8085 Syllabus :-

1. System bus
2. Internal architecture
   - Interrupts
3. Pin layout
4. Programming Model
5. Instruction format
6. Addressing modes
7. Timing diagram
8. Instruction set classification
9. Simple programs
10. Interfacing - I/o device
    - M/M's

	Models		Marks
I	Architecture	1 – 3	50 – 60 %
II	Programming	4 – 9	40 – 50 %
III	Interfacing	10	

Number System :-

		base
• Decimal	0 → 9	10
• octal	0 → 7	8
• binary	0, 1	2
• Hexa decimal	0 → 9 A B C D E F	16

octal → decimal

0 → 0
↓ ↓
7 7

$(10)_8 \rightarrow (8)_{10}$

$(11)_8 \rightarrow (9)_{10}$

$(12)_8 \rightarrow (10)_{10}$

binary → Decimal

0 → 0
1 → 1

$(10)_2 \rightarrow (2)_{10}$

$(11)_2 \rightarrow (3)_{10}$

$(100)_3 \rightarrow (4)_{10}$

Hexa-decimal → Decimal

0 → 0
↓ ↓
9 9

A → 10
B → 11
C → 12
D → 13
E → 14
F → 15

$10H \rightarrow (16)_{10}$

$11H \rightarrow (17)_{10}$

$12H \rightarrow (18)_{10}$

$13H \rightarrow (19)_{10}$

$14H \rightarrow (20)_{10}$

Hexa-decimal → binary $(2^3 2^2 2^1 2^0)$

0H → 0000
1H → 0001
2H → 0010
⋮
9H → 1001
A → 1010
B → 1011
C → 1100
D → 1101
E → 1110
F → 1111

Converting Decimal Value to other number system:-

Method:- Devide the decimal value with base of the required number system consider reminder from last to first.

Eg:- $(15)_{10} \rightarrow (1111)_2$

2	15	1
2	7	1
2	3	1
2	2	1
	1	

$(1111)_2$

$(20)_{10} \rightarrow (14H)_{16}$

16	20	
16	16	4
	1	

$(14H)_{16}$

16)20(1
16
4

Eg $(48)_{10} \rightarrow (30)_{16}$

16	48	
	48	3
	0	

$(30)_{16}$

eg $(1110\ 0111\ 0011\ 1000)_2 \rightarrow (E738)_{16}$

take group of 4 for base 16

eg $(001\ 110\ 011\ 100\ 11\ 1000)_2 \rightarrow (163470)_8$

1 6 3 4 7 0

take Group of 3 for base 8

Decimal

$$\begin{array}{r} 9 \\ +7 \\ \hline 16 \end{array} \qquad \begin{array}{r} 9 \\ +5 \\ \hline 14 \end{array}$$

Hexa-Deci

$$\begin{array}{r} F \\ +7 \\ \hline 16H \end{array} \qquad \begin{array}{r} F \\ +5 \\ \hline 14H \end{array}$$

Decimal

$$\begin{array}{r} 0 \\ \downarrow \\ 9 \\ \hline 10 \end{array}$$

+0 → 10

1 → 11

1

9 → 19

20

Hexadecimal

0
↓
9
↓
A
↓
F

10H
+ o → 10H
1 → 11H
⋮
9 → 19H
A → 1AH
⋮
F → 1FH
20H

x
0
↓
F

10H
+x
1xH

eg (FF)H
+(19)H
(108)H

eg (FFFF)H
(9365)H
(19364)H

eg (0EDE)H
+(5869)H
(3747)H

eg (A7C3)H
+(5B21)H
(102E4)H

Finding 1's Compliment of a binary Value:-

Suffle the bits from 0 to 1 or 1 to 0

eg 11001010
↳ 00110101

Finding 2's compliment:- Add 1 to the least Significant Bits(LSB) of 1's Compliment.

eg $(11101101)_2$ find 2's compliment

↳ 00010010 → 1's compliment

MSb ↩ ↪ LSb

+ 1

00010011 2's compliment

$(11101101)_2 \rightarrow (EDH)_{16}$

2's Compliment = -ve Value

⇒

$(11101101)_2 — (EDH)_{16}$

00010011 - 2's Compliment

Negative

$(13H)_{16} = -(EDH)$

eg -59 H = ? A7H

↓ ↘

01011001

↓

10100110 1's compliment

+1

10100111 2's Compliment

(A7H) = -59H

eg trick = Substract First digit from 16(10H) and second digit from 15(F) other also to get -ve Value

eg (EDH

⇒ F | 10H

-E | DH

13H

eg (59H

F | 10H

-5 | 9H

A7

eg $(6BD9)H$

$\Rightarrow$

F	F	P	F	10H
-6	B	D	9	H
9	4	2	7	H

first digit 16(10H)
other 15(F)

$(-6BD9)H = (9427)H$

Subtraction :-

Manual	processor
A − B	A + (−B)

eg

$(98)H \rightarrow A$	$('98)H \rightarrow A$	
$-(26)H \rightarrow B$	$+(DA)H \rightarrow (-B)$	
$72H$	carry $72H$	
	↓ discard	

-②

~~2 6 H~~

F	10H
-2	6H
D	A

eg

$(73)H \rightarrow$	$(73)H$	
$-(19)H \rightarrow$	$(E7)H$	
$(5A)H$	$(5A)H$	
	Carry discard	

F	10H
-1	9H
E	7H

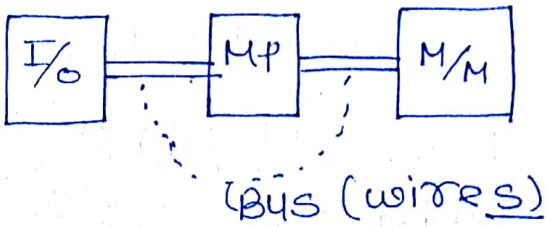
$-19H = E7H$

System bus:-

Bus is Group of wires or Conductors used for Communication between processor, Memory and I/o Devices.

There are three types of bus.

① Address bus
② Data bus
③ control bus



Address bus:- Purpose:- ① It is used to transfer the Address of either M/M or I/o from the processor.

② It defines the maximum memory that can be connected to a processor given by the relation

$$2^n = N$$

N = No. of Address/M/M location

n = No. of address line

length of Address bus : it is of 16 bits in length for 8085.

Direction :- it is unidirectional

Memory :- memory is group of Registors

→ A Registors is group of flip flop

→ A flip-flop is a memory cell which can store a Bit i.e. 0 or 1

Most of the memories (Semiconductor) are design to store or hold 8 bits per each register or memory location or address location therefore memory is represented in terms of byts.

Note ✡ The standard word length of memory is 8 bits or 1 byts

Relation b/w Address lines and Memory :

$2^1 \rightarrow$ 2 B
$2^2 \rightarrow$ 4 B
$2^3 \rightarrow$ 8 B
⋮
$2^{10} \rightarrow$ 1024 B
$\Rightarrow$ 1 KB

$2^{10} \rightarrow$ 1 kilo byte / 1 KB
$2^{20} \rightarrow$ 1 Mega byte / 1 MB
$2^{30} \rightarrow$ 1 Giga byte / 1 GB
$2^{40} \rightarrow$ 1 tera byte / 1 TB

Question:- A processor has 25 Address line find the max. memory that can be Connected.

Soln

$$N = 2^n = 2^{25} = (2^{20})\, 2^5$$
$$= 32 \text{ MB}$$

Question: A max. memory of 512 TB can be connected to a process find the address line requir

$2^n = 512\ TB$ $\quad 1TB = 2^{40}$ byte.

$2^n = 2^9 \cdot 2^{40} = 2^{49}$

Que It is require to interface 100 GB to a processor find the mini address line requir

