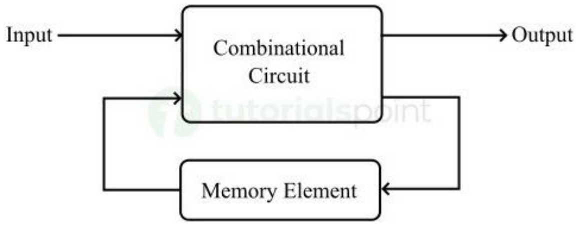
Digital Logic Circuits-II: Sequential Circuits, Flip Flops, Binary counters, Registers, Shift Registers, Ripple counters. Basic Structure of Computers: Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations, Von-Neumann Architecture.

Sequential Circuit: A sequential circuit is a type of digital logic circuit whose output depends on present inputs as well as past operation of the circuit.

A **sequential circuit** is a logic circuit that consists of a memory element to store history of past operation of the circuit. Therefore, the output of a sequential circuit depends on present inputs as well as past outputs of the circuit.

The **block diagram of a typical sequential circuit** is shown in the following figure −



Here, it can be seen that a sequential circuit is basically a combination of a combinational circuit and a memory element. The combinational circuit performs the logical operations specified, while the memory element records the history of operation of the circuit. This history is then used to perform various logical operations in future.

Sequential Circuits vs. Combinational Circuits:

Given below is a list of some of the main advantages of using sequential circuits over combinational logic circuits −

- Sequential circuits can retain the operation history which is important in various applications like data storage, feedback control systems, etc.
- Sequential circuits exhibit dynamic behaviour and can execute complex operation in real time.
- Sequential circuits comprise a feedback mechanism which improves the stability and optimizes the system performance.
- Synchronous sequential circuits use a common clock signal for synchronization that ensures reliable operation of the circuit.
- Sequential circuits can perform more complex operations using simpler circuit designs than combinational circuits. Hence, their hardware complexity is lesser.

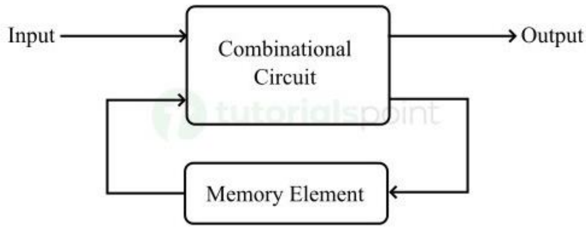
Types of Sequential Circuits: Based on structure, operation, and applications, the sequential circuits are classified into the following two types −

- Asynchronous Sequential Circuit
- Synchronous Sequential Circuit

Asynchronous Sequential Circuit

A type of sequential circuit whose operation does not depend on the clock signals is known as an asynchronous sequential circuit. This type of sequential circuits operates using the input pulses that means their state changes with the change in the input pulses.

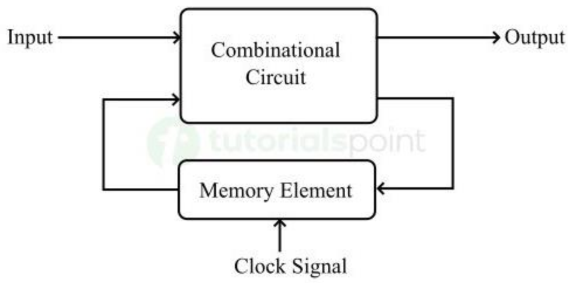
The main components of the asynchronous sequential circuits include un-clocked flip flops and combinational logic circuits. The block diagram of a typical asynchronous sequential circuit is shown in the following figure.



From this diagram, it is clear that an asynchronous sequential circuit is similar to a combinational logic circuit with a feedback mechanism.

Synchronous Sequential Circuit: A synchronous sequential circuit is a type of sequential circuit in which all the memory elements are synchronized by a common clock signal. Hence, synchronous sequential circuits take a clock signal along with input signals.

In synchronous sequential circuits, the duration of the output pulse is equivalent to the duration of the clock pulse applied. Take a look at the block diagram of a typical synchronous sequential circuit –



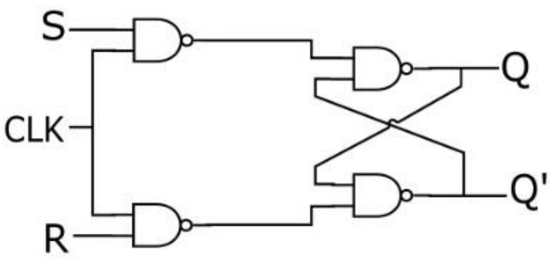
Flip-Flops: A flip-flop is a sequential digital electronic circuit having two stable states that can be used to store one bit of binary data. Flip-flops are the fundamental building blocks of all memory devices.

Types of Flip-Flops

- S-R Flip-Flop
- J-K Flip-Flop
- D Flip-Flop
- T Flip-Flop

S-R Flip-Flop:

This is the simplest flip-flop circuit. It has a set input (S) and a reset input (R). When in this circuit when S is set as active, the output Q would be high and the Q' will be low. If R is set to active then the output Q is low and the Q' is high. Once the outputs are established, the results of the circuit are maintained until S or R get changed, or the power is turned off.



Truth Table of S-R Flip-Flop

S	R	Q	State
0	0	0	No Change
0	1	0	Reset
1	0	1	Set
1	1	X	

Characteristics Table of S-R Flip-Flop

S	R	Q(t)	Q(t+1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X
1	1	1	X

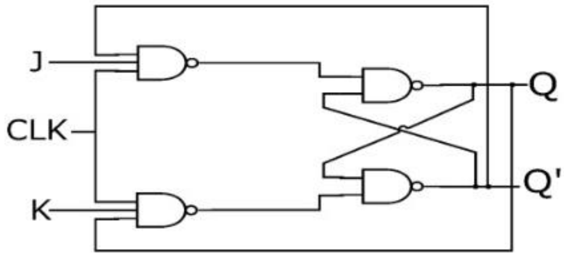
Characteristics Equation of S-R Flip-Flop

$Q(t+1)=S+R'Q(t)$

J-K Flip-Flop:

Because of the invalid state corresponding to S=R=1 in the SR flip-flop, there is a need of another flip-flop. The JK flip-flop operates with only positive or negative clock transitions. The operation of the JK flip-flop is similar to the SR flip-flop. When the input J and K are different then the output Q takes the value of J at the next clock edge.

When J and K both are low then NO change occurs at the output. If both J and K are high, then at the clock edge, the output will toggle from one state to the other.



Truth Table of JK Flip-Flop

J	K	Q	State
0	0	0	No Change
0	1	0	Reset
1	0	1	Set
1	1	Toggles	Toggle

Characteristics Table of JK Flip-Flop

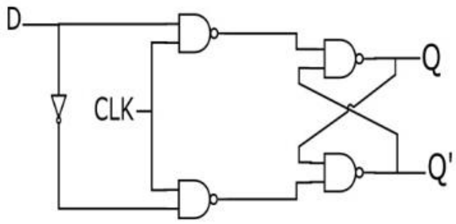
J	K	Q(t)	Q(t+1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Characteristics Equation of JK Flip-Flop

$Q(t+1)=jkQ(t)'+K'Q(t)$

D Flip-Flop:

In a D flip-flop, the output can only be changed at positive or negative clock transitions, and when the inputs changed at other times, the output will remain unaffected. The D flip-flops are generally used for shift-registers and counters. The change in output state of D flip-flop depends upon the active transition of clock. The output (Q) is same as input and changes only at active transition of clock.



Truth Table of D Flip-Flop:

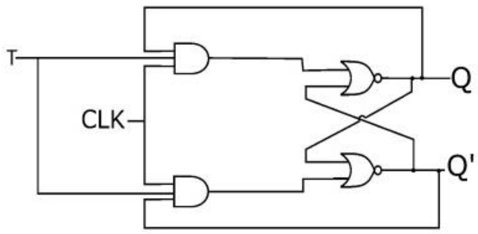
D	**Q**
0	0
1	1

Characteristics Equation of D Flip-Flops

$Q(t+1)=D$

T Flip-Flop:

A T flip-flop (Toggle Flip-flop) is a simplified version of JK flip-flop. The T flop is obtained by connecting the J and K inputs together. The flip-flop has one input terminal and clock input. These flip-flops are said to be T flip-flops because of their ability to toggle the input state. Toggle flip-flops are mostly used in counters.



