1}$,$\overline{SCK_2}$

Note: When bit 2 of SMR1 or SMR2 is set. if internal Clock is selected, D_8 or D_{11} becomes $\overline{SCK_1}$ or $\overline{SCK_2}$ output automatically. If external clock is selected at this time. these pins become $\overline{SCK_1}$ or $\overline{SCK_2}$ input.

SMR1,SMR2	STS instruction
Bit 3	
0	Disabled (not affect operating state)
1	Enabled (transfer clock wait state)

Note: Bit 3 of SMR1 and SMR2 is reset automatically after the STS instruction execution.

SMR1,SMR2				Serial interface operation mode
Bit 3	Bit 2	Bit 1	Bit 0	
1	1	0	0	Clocke continuous output mode
1	1	0	1	Transmit mode
1	1	1	0	Receive mode
1	1	1	1	Transmit/receive mode

Figure 27. Serial Mode Register (SMR1: $010, SMR2: $014) and Operation Mode Selection

Table 9. Serial Interrupt Request Flag (IFS2: $003,2, IFS1: $023,2)

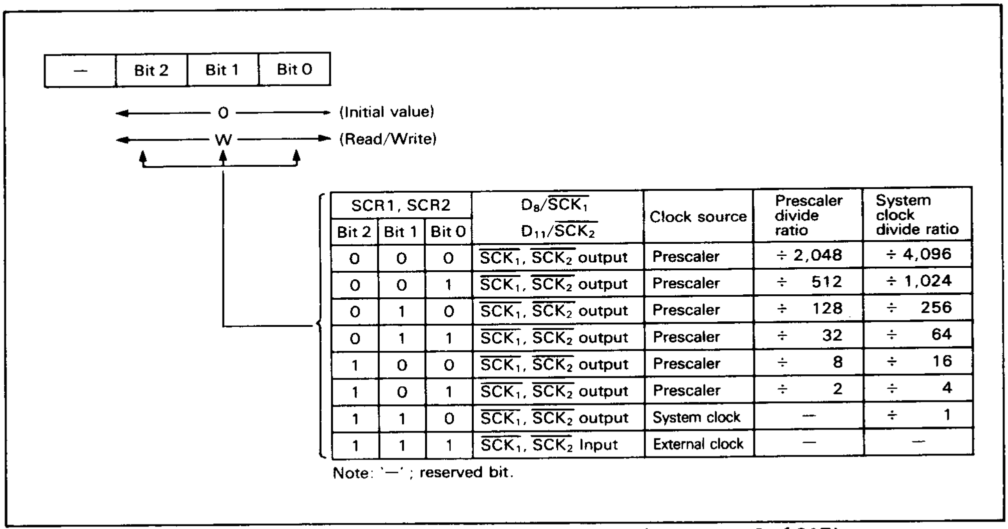
Serial Interrupt Request Flag	Interrupt Requests
0	No
1	Yes

Initial value: 0, R/W
Note: Reset by the REM and REMD instructions. 1 cannot be set by software. Tested by the TM and TMD instructions.

Table 10. Serial Interrupt Mask (IMS2: $003, 3, IMS1: $023, 3)

Serial interrupt mask	Interrupt Requests
0	Enabled
1	Masked (held)

Initial value: 1, R/W
Note: REM, REMD, SEM, SEMD, TM, and TMD instructions can be used.



SCR1, SCR2			$D_8/\overline{SCK_1}$ $D_{11}/\overline{SCK_2}$	Clock source	Prescaler divide ratio	System clock divide ratio
Bit 2	Bit 1	Bit 0				
0	0	0	$\overline{SCK_1}$, $\overline{SCK_2}$ output	Prescaler	÷ 2,048	÷ 4,096
0	0	1	$\overline{SCK_1}$, $\overline{SCK_2}$ output	Prescaler	÷ 512	÷ 1,024
0	1	0	$\overline{SCK_1}$, $\overline{SCK_2}$ output	Prescaler	÷ 128	÷ 256
0	1	1	$\overline{SCK_1}$, $\overline{SCK_2}$ output	Prescaler	÷ 32	÷ 64
1	0	0	$\overline{SCK_1}$, $\overline{SCK_2}$ output	Prescaler	÷ 8	÷ 16
1	0	1	$\overline{SCK_1}$, $\overline{SCK_2}$ output	Prescaler	÷ 2	÷ 4
1	1	0	$\overline{SCK_1}$, $\overline{SCK_2}$ output	System clock	—	÷ 1
1	1	1	$\overline{SCK_1}$, $\overline{SCK_2}$ Input	External clock	—	—

Note: '—' ; reserved bit.

Figure 28. Serial Clock Register (SCR1: $011, SCR2: $015)

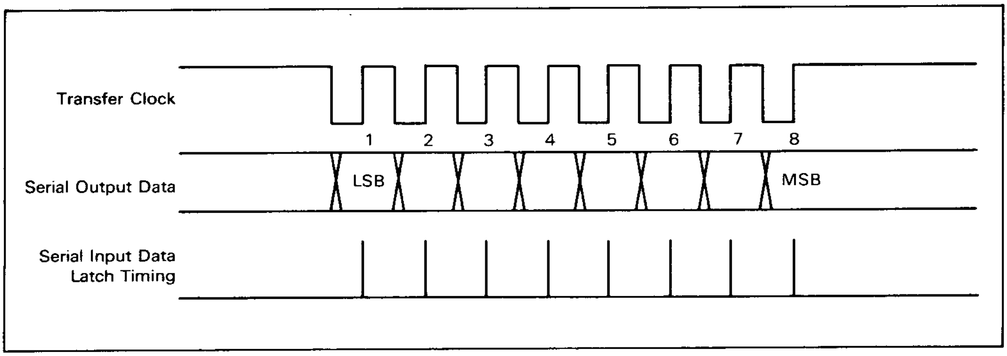


Figure 29. Serial Interface I/O Timing Chart

Serial Interrupt Request Flag (IFS1: $023 bit 2, IFS2: $003 bit 2): The serial interrupt request flag is set when the octal counter counts eight transfer clock signals, or when data transfer is discontinued by resetting the octal counter. See table 9.

Serial Interrupt Mask (IMS1: $023 bit 3, IMS2: $003 bit 3): The serial interrupt mask masks the interrupt request. See table 10.

Selection and Change of the Operation Mode: The serial interface operation modes which are determined by a combination of the value in the serial mode register are shown in figure 27.

Operating State of Serial Interface: The serial interface has four operating states, transfer disable state, STS waiting state, transfer clock waiting state, and transfer state, as shown in figure 30.

The transfer disable state is the initialization state of the serial interface. In this state, the serial interface does not operate even if the STS instruction is executed or the transfer clock is applied. Setting the enable bit of the desired serial interface enters it into STS waiting state.

When the STS instruction is executed in this state, the serial interface becomes transfer clock waiting state and serial transfer is enabled. When both enable bits of the serial interface 1 and 2 are set before executing the STS instruction, two serial interfaces start to operate. If the transfer clock is applied during the transfer clock waiting state, the serial interface changes to transfer state, while the octal counter counts up and the serial data register changes simultaneously. However, if the consecutive clock output mode is selected, only transfer clock is output consecutively without transfer.

In the transfer state, the octal counter becomes 000 by 8 transfer clocks and the serial interrupt flag is set. At this time, if internal clock is selected, the serial interface changes to the transfer disable state, and if external clock is selected, it changes to the transfer clock waiting state. A write signal to the serial mode register during the transfer state resets the octal counter to 000 to stop transfer and sets the interrupt request flag simultaneously.

Example of Transfer Clock Error Detection: The serial interface malfunctions when the transfer clock is disturbed by external noises. In this case, transfer clock error can be detected by the procedure shown in figure 31. If more than 8 transfer clocks are applied in the transfer clock waiting state, the state of the serial interface changes as the following sequence: first, transfer state, second, transfer clock waiting state, and third, transfer state again. The serial interrupt flag should be reset before entering into the STS waiting state by writing data to the serial mode register. This procedure causes the serial interrupt request flag to be set again.

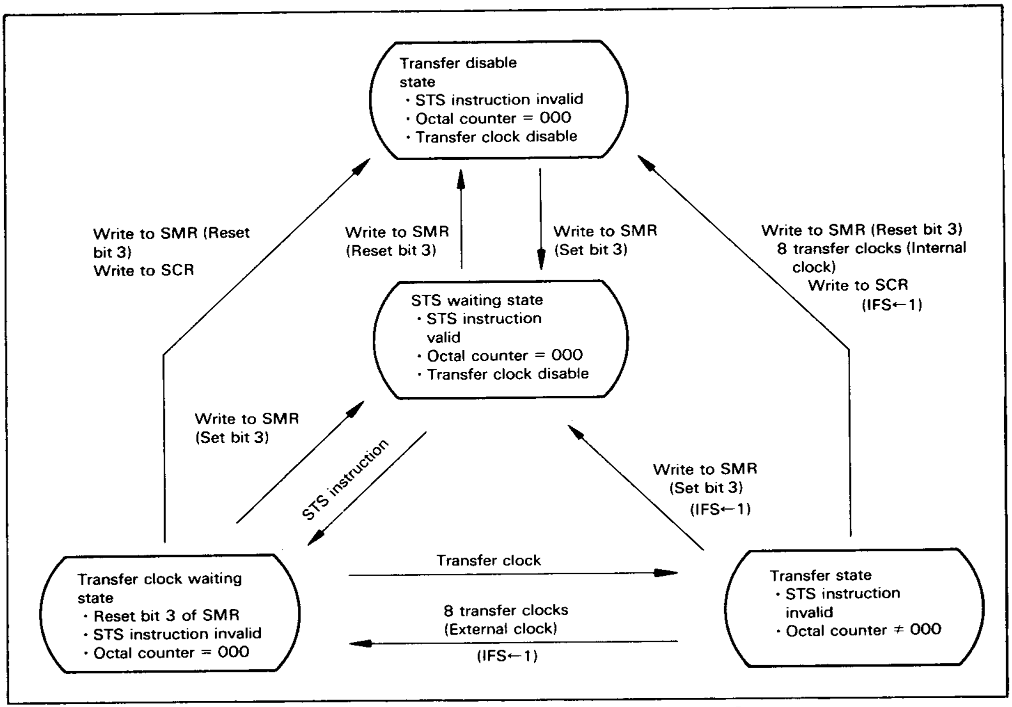


Figure 30. Operating State of Serial Interface

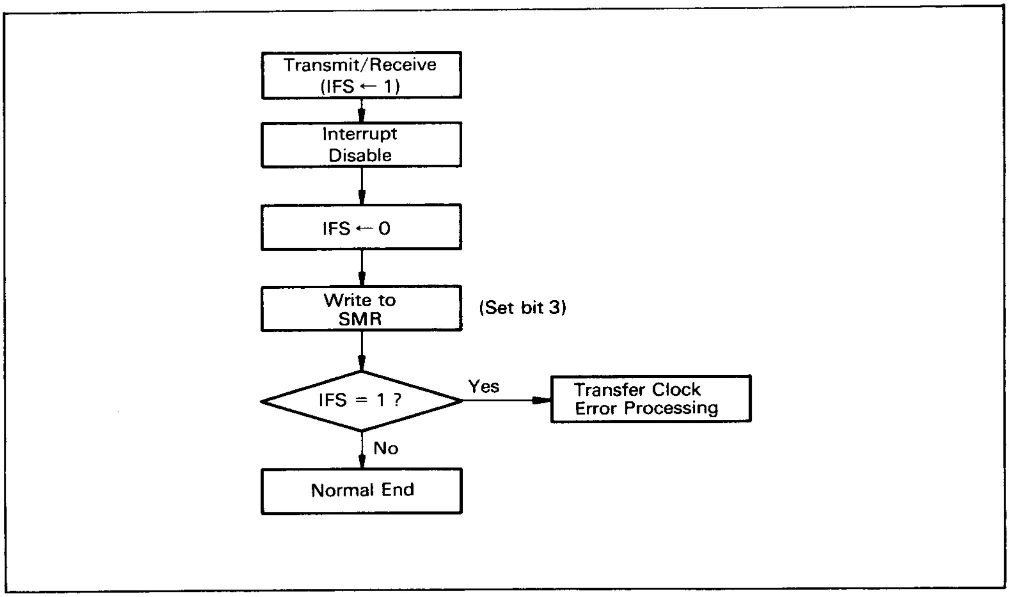


Figure 31. Example of Transfer Clock Error Detection

Timer

The MCU contains a prescaler, two timer/counters (timer-1, 2) and time-base prescaler (timer 3).Figures 32, 33 and 34 show the block diagrams of the timers.

Prescaler: The input to the prescaler is a system clock signal. The prescaler is initialized to $000 by MCU reset before dividing the system clock. The prescaler keeps counting up except in MCU reset and stop mode. Timer input clock and serial transfer clock are selected by the timer mode registers (TMR1, TMR2) and serial clock registers (SCR1, SCR2) among prescaler outputs.

Operation of Timer 1, Timer 2: Timer 1 and 2 are multifunctional timers. They can be used as a free-running timer, event counter, reload timer, or PWM circuit (Timer 2 does not have a PWM function). Each function is selected by the timer mode register (TMR1, TMR2) and timer output register (TOR1, TOR2).
The timer/counter (TC1, TC2) is initialized to $00 by MCU reset, and it starts to count up every input signal. As an input clock, the clock divided by the prescaler and an external clock can be used. When an external clock input is selected, external interrupt must be inhibited by setting the external interrupt mask bit (IM0 or IM1) since the external clock input pin is multiplexed with the external interrupt pin. (External interrupt cannot be inhibited by setting the interrupt enable bit of the timer output register.) If an input clock is applied after the timer counter reaches $FF, the timer interrupt request flag (IFT1, IFT2) is set. If an auto-reload function is not selected, timer 1 and 2 function as free-running timer/event counter and continue to count up after returning to $00. If the auto-reload function is selected, they are reload timers.
The timers initiate counting up from the timer load register (TLR1, TLR2) value loaded in the timer counter for each clock input after the timer counter reaches $FF.
Timer 1 has a timer output circuit. This circuit changes output level when a clock is input after the timer counter reaches $FF. An optional cycle clock signal can be output by combining this circuit and the reload timer. Also this output circuit becomes PWM output circuit if PWM function is specified by the timer output register. One cycle of the PWM output is 256 input clocks. The output becomes 1 during the time for the number of clocks set in the timer load register, and becomes 0 during the other time.
To obtaine the voltage level in proportion to the value set in the timer load register, PWM output should be combined with low-pass filter.

Timer 3 Operation: Timer 3 is a prescaler which sets interrupt request flag (IFT3) every 125 ms.

- Timer interrupt request flag (IFT3: $000 bit 2)
- Timer interrupt mask (IMT3: $000 bit 3)

The timer interrupt mask masks the timer 3 interrupt request.

Timer 3 is initialized in the following condition.

- Power-on Reset
- MCU Reset when the data retention bit (MIS$_3$: $004,3) is 0.

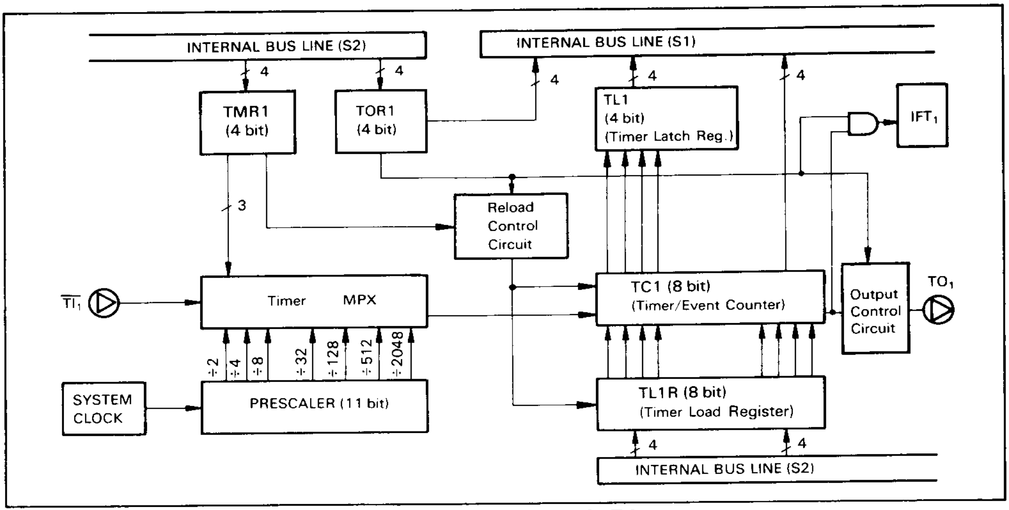


Figure 32. Timer 1 Block Diagram

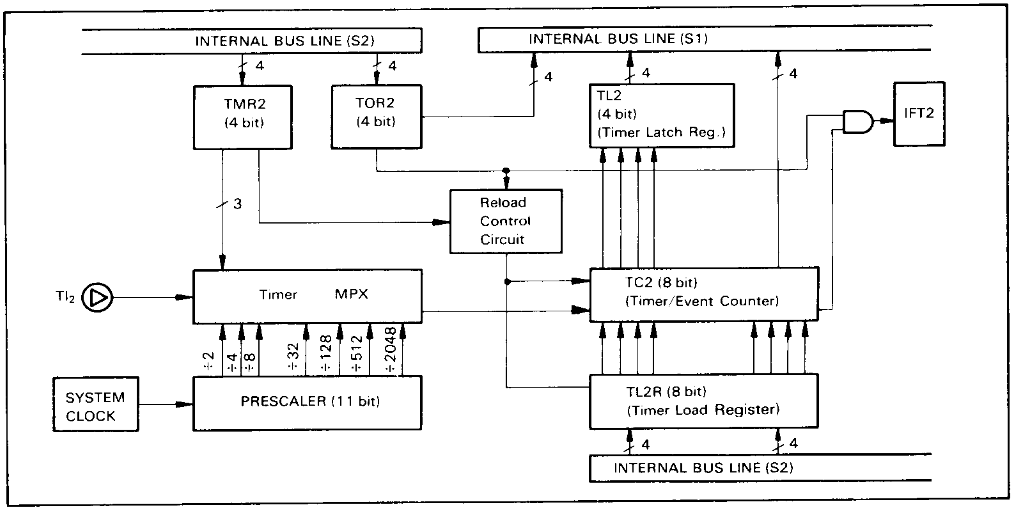


Figure 33. Timer 2 Block Diagram

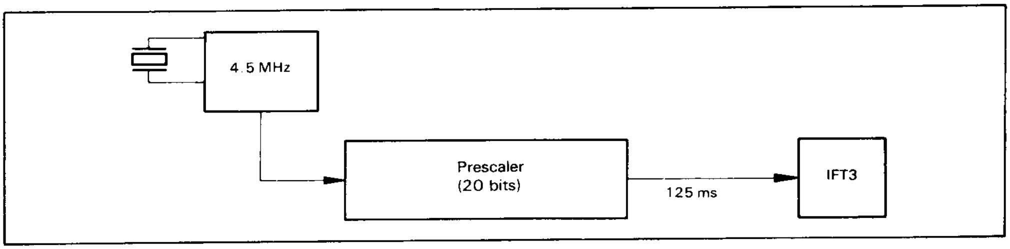


Figure 34. Timer 3 Block Diagram

Timer Mode Register 1, 2 (TMR1: $018, TMR2: $01C): The timer mode register is a 4-bit write-only register which selects the autoreload function, input clock, and prescaler divide ratio as shown in figure 35.

