HD4074008

Description

The HD4074008 is a ZTAT™ microcomputer incorporating 8 Kwords of programmable ROM and 512 digits of RAM. It is a CMOS 4-bit single-chip HMCS400-series microcomputer providing high programming productivity, high-speed operation, and low power dissipation.

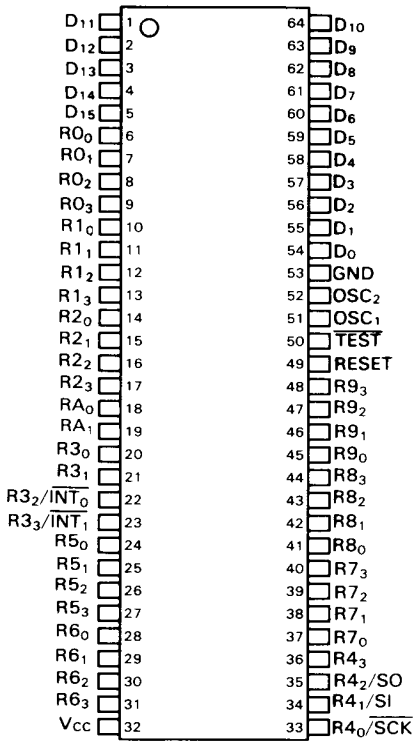
Features

- 8,192-word × 10-bit PROM (Programming compatible with the 27256 ROM)
- 512-digit × 4-bit RAM
- 58 I/O lines including 12 high-current (15 mA) I/O circuit type pins of open drain
- Two timer/counters
- Clock-synchronous 8-bit serial interface
- Five interrupt sources
  —Two by external sources
  —Three by internal sources
- Subroutine stack up to 16 levels, including interrupts
- Two low-power dissipation modes
  —Standby mode
  —Stop mode
- On-chip oscillator: Crystal or ceramic filter (Also externally drivable)
- Minimum instruction cycle time: 0.89 μs
- Package
  —64-pin shrink-type plastic DIP
  —64-pin shrink-type ceramic DIP with window
  —64-pin flat plastic package

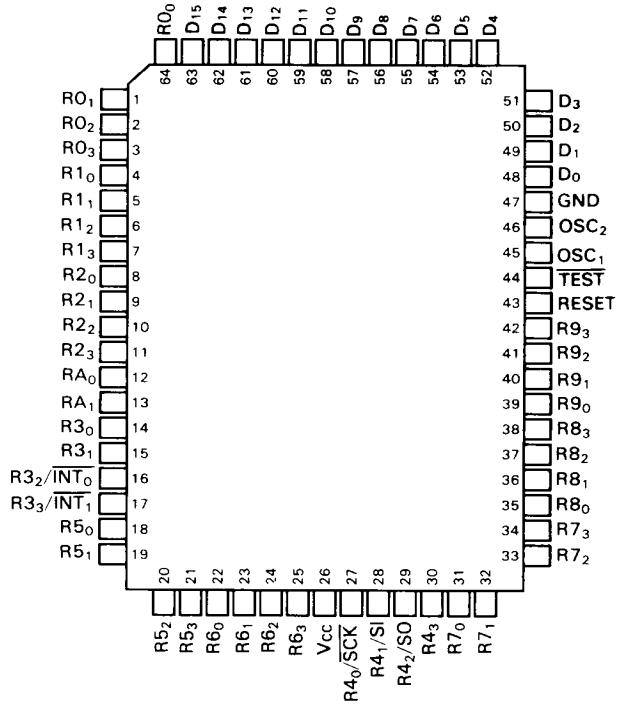
Ordering Information

Part Number	Clock Frequency	Package
HD4074008S	8 MHz	DP-64S
HD4074008C		DC-64S
HD4074008F		FP-64
HD4074008H		FP-64A

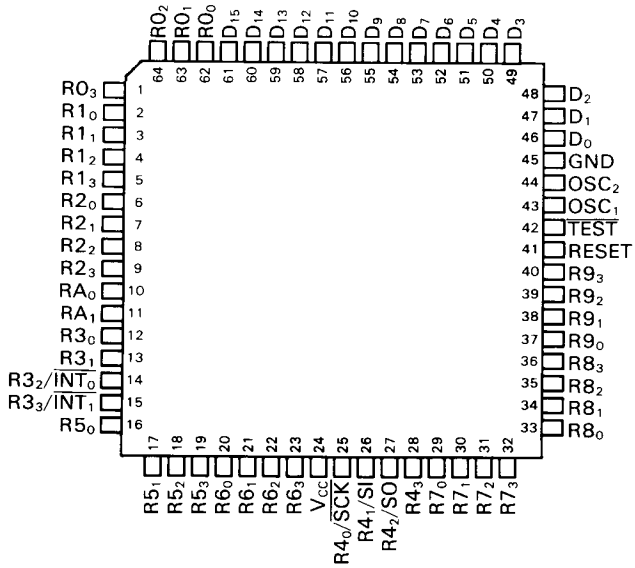
Pin Arrangement



(DP-64S, DC-64S)



(FP-64)



(FP-64A)

(Top view)

Pin Description

Pin Number			MCU Mode		PROM Mode	
DC-64S, DP-64S	FP-64	FP-64A	Symbol	I/O	Symbol	I/O
1	59	57	D_{11}	I/O	V_{CC}	
2	60	58	D_{12}	I/O		
3	61	59	D_{13}	I/O		
4	62	60	D_{14}	I/O		
5	63	61	D_{15}	I/O		
6	64	62	RO_0	O	A_1	I
7	1	63	RO_1	O	A_2	I
8	2	64	RO_2	O	A_3	I
9	3	1	RO_3	O	A_4	I
10	4	2	$R1_0$	I/O	A_5	I
11	5	3	$R1_1$	I/O	A_6	I
12	6	4	$R1_2$	I/O	A_7	I
13	7	5	$R1_3$	I/O	A_8	I
14	8	6	$R2_0$	I/O	A_0	I
15	9	7	$R2_1$	I/O	A_{10}	I
16	10	8	$R2_2$	I/O	A_{11}	I
17	11	9	$R2_3$	I/O	A_{12}	I
18	12	10	RA_0	I		
19	13	11	RA_1	I		
20	14	12	$R3_0$	I/O	A_{13}	I
21	15	13	$R3_1$	I/O	A_{14}	I
22	16	14	$R3_2/\overline{INT_0}$	I/O		
23	17	15	$R3_3/\overline{INT_1}$	I/O		
24	18	16	$R5_0$	I/O		
25	19	17	$R5_1$	I/O		
26	20	18	$R5_2$	I/O		
27	21	19	$R5_3$	I/O		
28	22	20	$R6_0$	O		
29	23	21	$R6_1$	O		
30	24	22	$R6_2$	O		
31	25	23	$R6_3$	O		
32	26	24	V_{CC}		V_{CC}	
33	27	25	$R4_0/\overline{SCK}$	I/O	O_4	I/O
34	28	26	$R4_1$/SI	I/O	O_5	I/O
35	29	27	$R4_2$/SO	I/O	O_6	I/O
36	30	28	$R4_3$	I/O	O_7	I/O
37	31	29	$R7_0$	O	$\overline{CE}$	I
38	32	30	$R7_1$	O	$\overline{OE}$	I
39	33	31	$R7_2$	O		
40	34	32	$R7_3$	O		
41	35	33	$R8_0$	O		
42	36	34	$R8_1$	O		
43	37	35	$R8_2$	O		
44	38	36	$R8_3$	O		
45	39	37	$R9_0$	I	V_{PP}	
46	40	38	$R9_1$	I	A_9	I
47	41	39	$R9_2$	I	$\overline{M_0}$	I
48	42	40	$R9_3$	I	$\overline{M_1}$	I
49	43	41	RESET	I	RESET	I
50	44	42	$\overline{TEST}$	I	$\overline{TEST}$	I
51	45	43	OSC_1	I		
52	46	44	OSC_2	O		
53	47	45	GND		GND	
54	48	46	D_0	I/O	O_0	I/O
55	49	47	D_1	I/O	O_1	I/O
56	50	48	D_2	I/O	O_2	I/O
57	51	49	D_3	I/O	O_3	I/O
58	52	50	D_4	I/O		
59	53	51	D_5	I/O		
60	54	52	D_6	I/O		
61	55	53	D_7	I/O		
62	56	54	D_8	I/O		
63	57	55	D_9	I/O		
64	58	56	D_{10}	I/O	V_{CC}	

Note: I/O: Input/Output Pins
I: Input Pins
O: Output Pins

Pin Functions

GND, V_{CC} (Power)

GND and V_{CC} are the power supply pins for the MCU. Connect GND to ground (0 V) and apply the V_{CC} power supply voltage to the V_{CC} pin.

$\overline{TEST}$ (Test)

$\overline{TEST}$ is for test purposes only. Connect it to V_{CC}.

RESET (Reset)

RESET resets the MCU. For details, see the Reset section.

OSC_1, OSC_2 (Oscillator Connections)

OSC_1 and OSC_2 are the pins for the internal oscillator circuit. They can be connected to a crystal resonator, ceramic filter resonator, or external oscillator circuits.

D_0–D_{15} (D Port)

The D port is an input/output port addressed by one bit. These 16 pins are all input, D_0 to D_3 are standard, and D_4 to D_{15} are large current standard pins. The circuit type for each pin can be selected using a mask option. For details, see the Input/Output section.

R0-RA (R Ports)

R0-R9 are 4-bit I/O ports. RA is a 2-bit I/O port. R0, R6, R7, and R8 are output ports, R9 and RA are input ports, and R1 to R5 are I/O ports. All pins of port R0-RA are standard pins. The circuit type of D_4-D_{15} and R0-R2 is PMOS open drain, and that of D_0-D_3 and R3-R8 is NMOS open drain. $R3_2$, $R3_3$ and $R4_0$, $R4_1$, $R4_2$ are multiplexed with $\overline{INT_0}$, $\overline{INT_1}$, $\overline{SCK}$, SI, and SO, respectively. The RA_1 pin should be used as RA_1 since it is PMOS open drain (MOS without pull-down). For details, see the Input/Output section

$\overline{INT_0}$, $\overline{INT_1}$ (Interrupts)

$\overline{INT_0}$ and $\overline{INT_1}$ are external interrupts for the MCU. $\overline{INT_1}$ can be used as an external event input pin for timer B. $\overline{INT_0}$ and $\overline{INT_1}$ are multiplexed with $R3_2$ and $R3_3$, respectively. For details, see the Interrupt section.

$\overline{SCK}$, SI, SO (Serial Interface)

The transmit clock I/O pin ($\overline{SCK}$), serial data input pin (SI), and serial data output pin (SO) are used for serial interface. $\overline{SCK}$, SI, and SO are multiplexed with $R4_0$, $R4_1$, and $R4_2$, respectively. For details, see the Serial Interface section.

V_{PP} (Program Voltage)

V_{PP} is the input for the program voltage (12.5 V $\pm$ 0.3 V) for programming the PROM.

$\overline{CE}$ (Chip Enable)

$\overline{CE}$ is the input for programming and verifying the internal PROM.

$\overline{OE}$ (Output Enable)

$\overline{OE}$ is the input of the data output control signal for verification.

A_0–A_{14} (Address Bus)

A_0-A_{14} are address input pins for the internal PROM.

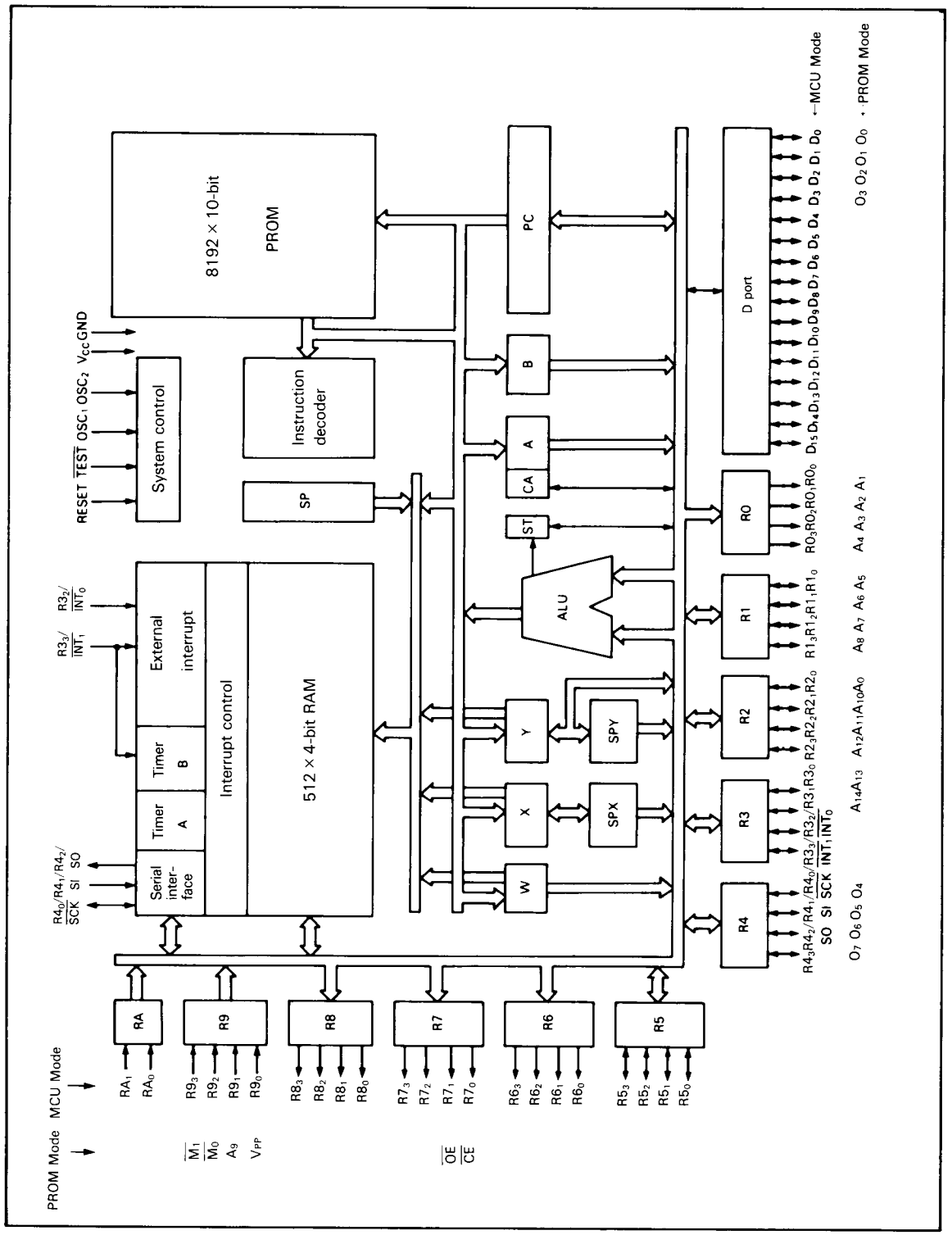
O_0–O_7 (PROM Data Bus)

These are data bus pins for the internal PROM.

$\overline{M_0}$, $\overline{M_1}$ (Mode)

$\overline{M_0}$ and $\overline{M_1}$, set the PROM mode. The PROM mode is set when $\overline{M_0}$, $\overline{M_1}$, and $\overline{TEST}$ pins are low and the RESET pin is high.

Block Diagram



Memory Map

ROM Memory Map

The MCU includes a 8,192-word × 10-bit ROM. It is described in the following paragraphs with the ROM memory map in figure 1.

Vector Address Area ($0000 to $000F): Locations $0000 through $000F are reserved for JMPL instructions to branch to the starting address of the initialization program and of the interrupt programs. After reset or an interrupt, the program is executed from the vector address.

Zero-Page Subroutine Area ($0000 to $003F): Locations $0000 through $003F are reserved for subroutines. The CAL instruction branches to these subroutines.

Pattern Area ($0000 to $0FFF): Locations $0000 through $0FFF are reserved for PROM data. The P instruction can reference the PROM data as a pattern.

Program Area ($0000 to $1FFF): Locations from $0000 to $1FFF can be used for program code.

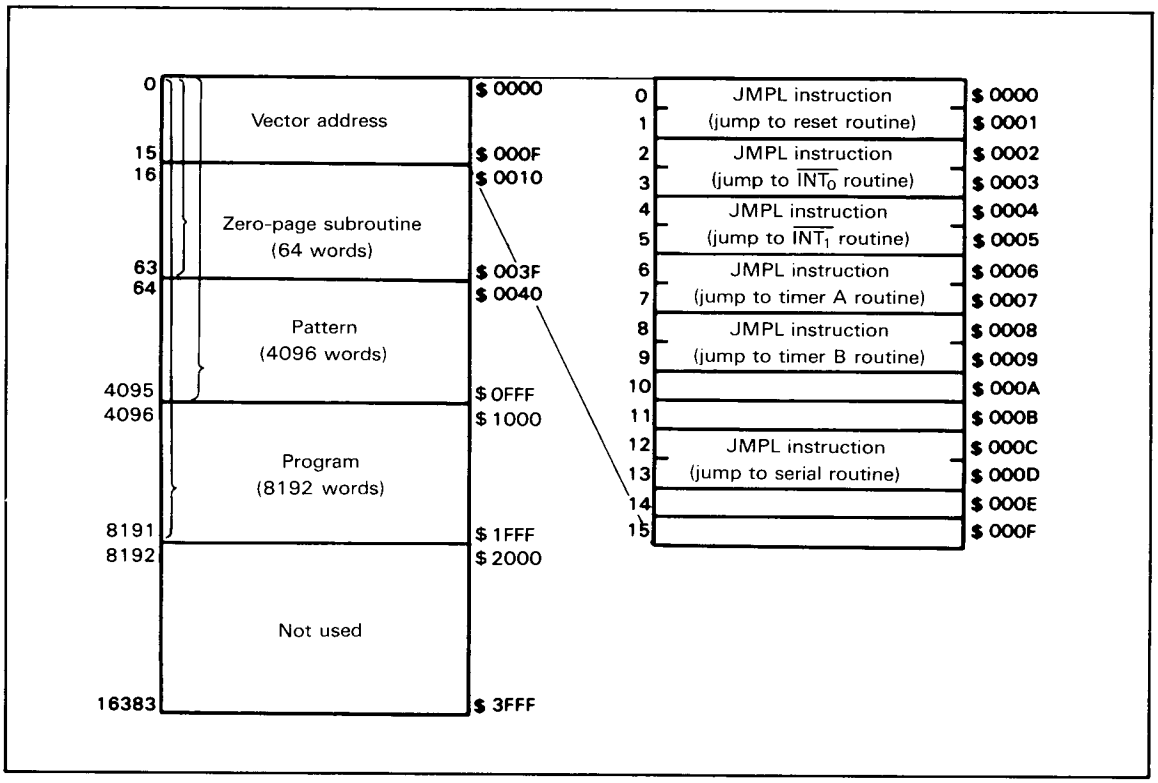


Figure 1 ROM Memory Map

RAM Memory Map

The MCU also contains a 512-digit × 4-bit RAM as the data and stack area. In addition to these areas, interrupt control bits and special function registers are also mapped on the RAM memory space. The RAM memory map (figure 2) is described in the following paragraphs.

