Section – C
Computer Architecture

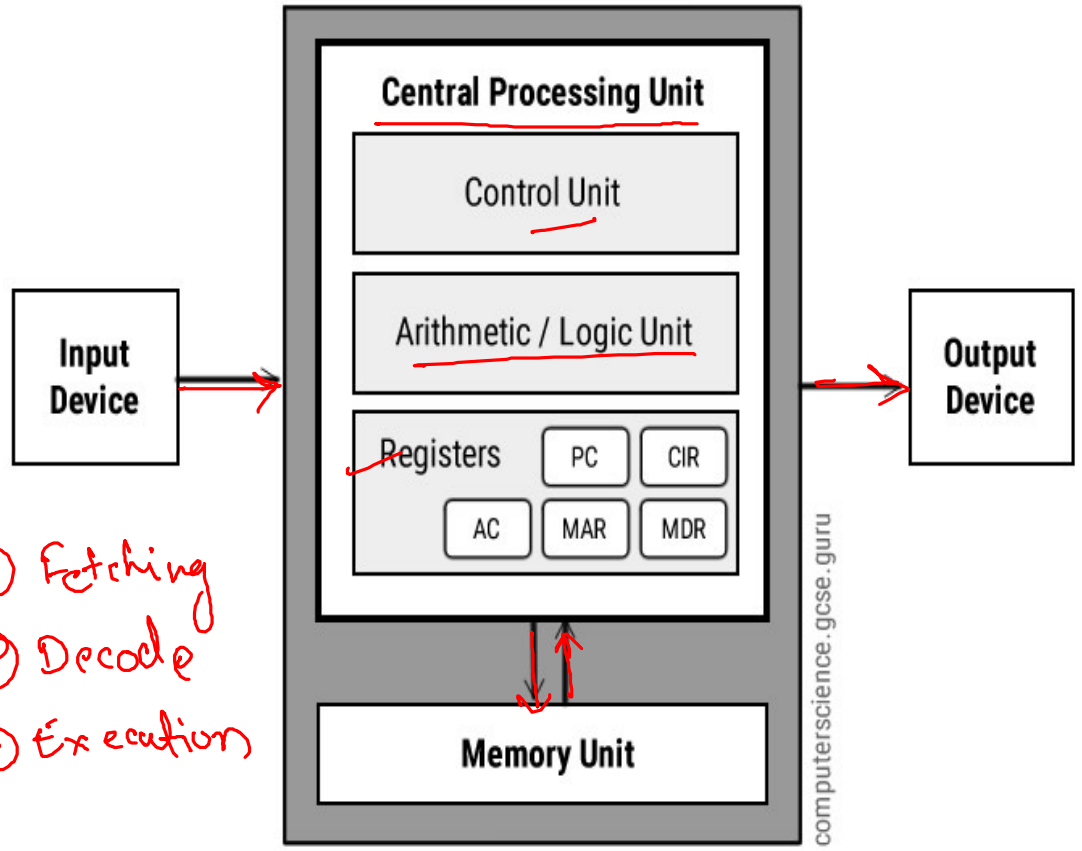
Trainer :- Sohail Inamdar

<sohail.inamdar@sunbeaminfo.com>

Introduction

- A typical programmable system can be represented by
  - CPU
  - Memory
  - Input/output devices

- In computer, the CPU executes each instruction provided to it, in a series of steps, this series of steps is called Machine Cycle.
- One machine cycle involves fetching of instruction, decoding the instruction, executing the instruction



1. Fetching
2. Decode
3. Execution

Computer Architecture

- Computer architecture design consists of a Control Unit, Arithmetic and Logic Unit(ALU), Memory Unit, Register and Inputs/Outputs.
- **Central Processing Unit (CPU).**
  - The CPU is the electronic circuit responsible for executing the instructions of a computer program.
  - It is sometimes referred to as the microprocessor or processor.
- **Arithmetic and Logic Unit (ALU)**
  - The ALU allows arithmetic (add, subtract etc) and logic (AND, OR, NOT etc) operations to be carried out.
- **Control Unit (CU)**
  - The control unit controls the operation of the computer's ALU, memory and input/output devices, telling them how to respond to the program instructions it has just read and interpreted from the memory unit.
  - The control unit also provides the timing and control signals required by other computer components.

Computer Architecture

- **Registers**
  - Registers are high speed storage areas in the CPU. All data must be stored in a register before it can be processed.
  - **MAR - Memory Address Register**
    - Holds the memory location of data that needs to be accessed
  - **MDR - Memory Data Register**
    - Holds data that is being transferred to or from memory
  - **AC – Accumulator**
    - Where intermediate arithmetic and logic results are stored
  - **Program Counter**
    - Contains the address of the next instruction to be execute.
  - **CIR (**Current Instruction Register)
    - Contains the current instruction during processing.

Computer Architecture

- **Buses**
  - Buses are the means by which data is transmitted from one part of a computer to another, connecting all major internal components to the CPU and memory.
  - **Address Buses**
    - Carries the addresses of data (but not the data) between the processor and memory
  - **Data Buses**
    - Carries data between the processor, the memory unit and the input/output devices
  - **Control Buses**
    - The control bus sends out control signal to memory, I/O ports and other peripheral devices to ensure proper operation.