The timer mode register is initialized to $0 by MCU reset.

The operation mode of the timer mode register is changed at the second instruction cycle after the timer mode register is written to. Initialization of the timer by a write to the timer mode register should be performed after the operation mode is completely changed.

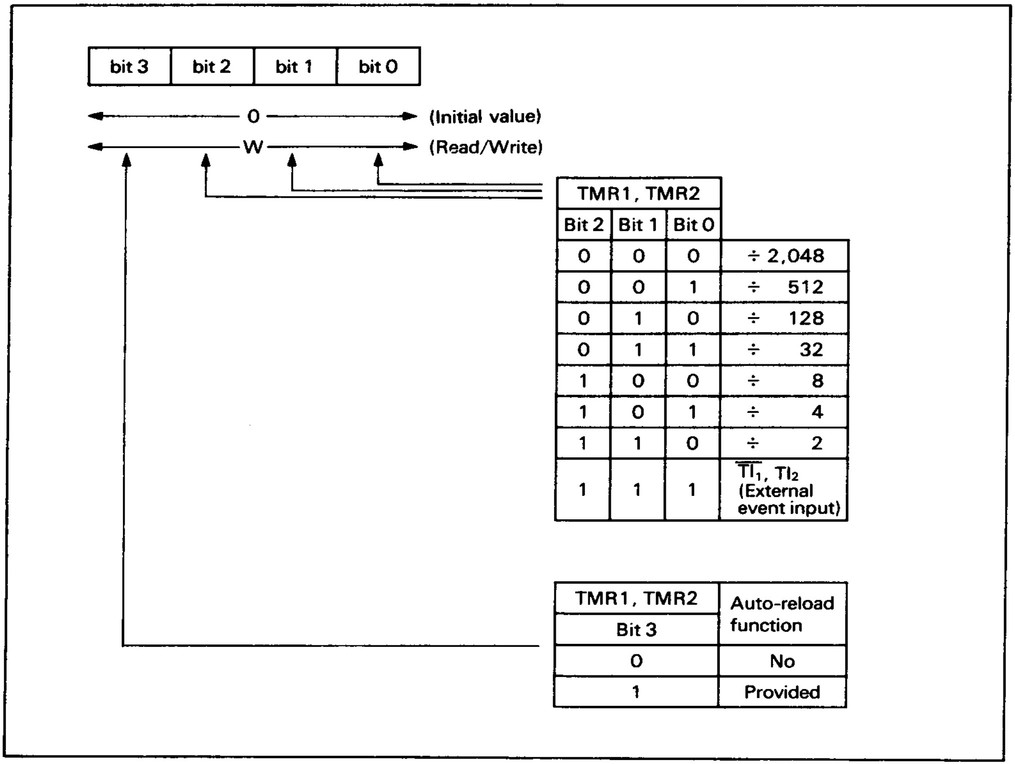
Timer Output Register (TOR1: $019, TOR2: $01D): The timer output register is a 4-bit read/write register which controls the timer output mode, PWM output, and external interrupts as shown in figure 36.

If bit 0 and 1 of the timer output register are set in the mode except for timer output disable mode, the pin becomes timer output pin automatically.

Combination of these modes and each mode of the timer 1 to 3 can produce optional frequency and optional duty clock signal. When PWM output is set, the timer output pin becomes PWM output pin independently of the timer output mode.

The timer output register is initialized to $0 by MCU reset.

The operation mode of the timer output register is changed at the second instruction cycle after the timer output register is written to.



TMR1, TMR2			
Bit 2	Bit 1	Bit 0	
0	0	0	÷ 2,048
0	0	1	÷ 512
0	1	0	÷ 128
0	1	1	÷ 32
1	0	0	÷ 8
1	0	1	÷ 4
1	1	0	÷ 2
1	1	1	$\overline{TI_1}$, TI_2 (External event input)

TMR1, TMR2	Auto-reload function
Bit 3	
0	No
1	Provided

Figure 35. Timer Mode Register 1, 2 (TMR1: $018, TMR2: $019)

Timer 1, 2 (TC1L: $01A, TC1U: $01B, TL1L: $01A, TL1U: $01B, TC2L: $01E, TC2U: $01F, TL2L: $01E, TL2U: $01F): Timer 1 and 2 consist of write-only timer load register (8 bits) and read-only timer/event counter (8 bits) respectively. Each of them has a lower digit (TC1L: $01A, TC2L: $01E) and a higher digit (TC1U: $01B, TC2U: $01F).

The timer/event counter can be initialized by writing data into the timer load register. In this case, write the lower digit first, and then the higher digit. The timer load register is initialized to $00 by the MCU reset.

The counter value of the timer can be obtained by reading the timer/event counter. In this case, read the higher digit first, and then the lower digit. The count value of the lower digit is latched at the time when the higher digit is read.

Timer Interrupt Request Flag (IFT1: $002 bit 0, IFT2: $003 bit 0): The timer interrupt request flag is set by the overflow output of timer 1 or timer 2. If the PWM function is selected, the timer interrupt request flag is not set even if the overflow output generates.

See table 11.

Timer Interrupt Mask (IMT1: $002 bit1, IMT2: $003 bit 1): The timer interrupt mask masks an interrupt request from timer 1 or timer 2.

See table 12.

Table 11. Timer Interrupt Request Flag

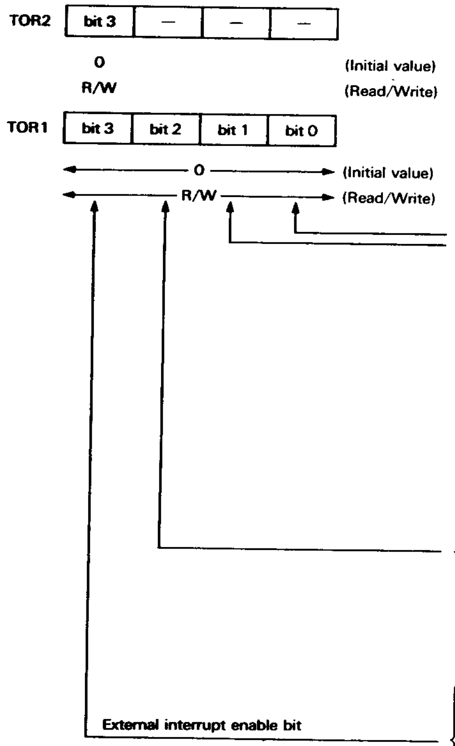
Timer Interrupt Request Flag	Interrupt Request
0	No
1	Yes

Initial value: 0, R/W

Table 12. Timer Interrupt Mask

Timer Interrupt Mask	Interrupt Request
0	Enabled
1	Disabled (masks)

Initial value: 1, R/W



TOR1		Timer Output Mode
Bit 1	Bit 0	
0	0	Timer output disabled
0	1	Outputs reverse level signal synchronously with the input clock after TC reactes $FF (Toggle output mode)
1	0	Output 0 synchronously with the input clock after TC reaches $FF (0 output mode)
1	1	Outputs 1 synchronously with the input clock after TC reaches $FF (1 output mode)

Note: When bit 1 and bit 0 of TOR1 are set to other than '00', $R3_2$ becomes output automatically.

TOR1	PWM Function
Bit 2	
0	Normal timer operation
1	PWM operation Output from TO_1 pin

Note: When bit 2 of TOR1 is set, $R3_2$ becomes output automatically.

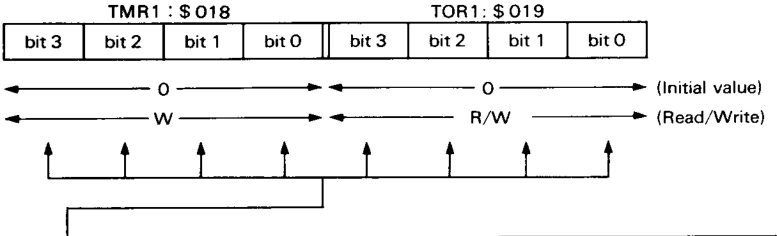
TOR1, TOR2	External Interrupts
Bit 3	$\overline{\text{INT0}}$, INT1
0	$\overline{\text{INT0}}$, INT1 disabled
1	$\overline{\text{INT0}}$, INT1 enabled

Note: When an external event input is specified by bits 0–3 of the timer mode register, an external interrupt is enabled irrespectively of the value of bit 3 of TOR.
'—' : reserved bit

Figure 36. Timer Output Register

Operating Mode of Timer 1, Timer 2: Since timer 1 and timer 2 can select several functions as free-running, reload, PWM (timer 1 only), and can control clock input and timer output, various operating states can be specified by combination of these functions as shown in figure 37, 38. Required data should be set in the timer mode register and timer output register for the selected operating state.

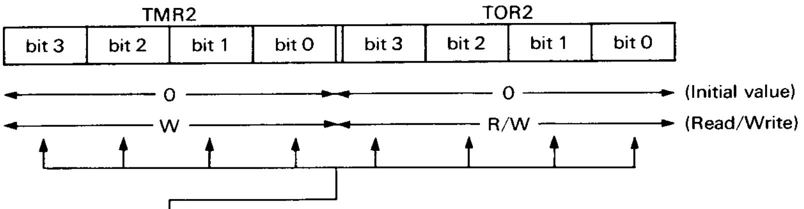
Example of the timer output waveform is shown in figure 39. The waveform depends on the operation mode.



TMR1				TOR1				Timer input pin	Timer output pin	Timer 1 Function
Bit 3	Bit 2	Bit 1	Bit 0	Bit 3	Bit 2	Bit 1	Bit 0			
0	*3	*3	*3	0	0	0	0	$R4_0$	$R3_2$	Free-running timer
0	*3	*3	*3	0	0	*2	*2	$R4_0$	TO_1	
0	*3	*3	*3	1	0	0	0	$\overline{INT_0}$	$R3_2$	
0	*3	*3	*3	1	0	*2	*2	$\overline{INT_0}$	TO_1	
0	1	1	1	*1	0	0	0	$\overline{TI_1}$, $\overline{INT_0}$*5	$R3_2$	Event counter
0	1	1	1	*1	0	*2	*2	$\overline{TI_1}$, $\overline{INT_0}$*5	TO_1	
1	*3	*3	*3	0	0	0	0	$R4_0$	$R3_2$	Reload timer
1	*3	*3	*3	0	0	*2	*2	$R4_0$	TO_1	
1	*3	*3	*3	1	0	0	0	$\overline{INT_0}$	$R3_2$	
1	*3	*3	*3	1	0	*2	*2	$\overline{INT_0}$	TO_1	
1	1	1	1	*1	0	0	0	$\overline{TI_1}$, $\overline{INT_0}$*5	$R3_2$	Event counter (with reload function)
1	1	1	1	*1	0	*2	*2	$\overline{TI_1}$, $\overline{INT_0}$*5	TO_1	
*1	*3	*3	*3	0	1	*1	*1	$R4_0$	TO_1	PWM*4
*1	*3	*3	*3	1	1	*1	*1	$\overline{INT_0}$	TO_1	
*1	1	1	1	*1	1	*1	*1	$\overline{TI_1}$, $\overline{INT_0}$*5	TO_1	PWM (external clock*4)

*1 Both 0 and 1 do not affect the operation.
*2 Either or both of the bit 0 and bit 1 of TOR is 1.
*3 One, two, or three bits of bit 0, 1, 2 of TMR are 0.
*4 When the timer load register is set to $ 00 while PWM output is selected, the timer functions in the same way as the free-running timer.
*5 When $\overline{TI_1}$ is selected, $\overline{INT_0}$ interrupt mask bit must have been set since $\overline{INT_0}$ is simultaneosly selected.

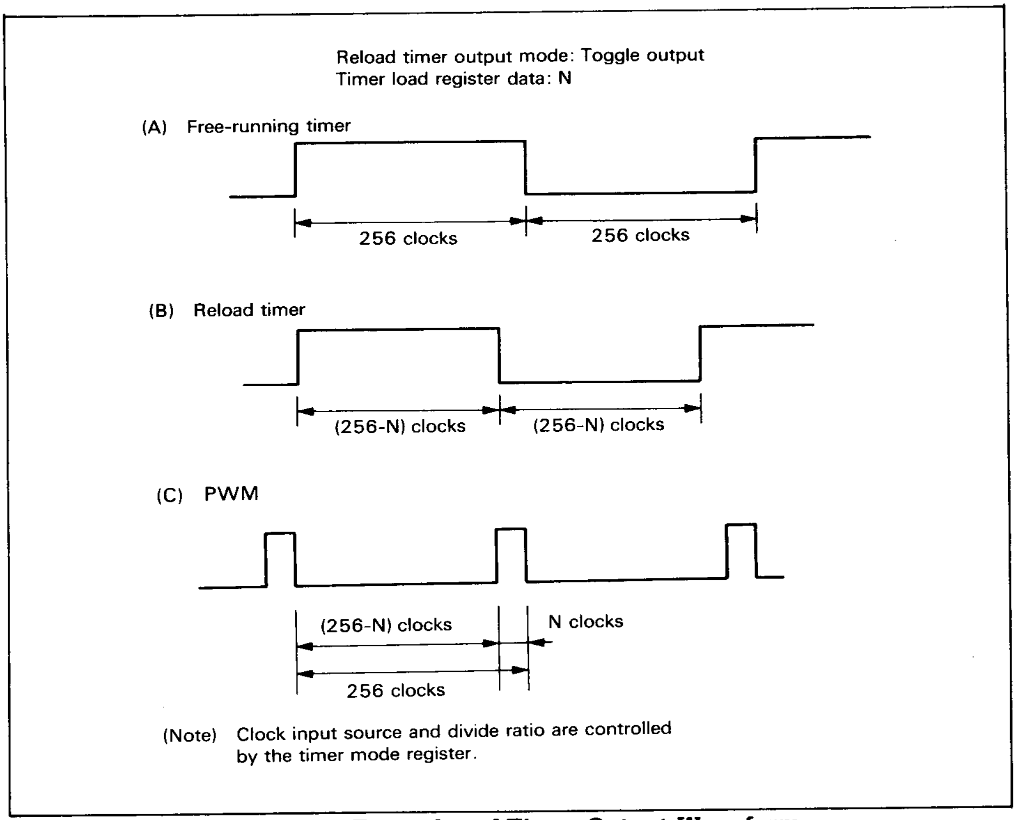
Figure 37. Combination of Timer 1 Operation Modes



TMR2				TOR2				Timer input pin	Timer 2 function
Bit 3	Bit 2	Bit 1	Bit 0	Bit 3	Bit 2	Bit 1	Bit 0		
0	*2	*2	*2	0	*1	*1	*1	$R4_1$	Free-running timer
0	*2	*2	*2	1	*1	*1	*1	INT_1	
0	1	1	1	*1	*1	*1	*1	TI_2, INT_1 *3	Event counter
1	*2	*2	*2	0	*1	*1	*1	$R4_1$	Reload timer
1	*2	*2	*2	1	*1	*1	*1	INT_1	
1	1	1	1	*1	*1	*1	*1	TI_2, INT_1 *3	Event counter (with reload function)

*1 Both 0 and 1 do not affect the operation.
*2 One, two, or three bits of bit 0, 1, 2 of TMR are 0.
*3 When TI_2 is selected, INT_1 interrupt mask bit must have been set since INT_1 is simultaneously selected.

Figure 38. Combination of Timer 2 Operation Modes



Reload timer output mode: Toggle output
Timer load register data: N

(Note) Clock input source and divide ratio are controlled by the timer mode register.

Figure 39. Examples of Timer Output Waveform

A/D Converter

The MCU incorporates an A/D converter operating in the sequential comparison method with a resistor ladder. It can measure two analog inputs with 8-bit resolution.
A block diagram of the A/D converter is shown in figure 40.

The A/D converter is composed of the following registers and bit.

A/D mode register (AMR: $00C): The A/D mode register is a write-only register. Bits 0 and 1 select an A/D conversion period, and bits 2 and 3 select a channel (figure 41).

A/D data register (ADRU: $00E, ADRL: $00D): The A/D data register is a read-only register.
Data read during A/D conversion is not guaranteed. After completion of a conversion, resulting data is maintained until the following conversion starts (figure 42).

A/D start bit, status flag (ADSF: $020, 0): Writing 1 into the A/D start bit initiates A/D conversion. On completion of A/D conversion, the converted data is stored in the A/D data register simultaneously with setting the A/D interrupt flag, and the A/D start flag is cleared.

Note that writing into ADSF should be performed by bit manipulation instruction SEM or SEMD. It is invalid with the REM/REMD instructions. ADSF must not be written to during A/D conversion. The configuration of ADSF is shown in figure 43.

Precautions on A/D converter use:
Port output instructions should not be executed during A/D conversion to allow the A/D converter to operate stably.

