UNIT 7

SEQUENTIAL LOGIC

Introduction to Sequential Circuit:

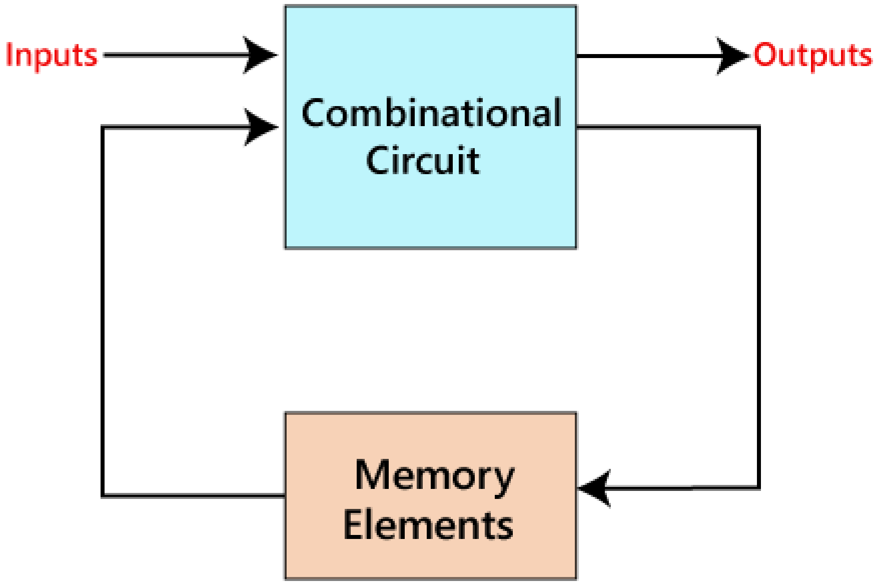


Fig: Synchronous/Sequential logic circuit.

- The sequential circuit is a type of circuit which output depends on present input as well as previous output.
- 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.

Types of Sequential Circuits – There are two types of sequential circuits:

- **Asynchronous sequential circuit** – These circuits do not use a clock signal but uses the pulses of the inputs.
- **Synchronous sequential circuit** – These circuits use clock signal and level inputs.

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

	Combinational Circuits	Sequential Circuits
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.

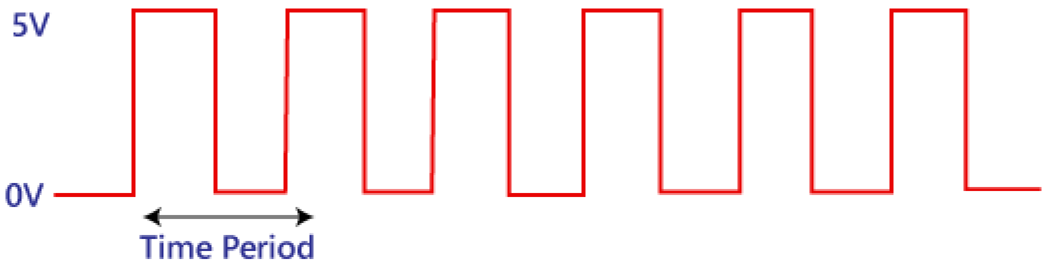
Difference between Flip-flop and Latch:

S. N	Flip-flop	Latch
1	It is an edge triggered device.	It is a level triggered device.
2	They are classified into asynchronous or synchronous flipflops.	There is no such classification in latches.
3	It forms the building blocks of many sequential circuits like counters.	These can be used for the designing of sequential circuits but are not generally preferred.
4	Flip-flop always have a clock signal	latches doesn’t have a clock signal
5	Flip-flop can be built from Latches	Latches can be built from gates
6	ex: D Flip-flop, JK Flip-flop	ex: SR Latch, D Latch

Clock Signal and Triggering:

Clock signal:

- A clock signal is a periodic signal in which ON time and OFF time need not be the same.
- When ON time and OFF time of the clock signal are the same, a square wave is used to represent the clock signal.
- Below is a diagram which represents the clock signal:



- A clock signal is considered as the square wave.
- Sometimes, the signal stays at logic, either high 5V or low 0V, to an equal amount of time.
- It repeats with a certain time period, which will be equal to twice the 'ON time' or 'OFF time'.

Types of Triggering:

- The switching of the circuit is known as the triggering.

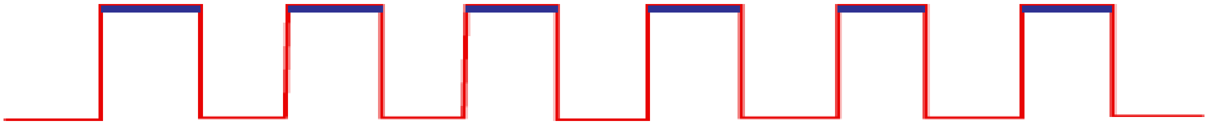
These are two types of triggering in sequential circuits: Level Triggering and Edge Triggering.

1. **Level triggering:**

- In the sequential circuit, if the output changes during the high voltage period or low voltage period, it is called level triggering.
- In other words, the output changes during either high voltage or low voltage period- not during the edges like in edge triggering.

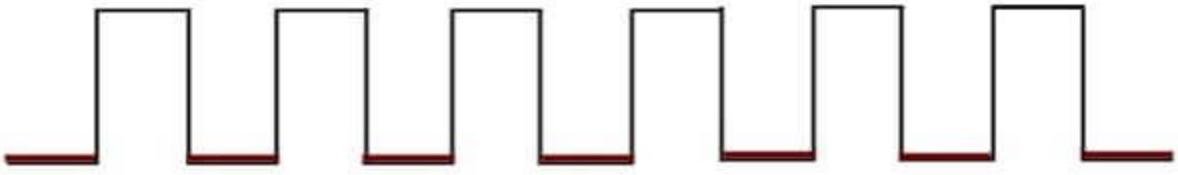
Positive level triggering:

- In a positive level triggering, the signal with Logic High occurs.
- So, in this triggering, the circuit is operated with such type of clock signal.
- Below is the diagram of positive level triggering:



Negative level triggering:

- In negative level triggering, the signal with Logic Low occurs.
- So, in this triggering, the circuit is operated with such type of clock signal.
- Below is the diagram of Negative level triggering:



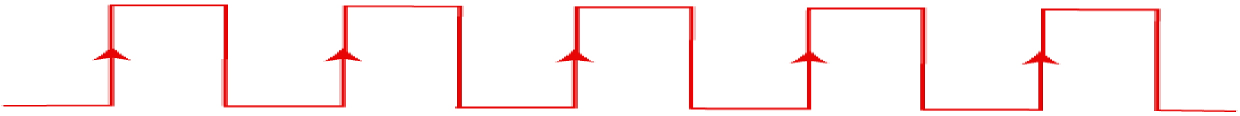
2. Edge triggering:

- In a sequential circuit, if the output changes when the signal transits from a high level to a low level or from a low level to a high level, we call it edge triggering.
- Here, the edge that changes the voltage from low level to the high level is called rising edge (positive edge).
- And, the edge that changes the voltage from high level to the low level is called falling edge (negative edge).

