UNIT-4

The bistable element

By cross coupling a pair of NAND gates which are both connected as inverters, a bistable element is formed. There are two possible states for the element: (a) Q = 0, $\bar{Q}$ =1 and (b) Q =1, $\bar{Q}$=0 (see Figure 6.1). Initially, when the circuit is switched on, the bistable element will take up one of these two states and without external intervention will remain in that state indefinitely, or until the power has been removed.

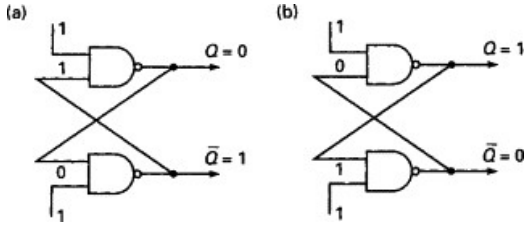
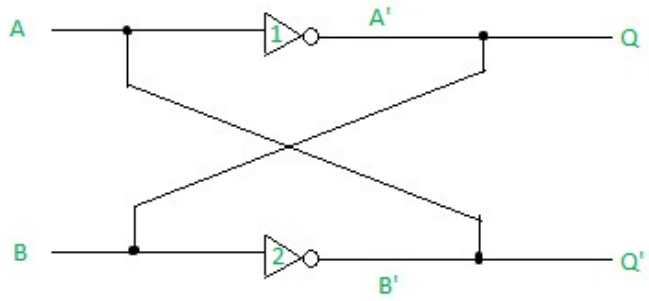


Figure 6.1. (a) and (b) The two states of a pair of cross-coupled NAND gates

One bit memory cell (or Basic Bistable element)

One Bit memory cell is also called **Basic Bistable element**. It has two cross-coupled inverters, 2 outputs Q and $\bar{Q}$. It is called "Bistable" as the basic bistable element circuit has two stable states logic 0 and logic 1.

The following diagram shows the Basic Bistable element:



(A) when **A=0**,

(i) In inverter1, Q = A'= B= 1

(ii)In inverter2, Q' = B' = A = 0

(B) when **A=1**,

(i) In inverter1, Q = A'= B= 0

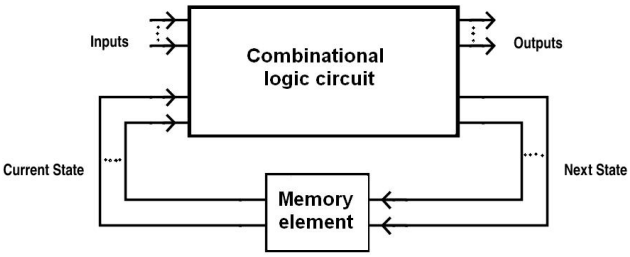
(ii) In inverter2, Q' = B' = A = 1

Some key points:

1. The 2 outputs are always complementary.
2. The circuit has 2 stable states. when Q=1, it is **Set state**. when Q=0, it is **Reset state**.
3. The circuit can store 1-bit of digital information and so it is called one-bit memory cell.
4. The one-bit information stored in the circuit is locked or latched in the circuit. This circuit is also called **Latch.**

Sequential Logic circuits:

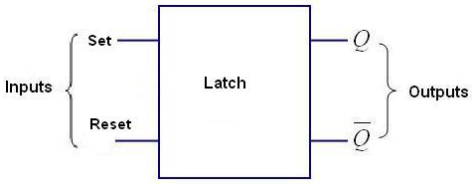
The logic circuit whose output depends upon both present and previous input is known as sequential logic circuit. This circuit comprises of a combinational logic circuit, memory element, input variables and output variables. The block diagram of a sequential logic circuit is shown in figure.



It is very important to note that the basic building block for combinational logic circuits is the 'LOGIC GATE' and for sequential logic circuits is the 'Flip-Flop'.

LATCH:

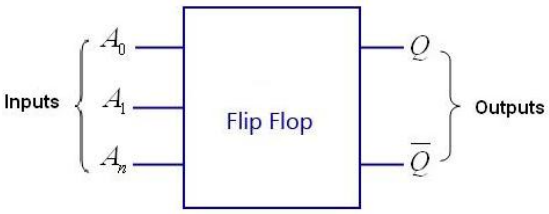
A latch is an electronic device, which changes its output immediately based on the applied input. It is used to store 1 or 0 at any specified time. It consists of two inputs namely 'SET' and 'RESET' and two outputs, which are complement to each other.



Latch is a type of bistable circuit, similar to a flip-flop, that can reside in either of two states by virtue of feedback arrangement. The main difference between latches and flip-flops is in the method used for changing their state.

FLIP-FLOP:

Flip-flop is a basic digital memory circuit, which stores one bit of information. The flip-flop is a 'bistable' logic circuit, which is a type of multivibrator with two stable states of output. These two states are known as "SET" and "RESET". Due to this character, these bistable elements are used as memory devices.



Flip-flop is a circuit that maintains a state until directed by input to change the state. A basic flip-flop can be constructed using four-NAND or four-NOR gates.

The block schematic representation of flip-flop is as shown figure.

Where, Q → Normal output; $\overline{Q}$ → Inverted output.

$A_0, A_1, A_n \rightarrow$ Inputs

Basically there are two categories of bistable elements. They are 'latches' and 'flip-flops'.

LATCHES:

Latches are basic storage elements that operate with signal levels (rather than signal transitions). Latches controlled by a clock transition are flip-flops. Latches are level-sensitive devices. Latches are useful for the design of the asynchronous sequential circuit. Latches are sequential circuit with two stable states. These are sensitive to the input voltage applied and does not depend on the clock pulse. Flip flops that do not use clock pulse are referred to as latch.

NAND/NOR GATE LATCH:

Consider SR (Set Reset) latches, formed with two cross coupled NAND gates and NOR gates as shown in figures (a) and (b). In these latches, S and R are the two inputs; and the two outputs are Q and $\overline{Q}$. Notice that the output of each gate is connected to the input of the other gate. This produces the feedback, which is the characteristic of all multivibrators.