Computer Architecture

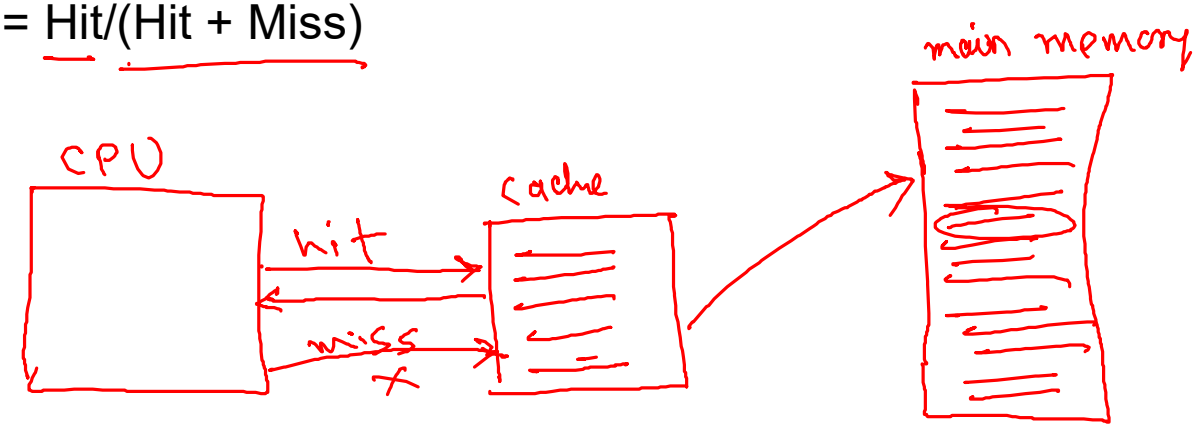
- **Memory Unit**
  - A Memory Unit is a collection of storage cells together with associated circuits needed to transfer information in and out of storage.
  - Two major types of memories are used in computer systems:
  - **Random Access Memory(RAM) and Read Only Memory(ROM).**
  - The memory unit consists of RAM, sometimes referred to as **primary or main memory**. Unlike a hard drive (secondary memory), this memory is fast and also directly accessible by the CPU.
  - RAM is split into partitions. Each partition consists of an address and its contents.
- **RAM: Random Access Memory**
  - **DRAM**: Dynamic RAM, is made of capacitors and transistors, and must be refreshed every 10~100ms. It is slower and cheaper than SRAM.
  - **SRAM**: Static RAM, has a six transistor circuit in each cell and retains data, until powered off.
  - **NVRAM**: Non-Volatile RAM, retains its data, even when turned off. Example: Flash memory.

Computer Architecture

- **ROM: Read Only Memory**
  - Read-only memory (Not writable).
  - This type of memory is non-volatile.
  - The information is stored permanently.
  - A ROM stores such instructions that are required to start (bootstrap) a computer.
  - **PROM(Programmable ROM), EPROM(Erasable PROM) and EEPROM(Electrically Erasable PROM)** are some commonly used ROMs.
- **Auxiliary Memory**
  - Devices that provide backup storage are called auxiliary memory.
  - For example: Magnetic disks and tapes are commonly used auxiliary devices. Other devices used as auxiliary memory are magnetic drums, magnetic bubble memory and optical disks.

Computer Architecture

- **Cache Memory**
  - The data or contents of the main memory that are used again and again by CPU, are stored in the cache memory so that we can easily access that data in shorter time.
  - Whenever the CPU needs to access memory, it first checks the cache memory. If the data is not found in cache memory then the CPU moves onto the main memory.
  - The performance of cache memory is measured in terms of a quantity called hit ratio. When the CPU refers to memory and finds the word in cache it is said to produce a hit. If the word is not found in cache, it is in main memory then it counts as a miss.
  - Hit Ratio = Hit/(Hit + Miss)



Computer Architecture

- **Word**
  - The memory stores binary information(1's and 0's) in groups of bits called words.
  - A memory word is a group of 1's and 0's and may represent a number, an instruction code, one or more alphanumeric characters, or any other binary coded information.
- **Byte**
  - A group of eight bits is called a byte. Most computer memories use words whose number of bits is a multiple of 8. Thus a 16-bit word contains two bytes, and a 32-bit word is made up of 4 bytes.
  - The capacity of memories in computers is usually stated as the total number of bytes that can be stored.
    - K(Kilo) is equal to 1024bytes = 2^{10}.
    - M(Mega) is equal to 1024Kbytes = 2^{20}
    - G(Giga) is equal to1024Mbytes = 2^{30}

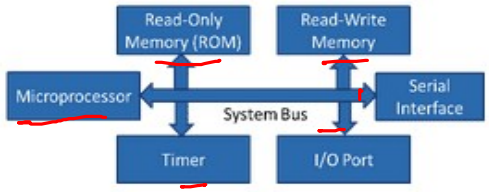
2^{10} = 1 KB → 1024 Bytes
2^{20} = 1 MB → 1024 KB
2^{30} = 1 GB → 1024 MB
2^{40} = 1 TB → 1024 GB

Computer Architecture

- **IO Device**
  - The Input unit allows programs and data to be entered into the computer.
  - e.g. Keyboard (primary), Mouse, Joystick, Touchpad, Touch pen, Scanner, Microphone, Webcam, Punch card, Bar code scanner, MICR scanner, Fingerprint, ...
  - The Output unit allows the results of processing to be exported to the outside world or other devices or saved to be used later.
  - e.g. Monitor (primary), printer, plotter, Speakers, projector, ...

