EC 8392 – DIGITAL ELECTRONICS

UNIT – III : SYNCHRONOUS SEQUENTIAL CIRCUITS

TRIGGERING OF FLIP-FLOPS

The state of a Flip-Flop is switched by a momentary change in the input signal. This momentary change is called a trigger and the transition it causes is said to trigger the Flip-Flop. Clocked Flip-Flops are triggered by pulses. A clock pulse starts from an initial value of 0, goes momentarily to 1and after a short time, returns to its initial 0 value.

Latches are controlled by enable signal, and they are level triggered, either positive level triggered or negative level triggered. The output is free to change according to the S and R input values, when active level is maintained at the enable input.

Flip-Flops are different from latches. Flip-Flops are pulse or clock edge triggered instead of level triggered.

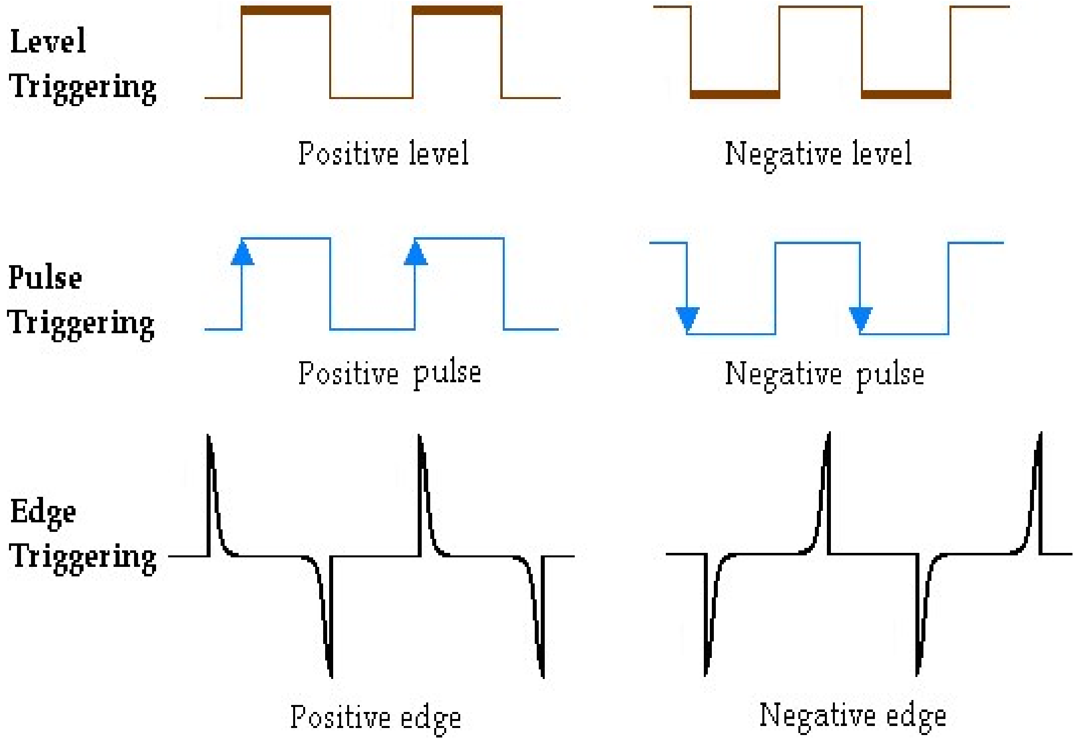


Fig : 3.12 – Various types of triggering

EDGE TRIGGERED FLIP-FLOPS

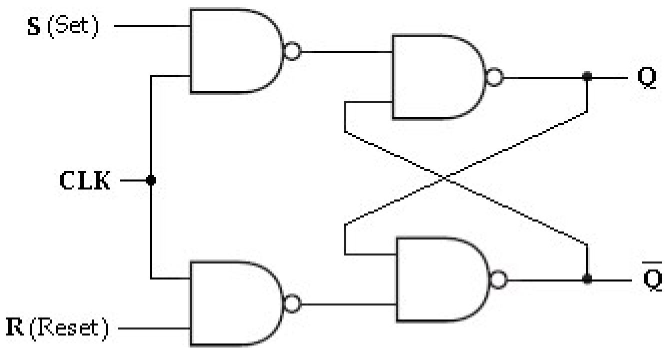
Flip-Flops are synchronous bistable devices (has two outputs Q and Q’). In this case, the term synchronous means that the output changes state only at a specified point on the triggering input called the clock (CLK), i.e., changes in the output occur in synchronization with the clock.

