

Date: / / Page: / /
Digital Electronics

21/08/19

Date: / / Page: / /

Digital Electronics/System → It is combination

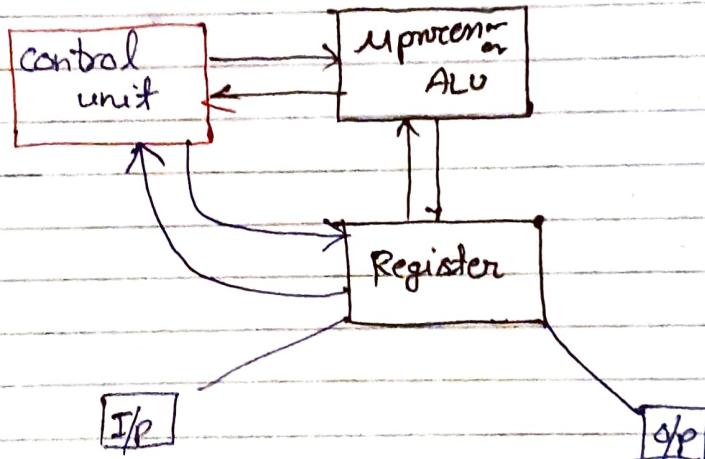
of devices that are designed to manipulate physical quantity of logical information that are in the form of digital or discrete.

& → Digital computer, Digital Audio & video fields.

Advantage of Digital system

- 1) Packaging density is high
- 2) Easy to design
- 3) D.S takes less memory

{ Microprocessor is the key point of D.S.



Digital computers
abstraction
& manipulation.

Number System → Positional & Non-positional Number System

→ What is the significant of the complement?

→ Codes → Weighted code

→ Non weighted code

BCD
Gray

, 8425

Weighted \rightarrow 7421, 5421, 5211, 4211, 2421, 7421

Non weighted code \rightarrow Excess-3 code

\rightarrow Gray

\rightarrow Alphanumeric

\rightarrow ASCII \rightarrow 128 different patterns

\rightarrow EBCDIC

\rightarrow Hollerith code

\rightarrow Error detecting code (Parity)

\rightarrow Error detecting and correcting code
(Hamming code)

\rightarrow Self complementary & reflective code

$\hookrightarrow 5211 = 9$ sum = 9 must be reflecting

sum = 9 must be

FF \rightarrow Flip F

one bit is
in clocked

\rightarrow A FF is also

To make
Type of F

S=1

CLK

R=1

Ex-

8421.

0 0 0 0 \rightarrow 0

0 1 1 1 \rightarrow 1

0 1 1 0 \rightarrow 2

1

1

1

1 0 0 0 \rightarrow 8

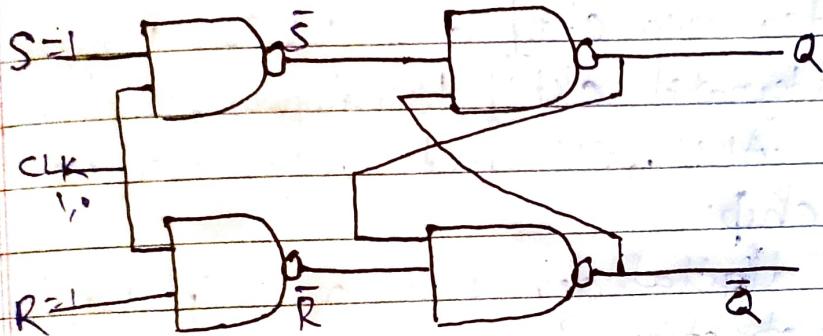
1 1 1 1 \rightarrow 9

FF → Flip Flop → It is a memory element which is capable of storing one bit of information. It is used in clocked sequential ckt.

→ A FF is also known as bistable multivibrator.

To make FF NAND or NOR Gate are used

Type of Flip-Flop → SR, JK, D, T



Microprocessor-

- A Microprocessor is an important part of a computer architecture without which you be able to perform anything on your computer system.
- It is a programmable device.
 - MP is the semiconductor (Digital) device that is manufactured by the LSI [large scale integrated ckt] and VLSI [Very Large Scale Integrated ckt] techniques, which includes ALU, CU and register on a single chip.
 - It is a digital device on a chip which can fetch instruction from memory, decode and execute them and give results.

Basic of MP → A MP takes a bunch of instruction in machine language and executes them, telling the processor what it has to do.

Microcontroller - MC is a semiconductor device that can be manufactured by LSI & VLSI technique that includes MP along with external peripheral such as Input & Output on a single chip.

Historical Development of MP

MP	Year	Word Length	Memory Capacity
----	------	-------------	-----------------

Word
pro
of

Sr. no MP

- 1) Intel (PMA) 4004
- 2) Intel 8008
- 3) 8080
- 4) 8085
- 5) Intel 8086
- 6) Intel (80)
- 7) Intel (80)
- 8) Intel (80)
- 9) Intel 80386
- 10) INTEL
- 11) Pentium
- 12) Pentium
- 13) Pentium
- 14) Pentium
- 15) Pentium

Sr.no	MP	year	word length	memory capac
1)	Intel(Pmos) 4004	1971	4 bits	640 B
2)	Intel 8008	1972	8 bits	16 Kb
3)	8080 n-Mos	1973	8 bits	64 Kb
4)	8085 (N-mos)	1976	8 bits	64 Kb
5)	Intel 8086(H-mos)	1978	16 bits	1 MB
6)	Intel (8088)	1980	8/16 bits	1 MB
7)	Intel (80186)	1982	16 bits	1 MB
8)	Intel 80286	1983	16 bits	16 MB(Real) 40MB(Virtual)
9)	Intel 80386	1983	32 bits	4 GB real 4 GB virtual
10)	INTEL 80486	1989	32 bits	4 GB(Real) 64 Tb(Virtual)
11)	Pentium-1	1993	64 bits	64 GB real
12)	Pentium-2	1997	64 bits	64 GB real
13)	Pentium-3	1999	64 bits	64 GB real
14)	Pentium-4	2000	"	"
15)	Pentium 4	2001	64 bits	64 GB Real

Word Length → The number of bits that can be processed by a MP is termed as word length of the MP. Word length of 8085 is 8 bits.

Basic Terminology in MP

- Nemonic → (ASSEMBLY)
- Programming
- Machine Language
- Assembly Language
- Low Level "
- High Level "
- Compiler, "

Processor 8085

It is an 8-bit MP designed by INTEL using NMOS Technology.

→ It has following configuration

-) 8 bit Data bus
- *) 16-bit address bus, which can address upto 64KB
-) A 16 bit program counter.

Address Bus → It is a group of 16-lines i.e. A₀ - A₁₅. It is unidirectional i.e. bits flow in one direction from the MP unit to the peripheral devices and uses the high order address bus.

Control
Control which controls
be ex
Mov the " by th

Parity
Byte
Word
Double
Stain

Location

5) Nibble

6) by

7) Integrat

8) Com
sim
code
Lang
bit

Control Line :- (Bidirectional)

Control line or unit is the bidirectional which basically generated timing and controlling signal.

Opcode & operand → An opcode is a single instruction that can be executed by the CPU. The opcode is the `MOV` instruction. The other parts are called the 'operands'. Operands are manipulated by the opcode.

Possible operand types in MP:

- i) Byte and short → 8 bit variable
- ii) Word and Integer → 16 bit variable.
- iii) Double word long integer → 32 bit variable.
- iv) String - series of bytes or series of word which is generally stored in memory location.

v) Nibble: 4 bit unit.

vi) byte word or double word - these are unsigned operands.

vii) Integer short & long → Signed operands.

viii) Compiler → Assembler → The compiler is a simple program which converts the source code written by the humans to a machine language. While the assembler has a little different work, it converts the assembly

language to the machine language.
Compiler works more directly than the assemblers.

Control & Timing Unit → The control and timing unit coordinates with all the actions of the CPU by the clock and gives control signals which are required for communication among the CPU as well as peripherals.

8085 PIN Description.

- 1) Von Neumann Architecture:
- 2) Harvard Architecture

The difference is that Von Neumann Architecture has common memory space for data and instruction while in Harvard it has separate memory for data and instructions.

→ Program & Data can be stored in same memory location Ex - 8085, 8086

Von-Neumann

⇒ 2) 8085

⇒ 8085 has 40 pins

Exact

Pin Diagram of 8085 uP

Date: / / Page:

X ₁	1	40	Vcc
X ₂	2	39	HOLD
RESET OUT	3	38	HLDA
SOD	4	37	CLOCK OUT)
SIO	5	36	RESET IN'
TRAP	6	35	READY
RST 7.5	7	34	I/O/M'
RST 6.5	8	33	S ₁
RST 5.5	9	32	RD
INTR	10	31	WR
INTA	11	30	ALE
A _{D0}	12	29	S ₀
A _{D1}	13	28	A ₁₅
A _{D2}	14	27	A ₁₄
A _{D3}	15	26	A ₁₃
A _{D4}	16	25	A ₁₂
A _{D5}	17	24	A ₁₁
A _{D6}	18	23	A ₁₀
A _{D7}	19	22	A ₉
V _{SS}	20	21	A ₈

Exact Pin Description of 8085 uP -

General Pin description of 8085

Date: 1-1-18 Page: 1

- bit capa
- It has addressing
- The crystal and the which is
- lower address bus of pin.
- To Demultiplex latch means
 $ALE = C_6$
- Disadvantage will be

Serial

Key feature of 8085

- 1) It is a 8 bit processor.
- 2) It has 8 data bus line, which is the

1)
2)
3)
4)

bit capacity of MP.

It has total 16th address line which addressing capacity of 64K.

The crystal frequency of processor is 6 MHz, and the clock frequency is $3.07 \approx (3\text{MHz})$ which is the half of the crystal frequency.

$$\rightarrow \boxed{\text{Clock frequency} = \frac{1}{2} \text{crystal frequency}}$$

\rightarrow lower order address bus is multiplexed with data bus in order to reduce the number of pin.

\rightarrow To Demultiplex address bus from Data, ALE (address latch Enable) signal is used if $\text{ALE}=1$ that means address will transfer to bus. $\text{ALE}=0$ Data transfer to bus.

\Rightarrow Disadvantage of multiplexing is that speed will be reduced.

Serial number	Signal	Pins
1)	Power supply	Vcc, Vss
2)	Clock signal	X ₁ , X ₂ , CLK out
3)	Reset Signal	Reset in, Reset out
4)	Interrupt Signal	TRAT, Restart, INTR, INRD (Interrupt RST, 7.5, 6.55.5)

Microprocessor

- 5) Address bus & Data Bus
6) Status signal & control signal

7) Serial I/O Signal

8) DMA Request Signal

- 1) Address Bus
2) Data Bus

- 3) Control Buses
4) System Bus

i) Bus is a group of wires or conductor that is used for communication b/w MP, memory, I/O device, peripheral.

A) Address Bus: Group of conducting wires which carries address only.

address bus is unidirectional because data flow is in one direction, from MP to memory or from MP to I/O devices.

length of address bus of 8085 MP is 16 Bit
→ It is defined as maximum no. of memory that can be connected with MP.

$$2^n = N$$

ii) Data bus: It carries Data only. It is used to transfer data b/w MP, memory, peripheral bidirectional. Data flow in both direction

Pins

A0-A15 and A16-A31,
ALE, I_O/memory
S0, S1, Read (RD),
WR, Ready

SID, SOD

HOLD & HLD_A

from MP
memory

3) Control

of the
control
peripheral
data
location

→ Mem

→ I/O

→ I/O

→ opcodes

Control

i) RD (R

pin
read

b) WR (W

pin
write

9) I/O

gi
per

from MP to memory or I/O devices and from memory or I/O devices to MP.

3) Control Bus :- It is a group of control signal required for various operation of the MP. It is used to generate timing and control signals to control all the associated peripherals, MP uses control bus to process data, that is what to do with selected memory location. Some control signals are:

- Memory Read
- Memory Write
- I/O read
- I/O write
- Opcode fetch.

