Sequential Logic

Sequential Circuit: Till now, we study combinational circuits in which the outputs at any instant of time are entirely dependent upon the inputs present at that time. Although every digital system is likely to have combinational circuits, most systems encountered in practice also include memory elements, which require that the system be described in terms of sequential logic.

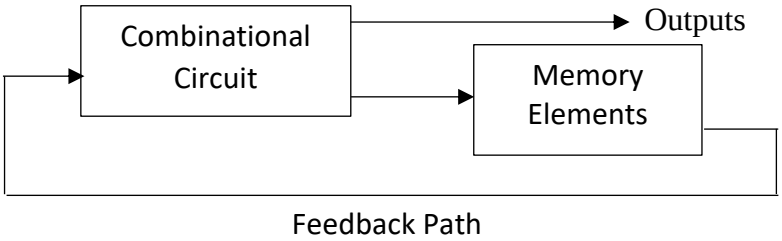


Fig: Block diagram of sequential circuit

Memory elements are devices capable of storing binary information within them. The binary information stored in the memory elements at any given time defines the state of the sequential circuit.

Block diagram shows external outputs in a sequential circuit are a function not only of external inputs, but also of the present state of the memory elements. Thus, a sequential circuit is specified by a time sequence of inputs, outputs, and internal states.

The logic circuits whose outputs at any instant of time depend not only on the present inputs but also on past outputs are known as sequential circuits. It consists of a combinational circuit to which storage elements are connected to form a feedback path. Examples of sequential circuits are: flip-flops, counters, registers etc.

Types of sequential circuits:

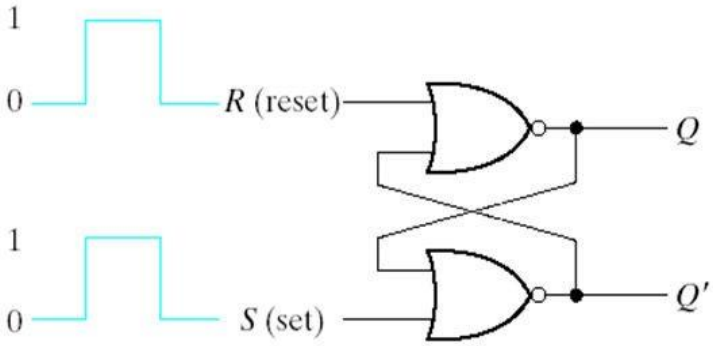
i) **Synchronous Sequential Circuit:** It is a system whose behavior can be defined from the knowledge of its signals at discrete instant of time. The change in input signal can effect memory element upon activation of clock signal. In this circuit memory elements are clocked flip-flops (i.e. it works only when clock signal is 1). This circuit can operate slower.
ii) Asynchronous Sequential circuit: It is a system whose behavior depends in the order in which its input signals change and can be affected at any instant of time. In this circuit memory elements are either unlocked flip-flops or time delay elements (i.e. it works either clock signal is 0 or 1). This circuit can operate faster.

Flip-Flops: The memory elements used in clocked sequential circuits are called flip-flops. These circuits are binary cells capable of storing one bit of information. A flip-flop circuit has two outputs, one for the normal value and one for the complement value of the bit stored in it. Binary information can enter a flip-flop in a variety of ways, a fact that gives rise to different types of flip-flops. A flip-flop circuit can maintain binary state indefinitely (as long as power is delivered to the circuit) until directed by an input signal to switch states.

A flip-flop is a sequential circuit which is properly known as basic digital electronic circuit. A flip-flop stores 1 bit, therefore, it is called as 1 bit memory cell. It has two stable states: logic 1 and logic 0. It can flip from one state to another and then flop back, so it is called flip-flop and also known as a bi-stable multi-vibrator. The outputs of circuit (Q and $\overline{Q}$) will always be complementary. This means that if Q = 0, then $\overline{Q}$ = 1 and vice-versa. They will never be equal; Q and $\overline{Q}$ = 0 or 1 is an invalid state. If Q=1, $\overline{Q}$=0, it called 1 or SET state. . If Q=0, $\overline{Q}$=1, it called 0 or RESET state.

Basic flip-flop circuit (direct-coupled RS flip-flop or SR latch): A flip-flop circuit can be constructed from two NAND gates or two NOR gates. These constructions are shown in the logic diagrams below. Each circuit forms a basic flip-flop upon which other more complicated types can be built. The cross-coupled connection from the output of one gate to the input of the other gate constitutes a feedback path. For this reason, the circuits are classified as asynchronous sequential circuits. Each flip-flop has two outputs, Q and Q', and two inputs, set and reset.

Basic flip-flop circuit: SR Latch with NOR Gates:



(a) Logic diagram

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

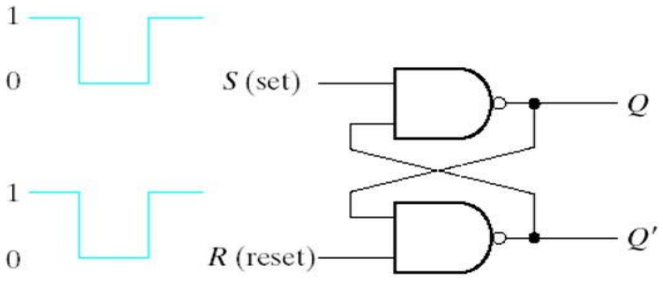
(b) Function table

Fig. *SR* Latch with NOR Gates

S	R	Q	Q`	Action
0	0	0	1	No change(Memory)
0	1	0	1	Reset
1	0	1	0	Set
1	1	0	0	Invalid

- Output of a NOR gate is 0 if any input is 1, and that the output is 1 only when all inputs are 0.
- First, assume that the set input is 1 and the reset input is 0. Since gate-2 has an input of 1, its output Q' must be 0, which puts both inputs of gate-1 at 0, so that output Q is 1.
- When the set input is returned to 0, the outputs remain the same i.e. output Q' stay at 0, which leaves both inputs of gate-1 at 0, so that output Q is 1. Similarly, 1 in the reset input changes output Q to 0 and Q' to 1.
- When the reset input returns to 0, the outputs do not change. When a 1 is applied to both the set and the reset inputs, both Q and Q' outputs go to 0. This condition violates the fact that outputs Q and Q' are the complements of each other.

