M5L8085AP

8-BIT PARALLEL MICROPROCESSOR

DESCRIPTION

The M5L8085AP is a family of single-chip 8-bit parallel central processing units (CPUs) developed using the N-channel silicon-gate ED-MOS process. It requires a single 5V power supply and has a basic clock rate of 3MHz.

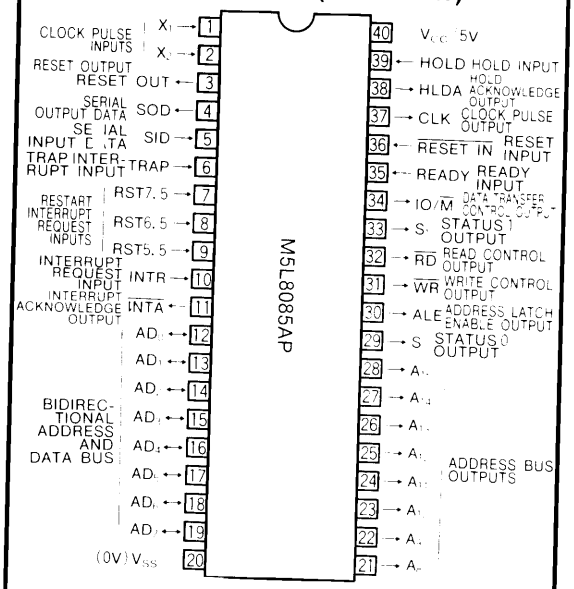
FEATURES

- Single 5V supply voltage
- TTL compatible
- Instruction cycle ..... 1.3 μs (min.)
- Clock generator (with an external crystal or RC circuit)
- Built-in system controller
- Four vectored interrupts (one of which is non-maskable)
- Serial I/O port ..... 1 each
- Decimal, binary, and double precision arithmetic operations
- Direct Addressing capability to 64K bytes of memory

APPLICATION

Central processing unit for a microcomputer

PIN CONFIGURATION (TOP VIEW)

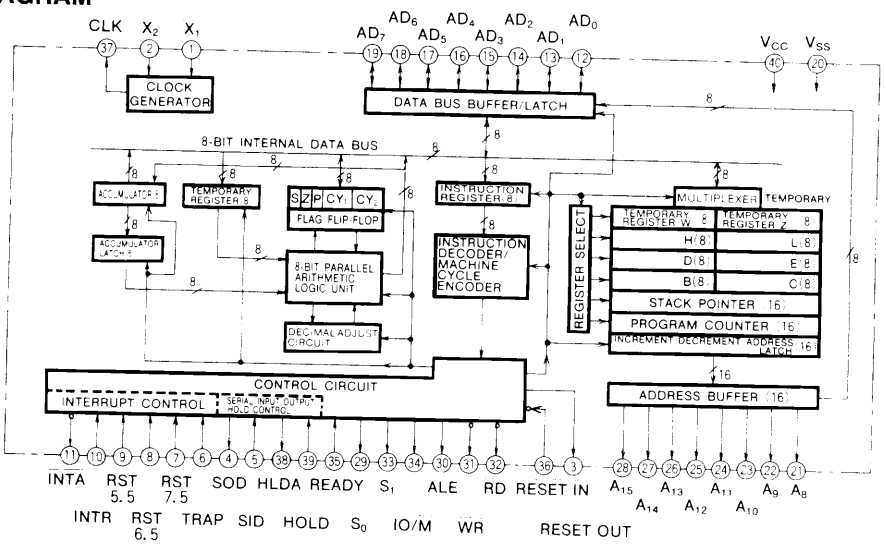


Outline 40P4

FUNCTION

Under the multiplexed data bus concept adopted, the high-order 8 bits of the address are used only as an adress bus and the low-order 8 bits are used as an address/data bus. During the first clock cycle of an instruction cycle, the address is transferred. The low-order 8 bits of the address are stored in the external latch by the address latch enable (ALE) signal. During the second and third clock cycles, the address/data bus functions as the data bus, transferring the data to memory or to the I/O. For bus control, the device provides RD, WR, and IO/M signals and an interrupt acknowledge signal (INTA). The HOLD, READY and all interrupt signals are synchronized with the clock pulse. For simple serial data transfer it provides both a serial input data (SID) line and a serial output data (SOD) line. It also has three maskable restart interrupts and one non-maskable trap interrupt.

BLOCK DIAGRAM



PIN DESCRIPTIONS

Pin	Name	Input or output	Functions
X_1, X_2	Clock input	In	These pins are used to connect an external crystal or RC circuit to the internal clock generator. An external clock pulse can also be input through X_1.
RESET OUT	Reset output	Out	This signal indicates that the CPU is in the reset mode. It can be used as a system RESET. The signal is synchronised to the processor clock.
SOD	Serial output data	Out	This is an output data line for serial data. The output SOD may be set or reset by means of the SIM instruction. It returns to high-level after the RESET.
SID	Serial input data	In	This is an input data line for serial data, and the data on this line is moved to the 7th bit of the accumulator whenever a RIM instruction is executed.
TRAP	Trap interrupt	In	A non-maskable restart which is recognized at the same time as an INTR it is not affected by any mask or another interrupt. It has the highest interrupt priority.
RST5.5 RST6.5 RST7.5	Restart interrupt request	In	Input timing is the same as for INTR for these three signals. They all cause an automatic insertion of an internal RESTART. RST 7.5 has the highest priority while RST 5.5 has the lowest. All three signals have a higher priority than INTR.
INTR	Interrupt request signal	In	This signal is for a general purpose interrupt and is sampled only during the last clock cycle of the instruction. When an interrupt is acknowledged, the program counter (PC) is held and an INTA signal is generated. During this cycle, a RESTART or CALL can be inserted to jump to an interrupt service routine. The interrupt request may be enable and disable by means of software. But it is disable by the RESET and immeadiately after an accepted interrupt.
$\overline{INTA}$	Interrupt acknowledge control signal	Out	This signal is used instead of RD during the instruction cycle after an INTR is accepted.
AD_0~AD_7	Bidirectional address and data bus	In/out	The low-order (I/O address) appears during the first clock cycle. During the second and third clock cycles, it becomes the data bus. It remains in the high-impedance state during the HOLD and HALT modes.
A_8~A_{15}	Address bus	Out	Output the high-order 8 bits of the memory address or the 8 bits of the I/O address. It remains in the high-impedance state during the HOLD and HALT modes.
S_0, S_1	Status	Out	Indicates the status of the bus. HALT: S_1 0, S_0 0 WRITE: 0, 1 READ, DAD: 1, 0 FETCH: 1, 1 The S_1 signal can be used as an advanced R/W status.
ALE	Address latch enable	Out	This signal is generated during the first clock cycle, to enable the address to be latched into the latches of peripherals. The falling edge of ALE is guaranteed to latch the address information. The ALE can also be used to strobe the status information, but it is kept in the low-level state during bus idle machine cycles.
$\overline{WR}$	Write control	Out	Indicates that the data on the data bus is to be written into the selected memory at the falling edge of the signal WR. It remains the high-impedance state during the HOLD and HALT modes.
$\overline{RD}$	Read control	Out	Indicates that the selected memory or I/O address is to be read and that the data bus is active for data transfer. It remains in the high-impedance state during the HOLD and HALT modes.
IO/$\overline{M}$	Data transfer control output	Out	This signal indicates whether the read/write is to memory or to I/Os. It remains in the high-ipedance state during the HOLD and HALT modes.
READY	Ready input	In	When it is at high-level during a read or write cycle, the READY indicates that the memory or peripheral is ready to send or receive date. When the signal is at low-level, the CPU will wait for the signal to turn high-level before completing the read or write cycle.
$\overline{RESET\ IN}$	Reset input	In	This signal (at least three clock cycles are necessary) sets the program counter to zero and resets the interrupt enable and HLDA flip-flops. None of the other flags or registers (except the instruction register) are affected. The CPU is held in the reset mode as long as the signal is applied.
CLK	Clock output	Out	Clock pulses are available from this pin when a crystal or RC circuit is used as an input to the CPU.
HLDA	Hold acknowledge signal	Out	By this signal the processor acknowledges the HOLD request signal and indicates that it will relinquish the buses in the next clock cycle. The signal is returned to the low-level state after the HOLD request is completed. The processor resumes the use of the buses one half clock cycle after the signal HLDA gose low-level.
HOLD	Hold request signal	In	When the CPU receives a HOLD request, It relinquishes the use of the buses as soon as the current machine cycle is completed. The CPU can regain the use of buses only after the HOLD state is removed. Upon acknowledging the HOLD signal, the address bus, the data bus, RD, WR and IO/$\overline{M}$ lines are put in the high-impedance state.

