

MODULE-1

Overview of Computer Architecture & Organization

Computer Architecture :- Abstract
(What to do)

Computer Organization :- Implementation
(How to do)

Computer Architecture :-

It refers to how hardware components connect to form a computer system.
→ It's a design blueprint & decided first
→ It acts as an interface between hardware & software.

Computer Organization :-

It refers to the structure & behaviour of the computer system as seen by the user.
→ It's decided after computer architecture.
→ It deals with the interconnection of components.

Difference between CA and CO

Computer Architecture

1) Architecture describes what the computer does.

Computer Organization

The organization describes how it does it.

CA

It deals with the functional behavior of computer systems.

It deals with high level design issues.

Architecture indicates its hardware.

As a programmer, you can view architecture as a series of instructions, addressing modes and registers.

For designing a computer, its architecture is fixed first.

CA is also called ISA (Inst^r set Arch^t)

CA comprises logical fun such as inst^r sets, registers, data types & addressing modes.

It makes the computer's H/w visible.

Arch^t coordinates the H/w & S/w of the system.

CO

CO deals with a structural relationship.

It deals with low level design issues.

Organization indicates its performance.

The implementation of the arch^t is called organization.

For designing a computer, an organization is decided after its arch^t.

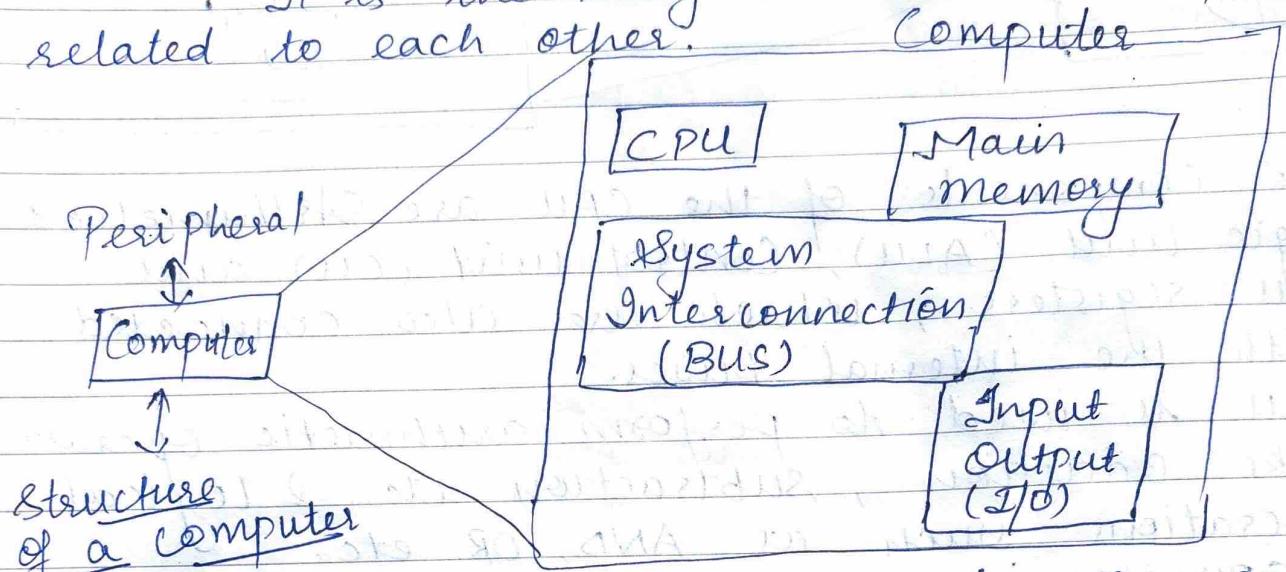
CO is frequently called microarchitecture.

CO consists of physical units like circuit designs, peripherals & address.

It offers details on how well the computer performs.

CO handles the segments of the N/w in a system.