Based on the transitions of the clock signal, there are the following types of edge triggering:

Positive edge triggering:

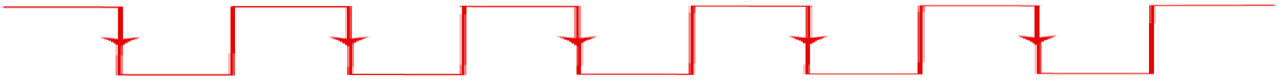
- The transition from Logic Low to Logic High occurs in the clock signal of positive edge triggering.
- So, in positive edge triggering, the circuit is operated with such type of clock signal. The diagram of positive edge triggering is given below.



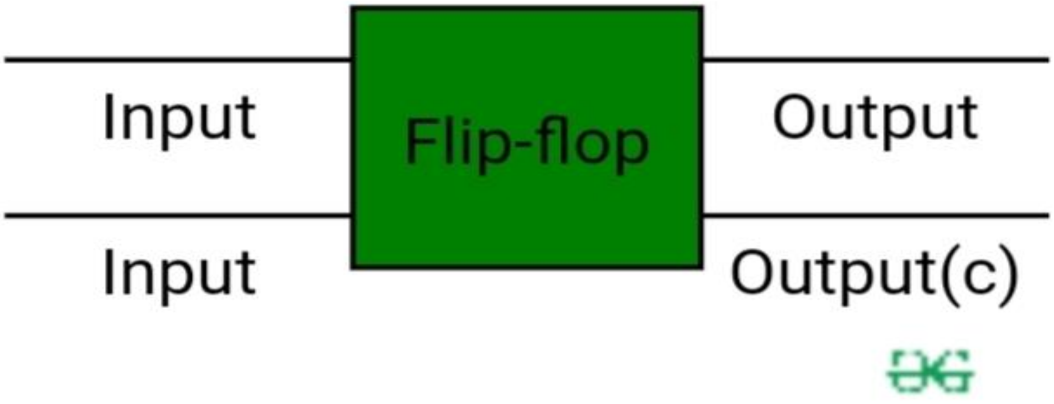
Negative edge triggering:

- The transition from Logic High to Logic low occurs in the clock signal of negative edge triggering. So, in negative edge triggering, the circuit is operated with such type of clock signal.

The diagram of negative edge triggering is given below.



Basics of Flip Flop:

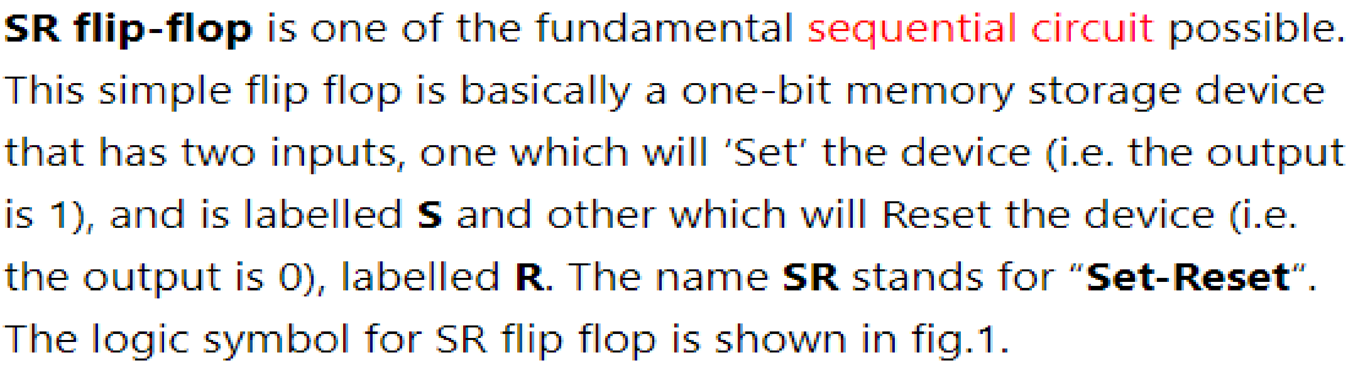


- A circuit that has two stable states is treated as a flip flop. It can store one bit data.
- These stable states are used to store binary data that can be changed by applying varying inputs.
- The flip flops are the fundamental building blocks of the digital system.
- Flip flops and latches are examples of data storage elements.
- In the sequential logical circuit, the flip flop is the basic storage element.
- The latches and flip flops are the basic storage elements but different in working.

There are the following types of flip flops:

- SR Flip-Flop
- D Flip-Flop
- JK Flip-Flop
- T Flip-Flop

1. SR Flip-Flop:



SR flip-flop is one of the fundamental sequential circuit possible. This simple flip flop is basically a one-bit memory storage device that has two inputs, one which will 'Set' the device (i.e. the output is 1), and is labelled **S** and other which will Reset the device (i.e. the output is 0), labelled **R**. The name **SR** stands for "**Set-Reset**". The logic symbol for SR flip flop is shown in fig.1.

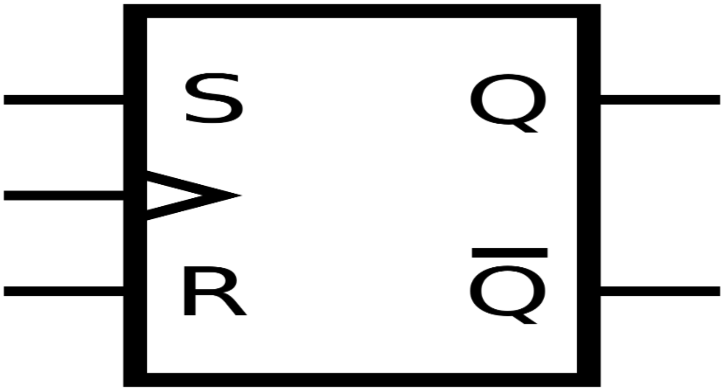


Fig 1: Block Diagram of SR flip flop

SR NAND flip flop

The circuit diagram of NAND SR flip flop is shown in fig.2.

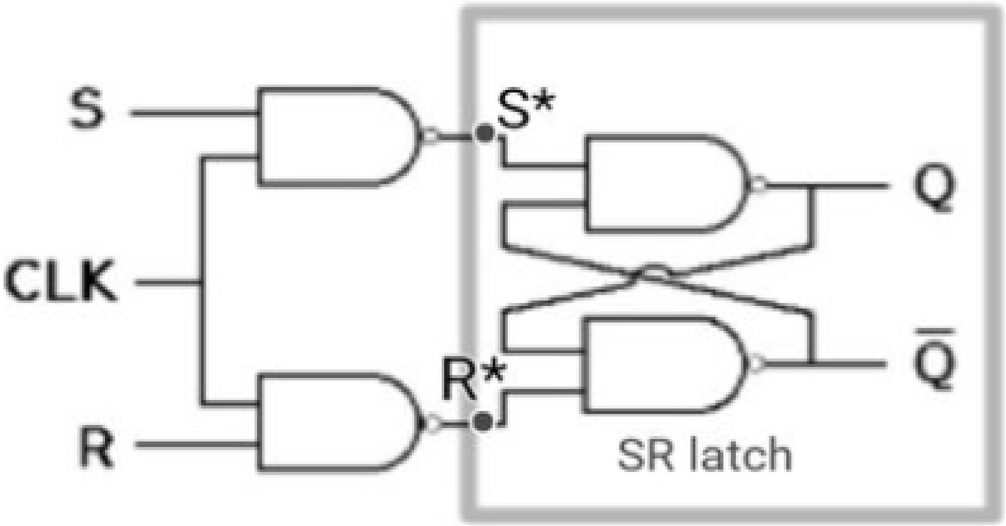


Fig. 2 SR NAND flip-flop

Working of SR NAND Flip-Flop:

It is clear from the fig.2 that

$$S^{*} = \overline{(S\ .\ CLK)} = \overline{S} + \overline{CLK}$$

and

$$R^{*} = \overline{(R\ .\ CLK)} = \overline{R} + \overline{CLK}$$

Case 1: Now if CLK is 0 then S*=1 and R*=1 and here S and R will be treated as don't care conditions, then we get Q and $\overline{Q}$

in memory state i.e. holding previous values.