Truth Table of T Flip-Flop

T	Q(t)	Q(t+1)
0	0	0
0	1	1
1	0	1
1	1	0

Characteristics Equation of T Flip-Flop

$Q(t+1)=T'Q(t)+TQ(t)'=T\oplus Q(t)$

Counters: A Counter is a device which stores (and sometimes displays) the number of times a particular event or process has occurred, often in relationship to a clock signal. Counters are used in digital electronics for counting purpose, they can count specific event happening in the circuit. For example, in UP counter a counter increases count for every rising edge of clock. Not only counting, a counter can follow the certain sequence based on our design like any random sequence 0,1,3,2… .They can also be designed with the help of flip flops.

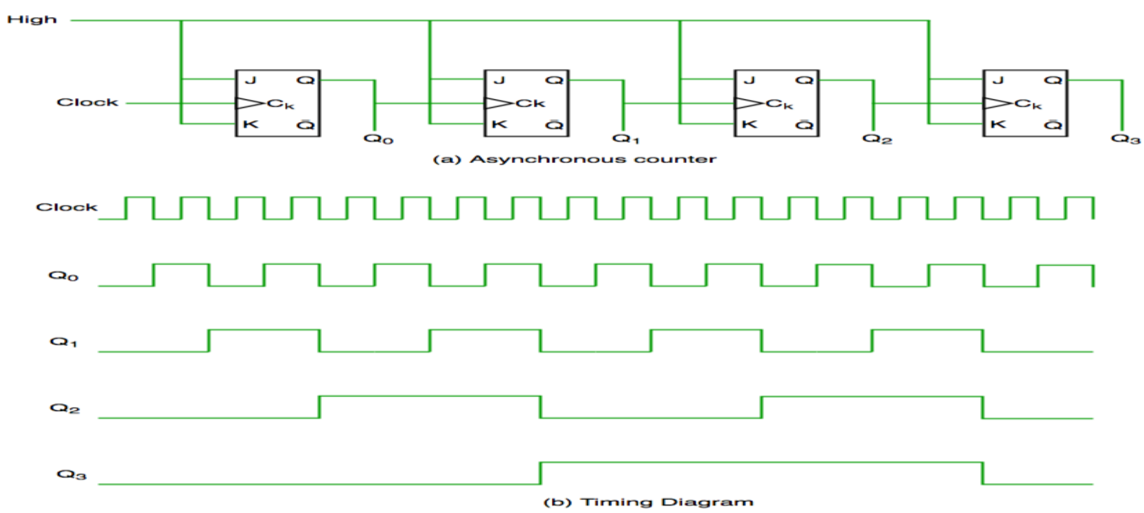
Counter Classification

Counters are broadly divided into two categories

1. Asynchronous counter
2. Synchronous counter

1. **Asynchronous Counter**

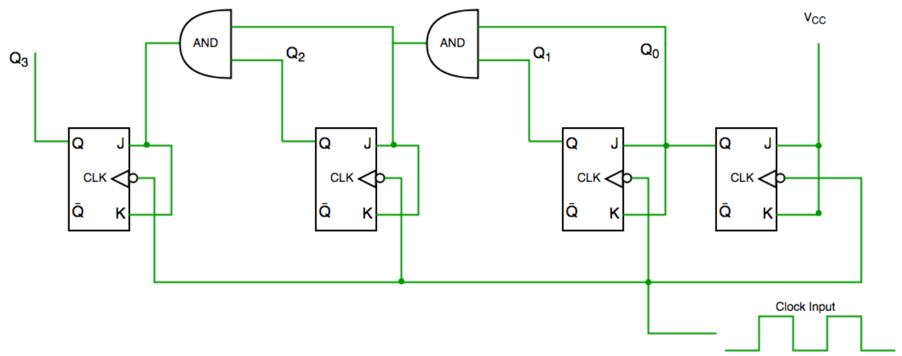
In asynchronous counter we don't use universal clock, only first flip flop is driven by main clock and the clock input of rest of the following flip flop is driven by output of previous flip flops. We can understand it by following diagram-



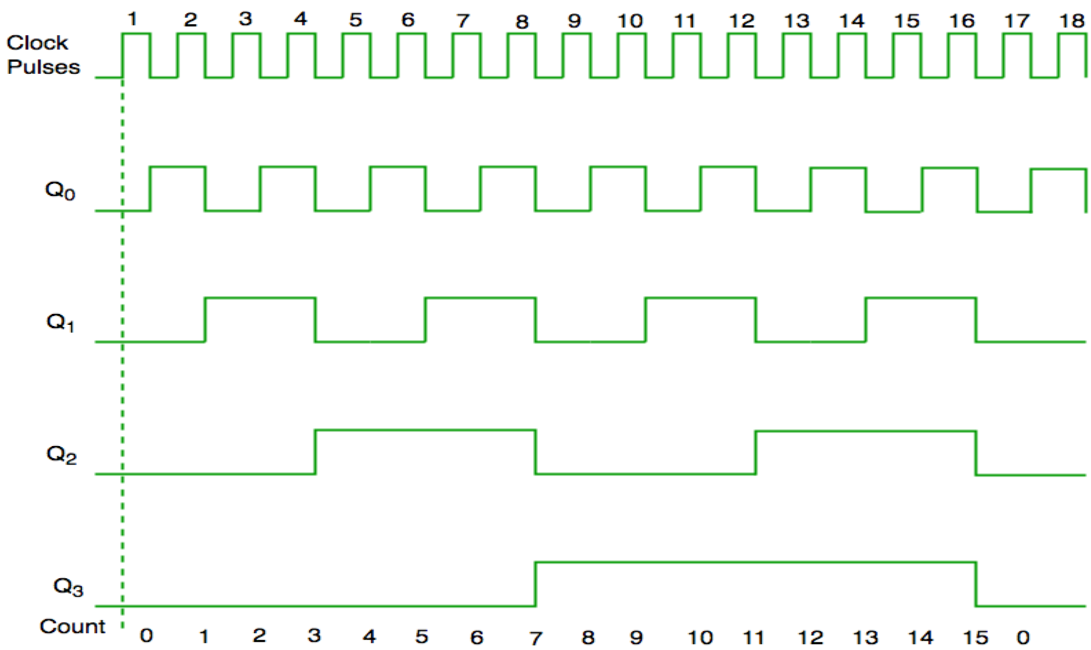
It is evident from timing diagram that Q0 is changing as soon as the rising edge of clock pulse is encountered, Q1 is changing when rising edge of Q0 is encountered(because Q0 is like clock pulse for second flip flop) and so on. In this way ripples are generated through Q0,Q1,Q2,Q3 hence it is also called **RIPPLE counter and serial counter.** A ripple counter is a cascaded arrangement of flip flops where the output of one flip flop drives the clock input of the following flip flop

2. **Synchronous Counter**

Unlike the asynchronous counter, synchronous counter has one global clock which drives each flip flop so output changes in parallel. The one advantage of synchronous counter over asynchronous counter is, it can operate on higher frequency than asynchronous counter as it does not have cumulative delay because of same clock is given to each flip flop. It is also called as parallel counter.