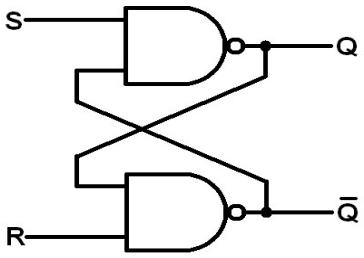


Fig. (a)

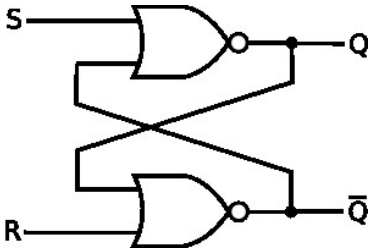


Fig. (b)

Inputs		Output
R	S	Q
0	0	Unpredictable state
0	1	1
1	0	0
1	1	No change

(a) NAND latch

Inputs		Output
R	S	Q
0	0	No change
0	1	1
1	0	0
1	1	Unpredictable state

(b) NOR latch

Operation:

i) When R and S inputs are both at logic 0, both outputs go to a logic 1. This is known as 'unpredictable state' (or) 'prohibited state' (or) 'invalid state' for the latch and is not used.

ii) When R=0 and S=1; the output Q is set logical 1. This is called the SET condition.

iii) When S=0 and R=1; the output Q is RESET (cleared) to 0. This is called the RESET condition.

iv) When both inputs R and S are at 1; this is the idle or at rest condition and leaves the outputs Q and $\overline{Q}$ in their previous complementary states. This is called the hold condition (or) no change state.

By observing the above four operations, we can rewrite the truth table for NAND latch with complete details as shown in below table.

Mode of operation	Inputs		Output		
	R	S	Q	$\overline{Q}$	Effect on Q
Prohibited	0	0	1	1	Do not use
Set	0	1	1	0	For setting Q to 1
Reset	1	0	0	1	For setting Q to 0
Hold	1	1	Q	$\overline{Q}$	Depends on Previous State

From the above table, we must note that whenever we use a NAND latch, we must avoid the condition when both inputs are at LOW at the same time.

We can rewrite the truth table for NOR latch with complete details as shown in below table.

Mode of operation	Inputs		Output		
	R	S	Q	$\overline{Q}$	Effect on Q
Prohibited	0	0	Q	$\overline{Q}$	Depends on Previous State
Set	0	1	1	0	For setting Q to 1
Reset	1	0	0	1	For setting Q to 0
Hold	1	1	0	0	Do not use

ASYNCHRONOUS AND SYNCHRONOUS SYSTEMS:

In any logic system, if the output responds to changes in one or more input states at any time, that system is called Asynchronous system. It is difficult to design and troubleshoot.

In a logic system, if the output state changes depending upon the status of a common input signal called "CLOCK" in addition to the states of other inputs, that system is called Synchronous system. The CLOCK is generally rectangular or square pulse train. This CLOCK signal is distributed to all parts of the system. The system output changes state when the CLOCK changes from 0 to 1 or from 1 to 0. The Synchronous system is easy to design and troubleshoot.

NECESSITY OF CLOCK IN FLIP-FLOP AND TYPES OF TRIGGERING:

In flip-flop, clock signal is used to match the time intervals (or) to achieve synchronization. A timing device called 'system clock' generates a clock pulse and this clock pulse synchronizes all the flip-flops in sequential circuits. There are mainly two types of triggering. They are,

1. Level triggering
2. Edge triggering

Triggering means making a circuit active. In **level triggering** the circuit will become active when the clock pulse is on a particular **level**. In edge **triggering** the circuit becomes active at negative or positive edge of the clock signal.

LEVEL TRIGGERING:

In level triggering, once the CLOCK becomes high or low, the output changes according to the state of other inputs. As the CLOCK signal is maintained at that level for a small time period, and if changes in the inputs occur in this time the outputs will respond to these changes. Hence there is a possibility of change in output more than once in one CLOCK cycle. This is called **RACING**.

The change in output state either at positive level or negative level of the enable input is known as level triggering or level clocking. There are two types of level triggering in flip-flops. They are,

i) Positive level triggering
ii) Negative level triggering

i) Positive level triggering:

In positive level triggering, triggering occurs on the positive level of the clock pulse.

Figure (1): Positive Level Triggering

ii) Negative level triggering:

In negative level triggering, triggering occurs on the negative level of the clock pulse.

Figure (2): Negative Level Triggering

EDGE TRIGGERING:

In order to avoid the Racing problem, Edge triggering is used. In this, the CLOCK pulses are made into narrow pulses. These narrow pulses can be obtained by passing the CLOCK signal through a Differentiator circuit. As the CLOCK pulses are very narrow, after the CLOCK changes its state and once the outputs respond to the states of inputs at that time, the CLOCK changes to its initial state immediately so that the next changes in inputs cannot change the outputs.

The change in output state either at positive edge (raising edge) or negative edge (falling edge) of the clock pulse is known as edge triggering. In this, flip-flop responds only during clock pulse transition at edges. The two types of edge triggering are,

i) Positive edge triggering

ii) Negative edge triggering

i) Positive edge triggering:

It occurs when the output response of the flip-flop changes only at the positive going edge of the clock pulse.

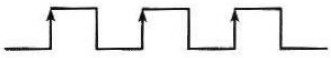


Figure (3): Positive Edge Triggering

ii) Negative edge triggering:

It occurs only when the output response of the flip-flop changes only at the negative going edge of the clock pulse.

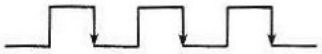


Figure (4): Negative Edge Triggering

RS FLIP-FLOP:

A Latch is similar to a flip-flop; that can reside in either of the two stable states. Therefore, the NAND and NOR latches are also called as "RS flip-flop". The logic symbol of RS flip-flop is shown in figure (a). The logic circuit of RS flip-flop is sgown in figure (b). The RS flip-flop has two inputs, S and R. The two outputs are Q and $\overline{Q}$. Note that the outputs are always are opposite, or complementary in the flip-flops. The output of each gate is connected to the input of the other gate. This produces the feedback, which is the characteristic of all multivibrators.

Operation:

i) When R and S inputs are both at logic 0, both outputs go to a logic 1. This is known as 'unpredictable state' (or) 'prohibited state' (or) 'invalid state' for the latch and is not used.

ii) When R=0 and S=1; the output Q is set logical 1. This is called the SET condition.

iii) When S=0 and R=1; the output Q is RESET (cleared) to 0. This is called the RESET condition.

iv) When both inputs R and S are at 1; this is the idle or at rest condition and leaves the outputs Q and $\overline{Q}$ in their previous complementary states. This is called the hold condition (or) no change state.

