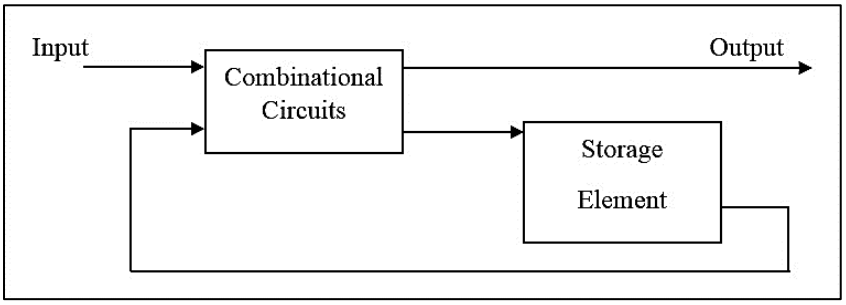
Chapter 5: Sequential Logic Circuits

Sequential Logic Circuits:

Up to now everything has been combinational circuits that is the output at any instant of time depends only on what the input is at that time.

Sequential logic differs from combinational logic in that the output of the logic device is dependent not only on the present inputs to the device, but also on past inputs. This implies that a sequential logic device has some kind of ***memory***.



There are two types of sequential logic circuit:

1) Asynchronous: The output can change state at any instant of time when the inputs are change.
2) Synchronous: The output can change state only at a discrete instants of time controlled by a signal called a "clock".

The Storage Element

The storage element used in an asynchronous sequential circuit called "Latches". While the storage element used in a synchronous (clocked) sequential circuit called "Flip Flop".

Latches: are the most basic type of flip flop operate with signal levels. The latches are useful for storing information and for the design an *asynchronous* sequential circuit.

Flip Flop: is a binary storage device capable of storing one bit of information. A sequential circuit may use many flip flop's to store as many bits as necessary. A flip flop is a clocked binary storage device, that storage either '0' or '1'. Under normal operation, the value will only change on the appropriate transition of the clock.

5.1 Latches

The latch is a type of bistable device that is normally placed is a category separate from that of flip flop. Latches are similar to flip flops because they are bistable device that can reside in either of two state. The main difference between latches & flip flops is in the method used for changing their state. There are many type of latches:

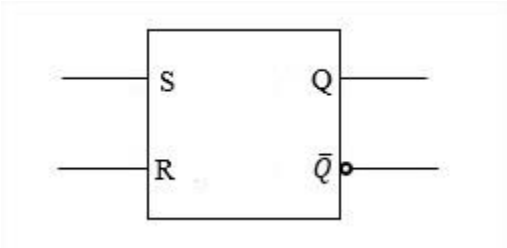
- S-R Latch
- Gated S-R Latch
- D-Latch

5.1.1 S-R Latch

This latch is called *SR-latch*, which stands for *set* and *reset*.

Active high SR-Latch Input

Input		Output		Comment
S	R	Q	$\bar{Q}$	
0	0	Q	$\bar{Q}$	Hold (no change)
0	1	0	1	Reset
1	0	1	0	Set
1	1	X	X	Don't care (invalid)

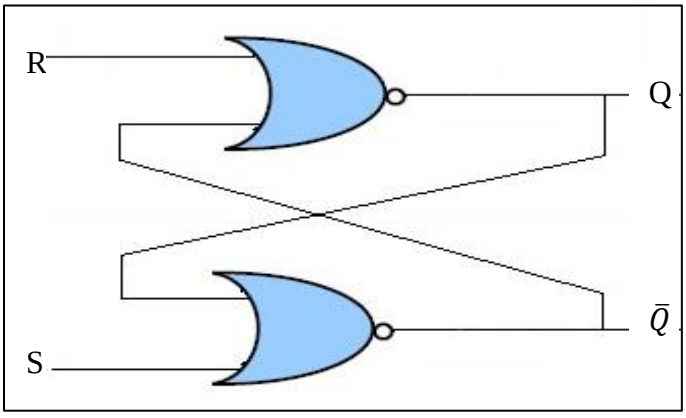


Active High Input

The characteristic table of SR latch can be written in another way:

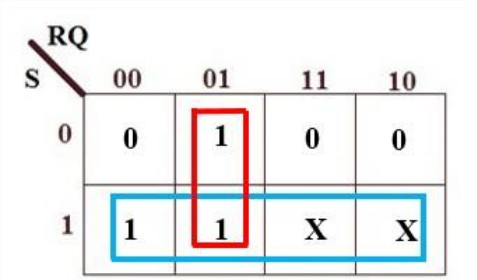
The stored output = Q_n

The new output = Q_n+1, n+1 mean next time



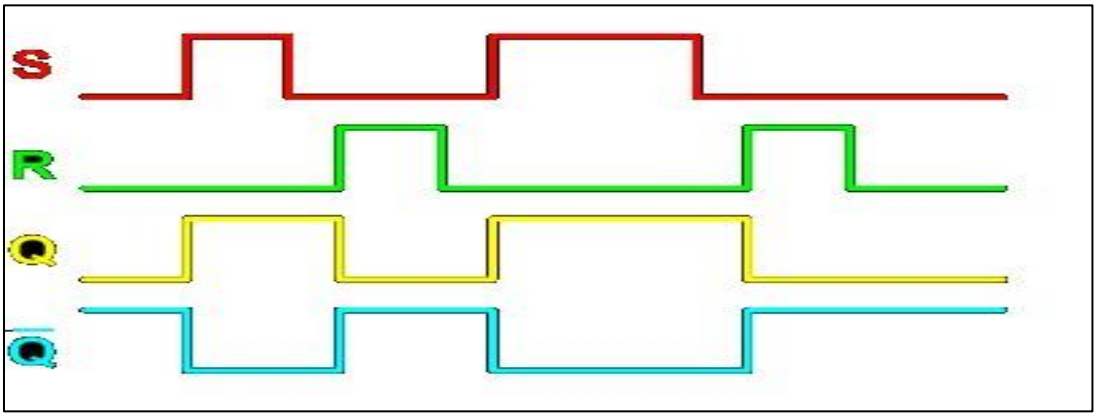
We use the term *present state* to mean the state of Q at the time the inputs are applied, and the term *next state* to mean the state of Q after the latch or flip-flop has reacted to the inputs.

Input			Output
S	R	Q	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	X
1	1	1	X



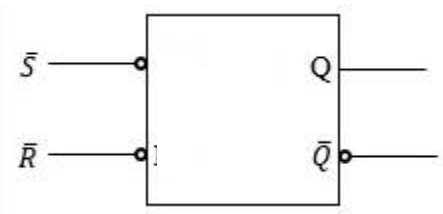
$Q_{n+1} = S + \bar{R}Q$

Example: If S and R waveforms shown below are applied to SR latch. Determine the waveform that would be observed on the output Q. Assume Q initially is '0'.

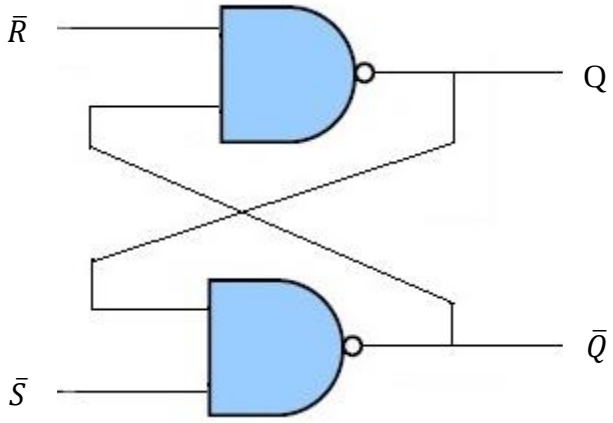


Active Low SR-Latch Input ($\bar{S}\bar{R}$-Latch):

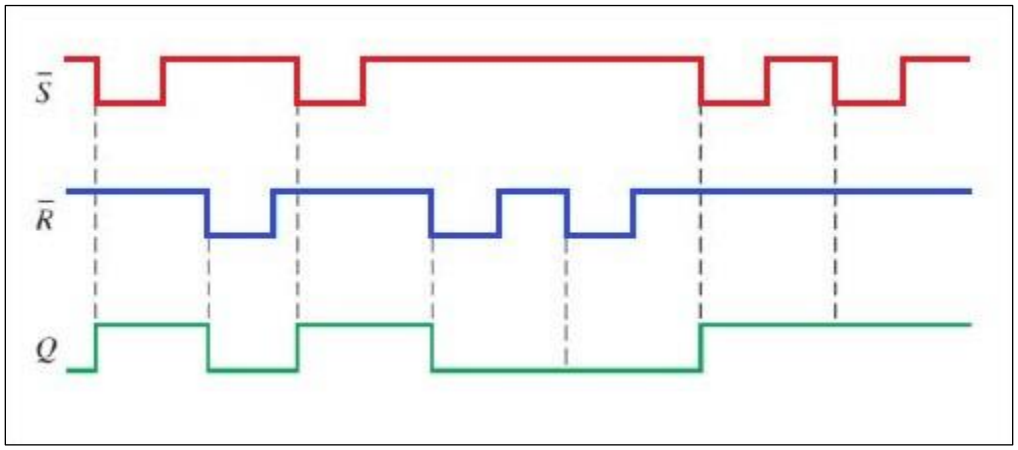
Input		Output		Comment
$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$	
0	0	X	X	Don't care (invalid)
0	1	1	0	Set
1	0	0	1	Reset
1	1	Q	$\bar{Q}$	Hold (no change)



Active Low Input

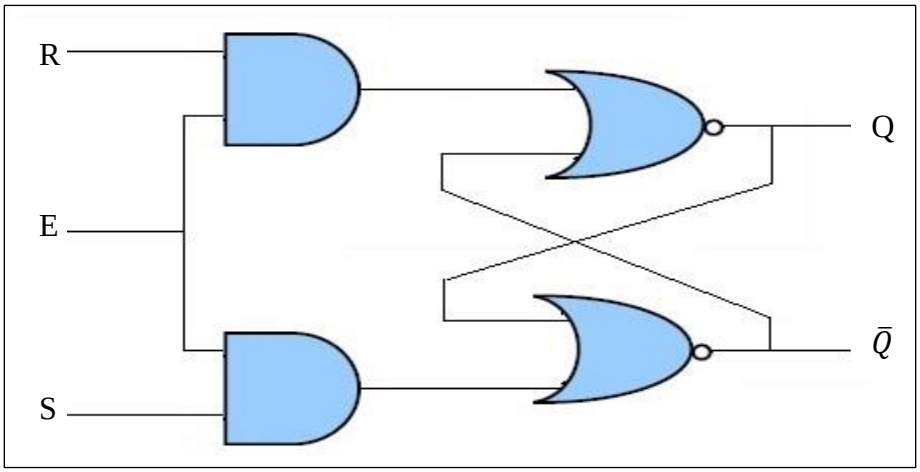


Example: If $\bar{S}$ and $\bar{R}$ waveforms shown below are applied to $\bar{R}$ $\bar{S}$ latch. Determine the waveform that would be observed on the output Q. Assume Q initially is '0'.