Synchronous counter circuit



Timing diagram synchronous counter

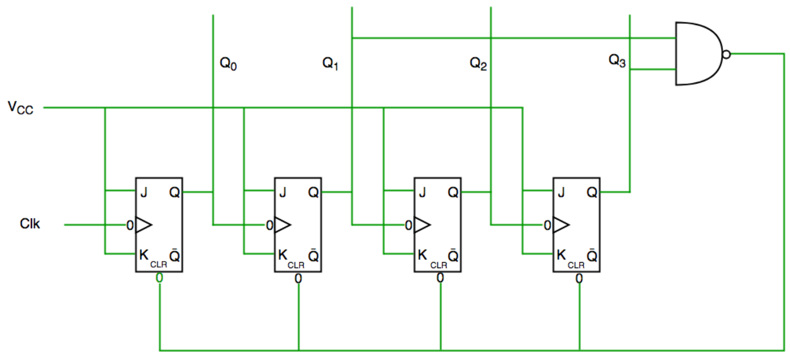
From circuit diagram we see that Q0 bit gives response to each falling edge of clock while Q1 is dependent on Q0, Q2 is dependent on Q1 and Q0 , Q3 is dependent on Q2,Q1 and Q0.

Decade Counter

A decade counter counts ten different states and then reset to its initial states. A simple decade counter will count from 0 to 9 but we can also make the decade counters which can go through any ten states between 0 to 15(for 4 bit counter).

Clock pulse	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	0	0	0	0

Truth table for simple decade counter



Decade counter circuit diagram

We see from circuit diagram that we have used nand gate for Q3 and Q1 and feeding this to clear input line because binary representation of 10 is—

1010

And we see Q3 and Q1 are 1 here, if we give NAND of these two bits to clear input then counter will be clear at 10 and again start from beginning.

<u>Important point</u>: Number of flip flops used in counter are always greater than equal to **($\log_2$ n)** where n=number of states in counter.

Shift Registers: Flip flops can be used to store a single bit of binary data (1 or 0). However, in order to store multiple bits of data, we need multiple flip-flops. N flip flops are to be connected in order to store n bits of data.

A **Register** is a device that is used to store such information. It is a group of flip-flops connected in series used to store multiple bits of data. The information stored within these registers can be transferred with the help of **shift registers**.

Shift Register is a group of flip flops used to store multiple bits of data. The bits stored in such registers can be made to move within the registers and in/out of the registers by applying clock pulses. The registers which will shift the bits to the left are called “Shift left registers”. The registers which will shift the bits to the right are called “Shift right registers”. Shift registers are basically of following types.

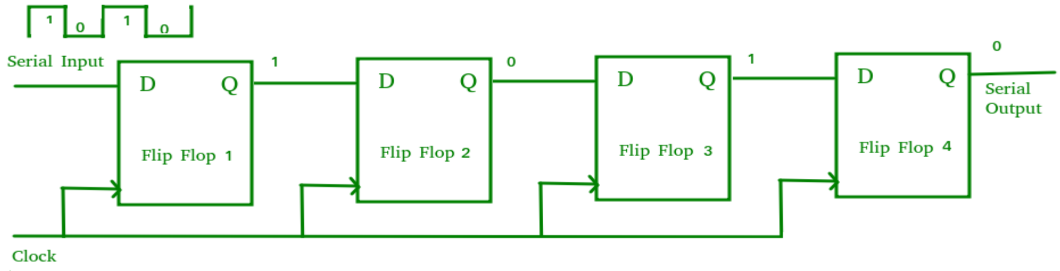
Types of Shift Registers

- Serial In Serial Out shift register
- Serial In parallel Out shift register
- Parallel In Serial Out shift register
- Parallel In parallel Out shift register
- Bidirectional Shift Register
- Universal Shift Register

Serial-In Serial-Out Shift Register (SISO)

The shift register, which allows serial input (one bit after the other through a single data line) and produces a serial output is known as a Serial-In Serial-Out shift register. Since there is only one output, the data leaves the shift register one bit at a time in a serial pattern, thus the name Serial-In Serial-Out Shift Register.

The circuit consists of four D flip-flops which are connected in a serial manner. All these flip-flops are synchronous with each other since the same clock signal is applied to each flip-flop.



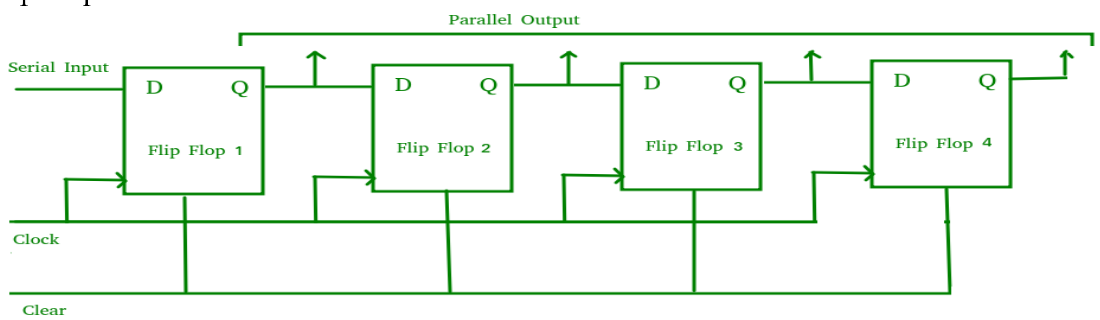
The above circuit is an example of a shift right register, taking the serial data input from the left side of the flip flop.

Serial-In Parallel-Out Shift Register (SIPO)

The shift register, which allows serial input (one bit after the other through a single data line) and produces a parallel output is known as the Serial-In Parallel-Out shift register.

The circuit consists of four D flip-flops which are connected. The clear (CLR) signal is connected in addition to the clock signal to all 4 flip flops in order to RESET them.

The output of the first flip-flop is connected to the input of the next flip flop and so on. All these flip-flops are synchronous with each other since the same clock signal is applied to each flip-flop.

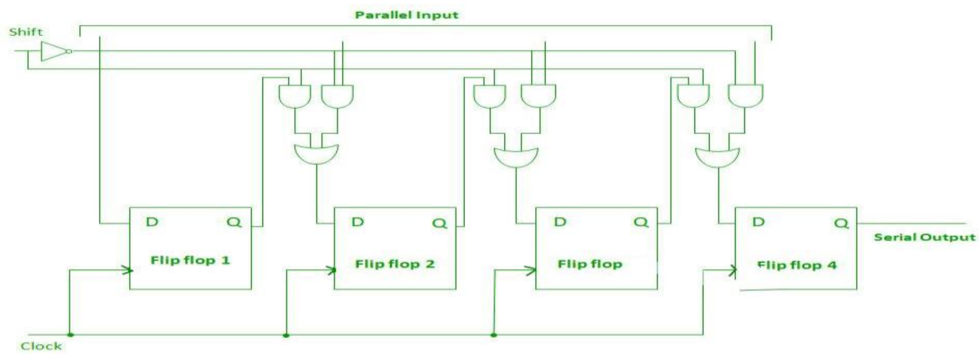


The above circuit is an example of a shift right register, taking the serial data input from the left side of the flip-flop and producing a parallel output.

Parallel-In Serial-Out Shift Register (PISO)

The shift register, which allows parallel input (data is given separately to each flip flop and in a simultaneous manner) and produces a serial output is known as a Parallel-In Serial-Out shift register.

The circuit consists of four D flip-flops which are connected. The clock input is directly connected to all the flip-flops but the input data is connected individually to each flip-flop through a multiplexer at the input of every flip-flop.



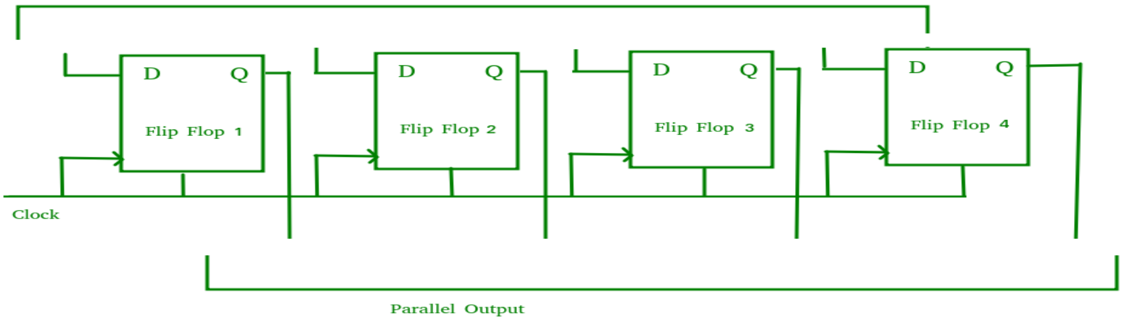
A Parallel in Serial Out (PISO) shift register is used to convert parallel data to serial data.