Note : HOLD, READY and all interrupt signals are synchronous with clock signal.

STATUS INFORMATION

Status information can be obtained directly from the M5L8085AP. ALE is used as a status strobe. As the status is partially encoded, it informs the user in advance what type of bus transfer is being performed. The IO/$\overline{M}$ cycle status signal is also obtained directly. Decoded S_0 and S_1 signals carry:

	S_1	S_0	
HALT	0	0	
WRITE	0	1	
READ	1	0	(except for second and third machine cycles of DAD instruction)
FETCH	1	1	

S_1 can be used in determining the R/W status of all bus transfers.

In the M5L8085AP the low-order 8 bits of the address are multiplexed with data. When entering the low-order of the address into memory or peripheral latch circuits, the ALE is used as a strobe.

INTERRUPT AND SERIAL I/O

The M5L8085AP has five interrupt inputs—INTR, RST 5.5, RST 6.5, RST 7.5, and TRAP. The three RST inputs, 5.5, 6.5, 7.5, are provided with programmable masks. TRAP has the same function as the restart interrupt, except that it is non-maskable.

When an interrupt is enabled and the corresponding interrupt mask is not set, the three RST interrupts will cause the internal execution of the RST. When non-maskable TRAP is applied, it causes the internal execution of an RST regardless of the state of the interrupt enable or masks. The restart addresses (hexadecimal) of the interrupts are:

Interrupt	Adress
TRAP	24_{16}
RST 5.5	$2C_{16}$
RST 6.5	34_{16}
RST 7.5	$3C_{16}$

Two different types of signal are used for restart interrupts. Both RST 5.5 and RST 6.5 are sensitive to high-level as in INTR, and are acknowledged in the same timing as INTR. RST 7.5 is sensitive to rising-edge, and existence of a pulse sets the RST 7.5 interrupt request. This condition will be maintained until the request is fulfilled or reset by a SIM or RESET instruction.

Each of the restart interrupts may be masked independently to avoid interrupting the CPU. An interrupt requested by an RST 7.5 will be stored even when its mask is set and the interrupt is disabled. Masks can be changed in a SIM instruction or the RESET. When two enabled interrupts are requested at the same time, the interrupt with the highest priority will be accepted. The TRAP has the highest priority followed in order by RST 7.5, RST 6.5, RST 5.5 and INTR. This priority system dose not take into consideration the priority of an interrupt routine that is already started. In other words, when an RST 5.5 interrupt is reenabled before the termination of the RST 7.5 interrupt routine, it will interrupt the RST 7.5.

The TRAP interrupt is very useful in preventing disastrous errors and bus errors resulting from power failures. The TRAP input is recognized in the same manner as any other interrupt, but it has the highest priority, and is not affected by any flags or masks. The TRAP input can be sensed by both edge and level. TRAP should be maintained high-level until it is acknowledged. But, it will not be acknowledged again unless it turns low-level and high-level again. In this manner, faulty operation due to noise or logic glitches is prevented.

The selial I/O system is also considered to be an interrupt as it is controlled by instructions RIM and SIM. The SID is read by instruction RIM and the SOD data is set by instruction SIM.

BASIC TIMING

The M5L8085AP is provided with a multiplexed data bus. The ALE is utilized as a strobe with which the low-order 8 bits of the address on the data bus are sampled. Fig.1 shows the basic cycle in which an out instruction is fetched, and memory is read and written to the I/O port. The I/O port address is stored in both the address bus and the address/data bus during the I/O write and read cycle. To enable the M5L8085AP to be used with a slow memory, the READY line is used for extending the read and write pulse width.