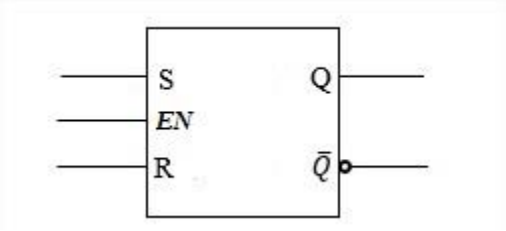


5.1.2 Gated S-R Latch

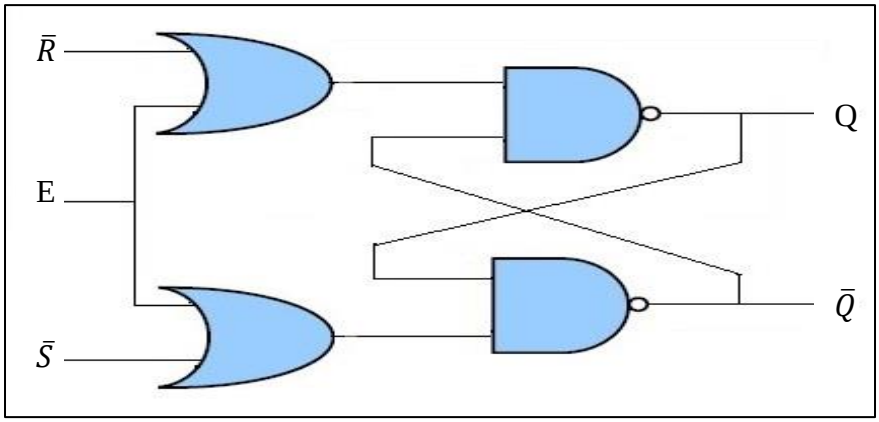
The gated latch has an additional input, called **enable (*EN*)** that must be HIGH in order for the latch to respond to the *S* and *R* inputs. This circuit is just an SR latch with the input passed through a pair of AND gates. When the input Enable = 0, without regard to the values of S and R, the latch does not change state, but keeps its current value. When the input Enable is 1, the SR latch responds to the input values of S and R as discussed above. The table below is same as that of SR Latch (active high inputs) except that it is controlled using enable signal.



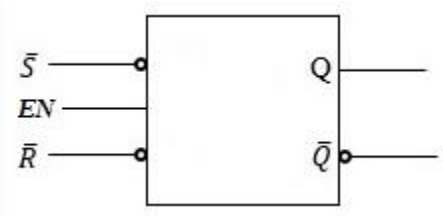
Input			Output		Comment
E	S	R	Q	$\bar{Q}$	
1	0	0	Q	$\bar{Q}$	Hold (no change)
1	0	1	0	1	Reset
1	1	0	1	0	Set
1	1	1	X	X	Don't care (invalid)
0	X	X	No	No	Hold (no change)



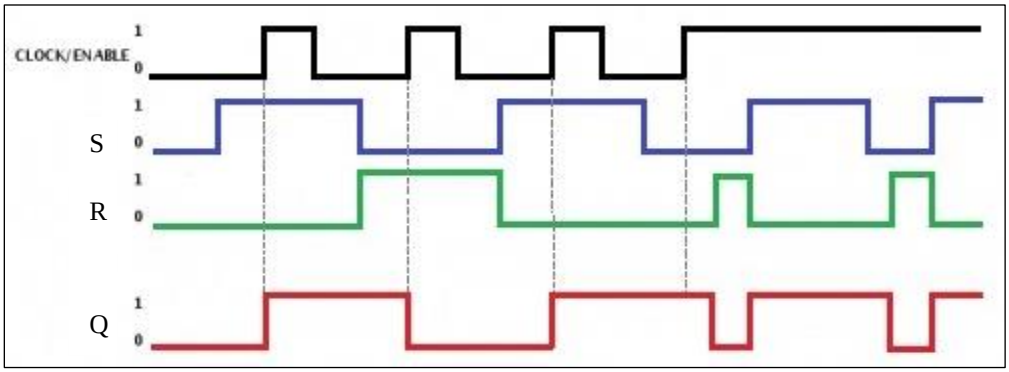
This circuit is just an SR latch with the input passed through a pair of OR gates. When the input Enable = 1, without regard to the values of $\bar{S}$ and $\bar{R}$, the latch does not change state, but keeps its current value. When the input Enable is 0, the $\bar{S}\bar{R}$ latch responds to the input values of $\bar{S}$ and $\bar{R}$ as discussed above. The table below is same as that of $\bar{S}\bar{R}$ Latch (active low inputs) except that it is controlled using enable signal.



Input			Output		Comment
E	$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$	
0	0	0	X	X	Don't care (invalid)
0	0	1	1	0	Set
0	1	0	0	1	Reset
0	1	1	Q	$\bar{Q}$	Hold (no change)
1	X	X	Q	$\bar{Q}$	Hold (no change)

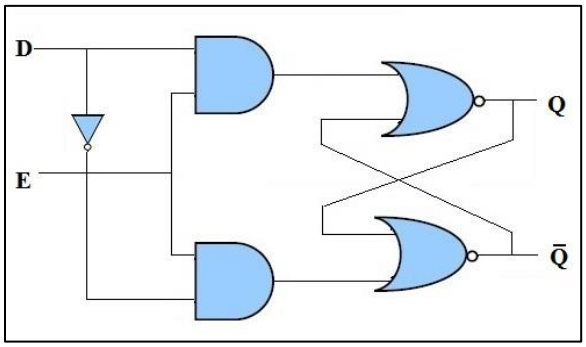


Example: If S, R and E waveforms shown below are applied to gated SR latch. Determine the waveform that would be observed on the output Q. Assume Q initially is '0'.

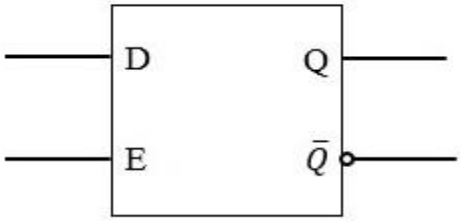


5.1.3 D latch

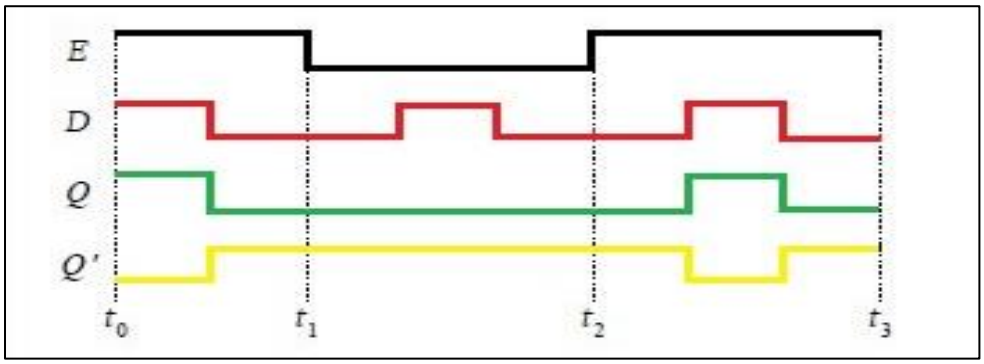
The D latch is a variation of the S-R latch but combines the S and R inputs into a single D input. SR latch requires careful design to ensure SR=11 never occurs, D latch relieves designer of that problem inserted inverter ensures R always opposite of S.

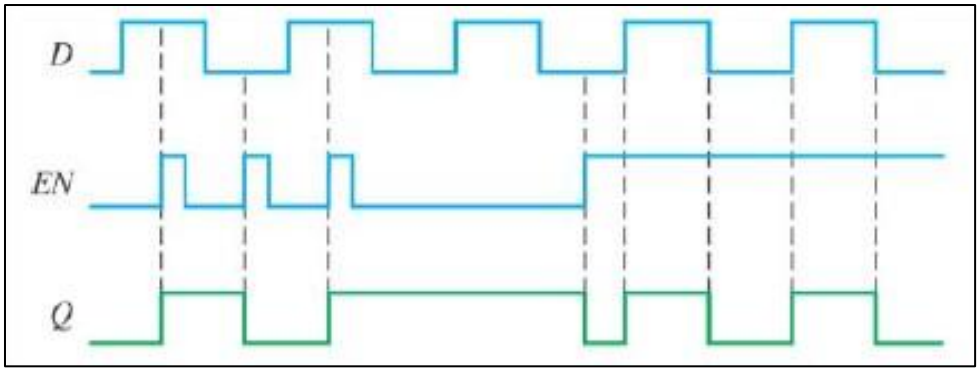


Input		Output		Comment
E	D	Q	$\bar{Q}$	
0	X	Q	$\bar{Q}$	Hold (no change)
1	0	0	1	Reset
1	1	1	0	Set



Example: Show the Q output with relation to the input signals of a gated D latch.

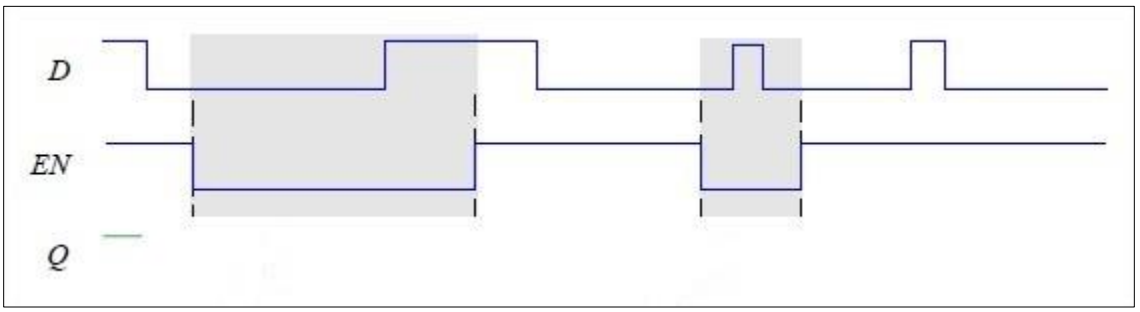




Homework: Complete the truth table.