$\Rightarrow$ 100 GB $\Rightarrow$ 64 — 128 GB

$2^{36} \rightarrow 64$ GB

$2^{37} - 128$ GB $= 2^7 \cdot 2^{30}$ B

=

For 8085

max. memory that can be connected to 8085 is

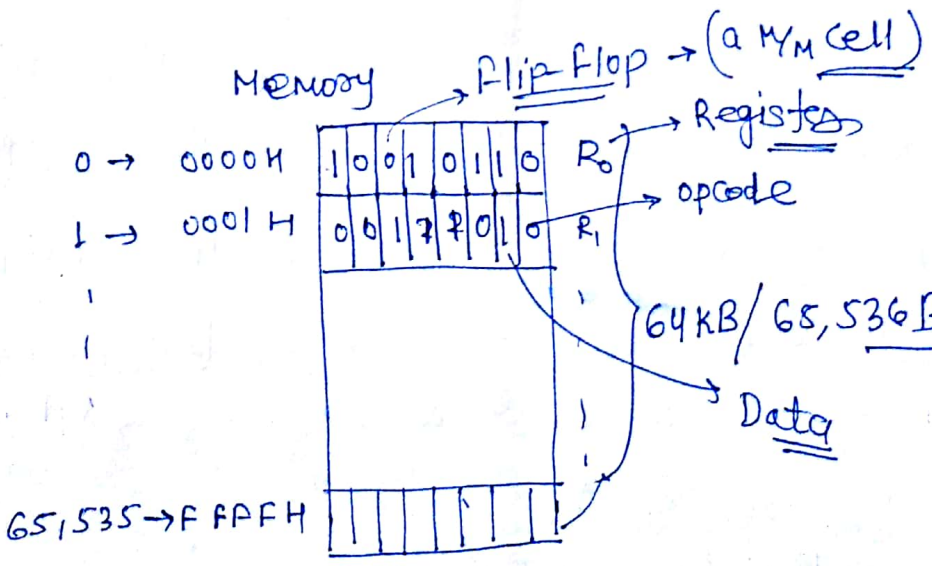
$2^{16} = 2^6(2^{10})$

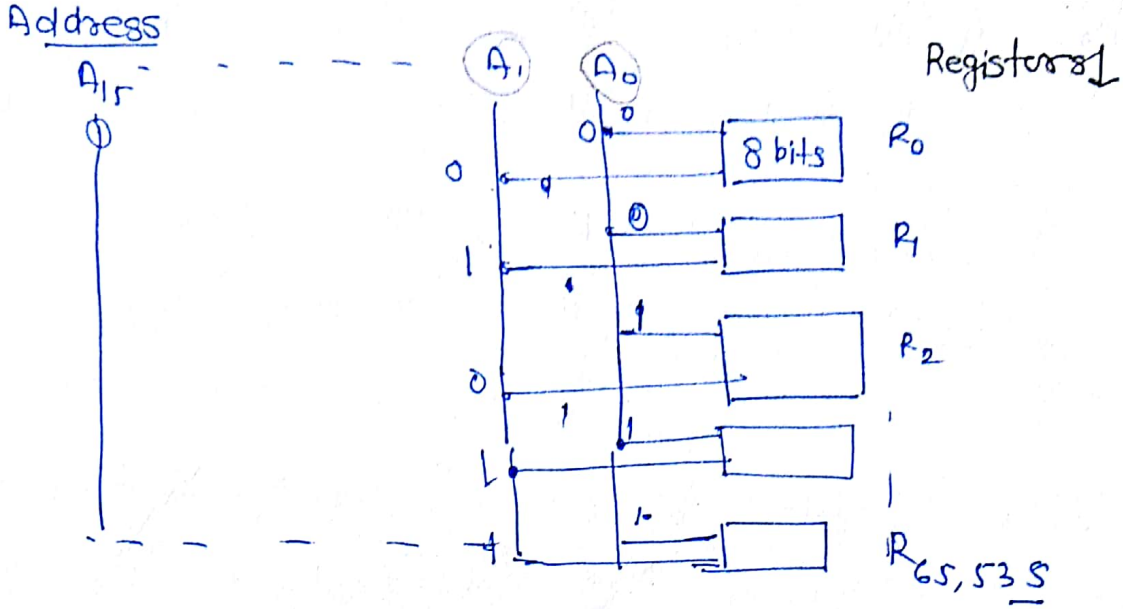
$= 2^6(1KB)$

$= 64\ KB$ ☆

$= 64 \times 1024$ B

$= 65,536$ B ☆





$A_0 - A_{15}$ $\quad 2^{16} - 65,536$

Data Bus :-

Purpose :- It is used to transfer data between memory, processor and I/o device

length :- It is of 8 bits in length for 8085

Direction :- data bus is bidirectional (sending / bring data)

Note :- There is no seprate data bus in 8085

The lower 8 address lines can be used either as address or data bus with the help of a signal know as ALE - Address latch enables. → it is a component

→ 1 : $A_{11} - A_{16}$ lines - Address bus

0 : $A_{15} - A_8$ Address bus

$AD_7 - AD_0$ Data bus

→ Multiplexed Address / data bus.

* $A_{15} - A_8$ / $AD_7 - AD_0$

Higher order Address bus → 0: Address bus	lower order Address bus Data bus

Control bus:- It is a group of different Control signals required for various operation of processor.

$\overline{RD}$ = Read Control
$\overline{WR}$ = write Control

* No direction, No length.

Note latch is used to hold lower byte of address till read / write operation Complited

Eg:- ↳ 74LS373 (Name of latch)

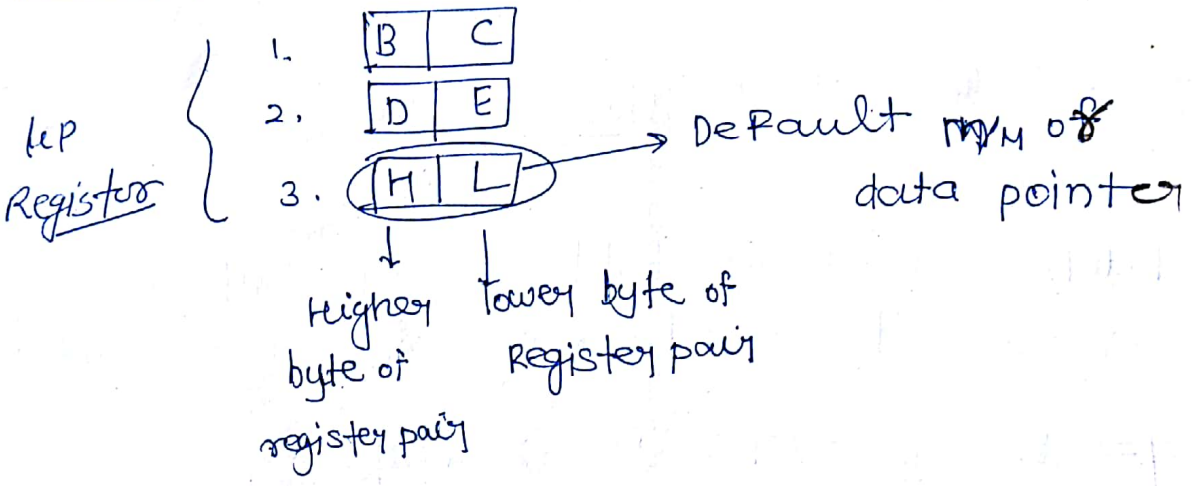
Internal Architecture:-

It is devided into five functional units

1. Register unit
2. ALU
3. timing/control unit
4. Interrupt control unit
5. serial control unit.

(1) Register Unit:- There are two type
- General purpose
- specific purpose.

General purpose :- 6, 8 bit

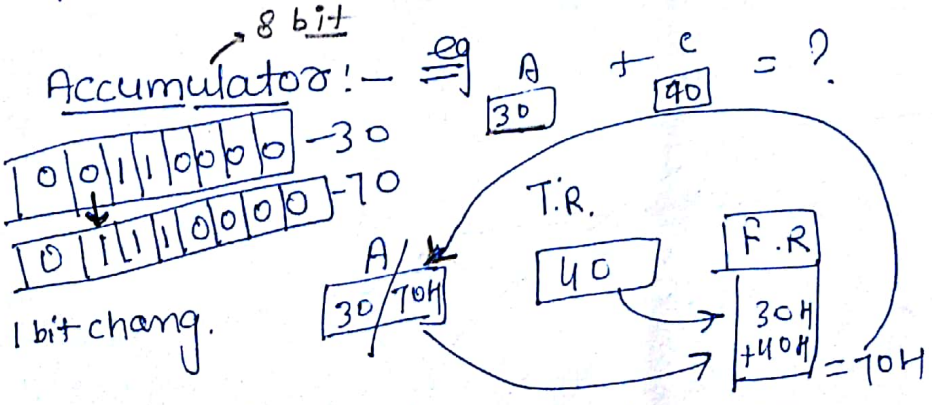


HL → Address — MOV D, M → memory content whose address present in HL only
Data — MOV L, H

Special purpose:-

8 bit

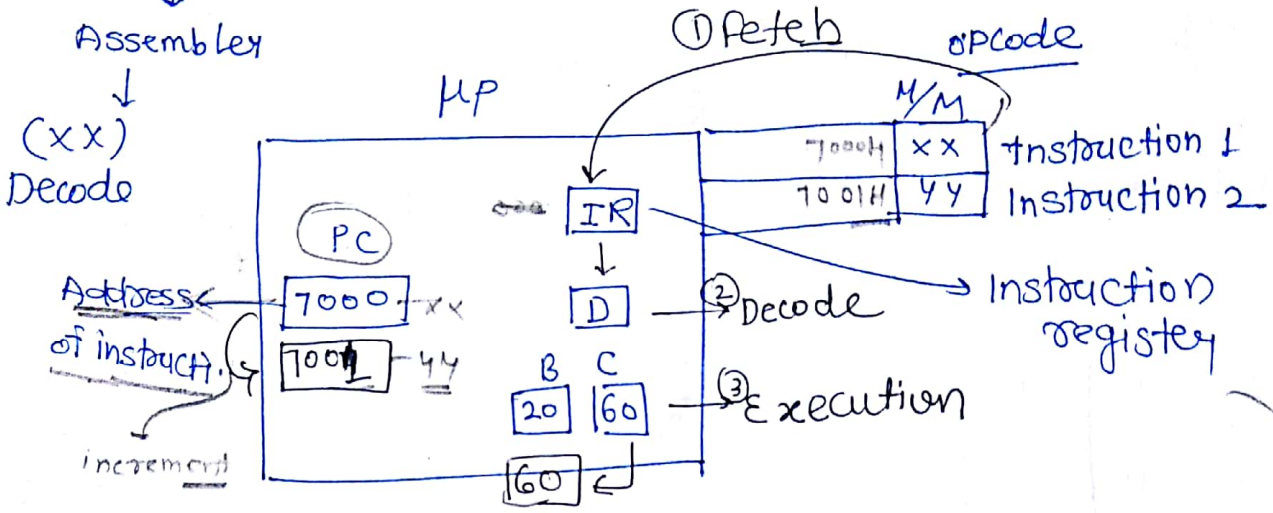
Accumulator:- eg A + C = ?



