Section C

Digital Electronics – Day3

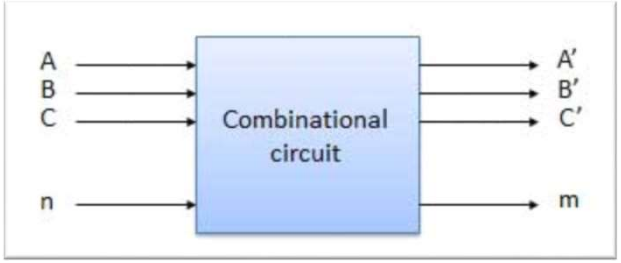
Trainer : Sohail Inamdar

Combinational Circuits & Sequential Circuit

Digital Electronics

Combinational Circuits

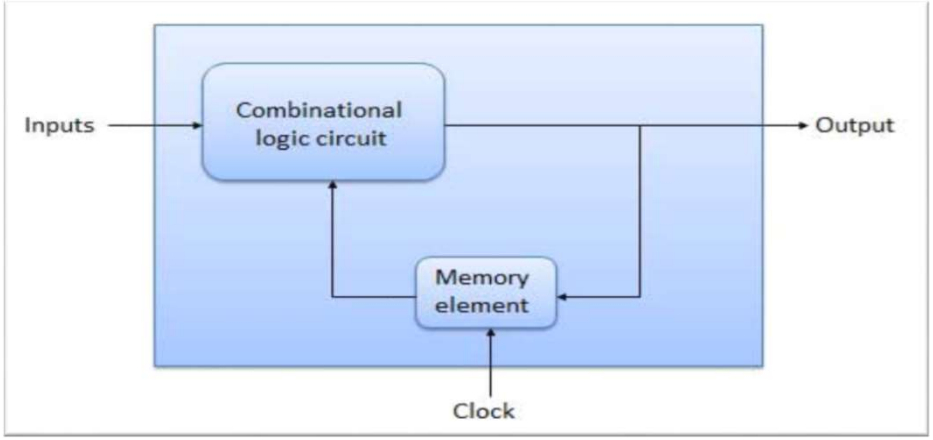
- O/p is only depending on the present I/p.



- No feedback.
- No memory.

Sequential Circuit

- O/p is depending on the present I/p as well as past outputs.



Combinational Circuits & Sequential Circuit

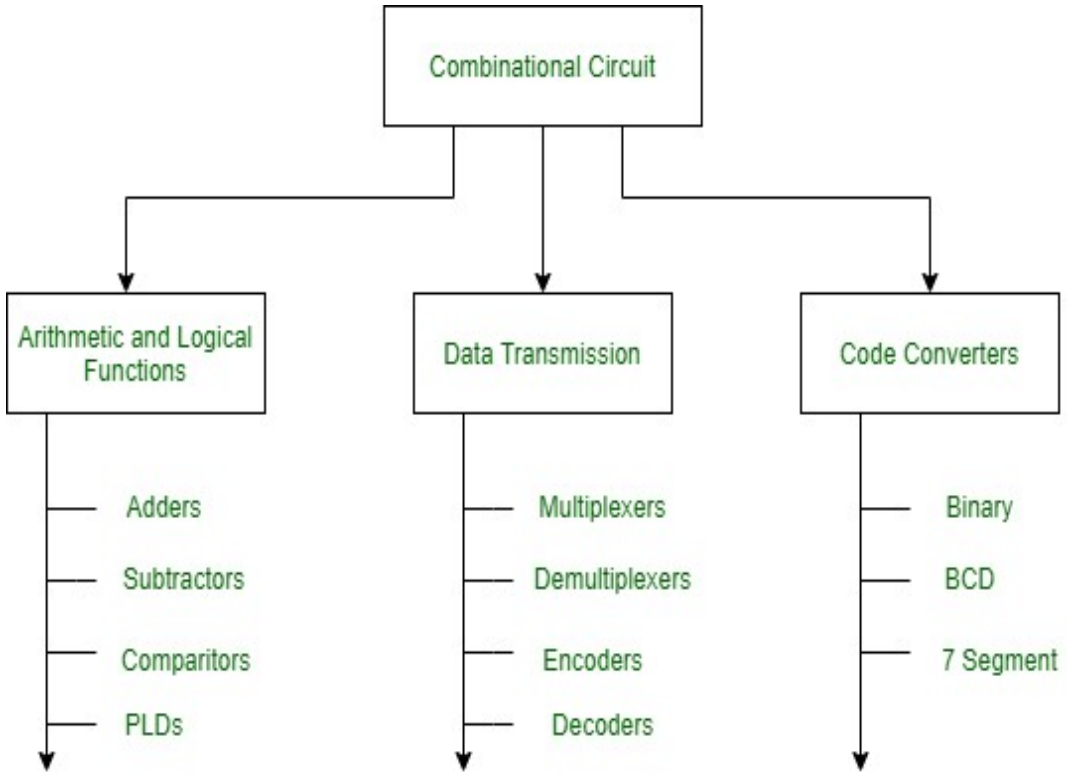
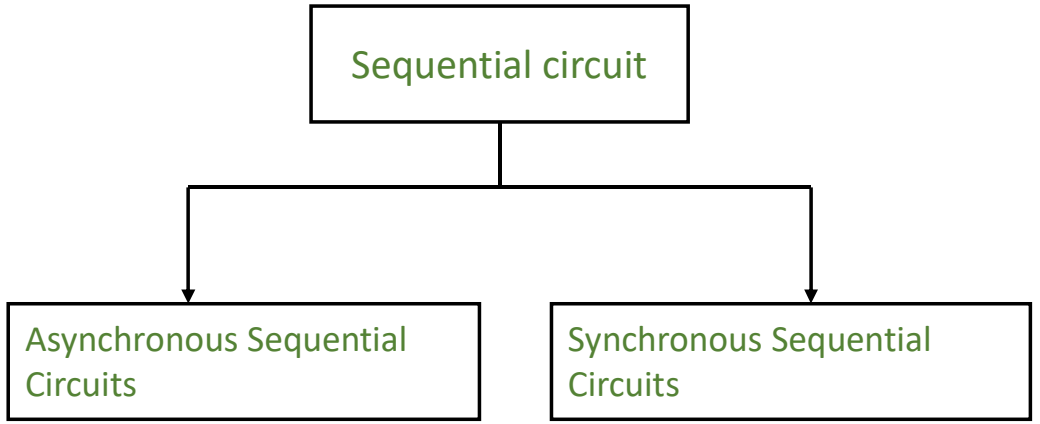
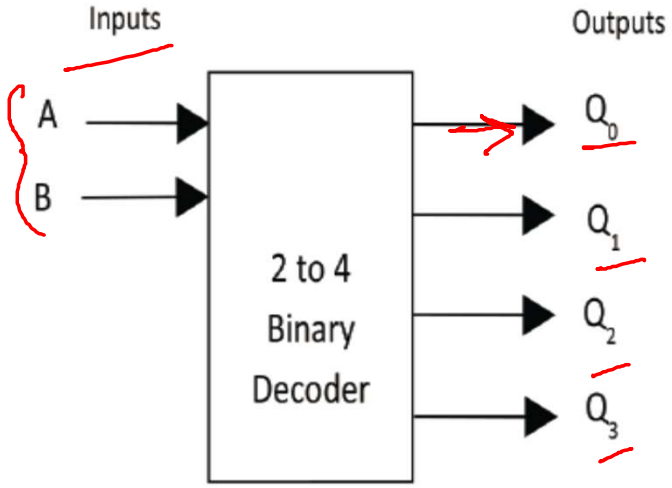