Control and Status Signal

- 1) RD (Read) → When the signal is low on this pin the MP performs memory or I/O read operation.
- 2) WR (Write) → When the signal is low on this pin MP performs writing operation with either memory or I/O.
- 3) I/O IO/M : This statu signal is used to give information of operation to be perform with memory or I/O devices

J0/M	\bar{R}	\bar{W}	operation
0	0	1	memory read (mrd)
0	1	0	memory write (mrw)
1	0	1	I/O read I/oR
1	1	0	I/O write I/oW

Status Signal \rightarrow (S₁, S₀)

S ₁	S ₀	operation
0	0	Halt (No operation)
0	1	write operation
1	0	Read operation
1	1	Opcode fetch (Reading Instruction)

Power Supply V_{cc}: +5V, V_{SS} \rightarrow it indicates ground signal

Frequency Signals X₁, X₂:

X₁, X₂ clock out.

X₁, X₂ - A crystal (RC, LC N/W) is connected at these two pins is used to set frequency of the internal clock generator. This frequency is internally divided by 2.

Clock Out \rightarrow This signal is used as the system clock for devices connected with the microprocessor.

\Rightarrow provide synchronization between CPU & peripheral

Serial I/O

i) SIO \rightarrow

micropro

ii) SOI \rightarrow

from I

Hardware

When

pins (

hardware

There

INTR, I

4.5 >

Software

insert

means

There

RSTO,

RSTO

operations
memory read (MREQ)
memory write (MWRE)
read I/O
write I/O

ion)

Instruction)

icates ground
signal

connected
set
for. This

system
with the

μP & peripheral

Serial I/O port → There are 2 serial signals, SID & SOD

- 1) SID → (Serial Input Data) This pin is used for receiving the data into microprocessor by serially.
- 2) SOD → (Serial output Data) → This pin is used for sending the data from MP by serially.

Hardware and Software Interrupts

When MP receive interrupt signals through pins (hardware) of MP, they are known as hardware interrupt.

There are 5 hardware interrupt in 8085 MP → INTR, RST 7.5, RST 6.5, RST 5.5, TRAP.

4.5 > 7.5 > 6.5 > 5.5 > INTR → priority order

Software interrupts are those which are inserted in between the program which means these are mnemonics of microprocessor. There are 8 software interrupt in 8085 MP RST0, RST1, RST2, RST3, RST4, RST5, RST6, RST7

RST0 RST1 ————— RST7
 ↓ ↓ ↓
 index order

Vectorized and Non vectorized Interrupts

Vector Interrupts are those which have fixed vector address (starting address of subroutine) and after executing these, program control is transferred to that address.

Vector addresses are calculated by the formula

(*) type

Interrupt

TRAP (RST 4.5)

RST 5.5

RST 6.5

RST 7.5

Vector Address

0024H

002CH

0034H

003CH

for software interrupts vector addresses are given by:

Interrupt

RST 0

RST 1

RST 2

RST 3

RST 4

RST 5

RST 6

RST 7

Vector Address

00H

08H

10H

18H

20H

28H

30H

38H

disabled

are either

so they can

INTR, RST

interrupts

Non Mask

cannot b

* TRAP is

gt consis

triggering

failure

RST 1

RST 7

RST 6

RST 5

INTR

Hardware

RESET

signal

- can

be reset

reset

RESET

Note: INTR is the only one non-vectorized interrupt in 8085 up

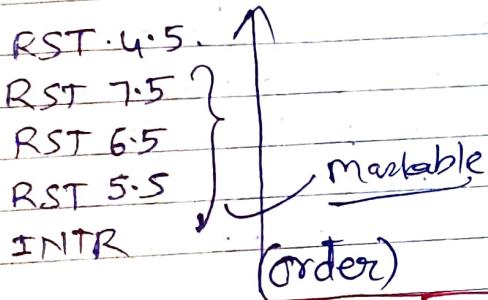
⇒ Maskable and Non-maskable interrupts or
Maskable Interrupts are those which can be

disabled or ignored by the MP. Then interrupt are either edge-triggered or level triggered. So they can be disabled.

INTR, RST 7.5, RST 6.5, RST 5.5 are maskable interrupts in 8085 MP.

⇒ Non maskable interrupts are those which cannot be disabled or ignored by MP.

* TRAP is non maskable interrupt. It consists of both level as well as edge triggering and is used in critical power failure conditions.



Hardware interrupt → Software interrupt
(priority order)

RESET IN Reset in Low signal. When the

signal on this pin is low (0), the program counter is set to zero, the buses are floated tristated and the MP unit is reset.

RESET OUT → It is used to reset the other peripheral devices.

Ready Signals It is used to interface to the few peripheral devices (M or S or) to the MP.

It senses whether a peripheral is ready to transfer data or not.

If ready is high (1) the peripheral is ready. If it is low (0) the MP waits till it goes high. It is useful for interfacing low speed devices.

HOLD AND HLDA

Holding a peripheral like Direct Memory Access DMA controller sends the whole request to the MP through this pin to leave the data bus which is being used by MP. It is fastest mode of Data transfer.

HLDA → Hold acknowledgement, by MP to peripheral.

It is a signal which indicates that the hold request has been received. After the removal of a HOLD request, the HLDA goes low.

Internal Architecture of 8085 MP

ALU → An ALU is a digital ckt used to perform arithmetic logic operations. It represents the fundamental building block of the CPU of a computer which includes → Accumulator + Programmable register

It is an 8 bit programmable register. All arithmetical & logical operation perform with contents of Accumulator (Acc) and results stored in ACC.

→ Temporary Register 8-bit Non-programmable register

It is used to hold data using arithmetical & logical operation.

Arithmetic and logic Unit

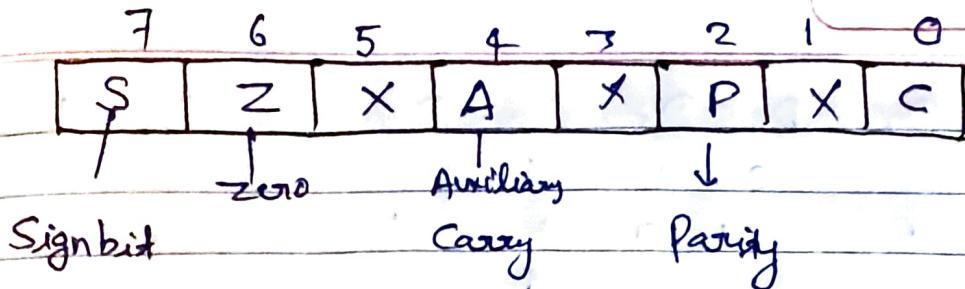
This unit performs the actually numerical & logical operations.

Flags → The flag generally reflects the status of Arithmetic and logical operation

→ Flag register is a special purpose register. Depending upon the value of result after any arithmetic and logical operation the flag bits become set(1) or reset(0). In 8085 CPU, Flag register consists of 8 bits and only 5 of them are useful.

Flag register (Q word)

Date: / / Page: / /



1) D7 - Sign Flag →

$D_7 - \text{MSB} = 1 \Rightarrow \text{-ve (Set)}$

$D_7 - \text{LSB} = 0 \Rightarrow \text{+ve (Reset)}$

Ex- MVI A 30 (Load 30H in register A)

MVI B 40 (load 40H in register B)

SUB B (A = A - B)

These set of instructions will set the sign flag to 1 as $30 - 40$ is a -ve number

2) Zero Flag :- After any arithmetical or logical operation if the result is 0(00)H the zero flag becomes set i.e 1, otherwise it becomes reset 0.

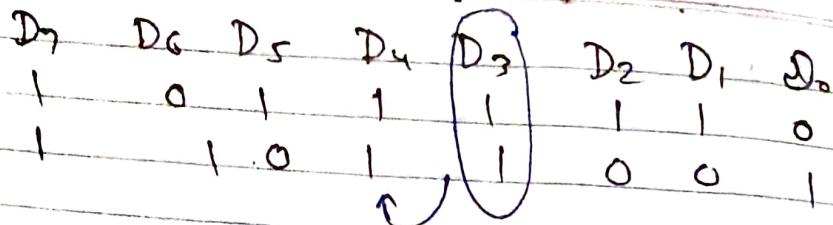
$D_6 = 0$ (Reset)

$D_6 = 1$ (Set)

3) Auxiliary Carry → This flag is used in BCD number system (0-9).

If after any arithmetic or logical operation D(3) generates any carry and passes it to D(4) this flag becomes Set '1', otherwise reset (0).

This is the only flag register which is not accessible by the programmer.



Important point
not al

Imp-2) Amor

intern'

The in

condit

Flags.

X i

The v

as o

instn

Regi

i) Temp

gt

Ex- MOV A 2B (load 2BH in register A)

MOV B 39 (load 39H in register B)

ADD B (A=A+B)

⇒ Parity Flag → If after any Arithmetic or logical operation the result has even parity, an even number of 1 bits, the flag is Set, otherwise Reset.

→ Carry Flag (CY) → Carry is generated when performing n bit operations and the result is more than n bits, then this flag becomes Set 1, otherwise it becomes reset.

→ During subtraction (A-B), if A>B it becomes reset and if (A<B) it becomes set.

Carry flag is also called borrow flag.

$$\begin{array}{r} \text{Ex- } 57 \\ + 96 \\ \hline \end{array} \quad \begin{array}{r} 01010111 \\ 10011110 \\ \hline 11101101 \end{array}$$

(ii) TR-

for

for

$S = 1$
$Z = 0$
$AC = 0$
$P \rightarrow 1$
$CY = 0$

(iii) Gene

cons

Cap

avai

→ Gr

mer

Important points → In each and every instruction not all flag affected at same time

In b-2) Among the 5 flag, AC flag is used internally for BCD Arithmetic operation. The instruction does not include any conditional jump instruction west on the AC Flags.

- X is the flag register unused flip-flop.
- The value of D_1, D_3, D_5 bits should be taken as 0 in the program while use PSW instruction.

→ program Status word

Register Array

i) Temporary DR - It is also called as opend registers (8 bit). It provides operands to the ALU.

ii) TR → (w,z) : This register is not available for users. It is internally used by MP for the internal operation.

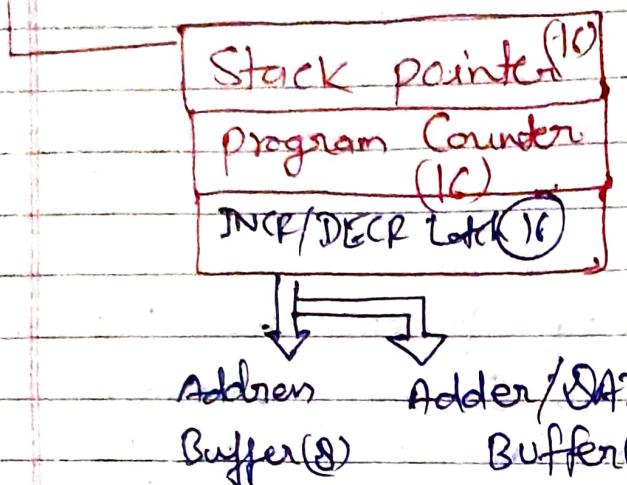
iii) General purpose Register : 8085 MP

consists 6 types of register : 8 bit register GPR. That is B, C, D, E, H, L. These are available for users. → 8 bit each.

→ GPR Put together is called scratch or memory [BC, DE, HL] → Hold data

(4) Special Purpose Register 8085 - up

provide 2 16 bit SPR. That is - PC and - stack pointer.



(a) Program Counter → It is 16 bit register used to hold next memory address while the execution of the program.

(b) Stack pointer → A stack is nothing but the portion of RAM.

- It works on the LIFO concept.
- Stack pointer maintains the address of the last byte that is entered into stack.
- Each time when the data is loaded into stack, stack pointer gets decremented.

Timing And Control Units

- Instruction Register and Decoder
- IR - Hold the opcode of the instruction that is decoded and Instruction

Decoder → The o/p of SR, connected to the Decoder. The Decoder Decodes the Instructions and establishes the sequence of event.