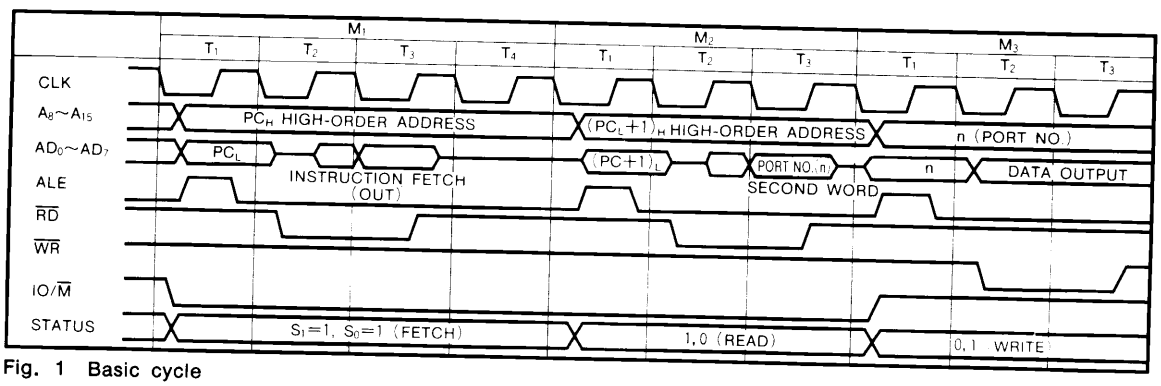


Fig. 1 Basic cycle

MACHINE INSTRUCTIONS

Instr class	Mnemonic	Instruction code D7 D6	D5 D4 D3	D2 D1 D0	16-mal notation	No. of states	No. of bytes	No. of cycles	Functions	Flags S Z P CY AC CY	Address bus Contents	Address bus Mach. cycle*	Data bus Contents	Data bus I/O	Data bus Mach. cycle*
Data transfer	MOV r1, r2	0 1	D D D	S S S		4	1	1	(r1) ← (r2)	X X X X X					
	MOV M, r	0 1	1 1 0	S S S		7	1	2	((H)(L)) ← (r)	X X X X X	M	M4	(r)	O	M4
	MOV r, M	0 1	D D D	1 1 0		7	1	2	(r) ← ((H)(L))	X X X X X	M	M4	((H)(L))	I	M4
	MVI r, n	0 0	D D D	1 1 0		7	2	2	(r) ← n	X X X X X			(B2)	I	M4
			(B2)												
	MVI M, n	0 0	1 1 0	1 1 0	3 6	10	2	3	((H)(L)) ← n	X X X X X	M	M5	(B2)	I	M5
			(B2)												
	LXI B, m	0 0	0 0 0	0 0 1	0 1	10	3	3	(C) ← (B2)	X X X X X			(B2)	I	M2
			(B2)						(B) ← (B3)				(B3)	I	M3
			(B3)												
	LXI D, m	0 0	0 1 0	0 0 1	1 1	10	3	3	(E) ← (B2)	X X X X X			(B2)	I	M2
			(B2)						(D) ← (B3)				(B3)	I	M3
			(B3)												
	LXI H, m	0 0	1 0 0	0 0 1	2 1	10	3	3	(L) ← (B2)	X X X X X			(B2)	I	M2
			(B2)						(H) ← (B3)				(B3)	I	M3
			(B3)												
	LXI SP, m	0 0	1 1 0	0 0 1	3 1	10	3	3	(SP) ← m	X X X X X			(B2)	I	M2
			(B2)										(B3)	I	M3
			(B3)												
	SPHL	1 1	1 1 1	0 0 1	F 9	6	1	1	(SP) ← ((H)(L))	X X X X X					
	STAX B	0 0	0 0 0	0 1 0	0 2	7	1	2	((B)(C)) ← (A)	X X X X X	(B)(C)	M4	(A)	O	M4
	STAX D	0 0	0 1 0	0 1 0	1 2	7	1	2	((D)(E)) ← (A)	X X X X X	(D)(E)	M4	(A)	O	M4
	LDAX B	0 0	0 0 1	0 1 0	0 A	7	1	2	(A) ← ((B)(C))	X X X X X	(B)(C)	M4	((B)(C))	I	M4
	LDAX D	0 0	0 1 1	0 1 0	1 A	7	1	2	(A) ← ((D)(E))	X X X X X	(D)(E)	M4	((D)(E))	I	M4
	STA m	0 0	1 1 0	0 1 0	3 2	13	3	4	(m) ← (A)	X X X X X	m	M4	(A)	O	M4
			(B2)												
			(B3)												
	LDA m	0 0	1 1 1	0 1 0	3 A	13	3	4	(A) ← (m)	X X X X X	m	M4	(m)	I	M4
			(B2)												
			(B3)												
	SHLD m	0 0	1 0 0	0 1 0	2 2	16	3	5							
			(B2)						(m) ← (L)	X X X X X	m	M4	(L)	O	M4
			(B3)						(m + 1) ← (H)		m + 1	M5	(H)	O	M5
	LHLD m	0 0	1 0 1	0 1 0	2 A	16	3	5							
			(B2)						(L) ← (m)	X X X X X	m	M4	(m)	I	M4
			(B3)						(H) ← (m + 1)		m + 1	M5	(m + 1)	I	M5
	XCHG	1 1	1 0 1	0 1 1	E B	4	1	1	(H)(L) ↔ (D)(E)						
	XTHL	1 1	1 0 0	0 1 1	E 3	16	1	5	(H)(L) ↔ ((SP) + 1)((SP))	X X X X X	(SP)	M2	((S))	I	M2
											(SP) + 1	M3	((SP) + 1)	I	M3
Arithmetic, logical compare	ADD r	1 0	0 0 0	S S S		4	1	1	(A) ← (A) + (r)						
	ADD M	1 0	0 0 0	1 1 0	8 6	7	1	2	(A) ← (A) + ((H)(L))	Where M (H)(L)	M	M4	(M)	I	M4
	ADI n	1 1	0 0 0	1 1 0	C 6	7	2	2	(A) ← (A) + n				(B2)	I	M4
			(B2)												
	ADC r	1 0	0 0 1	S S S		4	1	1	(A) ← (A) + (r) + (CY2)						
	ADC M	1 0	0 0 1	1 1 0	8 E	7	1	2	(A) ← (A) + ((H)(L)) + (CY2)	Where M (H)(L)	M	M4	(M)	I	M4
	ACI n	1 1	0 0 1	1 1 0	C E	7	2	2	(A) ← (A) + n + (CY2)				(B2)	I	M4
			(B2)												
	DAD B	0 0	0 0 1	0 0 1	0 9	10	1	3	((H)(L)) ← ((H)(L)) + (B)(C)						
	DAD D	0 0	0 1 1	0 0 1	1 9	10	1	3	((H)(L)) ← ((H)(L)) + (D)(E)						
	DAD H	0 0	1 0 1	0 0 1	2 9	10	1	3	((H)(L)) ← ((H)(L)) + ((H)(L))						
	DAD SP	0 0	1 1 1	0 0 1	3 9	10	1	3	((H)(L)) ← ((H)(L)) + (SP)						
	SUB r	1 0	0 1 0	S S S		4	1	1	(A) ← (A) − (r)						
	SUB M	1 0	0 1 0	1 1 0	9 6	7	1	2	(A) ← (A) − ((H)(L))	Where M (H)(L)	M	M4	(M)	I	M4
	SUI n	1 1	0 1 0	1 1 0	D 6	7	2	2	(A) ← (A) − n				(B2)	I	M4
			(B2)												
	SBB r	1 0	0 1 1	S S S		4	1	1	(A) ← (A) − (r) − (CY2)						
	SBB M	1 0	0 1 1	1 1 0	9 E	7	1	2	(A) ← (A) − ((H)(L)) − (CY2)	Where M (H)(L)	M	M4	(M)	I	M4
	SBI n	1 1	0 1 1	1 1 0	D E	7	2	2	(A) ← (A) − n − (CY2)				B2)	I	M4
			(B2)												
	ANA r	1 0	1 0 0	S S S		4	1	1	(A) ← (A) ∧ (r)						
	ANA M	1 0	1 0 0	1 1 0	A 6	7	1	2	(A) ← (A) ∧ ((H)(L))	Where M (H)(L)	M	M4	(M)	I	M4
	ANI n	1 1	1 0 0	1 1 0	E 6	7	2	2	(A) ← (A) ∧ n				(B2)	I	M4
			(B2)												
	XRA r	1 0	1 0 1	S S S		4	1	1	(A) ← (A) ∨ (r)						
	XRA M	1 0	1 0 1	1 1 0	A E	7	1	2	(A) ← (A) ∨ ((H)(L))	Where M (H)(L)	M	M4	(M)	I	M4
	XRI n	1 1	1 0 1	1 1 0	E E	7	2	2	(A) ← (A) ∨ n				(B2)	I	M4
			(B2)												
	ORA r	1 0	1 1 0	S S S		4	1	1	(A) ← (A) ∨ (r)						
	ORA M	1 0	1 1 0	1 1 0	B 6	7	1	2	(A) ← (A) ∨ ((H)(L))	Where M (H)(L)	M	M4	(M)	I	M4
	ORI n	1 1	1 1 0	1 1 0	F 6	7	2	2	(A) ← (A) ∨ n				(B2)	I	M4
			(B2)												
	CMP r	1 0	1 1 1	S S S		4	1	1	(A) − (r)						
	CMP M	1 0	1 1 1	1 1 0	B E	7	1	2	A − ((H)(L)) Compare Where M (H)(L)		M	M4	(M)	I	M4
	CPI n	1 1	1 1 1	1 1 0	F E	7	2	2	(A) − n				(B2)	I	M4
			(B2)												
Register increment decrement	INR r	0 0	D D D	1 0 0		4	1	1	(r) ← (r) + 1						
	INR M	0 0	1 1 0	1 0 0	3 4	10	1	3	((H)(L)) ← ((H)(L)) + 1	Where M (H)(L)	M	M4	(M)	I	M4
	DCR r	0 0	D D D	1 0 1		4	1	1	(r) ← (r) − 1						
	DCR M	0 0	1 1 0	1 0 1	3 5	10	1	3	((H)(L)) ← ((H)(L)) − 1	Where M (H)(L)	M	M4	(M)	I	M4
	INX B	0 0	0 0 0	0 0 1	0 3	6	1	1	(B)(C) ← (B)(C) + 1						
	INX D	0 0	0 1 0	0 1 1	1 3	6	1	1	(D)(E) ← (D)(E) + 1						
	INX H	0 0	1 0 0	0 1 1	2 3	6	1	1	(H)(L) ← (H)(L) + 1						
	INX SP	0 0	1 1 0	0 1 1	3 3	6	1	1	(SP) ← (SP) + 1						
	DCX B	0 0	0 0 1	0 1 1	0 B	6	1	1	(B)(C) ← (B)(C) − 1						
	DCX D	0 0	0 1 1	0 1 1	1 B	6	1	1	(D)(E) ← (D)(E) − 1						
	DCX H	0 0	1 0 1	0 1 1	2 B	6	1	1	(H)(L) ← (H)(L) − 1						
	DCX SP	0 0	1 1 1	0 1 1	3 B	6	1	1	(SP) ← (SP) − 1						
Rotate & shift contents of accumulator	RLC	0 0	0 0 0	1 1 1	0 7	4	1	1	Left shift (CY) ← A7, (A) ← (A) ∗	X X X X O X					
	RRC	0 0	0 0 1	1 1 1	0 F	4	1	1	Right shift (CY) ← A0, (A) ← (A) ∗	X X X X O X					
	RAL	0 0	0 1 0	1 1 1	1 7	4	1	1	Left shift (CY) ← A7, (A) ← (A) ∗	X X X X O X					
	RAR	0 0	0 1 1	1 1 1	1 F	4	1	1	Right shift (CY) ← A0, (A) ← (A) ∗	X X X X O X					
Accumulator	CMA	0 0	1 0 1	1 1 1	2 F	4	1	1	A ← $\overline{A}$						
	DAA	0 0	1 0 0	1 1 1	2 7	4	1	1	Results of binary addition are adjusted to BCD	X X X X X O					
Carry set	STC	0 0	1 1 0	1 1 1	3 7	4	1	1	(CY) ← 1	X X X 1 X					
	CMC	0 0	1 1 1	1 1 1	3 F	4	1	1	(CY) ← $\overline{(CY)}$	X X X O X					