Interrupt Control Bits Area ($000 to $003): The interrupt control bits area (figure 3) is used for interrupt control. It is accessible only by RAM bit manipulation instructions. However, the interrupt request flag cannot be set by software. The RSP bit is used only to reset the stack pointer.

Special Function Registers Area ($004 to $00B): The special function registers are the mode or data registers for the external interrupt, the serial interface, and the timer/counters. These registers are classified into three types: write-only, read-only, and read/write, as shown in figure 2. These registers cannot be accessed by RAM bit manipulation instructions.

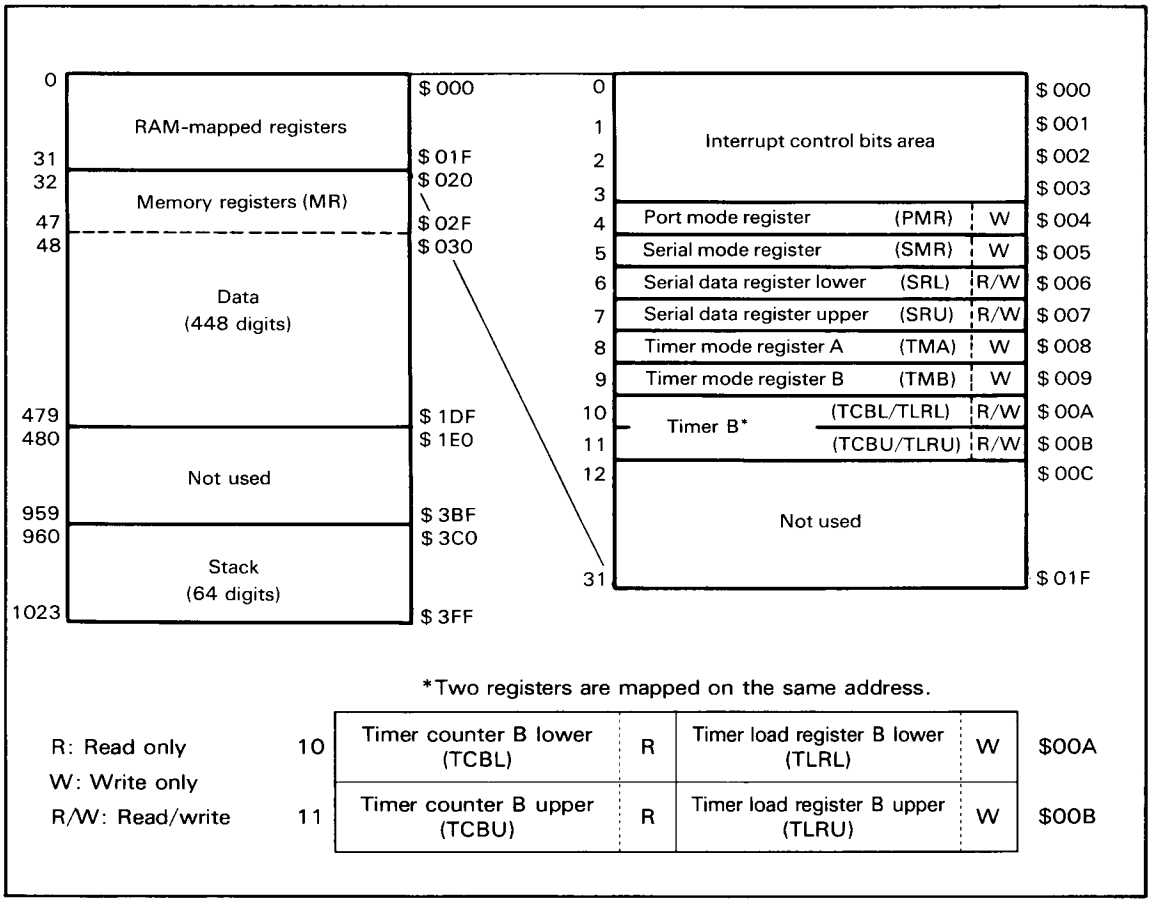


Figure 2 RAM Memory Map

Data Area ($020 to $1DF): The 16 digits of $020 through $02F are called memory registers (MR) and are accessible by the LAMR and XMRA instructions (figure 4).

Stack Area ($3C0 to $3FF): Locations $3C0 through $3FF are reserved for LIFO stacks to save the contents of the program counter (PC), status flag (ST), and carry flag (CA) when subroutine calls (CAL or CALL instruction) and interrupts are processed. This area can be used as a 16-level nesting stack in which one level requires 4 digits. Figure 4 shows the save condition. The program counter is restored by the RTN and RTNI instructions. Status and carry flags are restored only by the RTNI instruction. This area, when not used as a stack, is available as a data area.

	Bit 3	Bit 2	Bit 1	Bit 0	
0	IM0 (IM of $\overline{INT_0}$)	IF0 (IF of $\overline{INT_0}$)	RSP (Reset SP bit)	IE (Interrupt enable flag)	$000
1	IMTA (IM of timer A)	IFTA (IF of timer A)	IM1 (IM of $\overline{INT_1}$)	IF1 (IF of $\overline{INT_1}$)	$001
2	Not used	Not used	IMTB (IM of timer B)	IFTB (IF of timer B)	$002
3	Not used	Not used	IMS (IM of serial)	IFS (IF of serial)	$003

IF: Interrupt request flag
IM: Interrupt mask
IE: Interrupt enable flag
SP: Stack pointer
Note: Each bit of the interrupt control bit area is set by the SEM/SEMD instruction, reset by the REM/REMD instruction, and 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 value of the status flag becomes invalid when unusable bits are tested.

Figure 3 Configuration of Interrupt Control Bits Area

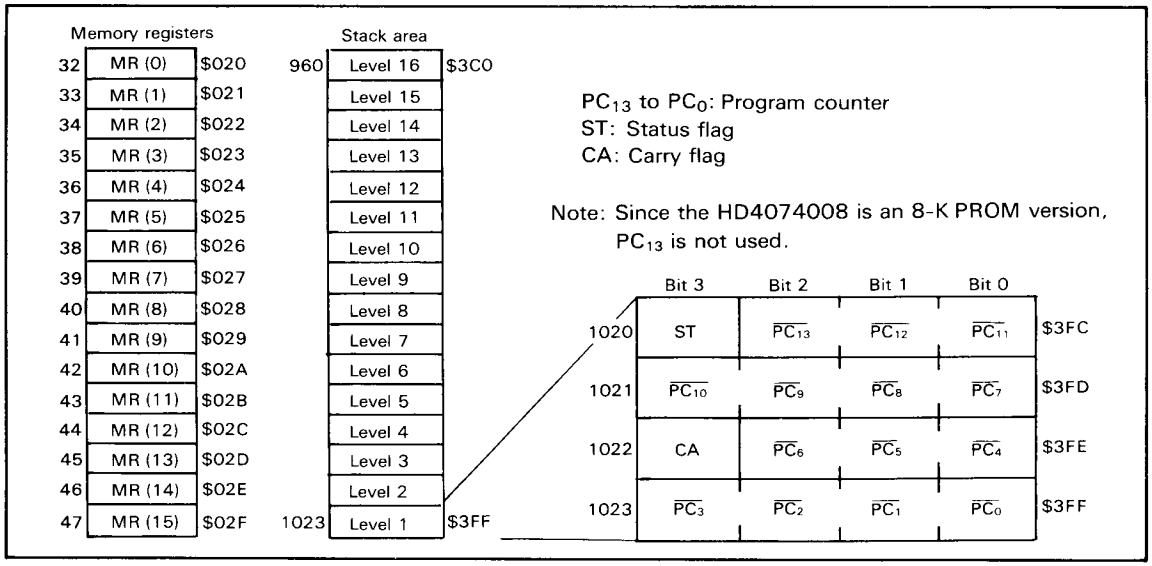


Figure 4 Configuration of Memory Registers, Stack Area, and Stack Position

Functional Description

Registers and Flags

The MCU has nine registers and two flags for the CPU operations (figure 5).

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

W Register (W), X Register (X), Y Register (Y): The 2-bit W register, and the 4-bit X and Y registers indirectly address RAM. The 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 Flag (CA): The carry flag (CA) stores the overflow from the ALU generated by an arithmetic operation. It is also affected by the SEC, REC, ROTL, and ROTR instructions.

During an interrupt, the carry flag is pushed onto the stack. It is restored by the RTNI instruction, but not by the RTN instruction.

Status Flag (ST): The status flag (ST) holds the ALU overflow, non-zero, and the results of a bit test instruction for the arithmetic or compare instructions. It is a branch condition of the BR, BRL, CAL, or CALL instructions. The value for the status flag remains unchanged until the next arithmetic, compare, or bit test instruction is executed. The status flag becomes a 1 after the BR, BRL, CAL, or CALL instruction was either executed or skipped. During an interrupt, the status flag is pushed onto the stack. It is restored back from the stack by the RTNI instruction, but not by the RTN instruction.

Program Counter (PC): The program counter is a 14-bit binary counter which controls the sequence in which the instructions stored in ROM are executed.

Stack Pointer (SP): The stack pointer (SP) is used to point to the address of the next stack area (up to 16 levels).

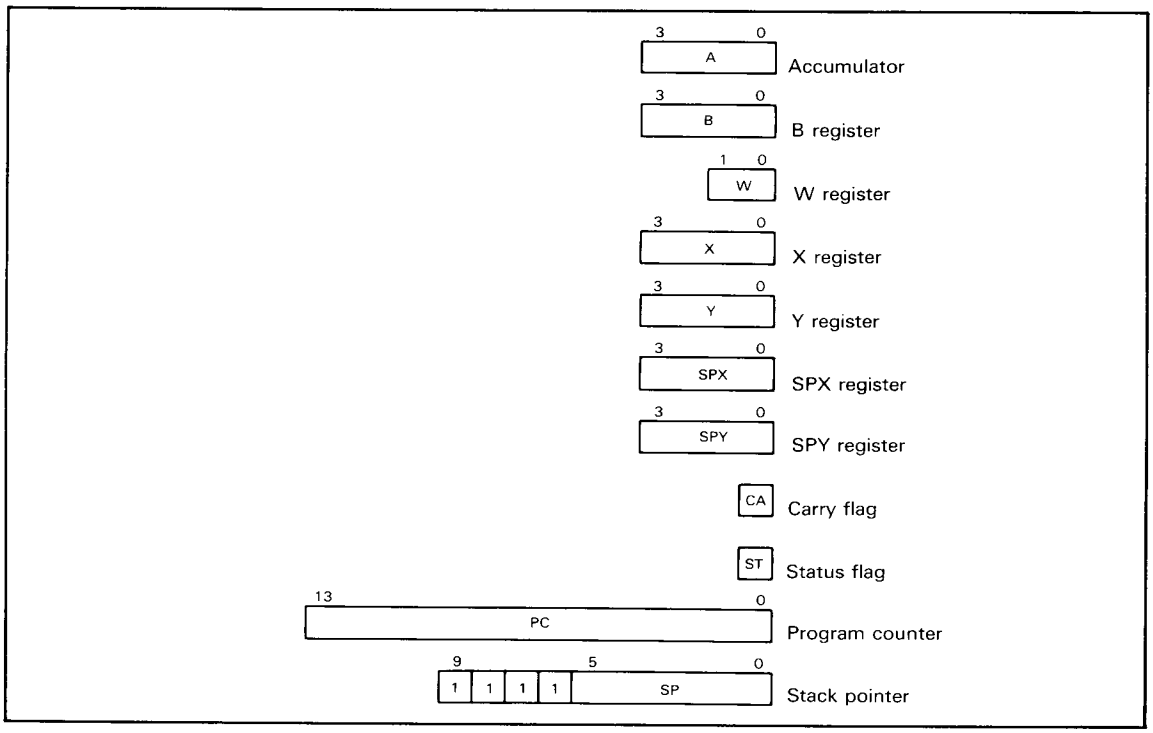


Figure 5 Registers and Flags

The stack pointer is initialized to RAM address $3FF. It is decremented by 4 when data is pushed onto the stack, and incremented by 4 when data is restored from it. The stack can only be used up to 16 levels deep because the high four bits of the stack pointer are fixed at 1111.

The stack pointer is initialized to $3FF either by MCU reset or by the RSP bit reset from the REM/REMD instruction.

Interrupts

Five interrupt sources are available on the MCU: external requests ($\overline{INT_0}$, $\overline{INT_1}$), timer/counters (timers A and B), and the serial port.

For each source, an interrupt request flag (IF), interrupt mask (IM), and interrupt vector addresses control and maintain the interrupt request. The interrupt enable flag (IE) also controls interrupt operations.

Interrupt Control Bits and Interrupt Processing: The interrupt control bits are mapped on $000 through $003 of the RAM space. They are accessible by RAM bit manipulation instructions. (The interrupt request flag (IF) cannot be set by software.) The interrupt enable flag (IE) and IF are cleared to 0, and the interrupt mask (IM) is set to 1 by MCU reset.

Figure 6 is a block diagram of the interrupt

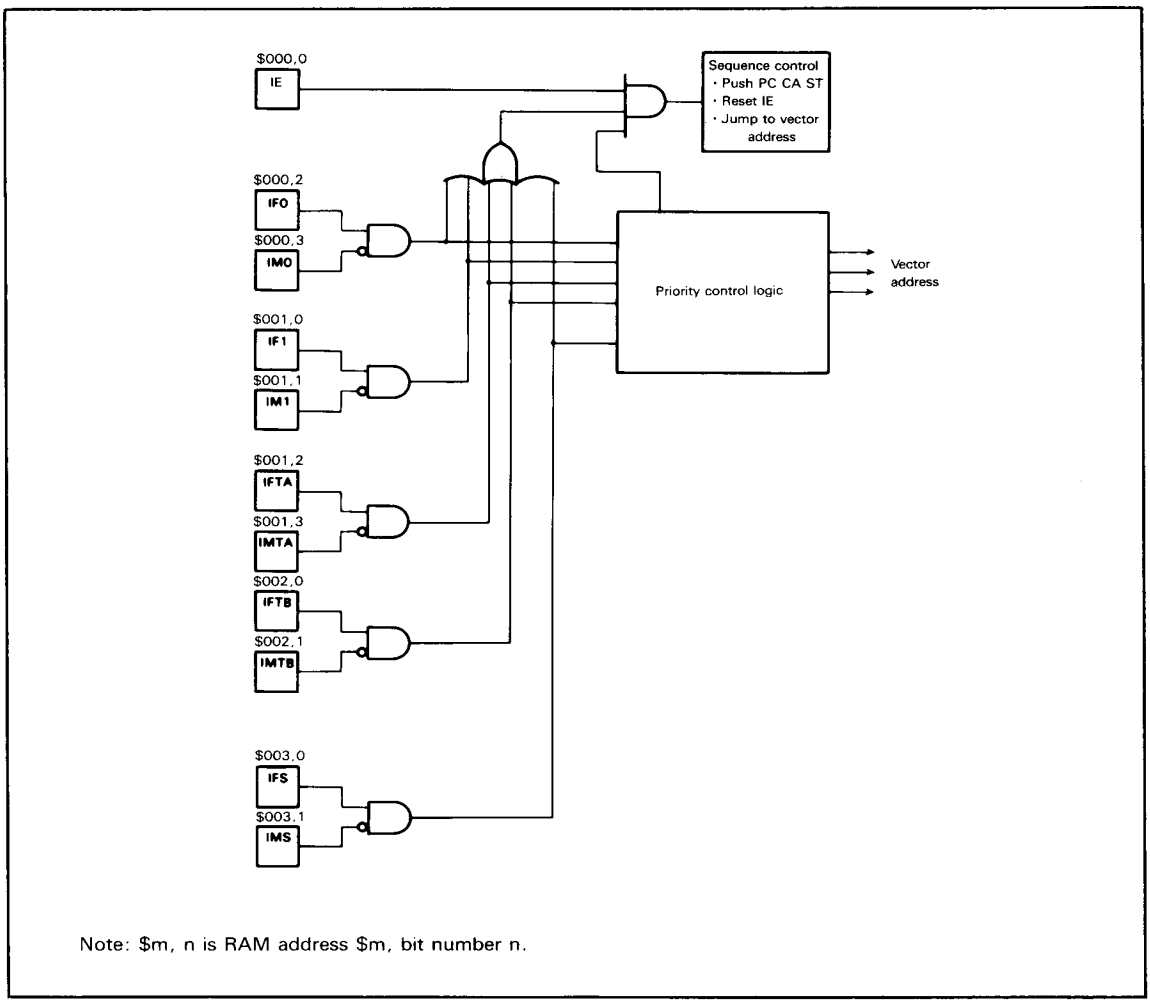


Figure 6 Interrupt Control Circuit Block Diagram

control circuit. Table 1 shows the interrupt priority and vector addresses, and table 2 shows the interrupt conditions corresponding to each interrupt source.

An interrupt request is generated when IF is set to 1 and IM is 0. If IE is 1 at this time, the interrupt will be activated and vector addresses will be generated from the priority PLA corresponding to the interrupt source.

Figure 7 shows the interrupt sequence, and figure 8 shows the interrupt processing flowchart. If an interrupt is requested, the instruction being executed finishes in the first cycle. The IE is reset in the second cycle. In the second and third cycles, the carry flag, status flag, and program counter are pushed onto the stack. In the third cycle, the instruction is re-executed after jumping to the vector address.

Table 1 Vector Addresses and Interrupt Priority

Reset/Interrupt	Priority	Vector Addresses
RESET	–	$0000
$\overline{INT_0}$	1	$0002
$\overline{INT_1}$	2	$0004
Timer A	3	$0006
Timer B	4	$0008
Serial	5	$000C

Table 2 Conditions of Interrupt Service

	Interrupt Source				
Interrupt Control Bit	**$\overline{INT_0}$**	**$\overline{INT_1}$**	**Timer A**	**Timer B**	**Serial**
IE	1	1	1	1	1
IF0·$\overline{IM0}$	1	0	0	0	0
IF1·$\overline{IM1}$	*	1	0	0	0
IFTA·$\overline{IMTA}$	*	*	1	0	0
IFTB·$\overline{IMTB}$	*	*	*	1	0
IFS·$\overline{IMS}$	*	*	*	*	1

* Don't care

Table 3 Interrupt Enable Flag

IE	Interrupt Enable/Disable
0	Disable
1	Enable

Table 4 External Interrupt Request Flags

IF0, IF1	Interrupt Request
0	No
1	Yes

Table 5 External Interrupt Masks

IM0, IM1	Interrupt Request
0	Enable
1	Disable (Mask)

At each vector address, program the JMPL instruction to branch to the starting address of the interrupt program. The IF which caused the interrupt must be reset by software in the interrupt program.