Basic Organization of Computer & Block Level Description of the functional Units: Structural components of a computer. It is the way in which components related to each other.



The components of a computer are central processing unit, Input and Output (I/O) devices and memory.

The three units are connected with the help of system interconnection i.e. buses.

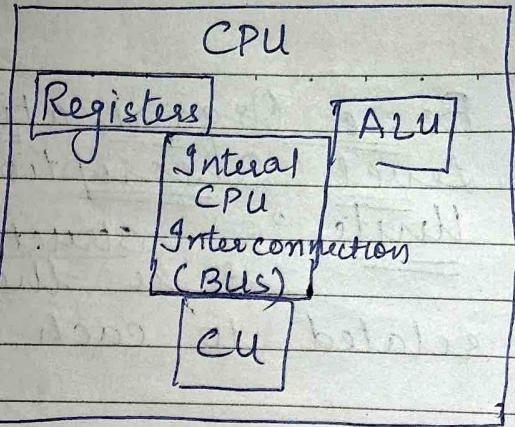
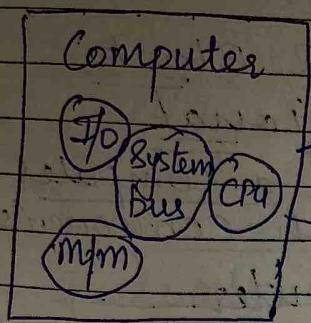
Memory is used to store code (Program) and data. It can be various kinds of like Semiconductor memory using ICs, magnetic memory or optical memory. etc

I/O devices are used to accept a input or give an output by the CPU.

These are various input devices like Keyboard, mouse, scanner and various output devices like CRT, Printer etc.

The CPU is further divided into three units

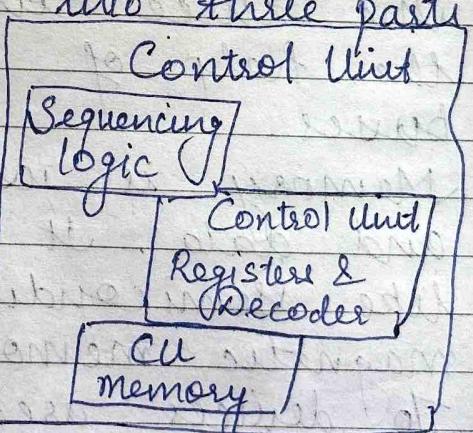
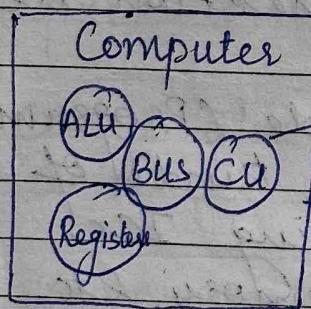
Structure of a CPU:



The Components of the CPU are Arithmetic & Logic Unit (ALU), Control unit (CU) and CPU registers, which are also connected with the internal buses.

- ALU is used to perform arithmetic operations like addition, subtraction etc. & logical operations such as AND, OR etc.
- CPU registers are used to store the data temporarily in the CPU to save memory access time.

The CU is further divided into three parts



Structure of Control unit:

Control unit comprises of control memory, Control unit register & sequencing logic. The control memory stores the micro inst & loads it into the control unit register & the sequencing logic gives these signals in a proper sequence to execute a instruction.