Case 2: if CLK=1 then S*= $\overline{S}$

and R*= $\overline{R}$, now there will be 4 more cases depending upon the values of S and R.

Case 2(a): S=0 and R= 0 then S* and R* both becomes 1 and we get outputs Q and $\overline{Q}$

holding memory state.

Case 2(b): S=0 and R=1 then S*=1 and R*= 0 then we get Q= 0 and $\overline{Q}$

=1, we get both outputs as complement of each other.

Case 2(c): S=1 and R=0 then S*=0 and R*=1 them we get Q= 1 and $\overline{Q}$

=0.

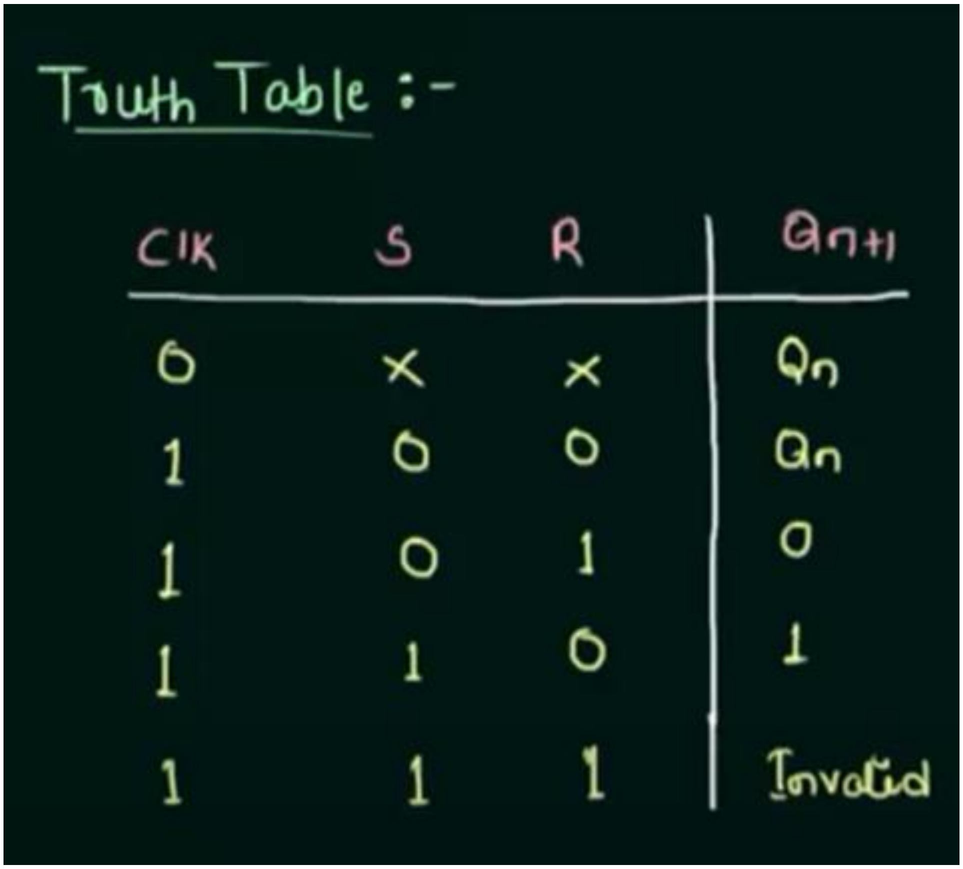
Case 2(d): S=1 and R=1 then S*=0 and R*=0 then we get Q and $\overline{Q}$

in the invalid state i.e. not used condition.

The truth table of SR NAND flip flop is given below.

CLK	S	R	Q	$\overline{Q}$
0	×	×	Memory state	
1	0	0	Memory state	
1	0	1	0	1
1	1	0	1	0
1	1	1	Invalid	

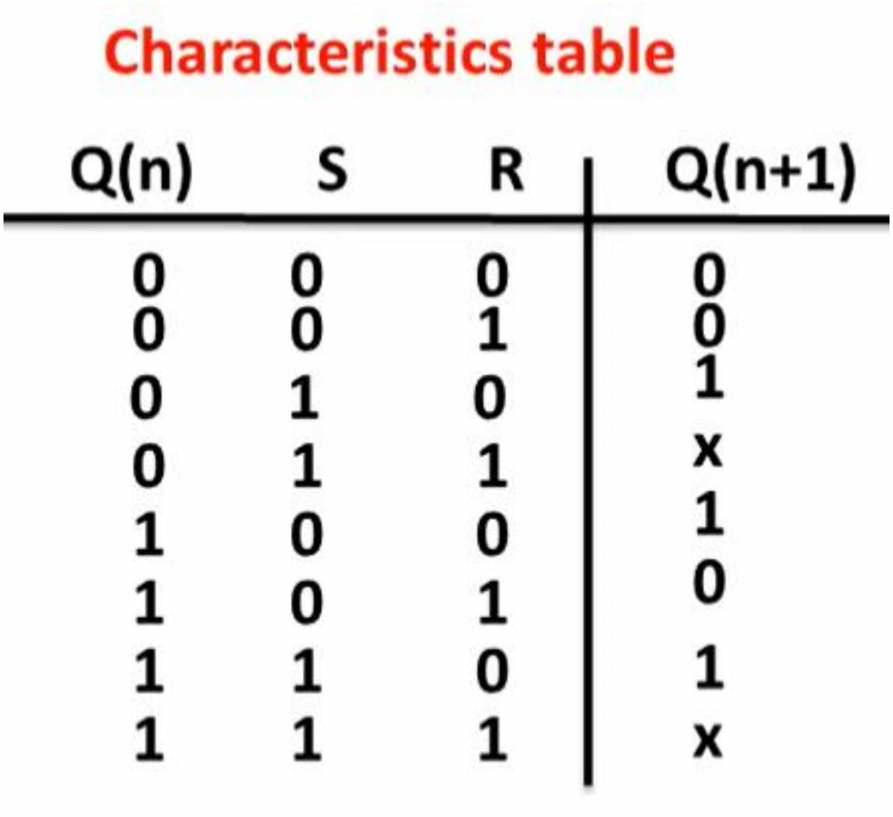
Note: × is the don't care condition.