Figure - Classifications of Combinations Circuits

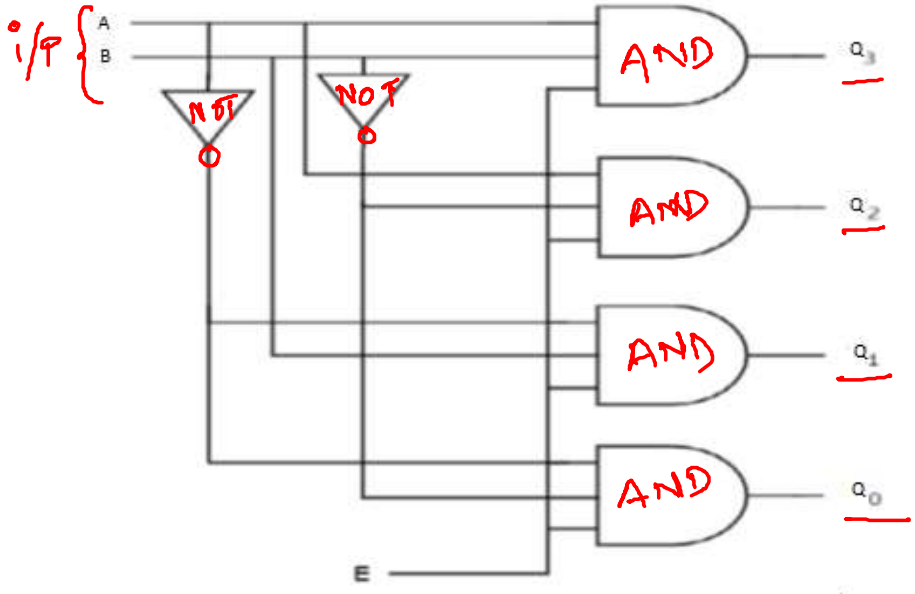


Decoders (n : 2^n)

- Decoder is multiple-input multiple-output combinational circuit
- Which converts coded input into coded output
- Where input and output codes are different
- Decoders has n inputs and 2^n outputs
- 2 to 4 line decoder



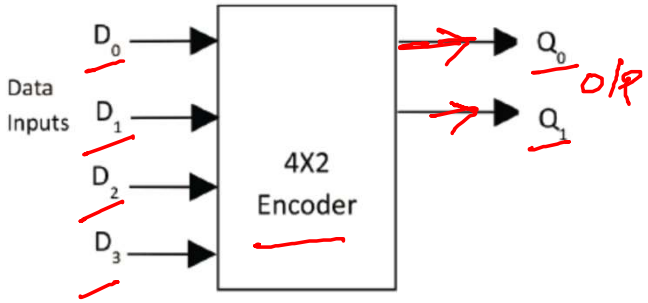
A	B	Q_0	Q_1	Q_2	Q_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



Encoders (2^n : n)

- Performs inverse operation of decoder
- Encoder has 2^n inputs and n outputs
- 4 to 2 line encoder
- e.g. octal to binary encoder

Inputs				Outputs	
D_3	D_2	D_1	D_0	Q_1	Q_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	0	1
0	0	0	0	X	X

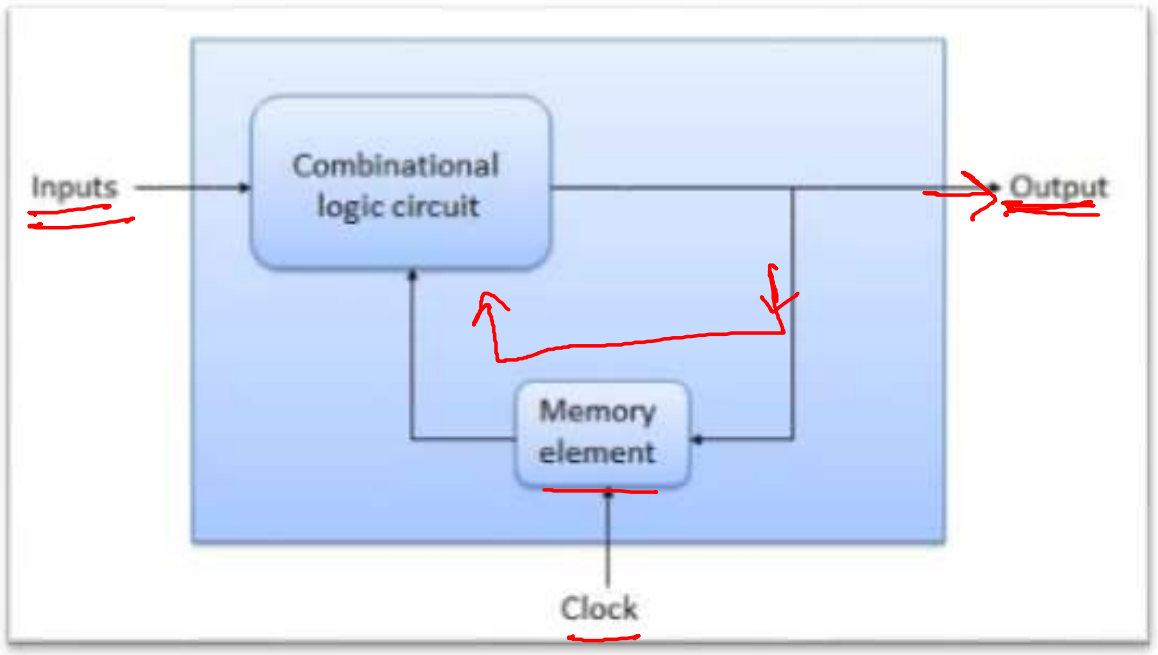


- **Priority encoder**

  If 2 or more inputs are 1 at same time, input having highest priority will take precedence

Sequential Circuits

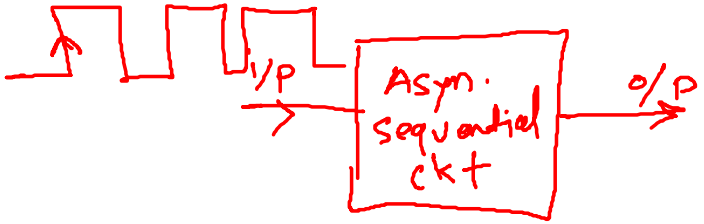
- Sequential circuit is a circuit whose output depends upon the present input, previous output and the sequence in which the inputs are applied.
- Sequential circuit has memory which are capable of storing binary information.



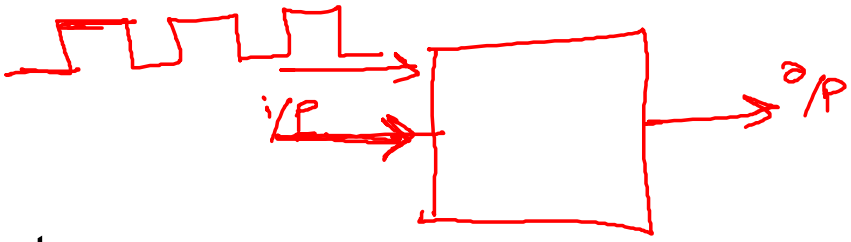
0
1

Sequential Circuits

- **Types of sequential circuits:**
  - Asynchronous sequential circuit
  - Synchronous sequential circuit.



- **Asynchronous sequential circuit**
  - These circuits do not use clock signal but uses the pulses of the inputs.
  - These circuits are faster than Synchronous sequential circuit.
  - These circuits are more difficult to design.



- **Synchronous sequential circuit**
  - These circuits uses clock signal and level inputs.
  - These circuits are bit slower than Asynchronous sequential circuit.
  - These circuits are easier to design.

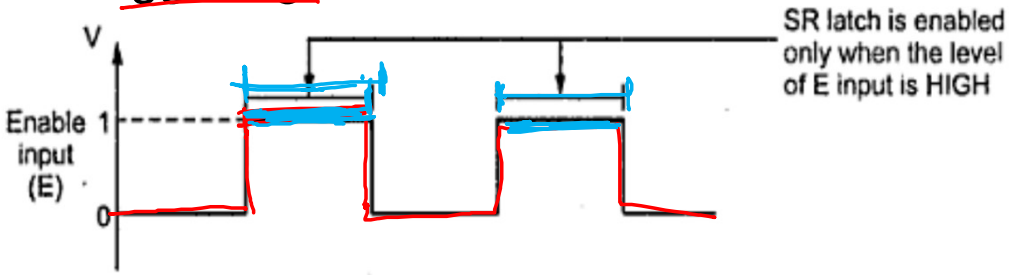
Sequential Circuits

- **Flip flop**
  - A flip-flop is a binary storage device capable of storing 1 bit of information. In a stable state the o/p of a flip-flop is either 0 or 1.
  - Flip-flop operates with clock, so output changes only at the clock signal.
  - Flip-flop is said to be edge sensitive.