Interrupt Enable Flag (IE: $000, Bit 0): The interrupt enable flag enables/disables interrupt requests as shown in table 3. It is reset by an interrupt and set by the RTNI instruction.

External Interrupts ($\overline{INT_0}$, $\overline{INT_1}$): The external interrupt request inputs ($\overline{INT_0}$, $\overline{INT_1}$) can be selected by the port mode register (PMR: $004). Setting bit 3 and bit 2 of PMR causes the $R3_3/\overline{INT_1}$ and $R3_2/\overline{INT_0}$ pins to be used as $\overline{INT_1}$ and $\overline{INT_0}$, respectively.

The external interrupt request flags (IF0, IF1) are set at the falling edge of $\overline{INT_0}$ and $\overline{INT_1}$ inputs. (Refer to table 4.)

The $\overline{INT_1}$ input can be used as a clock signal input to timer B, in which timer B counts up at each falling edge of the $\overline{INT_1}$ input. When using $\overline{INT_1}$ as the timer B external event input, the external interrupt mask (IM1) has to be set so that the interrupt request by $\overline{INT_1}$ will not be accepted. (Refer to table 5.)

External Interrupt Request Flags (IF0: $000, Bit 2; IF1: $001, Bit 0): The external interrupt request flags (IF0, IF1) are set at the falling edge of the $\overline{INT_0}$ and $\overline{INT_1}$ inputs, respectively.

Table 6 Port Mode Register

PMR3	$R3_3/\overline{INT_1}$ Pin
0	Used as $R3_3$ port input/output pin
1	Used as $\overline{INT_1}$ input pin

PMR1	$R4_1$/SI Pin
0	Used as $R4_1$ port input/output pin
1	Used as SI input pin

PMR2	$R3_2/\overline{INT_0}$ Pin
0	Used as $R3_2$ port input/output pin
1	Used as $\overline{INT_0}$ input pin

PMR0	$R4_2$/SO Pin
0	Used as $R4_2$ port input/output pin
1	Used as SO output pin

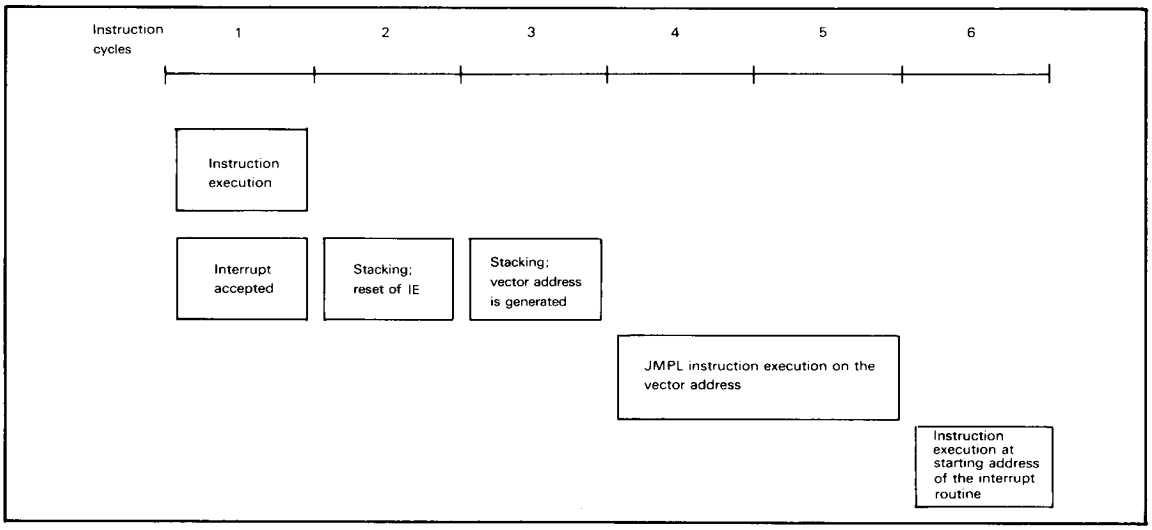


Figure 7 Interrupt Processing Sequence

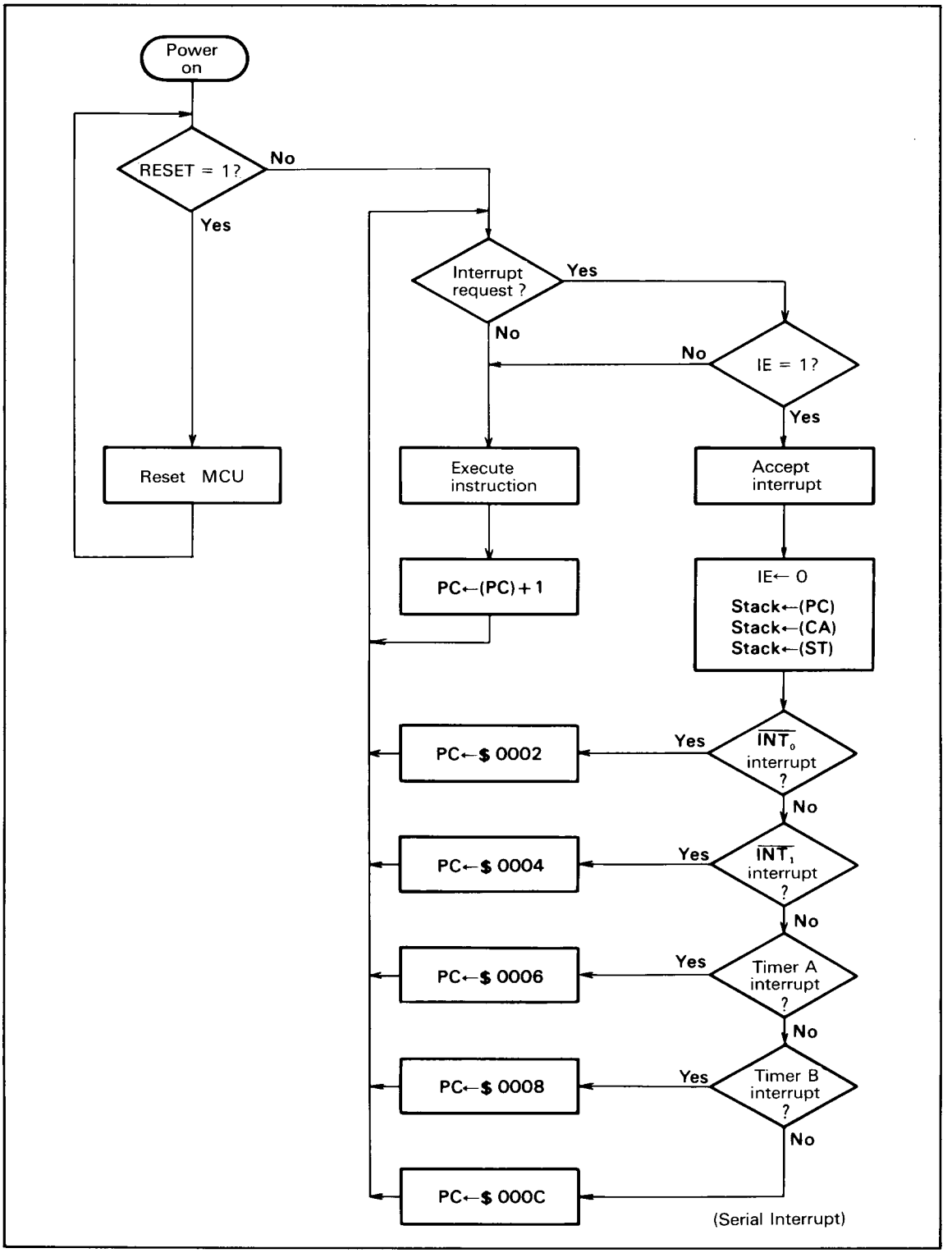


Figure 8 Interrupt Processing Flowchart

External Interrupt Masks (IM0: $000, Bit 3; IM1: $001, Bit 1): The external interrupt masks mask the external interrupt requests.

Port Mode Register (PMR: $004): The port mode register is a 4-bit write-only register which controls the $R3_2/\overline{INT_0}$, $R3_3/\overline{INT_1}$, $R4_1$/SI, and $R4_2$/SO pins as shown in table 6. The port mode register will be initialized to $0 by MCU reset. These pins are therefore initially used as ports.

Serial Interface

The serial interface is used to transmit/receive 8-bit data serially. It consists of the serial data register, the serial mode register, the octal counter, and the multiplexer as illustrated in figure 9. Pin $R4_0/\overline{SCK}$ and the transmit clock signal are controlled by the serial mode register. 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 transmit clock signal.

The STS instruction is used to initiate serial interface operations and to reset the octal counter to $0. The counter starts to count at the falling edge of the transmit clock ($\overline{SCK}$) signal and increments by one at the rising edge of $\overline{SCK}$. When the octal counter is reset to $0 after eight transmit clock signals, or when a transmit/receive operation is discontinued by resetting the octal counter, the serial interrupt request flag will be set.

Serial Mode Register (SMR: $005): The 4-bit write-only serial mode register controls the $R4_0/\overline{SCK}$ pin, prescaler divide ratio, and transmit clock source as shown in table 7.

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

The write signal to the serial mode register stops the serial data register and octal counter from accepting the transmit 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 will be changed on the second instruction cycle after the serial mode register has been written to. 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.

Serial Data Register (SRL: $006, SRU: $007): The 8-bit read/write serial data register consists of a low-order digit (SRL: $006) and a high-order digit (SRU: $007).

The data in the serial data register will be output from the SO pin, from LSB to MSB, synchronously with the falling edge of the transmit clock signal. At the same time, external data will be input from the SI pin to the serial data register, MSB first, synchronously with the rising edge of the transmit clock. Figure 10 shows the I/O timing chart for the transmit 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.

Serial Interrupt Request Flag (IFS: $003, Bit 0): The serial interrupt request flag will be set when the octal counter counts eight transfer clock signals, or when data transfer is discontinued by resetting the octal counter. Refer to table 8.

Serial Interrupt Mask (IMS: $003, Bit 1): The serial interrupt mask masks the interrupt request. Refer to table 9.

Selection and Change of the Operation Mode: Table 10 shows the serial interface operation modes which are determined by a combination of the value in the port mode register and in the serial mode register.

Initialize the serial interface by a write signal to the serial mode register, when the operation mode has changed.

Table 7 Serial Mode Register

SMR3	$R4_0/\overline{SCK}$
0	Used as $R4_0$ port input/output pin
1	Used as $\overline{SCK}$ input/output pin

			Transmit Clock			
SMR2	SMR1	SMR0	$R4_0/\overline{SCK}$ Port	Clock Source	Prescaler Divide Ratio	System Clock Divide Ratio
0	0	0	$\overline{SCK}$ output	Prescaler	÷ 2048	÷ 4096
0	0	1	$\overline{SCK}$ output	Prescaler	÷ 512	÷ 1024
0	1	0	$\overline{SCK}$ output	Prescaler	÷ 128	÷ 256
0	1	1	$\overline{SCK}$ output	Prescaler	÷ 32	÷ 64
1	0	0	$\overline{SCK}$ output	Prescaler	÷ 8	÷ 16
1	0	1	$\overline{SCK}$ output	Prescaler	÷ 2	÷ 4
1	1	0	$\overline{SCK}$ output	System clock	—	÷ 1
1	1	1	$\overline{SCK}$ input	External clock	—	—

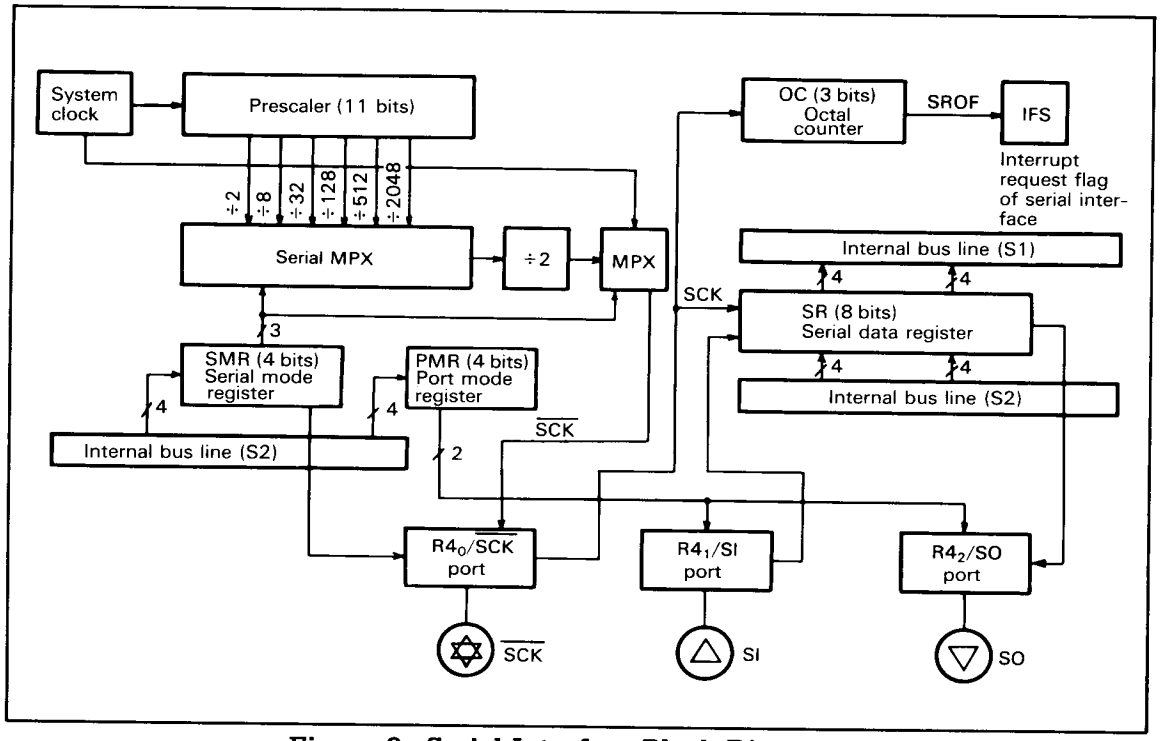


Figure 9 Serial Interface Block Diagram

Operating State of Serial Interface: The serial interface has three operating states: the STS waiting state, transmit clock wait state, and transfer state, as shown in figure 11.

The STS waiting state is the initialization state of the serial interface internal state. The serial interface enters this state in one of two ways: either by changing the operation mode through a data change in the port mode register, or by writing data into the serial mode register. In this state, the serial interface does not operate even if the transmit clock is applied. If the STS instruction is executed, the serial interface shifts to the transmit clock wait state.

In the transmit clock wait state the falling edge of the first transmit clock causes the serial interface to shift to the transfer state. While the octal counter counts up, the serial data register shifts simultaneously. As an exception, if the clock continuous output mode is selected, the serial interface stays in the transmit clock wait state while the transmit clock outputs continuously.

The octal counter becomes 000 again after 8 transmit clocks or by the execution of the STS instruction, so that the serial interface returns to the transmit clock wait state, and the serial interrupt request flag is set simultaneously.

When the internal transmit clock is selected, the transmit clock output is triggered by the execution of the STS instruction, and stops after 8 clocks.

Table 8 Serial Interrupt Request Flag

IFS	Interrupt Request
0	No
1	Yes

Table 9 Serial Interrupt Mask

IMS	Interrupt Request
0	Enable
1	Disable (Mask)

Table 10 Serial Interface Operation Mode

SMR3	PMR1	PMR2	Serial Interface Operating Mode
1	0	0	Clock continuous output mode
1	0	1	Transmit mode
1	1	0	Receive mode
1	1	1	Transmit/receive mode

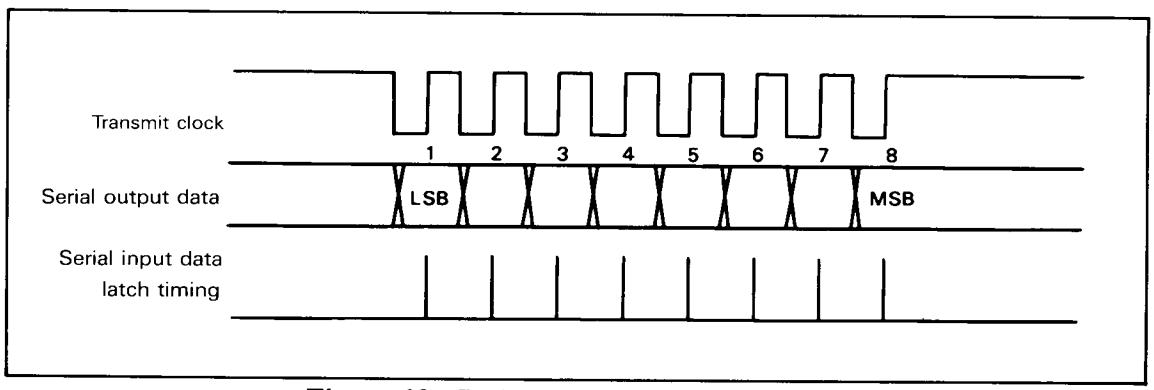


Figure 10 Serial Interface I/O Timing

Example of Transmit Clock Error Detection: The serial interface functions abnormally when the transmit clock is disturbed by external noise. In this case, transmit clock errors can be detected by the procedure shown in figure 12.

If more than 8 transmit clocks occur in the transmit clock wait state, the state of the serial interface shifts as follows: transfer state, transmit clock wait state, and transfer state. The serial interrupt flag should be reset before entering into the STS state by writing data to SMR. This procedure causes the serial IFS to be set again.

