II Semester -Computer Architecture

UNIT 1

Syllabus:

Number System: Binary, Octal, Hexa-decimal numbers, base conversion, addition, subtraction of binary numbers, one's and two's complements, positive and negative numbers, character codes ASCII, EBCDIC.

Computer Arithmetic: Addition and Subtraction, Multiplication and Division, Integer Arithmetic Operations. Digital Logic Circuits: Logic gates, Boolean algebra, Map Simplification.

Combinational Circuits: Half Adder, Full Adder, Flip Flops.

Sequential circuits: Shift registers, Counters, Integrated Circuits, Multiplexers, Demultiplexers, Encoder, Decoder.

Topics highlighted in syllabus to be referred from class notes

Binary Subtraction:

Subtraction can be performed in three ways:

1. Ordinary Binary Subtraction
2. Subtraction using 1's complement
3. Subtraction using 2's complement

Minued	Subtrahend	Difference	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

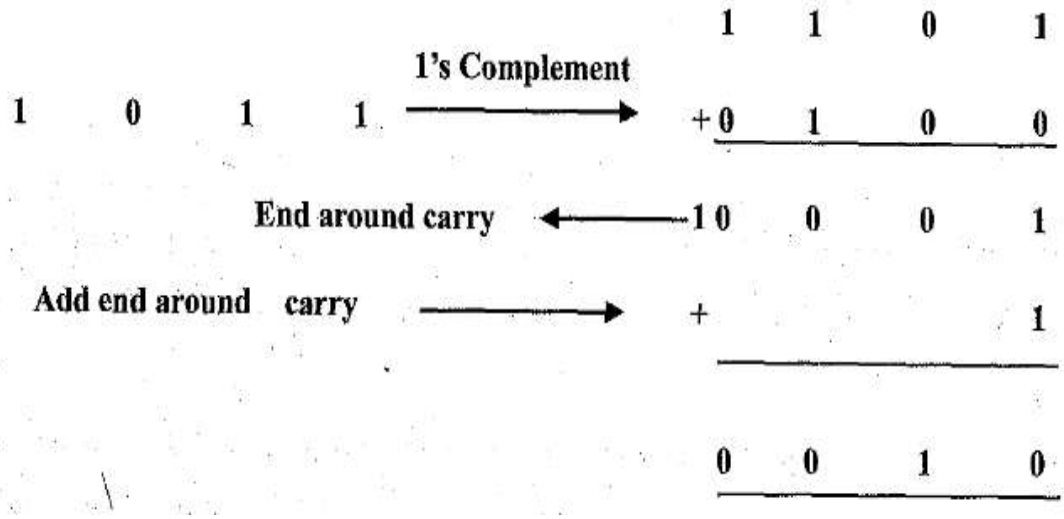
Example: $1101_2 - 1011_2 = ?$

Borrow	→	0	0	1	0
Minuend	→	1	1	0	1
Subtrahend	→	1	0	1	1
Difference	→	0	0	1	0

1. Subtraction using 1's complement
   - Steps:
     1. Obtain 1's complement of the smaller number
     2. Add the larger number
     3. The addition always results in a final carry called end around carry
     4. Remove the end around carry and add it to the result. The final result will be in true binary form

- Note: both number should have same number of bits

Example: Subtract 1011_2 from 1101_2 using 1's complement method.



2. **Subtraction using 2's complement**
   - Steps:
     1. Determine the 2's complement of the smaller number
     2. Add the 2's complement to the larger number
     3. There is always an end around carry. Discard the carry

•**Example:** Subtract 11011_2 from 101101_2 using 2's complement method.

2's complement of 011011 is 100100

$$\begin{array}{rr} & 100100 \\ & +\quad 1 \\ \hline & 100101 \end{array}$$

$$\begin{array}{rr} & 101101 \\ -011011 & +100101 \\ \hline & 1010010 \end{array}$$

Discard the carry

$\therefore\ 101101_2 - 011011_2 = 10010_2$

Combinational Circuit:

Combinational Logic Circuits are made from the basic and universal gates. The output is defined by the logic and it is depend only the present input states not the previous states.

Inputs and output(s) : logic 0 (low) or logic 1 (high).

Fig. Block diagram of a combinational circuits

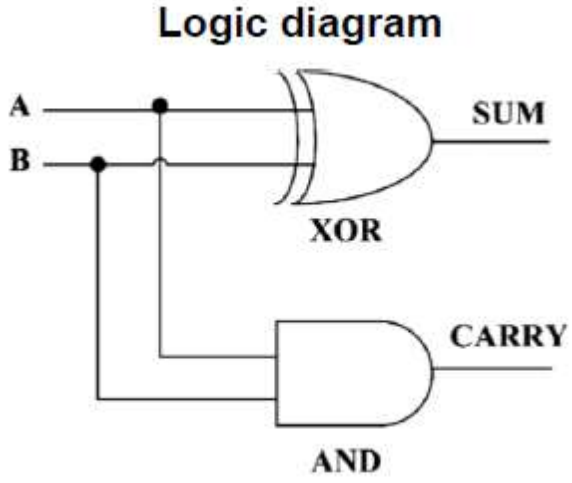
Adder

- The basic operation in digital computer is binary addition. The circuit which perform the addition of binary bits are called as Adder.
- The logic circuit which perform the addition of two bits is called half adder and 3 bits is called full adder.

Binary Addition:
0+0=0
0+1=1
1+0=1
1+1=10

Half Adder

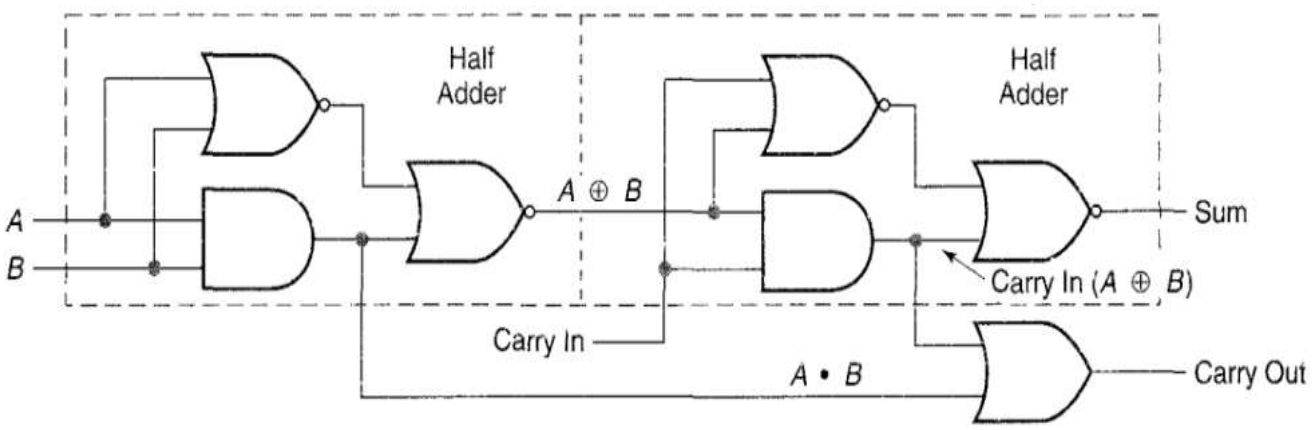
- Half adder is a combinational logic circuit with two inputs and two outputs.
- The half adder circuit is designed to add two single bit binary number A and B.
- It is the basic building block for addition of two single bit numbers. This circuit has two outputs carry and sum



Inputs		Outputs	
B	A	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Full Adder

Full adder is developed to overcome the drawback of Half Adder circuit. It can add two one-bit numbers A and B, and carry c. The full adder is a three input and two output combinational circuit.



Inputs			Outputs	
B	A	Carry In	Carry Out	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Flip Flop:A flip flop is a binary cell capable of storing one bit of information

Types of Flip Flop

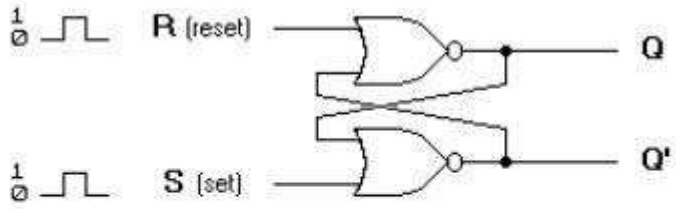
- S-R Flip Flop
- D Flip Flop
- J-K Flip Flop
- T Flip Flop

1. S-R Flip Flop

The SET-RESET flip flop is designed with the help of two NOR gates and also two NAND gates. These flip flops are also called S-R Latch.

