

Unit 1

Microprocessor evolution & types

Evolution & types

Generation	Technology	Timeline	Processors	Transistors	CPU Speed	Data Length (bits)
First	PMOS	1971	INTEL 4004	2300	108 kHz	4
Second	N MOS	1976	INTEL 8008	6500	3-5 MHz	8
Third	H MOS	1978	INTEL 8086	30,000	4 MHz	16
Fourth	HC MOS	1985	INTEL 80386	2,75,000	16 MHz	32
Fifth (Latest)	Super Scalar processing	1993	Pentium, Celeron, Dual core Quad core	More than 10 million transistors	3.5 GHz	32/64

Microprogram

Microprocessor is a programmable integrated circuit (IC) that has computing and decision making capability similar to that of Central Processing Unit (CPU) of a computer. means

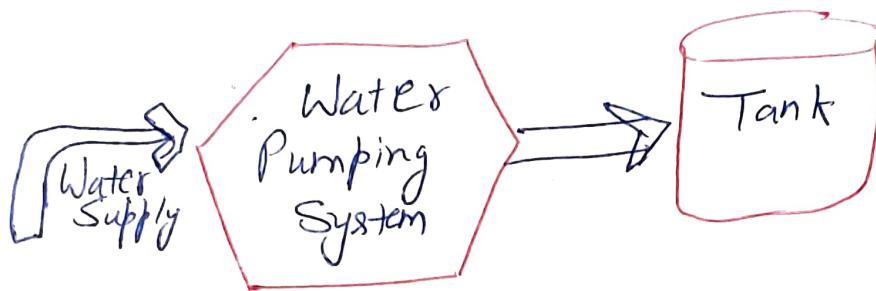
we have to program our microprocessor that's why we have to give some instruction

Introduction

used for automating any system

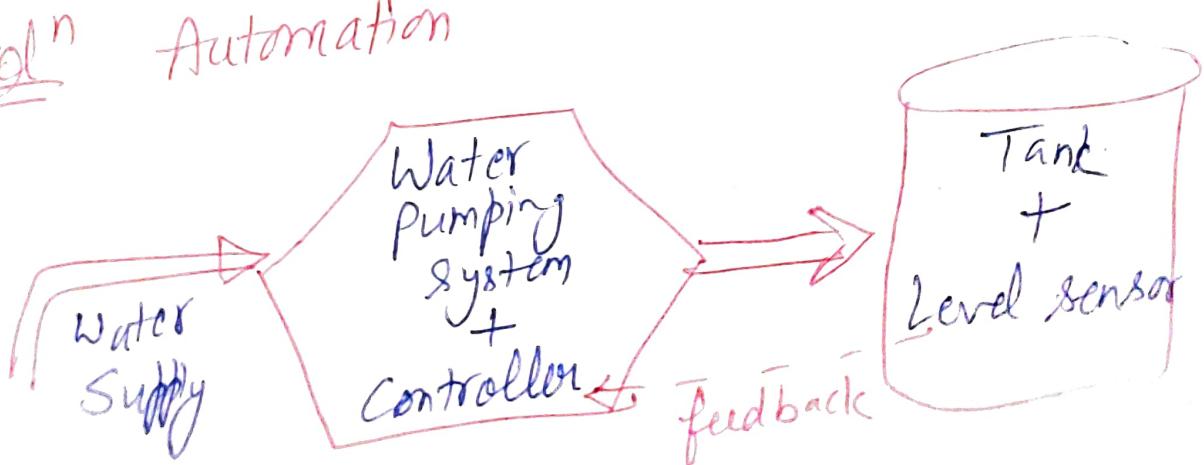
Ex.

Ex of water pumping system at home



- ⇒ switch off the pumping system on judgemental basis
- ⇒ forget to switch off the pump after sometime, leading to overflow of water from tank & hence water wastage
- ⇒ No means to check the level of water in tank

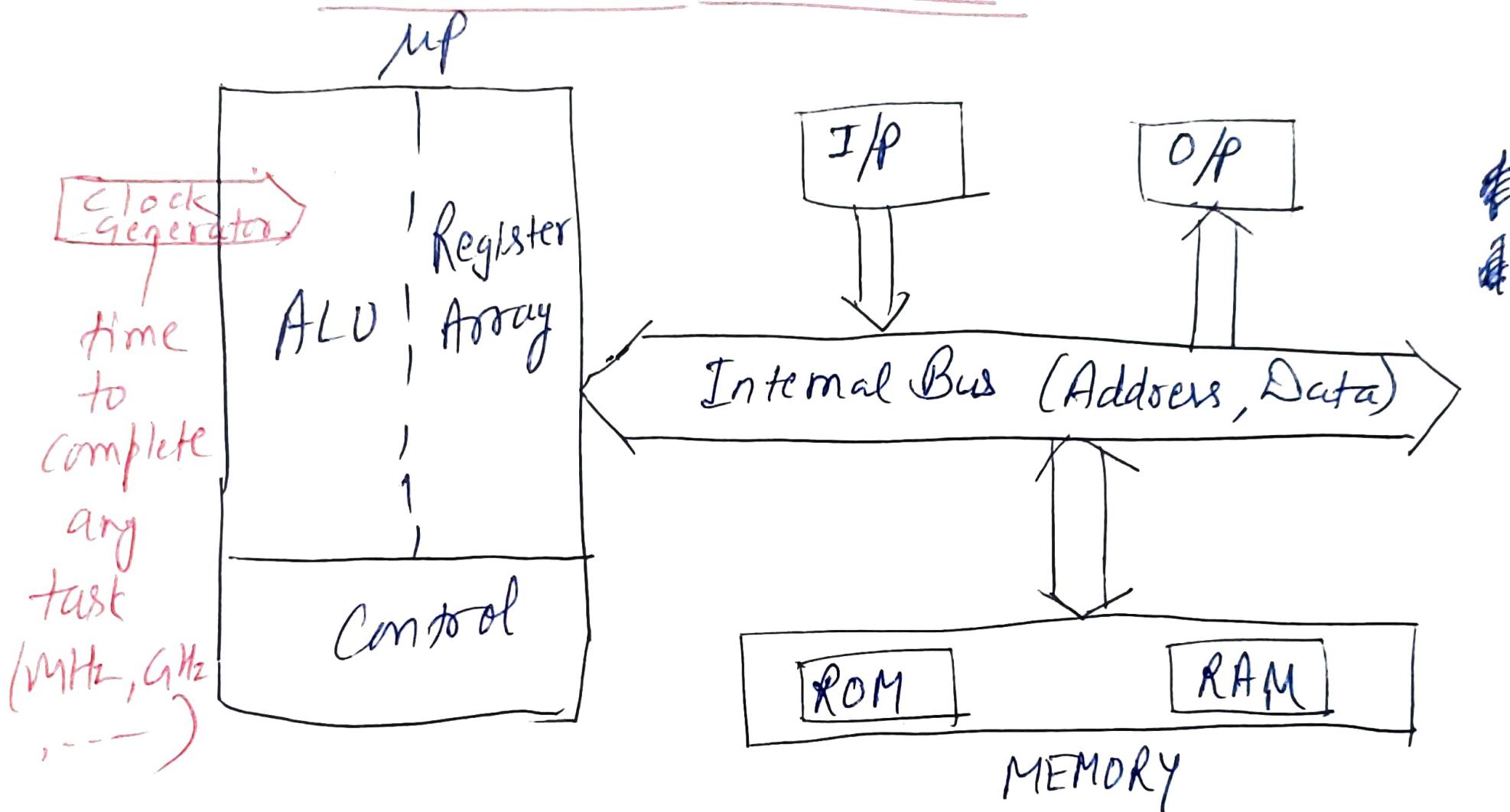
Solⁿ Automation



→ when the pre-defined level of the water reaches in tank, the controller will automatically switch off the pumping system, thereby, reducing water wastage.

This will help to reduce water wastage.

General Architecture

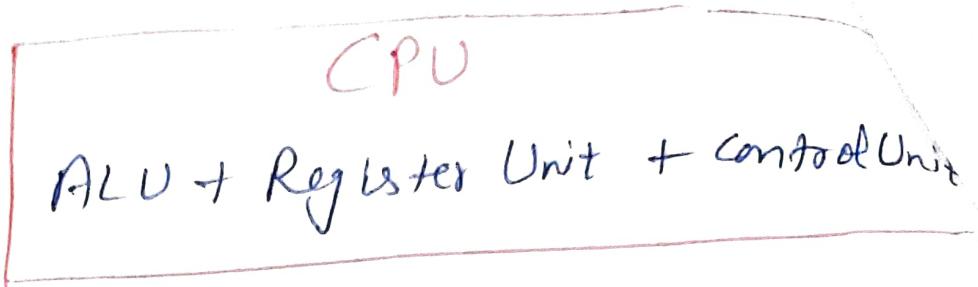


MP

To store the
data temporarily

④

Inig



Controls all the actions taken by microprocessor

Memory

RAM (User memory)

- ⇒ Main Memory
- ⇒ Data memory
- ⇒ Also called, Read Write memory

ROM

- ⇒ Secondary memory
- ⇒ Program memory
- ⇒ Also called, Hard disk drive

I/P + O/P devices

Internal bus (Address, data)

A bus has a wire for each bit & all bits are processed in parallel.

The width of a bus is the no. of signal lines/wires that constitute the bus.

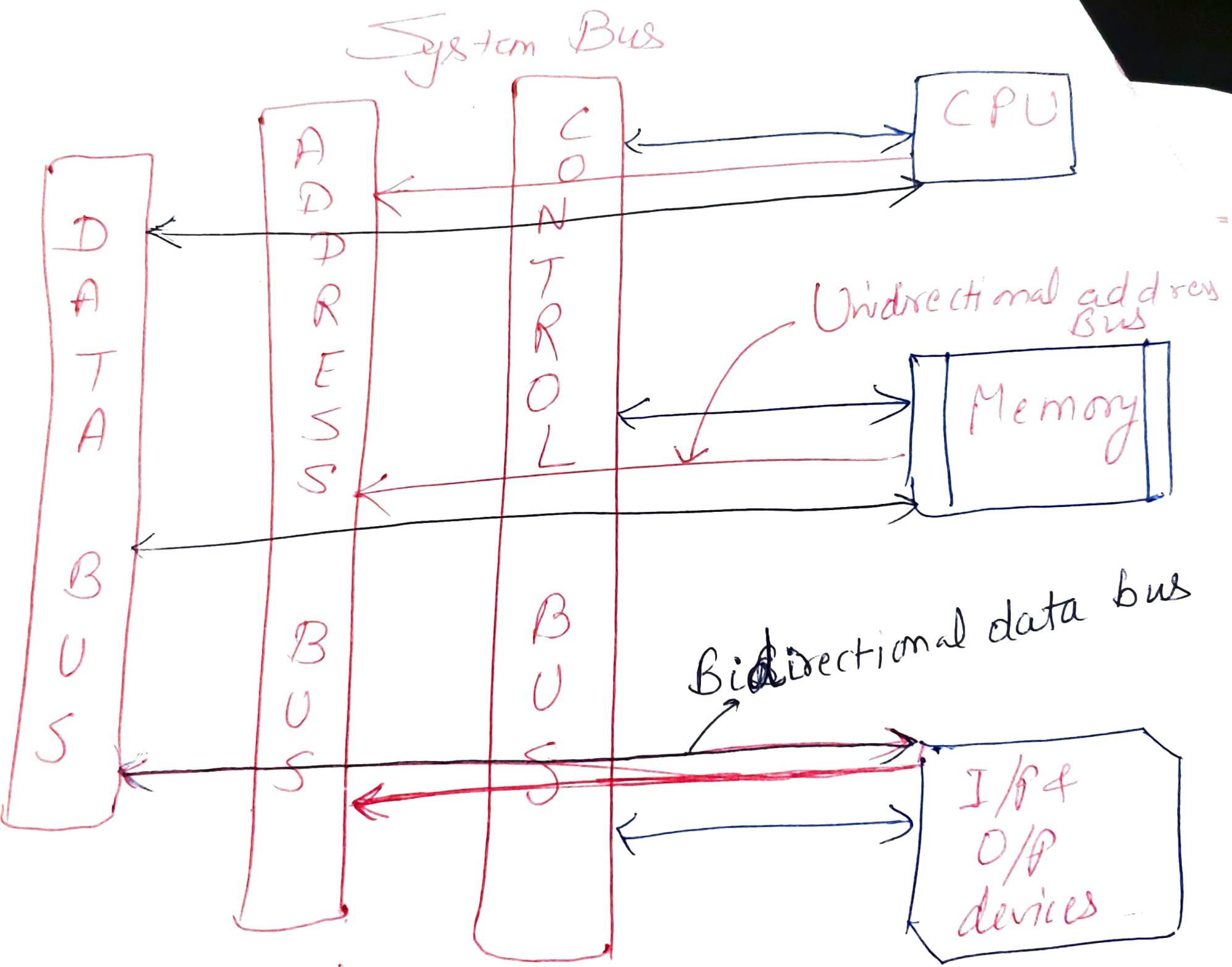
Important points regarding bus system

Bit-size

No. of bits processed simultaneously by ALU of microprocessor

Nibble = 4 bits \neq Byte = 8 bits

Time of processing will increase, if microprocessor perform calculations for numbers more than its bit-size.