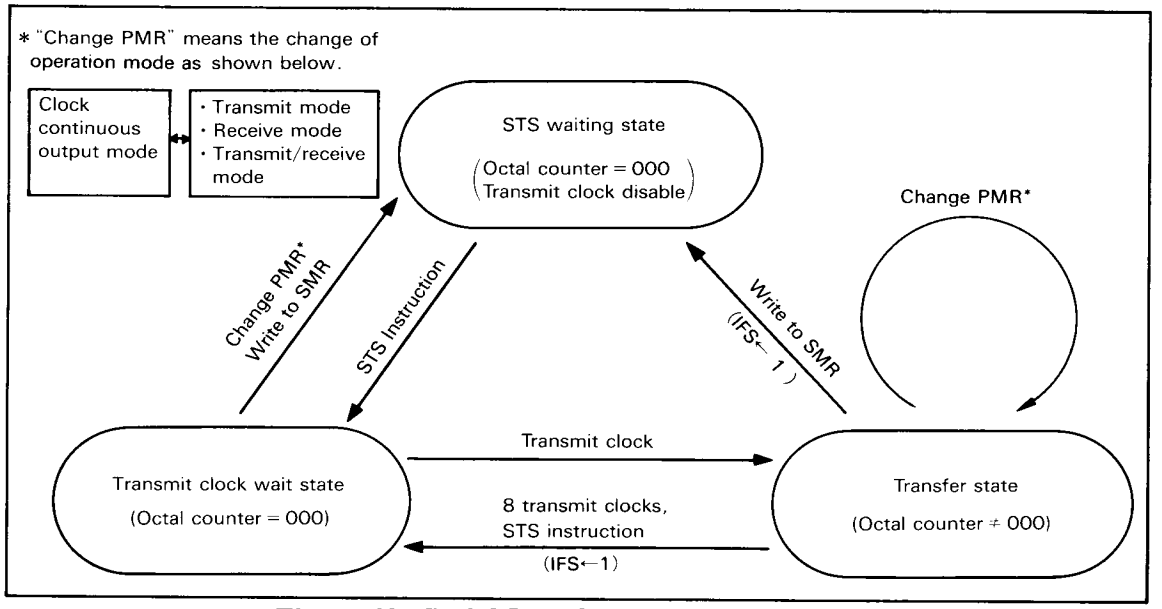


Figure 11 Serial Interface Mode Transition

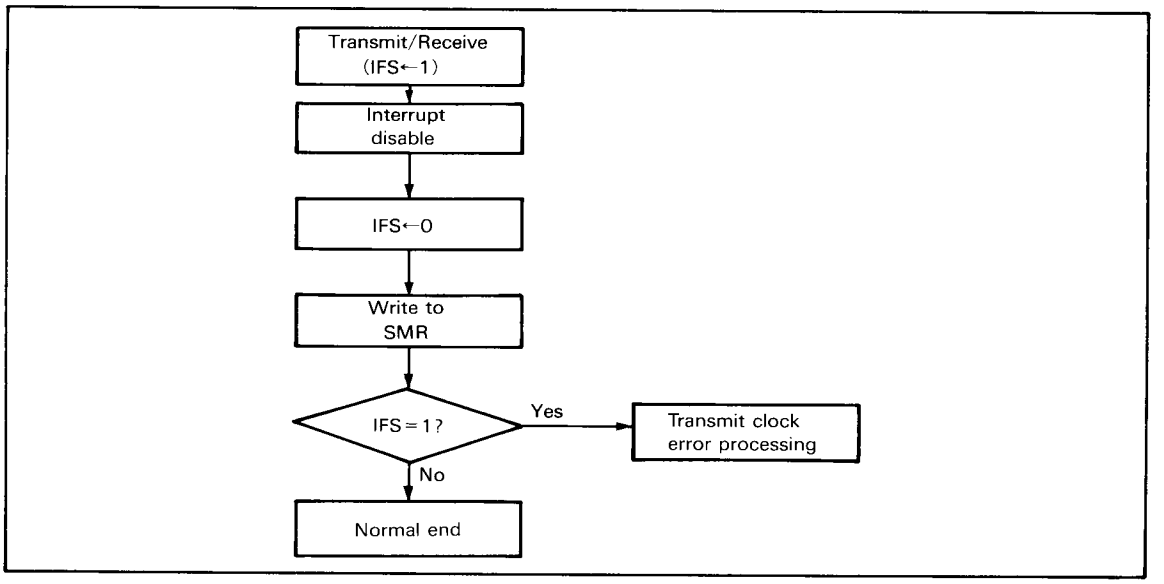


Figure 12 Transmit Clock Error Detection

Timers

The MCU contains a prescaler and two timer/counters (timers A and B). A block diagram is shown in figure 13. The prescaler is an 11-bit binary counter, timer A is an 8-bit free-running timer/counter, and timer B is an 8-bit auto-reload timer/event counter.

Prescaler: The input to the prescaler is the system clock signal. The prescaler is initialized to $0000 by MCU reset, and it starts to count up the system clock signal as soon as RESET input goes to logic 0. The prescaler keeps counting up except at MCU reset and stop mode. The prescaler provides clock signals to timer A, timer B, and the serial interface. The prescaler divide ratio is selected by timer mode register A (TMA), timer mode register B (TMB), and serial mode register (SMR).

Timer A Operation: After timer A is initialized to $00 by MCU reset, it counts up at every clock input signal. When the next clock signal is applied after timer A becomes $FF, it will generate an overflow and become $00. This overflow causes the timer A interrupt request flag (IFTA: $001, bit 2) to go to 1. This timer can function as an interval timer periodically generating overflow output at every 256th clock signal input.

The clock input signals to timer A are selected by timer mode register A (TMA: $008).

Timer B Operation: Timer mode register B (TMB: $009) selects the auto-reload function, input clock source, and the prescaler divide ratio of timer B. When the external event input is used as an input clock signal to timer B, select $R3_3/\overline{INT_1}$ as $\overline{INT_1}$ and set the external interrupt mask (IM1) to prevent an external interrupt request from occurring.

Timer B is initialized according to the data written into timer load register B by software. Timer B counts up at every clock input signal. When the next clock signal is applied to timer B after it is set to $FF, it will generate an overflow output. In this case, if the autoreload function is selected timer B is initialized according to the value of timer load register

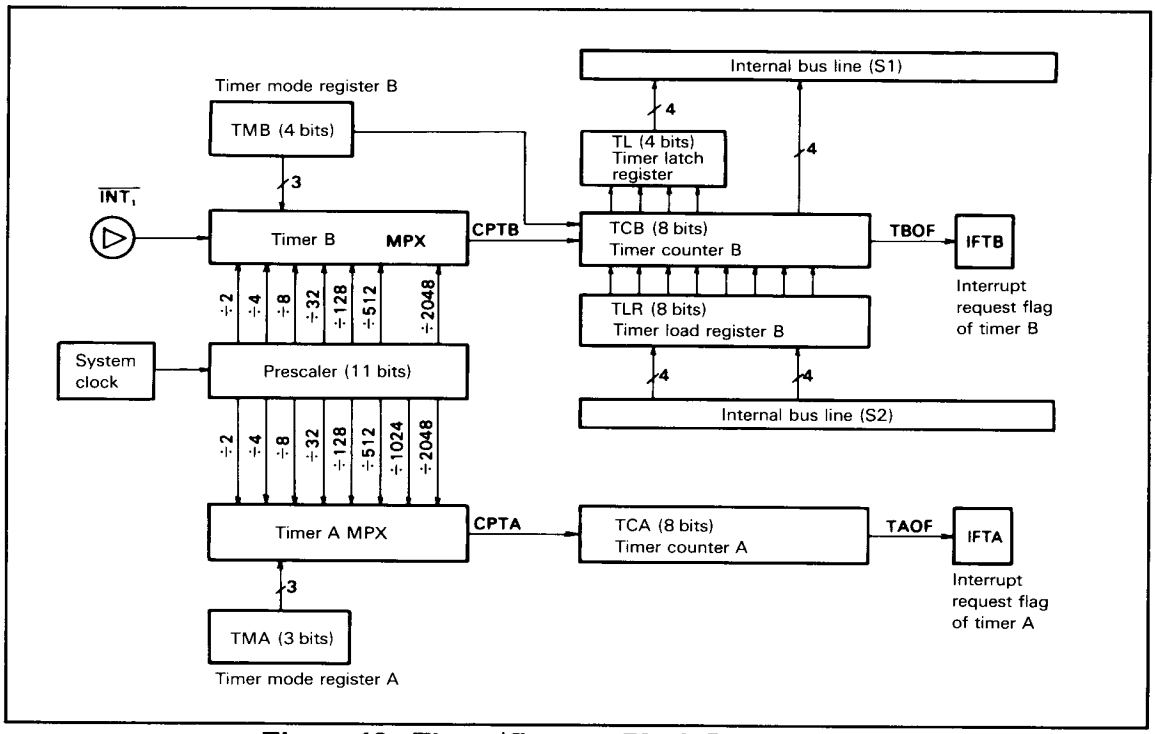


Figure 13 Timer/Counter Block Diagram

B. If it is not selected, timer B goes to $00. The timer B interrupt request flag (IFTB: $002, bit 0) will be set at this overflow output.

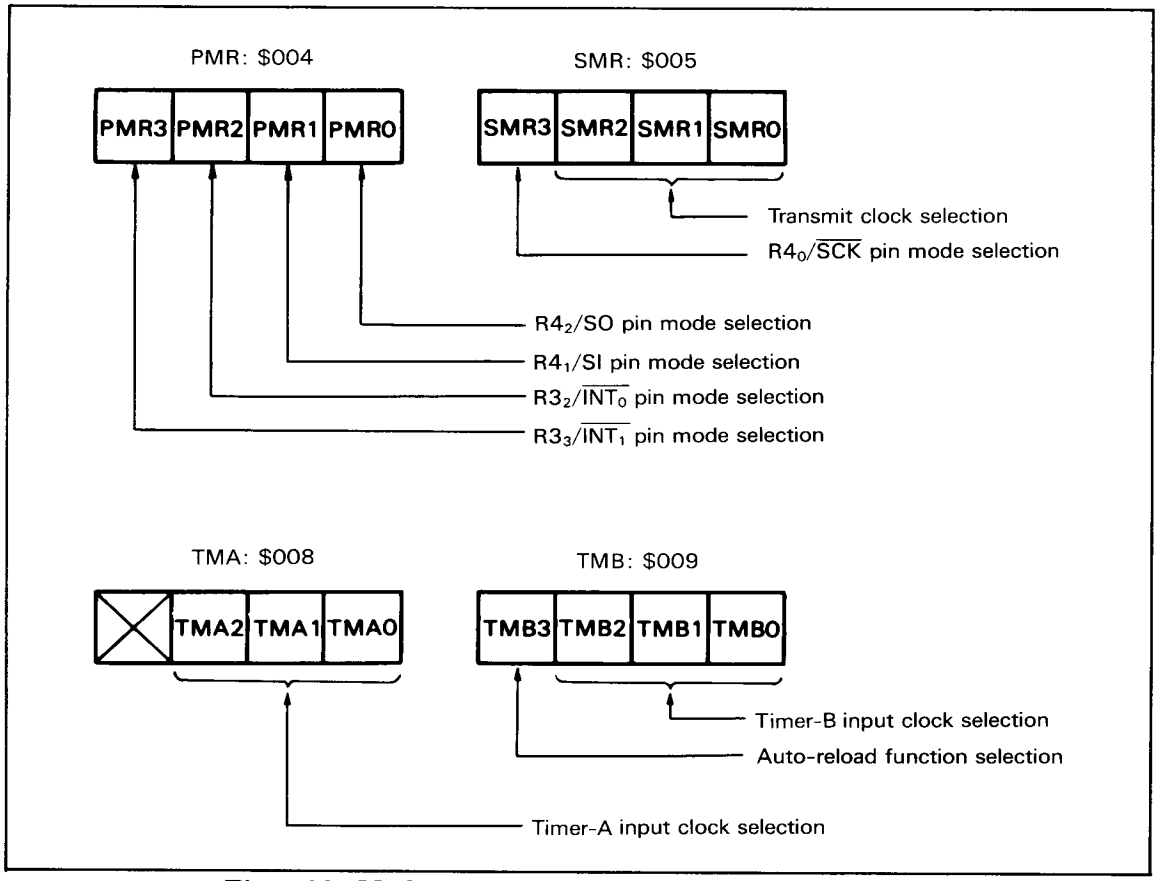
Timer Mode Register A (TMA: $008): Timer mode register A is a 3-bit write-only register. The TMA controls the prescaler divide ratio of timer A clock input, as shown in table 11. The timer mode register A is initialized to $0 by MCU reset.

Timer Mode Register B (TMB: $009): Timer mode register B (TMB) is a 4-bit write-only register which selects the auto-reload function, the prescaler divide ratio, or the source of the clock input signal, as shown in table 12. Timer mode register B is initialized to $0 by MCU reset.

The operation mode of timer B changes at the second instruction cycle after timer mode register B is written to. The initialization of timer B by writing data into timer load register B should be performed after the contents of TMB are changed. The configuration and function of timer mode register B is shown in figure 14.

Timer B (TCBL: $00A, TCBU: $00B, TLRL: $00A, TLRU: $00B): Timer B consists of an 8-bit write-only timer load register, and an 8-bit read-only timer counter. Each of them has a low-order digit (TCBL: $00A, TLRL: $00A) and a high-order digit (TCBU: $00B, TLRU: $00B). (Refer to figure 2.)

Timer counter B can be initialized by writing data into timer load register B. In this case, write the low-order digit first, and then the high-order digit. The timer counter is initialized when the high-order digit is written. The timer load register is initialized to $00 by MCU reset.



Fiure 14 Mode Register Configuration and Function

The counter value of timer B can be obtained by reading timer counter B. In this case, read the high-order digit first, and then the low-order digit. The count value of the low-order digit is latched at the time when the high-order digit is read.

Timer A Interrupt Request Flag (IFTA: $001, Bit 2): The timer A interrupt request flag is set by the overflow output of timer A (table 13).

Timer A Interrupt Mask (IMTA: $001, Bit 3): The timer A interrupt mask prevents an interrupt request from being generated by the timer A interrupt request flag (table 14).

Timer B Interrupt Request Flag (IFTB: $002, Bit 0): The timer B interrupt request flag is set by the overflow output of timer B (table 15).

Timer B Interrupt Mask (IMTB: $002, Bit 1): The timer B interrupt mask prevents an interrupt request from being generated by timer B interrupt request flag (table 16).

Table 11 Timer Mode Register A

TMA2	TMA1	TMA0	Prescaler Divide Ratio
0	0	0	÷ 2048
0	0	1	÷ 1024
0	1	0	÷ 512
0	1	1	÷ 128
1	0	0	÷ 32
1	0	1	÷ 8
1	1	0	÷ 4
1	1	1	÷ 2

Table 12 Timer Mode Register B

TMB3	Auto-Reload Function
0	No
1	Yes

TMB2	TMB1	TMB0	Prescaler Divide Ratio, Clock Input Source
0	0	0	÷ 2048
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	$\overline{INT_1}$ (External event input)

Table 13 Timer A Interrupt Request Flag

IFTA	Interrupt Request
0	No
1	Yes

Table 14 Timer A Interrupt Mask

IMTA	Interrupt Request
0	Enable
1	Disable (Mask)

Table 15 Timer B Interrupt Request Flag

IFTB	Interrupt Request
0	No
1	Yes

Table 16 Timer B Interrupt Mask

IMTB	Interrupt Request
0	Enable
1	Disable (Mask)

Input/Output

The MCU has 58 I/O pins including 12 high-current standard pins (15 mA). If an I/O pin is used as an input pin, output data should be in the state shown in table 18.

D Port: The D port is an I/O port which has 16 discrete I/O pins, each of which can be addressed independently. It can be set/reset through the SED/RED and SEDD/REDD instructions, and can be tested through the TD and TDD instructions. See table 17 for the I/O pin circuit types.

R Ports: The eleven R ports in the HD4074008 are composed of 20 I/O pins, 16 output-only pins, and 6 input-only pins. Data is input through the LAR and LBR instructions and output through the LRA and LRB instructions. The MCU will not be affected by writing into the input-only and/or non-existing ports, while invalid data will be read by reading from the output-only and/or non-existing ports.

The $R3_2$, $R3_3$, $R4_0$, $R4_1$, and $R4_2$ pins are multiplexed with the $\overline{INT_0}$, $\overline{INT_1}$, $\overline{SCK}$, SI, and SO pins, respectively. See table 17 for the I/O pin circuit types.

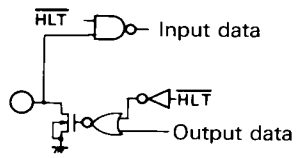
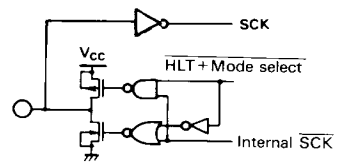
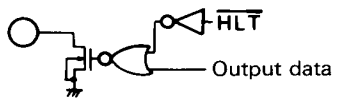
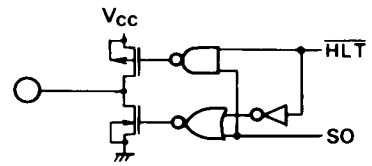
Unused I/O Pins: If the I/O pins not used in the user system are left floating, the LSI may malfunction because of noise. The electric potential of the I/O pins should be fixed as follows to prevent malfunction.

For PMOS open drain output pins, connect the pin to V_{CC} on the printed circuit of the user system.

For NMOS open drain output pins, connect the pin to GND on the printed circuit of the user system. Input pins should be connected to V_{CC} on the printed circuit of the user system.

Table 17 I/O Pin Circuit Types

Standard Pins

	MOS Without Pull-Up (NMOS Open Drain)	Pins	MOS Without Pull-Up (NMOS Open Drain)	Pins
I/O pins		D_0-D_3 $R3_0$-$R3_3$ $R4_0$-$R4_3$ $R5_0$-$R5_3$		$\overline{SCK}$ (Note 2) Output mode
Output pins		$R6_0$-$R6_3$ $R7_0$-$R7_3$ $R8_0$-$R8_3$		SO
Input pins		$R9_0$-$R9_3$		$\overline{INT_0}$ $\overline{INT_1}$ SI $\overline{SCK}$ (Note 2) Input mode

Notes: 1. In stop mode, $\overline{HLT}$ is 0 and HLT is 1. I/O pins are in high impedance.
2. During serial interface interrupt, the $\overline{SCK}$ pin is an input pin if external clock input mode is selected.

$R4_0/\overline{SCK}$ and $R4_2$/SO should be set to $R4_0$ and $R4_2$ by the serial mode register and port mode register, respectively.

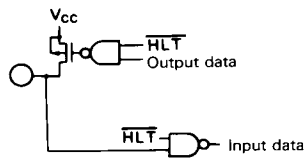
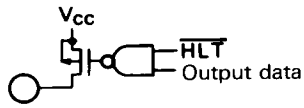
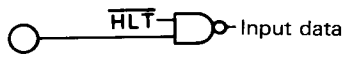
Reset

Setting the RESET pin high resets the MCU. At power-on or when cancelling the stop mode, the reset must satisfy t_{RC} for the oscillator to stabilize. In all other cases, at least two instruction cycles are required for the MCU to be reset.

Table 19 shows the components to be initialized by MCU reset, and the status of each.

Table 17 I/O Pin Circuit Types (cont)

Standard Pins

	MOS Without Pull-Down (PMOS Open Drain)	Pins
I/O pins		D_4-D_{15} $R1_0$-$R1_3$ $R2_0$-$R2_3$
Output pins		RO_0-RO_3
Input pins		RA_0, RA_1

Note: In stop mode, $\overline{HLT}$ is 0. I/O pins are in high impedance.

