SM803/SM803A SM805/SM805A

CMOS 8-Bit Single Chip Microcomuputers

■ Description

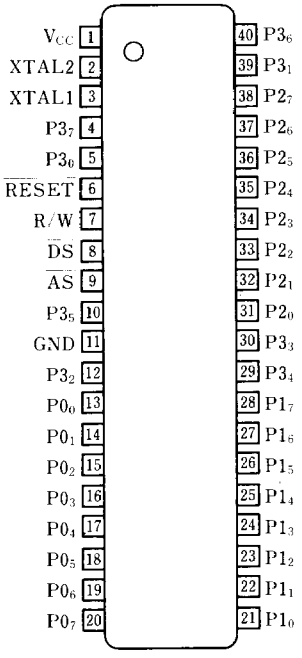
The SM803/A, SM805/A are CMOS 8-bit single chip microcomputers which have 4K bytes and 8K bytes of ROM respectively.

The devices offer faster execution; more efficient use of memory, more sophisticated interrupt, input/output and bit-manipulation capabilities, and easier system expansion.

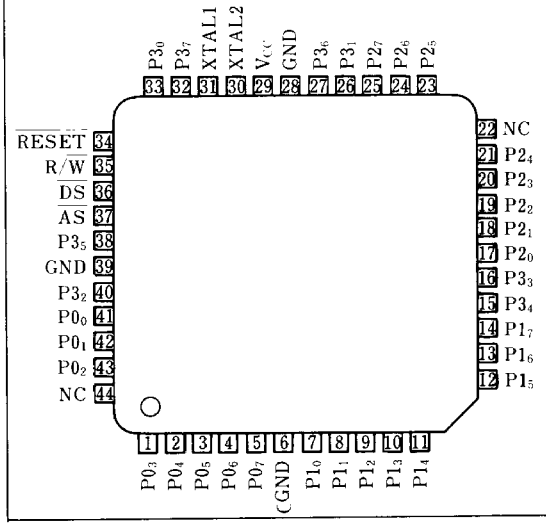
Under program control, the devices can be tailored to the user's needs. It can be configured as a stand-alone microcomputer with 4K bytes for the SM803/A or 8K bytes for the SM805/A of internal ROM, a traditional microprocessor that manages up to 120 bytes for the SM803/A or 112 bytes for the SM805/A of external memory, or a parallel processing device in a system with other processors and peripheral controllers linked by the BUS. In all configurations, a large number of pins remain available for I/O.

■ Pin Connections

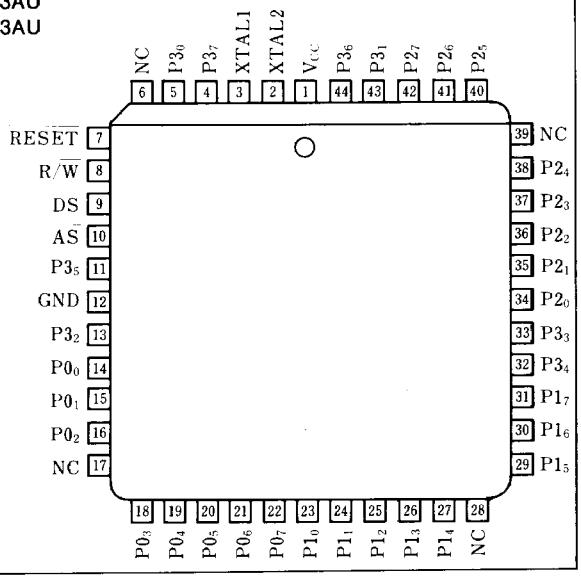
SM803/SM803A
SM805/SM805A



SM803M/SM803AM
SM805M/SM805AM



SM803U/SM803AU
SM805U/SM805AU



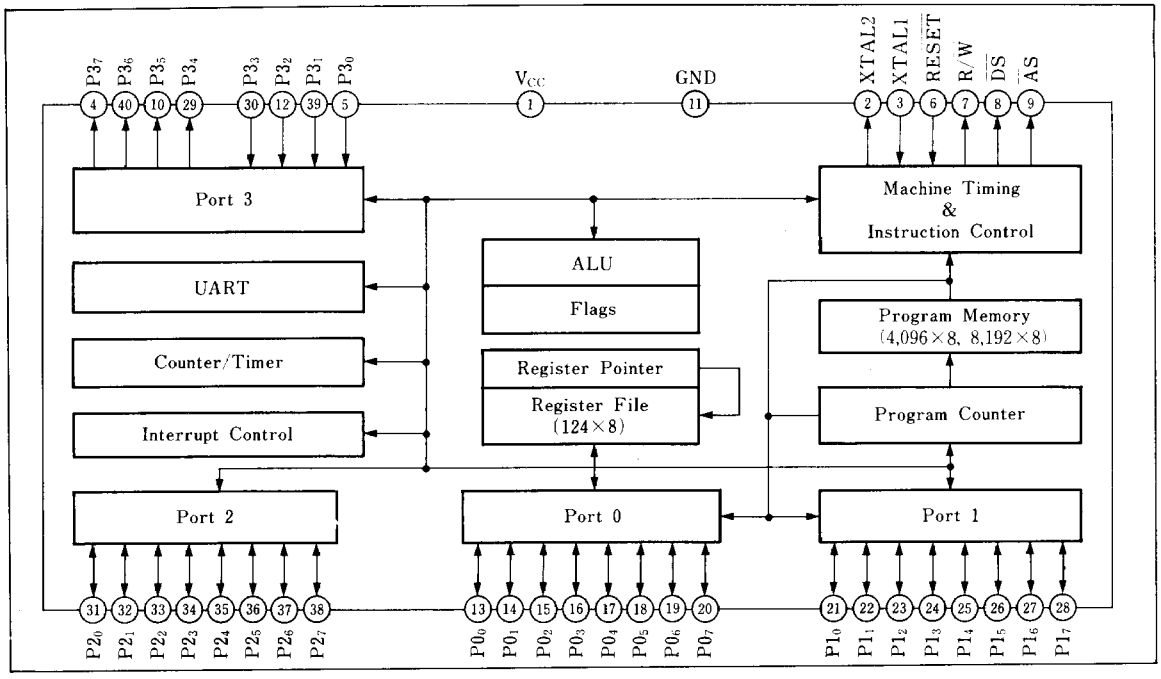
■ Features

1. Complete single-chip microcomputer with internal ROM, RAM and I/O
   RAM capacity: 124 bytes (SM803/A)
   : 236 bytes (SM805/A)
   ROM capacity: 4K bytes (SM803/A)
   : 8K bytes (SM805/A)
   I/O ports: 32
2. On-chip two programmable 8-bit couter/timers, each with a 6-bit programmable prescaler
3. Full-duplex UART
4. 144 byte register file (SM803/A)
   256 byte register file (SM805/A)
5. Register pointer so that short, fast instructions can access any working register groups
6. Vectored, priority interrupts for I/O, counter/timers, and UART
7. Up to 60K bytes for the SM803/A or 56K bytes for the SM805/A addressable external space each for program and data memory
8. On-chip oscillator
9. Maximum clock frequency
   8MHz (internal 4MHz): SM803/SM805
   8MHz (internal 6MHz): SM803A/SM805A
10. High speed instruction execution (8MHz/12MHz)
    Working register execution time:
    1.5 μs/1.0 μs
    Average instruction execution time:
    2.2 μs/1.5 μs
    Maximum instruction execution time:
    5.0 μs/3.3 μs
11. Single +5V power supply
12. 40-pin DIP (DIP40-P-600):
    SM803/A, SM805/A
    44-pin QFP (QFP44-P-1414)
    SM803M/AM, SM805M/AM
    44-pin QFJ (QFJ44-P-S650):
    SM803U/AU, SM805U/AU