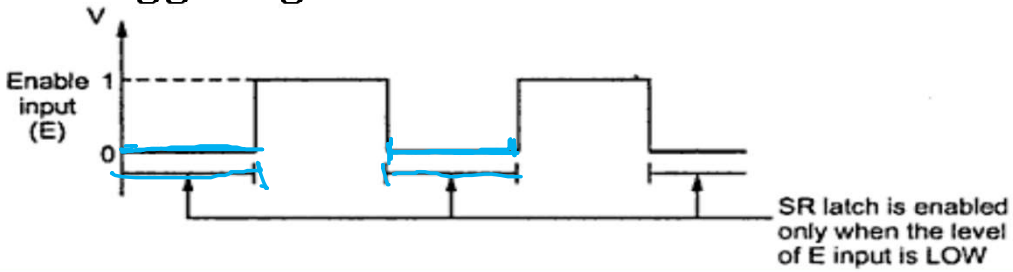
- **Latch**
  - A latch is a binary storage device capable of storing 1 bit of information.
  - Latch is an un-clocked flip-flop, so output changes at any instant of time doesn't depend on clock.
  - Latch is said to be level sensitive.

Sequential Circuits

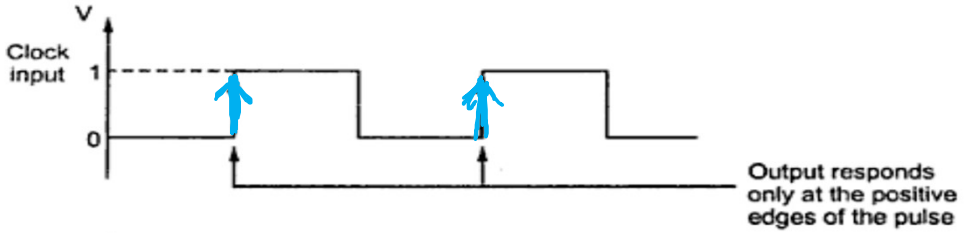
- **Positive Level-triggered**
  - The output changes during the high voltage period it is called positive level triggering.



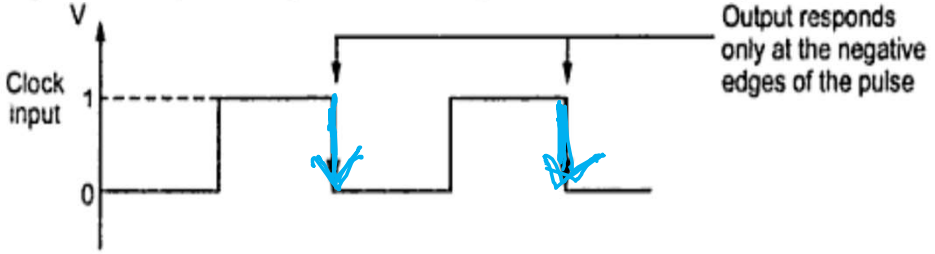
- **Negative Level-triggered**
  - The output changes during the low voltage period it is called negative level triggering.



- **Positive Edge-triggered**
  - An edge triggered flip-flop changes states at the positive edge (rising edge) of the clock pulses.



- **Negative Edge-triggered**
  - An edge triggered flip-flop changes states at the negative edge (falling edge) of the clock pulses.



Sequential Circuits

- A Flip Flop is a memory element that is capable of storing one bit of information.
- It is also called as Bi-stable multivibrator since it has two stable states either 0 or 1.

- **Types of Flip-Flops:**
  - SR Flip-Flop
  - D Flip-Flop
  - T Flip-Flop
  - JK Flip-Flop

NAND / NOR

SR Flip-Flop

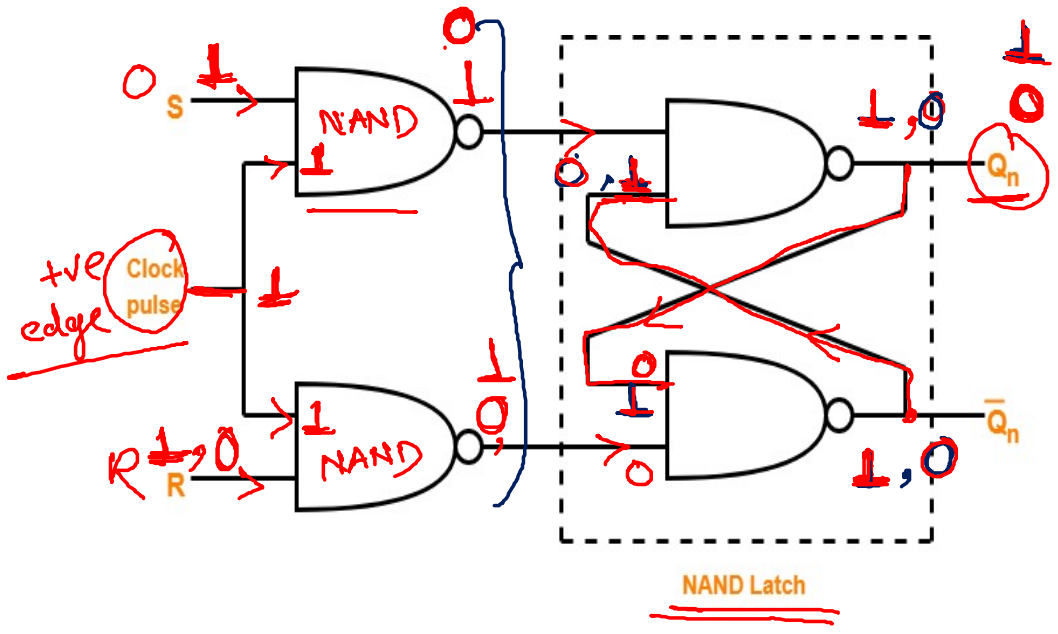
Qn → past value/o/p , Qn+1 → present

- SR Flip flop is the simplest type of flip-flops.
- It stands for **Set Reset flip-flop** and It is clocked flip flop.

Clk	S	R	Qn	Qn+1	State
↑	0	0	0	0	No change
↑	0	0	1	1	
↑	0	1	0	0	Reset
↑	0	1	1	0	
↑	1	0	0	1	Set
↑	1	0	1	1	
↑	1	1	0	X	Indetermind/
↑	1	1	1	X	Intermediate

NAND

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

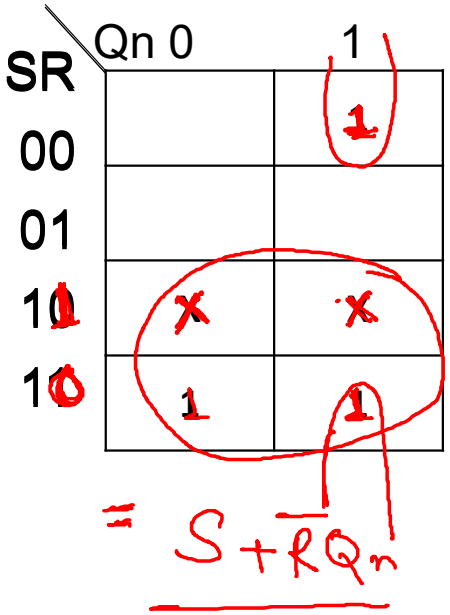


NAND Latch

SR Flip Flop Using NAND Latch

SR Flip-Flop

- **Characteristics Equation:**

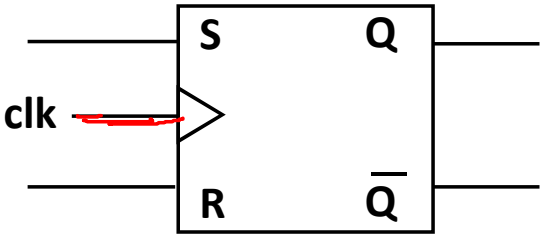


- **Excitation Table:**

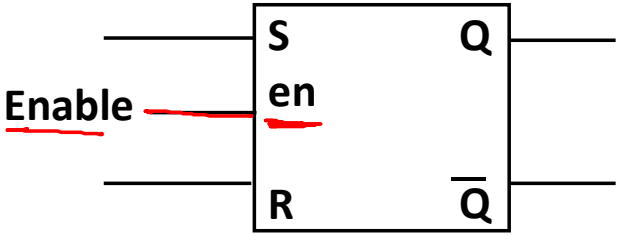
Qn	Qn+1	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