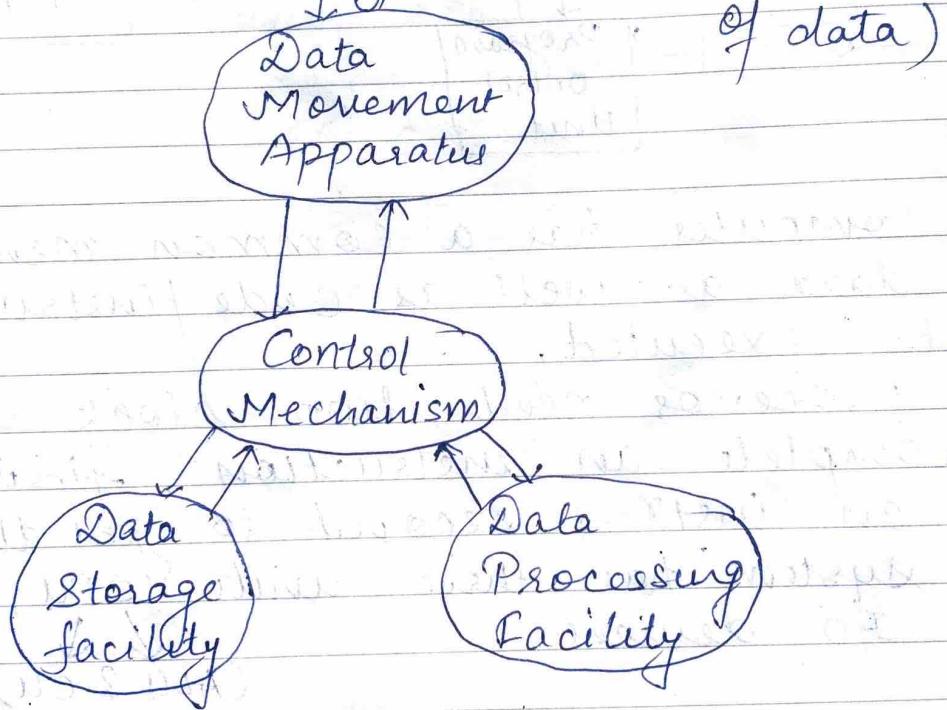
Functional View of a Computer

It is the operation of the individual component as the part of the structure.

The functions of a computer are data processing, data storage, data movement and control.

Various path followed by the data are:-

operating Env^o (source & destination)



The data may be taken into the processor from the input device, processed & then the result may be given at the output device

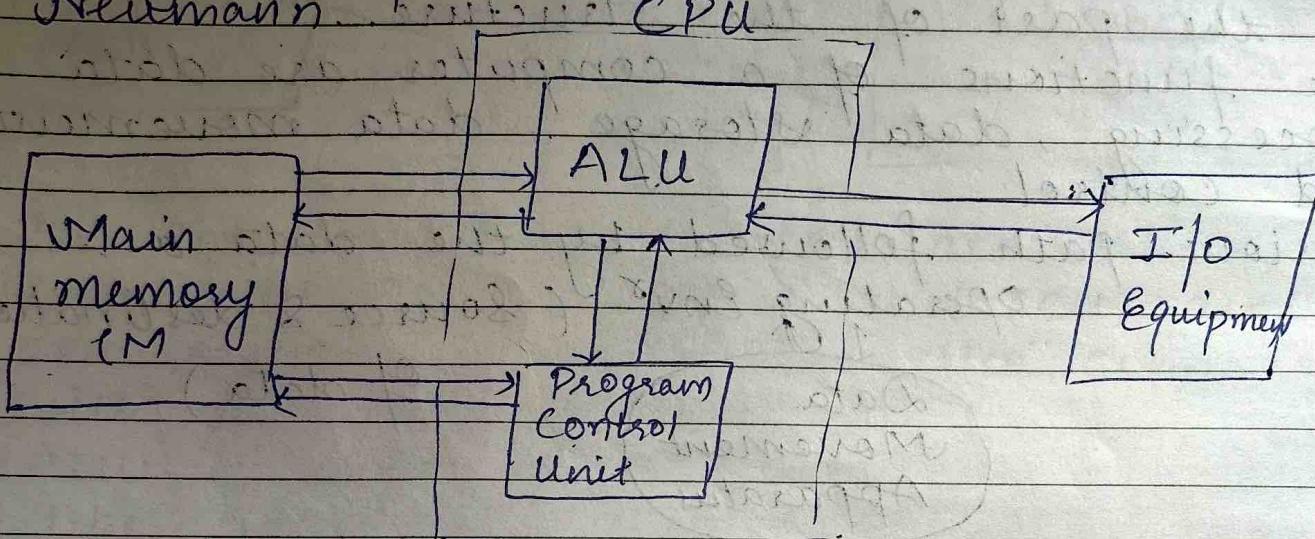
The data may be taken into the processor from the input device, processed and the result stored in memory

The data may be taken from memory, processed & the result given to output device

The data taken from an input device &

Von Neumann Architecture

The name is derived from the mathematician and early computer scientist John Von Neumann.