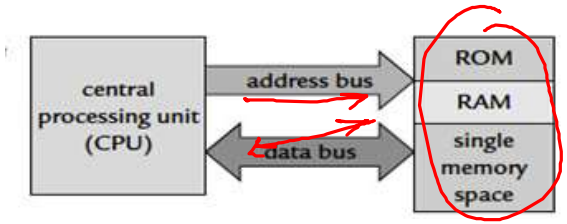
Difference between Microprocessor and Microcontroller

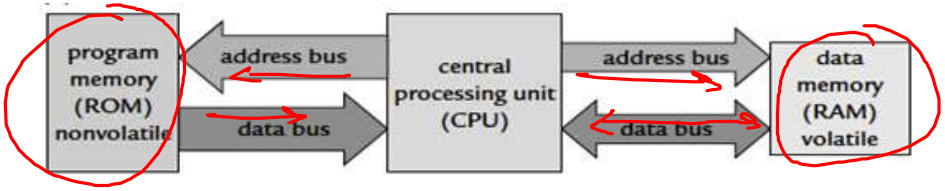
Microprocessor	Microcontroller
Microprocessor is heart of computer system	Microcontroller is heart of embedded system
Memory and IO components have to be connected externally	Microcontroller has external processor along with internal memory and IO components
Circuit becomes large	Circuit is small
Cannot be used in compact system and hence inefficient	Can be used in compact system and hence efficient
Cost and power consumption of entire system increases	Cost and power consumption of entire system is low
Relatively slower (each instruction needs external operation)	Speed is fast (most of the operations are internal operations)
Less number of registers (memory operations are more)	More number of registers
Microprocessors are based on Von Neumann	Microcontrollers are based on Harvard



Difference between Von Neumann and Harvard

Von Neumann	Harvard
Program and data memory is shared	Program and data memory are different
Processor needs two clocks cycles to execute instruction	Processor needs one clock cycle to execute instruction
Data transfers and instruction fetches can not be performed simultaneously	Data transfers and instruction fetches can be performed simultaneously
Used in personal computers, laptops and workstations	Used in microcontrollers and signal processing



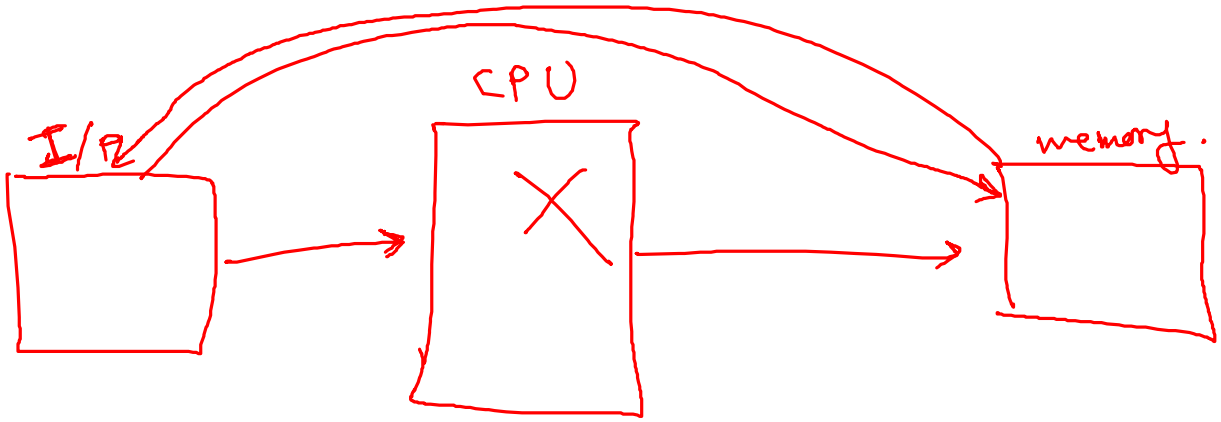


Difference between RISC and CISC

RISC	CISC
Less number of instructions	More number of instructions
All instructions are micro instructions - CPU don't further divide it into smaller parts	Some instructions are macro instructions - CPU further divide instruction in smaller parts and then executes
Most of the Instructions are executed in single CPU cycle	Need variable number of cycles
Instruction width is fixed	Instruction width is variable
large number of general purpose registers	less number of general purpose registers
Only load and store instructions can access memory	Many instructions access memory
Few addressing modes	Many addressing modes
eg. Power PC, ARM(Advanced RISC Machine), AVR(Advanced Virtual RISC Machine), SUN SPARC, 8051, MIPS	eg. 8080, 8085, 8086, 80286, 80386 Pentium etc

Computer Architecture

- **Direct Memory Access**
  - Removing the CPU from the path and letting the peripheral device manage the memory buses directly would improve the speed of transfer. This technique is known as DMA.
  - In this, the interface transfer data to and from the memory through memory bus. A DMA controller manages to transfer data between peripherals and memory unit.
  - Many hardware systems use DMA such as disk drive controllers, graphic cards, network cards and sound cards etc. It is also used for intra chip data transfer in multicore processors.



Computer Architecture

- **Interrupt**
  - Data transfer between the CPU and the peripherals is initiated by the CPU. But the CPU cannot start the transfer unless the peripheral is ready to communicate with the CPU. When a device is ready to communicate with the CPU, it generates an interrupt signal.
  - The main job of the interrupt system is to identify the source of the interrupt.

- **Priority Interrupt**
  - A priority interrupt is a system which decides the priority at which various devices, which generates the interrupt signal at the same time, will be serviced by the CPU.
  - For example, devices with high speed transfer such as magnetic disks are given high priority and slow devices such as keyboards are given low priority.
  - When two or more devices interrupt the computer simultaneously, the computer services the device with the higher priority first.

Computer Architecture

- **Types of Interrupts:**
- **Hardware Interrupts**
  - When the signal for the processor is from an external device or hardware then this interrupts is known as hardware interrupt.
  - Let us consider an example: when we press any key on our keyboard to do some action, then this pressing of the key will generate an interrupt signal for the processor to perform certain action.