Truth Table :-

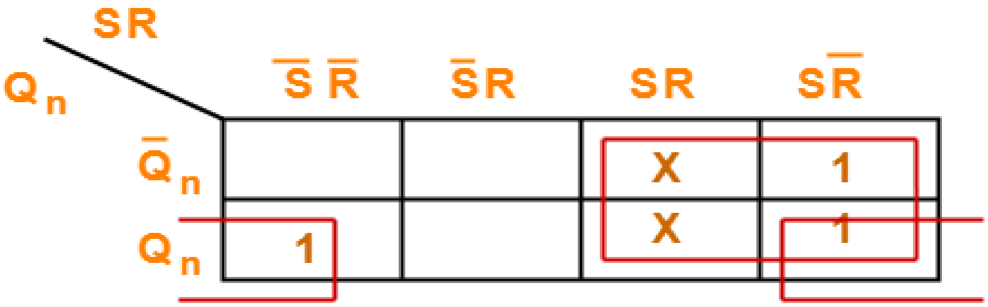
Clk	S	R	Q_{n+1}
0	×	×	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	Invalid

Characteristics Table of SR flip-Flop:



Characteristics table

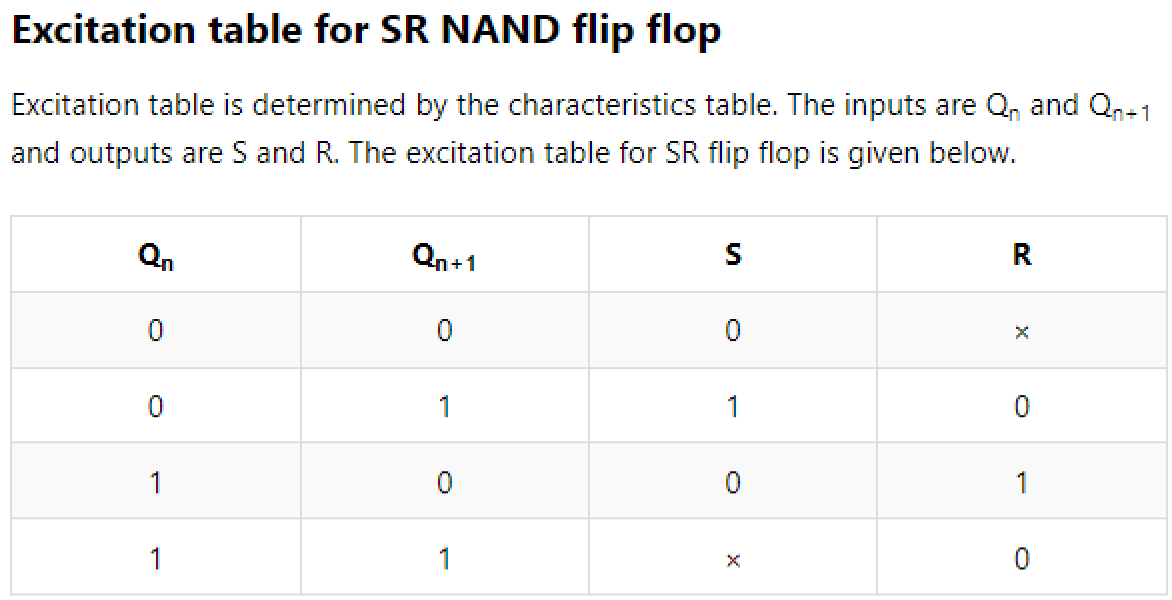
Q(n)	S	R	Q(n+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	x
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	x



	$\bar{S}\bar{R}$	$\bar{S}R$	SR	$S\bar{R}$
$\bar{Q}_n$			X	1
Q_n	1		X	1

K Map

$$Qn+1=S+ Qn. R'$$



Excitation table for SR NAND flip flop

Excitation table is determined by the characteristics table. The inputs are Q_n and Q_{n+1} and outputs are S and R. The excitation table for SR flip flop is given below.

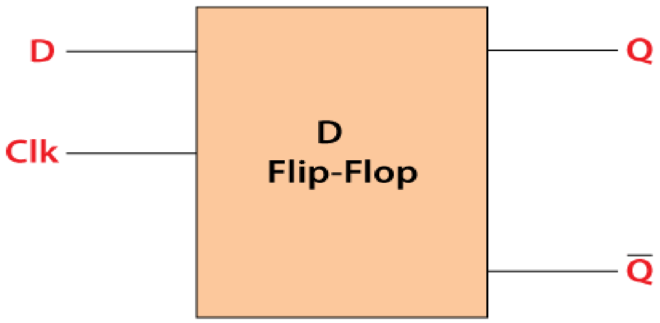
Q_n	Q_{n+1}	S	R
0	0	0	×
0	1	1	0
1	0	0	1
1	1	×	0

Similarly, we have to find the Boolean expression for S and R using K-map.

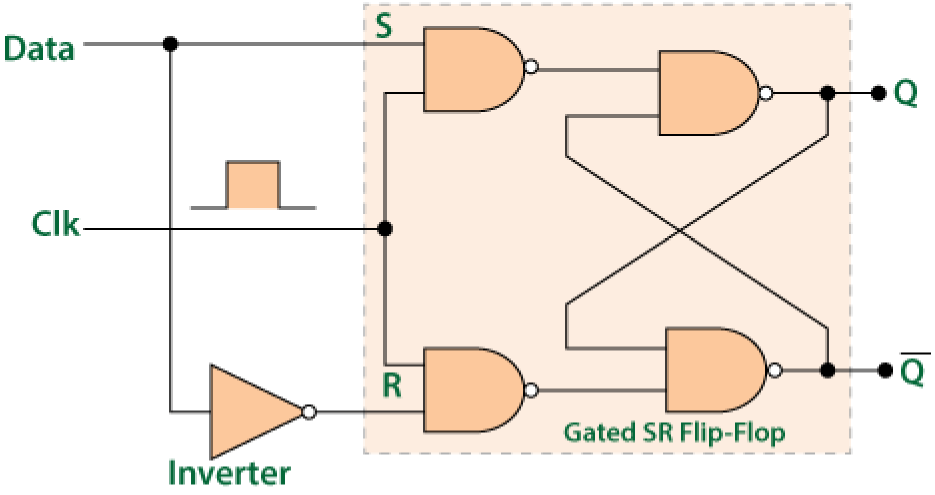
2. D Flip-Flop (Delay/Data Flip-Flop):

- In **SR Flip-Flop,** the undefined input condition of SET = "0" and RESET = "0" is forbidden. It is the drawback of the SR flip flop.
- We need an **inverter** to prevent this from happening.
- We connect the inverter between the Set and Reset inputs for producing another type of flip flop circuit called **D flipflop**
- A D (or Delay) Flip Flop is a digital electronic circuit used to delay the change of state of its output signal until the next rising edge of a clock timing input signal occurs.
- The Delay flip-flop is designed using a gated SR flip-flop