An ***edge-triggered Flip-Flop*** changes state either at the positive edge (rising edge) or at the negative edge (falling edge) of the clock pulse and is sensitive to its inputs only at this transition of the clock. The different types of edge-triggered Flip- Flops are—xS-R Flip-Flop, xJ-K Flip-Flop, xD Flip-Flop, xT Flip-Flop.

Although the S-R Flip-Flop is not available in IC form, it is the basis for the D and J-K Flip-Flops. Each type can be either positive edge-triggered (no bubble at C input) or negative edge-triggered (bubble at C input). The key to identifying an edge- triggered Flip-Flop by its logic symbol is the small triangle inside the block at the clock (C) input. This triangle is called the **dynamic input indicator**.

S-R Flip-Flop

The S and R inputs of the S-R Flip-Flop are called ***synchronous*** inputs because data on these inputs are transferred to the Flip-Flop's output only on the triggering edge of the clock pulse. The circuit is similar to SR latch except enable signal is replaced by clock pulse (CLK). On the positive edge of the clock pulse, the circuit responds to the S and R inputs.



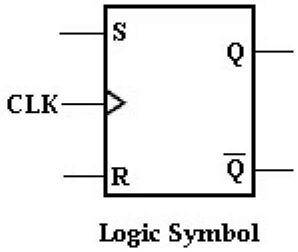


Fig : 3.13 - SR Flip-Flop

When S is HIGH and R is LOW, the Q output goes HIGH on the triggering edge of the clock pulse, and the Flip-Flop is SET. When S is LOW and R is HIGH, the Q output goes LOW on the triggering edge of the clock pulse, and the Flip-Flop is RESET. When both S and R are LOW, the output does not change from its prior state.

An invalid condition exists when both S and R are HIGH.

CLK	S	R	Qn	Qn+1	State
1 1	0 0	0 0	0 1	0 1	No Change (NC)
1 1	0 0	1 1	0 1	0 0	Reset
1 1	1 1	0 0	0 1	1 1	Set
1 1	1 1	1 1	0 1	x x	Indeterminate *
0 0	x x	x x	0 1	0 1	No Change (NC)

Truth table for SR Flip-Flop

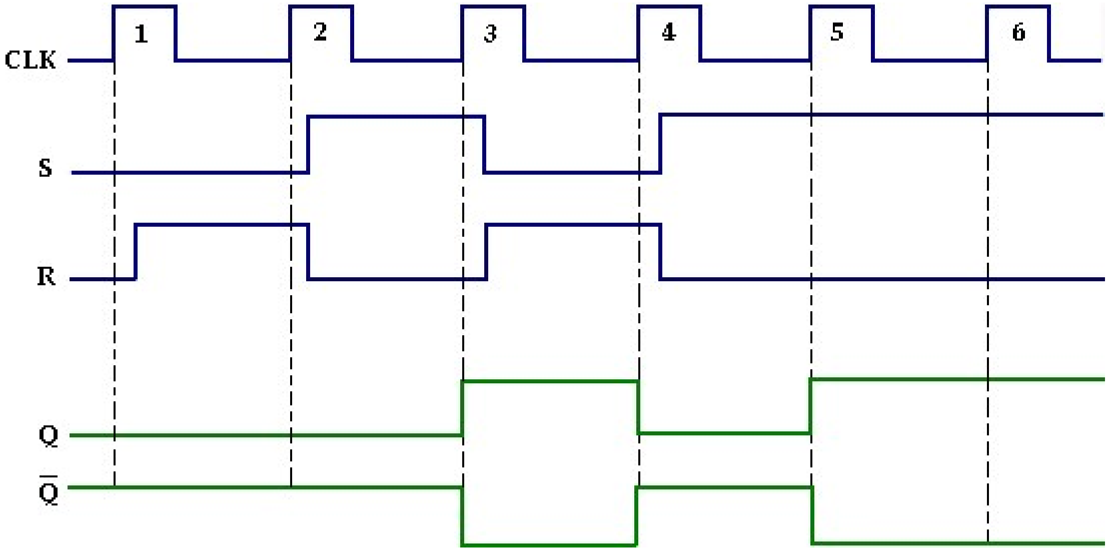


Fig : 3.14 - Input and output waveforms of SR Flip-Flop

J-K Flip-Flop:

JK means Jack Kilby, Texas Instrument (TI) Engineer, who invented IC in 1958. JK Flip-Flop has two inputs J(set) and K(reset). A JK Flip-Flop can be obtained from the clocked SR Flip-Flop by augmenting two AND gates as shown below.