Control Unit

(19)

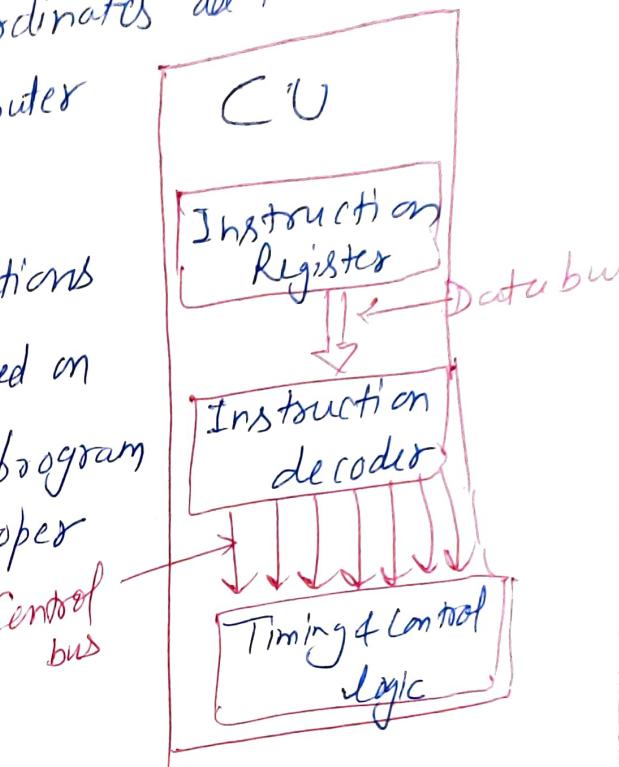
→ Controls & communicate with ALU, I/P/O/P devices, Primary & secondary storage devices.

⇒ CU controls & co-ordinates all the activities of the computer system.

⇒ It controls the operations of computer system based on the instructions in the program by executing them in proper order.

⇒ CU components :

- 1) Instruction register
- 2) Instruction decoder
- 3) Timing & control logic



1) Instruction register stores the instruction while it is being executed.

2) Instruction decoder will translate it in binary form

3) Timing & control logic generate the signals to execute it.

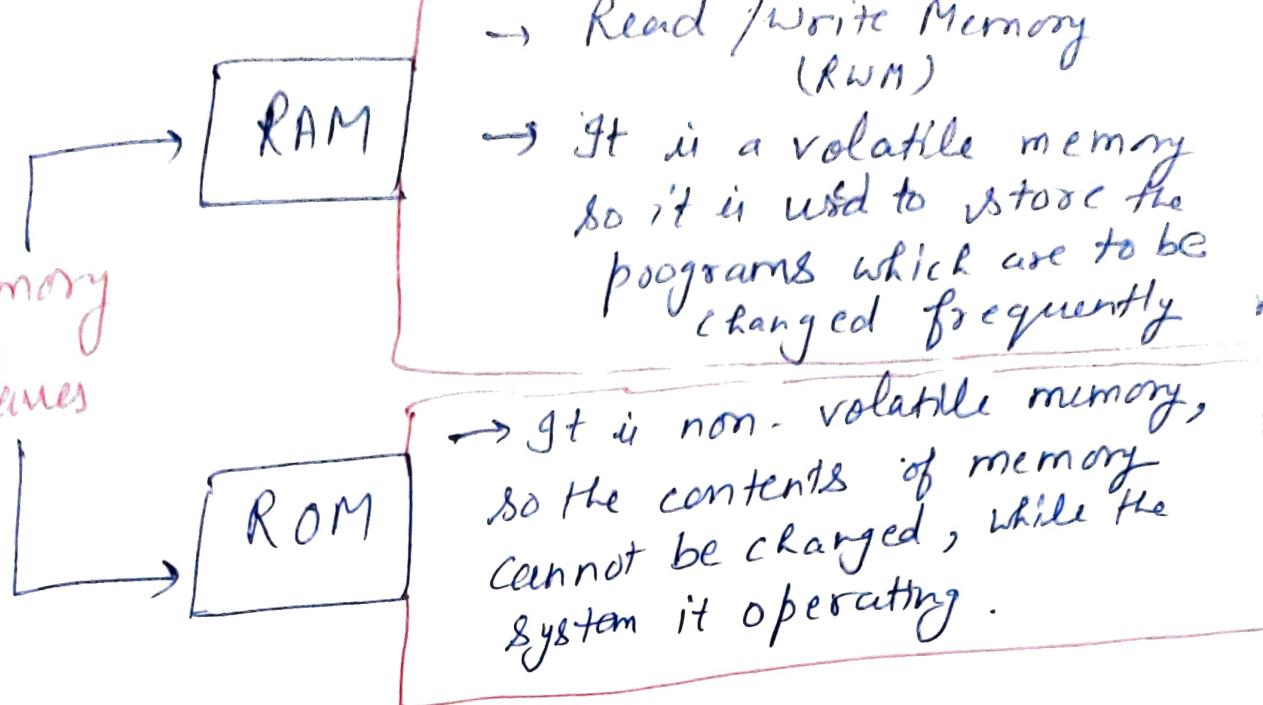
Registers

- ⇒ Registers are small, high speed temporary storage inside the CPU.
- ⇒ CPU use this temporary memory for store data or instructions during processing.
- ⇒ Registers also pass that data/information to the other parts of CPU or to main memory during the processing.
- ⇒ CPU contains several registers. Each register has its own pre-defined function.

Commonly used CPU registers are:

- Instruction Register
- Accumulator "
- Data "
- Program Counter Register
- Memory Address "

Memory Classes



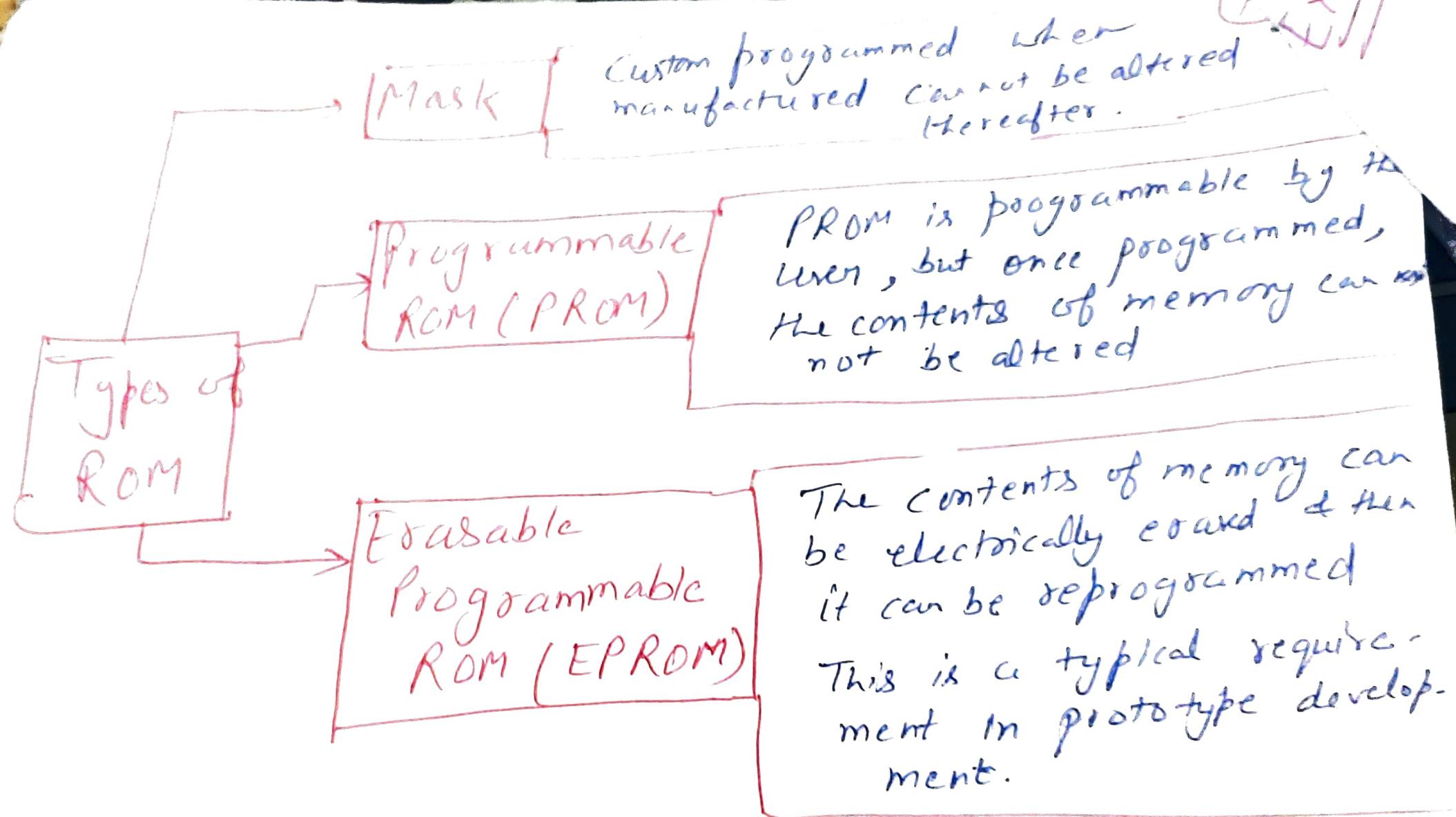
Types in RAM

Static RAM

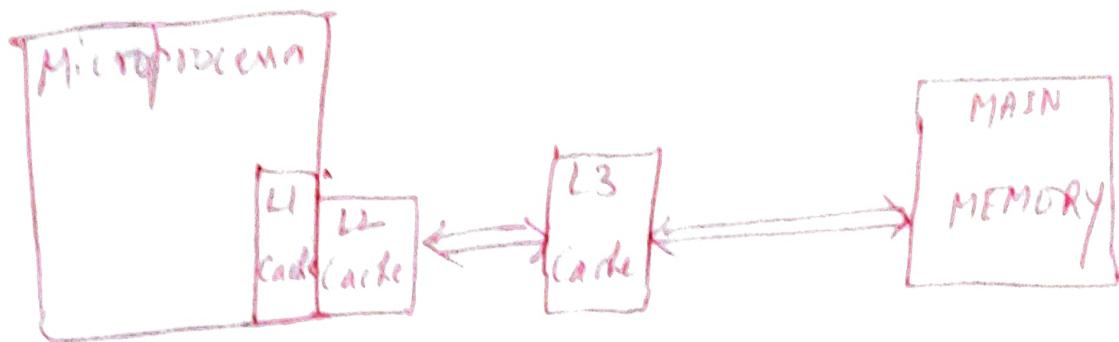
- made up of flip-flops
- Large in size
- Data stored in the form of voltage
- Much expensive as compare to dynamic RAM
- Low storage capacity
- consume more power
- Fast
- Data sustain with time

Dynamic RAM

- made up of capacitors
- small in size
- Data stored in the form of charge
- Less expensive as compare to static RAM
- High storage capacity
- consume less power
- Slow
- Data loses with time, so need refreshing circuit



Cache Memory



- Cache memory is a high speed small amount of memory inside the microprocessor also known as CPU memory because one portion of cache memory is directly integrated on CPU.
- CPU quickly access this memory as compared to RAM so cache is faster than main memory but it is very expensive.
- It stores data or information that is frequently required by CPU, means it keeps active portion of main memory.