- **S-R Flip Flop using NOR Gate**

The design of such a flip flop includes two inputs, called the SET [S] and RESET [R]. There are also two outputs, Q and Q'. The diagram and truth table is shown below.



(a) Logic diagram

S	R	Q	Q'	
1	0	1	0	
0	0	1	0	(after S=1, R=0)
0	1	0	1	
0	0	0	1	(after S=0, R=1)
1	1	0	0	

(b) Truth table

Basic flip-flop circuit with NOR gates

From the diagram it is evident that the flip flop has mainly four states. They are

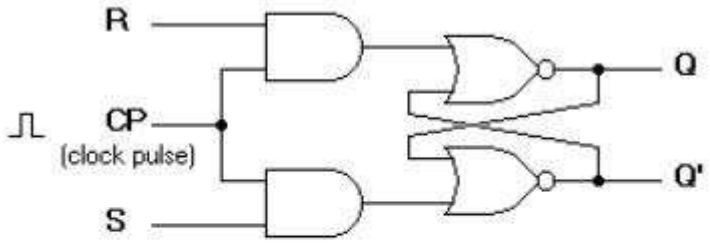
S=1, R=0—Q=1, Q'=0
This state is also called the SET state.
S=0, R=1—Q=0, Q'=1
This state is known as the RESET state.
In both the states you can see that the outputs are just compliments of each other and that the value of Q follows the value of S.
S=0, R=0—Q & Q' = Remember
If both the values of S and R are switched to 0, then the circuit remembers the value of S and R in their previous state.
S=1, R=1—Q=0, Q'=0 [Invalid]
This is an invalid state because the values of both Q and Q' are 0. They are supposed to be compliments of each other. Normally, this state must be avoided.

Clocked S-R Flip Flop:

It is also called a Gated S-R flip flop.

The problems with S-R flip flops using NOR and NAND gate is the invalid state. This problem can be overcome by using a bistable SR flip-flop that can change outputs when certain invalid states are met,

regardless of the condition of either the Set or the Reset inputs. For this, a clocked S-R flip flop is designed by adding two AND gates to a basic N\OR Gate flip flop. The circuit diagram and truth table is shown below.



(a) Logic diagram

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

(b) Truth table

Clocked SR flip-flop

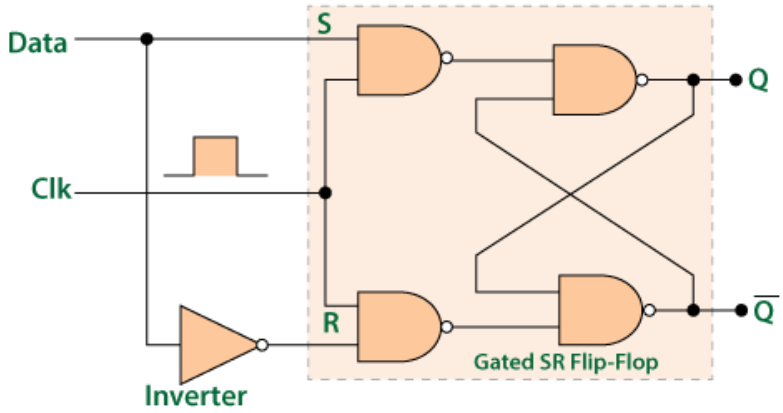
A clock pulse [CP] is given to the inputs of the AND Gate. When the value of the clock pulse is '0', the outputs of both the AND Gates remain '0'. As soon as a pulse is given the value of CP turns '1'. This makes the values at S and R to pass through the NOR Gate flip flop. But when the values of both S and R values turn '1', the HIGH value of CP causes both of them to turn to '0' for a short moment. As soon as the pulse is removed, the flip flop state becomes intermediate. Thus either of the two states may be caused, and it depends on whether the set or reset input of the flip-flop remains a '1' longer than the transition to '0' at the end of the pulse. Thus the invalid states can be eliminated.

2. D Flip Flop(Delay/Data FF):

- The D flip flop is the most important flip flop from other clocked types.
- It ensures that at the same time, both the inputs, i.e., S and R, are never equal to 1.
- The Delay flip-flop is designed using a gated SR flip-flop with an inverter connected between the inputs allowing for a single input D(Data).

- This single data input, which is labeled as "D" used in place of the "Set" input and for the complementary "Reset" input, the inverter is used.

The circuit diagram and truth table is given below.

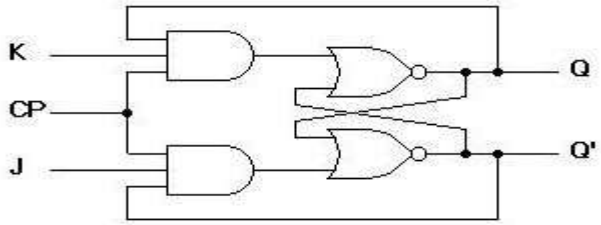


Clock	D	Q	Q'	Description
↓ » 0	X	Q	Q'	Memory no change
↑ » 1	0	0	1	Reset Q » 0
↑ » 1	1	1	0	Set Q » 1

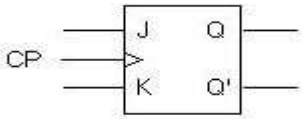
- The D input is passed on to the flip flop when the value of CP is '1'. When CP is HIGH, the flip flop moves to the SET state.
- If it is '0', the flip flop switches to the CLEAR state.

3. J-K Flip Flop

The circuit diagram and truth-table of a J-K flip flop is shown below.



(a) Logic diagram



(b) Graphical symbol

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

(c) Transition table

Clocked JK flip-flop

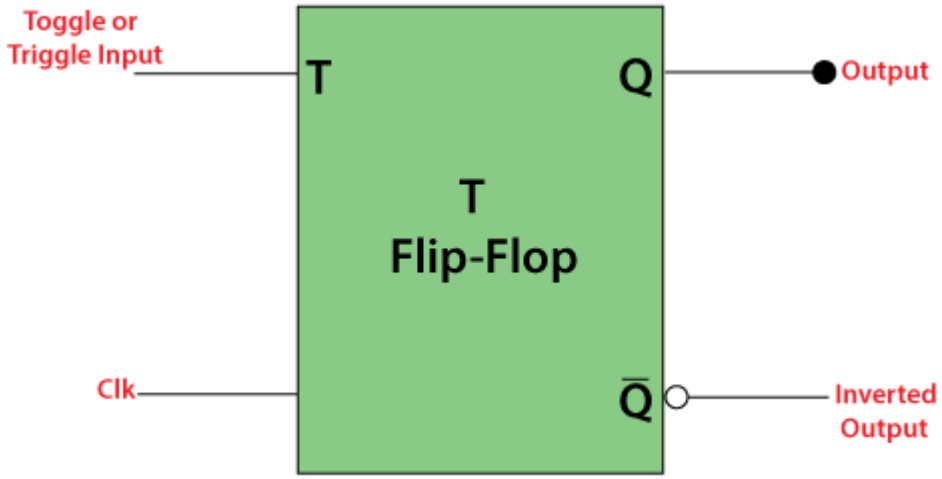
- A J-K flip flop can also be defined as a modification of the S-R flip flop. The only difference is that the intermediate state is more refined and precise than that of a S-R flip flop.
- The behavior of inputs J and K is same as the S and R inputs of the S-R flip flop. The letter J stands for SET and the letter K stands for CLEAR.
- When both the inputs J and K have a HIGH state, the flip-flop switch to the complement state. So, for a value of Q = 1, it switches to Q=0 and for a value of Q = 0, it switches to Q=1.
- The circuit includes two 3-input AND gates. The output Q of the flip flop is returned back as a feedback to the input of the AND along with other inputs like K and clock pulse [CP].
- So, if the value of CP is '1', the flip flop gets a CLEAR signal and with the condition that the value of Q was earlier 1.
- Similarly output Q' of the flip flop is given as a feedback to the input of the AND along with other inputs like J and clock pulse [CP]. So the output becomes SET when the value of CP is 1 only if the value of Q' was earlier 1.
- When both inputs 'J' and 'K' are set to 1, the JK toggles the flip flop

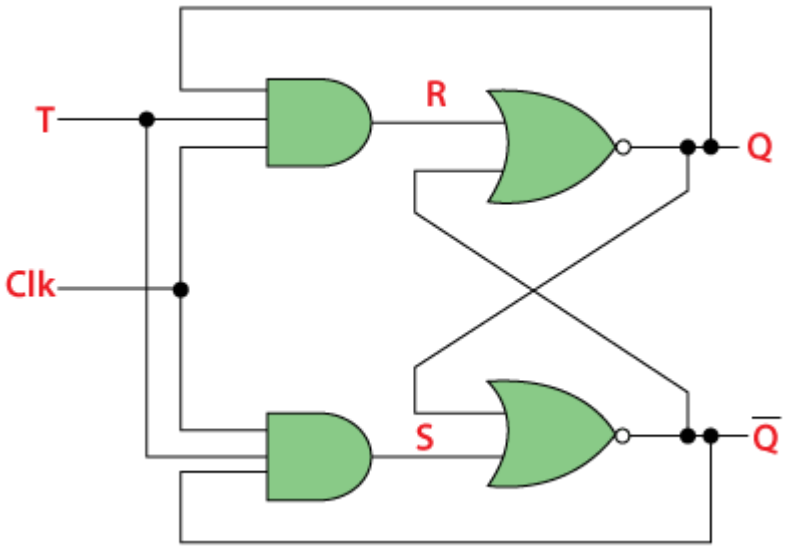
4. T Flip Flop

- In T flip flop, "T" defines the term "Toggle". In SR Flip Flop, we provide only a single input called "Toggle" or "Trigger" input to avoid an intermediate state occurrence.