Basic flip-flop circuit: SR Latch with NAND Gates:



(a) Logic diagram

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

(b) Function table

Fig. *SR* Latch with NAND Gates

S	R	Q	Q`	Action
0	0	1	1	Invalid
0	1	1	0	Set
1	0	0	1	Reset
1	1	1	0	No change(Memory)

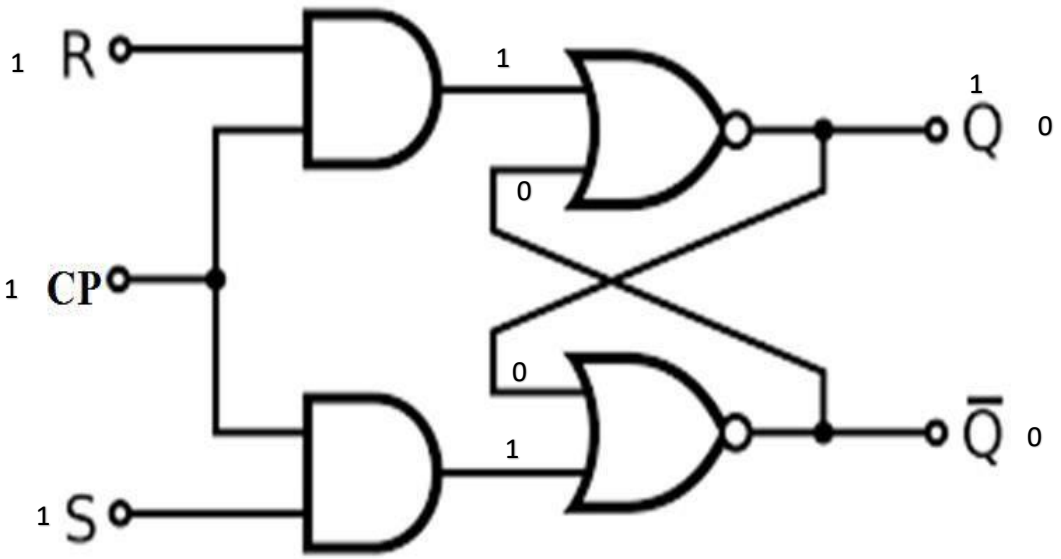
The NAND basic flip-flop circuit operates with both inputs normally at 1 unless the state of the flip-flop, has to be changed.

- The application of a momentary 0 to the set input causes output Q to go to 1 and Q' to go to 0, thus putting the flip-flop into the set state
- After the set input returns to 1, a momentary 0 to the reset input causes a transition to the clear state.
- When both inputs go to 0, both outputs go to 1- a condition avoided in normal flip-flop operation.

Clocked RS Flip-Flop: NOR gate

- It consists of a basic NOR flip-flop circuit and two additional AND gates along with clock pulse (CP) input.
- The outputs of the two AND gates remain at 0 as long as CP input is 0, regardless of the S and R input values. When the clock pulse goes to 1, information from S and R inputs is allowed to reach the basic flip-flop. The pulse input acts as an enable signal for the other two inputs.
- Set state: S = 1, *R* = 0, and *CP* = 1.
- Reset state: S = 0, *R* = 1, and *CP* = 1.
- Invalid condition: S = 1, and R = 1.
- In either case, when *CP* returns to 0, the circuit remains in its previous state. When *CP* = 1 and both the S and *R* inputs are equal to 0, the state of the circuit does not change.

Logic Diagram



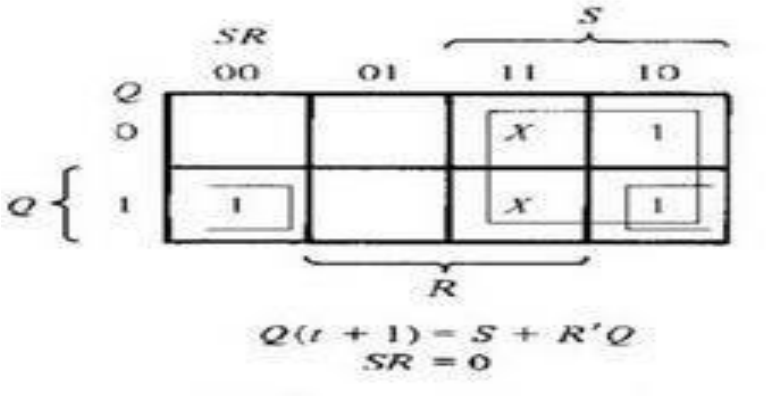
Characteristic table

Q	S	R	Q(t + 1)
0	0	0	0 (Previous State/No change)
0	0	1	0 (Reset)
0	1	0	1 (Set)
0	1	1	Indeterminate/Invalid Condition
1	0	0	1 (Previous State/No change)
1	0	1	0 (Reset)
1	1	0	1 (Set)
1	1	1	Indeterminate/Invalid Condition

S	R	CLK	Q(t+1)	Comments
0	0	X	Q(t)	No change
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	?	Invalid

Note: Apply Previous state, if either S = 0 or R = 0. I.e. when one cannot decide, what will be the output?

Characteristic equation: The characteristic equation of the flip-flop specifies the value of the next state as a function of the present and the input.

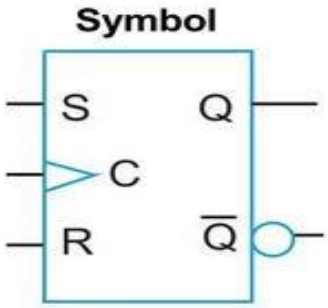


- **Characteristic Equation**

  $Q(t+1) = S + \overline{R}\,Q(t)$

  $S\,R = 0$ (S and R cannot be 1 simultaneously)

Graphic Symbol:



The graphic symbol of the RS flip-flop consists of a rectangular-shape block with inputs S, R, and C. The outputs are Q and Q', where Q' is the complement of Q (except in the indeterminate state).

Clocked RS Flip-Flop: NAND gate

It consists of a basic flip-flop circuit and two additional NAND gates along with clock pulse (CP) input. The pulse input acts as an enable signal for the other two inputs.

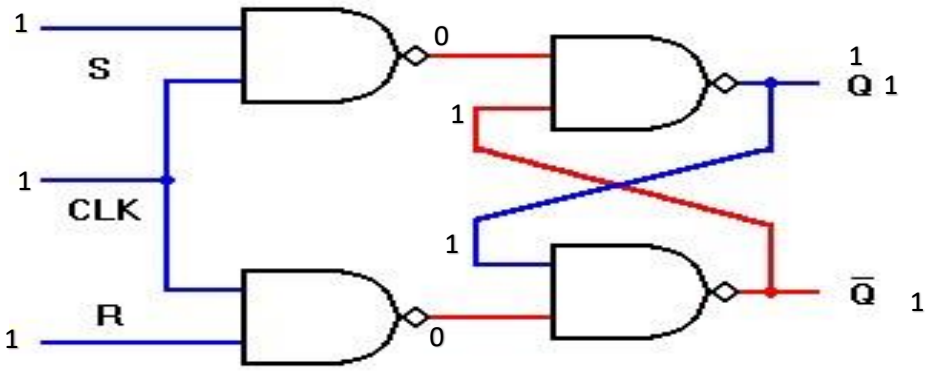


Figure : Clocked RS Flip Flop

When the pulse input goes to 1, information from the S or *R* input is allowed to reach the output.
Set state: S = 1, *R* = 0, and *CP* = 1.
Reset state: S = 0, *R* = 1, and *CP* = 1.
In either case, when *CP* returns to 0, the circuit remains in its previous state. When *CP* = 1 and both the S and *R* inputs are equal to 0, the state of the circuit does not change.