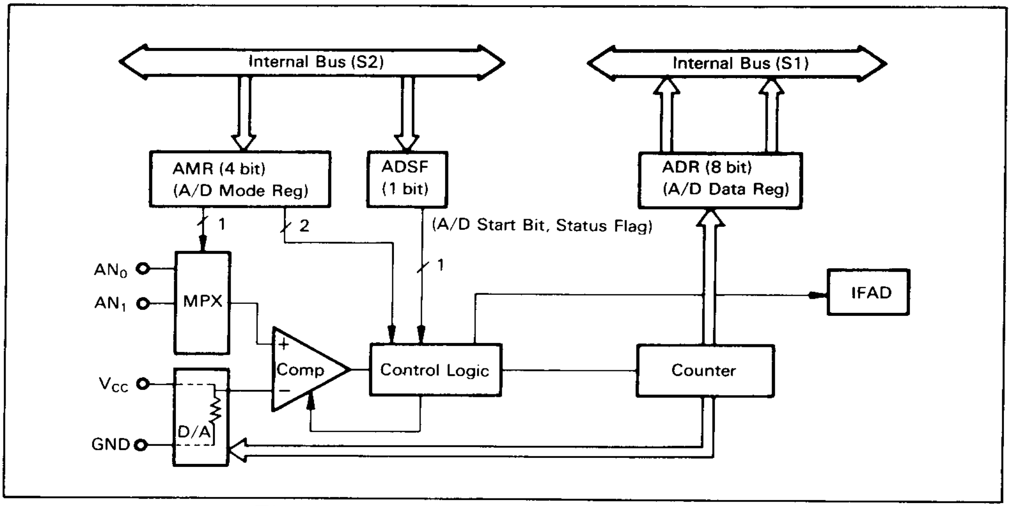
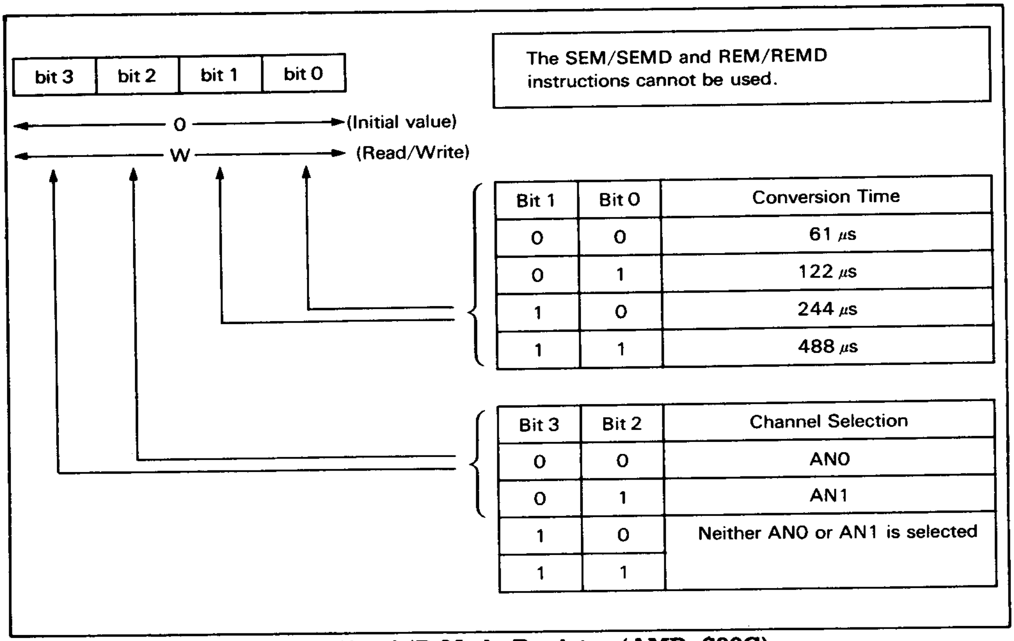


Figure 40. A/D Converter Block Diagram



The SEM/SEMD and REM/REMD instructions cannot be used.

Bit 1	Bit 0	Conversion Time
0	0	61 μs
0	1	122 μs
1	0	244 μs
1	1	488 μs

Bit 3	Bit 2	Channel Selection
0	0	AN0
0	1	AN1
1	0	Neither AN0 or AN1 is selected
1	1	

Figure 41. A/D Mode Register (AMR: $00C)

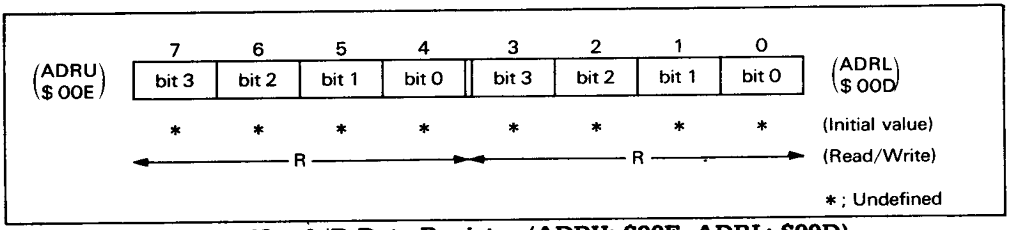


Figure 42. A/D Data Register (ADRU: $00E, ADRL: $00D)

bit 3	bit 2	bit 1	bit 0
—	—	—	ADSF
			0 (Initial value)
			R/W (Read/write)

bit 0	A/D start Bit,Status Flag
0	Indicate A/D conversion end
1	Start A/D conversion

— : Reserved bit

Instructions other than RAM bit manipulation instructions cannot be used. ADSF is set with the SEM/SEMD instructions and tested by the TM/TMD instructions. REM and REMD are invalid.

Figure 43 A/D Start Bit, Status Flag (ADSF: $020,0)

Liquid Crystal Display Circuit

A LCD circuit can directly drive a liquid crystal in three drive systems, static, 1/2 duty, and 1/3 duty with its three common signal pins and 28 segment signal pins. The LCD circuit is comprised of RAM, controller, and driver as shown in figure 44.

RAM ($3A0-$3BB): RAM in the LCD circuit is dual-port RAM which sends display data to the segment signal pin without software. Relation between display RAM area and segment signal is shown in figure 45. By writing data in RAM area, the corresponding segment pin automatically outputs it as display data. 1 means light on, and 0 means light off.

Control Section: The control section consists of the following three registers:
- LCD output register (LOR: $024) (figure 46)
- LCD control register (LCR: $025) (figure 47)
- LCD mode register (LMR: $026) (figure 48)

LCD Driver: The LCD driver has three common signal pins and 28 segment signal pins. 16 segment pins SEG13 to SEG28 are multiplexed with output ports R8, R7, R6, and R5.

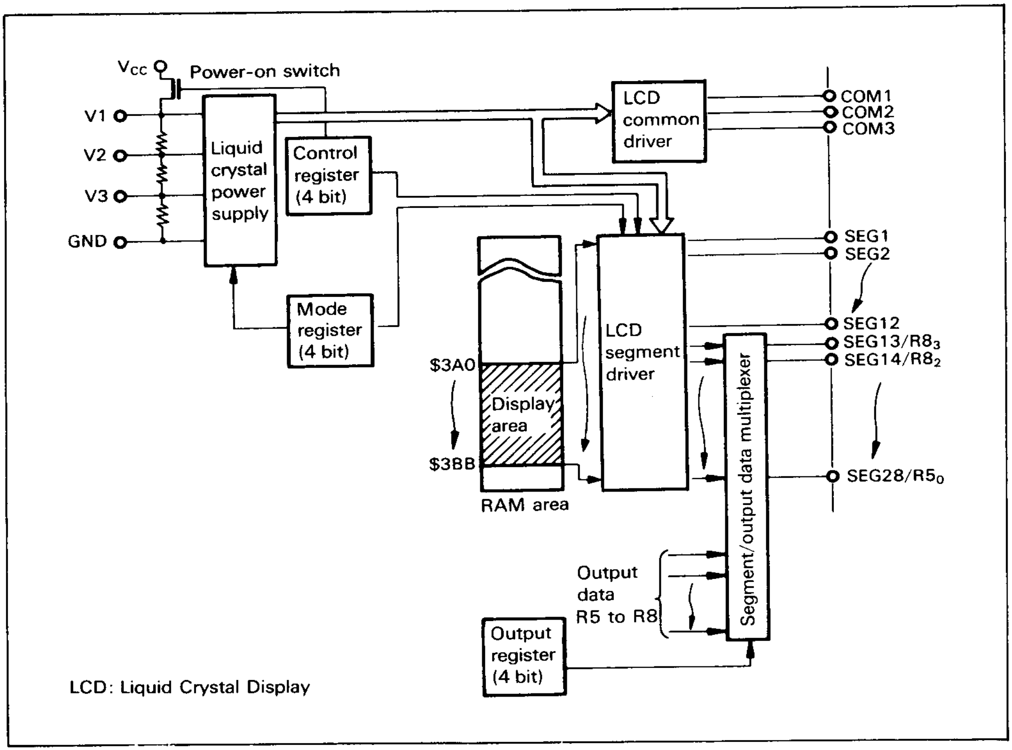
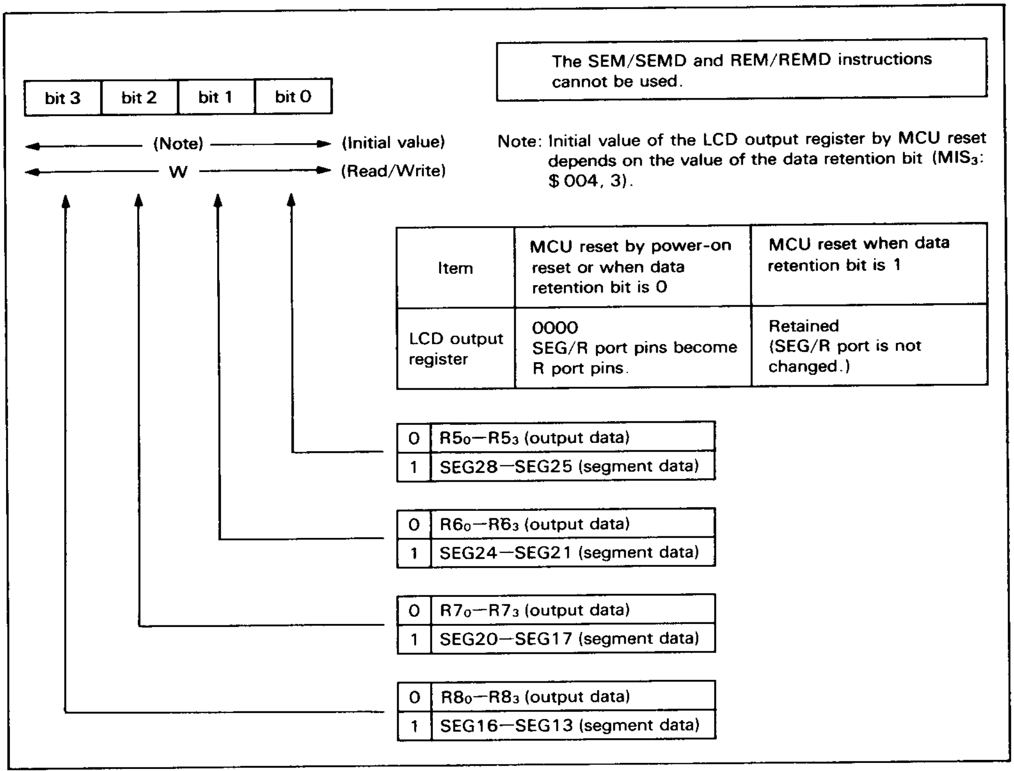


Figure 44. LCD Circuit Block Diagram

BIT 3	BIT 2	BIT 1	BIT 0	Address (Hexadecimal)
——	SEG 1	SEG 1	SEG 1	$3A0
——	SEG 2	SEG 2	SEG 2	$3A1
——	SEG 3	SEG 3	SEG 3	$3A2
——	SEG 4	SEG 4	SEG 4	$3A3
——	SEG 5	SEG 5	SEG 5	$3A4
——	SEG 6	SEG 6	SEG 6	$3A5
——	SEG 7	SEG 7	SEG 7	$3A6
——	SEG 8	SEG 8	SEG 8	$3A7
——	SEG 9	SEG 9	SEG 9	$3A8
——	SEG 10	SEG 10	SEG 10	$3A9
——	SEG 11	SEG 11	SEG 11	$3AA
——	SEG 12	SEG 12	SEG 12	$3AB
——	SEG 13	SEG 13	SEG 13	$3AC
——	SEG 14	SEG 14	SEG 14	$3AD
——	SEG 15	SEG 15	SEG 15	$3AE
——	SEG 16	SEG 16	SEG 16	$3AF
——	SEG 17	SEG 17	SEG 17	$3B0
——	SEG 18	SEG 18	SEG 18	$3B1
——	SEG 19	SEG 19	SEG 19	$3B2
——	SEG 20	SEG 20	SEG 20	$3B3
——	SEG 21	SEG 21	SEG 21	$3B4
——	SEG 22	SEG 22	SEG 22	$3B5
——	SEG 23	SEG 23	SEG 23	$3B6
——	SEG 24	SEG 24	SEG 24	$3B7
——	SEG 25	SEG 25	SEG 25	$3B8
——	SEG 26	SEG 26	SEG 26	$3B9
——	SEG 27	SEG 27	SEG 27	$3BA
——	SEG 28	SEG 28	SEG 28	$3BB
	COM3	COM2	COM1	

Note: Some areas do not correspond to a segment signal.
They can be read/written to.

Figure 45. LCD RAM Area and Segment Signals



The SEM/SEMD and REM/REMD instructions cannot be used.

Note: Initial value of the LCD output register by MCU reset depends on the value of the data retention bit (MIS$_3$: $ 004, 3).

Item	MCU reset by power-on reset or when data retention bit is 0	MCU reset when data retention bit is 1
LCD output register	0000 SEG/R port pins become R port pins.	Retained (SEG/R port is not changed.)

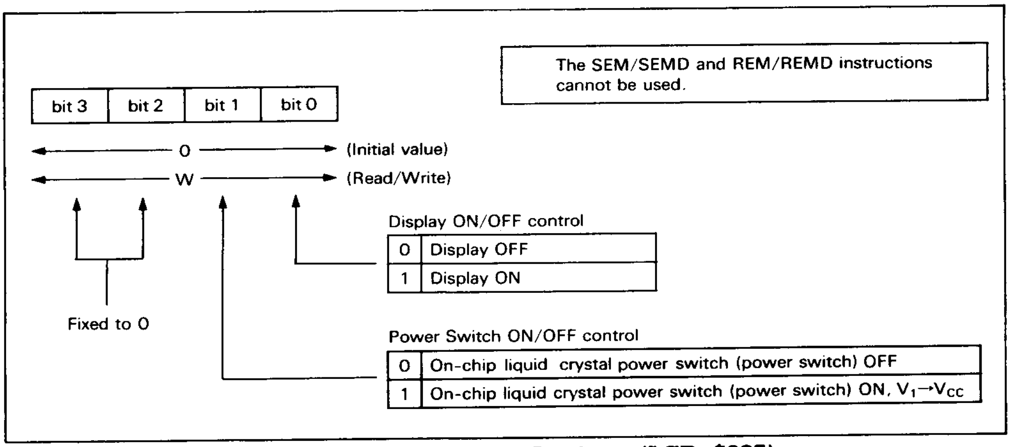
0	R5$_0$—R5$_3$ (output data)
1	SEG28—SEG25 (segment data)

0	R6$_0$—R6$_3$ (output data)
1	SEG24—SEG21 (segment data)

0	R7$_0$—R7$_3$ (output data)
1	SEG20—SEG17 (segment data)

0	R8$_0$—R8$_3$ (output data)
1	SEG16—SEG13 (segment data)

Figue 46. LCD Output Register (LOR: $024)



The SEM/SEMD and REM/REMD instructions cannot be used.

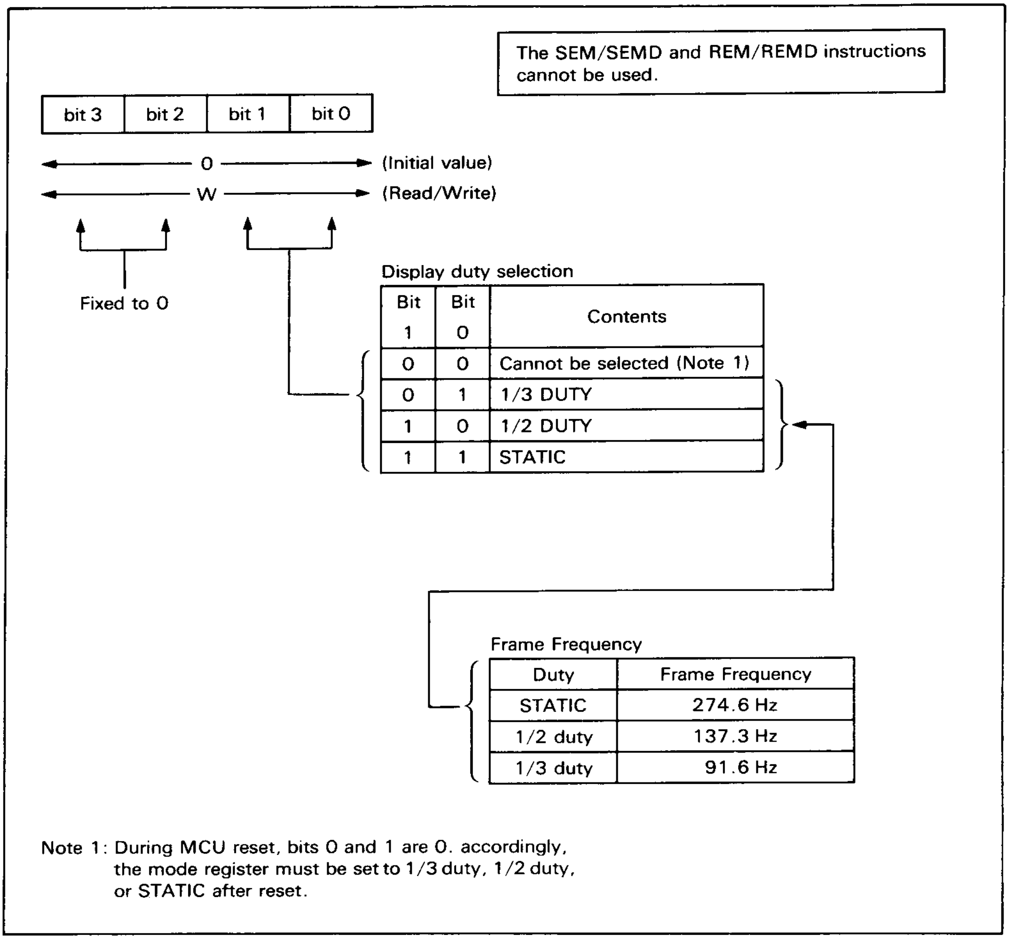
Display ON/OFF control

0	Display OFF
1	Display ON

Power Switch ON/OFF control

0	On-chip liquid crystal power switch (power switch) OFF
1	On-chip liquid crystal power switch (power switch) ON, $V_1 \rightarrow V_{CC}$

Figure 47. LCD Control Register (LCR: $025)



The SEM/SEMD and REM/REMD instructions cannot be used.

