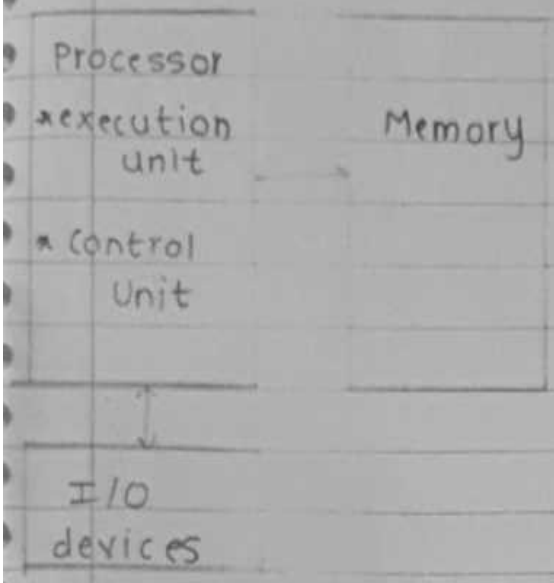
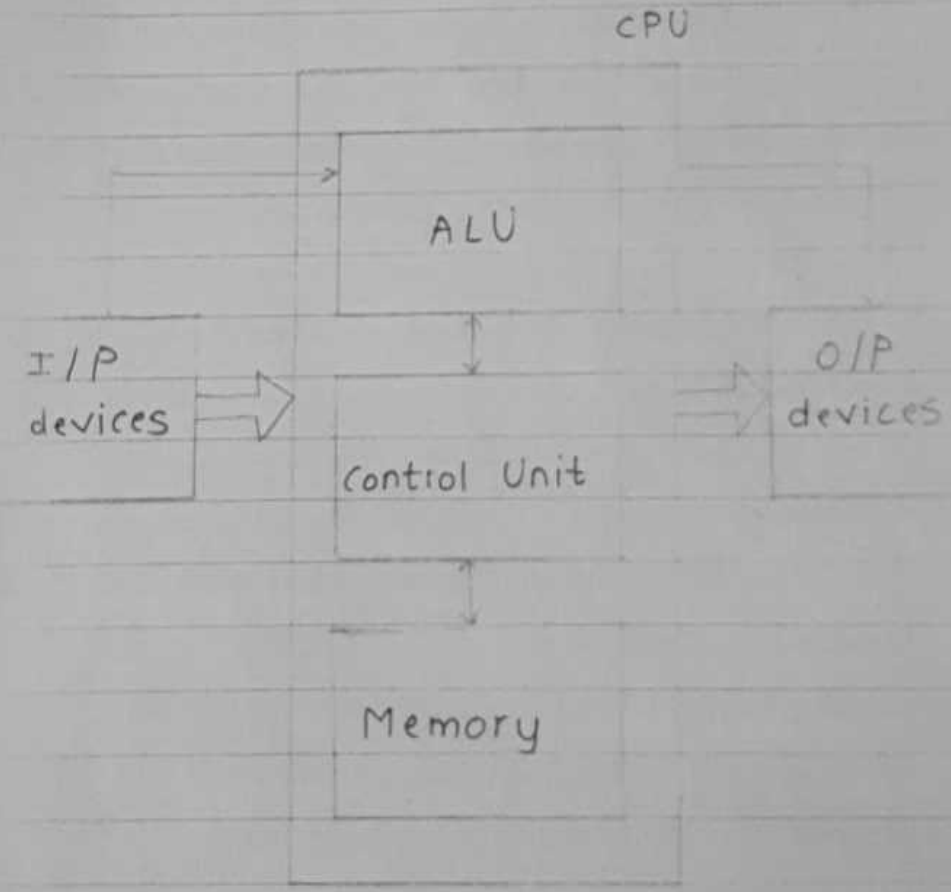
Module 1

Block diagram of Computer System:-





① I/p devices -
This section is responsible for transferring the data in computer system from outside world.

② CPU -
The main function of the CPU is to execute the instruction. It consists of 3 separate parts- ALU, Control Unit, Memory.

Control Unit -
- It controls the whole system by co-ordinating and organising on the operations of computer.
- It takes input from various i/p devices and organises the o/p to various o/p devices.
- It sends data to ALU to carry out operations and data to memory for storing purpose.

ALU -
- It performs all the arithemating and logical operations.

Memory -
- All the information stored inside the computer is in its memory.
- This information could be program or data.
- Permanent information is stored in ROM
- Runtime information is stored in RAM.