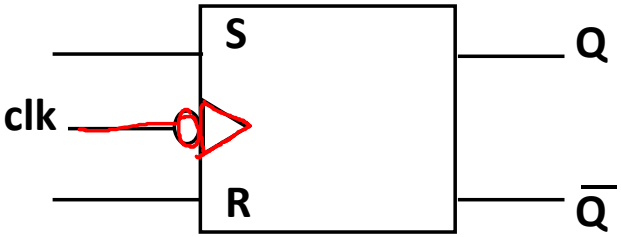
Sequential Circuits

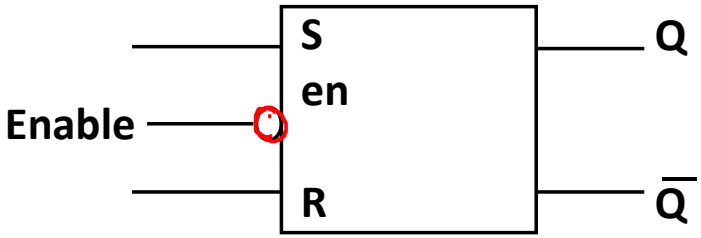
- SR Flip Flop



SR Latch



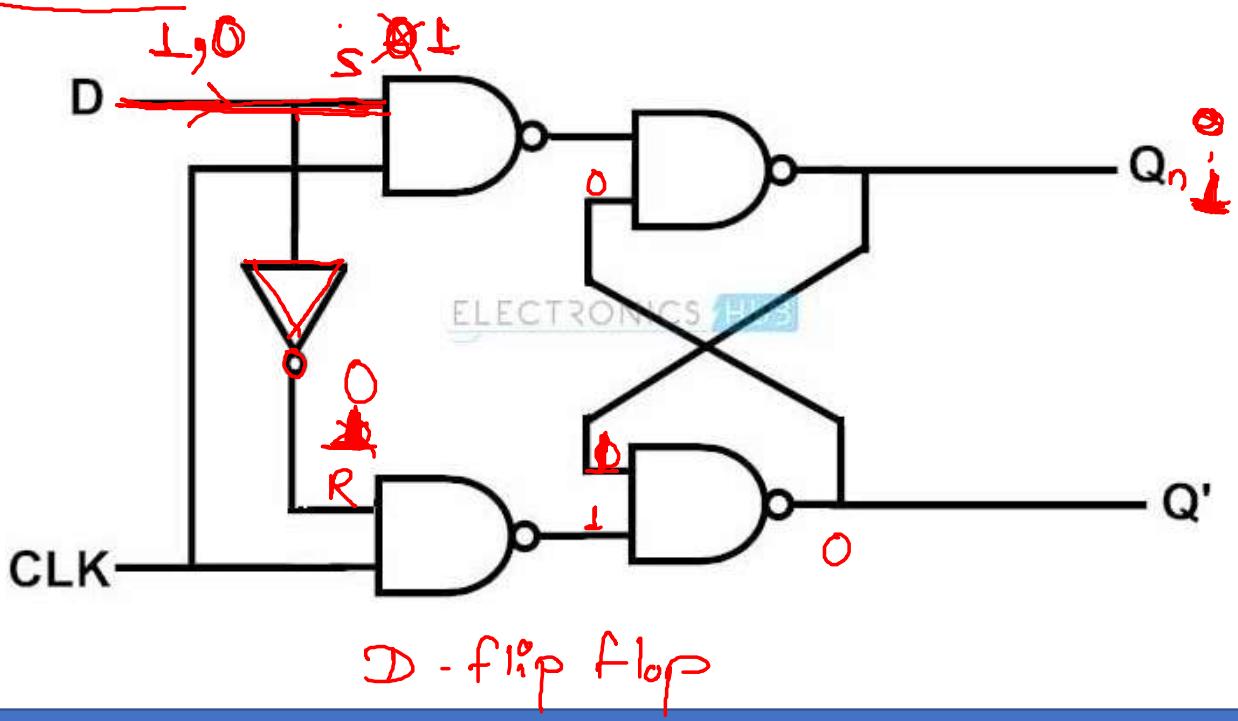




D Flip-Flop

- D flip-flops are also called as Delay flip–flop or Data flip–flop. They are used to store 1–bit binary data
- The S input is given with D input and the R input is given with inverted D input.
- D Flip-flop is also called as transparent latch.

Clk	D	Qn	Qn+1
↑	0	0	0
↑	0	1	0
↑	1	0	1
↑	1	1	1



D - flip flop

D Flip-Flop

- **Excitation Table:**

Qn	Qn+1	D
0	0	0
0	1	1
1	0	0
1	1	1

- **Characteristics Equation:**

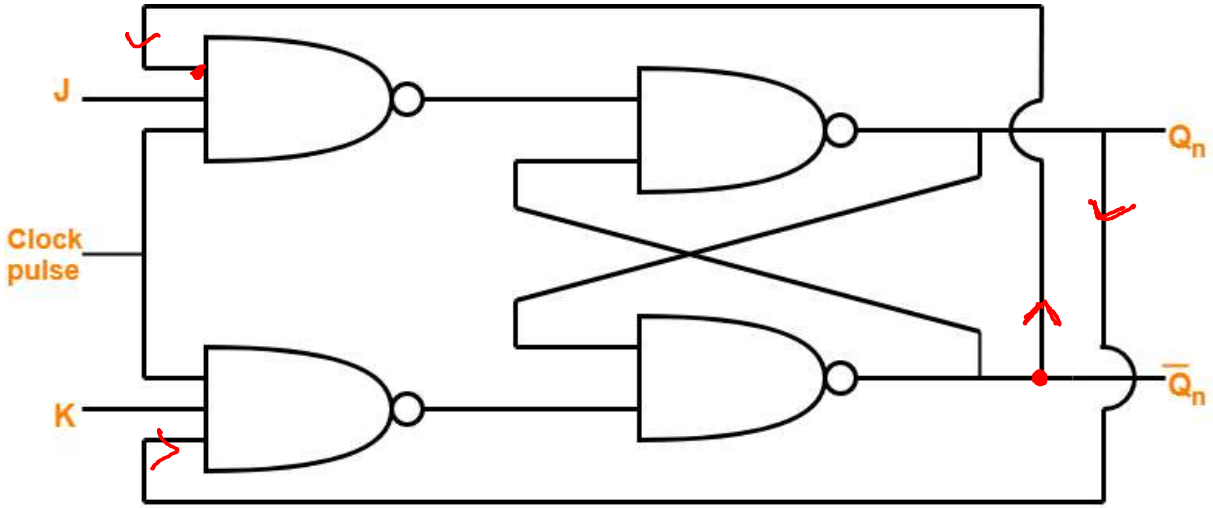
D \ Qn	0	1
0	0	0
1	1	

- **Applications of D**
  - Data storage registers.
  - Data transferring as shift registers.
  - Frequency division circuits.

JK Flip-Flop

- Input J behaves like input S of SR flip-flop which was meant to set the flip-flop.
- Input K behaves like input R of SR flip-flop which was meant to reset the flip flop.

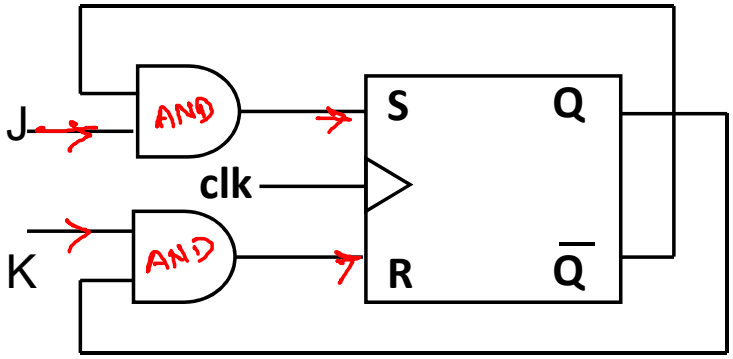
- **Applications**
  - Shift Registers
  - Frequency Dividers
  - Switching Applications
  - Parallel Data Transfer
  - Serial Data Transfer
  - Binary Counter
  - Sequence Detector



Logic Circuit For JK Flip Flop Using SR Flip Flop

(Constructed Using NAND Latch)