Display duty selection

Bit 1	Bit 0	Contents
0	0	Cannot be selected (Note 1)
0	1	1/3 DUTY
1	0	1/2 DUTY
1	1	STATIC

Frame Frequency

Duty	Frame Frequency
STATIC	274.6 Hz
1/2 duty	137.3 Hz
1/3 duty	91.6 Hz

Note 1: During MCU reset, bits 0 and 1 are 0. accordingly, the mode register must be set to 1/3 duty, 1/2 duty, or STATIC after reset.

Figure 48. LCD Mode Register (LMR: $026)

Large LCD Panel Driving and Driving Voltage (V_{LCD}): When using the large LCD panel, lower the dividing resistance by implementing the external resistors parallel to the internal dividing resistors as shown in figure 49.

Since the liquid crystal display board is of matrix configuration, the path of the charge/discharge current through the load capacitors is very complicated. Moreover, as it varies depending on display condition, a value of resistance cannot be simply determined by referring to the load capacitance of liquid crystal display. A value of resistance must be experimentally determined according to the demand for power consumption of the equipment in which the liquid crystal display is implemented. (A capacitor,C=0.1 to 0.3 μF can be used if necessary.) In general, R is 1 kΩ to 10 kΩ.

The following figure shows a connection when changing the liquid crystal driving voltage (V_{LCD}). In this case, power supply switch for dividing resistor (power switch) should be turned OFF. (Bit 1 of the LCR register is 0).

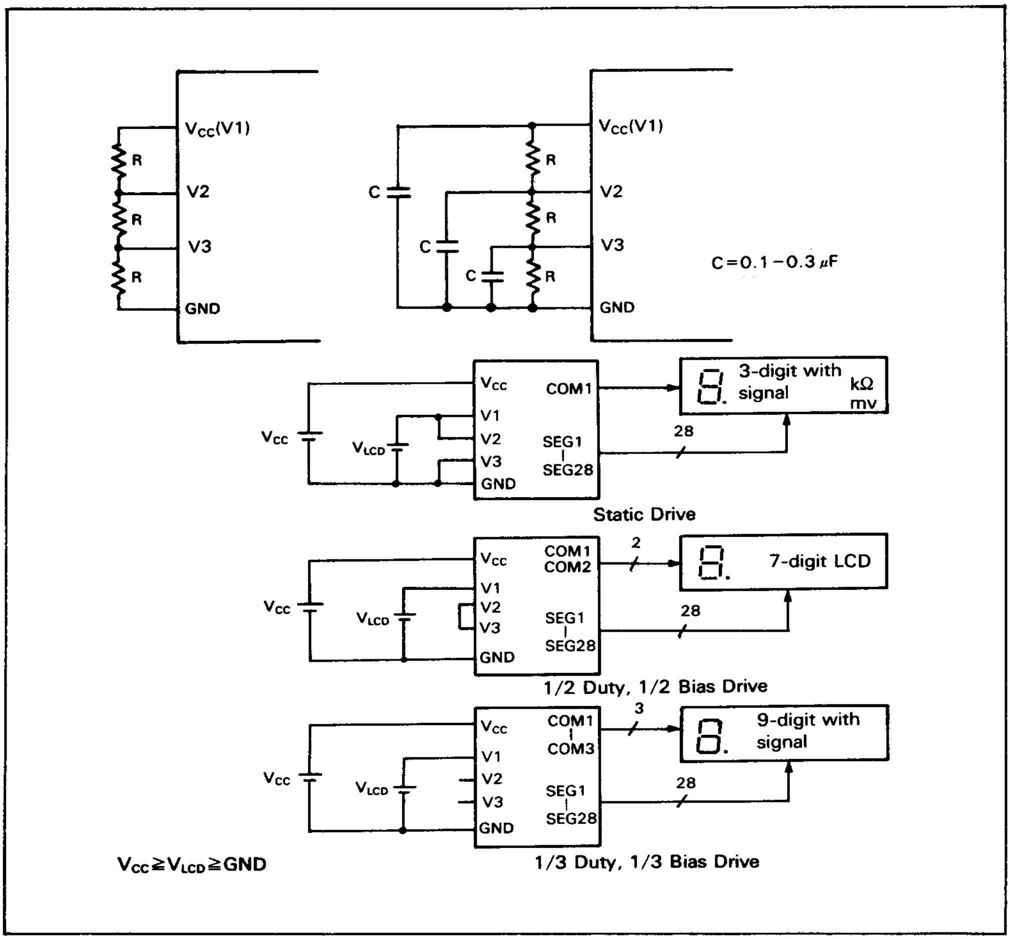


Figure 49. An Example of LCD Connection

Input/Output Port

The MCU has 47 I/O pins comprising 29 I/O pins, 2 input pins, and 16 output pins (multiplexed with SEG pins). The I/O ports (D, R0-R3) have individual data direction registers so that data direction (I/O) can be selected for each bit.Port R4 is an input port. All ports are standard ports.The registers which control the I/O ports of each module are shown in figure 52.

Port D (D_0-D_{13}): The D port is an I/O port composed of 14 I/O pins accessed on a bit basis. All bits of the port D data register are set to 1 by MCU reset. Data direction (input/output) of each pin can be selected by the data direction register provided for each bit (0: input, 1: output). The data direction register is cleared by reset, so that the D port becomes an input port by reset. (See figure 50.) The D port can be set and reset by the SED/RED and SEDD/REDD instructions, and can be tested by the TD/TDD instructions.
Pins D_0-D_2 and D_8-D_{13} are multiplexed with INT_2, INT_3, INT_4, $\overline{SCK_1}$, SI_1, SO_1, $\overline{SCK_2}$, SI_2, and SO_2, respectively.

Ports R0-R3: These are the I/O ports accessed in 4-bit units. The initial value of the port R data register by MCU reset depends on the value of the data retention bit. When it is 0, register value is 1, when it is 1, the register value before reset is retained. Data direction (input/output) of each bit of port R can be controlled by the port R data direction register (DDR0-DDR3) (0: input, 1: output). Since the data direction register is cleared by reset, the R port becomes an input port by reset. However, when the data retention bit is 1, I/O direction of the R port remains unchanged since the previous data is maintained without being cleared. (See figure 51.)
Data is input as well as output through ports R0-R3 by the LAR and LBR instruction.
Pin $R3_2$ of port R3 is multiplexed with timer output pin TO_1. TO_1 is an output pin of timer 1, which can output optional frequency clock and PWM output signal utilizing reload function. For details, see section "Timer".
Circuit types of I/O pins are shown in table 13.

Port R4: Port R4 is an input port accessed in 2-bit units. Pin $R4_0$ is multiplexed with external interrupt input $\overline{INT_0}$ and timer input $\overline{TI_1}$, and $R4_1$ is multiplexed with external interrupt input INT_1 and timer input TI_2. Operation mode of each pin is selected by the timer mode register and timer output register. (for details, see section "Timer".)
Data is input through port R4 by the LAR and LBR instructions.
Circuit types of the R4 port pins are shown in table 13.

Ports R5-R8: These ports are output ports accessed in 4-bit units. Pins $R5_0$-$R5_3$ are multiplexed with SEG28-SEG25; $R6_0$-$R6_3$, with SEG24-SEG21; $R7_0$-$R7_3$ with SEG20-SEG17; and $R8_0$-$R8_3$, with SEG16-SEG13, respectively. Operation mode of each pin is selected by the LCD output register (for details, see section "LCD Circuit").
During reset, these ports are used as R ports and 1 is output. However, when the data retention bit is 1, these pins are not switched to R ports and 1 is not set in the data register, so that the previous data is maintained.
Circuit types of the R5-R8 pins are shown in table 13.

Data Direction Register ($030-$033, $03B-$03E): The data direction register (DDR) is a 4-bit write-only register which controls input and output of the I/O ports. Each I/O port bit has a DDR and data direction can be selected on a bit basis. Data is input from each pin when a port is an input port. For an output port, data is accessed from the data register.
The I/O ports become input ports when DDR is cleared by reset. However, the R port data direction register is not cleared by reset when the data retention bit is 1, so that the previous data is retained.

D port data direction register (DDRD0: $03B, DDRD1: $03C, DDRD2: $03D, DDRD3: $03E): This register can select D port data direction on a bit basis. RAM bit manipulation instruction cannot be used. The D port becomes an input port by reset (figure 50).

R port data direction register (DDR0: $030, DDR1: $031, DDR2: $032, DDR3: $033): This register can select data direction for ports R0-R3 on a bit basis. RAM bit manipulation instruction cannot be used. When the R port data direction register is cleared by MCU reset, the R ports become input ports. However, if the data retention bit is 1, this register is not cleared by reset and the previous data is retained (figure 51).
Note: I/O pins unused on the user system.
If unused I/O pins are left floating, the LSI may malfunction because of noise. To prevent this, unused pins should be pulled up to V_{CC} through a resistor of approximately 100 kΩ.

	3	2	1	0	
DDRD0 ($ 03B)	D_3 DDR	D_2 DDR	D_1 DDR	D_0 DDR	
	0				(Initial value)
	W				(Read/Write)

	3	2	1	0	
DDRD1 ($ 03C)	D_7 DDR	D_6 DDR	D_5 DDR	D_4 DDR	
	0				(Initial value)
	W				(Read/Write)

	3	2	1	0	
DDRD2 ($ 03D)	D_{11} DDR	D_{10} DDR	D_9 DDR	D_8 DDR	
	0				(Initial value)
	W				(Read/Write)

	3	2	1	0	
DDRD3 ($ 03E)	—	—	D_{13} DDR	D_{12} DDR	
			0	0	(Initial value)
			W	W	(Read/Write)

Note: '—' : Reserved bit

The SEM/SEMD and REM/REMD instructions cannot be used.

Port I/O selection

Data direction register	Function
0	Input
1	Output

Figure 50. D Port Data Direction Register

	3	2	1	0	
DDR0 ($ 030)	$R0_3$ DDR	$R0_2$ DDR	$R0_1$ DDR	$R0_0$ DDR	
	*	*	*	*	(Initial value)
	W				(Read/Write)

	3	2	1	0	
DDR1 ($ 031)	$R1_3$ DDR	$R1_2$ DDR	$R1_1$ DDR	$R1_0$ DDR	
	*	*	*	*	(Initial value)
	W				(Read/Write)

	3	2	1	0	
DDR2 ($ 032)	$R2_3$ DDR	$R2_2$ DDR	$R2_1$ DDR	$R2_0$ DDR	
	*	*	*	*	(Initial value)
	W				(Read/Write)

	3	2	1	0	
DDR3 ($ 033)	—	$R3_2$ DDR	$R3_1$ DDR	$R3_0$ DDR	
		*	*	*	(Initial value)
		W			(Read/Write)

Port I/O selection

Data direction register	Function
0	Input
1	Output

Note 1: *; The initial value of the R port data direction register by MCU reset depends on the value of the data retention bit (MIS_3: $004,3).

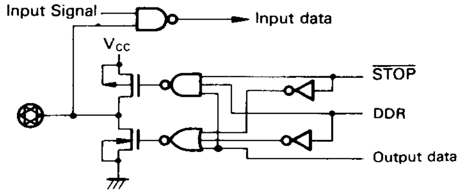
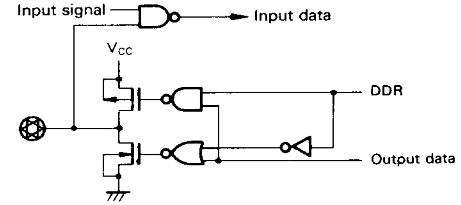
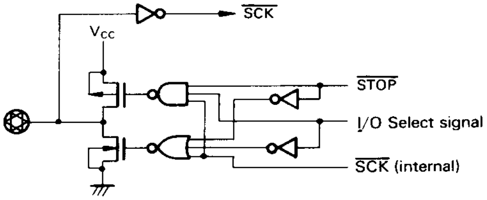
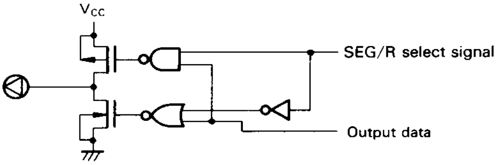
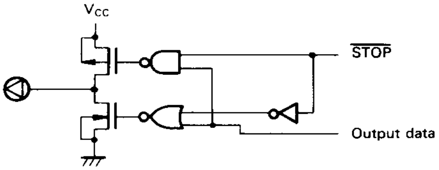
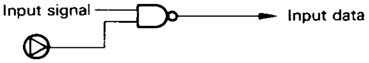
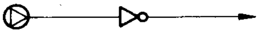
Item	Reset when power on reset and when data retention bit is 0	Reset when data retention bit is 1
R port data direction register	0000 R port becomes input port	Retained (R port function does not change)

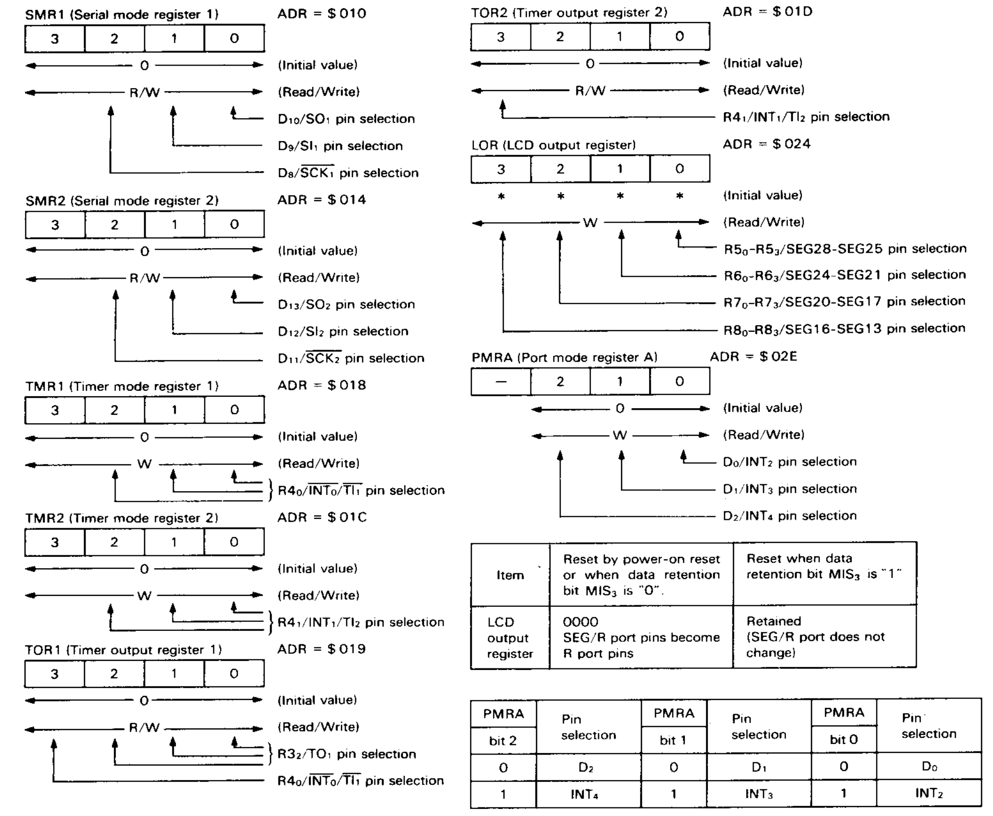
Note 2: —; Reserved bit

The SEM/SEMD and REM/REMD instructions cannot be used.

Figure 51. R Port Data Direction Register

Table 13. Circuit Type of I/O Pins

I/O Pins	Circuit Type	Applied Pins
I/O common pin	 Input Signal Input data V_{CC} $\overline{STOP}$ DDR Output data	D_0-D_{13}
	 Input signal Input data V_{CC} DDR Output data	R0-R3
	 $\overline{SCK}$ V_{CC} $\overline{STOP}$ I/O Select signal $\overline{SCK}$ (internal)	$\overline{SCK_1}$, $\overline{SCK_2}$
Output pin		R5-R8
		SO_1, SO_2, TO_1 ($\overline{STOP}$="1")
Input pin		R4
		$\overline{INT_0}$, INT_1, INT_2 INT_3, INT_4, SI_1 SI_2, $\overline{TI_1}$, TI_2



Item	Reset by power-on reset or when data retention bit MIS_3 is "0".	Reset when data retention bit MIS_3 is "1"
LCD output register	0000 SEG/R port pins become R port pins	Retained (SEG/R port does not change)

PMRA	Pin selection	PMRA	Pin selection	PMRA	Pin selection
bit 2		bit 1		bit 0	
0	D_2	0	D_1	0	D_0
1	INT_4	1	INT_3	1	INT_2

Notes: 1. Initial value of the LCD output register depends on the value of the data retention bit.
2. "—" : Reserved bit
3. See section "Serial Interface" for SMR1 and SMR2, section "Timer" for TMR1, TMR2, TOR1, and TOR2, and section "LCD circuit" for LOR.

Figure 52. I/O Mode Selection Registers

Oscillation Circuit

The oscillation circuit configuration is shown in figure 53.