- Now, this flip-flop work as a Toggle switch. The next output state is changed with the complement of the present state output. This process is known as "Toggling"'.
- We can construct the "T Flip Flop" by making changes in the "JK Flip Flop".
- The "T Flip Flop" has only one input, which is constructed by connecting the input of JK flip flop. This single input is called T. In simple words, we can construct the "T Flip Flop" by converting a "JK Flip Flop". Sometimes the "T Flip Flop" is referred to as single input "JK Flip Flop".

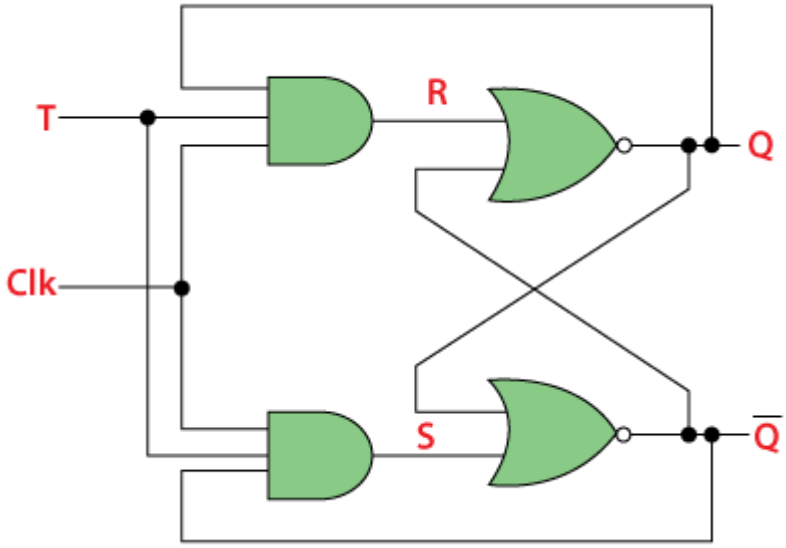
Block diagram of the "T-Flip Flop" is given where T defines the "Toggle input", and CLK defines the clock signal input.





- The "T Flip Flop" is designed by passing the AND gate's output as input to the NOR gate of the "SR Flip Flop".

- The inputs of the "AND" gates, the present output state Q, and its complement Q' are sent back to each AND gate.
- The toggle input is passed to the AND gates as input. These gates are connected to the Clock (CLK) signal. In the "T Flip Flop", a pulse train of narrow triggers are passed as the toggle input, which changes the flip flop's output state.
- The circuit diagram of the "T Flip Flop" using "SR Flip Flop" is given below:

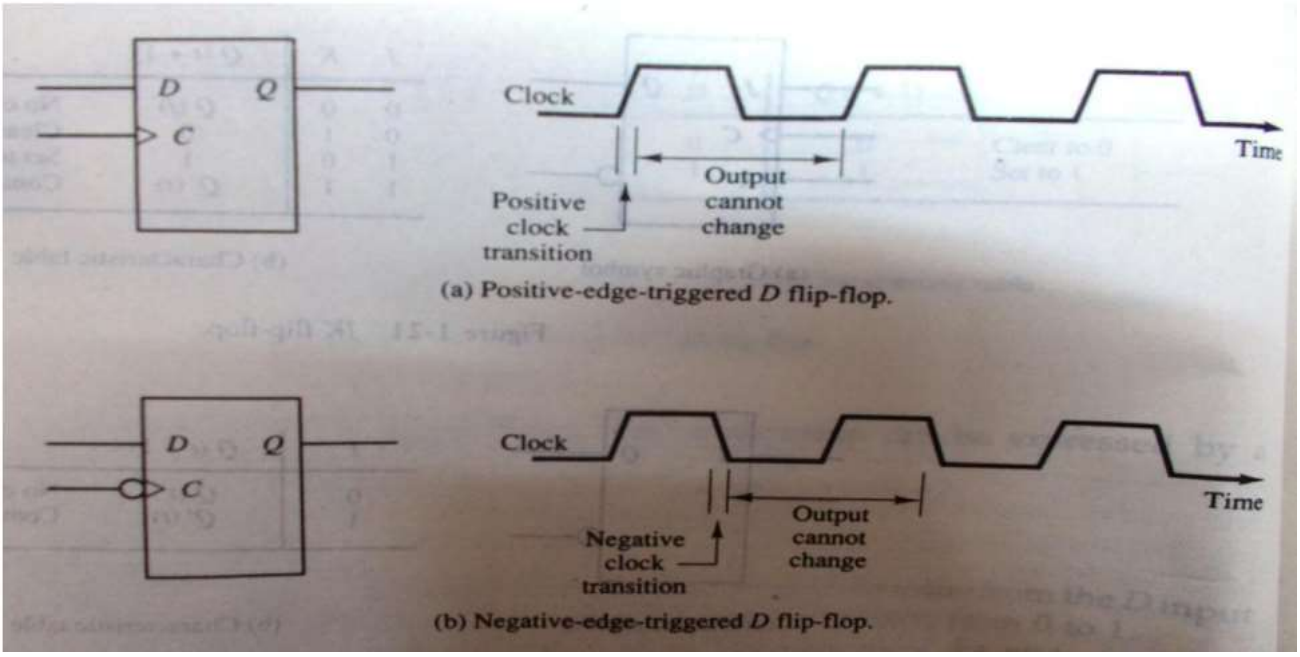


Truth Table of T Flip Flop

	Previous		Next	
T	Q	Q'	Q	Q'
0	0	1	0	1
0	1	0	1	0
1	0	1	1	0
1	1	0	0	1

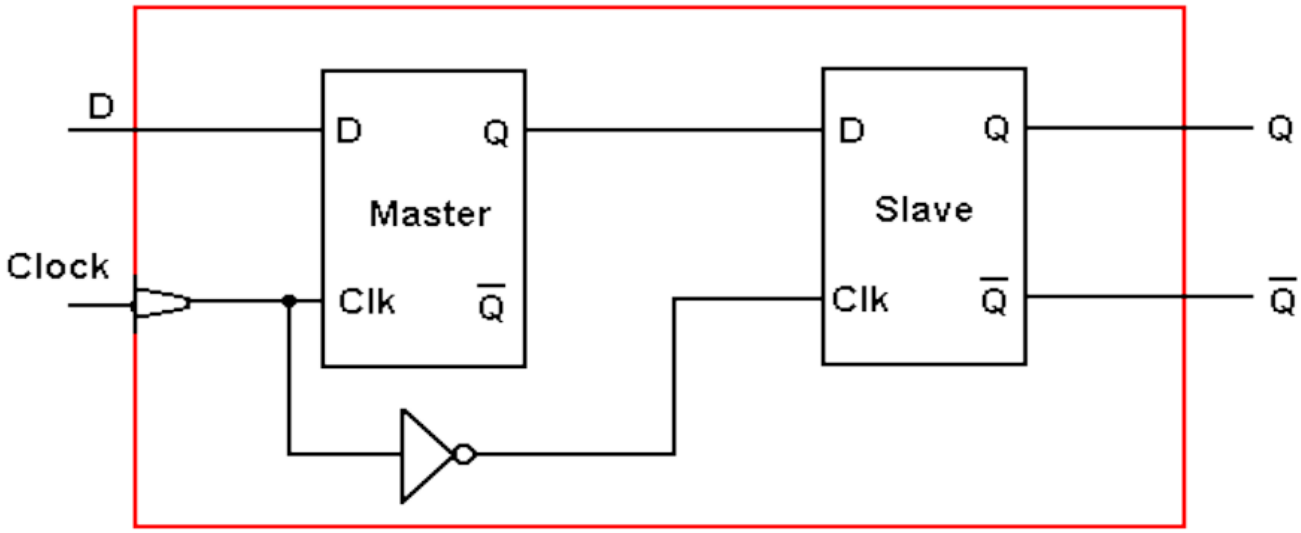
Edge Triggered Flip Flop:

- Output transitions occur at a specific level of clock pulse.
- When the pulse input level exceeds this threshold level, the inputs are locked out so that the flip flop is unresponsive to further changes in inputs until the clock pulse returns to 0 and another clock pulse occurs.
- Positive edge triggered flip flop
- Negative edge triggered flip flop



Master Slave Flip Flop

- It is constructed from 2 flip flops
- One circuit serves as a master other as a slave and overall circuit is referred to as a master slave flip flop
- It consists of a master flip flop, a slave flip flop and an inverter
- Both the flip flops are +ve level triggered, but inverter connected at the clock input of the slave flip flop forces it to trigger at the –ve level
- The output state of the master flip-flop is transferred as input to the slave flip flop



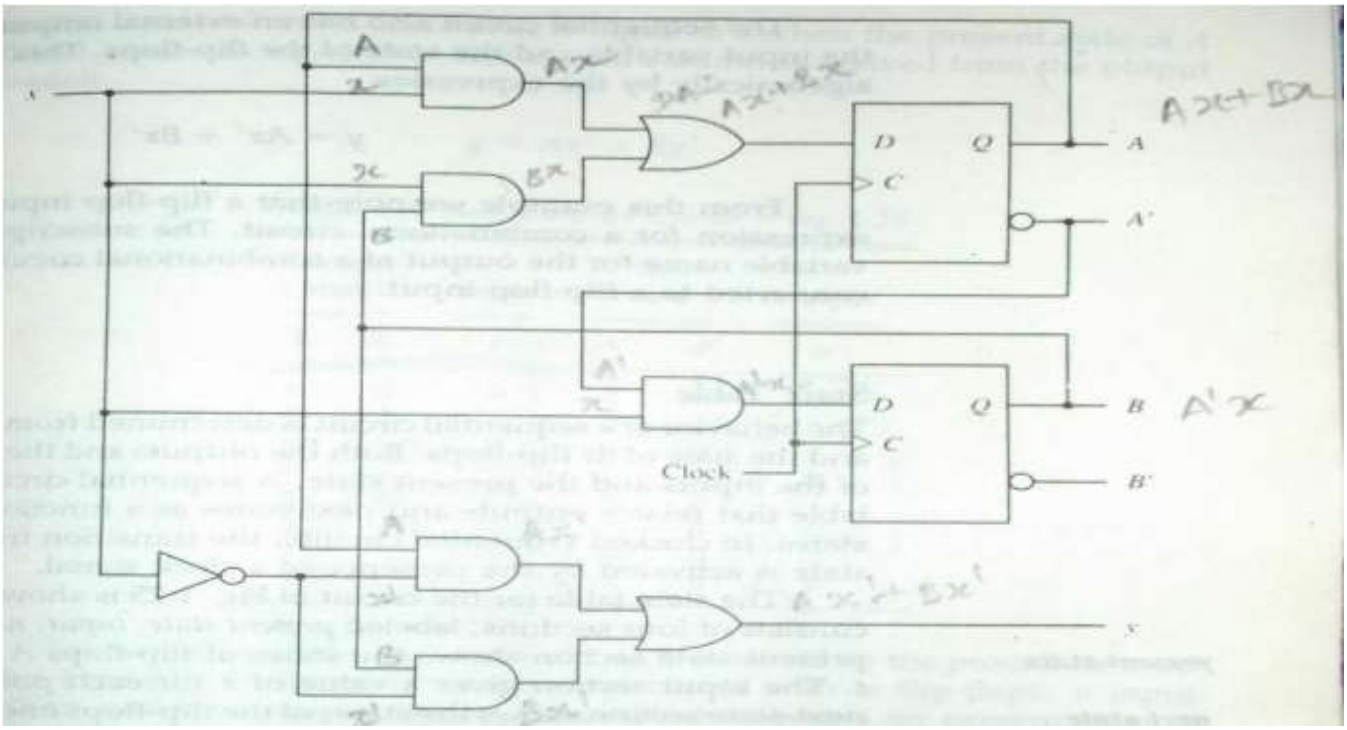
- Difference between the combinational circuits and sequential circuits are given below:

	Combinational	Sequential
1)	The outputs of the combinational circuit depend only on the present inputs.	The outputs of the sequential circuits depend on both present inputs and present state(previous output).
2)	The feedback path is not present in the combinational circuit.	The feedback path is present in the sequential circuits.
3)	In combinational circuits, memory elements are not required.	In the sequential circuit, memory elements play an important role and require.
4)	The clock signal is not required for combinational circuits.	The clock signal is required for sequential circuits.
5)	The combinational circuit is simple to design.	It is not simple to design a sequential circuit.

Sequential circuits:

- The sequential circuit is a special type of circuit that has a series of inputs and outputs.
- The outputs of the sequential circuits depend on both the combination of present inputs and previous outputs.
- The previous output is treated as the present state. So, the sequential circuit contains the combinational circuit and its memory storage elements.
- A sequential circuit doesn't need to always contain a combinational circuit. So, the sequential circuit can contain only the memory element.

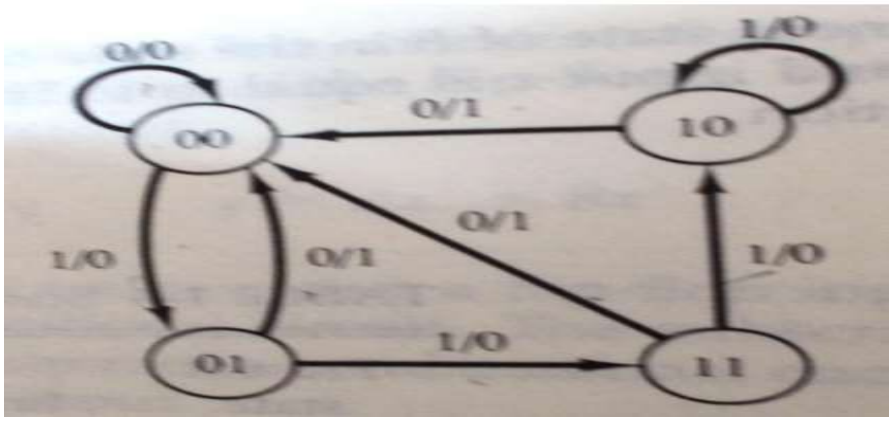
- It consists of 2 flip flops D_A and D_B. The external source x is applied to the logic gate, then the output of the logic gates is given as input to the flip flop, the output of flip flop is again given as an input to the logic gate
- We can frame input output equation with help of circuit diagram
- Input Equation:
  $D_A=Ax + Bx$
  $D_B=A'x$
- Output Equation:
  $Y=Ax'+Bx'$



State Table:

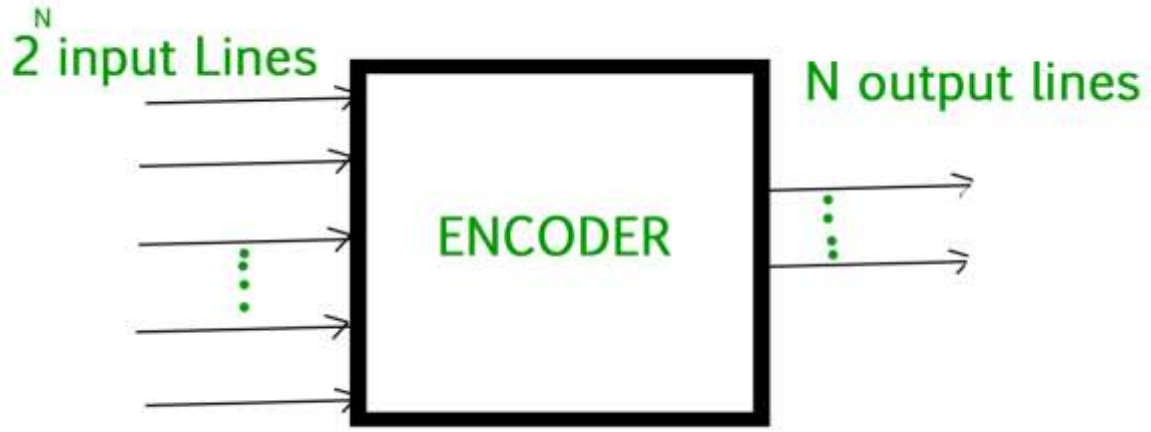
Present State		Input	Next State		Output
A	B	X	A	B	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