Table 18 Data Input from Common Input/Output Pins

I/O Pin Circuit Type		Input Possible	Input Pin State
Standard pins	CMOS	No	—
	MOS without pull-up (NMOS open drain)	Yes	1
	MOS without pull-down (PMOS open drain)	Yes	0

Table 19 Initial Values at MCU Reset

Item			Initial Value	Contents
Program counter (PC)			$0000	Execute the program from the top of ROM address
Status flag (ST)			1	Enable branching with conditional branch instructions
Stack pointer (SP)			$3FF	Stack level is 0
I/O pins, output registers	Standard pins	NMOS open drain [MOS without pull-up]	1	Enable input
		PMOS open drain [MOS without pull-down]	0	Enable input
Interrupt flags/ mask	Interrupt enable flag (IE)		0	Inhibit all interrupts
	Interrupt request flag (IF)		0	No interrupt request
	Interrupt mask (IM)		1	Mask interrupt request
Mode registers	Port mode register (PMR)		0000	See Port Mode Register section
	Serial mode register (SMR)		0000	See Serial Mode Register section
	Timer mode register A (TMA)		000	See Timer Mode Register A section
	Timer mode register B (TMB)		0000	See Timer Mode Register B section
Timer/counters, serial interface	Prescaler		$000	—
	Timer counter A (TCA)		$00	—
	Timer counter B (TCB)		$00	—
	Timer load register (TLR)		$00	—
	Octal counter		000	—

Note: MCU reset affects the other registers as shown in the following table.

Item	Abbr	After MCU Reset to Recover from Stop Mode	After MCU Reset to Recover from Other Modes
Carry flag	(CA)	The contents of the items before MCU reset are not retained. It is necessary to initialize them by software again.	The contents of the items before MCU reset are not retained. It is necessary to initialize them by software again.
Accumulator	(A)		
B register	(B)		
W register	(W)		
X/SPX registers	(X/SPX)		
Y/SPY registers	(Y/SPY)		
Serial data register	(SR)		
RAM		The contents of RAM before MCU reset (just before STOP instruction) are retained.	Same as above for RAM.

Internal Oscillator Circuit

Figure 15 outlines the internal oscillator circuit. A crystal oscillator or ceramic filter oscillator can be selected as the oscillator type. Refer to table 20 to select the oscillator type. In addition, see figure 16 for the layout of the crystal or ceramic filter.

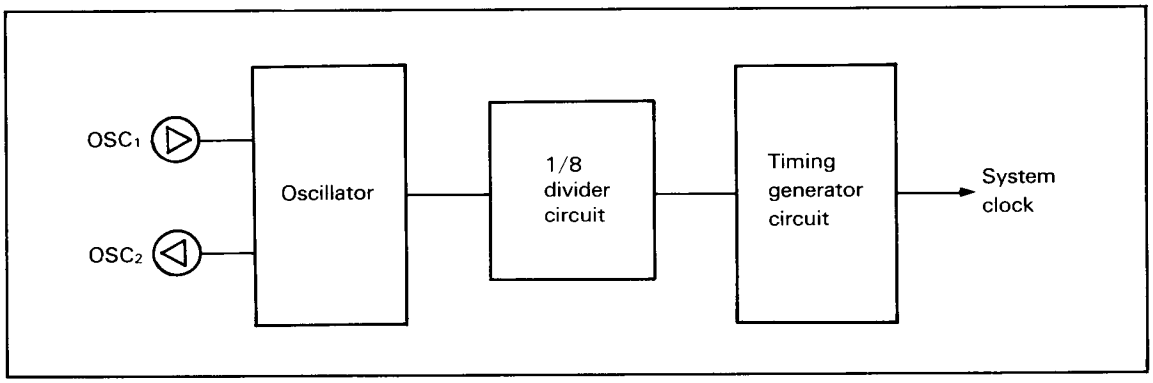


Figure 15 Internal Oscillator Circuit

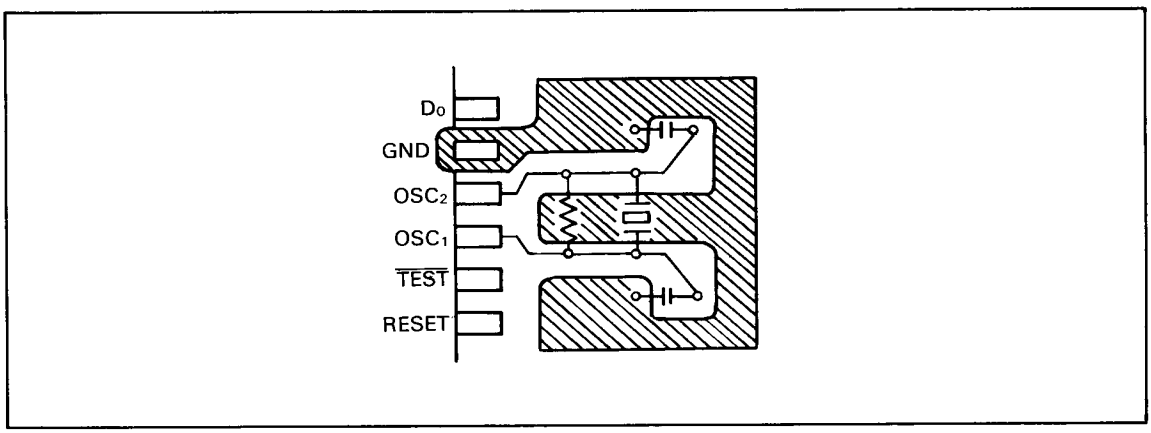
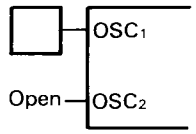
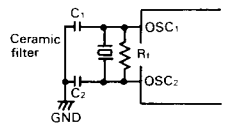
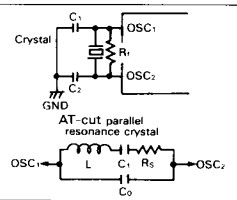


Figure 16 Layout of Crystal and Ceramic Filter

Table 20 Examples of Oscillator Circuits

	Circuit Configuration	Circuit Constants
External clock operation	Oscillator 	
Ceramic filter oscillator		Ceramic filter CSA8.00MT (Murata) R_f: 1 MΩ ± 20% C_1: 30 pF ± 20% C_2: 30 pF ± 20%
Crystal oscillator		R_f: 1 MΩ ± 20% C_1: 10-22 pF ± 20% C_2: 10-22 pF ± 20% Crystal: Equivalent circuit shown at bottom left C_0: 7 pF max. R_S: 100 Ω max. f: 1.0-9 MHz

Notes: 1. Since the circuit constant changes according to the crystal and ceramic filter resonator and to the stray capacitance of the board, consult with the crystal or ceramic filter maker to determine the circuit parameters.
2. Wiring among OSC_1, OSC_2, and elements should be as short as possible, and avoid crossing other wires. Refer to figure 16.

Operating Modes

The MCU has two low-power dissipation modes, standby mode and stop mode (table 21). Figure 17 is a mode transition diagram of these modes.

Standby Mode: Executing the SBY instruction puts the MCU into standby mode. In standby mode, the oscillator circuit is active and interrupts, timer/counters, and the serial interface remain working. On the other hand, the CPU stops since the clock related to the instruction execution stops. Registers, RAM, and I/O pins retain the states they were in just before the MCU went into standby mode.

Table 21 Low-Power Dissipation Modes Function

Low-Power Dissipation Mode	Instruction	Condition: Oscillator Circuit	Instruction Execution	Registers, Flags	Interrupt Function	RAM	Input/ Output Pins	Timer/ Counters, Serial Interface	Cancellation Method
Standby mode	SBY instruction	Active	Stop	Retained	Active	Retained	Retained[2]	Active	RESET input, interrupt request
Stop mode	STOP instruction	Stop	Stop	Reset[1]	Stop	Retained	High impedance	Stop	RESET input

Notes: 1. The MCU recovers from the stop mode by RESET input. Refer to table 19 for the contents of the flags and registers.
2. When an I/O circuit is active, an I/O current may flow, depending on the state of the I/O pin in the standby mode. This is the additional current for current dissipation in the standby mode.

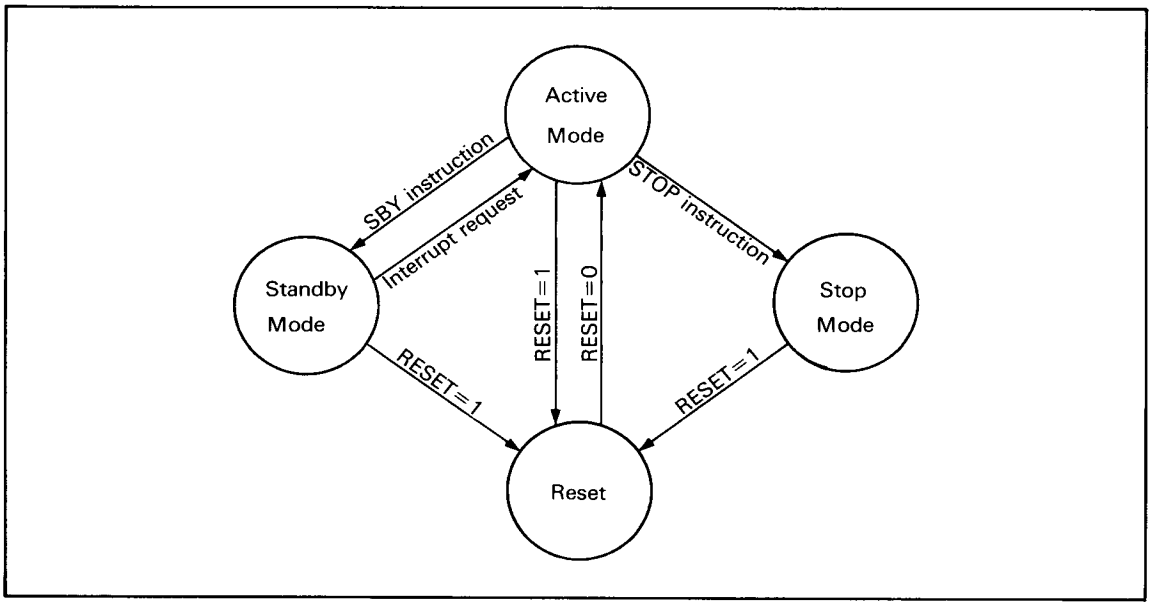


Figure 17 MCU Operation Mode Transition

The standby mode may be cancelled by inputting RESET or by asserting an interrupt request. In the former case the MCU is reset. If the interrupt enable flag is 1 when an interrupt request was asserted, the interrupt is executed; if it is 0, the interrupt request is put on hold and normal instruction execution continues. In the latter case, the MCU becomes active and executes the next instruction following the SBY instruction.

Figure 18 shows the flowchart of the standby mode.

Stop Mode: Executing the STOP instruction brings the MCU into the stop mode, in which the oscillator circuit and every function of the MCU stop.

The stop mode may be cancelled by resetting the MCU. At this time, as shown in figure 19, the reset input must be applied for at least t_{RC} for oscillation to stabilize. (Refer to the AC characteristics table.) After the stop mode is cancelled, RAM retains the state it was in just before the MCU went into stop mode, but the accumulator, B register, W register, X/SPX

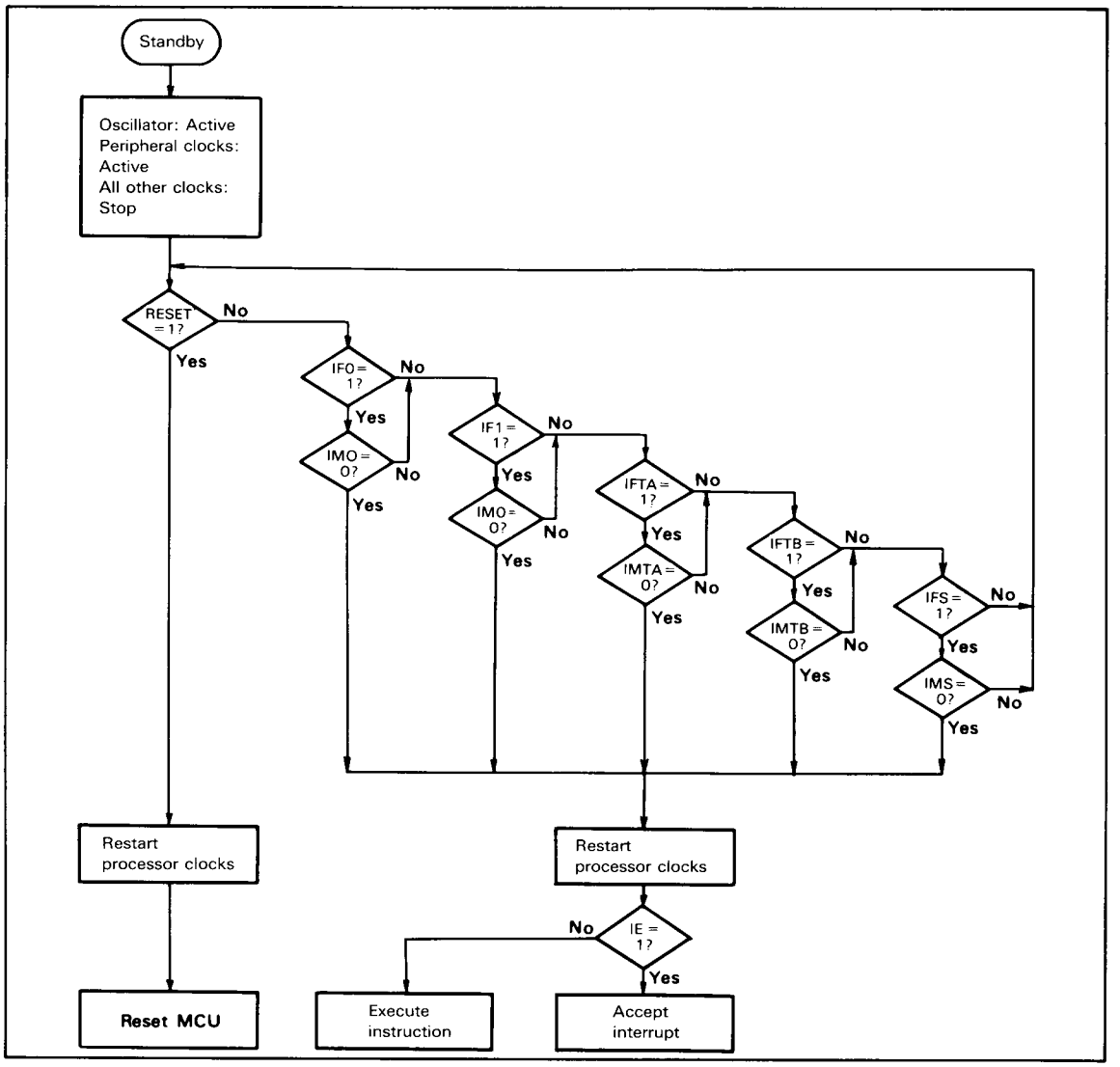


Figure 18 MCU Operating Flowchart in Standby Mode

registers, Y/SPY registers, carry flag, and serial data register will not retain their contents.

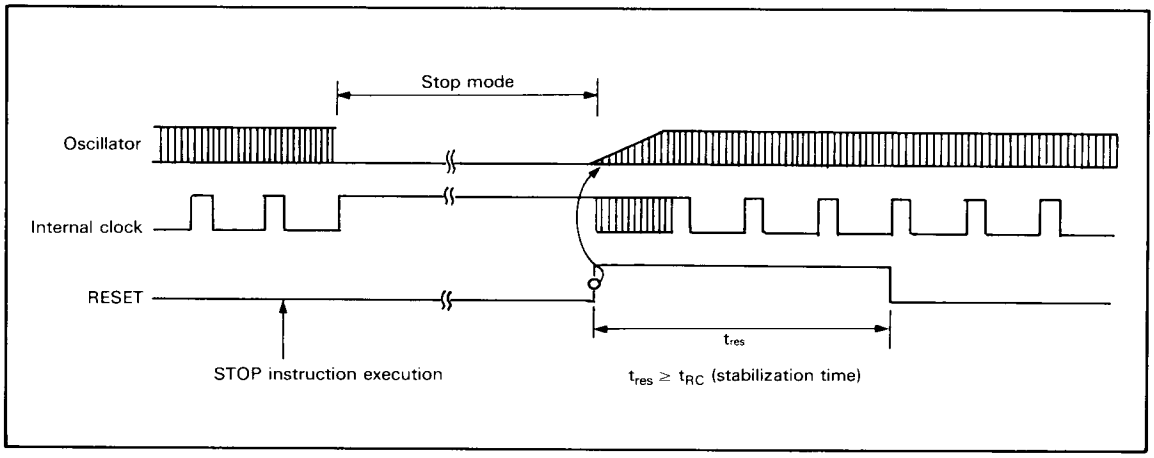


Figure 19 Timing of Stop Mode Cancellation

Programmable ROM Operation

The HD4074008's on-chip PROM is programmed in PROM mode (figures 20, 21 and 22). The PROM mode is set by pulling $\overline{TEST}$, $\overline{M_0}$, and $\overline{M_1}$ low, and RESET high as shown in figure 21. In the PROM mode, the MCU does not operate. It can be programmed like a standard 27256 EPROM using a standard PROM programmer and a 64-to-28-pin socket adapter. Table 23 lists recommended PROM programmers and socket adapters.

Since an instruction of the HMCS400 series consists of 10 bits, the HMCS400-series microcomputer incorporates a conversion circuit to be used as a general purpose PROM programmer. By this circuit, an instruction is read or programmed using 2 addresses, the lower 5 bits and upper 5 bits as shown in figure 22. For example, if 8 Kwords of on-chip PROM are programmed by a general purpose PROM programmer, 16 Kbytes of addresses ($0000-$3FFF) should be specified.

Programming and Verification

The HD4074008 can be programmed at high speed without causing voltage stress or affecting data reliability.

Table 22 shows how programming and verification modes are selected.

Figure 23 is a programming flowchart, and figure 33 is a timing chart.

Erasing

The PROMs with ceramic window packages can be erased by ultraviolet light. All erased bits become 1s.

Erasing conditions require an ultraviolet light of wavelength 2537 Å with a minimum irradiation of 15 W·sec/cm^2. These conditions are satisfied by exposing the LSI to a 12,000-μW/cm^2 UV source for 15 to 20 minutes at a distance of 1 inch.

Precautions

1. Addresses $0000 to $3FFF should be specfied if the PROM is programmed by a PROM programmer. If addresses of $4000 or higher are accessed, the PROM may not be programmed or verified. Note that the plastic package type cannot be erased and reprogrammed. Data in unused addresses should be set to $FF. (Only ceramic window packages can be erased and reprogrammed.)

2. Make sure that the PROM programmer, socket adapter, and LSI match properly. Using the wrong programmer for the 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 to the programmer.