Parallel-In Parallel-Out Shift Register (PIPO)

The shift register, which allows parallel input (data is given separately to each flip flop and in a simultaneous manner) and also produces a parallel output is known as Parallel-In parallel-Out shift register.

The circuit consists of four D flip-flops which are connected. The clear (CLR) signal and clock signals are connected to all 4 flip-flops.

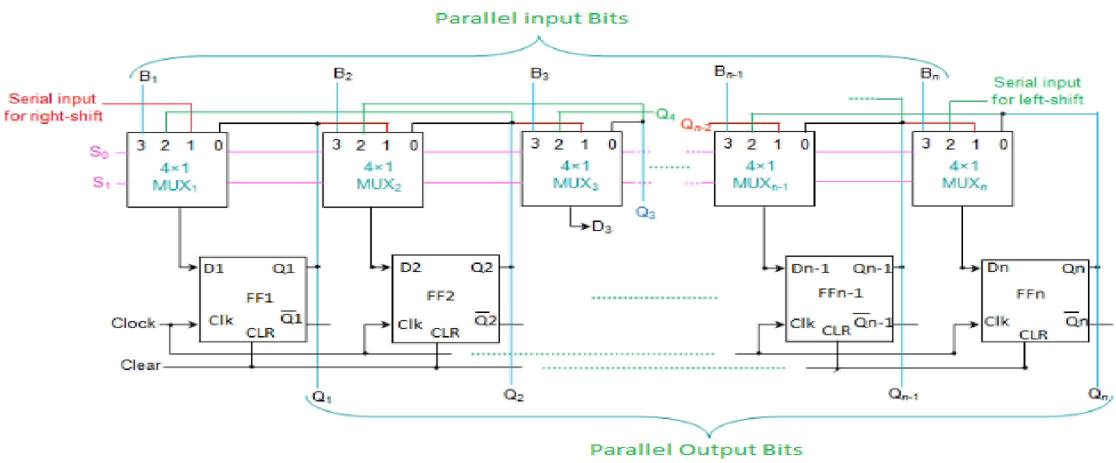
Data is given as input separately for each flip flop and in the same way, output is also collected individually from each flip flop.



A Parallel in Parallel out (PIPO) shift register is used as a temporary storage device

Universal Shift Register

Universal Shift Register is a type of register that contains the both right shift and the left shift. It has also parallel load capabilities. Generally, these types of registers are taken as memory elements in computers. But, the problem with this type of register is that it shifts only in one direction.



N-bit universal shift register consists of flip-flops and multiplexers. Both are N in size. In this, all the n multiplexers share the same select lines and this select input selects the suitable input for flip-flops.

Types of Computers:

A computer is an electronic device that has storage, computations, input (data), output (data) and networking capabilities.

The input and output data can be in different forms like text, images, audio and video. A computer processes the input according to the set of instructions provided to it by the user and gives the desired output. Computers are of various types and they can be categorized in two ways on the basis of size and on the basis of data handling capabilities.

There are two bases on which we can define the types of computers. the type of computers on the basis of size and data handling capabilities.

- Super Computer
- Mainframe computer
- Mini Computer
- Workstation Computer
- Personal Computer (PC)
- Server Computer
- Analog Computer
- Digital Computer
- Hybrid Computer
- Tablets and Smartphone

Supercomputer:

- When we talk about speed, then the first name that comes to mind when thinking of computers is supercomputers. They are the biggest and fastest computers (in terms of speed of processing data). Supercomputers are designed such that they can process a huge amount of data, like processing trillions of instructions or data just in a second.
- It is basically used in scientific and engineering applications such as weather forecasting, scientific simulations, and nuclear energy research. It was first developed by Roger Cray in 1976.
- It is used in scientific research areas for analyzing data obtained from exploring the solar system, satellites, etc.

Mainframe computer:

- Mainframe computers are designed in such a way that they can support hundreds or thousands of users at the same time.
- It also supports multiple programs simultaneously. So, they can execute different processes simultaneously.
- All these features make the mainframe computer ideal for big organizations like banking, telecom sectors, etc., which process a high volume of data in general.

Minicomputer:

- Minicomputer is a medium size multiprocessing computer. In this type of computer, there are two or more processors, and it supports 4 to 200 users at one time.
- Minicomputer is similar to Microcontroller. Minicomputers are used in places like institutes or departments for different work like billing, accounting, inventory management, etc.
- It is smaller than a mainframe computer but larger in comparison to the microcomputer. Because of its low weight, it is easy to carry anywhere.

Workstation Computer:

- A workstation computer is designed for technical or scientific applications. It consists of a fast microprocessor, with a large amount of RAM and a high-speed graphic adapter.
- It is a single-user computer. It is generally used to perform a specific task with great accuracy. They are exclusively made for complex work purposes.
- It provides large storage capacity, better graphics, and a more powerful CPU when compared to a PC. It is also used to handle animation, data analysis, CAD, audio and video creation, and editing.

Personal Computer (PC):

- Personal Computers is also known as a microcomputer. It is basically a general-purpose computer designed for individual use.
- It consists of a microprocessor as a central processing unit(CPU), memory, input unit, and output unit. In this limited number of software can be used.
- This kind of computer is suitable for personal work such as making an assignment, watching a movie, or at the office for office work, etc. For example, Laptops and desktop computers.

Server Computer

- Server Computers are computers that are combined data and programs. Electronic data and applications are stored and shared in the server computer.
- Examples of server computer are like Wikipedia, as when users put a request for any page, it finds what the user is looking for and sends it to the user.

Analog Computer

- Analog Computers are particularly designed to process analog data. Continuous data that changes continuously and cannot have discrete values are called analog data.
- So, an analog computer is used where we don't need exact values or need approximate values such as speed, temperature, pressure, etc.

Digital Computer

- Digital computers are designed in such a way that they can easily perform calculations and logical operations at high speed. It takes raw data as input and processes it with programs stored in its memory to produce the final output.
- All modern computers, like laptops, desktops including smartphones are digital computers.

Hybrid Computer

- As the name suggests hybrid, which means made by combining two different things. Similarly, the hybrid computer is a combination of both analog and digital computers. Hybrid computers are fast like analog computers and have memory and accuracy like digital computers.
- A processor which is used in petrol pumps that converts the measurements of fuel flow into quantity and price is an example of a hybrid computer.

Tablet and Smartphones

- Tablets and Smartphones are the types of computers that are pocket friendly and easy to carry is these are handy. This is one of the best use of modern technology.
- These devices have better hardware capabilities, extensive operating systems, and better multimedia functionality. smartphones and tablets contain a number of sensors and are also able to provide wireless communication protocols.

Generations of Computer:

Computers have evolved significantly over the years, and the history of computers is often divided into generations based on the technology used. Here are the five generations of computers:

1. **First Generation (1940s-1950s)**: The first computers used vacuum tubes for processing and magnetic drums for storage. They were large, expensive, and unreliable.
   Introduction:
   1946-1959 is the period of first generation computer.
   J.P.Eckert and J.W.Mauchy invented the first successful electronic computer called ENIAC, ENIAC stands for "Electronic Numeric Integrated And Calculator".
   Few Examples are:
   ENIAC
   EDVAC
   UNIVAC
   IBM-701
   IBM-650
2. **Second Generation (1950s-1960s):** The second generation of computers replaced vacuum tubes with transistors, making them smaller, faster, and more reliable. Magnetic core memory was also introduced, which was faster and more reliable than magnetic drums.
   Introduction:
   1959-1965 is the period of second-generation computer.
   3.Second generation computers were based on Transistor instead of vacuum tubes.
   Few Examples are:
   Honeywell 400
   IBM 7094
   CDC 1604
   CDC 3600
   UNIVAC 1108
3. **Third Generation (1960s-1970s):** The third generation of computers used integrated circuits, which allowed for even smaller and faster computers. They also introduced magnetic disk storage and operating systems.
   Introduction:
   1965-1971 is the period of third generation computer.
   These computers were based on Integrated circuits.
   IC was invented by Robert Noyce and Jack Kilby In 1958-1959.
   IC was a single component containing number of transistors.
   Few Examples are:
   PDP-8
   PDP-11
   ICL 2900
   IBM 360
   IBM 370
4. **Fourth Generation (1970s-1980s):** The fourth generation of computers saw the introduction of microprocessors, which made personal computers possible. They also introduced graphical user interfaces and networking.
   Introduction:
   1971-1980 is the period of fourth generation computer.
   This technology is based on Microprocessor.