INTERNAL CPU BUSES

- A bus is group of parallel wires used to carry data/information from one part of computer to another part of computer.
- ⇒ CPU is connected with all the devices through buses.
- ⇒ The buses found inside the CPU core known as Internal buses.

✓ Address bus

✓ Data bus

✓ Control bus

INTERNAL CPU BUSES (Cont. -)

- ⇒ Extension of these three buses is also available outside the CPU to communicate all the connected components with computer.
- ⇒ These buses are called external buses.



DP

Suppose we want to add

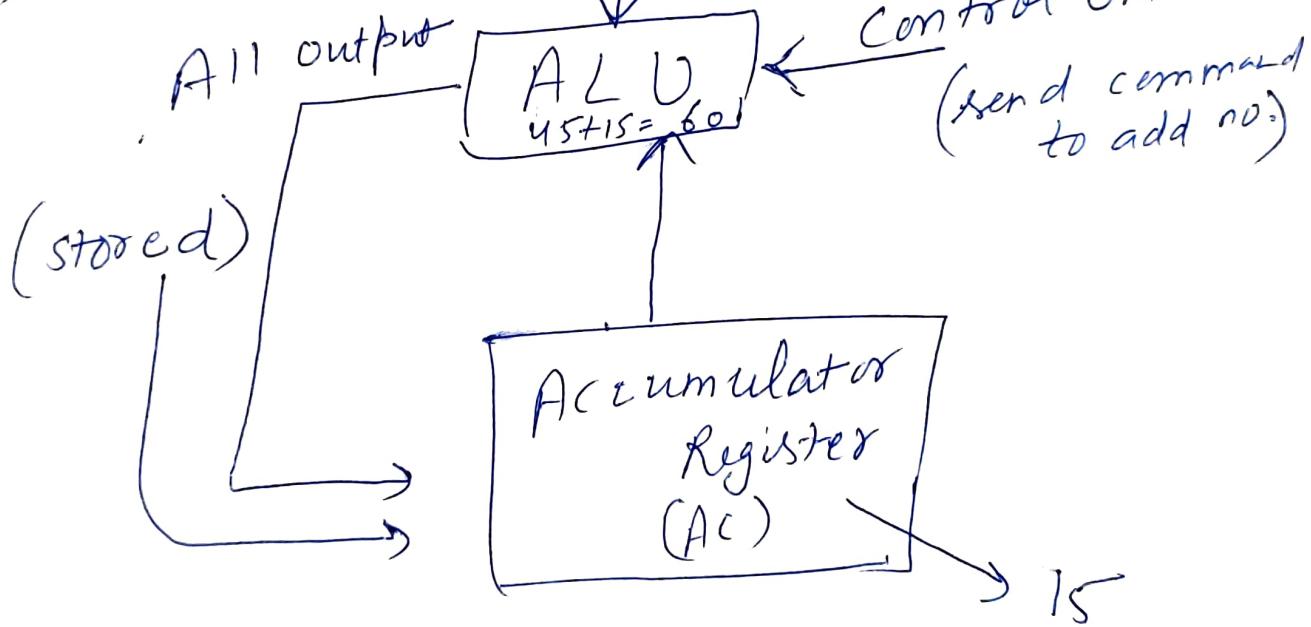
$$15 + 45$$

$$AC \rightarrow 15$$

$$DR \rightarrow 45$$

$$ALU \rightarrow 60$$

$$\text{Save} \leftrightarrow AC \\ (\text{stored})$$

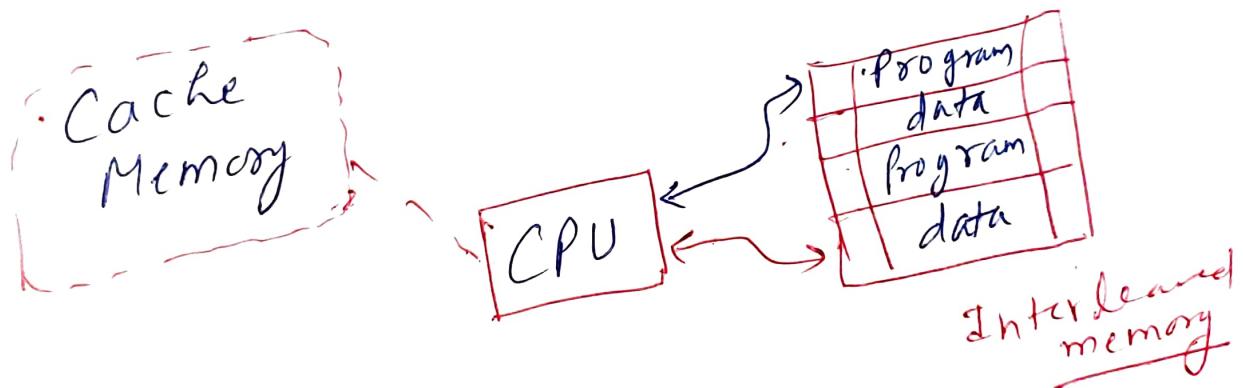
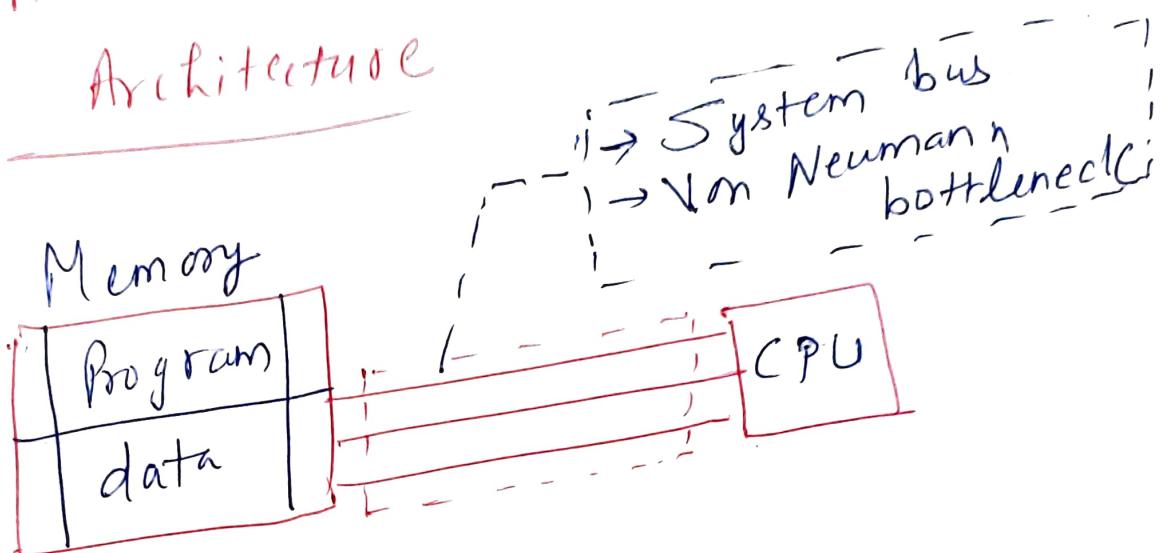


Conceptual diagram of ALU

Types of Architecture

Princeton or Von Neumann

Architecture

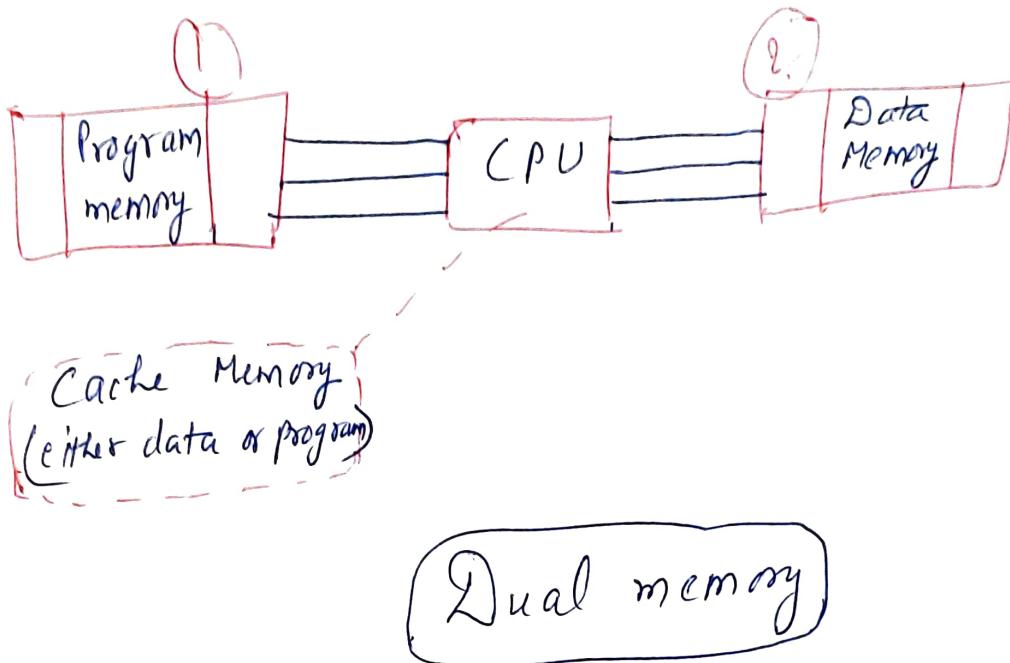


Single memory

Save time
(less idle time)

13

Harvard Architecture



How to choose a microprocessor?

Instruction set:

The set of instructions that the microprocessor can execute.

Bandwidth: (how much data it can handle e.g. 4-bit, 8-bit or any)
The no. of bits processed in a single instruction.

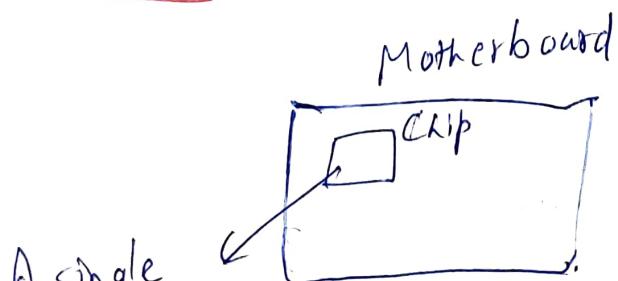
Clock speed:

The clock speed determines the no. of instructions per second, which processor can execute. It is provided in MHz or GHz.

Microprocessor

- ① CPU
- ② RAM, ROM, I/O ports can be added externally and can vary in numbers
- ③ heavier and costlier
- ④ Multitasking
- ⑤ Need high amount of resources

Microprocessor



A single chip is called microprocessor.
It is capable of processing data
It controls all components

Microcontroller

- ① CPU + RAM + ROM + other peripheral
- ② cannot be added externally, they are to be embedded on a chip and are fixed in number
- ③ light weight and cheaper
- ④ Specific task
- ⑤ Need small amount of resources

⇒ CPU is brain of micro computer.