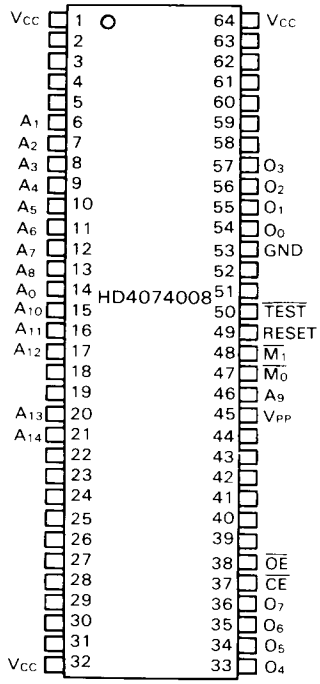
3. The PROM should be programmed with V_{PP}=12.5 V. Other PROMs use 21 V. If 21 V is applied to the HD4074008, the LSI may be permanently damaged. 12.5 V is the voltage for V_{PP} of Intel's 27256.

Table 22 PROM Mode Selection

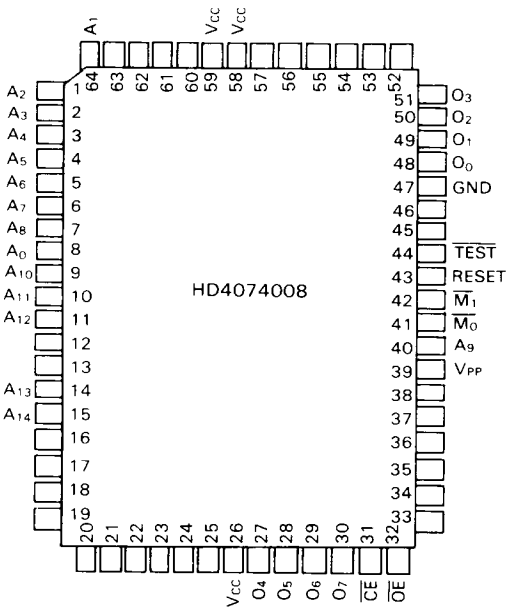
	Pin			
Mode	$\overline{CE}$	$\overline{OE}$	V_{PP}	
Programming	Low	High	V_{PP}	Data input
Verify	High	Low	V_{PP}	Data output
Programming inhibited	High	High	V_{PP}	High impedance

Table 23 PROM Programmers and Socket Adapters

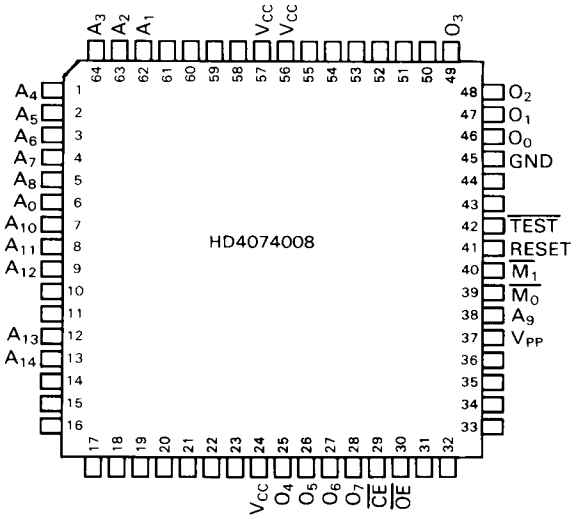
PROM Programmer		Socket Adapter		
Maker	**Type Name**	**Maker**	**Type Name**	**Package**
DATA I/O	22B 29B	Hitachi	HS408ESS11H	DP-64S DC-64S
			HS408ESF01H	FP-64
			HS408ESF03H	FP-64A
AVAL Corp.	PKW-1000 PKW-7000	Hitachi	HS408ESS21H	DP-64S DC-64S
			HS408ESF01H	FP-64
			HS408ESF03H	FP-64A



(DP-64S, DC-64S)



(FP-64)



(FP-64A)

(Top view)

GND: V_{SS} level
No pin name: Open

Figure 20 PROM Mode Pin Arrangement

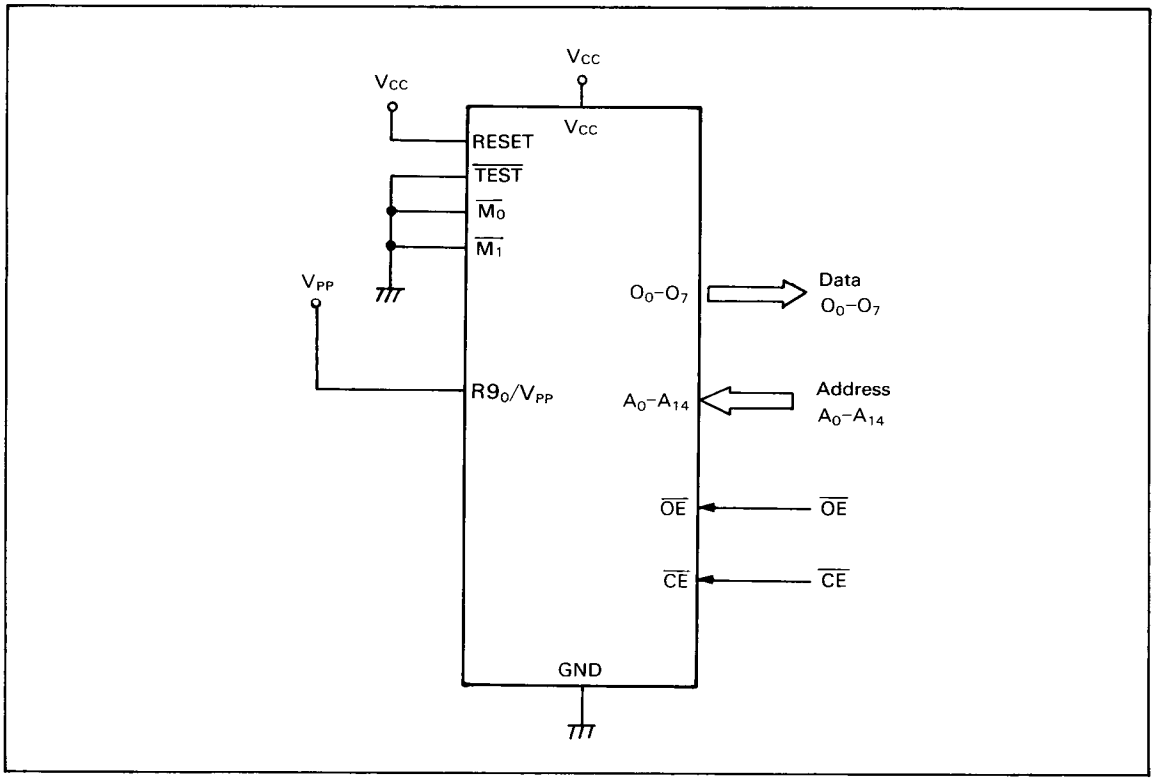


Figure 21 PROM Mode Connection Diagram

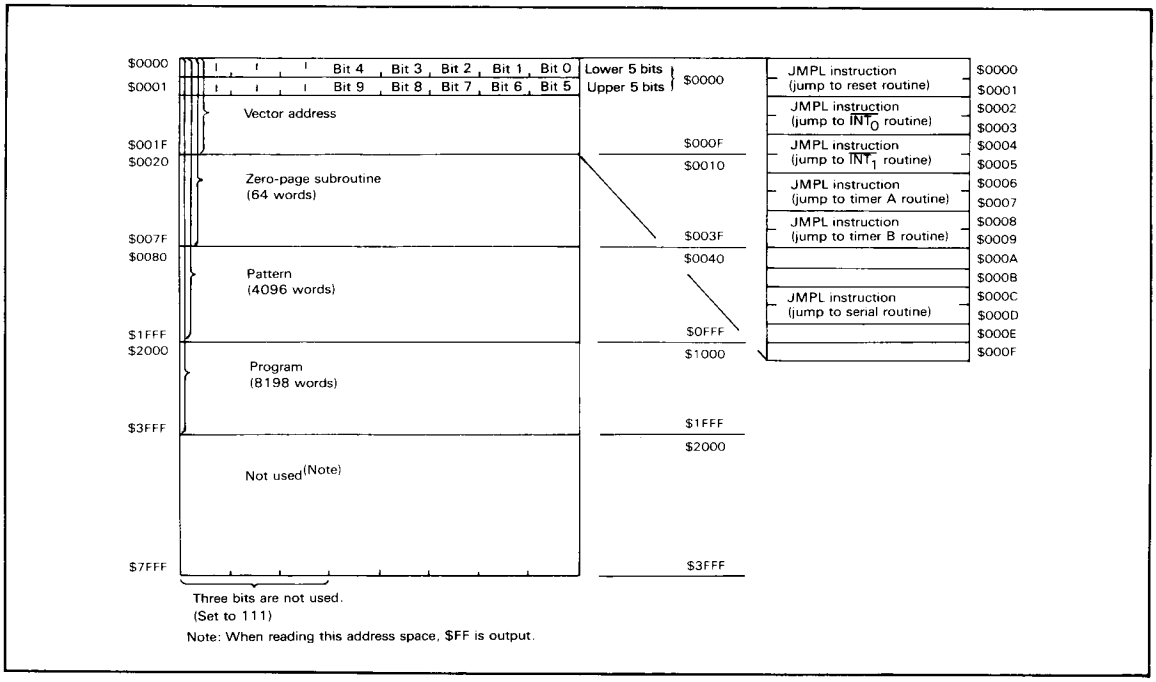


Figure 22 PROM Mode Memory Map

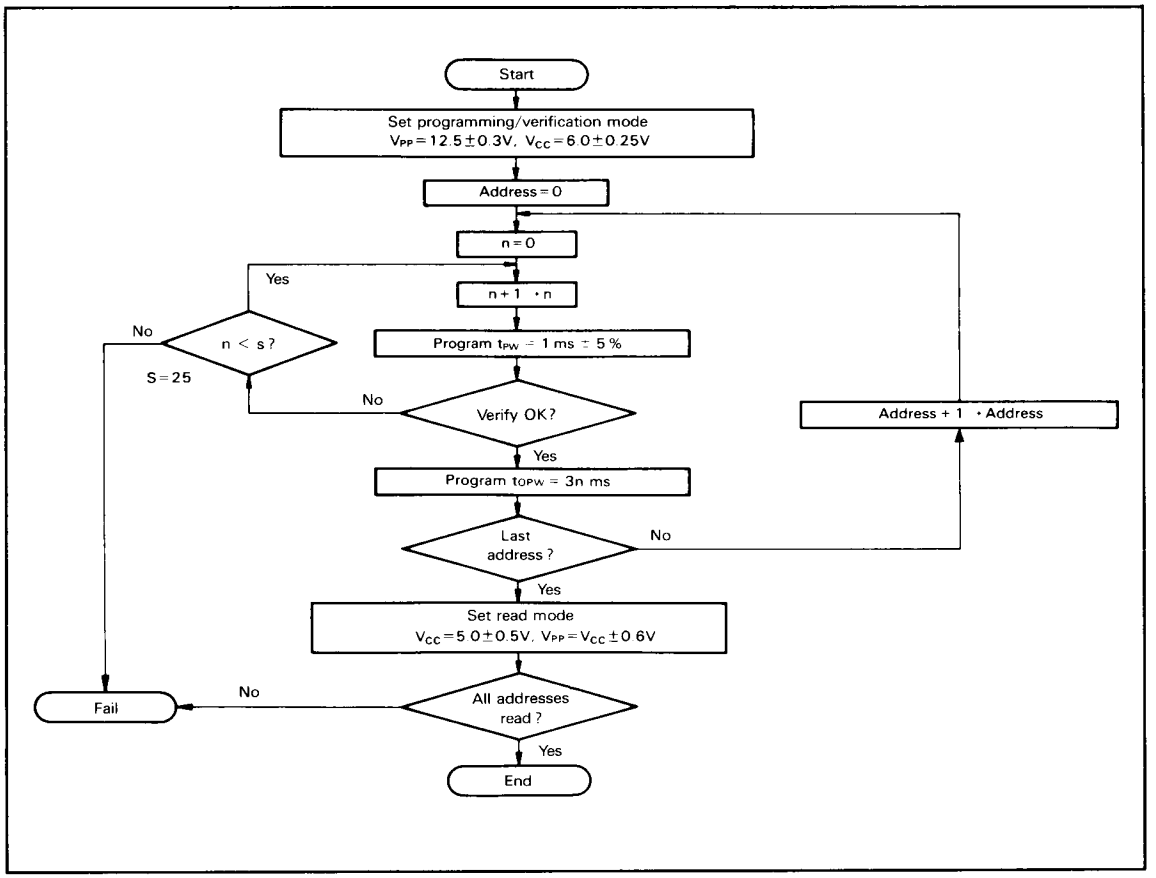


Figure 23 High-Speed Programming Flowchart

ZTAT™ MCU On-Chip PROM Characteristics and Precautions

Principles of Programming/Erasing: The ZTAT™ memory cells are the same as an EPROM's. Therefore they are programmed by applying a high voltage to its control gate and drain, which injects hot electrons into the floating gate. These electrons become stable, surrounded by an energy barrier of SiO_2 film. Such a cell becomes a 0 bit due to the memory threshold voltage change. A cell with no condensed electrons at its floating gate appears as a 1 bit (figure 24).

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

- Ultraviolet light: Discharged by photo-emitted electrons according to the erasure principle
- Heat: Discharged by thermal emitted electrons
- 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.

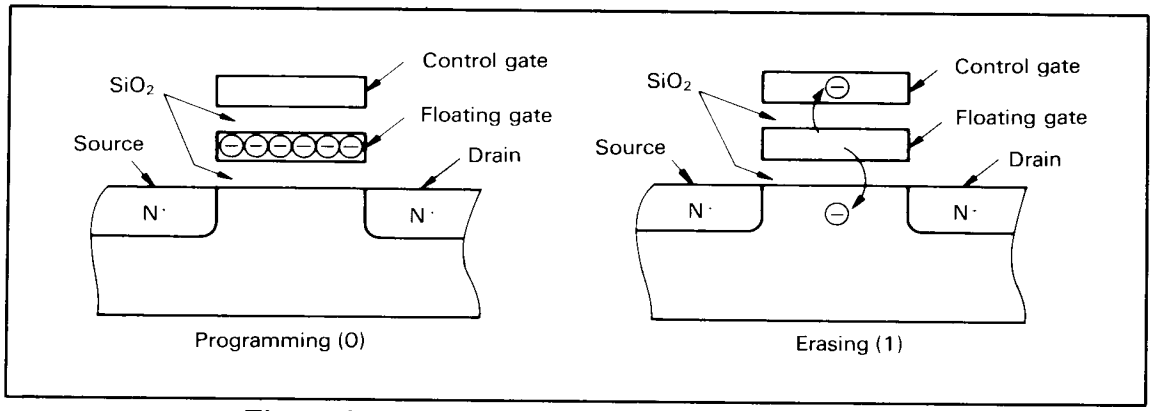


Figure 24 Cross-Sections of a EPROM Memory Cell

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 pn junction may be permanently damaged. Pay particular attention to PROM programmer overshoot. Negative voltage noise will cause a parasitic transistor effect, which may reduce breakdown voltage.

The ZTAT™ microcomputer is connected electrically to the PROM programmer through a socket adapter. Therefore, pay attention to the following:

- Confirm that the socket adapter is firmly fixed onto the PROM programmer.
- Do not touch the socket adapter or the LSI during programming.
- Misprogramming can be caused by poor contacts.

PROM Reliability after Programming: Generally, semiconductors are reliable except for initial failures that can be avoided by screening tests. Exposure to high temperature is a kind of screening which quickly removes PROM memory cells with data hold failures This is done to the ZTAT™s 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 25 shows the recommended screening procedure.

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

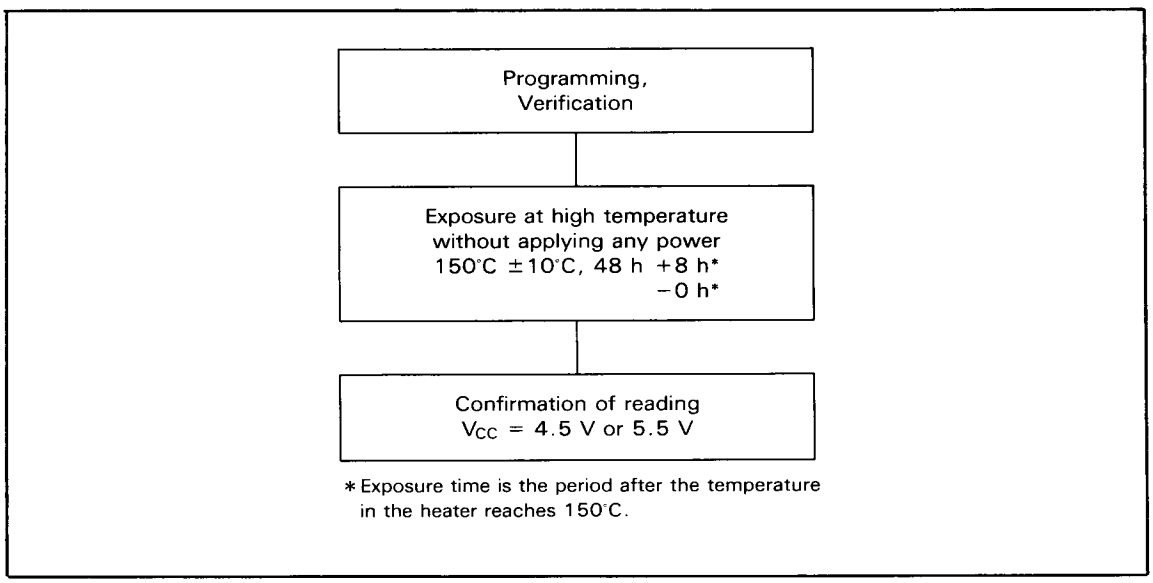


Figure 25 Recommended Screening Procedure

Addressing Modes

RAM Addressing Modes

As shown in figure 26, the MCU has three RAM addressing modes: register indirect addressing, direct addressing, and memory register addressing.

Register Indirect Addressing Mode: The W register, X register, and Y register contents (10 bits) are used as the RAM address.

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

Memory Register Addressing Mode: The memory registers (16 digits from $020 to $02F) are accessed by executing the LAMR and XMRA instructions.

ROM Addressing Modes and the P Instruction

The MCU has four ROM addressing modes as shown in figure 27.

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

Current Page Addressing Mode: The MCU has 8 pages of ROM with 256 words per page. By executing the BR instruction, the program can branch to an address on the current page. This instruction replaces the low-order eight bits of the program counter (PC_7 to PC_0) with 8-bit immediate data.

When the BR instrution is on a page boundary (256n + 255) (figure 28), executing it transfers the PC contents to the next page according to the hardware architecture. Consequently, the program branches to the next page when the BR instruction is used on a page boundary. The HMCS400 series cross macroassembler has an automatic paging facility for ROM pages.