State Diagram:



Encoders and Decoders in Digital Logic

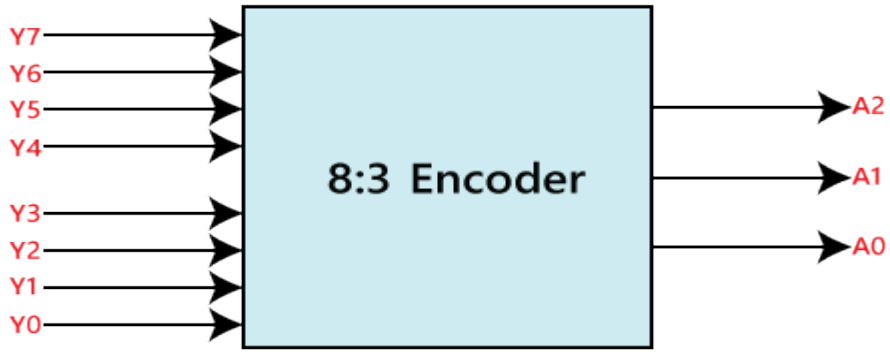
- **Encoders –**
  An encoder is a combinational circuit that converts binary information in the form of a 2^N input lines into N output lines, which represent N bit code for the input.
- For simple encoders, it is assumed that only one input line is active at a time. An **Encoder** is a combinational circuit that performs the reverse operation of Decoder.
- It has maximum of 2^n input lines and 'n' output lines. It will produce a binary code equivalent to the input, which is active High. Therefore, the encoder encodes 2^n input lines with 'n' bits.



8 to 3 line Encoder:

The 8 to 3 line Encoder is also known as **Octal to Binary Encoder**. In 8 to 3 line encoder, there is a total of eight inputs, i.e., Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, and Y_7 and three outputs, i.e., A_0, A1, and A_2. In 8-input lines, one input-line is set to true at a time to get the respective binary code in the output side. Below are the block diagram and the truth table of the 8 to 3 line encoder.

Block Diagram:



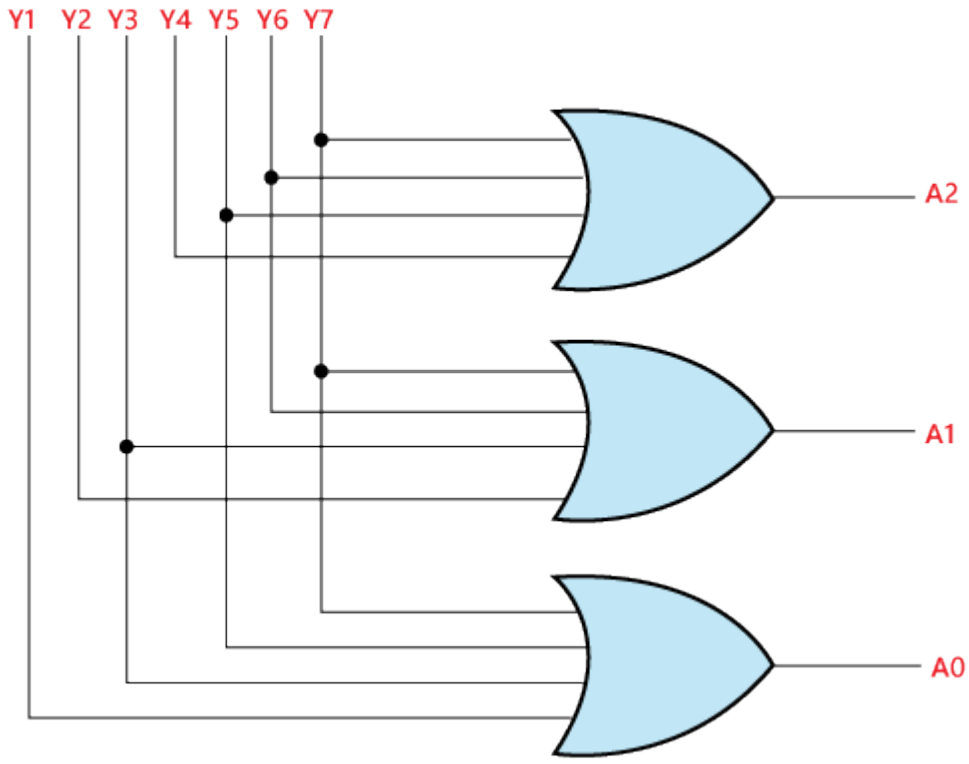
Truth Table:

INPUTS								OUTPUTS		
Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0	A_2	A_1	A_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

The logical expression of the term A0, A1, and A2 are as follows:

$A_2=Y_4+Y_5+Y_6+Y_7$
$A_1=Y_2+Y_3+Y_6+Y_7$
$A_0=Y_7+Y_5+Y_3+Y_1$

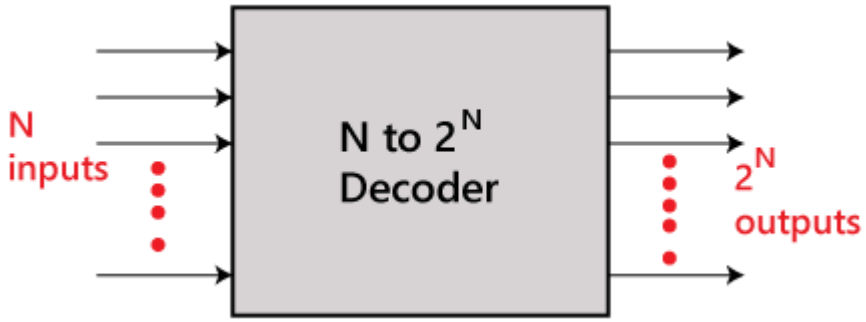
Logical circuit of the above expressions is given below:



Uses of Encoders:

1. These systems are very easy to use in all digital systems.
2. Encoders are used to convert a decimal number into the binary number. The objective is to perform a binary operation such as addition, subtraction, multiplication, etc.

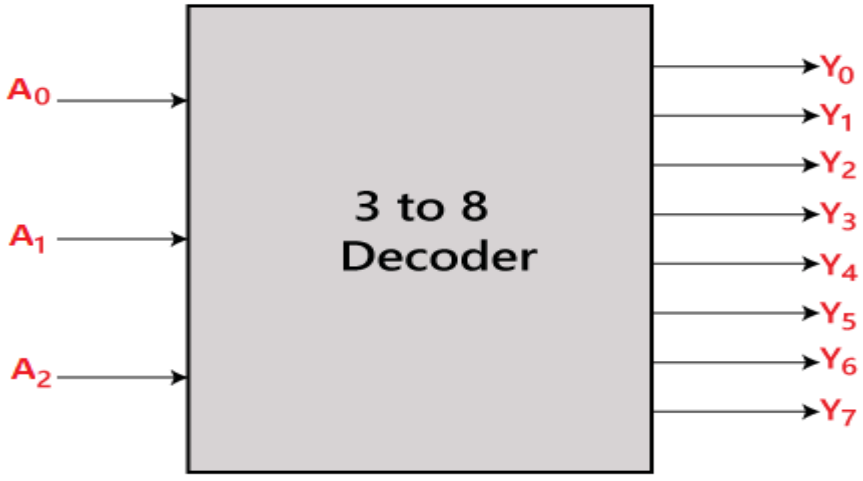
- **Decoder :**
  A decoder is also a combinational circuit as encoder but its operation is exactly reverse as that of the encoder.
- A decoder is a device that generates the original signal as output from the coded input signal and converts n lines of input into 2n lines of output. An AND gate can be used as the basic decoding element because it produces a high output only when all inputs are high.



3 to 8 line decoder:

- The 3 to 8 line decoder is also known as **Binary to Octal Decoder**.
- In a 3 to 8 line decoder, there is a total of eight outputs, i.e., Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, and Y_7 and three outputs, i.e., A_0, A1, and A_2.
- This circuit has an enable input 'E'. Just like 2 to 4 line decoder, when enable 'E' is set to 1, one of these four outputs will be 1.

The block diagram and the truth table of the 3 to 8 line encoder are given below.