Microprocessor and Microcontroller

↓
chip

[ALU, reg,
control unit]

MPU

(you have to design a
PLB if you want
this)

Microprocessor system



connected with
buses.

↓
Integrated electronic
computing device that
include three major
components on a single
chip.

Microprocessor

+

Memory

+

I/O ports

MCU

Microcontroller

CPU	RAM	ROM
I/O PORT	TIMER	SERIAL COM PORT

single chip

Why we prefer microcontroller if
want to automate any product/system?

(9)

Evolution of intel microprocessor

first generation (1939-1954) \Rightarrow vacuum tubes

Second generation (1954-1959) \Rightarrow transistors

Third generation (1959-1971) \Rightarrow transistors
along with IC
(Integrated Circuits)

fourth generation (1971 - present) \Rightarrow Microprocessor

\Rightarrow up is scaling from 4004 to
pentium 4

\Rightarrow up is identified with word size of data

e.g. ALU can perform 4-bit data operation
at a time. Then these type of
microprocessor is called 4-bit microprocessor.

Different Intel Microprocessor

* 4-Bit Processor

Intel 4004, Intel 4040 \Rightarrow ALU
can perform 4-bit of data at a time.

* 8-Bit Processor

8 bit of data at a time

8008, 8080, 8085, MOTOROLA 6800 (M6800)

↓

intron speed

* 16-Bit Processors
ALU can perform 16-bit operations at a time.

8086, 8088, Zilog Z800, 80186, 80286

* 32-bit Processors

ALU can perform 32-bit operations at a time.

INTEL 80386, 80387, 80486

INTEL PENTIUM, INTEL PENTIUM PRO

INTEL PENTIUM II, INTEL PENTIUM III

INTEL PENTIUM IV

INTEL DUAL CORE

* 64-Bit Processors → (very fast)

INTEL CORE 2,

INTEL CORE I7,

INTEL CORE I5,

INTEL CORE I3

Microprocessor Operations

- The internal logic design of the microprocessor called its architecture, determine how and what various operations are performed by the microprocessor.
- The microprocessor is programmable logic device designed with registers, flipflop and timing elements.
- All functions in microprocessor can be classified in three categories :
 1. Microprocessor Initiated operations
 2. Internal data operations
 3. Peripheral (or externally initiated) Operation

1. Microprocessor Initiated operations

Primarily microprocessor performs four operations :—

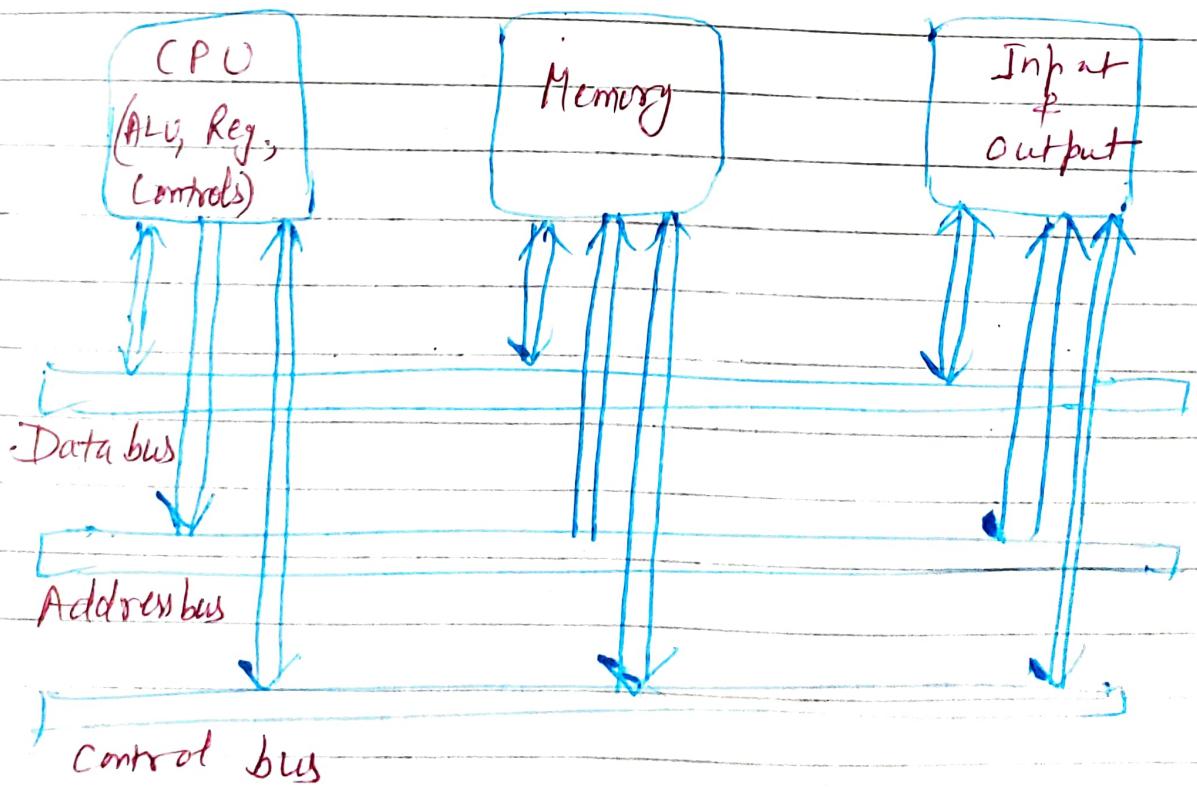
- a) Memory read (Read data from memory)
- b) Memory writes (write data into memory)
- c) I/O read (Accept data to o/p device)
- d) I/O writes (Sends data to o/p device)

Microprocessor initiated operations -
To communicate with a peripheral (or a memory location), The MPU needs to perform following steps:

1. Identify the peripheral
2. Transfer data
3. Provide timing or synchronization signals.

Microprocessor performed these functions using set of buses:

- Address bus
- Data bus
- Control bus



Internal Data Operations

The internal architecture of the 8085/8080A microprocessor determines how and what operation can be performed with the data. These operations are—

- Store 8-bit data
- Perform arithmetic and logical operations
- Test for conditions
- Sequence the execution of instructions
- Store data temporarily during execution in the defined R/W memory locations called the stack.

Peripheral or Externally Initiated Operations

External devices (or signals) can initiate the following operation for which individual pins on Microprocessor chip are assigned:

Reset, Interrupt, Ready, Hold

A) Reset: When reset is activated all internal operations are suspended and the program counter is cleared.

B) Interrupt: The microprocessor can be interrupted from normal execution and asked to execute other

instructions called "service routine" (emergency), Microprocessor resumes its operation after that.

C) Ready : 8085 has pin called ready, if the signal is low Microprocessor enters into wait state, this signal used to synchronize slower peripherals with Microprocessor.

D) Hold : When hold pin activated by external signal Microprocessor relinquishes control buses and allows the external peripheral to use the buses.
For exp. : Hold signal is used in direct memory access data transfer.

Addressing Modes

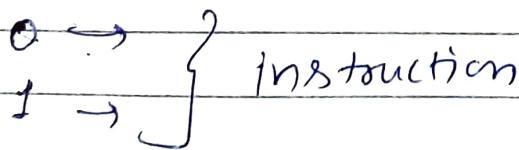
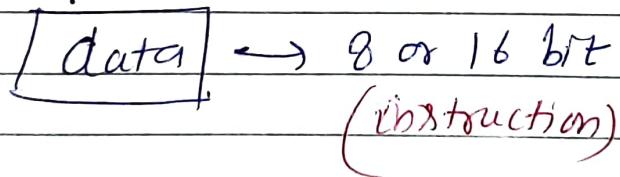
The various formats for specifying operands are called addressing modes.

Addressing mode Types:-

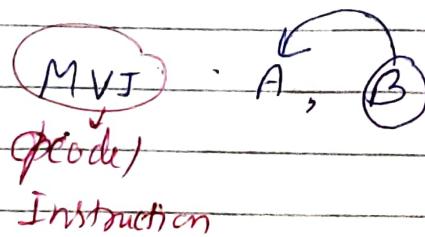
- 1) Immediate addressing mode
- 2) Register " "
- 3) Direct " "
- 4) Indirect " "
- 5) Implied " "

1) Immediate Addressing mode

In immediate addressing mode, the data may be of 8 bit or 16 bit and it is present in the instruction.



8 bit → 1 byte



Opcode | operand

2 byte

Opcode | operand | operand

↓

lower
order

8 bit

↓

higher
order

8 bit

→ Immediate addressing mode instruction is of 2-bytes or 3-bytes long.

→ Whenever the symbol (I) is present in the instruction, then it is an immediate addressing mode / immediate data.

e.g. MVI A, ^{immediate}AOH Immediate data
 ↓ ↓
 Move Accumulator

AOH generally transfers or copy the data in accumulator (A).

② Register Addressing mode

There are two terms in register addressing mode.

Destination

on

Source

↓
Where we have something
to copy and reach out
the destination.

↓
From where some-
thing or some data
is coming.

- In Register addressing mode the operands (i.e. source or destination) is a general purpose register.
- Register addressing instruction is generally 1 byte long that means only opcode is available or present in it.
- That means opcode specified that what the operation has to be performed or whom registers are used to do that operation.

e.g.-

ADD B
A

$$\begin{aligned} A &= 24 \\ B &= 2 \end{aligned}$$

Add B

$$A \rightarrow 24$$

$$B \rightarrow 2$$

$$\begin{aligned} A &\leftarrow A + B \\ 26 &\leftarrow 24 + 2 \end{aligned}$$

In this example, the content or data in B register or the data in accumulator (A) are add and then the output of this operation is saved in Accumulator (A).

Here accumulator (A) is present by default.

ADD B

$$\boxed{A \leftarrow A + B}$$

so in this type of addressing mode, the data is transferred by registers.

③ Direct Addressing Mode

- In direct addressing mode, the 16-bit address of operand are given in instruction.
- The instruction in direct addressing mode is generally 3-byte instruction.

1st byte - opcode

2nd byte - Lower order addressing mode byte

3rd byte - Higher order address byte.

e.g.

L D A, C200H

Load Accumulator

The data on this address load the content on accumulators.

(4) Indirect addressing mode:

The memory address where operands is located, is specified by the content of a register pair.

e.g.

MOV A, (M) → Memory Pointer

The memory pointer at HL register pair is transferred into accumulator (A)

MOV A, M → HL

(5) Implied or Implicit addressing mode

→ In this type of addressing mode, no operands are required.

