ST. LAWRENCE HIGH SCHOOL

A Jesuit Christian Minority Institution

STUDY MATERIAL - 10

Subject: COMPUTER SCIENCE **Class - 12**

Chapter: Latches & Flip-flops **Date: 30/06/2020**

Introduction to Sequential Circuits

A **Sequential circuit** combinational logic circuit that consists of inputs variable (X), logic gates (Computational circuit), and output variable (Z).

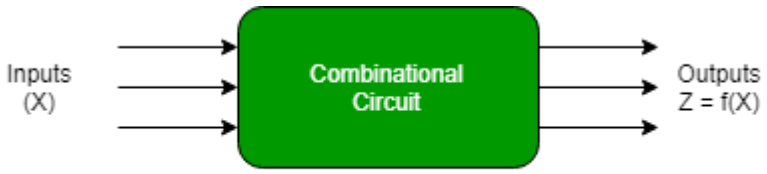


Figure: Combinational Circuits

Combinational circuit produces an output based on input variable only, but **Sequential circuit** produces an output based on **current input and previous input variables**. That means sequential circuits include memory elements which are capable of storing binary information. That binary information defines the state of the sequential circuit at that time. A latch is capable of storing one bit of information.

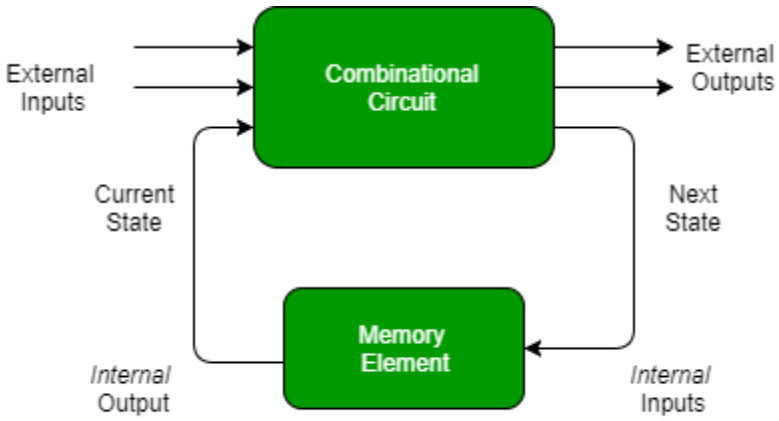


Figure: Sequential Circuit

As shown in figure there are two types of input to the combinational logic:

1. External inputs which not controlled by the circuit.
2. Internal inputs which are a function of a previous output states.

Secondary inputs are state variables produced by the storage elements, whereas secondary outputs are excitations for the storage elements.

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

1. **Asynchronous sequential circuit –** These circuits **do not use a clock signal** but uses the pulses of the inputs. These circuits are **faster** than synchronous sequential circuits because there is clock pulse and change their state immediately when there is a change in the input signal. We use asynchronous sequential circuits when speed of operation is important and **independent** of internal clock pulse.

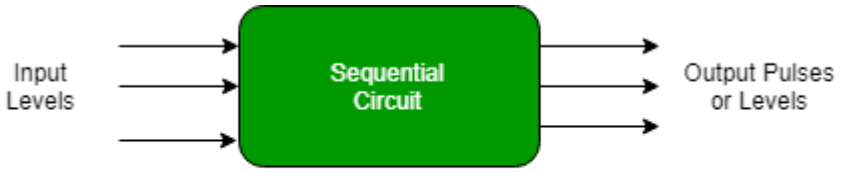


Figure: Asynchronous Sequential Circuit

But these circuits are more **difficult** to design and their output is **uncertain**.

2. **Synchronous sequential circuit –** These circuit **uses clock signal** and level inputs (or pulsed) (with restrictions on pulse width and circuit propagation). The output pulse is the same duration as the clock pulse for the clocked sequential circuits. Since they wait for the next clock pulse to arrive to perform the next operation, so these circuits are bit **slower** compared to asynchronous. Level output changes state at the start of an input pulse and remains in that until the next input or clock pulse.