The computer has a common memory for data as well as code/instructions to be executed.

The processor needs two clock cycles to complete an instruction, first to get an instr & second to get the data. The system has three units GPU, Memory and I/O devices.

(ALU & CU)

1) Input Unit:

A computer accepts input from the user through these devices i.e. input devices.

Some input devices are mouse, Keyboard, Camera, Scanner, Joystick etc.

The devices input the data accepted from the user in a proper coded form understood by the computer.

2) Output unit:-

The result is given back by the computer to the user through an output device.

Input devices and output devices are also called as human interface devices because they are used to interface the human to the computer.

Some output devices are printer, monitor, plotter, speaker etc.

3) Arithmetic & Logic Unit (ALU):-

ALU like multiplication, addition, division, AND, OR, EXOR etc are performed by ALU. Operands are brought into the ALU, where the necessary operation is performed.

4) Control Unit:-

It is the main unit that controls all the operations of the system, inside and outside the processor.

The I/m or I/O devices have to be controlled by the computer to perform the operation according to the instruction given to it.

5) Memory unit:-

It is used to store the program (instⁿ) and data for the computer.

The inst^s from the programs are taken by the processor, decoded and executed accordingly.

The data is also stored in the memory. The data is taken from memory and the operation is performed on that data, as well as the results are stored in the memory.

Memory in the Von Neumann system has a special organization wherein the data and instructions are stored in the same memory.

Key features of a Von Neumann m/c Arch^s

The Von Neumann m/c uses stored program concept.

The program and data are stored in the same memory unit.