Memory — Main memory
Memory — secondary memory
Memory — cache memory

Main memory - It is the actual memory accessed by CPU. Size depends on the address bus.

Secondary memory - It increases the storage space. They are non-volatile and used for permanent data and programs. eg. Hard disk. It is independent of address bus.

Cache memory - It increases the speed of the operation

CPU ⟷ Cache ⟷ Main memory

(3) o/p devices -
It is used to display and record the result and display it to user using user-friendly language.

(4) System Bus -
1.) Control Bus - carries control signal
2.) Address Bus - To carry address
3.) Data Bus - To carry data

8085 processor
16 bit address
8 bit data

2^{16} → locations
8 bit → data per location

Total size = $2^{16} \times 8$

Computer Architecture	Computer Organisation
1) Attributes of the computer system which are visible to the programmer or these attributes that have direct impact on the logical execution of the program is called as computer architecture.	Computer organisation defi the operational units and their interconnection that acheives the architectural specification.
2) Computer architecture attributes include- - no. of bits in architecture - instruction set - memory technology - I/O mechanism	Computer organisation includes - Hardware specifications (These are transparent to the user) - Control signals - Interfacing technical details
3) Basic computer architecture remains same for many years.	Each architecture can have no. of organisational modu with different performance characteristics.
4) Architecture tells 'what' does the system do.	Organisation tells 'how' to implement it.

Register organization

Visible (Accesible)

eg.
- → General purpose registers
- → Address register
- → stack pointer (SP)
- → status Register

Invisible (Inaccesible)

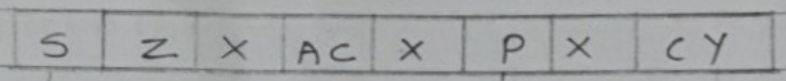
eg.
- → Program Counter
- → Memory Address Reg (MAR)
- → Memory data Reg (MDR)
- → Instruction Reg (IR)

General purpose register -
- These registers are used to store general data during program execution.
- They can be changed by arithematic operations performed by ALU
- eg. A register, B reg, C, D, E reg w.r.t 8085 processor.

Address register -
- This register is used to hold the memory address.

Stack pointer (SP) -
- It is used to hold the address of top of stack.
- A stack is set of memory locations working in LIFO manner.
- When we PUSH an element into the stack SP is decremented.
- When we POP the element from the stack SP will be incremented.

8 bit Status Register of 8085 -



Sign
+ → 0
- → 1

Result is zero or not

Auxillary Carry (Generated or not)

Parity (even/odd)

Carry (Generated or not)

S - If S is logic 0 ⇒ +ve number
If S is logic 1 ⇒ -ve number

Z - If Z is logic 1 ⇒ ans is 0
If Z is logic 0 ⇒ ans is not 0

AC -

eg. 0 1 0 1 | 0 1 1 0
0 1 1 1 | 1 1 0 0

P - If parity is odd ⇒ logic 0
If parity is even ⇒ logic 1

Program Counter -

- It contains the address of the next instructi to be fetched. After every current instruction fetched value of PC is incremented by the processor.
- Program counter value programmer cannot change directly.

Instruction Register -

- It is used to temporarily hold the newly fetched instruction for decoding.

MAR -

- It is the buffer register which is used to store address of current memory transfer
- This has been introduced because buses and CPU work on different frequency (speed)