A microprocessor is used in a computer for any logical and arithmetic function to be performed in any program.
Graphics User Interface (GUI) technology was exploited to offer more comfort to users.
Few Examples are:
IBM 4341
DEC 10
STAR 1000
PUP 11

5. **Fifth Generation (1980s-Present):** The fifth generation of computers is still ongoing, and is focused on artificial intelligence and parallel processing. This generation also saw the development of mobile computing and the internet.
Introduction:
The period of the fifth generation in 1980-onwards.
This generation is based on artificial intelligence.
The aim of the fifth generation is to make a device which could respond to natural language input and are capable of learning and self-organization.
This generation is based on ULSI(Ultra Large Scale Integration) technology resulting in the production of microprocessor chips having ten million electronic component.
Few Examples are:
Desktop
Laptop
NoteBook
UltraBook
Chromebook

Functional unit: - A computer consists of five functionally independent main parts input, memory, arithmetic logic unit (ALU), output and control unit.

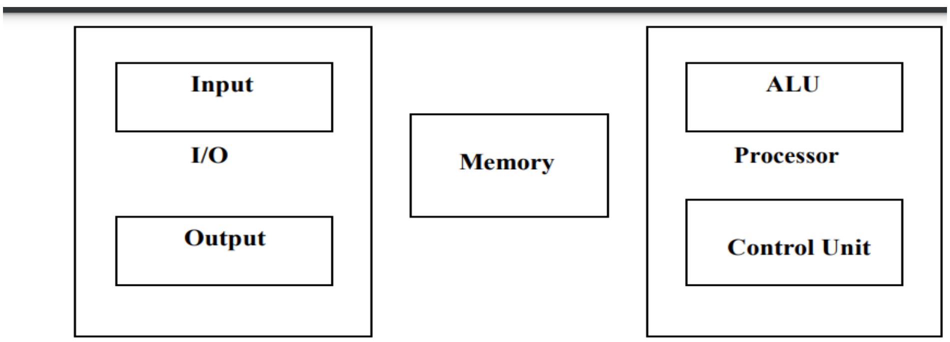


Fig a : Functional units of computer

Input device accepts the coded information as source program i.e. high level language. This is either stored in the memory or immediately used by the processor to perform the desired operations. The program stored in the memory determines the processing steps. Basically the computer converts one source program to an object program. i.e. into machine language.

Finally the results are sent to the outside world through output device. All of these actions are coordinated by the control unit.
Input unit: - The source program/high level language program/coded information/simply data is fed to a computer through input devices keyboard is a most common type.
Whenever a key is pressed, one corresponding word or number is translated into its equivalent binary code over a cable & fed either to memory or processor.

Examples: Joysticks, trackballs, mouse, scanners etc are other input devices.

Memory unit: - Its function into store programs and data.

It is basically to two types

1. Primary memory
2. Secondary memory

1. Primary memory: - Is the one exclusively associated with the processor and operates at the electronics speeds programs must be stored in this memory while they are being executed. The memory contains a large number of semiconductors storage cells. Each capable of storing one bit of information. These are processed in a group of fixed site called word.

To provide easy access to a word in memory, a distinct address is associated with each word location. Addresses are numbers that identify memory location.

Number of bits in each word is called word length of the computer. Programs must reside in the memory during execution. Instructions and data can be written into the memory or read out under the control of processor.

Memory in which any location can be reached in a short and fixed amount of time after specifying its address is called random-access memory (RAM).

The time required to access one word in called memory access time. Memory which is only readable by the user and contents of which can't be altered is called read only memory (ROM) it contains operating system.

Caches are the small fast RAM units, which are coupled with the processor and are aften contained on the same IC chip to achieve high performance. Although primary storage is essential it tends to be expensive.

2. Secondary memory: - Is used where large amounts of data & programs have to be stored, particularly information that is accessed infrequently.
Examples: - Magnetic disks & tapes, optical disks (ie CD-ROM's), floppies etc.,

Arithmetic logic unit (ALU):- Most of the computer operators are executed in ALU of the processor like addition, subtraction, division, multiplication, etc. the operands are brought into the ALU from memory and stored in high speed storage elements called register. Then according to the instructions the operation is performed in the required sequence.

The control and the ALU are may times faster than other devices connected to a computer system. This enables a single processor to control a number of external devices such as key boards, displays, magnetic and optical disks, sensors and other mechanical controllers.

Output unit:- These actually are the counterparts of input unit. Its basic function is to send the processed results to the outside world.
Examples:- Printer, speakers, monitor etc.

Control unit:- It effectively is the nerve center that sends signals to other units and senses their states. The actual timing signals that govern the transfer of data between input unit, processor, memory and output unit are generated by the control unit.

Basic operational concepts: -
To perform a given task an appropriate program consisting of a list of instructions is stored in the memory. Individual instructions are brought from the memory into the processor, which executes the specified operations. Data to be stored are also stored in the memory.
Examples: - Add LOCA, R0

This instruction adds the operand at memory location LOCA, to operand in register R0 & places the sum into register. This instruction requires the performance of several steps,
1. First the instruction is fetched from the memory into the processor.
2. The operand at LOCA is fetched and added to the contents of R0
3. Finally the resulting sum is stored in the register R0

The preceding add instruction combines a memory access operation with an ALU Operations. In some other type of computers, these two types of operations are performed by separate instructions for performance reasons.

Load LOCA, R1
Add R1, R0
Transfers between the memory and the processor are started by sending the address of the memory location to be accessed to the memory unit and issuing the appropriate control signals. The data are then transferred to or from the memory.

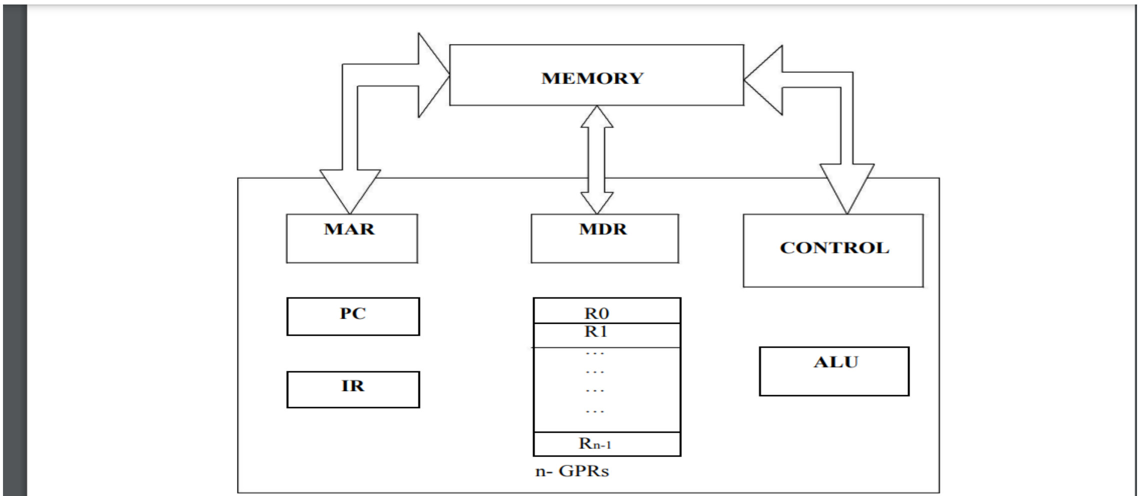


Fig b : Connections between the processor and the memory

The fig shows how memory & the processor can be connected. In addition to the ALU & the control circuitry, the processor contains a number of registers used for several different purposes.