JK Flip-Flop



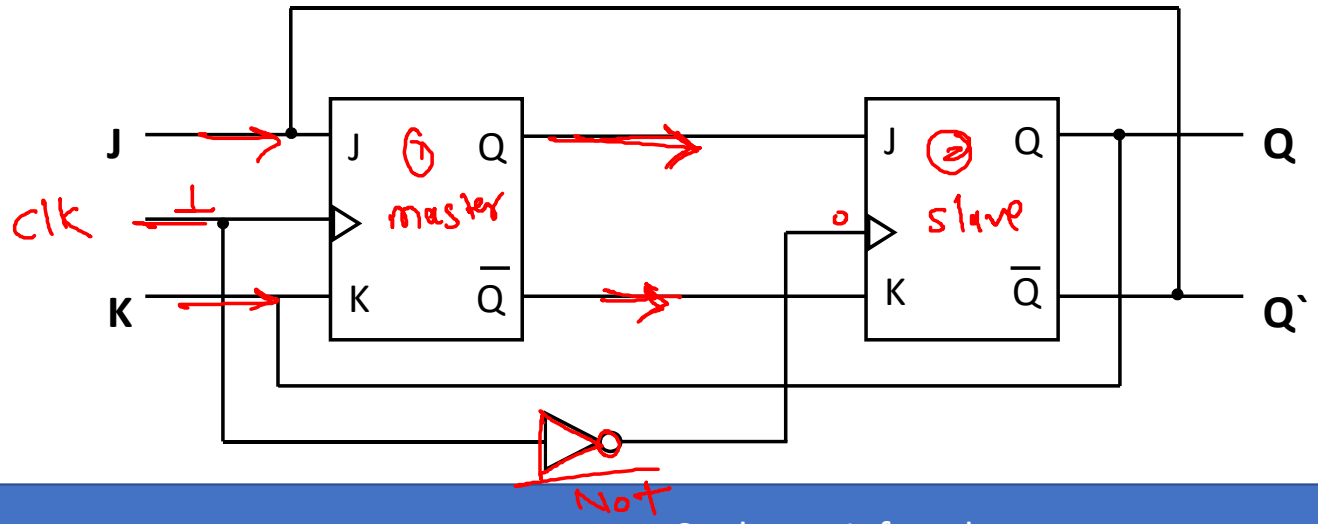
Excitation Table:

Qn	Qn+1	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Clk	J	K	Qn	Qn+1	State
↑	0	0	0	0	No change
↑	0	0	1	1	
↑	0	1	0	0	Reset
↑	0	1	1	0	
↑	1	0	0	1	set
↑	1	0	1	1	
↑	1	1	0	→ 1	Toggle
↑	1	1	1	→ 0	

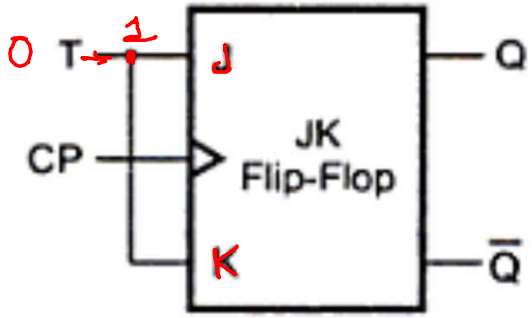
MASTER SLAVE J-K FLIP-FLOP

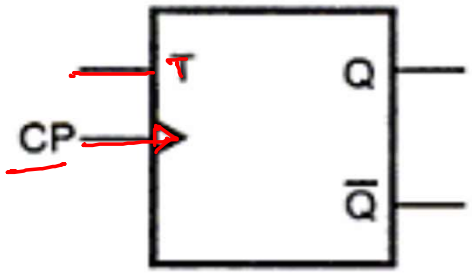
- **Race around condition**
  - In JK flip flop, when J = K = 1, the output toggles
  - This toggling will continue until flip flop is enabled and J=k=1
  - At the end of clock, the flip flop is disabled and output is uncertain
  - This condition is called as race around condition
  - This is overcome by using master slave flip flop



T Flip Flop

- Also known as "Toggle Flip Flop"
- It is modification of JK flip flop





Logic Symbol

- When T = 0, J = K = 0 -> no change
- When T = 1, J = K = 1 -> toggles

Qn	T	Qn+1
0	0	0
0	1	1
1	0	1
1	1	0

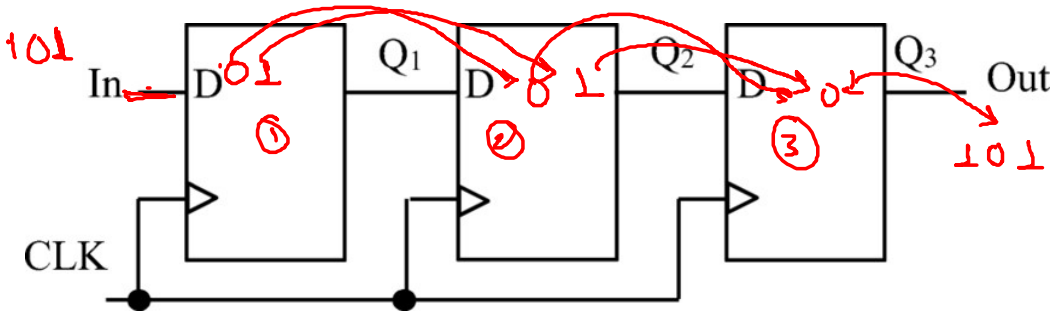
T		Qn+1
0	→	Qn
1	→	$\overline{Qn}$

Shift Registers

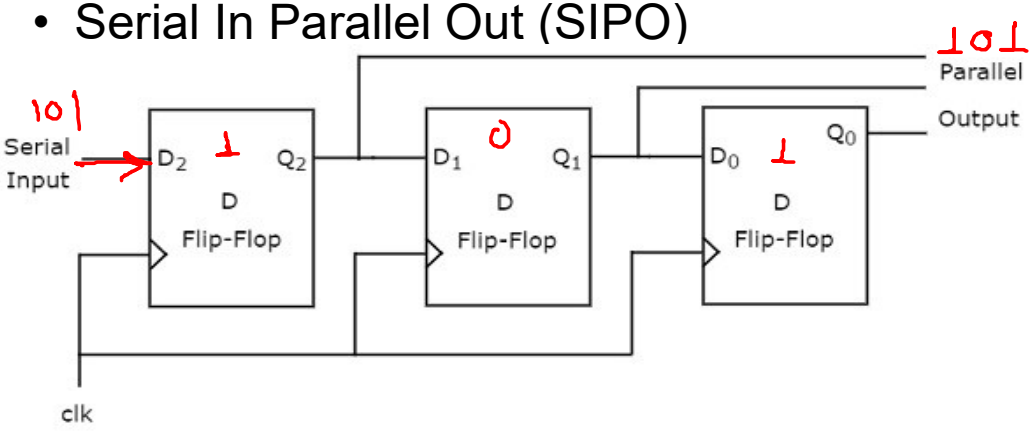
- Flip flops can store a single bit of binary data i.e. 1 or 0.
- Register is a group of flip flops used to store multiple bits of data.
- When a number of flip flops are connected in series, this arrangement is called a Register.
- The stored information can be transferred within the registers; these are called as 'Shift Registers'.
- Shift registers are of 4 types
  - Serial In Serial Out shift register SISO
  - Serial In parallel Out shift register SIPO
  - Parallel In Serial Out shift register PISO
  - Parallel In parallel Out shift register PIPO
- The registers which will shift the bits to left are called "Shift left registers".
- The registers which will shift the bits to right are called "Shift right registers".