Interrupt Control Unit

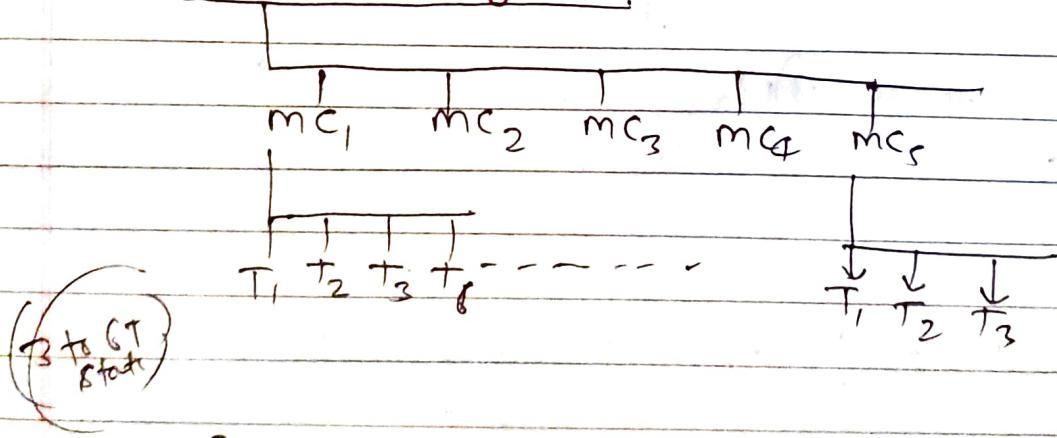
Instruction Set and Data format

Timing diagram related -

Instruction cycle → Fetch cycle + Execution cycle
 (1 to 5 machine cycles) (opcode fetch)

Instruction cycle → The CPU fetches one instruction from memory at a time and execute it - one instruction cycle make consists 1 to 5 machine cycles.

Instruction Cycle



Machine Cycle → Time required by the CPU to complete the operation of accessing memory or I/O device is called as MC.

→ Machine cycle consists 3 to GT state.

another words \rightarrow It is nothing but simply read or write operation.

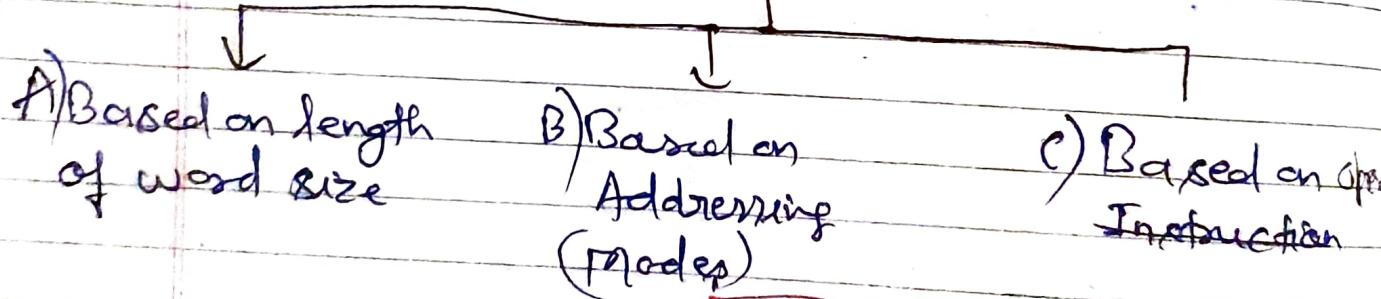
Imp: Different types of M.C. are:

- i) Opcode fetch MC \rightarrow required, 1 to 6 T steps
- ii) Memory read (3T)
- iii) Memory write (3T)
- iv) I/O read (3T)
- v) INTR Acknowledgement (3T)
- vi) Bus Id (3T)

Note 1st Machine cycle in any instruction is opcode fetch cycle.

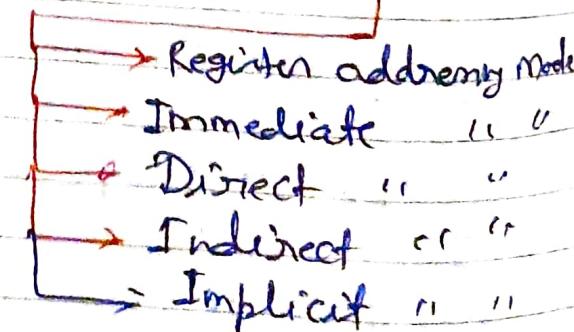
Classification of Instruction Set of 8085

Instruction Set



A) Based on length of word size-

- \rightarrow One byte length
- \rightarrow Two byte length
- \rightarrow Three byte length



(1) Based on Operation Modes.

- Data transfer
- ALU operation
- Logical operation
- Branch control
- Stack, I/O, And Machine control.

→ 1 byte Instruction Set, In 1-byte instruction, the opcode and the operand of an instruction are represented in one byte. These instruction required single memory location.

~~X~~

task - Add the contents of AC to the contents of reg B
Mnemonic - ADD B
Opcode - ADD
Operand - B
Hex code - 80H
Binary code - 1000,0000

(2) 2 byte - I.S. → It uses 1 byte to specifies the operation (opcode) and 2nd byte specifies to the operation data these instruction required at two successive memory location in the memory. 1st bit - opcode & 2nd bit - operand.

(3) 3 byte - I.S. → 1st byte store opcode (operation to the perform, 2nd byte store lower order 8 bit of 16 bits data, and 3rd byte store higher order 8 bit or 16 bit data.

opcode

Low byte data/address

High byte data/Address

B(i) Register addressing Mode :- In this addressing mode the source and destination are general purpose register.

Ex → M_{DD} A, B → Move the content of B register to A register.

ii) Immediate Addressing Mode :- In this mode the data is specified in the instruction itself.

Ex → MVI B, 3EH → Move the data 3EH given in the instruction to B register.

(iii) Direct addressing Mode - In this mode 16 bit address of the operand is given within the instruction itself.

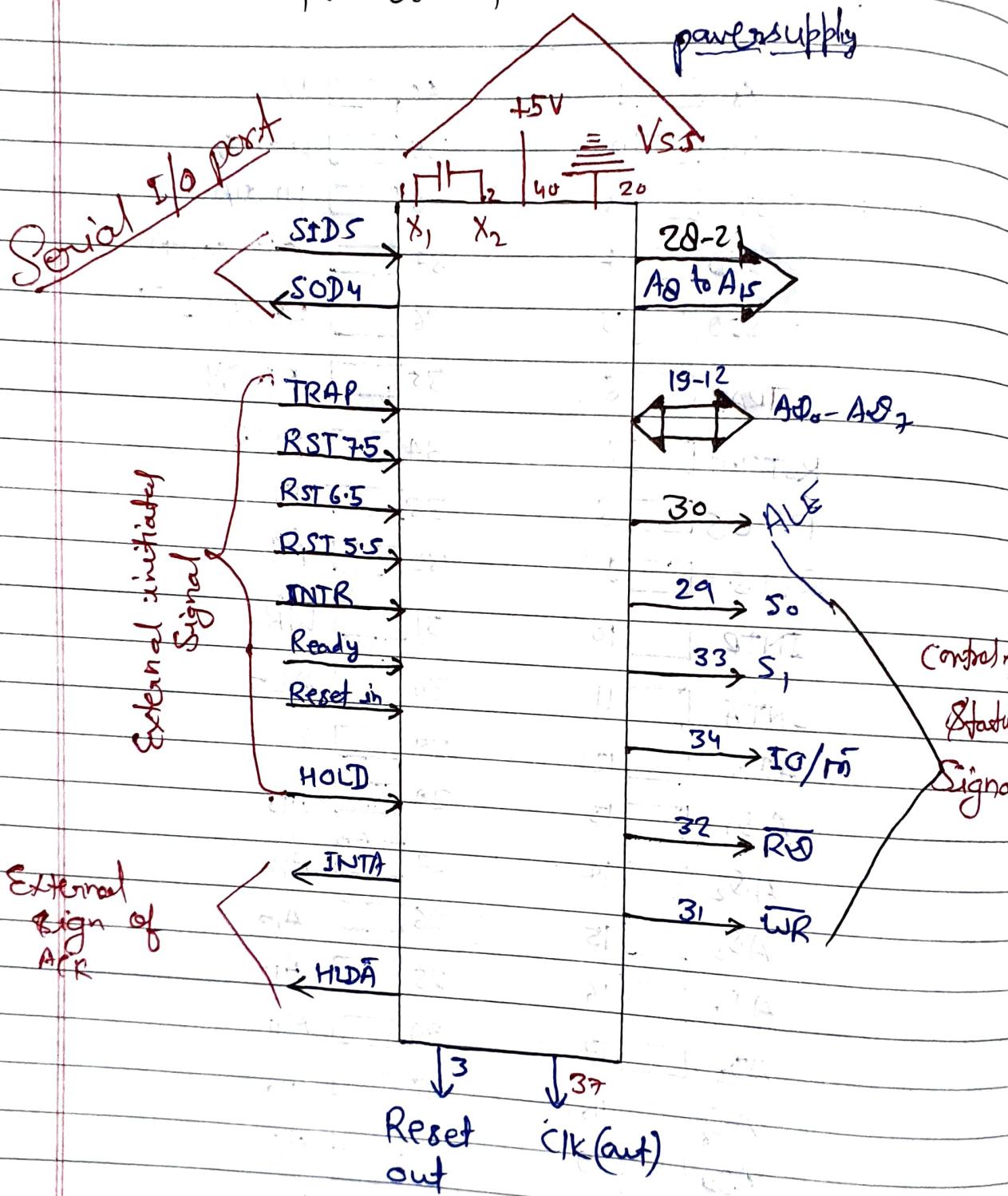
The address of the data is specified in the instruction directly.

Ex - LDA 1050H - Load the data available in memory location 1050H in Accumulator.

(iv) Indirect addressing Mode :-

Total no of output pins in 8085 is 27.
 The o/p pins are A8-A15 (8 pins), ADO-AD7 (8 pins), ALE, S0, S1, RD, WR, HLD/A, INTA, RESET OUT, CLK, SOD
 Date: / / Page: / /

General Pin description of 8085



Key feature of 8085

- (1) It is a 8 bit processor.
- (2) It has 8 data bus line, which is the

(3c) ALE → Address Latch Enable

Used to 8 bits of address of line AD₀-AD₇ into
an external latch.

Date: / /

Page:

language to the machine Language.
Compiler works more directly than the
assemblers.

Control & Timing Unit → The control and timing
unit coordinates with all
the actions of the CPU by the clock and
gives control signals which are required
for communication among the CPU as
well as peripherals.

8085 PIN Description

DMA (Direct Memory Access) is used between memory and I/O.

Date: / / Page: /

DMA allows data transfer between source and destination thus bypassing the CPU.

Microweb

Pins

Address bus & Data Bus
Status signal & Control signal

A0-A15 and ADO-ADS
ALE, $I_{O/MEMORY}$,
S0, S1, Read (RD),
 \overline{WR} , Ready.

DMA Request Signal

Type of Buses:

- 1) Address Bus
- 2) Control Bus
- 3) System Bus

Bus is a group of wires of conductor that is used for communication b/w CPU, memory, I/O device peripheral.

Address Bus: Group of conducting wires which carries address.

Only address bus is unidirectional because data flow in one direction, from CPU to memory or from I/O to CPU devices.

from CPU to memory or I/O devices and from memory or I/O devices to CPU.

3) Control Bus :- It is a group of control signal required for various operation of the CPU. It is used to generate timing and control signals to control all the associated peripherals, CPU uses control bus to process data that is used to do with selected memory location. Some control signals are:

- Memory Read
- Memory write
- I/O read
- I/O write
- Opcode fetch.

Control and Status Signal
Control (Read) → When the signal is low on this pin the CPU performs memory or I/O read operation.
Status (Write) → When the signal is low on this pin the CPU performs writing operation.

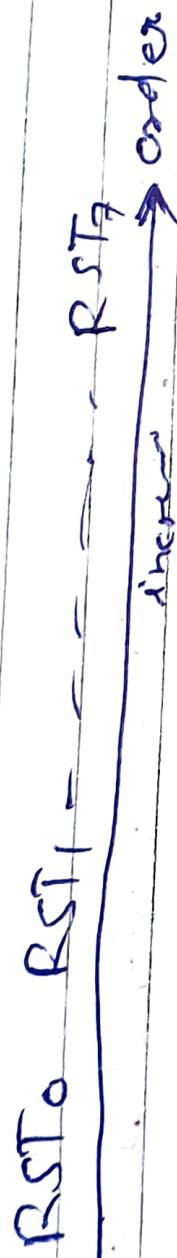
Device that does not interrupt de during can be interrupted instead instead of instruction complete the interrupt the CPU resents the CPU cycle they interrupt the CPU resents the CPU cycle they

lat can be

INTx

Software interrupts

are those which are inserted in between the program which means these are one mnemonics of microprocessor. There are 8 software interrupts in 8085 i/p RST0, RST1, RST2, RST3, RST4, RST5, RST6, RST7.



interrupts the
set by the EI
ate clock pulse
he CPU resets
he INTA cycle
ferences, they

machine cycle)

- ⇒ Trap in edge as well as level triggered.
- RST 7:5 is a five edge triggered interrupt.
- RST 6:5 and RST 5:5 are level triggered interrupt

0. S₁ high, RD,
0, bus and ALE
is the address

Priority order

Interrupts are those which are inserted in between the program which means these are mnemonics of microprocessor.