E	D	Q	Q_n
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Homework: Determine the *Q* output for the *D* latch, given the inputs shown.



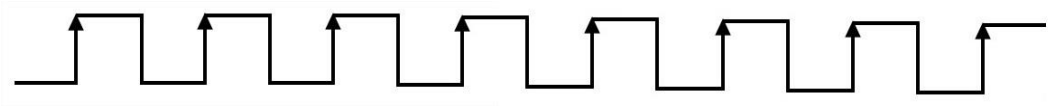
5.2 Flip flop's

Flip-flops are synchronous bistable devices (= bistable multi-vibrators). A flip-flop differs from a latch in the manner it changes states. The output changes state only at the transition of the clock (C or CLK). A flip-flop is a clocked device, in which only the clock edge determines when a new bit is entered. The active edge can be positive or negative. There are many type of latches:

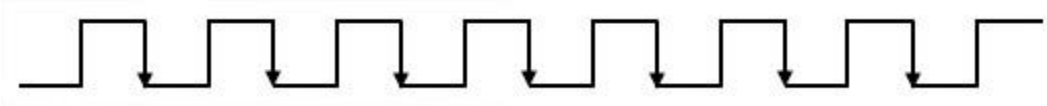
- S-R flip flop.
- D flip flop.
- J-K flip flop.
- T flip flop.

Triggering of flip flop's

For others, that changes takes place when the clock goes from 0 to 1, which is referred to as "Leading - edge triggered" or "Positive - edge triggered" or "+Ve edge".



Some flip flops change on the state takes place when the clock goes from 1 to 0, which is referred to as "Trailing - edge triggered" or "Negative - edge triggered" or " –Ve edge".

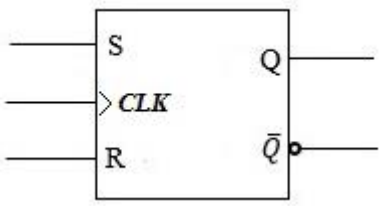


5.2.1 S-R flip flop

The S & R input of the S-R f.f are called the "synchronous input" because data on these input are transferred to the F.F output only on triggering edge of the clock pulse (+Ve edge or –Ve edge).

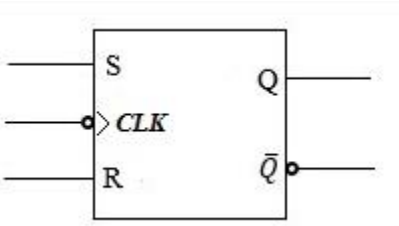
Positive *– edge triggering*

Input				Output		Comment
S	R	ClK		Q	$\bar{Q}$	
X	X	1 to 0	↓	Q	$\bar{Q}$	Hold (no change)
0	0	0 to 1	↑	Q	$\bar{Q}$	Hold (no change)
0	1	0 to 1	↑	0	1	Reset
1	0	0 to 1	↑	1	0	Set
1	1	0 to 1	↑	X	X	Don't care (invalid)

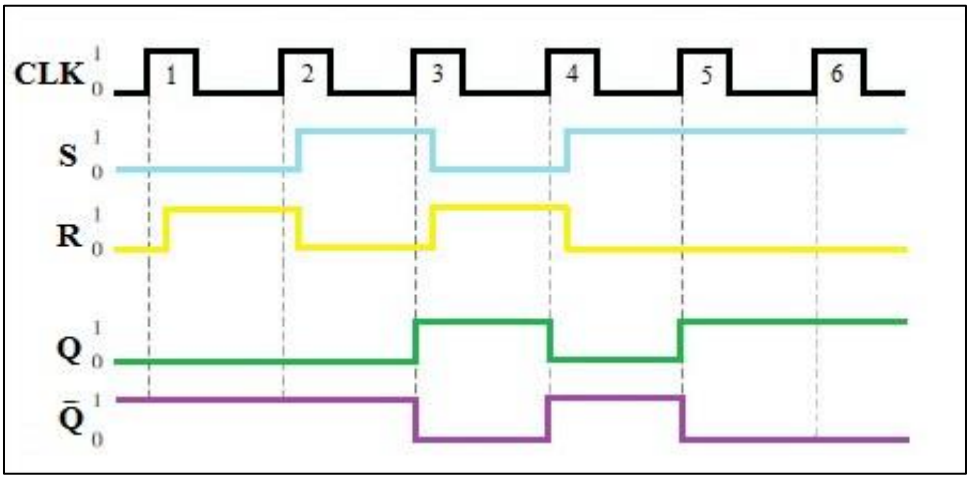


Negative *– edge triggering*

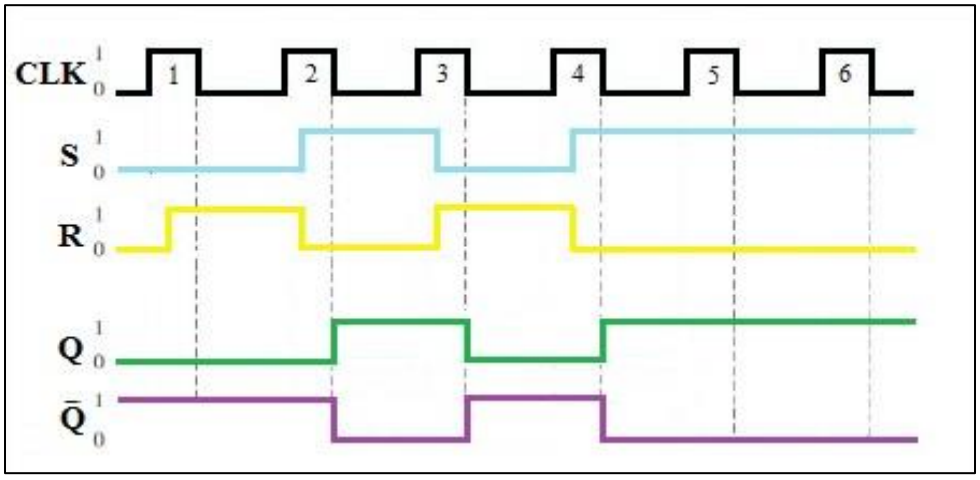
Input				Output		Comment
S	R	ClK		Q	$\bar{Q}$	
X	X	0 to 1	↑	Q	$\bar{Q}$	Hold (no change)
0	0	1 to 0	↓	Q	$\bar{Q}$	Hold (no change)
0	1	1 to 0	↓	0	1	Reset
1	0	1 to 0	↓	1	0	Set
1	1	1 to 0	↓	X	X	Don't care (invalid)



Example: Given the waveforms for S, R, & CLK flip flop input shown below, determine the output Q waveform. Assuming that the F.F is initially reset (Suppose positive edge).

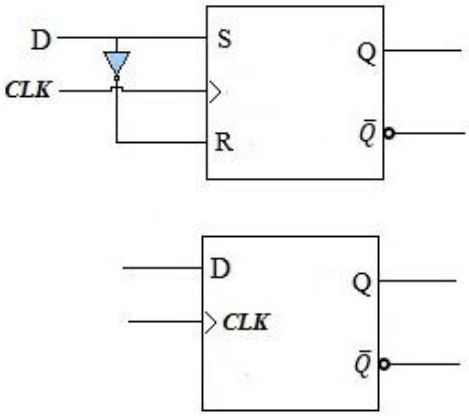


Example: Given the waveforms for S, R, & CLK flip flop input shown below, determine the output Q waveform. Assuming that the F.F is initially reset (Suppose negative edge).



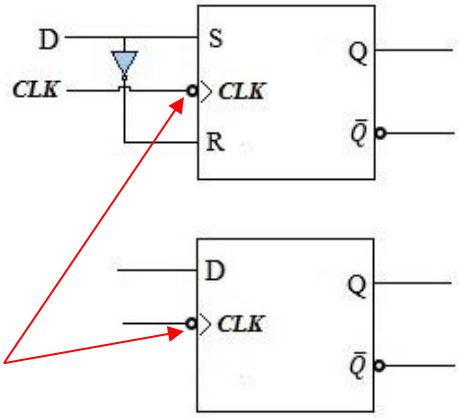
5.2.2 D flip flop

A **D flip-flop** is useful when a single data bit (1 or 0) is to be stored. An addition of an inverter to an S-R flip-flop creates a basic D flip-flop.



Input		Output		Comment
CLK	D	Q	$\bar{Q}$	
↓	X	Q	$\bar{Q}$	No Change
↑	0	0	1	Reset
↑	1	1	0	Set

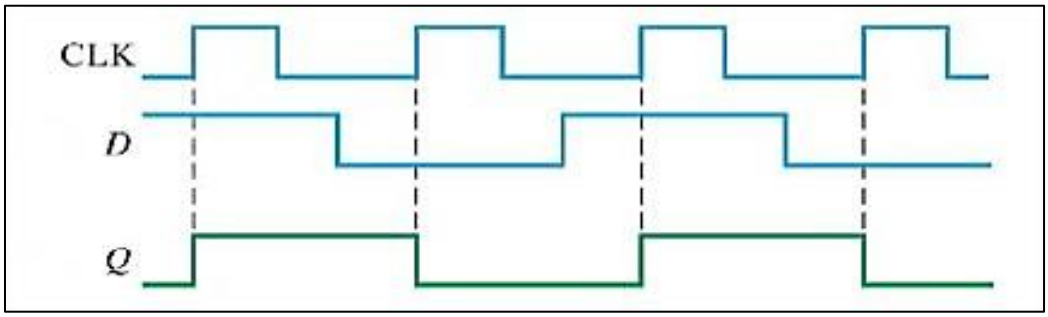
Positive Edge-Triggered



Input		Output		Comment
CLK	D	Q	$\bar{Q}$	
↑	X	Q	$\bar{Q}$	No Change
↓	0	0	1	Reset
↓	1	1	0	Set

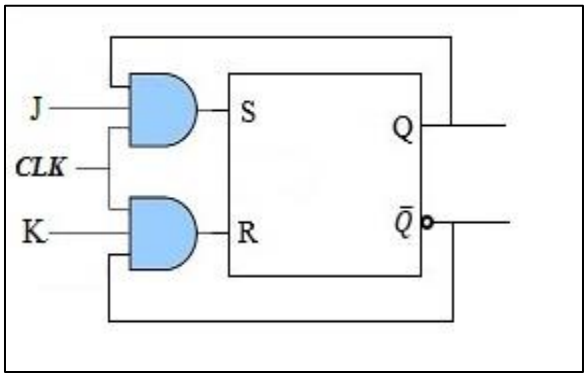
Negative-edge triggered

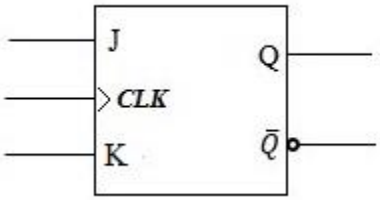
Example: Given the waveforms for D, CLK flip flop input shown below, determine the output Q waveform. (Suppose positive edge D F.F).