Almost ALU operation performed Result also store in ACC

Program Counter (PC) :— 16 bit Register

eg:- MOV B, C → means copy from C to B

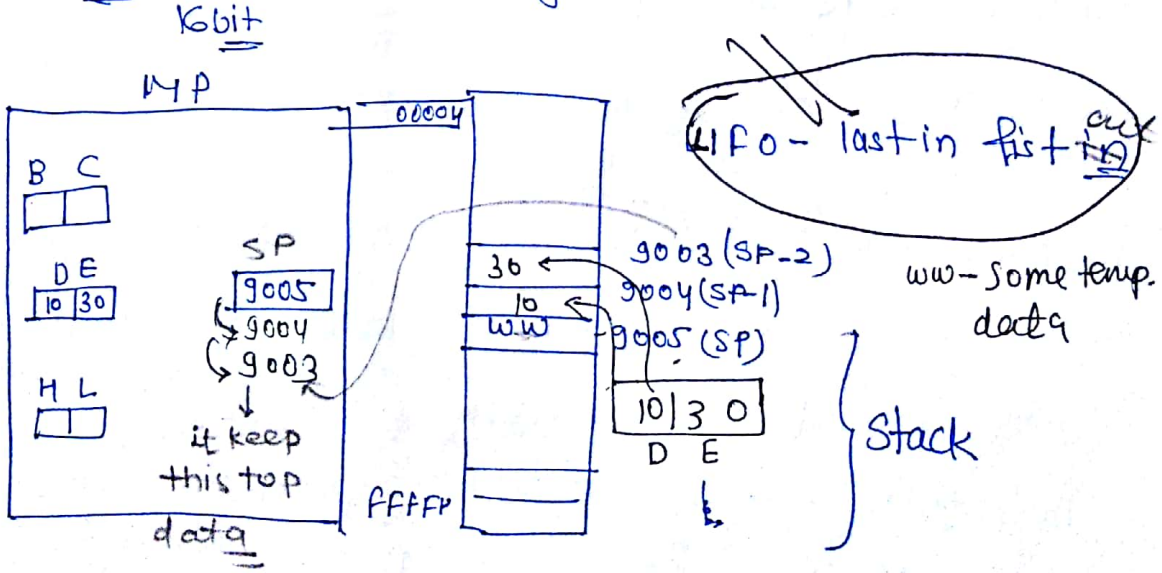


Basic steps of execution of an Instruction

① Fetch
② Decode
- - - → Read execute Address
③ Execute

Instruction Register :- 8 bit register which contains opcode of present instruction

Stack Pointer : (SP) : store temporary data.



Q SP → D009 H SP = ? final & Data @ top of stack

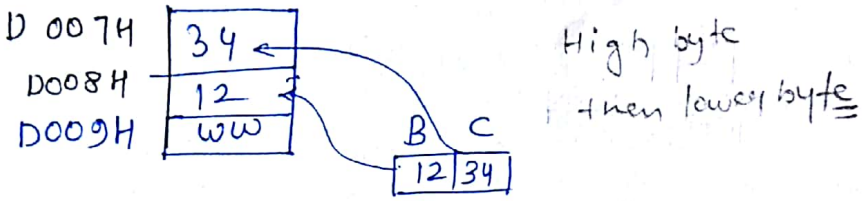
B	C
12	34

Soln

SP = D009 − 2 = D007H

Data @ top of stack = 34 → lower byte

∴



* stack pointer at High value (FFFFH) preferred to store more data.

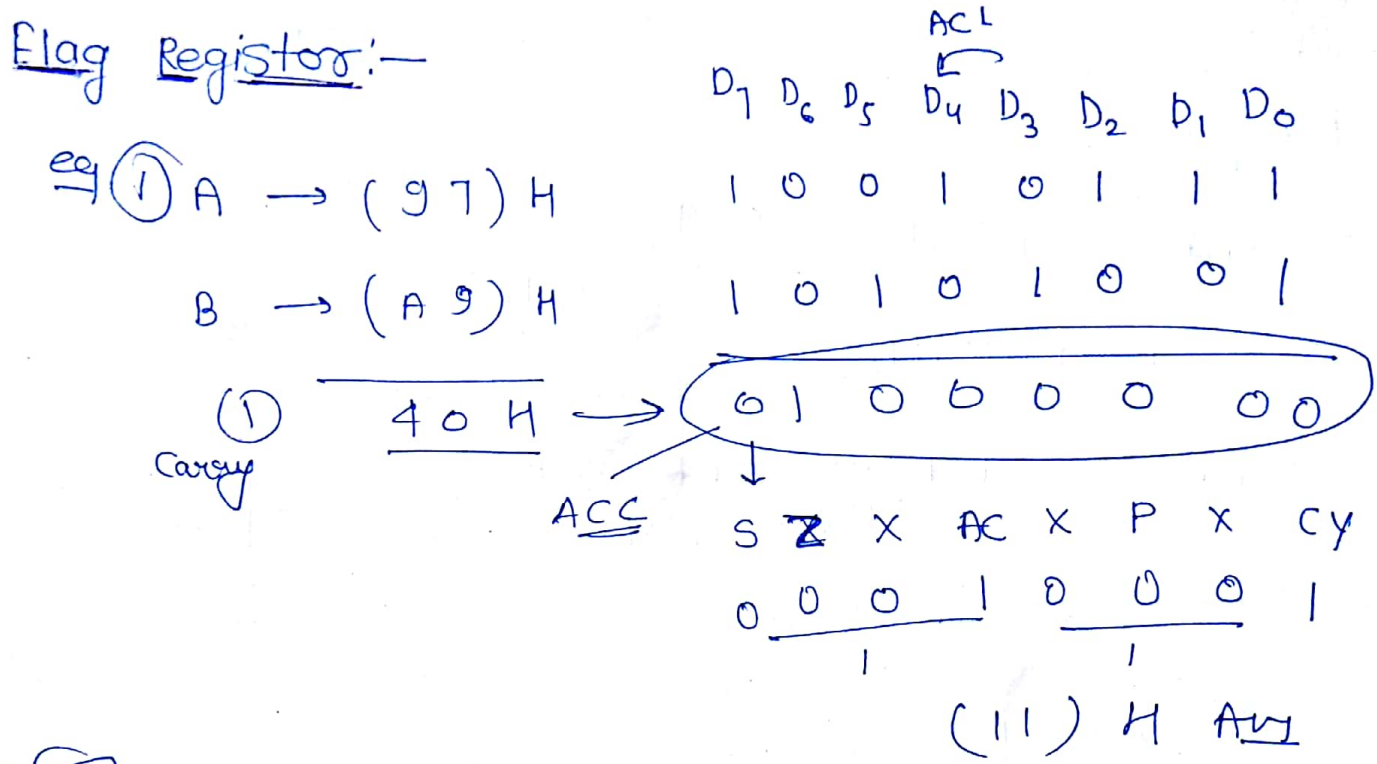
Instruction Decoder and machine cycle encode :–

After the OPCode is fetched into IR it is decoded in this block with the help of micro program the no. of operation are assigned according to type of instruction.

Microprogram :– It is program written by chip designer to make the processor to understand what an instruction is or it indicated the type of operation to be performed for an instruction.

It is present side the processor

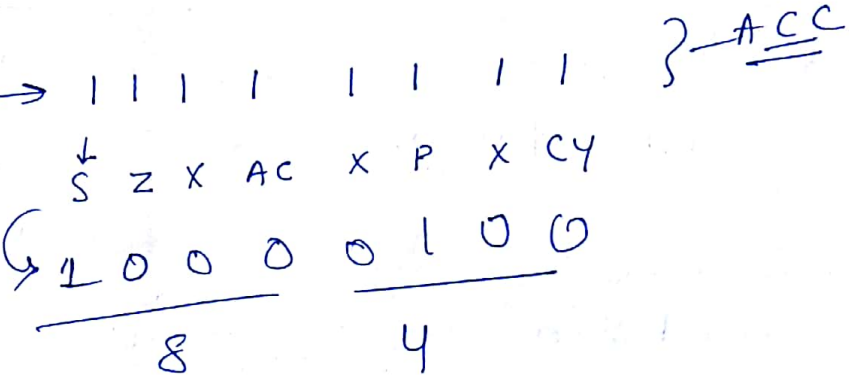
Flag Register:-



eg (2) 83 H →

+ 7C H

(FF) H →



Q (3) Find Flag

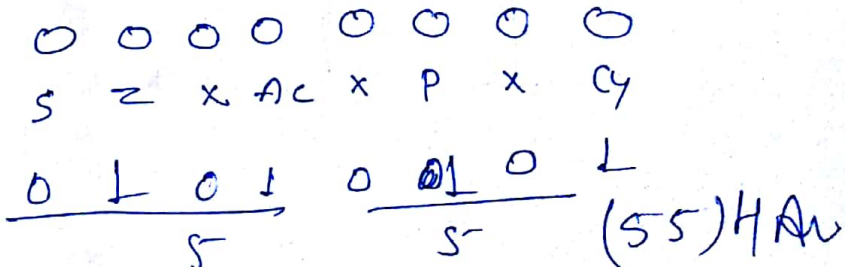
D6 H

+ 2A H

(1) (00) H

Carry

Result of (00) so Z = 1



Program/Processor status word (PSW):—

It is combined status of accumulator and flag register, where accumulator is higher byte. [Acc | Flag]