There are 8 software interrupt in 8085 up RST₀, RST₁, RST₂, RST₃, RST₄, RST₅, RST₆, RST₇

RST₀ RST₁ ----- RST₇ order

* Trap is edge as well as level triggered.

- RST_{7:5} is active edge triggered interrupt.
- RST_{6:5} and RST_{5:5} are level triggered interrupt

HLDA

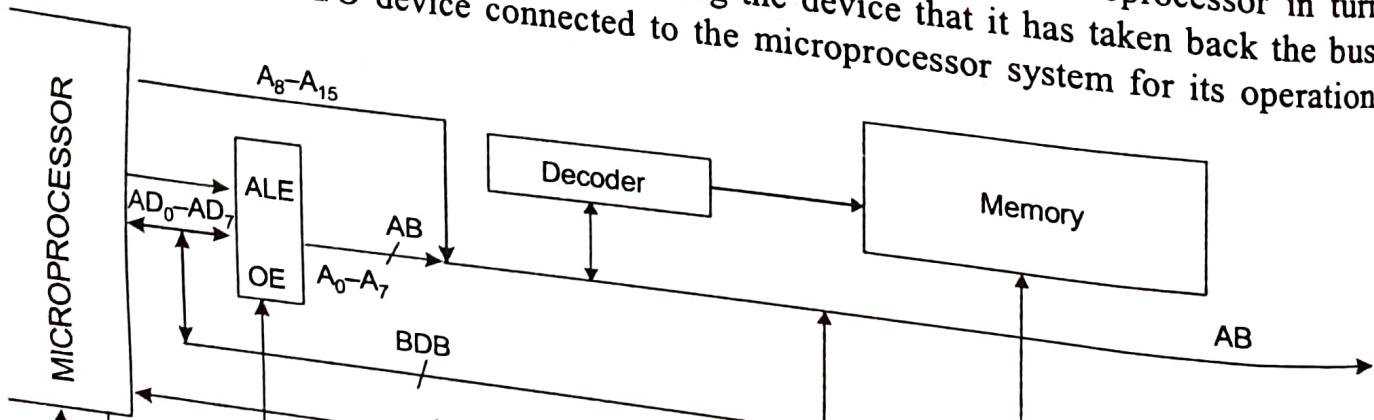
Hold Acknowledge signal goes in hand with HOLD signal of the microprocessor. This signal is made active by the microprocessor by raising it high after getting HOLD signal from the device and completing the present executing machine cycle. This signal is made inactive by the microprocessor after the I/O device has completed the DMA operation and makes the HOLD signal inactive.

DMA Operation of 8085 Microprocessor

The I/O device requests for direct memory access by raising HOLD pin high. The 8085 microprocessor makes the HOLD F/F = 1, and completes the present machine cycle. At the end of the machine cycle the microprocessor checks whether there is a request for DMA. If there is a request, it tri-states the address and data bus and goes into a state called T_{hold} state. It makes the HLDA signal high telling the I/O device that the bus is relinquished. Microprocessor will monitor the HOLD pin till the external I/O makes it low indicating that it has completed DMA operation.

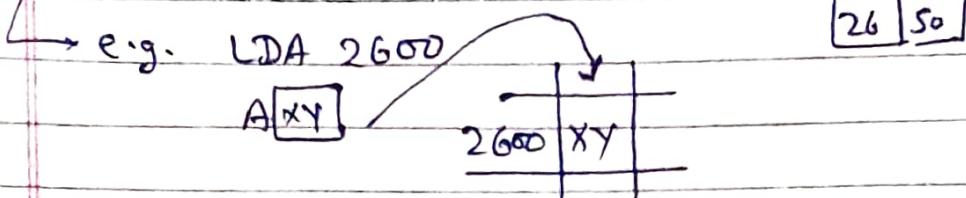
The I/O device looking at the HLDA signal starts its action as soon as it receives the HLDA high. It sends the required address and control signals. The control signal will select the required device and the data transfer takes place between the memory and I/O device. The time taken to transfer one byte of data is only 8 clock period. Unlike execution of many instructions by the processor, to transfer data between the memory and I/O. This results in speed up operation. Once the data transfer is complete the I/O device will relinquish the bus and makes the HOLD signal low.

Once the HOLD signal is low, the microprocessor will continue its operation from where it had left by making its address and data bus active. The microprocessor in turn will make the HLDA signal inactive (low) telling the device that it has taken back the bus. Figure 2.8 shows the I/O device connected to the microprocessor system for its operation.

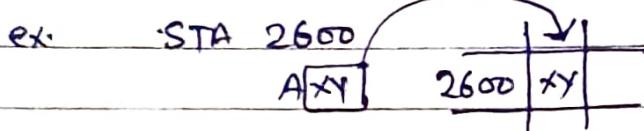


Data Transfer Type

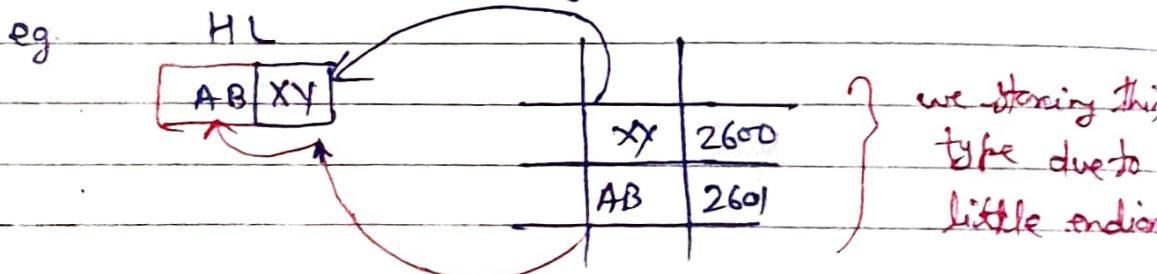
- MOV → Copy from Source to destination
- MVI → Move immediate
- ** LXI → load register pair immediate
- LDA → Load Accumulator directly.



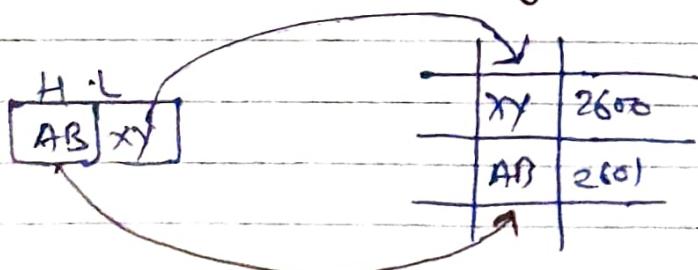
STA → Store Accumulator directly



→ LHLD → Load H and L register direct



→ SHLD: Store H and L register direct.

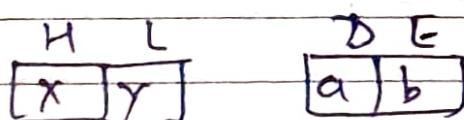


→ STAX: - Store Accumulator indirect

→ LDX: - Load Accumulator indirect

opcode	operand	Byte	Machinecycle	T-State
MOV	Rd/M / RS/M	1	1/2	4T/7T
MVI	Rd/m, 8 bit	2	2/3	7T/10T
LXI (optional)	RP, 16 bit data HL, BC or DE register pair	3	3	10T
LDA	16 bit address	3	4	13T
STA	"	"	"	"
LHLD	"	"	5	16T
SHLD	"	"	"	"
STAX	Rp	1	2	7T
LDAX	Rp	1	1	7T
XCHG	None	1 1	1	4T
PCHL	"	"	"	"

XCHG → Exchange H and L with DE.



PCHL → Load program control by the content of HL.
This instruction basically copies the content of HL register pair into program counter.

* Arithmetic Instruction Set

ADD R $\leftarrow A = A + R$

ADD M $\leftarrow A = A + M$

ADI 8bit data $\leftarrow A = A + 8\text{bit data}$

ADC R $\leftarrow A = A + R + \text{carry}$

ADC M $\leftarrow A = A + M(H.L) + \text{Carry}$

ACI 8bit data $\leftarrow A = A + 8\text{bit} + \text{carry}$

SUB R $\leftarrow A = A - R$

SUB M $\leftarrow A = A - M$

SUI 8bit $\leftarrow A = A - 8\text{bit data}$

SBB R $\leftarrow A = A - R - B$

SBBM $\leftarrow A = A - M - B$

SBI 8bit data $\leftarrow A = A - 8\text{bit}^{\text{data}} - B$

INR R $\leftarrow R = R + 1$

JNR M $\leftarrow M = M + 1$

INX Rp $\leftarrow R_p = R_p + 1$

DCR R $\leftarrow R = R - 1$

DCR M $\leftarrow R = M = M - 1$

DCX Rp $\leftarrow R_p = R_p - 1$

DAD Rp $\leftarrow R_p = R_p + R_p$

R = Register

M = memory

R_p = register pair

ADI : Add immediate to accumulator

ACI : Add immediate accumulator with carry

ADC : Add register to accumulator with carry

SBB : Subtract source and borrow from Accumulator

SUB : Subtract register of memory from Accumulator

SBI : Subtract immediate with borrow

SUI : Subtract immediate from accumulator without borrow

INR : Increment ~~counts~~ contents of memory of register by 1

DCR : Decrement contents of " " " " "

INX i Increment register pair by 1

DCX: Decrement " " by 1

Note: 1) In case of $INT + 4 DCX$ No any flags are affected.

2) $INR + DCR \rightarrow$ in this case flags are affected.

DAD: - Add Rp (HAL)

opcode	operand	byte	MachineC:	T-state
ADD	R/M	1	1/2	4T
ADC	R/M	1	1/2	4/7T
ACI	8 bit data itself	2	2	7T
ADI	"	2	2	"
SBB	R/M	1	1/2	4T/7T
SUB	R/M	1	1/2	4T/7T
SBJ	8 bit data itself	2	2	7T
SUI	"	2	2	7T
INR	R/M	1	1/3	4/10T
DCR	R/M	1	"	"
INX	Rp	1	1	6T
DCX	Rp	1	1	6T
DAD	Rp	1	3	10T
DAA	<u>None</u>	1	1	9T

AC \rightarrow Auxiliary carry

borrow DAA \rightarrow (i) If the value of the low order 4 bit ($D_3 - D_0$) in the AC is > 9 or if AC flag is set, the

LOGICAL INSTRUCTION SET :-

Logical instructions are the instructions which perform basic operations such as AND, OR etc. logical operation works on a bitwise level.

AND →

ANA R	$A \leftarrow A \text{ and } R$	ANR
ANA M	$A \leftarrow A \text{ and } M$	ANM
ANI 8bitdata	$A \leftarrow A \text{ and } 8\text{bitdata}$	AN 8bitdata

OR →

ORA R	$A \leftarrow A \text{ or } R$	AUR
ORA M	$A \leftarrow A \text{ or } M$	AUM
ORI 8bitdata	$A \leftarrow A \text{ or } 8\text{bitdata}$	AU 8bitdata

X-OR →

XRA R	$A \leftarrow A \text{ X-OR } R$	AVR
XRA M	$A \leftarrow A \text{ X-OR } M$	AVM
XRI 8bitdata	$A \leftarrow A \text{ X-OR } 8\text{bitdata}$	AV 8bitdata

CMP R Compares R with A and triggers A/Z/E R
the flag register

CMP M Compares M with A and triggers A/Z/E M
the flag register

CPI 8bitdata Compares 8bitdata with A and triggers A/Z/E ds
the flag register

STC Sets the carry flag

carry ← 1

CMA A = 1's complement of A

$A \leftarrow A$

CMC Complements the carry flag

$cy \leftarrow \bar{cy}$

RLC without carry

$cy \leftarrow 0$

→ Rotate accumulator left without carry

with carry

RAL

RRC

RAR

Rotate accumulator left with carry
 Rotate accumulator right without carry
 Rotate accumulator right with carry

operand

ANA

AND

ORA

opcode

R/M

8 bit

R/M

byte

1

2

1

2

M.C.