* State is T1 ** State is T2

MITSUBISHI ELECTRIC

Instr class	Mnemonic		Instruction code D7 D6	D5 D4 D3	D2 D1 D0	16mal notatn	No. of states	No. of bytes	No. of cycles	Functions	Flags S	Z	P	CY	AC	Address bus Contents	Mach cycle*	Data bus Contents	I/O	Mach cycle**
Jump	JMP	m	1 1	0 0 0	0 1 1	C3	10	3	3	(PC)←m	x	x	x	x	x			⟨B2⟩	I	M2
				⟨B2⟩														⟨B3⟩	I	M3
				⟨B3⟩																
	PCHL		1 1	1 0 1	0 0 1	E9	6	1	1	(PC)←(H)(L)	x	x	x	x	x					
	JC	m	1 1	0 1 1	0 1 0	DA	10/7	3	3/2	(CY)=1	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	JNC	m	1 1	0 1 0	0 1 0	D2	10/7	3	3/2	(CY)=0 If condition is true (PC)←m	x	x	x	x	x	If condition is true				
				⟨B2⟩																
				⟨B3⟩																
	JZ	m	1 1	0 0 1	0 1 0	CA	10/7	3	3/2	(Z)=1	x	x	x	x	x			⟨B2⟩	I	M2
				⟨B2⟩														⟨B3⟩	I	M3
				⟨B3⟩																
	JNZ	m	1 1	0 0 0	0 1 0	C2	10/7	3	3/2	(Z)=0	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	JP	m	1 1	1 1 0	0 1 0	F2	10/7	3	3/2	(S)=0 If condition is false (PC)←(PC)+3	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	JM	m	1 1	1 1 1	0 1 0	FA	10/7	3	3/2	(S)=1	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	JPE	m	1 1	1 0 1	0 1 0	EA	10/7	3	3/2	(P)=1	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	JPO	m	1 1	1 0 0	0 1 0	E2	10/7	3	3/2	(P)=0	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
Subroutine call	CALL	m	1 1	0 0 1	1 0 1	CD	18	3	5	((SP)−1)←((SP)−2)←(PC)+3 (PC)←m (SP)←(SP)−2	x	x	x	x	x			⟨B2⟩	I	M2
				⟨B2⟩														⟨B3⟩	I	M3
				⟨B3⟩														(SP)−1	O	M4
	RST	n	1 1	A A A	1 1 1		12	1	3	((SP)−1)←((SP)−2)←(PC)+1 (PC)←n·8, (SP)←(SP)−2 Where 0≤n≤7	x	x	x	x	x	(SP)−1	M4	(PC)+3	O	M4
																(SP)−2	M5	(PC)+3	O	M5
	CC	m	1 1	0 1 1	1 0 0	DC	18/9	3	5/2	(CY)=1	x	x	x	x	x	(SP)−1	M4	(PC)+1	O	M4
				⟨B2⟩												(SP)−2	M5	(PC)+1	O	M5
				⟨B3⟩																
	CNC	m	1 1	0 1 0	1 0 0	D4	18/9	3	5/2	(CY)=0	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	CZ	m	1 1	0 0 1	1 0 0	CC	18/9	3	5/2	(Z)=1 ((SP)−1)←((SP)−2)←(PC)+3	x	x	x	x	x	If condition is true				
				⟨B2⟩																
				⟨B3⟩																
	CNZ	m	1 1	0 0 0	1 0 0	C4	18/9	3	5/2	(Z)=0 (PC)←m (SP)←(SP)−2	x	x	x	x	x	(SP)−1	M4	(PC)+3	O	M4
				⟨B2⟩																
				⟨B3⟩												(SP)−2	M5	(PC)+3	O	M5
	CP	m	1 1	1 1 0	1 0 0	F4	18/9	3	5/2	(S)=0	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	CM	m	1 1	1 1 1	1 0 0	FC	18/9	3	5/2	(S)=1 If condition is false (PC)←(PC)+3	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	CPE	m	1 1	1 0 1	1 0 0	EC	18/9	3	5/2	(P)=1	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
	CPO	m	1 1	1 0 0	1 0 0	E4	18/9	3	5/2	(P)=0	x	x	x	x	x					
				⟨B2⟩																
				⟨B3⟩																
Return	RET		1 1	0 0 1	0 0 1	C9	10	1	3	(PC)←((SP)+1)((SP)) (SP)←(SP)+2	x	x	x	x	x	(SP)	M4	((SP+1))	I	M4
																(SP)+1	M5	((SP)+1)	I	M5
	RC		1 1	0 1 1	0 0 0	D8	12/6	1	3	(CY)=1 If condition is true	x	x	x	x	x	If condition is true				
	RNC		1 1	0 1 0	0 0 0	D0	12/6	1	3	(CY)=0	x	x	x	x	x					
	RZ		1 1	0 0 1	0 0 0	C8	12/6	1	3	(Z)=1 (PC)←((SP)+1)((SP))	x	x	x	x	x	(SP)	M4	((SP+1))	I	M4
	RNZ		1 1	0 0 0	0 0 0	C0	12/6	1	3	(Z)=0 (SP)←(SP)+2	x	x	x	x	x	(SP)+1	M5	((SP)+1)	I	M5
	RP		1 1	1 1 0	0 0 0	F0	12/6	1	3	(S)=0	x	x	x	x	x					
	RM		1 1	1 1 1	0 0 0	F8	12/6	1	3	(S)=1 If condition is false	x	x	x	x	x					
	RPE		1 1	1 0 1	0 0 0	E8	12/6	1	3	(P)=1 (PC)←(PC)+1	x	x	x	x	x					
	RPO		1 1	1 0 0	0 0 0	E0	12/6	1	3	(P)=0	x	x	x	x	x					
Input/ output control	IN	n	1 1	0 1 1	0 1 1	DB	10	2	3	(A)←(Input buffer)←(Input device of number n) (Input data)	x	x	x	x	x			⟨B2⟩	O	M4
				⟨B2⟩												⟨B2⟩ ⟨B2⟩	M5	Input data	I	M5
	OUT	n	1 1	0 1 0	0 1 1	D3	10	2	3	(Output device of number n)←(A)	x	x	x	x	x			⟨B2⟩	O	M4
				⟨B2⟩												⟨B2⟩ ⟨B2⟩	M5	(A)	O	M5
Interrupt control	EI		1 1	1 1 1	0 1 1	FB	4	1	1	(INTE)←1	x	x	x	x	x					
	DI		1 1	1 1 0	0 1 1	F3	4	1	1	(INTE)←0	x	x	x	x	x					
Stack control	PUSH	PSW	1 1	1 1 0	1 0 1	F5	12	1	3	((SP)−1)←(A) ((SP)−2)←(F) (SP)←(SP)−2	x	x	x	x	x	(SP)−1	M4	(A)	O	M4
																(SP)−2	M5	(F)	O	M5
	PUSH	B	1 1	0 0 0	1 0 1	C5	12	1	3	((SP)−1)←(B) ((SP)−2)←(C) (SP)←(SP)−2	x	x	x	x	x	(SP)−1	M4	(B)	O	M4
																(SP)−2	M5	(C)	O	M5
	PUSH	D	1 1	0 1 0	1 0 1	D5	12	1	3	((SP)−1)←(D) ((SP)−2)←(E) (SP)←(SP)−2	x	x	x	x	x	(SP)−1	M4	(D)	O	M4
																(SP)−2	M5	(E)	O	M5
	PUSH	H	1 1	1 0 0	1 0 1	E5	12	1	3	((SP)−1)←(H) ((SP)−2)←(L) (SP)←(SP)−2	x	x	x	x	x	(SP)−1	M4	(H)	O	M4
																(SP)−2	M5	(L)	O	M5
	POP	PSW	1 1	1 1 0	0 0 1	F1	10	1	3	(F)←((SP)) (A)←((SP)+1) (SP)←(SP)+2	○	○	○	○	○	(SP)	M4	((SP))	I	M4
																(SP)+1	M5	((SP)+1)	I	M5
	POP	B	1 1	0 0 0	0 0 1	C1	10	1	3	(C)←((SP)) (B)←((SP)+1) (SP)←(SP)+2	x	x	x	x	x	(SP)	M4	((SP))	I	M4
																(SP)+1	M5	((SP)+1)	I	M5
	POP	D	1 1	0 1 0	0 0 1	D1	10	1	3	(E)←((SP)) (D)←((SP)+1) (SP)←(SP)+2	x	x	x	x	x	(SP)	M4	((SP))	I	M4
																(SP)+1	M5	((SP)+1)	I	M5
	POP	H	1 1	1 0 0	0 0 1	E1	10	1	3	(L)←((SP)) (H)←((SP)+1) (SP)←(SP)+2	x	x	x	x	x	(SP)	M4	((SP))	I	M4
																(SP)+1	M5	((SP)+1)	I	M5
Others	HLT		0 1	1 1 0	1 1 0	76	5	1	1	(PC)←(PC)+1	x	x	x	x	x					
	NOP		0 0	0 0 0	0 0 0	00	4	1	1	(PC)←(PC)+1	x	x	x	x	x					
Mask set instructions	RIM		0 0	1 0 0	0 0 0	20	4	1	1	All RST interrupt masks, any pending RST interrupt requests, and the serial input data from the SID pin are read into the accumulator	x	x	x	x	x					
	SIM		0 0	1 1 0	0 0 0	30	4	1	1	Mask is enabled (or disabled) to the RST interrupt corresponding to the contents (bit pattern) of the accumulator. The serial output is enabled and the serial output bit is loaded into the SOD latch	x	x	x	x	x					