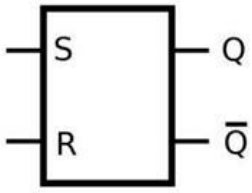


Fig. (a)

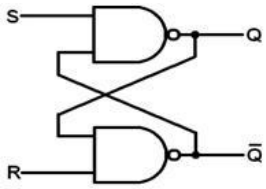


Fig. (b)

Mode of operation	Inputs		Output		
	R	S	Q	$\overline{Q}$	State
Prohibited	0	0	1	1	Do not use
Set	0	1	1	0	For setting Q to 1
Reset	1	0	0	1	For setting Q to 0
Hold	1	1	Q	$\overline{Q}$	Depends on Previous State

Truth Table

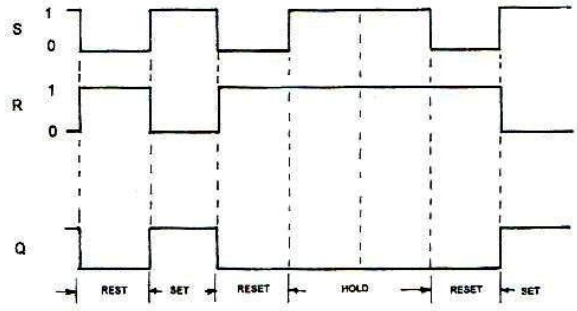


FIG. (c) TIMING DIAGRAM FOR AN RS FLIP-FLOP

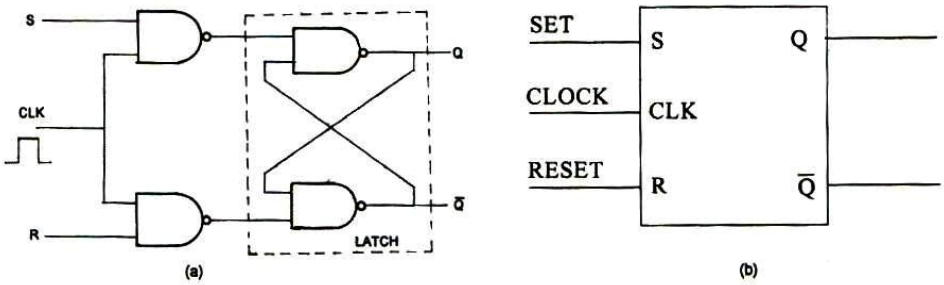
Figure (c) shows the timing diagram for an RS flip-flop. Notice that the output Q goes HIGH whenever R goes LOW; and the output Q goes LOW whenever S goes LOW. The output Q holds, whenever the both inputs are HIGH.

CLOCKED RS FLIP-FLOP (RST FLIP-FLOP):

The flip-flops are synchronous bistable devices. The term synchronous indicates that the output changes its state only at a specified point on a triggering input called the 'clock' (known as a control input C); i.e., changes in the output occur in synchronisation with the clock.

The block diagram and circuit of Clocked RS flip-flop using NAND gates are shown in figure (a) and (b) respectively. The output of the clocked RS flip-flop changes its state with respect to the inputs on the occurance of clock pulse. The applied clock may be edge triggered or level triggered.

The two inputs R and S along with clock signal are coupled to the flop-flop inputs.



Operation:

When the clock signal is LOW, the inputs does not affect the output i.e., remains in No change state. When the clock signal is HIGH, the following changes occur based on applied inputs.

Case (i): If both the inputs of SR flip-flop are '0' (i.e., S=R=0), then the flip-flop enters into hold mode. In this mode, output remains same as that of the previous state (i.e., no change in the output).

Case (ii): If S=0 and R=1, then the flip-flop enters into reset mode. In the reset mode, the data inputs reset the output 'Q'. Thus, the outputs are $Q=0, \overline{Q}=1$.

Case (iii): If S=1 and R=0, then the flip-flop enters into set mode. In the set mode, the data inputs set the output 'Q'. Thus, the outputs are $Q=1, \overline{Q}=0$.

Case (iv): If both the inputs of SR flip-flop are '1' (i.e., S=R=1), then the flip-flop outputs are $Q=1, \overline{Q}=1$ which is invalid state and will not be used. This state is called intermediate state.

The truth table for a clocked RS flip-flop is shown in table.

Inputs			Outputs		
CLK	R	S	Q	$\overline{Q}$	Effect on Q
0	0	0	Q	$\overline{Q}$	No change
0	0	1	Q	$\overline{Q}$	No change
0	1	0	Q	$\overline{Q}$	No change
0	1	1	Q	$\overline{Q}$	No change
1	0	0	Q	$\overline{Q}$	No change
1	0	1	1	0	Set
1	1	0	0	1	Reset
1	1	1	1	1	Intermediate (Invalid state)

Figure (c) shows the timing diagram for a clocked RS flip-flop. When S and CLK are HIGH, then Q goes HIGH. When R and CLK are HIGH, then Q returns to LOW. From the timing diagram, note the following:

a) The clock pulse has no effect on output Q when S and R are at LOW level.

b) We can say that this flip-flop operates 'synchronously', as it operates in step with the clock. This characteristic is very important in the design of various digital systems.

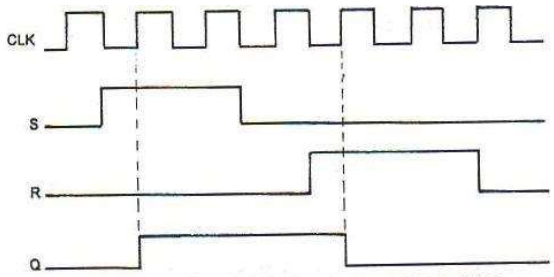


Fig. C TIMING DIAGRAM FOR A CLOCKED RS FLIP-FLOP

TOGGLE FLIP-FLOP (T FLIP-FLOP):

T flip-flop is also known as "Toggle Flip-flop". To avoid the occurrence of intermediate state in SR flip-flop, we should provide only one input to the flip-flop called Trigger input or Toggle input (T). Then the flip-flop acts as a Toggle switch. Toggling means 'Changing the next state output to complement of the present state output'. Figure (a) shows the logic diagram of a clocked T flip-flop; which has only one input referred to as T-input. Figure (b) shows the logic symbol of T flip-flop and its truth table is shown in the figure (c).