Y₂

2

Y₂

2

T.State

4T/7T

7T

4T/7T

7T

Logically OR with AC → 8 bit itself

ORI

Logically OR Immediate

XRA

R/M

1

Y₂

4T/7T

→ OR with accum

XRI

8 bit itself

2

2

7T

→ OR Immediate

CMA

None

1

1

4T

Complement Accumulator

CPI

8 bit itself

2

2

7T

Compare immediate with Acc.

CMP

R/M

1

Y₂

4T/7T

compare with Accumulator

CMC

—

1

1

4T

STC

—

1

1

4T

Set carry

Note: In case of CPI & CMP flag will be affected in some way like SUB only.

Compare and SUB operation almost one same

A-B if A>B

$$Z=0 \\ CY=0$$

cmp. B/m A-B, if (A=B)

$$Z=1 \\ CY=1$$

A-B if A<B

$$\begin{cases} Z=0 \\ CY=1 \end{cases}$$

Date: / /

Page

Rotate Operation :- Rotate is a logical

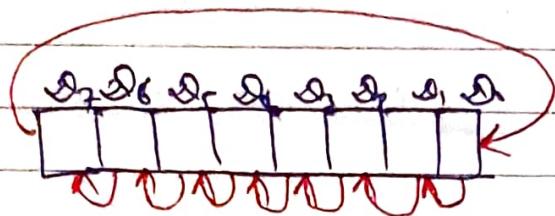
operation of 8085 MP. It is a 1 byte instruction. This instruction does not require any operand after the opcode. The rotate instruction is used to rotating the bits of accumulator.

There are 4 types of Rotate instruction

- 1) RLC [Rotate accumulator left without carry]:

In this instruction, each bit is shifted to the adjacent left position.

opcode	operand	Byte	T-Cycle	T-State
RLC	None	1	1	4T



Bit D₇ is placed in the position of D₀ and so on.

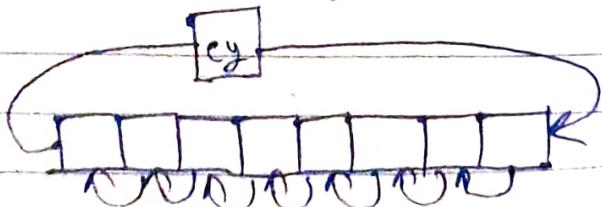
Ex-1) A = 10101010; CY = 0

after RLC A = 01010101; CY = 1]

after 2nd RLC A = 10101010; CY = 0]

- 2) RAL (Rotate accumulator left through carry): each bit is shifted to the adjacent left position. Bit D₇ becomes the carry bit and the carry bit is shifted into D₀. Carry flag CY is modified according to the bit D₇.

opcode	operand	1	1	4T
RAL	None	1	1	4T



Ex-1

$$A = [D_7 | D_6 | D_5 | D_4 | D_3 | D_2 | D_1 | D_0]$$

$$A = 10101010 ; \text{ CY} = 0$$

after 1st RAL

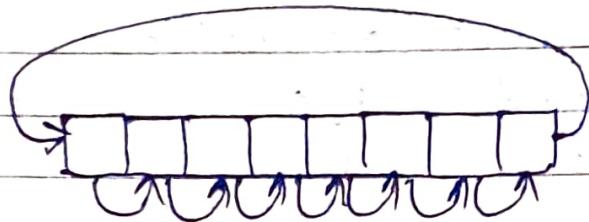
$$A = 01010100 ; \text{ CY} = 1 \quad \left. \right\}$$

after 2nd RAL

$$A = 10101001 ; \text{ CY} = 0 \quad \left. \right\}$$

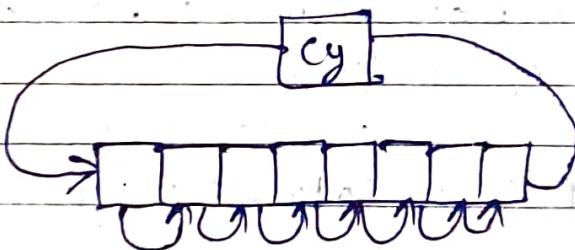
③ RRC: (Rotate accumulator right) :-

[RRC	None	1	1	4T
-------	------	---	---	----



④ RAR: (Rotate accumulator right through carry)

[RAR	None	1	1	4T
-------	------	---	---	----



Imp Note - Nibble or byte can be interchange by executing RRC & RLC 4 times, after 8 times we get the same value.

Branch Control Instructions:

Brnch

Un-conditional

Conditional

- JMP 16 bit address
- CALL 16 bit ..;
- RETURN
- RST_n (n=0 to 7)
- PCHL (Copy HL values in Pr)

opcode	operand	Meaning	Byte	M/c	T-Sck
JMP	16 bit address	Jump un-conditionally	3	3	10T

The program sequence is transferred to the memory address given in the operand memory location specifying to the 16 bit add.

Conditionally Instructions.

Jump (JMP)

Call (CALL)

Return (RET)

opcode	Description	Flag Status	operand
JC	Jump on carry	CY=1	
JNC	Jump on no carry	CY=0	
JZ	Jump on zero	Z=1	
JNZ	Jump on no zero	Z=0	
JP	Jump on positive	S=0	
JM	Jump on minus	S=1	
JPO	Jump on parity even	P=0	
JPE	Jump on parity odd	P=1	

JUMP

The program sequence is transferred to the memory address given in the operand based on the specified flag of the PSW.

CALL

opcode	Description	operand	
CC	Call on carry		
CNC	Call on no carry		
CP	Call on tve		
CM	Call on minus		
CZ	Call on zero		
CNZ	Call on no zero		
CPE	Call on parity even		
CPO	Call on not parity odd		

The program sequence is transferred to the memory address given in the operand. Before transferring the address of the next instruction after CALL is pushed onto the stack.

Unconditional Subroutine CALL

Return (RET)

	Description	
RC	Return on carry	16 bit address
RNC	Return on no carry	"
RZ	Return on zero	"
RNZ	Return on no zero	"
RP	Return on tve	"
RM	Return on -ve	"
RPO	Return on parity odd	"
RPE	Return on parity even	"

In case of conditional, if condition is true, then go with the even instructions.

RET → The program sequence is transferred from the subroutine to the calling program.

JUMP Conditional -

opcode	operand	bytes	M-cycle	T-state
JC JNC	16 bit add. " " " "	3 " " "	2/3 " " "	7/10T " " "
JZ JNZ	" " " "	" " "	" " "	" " "
JM — (-) JP — (+)	" " " "	" " "	" " "	" " "
JE JPO	→ 1 → 0	" " " "	" " "	" " "

JUMP execution of program at the given address.
If address is True

3 Machine cycles and 10T state consumed

PCHL :- Load the program counter with HL contents.
The contents of registers H4L are copied into the program counter. The contents of H are placed as the high-order byte and the contents of L as the lower byte.

opcode	operand	bytes	M.C.	T-state
PCHL	None	1	1	6T

* No flags are affected.

→ Call unconditional Subroutine :-

opcode	operand	byte	M.C.	T-State
CALL	16 bit address	3	5	10T

No flags are affected

The program sequence is transferred to the address specified by the operand. Before the transfer, the address of the next instruction call is pushed to the another instruction.

Conditional call -

opcode :-

call carry	16 bit address	byte	M.C.	T-S
C C → 1	" "	3	2/5	9/10
CNC → 0	" "	"	"	"
LCZ → 1	" "	"	"	"
zern ↴ CNZ → 0	" "	"	"	"
CM → (-)	" "	"	"	"
Sign ↴ CP → (+)	" "	"	"	"
Parity ↴ CPE → even	" "	"	"	"
CP0 → odd	" "	"	"	"

Return Unconditional →

opcode	operand	Byte	MachineCycle	T-state
Return	None	1	3	10T

The program sequence is transferred from subroutine to the calling program.

The Instruction

POP program

No flags are affected

Conditional Return:

in True condition

opcode	operand	Bytes	M.C	T-State
RC	-	1	1/3	G/12T
RNC	-	1	1/3	G/12T
RZ-1	-	"	"	"
RNZ-0	-	"	"	"
RM → E	-	"	"	"
Sign RP → F	-	"	"	"
RPE → even Parity RPO → odd	-	"	"	"

→ XTHL → In 8085 instruction set, XTHL is a mnemonic that stands for exchange Top of stack with HL.

Operand	opcode	Byte	M.C	T-State
XTHL	-	1	5	16T

The content of N registers are exchanged with the stack location pointed out by the content of the stack pointer register.

→ The content of H register is exchanged with the next stack pointer register.

#* STACK, I/O, Machine Control INSTRUCTIONS.

Machine control	I/O related	Stack
SIM	IN - 8 bit pointer addressing	CXI SP 16bit add
RIM		SPHL
EI	OUT - " "	PUSH Rp
DI	"	PUSH PSW
-NOP		POP Rp
HLT		POP RSW
		XTHL

* NOP - No operation are performed

opcode	operand	bytes	MC	T-State
NOP	None	1	1	4T

** These instructions are used to fill in time delay or to delete an insert instruction while troubleshooting.

(S, S+)

* HLT : Halt and interweight state.
Halt the CPU.

opcode	operand	bytes	MC	T.State
HLT	—	1	2 or more	Same

The CPU finishes executing the current instruction and Halt and wait for further instructions.

~~Diff~~ • NOP & HLT Both are same [No operand] However when NOP is executed, PC increases by 1 time, whereas for HLT PC remains same.

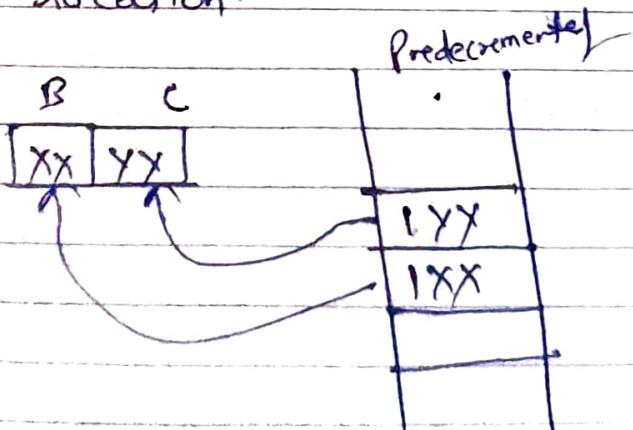
Ans:

* PUSH Rp → This instruction stores contents of register pair Rp by pushing it into two locations above the top of the stack.
 $\boxed{Rp \rightarrow BC, DE, HL, \text{ or } PSW}$

opcode	operand	Bytes	M.C.	T-state
PUSH	Rp	1	3	12T

The content of the register pair designated in the operand is copied into the stack in the following sequences

- 1) The stack pointer register is decremented and content of higher order Register copied into that location.
- 2) SP register is decremented again and content of lower order Register copied into that location.



No flags are modified.

POP Rp :

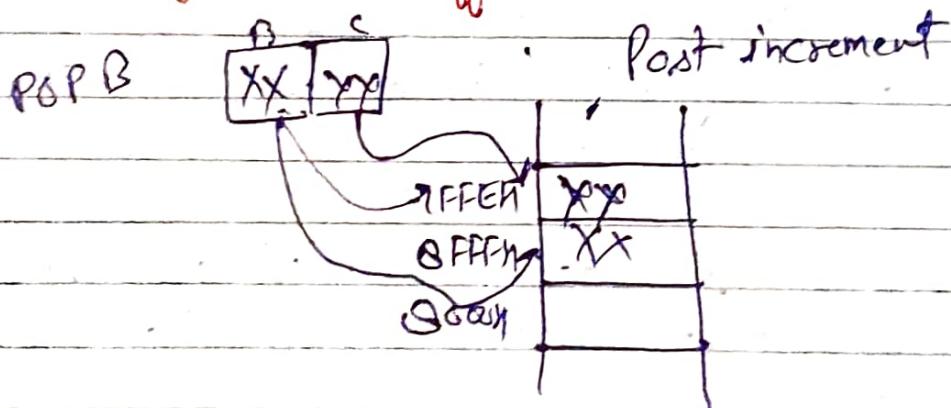
opcode	operand	Byte	M.C	T-state
POP	Rp	1	3	12T

The contents of memory location pointed out by the SP register R or copied to the low order register of the operand.

→ The Stack pointer is incremented by 1 and content of that memory location are copied into the higher order register of the operand.