→ The data is specified in opcode

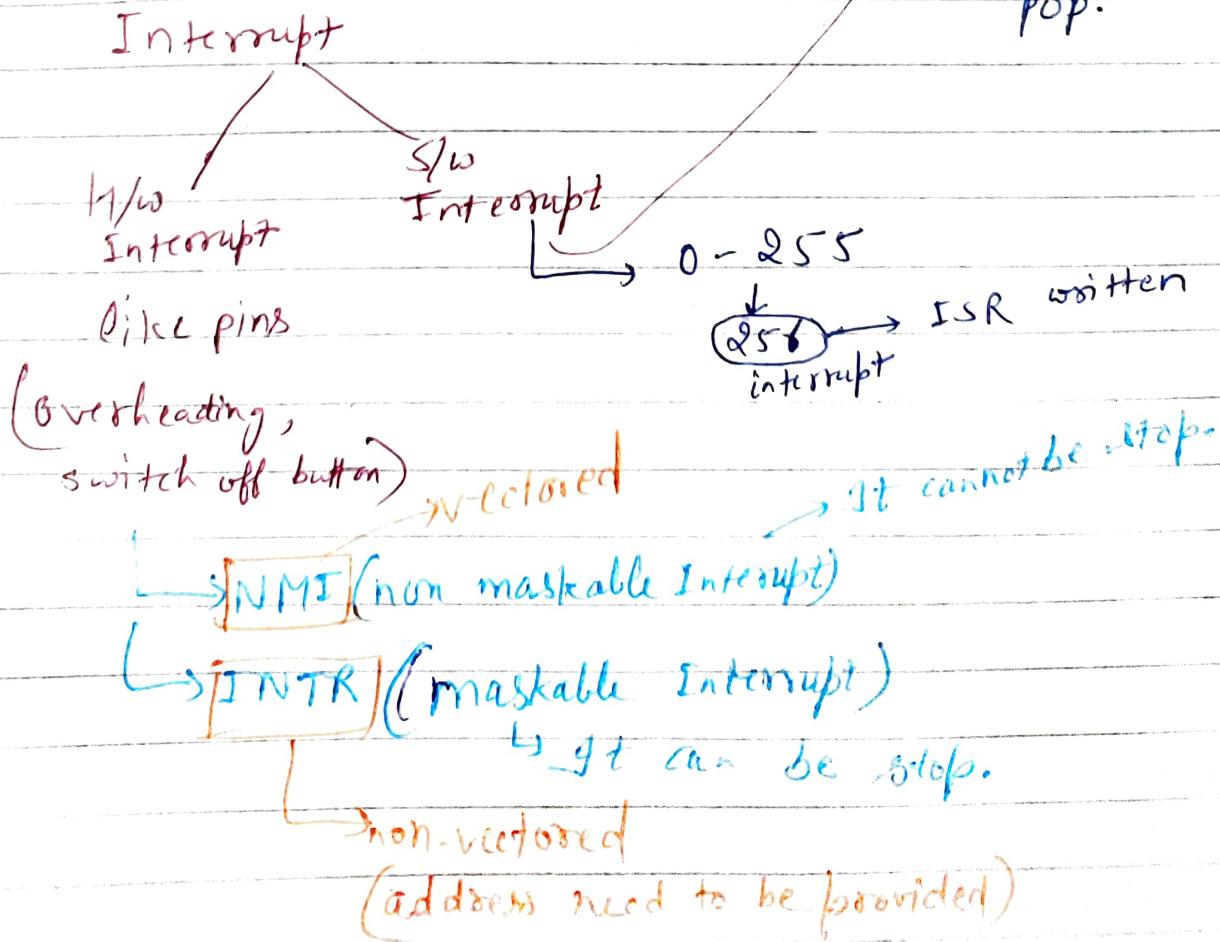
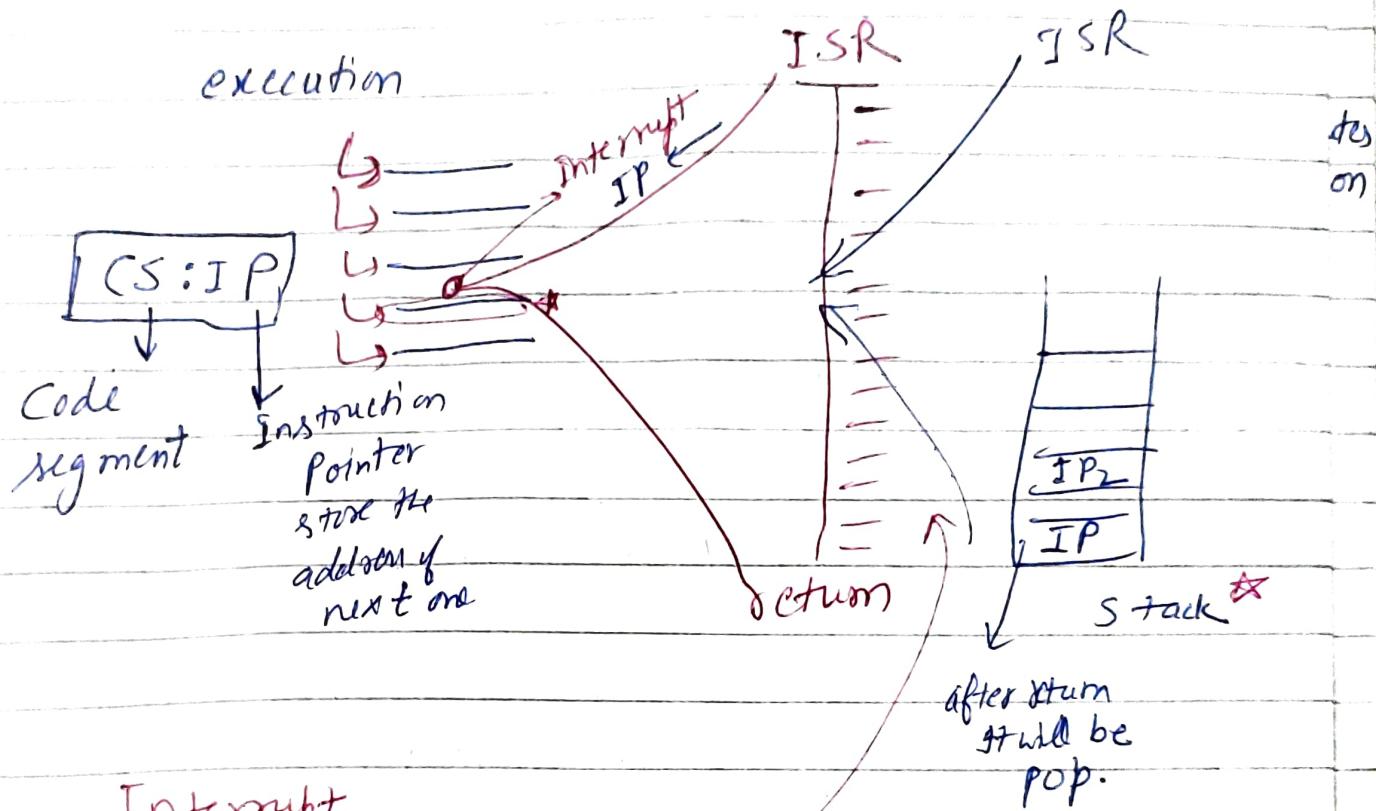
Generally the implied addressing mode is of 1 byte instruction.

e.g.

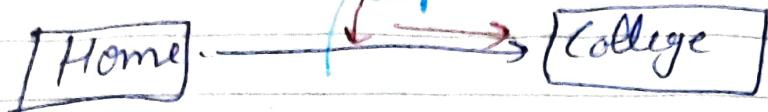
RAL → Left
Rotate Accumulator

Interrupts (Summary)

It is a condition that makes CPU to execute ISR (Interrupt service routine).



Interrupt



ISR
means watch
called you to
friends called you to
watch movie
send
interrupt
answer

normal execution

Definition

An external I/O signal or an instruction can suspend normal flow of execution and go to "interrupt service routine". Once serviced, program resumes.

What is INTERRUPT SERVICE ROUTINE (ISR)?

A small program or a routine that when executed services the corresponding interrupting source is called as an ISR.

See the
example

Sequential execution
of program
commands

PROGRAM

Program control
jumps to ISR

ISR

ISR executes
to completion

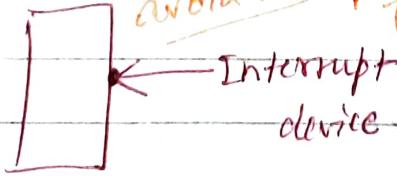
H/w interrupt
caught here

Program control
returns to -interrupted
program

Sequential
execution of
program picks up
where it left off

Maskable / Non-Maskable interrupt

Hide / Ignore / Disable



avoids it & proceed further means mask the interrupt

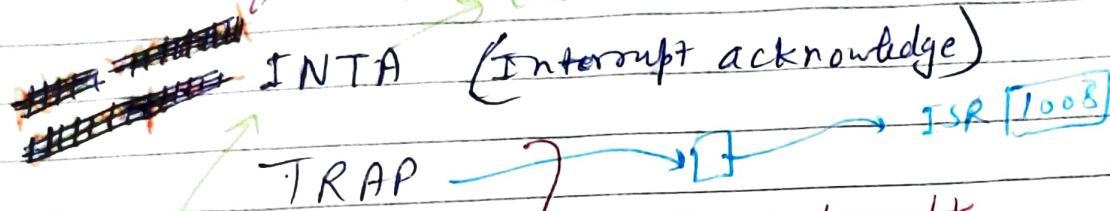
⇒ The interrupt to which we can ignore, known as Maskable interrupt.

⇒ The interrupt to which we cannot ignore, known as non-maskable interrupt.

e.g. TRAP in 8085

↓
It can be held/delay but
can not be ignored

There are 6 pins available in 8085
for interrupt: INTA → Outcome interrupt i.e. op

 INTA (Interrupt acknowledge)

TRAP → ISR [1008]

RST 7.5

RST 6.5

RST 5.5

INTA

H/w interrupt
(vectored interrupt)

↓
location is fix

Non-vectored
interrupt

(location is
not fixed)

Q. What happens when interrupt comes?

- INTA (Active low) signal to the peripheral.
- vectored address of particular interrupt is stored in program counter.
- The processor executes an ISR.

8085 Interrupt

RST0

RST1

RST2

RST3

RST4

RST5

RST6

RST7

Location
of

Vector Address = Interrupt no. * 8 ISR

For exp



Hexa decimal form
?

RST1 : Vector address = $1 * 8 = 08$ (0008H)

RST2 : " " " = $2 * 8 = 16$ (0010H)

RST3 : " " " = $3 * 8 = 24$ (0018H)
= 32 (0020H)

RST4 : " " " = 40 (0028H)

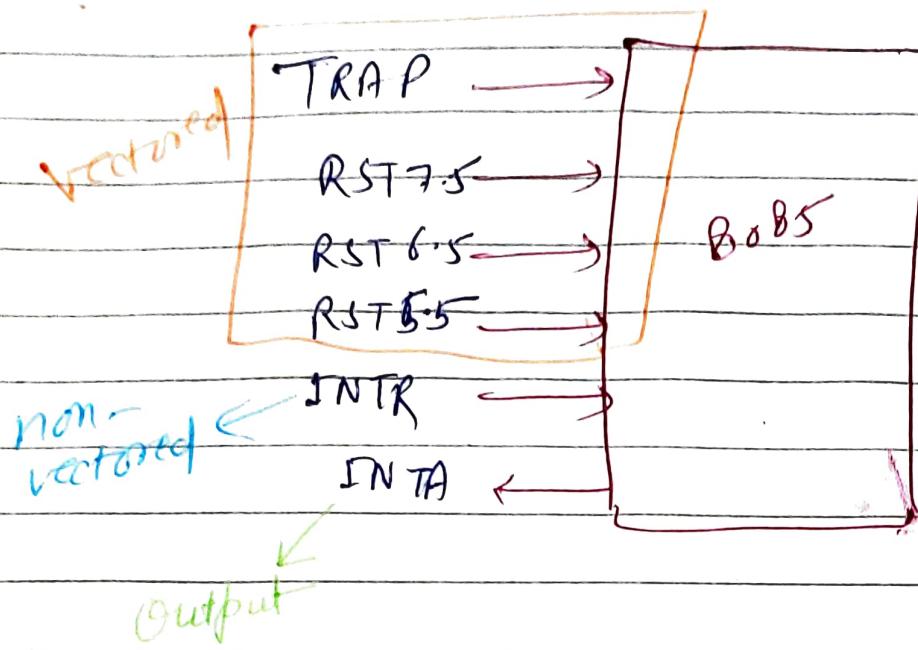
RST5 : " " " = 48 (0030H)

RST6 : " " " = 56 (0038H)

RST7 : " " " = 60 (0040H)

RST0 : " " " = 00 (0000H)

Hardware Interrupt



→ 5 Interrupt I/P

- TRAP, RST 7.5, 6.5, 5.5 are vectored interrupts
- TRAP IN NMI
- INTR is non vectored interrupt
- INTA is acknowledgement signal

Data Transfer Schemes

- Concept of Programmed I/O operations
- Synchronous Data Transfer
- Asynchronous Data Transfer (Handshaking)
- Interrupt driven data transfer
- DMA (Direct Memory Access)
- Serial output data
- Parallel output data

Concept

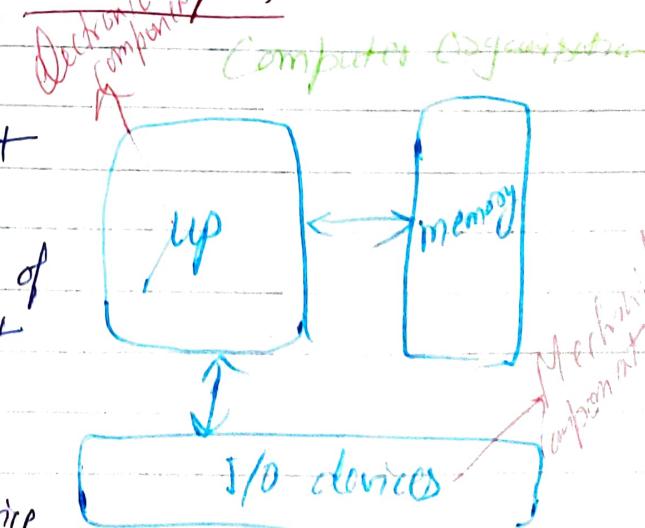
Technique to transfer data in between parts of a computer organization, i.e. CPU, memory and I/O devices, is called data transfer techniques/schemes.