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

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

P Instruction: ROM data addressed by table data addressing can be referenced by the P instruction (figure 29). When bit 8 in the referred ROM data is 1, 8 bits of ROM data are written into the accumulator and B register. When bit 9 is 1, 8 bits of ROM data are written into the R1 and R2 port output registers. When both bits 8 and 9 are 1, ROM data are written into the accumulator and B register, and also to the R1 and R2 port output registers at the same time.

The P instruction has no effect on the program counter.

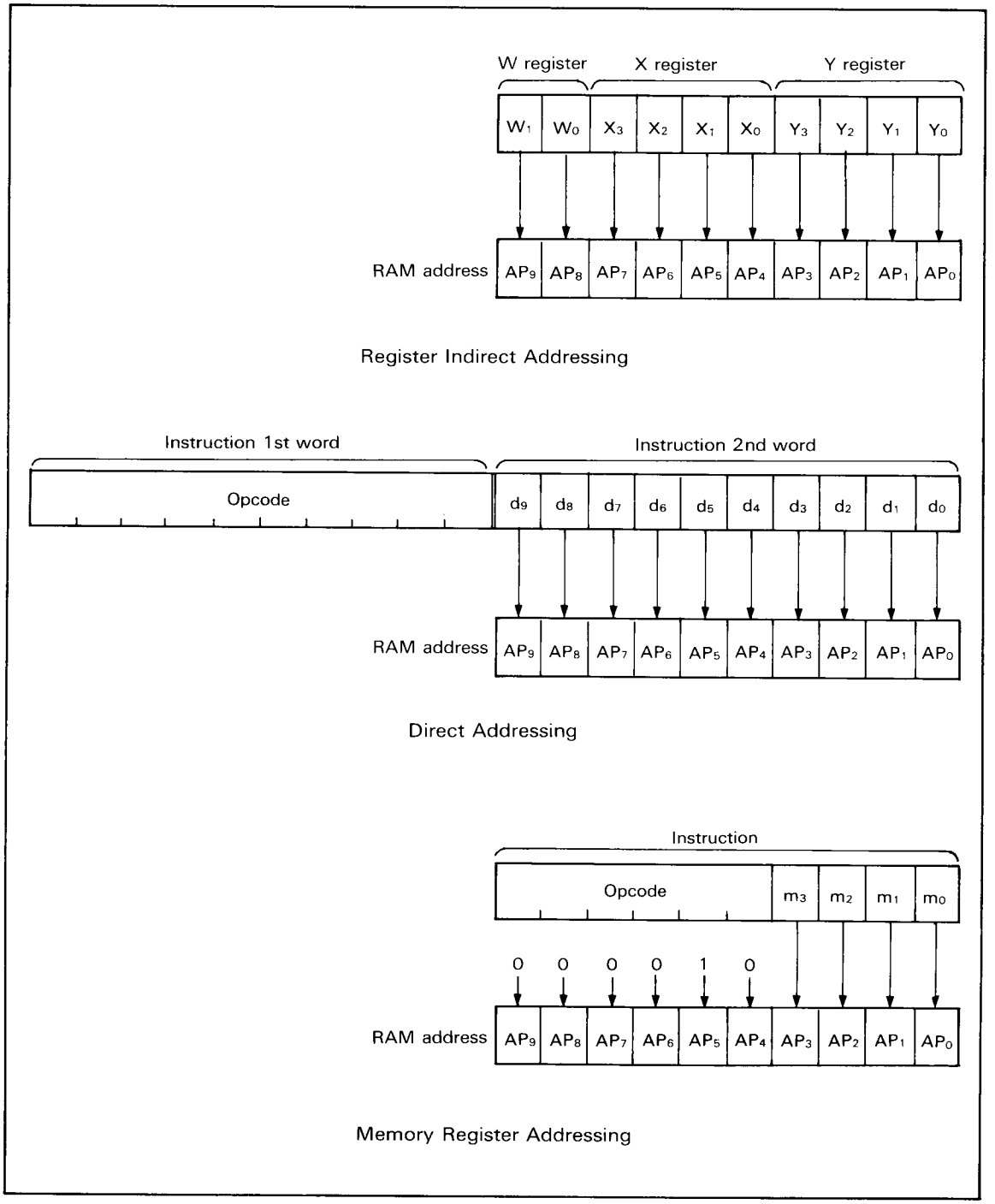


Figure 26 RAM Addressing Modes

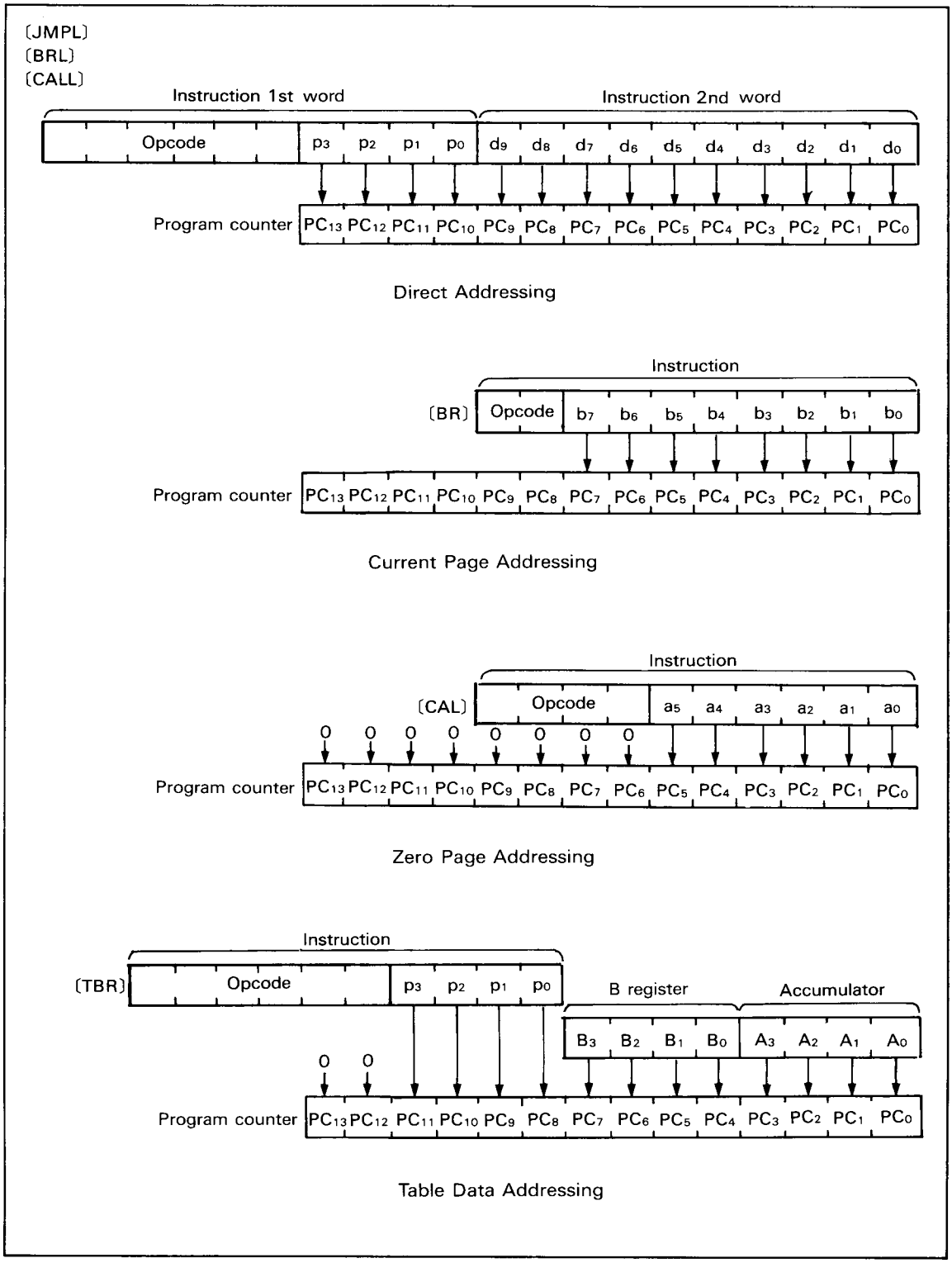


Figure 27 ROM Addressing Modes

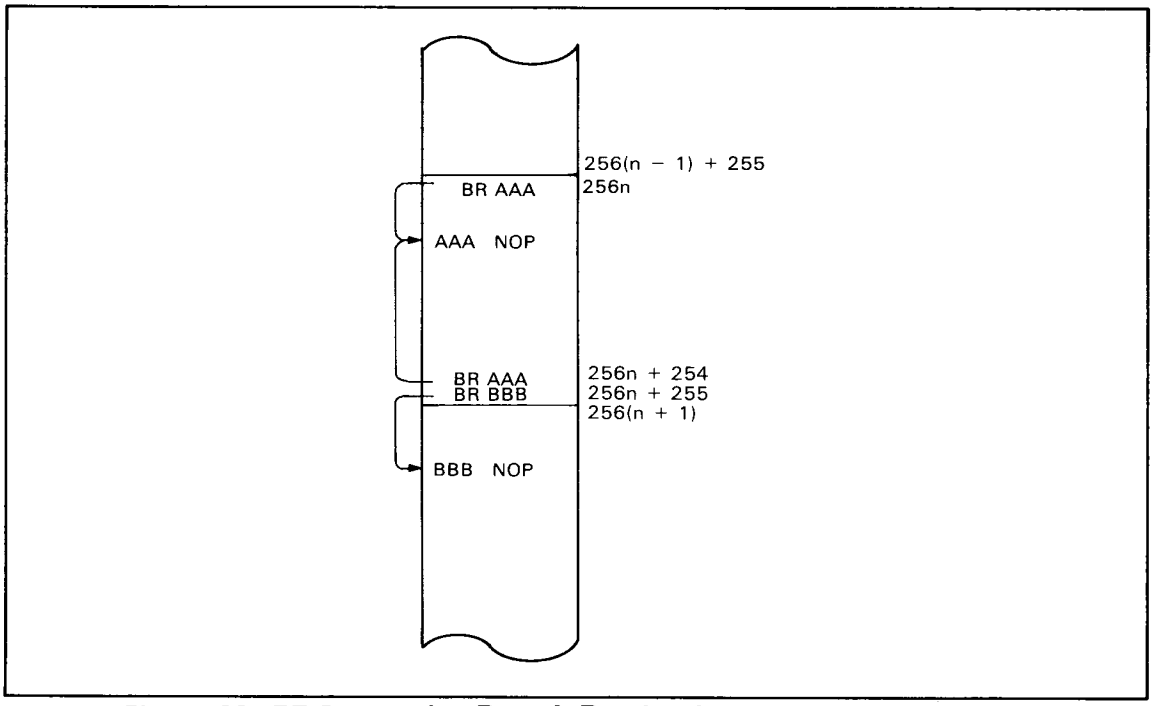


Figure 28 BR Instruction Branch Destination on a Page Boundary

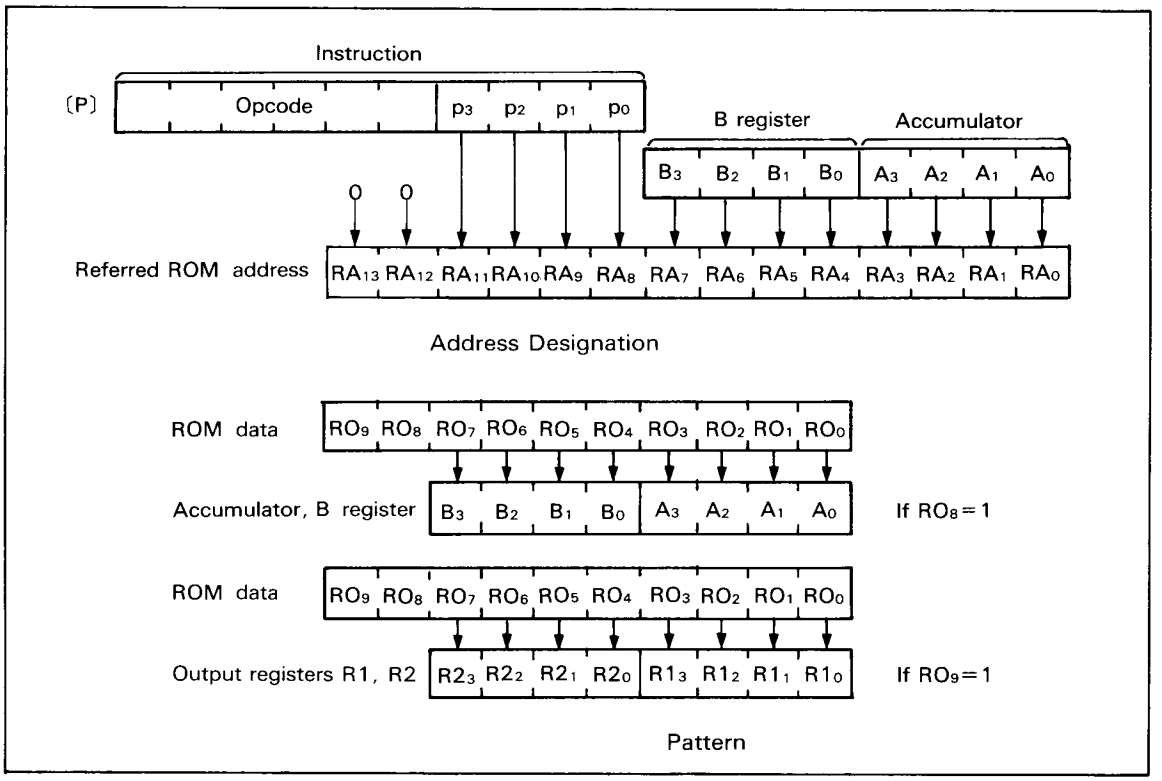


Figure 29 P Instruction

Absolute Maximum Ratings

Item	Symbol	Value	Unit	Note
Supply voltage	V_{CC}	−0.3 to +7.0	V	
Programming voltage	V_{PP}	−0.3 to +14	V	3
Pin voltage	V_T	−0.3 to V_{CC} + 0.3	V	4
Total permissible input current	ΣI_o	50	mA	5
Maximum input current	I_o	15	mA	7, 8
Maximum output current	$-I_o$	4	mA	9, 10
		6	mA	9, 11
		30	mA	9, 12
Total permissible output current	$-\Sigma I_o$	150	mA	6
Operating temperature	T_{opr}	−20 to + 75	°C	
Storage temperature	T_{stg}	−55 to + 125	°C	

Notes: 1. Permanent damage may occur if these absolute maximum ratings are exceeded. Normal operation should be under the conditions of the electrical characteristics tables. If these conditions are exceeded, the LSI may malfunction or the reliability may be affected.
2. All voltages are with respect to GND.
3. Applied to $R9_0$ (V_{PP}).
4. Standard pins.
5. Total permissible input current is the total sum of input currents which flow in from all I/O pins to GND simultaneously.
6. Total permissible output current is the total sum of the output currents which flow out from V_{CC} to all I/O pins simultaneously.
7. Maximum input current is the maximum amount of input current from each I/O pin to GND.
8. D_0—D_3 and R3—R8.
9. Maximum output current is the maximum amount of output current from V_{CC} to each I/O pin.
10. D_0—D_3 and R3—R8.
11. R0—R2.
12. D_4—D_{15}.

Electrical Characteristics

DC Characteristics (V_{CC} = 5 V ±10%, GND = 0 V, T_a = −20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition	Note
Input high voltage	V_{IH}	RESET, SCK, $\overline{INT_0}$, $\overline{INT_1}$	0.8V_{CC}		V_{CC} + 0.3	V		
		OSC_1	V_{CC} − 0.5		V_{CC} + 0.3	V		
		SI	0.7V_{CC}		V_{CC} + 0.3	V		
Input low voltage	V_{IL}	RESET, SCK, $\overline{INT_0}$, $\overline{INT_1}$	−0.3		0.2V_{CC}	V		
		OSC_1	−0.3		0.5	V		
		SI	−0.3		0.3V_{CC}	V		
Output high voltage	V_{OH}	SCK, SO	V_{CC}−1.0			V	$-I_{OH}$ = 1.0 mA	
			V_{CC}−0.5			V	$-I_{OH}$ = 0.5 mA	
Output low voltage	V_{OL}	SCK, SO		0.4		V	I_{OL} = 1.6 mA	
Input/output leakage current	$\|I_{IL}\|$	RESET, SCK, $\overline{INT_0}$, $\overline{INT_1}$, OSC_1, SI, SO			1	μA	V_{in} = 0 V to V_{CC}	1
Current dissipation in active mode	I_{CC}	V_{CC}			4.5	mA	V_{CC} = 5 V	2,5
Current dissipation in standby mode	I_{SBY}	V_{CC}			1.7	mA	Maximum logic operation V_{CC} = 5 V	3,5
Current dissipation in stop mode	I_{STOP}	V_{CC}			10	μA	V_{in}:$\overline{TEST}$ = V_{CC} − 0.3 V to V_{CC}, V_{in}:RESET = 0 V to 0.3 V	4
Stop mode retaining voltage	V_{STOP}	V_{CC}	2			V		

Notes: 1. Excluding pull-up MOS current and output buffer current.
2. The MCU is in the reset state. Input/output current does not flow.
   - MCU in reset state, operation mode
   - RESET, $\overline{TEST}$: V_{CC}
   - D_0-D_3, R3-R9: V_{CC}
   - D_4-D_{15}, R0-R2, RA_0, RA_1: GND
3. The timer/counter operates with the fastest clock. Input/output current does not flow.
   - MCU in standby mode
   - Input/output in reset state
   - Serial interface: Stop
   - RESET: GND
   - $\overline{TEST}$: V_{CC}
   - D_0-D_3, R3-R9: V_{CC}
   - D_4-D_{15}, R0-R2, RA_0, RA_1: GND
4. Excluding pull-down MOS current.
5. When f_{OSC} = x MHz, estimate the current dissipation as follows:
   Maximum value at x MHz = (x/8) × (max. value at 8 MHz)

Input/Output Characteristics for Standard Pins: NMOS Open Drain (V_{CC} = 5 V ±10%, GND = 0 V, T_a = −20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition
Input high voltage	V_{IH}	D_0-D_3, R3-R5, R9	0.7V_{CC}		V_{CC} + 0.3	V	
Input low voltage	V_{IL}	D_0-D_3, R3-R5, R9	−0.3		0.3V_{CC}	V	
Output low voltage	V_{OL}	D_0-D_3, R3-R8			0.4	V	I_{OL} = 1.6 mA
Input/output leakage current(Note)	$\|I_{IL}\|$	D_0-D_3, R3-R9			1	μA	V_{in} = 0 V to V_{CC}

Note: Pull-up MOS current and output buffer current are excluded.