■ Ordering Information

SM80X X X
- Package
  - Blank: 40-pin DIP (DIP40-P-600)
  - M: 44-pin QFP (QFP44-P-1414)
  - U: 44-pin QFJ (QFJ44-P-S650)
- Clock frequency
  - Blank: 8MHz
  - A: 12MHz
- Model No.
  - SM803 (On-chip 4K byte ROM)
  - SM805 (On-chip 8K byte ROM)

■ Block Diagram



Note: Pin numbers apply to 40-pin DIP.

Pin Description

Pin	Meaning	I/O	Function
$P0_0-P0_7$	Port 0	I/O	8-bit I/O port, programmable for I/O.
$P1_0-P1_7$	Port 1	I/O	Programmable for I/O in btyes.
$P2_0-P2_7$	Port 2	I/O	Programmable for I/O in bits.
$P3_0-P3_7$	Port 3	I/O	$P3_0$-$P3_3$ for input, $P3_4$-$P3_7$ for output.
$\overline{AS}$	Address Strobe	O	Active "Low", activated for external address memory transfer.
$\overline{DS}$	Data Strobe	O	Active "Low", activated for external data memory transfer.
R/$\overline{W}$	Read/Write	O	Read at "High", Write at "Low".
$\overline{RESET}$	Reset	I	Active "Low". Initializes.
XTAL1	Clock 1	I	Clock terminal pin.
XTAL2	Clock 2	O	Clock terminal pin.

Absolute Maximum Ratings

Parameter	Symbol	Ratings	Unit	Note
Input voltage	V_{IN}	−0.3 to V_{CC}	V	1
Output voltage	V_{OUT}	−0.3 to V_{CC}	V	
Operating temperature	Topr	0 to +70	℃	
Storage temperatuer	Tstg	−65 to +150	℃	

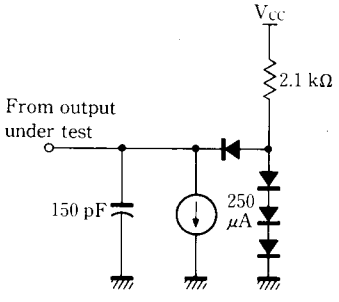
Note: The maximum applicable voltage on any pin with respect to GND.

DC Characteristics

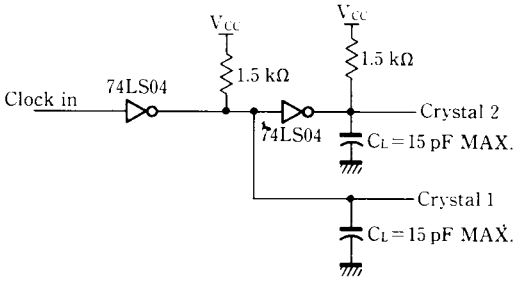
(V_{CC}=5V±10%, Ta=0 to +70℃)

Parameter	Symbol	Condition	8MHz		12MHz		Unit	Note
			MIN.	MAX.	MIN.	MAX.		
Clock input high voltage	V_{CH}	Driven by external clock oscillator	3.8	V_{CC}	3.8	V_{CC}	V	
Clock input low voltage	V_{CL}	Driven by external clock oscillator	−0.3	0.8	−0.3	0.8	V	
Input high voltage (handshaking)	V_{IH}		2.0 (2.2)	V_{CC}	2.0 (2.2)	V_{CC}	V	1
Input low voltage (handshaking)	V_{IL}		−0.3	0.8 (0.5)	−0.3	0.8 (0.5)	V	
Reset input high voltage	V_{RH}		3.8	V_{CC}	3.8	V_{CC}	V	
Reset input low voltage	V_{RL}		−0.3	0.8	−0.3	0.8	V	
Output high voltage	V_{OH}	I_{OH}=−250 μA	2.4		2.4		V	
Output low voltage	V_{OL}	I_{OL}=+2.0mA		0.4		0.4	V	
Input leakage current	I_{IL}	0V≦V_{IN}≦+5.5V	−10	10	−10	10	μA	
Output leakage current	I_{OL}	0V≦V_{IN}≦+5.5V	−10	10	−10	10	μA	
Reset input current	I_{IR}	V_{CC}=5.5V, V_{RL}=0V					μA	
Supply current	I_{CC}						mA	
Standby current	I_{CC1}	HALT instruction		7		10	mA	
	I_{CC2}	STOP intstruction		200		200	μA	

Note1: For the SM805/A, the minimum value should be 2.2V as well as when handshaking.



Test load 1

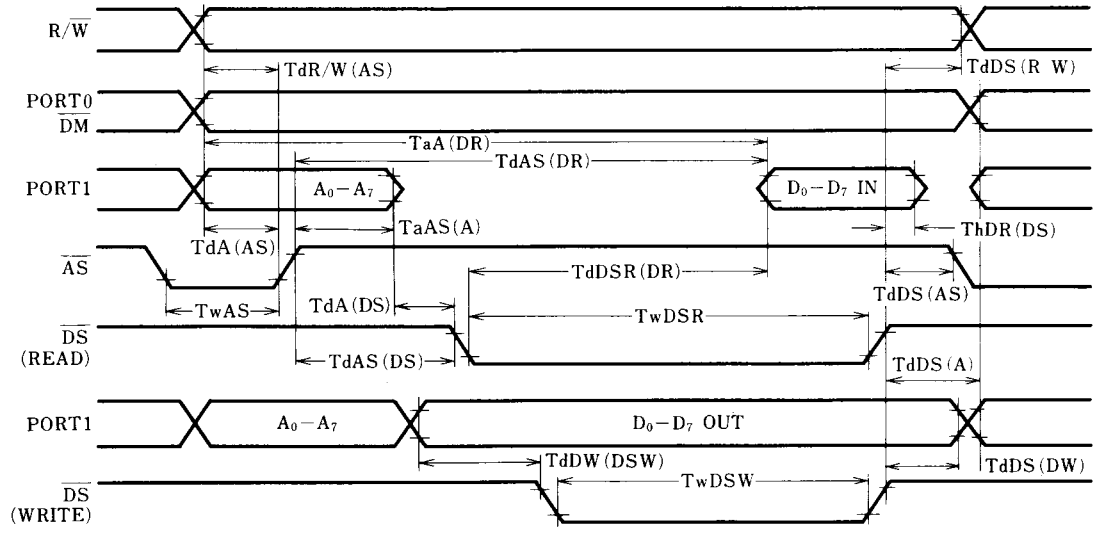


External clock generator circuit

■ External I/O or Memory Read/Write (Note 1) (V_{CC}=5V±10%, Ta=0 to +70℃)