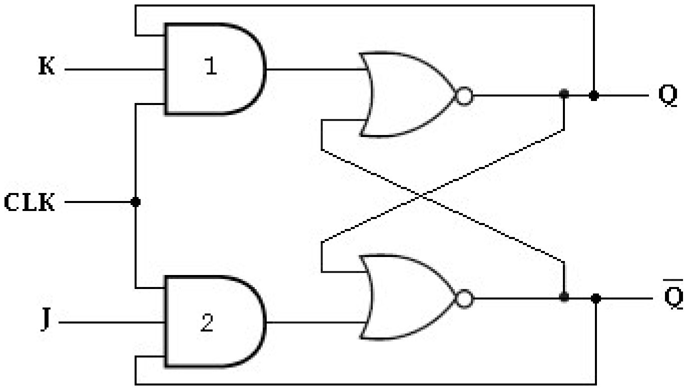


Fig : 3.15 - JK Flip Flop

The data input J and the output Q' are applied o the first AND gate and its output (JQ') is applied to the S input of SR Flip-Flop. Similarly, the data input K and the output Q are applied to the second AND gate and its output (KQ) is applied to the R input of SR Flip-Flop.

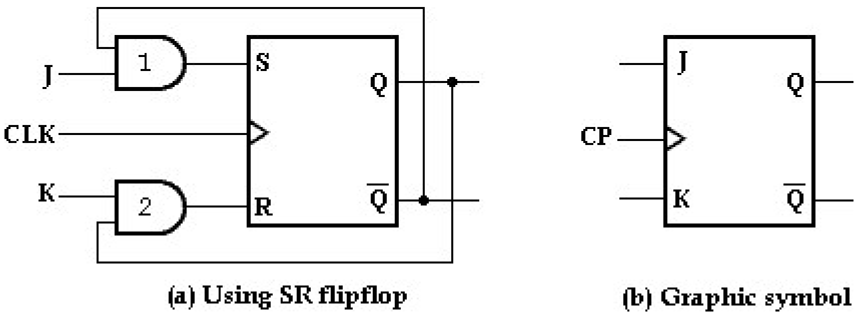


Fig : 3.16 – JK Flipflop using SR Flipflop

J= K= 0

When J=K= 0, both AND gates are disabled. Therefore clock pulse have no effect, hence the Flip-Flop output is same as the previous output.

J= 0, K= 1

When J= 0 and K= 1, AND gate 1 is disabled i.e., S= 0 and R= 1. This condition will reset the Flip-Flop to 0.

J= 1, K= 0

When J= 1 and K= 0, AND gate 2 is disabled i.e., S= 1 and R= 0. Therefore the Flip-Flop will set on the application of a clock pulse.

J= K= 0

When J=K= 1, it is possible to set or reset the Flip-Flop. If Q is High, AND gate 2 passes on a reset pulse to the next clock. When Q is low, AND gate 1 passes on a set pulse to the next clock. Eitherway, Q changes to the complement of the last state i.e., toggle. Toggle means to switch to the opposite state.

The truth table of JK Flip-Flop is given below.

CLK	**Inputs**		**Output**	**State**
	J	**K**	**Qn+1**	
1	0	0	Qn	No Change
1	0	1	0	Reset

1	1	0	1	Set
1	1	1	Q_n'	Toggle

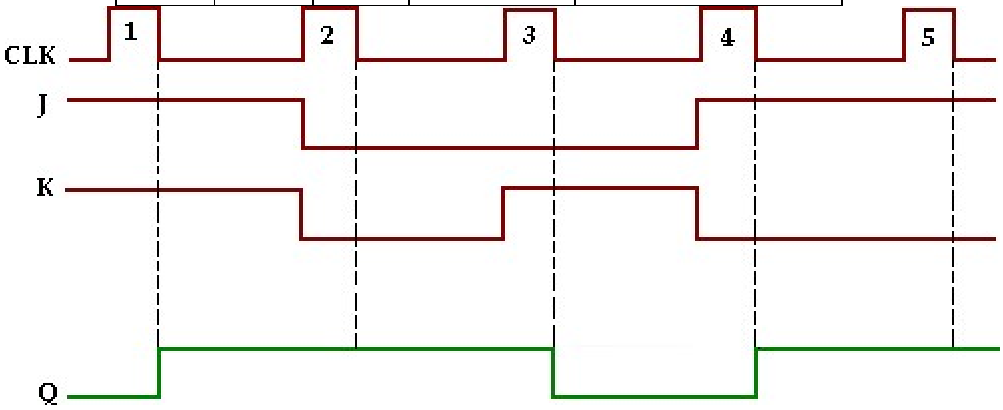


Fig : 3.17 - Input and output waveforms of JK Flip-Flop

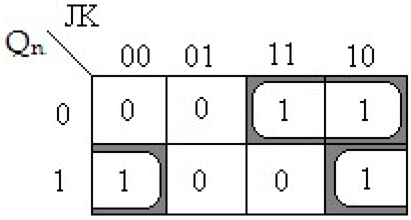
Characteristic table and Characteristic equation:

The characteristic table for JK Flip-Flop is shown in the table below. From the table, K-map for the next state transition (Q_{n+1}) can be drawn and the simplified logic expression which represents the characteristic equation of JK Flip-Flop can be found.