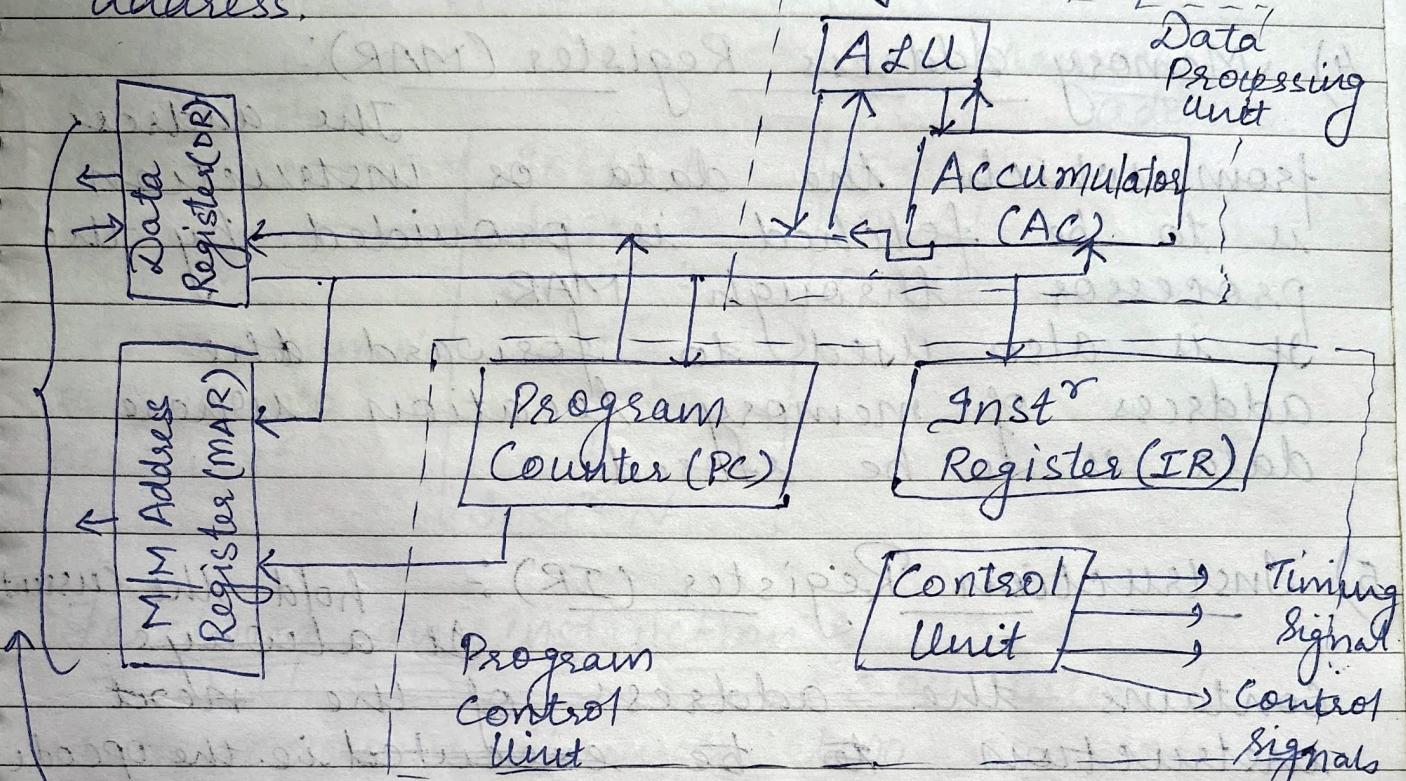
Each location of the memory has a unique address i.e. no. two locations have the same memory address.

Execution of instruction in Von Neumann m/c is carried out in a sequential manner (unless explicitly altered by the program itself) from one inst^s to the next.

Detailed Structure of the CPU

The block diagram of the computer proposed by Von Neumann have a minimal number of

registers along with the above blocks. This computer has a small set of instruction and an instruction was allowed to contain only one operand address.



The structure contains following registers:-

1) Accumulator (AC):

It provides one of the operand to ALU and store the result.

2) Data Register:

It acts as buffer storage between the CPU and main memory or I/O devices.

(PC)

3) Program Counter:-

It always contains the address of the next instruction to be executed.

4) Memory Address Register (MAR):-

The address from which the data or instruction is to be fetched is provided by the processor through MAR.

It is also used to forward the address of memory location where data is to be stored.

5) Instruction Register (IR):-

hold the current It always contains the address of the next instruction to be executed i.e. the opcode & operand of the inst^r to be executed.

Execution of a Program by Von Neumann:-

The program to be executed is stored in memory.

A register, PC always points to the first instruction of the program when the computer starts. CPU fetches the instruction pointed by PC.

PC contents are automatically incremented to point to the next instruction.

If the initial value of PC = 00000000 (in binary), first instruction to be fetched for execution.

After fetching "Instruction Number 1", PC will be incremented by one. It is assumed that the size of each instruction is 1 byte.

$$PC = PC + 1 = 0 + 1$$

$\underbrace{\text{Inst}^x \text{ pointed}}_{\text{by PC if fetched}} = 1$

PC	Memory address
	0000
1	0001
↑	0 0 1 0
PC is incremented to point to next inst ^x	Inst ^x N

Fetching an instruction :-

CPU interacts with memory through two special registers.

1) MAR (Memory Address Register) :-

It provides address of memory location from where data or instruction is to be retrieved or to which data is to be stored.