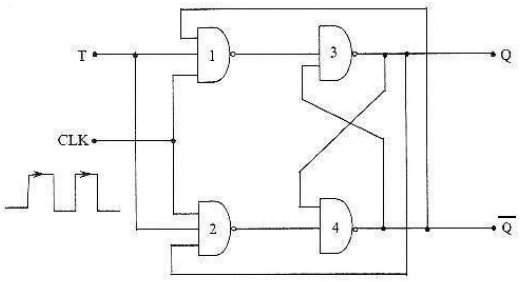
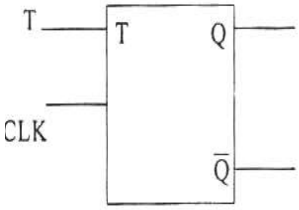


Fig. (a) LOGIC DIAGRAM OF T FLIP-FLOP



(b) LOGIC SYMBOL

CLK	Input (T)	Output (Q)
0	0	No change
0	1	
1	0	Q
1	1	$\overline{Q}$

(c) TRUTH TABLE

Fig. LOGIC DIAGRAM, LOGIC SYMBOL AND TRUTH TABLE OF T FLIP-FLOP

Operation:

From the above truth table, it is clear that T-flip-flop produces toggled output, when T input and clock signal are HIGH, i.e., for T=CLK=1, output = $\overline{Q}$. In other case i.e., when T-input is LOW, there is no change in the output, it maintains the previous state.

When CLK is LOW, whatever may be the value of other inputs, the outputs of both (1) and (2) are 1. The S and R inputs of the NAND latch are 1. The outputs will not change. This is no change state.

When CLK is high, the operation of the circuit is as follows.

When T=0, one of the inputs of both (1) and (2) will be 0. Hence their outputs will be 1. Hence S=R=1 for the NAND latch. The outputs Q and $\overline{Q}$ will not change.

When T=1, one of the inputs of both (1) and (2) are 1. If the previous outputs are Q=0 & $\overline{Q}$=1, all the inputs of (1) become 1. Hence its output becomes 0. The output Q= 0 is applied at the third input of (2). The output of (2) becomes 1. The inputs of NAND latch becomes S=0 and R=1. The latch will be set. The output Q will be changed to 1 and $\overline{Q}$ will be changed to 0. If the previous outputs are Q=1 & $\overline{Q}$=0, all the inputs of (2) become 1. Hence its output becomes 0. The output $\overline{Q}$=0 is applied at the third input of (1). The output of (1) becomes 1. The inputs of NAND latch becomes S=1 and R=0. The latch will be reset.

The output Q will be changed to 0 and $\overline{Q}$ will be changed to 1. Hence, if T=1, the output changes from 0 to 1 or from 1 to 0. This is called toggling.

The T or "toggle" flip-flop changes its output on each clock edge, giving an output which is half the frequency of the signal to the T input. It is useful for constructing binary counters, frequency dividers, and general binary addition devices.

The timing diagram of T- flip-flop is as shown in figure (d).

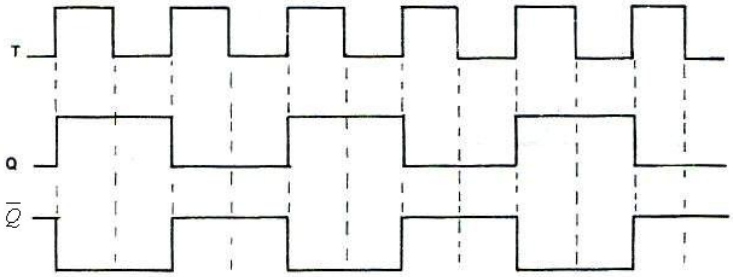
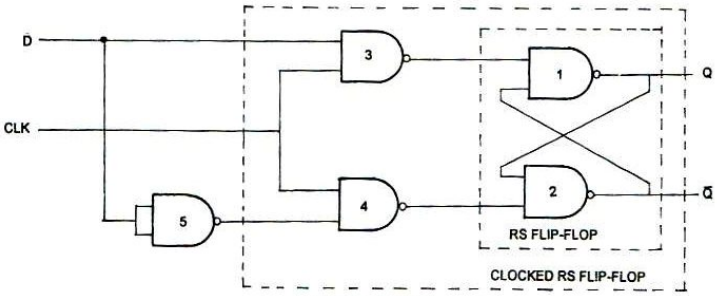


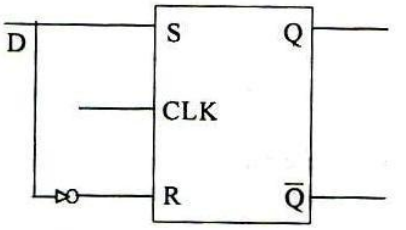
Fig. (d) INPUT AND OUTPUT WAVE FORMS OF T FLIP-FLOP

DELAY FLIP-FLOP (D FLIP-FLOP):

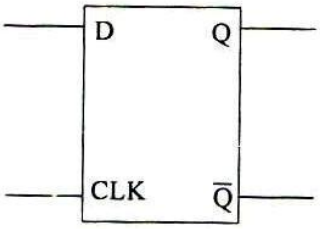
The D (Delay) flip-flop is used for storing the information. It is basically an Rs flip-flop with an inverter in the R input. Figure (a) shows a clocked D flip-flop. NAND gates 1 and 2 form a basic RS flip-flop and gates 3 and 4 modify it into a clocked RS flip-flop. The D input is to the S input and its complement through gate 5 is applied to the R input.



(a) LOGIC CIRCUIT



(b) MODIFICATION OF CLOCKED RS FLIP-FLOP INTO D SLIP FLOP



(c) LOGIC SYMBOL

Inputs		Outputs	
CLK	D	Q	$\overline{Q}$
0	0	No change	
0	1		
1	0	0	1
1	1	1	0

(d) TRUTH TABLE

Fig. LOGIC CIRCUIT, CONVERSION OF RS FLIP-FLOP INTO D FLIP-FLOP, TRUTH TABLE AND LOGIC SYMBOL OF D FLIP-FLOP

Thus D flip-flop may be formed from a clocked RS flip-flop by adding an inverter, as shown in the figure (b). Figure (c) shows the complete logic symbol for D flip-flop. D and CLK are the inputs and Q and $\overline{Q}$ are the outputs. The D flip-flop is also called as Delay flip-flop. The word 'delay' describes what happens to the data or information at input D. In other words, the data i.e., 0 or 1 at the input D is delayed by one clock pulse from getting to output Q.