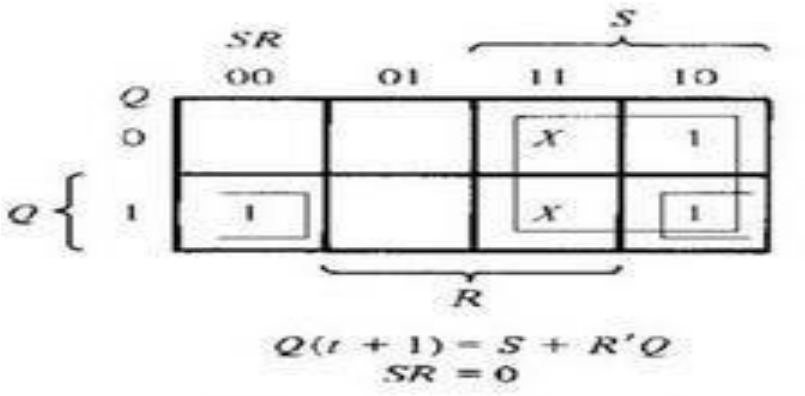
Characteristic table

Q	S	R	Q(t + 1)
0	0	0	0 (Previous State/No change)
0	0	1	0 (Reset)
0	1	0	1 (Set)
0	1	1	Indeterminate/Invalid Condition
1	0	0	1 (Previous State/No change)
1	0	1	0 (Reset)
1	1	0	1 (Set)
1	1	1	Indeterminate/Invalid Condition

Note: Apply previous state, if either S = 1 or R = 1. I.e. when one cannot decide, what will be the output?

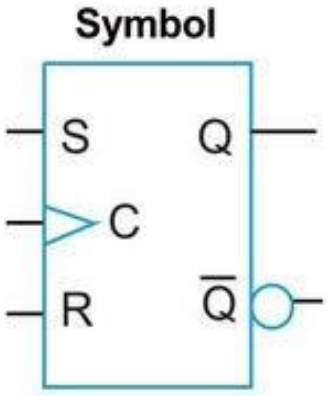
S	R	CLK	Q(t+1)	Comments
0	0	X	Q(t)	No change
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	?	Invalid

Characteristic equation: The characteristic equation of the flip-flop specifies the value of the next state as a function of the present and the input.



- **Characteristic Equation**
  $Q(t+1) = S + \overline{R}\,Q(t)$
  S R = 0 (S and R cannot be 1 simultaneously)

Graphic Symbol:



The graphic symbol of the RS flip-flop consists of a rectangular-shape block with inputs S, R, and C. The outputs are Q and Q', where Q' is the complement of Q (except in the indeterminate state).

Clocked JK Flip-Flop

A *JK* flip-flop is a refinement of the *RS* flip-flop in that the indeterminate state of the *RS* type is defined in the *JK* type.

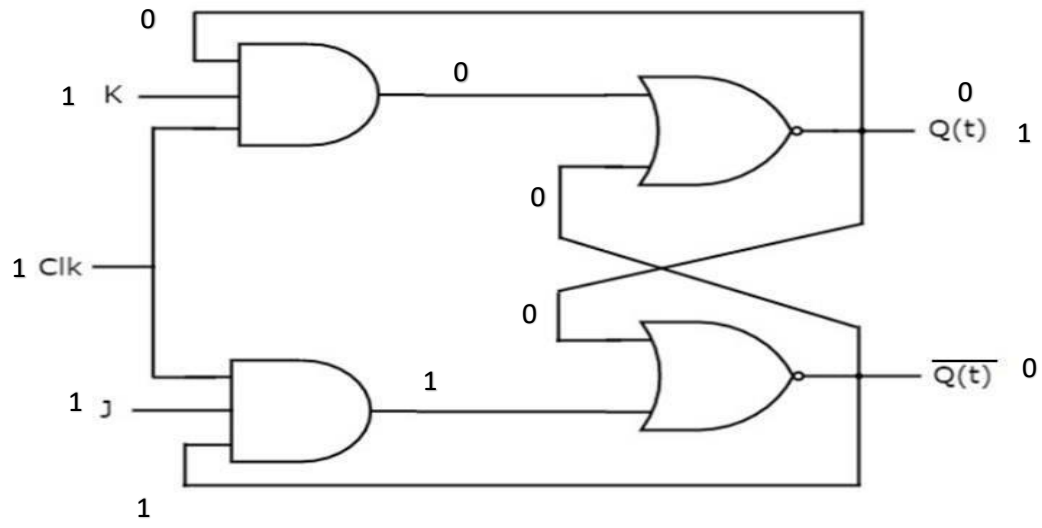
Inputs *J* and *K* behave like inputs *S* and *R* to set and clear the flip-flop, respectively.

The input marked *J* is for *set* and the input marked *K* is for *reset.*

When both inputs *J* and *K* are equal to 1, the flip-flop switches to its complement state, that is, if *Q* = 1, it switches to *Q* = 0, and vice versa.

A *JK* flip-flop constructed with two cross-coupled NOR gates and two AND gates is shown in Fig. below:

Logic diagram



A *JK* flip-flop constructed with two cross-coupled NOR gates and two AND gates.

- Output *Q* is ANDed with *K* and *CP* inputs so that the flip-flop is cleared during a clock pulse only if *Q* was previously 1.
- Similarly, output *Q'* is ANDed with J and *CP* inputs so that the flop-flop is set with a clock pulse only when *Q'* was previously 1.
- Because of the feedback connection in the *JK* flip-flop, a *CP* pulse that remains in the 1 state while both *J* and *K* are equal to 1 will cause the output to complement again and repeat complementing until the pulse goes back to 0.

Characteristic table

Q	J	K	Q(t + 1)
0	0	0	0 (Previous State/No change)
0	0	1	0 (Reset)
0	1	0	1 (Set)
0	1	1	1 (Toggle / Complement)
1	0	0	1 (Previous State/No change)
1	0	1	0 (Reset)
1	1	0	1 (Set)
1	1	1	0 (Toggle / Complement)

Flip-Flop Characteristic Tables

***JK* Flip-Flop**

J	*K*	*Q*(*t* + 1)	
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	$Q'(t)$	Complement

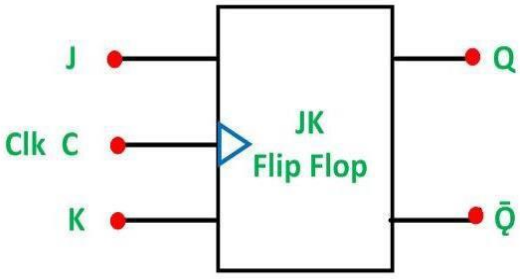
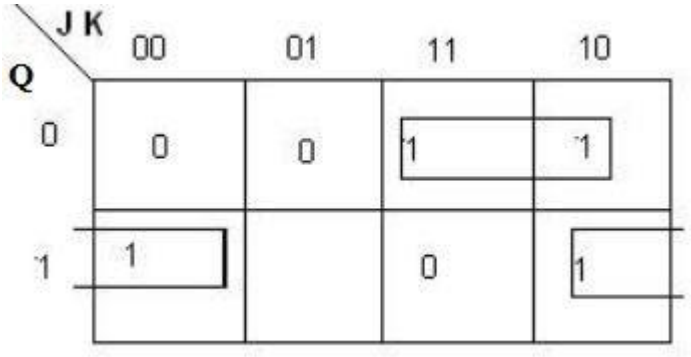


Fig: Graphic Symbol

Note: Feedback is applied, when J = 1 or k = 1.