Block Diagram

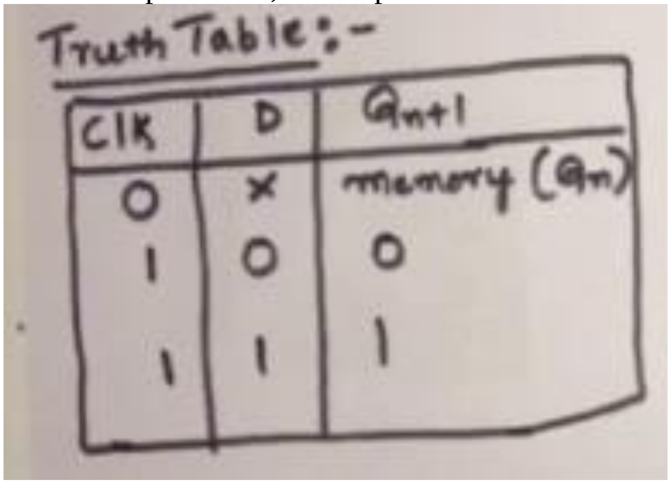


Circuit Diagram



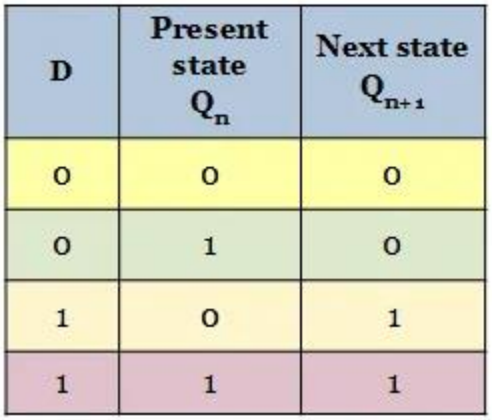
Operation or Working of D Flip-Flop:

- In D flip flop, the single input "D" is referred to as the "Data" input.
- When the data input is set to 1, the flip flop would be set, and when it is set to 0, the flip flop would change and become reset.
- However, this would be pointless since the output of the flip flop would always change on every pulse applied to this data input.
- The "CLOCK" or "ENABLE" input is used to avoid this for isolating the data input from the flip flop's latching circuitry.
- When the clock input is set to true, the D input condition is only copied to the output Q.
- This forms the basis of another sequential device referred to as D Flip Flop.
- When the clock input is set to 1, the "set" and "reset" inputs of the flip-flop are both set to 1.
- So, it will not change the state and store the data present on its output before the clock transition occurred. In simple words, the output is "latched" at either 0 or 1.

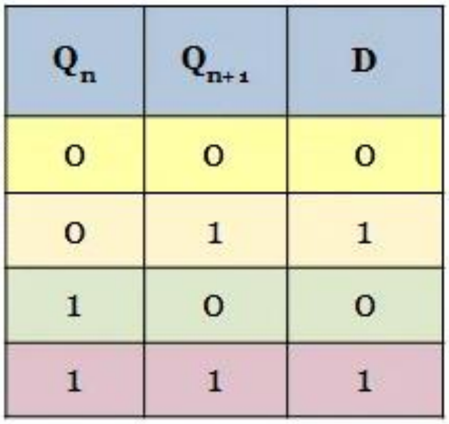


Truth Table:-

Clk	D	Q_{n+1}
0	x	memory (Q_n)
1	0	0
1	1	1

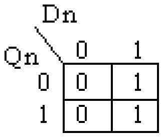


D	Present state Q_n	Next state Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1



Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Fig: Characteristics Table **Fig: Excitation Table**



Qn \ Dn	0	1
0	0	1
1	0	1

K-Map

$$\boxed{Q_{n+1} = D_n}$$

Characteristic Equation

Need to find K-map for D flip flop from Excitation Table above

3. JK Flip-Flop:

- The JK flip flop is basically a gated SR flip-flop with the addition of a clock input circuitry that prevents the illegal or invalid output condition that can occur when both inputs S and R are equal to logic level "1".
- Due to this additional clocked input, a JK flip-flop has four possible input combinations, "logic 1", "logic 0", "no change" and "toggle".
- The letters JK chosen by its inventor Jack Kilby to distinguish the flip-flop design from other types.

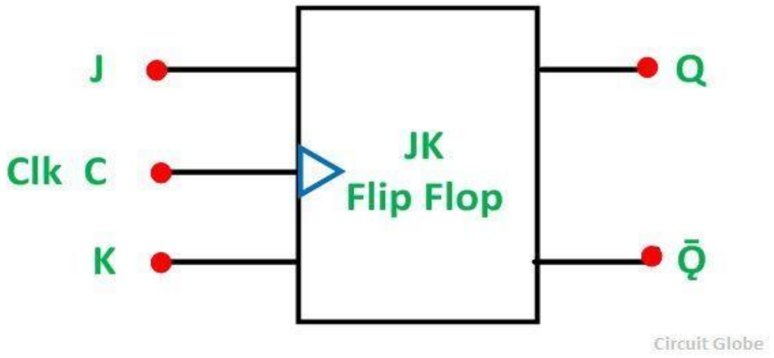


Fig: Block Diagram

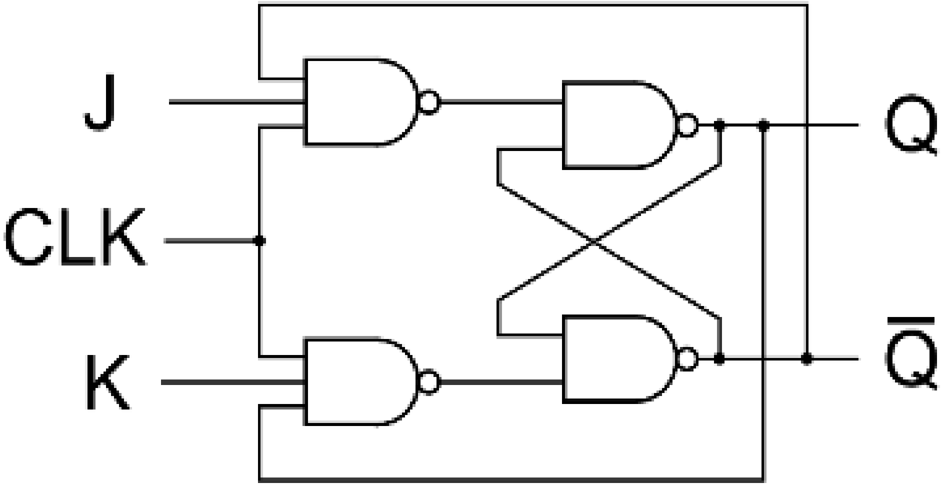


Fig: Circuit Diagram of JK Flip-Flop

<u>Operation/Working of JK Flip-Flop:</u>

<u>Case 1 : J = K = 0</u>

- For this input condition, irrespective of the other inputs for NAND gates A and B, = 1 and = 1.
- For present state outputs, Q = 1 and = 0, the next state outputs are Q+1 = 1, = 0.
- There is no change in the output.
- For Q = 0 and = 1, the next state outputs are Q+1 = 0, = 1.
- Thus there is NO CHANGE in the next state output.

<u>Case 2 : J = 0, K = 1</u>

- For this input condition, irrespective of other input for NAND gate A, the output produced is = 1. Let the present state output be Q = 0 and = 1.
- The inputs for NAND gate B are K = 1 and Q = 0, which produces = 1.
- Now the inputs for NAND gate C are = 1, = 1 and the next state output produced is Q+1 = 0. For NAND gate D, the inputs are = 1, Q = 0 and the prouced output is = 1.
- Thus, for inputs J = 0, K = 1 and present state inputs Q = 0 and = 1, the obtained next state outputs are Q+1 = 0 and = 1, so there is NO CHANGE is the state of JK flip flop.
- Let the present state inputs be Q = 1 and = 0.
- For J = 0, K = 1, the output produced by NAND gates A and B are = 1 and = 0.
- For this SR input value, when you look at the truth table of SR flip flop, the flip flop will RESET its state.