Qn	J	K	Qn+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

Characteristic table

K-map Simplification:



Q_n \ JK	00	01	11	10
0	0	0	1	1
1	1	0	0	1

Characteristic equation: $\mathbf{Q_{n+1}= JQ'+ K'Q.}$

D Flip-Flop:

Like in D latch, in D Flip-Flop the basic SR Flip-Flop is used with complemented inputs. The D Flip-Flop is similar to D-latch except clock pulse is used instead of enable input.

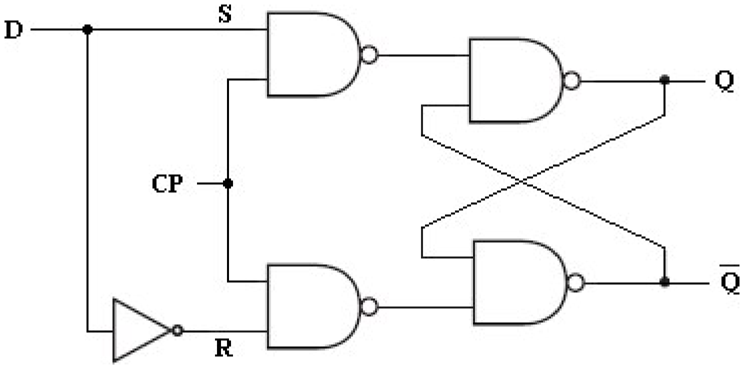


Fig : 3.18 - D Flip-Flop

To eliminate the undesirable condition of the indeterminate state in the RS Flip-Flop is to ensure that inputs S and R are never equal to 1 at the same time. This is done by D Flip-Flop. The D (***delay***) Flip-Flop has one input called delay input and clock pulse input. The D Flip-Flop using SR Flip-Flop is shown below.

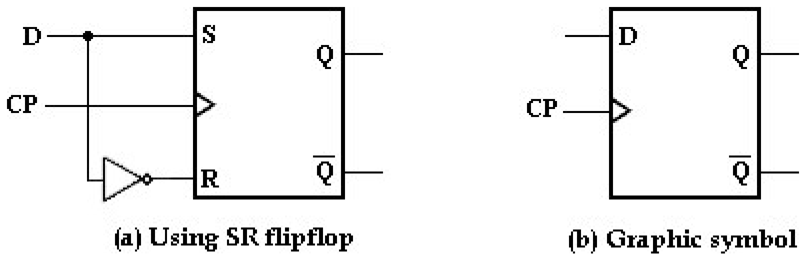


Fig : 3.19 – D Flipflop using SR Flipflop

The truth table of D Flip-Flop is given below.

Clock	D	Qn+1	State
1	0	0	Reset
1	1	1	Set
0	x	Qn	No Change

Truth table for D Flip-Flop

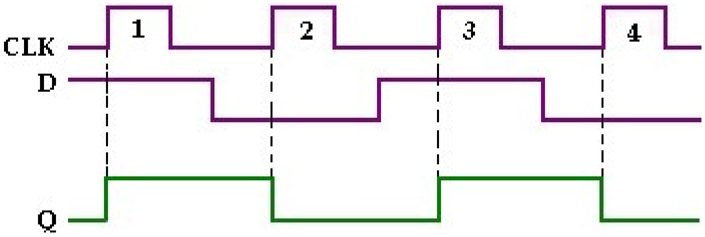


Fig : 3.20 - Input and output waveforms of clocked D Flip-Flop

Looking at the truth table for D Flip-Flop we can realize that Q_{n+1} function follows the D input at the positive going edges of the clock pulses.

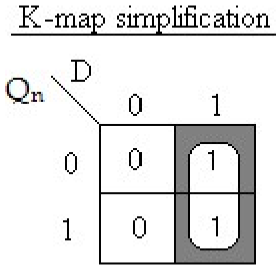
Characteristic table and Characteristic equation:

The characteristic table for D Flip-Flop shows that the next state of the Flip- Flop is independent of the present state since Q_{n+1} is equal to D. This means that an input pulse will transfer the value of input D into the output of the Flip-

Flop independent of the value of the output before the pulse was applied. The characteristic equation is derived from K-map.

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

Characteristic table



Characteristic equation: **Q_{n+1}= D.**

T Flip-Flop

The T (***Toggle***) Flip-Flop is a modification of the JK Flip-Flop. It is obtained from JK Flip-Flop by connecting both inputs J and K together, i.e., single input. Regardless of the present state, the Flip-Flop complements its output when the clock pulse occurs while input T= 1.