Characteristic equation



$$Q(t+1)=JQ'+K'Q$$

D Flip-Flop

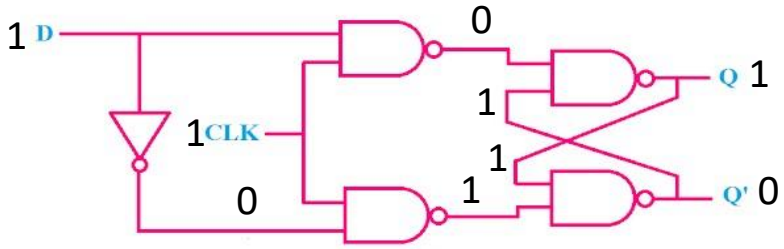
D means delay. It holds the input data. When we apply the clock pulse it sends it to the output. It helps in synchronization. I.e. It slows down the portion of the circuit that works faster than other to reach output at the same time.

- One way 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 in the *D* flip-flop
- The *D* flip-flop has only two inputs: *D* and *CP*.
- The *D* input goes directly to the S input and its complement is applied to the *R* input.
- As long as CP is 0, the circuit cannot change state regardless of the value of *D*.

The *D* input is sampled when *CP* = 1.

- If *D* is 1, the *Q* output goes to 1, placing the circuit in the **set state**.
- If *D* is 0, output *Q* goes to 0 and the circuit switches to the **clear state**.

Logic diagram

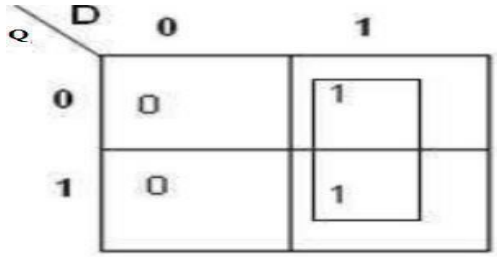


Characteristic Table

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

Q	D	Q(T+1)
0	0	0
0	1	1
1	0	0
1	1	1

Characteristic Equation



Characteristic Equation

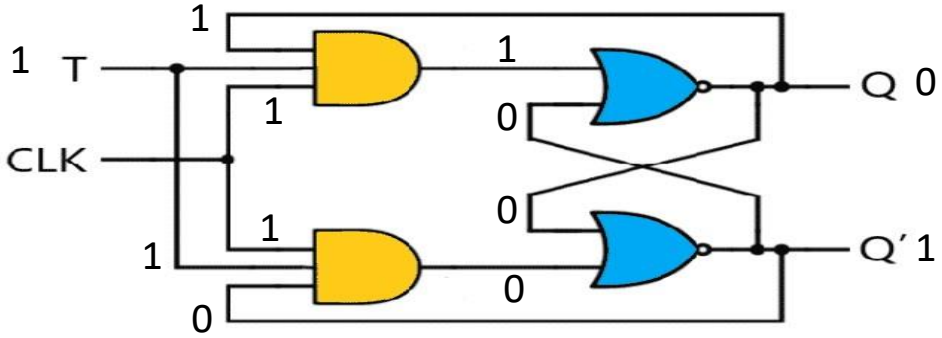
$$Q(t+1) = D$$

T Flip-Flop

- The *T* flip-flop is a single-input version of the *JK* flip-flop and is obtained from the *JK* flip-flop when both inputs are tied together.
- The designation *T* comes from the ability of the flip-flop to "toggle," or complement, its state.
- Regardless of the present state, the flip-flop complements its output when the clock pulse occurs while input *T* is 1.
- The characteristic table and characteristic equation show that:

— When $T = 0$, $Q(t + 1) = Q$, that is, the next state is the same as the present state and no change occurs. When $T = 1$, then $Q(t + 1) = Q'$, and the state of the flip-flop is complemented.

Logic Diagram

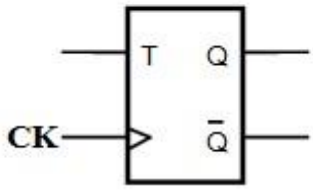


Characteristic Table

T Flip-Flop

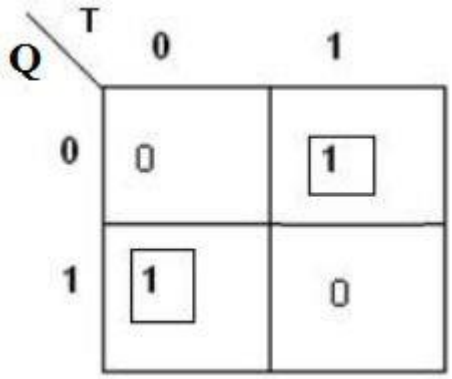
T	$Q(t + 1)$	
0	$Q(t)$	No change
1	$Q'(t)$	Complement

Q	T	Q(T+1)
0	0	0
0	1	1
1	0	1
1	1	0



Graphical symbol

Characteristic equation



$$Q(t+1) = T.Q' + T'.Q$$

Clocked JK Flip-Flop: Race around condition

- If the inputs of JK flip Flop are J=K=1 and Q=0 and clock pulse as shown in fig., After a time interval t_p equal to propagation delay of NAND gates, the output will change to Q=1.Now we have J=1,K=1 and Q=1.If duration of clock pulse(T) is greater than propagation delay t_p, after another time interval of t_p the output will change back to Q=0,hence the output will oscillate back and forth between 0 and 1.The output is uncertain at the end of clock pulse if flip flop is level trigger. This situation is called race around condition.

Master-Slave Flip-Flop

A master-slave flip-flop is constructed from two separate flip-flops. One circuit serves as a master and the other as a slave, and the overall circuit is referred to as a master slave flip-flop.

Master-Slave JK flip-flop

Master-slave *JK* flip-flop constructed with NAND gates is shown in Fig. below. It consists of two flip-flops; gates 1 through 4 form the **master flip-flop**, and gates 5 through 8 form the **slave flip-flop**. The information present at the *J* and *K* inputs is transmitted to the master flip-flop on the positive edge of a clock pulse and is held there until the negative edge of the clock pulse occurs, after which it is allowed to pass through to the slave flip-flop.