2) DR (Data Register) :-

It acts as buffer storage between the main memory and the CPU.
Let's take an example:-

The inst^r to be executed is brought from the memory to the CPU, through following steps:-

- 1) The address of the inst^r is transferred from PC to MAR
 $\text{MAR} \leftarrow \text{PC}$
- 2) MAR puts this address on the address bus for selection of the required location of the memory.
- 3) Control unit generates the RD (read control signal) signal to perform read operation on memory. Required instruction is given on data bus by the memory. Instruction on data bus is accepted in DR (Data Register).

Execution of instruction:

The fetched instruction is in the form of binary code and is loaded into instruction register (IR) from DR (Data Register).

- The instruction specifies what action the CPU has to take.
- The CPU interprets the instruction & performs the required action.
The action could be:-

- 1) Data transfer b/w CPU & memory
- 2) Data transfer b/w CPU & I/O
- 3) The CPU may perform an arithmetic or logic operation on data.

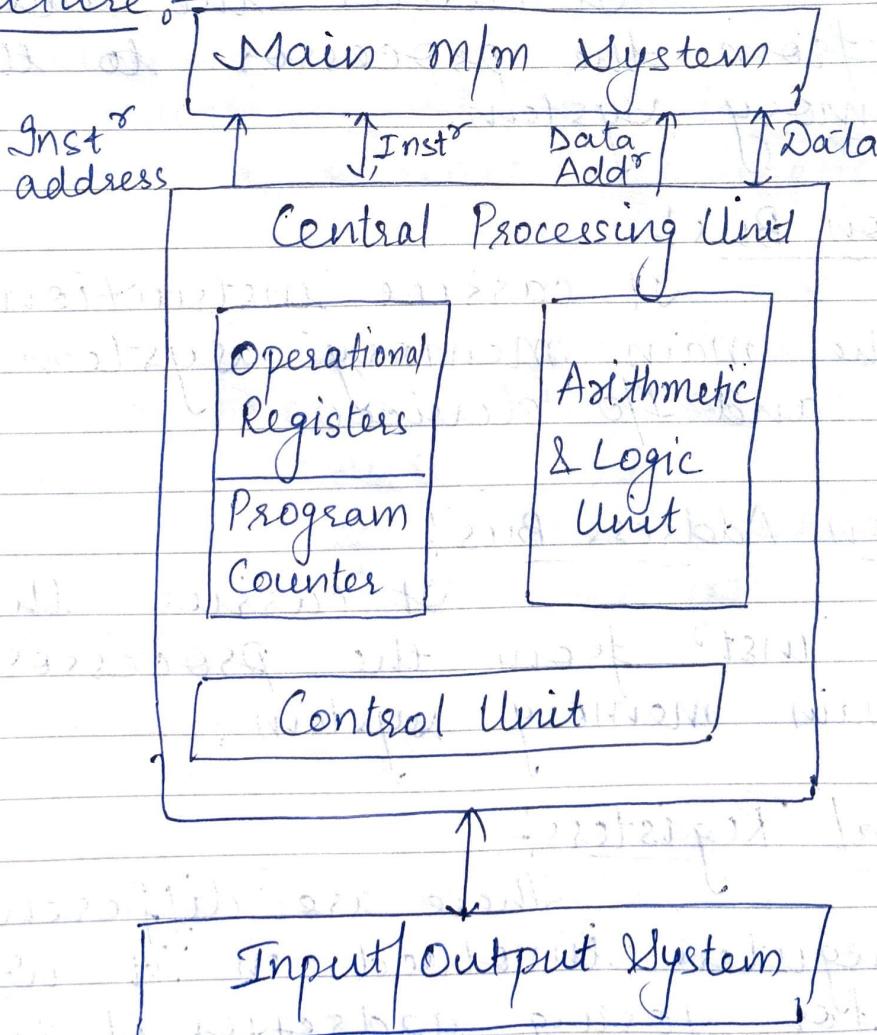
Harvard Model

The name is originated from Harvard's Mark I a relay base old computer.

Harvard Arch^r is the Computer arch^r that contains separate storage & separate buses (signal path) for inst^r & data.