Acc — Result

Qu ① 40H & 11H → Flash

Acc

PSW [40 | 11]

Acc / flag

Que ② ~~FF~~ Acc = FF — High

Flash = 84 — Lower

PSW = [FF | 84]

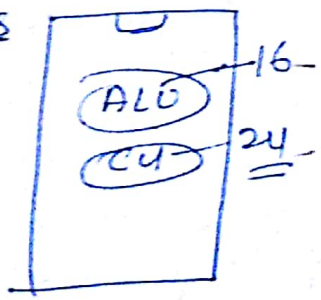
Qu ③ PSW = [00 | 55] A

Temporary Registers:— only processor can use them operator can not access them.

Can be directly used data operator → BC, D E, HL, SP → used define.

Indirectly used → PC, IR, FR, W

40 pins



Arithmetic & logic Unit :- (ALU) :-

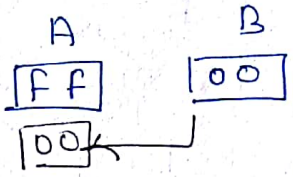
Combination of (ACC, TR, FR, LC) → ALU

- Addition
- subtraction
- Increment
- Decrement
- AND
- OR
- Ex-OR
- compare
- Compliment
- Rotate

* Flags are modified only for ALU operation.

ALU

Ques MOV A, B



Z 'flash' ?

(a) 1
(b) 0
(c) a/b ✓

it is not a ALU operation
so Flag can be 0/1

Ques if zero Flag Z = 0 initially

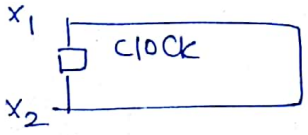
(a) 1
(b) 0 ✓
(c) a, b

it can't affect flag
so b Ans

Timing and Control Unit :-

↳ Brain of processor
- Responsible for Generate control.

X1 & X2 Pin :- crystal oscillator is connected to provide stable oscillation



R, C, L → other oscillator are temp. dependent

$$\text{frequency} = \frac{f_{crystal}}{2}$$

$f_{CLK} = 3$ MHz (8085)

↓

Range ⇒ 3–6 MHz

ALE : address latch enable
(Already)

$\overline{RD}$ – Read control signal — (bar) indicates low active
↙ low active signal
→ 0 : Active
→ 1 : Inactive.

$\overline{WR}$ – write control signal
→ 0 : Active
→ 1 : Inactive

IO/$\bar{M}$:- Status signal which ~~indicates~~ either M/M or I/O operation

→ 0 : Memory operation
1 : I/O operation.

both cannot performed parallelly

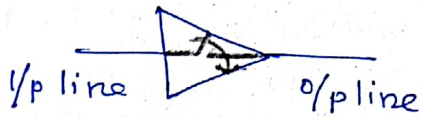
IO/M	RD	WR	operation	control signal
M/M opn → 0	0 Active	1 inactive	Fetch, M/M Read	MEMR
0	1 inactive	0 Active	M/M write	MEMW
I/O opn → 1	0 active	1 inactive	I/O Read	IOR
1	1 inactive	0 active	I/O write	IOW

Status line : S1/S0 :-

IO/M	S_1	S_0	status
Z	0	0	Halt
0	0	1	write
0	1	0	Read
0	1	1	fetch
1	0	1	I/O write
1	1	0	I/O Read
1	1	1	INTA
Z	X	X	HOLD
Z	X	X	Reset

Z – High Impedance state / Tristate State :–

Tristate Buffer :– It select one device from many device. It used when so many device connected to bus. To increase power or current level of bus.



E = 0 : active logic

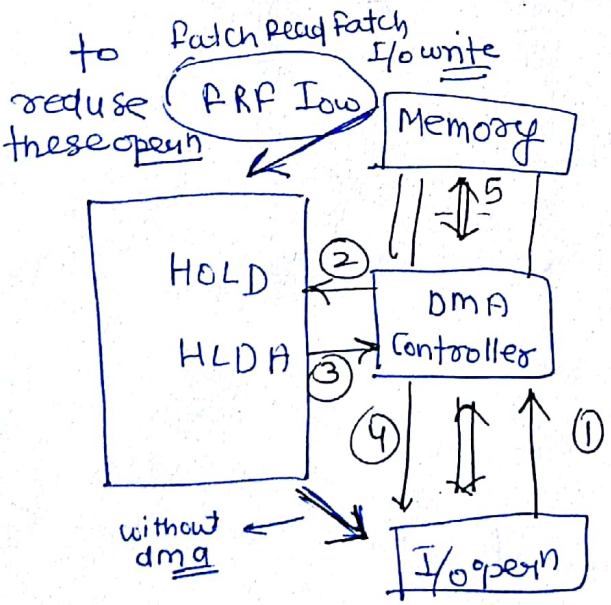
1 : Inactive logic / High Impedance state

eg ① 74LS244 Unidirn buffer

② 74LS245 Bi-dirn buffer.

Imp

HOLD / HLDA :– to transfer 1 byte from M/M to O/p device.



⇒ Direct M/M access operation eg 8257/8237

↘ HOLD – input
↘ HLDA – output
① DMA – request
④ DMA – Acknowledge

DMA = used to transfer more data at fast rate

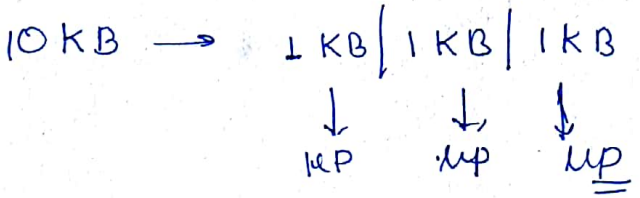
Modes of DMA :-

1. Burst Mode
2. cycle stealing techneque or short burst mode
3. Interleaved DMA.

1. **Burst mode** - HOLD signal remains high till total data is transmitted

eg:- burning of CD/DVD.

2. **Cycle stealing** :- Blocks of data transfered and processer can used buses.



eg: Audio and file transfer application together in a computer

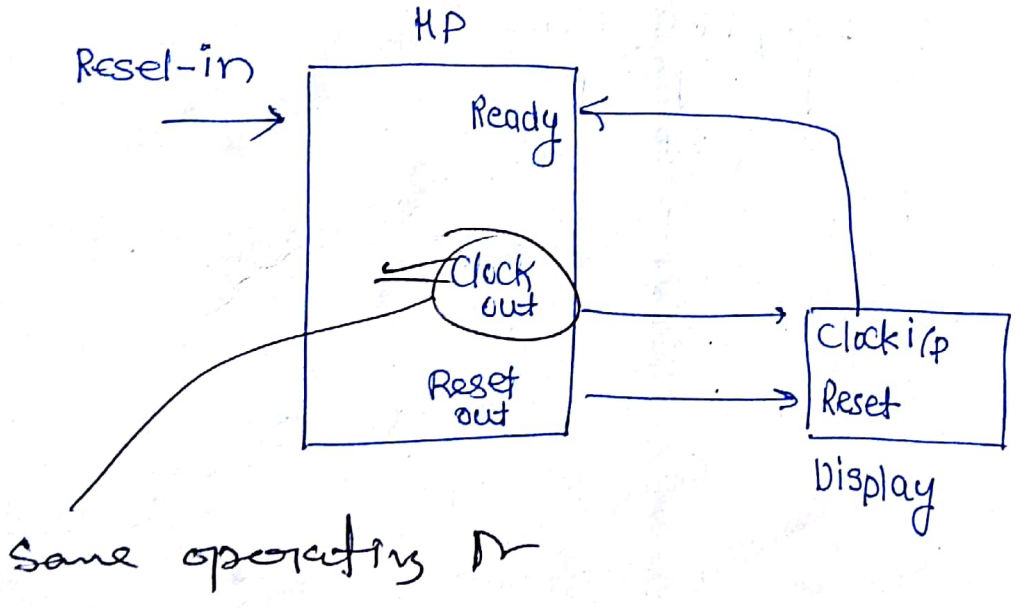
3. **Interleaved DMA** :- In this mode 1 byte or word is transfered per instruction. when processor doing internal work buses can be use by DMA.

Reset:- low active Input signal to the processor to reset it. PC = 0000H μp

Reset out:- o/p signal which indicate that the processor is reset.

Clock out:- o/p pin on which same operating frequency

Ready: → 1 µp transfer/receive data.
→ 0, µp waits for the i/o device



Interrupt control unit :-

① Hardware interrupts :- There are five hardware interrupt according to priority.