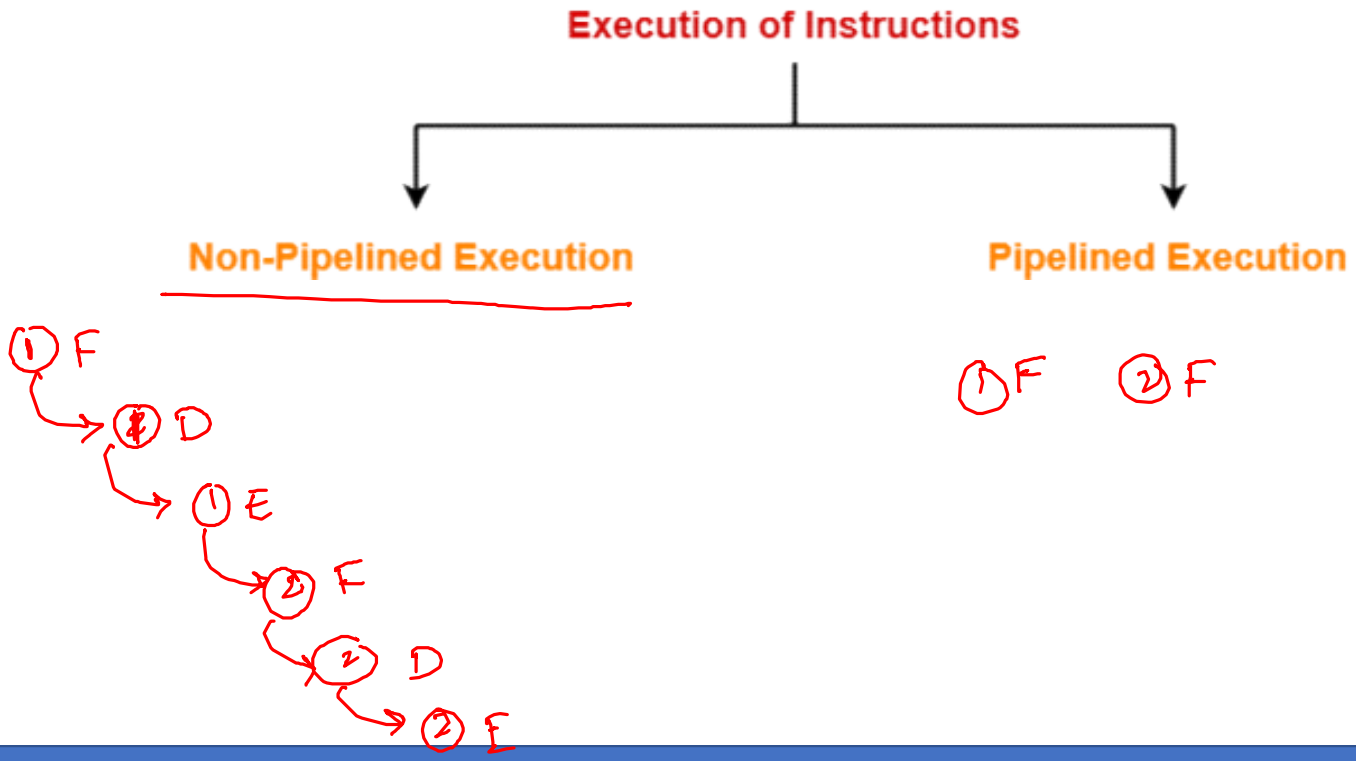
  - **Maskable Interrupt**
    - The hardware interrupts which can be delayed when a much high priority interrupt has occurred at the same time.
  - **Non Maskable Interrupt**
    - The hardware interrupts which cannot be delayed and should be processed by the processor immediately.

Computer Architecture

- **Software Interrupts**
  - The interrupt that is caused by any internal system of the computer system is known as a software interrupt.
    - **Normal Interrupt**
      - The interrupts that are caused by software instructions are called normal software interrupts.
    - **Exception**
      - Unplanned interrupts which are produced during the execution of some program are called exceptions, such as division by zero

Computer Architecture

- A program consists of several number of instructions.
- These instructions may be executed in the following two ways



Computer Architecture

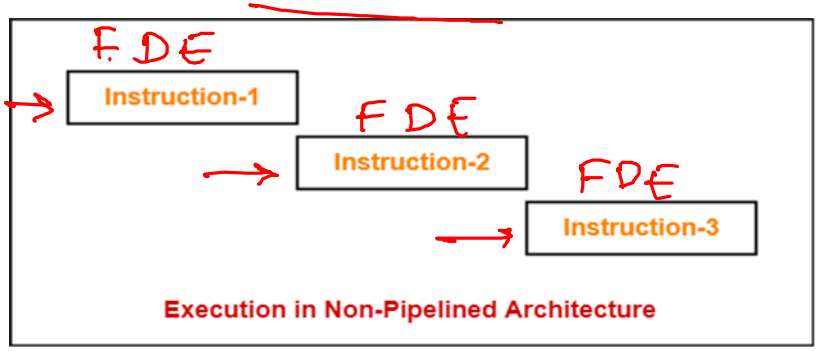
- **Non-Pipelined Execution**

  - In non-pipelined architecture, All the instructions of a program are executed sequentially one after the other.
  - A new instruction executes only after the previous instruction has executed completely.
  - This style of executing the instructions is highly inefficient.
  - Example:
    - Consider a program consisting of three instructions.
    - In a non-pipelined architecture, these instructions execute one after the other as
    - If time taken for executing one instruction = t, then-
    - Time taken for executing 'n' instructions = n x t

$n = 3, \quad t = 0.25 \text{ sec}$

$= 3 \times 0.25$

$= 0.75 \text{ sec}$



Execution in Non-Pipelined Architecture

Computer Architecture

- No. of cycle required to execute = K x N
- Where K à Number of stages N à Number of Instruction

- For Example :
- If we have 5 number of instruction and 4 number of stages then total how many cycle are required in non-pipelining ?

No. of cycles = K x N
= 4 x 5
= 20

Computer Architecture

- **Pipelined Execution-**
  - In pipelined architecture, Multiple instructions are executed simultaneously.
  - Pipelining is the process of accumulating instruction from the processor through a pipeline. It allows storing and executing instructions in an orderly process.