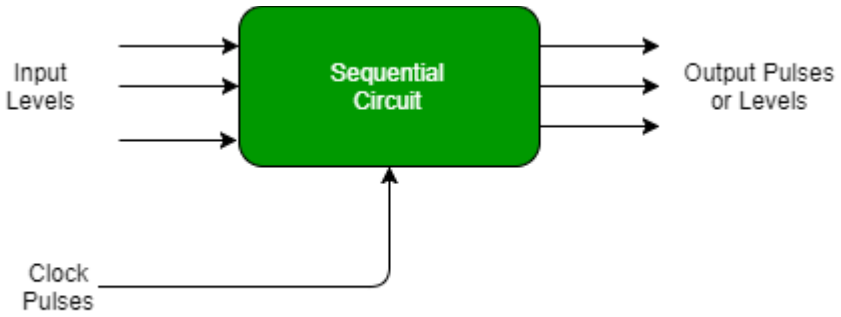


Figure: Synchronous Sequential Circuit

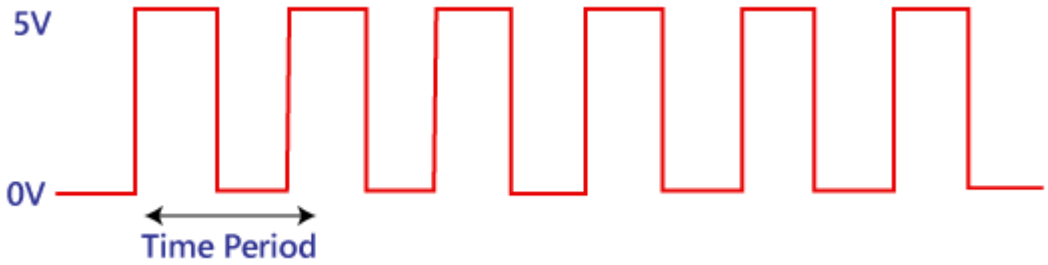
We use synchronous sequential circuit in synchronous counters, flip flops, and in the design of MOORE-MEALY state management machines.

We use sequential circuits to design Counters, Registers, RAM, MOORE/MEALY Machine and other state retaining machines.

Clock Signal and Triggering

Clock signal

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



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

Types of Triggering

These are two types of triggering in sequential circuits:

Level triggering

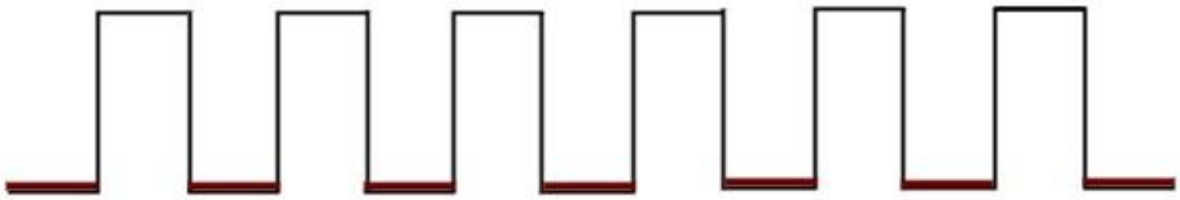
The logic High and logic Low are the two levels in the clock signal. In level triggering, when the clock pulse is at a particular level, only then the circuit is activated. There are the following types of level triggering:

Positive level triggering

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

Negative level triggering

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



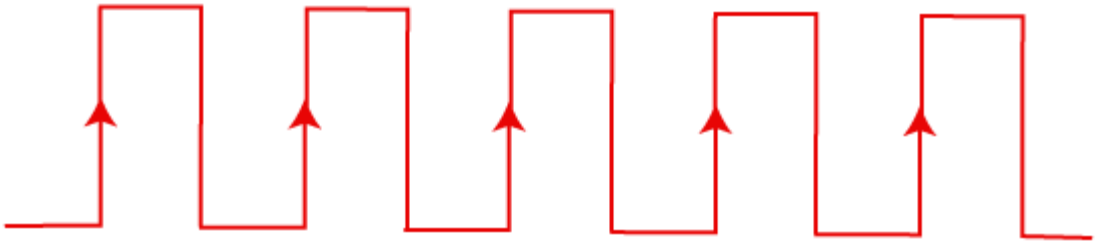
Edge triggering

In clock signal of edge triggering, two types of transitions occur, i.e., transition either from Logic Low to Logic High or Logic High to Logic Low.