- It was basically developed to overcome the bottleneck of Von Neumann's Arch^r. The main advantage of having separate buses for inst^r & data is that the CPU can access inst^r & read/ write data at the same time.

Structure



Buses

Buses are used as signal pathways. In Harvard Arch, there are separate buses for both inst & data types.

Types of Buses:

a) Data Bus:

It carries data among the main memory system, processor and I/O devices.

b) Data Address Bus:

It carries the address of data from the processor to the main memory system.

c) Instruction Bus:

It carries instructions among the main memory system, processor and I/O devices.

d) Instruction Address Bus:

It carries the address of inst^x from the processor to the main memory system.

Operational Registers:-

These are different types of registers involved in it which are used for storing addresses of different types of inst^x. For example - the memory

address register & memory data register
are operational registers.

1) Program Counter:

It has the location of the next instruction to be executed. The program counter then passes this next address to the memory address register.

2) Arithmetic & Logic Unit:

The arithmetic logic unit is part of the CPU that operates all the calculations needed. It performs addition, subtraction, comparison, logical operations, bit shifting operations & various arithmetic operations.

3) Control Unit:

The CU is the part of the CPU that operates all processor control signals. It controls the input & output devices & also controls the movement of inst^r and data within the system.

4) Input Output System:

Input devices are used to read data into main m/m with the help of CPU input inst^r. The information from a computer as output is given through output devices. The computer gives the result of computation with the help of output devi

Features:

- Separate Memory Spaces
- Fixed instruction length
- Parallel inst & data access
- More efficient memory usage
- Suitable for embedded systems
- Limited flexibility

Advantages:

- Fast & efficient data access
- Better Performance
- Suitable for real time applications
- Security

Disadvantages:

- Complexity
- Limited flexibility
- Higher memory requirements
- Code size limitations

Performance Issues:

There are various parameters that are used to measure the performance of a parallel system.

Some performance parameters are:-

- 1) Sequential execution time: The time required for a program to be executed on a sequential

(uni-processor) system is called as the sequential execution time with respect to parallel processor. It is represented by $T(1)$.

2) Parallel Execution time:

The time required for a program to be executed on a parallel processor system is called as parallel execution time for 'n' processors. It is represented by $T(n)$, where n is the number of processors.

3) Speed-up:

The speed increase because of the parallel system compared to the uni-processor system is called as the speed up.

It is the ratio of the speed of parallel system to that of sequential system.

It can also be given as the ratio of time required to execute a program on sequential system to that of the parallel system.

It is a very important metric to measure the performance of a parallel system.

It is represented as $S(n)$ and given as below:

$$S(n) = \frac{T(1)}{T(n)}$$

4) Efficiency :- It is the ratio of the actual speed up obtained by a system to the ideal speed up that should be achieved according to the number of processors in the parallel system.

$$\text{Efficiency } (\eta) = \frac{\text{Actual Speed-up}}{\text{Ideal Speed-up}} = \frac{T(1)}{nT(n)}$$

$\frac{1}{T(n)} = \text{Actual Speed up} \quad \frac{n}{T(1)} = \text{Ideal Speed up}$

5) Clocks Per Instruction (CPI):-

This is as the name says a measure of the clock pulses required per instruction.

It is the ratio of the clock cycles required for a program to the number of instructions in the program.

6) Million Instruction Per Second (MIPS):-

It is the count of instructions executed per unit second in millions.

For e.g., if a system has 5 MIPS, it means it can execute 5 million instructions in a second.

7) Millions Floating Point instⁿ per second (MFLOPS):- This is

similar to the MIPS, only the difference being here floating point instructions are taken into account.

8) Throughput: The throughput of a system is defined as the number of programs executed per unit time.

$$W_s = \frac{\text{Number of Programs}}{\text{Time in seconds}}$$

9) Scalability: A system is said to be scalable if the same efficiency is obtained by increasing the number of processors.