- **Four-Stage Pipeline-**
  - In four stage pipelined architecture, the execution of each instruction is completed in following 4 stages-

    - Instruction fetch (IF)
    - Instruction decode (ID)
    - Instruction Execute (IE)
    - Write back (WB)
    - Read back (RB) → 5

Computer Architecture

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7	Stage 8
Instruction 1	IF	ID	IE	WB				
Instruction 2		IF	ID	IE	WB			
Instruction 3			IF	ID	IE	WB		
Instruction 4				IF	ID	IE	WB	
Instruction 5					IF	ID	IE	WB

No. of cycle required to execute = K +(N -1)

Where K → Number of stages

N → Number of Instruction

For Example :

If we have 5 number of instruction and 4 number of stages then total how many cycle are required in pipelining ?

⇒ 4 + (5-1)

= 8 no. of cycles.

Computer Architecture

- Advantages of Pipelining
  - The cycle time of the processor is reduced.
  - It increases the throughput of the system.
  - It makes the system reliable.

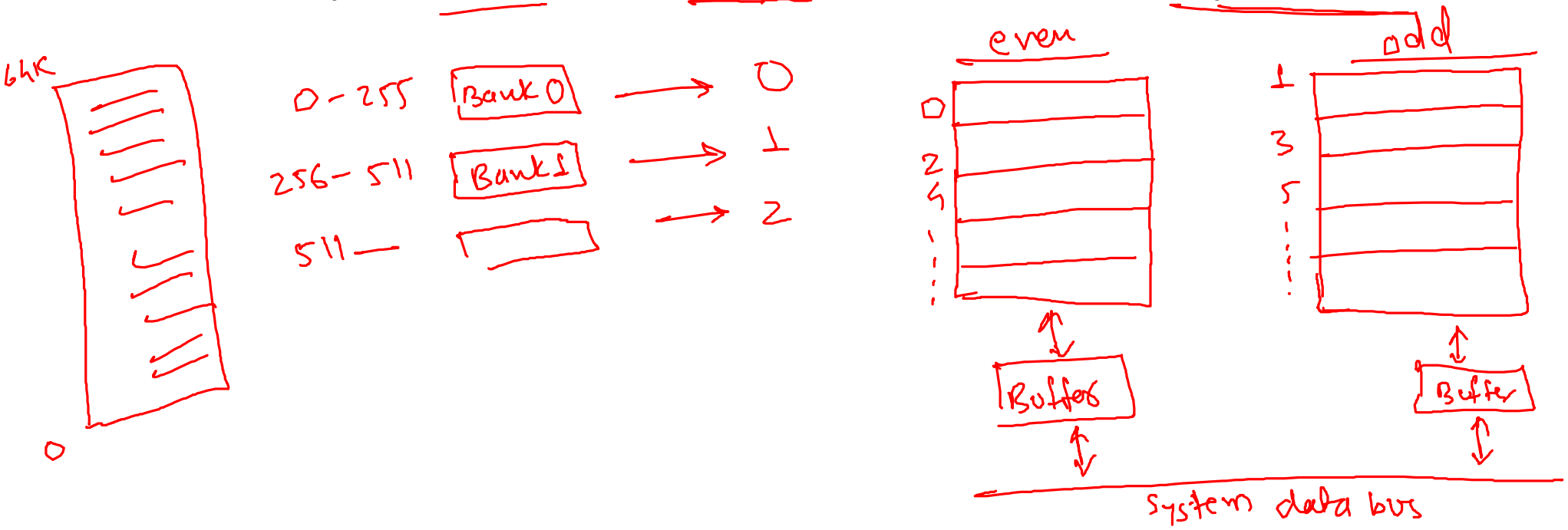
- Disadvantages of Pipelining
  - The design of pipelined processor is complex and costly to manufacture.

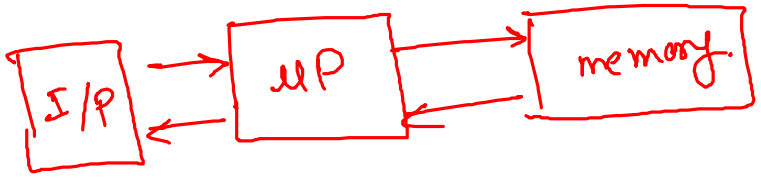
Computer Architecture

- **What is Interleaved Memory?**
- In this technique, the main memory is divided into memory banks which can be accessed individually without any dependency on the other.
- For example:
- If we have 4 memory banks(4-way Interleaved memory), with each containing 256 bytes, then, the Block Oriented scheme(no interleaving), will assign virtual address 0 to 255 to the first bank, 256 to 511 to the second bank. But in Interleaved memory, virtual address 0 will be with the first bank, 1 with the second memory bank, 2 with the third bank and 3 with the fourth.
- CPU can access alternate sections immediately without waiting for memory to be cached.
- Memory interleaving is a technique for increasing memory speed. It is a process that makes the system more efficient, fast and reliable

Computer Architecture

- The organization of two physical banks of n long words. All even long words of logical bank are located in physical bank 0 and all odd long words are located in physical bank 1.