5.2.3 J-K flip flop

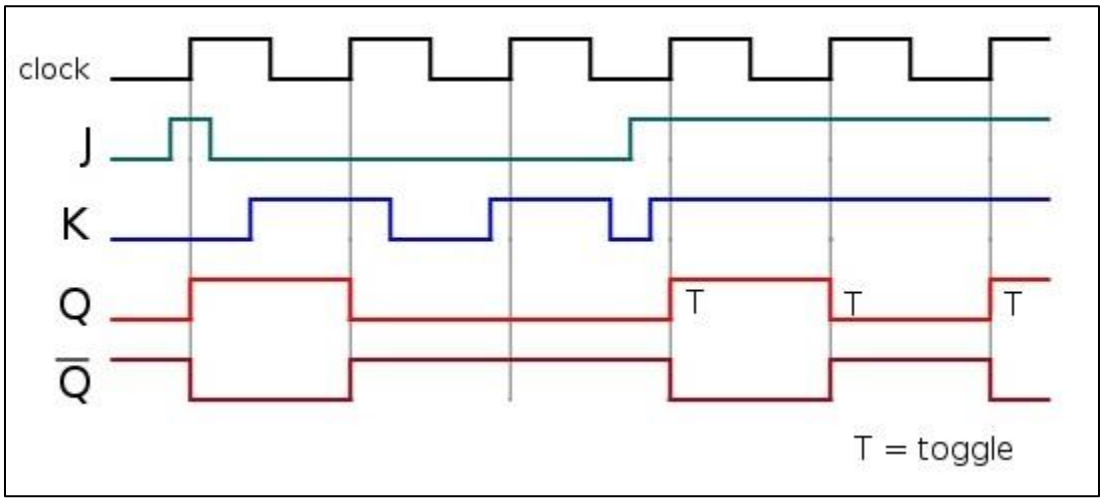
The J-K flip flop is the most widely used type of flip flop. It is are very similar to SR flip-flops. The J input is just like the S input in that when J=1, it sets the flip-flop. Similarly, the K input is like the R input where it clears the flip-flop when K=1. The only difference is when both inputs are =1, for the SR flip-flop the next state is undefined, whereas for the JK flip-flop the next state is the inverse of the current state. In other words, the JK flip-flop *toggles* its state when both inputs are =1. The circuit, truth table and the logic symbol for the JK flip-flop is shown below.



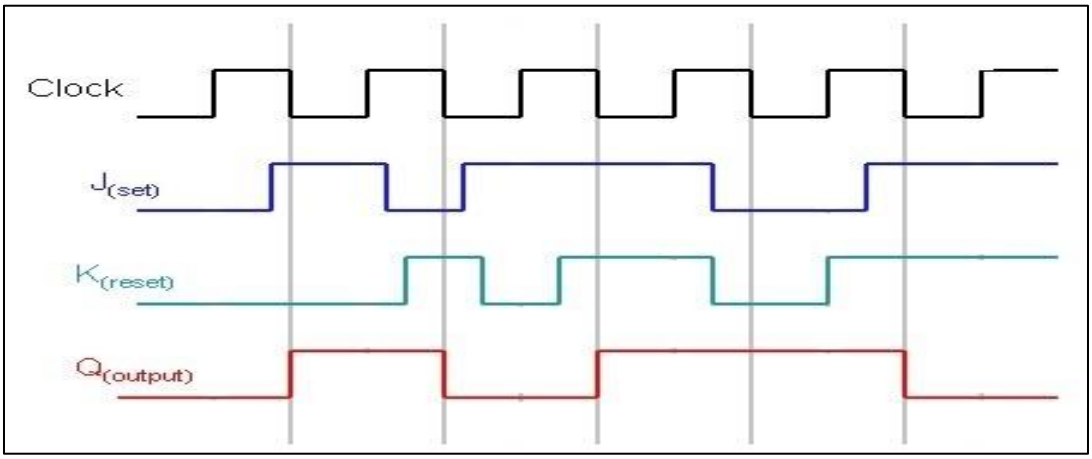


Input				Output		Comment
J	K	ClK		Q	$\bar{Q}$	
X	X	1 to 0	↓	Q	$\bar{Q}$	Hold (no change)
0	0	0 to 1	↑	Q	$\bar{Q}$	Hold (no change)
0	1	0 to 1	↑	0	1	Reset
1	0	0 to 1	↑	1	0	Set
1	1	0 to 1	↑	$\bar{Q}$	Q	Toggle

Example: Given the waveforms for J, K, & CLK flip flop input shown below, determine the output Q waveform. Assuming that the F.F is initially reset (Suppose +ve edge).

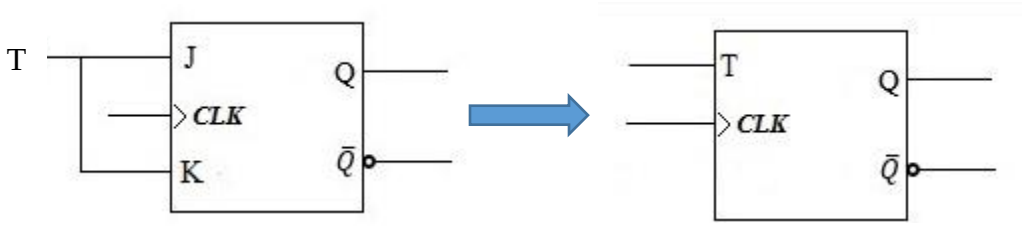


Example: Given the waveforms for J, K, & CLK flip flop input shown below, determine the output Q waveform. Assuming that the F.F is initially reset (Suppose -ve edge).



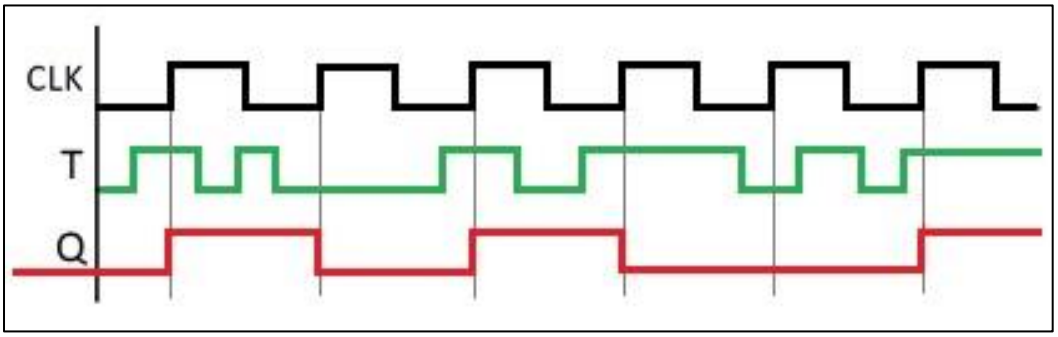
5.2.4 T flip flop

Another type of flip flop is T F.F. This is essentially a J-K FF with its J and K inputs connected together and renamed as the T input.



Input		Output	Comment
CLK	T	Q	
↓	X	Q	No Change
↑	0	Q	No Change
↑	1	$\bar{Q}$	Toggle

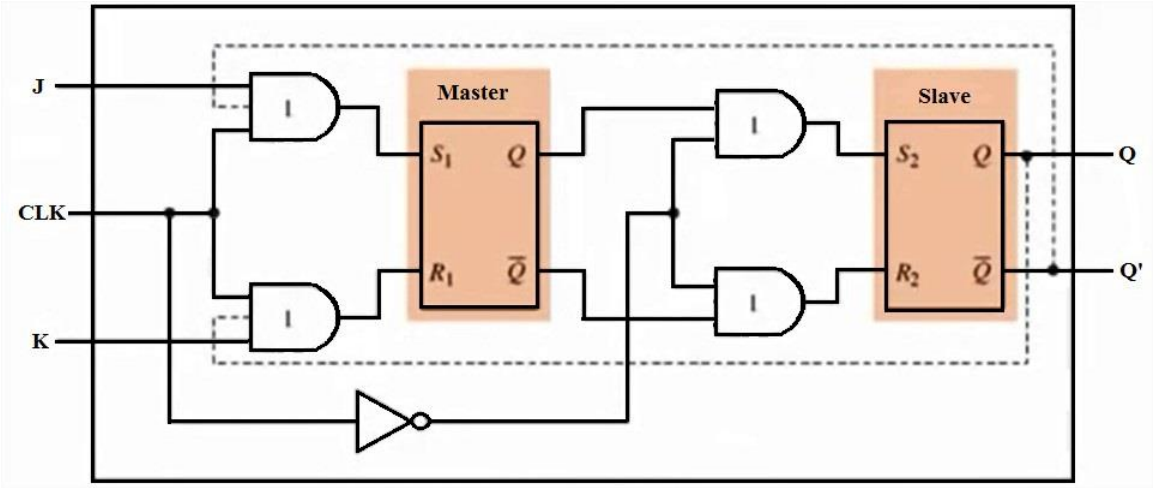
Example: Given the waveforms for T & CLK flip flop input shown below, determine the output Q waveform. Assuming that the F.F is initially reset (Suppose +ve edge).

