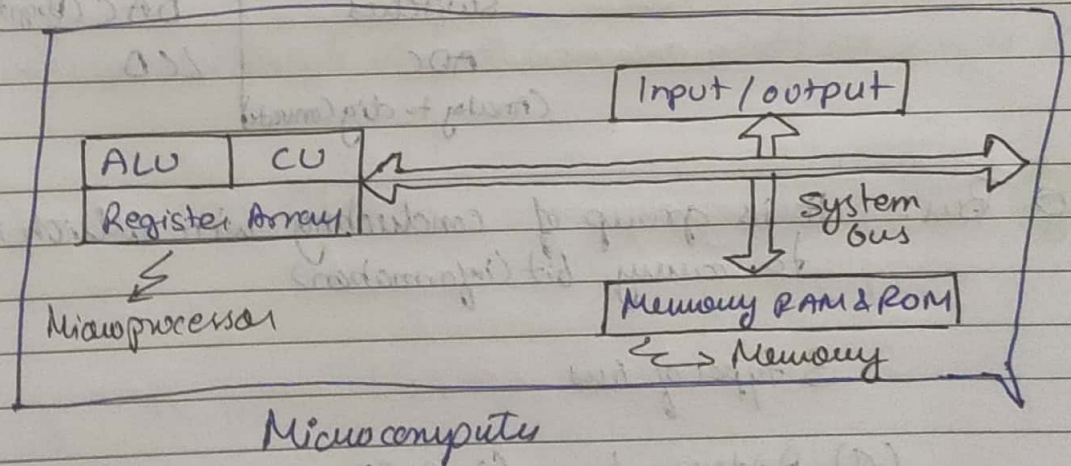


Microprocessor TCS-403

⑥ Microprocessor :- Its a CPU in a single chip. It contains registers (as internal memory), CU (to control all associated peripherals connected with MP) and ALU (to perform arithmetic & logical operations).

* Its a multipurpose, programmable, clock driven semiconductor device which takes input in binary form and process according to given instructions & generate output in binary form.

⑦ Microcomputer :- A digital computer whose CPU is a microprocessor is called a microcomputer.



⑧ ALU :- Various computing functions are performed on data such as addition, subtraction, logical operations (AND, OR and exclusive OR).

⑨ Register Array :- Used to store data temporarily during execution of program & are accessible to users through instructions. RA are identified by B, C, D, E, H & L.

⑥ Control Unit:- Provides the necessary timing and control signals to all the operations in the computer.

* Controls the flow of data b/w μ p and memory & peripheral

⑥ Memory:-

→ ROM (read only memory) (code memory)

→ RAM (random access memory) (data memory)

⑥ Input/Output

devices :-

INPUT

Keyboard

Switches

ADC

(Analog to dig Convert)

OUTPUT

LED

DAC (Digital to Analog Conv)

LCD

⑥ Bus:- It is group of conducting wire which is used to carry bit (information)

3 types of bus

(a) Address bus (in 8085 \rightarrow 16 bit)

(b) Data bus (in 8085 \rightarrow 8 bit)

(c) Control bus (to generate timing & control signals)