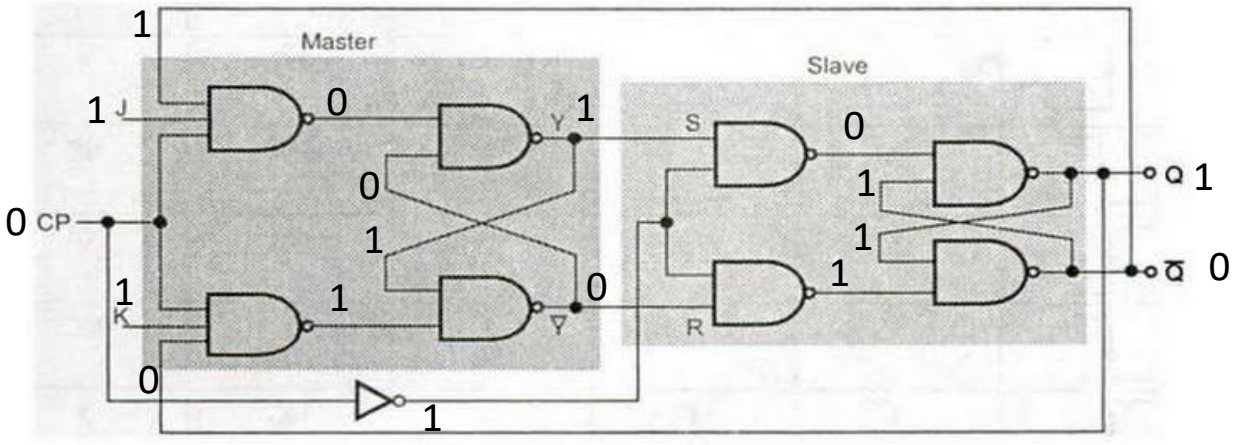


Fig. Master-slave JK flip-flop

- The following is truth table of master slave flip flop.

Case	Inputs			Outputs		Remark
	CLK	J	K	Q_{n+1}	$\bar{Q}_{n+1}$	
I	×	0	0	Q_n	$\bar{Q}_n$	No change
II	⎍(1)	0	0	Q_n	$\bar{Q}_n$	No change
III	⎍(1)	0	1	0	1	Reset
IV	⎍(1)	1	0	1	0	Set
V	⎍(1)	1	1	$\bar{Q}_n$	Q_n	Toggle

Fig . Truth table of Master slave JK FF

Operation:

- The clock input is normally 0, which prevents the J and K inputs from affecting the master flip-flop.
- The slave flip-flop is a clocked RS type, with the master flip-flop supplying the inputs and the clock input being inverted by NOT gate
- When the clock is 0, Q = Y, and Q' = Y'. • When the positive edge of a clock pulse occurs, the master flip-flop is affected and may switch states.
- The slave flip-flop is isolated as long as the clock is at the 1 level.
- When the clock input returns to 0, the master flip-flop is isolated from the J and K inputs and the slave flip-flop goes to the same state as the master flip-flop.

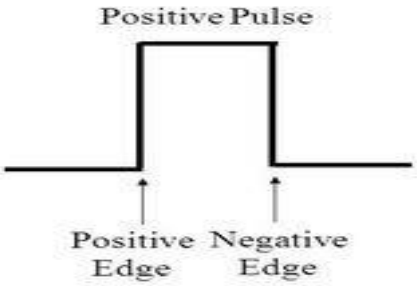
When J = 1 and K = 1, master toggles on the positive clock and slave then copies the output of master on the negative clock. At this instant, feedback inputs to the master flip-flop are complemented but as it is negative half of the clock pulse master flip-flop is inactive. This prevents race around condition. Fig. shows input and output waveforms of master-slave JK flip-flop.

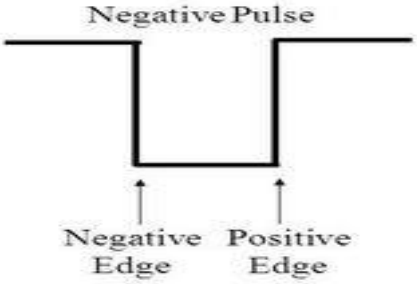
When J =1 and K = 0, the master sets on the positive clock. The high Y output of the master drives the S input of the slave, so at negative clock, slave sets, copying the action of the master.

When J = 0 and K = 1, the master resets on the positive clock. The high $\bar{Y}$ output of the master goes to the R input of the slave. Therefore, at the negative clock slave resets, again copying the action of the master.

Triggering of Flip Flop

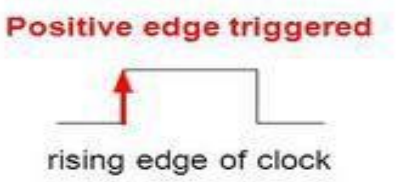
- The state of a flip-flop is switched by a momentary change in the input signal. This momentary change is called a ***trigger.***
- Clocked flip-flops are triggered by ***pulses***.
- A pulse starts from an initial value of 0, goes momentarily to 1, and after a short time, returns to its initial 0 value.
- A clock signal is a periodic square wave that indefinitely switches values from 0 to 1 and 1 to 0 at fixed intervals.
  A clock pulse may be either positive or negative. – A positive clock source remains at 0 during the interval between pulses and goes to 1 during the occurrence of a pulse.
  - An edge triggered flip flop changes state either at positive edge(rising edge) or at negative edge(falling edge) of the clock pulse and is sensitive to its input only at this transition of the clock.
  - A pulse triggered flip flop changes state either at the positive pulse (positive level of the pulse) or at negative pulse(negative level of pulse)of the applied clock pulse.

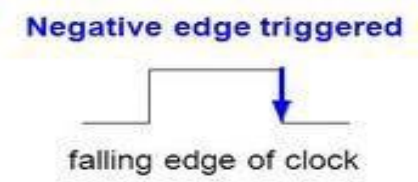


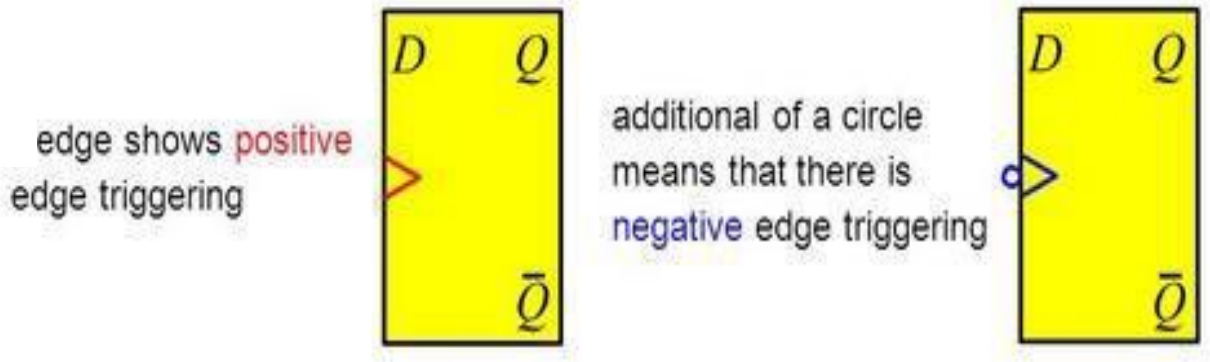


Level Triggering	Edge Triggering
• Outputs change based on inputs whenever *clock* is high • Memory will be considered to be level triggered (for cost reasons)	• Outputs change based on inputs only when clock transitions • Positive edge triggered logic when leading edge cause triggering • Negative edge triggered when trailing edge causes triggering

These circuits respond to their inputs on either the **rising** or **falling** edge of the clock — a precise point in time rather than an interval.