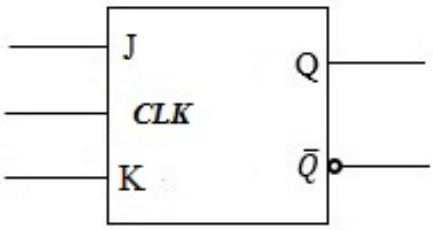


5.3 Master-Slave flip flop (Plus Triggered)

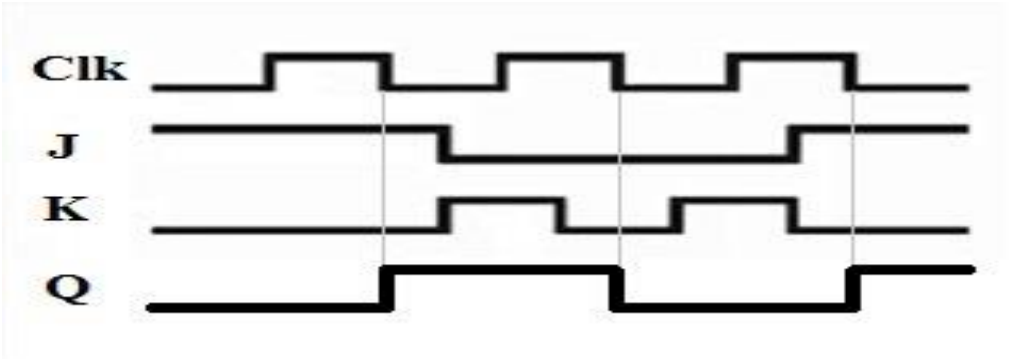
One of the most important clocked logic devices is the master-slave JK flip-flop. Master-slave JK flip-flop has memory only until another clock pulse comes along, the JK flip-flop has true memory. The term plus triggered means that the data are entered into the F.F on leading edge of the clock plus, but the output does not reflect the input state until the trailing edge of the clock plus. The input must be set up prior to clock plus leading edge, but the output is postponed until the trailing edge of the clock. A major restriction of plus-triggered F.F is that the data input must not change while the clock plus is HIGH because the F.F is sensitive to any change of input level during this time.

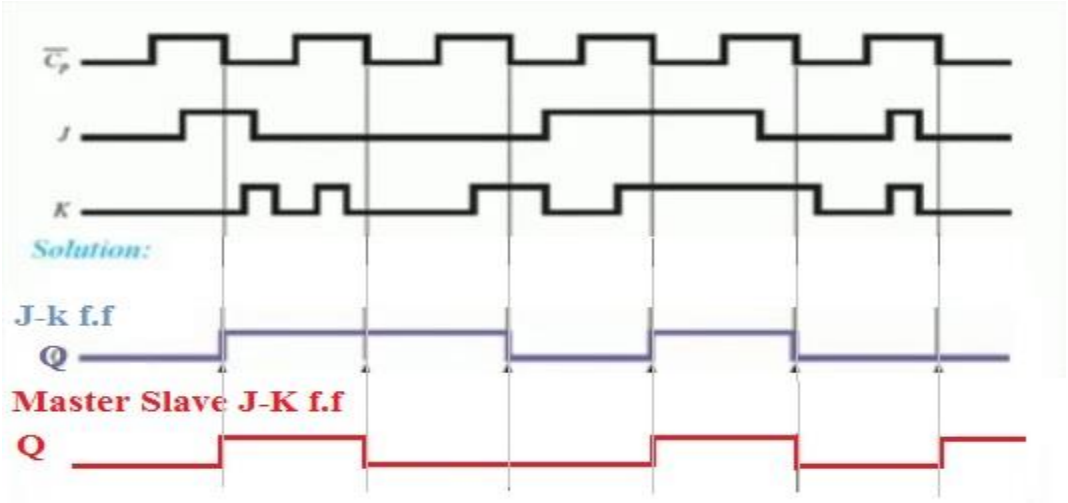
Notice that there is no dynamic input indicator ($>$) at clock input.





Example: Given the waveforms for J, K plus triggered F.F shown below, determine the output Q waveform. Assuming that the F.F is initially reset.



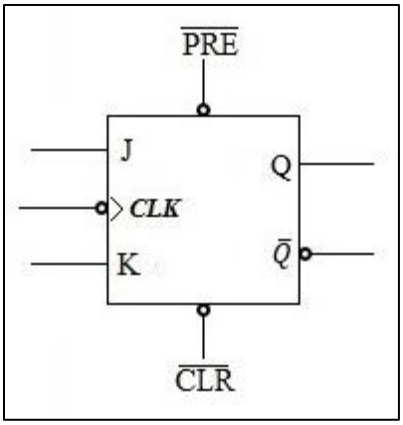


5.4 Asynchronous input of flip flop

For the flip flop's just discussed, the S-R, D, J-k, & T input are called synchronous inputs because data on these input are transferred to the F.F output only on the triggering edge of the clock pulse that is the data are transferred synchronously with the clock.

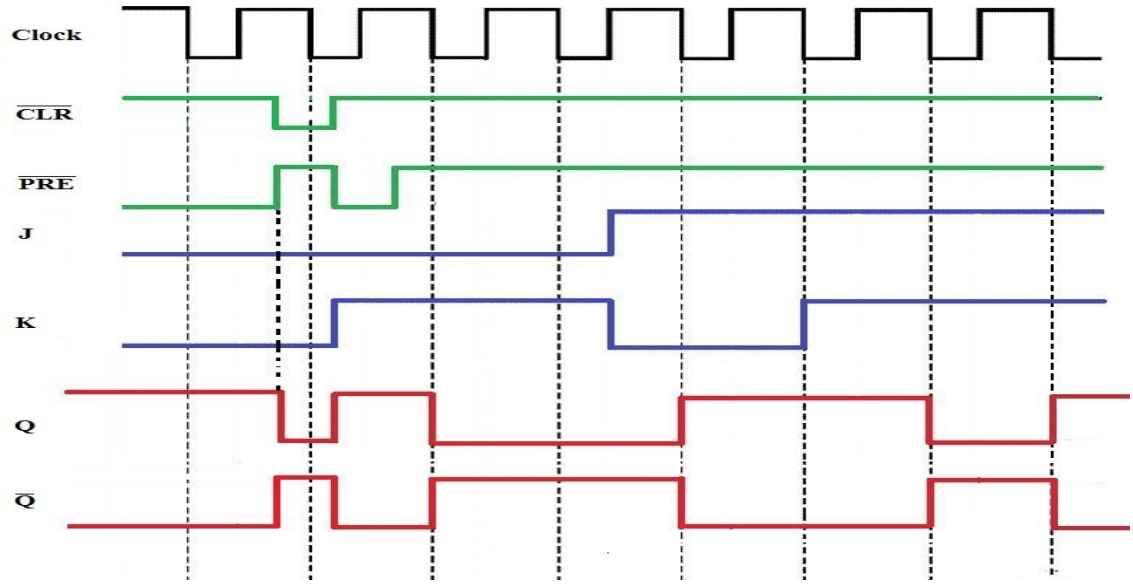
Most F.F's also have asynchronous input. These are inputs that affect the state of the F.F independent of the clock. They are normally labeled PRE (Preset) and CLR (Clear). An active level on PRE input will reset the flip flop, and an active level on CLR input will set it.

The most common F.F's with asynchronous input is the J-K F.F's.

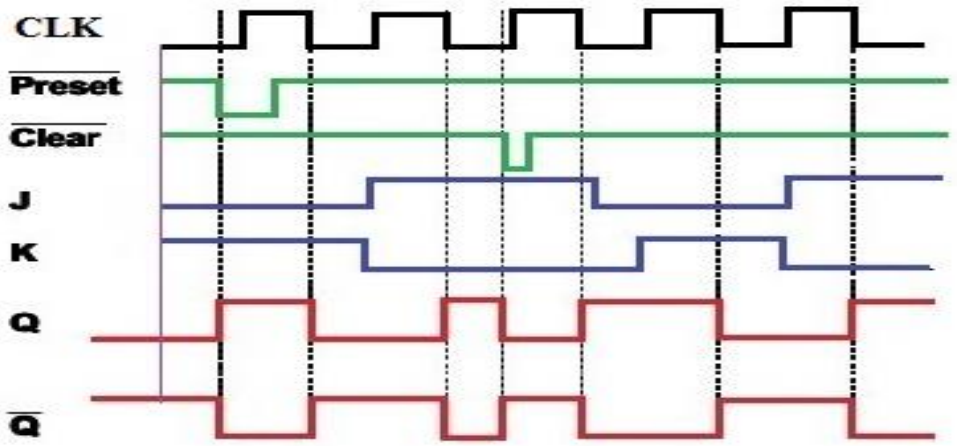


Input						Output		
$\overline{CLR}$	$\overline{PRE}$	J	K	CLK		Q	$\bar{Q}$	Comment
0	0	X	X	X		X	X	Don't Care
1	0	X	X	X		1	0	Set
0	1	X	X	X		0	1	Reset
1	1	0	0	1 to 0	↓	Q	$\bar{Q}$	No Change
1	1	0	1	1 to 0	↓	0	1	Reset
1	1	1	0	1 to 0	↓	1	0	Set
1	1	1	1	1 to 0	↓	$\bar{Q}$	Q	Toggle
1	1	X	X	0 to 1	↑	Q	$\bar{Q}$	No Change

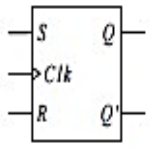
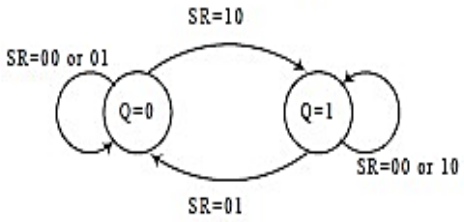
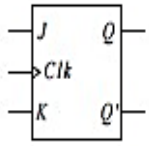
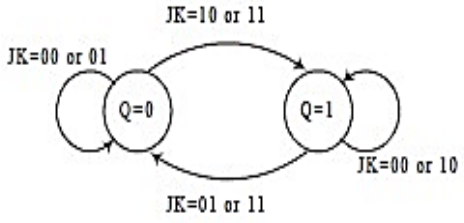
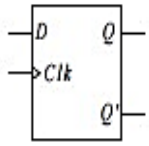
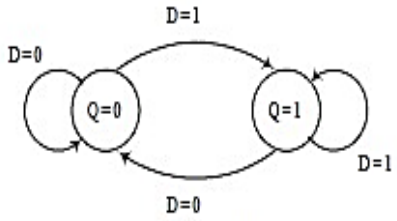
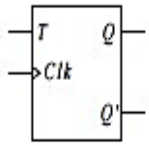
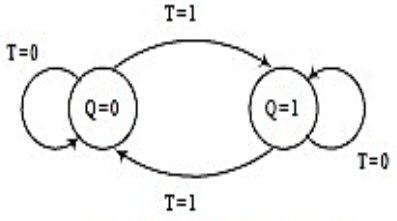
Example: Given the waveforms for J, K F.F with preset and clear input shown below, determine the output Q waveform. Assuming that the F.F is initially set (Suppose –Ve edge).