Parameter		Symbol	8MHz		12MHz		Unit	Note
			MIN.	MAX.	MIN.	MAX.		
Address valid to $\overline{AS}$↑ delay		TdA (AS)	50		35		ns	2, 3
$\overline{AS}$↑ to input data required valid delay		TdAS (DR)		360		220	ns	2, 3, 4
$\overline{AS}$↑ low width		TwAS	80		55		ns	2, 3
$\overline{DS}$ low width	Read	TwDSR	250		185		ns	2, 3, 4
	Write	TwDSW	160		110		ns	2, 3, 4
$\overline{DS}$↓ to input data required valid		TdDSR (DR)		200		130	ns	2, 3, 4
Input data hold time		ThDSR (DS)	0		0		ns	2
$\overline{DS}$↑ to address active delay		TdDS (A)	70		45		ns	2, 3
$\overline{DS}$↑ to $\overline{AS}$↓ delay		TdDS (AS)	70		55		ns	2, 3
Read valid to $\overline{AS}$↑ delay		TdR/W (AS)	50		30		ns	2, 3
$\overline{DS}$↑ to read not valid		TdDS (R/W)	60		35		ns	2, 3
Output data valid to $\overline{DS}$↓ delay		TdDW (DSW)	50		35		ns	2, 3
$\overline{DS}$↑ to output data not valid delay		TdDS (DW)	70		45		ns	2, 3
Write valid to $\overline{AS}$↑ delay		TdA (DR)		410		255	ns	2, 3, 4
$\overline{DS}$ to write not valid delay		TdAS (DS)	80		55		ns	2, 3

Note 1: All timing references use 2.0 V for a logic "1" and 0.8 V for a logic "0".
Note 2: Test load 1
Note 3: The timing is defined at the minimum cycle of TpC.
Note 4: Apply double cycle of input clock TpC for the expansion memory timing.

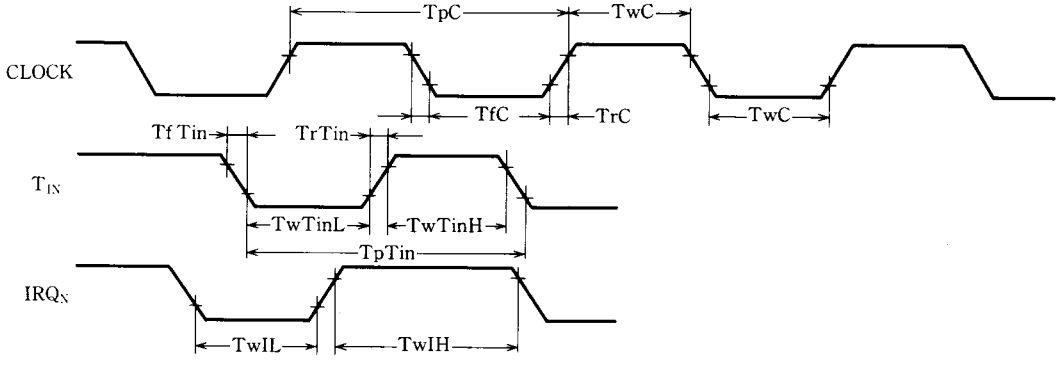


■ Input Clock, Timer Input, Interrupt Request Input

(V_{CC}=5V±10%, to=0 to +70℃)

Parameter	Symbol	8MHz		12MHz		Unit	Note
		MIN.	MAX.	MIN.	MAX.		
Input clock cycle	TpC	125	1000	83	1000	ns	1
Input clock rise, fall time	TrC, TfC		25		15	ns	1
Input clock width	TwC	37		26		ns	1
Timer input low width	TwTinL	100		70		ns	2
Timer input high width	TwTinH	3TpC		3TpC		ns	2
Timer input cycle	TpTin	8TpC		8TpC		ns	2
Timer input rise, fall time	TrTin. TfTin		100		100	ns	2
Interrupt request input low time	TwIL	100		70		ns	2, 3
		3TpC		3TpC		ns	2, 4
Interrupt request input high time	TwIH	3TpC		3TpC		ns	2, 3

Note 1: The clock timing references use 3.8V for a logic "1" and 0.8V for logic "0".
Note 2: The timing references use 2.0V (2.2V for SM805/A) for a logic "1" and 0.8V for a logic "0".
Note 3: Interrupt request from port 3 ($P3_1$-$P3_3$).
Note 4: Interrupt request from port 3 ($P3_0$).

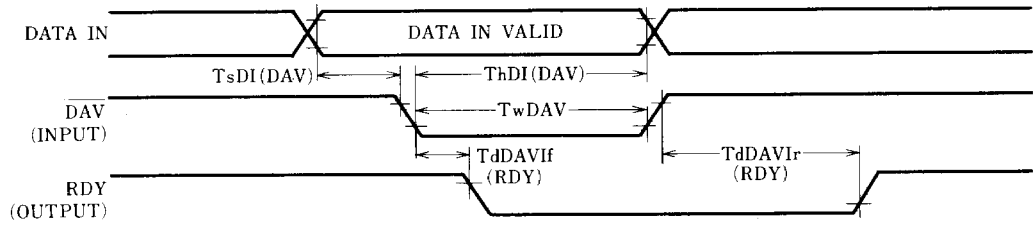


■ Handshake Timing (Note 1)

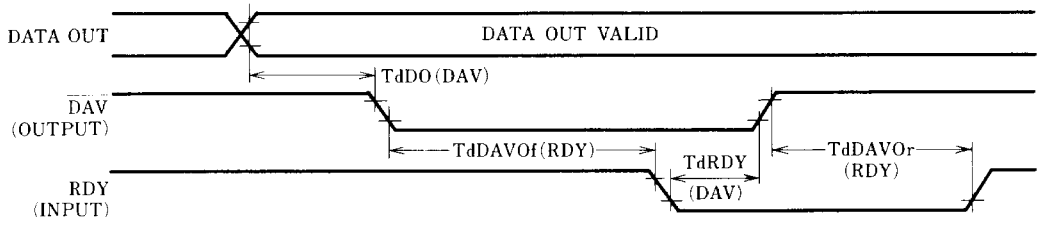
(V_{CC}=5V±10%, Ta=0 to +70℃)