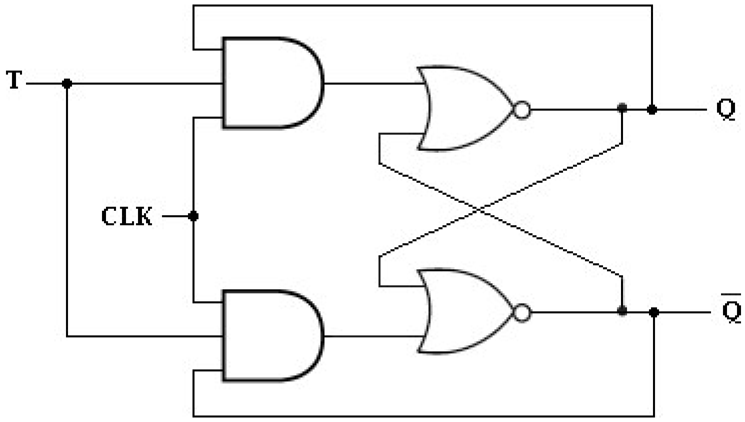


Fig : 3.21 - T Flip-Flop

When T= 0, Q_{n+1}= Q_n, ie., the next state is the sameas the present state and no change occurs. When T= 1, Qn+1= Qn',ie., the next state is the complement of the present state.

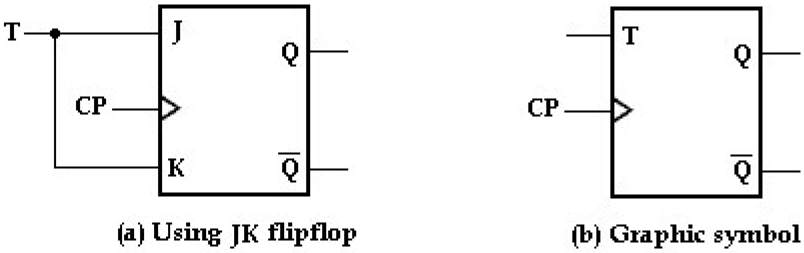


Fig : 3.22 – T flipflop using JK Flipflop

The truth table of T Flip-Flop is given below.

T	Qn+1	State
0	Qn	No Change
1	Qn'	Toggle

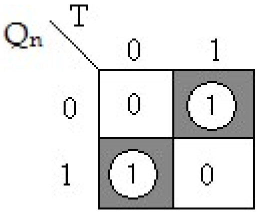
Truth table for T Flip-Flop

Characteristic table and Characteristic equation:

The characteristic table for T Flip-Flop is shown below and characteristic equation is derived using K-map.

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

K-map Simplification:



Q_n \ T	0	1
0	0	1
1	1	0

Characteristic equation: $\mathbf{Q_{n+1}= TQ_n' + T'Q_n}$.

Master-Slave JK Flip-Flop

A master-slave Flip-Flop is constructed using two separate JK Flip-Flops. The first Flip-Flop is called the master. It is driven by the positive edge of the clock pulse. The second Flip-Flop is called the slave. It is driven by the negative edge of the clock pulse. The logic diagram of a master-slave JK Flip-Flop is shown below.

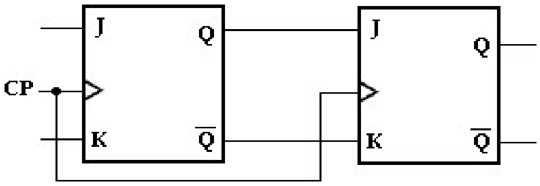


Fig : 3.32 - Logic diagram

When the clock pulse has a positive edge, the master acts according to its J- K inputs, but the slave does not respond, since it requires a negative edge at the clock input.

When the clock input has a negative edge, the slave Flip-Flop copies the master outputs. But the master does not respond since it requires a positive edge at its clock input. The clocked master-slave J-K Flip-Flop using NAND gates is shown below.