Shift Registers

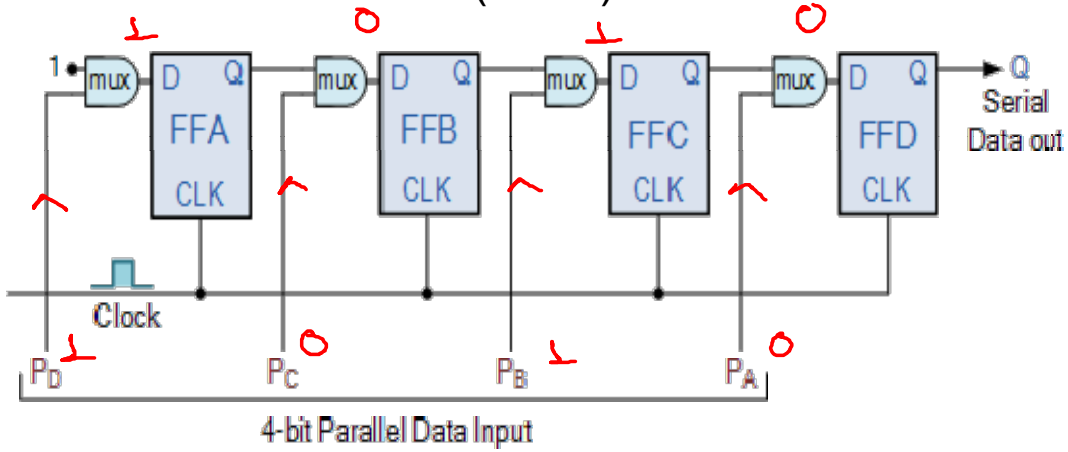
- Serial In Serial Out (SISO)



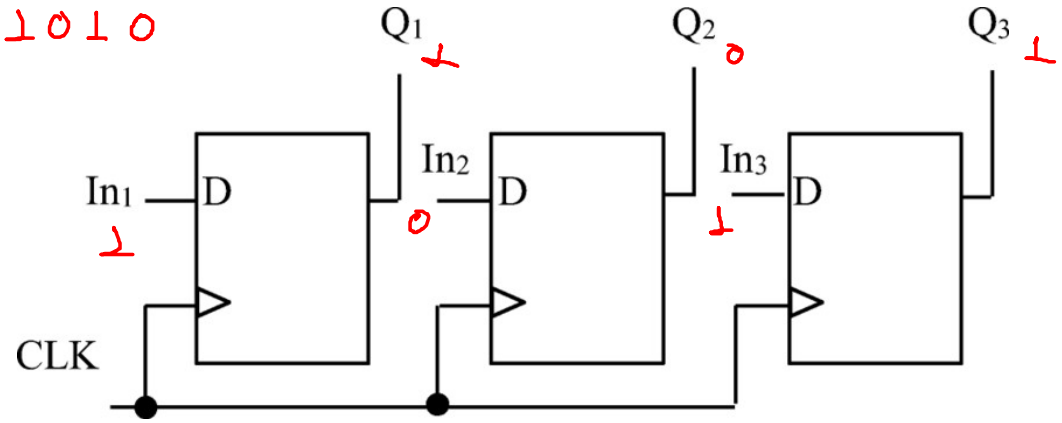
- Serial In Parallel Out (SIPO)



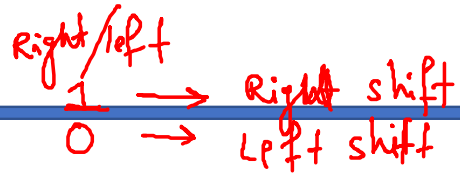
- Parallel In Serial Out (PISO)



- Parallel In Parallel Out (PIPO)



Bidirectional and Universal Shift Register



Bidirectional Shift Register

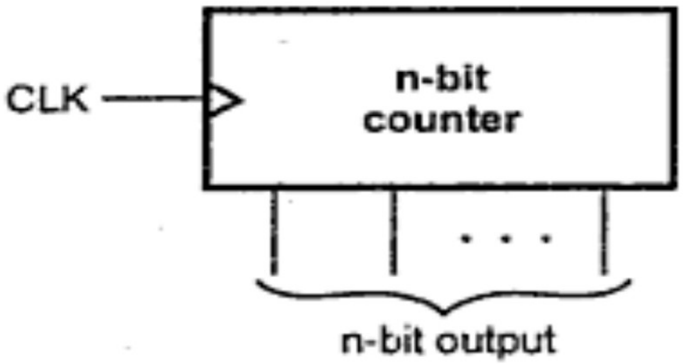
- The bidirectional shift register can be defined as the register in which the data can be shifted either left or right.
- Right /Left is the mode signal. When Right /Left is a 1, the logic circuit works as a right shift register. When Right /Left is a 0, the logic circuit works as a left shift register.

Universal Shift Register.

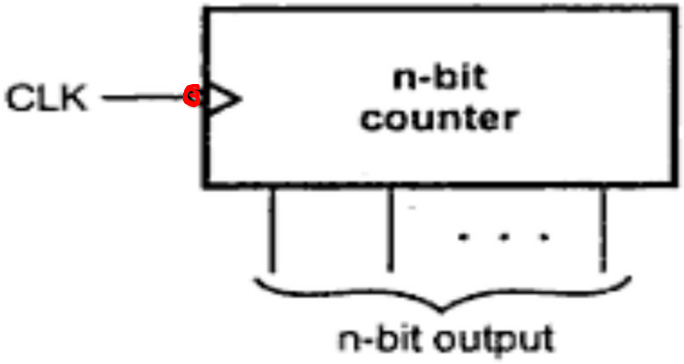
- The universal shift register can be defined as the register which can be used to shift the data in both the directions like left, right and can load parallel data as well.
- It is called Universal Shift Register as it can be used for left shift, right shift, serial to serial, serial to parallel, parallel to serial and parallel to parallel operations.

Counters

- A digital circuit which is used for a counting pulses is known counter, counter is sequential circuit.
- Counters will count the number of pulses applied to the count input.
- Counters are constructed using Flip flops and logic gates.
- Counters can be positive edge triggered or negative edge triggered



(a) Positive edge triggered n-bit counter

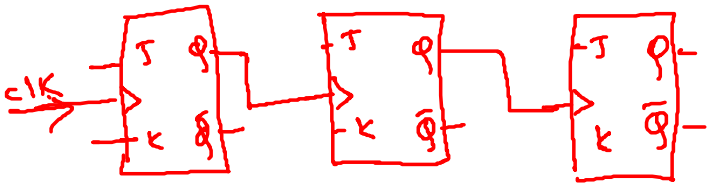


(b) Negative edge triggered n-bit counter

Counters

Types of counters

- Depending on type of clock
- There are two types of counters



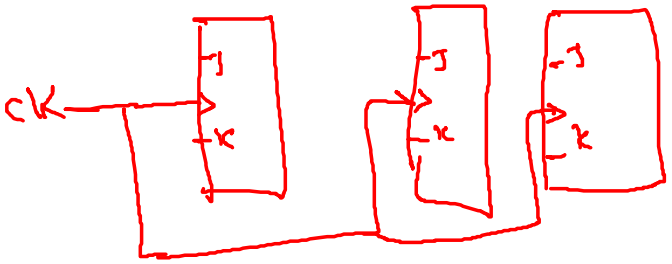
1. Asynchronous/Ripple counter
   - The first flip-flop is clocked by the external clock pulse and then each successive flip-flop is clocked by the output of the preceding flip flop.
   - Asynchronous counters are also called as Truncated counters.
2. Synchronous counter

Counters

Types of counters

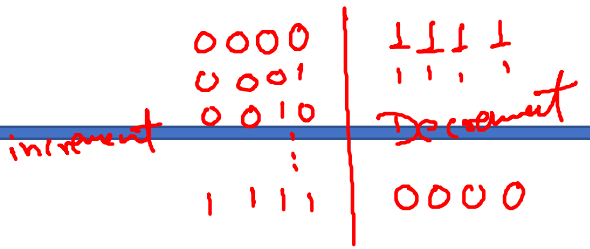
There are two types of counters

1. Asynchronous/Ripple counter
   - The first flip-flop is clocked by the external clock pulse and then each successive flip-flop is clocked by the output of the preceding flip flop.
   - Asynchronous counters are also called as Truncated counters.



2. Synchronous counter
   - In synchronous counters, the clock input is connected to all of the flip-flops so that they are clocked simultaneously.