Edge-Triggered Flip-Flop

- Edge-triggered flip-flop (alternative to master-slave) synchronizes the state changes during clock-pulse transitions.
- In this type of flip-flop, output transitions occur at a specific level of the clock pulse.
- When the pulse input level exceeds this threshold level, the inputs are locked out and the flip-flop is therefore unresponsive to further changes in inputs until the clock pulse returns to 0 and another pulse occurs.
- Some edge-triggered flip-flops cause a transition on the positive edge of the pulse, and others cause a transition on the negative edge of the pulse.
- The logic diagram of a D-type positive-edge-triggered flip-flop is shown below.
  It consists of three basic flip-flops. NAND gates 1 and 2 make up one basic flip-flop and gates 3 and 4 another. The third basic flip-flop comprising gates 5 and 6 provides the outputs to the circuit. Inputs S and R of the third basic flip-flop must be maintained at logic-1 for the outputs to remain in their steady state values.
- When S = 0 and R = 1, the output goes to the set state with Q = 1. • When S = 1 and R = 0, the output goes to the clear state with Q = 0.

Inputs S and R are determined from the states of the other two basic flip-flops. These two basic flip-flops respond to the external inputs D (data) and CP (clock pulse).

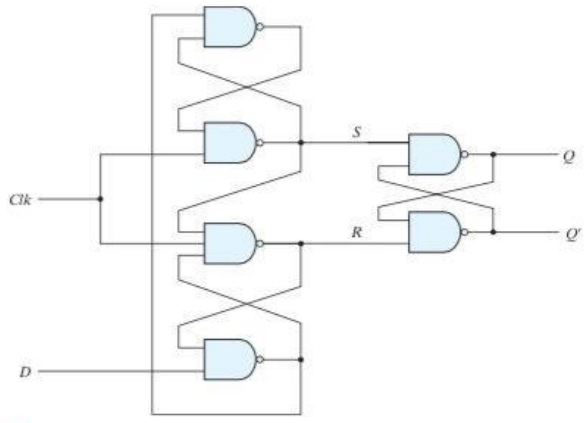
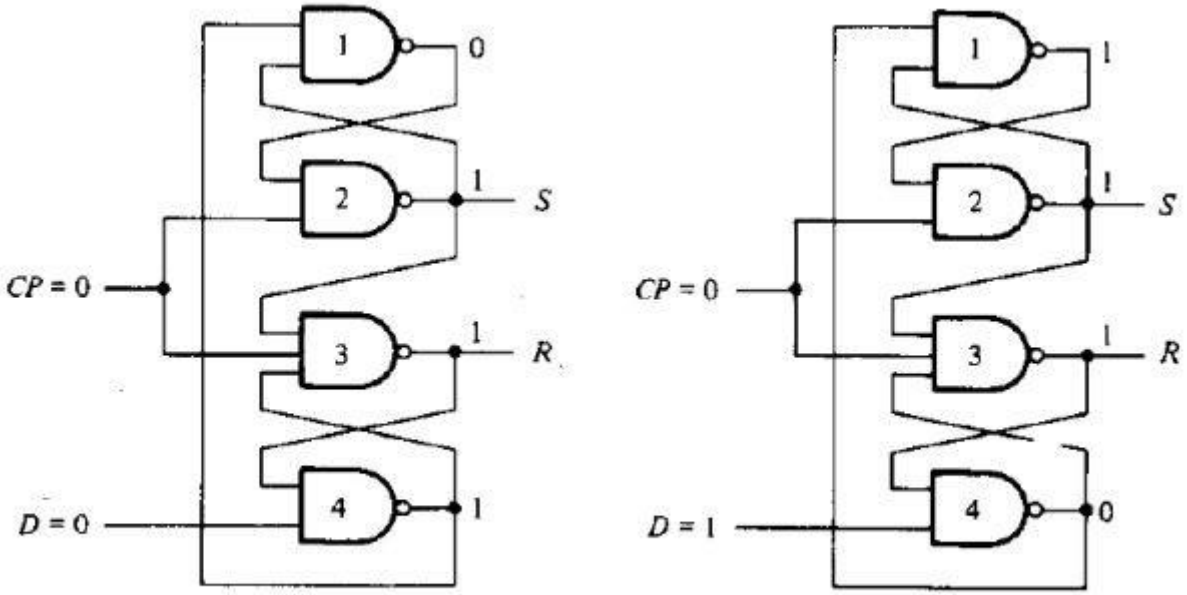


FIGURE
D-type positive-edge-triggered flip-flop

Truth Table

CP	D	S	R	Q(t + 1)
0	0	1	1	0 (Previous State)
0	1	1	1	0 (Previous State)
1	0	1	0	0 (Reset)
1	1	0	1	1 (Set)

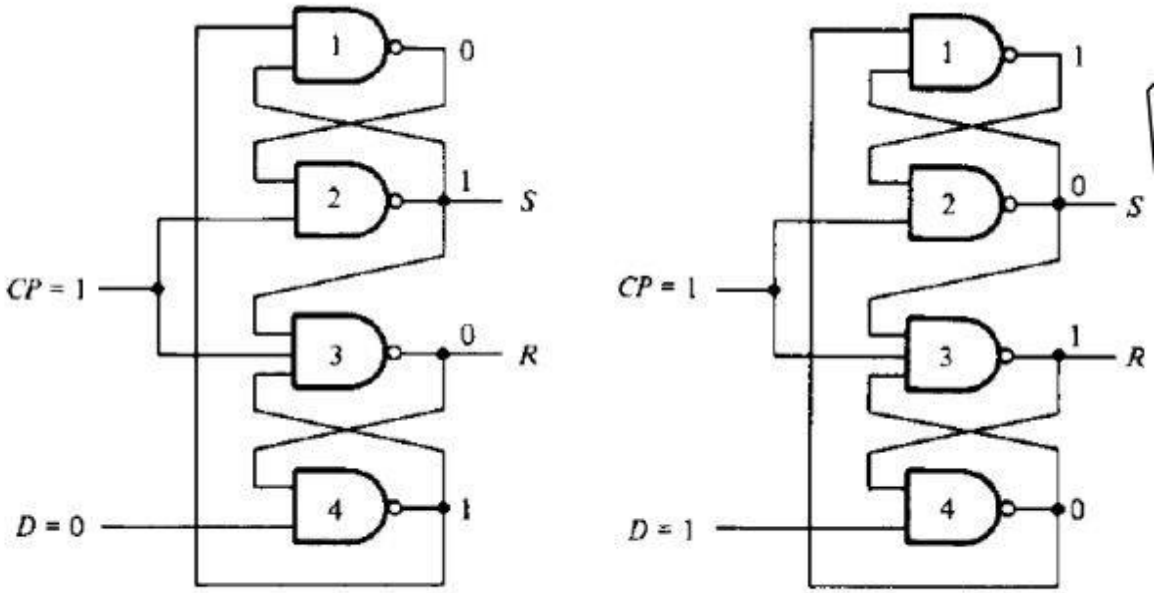
When CP = 0



(a) With $CP = 0$

Operation:

Fig (a) shows the binary values at the outputs of the four gates when CP = 0. Input D may be equal to 0 or 1. In either case, a CP of 0 causes the outputs of gates 2 and 3 to go to 1, thus making S = R = 1, which is the condition for a steady state output.