Parameter	Symbol	8MHz		12MHz		Unit	Note
		MIN.	MAX.	MIN.	MAX.		
Data input setup time	TsDI (DAV)	0		0		ns	
Data input hold time	ThDI (DAV)	230		160		ns	
Data valid signal input width	TwDAV	175		120		ns	
$\overline{DAV}$ ↓ input to RDY ↓ delay time	TdDAVIf (RDY)		175		120	ns	2, 3
$\overline{DAV}$ ↓ output to RDY ↓ delay time	TdDAVOf (RDY)	0		0		ns	2, 4
$\overline{DAV}$ ↑ input to RDY ↑ delay time	TdDAVIr (RDY)		175		120	ns	2, 3, 5
$\overline{DAV}$ ↑ output to RDY ↑ delay time	TdDAVOr (RDY)	0		0		ns	2, 4
Data output to $\overline{DAV}$ ↓ delay time	TdDO (DAV)	50		30		ns	2
RDY ↓ input to $\overline{DAV}$ ↑ delay time	TdRDY (DAV)	0	200	0	140	ns	2

Note 1: All timing references use 2.0V for a logic "1" and 0.8V for a logic "0".
Note 2: Test load 1.
Note 3: Input handshake
Note 4: Output handshake
Note 5: When read out from the port before $\overline{DAV}$ ↑ input.



Input handshake



Output handshake

■ Architecture

(1) Address Spaces

(i) Program Memory The 16-bit program counter addresses 64K bytes of program memory space. Program memory can be located in two areas : one internal and the other external (Fig. 2). The first 2048 bytes consist of on-chip mask-programmed ROM. At addresses 2048 and greater, the Z8 executes external program memory fetches.

The first 12 bytes of program memory are reserved for the interrupt vectors. These locations contain six 16-bit vectors that correspond to the six available interrupts.

(ii) Data Memory The Z8 can address 62K bytes of external data memory beginning at location 2048 (Fig. 3). External data memory may be include with or separated from the external program memory space. $\overline{DM}$, an optical I/O function that can be programmed to appear on pin $P3_1$, is used to distinguish between data and program memory space.

(iii) Register File The 144-byte register file includes four I/O port registers (R0-R3), 124 general-purpose registers (R4-R127) and 16 control and status registers (R240-R255). These registers are assigned the address locations shown in Fig. 4.

Z8 instructions can access registers directly or indirectly with an 8-bit address field. The Z8 also allows short 4-bit register addressing using the Register Pointer (one of the control registers). In the 4-bit mode, the register file is divided into nine working-register groups, each occupying 16 contiguous locations. The Register Pointer addresses the starting location of the active working-regsister group.

The 4-bit address specifies the nth (0 to 15) address from the starting location (see Fig. 5).

*The addresses OEO_H-OEF_H of SM805 register file can not be directly accessed due to the essential function of the register pointer. Either of the following two methods is available for accessing those 16 registers.

1) Working register addressing
   SRP #OEO_H (set the RP to $0EO_H$)
2) Register indirect addressing
   ex.) LD 70H, #$0EO_H$
   LD 40H, @70H (read)
   LD @70H, 40H (write)

(iv) Stacks Either the internal register file or the external data memory can be used for the stack. A 16-bit Stack Pointer (R254 and R255) is used for the external stack, which can reside anywhere in data memory between locations 4096 (8192 for SM805/A) and 65535. An 8-bit Stack Pointer (R255) is used for the internal stack that resides within the 124 (236 for SM805/A) general-purpose registers.

Either an internal stack or an external stack may be selected with ports 0, 1 and the bit D_2 of mode register (248). The internal stack is specified with the device to be reset.