Need of Data Transfer Techniques

⇒ CPU 8085 works at about 3 MHz clock frequency. But speed of I/O may be different from it.

⇒ More than one I/O device try to access CPU at the same time for data transfer.

⇒ I/O device try to access memory device directly.

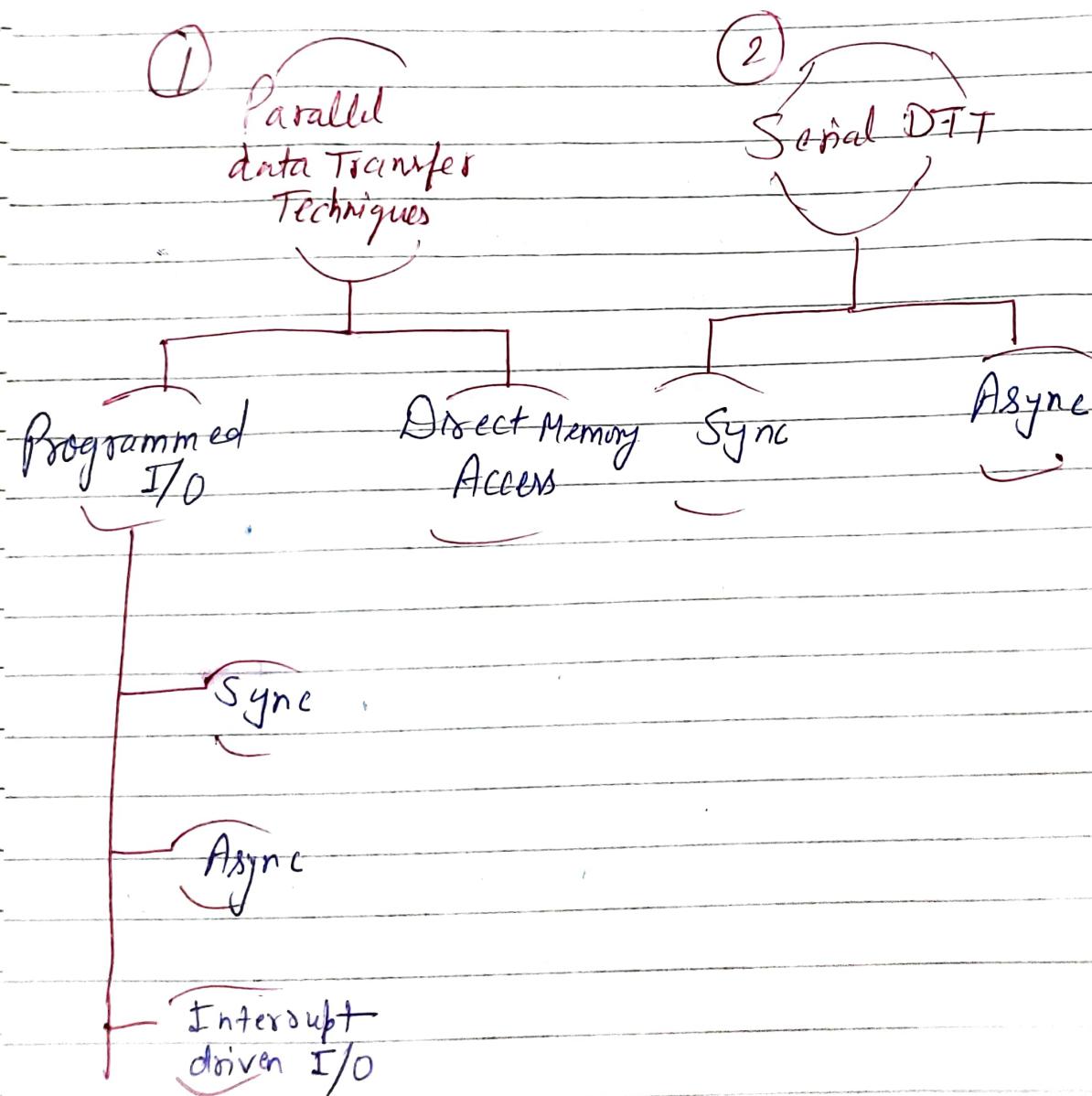


Data Transfer Modes

μP - can send/receive data in two modes—

1. parallel data transfer technique
2. serial data transfer technique

⇒ 8085 MP is a parallel device, since it transfers 8 bits of data at a time over 8 data lines.



Synchronous DTT

Synchronous means "at the same time"

So if

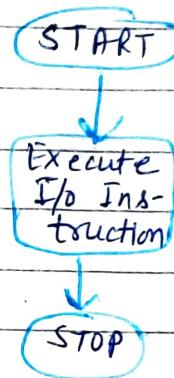
speed of 8085 = speed of connected Peripheral
or

speed of connected peripheral is known

This technique will be used.

Advantage : Simplicity

Disadvantage : Not universal



Asynchronous DTT

means "not at the same time"

So if

speed of 8085 ≠ speed of connected Peripheral

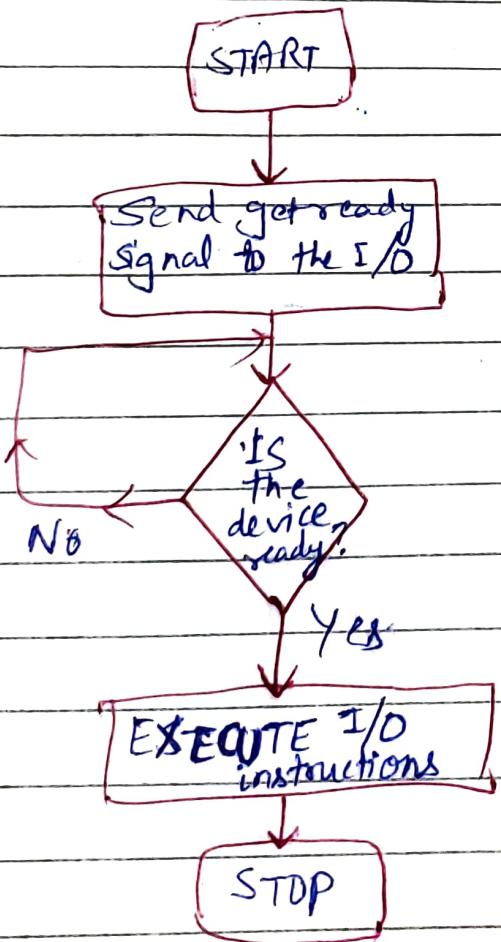
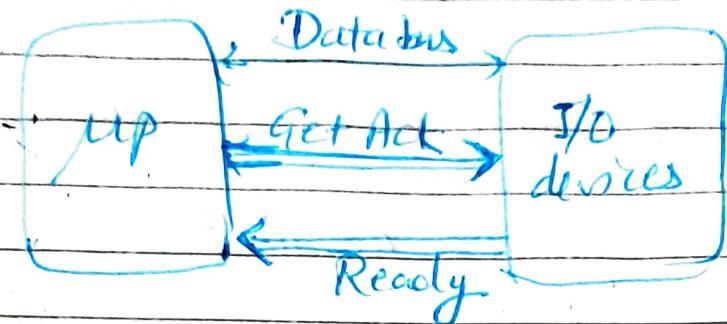
or

speed of connected peripheral is unknown

This technique will be used

How?: In this technique the status of the I/O device is checked by the microprocessor before the data is transferred.

It's also called as "HANDSHAKING" mode.