⇒ TRAP to INTR (see table)

② Software interrupts :- There are eight software interrupt which can be use either as instruction or along with INTR.

eg. ~~RSTn~~ RSTn n → (0 to 7)
(Vector address)

instruction
- RST 0 → 0000 H
- RST 1 → 0008 H
- ⋮
- RST 7 → 0038 H

③ Maskable Interrupt :- Interrupt which can be ignored or avoided are maskable

maskable
- RST 7.5
- RST 6.5
- RST 5.5
- INTR

④ Non-maskable :- Interrupt which can not be avoided when they are triggered eg. TRAP.
It must be connected to highly prioritized event like power failure or power related.

(5) Vectored interrupt: Interrupt which have specific location in the memory.

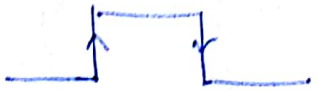
(6) Non Vectored :- interrupts which do not have specific address location in the memory eg. INTR.

It is also known as POLLED interrupt.

Priority		Interrupt	type of triggering	Vector Address.	
1.	Non maskable	TRAP/NMI/RST4.5	⊓	0024H	Vectored
2.	maskable	RST 7.5	⊓	003CH	
3.		RST 6.5	⊓	0034H	
4.		RST 5.5	⊓	002CH	
5		INTR	⊓	—	non vectored

level triggering

Edge triggering



ISR :- Interrupt service routine / Interrupt service Sub routine.

It is a program executed in response to an interrupt every Vectored interrupt in 8085 is allocated 8 bytes in M/M to store Corresponding ISR.

Vector Address calculation for Interrupts :-

RST n

$(n \times 8) = (x)_{10} = (y)_{16}$

(1) RST 6 → 0030H

$6 \times 8 = (48)_{10}$

$= (30)_{16}$

16 | 48 | 3
48
0

(2) RST 7.5

$7.5 \times 8 = (60)_{10}$

$= 003CH$

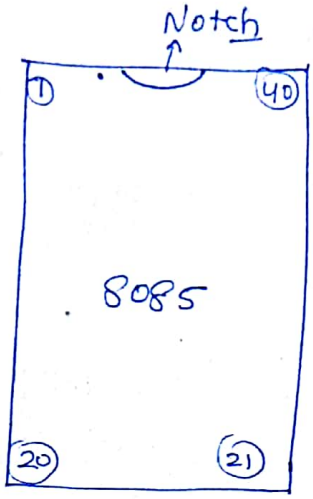
16 | 60 | 3
48
12 - C

Serial i/o Control Unit :-

SID ⇒ serial i/p data pin
SOD ⇒ serial o/p data pin

eg: computer & printer (RS 232C - Standard)

③ Pin layout :-



AD_7 - AD_0 (8+8)

input pins = 13/21

output pins = 27

① Programming Model :-

assembly language :- MOV A, B — 78 H

ADD C — 81 H

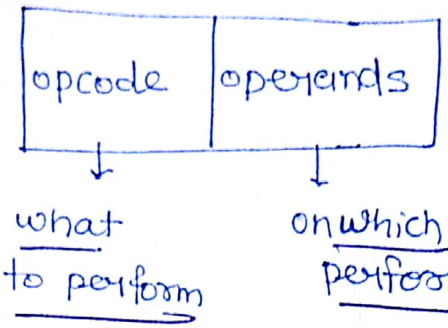
↑ assembly ↑ machine code.

low level language — Machine dependent

High level language :- C, C++, Java → Machine independent

Compiler : high level to machine level, where entire program is converted at a time.

⑤ Instruction format :-



Instruction length :- No of bytes an instruction occupies in M/M.

three type

① 1 bytes instⁿ

ex:- MOV A, B → xx

MOV → opcode, A, B → operands

② 2 byte Instⁿ (8 bit)

ex :- MVI C, (90)H → yy, 90H

move immediately

MVI → opcode, C → operands, (90)H → operands

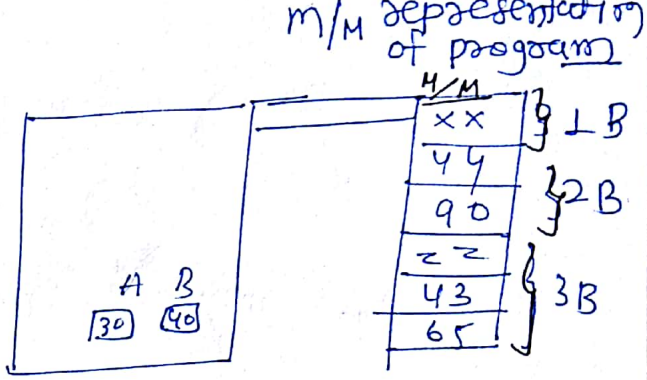
③ 3 bytes Instⁿ (16 bit)

ex: LXI H, 6543H → zz, 43H, 65H

LXI → opcode, H → operands, 6543H → operands

60 % — 1 B , 40 % — 2B/3B

M/M representation of program



Ex: OUT 60H — 2B
MOV B, M — 2B
LDA 4000H — 3B
LDAX B
LHLD 2000H — 3B

After decoding opcode

① length of Instⁿ

② No. of operation

XX, YY, ZZ - opcode ⇒ F (Fatch)

90, 43, 65 - data ⇒ R (Read)

eg OUT 60H = 2B → min 2 operⁿ

(FR) Fatch and Read

eg LDA, 4000H = 3B → mini 3 operⁿ

(FRR)

eg MOV M, C ⇒ move C to Memory

So FW (fatch write / transfer)

2 operation

* There is no direct relation b/w length of instruction and no of operation but mini operation can be found by length

Ques

MVI A, 60H ⇒ 2B

MVI B, 30H ⇒ 2B

ADD B → 1B

STA 3000H → 3B

Halt रोकना → HLT → 1B

length of program = 9B

output → B: 30, A: 60

FRFRFFRRF

60 + 30 = 90H, store in 3000H

So content at 3000H is 90H Ans

⑥ Addressing Modes :- 5 types mode

1. Immediate :- MVI A, 60H
   ↓
   Instⁿ have data

2. Direct :- LDA 7000H
   - LDA → load → Accumulator
   - 7000H → [Address of data] [take from 7000H M/M]
   - यहाँ जाकर Data लाना है
   - (M/M Read)

3. Indirect :- LDA X B = 1 bytes
   ↳ point data indirectly
   (M/M Read).

4. Register : MOV B, C ⇒ 2 B
   address in register pair

5. Implicit/implied :- eg CMA, RAL ⇒ 1 B
   address in instⁿ itself
   CMA ⇒ compliment Content of accumulator (ACC).

⑦ Timing Diagram :-

Instruction Cycle
↑
Machine Cycle (Sub-Set)
↑
T- state (Sub-sub set)
↳ one Sub-division of one operation

T-State (8085)

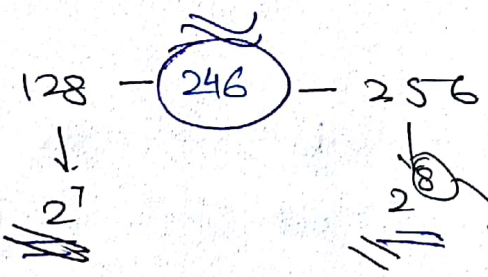
$f_{CLK} = 3\ MHz$

$T = \frac{1}{f} = \frac{1}{3 \times 10^6} = 0.33\ \mu s \rightarrow$ T-states

* 1 Machine may have 3-6 T-state

- F — 4 T-state
- MR, MW, IOR, IOW — 3 T-state

Note:- There are 74 different instruction pattern (ex. MOV) which result in 246 instruction and hence opcode length is 8 bit / 1 byte in 8085.



1 byte ⇒ length of 8085

Ques Total T-state of a program are 27 are fine total execution time

Sol

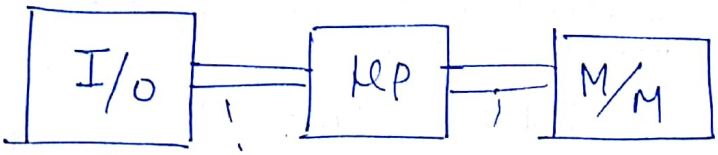
$$\text{Total execution time} = \frac{1}{f_{CLK}} \times \text{T-State} \times \text{Count Value}$$

(Not given take 1)

$$TET = \frac{1}{3 \times 10^6} \times 27 \times 1 = 9\ \mu s$$

(10) **Interfacing** :- Designing logic Circuit and writing program to make the processor to communicate either m/M or I/O device is known as **Interfacing**.

The logic circuit used are know as interfacing circuit or I/O port.