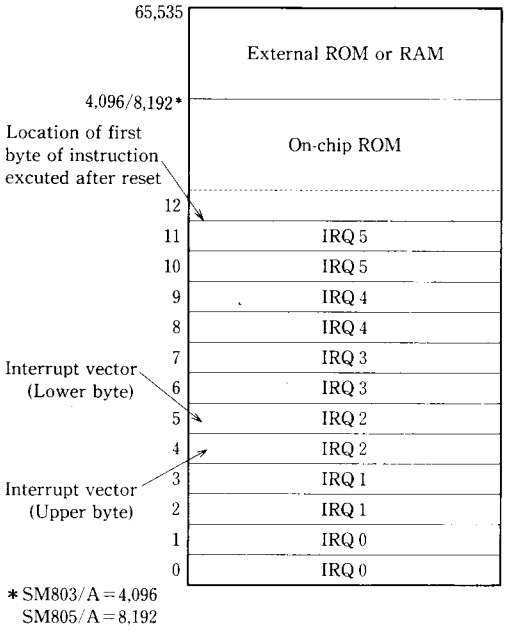


Fig. 2 Program memory map

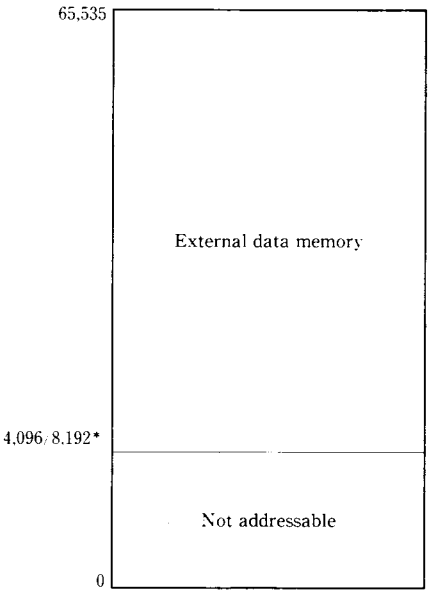


Fig. 3 Data memory map

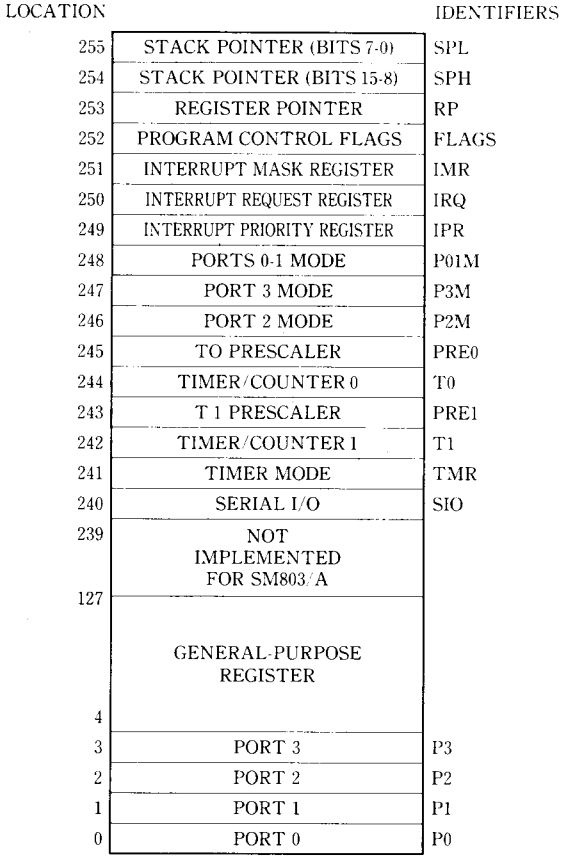


Fig. 4 The register file

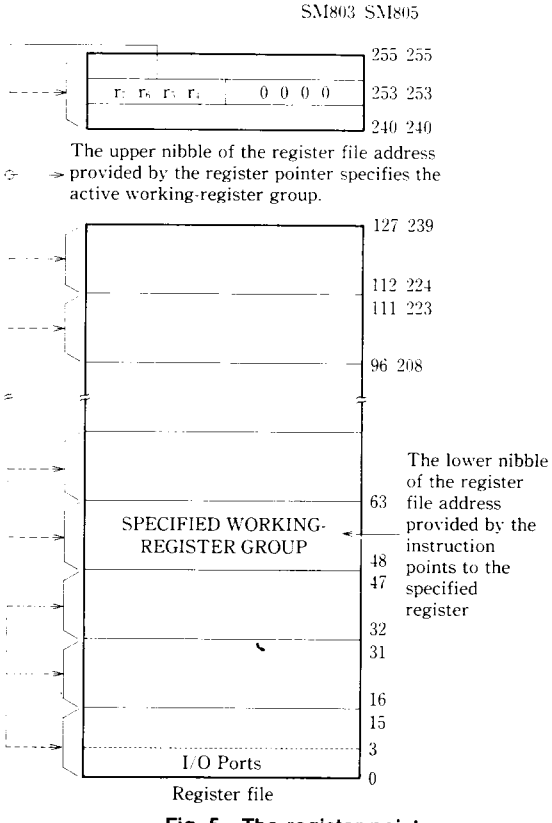


Fig. 5 The register pointer

(2) I/O ports

The Z8 has 32 lines dedicated to input and output. These lines are grouped into four ports of eight lines each and are configurable as input, output or address/data. Under software control, the ports can be programmed to provide address outputs, timing, status signals, serial I/O, and parallel I/O with or without handshake. All ports have active pull-ups and pull-downs compatible with TTL loads.

(i) Port 1 can be programmed as a byte I/O port or an address/data port for interfacing external memory.

Memory locations greater than 4095 (8191 for SM805/A) are referenced through Port 1. To interface external memory, Port 1 must be programmed for the multiplexed Adress/Data mode. If more than 257 external locations are required, Port 0 must output the additional lines.

(ii) Port 0 can be programmed as a nibble I/O port, or as an address port for interfacing external memory.

For external memory references, Port 0 can provide address bits A_8A_{11} (lower nibble) or A_8-A_{15} (lower and upper nibble) depending on the required address space.

(iii) Port 2 bits can be programmed independently as input or output. The port is always available for I/O operations. In addition, Port 2 can be configured to provide open-drain outputs.

(iv) Port 3 lines can be configured as I/O or control lines. In either cases, the direction of the eight lines is fixed as four input ($P3_0$-$P3_3$) and four output ($P3_4$-$P3_7$). For serial I/O, lines $P3_0$ and $P3_7$ are programmed as serial in and serial out respectively

- handshake for Ports 0, 1 ind 2 ($\overline{DAV}$ and RDY)
- four external interrupt request signals (IRQ_0-IRQ_3)
- timer input and output signals (T_{IN} and T_{OUT})
- Data Memory Select ($\overline{DM}$).

(3) Serial Input/Output

Port 3 lines $P3_0$ and $P3_7$ can be programmed as serial I/O lines for full-duplex serial asynchronous receiver/transmitter operation. The bit rate is controlled by Counter/Timer 0, with a maximum rate of 62.5K-bits/second.

The device automatically adds a start bit and two stop bits to transmitted data (Fig. 6). Odd parity is also aviliable as an option.

(4) Counter/Timer

The device contains two 8-bit programmable counter/timers (T_0 and T_1), each driven by its own 6-bit programmable prescaler. The T_1 prescaler can be driven by internal or external clock sources; however, the T_0 prescaler is driven by the internal clock only.

The counters can be started, stopped, restarted to continue, or restarted from the initial value. The counters can also be programmed to stop upon reaching zero (single-pass mode) or to automatically reload the initial value and continue counting (modulo-n continuos mode). The conters, but not the prescalers, can be read any time without disturbing their value or count mode.

(5) Interrupts

The device allows six different interrupts from eight sources: the four Port 3 lines $P3_0$-$P3_3$, Serial In, Serial Out, and the two conter/timers. These interrupts are both maskable and prioritized.

All device interrupts are vectored. Polled interrupt systems are also supported.

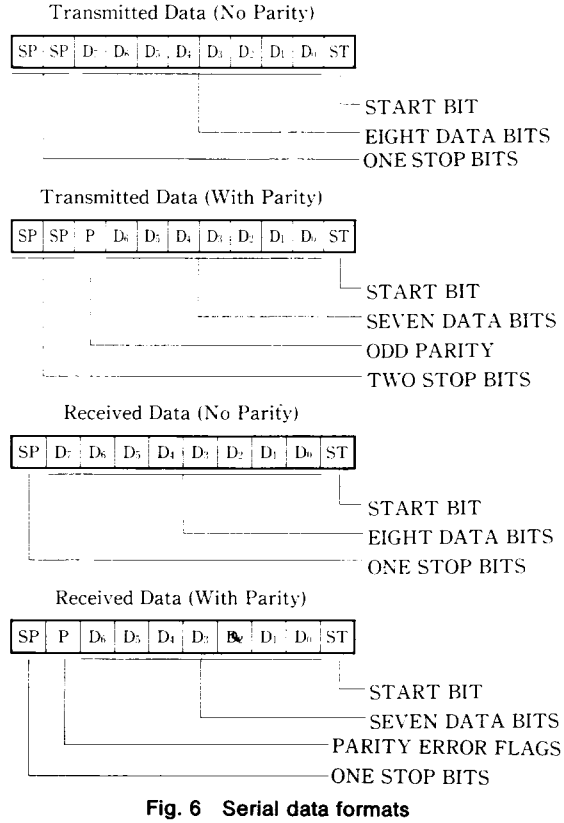


Fig. 6 Serial data formats

■ Instruction Set Notation

(1) Addressing modes

The following notation is used to describe the addressing modes and instruction operations as shown in the instruction summary.