The instruction register (IR):- Holds the instructions that is currently being executed. Its output is available for the control circuits which generates the timing signals that control the various processing elements in one execution of instruction.

The program counter PC:- This is another specialized register that keeps track of execution of a program. It contains the memory address of the next instruction to be fetched and executed.

Besides IR and PC, there are n-general purpose registers R0 through Rn1. The other two registers which facilitate communication with memory are: -

1. MAR – (Memory Address Register):- It holds the address of the location to be accessed.
2. MDR – (Memory Data Register):- It contains the data to be written into or read out of the address location.

Operating steps are

1. Programs reside in the memory & usually get these through the I/P unit.
2. Execution of the program starts when the PC is set to point at the first instruction of the program.
3. Contents of PC are transferred to MAR and a Read Control Signal is sent to the memory.
4. After the time required to access the memory elapses, the address word is read out of the memory and loaded into the MDR.
5. Now contents of MDR are transferred to the IR & now the instruction is ready to be decoded and executed.
6. If the instruction involves an operation by the ALU, it is necessary to obtain the required operands.
7. An operand in the memory is fetched by sending its address to MAR & Initiating a read cycle.
8. When the operand has been read from the memory to the MDR, it is transferred from MDR to the ALU.
9. After one or two such repeated cycles, the ALU can perform the desired operation.
10. If the result of this operation is to be stored in the memory, the result is sent to MDR.
11. Address of location where the result is stored is sent to MAR & a write cycle is initiated.
12. The contents of PC are incremented so that PC points to the next instruction that is to be executed.

Normal execution of a program may be preempted (temporarily interrupted) if some devices require urgent servicing, to do this one device raises an Interrupt signal.

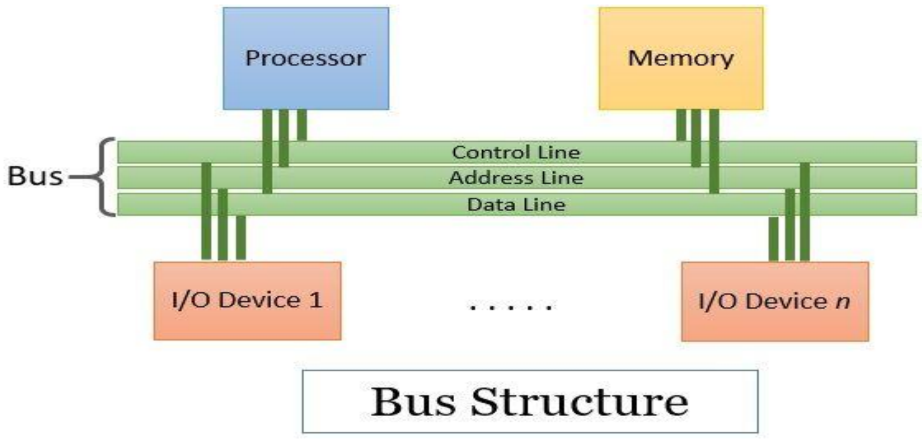
An interrupt is a request signal from an I/O device for service by the processor. The processor provides the requested service by executing an appropriate interrupt service routine.

The Diversion may change the internal stage of the processor its state must be saved in the memory location before interruption. When the interrupt-routine service is completed the state of the processor is restored so that the interrupted program may continue.

Bus Structures: Bus structures in computer plays important role in connecting the internal components of the computer. The bus in the computer is the *shared transmission medium*. This means multiple components or devices use the same bus structure to transmit the information signals to each other.

At a time, only one pair of devices can use this bus to communicate with each other successfully. If multiple devices transmit the information signal over the bus at the same time, the signals overlap each other and get jumbled.

A system bus has typically from fifty to hundreds of distinct lines where each line is meant for a certain function. These lines can be categorised into three functional groups i.e., data lines, address lines, and control lines.



1. Data Lines

Data lines coordinate in transferring the data among the system components. The data lines are collectively called data bus. A data bus may have 32 lines, 64 lines, 128 lines, or even more lines. The number of lines present in the data bus defines the *width* of the data bus.

Each data line is able to transfer only one bit at a time. So the number of data lines in a data bus determines how many bits it can transfer at a time.

2. Address Lines

The content of the address lines of the bus determines the source or destination of the data present on the data bus. The number of address lines together is referred to as the address bus. The number of address lines in the address bus determines its *width*.

The width of the address bus determines the memory capacity of the system. The content of address lines is also used for addressing I/O ports. The higher-order bits determine the bus module, and the lower-ordered bits determine the address of memory locations or I/O ports.

Whenever the processor has to read a word from memory, it simply places the address of the corresponding word on the address line.

3. Control Lines

The address lines and data lines are shared by all the components of the system, so there must be some means to control the use and access of data and address lines. The control signals placed on the control lines control the use and access to the address and data lines of the bus.

The control lines include the lines for:

- **Memory Write:** This command causes the data on the data bus to be placed over the addressed memory location.
- **Memory Read:** This command causes the data on the addressed memory location to be placed on the data bus.
- **I/O Write:** The command over this control line causes the data on the data bus to be placed over the addressed I/O port.
- **I/O Read:** The command over this control line causes the data from the addressed I/O port to be placed over the data bus.
- **Transfer ACK:** This control line indicates the data has been received from the data bus or is placed over the data bus.

- **Bus Request:** This control line indicates that the component has requested control over the bus.
- **Bus Grant:** This control line indicates that the bus has been granted to the requesting component.
- **Interrupt Request:** This control line indicates that interrupts are pending.
- **Interrupt ACK:** This control line acknowledges when the pending interrupt is serviced.
- **Clock:** This control line is used to synchronize the operations.
- **Reset:** The bit information over this control line initializes all the modules.

Suppose a component connected to the bus wishes to send data to another connected component. In that case, it first has to acquire control over the bus, and then it can transfer the data to another component over the bus. The same happens when a component request data from another component.

During data transfer between two components, one component act as a master and the other act as a slave. The device initiating the data transfer is referred to as the master, and usually, it is a processor, or sometimes it may be some other device or component. The component addressed by the master component is referred to as a slave.

Software: Software, which is abbreviated as SW or S/W, is a set of programs that enables the hardware to perform a specific task. All the programs that run the computer are software. The software can be of three types: **system software, application software, and programming software.**

1) System Software

The system software is the main software that runs the computer. When you turn on the computer, it activates the hardware and controls and coordinates their functioning. The application programs are also controlled by system software. An operating system is an example of system software.

i) Operating System:

An operating system is the system software that works as an interface to enable the user to communicate with the computer. It manages and coordinates the functioning of hardware and software of the computer. The commonly used operating systems are Microsoft Windows, Linux, and Apple Mac OS X.

Some other examples of system software include:

- **BIOS:** It stands for basic input output system. It is a type of system software, which is stored in Read Only Memory (ROM) located on the motherboard. However, in advanced computer systems, it is stored in flash memory. BIOS is the first software that gets activated when you turn on your computer system. It loads the drivers of the hard disk into memory as well as assists the operating system to load itself into the memory.
- **Boot Program:** Boot refers to starting up a computer. When you switch on the computer, the commands in the ROM are executed automatically to load the boot program into memory and execute its instructions. The BIOS program has a basic set

of commands that enables the computer to perform the basic input/output instructions to start the computer.

- **An assembler:** It plays the role of a converter as it receives basic computer instructions and converts them into a pattern of bits. The processor uses these bits to perform basic operations.
- **A device driver:** This system software controls hardware devices connected to a computer. It enables the computer to use the hardware by providing an appropriate interface. The kernel of a Computer's CPU communicates with different hardware through this software. Operating systems generally come with most of the device drivers. If the operating system does not have a device driver for hardware, you have to install the device driver before using that hardware device.