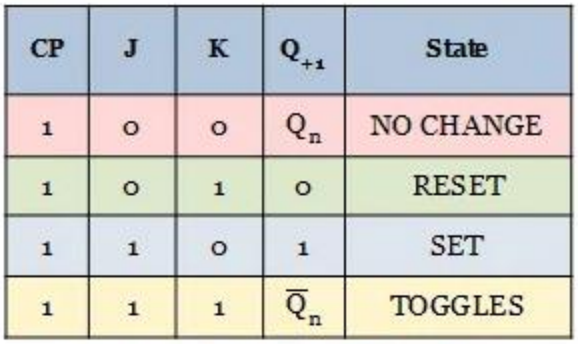
<u>Case 3 : J = 1, K = 0</u>

- In this case, the output of NAND gate B is = 1, irrespective of its other input.
- Now, let us consider the present state be Q = 0 and = 1.
- The inputs of NAND gate A are J = 1 and = 1, the output thus produced is = 0.
- For the SR input values, = 0 and = 1, when you look at the truth table of SR Flip lop, the flip flop will SET.
- Now, for the present state values Q = 1 and = 0, the outputs of NAND gate A and B are = 1 and = 1.
- For this SR input values(truth table of SR flip flop), the state of the flip flop has NO CHANGE.

<u>Case 4 : J = K = 1</u>

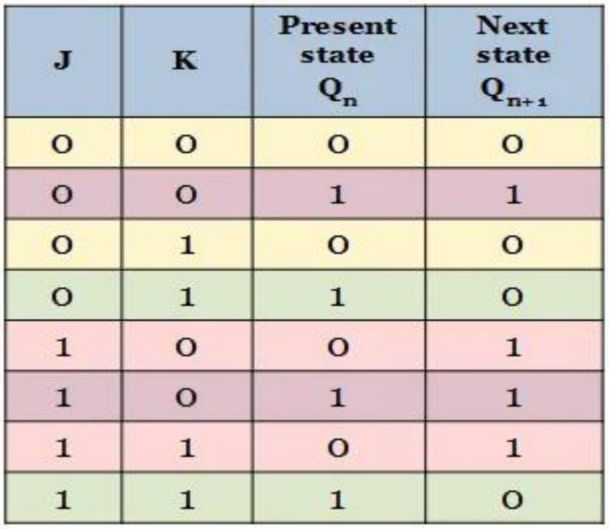
- For the present state inputs Q = 0 and = 1, the NAND gate outputs are A and B are = 0 and = 1.
- From the truth table of SR flip flop, it can be observed that, for = 0 and = 1, the flip flop will SET its state.
- Now, for the present state inputs, Q = 1 and = 0, the gate outputs of A and B are = 1 and = 0.
- From the truth table of SR flip flop, for the obtained SR inputs, the flip flop will RESET its state.
- For J = K = 1, the flip flop continuously changes its state from SET to RESET.

- It means, the flip flop toggles the flip flop output.
- As long as the input is J = K = 1 and for high clock pulse, the flip flop output will toggle.
- This leads to uncertainty in determining the output Q of the flip flop.
- This problem is referred to as race-around condition.
- To overcome this problem, Master-slave configuration of JK flip flop is developed.

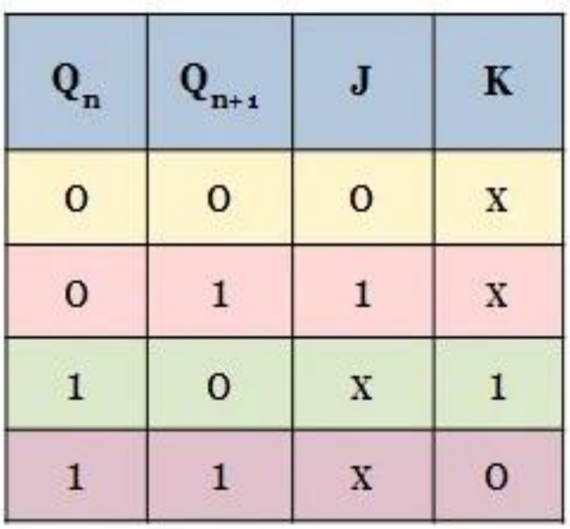


CP	J	K	Q_{+1}	State
1	0	0	Q_n	NO CHANGE
1	0	1	0	RESET
1	1	0	1	SET
1	1	1	$\overline{Q}_n$	TOGGLES

Simplified Truth Table of JK flip flop

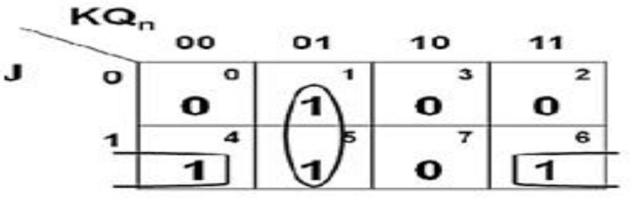


J	K	Present state Q_n	Next state Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0



Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Fig: Characteristics Table **Fig: Excitation Table**



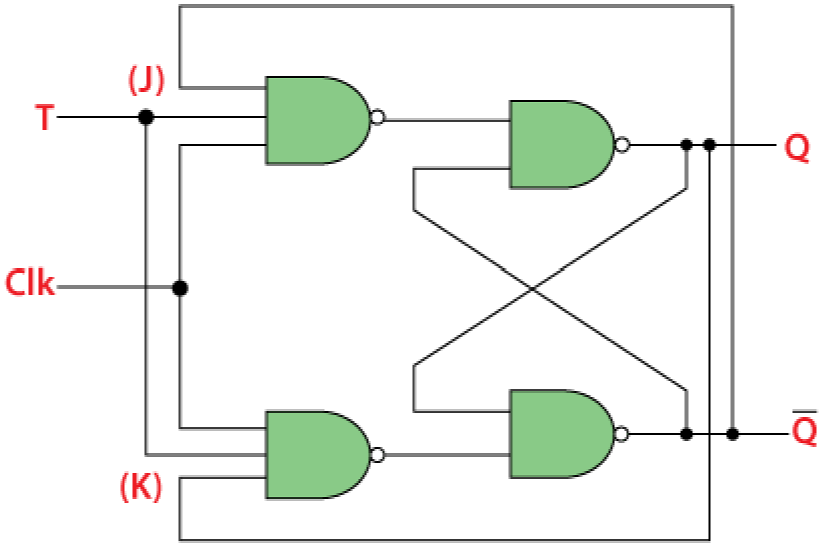
J \ KQ_n	00	01	10	11
0	0	1	0	0
1	1	1	0	1

$Q_{n+1} = JQ_n' + K'Q_n$

4. T-Flip-Flop:

- T-flip-flop is known as toggle flip-flop. The t-flip-flop is the modification of J-K flip-flop.
- T-flip-flop is obtained from J-K flip-flop by connecting both J and K together.