Figure 54 shows an internal oscillation circuit block diagram. 4.5 MHz crystal should be attached.

Figure 55 is an example of a crystal oscillation circuit, and its layout example is shown in figure 56.

Notes: 1. The circuit parameters are changed by crystal resonator and the floating capacitance in the board. In employing a resonator, please consult with the engineers of the crystal maker to determine the actual circuit parameter.

2. Wiring between OSC1, OSC2, and elements should be as short as possible, and never cross other wires. (See figure 56.)

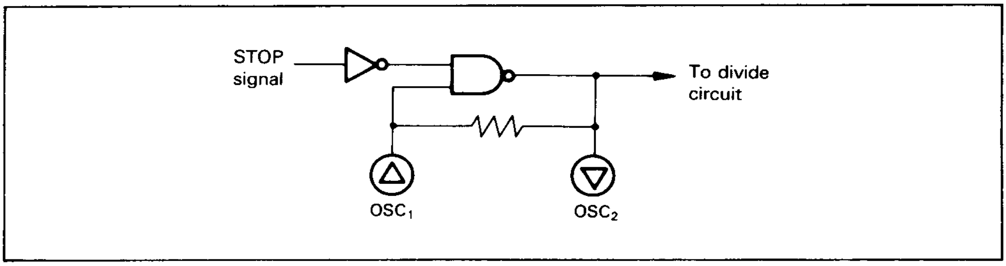


Figure 53. Oscillator

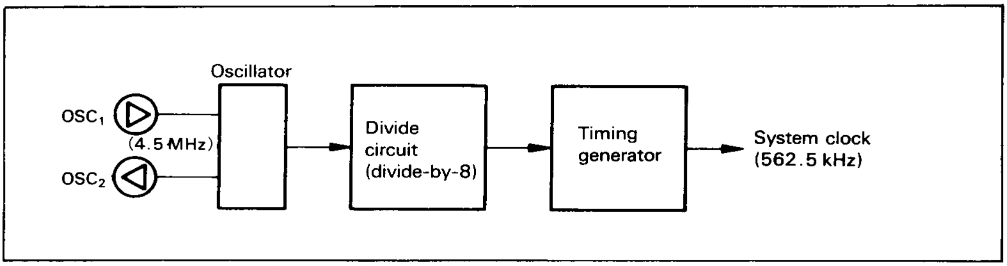


Figure 54. Internal Oscillation Circuit

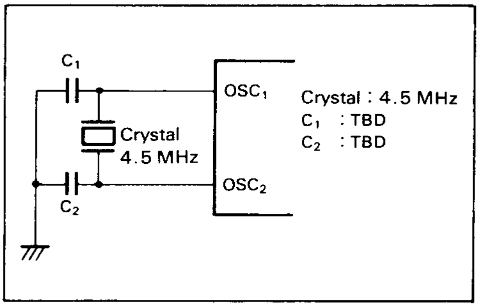


Figure 55. Crystal Oscillation Circuit

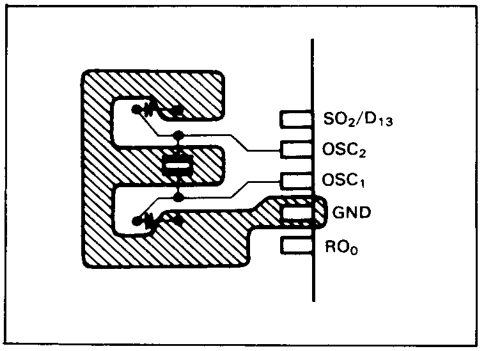


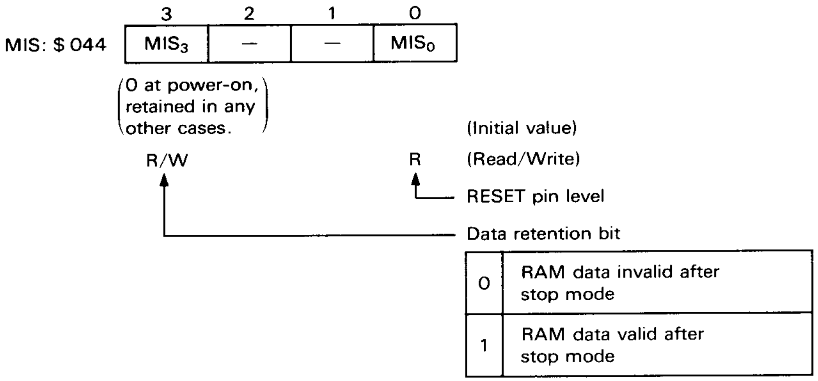
Figure 56. Crystal Oscillation Circuit Layout

Miscellaneous Register

The function of the miscellaneous register, shown in figure 57, is described below.

RESET: The RESET pin level can be read by reading the MIS_0 bit.

Data retention bit: This bit is cleared if V_{CC} voltage supply stops. This bit is set before the MCU enters into stop mode. Then, if this bit is set when the MCU recovers from stop mode by MCU reset, V_{CC} voltage is applied during stop mode and RAM data is retained.



0	RAM data invalid after stop mode
1	RAM data valid after stop mode

Notes: 1. MIS_0 cannot be written to. When the miscellaneous register is written to, MIS_0 does not change.
2. "—": Reserved bit.
3. The TM/TMD instructions can be used. The SEM/SEMD and REM/REMD instructions can not be used.

Figure 57. Miscellaneous Register

PROM Mode Pin Description

The HD4074509 is a ZTAT microcomputer incorporating PROM. When the MCU enters PROM mode, the MCU stops to program the PROM. PROM mode signals are listed in table 14, and PROM mode pin assignment is in figure 58.

Table 14. PROM Mode Signals

	MCU mode		PROM mode			MCU mode		PROM mode	
Pin No.	**Pin name**	**I/O**	**Pin name**	**I/O**	**Pin No.**	**Pin name**	**I/O**	**Pin name**	**I/O**
1	AN$_0$	I	$\overline{M_0}$	I	41	R2$_2$	I/O	A$_{11}$	I
2	V1				42	R2$_3$	I/O	A$_{12}$	I
3	V2				43	R3$_0$	I/O	A$_{13}$	I
4	V3				44	R3$_1$	I/O	A$_{14}$	I
5	IF	I			45	R3$_2$/TO$_1$	I/O		
6	$\overline{\text{TEST}}$	I	$\overline{\text{TEST}}$	I	46	R4$_0$/$\overline{\text{INT}_0}$/$\overline{\text{TI}_1}$	I	V$_{PP}$	
7	ϕ_1	O			47	R4$_1$/INT$_1$/TI$_2$	I	A$_9$	I
8	ϕ_2	O			48	V$_{CC}$		V$_{CC}$	
9	V$_{CC}$ (PLL)		V$_{CC}$		49	SEG1	O	O$_0$	I/O
10	PLL(D)	I			50	SEG2	O	O$_1$	I/O
11	PLL(P)	I			51	SEG3	O	O$_2$	I/O
12	RESET	I	RESET	I	52	SEG4	O	O$_3$	I/O
13	GND (PLL)		GND		53	SEG5	O	O$_4$	I/O
14	INT$_2$/D$_0$	I/O	O$_0$	I/O	54	SEG6	O	V$_{CC}$	
15	INT$_3$/D$_1$	I/O	O$_1$	I/O	55	SEG7	O		
16	INT$_4$/D$_2$	I/O	O$_2$	I/O	56	SEG8	O		
17	D$_3$	I/O	O$_3$	I/O	57	SEG9	O		
18	D$_4$	I/O	O$_4$	I/O	58	SEG10	O		
19	D$_5$	I/O	O$_5$	I/O	59	SEG11	O		
20	D$_6$	I/O	O$_6$	I/O	60	SEG12	O		
21	D$_7$	I/O	O$_7$	I/O	61	SEG13/R8$_3$	O		
22	$\overline{\text{SCK}_1}$/D$_8$	I/O	$\overline{\text{OE}}$	I	62	SEG14/R8$_2$	O		
23	SI$_1$/D$_9$	I/O	$\overline{\text{CE}}$	I	63	SEG15/R8$_1$	O		
24	SO$_1$/D$_{10}$	I/O	V$_{CC}$		64	SEG16/R8$_0$	O		
25	$\overline{\text{SCK}_2}$/D$_{11}$	I/O	V$_{CC}$		65	SEG17/R7$_3$	O		
26	SI$_2$/D$_{12}$	I/O			66	SEG18/R7$_2$	O		
27	SO$_2$/D$_{13}$	I/O			67	SEG19/R7$_1$	O		
28	OSC$_2$	O			68	SEG20/R7$_0$	O		
29	OSC$_1$	I			69	SEG21/R6$_3$	O		
30	GND		GND		70	SEG22/R6$_2$	O		
31	R0$_0$	I/O	A$_1$	I	71	SEG23/R6$_1$	O		
32	R0$_1$	I/O	A$_2$	I	72	SEG24/R6$_0$	O		
33	R0$_2$	I/O	A$_3$	I	73	SEG25/R5$_3$	O		
34	R0$_3$	I/O	A$_4$	I	74	SEG26/R5$_2$	O		
35	R1$_0$	I/O	A$_5$	I	75	SEG27/R5$_1$	O		
36	R1$_1$	I/O	A$_6$	I	76	SEG28/R5$_0$	O		
37	R1$_2$	I/O	A$_7$	I	77	COM1	O		
38	R1$_3$	I/O	A$_8$	I	78	COM2	O		
39	R2$_0$	I/O	A$_0$	I	79	COM3	O		
40	R2$_1$	I/O	A$_{10}$	I	80	AN$_1$	I	$\overline{M_1}$	I

Notes: 1. I/O: Input/output pin, I: Input pin, O: Output pin
2. Pins O$_0$ to O$_4$ each have two pins. Each pair must be closed when using.

Pins for PROM mode

V_{PP}: V_{PP} applies program voltage (12.5 V ± 0.3 V).

$\overline{CE}$: $\overline{CE}$ inputs the control signal which enables PROM programming and verifying.

$\overline{OE}$: $\overline{OE}$ inputs the data output control signal for verify.

A_0-A_{14}: These are the address input pins of on-chip PROM.

O_0-O_7: These are the data I/O pins of on-chip PROM.

$\overline{M_0}$, $\overline{M_1}$: These are used for setting PROM mode. PROM mode is set by bringing $\overline{M_0}$, $\overline{M_1}$, and $\overline{TEST}$ pins to low and RESET pin to high.

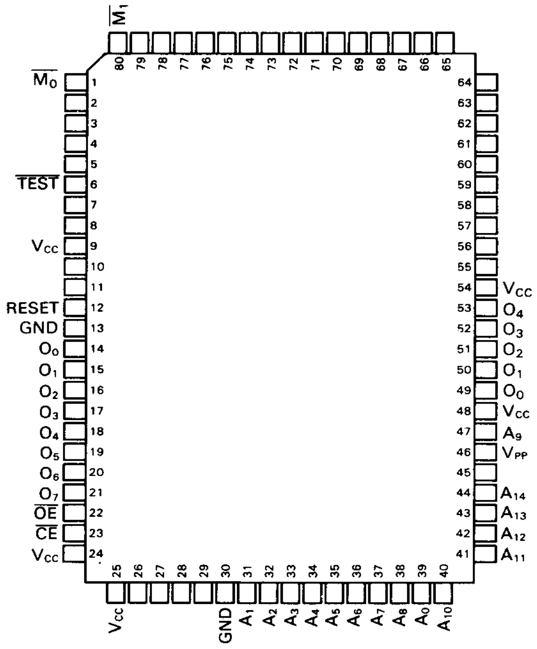


Figure 58. Pin Arrangement in PROM Mode

Programmable ROM Programming

The MCU enters into PROM mode by setting $\overline{\text{TEST}}$, $\overline{M_0}$, $\overline{M_1}$, to low, and RESET pin to high (figure 59).

The specification of the on-chip PROM is the same as that of EPROM 27256. It can be programmed by the standard PROM programmer using a 80-to-28 pin socket adapter. In order to program the PROM using a general-purpose PROM programmer, the HD4074509 incorporates the conversion circuit which divides a 10-bit HMCS400 series instruction into 5 higher bits and 5 lower bits as shown in figure 60. One MCU address is assigned to two PROM addresses. For example, in programming an 16 k word PROM with a general-purpose PROM programmer, the user should assign 32 kbyte address locations ($0000-$7FFF).

Precautions

1. Addresses $0000 to $7FFF should be specified if the PROM is programmed by the PROM programmer. If addresses of $8000 or higher is accessed, the PROM may not be programmed or verified. Note that the plastic package type cannot be erased and reprogrammed. Unused address data must be set to $FF.
2. Be careful that an index of the PROM programmer socket, socket adapter, and LSI match. Using the wrong programmer of socket adapter may cause an overvoltage and damage the LSI. Make sure that the LSI is firmly fixed in the socket adapter, and that the socket adapter is firmly fixed in the programmer.
3. The PROM should be programmed with V_{PP} = 12.5 V. Other PROMs also use 21 V. If 21 V is applied to the HD4074509, the LSI may be permanently damaged.
   12.5 V is Intel's 27256 V_{PP}.

Programming and verification

The HD4074509 can accomplish high-speed programming without causing voltage stress or degrading data reliability. The flowchart is shown in figure 61. For details, see "Characteristics and Applications of PROM".

Table 15. Mode Selection

	Pin			
Mode	$\overline{\text{CE}}$	$\overline{\text{OE}}$	V_{PP}	O_0-O_7
Programming	Low	High	V_{PP}	Data input
Verify	High	Low	V_{PP}	Data output
Programming inhibited	High	High	V_{PP}	High impedance

Table 16. PROM Programmers and Socket Adapters

PROM Programmer		Socket Adapters	
Maker	**Type name**	**Maker**	**Type name**
DATA I/O	22 B 29 B	Hitachi	HS450ESF01H
AVAL Corp	PKW-1000 PKW-7000	Hitachi	HS450ESF01H

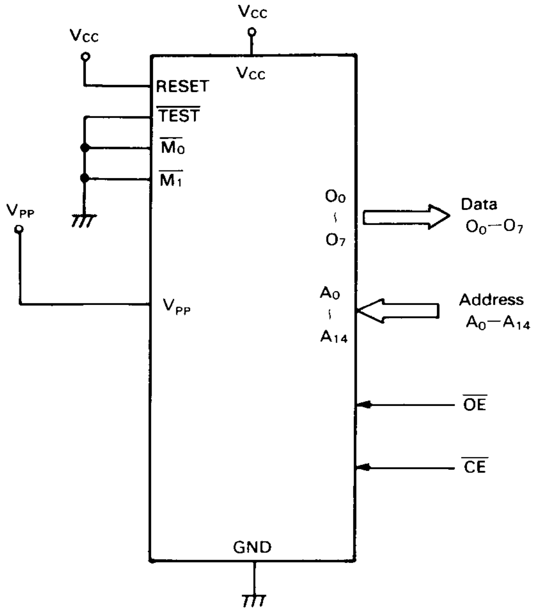
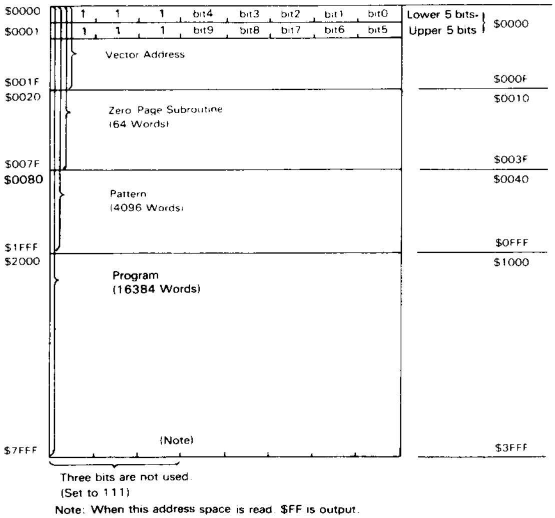


Figure 59. PROM Mode



Note: When this address space is read. $FF is output.

Figure 60. PROM Mode Memory Map

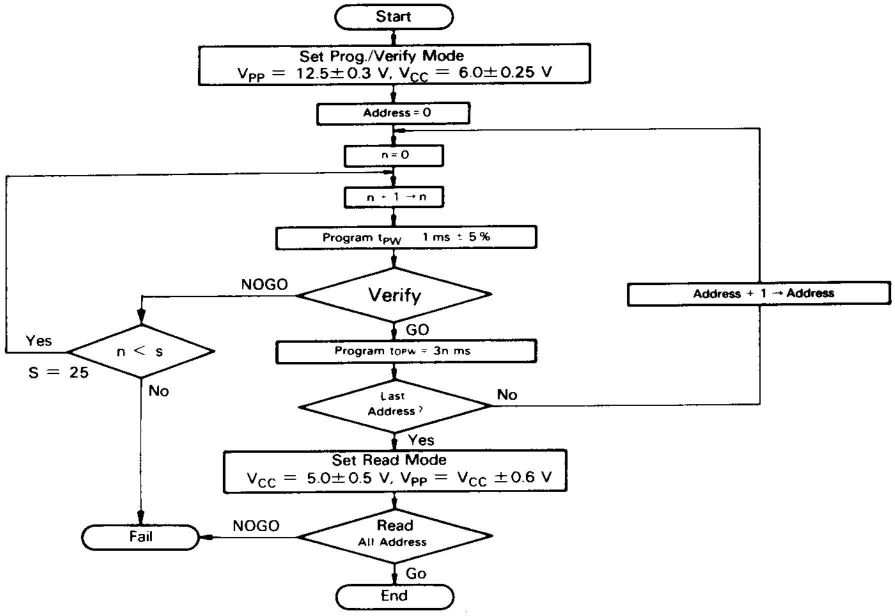


Figure 61. A Sequence of High Speed Programming

Characteristics and Applications of PROM

Principles of Programming/Erasing

The configuration of the ZTAT micros' memory cells are the same as that of an EPROM's (figure 62). Therefore they are programmed by applying high voltage to control gates and drains, which injects hot electrons into the floating gate. The stored electrons become stable since they are surrounded by an energy varrier of SiO_2 film. Such a cell becomes a 0 bit due to the memory threshold voltage change. A cell with so condensed electrons at its floating gate appears as a 1 bit.