Example: Given the waveforms for J, K F.F with preset and clear input shown below, determine the output Q waveform. Assuming that the F.F is initially reset (Suppose –Ve edge).



Summary of type of flip flop behavior

Name / Symbol	Characteristic (Truth) Table	State Diagram / Characteristic Equations	Excitation Table
SR 	S R Q Qnext 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 ×	 $Q_{next} = S + R'Q$ $SR = 0$	Q Qnext S R 0 0 0 × 0 1 1 0 1 0 0 1 1 1 × 0
JK 	J K Q Qnext 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	 $Q_{next} = J'K'Q + JK' + JKQ'$ $= J'K'Q + JK'Q + JK'Q' + JKQ'$ $= K'Q(J'+J) + JQ'(K'+K)$ $= K'Q + JQ'$	Q Qnext J K 0 0 0 × 0 1 1 × 1 0 × 1 1 1 × 0
D 	D Q Qnext 0 × 0 1 × 1	 $Q_{next} = D$	Q Qnext D 0 0 0 0 1 1 1 0 0 1 1 1
T 	T Q Qnext 0 0 0 0 1 1 1 0 1 1 1 0	 $Q_{next} = TQ' + T'Q = T \oplus Q$	Q Qnext T 0 0 0 0 1 1 1 0 1 1 1 0

5.5 Sequential Circuit Design

Logic Symbol

The logic or graphical symbol describes the flip-flop's inputs and outputs, the names given to these signals, and whether they are active high or low. All the flip-flops have Q and $\bar{Q}$ as their outputs. All of them also have a CLK input. The small triangle at the clock input indicates that the circuit is a flip-flop and so it is triggered by the edge of the clock signal; if there is a circle in front, then it is the falling edge, otherwise, it is the rising edge of the clock signal.

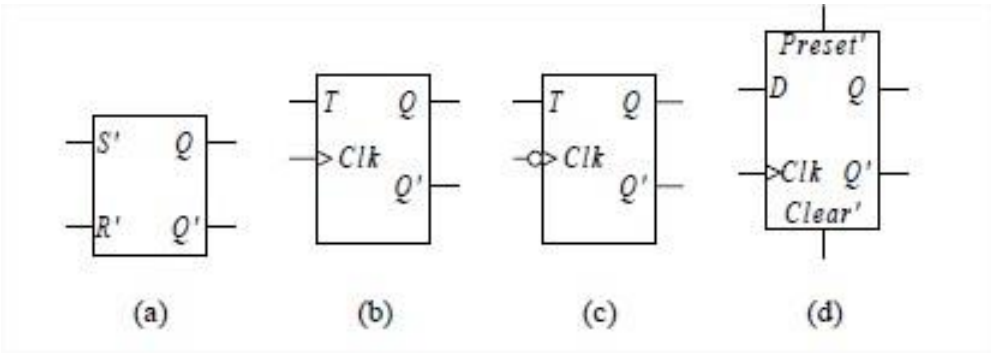


Figure 1. Various logic symbols: (a) Active low SR latch; (b) positive-edge-triggered active high T flip-flop; (c) negative-edge-triggered T flip-flop; (d) positive-edge-triggered D flip-flop with asynchronous active low preset and clear.

Without the small triangle, the circuit is a latch. In addition, the flip-flops have one or two more inputs that characterize the flip-flop and give it its name. Figure 1 shows several sample logic symbols for various memory elements.

Characteristic Table (Truth table)

The characteristic table is just the truth table but usually written in a shorter format. For example, the characteristic table for the JK flip-flop in Figure 2. The truth table, as we have seen, simply lists all possible combinations of the input signals, the current state (or content) of the flip-flop, and the next state that the flip-flop will go to at the next active edge of the clock signal. The characteristic table answers the question of what is the next state when given the inputs and the current state, and is used in the analysis of sequential circuits.

J	K	Q_{next}
0	0	Q
0	1	0
1	0	1
1	1	Q'

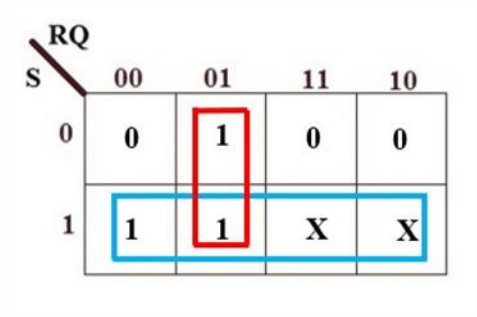
Figure 2. JK flip-flop characteristic table.

Characteristic Equation

The characteristic equation is the functional Boolean equation that is derived from the characteristic table. This equation formally describes the functional behavior of the flip-flop. Like the characteristic table, it specifies the flipflop's next state as a function of its current state and inputs. For example, the characteristic equation for the JK flipflop can be derived from the truth table as follows:

$$Qnext = J\bar{K}Q + J\bar{K}Q + J\bar{K}\bar{Q} + JK\bar{Q}$$
$$= \bar{K}Q(\bar{J}+J) + J\bar{Q}(\bar{K}+K)$$
$$= \bar{K}Q + J\bar{Q}$$

The characteristic equation can also be obtained from the truth table using the K-map method as follows for the SR flip-flop:



Thus, the characteristic equation for the SR flip-flop is

$$Q_{next} = S + \bar{R}Q.$$

State Diagram

A state diagram is a graph that shows the flip-flop's operations in terms of how it transitions from one state to another.

A. A state is represented by a "Circle".
B. Transition between states is indicated by directed lines connecting the circle.
C. Binary number inside each circle identifies the state of each circle.
D. Directed lines are labeled with two binary numbers separated by a "/ ". The input value that causes the state transition is labeled first, the number after the symbol / gives the value of the output during the present state.

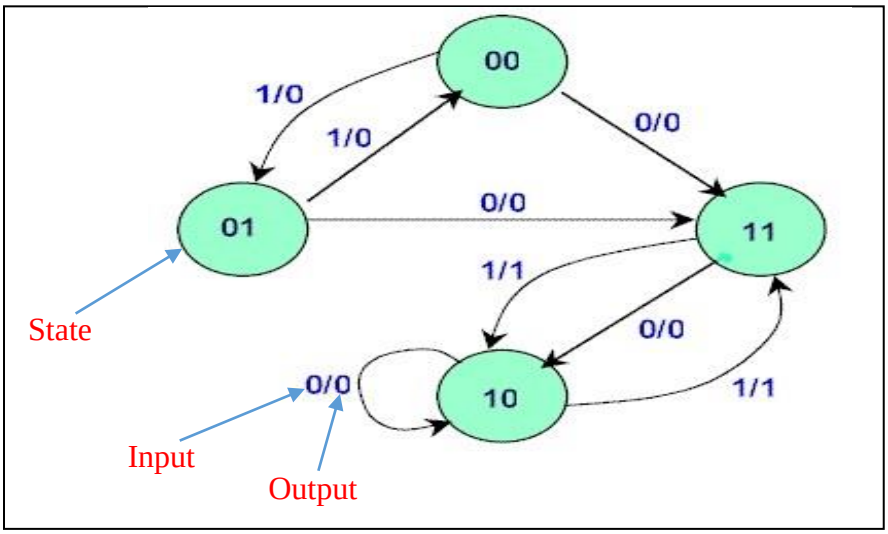


Figure 3. State diagram for the SR flip-flop.

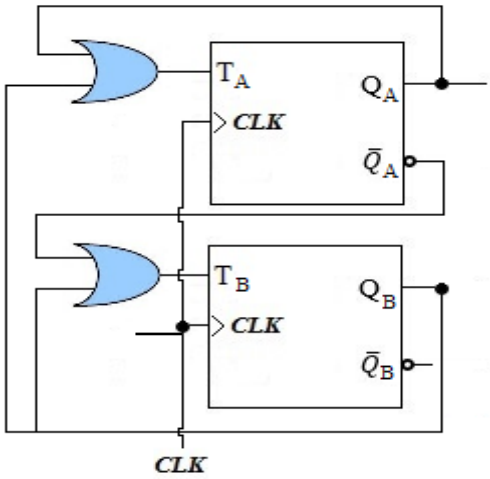
Excitation Table

The excitation table gives the value of the flip-flop's inputs that are necessary to change the flip-flop's current state to the desired next state at the next active edge of the clock signal. The excitation table answers the question of what should the inputs be when given the current state that the flip-flop is in and the next state that we want the flip flop to go to. This table is used in the synthesis of sequential circuits.

Q_n	Q_{n+1}	S	R	J	K	D	T
0	0	0	X	0	X	0	0
0	1	1	0	1	X	1	1
1	0	0	1	X	1	0	1
1	1	X	0	X	0	1	0

Figure 4. Flip-flop excitation table.

Example: For the sequential circuit below drive the truth table and start diagram.