*: State is T_1 **: State is T_2

MACHINE INSTRUCTIONS SYMBOL MEANING

Symbol	Meaning
r	Register
m	Two byte data
n	One byte data
⟨B2⟩	Second byte of instruction
⟨B3⟩	Third byte of instruction
AAA	Binary representation for RST instruction n
F	8-bit data from the most to the least significant bit S Z X CY1 0 P X CY2 X is indefinite
PC	Program counter
SP	Stack pointer

Symbol	Meaning
SSS or DDD	Bit pattern designating register or memory

Register or memory	SSS or DDD
B	000
C	001
D	010
E	011
H	100
L	101
M	110
A	111

Where M (H)(L)

Symbol	Meaning
←	Data is transferred in direction shown
( )	Contents of register or memory location
V	Inclusive OR
⊻	Exclusive OR
∧	Logical AND
‾	1's complement
X	Content of flag is not changed after execution
O	Content of flag is set or reset after execution
I	Input mode
O	Output mode

INSTRUCTION CODE LIST

$D_3 \sim D_0$ \ $D_7 \sim D_4$	Hexadecimal notation	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NOP	(—)	RIM	SIM	MOV B, B	MOV D, B	MOV H, B	MOV M, B	ADD B	SUB B	ANA B	ORA B	RNZ	RNC	RPO	RP
0001	1	LXI B	LXI D	LXI H	LXI SP	MOV B, C	MOV D, C	MOV H, C	MOV M, C	ADD C	SUB C	ANA C	ORA C	POP B	POP D	POP H	POP PSW
0010	2	STAX B	STAX D	SHLD	STA	MOV B, D	MOV D, D	MOV H, D	MOV M, D	ADD D	SUB D	ANA D	ORA D	JNZ	JNC	JPO	JP
0011	3	INX B	INX D	INX H	INX SP	MOV B, E	MOV D, E	MOV H, E	MOV M, E	ADD E	SUB E	ANA E	ORA E	JMP	OUT	XTHL	DI
0100	4	INR B	INR D	INR H	INR M	MOV B, H	MOV D, H	MOV H, H	MOV M, H	ADD H	SUB H	ANA H	ORA H	CNZ	CNC	CPO	CP
0101	5	DCR B	DCR D	DCR H	DCR M	MOV B, L	MOV D, L	MOV H, L	MOV M, L	ADD L	SUB L	ANA L	ORA L	PUSH B	PUSH D	PUSH H	PUSH PSW
0110	6	MVI B	MVI D	MVI H	MVI M	MOV B, M	MOV D, M	MOV H, M	HLT	ADD M	SUB M	ANA M	ORA M	ADI	SUI	ANI	ORI
0111	7	RLC	RAL	DAA	STC	MOV B, A	MOV D, A	MOV H, A	MOV M, A	ADD A	SUB A	ANA A	ORA A	RST 0	RST 2	RST 4	RST 6
1000	8	(—)	(—)	(—)	(—)	MOV C, B	MOV E, B	MOV L, B	MOV A, B	ADC B	SBB B	XRA B	CMP B	RZ	RC	RPE	RM
1001	9	DAD B	DAD D	DAD H	DAD SP	MOV C, C	MOV E, C	MOV L, C	MOV A, C	ADC C	SBB C	XRA C	CMP C	RET	(—)	PCHL	SPHL
1010	A	LDAX B	LDAX D	LHLD	LDA	MOV C, D	MOV E, D	MOV L, D	MOV A, D	ADC D	SBB D	XRA D	CMP D	JZ	JC	JPE	JM
1011	B	DCX B	DCX D	DCX H	DCX SP	MOV C, E	MOV E, E	MOV L, E	MOV A, E	ADC E	SBB E	XRA E	CMP E	(—)	IN	XCHG	EI
1100	C	INR C	INR E	INR L	INR A	MOV C, H	MOV E, H	MOV L, H	MOV A, H	ADC H	SBB H	XRA H	CMP H	CZ	CC	CPE	CM
1101	D	DCR C	DCR E	DCR L	DCR A	MOV C, L	MOV E, L	MOV L, L	MOV A, L	ADC L	SBB L	XRA L	CMP L	CALL	(—)	(—)	(—)
1110	E	MVI C	MVI E	MVI L	MVI A	MOV C, M	MOV E, M	MOV L, M	MOV A, M	ADC M	SBB M	XRA M	CMP M	ACI	SBI	XRI	CPI
1111	F	RRC	RAR	CMA	CMC	MOV C, A	MOV E, A	MOV L, A	MOV A, A	ADC A	SBB A	XRA A	CMP A	RST 1	RST 3	RST 5	RST 7

This list shows the machine codes and corresponding machine instruction. $D_3 \sim D_0$ indicate the low-order 4 bits of the machine code and $D_7 \sim D_4$ indicate the high-order 4 bits. Hexadecimal numbers are also used to indicate this code.

The instruction may consists of 1 , 2 , or 3 bytes, but only the first byte is listed.

- ☐ indicates a 3-byte instruction.
- ▨ indicates a 2-byte instruction.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5～7	V
V_I	Input voltage	With respect to V_{SS}	−0.5～7	V
Pd	Power dissipation	T_a=25℃	1.5	W
Topr	Operating free-air temperature range		−20～75	℃
Tstg	Storge temperature range		−65～150	℃

RECOMMENDED OPERATING CONDITIONS (T_a=−20～75℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage (GND)		0		V

ELECTRICAL CHARACTERISTICS (T_a=−20～75℃, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage (Except for X_1, X_2)		2.2		V_{CC}+0.5	V
V_{IL}	Low-level input voltage		−0.5		0.6	V
$V_{IH\ RESIN}$	High-level reset input voltage		2.4		V_{CC}+0.5	V
$V_{IL\ RESIN}$	Low-level reset input voltage		−0.5		0.8	V
V_{IHX}	X_1, X_2 High-level voltage		4.0		V_{CC}+0.5	V
V_{OH}	High-level output voltage	I_{OH}=−400μA	2.4			V
V_{OL}	Low-level output voltage	I_{OL}=2mA			0.45	V
I_{CC}	Supply current from V_{CC}	(Note 2)			200	mA
I_I	Input leak current, except RESET IN (Note 1)	V_I=V_{CC}	−10		10	μA
I_{OZL}	Output floating leak current	V_O=0.45V～V_{CC}	−10		10	μA
V_{IH}−V_{IL}	Hysterisis RESET IN input		0.25			V

Note 1 : The input RESET IN is pulled up to V_{CC} with the resistor 3kΩ (typ) when $V_I \leqq V_{IH\ RESIN}$
2 : Maximum I_{CC} is 170mA at T_a = 0～70℃

TIMING REQUIREMENTS (T_a=−20～75℃, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

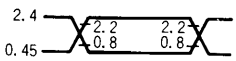
Symbol	Paramater	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{C(CLK)}$	Clock cycle time		320		2000	ns
$t_{su(DA-AD)}$	DA input setup time		−575			ns
$t_{su(DA-\overline{RD})}$	DA input setup time		−300			ns
$t_{h(DA-\overline{RD})}$	DA input hold time		0			ns
$t_{su(RDY-AD)}$	READY input setup time		−220			ns
$t_{su(RDY-CLK)}$	READY input setup time	$t_{C(CLK)} \geqq 320$ns			−110	ns
$t_{h(RDY-CLK)}$	READY input hold time	C_L=150pF	0			ns
$t_{su(DA-ALE)}$	DA input setup time		−460			ns
$t_{su(HLD-CLK)}$	HOLD input setup time		170			ns
$t_{h(HLD-CLK)}$	HLD input hold time		0			ns
$t_{su(INT-CLK)}$	Interrupt setup time		160			ns
$t_{h(INT-CLK)}$	Interrupt hold time		0			ns
$t_{su(RDY-ALE)}$	READY input setup time		−110			ns

SWITCHING CHARACTERISTICS (T_a=−20～75℃, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