Interfacing circuit
or
I/O port

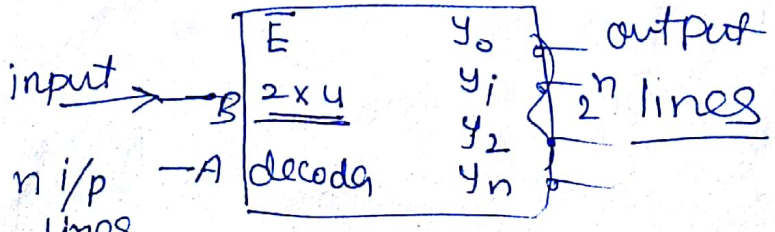
Basic Interfacing Component :-

(1) Tri-State Buffer :-

(2) **Decoder** :- Used for **decoding** logic of **M/M** or **I/O** device

ex:- **74LS138 ⇒ 3x8 decoder**

It is a logic circuit which can identify Combination of input signal and select one of the output.

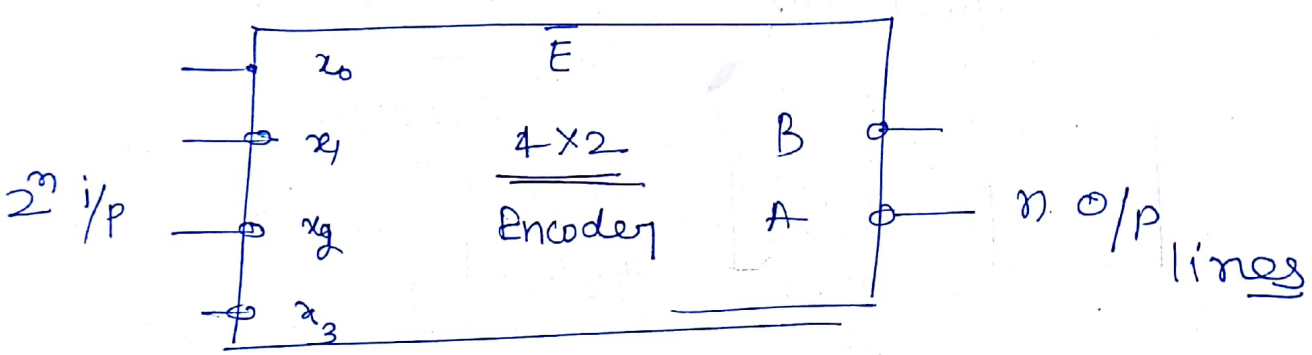


decoding logic

i/p B/A	o/p
0 0	y_0
0 1	y_1
1 0	y_2
1 1	y_3

③ Encoder :- it is logic circuit which can generate / produces the code of on of input signal in output.

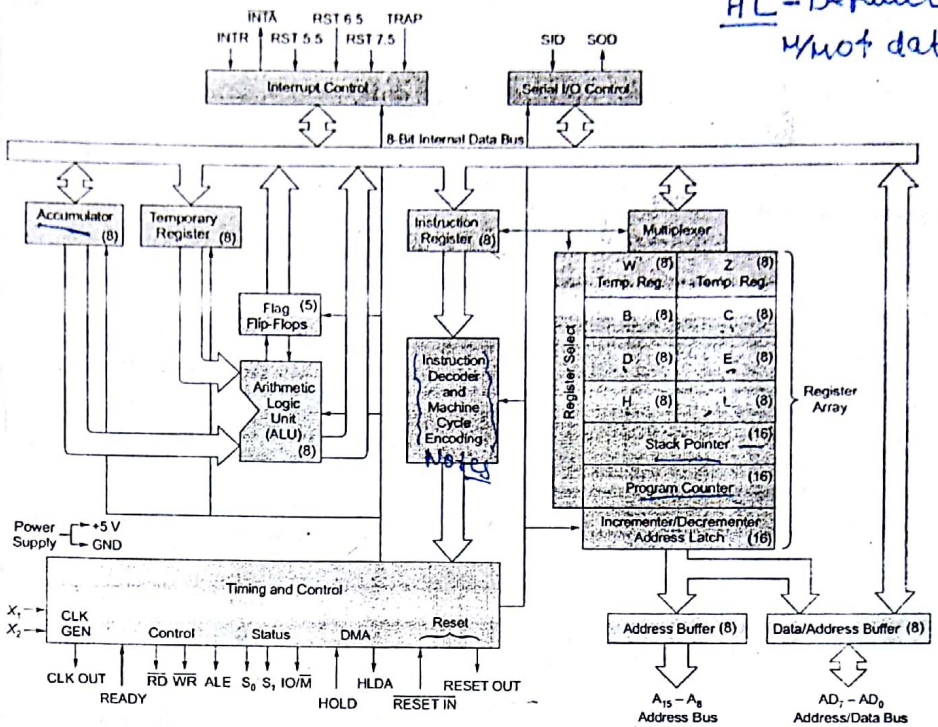
Used in communication system and also interface a keyboard to a computer.



eg ⑧ 74L5148 → 8×3 priority encoder

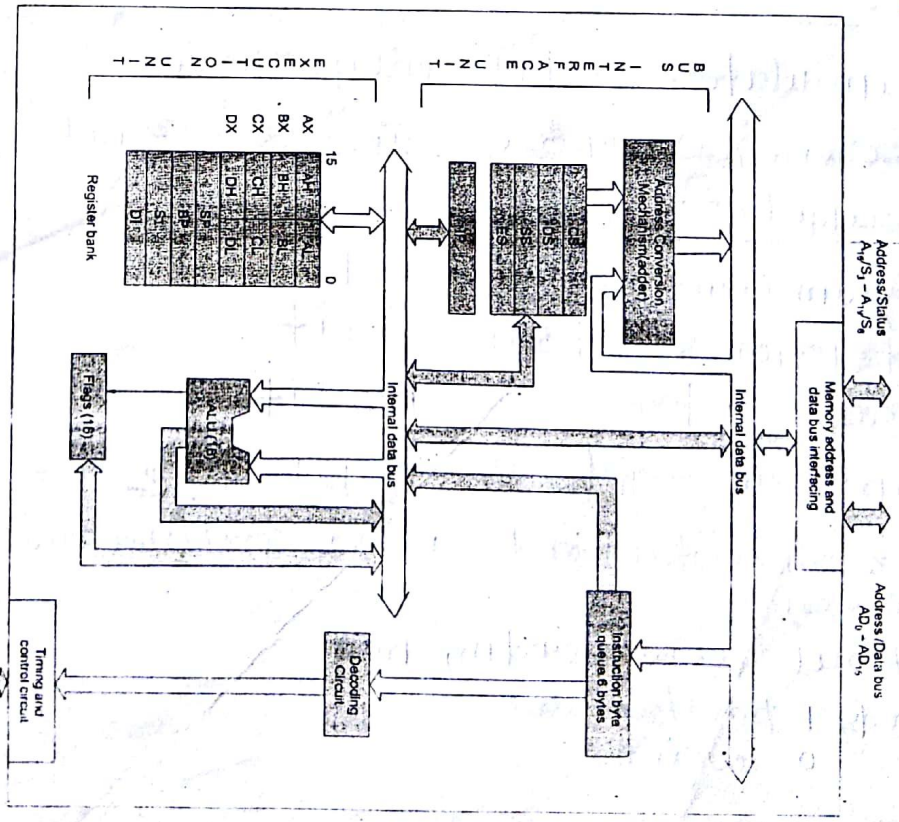
Higher byte of register pair { B, D, H } { C, E, L } lower byte of register pair

HL - Default M/mot data pt



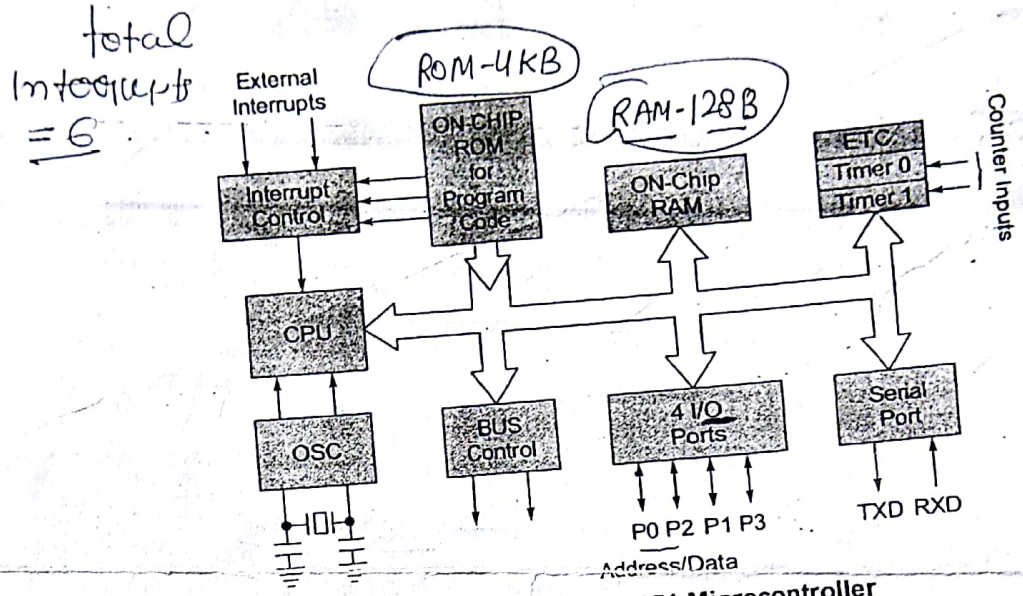
Architecture of INTEL 8085 Microprocessor

Dilkhush Meena.