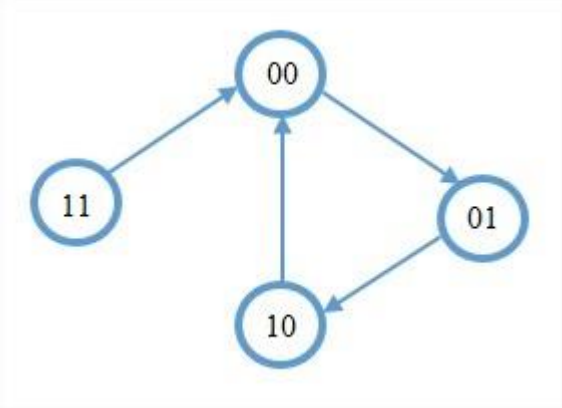
Solution:

STEP 1: First we derive the Boolean expressions for the inputs of each flip-flops in the schematic, $T_A = QA + QB$, $T_B = \overline{QA} + QB$

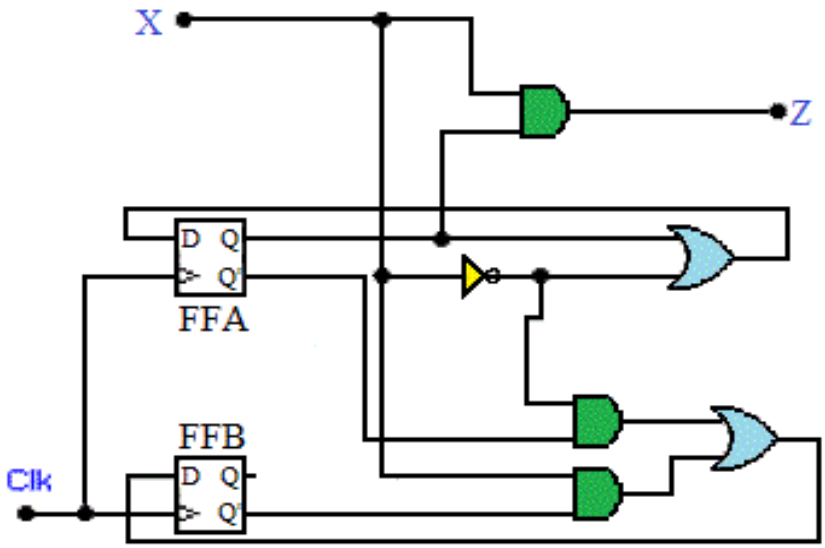
STEP 2: Now convert these next-state equations into tabular form called the next-state table.

Input		Output			
Q_A	Q_B	T_A	T_B	Q_{A+1}	Q_{B+1}
0	0	0	1	0	1
0	1	1	1	1	0
1	0	1	0	0	0
1	1	1	1	0	0

STEP 3: The state diagram is generated directly from the next-state table,



Example: For the sequential circuit below drive the truth table and start diagram.



Solution:

STEP 1: First we derive the Boolean expressions for the inputs of each flip-flops in the schematic, in terms of external input X and the flip-flop outputs Q_A and Q_B. Since there are two D flip-flops in this example, we derive two expressions for D_A and D_B:

$$Z = XQ_A$$

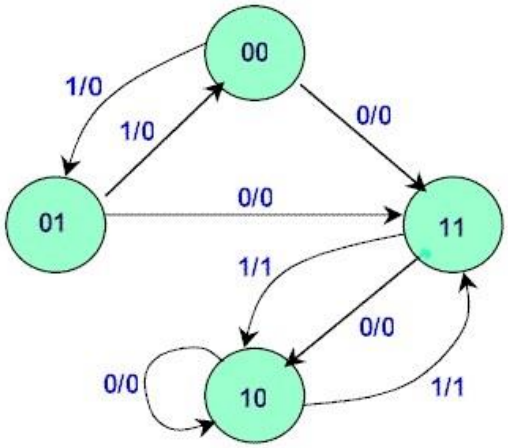
$$D_A = \bar{X} + Q_A$$

$$D_B = X\bar{Q}_B + \bar{X}\,\bar{Q}_A$$

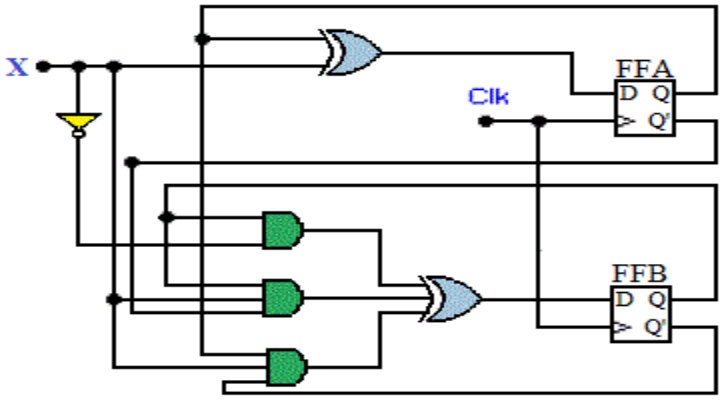
STEP 2: Now convert these next-state equations into tabular form called the next-state table.

Present State			Next State				Output
X	Q_A	Q_B	D_A	D_B	Q_{A+1}	Q_{B+1}	Z
0	0	0	1	1	1	1	0
0	0	1	1	1	1	1	0
0	1	0	1	0	1	0	0
0	1	1	1	0	1	0	0
1	0	0	0	1	0	1	0
1	0	1	0	0	0	0	0
1	1	0	1	1	1	1	1
1	1	1	1	0	1	0	1

STEP 3: The state diagram is generated directly from the next-state table,



Example: Derive the state table and state diagram for the sequential circuit shown below:



STEP 1:

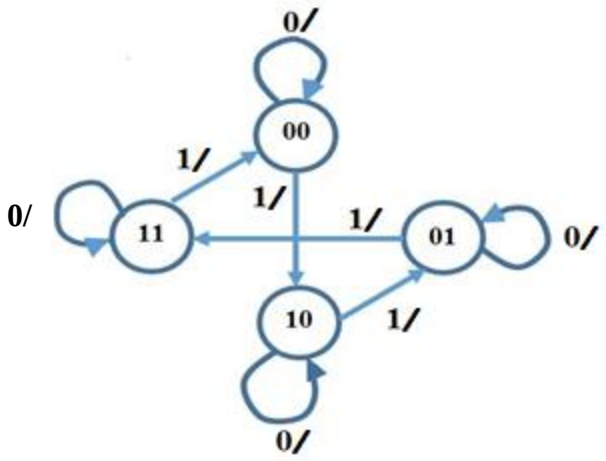
$D_A = X \oplus Q_A$

$D_B = \bar{X}\, Q_B + X\, \bar{Q}_B\, Q_A + X\, Q_B\, \bar{Q}_A$

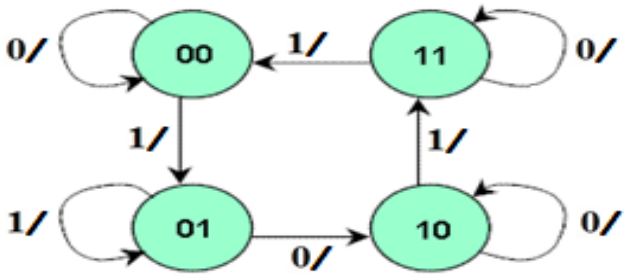
STEP 2: Now convert these next-state equations into tabular form called the next-state table.

Present State			Next State			
X	Q_A	Q_B	D_A	D_B	Q_{A+1}	Q_{B+1}
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	1	0	1	0
0	1	1	1	1	1	1
1	0	0	1	0	1	0
1	0	1	1	1	1	1
1	1	0	0	1	0	1
1	1	1	0	0	0	0

STEP 3: The state diagram is generated directly from the next-state table,



Example: Design a synchronous sequential circuit whose state diagram is shown below. Using J-K flip-flop.



Solution:

From the state diagram, we can generate the state table shown below. Note that there is no output section for this circuit. Two flip-flops are needed to represent the four states and are designated Q_AQ_B. The input variable is labelled x.

Present State			Next State	
X	Q_A	Q_B	Q_{A+1}	Q_{B+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	1	1
1	1	1	0	0

Remember, the excitable for the JK flip-flop was derive in Table below.

Q	$Q_{(next)}$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

We shall now derive the excitation table and the combinational structure. The table is now arranged in a different form shown below, where the present state and input variables are arranged in the form of a truth table.

Present State Q_A Q_B	Input X	Next State Q_A Q_B	Flip Flop Input J_A K_A	J_B K_B
0 0	0	0 0	0 X	0 X
0 0	1	0 1	0 X	1 X
0 1	0	1 0	1 X	X 1
0 1	1	0 1	0 X	X 0
1 0	0	1 0	X 0	0 X
1 0	1	1 1	X 0	1 X
1 1	0	1 1	X 0	X 0
1 1	1	0 0	X 1	X 1

Present State X Q_A Q_B	Next State Q_{A+1} Q_{B+1}	Flip Flop Input J_A K_A	J_B K_B
0 0 0	0 0	0 X	0 X
0 0 1	1 0	1 X	X 1
0 1 0	1 0	X 0	0 X
0 1 1	1 1	X 0	X 0
1 0 0	0 1	0 X	1 X
1 0 1	0 1	0 X	X 0
1 1 0	1 1	X 0	1 X
1 1 1	0 0	X 1	X 1

The simplified Boolean functions for the combinational circuit can now be derived. The input variables are Q_A, Q_B, and x; the output are the variables J_A, K_A, J_B and K_B. The information from the truth table is plotted on the Karnaugh maps shown below.

X \ $Q_A Q_B$	00	01	11	10
0	0	1	X	X
1	0	0	X	X

$J_A = \bar{X}\, Q_B$

X \ $Q_A Q_B$	00	01	11	10
0	X	X	0	0
1	X	X	1	0

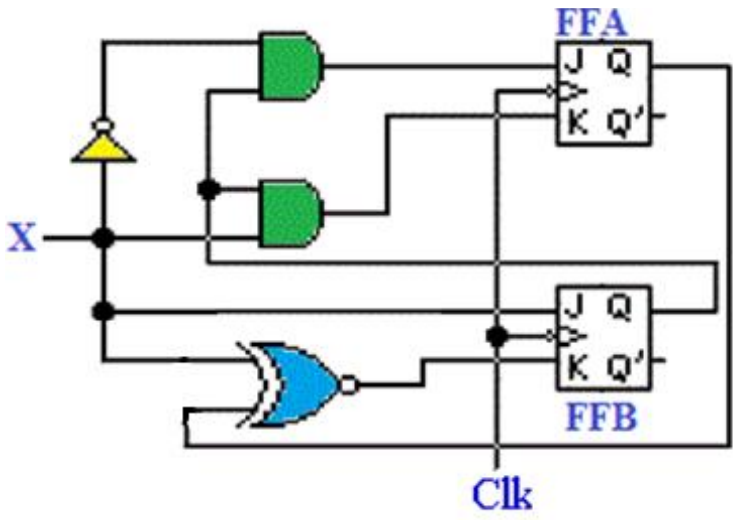
$K_A = X\, Q_B$

X \ $Q_A Q_B$	00	01	11	10
0	0	X	X	0
1	1	X	X	1

$J_B = X$

X \ $Q_A Q_B$	00	01	11	10
0	X	1	0	X
1	X	0	1	X

$K_B = \bar{X}\, \bar{Q}_A + X\, Q_A = \overline{QA \oplus X}$



Example: Design a synchronous sequential circuit whose state table are shown below. Using J-K flip-flop.