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

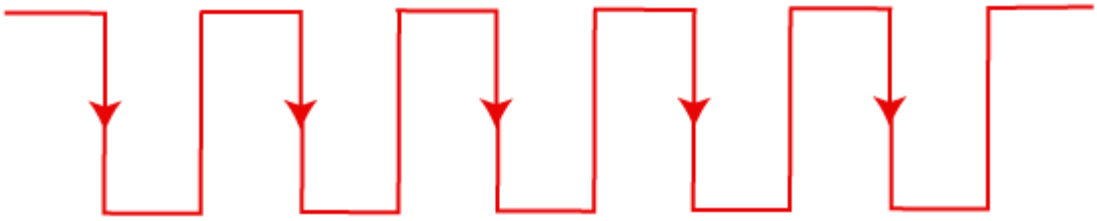
Positive edge triggering

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



Negative edge triggering

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



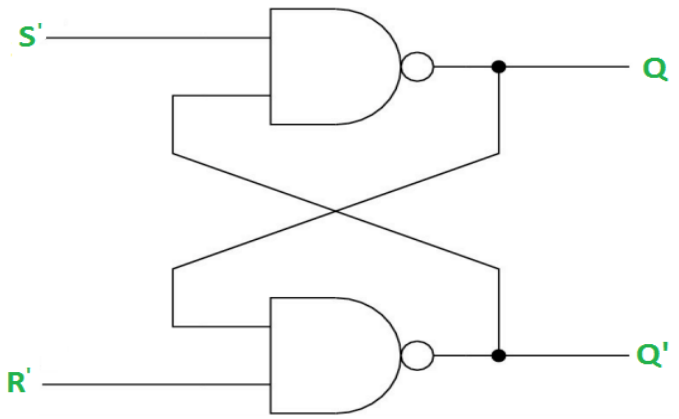
Latches in Digital Logic

Latches are basic storage elements that operate with signal levels (rather than signal transitions). Latches controlled by a clock transition are flip-flops. Latches are level-sensitive devices. Latches are useful for the design of the asynchronous sequential circuit.

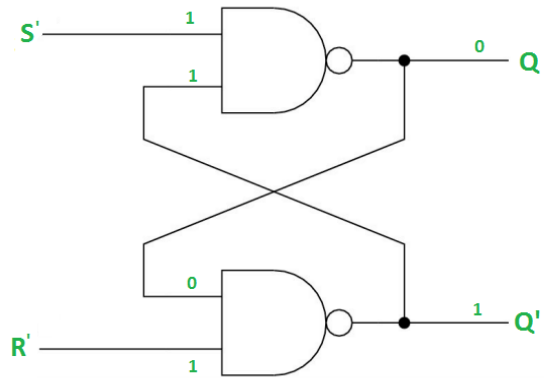
SR (Set-Reset) Latch – SR Latch is a circuit with:
(i) 2 cross-coupled NOR gate or 2 cross-coupled NAND gate.
(ii) 2 input S for SET and R for RESET.
(iii) 2 output Q, Q'.

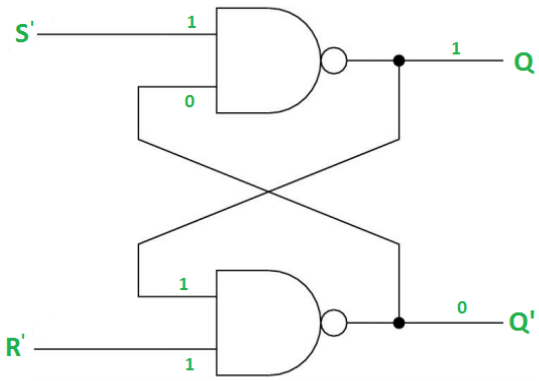
Q	Q'	STATE
1	0	Set
0	1	Reset

Under normal conditions, both the input remains 0. The following is the RS Latch with NAND gates:

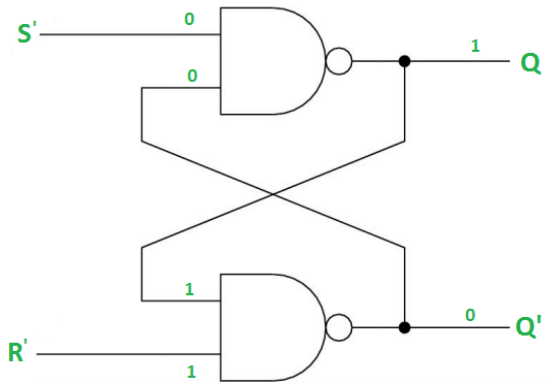


Case-1: S'=R'=1 (S=R=0) –
If Q = 1, Q and R' inputs for 2nd NAND gate are both 1.
If Q = 0, Q and R' inputs for 2nd NAND gate are 0 and 1 respectively.

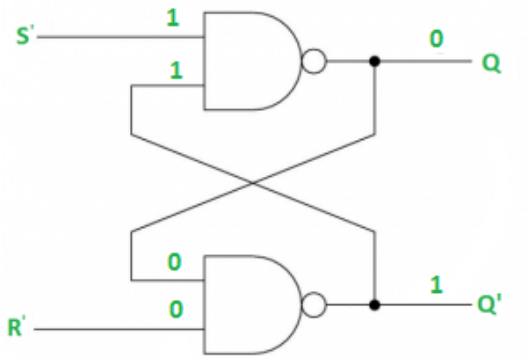




Case-2: S'=0, R'=1 (S=1, R=0) –
As S'=0, the output of 1st NAND gate, Q = 1(**SET state**). In 2nd NAND gate, as Q and R' inputs are 1, Q'=0.

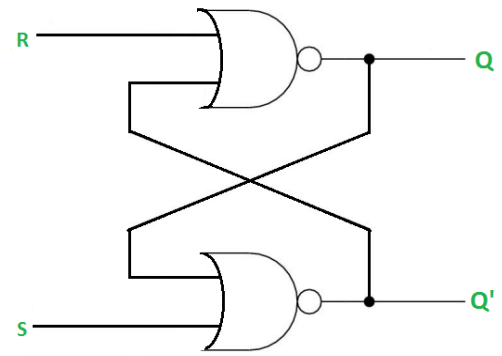


Case-3: S'= 1, R'= 0 (S=0, R=1) –
As R'=0, the output of 2nd NAND gate, Q' = 1. In 1st NAND gate, as Q and S' inputs are 1, Q=0(**RESET state**).