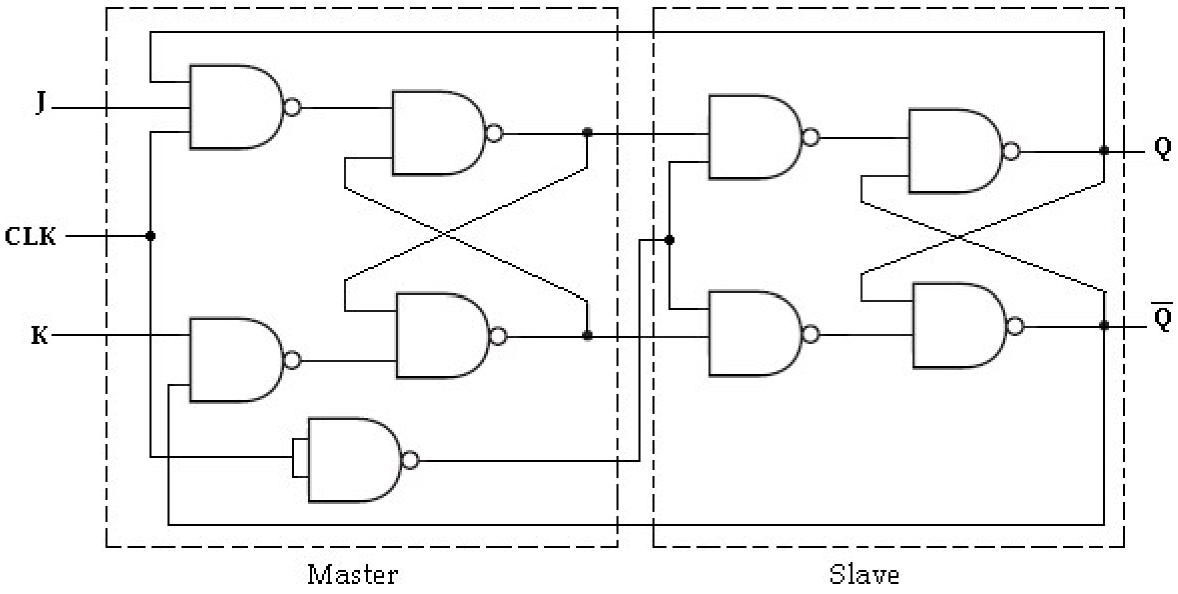


Fig : 3.33 - Master-Slave JK Flip-Flop

APPLICATION TABLE (OR) EXCITATION TABLE:

The ***characteristic table*** is useful for **analysis** and for defining the operation of the Flip-Flop. It specifies the next state (Q_{n+1}) when the inputs and present state are known.

The ***excitation or application table*** is useful for **design** process. It is used to find the Flip-Flop input conditions that will cause the required transition, when the present state (Q_n) and the next state (Q_{n+1}) are known.

SR Flip-Flop:

Present State	Inputs		Next State
Qn	S	R	Qn+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

Characteristic Table

Present State	Next State	Inputs		Inputs	
Qn	Qn+1	S	R	S	R
0	0	0	0	0	x
0	0	0	1		
0	1	1	0	1	0
1	0	0	1	0	1
1	1	0	0	x	0
1	1	1	0		

Modified Table

Present State	Next State	Inputs	
Qn	Qn+1	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

Excitation Table

The above table presents the excitation table for SR Flip-Flop. It consists of present state (Q_n), next state (Q_{n+1}) and a column for each input to show how the required transition is achieved.

There are 4 possible transitions from present state to next state. The required Input conditions for each of the four transitions are derived from the information available in the characteristic table. The symbol ‘x’ denotes the don’t care condition, it does not matter whether the input is 0 or 1.

JK Flip-Flop:

Present State	Inputs		Next State
Qn	**J**	**K**	**Qn+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

Characteristic Table

Present State	Next State	Inputs		Inputs	
Qn	**Qn+1**	**J**	**K**	**J**	**K**
0	0	0	0	0	x
0	0	0	1		
0	1	1	0	1	x
0	1	1	1		
1	0	0	1	x	1
1	0	1	1		
1	1	0	0	x	0
1	1	1	0		

Modified Table

Present State	Next State	Inputs	
Qn	**Qn+1**	**J**	**K**
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

Excitation Table

D Flip-Flop

Present State	Input	Next State
Qn	**D**	**Qn+1**
0	0	0
0	1	1
1	0	0
1	1	1

Present State	Next State	Input
Qn	**Qn+1**	**D**
0	0	0
0	1	1
1	0	0
1	1	1

Characteristic Table

Excitation Table

T Flip-Flop

Present State	Input	Next State
Qn	**T**	**Qn+1**
0	0	0
0	1	1
1	0	1
1	1	0

Characteristic Table

Present State	Next State	Input
Qn	**Qn+1**	**T**
0	0	0
0	1	1
1	0	1
1	1	0

Modified Table

REALIZATION OF ONE FLIP-FLOP USING OTHER FLIP-FLOPS

It is possible to convert one Flip-Flop into another Flip-Flop with some additional gates or simply doing some extra connection. The realization of one Flip-Flop using other Flip-Flops is implemented by the use of characteristic tables and excitation tables. Let us see few conversions among Flip-Flops.