Counters



- **Up/Down counter**
  - An 'N' bit binary counter consists of 'N' flip-flops. If the counter counts from 0 to $2^n - 1$(zero-max), then it is called as binary up counter.
  - Example : 4- bit counter has 4 FF and counter counts from 0000 to 1111
  - 0Similarly, if the counter counts down from $2^n - 1$ to 0(max-zero), then it is called as binary down counter.
  - Example : 4- bit counter has 4 FF and counter counts from 1111 to 0000
- **Application of counters**
  - Frequency counters
  - Digital clock
  - Time measurement
  - A to D converter
  - Frequency divider circuits
  - Digital triangular wave generator

Difference between asynchronous and synchronous counters

Asynchronous counters	Synchronous counters
Output of first flip flop drives clock input of the next flip flop	No connection between output of first flip flop and clock input
All flip flops are not clocked simultaneously	All flip flops are clocked simultaneously
Logic circuit is very simple even for more number of states	Design involves complex logic as number of states increases
Low speed counters	High speed counters

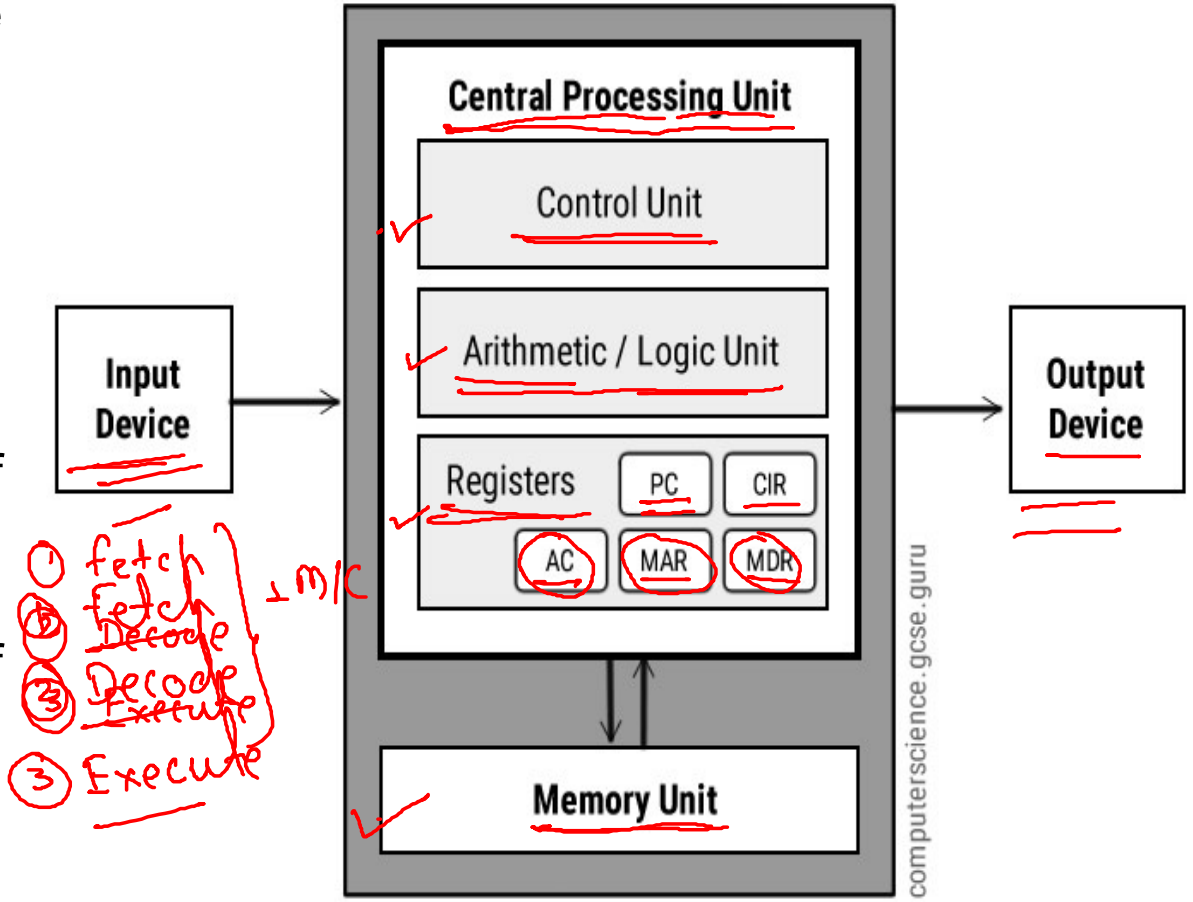
Section – C
Computer Architecture

Trainer :- Sohail Inamdar

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

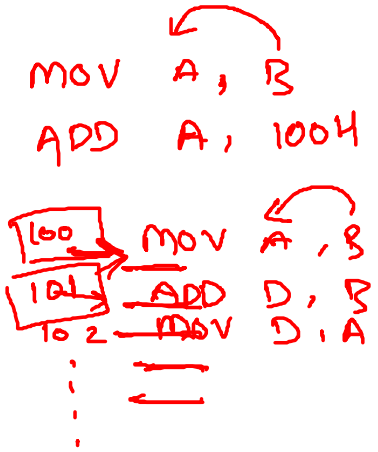


Computer Architecture

- 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

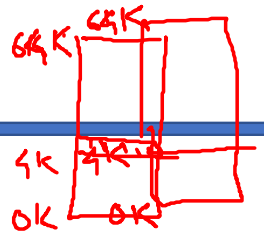
- **Registers**
  - Registers are high speed storage areas in the CPU. All data must be stored in a register before it can be processed.
  - **MAR - Memory Address Register**
    - Holds the memory location of data that needs to be accessed
  - **MDR - Memory Data Register**
    - Holds data that is 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

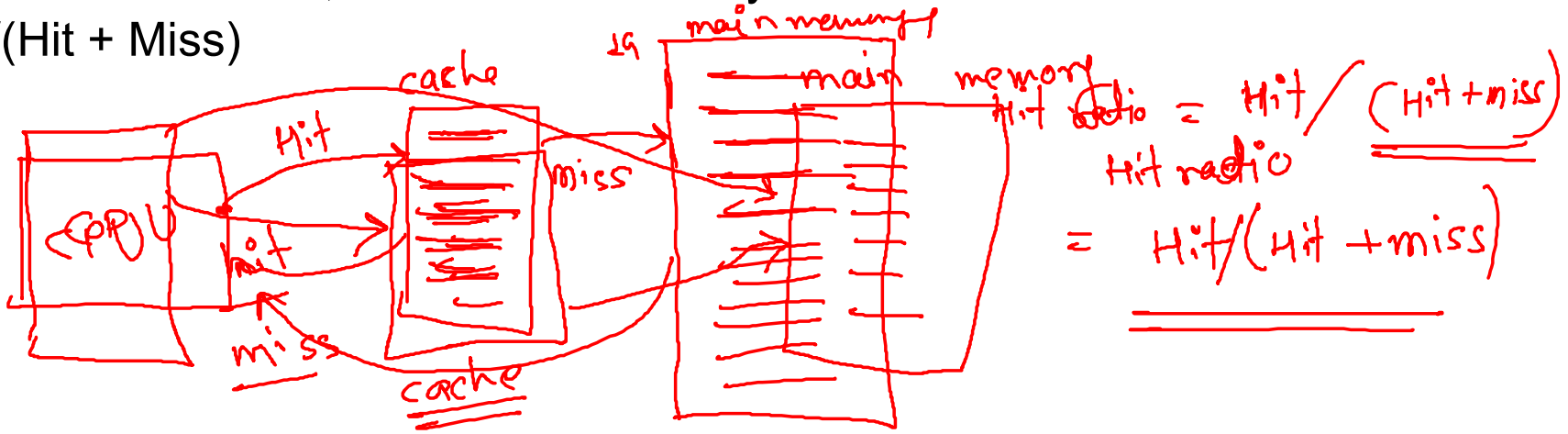


- **Memory Unit**
  - A Memory Unit is a collection of storage cells together with associated circuits needed to transfer information in and out of storage.
  - Two major types of memories are used in computer systems:
  - **Random Access Memory(RAM) and Read Only Memory(ROM).**
  - The memory unit consists 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.

- **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} = 1024 bytes = 1 KB = 1 kb

2^{20} = 1024 kbytes = 1 MB = 1 Mb

2^{30} = 1024 Mbytes = 1 GB = 1 Gb

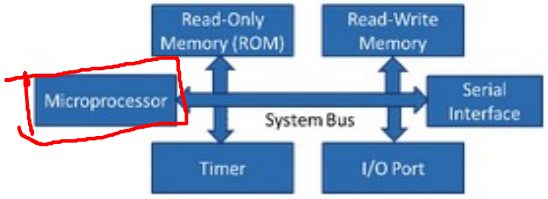
2^{40} = 1024 GBytes = 1 TB = 1 Tb

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

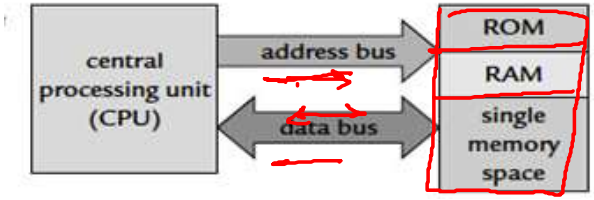
Microprocessor	Microcontroller
Microprocessor is heart of computer system	Microcontroller is heart of embedded system
Memory and IO components have to be connected externally	Microcontroller has external processor along with internal memory and IO components
Circuit becomes large	Circuit is small
Cannot 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