Circuit Diagram



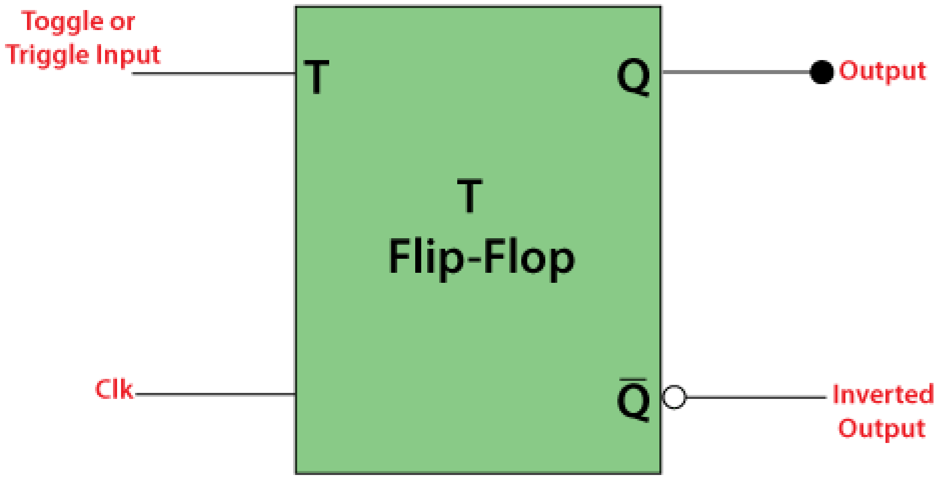
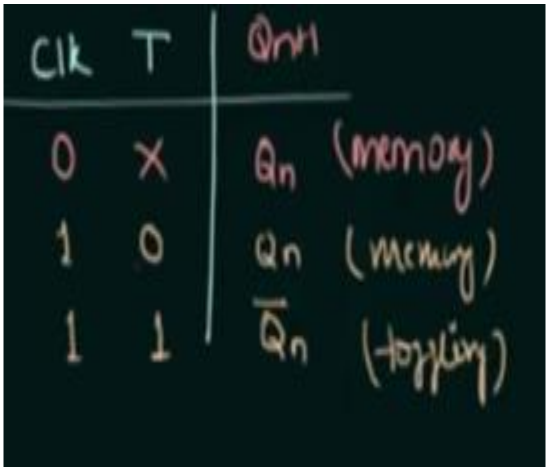


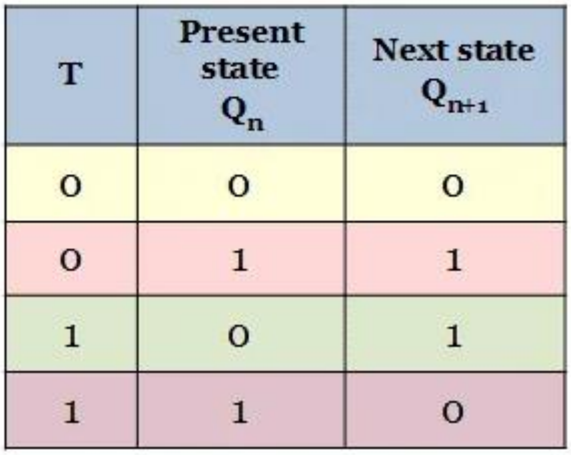
Fig: Block Diagram

Operation/working:

- The next sate of the T flip flop is similar to the current state when the T input is set to false or 0.
  - If toggle input is set to 0 and the present state is also 0, the next state will be 0.
  - If toggle input is set to 0 and the present state is 1, the next state will be 1.
- The next state of the flip flop is opposite to the current state when the toggle input is set to 1.
  - If toggle input is set to 1 and the present state is 0, the next state will be 1.
  - If toggle input is set to 1 and the present state is 1, the next state will be 0.
- The "T Flip Flop" is toggled when the set and reset inputs alternatively changed by the incoming trigger.
- The "T Flip Flop" requires two triggers to complete a full cycle of the output waveform.
- The frequency of the output produced by the "T Flip Flop" is half of the input frequency.
- The "T Flip Flop" works as the "Frequency Divider Circuit."
- In "T Flip Flop", the state at an applied trigger pulse is defined only when the previous state is defined.
- It is the main drawback of the "T Flip Flop".



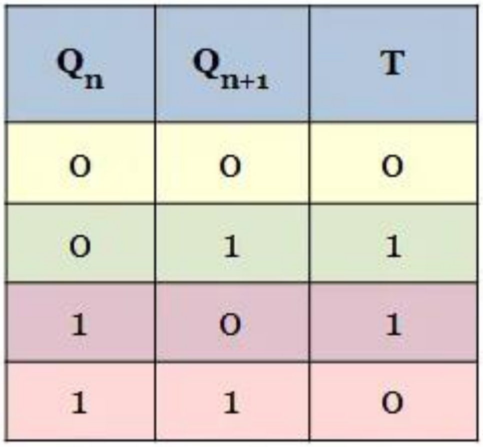
Clk	T	Q_{n+1}
0	x	Q_n (memory)
1	0	Q_n (memory)
1	1	$\overline{Q}_n$ (toggling)



T	Present state Q_n	Next state Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

Fig: Truth Table

Fig: Characteristics Table



Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

Fig: Excitation Table

Master-Slave JK Flip-Flop:

- The master-slave flip-flop is basically a combination of two J-K flipflops connected together in a series configuration. out of these one-acts as a master other as slave.
- The output from the master flip-flop is connected to the two input of the slave flip-flop whose output is feedback to the input of the master flip-flop.

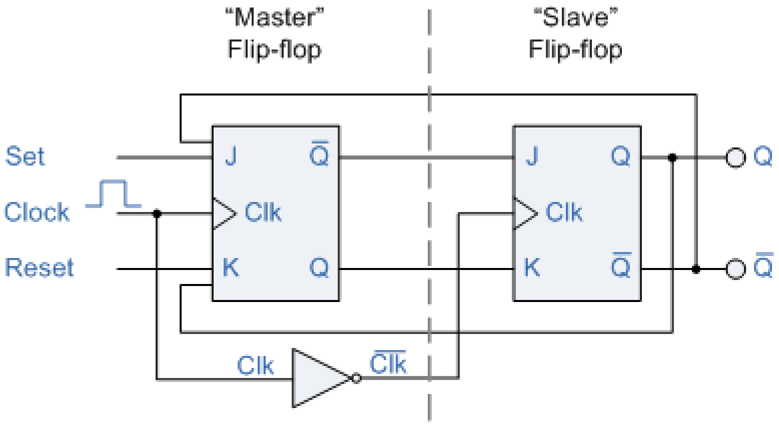


Fig: Block Diagram

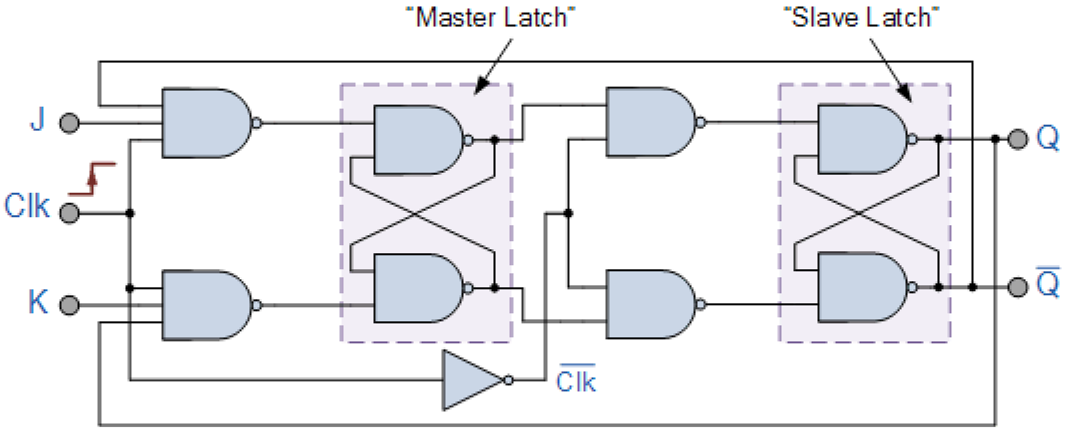


Fig: Circuit Diagram

A working condition of the master-slave flip-flop.