(b) With $CP = 1$

Operation:

If D = 1 then S changes to 0, but R remains at 1, which causes the output of the flip-flop Q to go to 1 (set state).

If D = 0 then S = 1 and R = 0. Flip-flop goes to clear state (Q = 0).

Excitation Table:

A table that lists required inputs for a given change of state (Present to next-state) is called an *excitation* table. The required input conditions are derived from the information available in the characteristic table.

Flip-Flop Excitation Tables

$Q(t)$	$Q(t+1)$	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

(a) *RS*

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

(b) *JK*

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1

Characteristic Table of SR Flip-flop

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

Truth table

Characteristic Table of JK Flip-flop

Q	J	K	Q(t + 1)
0	0	0	0 (PS)
0	0	1	0 (Reset)
0	1	0	1 (Set)
0	1	1	1 (Toggle)
1	0	0	1 (PS)
1	0	1	0 (Reset)
1	1	0	1 (Set)
1	1	1	0 (Toggle)

Characteristic Table of D Flip-flop

Q	D	Q(T+1)
0	0	0
0	1	1
1	0	0
1	1	1

Characteristic Table of T Flip-flop

Q	T	Q(T+1)
0	0	0
0	1	1
1	0	1
1	1	0

Flipflop excitation table

It is very important in analysis of problem. It is just a reverse of truth table. Here, based on present output (Q_n) and next output (Q_{n+1}) the input of flipflop is generated.

Excitation table of flipflops

Transition		RS flipflop		JK flipflop		D flipflop	T flipflop
$Q_n \rightarrow Q_{n+1}$		S	R	J	K	D	T
0	0	0	×	0	×	0	0
0	1	1	0	1	×	1	1
1	0	0	1	×	1	0	1
1	1	×	0	×	0	1	0

Flip-flop Conversion:

RS flip-flop to JK flip-flop:

⇒ We need JK flip-flop we have to draw characteristics table of JK flip-flop.

PS → Q(t)	J	K	Q(t+1) ← NS
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

Available flip-flop

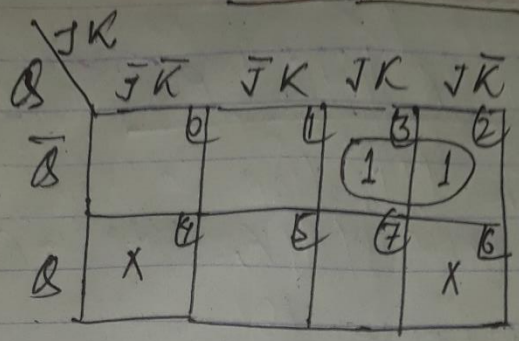
Excitation table for RS flip-flop

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

Flip-flop inputs using excitation table:

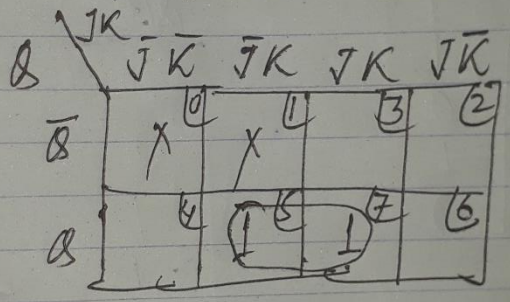
Q(t)	J	K	Q(t+1)	Excitation Inputs S	R
0	0	0	0	0	X
0	0	1	0	0	X
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	1	X	0
1	0	1	0	0	1
1	1	0	1	X	0
1	1	1	0	0	1

K-Map for S



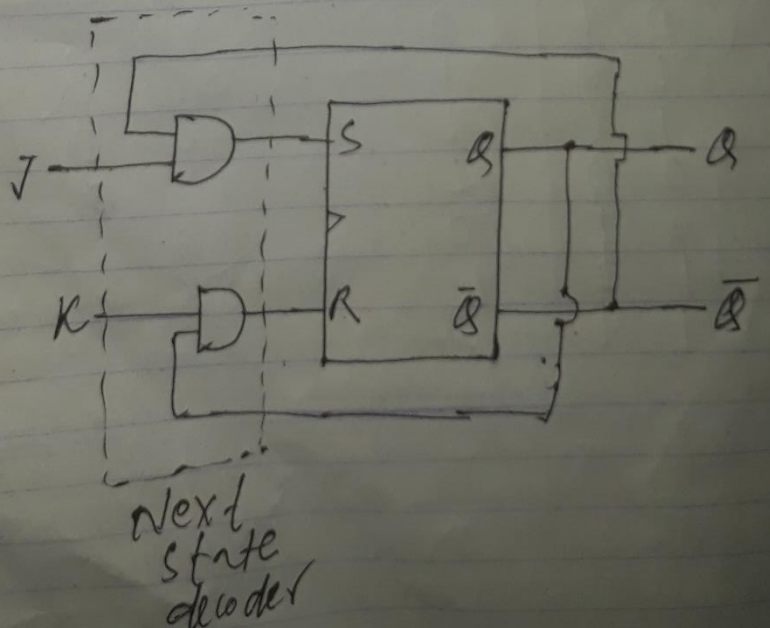
$S = \bar{Q}J$

K-Map for R



$R = QK$

Logic diagram



SR to D flip-flop conversion:

We need D flip-flop so we have to make characteristics table of D flip-flop:

Characteristics table of D flip-flop:

Q(t)	D	Q(t+1)
0	0	0
0	1	1
1	0	0
1	1	1

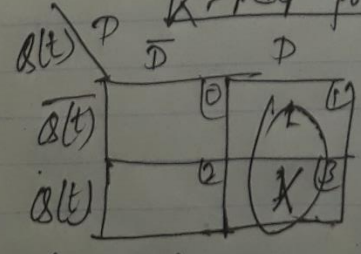
Excitation table of available flip-flop i.e SR:

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

Excitation table to find flip-flop inputs:

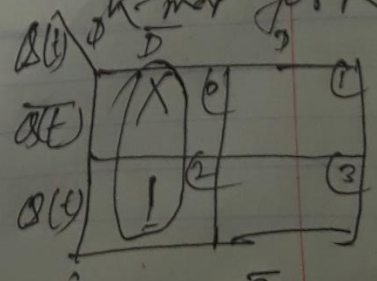
Q(t)	D	Q(t+1)	Excitation inputs S	R
0	0	0	0	X
0	1	1	1	0
1	0	0	0	1
1	1	1	X	0