When CLK is low, whatever may the value of D, one of the inputs of both NAND gates 3 and 4 is low. Hence their outputs are 1. The S and R inputs of NAND latch are 1. The outputs will not change. The previous states are maintained as it is.

When CLK is high, D is low, and the previous outputs are Q=1 and $\overline{Q}$=0, all the inputs of N4 are 1. Its output is 0. One of the inputs of N3 is 1, and the other inputs are 0. Its output will be 1. The inputs for the NAND latch are S=1 and R=0. Hence the output Q will become 0 and $\overline{Q}$ will become 1. If the previous outputs are Q=0 and $\overline{Q}$=1, one of the inputs of both N3 and N4 are 0. Their outputs are 1. The inputs for NAND latch will be S=R=1. Hence the outputs will not change, and remain as Q=0 and $\overline{Q}$=1.

When CLK is high, D=1, and the previous outputs are Q=0 and $\overline{Q}$=1, all the inputs of N3 are 1. Its output is 0. One of the inputs of N4 is 1, and the other inputs are 0. Its output will be 1. The inputs for the NAND latch are S=0 and R=1. Hence the output Q will become 1 and $\overline{Q}$ will become 0. If the previous outputs are Q=1 and $\overline{Q}$=0, one of the inputs of both N3 and N4 are 0. Their outputs are 1. The inputs for NAND latch will be S =R=1. Hence the outputs will not change, and remain as Q=1 and $\overline{Q}$=0.

Whatever may be the value of D, the same will be transferred to the output when the next CLOCK pulse appears. The input applied at the D input will be transferred to the output with some time delay.

Operation:

The output of the D flip-flop depends on the clock signal applied at its input.

Case (i): When clock signal is low (i.e., clock=0) there is no change in the output.

Case (ii): If clock signal is high (i.e., clock=1), data storage takes place.

For D=0; Reset = High; Q=0

For D=1; Set = High; Q=1

This indicates that the input data appears at the output after some delay i.e., at the end of the clock pulse. Thus, it is referred as delay flip-flop. The timing diagram of D flip-flop is shown in figure.

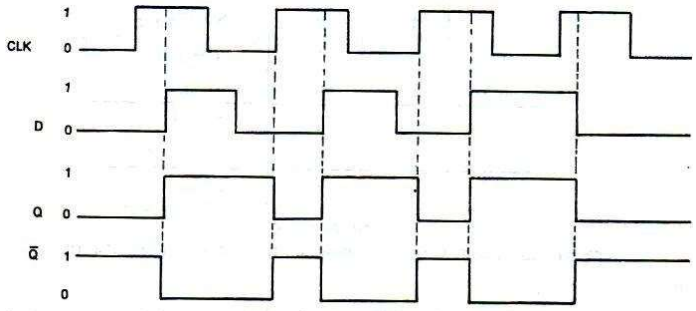


Fig. TIMING DIAGRAM FOR A CLOCKED D FLIP-FLOP

JK FLIP-FLOP:

The JK flip-flop has the features of all other flip-flops, and hence it can be considered as 'universal' flip-flop. This JK flip-flop is a refinement of the RS flip-flop. The occurrence of intermediate state in RS flip-flop can be avoided in JK flip-flop. The logic circuit and logic symbol of JK flip-flop is shown in figures a & b.

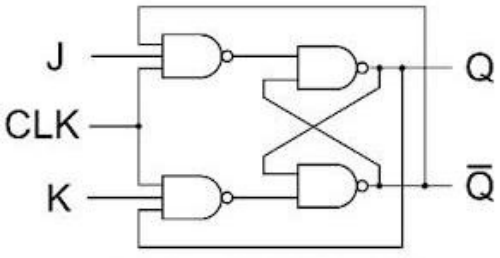


Fig. (a) LOGIC DIAGRAM

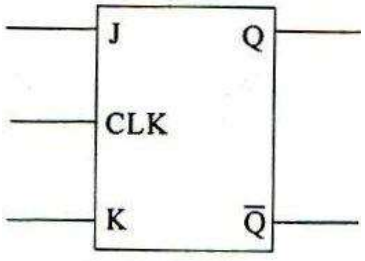


Fig.(b) LOGIC SYMBOL

Where, J,K → Data inputs; CLK → Clock input; Q → Normal output; $\overline{Q}$=Complementary output.

Operation:

When the clock signal is LOW, the inputs does not affect the output i.e., remains in No change state. When the clock signal is HIGH, the following changes occur based on applied inputs.

Case (i): If both the inputs of JK flip-flop are '0' (i.e., J=K=0), then the flip-flop remains in the "hold" mode. In this mode, ourput remains same as that of the previous state (i.e., no change in the output).

Case (ii): If J=0 and K=1, then the flip=flop enters into reset mode. In the reset mode the output Q is cleared to 0. Thus the outputs are Q=0, $\overline{Q}$=1.

Cse (iii): If J=1 and K=0, then the flip-flop enters into set mode. In the set mode, the output Q is set to 1. Thus the outputs are Q-1, $\overline{Q}$=0.

Case (iv): If both the inputs of JK flip-flop are '1' (i.e., J=K=1), then the flip-flop enters into toggle mode, in which output continuously shifts between logic '0' and logic '1' for complete clock pulse. Thus uncertain output is obtained.

The truth table and timing diagrams of JK flip-flop is shown below.

Inputs			Outputs	
CLK	J	K	Q	$\overline{Q}$
0	0	0		
0	0	1		
0	1	0	No change	
0	1	1	(HOLD)	
1	0	0		
1	0	1	0	1
1	1	0	1	0
1	1	1	Toggle	

(c) TRUTH TABLE

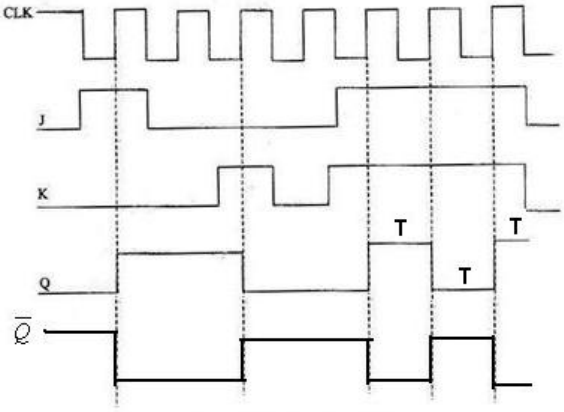


Fig. Timing diagram of JK Flip-flop

SYNCHRONOUS AND ASYNCHRONOUS INPUTS OF A FLIP-FLOP :

Synchronous Inputs: The inputs, whose effect on the flip-flop output is synchronised with the clock input are known as synchronous inputs. Examples of synchronous inputs are R, S, J, K and D inputs.