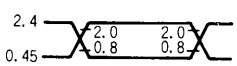
Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{w(\overline{CLK})}$	CLK output low-level pulse width			80			ns
$t_{w(CLK)}$	CLK output high-level puls width			120			ns
$t_{r(CLK)}$	CLK output rise time					30	ns
$t_{f(CLK)}$	CLK output fall time					30	ns
$t_{d(X1-CLK)}$	Delay time, X_1 to CLK			30		120	ns
$t_{d(X1-\overline{CLK})}$	Delay time, X_1 to $\overline{CLK}$			30		150	ns
$t_{d(AD-ALE)}$	Delay time, address output to ALE signal	AD_0～AD_7		90			ns
		A_8～A_{15}		115			
$t_{d(ALE-AD)}$	Delay time, ALE signal to address output			100			ns
$t_{w(ALE)}$	ALE pulse width			140			ns
$t_{d(ALE-CLK)}$	Delay time, ALE to CLK			100			ns
$t_{d(ALE-\overline{CONT})}$	Delay time, ALE to control signal			130			ns
$t_{DXZ(\overline{RD}-AD)}$	Address disable time from read		$t_{C(CLK)}$≧320ns			0	ns
$t_{DZX(\overline{RD}-AD)}$	Address enable time from read		C_L=150pF	150			ns
$t_{d(\overline{CONT}-AD)}$	Address valid time after control signal			120			ns
$t_{d(DA-\overline{WR})}$	Delay time, data output to $\overline{WR}$ signal			420			ns
$t_{d(\overline{WR}-DA)}$	Delay time, $\overline{WR}$ signal to data output			100			ns
$t_{w(\overline{CONT})}$	Control signal pulse width			400			ns
$t_{d(CONT-ALE)}$	Delay time, CLK to ALE signal			50			ns
$t_{d(CLK-HLDA)}$	Delay time, CLK to HLDA signal			110			ns
$t_{DXZ(HLDA-BUS)}$	Bus disable time from HLDA					210	ns
$t_{DZX(HLDA-BUS)}$	Control signal disable time					210	ns
$t_{d(CONT-\overline{CONT})}$	Control signal disable time			400			ns
$t_{d(AD-\overline{CONT})}$	Delay time, address output to control signal	AD_0～AD_7		240			ns
		A_8～A_{15}		270			
$t_{d(ALE-DA)}$	Delay time, ALE to data output					200	ns
$t_{d(WRHL-DA)}$	Delay time, $\overline{WR}$ signal to data output					40	ns

Note 3 : at A_8～A_{15}, and IO/$\overline{M}$ $t_{d(AD-\overline{CONT})}$ after the release of the high-impedance state is 200ns
4 : $t_{w(\overline{CLK})}$, $t_{w(CLK)}$ are 100ns(Min), 150ns(Min) respectively when 50pF+1TTL loaded
5 : A. C Testing waveform

Input pulse level	0.45～2.4V
Input pulse rise time	20ns
Input pulse fall time	20ns
Reference level input	V_{IH}=2.2V, V_{IL}=0.8V
output	V_{OH}=2.0V, V_{OL}=0.8V



Input



Output

Parameters described in the timing requirements and switching characteristics take relevant values in accordance with the relational expression shown in the following tables when the frequency is varied.

Relational Expression with the frequency T ($t_{C(CLK)}$) in the M5L8085AP

TIMMING REQUIREMENTS (T_a=−20~75℃, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Relational expression (Note 6)	Limit
$t_{SU(DA-AD)}$	DA input setup time	C_L=150pF	225−(5/2+N)T	Min
$t_{SU(DA-\overline{RD})}$	DA input setup time		180−(3/2+N)T	Min
$t_{SU(RDY-AD)}$	READY input setup time		260−(3/2)T	Min
$t_{SU(DA-ALE)}$	DA input setup time		180−2T	Min

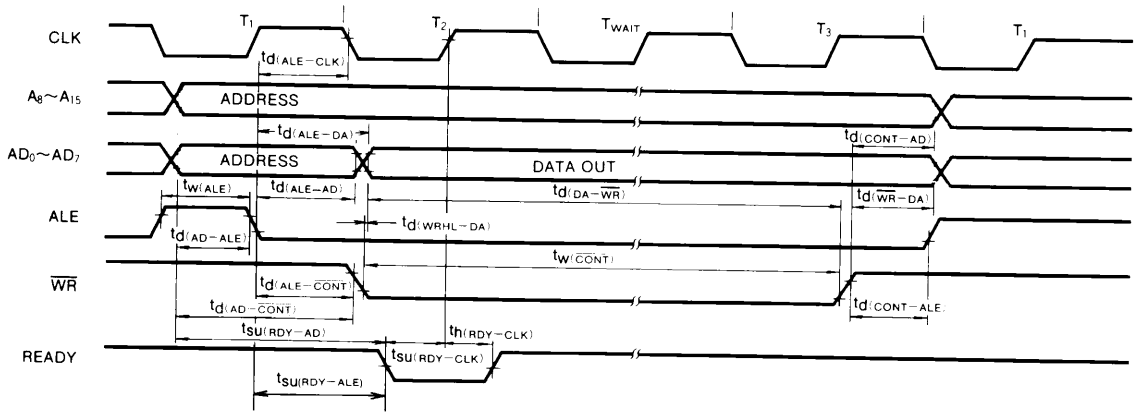
SWITCHING CHARACTERISTICS (T_a=−20~75℃, V_{CC}=5V±5%, V_{SS}=0V, unless otherwise notes)

Symbol	Palameter		Test conditions	Relational expression (Note 6)	Limit
$t_{W(\overline{CLK})}$	CLK output low-level pulse width		C_L=150pF	(1/2)T−80	Min
$t_{W(CLK)}$	CLK output high-level pulse width			(1/2)T−40	Min
$t_{d(AD-ALE)}$	Delay time, address output to ALE signal	AD_0~AD_7		(1/2)T−70	Min
		A_8~A_{15}		(1/2)T−45	
$t_{d(ALE-AD)}$	Delay time, ALE signal to address output			(1/2)T−60	Min
$t_{W(ALE)}$	ALE pulse width			(1/2)T−20	Min
$t_{d(ALE-CLK)}$	Delay time, ALE to CLK			(1/2)T−60	Min
$t_{d(ALE-\overline{CONT})}$	Delay time, ALE to control signal			(1/2)T−30	Min
$t_{DZX(\overline{RD}-AD)}$	Address enable time from read			(1/2)T−10	Min
$t_{d(CONT-AD)}$	Address valid time after control signal			(1/2)T−40	Min
$t_{d(DA-\overline{WR})}$	Delay time, data output to $\overline{WR}$ signal			(3/2+N)T−60	Min
$t_{d(\overline{WR}-DA)}$	Delay time $\overline{WR}$ signal to data output			(1/2)T−60	Min
$t_{W(\overline{CONT})}$	Control signal pulse width			(3/2+N)T−80	Min
$t_{d(CONT-ALE)}$	Delay time, CONT to ALE signal			(1/2)T−110	Min
$t_{d(CLK-HLDA)}$	Delay time, CLK to HLDA signal			(1/2)T−50	Min
$t_{DXZ(HLDA-BUS)}$	Bus disable time from HLDA			(1/2)T+50	Max
$t_{DZX(HLDA-BUS)}$	Bus enable time from HLDA			(1/2)T+50	Max
$t_{d(CONT-\overline{CONT})}$	Control signal disable time			(3/2)T−80	Min
$t_{d(AD-\overline{CONT})}$	Delay time, address output to control signal	AD_0~AD_7		T−80	Min
		A_8~A_{15}		T−50	

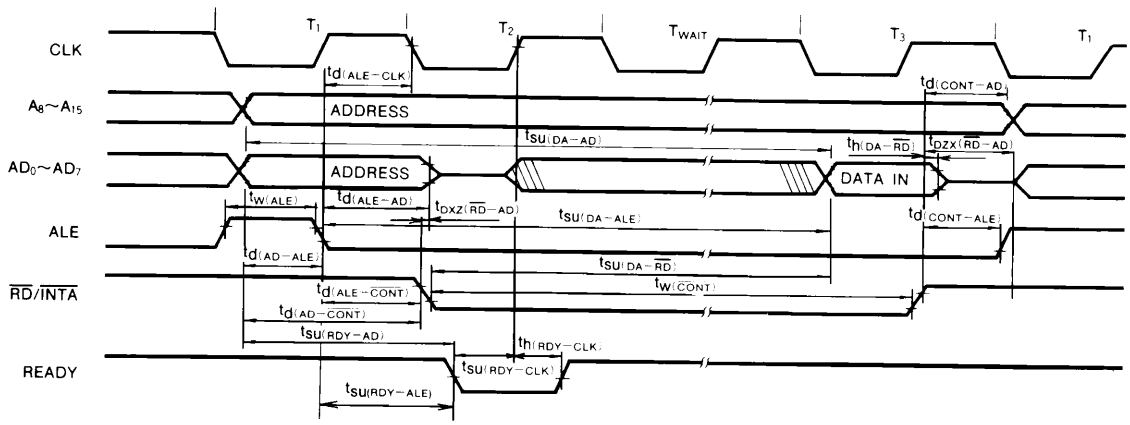
Note 6 : N indicates the total number of wait cycles.
$T=t_{C(CLK)}$