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

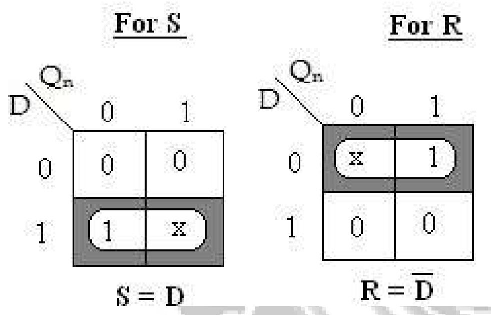
SR Flip-Flop to D Flip-Flop:

1. Write the characteristic table for required Flip-Flop (D Flip-Flop). x Write the excitation table for given Flip-Flop (SR Flip-Flop).
2. Determine the expression for the given Flip-Flop inputs (S and R) by using K-map.
3. Draw the Flip-Flop conversion logic diagram to obtain the required Flip- Flop (D Flip-Flop) by using the above obtained expression.

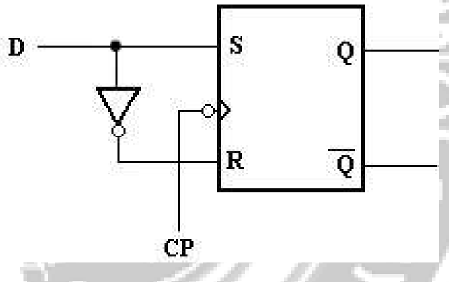
The excitation table for the above conversion is

Required Flip-Flop (D)			Given Flip-Flop (SR)	
Input	Present state	Next state	Flip-Flop Inputs	
D	Qn	Qn+1	S	R
0	0	0	0	x
0	1	0	0	1
1	0	1	1	0
1	1	1	x	0

K-map simplification



Logic diagram



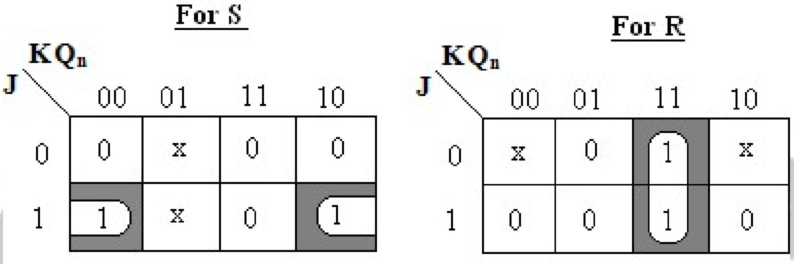
SR Flip-Flop to JK Flip-Flop

The excitation table for the above conversion is,

Inputs		Present state	Next state	Flip-Flop Input	
J	K	Qn	Qn+1	S	R
0	0	0	0	0	x
0	0	1	1	x	0
0	1	0	0	0	x
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	x	0

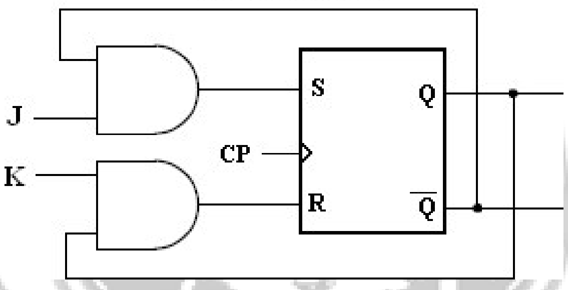
1	1	0	1	1	0
1	1	1	0	0	1

K-map simplification



$S = J\overline{Q}_n$ $R = KQ_n$

Logic diagram

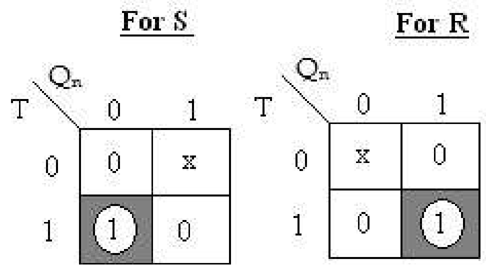


SR Flip-Flop to T Flip-Flop

The excitation table for the above conversion is

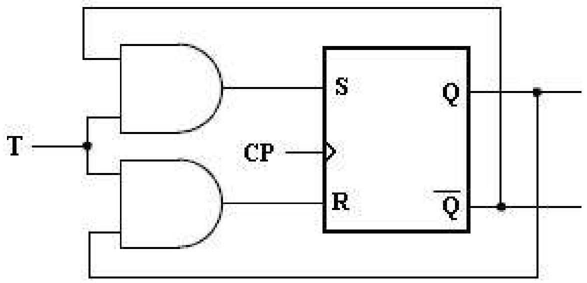
Input	Present state	Next state	Flip-Flop Inputs	
T	Qn	Qn+1	S	R
0	0	0	0	x
0	1	1	x	0
1	0	1	1	0
1	1	0	0	1