1. When the clock plus go to one the slave is isolated J and K may affect the state of the system. The slave flip-flop is isolated unit the clock plus goes to zero. When the clock plus goes to zero information is passed from master to the slave then, the output is obtained.
2. Firstly, master flip-flop is a positively level trigger so the master responded before the slave.
3. If J=0, K=0 the flip-flop is disabled and Q remain constant.
4. When J=0 and K=1 the high Q-output of master goes to the K-input of slave the positive transection of the clock sets the slave copying the master.
5. If J=1, K=0 the high Q-output of the master goes to the J-input of the slave and negative transection of the clock sets the slave copying the master.
6. If J=1, K=1 toggles on the positive transection of a clock and thus slave toggle on the negative transection of clock.

State Table, State Diagram, State Equation, State Reduction and State Assignment:

State Table:

- It is defined as the table that shows the relation between present state, Next state and Output.

Present State (P.S)		Input	Next State		Output
Q_A	Q_B	X	Q_A^+	Q_B^+	Y

Fig: Sample of State Table

State Diagram:

- Diagrammatic Representation of State table is known as State Diagram

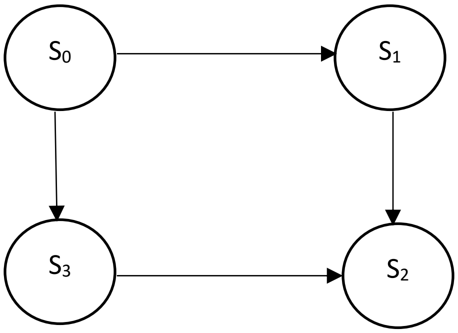


Fig: Sample State Diagram

State Equation:

- A State equation is nothing but the LHS is the Next state and RHS is the combination of PS and Input and this combination will give LHS as 1.

Example State Equation of JK:

$Q_{n+1}=Q_n'. J + Q_n. K'$

State Reduction:

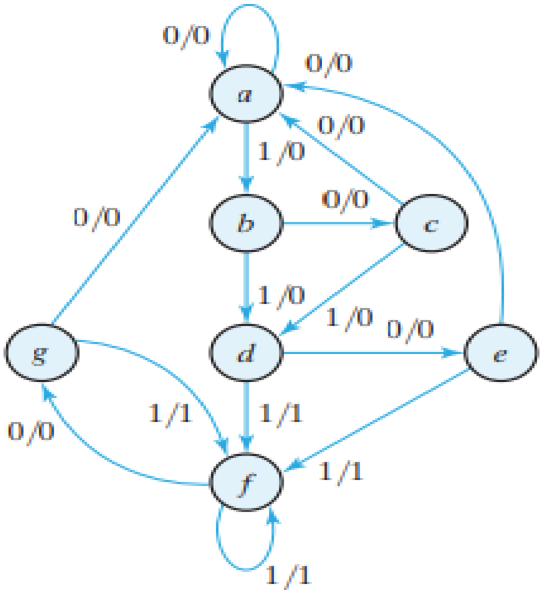
- State Reduction is process of Reducing the number of Flipflops or states given in any state diagram.

NOTE: State will be reduced if the next state and Output are same.

State Assignment:

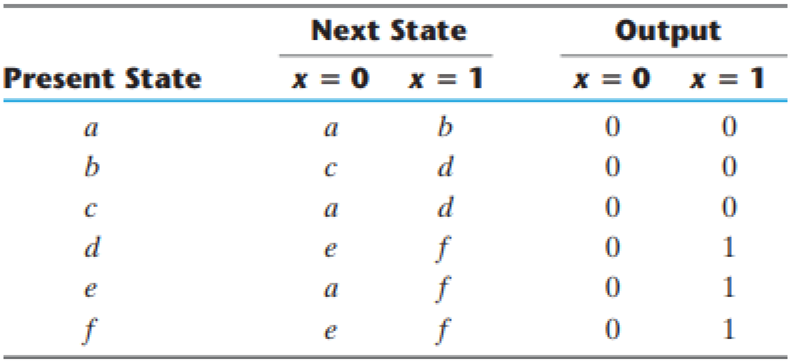
- State assignment is the process of assigning binary codes to the states Example S_0=00, S_1=01 etc.

Example:



State table representation of above state diagram

	Next State		Output	
Present State	**x = 0**	**x = 1**	**x = 0**	**x = 1**
a	*a*	*b*	0	0
b	*c*	*d*	0	0
c	*a*	*d*	0	0
d	*e*	*f*	0	1
e	*a*	*f*	0	1
f	*g*	*f*	0	1
g	*a*	*f*	0	1



	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	e	f	0	1

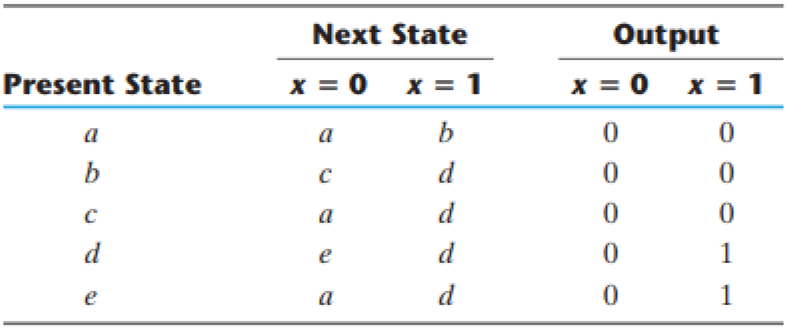
First Reduced State table

Now again checking for Same Next state and Output

We found the one i.e., d and f

So again, rewriting the state table

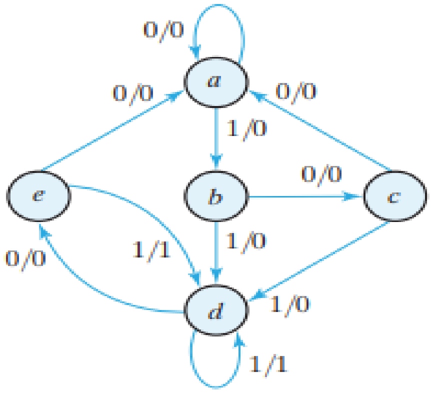
i.e.



	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Reduced state table

No more Same next sate and Output
So, we stop here and draw the final state diagram
i.e.



Design Procedure:

Following are the steps we have to follow to design the sequential circuit from the given state diagram:

STEP 1: Check for the State diagram whether it is given or not.

STEP 2: Design the State Table for given State Diagram.

STEP 3: Reduce the state if needed or possible.

STEP 4: DO the state assignment (if needed).

STEP 5: Determine the required flipflop and assign the letter symbol.

STEP 6: Decide the type or flip flop por look the type of flip flop given in question to be used.

STEP 7: Derrive the Circuit Excitation table from state table.

STEP 8: Obtain the expression for flip flop and output.

STEP 9: Implement the Circuit.