Asynchronous Inputs: The inputs, whose effect on the flip-flop output is independent of the clock input are known as asynchronous inputs. Examples of asynchronous inputs are preset and clear. Asynchronous inputs are also known as over inputs as, it override the state of the flip-flop to other.

Depending on the requirement of flip-flop, either of the synchronous inputs or asynchronous inputs will be in active state and the other will be in in-active state.

Distinguish between synchronous and asynchronous inputs of a flip-flop:

	Synchronous input		Asynchronous input
1	The inputs, whose effect on the flip-flop output is synchronised with the clock input are known as synchronous inputs.	1	The inputs, whose effect on the flip-flop output is synchronised with the clock input are known as synchronous inputs.
2	Polarity of these inputs is invariable.	2	Polarity of these inputs is variable.
3	These are edge triggered with respect to the clock signal.	3	These are level sensitive.
4	Example: data inputs to flip-flops.	4	Example: Preset and clear.

RACE AROUND CONDITION:

In a JK flip-flop, if J=K=1, when clock becomes high, the outputs will toggle. The changed outputs will be fed to the inputs. If the clock is at high level by the time the changed outputs are fed back to the inputs, the outputs will again toggle. There is a chance of toggling more than once in one clock cycle. This is called Race around condition. The Race around condition can be avoided by either

1) Using Edge triggering, or

2) Using Master-Slave JK flip-flop.

NEED FOR A MASTER SLAVE FLIP-FLOP:

Master-Slave flip-flop is needed to eliminate race around conditions in flip-flops. It reads the information into flip-flop and establishes new output values at different time periods. It completely eliminates the inputs that reverse through some gates outside the master-slave flip-flop.

The master-slave flip-flop responds to its input only at the leading edge of next clock pulse. This helps to avoid oscillatory races.

PRESET and CLEAR inputs of JK flip-flop:

Preset and clear inputs are used to reset and clear the flip-flop at any time. These are asynchronous inputs which are applied between the clock pulses. Preset and clear inputs changes the state of the flip-flop without using the clock pulses. Preset is also known as direct set. It forces the output Q to 1 to set the flip-flop. Clear is also known as direct reset. This forces the output Q to 0 to reset (clear) the flip-flop.

MASTER SLAVE FLIP-FLOP:

One flip-flop serves as a MASTER and the other as a SLAVE; and the overall circuit is known as 'Master-Slave flip-flop'. Whenever the clock is HIGH, the MASTER is active. According to the state of the data inputs, the output of the master is set or reset. At this stage, the SLAVE is inactive and its output remains in the previous state. Whenever the input clock is LOW, then the SLAVE is active. Note that final output of the Master-Slave flip-flop is the output of the Slave flip-flop. Hence the final output of the Master-Save flip-flop is available at the end of a clock pulse.

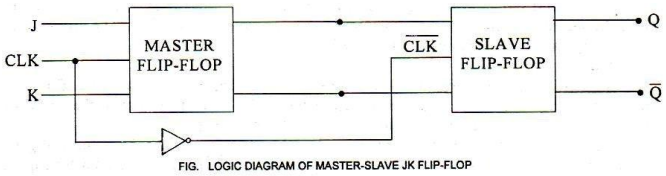


FIG. LOGIC DIAGRAM OF MASTER-SLAVE JK FLIP-FLOP

Here, J, K → Data inputs

CLK → Clock inputs

$\overline{CLK}$ → Inverter Clock input

Q → Output

$\overline{Q}$ → Complementary output

The logic symbol and circuit diagram of Master-Slave JK flip-flop is as shown in figure (2).

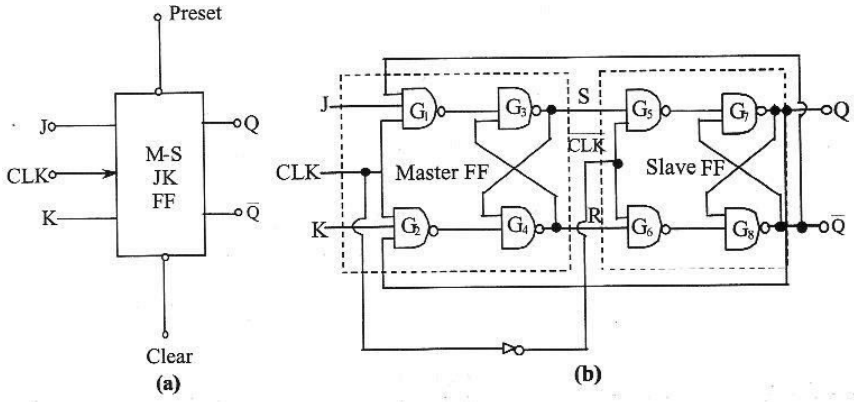


Figure (2)

In this type of flip-flop, the master flip-flop operates during positive level of clock cycle and the slave flip-flop operates during negative level of clock cycle.

Operation:

Case (i): When J=K=0 and CLK=0 (or) 1, both NAND gates G1 and G2 are disabled and circuit is inactive. Thus, no change in output.

Case (ii): When J=0, K=1 and CLK=1, the master flipflop enters into reset state, and produces outputs S=0, R=1. These outputs are fed to the slave flipflop. When clock signal goes low, the S and R inputs force the slave flipflop to reset the output because, slave flipflop operates only during low clock signal. Therefore, the outputs are Q=0 and $\overline{Q}$=1.

Case (iii): When J=1, K=0 and CLK=1, the master flopflop enters into set state, and produces outputs S=1, R=0. But, the outputs Q remains constant because the slave flipflop is inactive during the high clock. Thus, when clock signal goes low, slave flipflop becomes active and set the outputs to Q=1 and $\overline{Q}$=0.

Case (iv): When J=K=1 and CLK=1, the master-slave flipflop enters into toggle state. Thus, the outputs of master flipflop toggle between '0' and '1' till the completion of positive level of clock cycle. As soon as the negative level of clock cycle starts, slave flipflop takes the outputs of master flipflop and produces its corresponding outputs i.e., Q=1 and $\overline{Q}$=0.