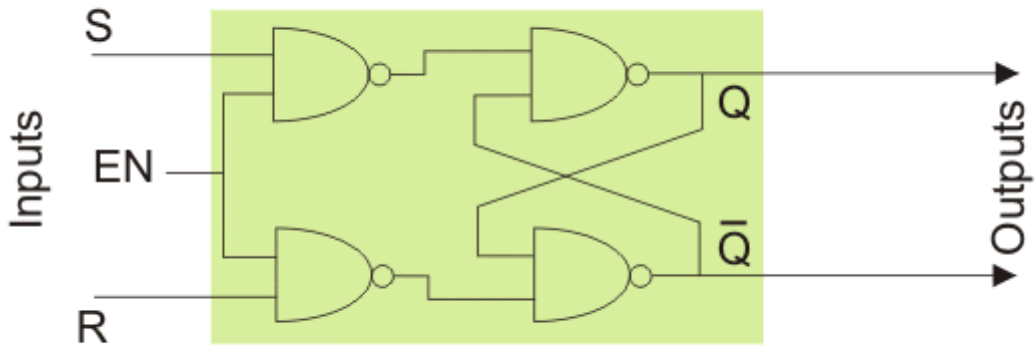
Case-4: S'= R'= 0 (S=R=1) –
When S=R=1, both Q and Q' becomes 1 which is not allowed. So, the input condition is prohibited.

The SR Latch using NOR gate is shown below:



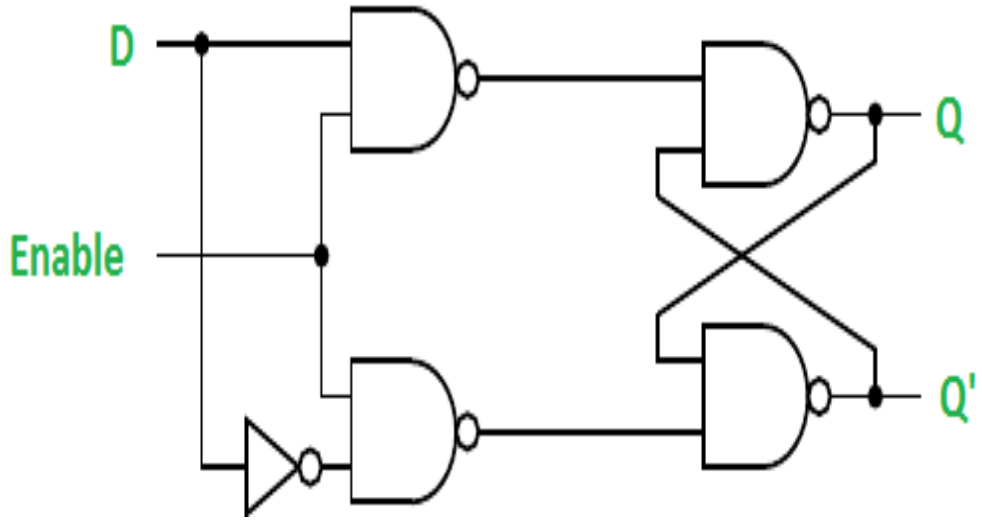
Gated SR Latch –

A Gated SR latch is a SR latch with enable input which works when enable is 1 and retains the previous state when enable is 0.



Gated D Latch –

D latch is similar to SR latch with some modifications made. Here, the inputs are complements of each other. The design of D latch with Enable signal is given below:



The truth table for the D-Latch is shown below:

Enable	D	Q(n)	Q(n+1)	STATE
1	0	X	0	RESET
1	1	X	1	SET
0	x	X	Q(n)	No Change

As the output is same as the input D, D latch is also called as *Transparent Latch*.

Flip-Flops in Digital Logic

SR Flip Flop

The SR flip flop is a 1-bit memory bistable device having two inputs, i.e., SET and RESET. The SET input 'S' set the device or produce the output 1, and the RESET input 'R' reset the device or produce the output 0. The SET and RESET inputs are labelled as **S** and **R**, respectively.