MDR -

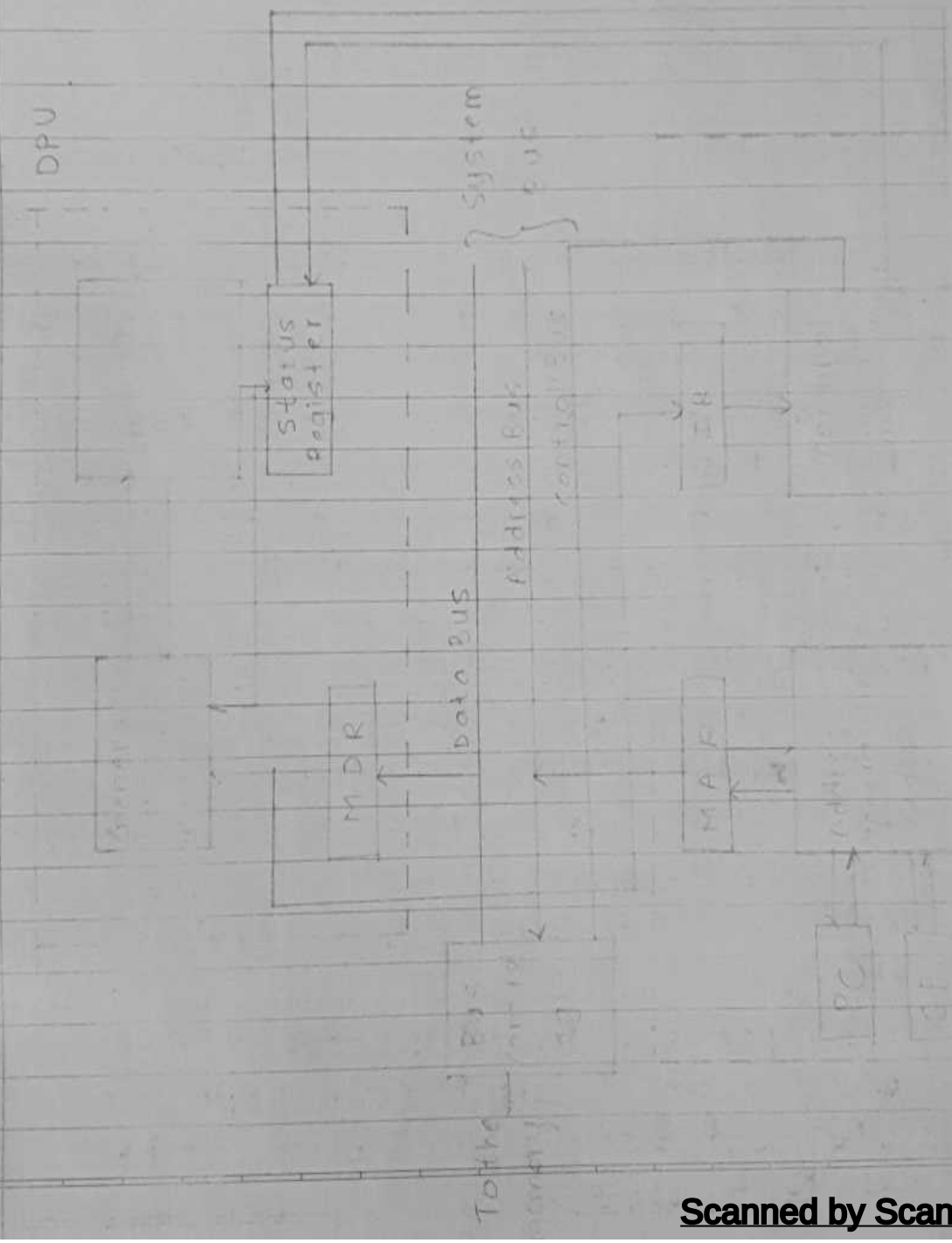
- It contains the data that is currently transferred from memory.

Von Neumann Architecture :-
(Stored program concept.)

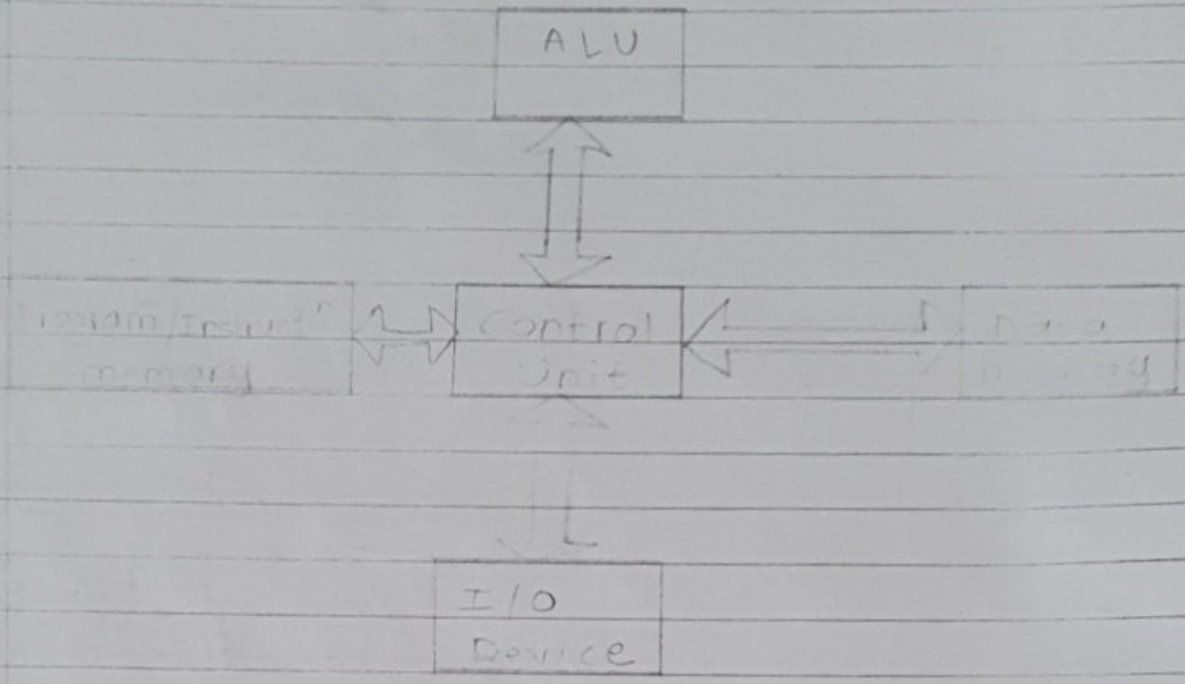
Program control Unit (PCU)	Data Processing Unit (DPU)
→ PC	→ MDR
→ SP	→ General purpose reg.
→ MAR	→ ALU
→ IR	→ status Reg.
→ Control Unit	