The electron charge in memory cells may decrease as time goes by. This can be caused by:

① Ultraviolet light, discharged by photo-emitting electrons (erasure principle)
② Heat, discharged by thermal emitting electrons
③ High voltage, discharged by a high electric field at the control gate or drain

If the oxide film covering a floating gate is defective, the erasure rate is great. Normally, electron erasure does not occur, because such defective devices are found and removed during testing.

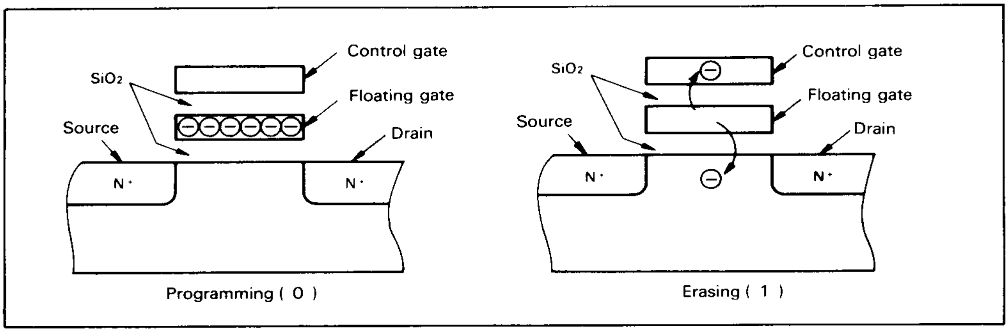


Figure 62. Cross-section of PROM Memory Cell

Programming precautions

The PROM memory cells should be programmed under specific voltage and timing conditions. The higher the program voltage and the longer the program pulse is applied, the more electrons will be injected into the floating gate. However, if an overvoltage is applied to V_{PP}, the p-n junction may be permanently damaged. Pay particular attention to PROM programmer overshot. Negative voltage noise will cause a parasitic transistor effect, which may reduce break-down voltage.

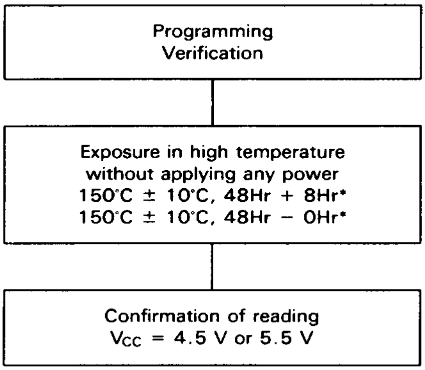
The ZTAT micros are connected electrically to the PROM programmer through a socket adapter. Therefore, pay attention to the followings:

① Confirm that the socket adapter is firmly fixed on the PROM programmer.

② Do not touch the socket adapter or the LSI during programming.
Misprogramming can be caused by poor contacts.

On-chip PROM reliability after programming

Generally, semiconductors are reliable except for initial failures. To avoid failures, screening can be performed. Exposure to high temperature is a kind of screening which removes PROM memory cells with data hold failures in a short time. This is done to the ZTATs in the wafer stage, so ZTAT data hold characteristics are high. Exposing the LSI to 150 °C after user programming can effectively upgrade these characteristics. Figure 63 shows the recommended screening flow.


Programming
Verification
Exposure in high temperature
without applying any power
150°C ± 10°C, 48Hr + 8Hr*
150°C ± 10°C, 48Hr − 0Hr*
Confirmation of reading
Vcc = 4.5 V or 5.5 V

* Exposing time is the time after the temperature in fireplace reaches 150°C.

Note: If programming erros occur continuously during programming with one PROM programmer, stop programming and check the PROM programmer or socket adapter.
If trouble occurs in verification after programming, or after exposure to high temperatures, please inform a Hitachi engineer.

Figure 63. Recommended Screening Flow

RAM Addressing Mode

As shown in figure 64, the MCU has three RAM addressing modes: register indirect addressing, direct addressing, and memory register addressing.

Register Indirect Addressing: Contents of registers W, X, and Y (10 bits) are used as the RAM address.

Direct Addressing: A direct addressing instruction consists of two words, with the word (10 bits) following the opcode used as the RAM address.

Memory Register Addressing: The memory register (16 addresses; from \$040 to \$04F) is accessed by executing the LAMR and XMRA instructions.

ROM Addressing Mode and P Instructions

The MCU has four ROM addressing modes, as shown in figure 65.

Direct Addressing Mode: The program can branch to any address in the ROM memory space by executing a JMPL, BRL, or CALL instruction. These instructions replace the 14 program counter bits (PC_{13} to PC_0) with the 14-bit immediate data.

Current Page Addressing Mode: A page is 256 words. By executing a BR instruction, the program can branch to an address in the current page. This instruction replaces the low-order eight bits of the program counter (PC_7 to PC_0) with the 8-bit immediate data.

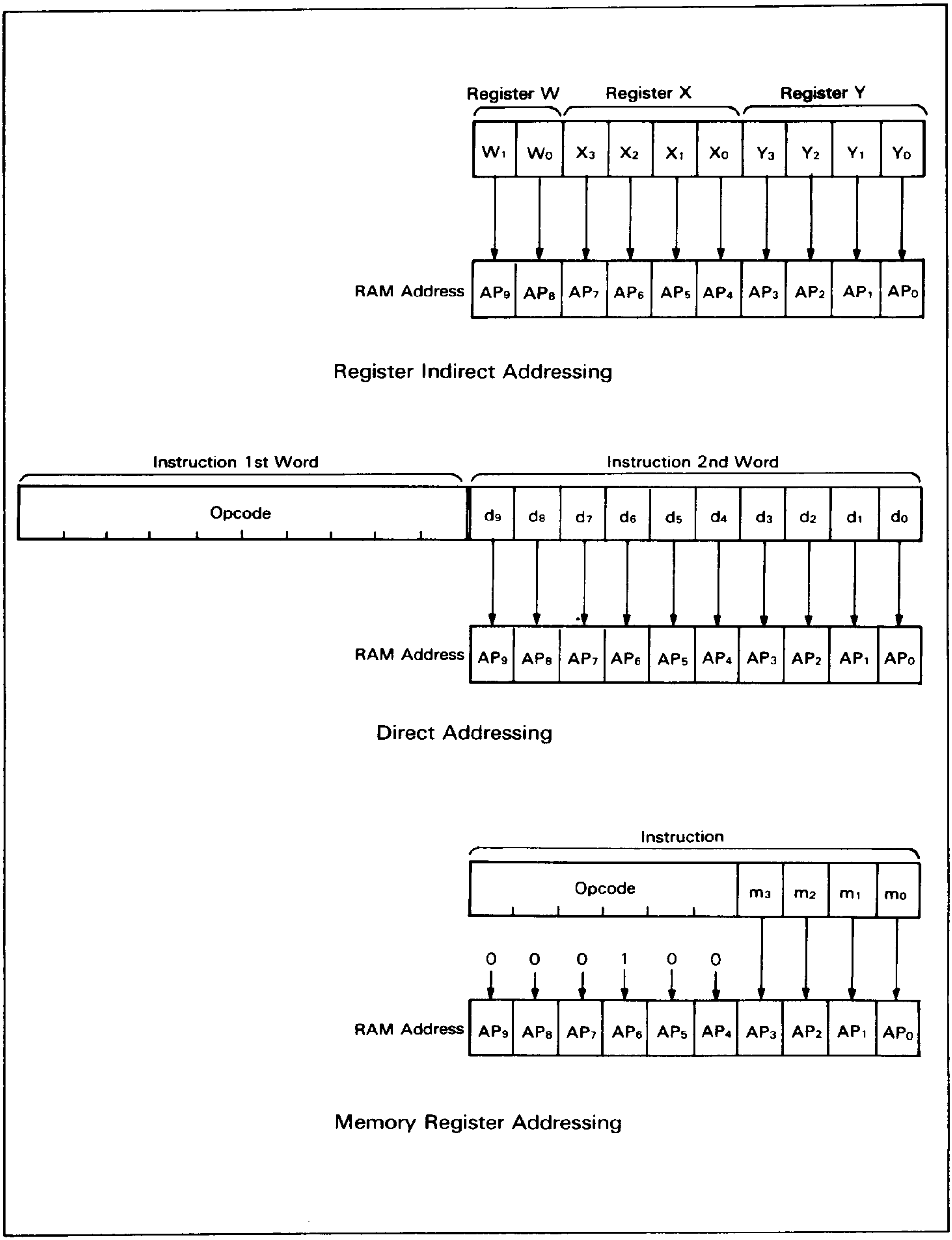
When BR instruction is on a page boundary (256n + 255) (figure 67), executing a BR instruction transfers the PC contents to the next page according to the hardware architecture. Consequently, the program branches to the next page when BR is used on a page boundary. The HMCS400-series cross macro assembler has an automatic paging facility for ROM pages.

Zero-Page Addressing Mode: By executing a CAL instruction, the program can branch to the zero-page subroutine area, which is located at \$0000-\$003F. When a CAL instruction is executed, 6 bits of immediate data are placed in the low-order 6 bits of the program counter (PC_5 to PC_0) and 0s are placed in the high-order 8 bits (PC_{13} to PC_6).

Table Data Addressing: By executing a TBR instruction, the program can branch to the address determined by the contents of the 4-bit immediate data, accumulator, and register B.

P Instruction: ROM data accessed by table data addressing can be referred to by a P instruction (figure 66). When bit 8 in the referred ROM data is 1, 8 bits of ROM data are written into the accumulator and register B. When bit 9 is 1, 8 bits of ROM data are written into the R1 and R2 port data registers. When both bits 8 and 9 are 1, ROM data are written into the accumulator and register B and also to the R1 and R2 port data registers at the same time.

The P instruction has no effect on the program counter.


Register W
Register X
Register Y
W_1 W_0 X_3 X_2 X_1 X_0 Y_3 Y_2 Y_1 Y_0
RAM Address
AP_9 AP_8 AP_7 AP_6 AP_5 AP_4 AP_3 AP_2 AP_1 AP_0

Register Indirect Addressing

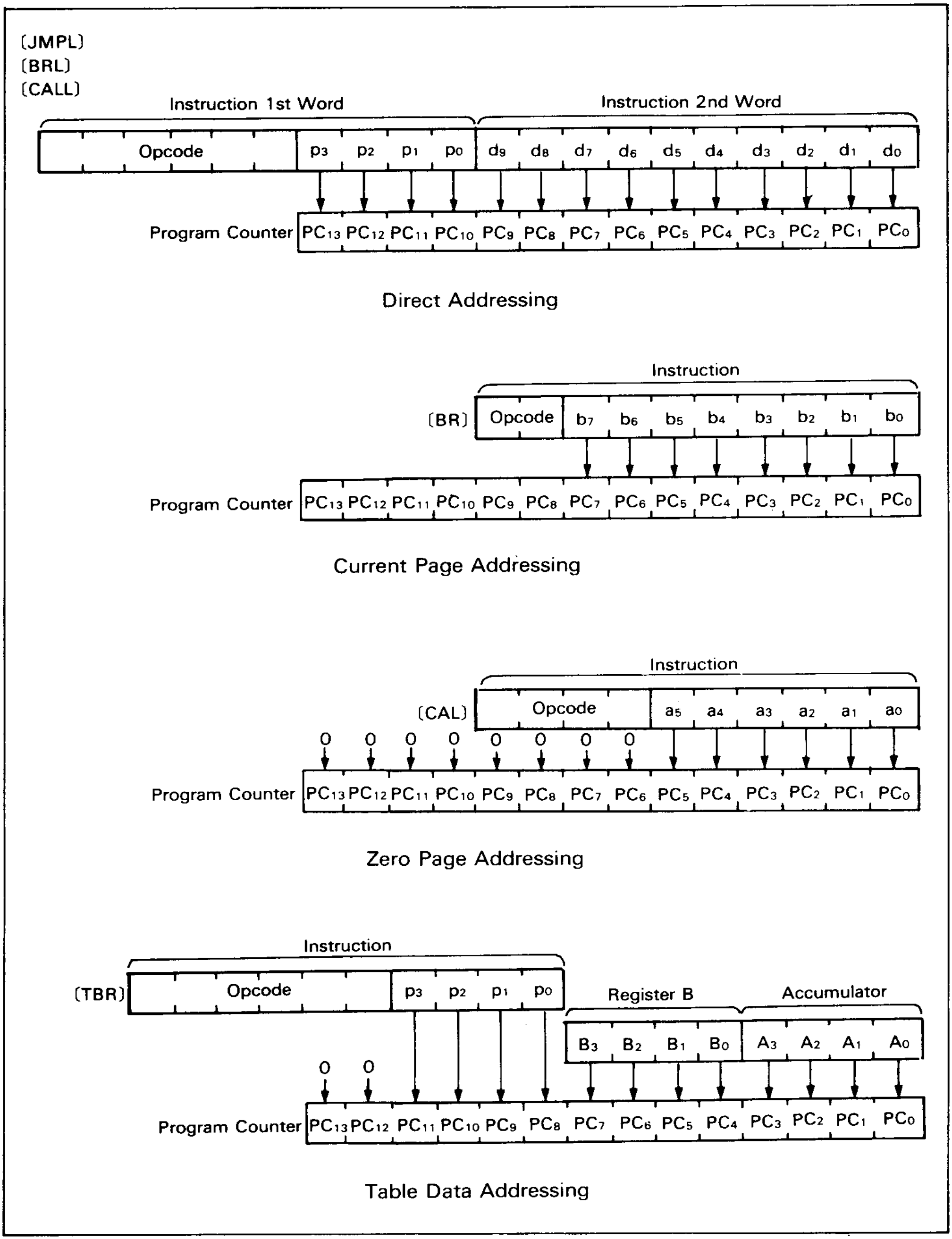
Instruction 1st Word
Instruction 2nd Word
Opcode
d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0
RAM Address
AP_9 AP_8 AP_7 AP_6 AP_5 AP_4 AP_3 AP_2 AP_1 AP_0

Direct Addressing

Instruction
Opcode
m_3 m_2 m_1 m_0
0 0 0 1 0 0
RAM Address
AP_9 AP_8 AP_7 AP_6 AP_5 AP_4 AP_3 AP_2 AP_1 AP_0

Memory Register Addressing

Figure 64. RAM Addressing Modes



Direct Addressing

Current Page Addressing

Zero Page Addressing

Table Data Addressing

Figure 65. ROM Addressing Mode

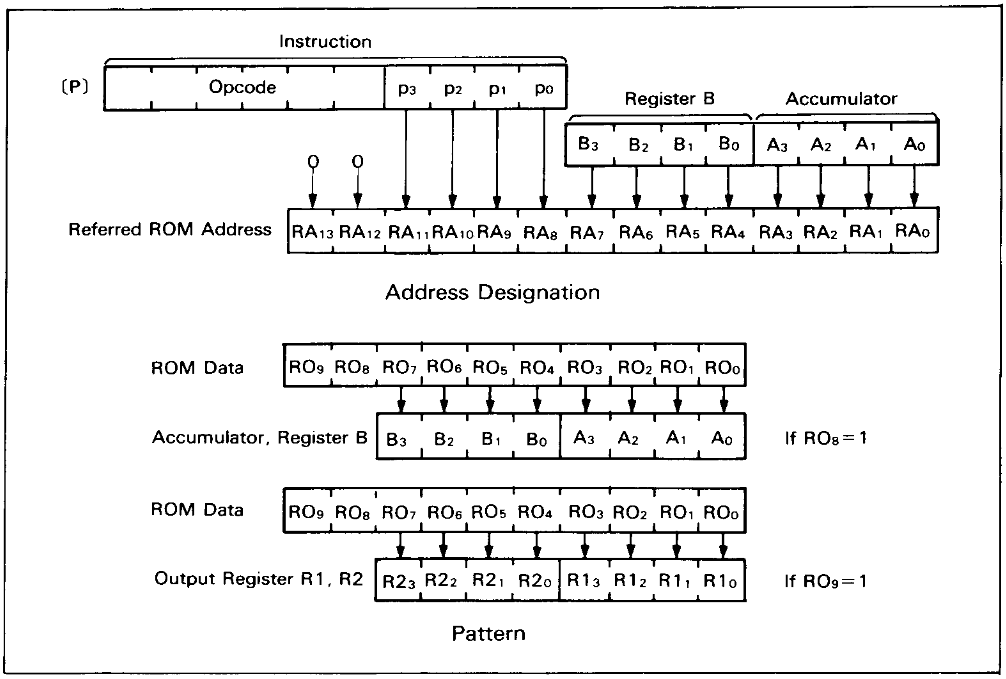

Instruction
(P)
Opcode
p_3 p_2 p_1 p_0
Register B
Accumulator
B_3 B_2 B_1 B_0 A_3 A_2 A_1 A_0
0 0
Referred ROM Address
RA_{13} RA_{12} RA_{11} RA_{10} RA_9 RA_8 RA_7 RA_6 RA_5 RA_4 RA_3 RA_2 RA_1 RA_0
Address Designation
ROM Data
RO_9 RO_8 RO_7 RO_6 RO_5 RO_4 RO_3 RO_2 RO_1 RO_0
Accumulator, Register B
B_3 B_2 B_1 B_0 A_3 A_2 A_1 A_0
If $RO_8 = 1$
ROM Data
RO_9 RO_8 RO_7 RO_6 RO_5 RO_4 RO_3 RO_2 RO_1 RO_0
Output Register R1, R2
$R2_3$ $R2_2$ $R2_1$ $R2_0$ $R1_3$ $R1_2$ $R1_1$ $R1_0$
If $RO_9 = 1$
Pattern

Figure 66. P Instruction

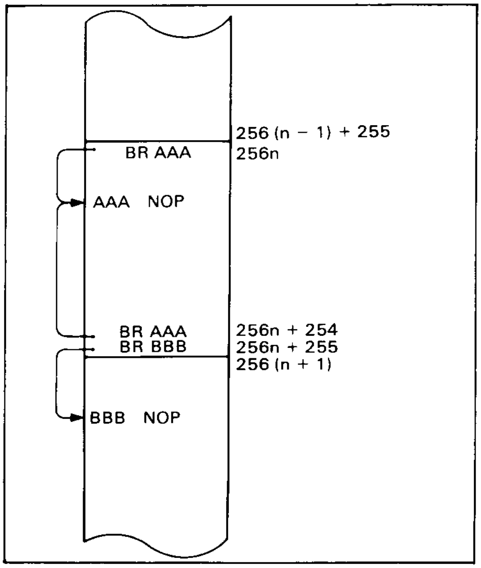


Figure 67. Branch Destination of BR Instruction on the Boundary between Pages

Instruction Set

The MCU provides 101 instructions which are classified into 10 groups as follows:

1. Immediate instructions
2. Register-to-register instructions
3. RAM address instructions
4. RAM-register instructions
5. Arithmetic instructions
6. Compare instructions
7. RAM bit manipulation instructions
8. ROM address instructions
9. Input/output instructions
10. Control instructions

Tables 17-26 list their functions, and table 27 is an opcode map.