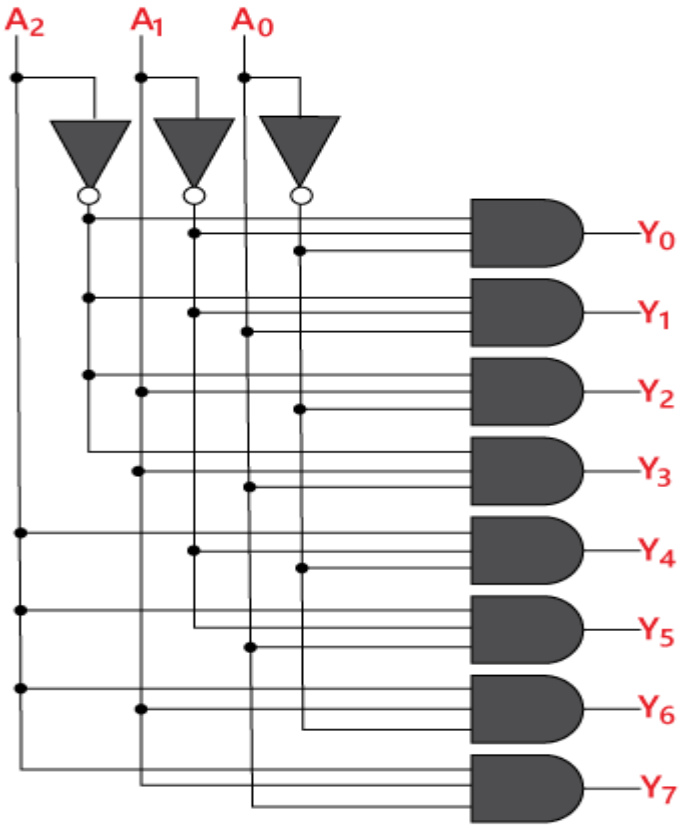
Truth table:

Inputs			Outputs							
A	B	C	D0	D1	D2	D3	D4	D5	D6	D7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

The logical expression of the term Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, and Y_7 is as follows:

$Y_0=A_0'.A_1'.A_2'$
$Y_1=A_0.A_1'.A_2'$
$Y_2=A_0'.A_1.A_2'$
$Y_3=A_0.A_1.A_2'$
$Y_4=A_0'.A_1'.A_2$
$Y_5=A_0.A_1'.A_2$
$Y_6=A_0'.A_1.A_2$
$Y_7=A_0.A_1.A_2$

Logical circuit of the above expressions is given below:



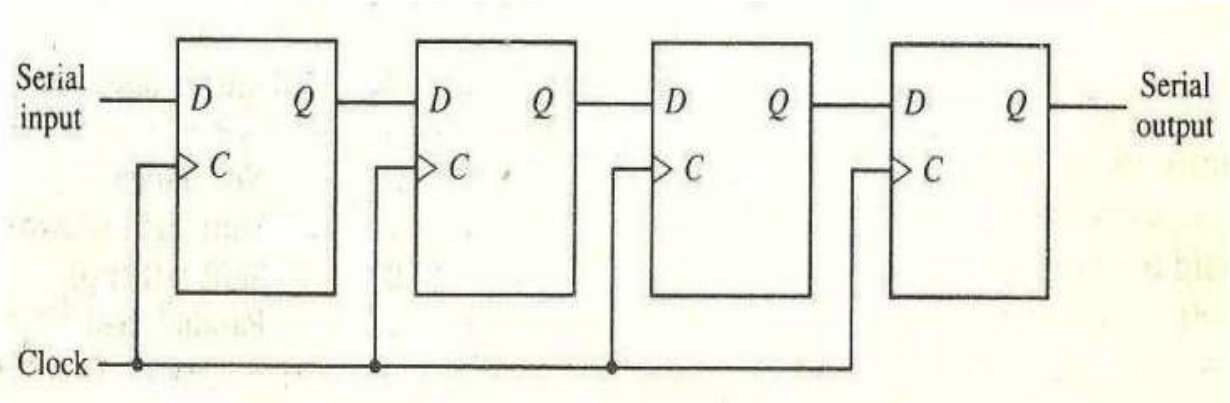
Difference between encoder and decoder :

S.No.	ENCODER	DECODER
1	Encoder circuit basically converts the applied information signal into a coded digital bit stream.	Decoder performs reverse operation and recovers the original information signal from the coded bits.
2	In case of encoder, the applied signal is the active signal input.	Decoder accepts coded binary data as its input.
3	The number of inputs accepted by an encoder is 2n.	The number of input accepted by decoder is only n inputs.
4	The output lines for an encoder is n.	The output lines of an decoder is 2n.
5	The encoder generates coded data bits as its output.	The decoder generates an active output signal in response to the coded data bits.
6	The operation performed is simple.	The operation performed is complex.

Shift Register

- A group of flip flops which is used to store multiple bits of data and the data is moved from one flip flop to another is known as **Shift Register**.
- The bits stored in registers shifted when the clock pulse is applied within and inside or outside the registers.
- To form an n-bit shift register, we have to connect n number of flip flops.
- So, the number of bits of the binary number is directly proportional to the number of flip flops.

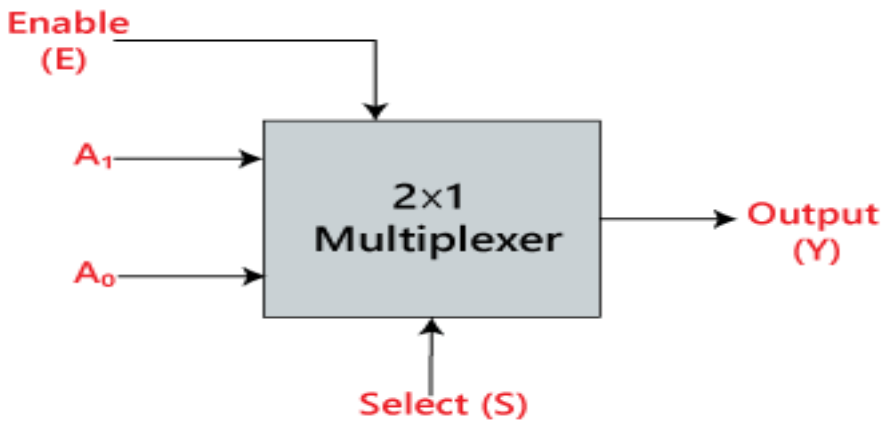
- The flip flops are connected in such a way that the first flip flop's output becomes the input of the other flip flop.
- A **Shift Register** can shift the bits either to the left or to the right. A **Shift Register**, which shifts the bit to the left, is known as **"Shift left register"**, and it shifts the bit to the right, known as **"Right left register"**.



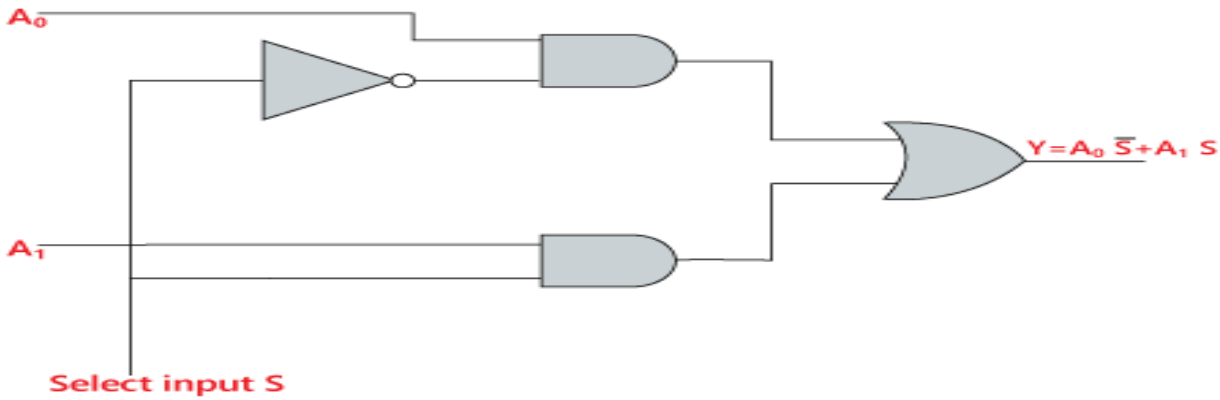
Multiplexers:

- A multiplexer is a combinational circuit that receives binary information from one of 2^n input data lines and directs it to a single output line
- A 2^n to 1 multiplexer has 2^n input data lines and n input selection lines
- A 4-1 line multiplexer has 4 data inputs (I_0, I_1, I_2, I_3) and selection input S_0, S_1
- In 2×1 multiplexer, there are only two inputs, i.e., A_0 and A_1, 1 selection line, i.e., S_0 and single outputs, i.e., Y.
- On the basis of the combination of inputs which are present at the selection line S^0, one of these 2 inputs will be connected to the output.
- Unlike encoder and decoder, there are n selection lines and 2^n input lines. So, there is a total of 2^N possible combinations of inputs. A multiplexer is also treated as **Mux**