Input X	Present State Q_A Q_B	Next State Q_{A+1} Q_{B+1}	Output Z
0	0 0	0 0	0
0	0 1	0 0	0
0	1 0	1 1	0
0	1 1	0 0	0
1	0 0	0 1	0
1	0 1	1 0	0
1	1 0	1 0	0
1	1 1	0 1	1

Solution:

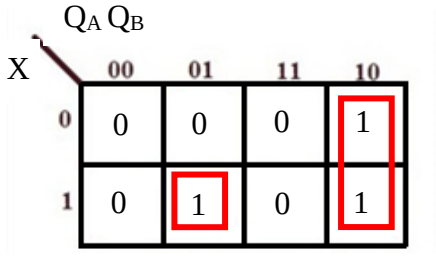
The excitable for the D flip-flop was derive in Table below.

Q	Q(next)	D
0	0	0
0	1	1
1	0	0
1	1	1

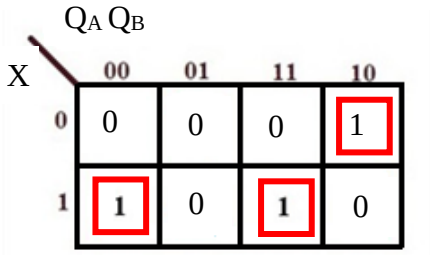
Next step is to drive the excitation table for the design circuit, which is shown below. The output of the circuit is labelled Z.

Input X	Present State Q_A Q_B	Next State Q_A Q_B	Flip Flop Input D_A D_B	Output Z
0	0 0	0 0	0 0	0
0	0 1	0 0	0 0	0
0	1 0	1 1	1 1	0
0	1 1	0 0	0 0	0
1	0 0	0 1	0 1	0
1	0 1	1 0	1 0	0
1	1 0	1 0	1 0	0
1	1 1	0 1	0 1	1

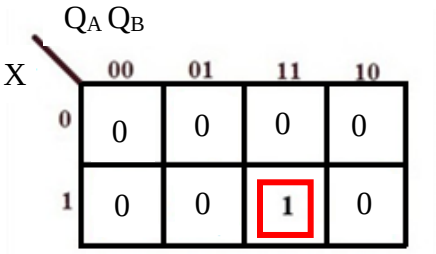
Now plot the flip-flop inputs and output function on the K-map to derive the Boolean expression,



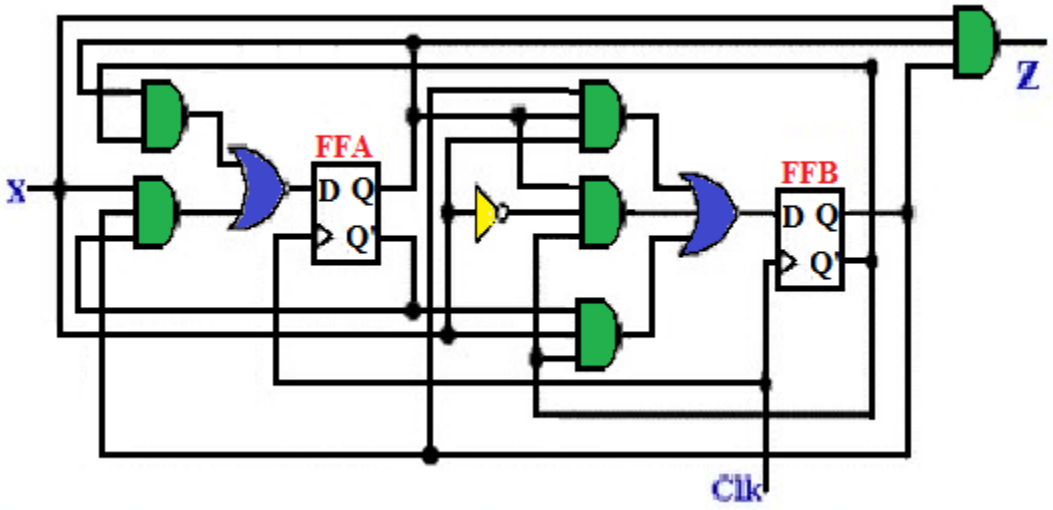
$$J_A = X\,\bar{Q}_A\,Q_B + Q_A\,\bar{Q}_B$$



$$J_B = X\,\bar{Q}_A\,\bar{Q}_B + \bar{X}\,Q_A\,\bar{Q}_B + X\,Q_A\,Q_B$$



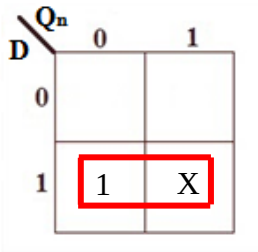
$$Z = X\,Q_A\,Q_B$$



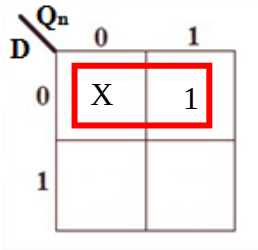
5.6 Conversion of flip flop

1. Conversion of S-R to D flip flop

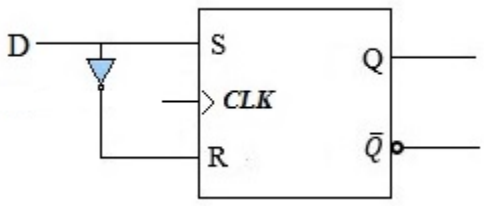
Input		Output		
D	Q_n	Q_{n+1}	S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0



S=D

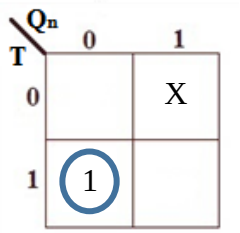


R=$\overline{D}$

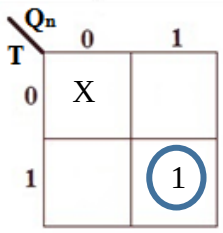


2. Conversion of S-R to T flip flop

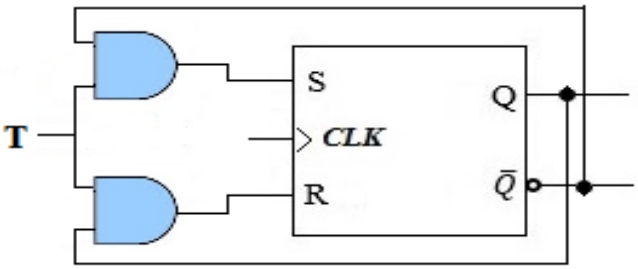
Input		Output		
T	Q_n	Q_{n+1}	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1



S=T $\overline{Qn}$

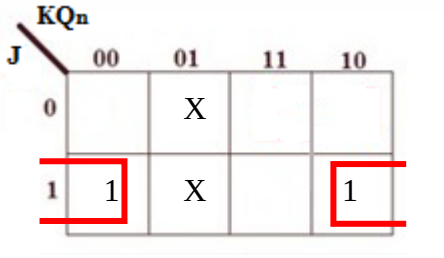


R= T Q_n

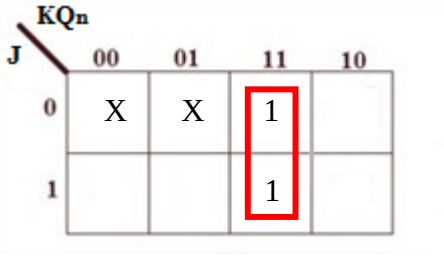


3. Conversion of S-R to J-K flip flop

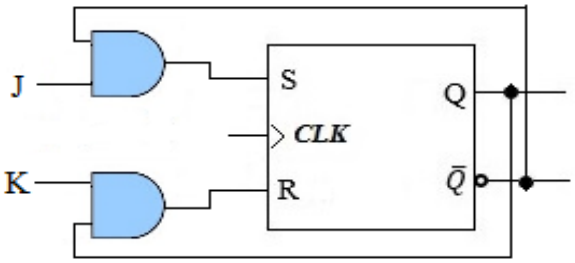
Input			Output		
J	K	Qn	Qn+1	S	R
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	1	0	X
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	X	0
1	1	0	1	1	0
1	1	1	0	0	1



S= $J\overline{Qn}$

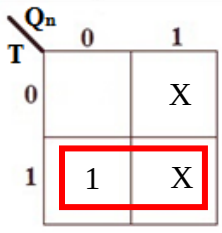


R= K Q_n

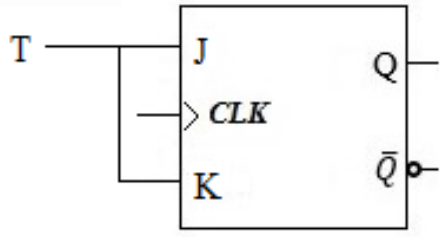


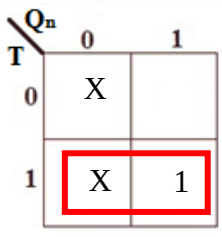
4. Conversion of J-K to T flip flop

Input		Output		
T	Q_n	Q_{n+1}	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1



J= T

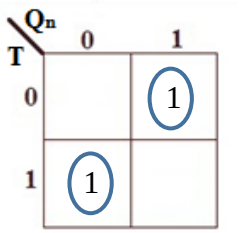




K= T

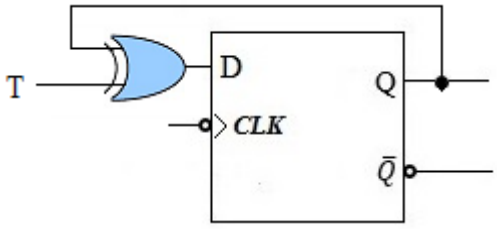
5. Conversion of D to T flip flop

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



$$D = \overline{Qn}T + Qn\bar{T}$$

$$D = Qn \oplus T$$



Homework: Convert the following flip flop

- T to D flip flop
- J-K to S-R flip flop
- J-K to D flip flop