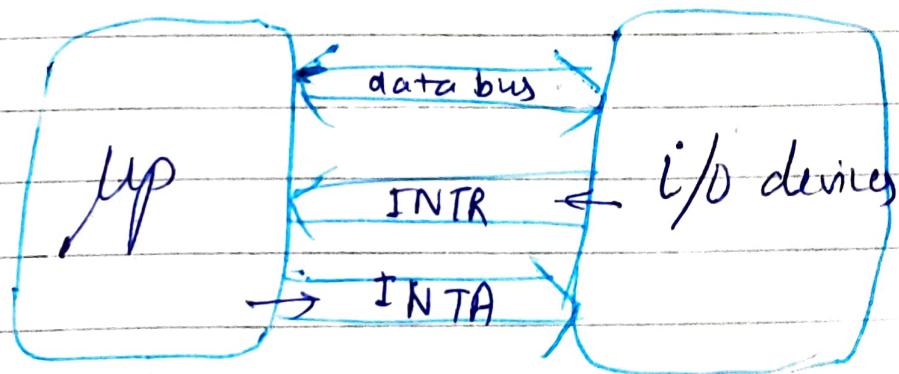


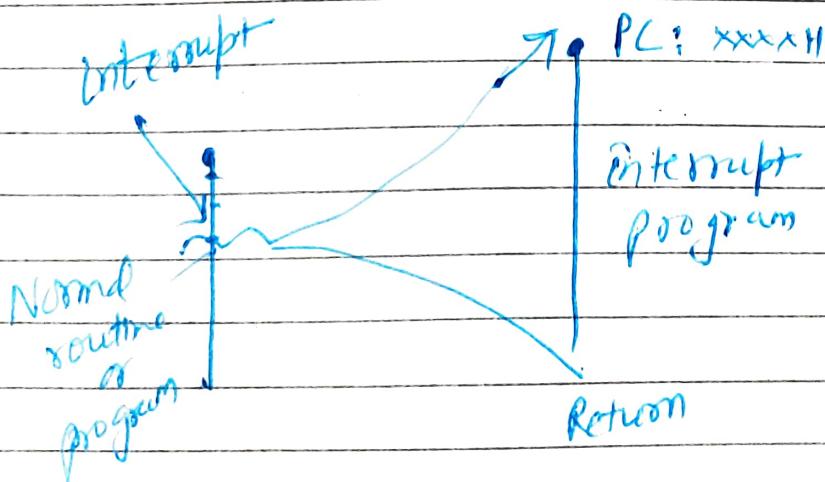
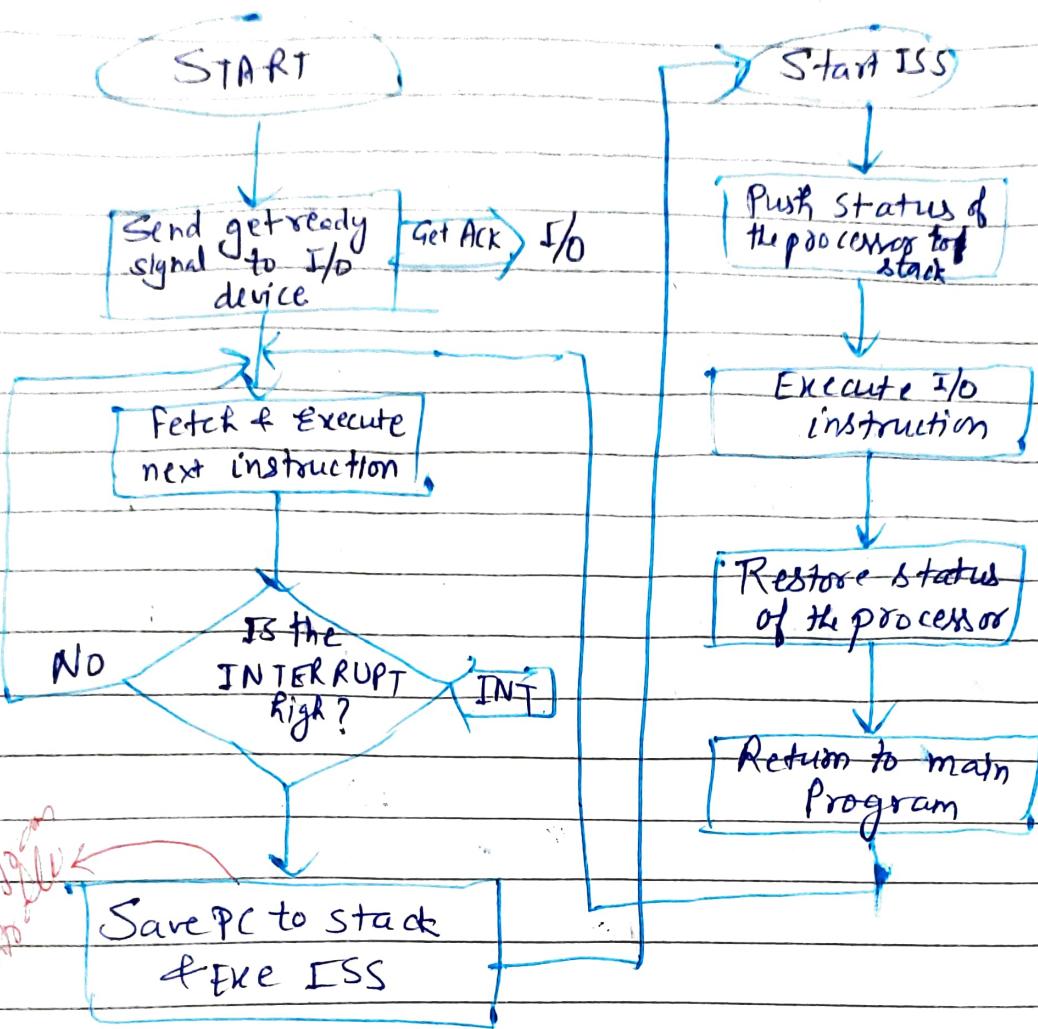
Advantage : Speed of I/O is not priority

Disadvantage:
Waste a lot of CPU time to check for ready signal

Interrupt driven DTT

- ⇒ After execution of each instruction, MP checks for interrupt status.
- ⇒ If there is any interrupt request, MP shifts from normal routine to ISR.
- ⇒ ISR is a hardwired memory location, for each interrupt a memory location is fixed except INTR.
- ⇒ For INTR, MP asks for I/O peripheral to provide location address.





$PC = TOS \rightarrow$ Top of stack

Polymon control (hex)

Types of interrupt driven DTT

1. Software interrupts
2. Hardware interrupts

H/W interrupt

TRAP (RST 4.5)

RST 5.5

RST 6.5

RST 7.5

INTR

Vector address

0024H

002CH

0034H

003CH

N/D (not defined)

S/W interrupt

vector address

0000H

0008H

0010H

0018H

0020H

0028H

0030H

0038H

Restart

will
choose
or
shift
in
any of
these

RST0

RST1

RST2

RST3

RST4

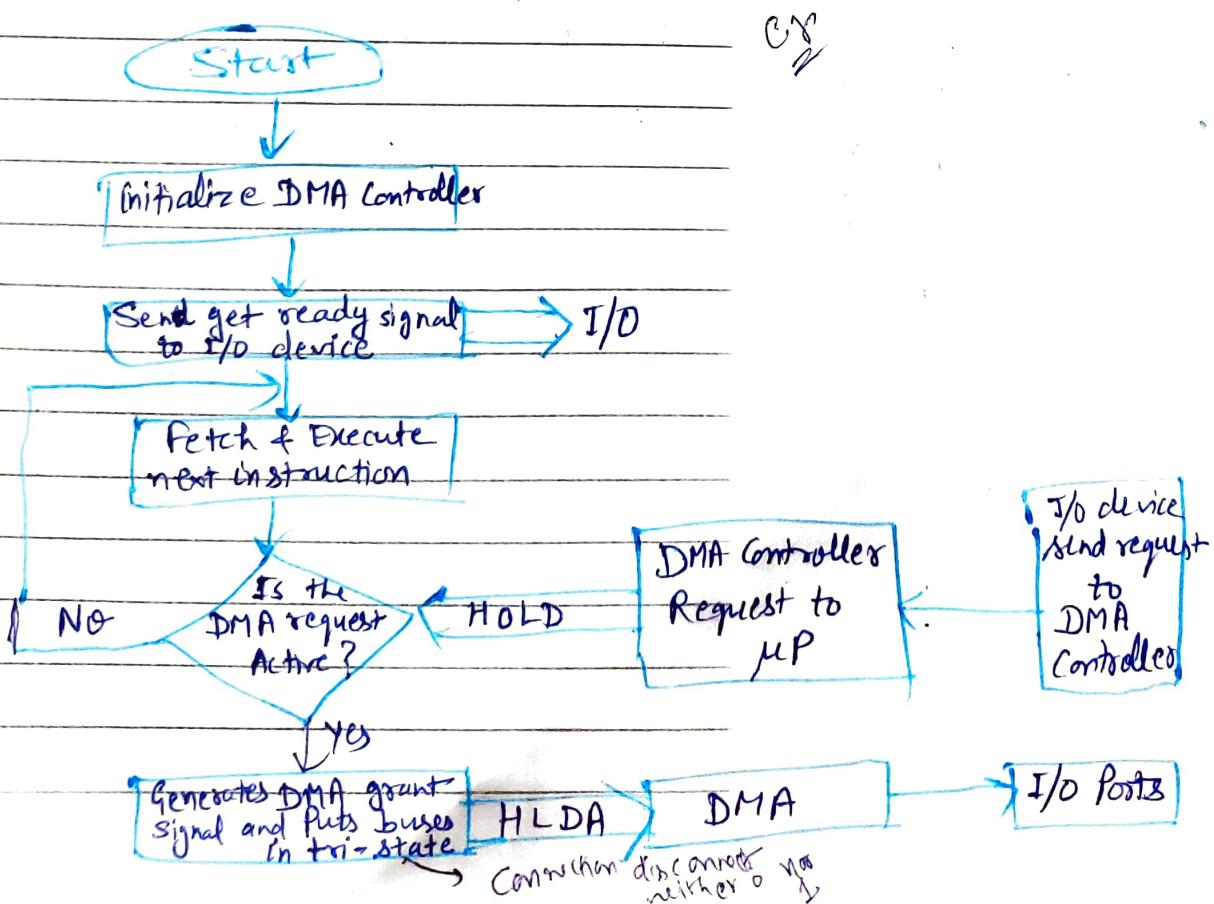
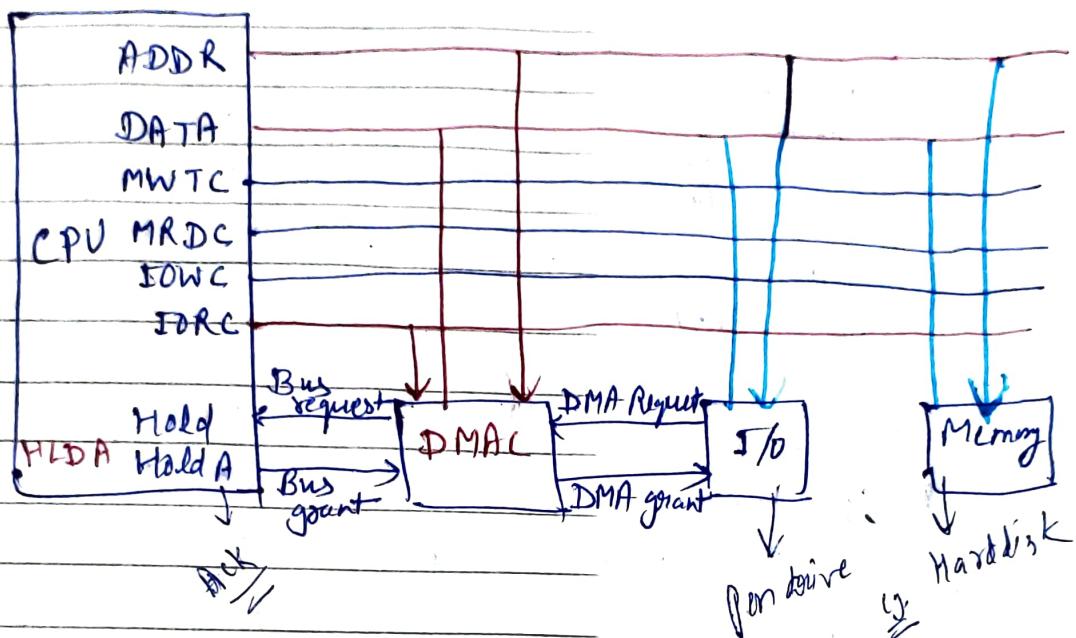
RST5

RST6

RST7

Direct Memory Address (DMA)

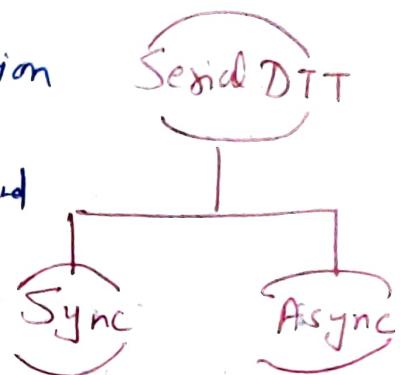
It provides direct access of memory to I/O devices with help of a DMA controller.



Serial DTT

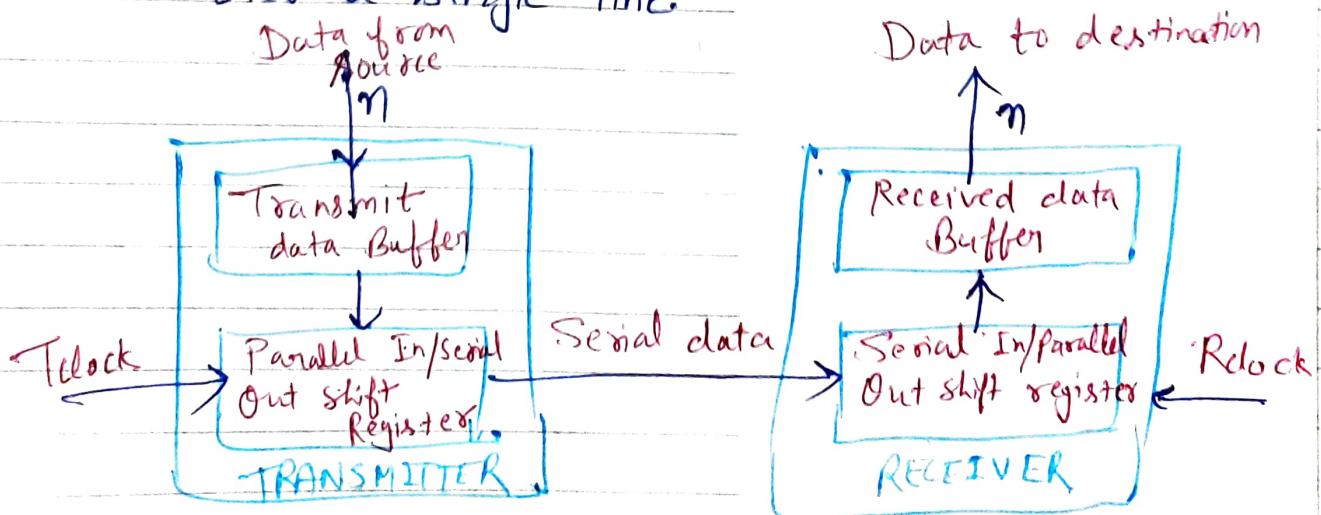
Problems with Parallel I/O:

- Needs a wire for each bit
- When the source and destination are more than a few feet the parallel cable can be bulky and expensive
- Induced noises during long distance communication.