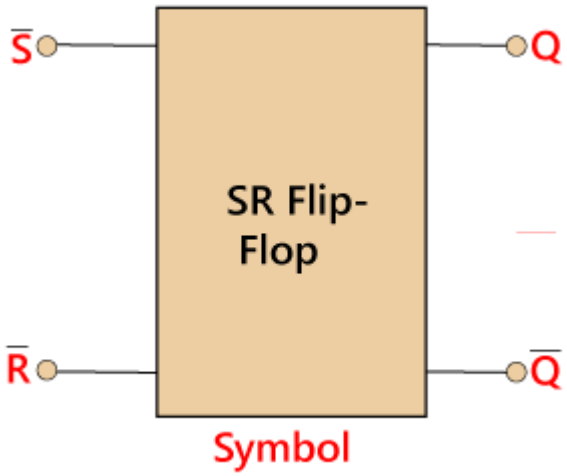
The SR flip flop stands for "Set-Reset" flip flop. The reset input is used to get back the flip flop to its original state from the current state with an output 'Q'. This output depends on the set and reset conditions, which is either at the logic level "0" or "1".

The NAND gate SR flip flop is a basic flip flop which provides feedback from both of its outputs back to its opposing input. This circuit is used to store the single data bit in the memory circuit. So, the SR flip flop has a total of three inputs, i.e., 'S' and 'R', and current output 'Q'. This output 'Q' is related to the current history or state. The term "flip-flop" relates to the actual operation of the device, as it can be "flipped" to a logic set state or "flopped" back to the opposing logic reset state.

The NAND Gate SR Flip-Flop

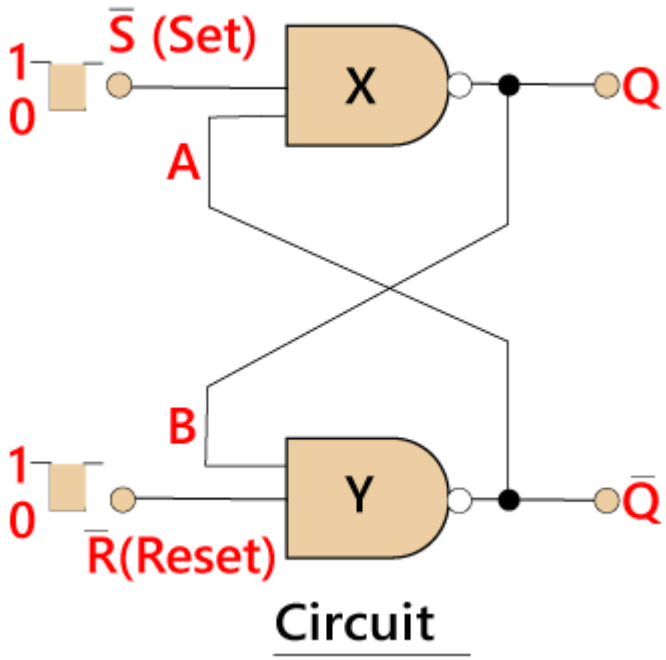
We can implement the set-reset flip flop by connecting two cross-coupled 2-input NAND gates together. In the SR flip flop circuit, from each output to one of the other NAND gate inputs, feedback is connected. So, the device has two inputs, i.e., Set 'S' and Reset 'R' with two outputs Q and Q' respectively. Below are the block diagram and circuit diagram of the S-R flip flop.

Block Diagram:



Symbol

Circuit Diagram:



The Set State

In the above diagram, when the input R is set to false or 0 and the input S is set to true or 1, the NAND gate Y has an input 0, which will produce the output Q' 1. The value of Q' is faded to the NAND gate 'X' as input 'A', and now both the inputs of the NAND gate 'X' are 1(S=A=1), which will produce the output 'Q' 0.

Now, if the input R is changed to 1 with 'S' remaining 1, the inputs of NAND gate 'Y' is R=1 and B=0. Here, one of the inputs is also 0, so the output of Q' is 1. So, the flip flop circuit is set or latched with Q=0 and Q'=1.

Reset State

The output Q' is 0, and output Q is 1 in the second stable state. It is given by R =1 and S = 0. One of the inputs of NAND gate 'X' is 0, and its output Q is 1. Output Q is faded to NAND gate Y as input B. So, both the inputs to NAND gate *Y* are set to 1, therefore, Q' = 0.

Now, if the input S is changed to 0 with 'R' remaining 1, the output Q' will be 0 and there is no change in state. So, the reset state of the flip flop circuit has been latched, and the set/reset actions are defined in the following truth table:

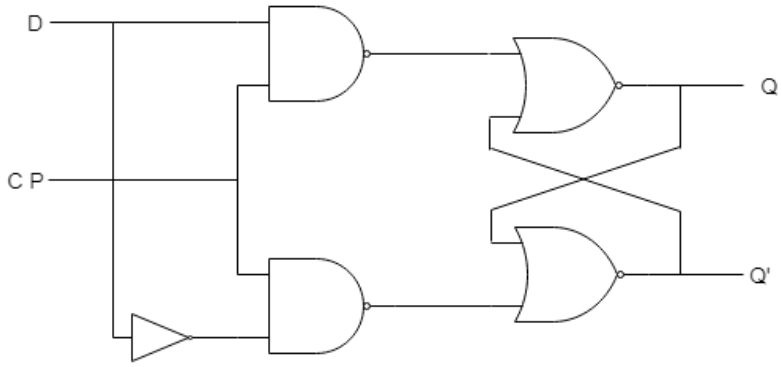
State	S	R	Q	Q'	Description
Set	1	0	0	1	Set Q'>>1
	1	1	0	1	No change
Reset	0	1	1	0	Reset Q'>>0
	1	1	1	0	No change
Invalid	0	0	1	1	Invalid Condition

From the above truth table, we can see that when set 'S' and reset 'R' inputs are set to 1, the outputs Q and Q' will be either 1 or 0. These outputs depend on the input state S or R before the input condition exist. So, when the inputs are 1, the states of the outputs remain unchanged.