→ The Stack pointer register is incremented by 1 again.

* No flags are affected.



Note → The content of source stack location are load & altered after push and pop operation.

RSTn : "Restart. n":

opcode	operand	Bytes	M.C	T-state
RSTn	—	1	3	12T

And in this case, n has a value from 0 to 7 only.

Thus the 8 possible RST instruction, e.g. RST0, RST1--- RST7

They are ~~≡~~ 1 Byte call instruction.

No flags are affected.

$$\boxed{\text{RST } n = \text{CALL } n * 8} = (n \times 8)_{10} = (Y)_{16}$$

\Rightarrow SPHL → Copy H & L register to the Stack pointer.

opcode	operand	Byte	m.c	T.S
SPHL	-	1	1	GT

* IN :- Input data to accumulator from a opcode with 8 bit address.

opcode	operand	byte	m.c	T.S
IN	8 bit port address	2	3	10T

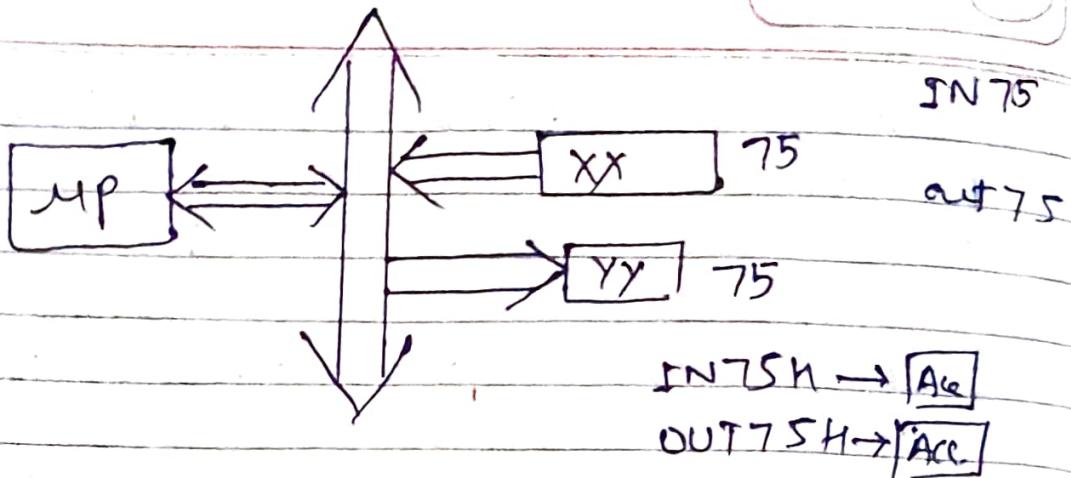
Thus we can have $2^8 = 256$ Input ports and o/p ports are possible in 8085 Microprocessor.

The content of Input port designated into operand are read & loaded into Accumulator.

* OUT → OUTPUT data from accumulator to a port with 8 bit address.

opcode	operand	Byte	m.c	T.S
OUT	8 bit port address	2	3	10T

The content of Accumulator are copied into the output port specified by operand.



* EI → Enable Interrupt

EI None 1 1 1 4T

The interrupt enable F/F is set and all interrupts are enable.

No flags are affected.

Comments :- after a system reset the acknowledge of an interrupt the interrupt enable F/F is reset.

This instruction is necessary to enable interrupt (Except TRAP)

* DI : Disable interrupt

DI None 1 1 1 4T

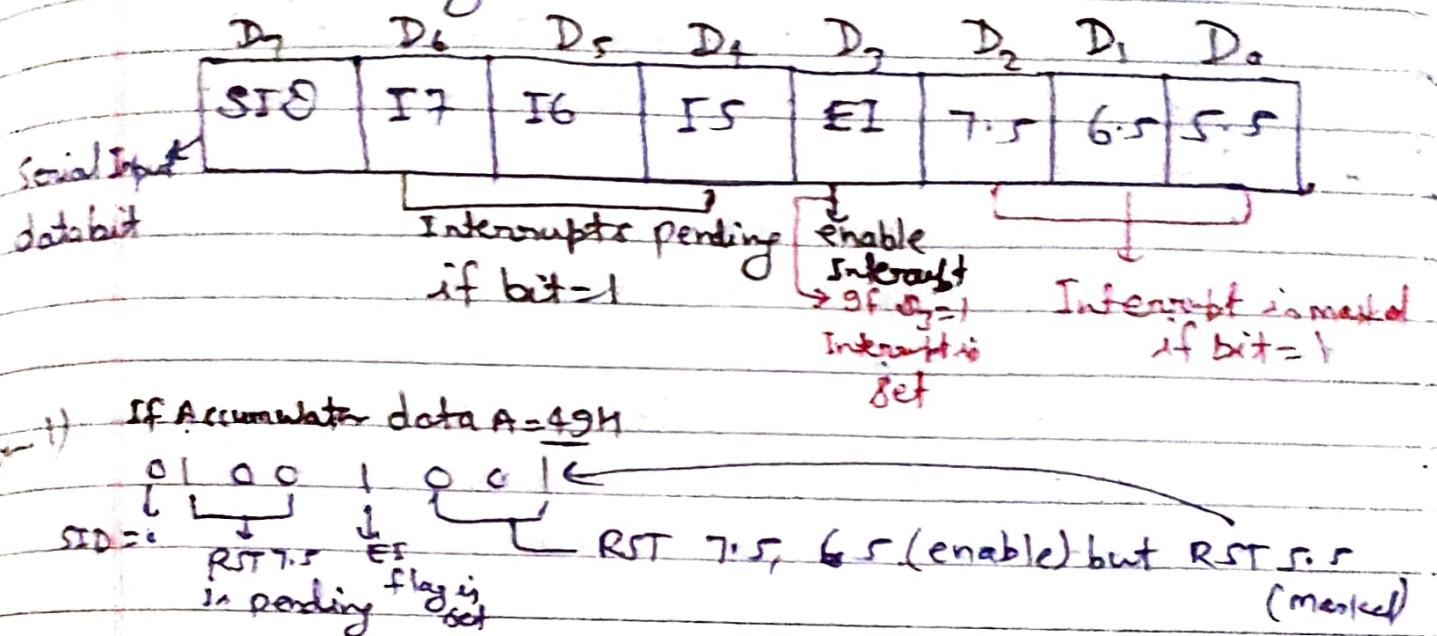
The interrupt enable F/F is reset and all interrupt except TRAP are enable.

• RIM :- Read Interrupt Mask :-

This is multipurpose instruction used to read the status of interrupt 7, 5, 6, 5, 5, 5

opcode	operand	byte	Mc	T-State
RDM	None	1	1 Done	4T

and to read serial data input bit.
This instruction loads eight bit data of accumulator with following Instruction.



SIM ————— 1 1 4T

- Set Interrupt Mask
- This Instruction is a multibitper ins. and used to implement the 8085 Interrupt (RST 6.5, 7.5, 5.5 and serial data output)

Serial output data	SOD	SDE	X	R7.5	MSE	M7.5	M6.5	M5.5
Input	Serial data enable	don't care	Reset	make set enable	R7.5=1			mask interrupt

Interrupt	Triggering	Vector address	Maskable / Non maskable	Instruction
RST 4.5 TRAP	Edge & Level Trigger	0024H	Non maskable	Independent RST 4DI
RST 7.5	edge	003CH	Maskable	controlled by EI 4DI
RST 6.5	Level Triggering	0034H	"	"
RST 5.5	"	002CH	"	"
I.TR	"	Non vector	"	" M

Imp Notes:

Important Points

- 1) Data copying instructions do not affect the flag.
 - 2) Operand PSW (Program Status Word) represent the content of the accumulator and the flag register.
 - 3) The accumulator is high order registers while flag are low order register.
 - 4) Addition & subtraction are performed in relation to the content of accumulator. However the increment and decrement operation can be performed in any register or register pair and memory location.
 - 5) OUT of all instruction PUSH, CALL, RET, RST_n, INX, DCX, SPHL, PCHL, user GT state for fetch machine cycle.
 DAD → GT is a mnemonic which stands for Double ADD.
 ↳ User Bus Idle Machine cycle.)
- NOP operation does not affect any flag.
 - JUMP instruction is a type of immediate address mode.
 - Conditional Jump Instruction allows the CPU to make decision based on certain condition indicated by flag.
 - INX, DCX does not affect any of the flag.
 - Compare Instruction work like Subtraction but content of Accumulator and register does not change.

- Stack and Stack pointer. are two different thing.
- Subroutine is also a program written outside a program.
- CALL & Return instructions. are used for execution of subroutine and after execution return to the main program.

Q-1 LXI SP 8000H
 PUSH B
 PUSH D
 CALL 2900H
 POP D
 Value of SP?

2) MVI A, 10H
 MVI B, 10H
 Back: ADD B
 RLC
 JNC: Back
 HLT

How many times loop will break

Q-2) LXI H, 0A79
 MOV A, L
 ADD H
 DAA
 MOV, HA
 PCHL
 PC = ?

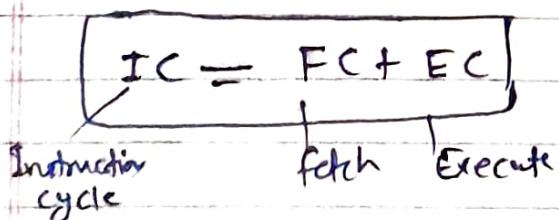
(a) A = 3AH
 B = 49H

Perform SUB B, Status of Flags?

Q) MVI A, 01H
 MOV B, B
 MVI A, C3H
 Sachin: ADDB
 DCRA
 JNC: Sachin
 OUT 10H
 HLT

Q) PUSH PSW \rightarrow SP: A001
 XTHL
 PUSHD
 JMP FC70H
 What is the position of SP?

Timing Diagram: It is one of the best way to understand the process of MP/MC.
 It is the graphical representation of process in steps with respect to time.

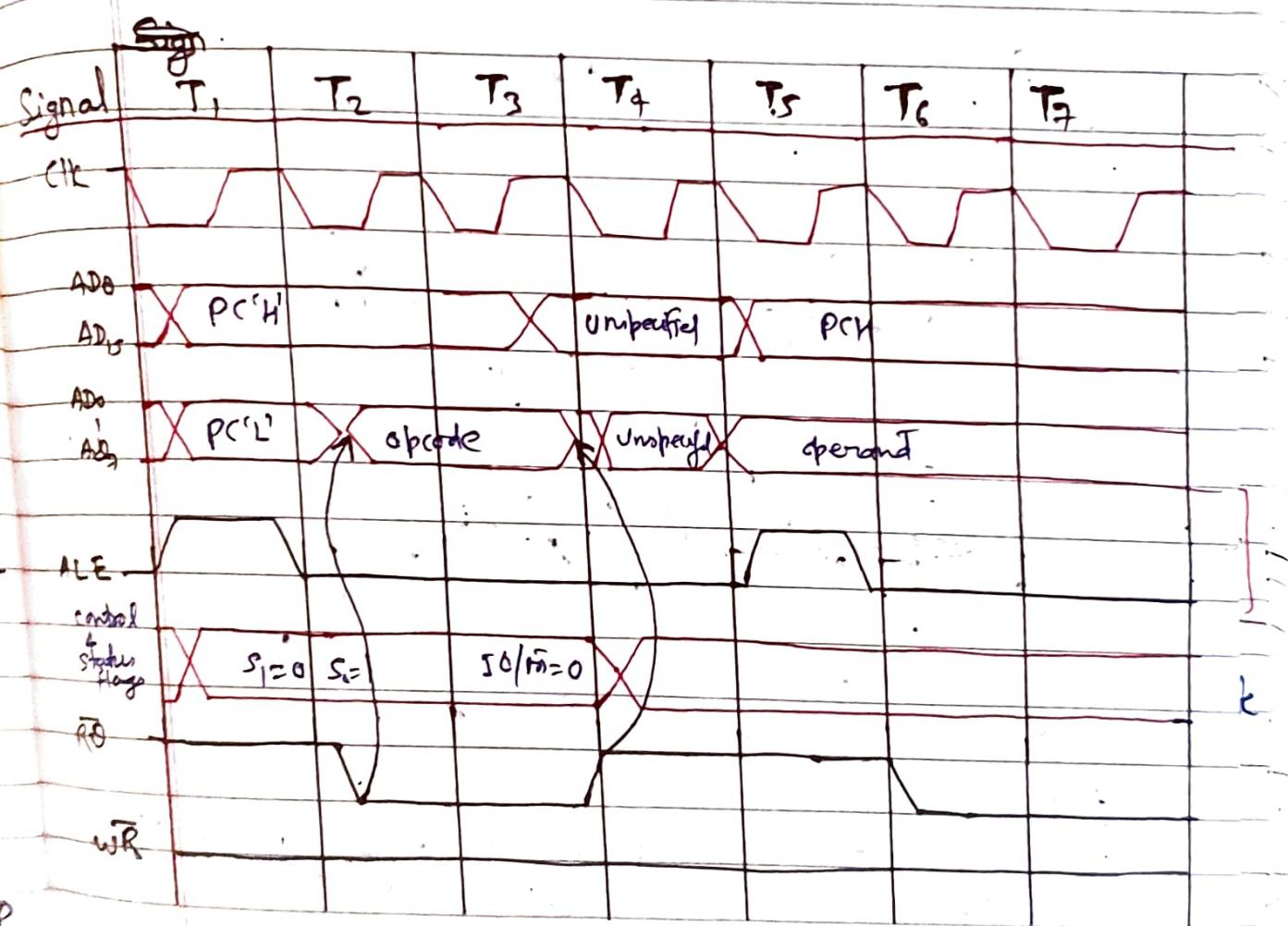


Status flags - control flags

WR	RJ	operation
0	0	Halt
0	1	WR
1	0	RJ
1	1	operate

Status Signal

$\text{IO}/\bar{\text{M}}$	$\text{R}\bar{\text{O}}$	$\bar{\text{W}}\text{R}$	operation
0	0	1	IOR
0	1	0	IOW
1	0	1	MEMR
1	1	0	MEMW