Table 17. Immediate Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Load A from Immediate	LAI i	1 0 0 0 1 1 i_3 i_2 i_1 i_0	i · A		1/1
Load B from Immediate	LBI i	1 0 0 0 0 0 i_3 i_2 i_1 i_0	i · B		1/1
Load Memory from Immediate	LMID i,d	0 1 1 0 1 0 i_3 i_2 i_1 i_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	i · M		2/2
Load Memory from Immediate, Increment Y	LMIIY i	1 0 1 0 0 1 i_3 i_2 i_1 i_0	i ·M, Y+1 ·Y	NZ	1/1

Table 18. Register-to-Register Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Load A from B	LAB	0 0 0 1 0 0 1 0 0 0	B · A		1/1
Load B from A	LBA	0 0 1 1 0 0 1 0 0 0	A · B		1/1
Load A from W	LAW	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	W · A		2/2 (Note)
Load A from Y	LAY	0 0 1 0 1 0 1 1 1 1	Y · A		1/1
Load A from SPX	LASPX	0 0 0 1 1 0 1 0 0 0	SPX · A		1/1
Load A from SPY	LASPY	0 0 0 1 0 1 1 0 0 0	SPY · A		1/1
Load A from MR	LAMR m	1 0 0 1 1 1 m_3 m_2 m_1 m_0	MR(m) · A		1/1
Exchange MR and A	XMRA m	1 0 1 1 1 1 m_3 m_2 m_1 m_0	MR(m) ·– A		1/1

Note: An operand is automatically provided for the second word of LAW and LWA instruction by assembler.

Table 19. RAM Address Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Load W from Immediate	LWI i	0 0 1 1 1 1 0 0 i_1 i_0	i → W		1/1
Load X from Immediate	LXI i	1 0 0 0 1 0 i_3 i_2 i_1 i_0	i → X		1/1
Load Y from Immediate	LYI i	1 0 0 0 0 1 i_3 i_2 i_1 i_0	i → Y		1/1
Load W from A	LWA	0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0	A → W		2/2 (Note)
Load X from A	LXA	0 0 1 1 1 0 1 0 0 0	A → X		1/1
Load Y from A	LYA	0 0 1 1 0 1 1 0 0 0	A → Y		1/1
Increment Y	IY	0 0 0 1 0 1 1 1 0 0	Y+1 → Y	NZ	1/1
Decrement Y	DY	0 0 1 1 0 1 1 1 1 1	Y−1 → Y	NB	1/1
Add A to Y	AYY	0 0 0 1 0 1 0 1 0 0	Y+A → Y	OVF	1/1
Subtract A from Y	SYY	0 0 1 1 0 1 0 1 0 0	Y−A → Y	NB	1/1
Exchange X and SPX	XSPX	0 0 0 0 0 0 0 0 0 1	X ↔ SPX		1/1
Exchange Y and SPY	XSPY	0 0 0 0 0 0 0 0 1 0	Y ↔ SPY		1/1
Exchange X and SPX,Y and SPY	XSPXY	0 0 0 0 0 0 0 0 1 1	X↔SPX, Y↔SPY		1/1

Note: An operand is automatically provided for the second word of LAW and LWA instruction by the assembler.

Table 20. RAM-Register Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Load A from Memory	LAM(XY)	0 0 1 0 0 1 0 0 y x	M → A, (X↔SPX, Y↔SPY)		1/1
Load A from Memory	LAMD d	0 1 1 0 0 1 0 0 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	M → A		2/2
Load B from Memory	LBM(XY)	0 0 0 1 0 0 0 0 y x	M → B, (X↔SPX, Y↔SPY)		1/1
Load Memory from A	LMA(XY)	0 0 1 0 0 1 0 1 y x	A → M, (X↔SPX, Y↔SPY)		1/1
Load Memory from A	LMAD d	0 1 1 0 0 1 0 1 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	A → M		2/2
Load Memory from A, Increment Y	LMAIY(X)	0 0 0 1 0 1 0 0 0 x	A → M, Y+1 → Y (X↔SPX)	NZ	1/1
Load Memory from A, Decrement Y	LMADY(X)	0 0 1 1 0 1 0 0 0 x	A → M, Y−1 → Y (X↔SPX)	NB	1/1
Exchange Memory and A	XMA(XY)	0 0 1 0 0 0 0 0 y x	M ↔ A, (X↔SPX, Y↔SPY)		1/1
Exchange Memory and A	XMAD d	0 1 1 0 0 0 0 0 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	M ↔ A		2/2
Exchange Memory and B	XMB(XY)	0 0 1 1 0 0 0 0 y x	M ↔ B, (X↔SPX, Y↔SPY)		1/1

Note: (XY) and (X) have the following meaning:

(1) The instructions with (XY) have 4 mnemonics and 4 object codes for each (example of LAM (XY) is given below). The op-code X or Y is assembled as follows:

Mnemonic	Y	X	Function
LAM	0	0	
LAMX	0	1	X ↔ SPX
LAMY	1	0	Y ↔ SPY
LAMXY	1	1	X↔SPX, Y↔SPY

(2) The instructions with (X) have 2 mnemonics and 2 object codes for each (example of LMAIY(X) is given below). The op-code X is assembled as follows:

Mnemonic	X	Function
LMAIY	0	
LMAIYX	1	X ↔ SPX

Table 21. Arithmetic Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Add Immediate to A	AI i	1 0 1 0 0 0 i_3 i_2 i_1 i_0	$A + i \rightarrow A$	OVF	1/1
Increment B	IB	0 0 0 1 0 0 1 1 0 0	$B + 1 \rightarrow B$	NZ	1/1
Decrement B	DB	0 0 1 1 0 0 1 1 1 1	$B - 1 \rightarrow B$	NB	1/1
Decimal Adjust for Addition	DAA	0 0 1 0 1 0 0 1 1 0			1/1
Decimal Adjust for Subtraction	DAS	0 0 1 0 1 0 1 0 1 0			1/1
Negate A	NEGA	0 0 0 1 1 0 0 0 0 0	$\overline{A} + 1 \rightarrow A$		1/1
Complement B	COMB	0 1 0 1 0 0 0 0 0 0	$\overline{B} \rightarrow B$		1/1
Rotate Right A with Carry	ROTR	0 0 1 0 1 0 0 0 0 0			1/1
Rotate Left A with Carry	ROTL	0 0 1 0 1 0 0 0 0 1			1/1
Set Carry	SEC	0 0 1 1 1 0 1 1 1 1	$1 \rightarrow CA$		1/1
Reset Carry	REC	0 0 1 1 1 0 1 1 0 0	$0 \rightarrow CA$		1/1
Test Carry	TC	0 0 0 1 1 0 1 1 1 1		CA	1/1
Add A to Memory	AM	0 0 0 0 0 0 1 0 0 0	$M + A \rightarrow A$	OVF	1/1
Add A to Memory	AMD d	0 1 0 0 0 0 1 0 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$M + A \rightarrow A$	OVF	2/2
Add A to Memory with Carry	AMC	0 0 0 0 0 1 1 0 0 0	$M + A + CA \rightarrow A$ $OVF \rightarrow CA$	OVF	1/1
Add A to Memory with Carry	AMCD d	0 1 0 0 0 1 1 0 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$M + A + CA \rightarrow A$ $OVF \rightarrow CA$	OVF	2/2
Subtract A from Memory with Carry	SMC	0 0 1 0 0 1 1 0 0 0	$M - A - \overline{CA} \rightarrow A$ $NB \rightarrow CA$	NB	1/1
Subtract A from Memory with Carry	SMCD d	0 1 1 0 0 1 1 0 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$M - A - \overline{CA} \rightarrow A$ $NB \rightarrow CA$	NB	2/2
OR A and B	OR	0 1 0 1 0 0 0 1 0 0	$A \cup B \rightarrow A$		1/1
AND Memory with A	ANM	0 0 1 0 0 1 1 1 0 0	$A \cap M \rightarrow A$	NZ	1/1
AND Memory with A	ANMD d	0 1 1 0 0 1 1 1 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$A \cap M \rightarrow A$	NZ	2/2
OR Memory with A	ORM	0 0 0 0 0 0 1 1 0 0	$A \cup M \rightarrow A$	NZ	1/1
OR Memory with A	ORMD d	0 1 0 0 0 0 1 1 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$A \cup M \rightarrow A$	NZ	2/2
EOR Memory with A	EORM	0 0 0 0 0 1 1 1 0 0	$A \oplus M \rightarrow A$	NZ	1/1
EOR Memory with A	EORMD d	0 1 0 0 0 1 1 1 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$A \oplus M \rightarrow A$	NZ	2/2

Note: $\cap$: Logical AND
$\cup$: Logical OR
$+$: Exclusive OR

Table 22. Compare Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Immediate Not Equal to Memory	INEM i	0 0 0 0 1 0 i_3 i_2 i_1 i_0	$i \neq M$	NZ	1/1
Immediate Not Equal to Memory	INEMD i,d	0 1 0 0 1 0 i_3 i_2 i_1 i_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$i \neq M$	NZ	2/2
A Not Equal to Memory	ANEM	0 0 0 0 0 0 0 1 0 0	$A \neq M$	NZ	1/1
A Not Equal to Memory	ANEMD d	0 1 0 0 0 0 0 1 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$A \neq M$	NZ	2/2
B Not Equal to Memory	BNEM	0 0 0 1 0 0 0 1 0 0	$B \neq M$	NZ	1/1
Y Not Equal to Immediate	YNEI i	0 0 0 1 1 1 i_3 i_2 i_1 i_0	$Y \neq i$	NZ	1/1
Immediate Less or Equal to Memory	ILEM i	0 0 0 0 1 1 i_3 i_2 i_1 i_0	$i \leq M$	NB	1/1
Immediate Less or Equal to Memory	ILEMD i,d	0 1 0 0 1 1 i_3 i_2 i_1 i_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$i \leq M$	NB	2/2
A Less or Equal to Memory	ALEM	0 0 0 0 0 1 0 1 0 0	$A \leq M$	NB	1/1
A Less or Equal to Memory	ALEMD d	0 1 0 0 0 1 0 1 0 0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	$A \leq M$	NB	2/2
B Less or Equal to Memory	BLEM	0 0 1 1 0 0 0 1 0 0	$B \leq M$	NB	1/1
A Less or Equal to Immediate	ALEI i	1 0 1 0 1 1 i_3 i_2 i_1 i_0	$A \leq i$	NB	1/1

Table 23. RAM Bit Manipulation Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Set Memory Bit	SEM n	0 0 1 0 0 0 0 1 n_1 n_0	1 · M(n)		1/1
Set Memory Bit	SEMD n,d	0 1 1 0 0 0 0 1 n_1 n_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	1 · M(n)		2/2
Reset Memory Bit	REM n	0 0 1 0 0 0 1 0 n_1 n_0	0 ·M(n)		1/1
Reset Memory Bit	REMD n,d	0 1 1 0 0 0 1 0 n_1 n_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	0 · M(n)		2/2
Test Memory Bit	TM n	0 0 1 0 0 0 1 1 n_1 n_0		M(n)	1/1
Test Memory Bit	TMD n,d	0 1 1 0 0 0 1 1 n_1 n_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0		M(n)	2/2

Table 24. ROM Address Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Branch on Status 1	BR b	1 1 b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0		1	1/1
Long Branch on Status 1	BRL u	0 1 0 1 1 1 p_3 p_2 p_1 p_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0		1	2/2
Long Jump Unconditionally	JMPL u	0 1 0 1 0 1 p_3 p_2 p_1 p_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0			2/2
Subroutine Jump on Status 1	CAL a	0 1 1 1 a_5 a_4 a_3 a_2 a_1 a_0		1	1/2
Long Subroutine Jump on Status 1	CALL u	0 1 0 1 1 0 p_3 p_2 p_1 p_0 d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0		1	2/2
Table Branch	TBR p	0 0 1 0 1 1 p_3 p_2 p_1 p_0			1/1
Return from Subroutine	RTN	0 0 0 0 0 1 0 0 0 0			1/3
Return from Interrupt	RTNI	0 0 0 0 0 1 0 0 0 1	1 → I/E CA Restore	ST	1/3

Table 25. Input/Output Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
Set Discrete I/O Latch	SED	0 0 1 1 1 0 0 1 0 0	1 → D(Y)		1/1
Set Discrete I/O Latch Direct	SEDD m	1 0 1 1 1 0 m_3 m_2 m_1 m_0	1 → D(m)		1/1
Reset Discrete I/O Latch	RED	0 0 0 1 1 0 0 1 0 0	0 → D(Y)		1/1
Reset Discrete I/O Latch Direct	REDD m	1 0 0 1 1 0 m_3 m_2 m_1 m_0	0 → D(m)		1/1
Test Discrete I/O Latch	TD	0 0 1 1 1 0 0 0 0 0		D(Y)	1/1
Test Discrete I/O Latch Direct	TDD m	1 0 1 0 1 0 m_3 m_2 m_1 m_0		D(m)	1/1
Load A from R Port Register	LAR m	1 0 0 1 0 1 m_3 m_2 m_1 m_0	R(m) → A		1/1
Load B from R Port Register	LBR m	1 0 0 1 0 0 m_3 m_2 m_1 m_0	R(m) → B		1/1
Load R Port Register from A	LRA m	1 0 1 1 0 1 m_3 m_2 m_1 m_0	A → R(m)		1/1
Load R Port Register from B	LRB m	1 0 1 1 0 0 m_3 m_2 m_1 m_0	B → R(m)		1/1
Pattern Generation	P p	0 1 1 0 1 1 p_3 p_2 p_1 p_0			1/2

Table 26. Control Instructions

Operation	Mnemonic	Operation Code	Function	Status	Words/ Cycles
No Operation	NOP	0 0 0 0 0 0 0 0 0 0			1/1
Start Serial	STS	0 1 0 1 0 0 1 0 0 0			1/1
Standby Mode	SBY	0 1 0 1 0 0 1 1 0 0			1/1
Stop Mode	STOP	0 1 0 1 0 0 1 1 0 1			1/1

Table 27. Opcode Map

	R8	0																1															
R9	H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	NOP	XSPX	XSPY	XSPXY	ANEM				AM				ORM				LAW				ANEMD				AMD				ORMD			
	1	RTN	RTNI			ALEM				AMC				EORM				LWA				ALEMD				AMCD				EORMD			
	2	INEM i(4)																INEMD i(4)															
	3	ILEM i(4)																ILEMD i(4)															
	4	LBM(XY)				BNEM				LAB				IB				COMB				OR				STS				SBY	STOP		
	5	LMAIY(X)				AYY				LASPY				IY				JMPL p(4)															
	6	NEGA				RED				LASPX							TC	CALL p(4)															
	7	YNEI i(4)																BRL p(4)															
	8	XMA(XY)				SEM n(2)				REM n(2)				TM n(2)				XMAD				SEMD n(2)				REMD n(2)				TMD n(2)			
	9	LAM(XY)				LMA(XY)				SMC				ANM				LAMD				LMAD				SMCD				ANMD			
	A	ROTR	ROTL					DAA				DAS					LAY	LMID i(4)															
	B	TBR p(4)																P p(4)															
	C	XMB(XY)				BLEM				LBA							DB	CAL a(6)															
	D	LMADY(X)				SYY				LYA							DY																
	E	TD				SED				LXA				REC			SEC																
	F	LWI i(2)																															
1	0	LBI i(4)																BR b(8)															
	1	LYI i(4)																															
	2	LXI i(4)																															
	3	LAI i(4)																															
	4	LBR m(4)																															
	5	LAR m(4)																															
	6	REDD m(4)																															
	7	LAMR m(4)																															
	8	AI i(4)																															
	9	LMIIY i(4)																															
	A	TDD m(4)																															
	B	ALEI i(4)																															
	C	LRB m(4)																															
	D	LRA m(4)																															
	E	SEDD m(4)																															
	F	XMRA m(4)																															

◣···**1-word/2-cycle Instruction**

◥···**1-word/3-cycle Instruction**

◤◢···**RAM Direct Address Instruction (2-word/2-cycle)**

◤···**2-word/2-cycle Instruction**

Absolute Maximum Ratings

Item	Symbol	Constant	Unit	Notes
Supply Voltage	V_{CC}	−0.3 to + 7.0	V	
Programming Voltage	V_{PP}	−0.3 to + 14.0	V	2, 8
Terminal Voltage	V_T	−0.3 to V_{CC} + 0.3	V	
Total Allowance of Input Current	ΣI_O	50	mA	3
Total Allowance of Output Current	$-\Sigma I_O$	50	mA	4
Maximum Input Current	I_O	4	mA	5, 7
Maximum Output Current	$-I_O$	4	mA	6, 7
Operating Temperature	T_{opr}	−40 to + 85	°C	
Storage Temperature	T_{stg}	−55 to + 125	°C	
Storage Temperature (bias)	Tbias	−45 to +90	°C	8

Notes:
1. Permanent damage may occur if absolute maximum ratings are exceeded. Normal operation should be under the conditions of electrical characteristics. If these conditions are exceeded, it may cause a malfunction or affect the reliability of LSI.
2. $R4_0/\overline{INT_0}/\overline{TI_1}$ (V_{PP})
3. Total allowance of input current is the total sum of input current which flows in from all I/O pins to GND simultaneously.
4. Total allowance of output current is the total sum of the output current which flows out from V_{CC} to all I/O pins simultaneously.
5. Maximum input current is the maximum amount of input current from each I/O pin to GND.
6. Maximum output current is the maximum amount of output current from V_{CC} to each I/O pin.
7. R0-R3, R5-R8, D_0-D_{13}, $\phi 1$, $\phi 2$.
8. Applied for the HD4074509.