The condition in which both the inputs states are set to 0 is treated as invalid and must be avoided.

D Flip-Flop

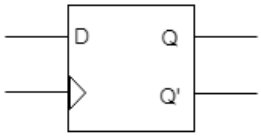
D flip-flop is a slight modification of clocked SR flip-flop.



(a) Logic diagram with Nand gates

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

(c) Transition table



(b) Graphic Symbol

fig. Clocked D flip flop

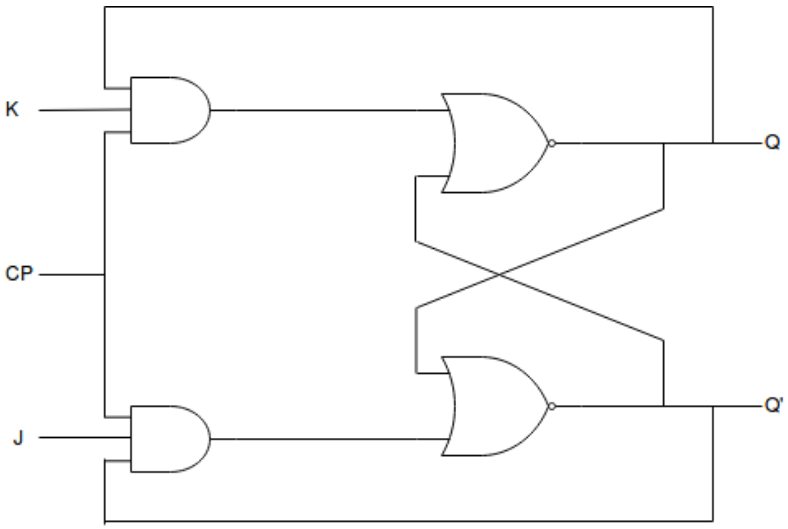
From the above figure, you can see that the D input is connected to the S input and the complement of the D input is connected to the R input.

When the value of CP is '1' (HIGH), the flip-flop moves to the SET state if it is '0' (LOW), the flip-flop switches to the CLEAR state.

J-K Flip-Flop

J-K flip-flop can be considered as a modification of the S-R flip-flop.

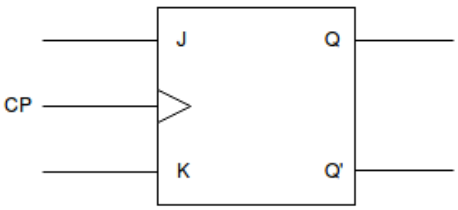
The main difference is that the intermediate state is more refined and precise than that of an S-R flip-flop.



a) Logic diagram

Q	J	K	Q(t+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

b) Transition table



c) Graphical symbol

fig. Clocked JK flip flop

The characteristics of inputs ‘J’ and ‘K’ are same as the 'S' and 'R' inputs of the S-R flip-flop.

J stands for SET, and 'K' stands for CLEAR.

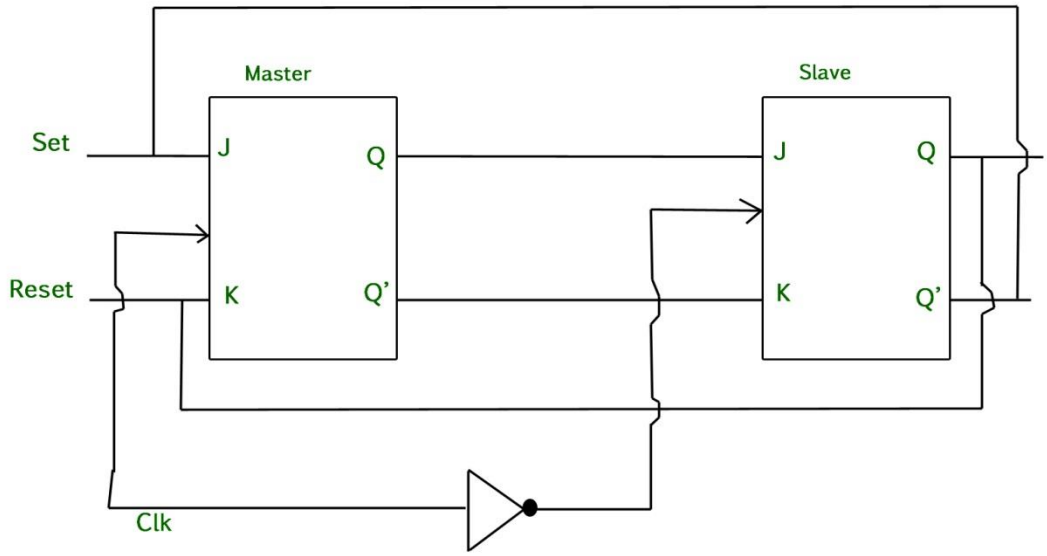
When both the inputs J and K have a HIGH state, the flip-flop switches to the complement state, so, for a value of Q = 1, it switches to Q=0, and for a value of Q = 0, it switches to Q=1.