TIMING DIAGRAM

Write Cycle

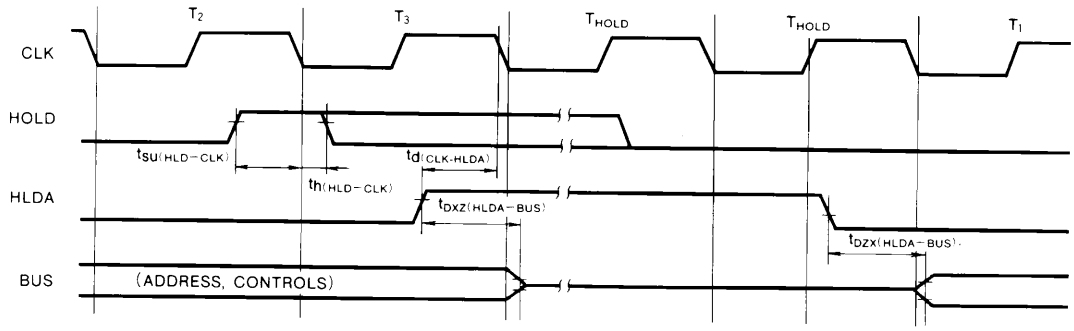


Read Cycle

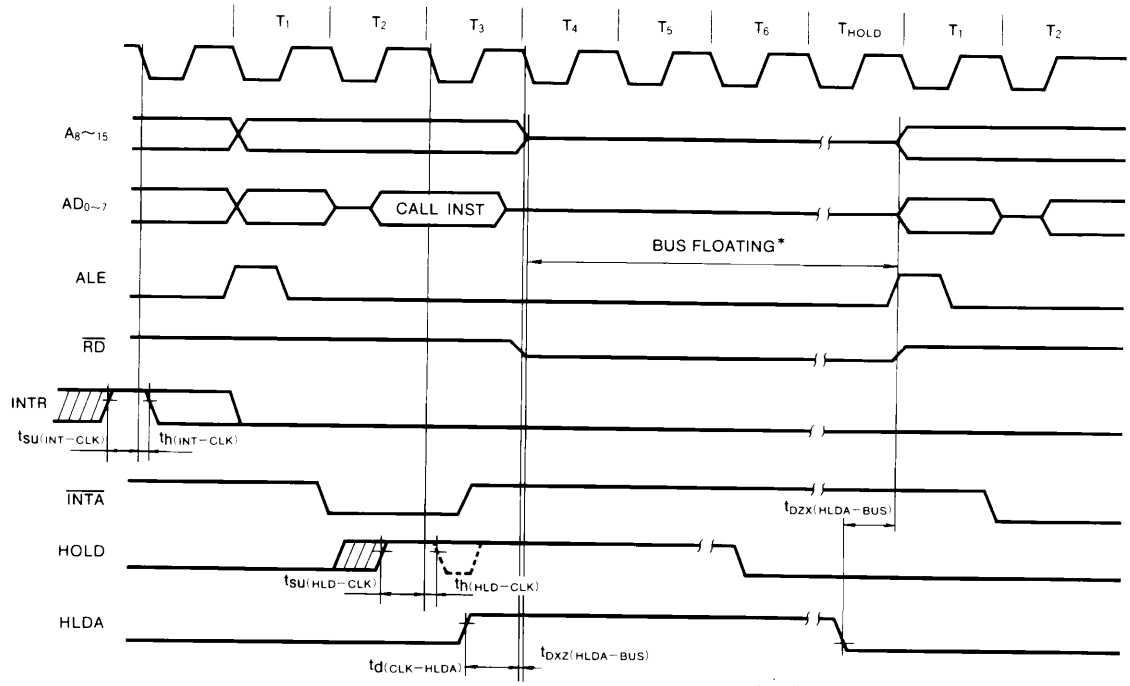


Note : READY must remain stable during setup and hold times

Hold Cycle

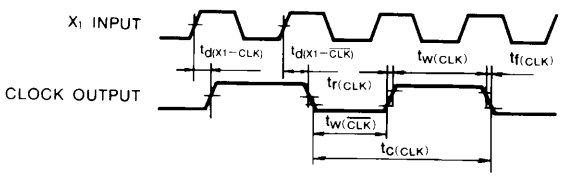


Interrupt and Hold Cycle



*IO/M is floating during this time

Clock Output Timing Waveform



TRAP INTERRUPT AND RIM INSTRUCTIONS

TRAP generates interrupts regardless of the interrupt enable filp-flop (INTE FF). The current state of the INTE FF is stored in flip flop A (A FF) of the CPU and then the INTE FF is reset. The first RIM instruction after the generation of a TRAP interrupt differs in function from the ordinary RIM instruction. That is, the bit 3 (INTE FF information) in the accumulator ($(A)_3$) after the execution of the RIM instruction contains the contents of the A FF, regardless of the state of the INTE FF at the time the RIM instruction is executed. These details are shown in Figs.2 and 3, Table 1.

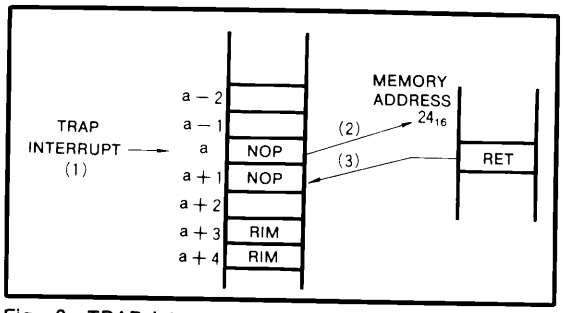


Fig. 2 TRAP interrupt processing

Below are the explanations of Fig. 2.

1. The TRAP interrupt request is issued while the instruction in address a is being executed.
2. The TRAP interrupt causes the same action as an RST instruction and then jumps to address 24_{16}.
3. It returns to address a+1 after executing the RET instruction.

Table 1 shows the information in the INTE FF at address a+3 and a+4 when the instructions EI and/or DI are executed at addresses a−1 and a+2.

Fig. 4 is a flow chart of the TRAP interrupt processing routine.

Table 1 TRAP interrupt and RIM instructions

Condition \ Number	1	2	3	4	5	6
instruction in address a-1	EI	EI	EI	DI	DI	DI
Instruction in address a+2	EI	NOP	DI	EI	NOP	DI
Contents of $(A)_3$ after the execution of the RIM instruction in address a+3	1	1	1	0	0	0
State of INTE FF after the execution of the RIM instruction in address a+3	1	0	0	1	0	0
Contents of $(A)_3$ after the execution of the RIM instruction in address a+4	1	0	0	1	0	0
State of INTE FF after the execution of the RIM instruction in address a+4	1	0	0	1	0	0

Note 7 : The contents of $(A)_3$ after the excution of the RIM instruction is an information of the INTE FF. The INTE FF assumes state 1 when it is in the EI state, and 0 when it is in the DI state.