2×1 Multiplexer:



- The logical expression of the term Y is as follows:
- $Y=S_0'.A_0+S_0.A_1$



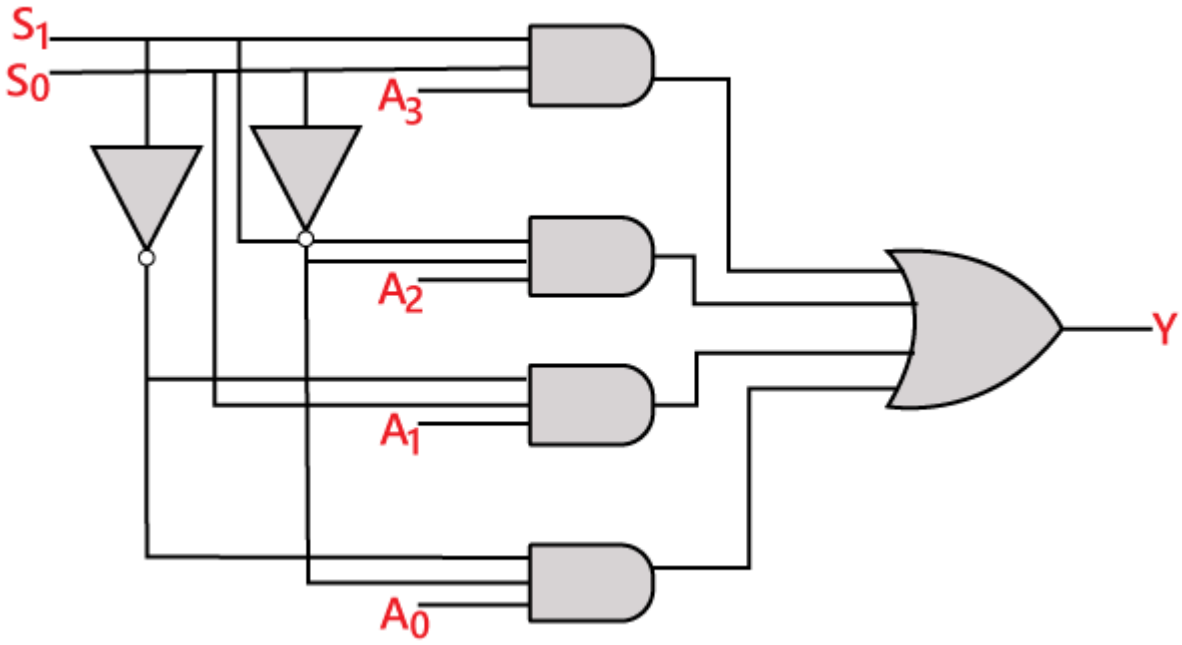
4×1 Multiplexer:

- In the 4×1 multiplexer, there is a total of four inputs, i.e., A_0, A_1, A_2, and A_3, 2 selection lines, i.e., S_0 and S_1 and single output, i.e., Y.
- On the basis of the combination of inputs that are present at the selection lines S^0 and S_1, one of these 4 inputs are connected to the output

INPUTS		Output
S_1	S_0	Y
0	0	A_0
0	1	A_1
1	0	A_2
1	1	A_3

- The logical expression of the term Y is as follows:
- $Y=S_1' S_0' A_0+S_1' S_0 A_1+S_1 S_0' A_2+S_1 S_0 A_3$

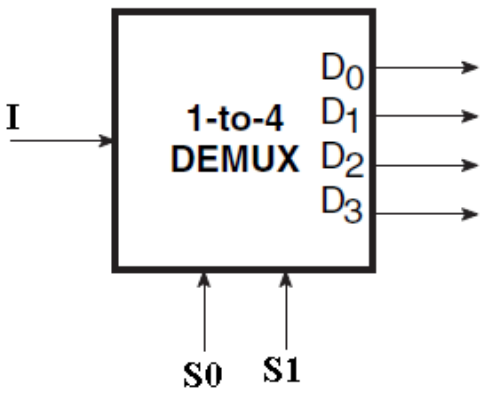
Logical circuit of the above expression is given below:



Demultiplexer:

A De-multiplexer (De-Mux) can be described as a combinational circuit that performs the reverse operation of a Multiplexer.

A De-multiplexer has a single input, 'n' selection lines and a maximum of 2^n outputs.

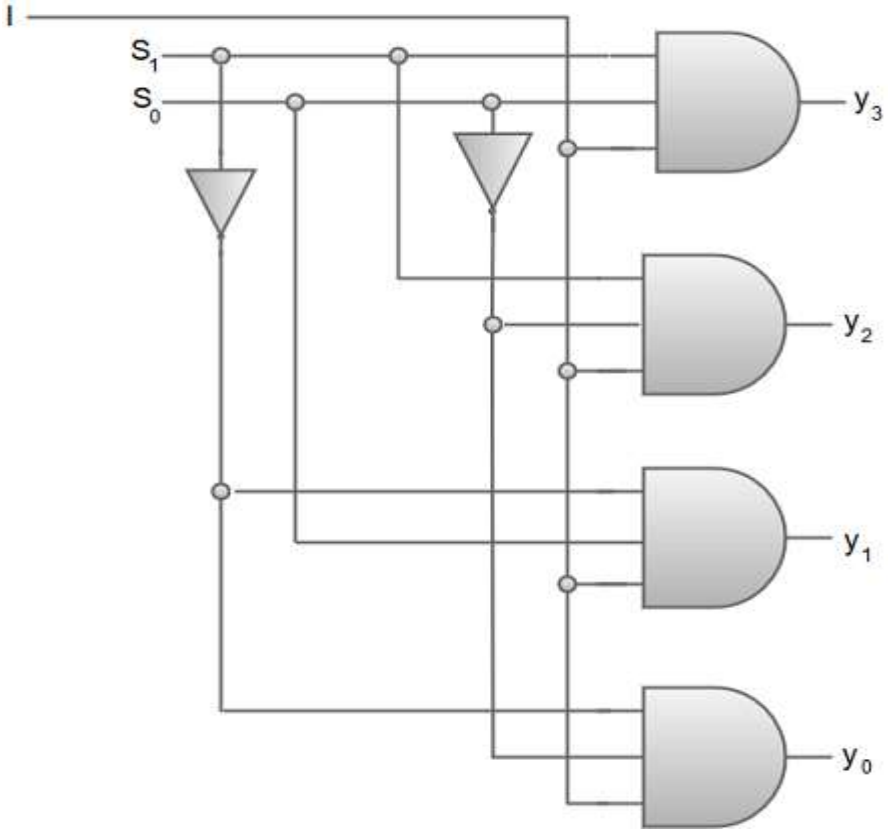


I	Select		O/P			
	S0	S1	D0	D1	D2	D3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

From the above function table, we can write the Boolean function for each output as:

y3 = S1S0 I, y2 = S1S0' I, y1 = S1' S0 I, y0 = S1'S0' I

The above equations can be implemented using inverters and three-input AND gates.



Counters:

- A special type of sequential circuit used to count the pulse is known as a counter, or a collection of flip flops where the clock signal is applied is known as counters.
- The counter is one of the widest applications of the flip flop. Based on the clock pulse, the output of the counter contains a predefined state.
- The number of the pulse can be counted using the output of the counter.