Master-Slave JK Flip Flop

Race Around Condition In JK Flip-flop – For J-K flip-flop, if J=K=1, and if clk=1 for a long period of time, then Q output will toggle as long as CLK is high, which makes the output of the flip-flop unstable or uncertain. This problem is called race around condition in J-K flip-flop. This problem (Race around Condition) can be avoided by ensuring that the clock input is at logic "1" only for a very short time. This introduced the concept of **Master Slave JK** flip flop.

Master Slave JK flip flop –
The Master-Slave Flip-Flop is basically a combination of two JK flip-flops connected together in a series configuration. Out of these, one acts as the **"master"** and the other as a **"slave"**. The output from the master flip flop is connected to the two inputs of the slave flip flop whose output is fed back to inputs of the master flip flop.

In addition to these two flip-flops, the circuit also includes an **inverter**. The inverter is connected to clock pulse in such a way that the inverted clock pulse is given to the slave flip-flop. In other words if CP=0 for a master flip-flop, then CP=1 for a slave flip-flop and if CP=1 for master flip flop then it becomes 0 for slave flip flop.



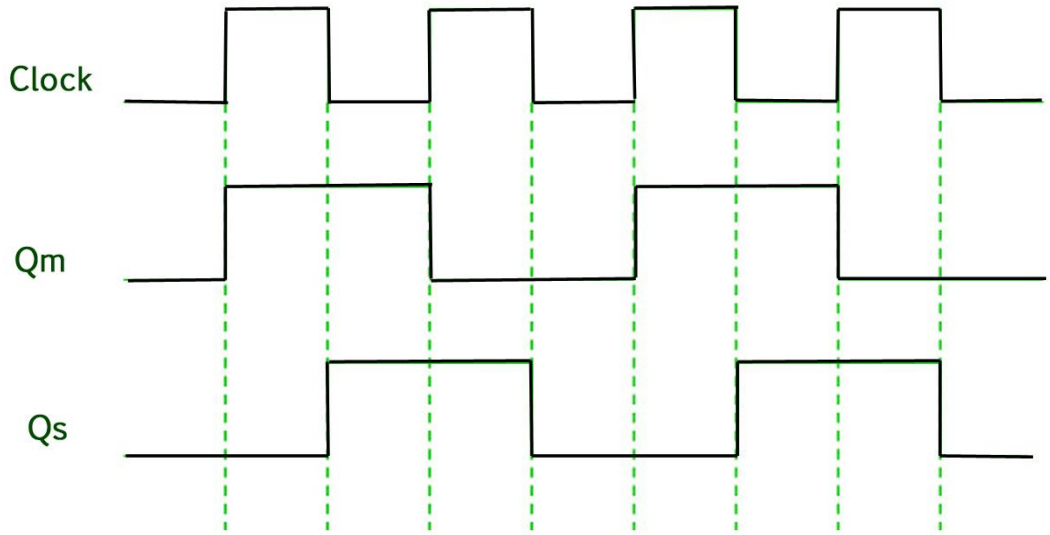
Master-slave JK Flip-Flop

Working of a master slave flip flop –

1. When the clock pulse goes to 1, the slave is isolated; J and K inputs may affect the state of the system. The slave flip-flop is isolated until the CP goes to 0. When the CP goes back to 0, information is passed from the master flip-flop to the slave and output is obtained.
2. Firstly the master flip flop is positive level triggered and the slave flip flop is negative level triggered, so the master responds before the slave.

3. If J=0 and K=1, the high Q' output of the master goes to the K input of the slave and the clock forces the slave to reset, thus the slave copies the master.
4. If J=1 and K=0, the high Q output of the master goes to the J input of the slave and the Negative transition of the clock sets the slave, copying the master.
5. If J=1 and K=1, it toggles on the positive transition of the clock and thus the slave toggles on the negative transition of the clock.
6. If J=0 and K=0, the flip flop is disabled and Q remains unchanged.