- PC provides address to AGL. AGL processes the address to generate physical address which is given to MAR.
- MAR loads the address on the address bus.
- This address is given to BCL which fetches the data on that address and loads it on data bus
- Data bus provides data to MDR. If data is opcode then it is sent to IR. If data is operands it is sent to GPR.
- IR sends opcode to Control Unit and acc. to the instructions control signals are generated.
- Control signals are sent to ALU and operands from GPR are also available to ALU. Hence acc. to the CS operations are performed on the data.
- Result generated by ALU is stored in GPR. It is also set to Status Register. status Register shows the status and sends it to CU for dependent processes.

Stored program concept –
In single memory program/intructions as well as data is stored.



Harvard Architecture :-



Von Neumann	Harvard
- It has single storage system memory for storing data and programs to be executed.	It has 2 separate memories for storing data and programs
- Processor needs minimum 2 clock cycles to complete an instruction. 1 clk cycle - opcode 1 clk cycle - operands	In one single clk cycle processor can complete 1 instruction (simultaneous fetching of opcode and operand).

- Pipelining of instructions is not possible.	Appropriate pipelining strategies can be implemented.
- It has single set of address and data bus between CPU & memory.	It has 2 sets of address and data buses between CPU and memory hence 2 simultaneous memory fetches are allowed.
- Simple in design as only 1 system bus and 1 memory is needed.	Complex in design as 2 system buses and 2 memory are needed.
- Slower in speed.	Faster in speed.
- eg. 8085, 8086 processor	eg. 8051 micro-controller, all advance processors

Addressing modes -

The manner in which operands are fetched for an instruction is called as addressing mode.

① Immediate addressing mode -

- In this mode the operand is specified in the instruction itself.
- Hence when the instruction is fetched the operand comes along with it so the processor can act on it immediately.

eg. MVI A 35 H

(Move immediately the value 35 H to the accumulator)

Advantages -
- Programmer can easily identify the operands.
- Execution is fast.

(2.) Register addressing modes -
- In this mode the operand is specified in the registers.
- These instruction uses the register present inside the processor to supply the data.
eg. MOV B, C
(Move the content of register C to register B.)

Advantages -
- Instruction will be small in size.
- Execution will be very fast as data is already present on the on-chip registers.

(3.) Direct addressing mode -
- In this mode the instruction only contains the address of the operand.
- The processor will then use this address to fetch the operand from the memory.
eg. LDA 2000H
(load the accumulator with the content of memory location 2000H)

Advantage -
- Address of operand is known to user.
- No need to store operand as it is already stored in the memory.

(4.) Indirect addressing mode -

- In this mode instruction only contains name of register pair containing the address of the operands.
- The address of the operand is given indirectly by the registers.
- As the value of registers can be changed, the same instruction can be used to operate on different memory locations.

eg. STAX B & C.

Advantages -

- length will be smaller compared to direct addressing
- Memory locations can be accesed sequentially to store data sequentially by using it in a loop.

(5) Implide Addressing mode -

- In this mode the operand is implied by the instruction.
- Here there is no need to mention the operand separately as the instruction is designed to work with a particular operand only.

eg. STC
(set the carry flag from status register)

Advantages -

- Smallest in size (1 byte) as operand is not present.