Input/Output Characteristics for Standard Pins: PMOS Open Drain (V_{CC} = 5 V ±10%, GND = 0 V, T_a = −20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition
Input high voltage	V_{IH}	D_4-D_{15}, R1, R2, RA_0, RA_1	0.7V_{CC}		V_{CC} + 0.3	V	
Input low voltage	V_{IL}	D_4-D_{15}, R1, R2, RA_0, RA_1	−0.3		0.3V_{CC}	V	
Output high voltage	V_{OH}	D_4-D_{15}	V_{CC} − 3.0			V	− I_{OH} = 15 mA, V_{CC} = 5 V
			V_{CC} − 2.0			V	− I_{OH} = 10 mA, V_{CC} = 5 V
			V_{CC} − 1.0			V	− I_{OH} = 4 mA, V_{CC} = 5 V
		R0-R2	V_{CC} − 3.0			V	− I_{OH} = 3 mA, V_{CC} = 5 V
			V_{CC} − 2.0			V	− I_{OH} = 2 mA, V_{CC} = 5 V
			V_{CC} − 1.0			V	− I_{OH} = 0.8 mA, V_{CC} = 5 V
Input/output leakage current	$\|I_{IL}\|$	D_4-D_{15}, R0-R2, RA_0, RA_1			1	μA	V_{in} = 0 to V_{CC}

Note: Pull-down MOS current and output buffer current are excluded.

AC Characteristics (V_{CC} = 5 V ±10%, GND = 0 V, T_a = −20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition	Note
Oscillation frequency	f_{OSC}	OSC_1, OSC_2	0.4	8	9	MHz	Divide by 8	
Instruction cycle time	t_{cyc}		0.89	1	20	μs		
Oscillator stabilization time	t_{RC}	OSC_1, OSC_2			20	ms		1
External clock high and low widths	t_{CPH}, t_{CPL}	OSC_1	41			ns	Divide by 8	2
External clock rise time	t_{CPr}	OSC_1			15	ns		2
External clock fall time	t_{CPf}	OSC_1			15	ns		2
$\overline{INT_0}$ high width	t_{I0H}	$\overline{INT_0}$	2			t_{cyc}		3
$\overline{INT_0}$ low width	t_{I0L}	$\overline{INT_0}$	2			t_{cyc}		3
$\overline{INT_1}$ high width	t_{I1H}	$\overline{INT_1}$	2			t_{cyc}		3
$\overline{INT_1}$ low width	t_{I1L}	$\overline{INT_1}$	2			t_{cyc}		3
RESET high width	t_{RSTH}	RESET	2			t_{cyc}		4
Input capacitance	C_{in}	All pins			15	pF	f = 1 MHz, V_{in} = 0 V	
RESET fall time	t_{RSTf}				20	ms		4

Notes: 1. The oscillator stabilization time is the period from when V_{CC} reaches its minimum allowable voltage at power-on until when the oscillator stabilizes, or after RESET goes high. At power-on or recovering from stop mode, apply the RESET input for more than t_{RC} to meet the necessary time for oscillator stabilization. Since t_{RC} depends on the crystal or ceramic filter's circuit constant and stray capacitance, consult with the crystal or ceramic filter manufacturer when designing the reset circuit.
2. See figure 30.
3. See figure 31.
4. See figure 32.

Serial Interface Timing Characteristics (V_{CC} = 5 V ±10%, GND = 0 V, T_a = −20°C to +75°C, unless otherwise specified)

During Transmit Clock Output

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition	Note
Transmit clock cycle time	t_{Scyc}	$\overline{SCK}$	1			t_{cyc}		1, 2
Transmit clock high and low widths	t_{SCKH}, t_{SCKL}	$\overline{SCK}$	0.5			t_{Scyc}		1, 2
Transmit clock rise and fall times	t_{SCKr}, t_{SCKf}	$\overline{SCK}$			100	ns		1, 2
Serial output data delay time	t_{DSO}	SO			250	ns		1, 2
Serial input data setup time	t_{SSI}	SI	300			ns		1
Serial input data hold time	t_{HSI}	SI	150			ns		1

Refer to notes below.

During Transmit Clock Input

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition	Note
Transmit clock cycle time	t_{Scyc}	$\overline{SCK}$	1			t_{cyc}		1
Transmit clock high and low widths	t_{SCKH}, t_{SCKL}	$\overline{SCK}$	0.5			t_{Scyc}		1
Transmit clock rise and fall times	t_{SCKr}, t_{SCKf}	$\overline{SCK}$			100	ns		1
Serial output data delay time	t_{DSO}	SO			250	ns		1, 2
Serial input data setup time	t_{SSI}	SI	300			ns		1
Serial input data hold time	t_{HSI}	SI	150			ns		1

Notes: 1. See figure 33.
2. See figure 34.

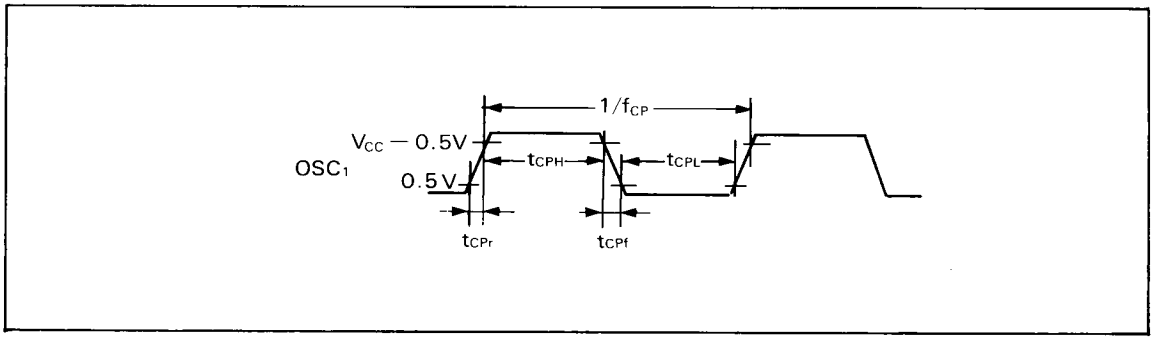


Figure 30 Oscillator Timing

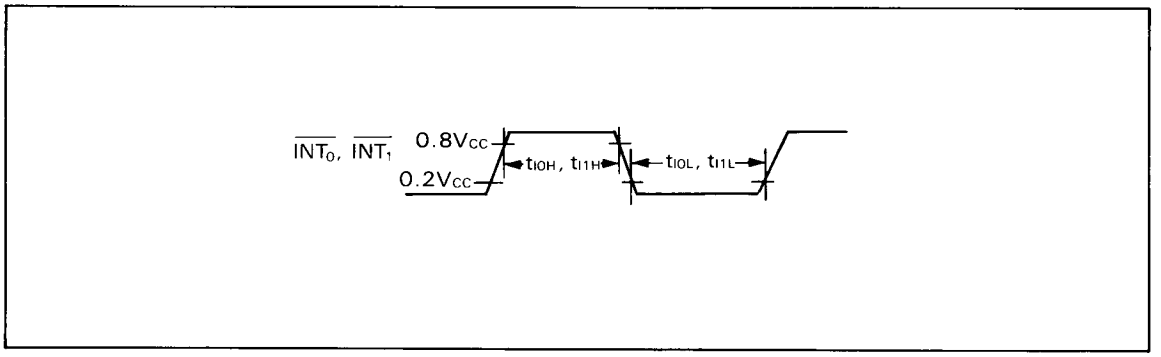


Figure 31 Interrupt Timing

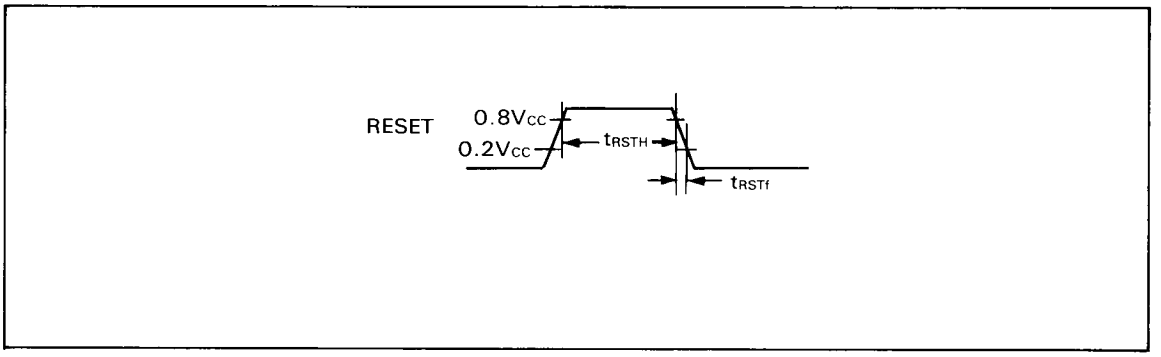
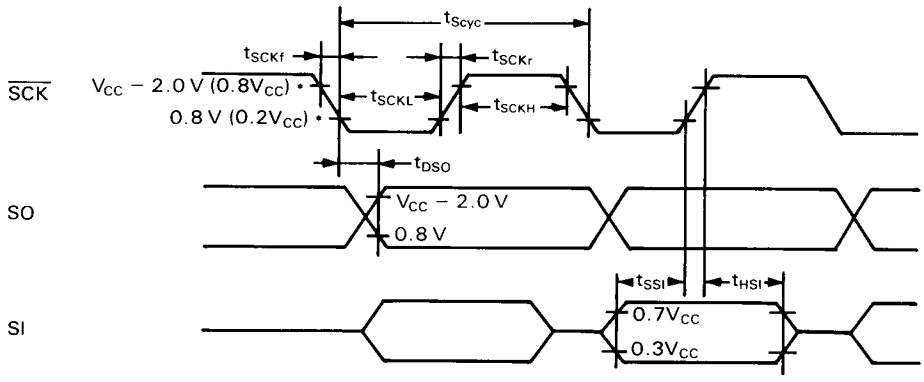


Figure 32 Reset Timing



*V_{CC} − 2.0 V and 0.8 V are the threshold voltages for transmit clock output.
0.8V_{CC} and 0.2V_{CC} are the threshold voltages for transmit clock input.

Figure 33 Timing of Serial Interface

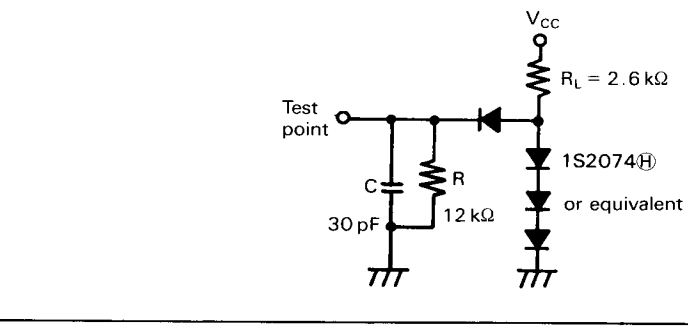


Figure 34 Timing Load Circuit

Programming Electrical Characteristics

Write and Verify Mode

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 specified)

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition
Input high voltage	V_{IH}	O_0-O_7, A_0-A_{14}, $\overline{OE}$, $\overline{CE}$	2.2		V_{CC}+0.3	V	
Input low voltage	V_{IL}	O_0-O_7, A_0-A_{14}, $\overline{OE}$, $\overline{CE}$	−0.3		0.8	V	
Output high voltage	V_{OH}	O_0-O_7	2.4			V	I_{OH} = −200 μA
Output low voltage	V_{OL}	O_0-O_7			0.4	V	I_{OL} = 1.6 mA
Input leakage current	$\|I_{IL}\|$	O_0-O_7, A_0-A_{14}, $\overline{OE}$, $\overline{CE}$			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 specified)

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Address setup time	t_{AS}	2			μs	Fig. 35*
$\overline{OE}$ setup time	t_{OES}	2			μs	
Data setup time	t_{DS}	2			μs	
Address hold time	t_{AH}	0			μs	
Data hold time	t_{DH}	2			μs	
Output disable delay time	t_{DF}			130	ns	
V_{PP} setup 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} setup time	t_{VCS}	2			μs	
Data output delay time	t_{OE}	0		500	ns	

*Input pulse level: 0.8-2.2 V
Input rising/falling time ≤ 20 ns

Input timing reference levels: 1.0 V, 2.0 V
Output timing reference levels: 0.8 V, 2.0 V

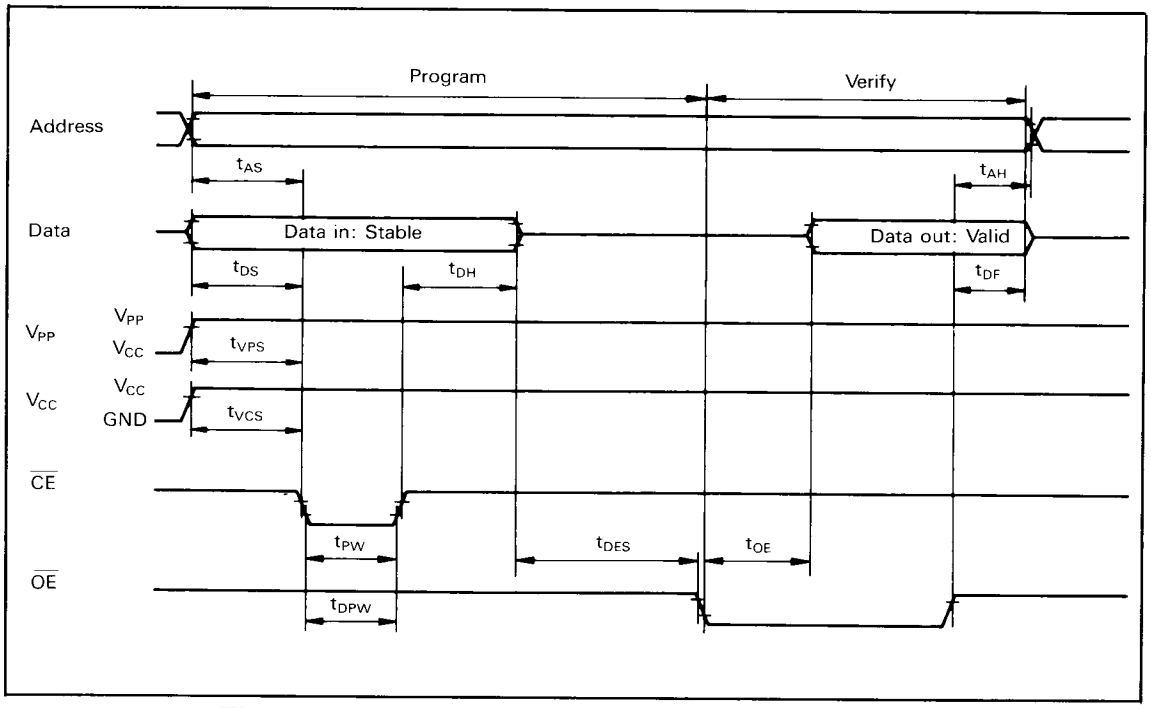


Figure 35 PROM Programming/Verification Timing

Read Mode

DC Characteristics (V_{SS} = 5 V ±10%, V_{PP} = V_{CC} ±0.6 V, T_a = 25°C ±5°C, unless otherwise specified)

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Input leakage current	I_{IL}			1	μA	V_{CC} = 5.5 V, V_{in} = GND to V_{CC}
Output leakage current	I_{OL}			1	μA	V_{CC} = 5.5 V, V_{out} = GND to V_{CC}
Progmam current	I_{PP}		1	100	μA	V_{PP} = V_{CC} + 0.6 V
Current dissipation active mode	I_{CC}			30	mA	
Input voltage	V_{IL}	−0.3		0.8	V	
	V_{IH}	2.2		V_{CC}+0.3	V	
Output voltage	V_{OL}			0.40	V	I_{OL} = 1.6 mA
	V_{OH}	2.4			V	I_{OH} = −200 μA

AC Characteristics (V_{CC} = 5 V ±10%, V_{PP} = V_{CC} ±0.6 V, T_a = 25°C ±5°C, unless otherwise specified)

Item	Symbol	Min	Max	Unit	Test Condition
Access time	t_{ACC}		500	ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}
$\overline{CE}$ output delay time	t_{CE}		500	ns	$\overline{OE}$ = V_{IL}
$\overline{OE}$ output delay time	t_{OE}	10	150	ns	$\overline{CE}$ = V_{IL}
Output disable delay time*	t_{DF}	0	105	ns	$\overline{CE}$ = V_{IL}
Data output hold time	t_{OH}	0		ns	$\overline{CE}$ = $\overline{OE}$ = V_{IL}

* T_{DF} is defined when the output becomes high impedence and cannot be referenced.

Switching characteristics:
Input pulse level: 0.8 to 2.2 V
Input rise/fall time ≤ 20 ns

Output load: 1 TTL gate + 100 pF
Output timing reference levels: 1 V, 2 V
Input timing reference levels: 0.8 V, 2 V

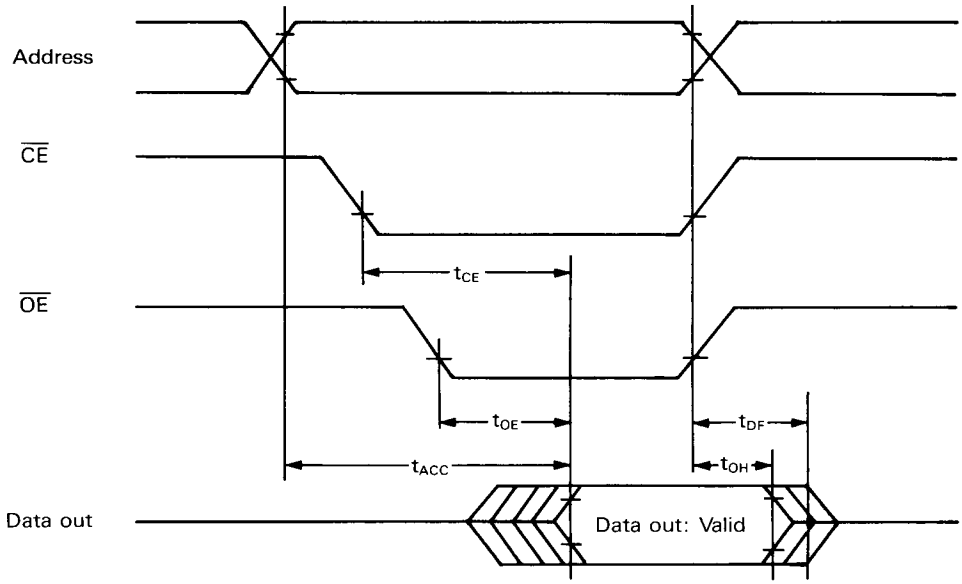


Figure 36 PROM Read Timing