4-bit → 4004 → Intel
8-bit → 8008 → 1972
↳ 8070 →
→ 8085 → 1976
16 bit → 8086 → pipeling

→ 8085
→ 8086
386
pentium
:

Section – C

Microprocessor 8085

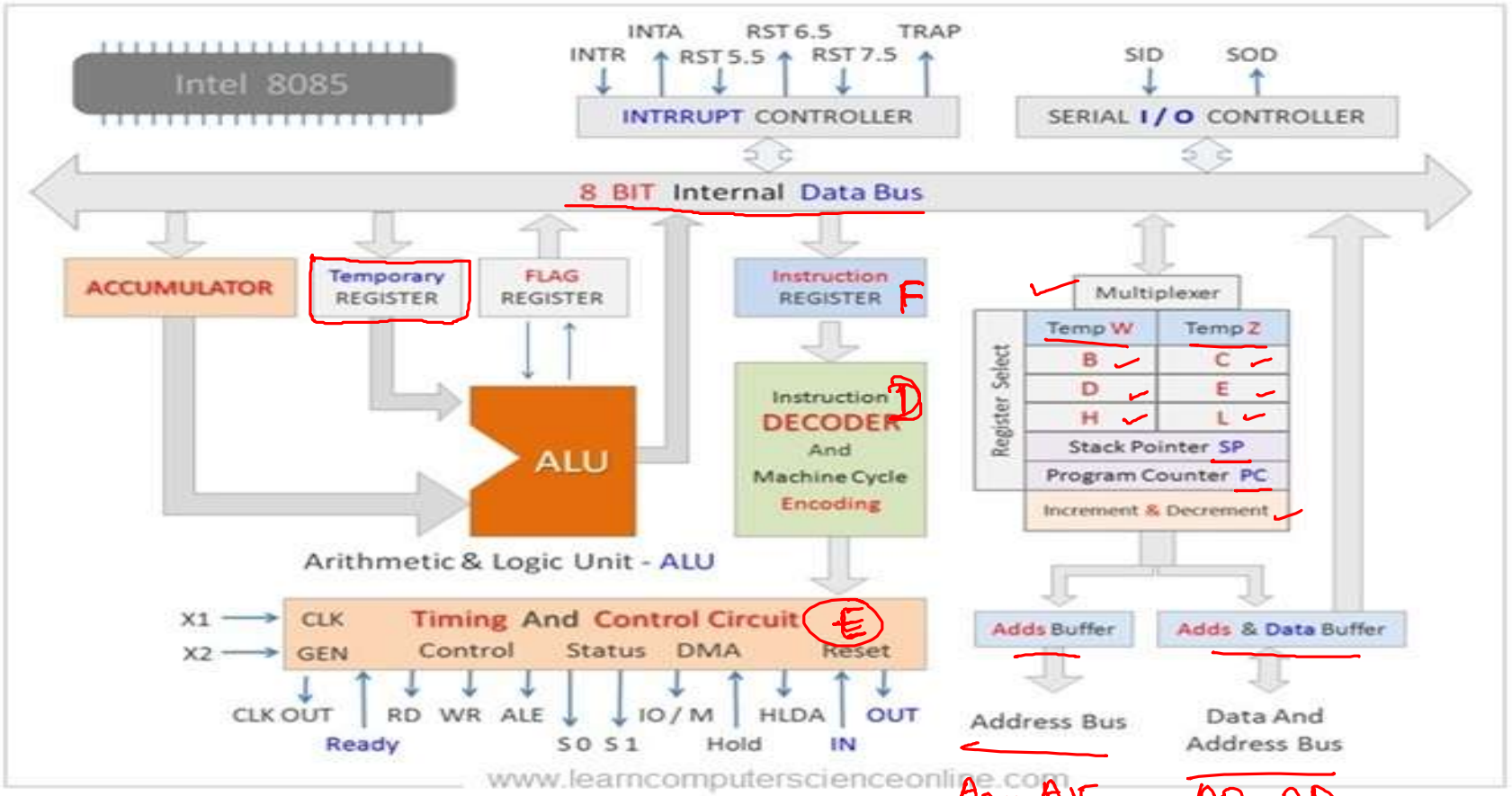
Microprocessor 8085

- 8085 architecture generally called as "eighty-eighty-five" microprocessor. It is an 8-bit microprocessor which was introduced by Intel in the year 1976 using NMOS technology.
- It has the following configuration − § 8-bit data bus.
- 16-bit address bus, which can address up to 64KB
- A 16-bit program counter
- A 16-bit stack pointer
- Six 8-bit registers arranged in pairs: BC, DE, HL
- Requires +5V supply to operate at 3 MHZ single phase clock

Microprocessor 8085

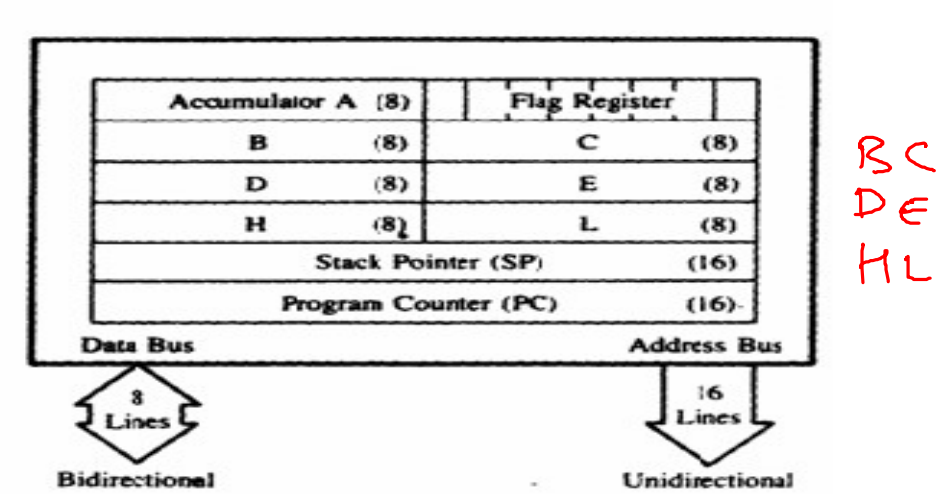
- **Bus Structure in 8085**
  - There are three buses in Microprocessor: 1. Address Bus 2. Data Bus 3. Control Bus
- **Data Bus**
  - The data bus width is 8-bit i.e. $2^8 = 256$ combination of binary digits, identified as D0–D7. Data bus is bidirectional since it carries data in binary form between microprocessor and other external units such as memory. It is used to transmit data. As it is 8-bit wide then largest number is 11111111.
- **Address Bus**
  - 8085 microprocessor contains 16-bit address bus i.e. $2^{16} = 65536 = 64KB$ memory location it can access, identified A0 – A15. Address bus is unidirectional. The address bus carries addresses from microprocessor to the memory or other devices. The higher order address lines are A8 – A15 and the lower order lines A0 – A7 are multiplexed with the eight bits data lines (D0 – D7)..