Microprocessor

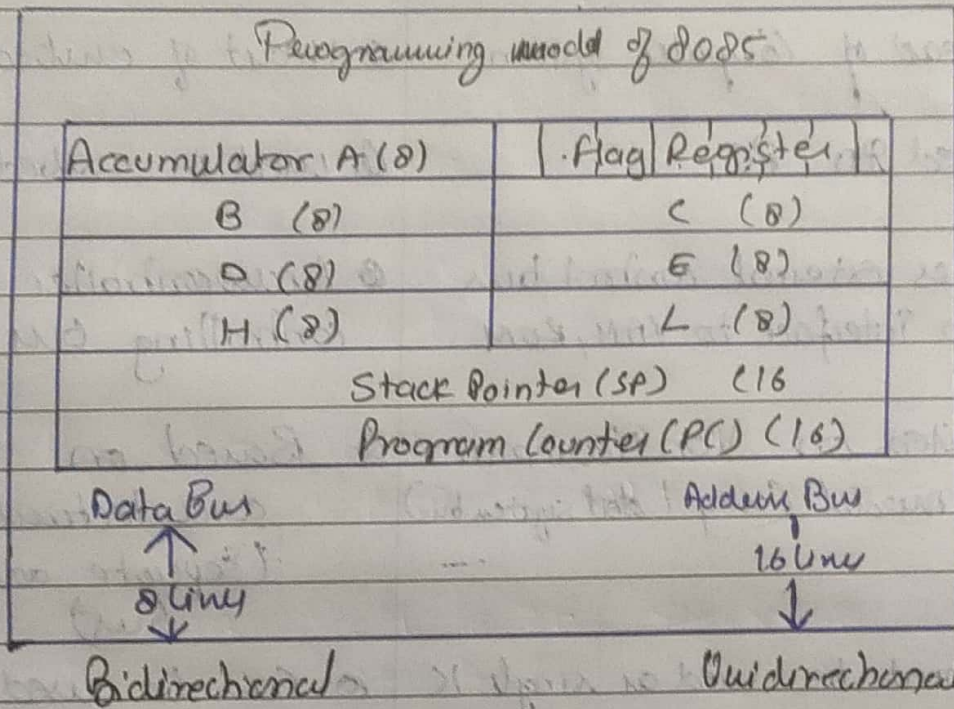
- ① Contains only CPU
- ② Heart of Computer system
- ③ Used in PC
- ④ Uses external control bus to interface to RAM, ROM
- ⑤ Based on Von Neumann model (only 1 system bus)
- ⑥ Cannot be used as single IC

Microcontroller

- ① Contains CPU, memory, I/O all integrated on single chip
- ② Heart of embedded system
- ③ Used in embedded system
- ④ Microcontroller uses internal controlling bus
- ⑤ Based on Harvard architecture (separate address & data bus)
- ⑥ Can be used as single IC

Programming model of 8085

⑥ Information needed to write an assembly language programme is called programming model



$\text{Memory capacity} = 2^{AB}$	$AB = \text{Length of Address Bus}$ in 8085 $AB = 16 \text{ bit}$
-----------------------------------	--

8085 has 64 Kb memory capacity

* Registers of 8085

1 General Purpose Registers :- To store temporary data during execution of a program.

There are :- B, C, D, E, H, L

These can be combined together as register pairs
 BC }
 DE } to perform 16-bit operations.
 HL }

2 Memory Registers :- There are 2 16-bit registers (SP & PC) used to hold memory addresses

(a) Program Counter (PC) :- It holds the memory address of next instruction which has to be fetched. It is incremented by n if size of next instruction is n bytes

(b) Stack Pointer (SP) :- Points to memory location in RAM called the stack. It's always incremented/decremented by 2 during push and pop operation

3 Specific Purpose Registers :-

(a) Accumulator :- Part of ALU. After performing ALU operation result is stored in Accumulator

In 8085 destination operand must be ACC

⑥ Flag Register :-

B ₇	B ₆	B ₅	B ₄	B ₃	B ₂	B ₁	B ₀
S	Z	—	AC	—	P	—	CV

Bit position of various flags in Flag Register of 8085

* It consists of 8 bits and only 5 bits are useful
 * These 5 are set and reset

if Flag = 0 \Rightarrow reset
 " = 1 \Rightarrow set

1. Sign flag (S) :- if MSB (B(7)) = 1 \Rightarrow -ve = set
 if MSB (B(7)) = 0 \Rightarrow +ve = reset

- from 00H to 7F, S = 0
- from 80H to FF, S = 1

2. Zero flag (Z) :- if result is 0 (00)H, Z = 1 = set
 else Z = 0 = reset

- 00H Z = 1
- from 01H to FF, Z = 0

use of zero flag :- while applying loop in Assembly language

3. Auxiliary carry flag (AC) :- Used in BCD (0-9)

if after an operation D(3) generates any carry and passes on to B(4) AC = 1 = set
 else AC = 0 = reset

* 8085 FR cannot be accessed by programmer.