Electrical Characteristics

DC Characteristics
(V_{CC} = 5 V ± 10 %, GND = 0 V, T_a = −40 to + 85°C, unless otherwise noted)

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
Input High Voltage	V_{IH}	RESET, $\overline{INT_0}$, INT_1, INT_2, INT_3, INT_4, $\overline{TI_1}$, $\overline{TI_2}$, $\overline{SCK_1}$, $\overline{SCK_2}$, SI_1, SI_2	0.8 V_{CC}		V_{CC} + 0.3	V		
Input low Voltage	V_{IL}		− 0.3		0.2 V_{CC}	V		
Output High Voltage	V_{OH}	$\overline{SCK_1}$,$\overline{SCK_2}$, SO_1, SO_2, TO_1	V_{CC} − 1.0			V	$-I_{OH}$ = 1.0mA	
		$\phi1$, $\phi2$	0.5 V_{CC}			V	$-I_{OH}$ = 2mA	
Output Low Voltage	V_{OL}	$\overline{SCK_1}$,$\overline{SCK_2}$, SO_1, SO_2, TO_1			0.4	V	I_{OL} = 1.6mA	
		$\phi1$, $\phi2$			0.4	V	I_{OL} = 2mA	
Input Leakage Current	$\lvert I_{IL} \rvert$	RESET, $\overline{INT_0}$, INT_1, INT_2, INT_3, INT_4,$\overline{TI_1}$,$\overline{TI_2}$, $\overline{SCK_1}$,$\overline{SCK_2}$			1	μA	V_{in} = 0V to V_{CC}	
Three-state Current	$\lvert I_{TS} \rvert$	$\phi1$, $\phi2$			1	μA	V_{in} = 0V to V_{CC}	
Power Dissipation in CPU Operation	I_{CC}	V_{CC}		TBD	TBD	mA	V_{CC} = 5V, f_{OSC} = 4.5MHz PLL halts	1
Power Dissipation in PLL Operation	I_{CC}	V_{CC}		TBD	TBD	mA	V_{CC} = 5V, CPU, PLL operating (f_P = 160MHz) f_{OSC} = 4.5MHz	1
Power Dissipation in Standby Mode	I_{SBY}	V_{CC}		TBD	TBD	mA	V_{CC} = 5V, f_{OSC} = 4.5MHz	1
Power Dissipation in Stop Mode	I_{stop}	V_{CC}			10	μA	V_{CC} = 5V	1
Stop Mode Retain Voltage	V_{stop}	V_{CC}	2			V		

Notes: 1. Output buffer current is excluded.

Input/Output Characteristics for Standard Pins
(V_{CC} = 5 V ± 10 %, GND = 0 V, T_a = −40 to + 85°C, unless otherwise noted)

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
Input High Voltage	V_{IH}	D_0-D_{13} R0-R4	0.7 V_{CC}		V_{CC} + 0.3	V		
Input Low Voltage	V_{IL}		− 0.3		0.22 V_{CC}	V		
Output High Voltage (1)	V_{OH1}	D_0-D_{13} R0-R3	V_{CC} − 1.0			V	$-I_{OH}$ = 1.0mA	
Output Low Voltage (1)	V_{OL1}				0.4	V	I_{OL} = 1.6mA	
Output High Voltage (2)	V_{OH2}	R5-R8	V_{CC} − 1.0			V	$-I_{OH}$ = 0.2mA	
Output Low Voltage (2)	V_{OL2}				0.4	V	I_{OL} = 0.3mA	
Input/Output Leakage Current	$\|I_{IL}\|$	D_0-D_{13} R0-R3, $R4_1$			1	μA	V_{in} = 0V to V_{CC}	1
		$R4_0$ (V_{PP})			TBD	μA	V_{in} = 0V to V_{CC}	

Notes: 1. Output buffer current is excluded.

A/D Converter Characteristics
(V_{CC} = 5 V ± 10 %, GND = 0 V, T_a = −40 to + 85°C, unless otherwise noted)

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
Resolution				8		bit		
Absolute Accuracy					±6	LSB		

Liquid Crystal Circuit Characteristics
(V_{CC} = 5 V ± 10 %, GND = 0 V, T_a = −40 to + 85°C, unless otherwise noted)

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
Segment Driver Descending Voltage	V_{ds}	SEG1-SEG28			0.6	V	I_d = 3μA	1
Common Driver Descending Voltage	V_{dc}	COM1-COM3			0.3	V	$-I_d$ = 3μA	1
LCD Power Supply Dividing Resistor	R_{WELL}		TBD	TBD	TBD	kΩ	V_{CC} = 5V	
LCD Frame Frequency	f_F	SEG1-SEG28		274.6		Hz	STATIC	
		COM1-COM3		137.3		Hz	1/2 duty	
				91.6		Hz	1/3 duty	

Note 1: Descending voltage from power supply pins V1, V2, V3, and GND to the segment and common pins.

PLL and IF Circuit Characteristics
(V_{CC} = 5 V ± 10 %, GND=GND (PLL) = 0 V, T_a = −40 to + 70°C, unless otherwise noted)

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
PLL Power Supply Voltage	V_{CC} (PLL)	V_{CC} (PLL)	$V_{CC}-0.3$	V_{CC}	$V_{CC}+0.3$	V		
Input Frequency	f_{in1}	PLL(P)	20	–	160	MHz	V_{in} = 0.3 V_{P-P}	
Input Frequency	f_{in2}	PLL(D)	0.4	–	20	MHz	V_{in} = 0.3 V_{P-P}	
Input Frequency	f_{in3}	IF	0.4	–	15	MHz	V_{in} = 0.3 V_{P-P}	

AC Characteristics
(V_{CC} = 5 V ± 10 %, GND = 0 V, T_a = −40 to + 85°C, unless otherwise noted)

Item		Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
Crystal Oscillation	Clock Oscillation Frequency	f_{OSC}	OSC_1, OSC_2		4.5		MHz		
	Instruction Cycle Time	t_{cyc}			1.78		μs	Divide-by-8	
External Interrupt Signal High, Low Width		t_{IH}, t_{IL}	$\overline{INT_0}$, INT_1, INT_2, INT_3, INT_4	2			t_{cyc}		1
RESET High Width		t_{RSTH}	RESET	2			t_{cyc}	Other than stop mode	2
				20			ms	Stop mode	
RESET Rise Time		t_{RSTr}	RESET			TBD	ms		2
Input Capacity		C_{in}	All pins other than $R4_0(V_{PP})$			15	pF	f=1 MHz, V_{in}=0V, T_a=25 C	
			$R4_0(V_{PP})$			TBD			
Power Supply Voltage Rise Time		t_{rcc}		TBD		TBD	ms		3

Notes: 1. See figure 68
2. See figure 69
3. See figure 70

Serial Interface Timing Characteristics
(V_{CC} = 5 V ± 10 %, GND = 0 V, T_a = −40 to + 85°C, unless otherwise noted.)

- At Transfer Clock Output

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition	Note
Transfer Clock Cycle Time	t_{Scyc}	$\overline{SCK_1}$, $\overline{SCK_2}$	1	—	—	t_{cyc}	Load circuit in Fig. 72	1,2
Transfer Clock "High" Level Width	t_{SCKH}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.4	—	—	t_{Scyc}		1,2
Transfer Clock "Low" Level Width	t_{SCKL}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.4	—	—	t_{Scyc}		1,2
Transfer Clock Rise Time	t_{SCKr}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	100	ns		1,2
Transfer Clock Fall Time	t_{SCKf}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	100	ns		1,2
Serial Output Data Delay Time	t_{DSO}	SO_1, SO_2	—	—	300	ns		1,2
Serial Input Data Set-up Time	t_{SSI}	SI_1, SI_2	300	—	—	ns		1
Serial Input Data Hold Time	t_{HSI}	SI_1, SI_2	150	—	—	ns		1

- At Transter Clock input

Item	Symbol	Pin Name	Min	Typ	Max	Unit	Test Condition	Note
Transfer Clock End Detection Time	t_{SCKHD}	$\overline{SCK_1}$, $\overline{SCK_2}$	1	—	—	t_{cyc}		1, 3
Transfer Clock "High" Level Width	t_{SCKH}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.4	—	—	t_{cyc}		1
Transfer Clock "Low" Level Width	t_{SCKL}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.4	—	—	t_{cyc}		1
Transfer Clock Rise Time	t_{SCKr}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	100	ns		1
Transfer Clock Fall Time	t_{SCKf}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	100	ns		1
Serial Output Data Delay Time	t_{DSO}	SO_1, SO_2	—	—	250	ns	Load circuit in Fig. 72	1,2
Serial Input Data Set-up Time	t_{SSI}	SI_1, SI_2	300	—	—	ns		1
Serial Input Data Hold Time	t_{HSI}	SI_1, SI_2	150	—	—	ns		1

Notes:
1. See figure 71
2. See figure 72
3. Transfer clock end detection time is an input transfer clock high level width after 8 transfer clock inputs. If the following clock is input before this, the serial interrupt request flag may not be set.

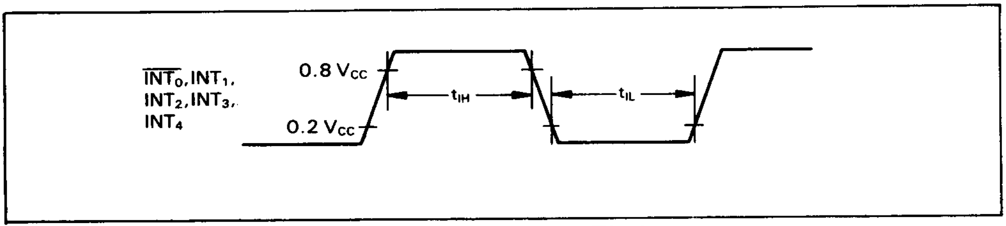


Figure 68. Interrupt Timing

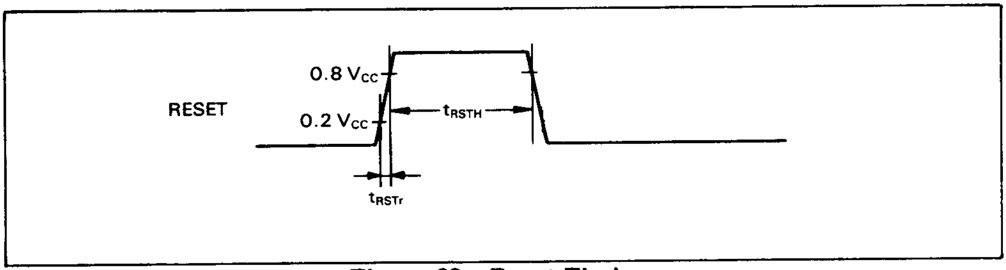


Figure 69. Reset Timing

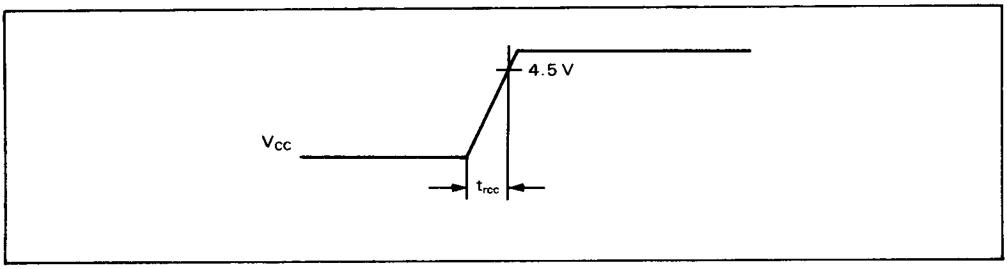
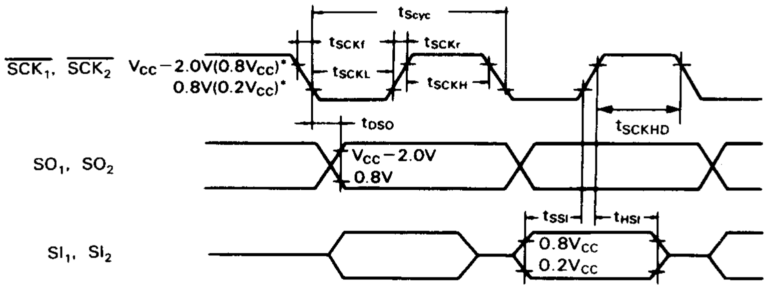


Figure 70. Supply Voltage Rise Time



* V_{CC}—2.0 V and 0.8 V are the threshold voltage for transfer clock output.
0.8 V_{CC} and 0.2 V_{CC} are the threshold voltage for transfer clock input.

Figure 71. Serial Interface Timing

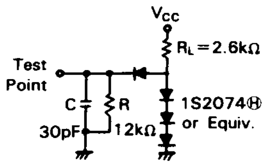


Figure 72. Timing Load Circuit

Programming Electrical Characteristics for HD4074509

DC Characteristics
(V_{CC} = 6 V ± 0.25 V, V_{PP} = 12.5 V ± 0.3 V, V_{SS} = 0 V, T_a = 25°C ± 5°C, unless otherwise noted)

Item		Symbol	Min	Typ	Max	Unit	Test Condition
Input high voltage	O_0—O_7, A_0—A_{14}, $\overline{OE}$, $\overline{CE}$	V_{IH}	2.2		V_{CC}+0.3	V	
Input low voltage	O_0—O_7, A_0—A_{14}, $\overline{OE}$, $\overline{CE}$	V_{IL}	−0.3		0.8	V	
Output high voltage	O_0—O_7	V_{OH}	2.4			V	I_{OH} = −200 μA
Output low voltage	O_0—O_7	V_{OL}			0.4	V	I_{OL} = 1.6 mA
Input leakage current	O_0—O_7, A_0—A_{14}, $\overline{OE}$, $\overline{CE}$	$\mid I_{LI} \mid$			2	μA	V_{in} = 5.25 V/0.5 V
V_{CC} current		I_{CC}			30	mA	
V_{PP} current		I_{PP}			40	mA	

AC Characteristics
(V_{CC} = 6 V ± 0.25 V, V_{PP} = 12.5 V ± 0.3 V, T_a = 25°C ± 5°C, unless otherwise noted)

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Address set-up time	t_{AS}	2			μs	Fig. 73 Input Pulse level: 0.8−2.2 V Input rise/fall time ≦ 20 ns Timing reference level input: 1.0 V, 2.0 V output: 0.8 V, 2.0 V
$\overline{OE}$ set-up time	t_{OES}	2			μs	
Data set-up time	t_{DS}	2			μs	
Address hold time	t_{AH}	0			μs	
Data hold time	t_{DH}	2			μs	
Data output disable time	t_{DF}			130	ns	
V_{PP} set-up time	t_{VPS}	2			μs	
Program pulse width	t_{PW}	0.95	1.0	1.05	ms	
$\overline{CE}$ pulse width when overprogramming	t_{OPW}	2.85		78.75	ms	
V_{CC} set-up time	t_{VCS}	2			μs	
Data output delay time	t_{OE}	0		500	ns	

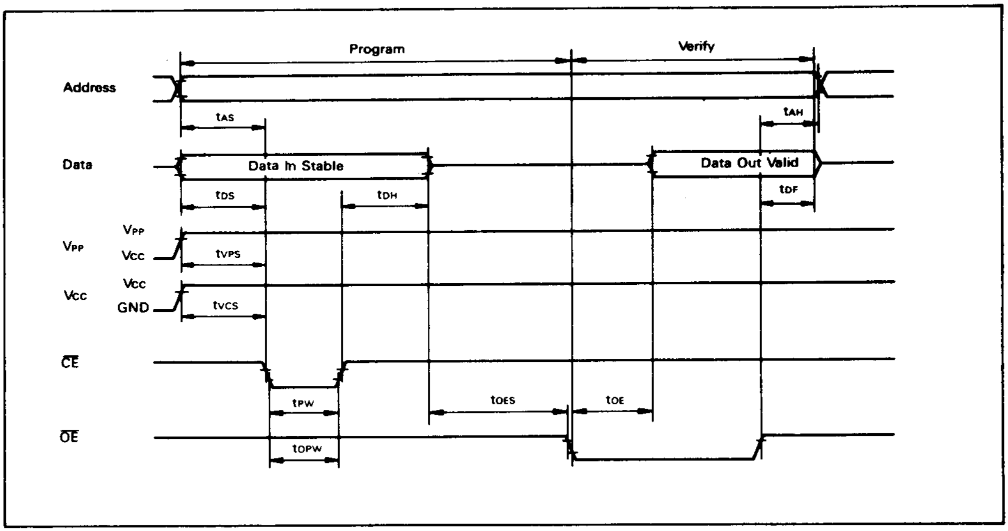
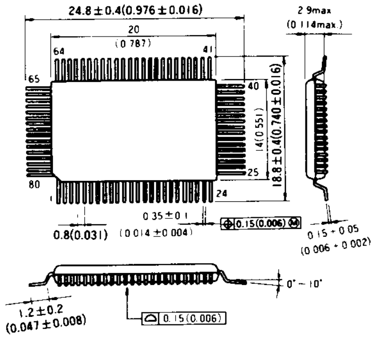


Figure 73. PROM Programming/Verify Timing

Package Dimensions

Unit: mm (inch)

FP-80B


24.8 ± 0.4(0.976 ± 0.016)
20
(0.787)
64
41
65
40
14(0.551)
18.8 ± 0.4(0.740 ± 0.016)
25
80
1
24
0.35 ± 0.1
(0.014 ± 0.004)
0.8(0.031)
⊕ 0.15(0.006) Ⓜ
2.9max
(0.114max.)
0.15 + 0.05
(0.006 + 0.002)
0° – 10°
1.2 ± 0.2
(0.047 ± 0.008)
⌓ 0.15(0.006)