K-map simplification



S = T$\bar{Q}_n$ R = TQ_n

Logic diagram

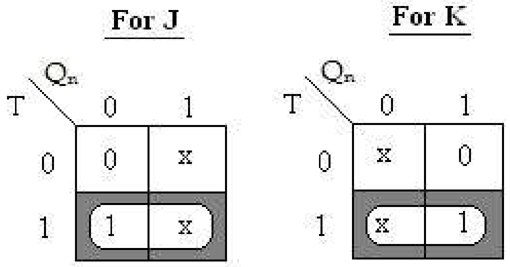


JK Flip-Flop to T Flip-Flop

The excitation table for the above conversion is

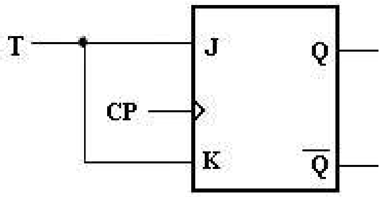
Input	Present state	Next state	Flip-Flop Inputs	
T	Qn	Qn+1	J	K
0	0	0	0	x
0	1	1	x	0
1	0	1	1	x
1	1	0	x	1

K-map simplification



J = T K = T

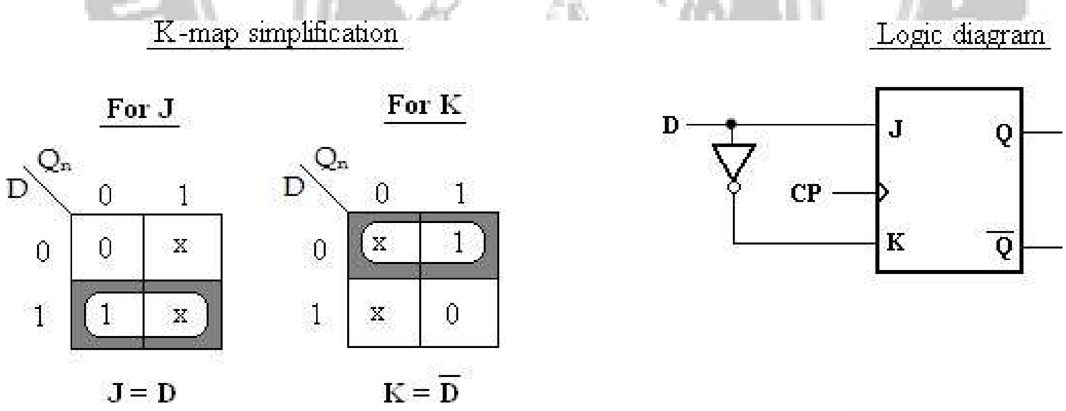
Logic diagram



JK Flip-Flop to D Flip-Flop

The excitation table for the above conversion is

Input	Present state	Next state	Flip-Flop Inputs	
D	Qn	Qn+1	J	K
0	0	0	0	x
0	1	0	x	1
1	0	1	1	x
1	1	1	x	0

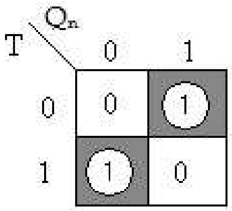


D Flip-Flop to T Flip-Flop

The excitation table for the above conversion is

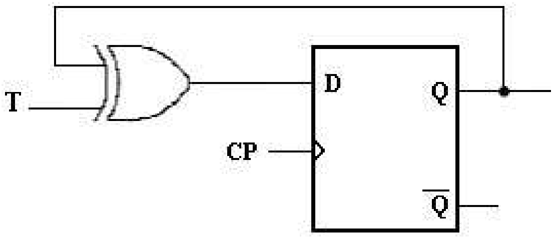
Input	Present state	Next state	Flip-Flop Input
T	Qn	Qn+1	D
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

K-map simplification



$$D = \overline{T}Q_n + T\overline{Q}_n$$
$$= T \oplus Q_n$$

Logic diagram



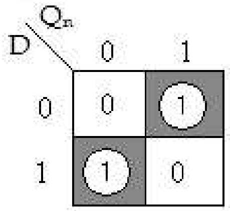
T Flip-Flop to D Flip-Flop

The excitation table for the above conversion is

Input	Present state	Next state	Flip-Flop Input
D	Qn	Qn+1	T
0	0	0	0
0	1	0	1

1	0	1	1
1	1	1	0

K-map simplification



$$T = D\overline{Q}_n + \overline{D}Q_n$$
$$= D \oplus Q_n$$

Logic diagram