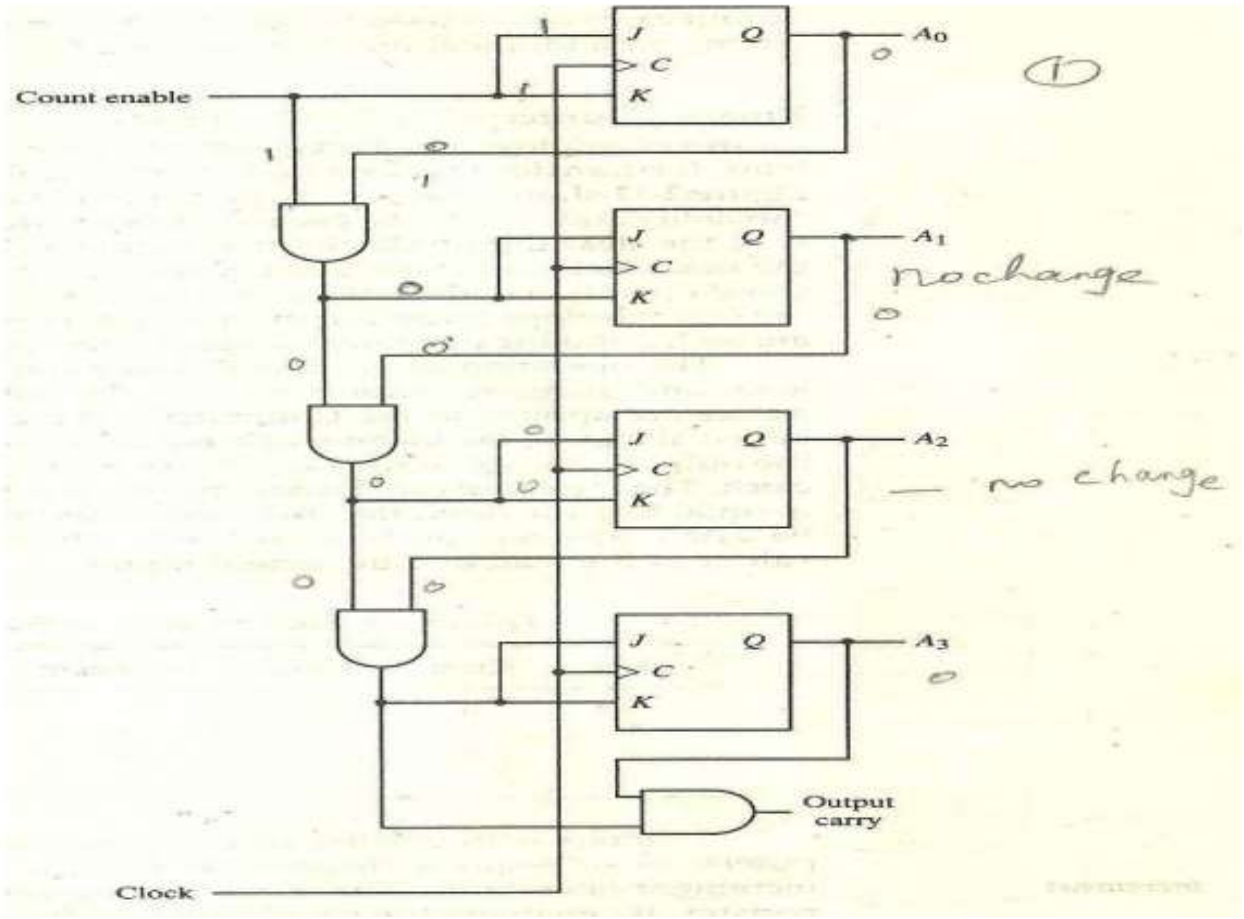
Synchronous counters

In the **Asynchronous counter**, the present counter's output passes to the input of the next counter. So, the counters are connected like a chain. The drawback of this system is that it creates the counting delay, and the propagation delay also occurs during the counting stage. The **synchronous counter** is designed to remove this drawback.

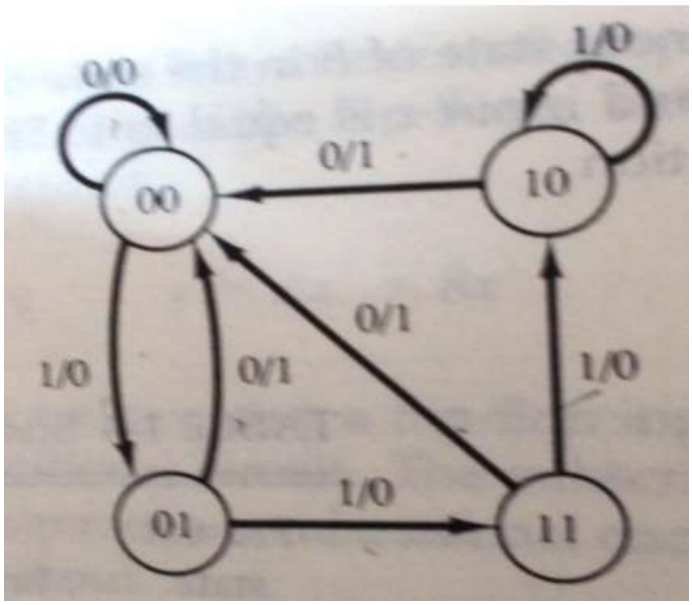
In the **synchronous counter**, the same clock pulse is passed to the clock input of all the flip flops. The clock signals produced by all the flip flops are the same as each other.

4 bit synchronous binary counter:

- The C inputs of all flip flops receive the common clock
- If the count enabled is 0, all J and K inputs are maintained at 0 and the output of counter does not change
- A_0 is complemented when counter is enabled and clock goes to positive transition
- Each of the other three flip flops are complemented when all previous least significant flip flop are equal to 1 and count is enabled



Present State				Next State			
A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0



Integrated Circuits

- An integrated circuit (IC) is manufactured using silicon material and mounted in a ceramic or plastic container (known as Chip).
- The basic components of an IC consist of electronic circuits for the digital gates. The various gates are interconnected inside an IC to form the required circuit.
- The following categories can broadly classify an Integrated Circuit (IC):

SSI (Small Scale Integration Devices)

- These types of devices contain several independent gates in a single package. The inputs and outputs of these gates are connected directly to the pins in the package.
- The number of logic gates are usually less than 10 and are limited by the number of pins available in the IC.

MSI (Medium Scale Integration Devices)

This type of devices has a complexity of approximately 10 to 200 gates in a single package. The basic components include decoders, adders, and registers.

LSI (Large Scale Integration Devices)

LSI devices contain about 200 to a few thousand gates in a single package. The basic components of an LSI device include digital systems, such as processors, memory chips, and programmable modules.

VLSI (Very Large Scale Integration Device)

This type of devices contains thousands of gates within a single package. The most common example of a VLSI device is a complex microcomputer chip.

TTL (Transistor-transistor Logic)

- The TTL technology was an upgraded version of a previous technology called as DTL (Diode-Transistor Logic).
- TTL came in existence when these diodes are replaced with transistors to improve the circuit operation.
- There are several variations of the TTL like high-speed TTL, low-power TTL, Schottky TTL, low-power Schottky TTL, and advanced Schottky TTL.

TTL Applications

- TTL is used as a switching device in driving lamps and relays.
- TTL is used in controller application for providing 0 to 5Vs.

ECL (Emitter-coupled Logic)

The ECL technology provides the highest-speed digital circuits in integrated form. An ECL circuit is used in supercomputers and signal processors where high speed is essential.

Features of ECL Family

- The logic gates continuously draw current even in the inactive state. Hence power consumption is more as compared to other logic families.
- ECL uses bipolar transistor logic where the transistors are not operated in the saturation region.

MOS (Metal-oxide semiconductor)

- The MOS (Metal-oxide semiconductor) is a unipolar transistor that depends on the flow of only one type of carrier, which may be electrons (n-channel) or holes (p-channel).
- MOS technology is generally categorized in two basic forms:
  1. A p-channel MOS is referred to as PMOS.
  2. An n-channel MOS is referred to as NMOS.

PMOS

- The operations performed by a PMOS logic family can be explained by considering a PMOS NAND gate.
- When a low logic is applied to either A or B, the transistor gets activated. This makes a connection between power supply and the output terminal.

NMOS

The structure of NMOS logic is similar to that of PMOS. However, instead of using PMOS transistors, here we will use NMOS transistors along with a pull-up resistor R.

Sample Questions:

1. Conversion from one number system to another
2. With steps and an example demonstrate normal subtraction, 1's complement, 2's complement subtraction
3. Explain various logic gates with logic diagram and truth table
4. Simplify equation using Boolean algebra
5. Problem based on K Map
6. What is half adder? Explain with circuit diagram and truth table.
7. What is full adder? Explain with circuit diagram and truth table.
8. Explain following flip flops with logic diagram and truth table:
   a. SR Flip Flop
   b. D Flip Flop
   c. JK Flip Flop
   d. T Flip Flop
9. Explain master slave flip flop
10. What is edge triggered flip flop? Explain Positive and Negative Edge Triggered
11. What are sequential circuits? Construct sequential circuit for:
    Input Equation:
    D_A=Ax + Bx
    D_B=A'x
    Output Equation:
    Y=Ax'+Bx'
    Write the state table and state diagram for the same.
12. Explain shift register with diagram
13. What are counters? Explain 4 bit synchronous counter with state table and logic diagram
14. What is IC's? How are they classified?
15. What is Multiplexer? Explain 4 X 1 MUX with diagram and truth table.
16. What is Decoder? Explain 3X8 decoder with truth table and diagram.