IRR	Indirect register pair or indirect working-register pair address
Irr	Indirect working-register pair only
X	Indexed address
DA	Direct address
RA	Relative address
IM	Immediate
R	Register or working-register address
r	Working-register address only
IR	Indirect-register or indirect working-register address
Ir	Indirect working-register address only
RR	Register pair or working register pair address

(2) Symbols

The following symbols are used in describing the instruction set.

dst	Destination location or contents
src	Source location or contents
cc	Condition code (see list)
@	Indirect address prefix
SP	Stack pointer (control registers 254-255)
PC	Program counter
FLAGS	Flag register (control register 252)
RP	Register pointer (control register 253)
IMR	Interrupt mask register (control register 251)

Assignment of a value is indicated by the symbol "←". For example.

dst ← dst + src

indicates that the source data is added to the destination data and the result is stored in the destination location. The notation "addr (n)" is used to refer to bit "n" of a given location. For example, dst (7) refers to bit 7 of the destination operand.

(3) Flags

Control Register R252 contains the following six flags :

C	Carry flag
Z	Zero flag
S	Sign flag
V	Overflow flag
D	Decimal-adjust flag
H	Half-carry flag

Affected flags are indicated by :

0	Cleared to zero
1	Set to one
*	Set or cleared according to operation
–	Unaffected
×	Undefined

(4) Condition codes

See Table 1.

Table 1 Condition codes

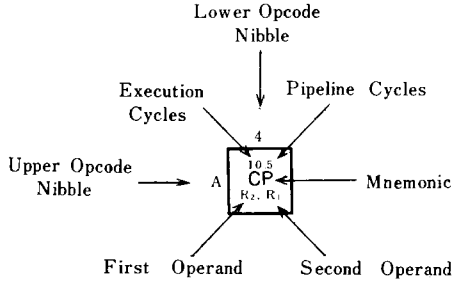
Value	Mnemonic	Meaning	Flags set
1000		Always true	.....
0111	C	Carry	C=1
1111	NC	No carry	C=0
0110	Z	Zero	Z=1
1110	NZ	Not Zero	Z=0
1101	PL	Plus	S=0
0101	MI	Minus	S=1
0100	OV	Overflow	V=1
1100	NOV	No overflow	V=0
0110	EQ	Equal	Z=1
1110	NE	Not equal	Z=0
1001	GE	Greater than or equal	(S XOR V) =0
0001	LT	Less than	(S XOR V) =1
1010	GT	Greater than	[Z OR (S XOR V)] =0
0010	LE	Less than or equal	[Z OR (S XOR V)] =1
1111	UGE	Unsigned greater than or equal	C=0
0111	ULT	Unsigned less than	C=1
1011	UGT	Unsigned greater than	(C=0 AND Z=0) =1
0011	ULE	Unsigned less than or equal	(C OR Z) =1
0000		Never true	.....

(5) Opcode map

Lower Nibble (Hex)

Upper Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	6.5 DEC R_1	6.5 DEC IR_1	6.5 ADD r_1, r_2	6.5 ADD r_1, Ir_2	10.5 ADD R_2, R_1	10.5 ADD IR_2, R_1	10.5 ADD R_1, IM	10.5 ADD IR_1, IM	6.5 LD r_1, R_2	6.5 LD r_2, R_1	12/10.5 DJNZ r_1, RA	12/10.0 JR cc, RA	6.5 LD r_1, IM	12/10.0 JP cc, DA	6.5 INC r_1	
1	6.5 RLC R_1	6.5 RLC IR_1	6.5 ADC r_1, r_2	6.5 ADC r_1, Ir_2	10.5 ADC R_2, R_1	10.5 ADC IR_2, R_1	10.5 ADC R_1, IM	10.5 ADC IR_1, IM								
2	6.5 INC R_1	6.5 INC IR_1	6.5 SUB r_1, r_2	6.5 SUB r_1, Ir_2	10.5 SUB R_2, R_1	10.5 SUB IR_2, R_1	10.5 SUB R_1, IM	10.5 SUB IR_1, IM								
3	8.0 JP IRR_1	6.1 SRP IM	6.5 SBC r_1, r_2	6.5 SBC r_1, Ir_2	10.5 SBC R_2, R_1	10.5 SBC IR_2, R_1	10.5 SBC R_1, IM	10.5 SBC IR_1, IM								
4	8.5 DA R_1	8.5 DA IR_1	6.5 OR r_1, r_2	6.5 OR r_1, Ir_2	10.5 OR R_2, R_1	10.5 OR IR_2, R_1	10.5 OR R_1, IM	10.5 OR IR_1, IM								
5	10.5 POP R_1	10.5 POP IR_1	6.5 AND r_1, r_2	6.5 AND r_1, Ir_2	10.5 AND R_2, R_1	10.5 AND IR_2, R_1	10.5 AND R_1, IM	10.5 AND IR_1, IM								
6	6.5 COM R_1	6.5 COM IR_1	6.5 TCM r_1, r_2	6.5 TCM r_1, Ir_2	10.5 TCM R_2, R_1	10.5 TCM IR_2, R_1	10.5 TCM R_1, IM	10.5 TCM IR_1, IM								6.0 STOP
7	10/12.1 PUSH R_1	12/14.1 PUSH IR_2	6.5 TM r_1, r_2	6.5 TM r_1, Ir_2	10.5 TM R_2, R_1	10.5 TM IR_2, R_1	10.5 TM R_1, IM	10.5 TM IR_1, IM								6.0 HALT
8	10.5 DECW RR_1	10.5 DECW IR_1	12.0 LDE r_1, Irr_2	18.0 LDEI Ir_1, Irr_2												6.1 DI
9	6.5 RL R_1	6.5 RL IR_1	12.0 LDE r_2, Irr_1	18.0 LDEI Ir_2, Irr_1												6.1 EI
A	10.5 INCW RR_1	10.5 INCW IR_1	6.5 CP r_1, r_2	6.5 CP r_1, Ir_2	10.5 CP R_2, R_1	10.5 CP IR_2, R_1	10.5 CP R_1, IM	10.5 CP IR_1, IM								14.0 RET
B	6.5 CLR R_1	6.5 CLR IR_1	6.5 XOR r_1, r_2	6.5 XOR r_1, Ir_2	10.5 XOR R_2, R_1	10.5 XOR IR_2, R_1	10.5 XOR R_1, IM	10.5 XOR IR_1, IM								16.0 IRET
C	6.5 RRC R_1	6.5 RRC IR_1	12.0 LDC r_1, Irr_2	18.0 LDCI Ir_1, Irr_2				10.5 LD r_1, x, R_2								6.5 RCF
D	6.5 SRA R_1	6.5 SRA IR_1	12.0 LDC r_2, Irr_1	18.0 LDCI Ir_2, Irr_1	20.0 CALL* IRR_1		20.0 CALL DA	10.5 LD r_2, x, R_1								6.5 SCF
E	6.5 RR R_1	6.5 RR IR_1		6.5 LD r_1, Ir_2	10.5 LD R_2, R_1	10.5 LD IR_2, R_1	10.5 LD R_1, IM	10.5 LD IR_1, IM								6.5 CCF
F	8.5 SWAP R_1	8.5 SWAP IR_1		6.5 LD Ir_1, r_2		10.5 LD R_2, IR_1										6.0 NOP

Bytes per Instruction: 3 | 2 | 3 | 1



Legend:
R = 8-Bit Address
r = 4-Bit Address
R_1 or r_1 = Dst Address
R_2 or r_2 = Src Address

Sequence:
Opcode, First Operand, Second Operand

Note: The blank areas are not defined.