Architecture of INTEL 8086 Microprocessor

total Interrupts = 6



Block Diagram of 8051 Microcontroller

4K ROM and 128 B RAM

Freq :- F = 11.24 – 12 MHz

Accumulator : 8 bit multipurpose

General purpose registers = 8 bit

Accumulator : 8 bit multipropose

Higher order: B C D E H L — defau
lower order

Program Counter (PC) : 16 bit

Instruction register : 8 bit

Stack pointer : 16 bit

temporary register : 8 bit W, Z

Program status word : PSW (Combination of ACC + flag (processor)

Clock out : same operating freq

Ready 1 trans/rece. data
0 wait

Internal Architecture of 8085:

It is divided into 5 functional units:

1. Register Unit
2. ALU
3. Timing and control Unit
4. Interrupt control Unit
5. Serial I/O control Unit

1. Register Unit:

There are two types of registers

1. General purpose Registers
2. Special purpose Registers

General Purpose Registers:

There are 6, 8bit general purpose registers, namely

B, C, D, E, H & L . They can also be used as register pairs of 16bit length.

1. BC 2. DE 3. HL

Any of them can be used to point the memory but HL pair is also known as default memory or data pointer.

Ex. MOV D,M

'M' –indicates memory content or memory location whose address is present in HL pair only.

Special Purpose Registers:

Accumulator: 8-bit Multi purpose register by which almost all ALU operations are performed. One of the bytes must be present in 'ACC'/'A', result is also stored in ACC.

- ✓ Program Counter:--PC

16 bit register which contains the address of the next instruction to be executed or it takes of the program flow or control. It is automatically incremented by one after an operation depending on the instruction.

Instruction Register:

It is an 8 bit register which contains the opcode of present instruction. It does not contain data.

- ✓ Stack Pointer: SP

It is a 16bit register which contains the address of the data present at the top of stack memory or it points to top of the stack.

- ✓ Stack: It is a part of Read/Write memory, used to store temporary data and also the content of program counter when *subroutines* are used.
- ✓ The technique involved in stack is LIFO i.e. Last in First out.

When data is stored in to STACK, SP is decremented, similarly SP is incremented when data is accessed from stack.

Flag Register:

D7	D6	D5	D4	D3	D2	D1	D0
S	Z	X	AC	X	P	X	CY

S- Sign; Z- Zero; AC – Auxiliary Carry; P- Parity;

Cy-Carry

S- 0/1 -- Depending on the D7 bit of accumulator .In Signed operations S→ 0; +ve Result; S→ 1; -ve Result

0/1

Z→ 1; if result in ACC=00H
→ 0; Otherwise.
AC→ 1; if there is carry from D4 bit ←D3 bit or
Higher nibble ← Lower nibble
→0; otherwise
P→1; if the number of binary 1's in ACC=Even
→0, if the number of binary 1's in ACC=odd
CY→ 1; if there is a carry i.e out of D7 bit.
→0; otherwise.

1's even P=1
odd P=0

Temporary Registers:

W & Z are the two 8 bit temporary registers which cannot be accessible by programmer. They are used by the processor in some instructions.

ALU: Arithmetic & Logic Unit:

It is the combination of ACC, Temporary register, Flag register & arithmetic and Logic circuits.
The various ALU operations possible in 8085 are:
Addition, subtraction, Increment, Decrement
AND, OR, EX-OR, COMPARE, COMPLIMENT & ROTATE
- ✓ Flags are modified only for ALU operations

Timing & Control Unit:

It is responsible for generating various control, timing and status signals used for the operations of the processor. Like M/M Read, M/M Write.
X1 & X2→ a crystal oscillator is connected to provide stable oscillations.
- ✓ Operating frequency of 8085 is 3 MHz.(it is half of crystal frequency)

F_{CLK} = 3MHz. Range→ 3 – 6 Mhz.

ALE—Address Latch enable
Used to make $AD_7 - AD_0$ either as address or data bus.
$\overline{RD}$→ Read control signal; low active signal; $\overline{WR}$→ Write control signal; low active signal
→0; Active →0; Active
IO/$\overline{M}$→ 0; Memory operation
→1; I/O operation
S1 & S0→ status signals which indicate the status of Bus cycle for an operation. They are used by other systems to know the status of processor.

IO/$\overline{M}$	S1	S0	STATUS
Z	0	0	HALT
0	0	1	M/M Write
0	1	0	M/M Read
0	1	1	Fetch
1	0	1	I/O Write
1	1	0	I/O Read
1	1	1	INTA
Z	X	X	HOLD
Z	X	X	RESET

2-Tristate state/High impedance state

Tristate Buffer:

It is used when many devices are connected to common bus. It is used to amplify or increase the power or current level of the bus.

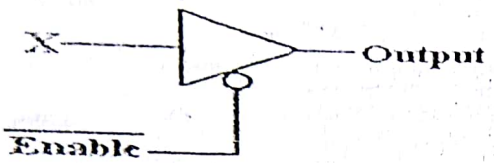
It has three states 1). Logic 0 2). Logic 1 3).High impedance state (Z/ Tristate)

- ✓ In high impedance state, the line doesn't draw any current from the system.

Circuit symbol of Tristate Buffer:

If E=0; Active/logic 0/1

=1; inactive/z/high impedance state/tristate



Dilkhush

HOLD & HLDA:

Purpose of DMA : when more data has to be transferred between memory and I/O devices at a faster rate , Direct memory access operation is used with the help of DMA controller.

HOLD: High active input signal to the processor from DMA controller requesting buses for DMA operation.

- ➢ Sequence for the DMA operation (considering data to be accessed from memory to output device).
  1. µp receives HOLD signal DMA controller.
  2. µp Completes current operation and responds with HLDA.
  3. The control of buses is given to DMA controller.
  4. After completing data transfer, the control of buses is accessed by the µp.

This process of data transfer is known as DMA operation.

By using using DMA operation, there is no need of opcode fetch for every single data byte. Once the address and count registers are loaded in DMA controller, the data is transferred till the count becomes 0.

HLDA: hold acknowledgement—out put signal in response to hold request.

- ✓ Reset-in—low active input signal to the processor to reset it. PC= 0000H

Reset out: o/p signal which indicates that the processor is reset. It can be used to reset the i/o devices.

Clock out: o/p pin on which same operating frequency of the processor is available.

Ready: i/p signal to the processor from a slow speed i/o device or memory.

Ready→1; mp transfers or receives data from i/o

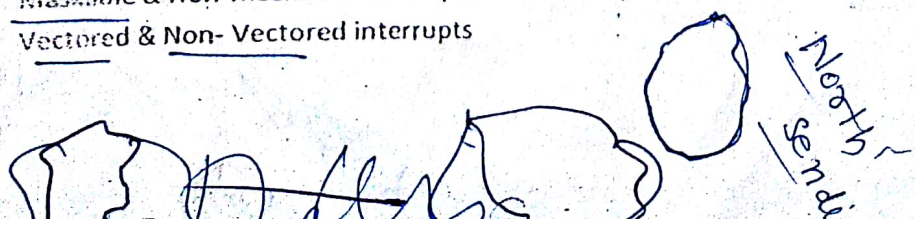
→0; mp waits for the i/o device.

Interrupt control Unit:

Interrupt is an external signal or an instruction which may disturb or alter the sequence of execution of the processor or it is a method by which an i/o informs the processor that it requires the service from the processor.

Interrupts can be classified as :

1. Hardwired & software interrupts
2. Maskable & Non-Maskable interrupts
3. Vectored & Non- Vectored interrupts



Pins

	Vector address	Type of triggering
1. TRAP/NMI/RST 4.5	0024H	Edge & level
2. RST 7.5	003CH	Edge
3. RST 6.5	0034H	Level
4. RST 5.5	002CH	Level
5. INTR	No address	Level

INTR---Non-vectored interrupt(it does not have any specific location).

INTA- Interrupt acknowledgement- o/p signal in response to interrupt request.

It is required only for INTR interrupt not for vectored interrupts.

Serial i/o Control Unit:

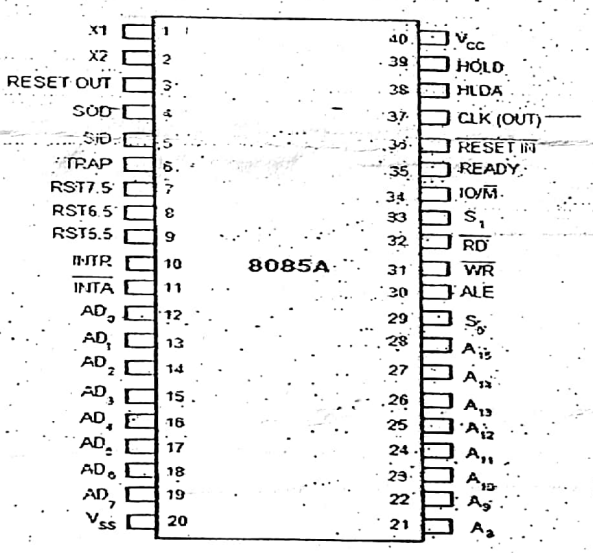
Used for serial communication between mp and serial i/o devices. Bit by bit transmission is done in serial communication. Ex: connection between computer and a printer(RS232 C-Serial communication standard)

SID- Serial i/p data pin—used to receive serial data

SOD- Serial o/p data pin – used to transmit serial data.