Serial Data Transfer Technique

- An (8 bit) parallel data is converted into a stream of eight serial bits; known as parallel-to-serial conversion.
- After it, one bit at a time is transferred over a single line.



There are two basic types of serial communications :—

- Synchronous
- Asynchronous

Synchronous

→ Transmitter and receiver are synchronized

→ Need extra h/w for clock synchronization

→ Having faster data transfer rate

→ USART (Universal Synchronous / Asynchronous Receiver / transmitter)

chip

Asynchronous

→ Transmitter + receiver use different clocks.

→ No clock synchronization is required

→ Used in many applications such as keyboards, mice, modems

→ UART (Universal asynchronous Receiver / transmitter)

Serial Data Communication

(W.r.t. μ P 8085)

μ P 8085 provides two serial I/O pins as

1. SOD (Serial Output data)
2. SID (" . Input ")

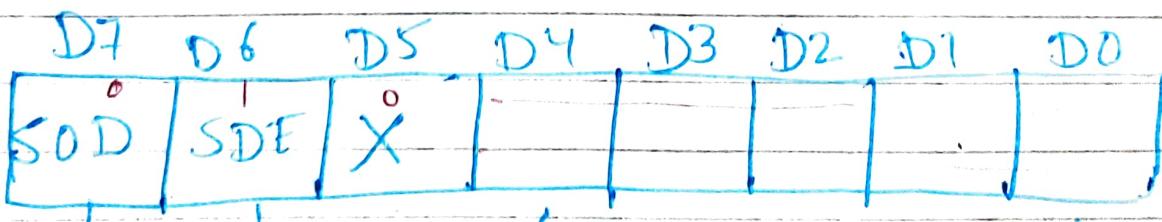
Data transfer in μ P 8085 controlled with two instructions —

1. SIM (Set interrupt Mask)
2. RIM (Reset " ")

These are multi-function instruction which not only use for serial data communication but also to mask interrupts.

SOD (Serial Output Data)

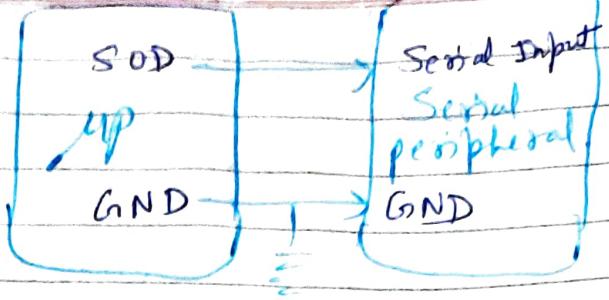
Accumulator (Reg A) contents by **SIM** Instruction



For serial data transmission

1 = Enable SOD
0 = Disable SOD

for Interrupt control



⇒ The μP 8085 transmits asynchronous serial data to peripherals through the SOD pin.

⇒ The instruction SIM is used to output data serially from SOD line.

⇒ SIM uses D7 bit of Accumulator to do so.

- For output 0 at SOD pin put 0100 0000 in A
- for output 1 at SOD pin put 1100 0000 in A

MVI A, 40

SIM

Set D7=0 and D6=1

for logic 0

MVI A, C0

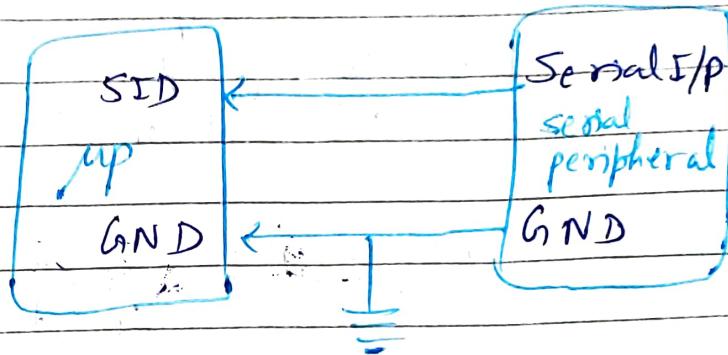
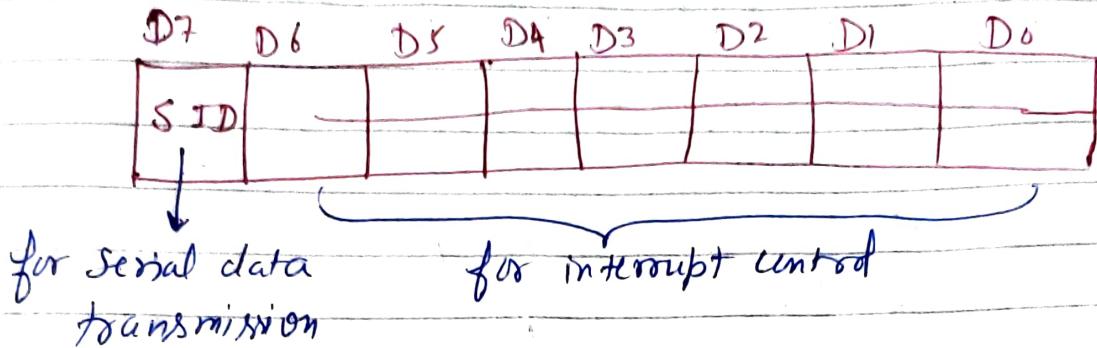
SIM

Set D7=1 and D6=1

For logic 1

SID (Serial Input Data)

Accumulator (Reg A) Contents by RIM instruction

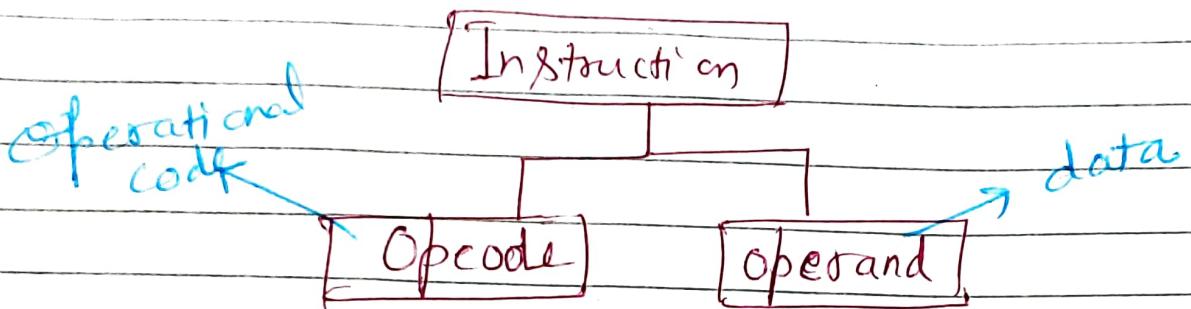


- ⇒ The μP 8085 receives asynchronous serial data from peripherals through the SID pin.
- ⇒ The instruction RIM is used to take input data serially from SID line.
- ⇒ RIM uses D7 bit of Accumulator to do so.

Instruction and data flow

time \Rightarrow Instruction is a bit pattern that instruct the microprocessor to perform a specific function.

Every microprocessor has its own instruction set that is provided by manufacturer.



Entire group of instruction that a microprocessor can handle is called its INSTRUCTION SET, this determines microprocessor's functionality.

Instructions are classified based on parameters such as functionality, length and operand addressing.

$$\begin{array}{r} 56 \\ + 64 \\ \hline 120 \end{array}$$

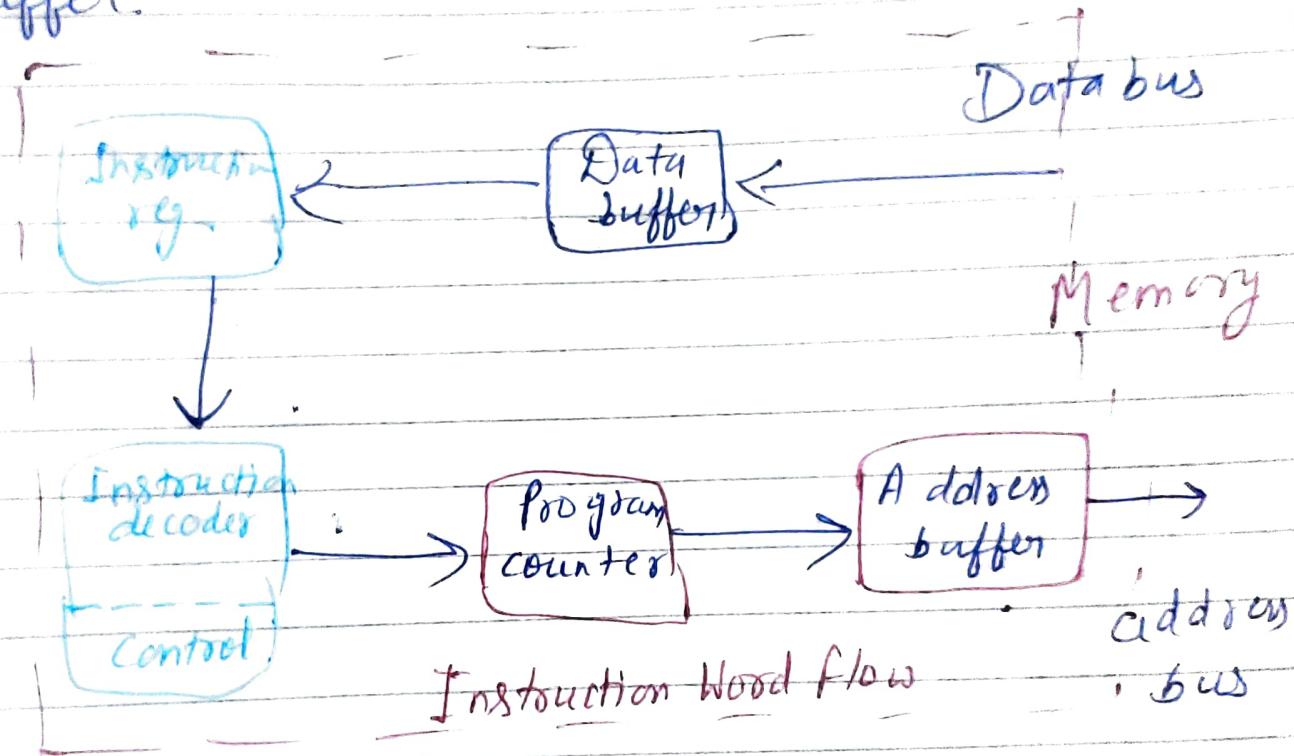
Result
R

ADD B₁

ADD D₂

instruction
Register

- At the starting of fetch cycle, the content of program counter is transferred in memory address register or address buffer.



- ⇒ Then the content of address buffer is transferred to memory through address bus. Microprocessor send some control signal to memory. It indicates that data has to be read or write.
- ⇒ Then memory send the opcode to the microprocessor through data bus.
- ⇒ Opcode firstly comes in data buffer. Then

operation code is placed to instruction register and it decode the it by the instruction decoder.

- ⇒ Data word flow: Data word is received from the memory or input.
- ⇒ The data word is transferred to the microprocessor through the data bus or store in accumulator or some other general purpose register, depends up on the instruction.
- ⇒ After the execution, the result is send to memory or output device.
- ⇒ When data is write to memory then data buffer holds the data until the completion of write operation.