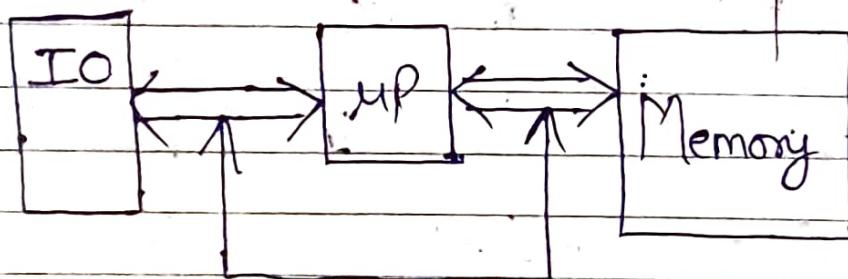
opcode fetch Time Diagram

$\frac{S_1}{\text{WR}}$	$\frac{S_0}{\text{RO}}$	operation
0	0	Halt
0	1	Write
1	0	Read
1	1	Op code

- ALE Signal is always generated at 1st rd pulse
- PC is always increased by 1
- RD/WR → it always generate 2nd clock pulse +ve ending of 3rd clock pulse.

Q- $f = 5 \text{ MHz}$ required time cycles to execute
 Any. $\therefore \left[n = \frac{1}{f} = 0.2 \mu\text{sec} \right]$, $IC \rightarrow 4T$

Interfacing of MP



The primary function of a memory interfacing circuit is to aid the MP in reading and writing a data to the given register of a memory chip.

The interfacing process involves matching the memory requirements with the MP signals.

→ In other words,

Designing logic circuit and writing program to make the MP communication with either Memory or I/O or peripheral.

Interfacing are two types →

- 1) Programmable
- 2) Non-Programmable.

$$\boxed{\text{Memory Capacity} = 2^A \times 2^D}$$

A = number of address line

D = number of data line

$$n = \frac{m_2}{m_1} \rightarrow \text{size of memory to be design}$$

number of chip \rightarrow available size

- 3) If initial address and size of memory is given then formula to calculate final address =

$$\begin{aligned} \text{Final Address} &= \text{Initial Address in Hexadecimal} + (\text{Memory Size})_{\text{Hex}} - (1)_{\text{Hex}} \\ &= (\text{IA})_{\text{H}} + (n \cdot s)_{\text{H}} - (1)_{\text{H}} \end{aligned}$$

- 4) If memory range is given then formula to calculate size of memory

$$\begin{aligned} \text{Memory size} &= (\text{Final Address})_{\text{H}} - (\text{Initial Address})_{\text{H}} + 1(\text{H}) \\ n \cdot s &= (\text{F.A.})_{\text{H}} - (\text{I.A.})_{\text{H}} + 1(\text{H}) \end{aligned}$$

0000H

0001H

} 256

00FFH

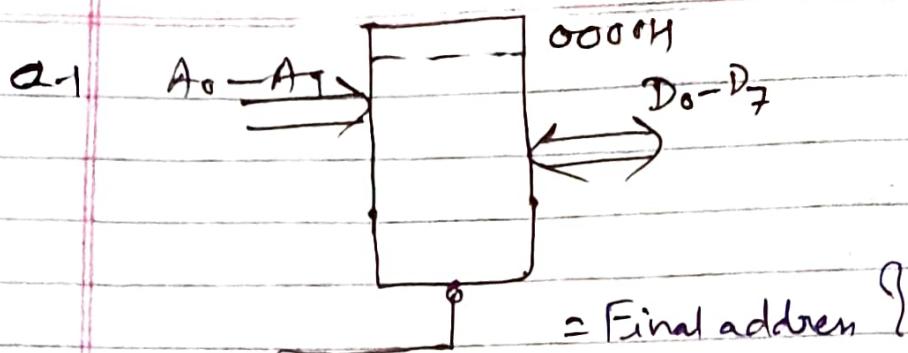
0100H

} 256

01FFH

$$\Rightarrow 256 \times 256 = 65536$$

≈ 64 K

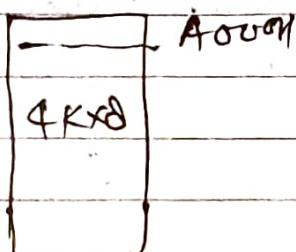


Ans $FA = 0000H + (0400H) - 1$
 $= (03FF)_{16} A$

Q-2 $0000H$ to $9FFFH$? find the memory address

$$\begin{aligned} FA &= (IA)_{16} + (MA)_{16} - (I)_{16} \\ (MA)_{16} &= (FA)_{16} + (I)_{16} - (IA)_{16} \\ &= (9FFF)_{16} - (I)_{16} - (8000)_{16} \\ &= 4K \end{aligned}$$

Q-3

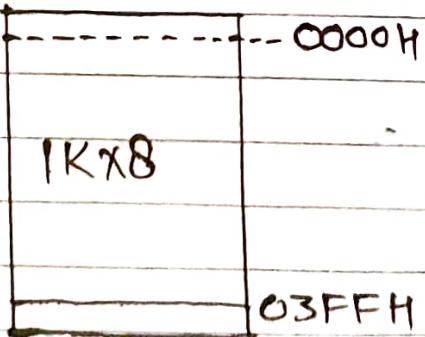


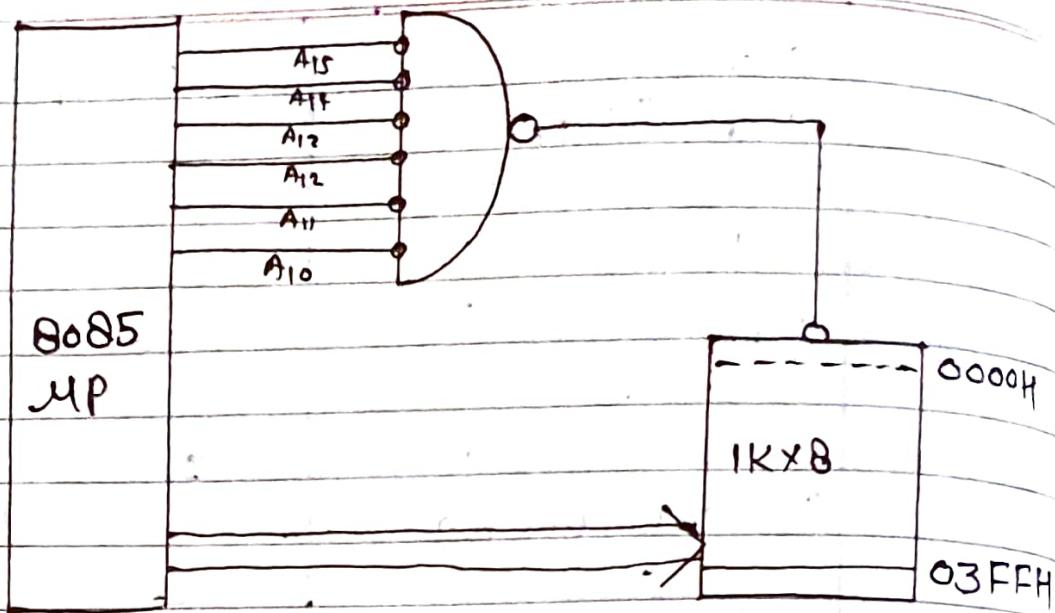
I/O Ports →

There are 2 ways through which \leftarrow converted.

- 1 Memory mapped I/O Port
- 2 I/O

Characteristics	Memory mapped I/O port	I/O Mapped I/O port
1) Device address	16 bit	8 bit
2) Control signal port I/O	MEMRD, MEMWR	IWR
3) Instructions available	Memory related Instruction (MOV, M.R, MOV R, m)	IN OUT
4) Maximum number of devices can be connected	2^{16} (64 K)	$2^8 \rightarrow \text{IN}$ $2^8 \rightarrow \text{OUT}$ $= 512$
5) Execution speed	13T state (STA, LOA)	10T
6) Hardware require- ments	More hardware is needed to decode 16 bit address	less hardware is needed to decode 8 bit address
7) Other	by default each every MP support memory mapped I/O	Not

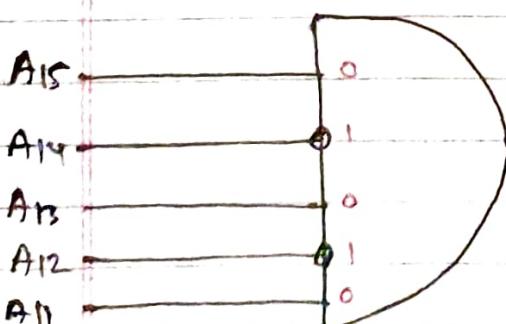


Ex-1

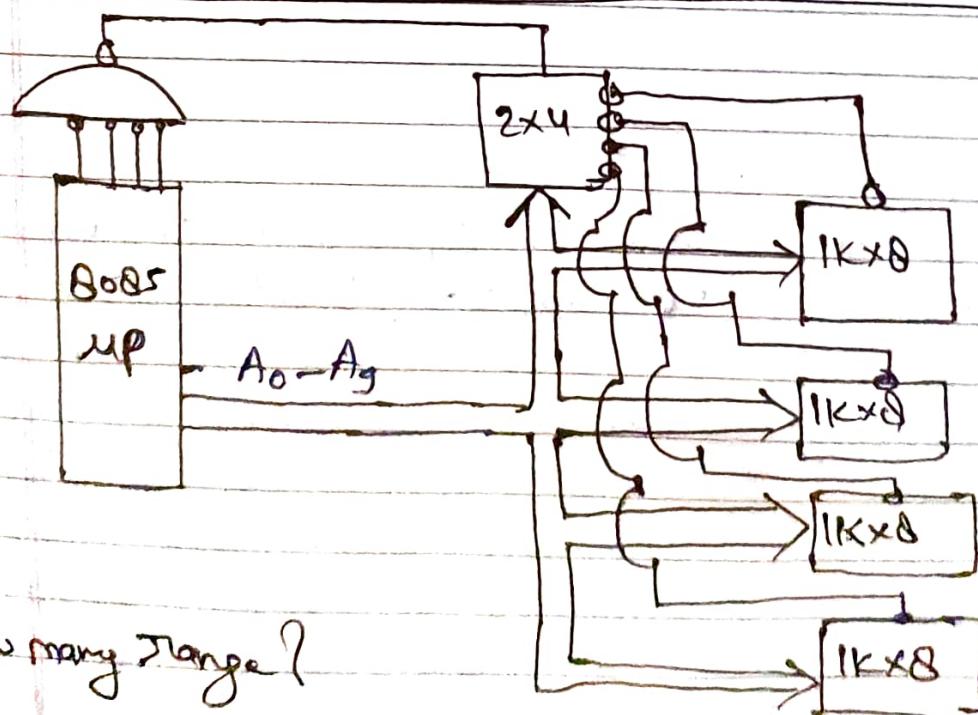
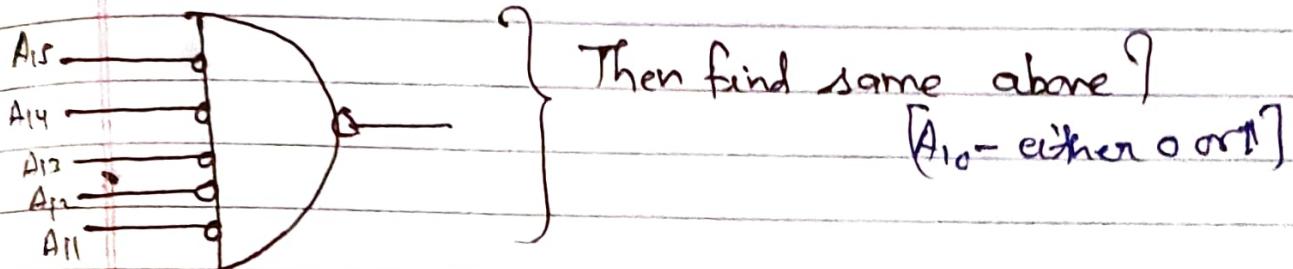
A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

Each memory range is this one E000H → E3FFH

Ex-2 Address lines A₀ to A₁₀ are connected to address pin of memory IC. A₁₁-A₁₅ are used for chip selection as shown in figure, then what is range of address of μP that can be used to communicate with memory and what is capacity of memory and number of page required.