PIN LAYOUT:

- 8085 is a 40 pin IC in DIP – dual in-line package, designed with NMOS technology(n-type metal oxide technology)
- i/p pins---13/21, o/p pins----27.



PROGRAMMING MODEL:

Program: set of instructions

Instruction: it is a command given to a computer to perform a specific task.

Machine level language: Binary medium of communication with a computer through a designed set of instructions, specific to a system.

Assembly level language: Instructions are written in separate words known as mnemonics.

Ex: MOV A,B ADD C

- ✓ Both assembly and machine level languages are known as low level languages

High level language: they are machine independent. Ex. C, C++, JAVA

Softwares:

Source program = Assembly program

Compiler

Ex; Turbo C, XLC, Javac

Interpreter: HLL – MLL line by line, Ex: M-BASIC o/p

Assembler: Assembly program to Machine code, MASM – microsoft macro assembler.

Loader: used to load the program into memory, it also converts hex code to binary(ex; .EXE file) convert exe file in binary

Source code: Assembly program which is written for a certain application.

Object code: o/p of the assembler

Monitor: it is a program which is similar to OS (Operating system) software in microprocessor environment. It takes care of initialization procedure and user interactions. It is present in the external memory. monitor program should be in ROM

It loads the PC with the starting address of actual program when the "enter or execute" key is pressed.

- ✓ **Note: Programmer cannot load the PC directly with the starting address of program.**

Basic steps of execution: 1). Fetch 2). Decode 3). Execute

Instruction format:

Every instruction has two parts 1). Opcode 2). Operands.

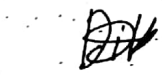
Opcode: indicates the type of operation to be performed

Operands: It is the data on which operation is to be performed.

Instruction length: Number of bytes an instruction occupies in the memory.

There are 3 types of instructions classified according to the length:

1. 1 byte/word instructions 2. 2 byte/word instructions 3. 3Byte/Word instructions.



Addressing modes: → Data को कैसे लाएगा है ?

These are various formats specifying the operands or they indicate how data is accessed for an instruction

1. Immediate Addressing mode: Data is in the instruction. Ex. MVI B, 90H
2. Direct addressing mode: Address of the data is in the instruction. EX. LDA 6000H
3. Indirect addressing mode: Address of the data is present as the content of another register pair EX. LDAX B
4. Register addressing mode: Data is transferred between registers. EX. MOV B,C
5. Implicit/ Implied addressing mode: Address of the data is in the instruction itself. EX. CMA,RAL

TIMING DIAGRAM:

It is the pictorial representation of execution of an instruction with the help of various control ,timing and status signals.

T-State: it is one subdivision of an operation performed in one clock period.

Machine cycle: It is the time required to access memory or i/o (either for read/write)

1 M/c may have 3 to 6 T-States. OF, MR, MW, IOR, IOW

Instruction cycle: It is the time required to complete the execution of an instruction.

1 I/C may have 1 to 5 machine cycles.

Note: maximum T-State for an instruction in 8085 is 18. Ex: Call 16 bit address—3B,5 M/C's, 18- T-States.

Instructions in 8085

1. **Data transfer or copy instructions: data is transferred from source to destination.**
2. **Arithmetic instructions:**
   Perform operations like Addition, subtraction, increment & Decrement.
3. **Logical instructions: AND, OR, EX-OR, Compare ,Compliment & Rotate**
4. **Branching Instructions: Program control is transferred from one location to another conditionally or unconditionally .Ex: JUMP, CALL , RETURN & RST n.**
5. **Machine control Instructions: Used for internal machine control operations of processor.**

Depending on the type of instruction, the number of operations may one or more.

Data transfer instructions		
Instruction	**Meaning**	**Example**
MVI R, 8 Bit data	Move immediate 8-bit data to register	MVI C, 90H
MOV R_d,R_s	Move content of source register to destination	MOV A,B
LXI R_P, 16bit value	Load immediate 16bit value into register pair	LXI H, 8000H
MOV R, M	Move content of memory to register i.e & ref of HL pair	MOV D,M

MOV M, R	Move content of register to memory	MOV M, E
MVI M, 8bit data	Move immediate 8bit data to memory	MVI M, 75H
LDA 16Bit address	Load data @ 16bit address to accumulator	LDA 4000H
STA 16bit address	Store the content of 'A' @ 16bit address	STA 3000H
LDAX Rp	Load Accumulator with data , whose address @ reference of R_p	LDAX B
STAX R_p	Store Accumulator content @address present in R_p	STAX D
PUSH R_p	Store the content of R_p into stack memory. SP→SP-2	PUSH D
POP R_p	Access data from top of stack in R_p. SP→SP+2	POP H
IN 8-bit port address	Read the data from 8bit port address in A	IN 60H
OUT 8-bit port address	Transfer the content of A to o/p port	OUT 70H

ARITHMETIC INSTRUCTIONS

ADD R	ADD M	ADI 8 bit data
ADC R	ADC M	ACI 8-bit data
SUB R	SUB M	SUI 8-bit data
SBB R	SBB M	SBI 8-bit data
ANA R	ANA M	ANI 8-bit data
ORA R	ORA M	ORI 8-bit data
XRA R	XRA M	XRI 8-bit data
CMP R	CMP M	CPI 8-bit data

Compliment–CMA – compliment content of accumulator

- CMC – Compliment carry

ROTATE – RLC, RAL , RRC , RAR

Branching Instructions

JUMP : To transfer control of program from one location to another

There are conditional and unconditional instructions

Conditional: depend on status of flags

Ex: JZ 16bit address(if Z=1, in previous ALU operation, then jump to the address , else go to next instruction in the program.)

Similarly all flags except AC have the instructions.

JNZ,JC,JNC,JPE,JPO,JM,JP

Unconditional Jump : JMP 16bit address—go to the address mentioned in the instruction without any condition.

CALL : To call a subroutine within the main program.

Subroutine: A program which performs specific function can be written as a separate program away from main program. It can be repeatedly used in the main program.

CALL also has conditional and unconditional instructions.
Unconditional CALL:- CALL 16bit address

When CALL (or interrupt is activated) is executed, the steps followed by processor:

1. Store/Save the address of next instruction(PC content) in the stack ,i.e SP→SP-2
2. Go to the subroutine address and continue the program

Conditional CALL → operation is same as unconditional CALL, but flag condition is verified before the control is transferred.
CZ,CNZ,CC,CNC,CPE,CPO,CM,CP

RETURN: used to return the control of a program from a subroutine or an interrupt service routine. It also has conditional and unconditional .
Unconditional return →RET
When RET is executed:

1. Processor accesses the data(2B) at top of stack in to PC, i.e SP→SP+2
2. Program control is transferred to the 16bit address value

For conditional Return instructions also operation is same except that the flags condition is verified before the transfer.
Ex. RZ,RNZ,RC,RNC,RPE,RPO,RM, RP.

RSTn—Software interrupts , where n=0-7

- Also known as 1Byte unconditional CALL instruction.
- Operation is similar to CALL instruction.

MACHINE CONTROL INSTRUCTIONS
HLT— stop the execution of a program.
NOP—No operation
DI—Disable maskable interrupts
EI- Enable maskable interrupts
SIM—Set interrupt Mask—to mask the interrupts or make them available & also for sending serial data out of SOD pin.
RIM—Read interrupt mask—to know the status of pending interrupts and to receive serial data through SID pin.

SPECIAL INSTRUCTIONS	
LHLD 16bit address	Load HL pair with the data @ 16bit address i.e 2bytes
SHLD 16bit address	Store the contents of HL pair @ 16 bit address
PCHL	Copy the content of HL pair to PC • Also known as 1B JUMP instruction
SPHL	Copy the content of HL to SP
XCHG	Exchange the contents of DE & HL pairs
XTHL	Exchange the contents at top of the stack with the HL pair. • SP is unchanged
DAD R_P	Add the contents of Register pair to HL pair. Result is stored in HL Pair only.

DAA	Decimal adjust Accumulator— Used in the BCD conversions. *This is the only instruction which works with the status of AC flag Converts 8bit value in Accumulator into two 4bit BCD values.

RAM- Random Access Memory

- Also known as Read /write or temporary or volatile memory
- Data is present Only with the supply of power

	SRAM	DRAM
Data storage	Flip flops are used, bit is stored as voltage	Data is stored as charge between GATE to substrate of a MOS Transistor. As the charge may leak off, it must be refreshed frequently
Each Memory cell has	3- 6 transistors , it may be BJT'S -2 & Resistors -2 Or 4 MOS transistors	one transistor
Density or storage capacity	less	More
Speed	Faster than DRAM, i.e low access time	Slow in operation
Cost and power consumption	More	Less

ROM→ Read only memory / Non-Volatile memory – data can be permanently stored.

TYPES:

MROM- masked ROM, PROM- Programmable ROM

EPROM—Erasable PROM, EEPROM—Electrical erasable PROM

FLASH→ same as EEPROM but data can be erased as a block. Ex. Pendrive or USB drive