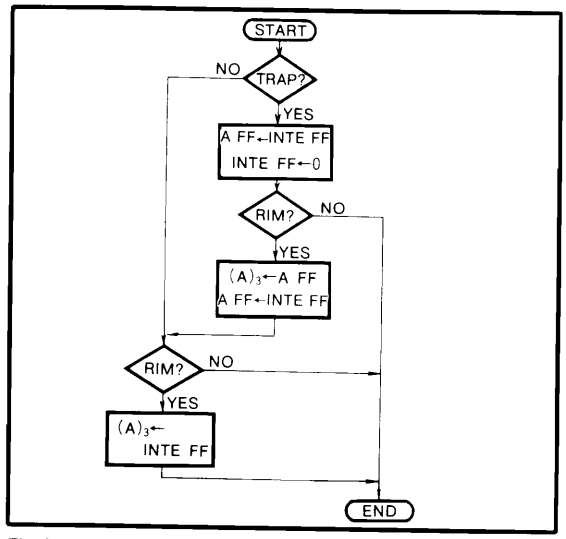


Fig.3 TRAP interrupt and INTE FF processing

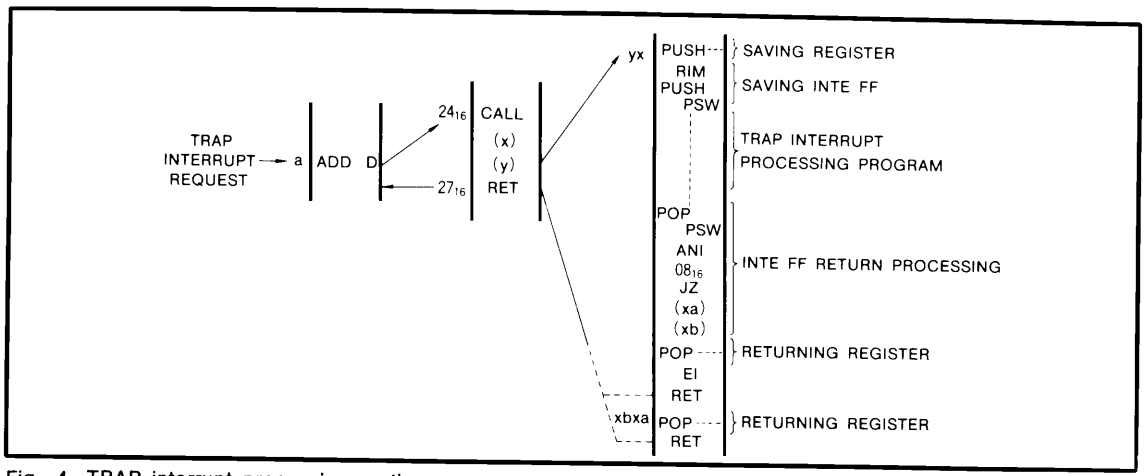


Fig. 4 TRAP interrupt processing routine

PULL-UP OF THE RESET IN INPUT

In order to increase the noise margin, the RESET IN input terminal is pulled up by about 3kΩ (typ) when the condition $V_I \geqq V_{IH(\overline{RESIN})}$ is satisfied. Fig. 5 is a connection diagram of the RESET IN input, and Fig. 6 shows the relation between input voltage and input current.

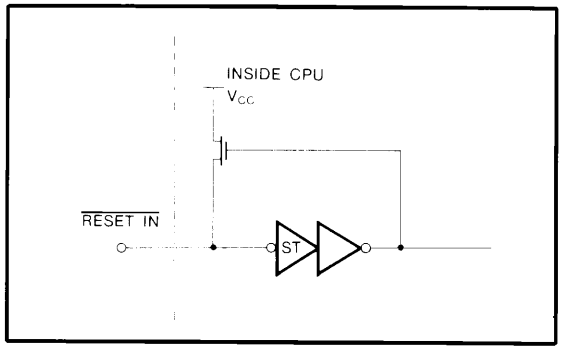


Fig. 5 Connections of RESET IN input

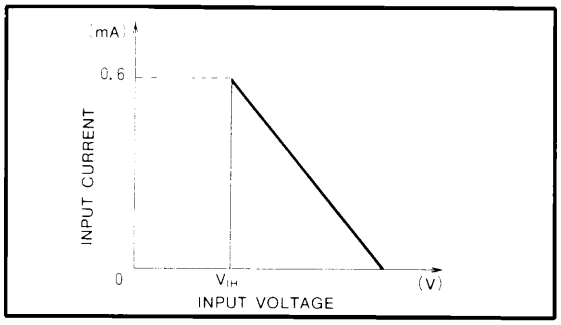


Fig. 6 RESET IN input current vs input voltage

DRIVING CIRCUIT OF X_1 AND X_2 INPUTS

Input terminals, X_1 and X_2 of the M5L8085AP can be driven by either a crystal, RC network, or external clock. Since the driver clock frequency is divided to 1/2 internally, the input frequency required is twice the actual execution frequency (6MHz for the M5L8085AP which is operated at 3MHz). Fig. 7 are typical connection diagram for a crystal circuit respectively.

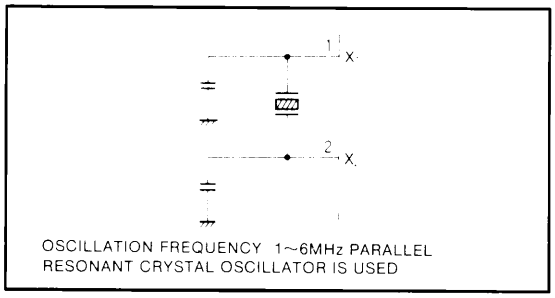


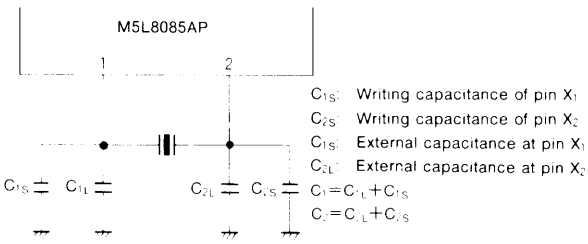
Fig. 7 Connections when crystal is used for X_1 and X_2 inputs

Conditions for Using a Quartz Crystal Element

1. Quartz Crystal Specifications

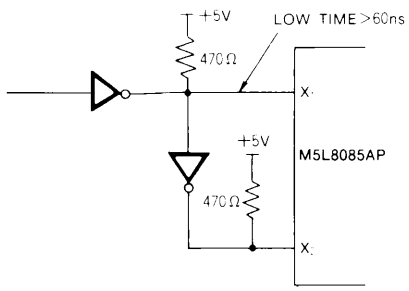
- Parallel resonance
- The frequency is 2 times the operation frequency (2 ~ 6.25MHz)
- Internal load capacitance: Approx. 16pF
- Parallel capacitance: Below 7pF
- Equivalent resistance: Below 75Ω (for operation above 4MHz)
- For operation in the range 2 ~ 4MHz, the resistance showld be made as small as possible.
- Drive capability: Above 5mW (the power at which the crystal will be destoryed)

2. External Circuitry



- For operation above 4MHz:
  $C_1=C_2=10pF$
- For operation below 4MHz:
  $C_1=C_2=15pF$

External Clock Driver Circuit



Pullup resistors are required to assure that the high level voltage of the input is at least 4V.

WAIT STATE GENERATOR

Fig. 8 shows a typical 1-wait state generator for low speed RAM and ROM applications.

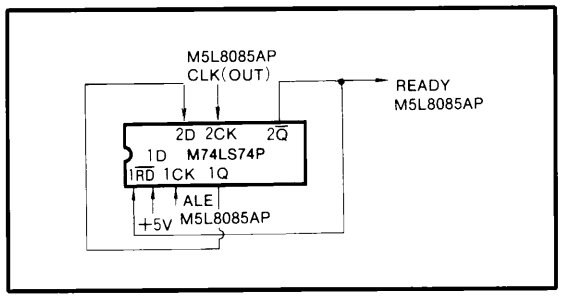


Fig. 8 1-wait state generator

Relation of Rim and Sim Instructions With The Accumulator (Supplementary Description).

The contents of the accumulator after the execution of a RIM instruction is shown in Fig. 9 .

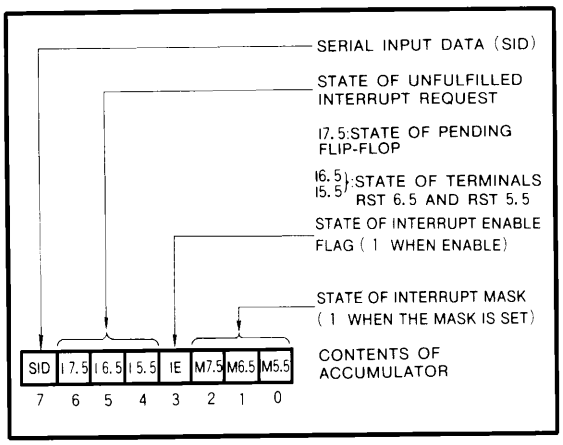


Fig. 9 Relation of the instruction RIM with the accumulator

The contents of the accumulator after the execution of a SIM instruction is shown in Fig.10.

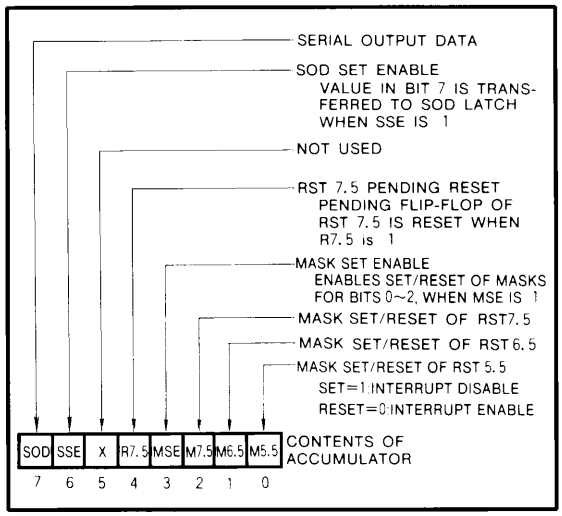


Fig. 10 Relation of the SIM instruction with the accumulator