(6) Instruction Summary

Instruction and Operation	Addr Mode dst	src	Opcode Byte (Hex)	Flags Affected C Z S V D H
ADC dst,src dst←dst+src+C	(Note 1)		1□	* * * * 0 *
ADD dst,src dst←dst+src	(Note 1)		0□	* * * * 0 *
AND dst,src dst←dst AND src	(Note 1)		5□	- * * 0 - -
CALL dst SP←SP−2 @SP←PC;PC←dst	DA IRR		D6 D4	- - - - - -
CCF C←NOT C			E F	* - - - - -
CLR dst dst← 0	R IR		B0 B1	- - - - - -
COM dst dst←NOT dst	R IR		60 61	- * * 0 - -
CP dst,src dst←src	(Note 1)		A□	* * * * - -
DA dst dst←DA dst	R IR		40 41	* * * X - -
DEC dst dst←dst−1	R IR		00 01	- * * * - -
DECW dst dst←dst−1	RR IR		80 81	- * * * - -
DI IMR(7)←0			8F	- - - - - -
DJNZ r,dst r←r−1 if r 0 PC←PC+dst Range: +127,−128	RA		rA r=0−F	- - - - -
EI IMR(7) ←1			9F	- - - - - -
HALF			FF 7F	- - - - - -
INC dst dst←dst+1	r R IR		rE r=0−F 20 21	- * * * - -
INCW dst dst←dst+1	RR IR		A0 A1	- * * * - -
IRET FLAGS←@SP; SP←SP+1 PC←@SP;SP←SP+2;IMR(7)←1			BF	* * * * * *
JP cc,dst if cc is true PC←dst	DA IRR		cD c=0−F 30	- - - - - -
JR cc,dst if cc is true, PC←PC+dst Range: +127,−128	RA		cB c=0−F	- - - - - -
LD dst,src dst←src	r IM r R R r r X X r r Ir Ir r R R R IR R IM IR IM IR R		rC r8 r9 r=0−F C7 D7 E3 F3 E4 E5 E6 E7 F5	- - - - - -
LDC dst,src dst←src	r Irr Irr r		C2 D2	- - - - - -
LDCI dst,src dst←src r←r+1;rr←rr+1	Ir Irr Irr Ir		C3 D3	- - - - - -
LDE dst,src dst←src	r Irr Irr r		82 92	- - - - - -
LDEI dst,src dst←src r←r+1;rr←rr+1	Ir Irr Irr Ir		83 93	- - - - - -
NOP			FF	- - - - - -
OR dst,src dst←dst OR src	(Note 1)		4□	- * * 0 - -
POP dst dst←@SP SP←SP+1	R IR		50 51	- - - - - -
PUSH src SP←SP−1;@SP←src	R IR		70 71	- - - - - -
RCF C← 0			C F	0 - - - - -
RET PC @SP;SP←SP+2			A F	- - - - - -
RL dst	R IR		90 91	* * * * - -
RLC dst	R IR		10 11	* * * * - -
RR dst	R IR		E0 E1	* * * * - -
RRC dst	R IR		C0 C1	* * * * - -
SBC dst,src dst←dst−src−C	(Note 1)		3□	* * * * 1 *
SCF C←1			D F	1 - - - - -
SRA dst	R IR		D0 D1	* * * 0 - -
SRP src RP←src	IM		31	- - - - - -
STOP			FF 6F	- - - - - -
SUB dst,src dst←dst−src	(Note 1)		2□	* * * * 1 *
SWAP dst	R IR		F0 F1	X * * X - -
TCM dst,src (NOT dst) AND src	(Note 1)		6□	- * * 0 - -
TM dst,src dst AND src	(Note 1)		7□	- * * 0 - -
XOR dst,src dst←dst XOR src	(Note 1)		B□	- * * 0 - -

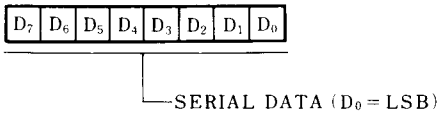
Note 1 These instructions have an identical set of addressing modes, which are encoded for brevity. The first opcode nibble is found in the instruction set table above. The second nibble is expressed symbolically by a □ in this table, and its value is found in the following table to the left of the applicable addressing mode pair.

For example, to determine the opcode of an ADC instruction use the addressing modes r (destination) and Ir (source). The result is 13

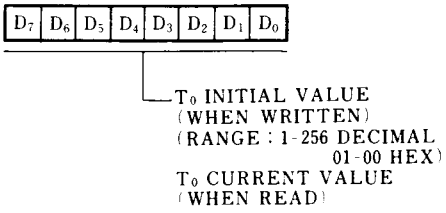
Addr Mode dst	Addr Mode src	Lower Opcode Nibble
r	r	2
r	Ir	3
R	R	4
R	IR	5
R	IM	6
IR	IM	7

■ Register

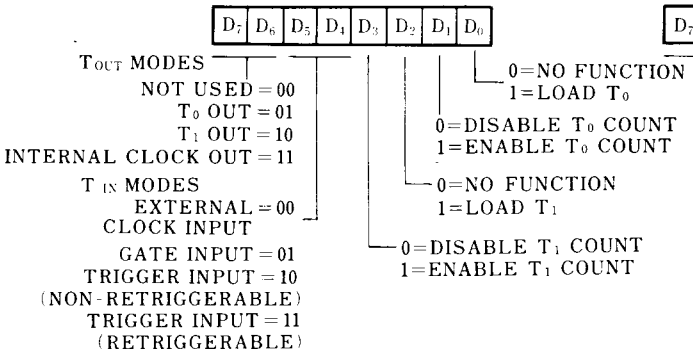
R240 (SIO)
Serial I/O Register
($F0_H$: Read/Write)