K-map for S



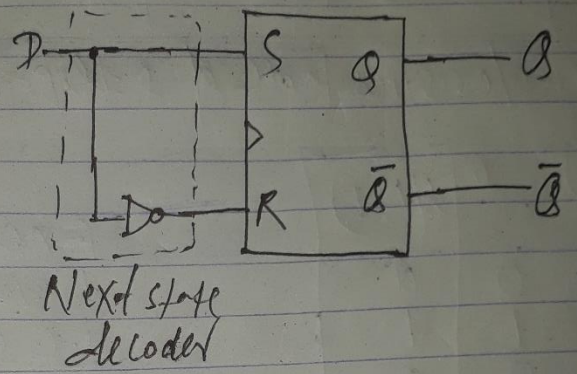
$S = D$

K-map for R



$R = \bar{D}$

Logic diagram



D flip-flop to SR flip-flop conversion

Characteristics table of SR flip-flop

Q(t)	S	R	Q(t+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

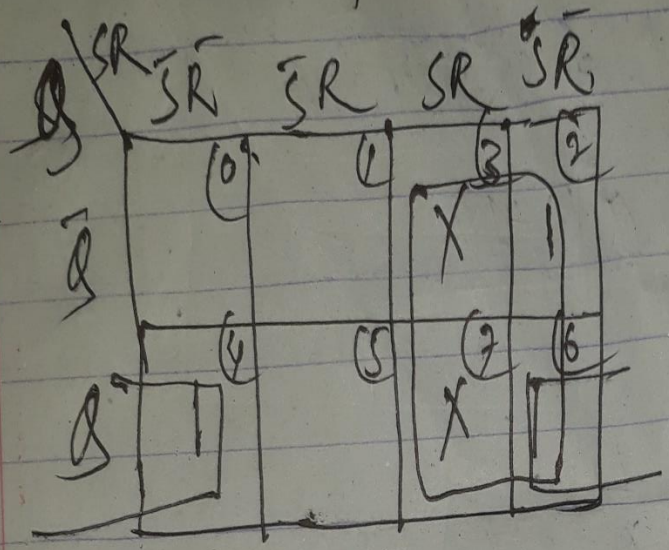
Excitation table for D flip-flop

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

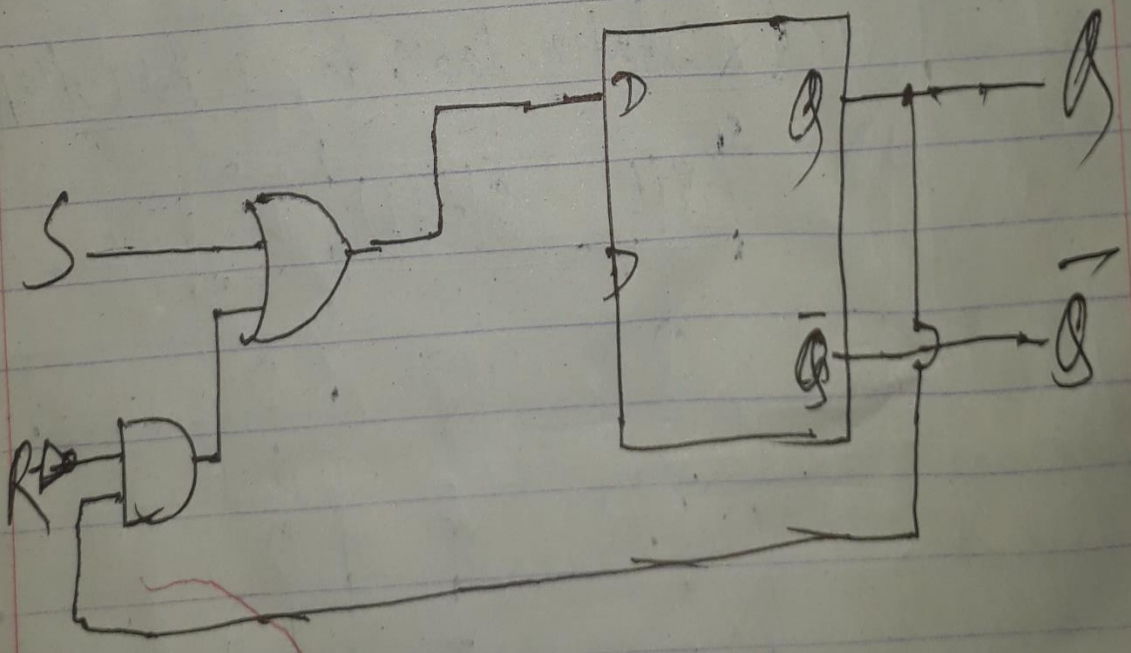
flip-flop inputs using excitation table.

Q(t)	S	R	Q(t+1)	excitation inputs D
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	X	X
1	0	0	1	1
1	0	1	0	0
1	1	0	1	1
1	1	1	X	X

K-Map for D



$$D = S + Q\bar{R}$$



JK flip-flop to T flip-flop

Characteristics table of T flip-flop

Q	T	Q(t+1)
0	0	0
0	1	1
1	0	1
1	1	0

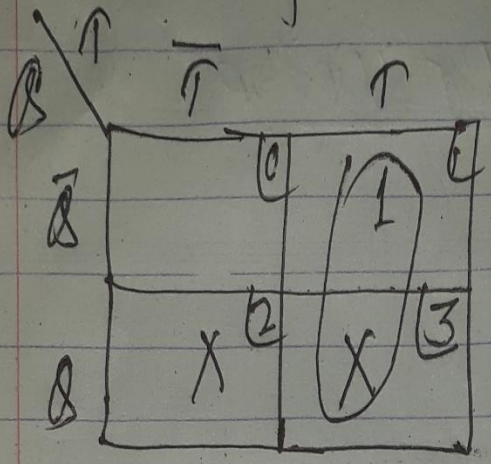
Excitation table of JK flip-flop

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

flip-flop inputs using excitation Inputs

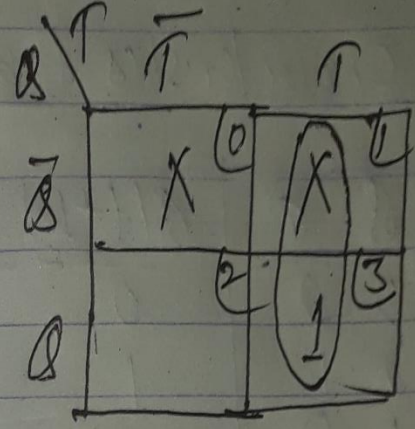
Q	T	Q(t+1)	Excitation Inputs J	K
0	0	0	0	X
0	1	1	1	X
1	0	1	X	0
1	1	0	X	1

K-map for J



$J = T$

K-map for K



$K = T$

Logic diagram