Microprocessor 8085



- **Accumulator**
  - It is an 8-bit register used for general purposes. It also helps in arithmetic, logical, I/O & LOAD/STORE operations. It is connected to internal data bus & ALU. The data is stored in this register. Arithmetic and logic unit.
  - It is helpful in arithmetic and logical operations like Addition, Subtraction, AND, OR, etc. on 8-bit data.
- **General purpose register.**
  - There are 6 general purpose registers in 8085 processor, i.e. B, C, D, E, H & L. Each register can hold 8-bit data.
  - These registers can work in pairs in order to hold 16-bit data and their pairing combination looks like B-C, D-E & H-L.
- **Program counter.**
  - It is a type of 16-bit register used to store the address of the instructions that is to be executed. Whenever each instruction get fetched from program counter its store value increases by 1.

- **Stack pointer**
  - It is also a 16-bit register works like stack, which is always incremented/ decremented by 2 during push & pop operations.
- **Temporary register**
  - It is an 8-bit register, as the name suggests it holds the temporary data of arithmetic and logical operations. W and Z are temporary register.



- **Flag register**
- It is an 8-bit register which has five 1-bit flip-flops, which contains either 0 or 1 based on the result data that is stored in the accumulator.

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

- Sign (S)
- Zero (Z)
- Auxiliary Carry (AC)
- Parity (P)
- Carry (C)

- **Instruction register and decoder**

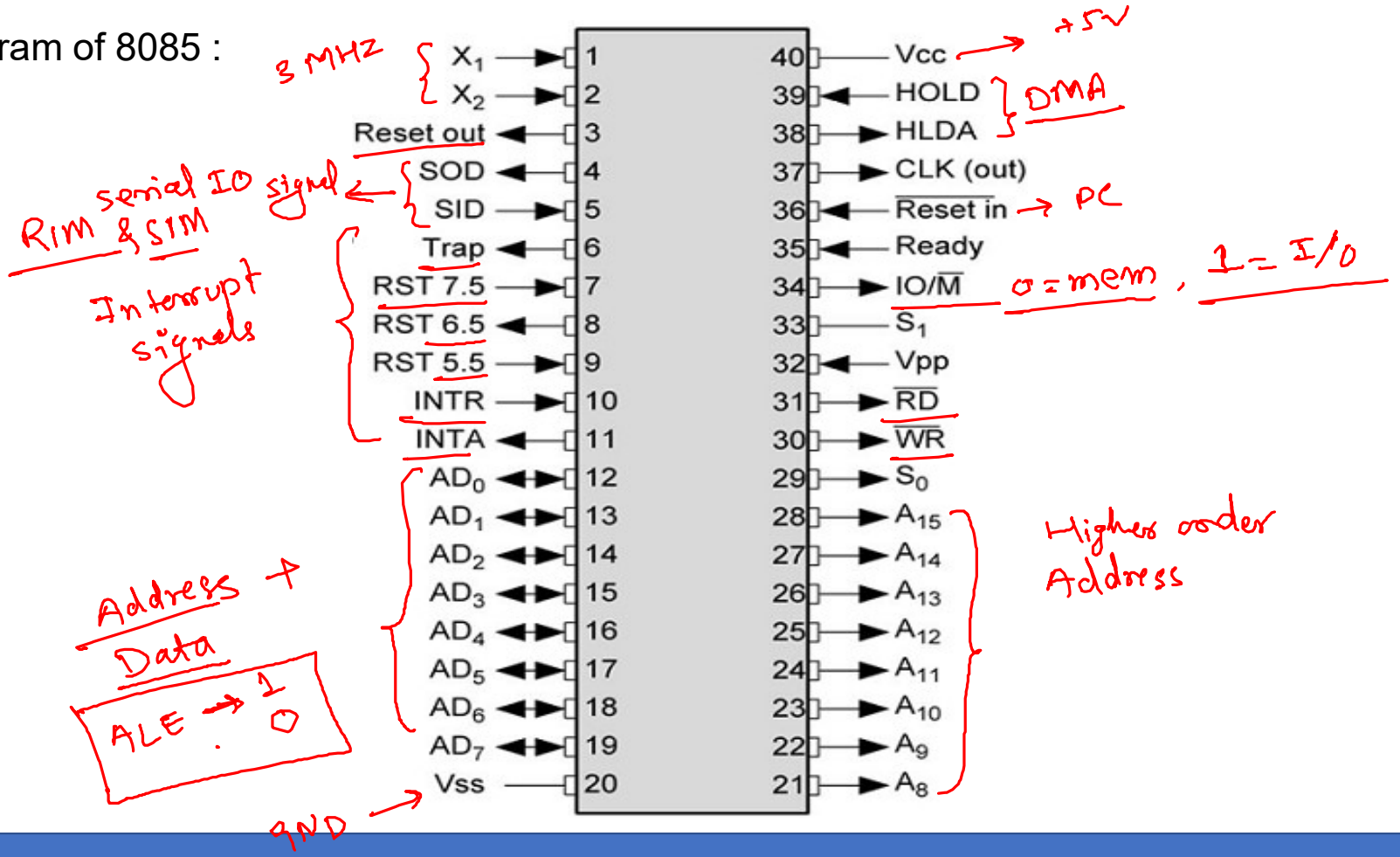
- The instructions that is fetched from the memory. Instruction decoder decodes the information present in the Instruction register.