Thus, there is no continuous toggling (i.e., variation) in output of master-slave JK flipflop. Hence, race around condition is avoided.

The turuth table and timing diagram of master-slave JK flipflop are as shown in figures (c) and (d).

CLK	J	K	Q	$\overline{Q}$	Mode
0	X	X	Q	$\overline{Q}$	No Change
1	0	0	Q	$\overline{Q}$	No Change
1	0	1	0	1	Reset
1	1	0	1	0	Set
1	1	1	$\overline{Q}$	Q	Toggle

Fig. (c) Truth table of Master-Slave JK Flipflop

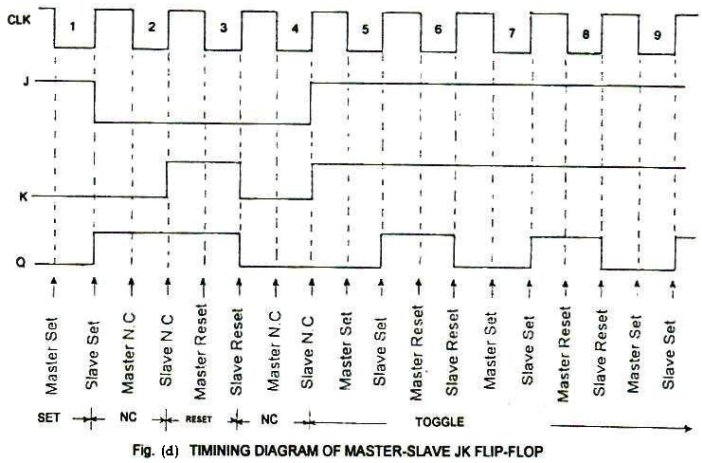
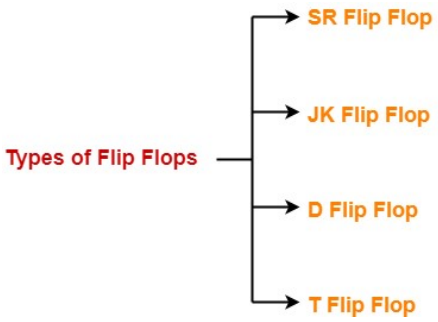


Fig. (d) TIMINING DIAGRAM OF MASTER-SLAVE JK FLIP-FLOP

EXCITATION TABLES, CHARACTERISTIC TABLES, CHARACTERISTIC EQUATIONS

Flip Flops-

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

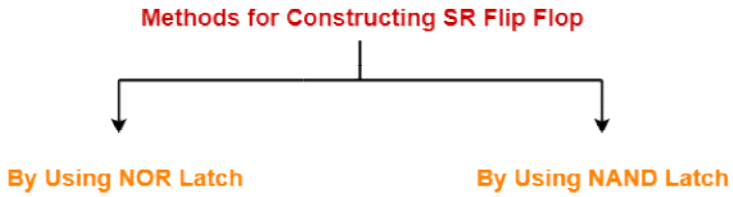


SR Flip Flop

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

Construction of SR Flip Flop-

There are following two methods for constructing a SR flip flop-



1. By using NOR latch
2. By using NAND latch

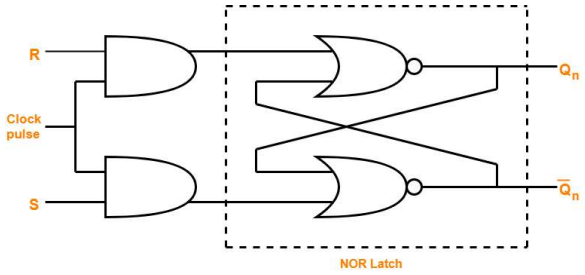
1. Construction of SR Flip Flop By Using NOR Latch-

This method of constructing SR Flip Flop uses-

- NOR latch
- Two AND gates

Logic Circuit-

The logic circuit for SR Flip Flop constructed using NOR latch is as shown below-



SR Flip Flop Using NOR Latch

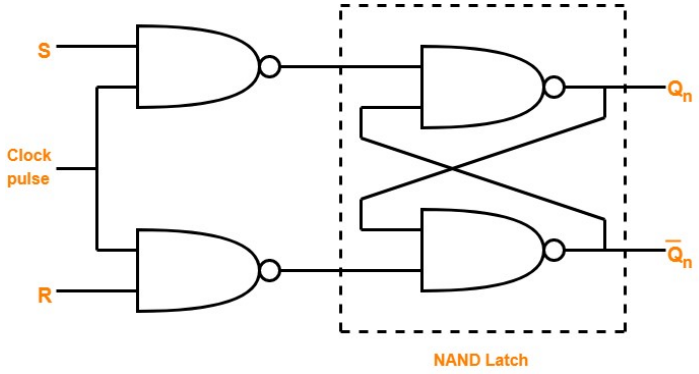
2. Construction of SR Flip Flop By Using NAND Latch-

This method of constructing SR Flip Flop uses-

- NAND latch
- Two NAND gates

Logic Circuit-

The logic circuit for SR Flip Flop constructed using NAND latch is as shown below-



SR Flip Flop Using NAND Latch

Logic Symbol-

The logic symbol for SR Flip Flop is as shown below-

Logic Symbol

Truth Table-

The truth table for SR Flip Flop is as shown below-

INPUTS			OUTPUTS
S	**R**	**Q_n (Present State)**	**Q_{n+1} (Next State)**
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	Indeterminate
1	1	1	Indeterminate

Truth Table

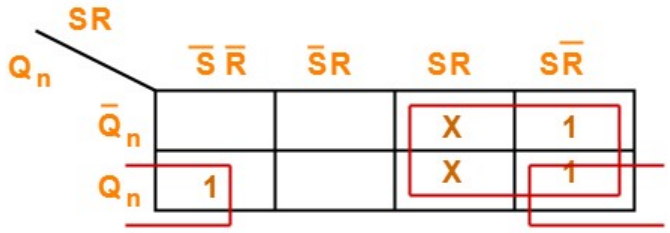
The above truth table may be reduced as-

INPUTS			OUTPUTS	REMARKS
S	**R**	**Q_n (Present State)**	**Q_{n+1} (Next State)**	**States and Conditions**
0	0	X	Q_n	Hold State condition S = R = 0
0	1	X	0	Reset state condition S = 0 , R = 1
1	0	X	1	Set state condition S = 1 , R = 0
1	1	X	Indeterminate	Indeterminate state condition S = R = 1

Truth Table

Characteristic Equation-

Draw a k map using the above truth table-



K Map

From here-

$$Q_{n+1} = (SR + SR') (Q_n + Q'_n) + Q_n (S'R' + SR')$$

$$\mathbf{Q_{n+1} = S + Q_nR'}$$

Excitation Table-

What is excitation table?