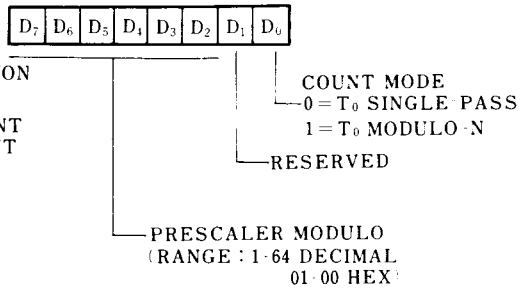
R244 (T0)
Counter/Timer 0 Register
($F4_H$: Read/Write)



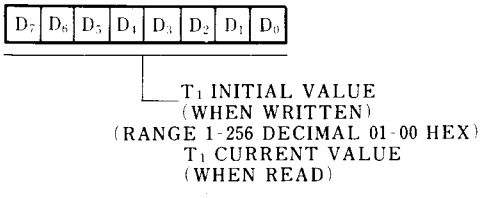
R241 (TMR)
Timer Mode Register
($F1_H$: Read/Write)



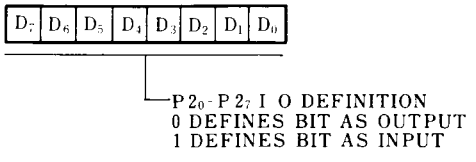
R245 (PRE0)
Prescaler 0 Register
($F5_H$: Write Only)



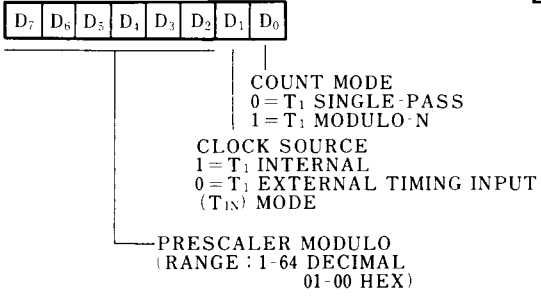
R242 (T1)
Counter Timer 1 Register
($F2_H$: Read/Write)



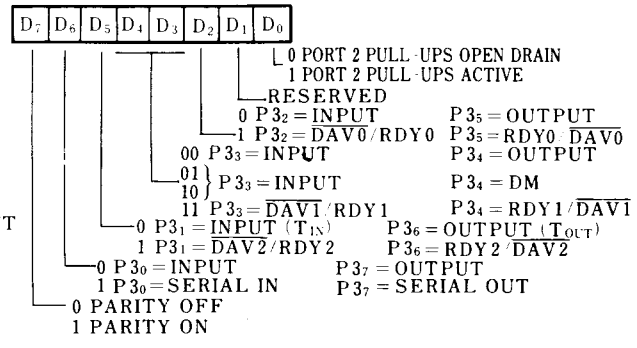
R246 (P2M)
Port 2 Mode Register
($F6_H$: Write Only)



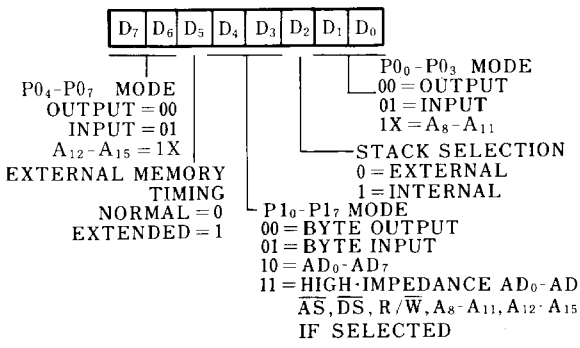
R243 (PRE1)
Prescaler 1 Register
($F3_H$: Write Only)



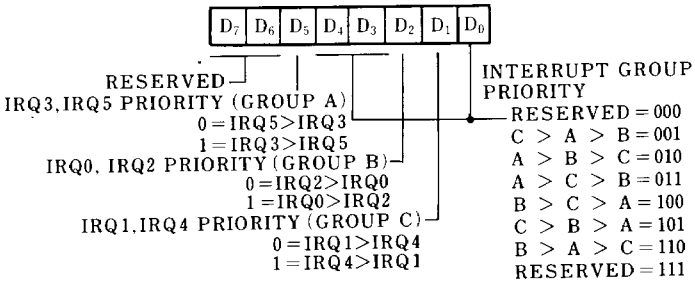
R247 (P3M)
Port 3 Mode Register
($F7_H$: Write Only)



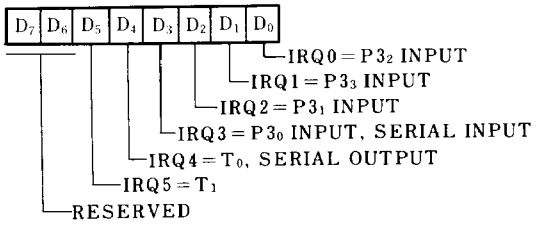
R248 (P01M)
Port 0 and 1 Mode Register
($F8_H$: Write Only)



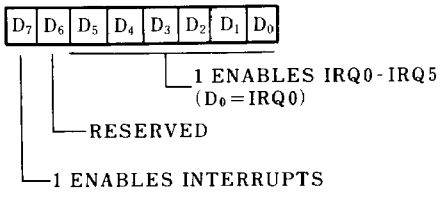
R249 (IPR)
Interrupt Priority Register
($F9_H$: Write Only)



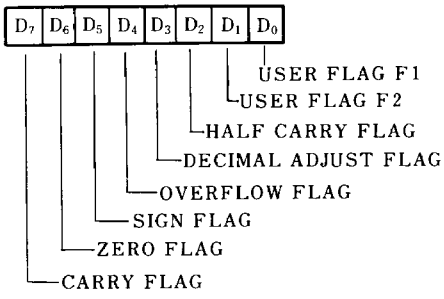
R250 (IRQ)
Interrupt Request Register
(FA_H : Read/Write)



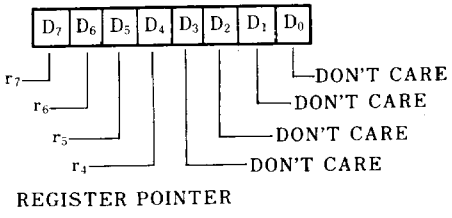
R251 (IMR)
Interrupt Mask Register
(FB_H : Read/Write)



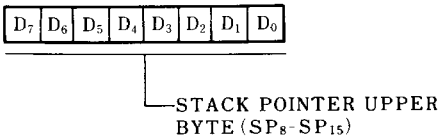
R252 (FLAGS)
Flag Register
(FC_H : Read/Write)



R253 (RP)
Register Pointer
(FD_H : Read/Write)



R254 (SPH)
Stack Pointer
(FE_H : Read/Write)



R255 (SPL)
Stack Pointer
(FF_H : Read/Write)