- **Timing and control unit**

  - It supplies timing and control signal to the microprocessor to perform certain operations.
  - Following are the timing and control signals, which control external and internal circuits −
  - Control Signals: READY, RD', WR', ALE
  - Status Signals: S0, S1, IO/M'
  - DMA Signals: HOLD, HLDA
  - RESET Signals: RESET IN, RESET OUT

- 8085 is a 40pin IC. Works on +5v power supply with 3MHz frequency. 8085 signals are classified into six groups:
  - Address bus
  - Data bus
  - Control & status signals
  - Power supply and frequency signals
  - Externally initiated signals & Serial I/O signals

- Pin Diagram of 8085 :



- **Address bus**
- A15-A8, it carries the most significant 8-bits of memory/IO address.
- **Data bus**
- AD7-AD0, it carries the least significant 8-bit address and data bus.
- **Control and status signals**
- These signals are used to recognize the nature of operation. There are 3 control signal and 3 status signals. Three control signals are RD, WR & ALE.
- **RD** − This signal signifies the selected IO or memory device that is to be read and helps in accepting data readily available on the data bus.
- **WR** − This signal indicates that the data on the data bus is to be written into a selected memory or IO location.
- **ALE** − It is Address Latch Enable signal. It is a positive going pulse produced when a new operation is started by the microprocessor. When the pulse goes high, it indicates address. When the pulse goes down it indicates data.

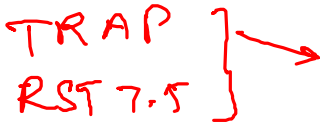
- Three status signals are IO/M, S0 & S1.
- **IO/M**
- This signal is used to distinguish between IO and Memory operations, i.e. when it is high indicates IO operation and when it is low then it indicates memory operation.
- **S1 & S0**
- These signals are used to identify the type of current operation.
- **Power supply**
- There are 2 power supply signals − VCC & VSS. VCC indicates +5v power supply and VSS indicates ground signal.
- There are 3 clock signals, i.e. X1, X2, CLK OUT.
- **X1, X2** − A crystal is connected at these two pins and is used to set frequency of the internal clock generator. This frequency is internally divided by 2.
- **CLK OUT** − This signal is used when the system clock for devices connected with the microprocessor.

- **Interrupts & externally initiated signals**
  - Interrupts are the signals produced by external devices to request the microprocessor to perform a specific task. There are 5 interrupt signals, i.e. TRAP, RST 7.5, RST 6.5, RST 5.5, and INTR.
- **INTA** − It is an interrupt acknowledgment signal.
- **RESET IN** − This signal is used to reset the microprocessor by setting the program counter to zero.
- **RESETOUT**− This signal is used to reset all the connected devices when the microprocessor is reset.
- **READY** − This signal indicates that the device is ready to send or receive data. If READY is low, then the CPU has to wait for READY to go high.
- **HOLD** − This signal gives hint that another master is requesting the use of the address and data buses.
- **HLDA** (HOLD Acknowledge) − It indicates that the CPU has received the HOLD request and it will relinquish the bus in the next clock cycle. HLDA is set to low after the HOLD signal is removed

- **Serial I/O signals**
- There are 2 serial signals, i.e. SID and SOD and these signals are used for serial communication.
- **SOD** (Serial output data line) – According to the SIM instruction the output SOD is set/reset.
- **SID** (Serial input data line) - Whenever the data instruction is executed the data on this line is loaded into accumulator

- **Interrupts in 8085**
- Interrupts are the signals that are sent by the external devices to the microprocessor to perform a particular task or work in request format. There are 5 interrupt signals, i.e. TRAP, RST 7.5, RST 6.5, RST 5.5, and INTR.
- Interrupt are classified into following groups based on their parameter −
- **Vector interrupt** − In this type of interrupt, the interrupt address is known to the processor.
- For example: RST7.5, RST6.5, RST5.5, TRAP.
- **Non-Vector interrupt** − In this type of interrupt, the interrupt address is not known to the processor so, the interrupt address require to send externally by the device to perform interrupts.
- For example: INTR. + Address
- **Maskable** interrupt − In this type of interrupt, you can neglect the interrupt by giving some instructions to the program.
- For example:RST7.5, RST6.5, RST5.5

- **Non-Maskable interrupt** − In this type of interrupt, interrupts cannot be disabled by writing some instructions into the program.
- For example: TRAP.
- **Software interrupt**− In this type of interrupt, as it is software the programmer has to give the instructions to the program in order to execute the interrupts. There are 8 software interrupts in 8085, i.e. RST0, RST1, RST2, RST3, RST4, RST5, RST6, and RST7.
- Hardware interrupt − There are 5 interrupt pins in 8085 used as hardware interrupts, i.e. TRAP, RST7.5, RST6.5, RST5.5, INTA.
- Note − TRAP has the highest priority, then RST7.5.



- **TRAP**
- It is called as a non-maskable interrupt, which has the highest priority among all interrupts. It is subroutine calls which are forced by the microprocessor when it identifies any interruption in the instructions. So, when the power failure occurs, it executes as ISR and sends the data to backup memory. This interrupt transfers the control to the location 0024H.
- **RST7.5**
- It is called as maskable interrupt, which has the second highest priority among all interrupts. When this interrupt is executed, the processor saves the content of the PC register into the stack and branches to 003CH address.
- **RST 6.5**
- It is a maskable interrupt, having the third highest priority among all interrupts. When this interrupt is executed, the processor saves the content of the PC register into the stack and branches to 0034H address

- **RST 5.5.**
- It is a maskable interrupt. When this interrupt is executed, the processor saves the content of the PC register into the stack and branches to 002CH address.
- **INTR**
- It is a maskable interrupt, having the lowest priority among all interrupts. It can be deleted by resetting the microprocessor.