Q-2 In place of Q-2 A_0-A_4 are used to address line and $A_{11}-A_{15}$ are used for chip selection where now figure?



	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
(A)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
(B)	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
(C)	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
(D)	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
(E)	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1

(B) range \rightarrow 0400H \rightarrow 07FFH

(C) range \rightarrow 0800H \rightarrow 0FFFH

(D) range \rightarrow 0C00H \rightarrow OFFFH

Q → How many IC capacity of 256×2 are required to construct 4K byte memory?

$$\text{Memory capacity} = \frac{4 \times 1024 \times 8}{256 \times 2}$$

$$= 64$$

Q - How many memory IC capacity of 4K nibble are required 26.5 byte?

$$\text{Memory capacity} = \frac{4 \times 26 \times 1000}{4000 \times 2} = 13$$

{ always even number }

Interfacing

A) Non Programmable

- (1) 8212 → 8 bit I/O Port
- (2) 74LS373 octal latch
- (3) 74LS244 Octal unidirectional buffer
- (4) 74LS245 Octal bidirectional buffer

(5) Decoder, Encoder, Mux, demux.

Note → We can't change behavior of the circuit without changing the electrical connections.

B) Programmable Type

(1) Intel 8155

(2) Intel 8255 → Programmable peripheral interface/adapter

(3) Intel 8251 → Programmable peripheral communication interface (USART)
It is useful for USART

Universal synchronous Asynchronous Receiver Transmitter

(4) 8253 → programmable Interval Timer

(5) Intel 8257 → Programmable DMA controller.

(6) 8259 → Programmable Interrupt controller

(7) 8272 → Flopy disk controller

(8) 8275 → Programme _____ controller

(9) 8279 → P.P. , Keyboard / display interface

CWR = control word register] Through this relation

MWR = mode word register]

we can change the behaviour of Programmable I/P

8086 MICROPROCESSOR

- Intel 8086 MP is the enhanced version of 8085 MP. It was designed by Intel in 1978.
- It is a 16-bit, N-channel, HMS MP.
- It has 20 address lines.

$$\begin{aligned} 2^{20} &= 1 \text{ MB} \\ 2^{20} &= 2^4(2^{16}) \\ &= 16(64)K \end{aligned}$$

HMS → High Speed Metal Oxide Semiconductor
q memory,
- all registers are 16 bit in length.
- It is built on a single semiconductor chip and packaged in a 40 pin IC package.
- 20 address lines + 16 data lines.

There are 2 types of register :-

- Special purpose register
- General purpose register

GPR: 16 bit GPR AX (Accumulator register), BX (Base register), CX (Counter) and DX. These are further subdivided into 8 bit register.

16 bit register	8 bit high order register	8 bit low order register
AX	AH	AL
BX	BH	BL
CX	CH	CL
DX	DH	DL

- SPR \Rightarrow

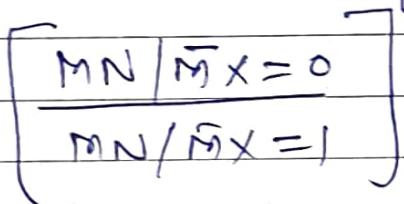
CS	\rightarrow Code Segment Register
DS	\rightarrow Data Segment "
SS	\rightarrow Stack "
ES	\rightarrow Extra " "

Flags S/PSW

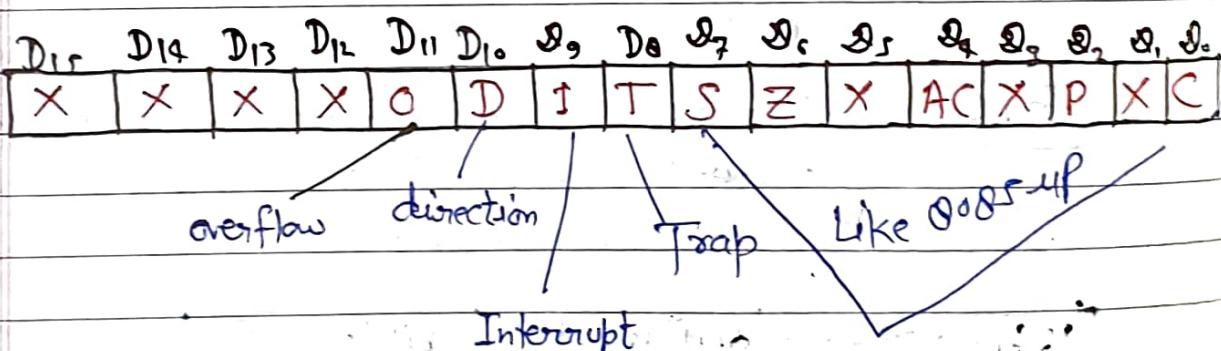
|
Segment register

SP	Index
BP	pointer
SI	register
DI	
IP	

\rightarrow It can be operated in two different modes with the help of control signals.



\rightarrow There are 9 flags in 8086 MP.



\rightarrow Control flags: O, S, Z, AC, P, C

Status Flag: D, I, T

\Rightarrow Again 8086 architecture followed by Non Neumann Architecture but it is segmented memory.

Difference between 8085 & 8086 UP

8085 UP

- 1) It is an 8-bit UP
- 2) It has 16 bit address line
- 3) It has a 8 bit databus
- 4) The memory capacity is 64 KB
- 5) It has five flags.
- 6) It does not support pipelining
- 7) It is accumulator based processor
- 8) The cost of 8085 is low
- 9) In 8085 only one processor is used

8086 UP

- 1) It is a 16-bit UP
- 2) It has 20 bit address line
- 3) It has 16 bit databus.
- 4) Memory capacity is 1MB
- 5) It has 9 flags.
- 6) It supports pipelining.
- 7) It is general purpose register based processor.
- 8) The cost of 8086 is high
- 9) In 8086, more than one processor is used.

10) Memory space is not segmented

11) It does not have multiplication and division instructions

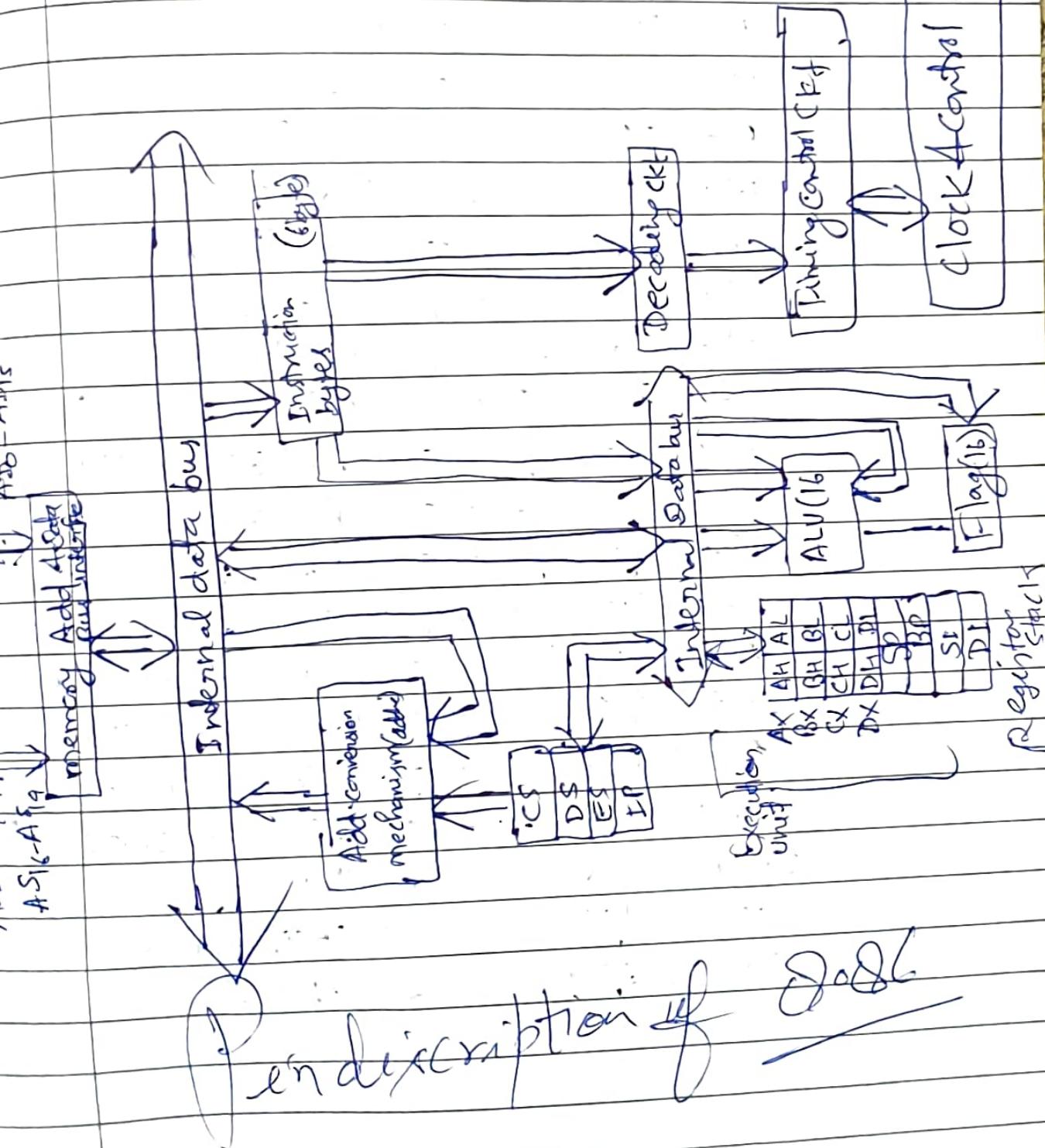
10) Memory Space is segmented

11) It has multiplication and division instructions.

Internal Architecture of 8086

Date: _____

Sr. No. _____



Signature _____

GND	1		VCC
AD ₁₄	2	31	AD ₁₅
AD ₁₃	3	30	A _{16/S₃}
AD ₁₂	4	29	A _{17/S₄}
AD ₁₁	5	28	A _{18/S₅}
AD ₁₀	6	27	A _{19/S₆}
AD ₉	7	26	BHE/S ₇
AD ₈	8	25	MN/MX
AD ₇	9	24	RD
AD ₆	10	23	RQ/GT ₀ → HOLD
AD ₅	11	22	RQ/GT ₁ → HOLD
AD ₄	12	21	Lock → WR
AD ₃	13	20	S ₂ → M/I/O
AD ₂	14	19	S ₁ → SDT/R
AD ₁	15	18	S ₀ → DEN
AD ₀	16	17	QS ₀ → ALE
NMI	17	16	QS ₁ → INTA
INTR	18	15	Test
CLK	19	14	Reddy
GND	20	13	Reset