2) Application Software:

Application software is a set of programs designed to perform a specific task. It does not control the working of a computer as it is designed for end-users. A computer can run without application software. Application software can be easily installed or uninstalled as required. It can be a single program or a collection of small programs. Microsoft Office Suite, Adobe Photoshop, and any other software like payroll software or income tax software are application software. As we know, they are designed to perform specific tasks. Accordingly, they can be of different types such as:

- **Word Processing Software:** This software allows users to create, edit, format, and manipulate the text and more. It offers lots of options for writing documents, creating images, and more. For example, MS Word, WordPad, Notepad, etc.
- **Spreadsheet Software:** It is designed to perform calculations, store data, create charts, etc. It has rows and columns, and the data is entered in the cell, which is an intersection of a row and column, e.g., Microsoft Excel.
- **Multimedia Software:** These software are developed to perform editing of video, audio, and text. It allows you to combine texts, videos, audio, and images. Thus, you can improve a text document by adding photos, animations, graphics, and charts through multimedia software. For example, VLC player, Window Media Player, etc.
- **Enterprise Software:** These software are developed for business operational functions. It is used in large organizations where the quantum of business is too large. It can be used for accounting, billing, order processing and more. For example, CRM (Customer Relationship Management), BI (Business Intelligence), ERP (Enterprise Resource Planning), SCM (Supply Chain Management), customer support system, and more.

3) Programming Software:

It is a set or collection of tools that help developers in writing other software or programs. It assists them in creating, debugging, and maintaining software or programs or applications. We can say that these are facilitator software that helps translate programming language such as Java, C++, Python, etc., into machine language code. So, it is not used by end-users. For example, compilers, linkers, debuggers, interpreters, text editors, etc. This software is also called a programming tool or software development tool.

Some examples of programming software include:

- **Eclipse:** It is a java language editor.
- **Coda:** It is a programming language editor for Mac.
- **Notepad++:** It is an open-source editor for windows.
- **Sublime text:** It is a cross-platform code editor for Linux, Mac, and Windows.

Performance: The performance of a computer system depends on multiple factors such as CPU, memory, and input/output systems.

Here are several factors that can impact the performance of a computer system, including:

- **Processor speed:** The speed of the processor, measured in GHz (gigahertz), determines how quickly the computer can execute instructions and process data.
- **Memory:** The amount and speed of the memory, including RAM (random access memory) and cache memory, can impact how quickly data can be accessed and processed by the computer.
- **Storage:** The speed and capacity of the storage devices, including hard drives and solid-state drives (SSDs), can impact the speed at which data can be stored and retrieved.
- **I/O devices:** The speed and efficiency of input/output devices, such as keyboards , mice, and displays, can impact the overall performance of the system.
- **Software optimization:** The efficiency of the software running on the system, including operating systems and applications, can impact how quickly tasks can be completed.

Improving the performance of a computer system typically involves optimizing one or more of these factors to reduce the time and resources required to complete tasks. This can involve upgrading hardware components, optimizing software, and using specialized performance-tuning tools to identify and address bottlenecks in the system.

It basically depends on the response time, throughput, and execution time of a computer system. **Response time** is the time from the start to completion of a task. This also includes:

- Operating system overhead.
- Waiting for I/O and other processes
- Accessing disk and memory
- Time spent executing on the CPU or execution time.

Throughput is the total amount of work done in a given time.

CPU execution time is the total time a CPU spends computing on a given task.

Performance is determined by execution time as performance is inversely proportional to execution time.

Performance = (1 / Execution time)

And,

(Performance of A / Performance of B)
= (Execution Time of B / Execution Time of A)

If given that Processor A is faster than processor B, that means execution time of A is less than that of execution time of B. Therefore, performance of A is greater than that of performance of B. **Example –** Machine A runs a program in 100 seconds, Machine B runs the same program in 125 seconds

(Performance of A / Performance of B)
= (Execution Time of B / Execution Time of A)
= 125 / 100 = 1.25

That means machine A is 1.25 times faster than Machine B. And, the time to execute a given program can be computed as:

Execution time = CPU clock cycles x clock cycle time

Since clock cycle time and clock rate are reciprocals, so,

Execution time = CPU clock cycles / clock rate

The number of CPU clock cycles can be determined by,

CPU clock cycles
= (No. of instructions / Program) x (Clock cycles / Instruction)
= Instruction Count x CPI

Which gives,

Execution time
= Instruction Count x CPI x clock cycle time
= Instruction Count x CPI / clock rate

Units for CPU Execution Time

$$\text{CPU time} = \frac{\text{Seconds}}{\text{Program}} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Cycle}}$$

How to Improve Performance?

To improve performance you can either:

- Decrease the CPI (clock cycles per instruction) by using new Hardware.
- Decrease the clock time or Increase clock rate by reducing propagation delays or by use pipelining.
- Decrease the number of required cycles or improve ISA or Compiler.

Execution Upgrade Procedures

The Exhibition upgrade on computer chip execution time is worked with by the accompanying variables in a significant manner.

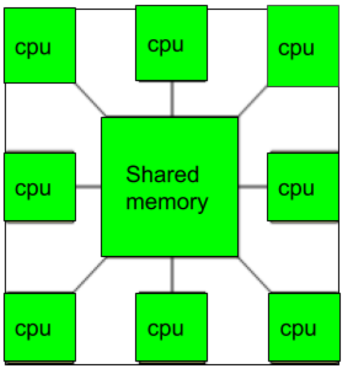
1.Internal Engineering of the computer chip
2.Instruction Arrangement of the central processor
3.Memory Speed and transmission capacity
4.Percentage utilization of the registers in execution (note: Registers are something like multiple times quicker than memory).

Multiprocessor and Multicomputer:

Multiprocessor: A Multiprocessor is a computer system with two or more central processing units (CPUs) share full access to a common RAM. The main objective of using a multiprocessor is to boost the system's execution speed, with other objectives being fault tolerance and application matching.

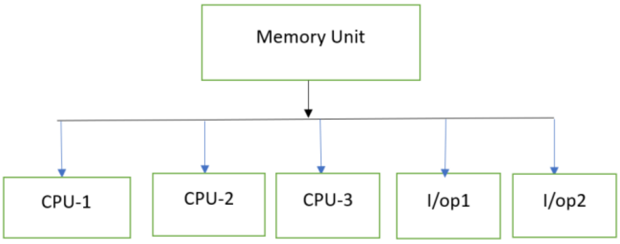
There are two types of multiprocessors, one is called **shared memory multiprocessor** and another is **distributed memory multiprocessor**. In shared memory multiprocessors, all the CPUs shares the common memory but in a distributed memory multiprocessor, every CPU has its own private memory.

The interconnection among two or more processor and shared memory is done with three methods