4 Parity Flag (P) :- if after ALU operation if

- no. of 1's = even = set = 1
- else = unset = 0

5 Carry Flag (CY) :- After any operation if carry is generated

CY = 1 = set

else CY = 0 = unset

* during addition if $(n+1)^{th}$ bit is generated during n bit operation CY = 1 = set

* during subtraction if $(A-B)$

if $A > B$ CY = 0 = unset

else if $A < B$ CY = 1 = set

Ex

2000 : MVI A 25H

2002 : MVI B 30H

2004 : ADD B

S = 0

Z = 0

AC = 1

P = 0

CY = 0

25 = 00000101

+ 30 = 00111101

1011000101

4x

A = R1 = 11111111 11110001

B = R0 = 00001111

10000000

S = 0, Z = 0, AC = 1, P = 0, CY = 1

⑥ 8085 Instruction Set

- * An Instruction is a binary pattern designed inside a microprocessor to perform a specific function.
- * The entire group of instructions that a μp supports is called instruction set.
- * 8085 has 246 instructions
- * Each instruction is represented by an 8-bit binary value
- * The 8-bit binary value are called op-code or instruction byte.

Mnemonic: $MOV(A, B) \rightarrow$ operands

⑦ Classification of Instruction Set

- Data transfer Instructions
 - These instructions move data b/w registers or b/w memory and registers.
 - These instructions copy data from source to destination
 - Content of source are not modified.

Opcode	Operand	Description
<u>1</u> MOV	Rd, Rs M, Rs Rd M	Copy from source to destination " " " " Memory " " Memory location to destination

* Size of each operand should be same.

MOV B, C
B = C

<u>2</u> MVI	Rd, data M, data	move immediate 8-bit
--------------	---------------------	----------------------

* Source is always data.

MVI B, 57H
B = 57

<u>3</u> LDA	16-bit Address	Load Accumulator
--------------	----------------	------------------

* LDA 2034H (Content of 2034 location copied to Accumulator)

$A \leftarrow 2034$

<u>4</u> LDAX	B/D Register Pair	Load Accumulator Indirect
---------------	-------------------	---------------------------

* LDAX B

if B = 20, C = 50 then the content of 2050 will be copied to Accumulator.

* When we use register pair in instruction we always mention high order register.

Oprode	operand	Description
<u>5</u> LXI	Reg. pair 16 bit data	Load register pair immediate

* LXI H, (2034H) \rightarrow This represents data if I

* Contains of HL pair register's memory location.

<u>6</u> LHLD	16 bit address	Load H-L register direct
---------------	----------------	--------------------------

* default register pair in 8085 is H-L

LHLD (2040H) \rightarrow Its address
 $L \leftarrow 2040$
 $H \leftarrow 2041$

<u>7</u> STA	16 bit address	Store Accumulator direct
--------------	----------------	--------------------------

* STA 2500H \rightarrow contents of Accumulator will be copied to 2500 memory location

<u>8</u> STAX	Reg. pair	Store Accumulator direct
---------------	-----------	--------------------------

* STAX B

Contents of register Accumulator will be copied to B-C pair address

<u>9</u> SHLD	16 bit address	Stores H-L register direct
---------------	----------------	----------------------------

* SHLD 2550H

Contents of L will be stored to 2550
" " " " " " " " 2551

© When we have to perform 16-bit operation we have to use LHLD & SHLD. to load & store respectively.

<u>10</u> XCHG	None	Exchange H-L with D-E
----------------	------	-----------------------

in this we use two temporary registers.

<u>11</u> SPHL	None	Copy H-L pair to stack pointer
----------------	------	--------------------------------

* SPHL
LX 1H2050H
SPHL
SP = 2050

<u>12</u> XTHL	None	Exchange H-L with to of stack
----------------	------	-------------------------------

* XTHL
Contents of L are exchanged with location pointed out by the contents of SP
Contents of H register are exchanged with the next location (SP+1)

<u>13</u>	PCHL	None	Load program counter with H-L contents
-----------	------	------	--

* Contents of H and L are copied to Program Counter

LXI H 2050

PCHL

PC = 2050

<u>14</u>	PUSH	Reg Pair	Push register pair onto stack
-----------	------	----------	-------------------------------

<u>15</u>	POP	Reg Pair	Pop stack to register pair
-----------	-----	----------	----------------------------

<u>16</u>	OUT	8 bit port Address	Copy data from Accumulator to a port with 8 bit address
-----------	-----	--------------------	---

* OUT $(78H) \rightarrow$ its address
 $78H \leftarrow A$

<u>17</u>	IN	8 bit address	Copy data from Accumulator from a port with 8 bit address
-----------	----	---------------	---

* IN $(8CH) \rightarrow$ Input port address
 $A \leftarrow 8CH$