For a given combination of present state Q_n and next state Q_{n+1}, excitation table tell the inputs required.

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

Excitation Table

JK Flip Flop

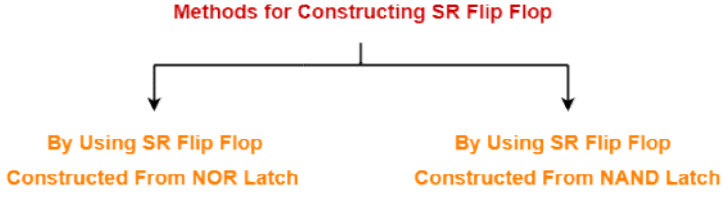
JK flip flop is a refined & improved version of **SR Flip Flop** that has been introduced to solve the problem of indeterminate state that occurs in SR flip flop when both the inputs are 1.

In 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.

Construction of JK Flip Flop-

There are following two methods for constructing a JK flip flop-



1. By using SR flip flop constructed from NOR latch
2. By using SR flip flop constructed from NAND latch

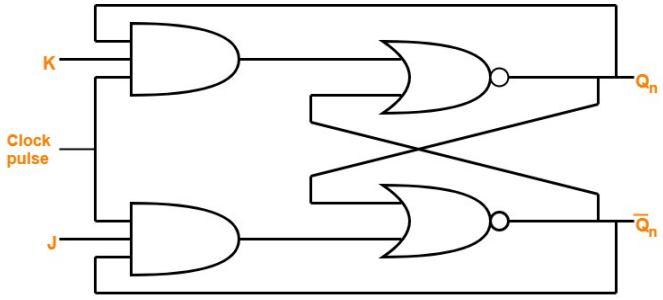
1. Construction of JK Flip Flop By Using SR Flip Flop Constructed From NOR Latch-

This method of constructing JK Flip Flop uses-

- SR Flip Flop constructed from NOR latch
- Two other connections

Logic Circuit-

The logic circuit for JK Flip Flop constructed using SR Flip Flop constructed from NOR latch is as shown below-



Logic Circuit For JK Flip Flop Using SR Flip Flop

(Constructed From NOR Latch)

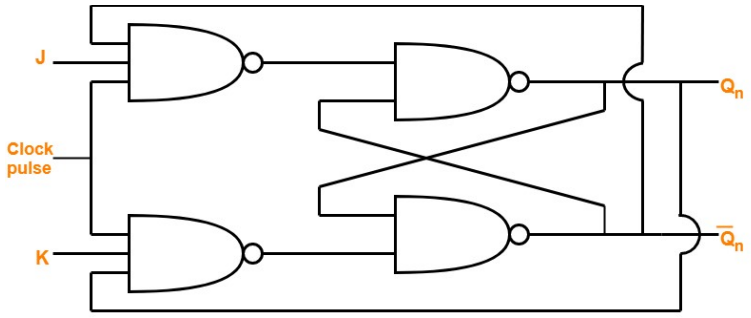
2. Construction of JK Flip Flop By Using SR Flip Flop Constructed From NAND Latch-

This method of constructing JK Flip Flop uses-

- SR Flip Flop constructed from NAND latch
- Two other connections

Logic Circuit-

The logic circuit for JK Flip Flop constructed using SR Flip Flop constructed from NAND latch is as shown below-



Logic Circuit For JK Flip Flop Using SR Flip Flop

(Constructed Using NAND Latch)

Logic Symbol-

The logic symbol for JK Flip Flop is as shown below-

Logic Symbol

Truth Table-

The truth table for JK Flip Flop is as shown below-

INPUTS			OUTPUTS
J	**K**	**Q_n (Present State)**	**Q_{n+1} (Next State)**
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

Truth Table

The above truth table may be reduced as-

INPUTS			OUTPUTS	REMARKS
J	**K**	**Q_n (Present State)**	**Q_{n+1} (Next State)**	**States and Conditions**
0	0	X	Q_n	Hold State condition J = K = 0
0	1	X	0	Reset state condition J = 0 , K = 1
1	0	X	1	Set state condition J = 1 , K = 0
1	1	X	Q'_n	Toggle state condition J = K = 1

Truth Table

Characteristic Equation-

Draw a k map using the above truth table-

K Map

From here-

$$Q_{n+1} = Q'_n (JK + JK') + Q_n (J'K' + JK')$$

$$\mathbf{Q_{n+1} = Q'_nJ + Q_nK'}$$

Excitation Table-

The excitation table of any flip flop is drawn using its truth table.

What is excitation table?

For a given combination of present state Q_n and next state Q_{n+1}, excitation table tell the inputs required.

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

Excitation Table

D Flip Flop

Truth Table-

The truth table for D Flip Flop is as shown below-

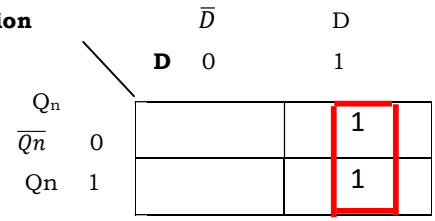
Inputs		Outputs	
CLK	D	Q	$\bar{Q}$
0	0	No change	
0	1		
1	0	0	1
1	1	1	0

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

Characteristic Table-

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

Characteristic equation



$Q_{n+1} = D$

Excitation Table

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

T Flip Flop

Truth Table-

The truth table for T Flip Flop is as shown below

CLK	Input (T)	Output (Q)
0 0	0 1	No change
1 1	0 1	Q $\overline{Q}$

CLK	T	Q_{n+1}
0	X	Q_n
1	0	Q_n
1	1	$\overline{Qn}$

Characteristic Table-

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

Characteristic equation

Qn	$\overline{T}$	T
	T 0	1
$\overline{Qn}$ 0		1
Qn 1	1	

$$Q_{n+1} = \overline{T}\,Qn + T\,\overline{Qn}$$

Excitation Table

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

***