1)Time shared common bus
2)Multiport memories
3)Crossbar switch network

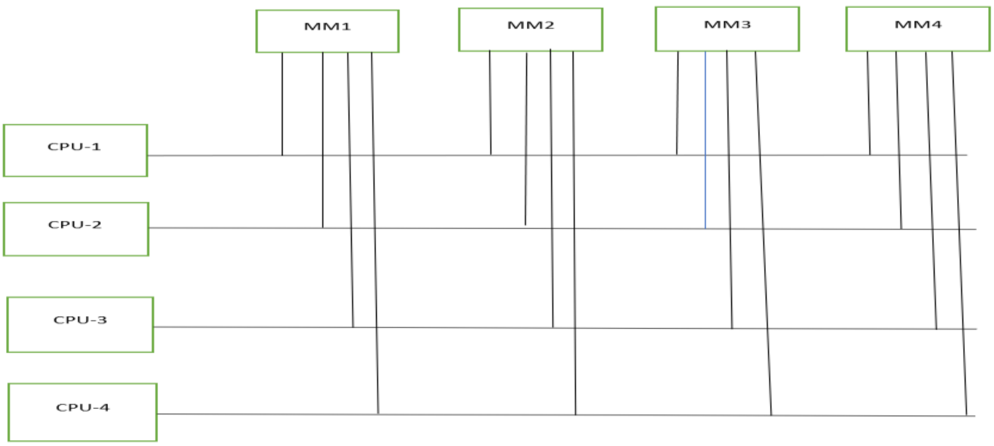
1)Time shared common bus



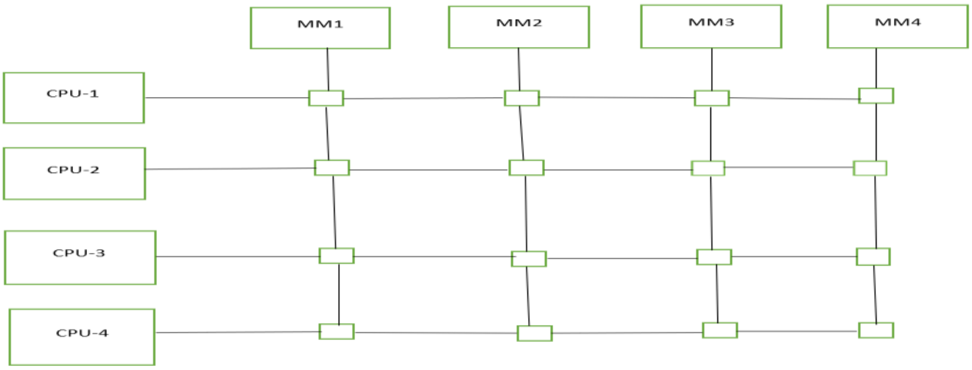
In this method is contains a single shared bus through which all processor & memory unit can be communicated.
Consider CPU-1 is interacting with memory unit using common shared bus in that case all other processor must be idle as we have only one bus to communicate.
Disadvantage:

- **Data transfer rate is slow.**

2)Multiport memories
Unlike in the shared common bus method, hence it contains separate bus for each processor to communicate with the memory module.
Suppose CPU-1 wants to interact with memory module 1 then port mm1 is enabled. Similarly CPU-4 wants to
to interact with memory module 4 then port mm4 is enabled. Hence all the process can be communicated parallelly. If more than one CPU request for same time memory module, priority will be given in the order of CPU-1,CPU-2,CPU-3,CPU-4.



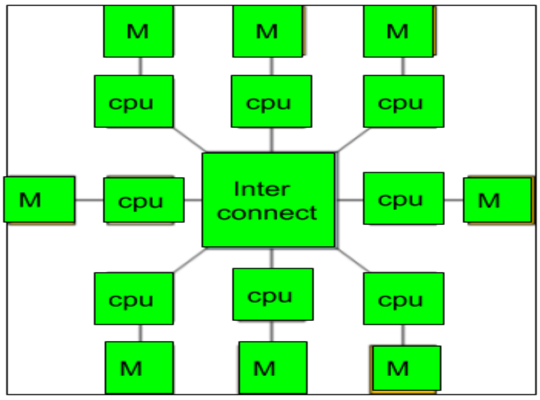
3)Crossbar switch network: Here instead multiport unlike in multiport memories, a switch will be installed between memory unit and CPU. Switch is responsible for whether to pass the request to a particular memory module or not based on the request made for.



Benefits of using a Multiprocessor –

- Enhanced performance.
- Multiple applications.
- Multi-tasking inside an application.
- High throughput and responsiveness.
- Hardware sharing among CPUs.

2. Multicomputer: A multicomputer system is a computer system with multiple processors that are connected together to solve a problem. Each processor has its own memory and it is accessible by that particular processor and those processors can communicate with each other via an interconnection network.



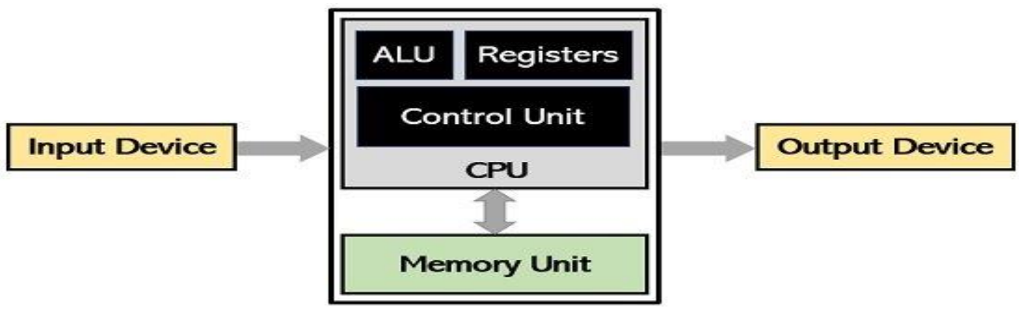
As the multicomputer is capable of messages passing between the processors, it is possible to divide the task between the processors to complete the task. Hence, a multicomputer can be used for distributed computing. It is cost effective and easier to build a multicomputer than a multiprocessor.

Difference between multiprocessor and Multicomputer:

Features	Multiprocessor System	Multicomputer System
Definition	It is a system with multiple processors that enables programs to be processed at the same time.	It is a collection of processors linked by a communication network that collaborate to solve a computation task.
Programming	It is easy to program.	It is complex to program.
Computing	It supports parallel computing.	It supports distributed computing.
Construction	It is easy and less expensive to develop.	It is complex and expensive to develop.
Type of network	It is a type of dynamic network.	It is a type of static network.
Communication between processing elements	It requires proper communication between the processing components and memory for successful resource allocation.	There is no interaction between processor units or memory resources.
Another name	It is also known as the tightly coupled system.	It is also known as loosely coupled systems.
Example	Sequent symmetry S-81 is an example of a multiprocessor system.	Message-passing multicomputer is an example of the multicomputer system.
Execution	It may execute the programs very quickly.	It may run slowly.

Von Neumann Architecture: A computer architecture that uses a single memory unit within which both data and instructions get stored is known as Von Neumann architecture.

The figure below represents the architectural representation of the Von Neumann Model:



Von Neumann Architecture

There are three major components that constitute this architecture:

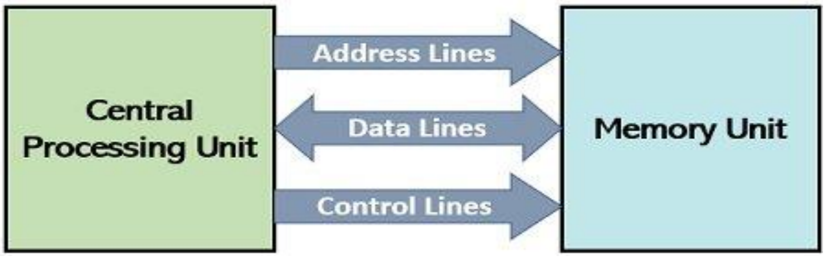
- CPU
- Memory
- I/O interface

The central processing unit is composed of the control unit, arithmetic and logic unit, and registers. The control unit is responsible for generating control signals based on the order of executing the instructions.

The ALU is responsible for executing the mathematical operations. While the registers are responsible for temporarily storing data and instructions.

The memory unit also serves as a crucial part of the overall system as it stores both data as well as codes. A computer's memory is mainly classified as RAM and ROM. RAM is the temporary storage unit that stores data along with general-purpose instructions. While ROM is a permanent type of memory that holds the instructions responsible for booting up the system.

Like we have said that in this architecture, data and instructions both reside in a single memory unit hence a single set of buses is used by the CPU to access the memory.



After the execution of the program, data gets stored in memory from where it is provided to output devices.

Attribute	**Harvard Architecture**	**Von Neumann Architecture**
Memory	Separate memory for instructions and data	Single memory for instructions and data
Instruction Execution	Instructions and data are fetched from separate memories simultaneously	Instructions and data are fetched from the same memory sequentially
Instruction Set	Can have different instruction sets for instructions and data	Uses a single instruction set for both instructions and data
Performance	Can potentially have faster execution due to simultaneous fetching	May have slower execution due to sequential fetching
Complexity	Can be more complex to implement due to separate memory systems	Relatively simpler to implement with a single memory system
Flexibility	Allows for more flexibility in terms of instruction and data handling	Less flexible in terms of instruction and data handling