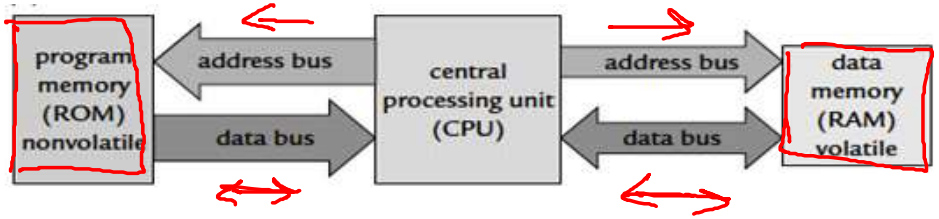


Difference between Von Neumann and Harvard

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

ARM, AVR, Power PC



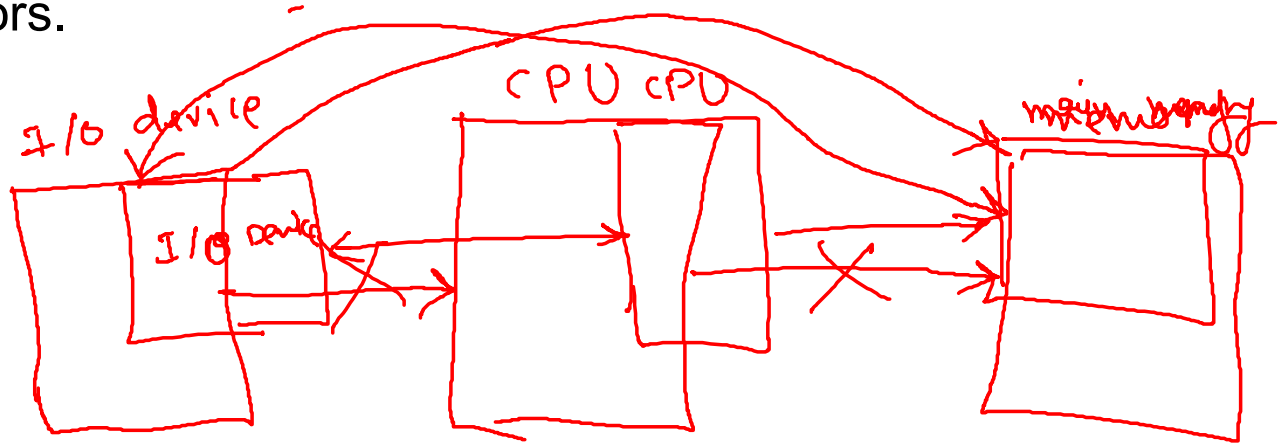


Difference between RISC and CISC

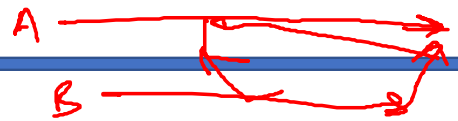
RISC	CISC
Less number of instructions	More number of instructions
All instructions are micro instructions - CPU don't further divide it into smaller parts	Some instructions are macro instructions - CPU further divide instruction in smaller parts and then executes
Most of the Instructions are executed 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.

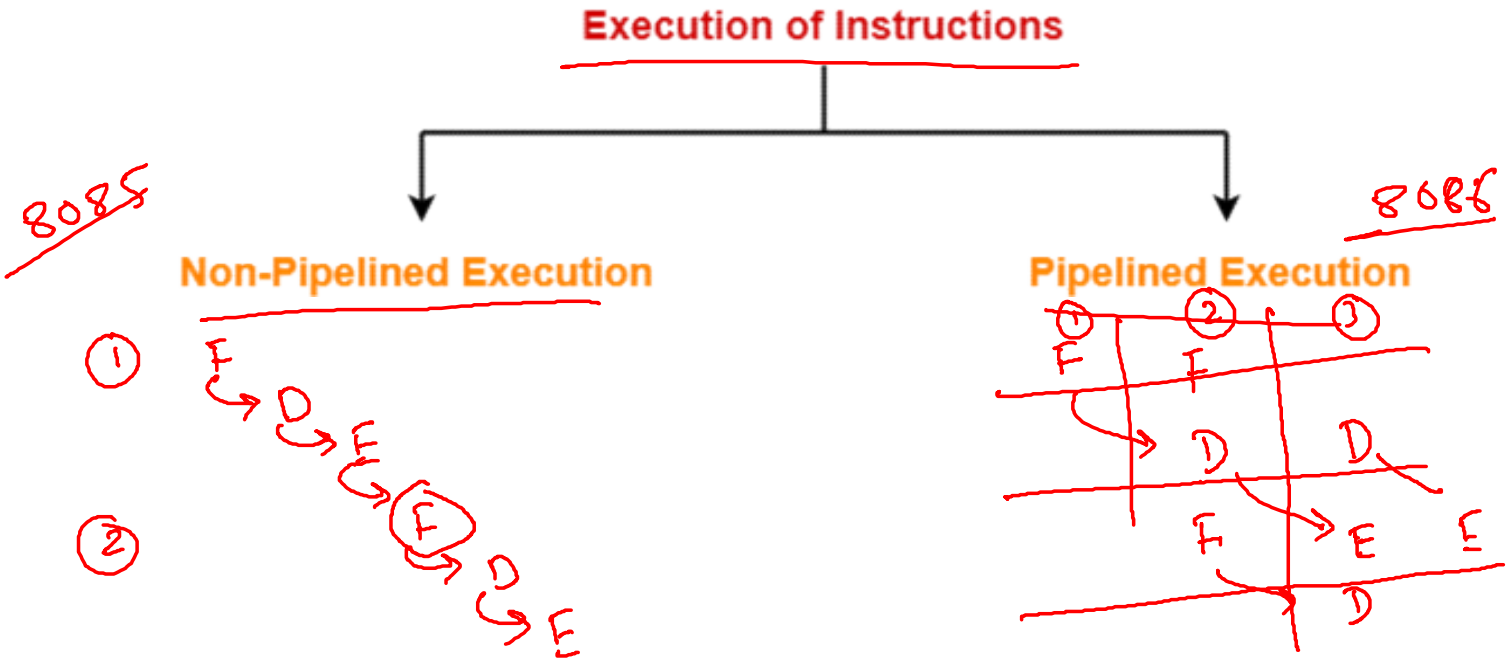
Computer Architecture

- **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

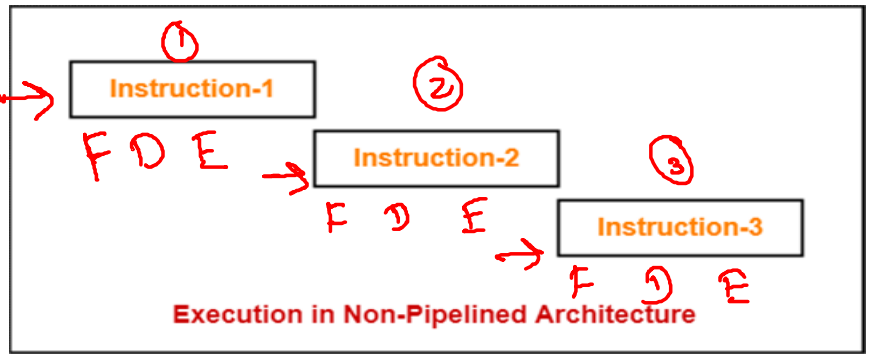
Fetch
Decode
Execute



- **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

t = 0.5 sec
n = 3
= n x t
= 0.5 x 3 = 1.5 sec



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 cycle = 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)
      - Readback (RB)

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 :

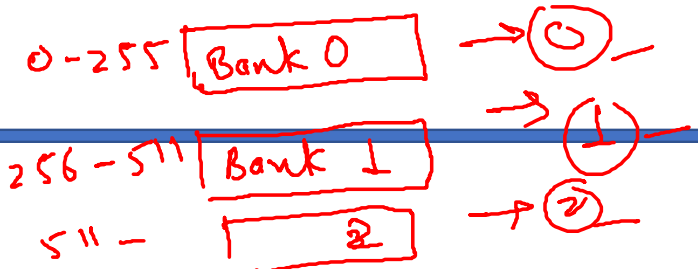
If we have 5 number of instruction and 4 number of stages then total how many cycle are required in pipelining ?

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

64K

- 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.