Timing Diagram of a Master flip flop –

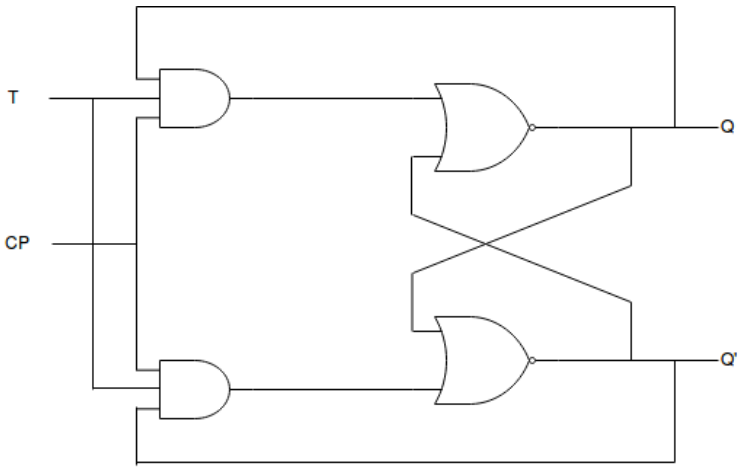


1. When the Clock pulse is high the output of master is high and remains high till the clock is low because the state is stored.
2. Now the output of master becomes low when the clock pulse becomes high again and remains low until the clock becomes high again.
3. Thus toggling takes place for a clock cycle.
4. When the clock pulse is high, the master is operational but not the slave thus the output of the slave remains low till the clock remains high.
5. When the clock is low, the slave becomes operational and remains high until the clock again becomes low.
6. Toggling takes place during the whole process since the output is changing once in a cycle.

This makes the Master-Slave J-K flip flop a Synchronous device as it only passes data with the timing of the clock signal.

T Flip-Flop

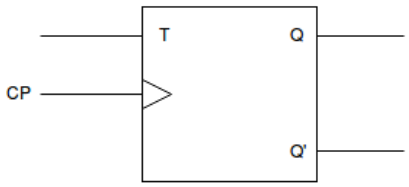
T flip-flop is a much simpler version of the J-K flip-flop.



a) Logic diagram

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

b) Transition table



c) Graphical symbol

fig. Clocked T flip flop

Both the J and K inputs are connected and are also called as a single input J-K Flip-flop.

Answer the following questions:

1. **How do you differentiate between a latch and a flip-flop?**

   Ans:

Flip-flop	Latch
Flip-flop is a bistable device i.e., it has two stable states that are represented as 0 and 1.	Latch is also a bistable device whose states are also represented as 0 and 1.
It checks the inputs but changes the output only at times defined by the clock signal or any other control signal.	It checks the inputs continuously and responds to the changes in inputs immediately.
It is an edge triggered device.	It is a level triggered device.
Gates like NOR, NOT, AND, NAND are building blocks of flip flops.	These are also made up of gates.
They are classified into asynchronous or synchronous flip-flops.	There is no such classification in latches.
It forms the building blocks of many sequential circuits like counters.	These can be used for the designing of sequential circuits but are not generally preferred.
A Flip-flop always have a clock signal	latches doesn't have a clock signal
Flip-flop can be built from Latches	Latches can't build from gates

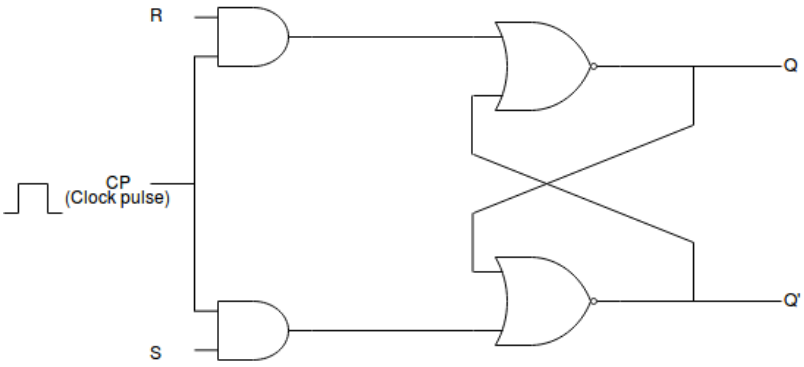
2. **Differentiate between sequential and combinational circuits.**

   Ans:

Combinational Circuit	Sequential Circuit
Combinational Circuit is the type of circuit in which output is independent of time and only relies on the input present at that particular instant.	On other hand Sequential circuit is the type of circuit where output not only relies on the current input but also depends on the previous output.
In Combinational circuit as output does not depend on the time instant, no feedback is required for its next output generation.	On other hand in case of Sequential circuit output relies on its previous feedback so output of previous input is being transferred as feedback used with input for next output generation.
As the input of current instant is only required in case of Combinational circuit, it is faster and better in performance as compared to that of Sequential circuit.	On other hand Sequential circuit are comparatively slower and has low performance as compared to that of Combinational circuit.
No implementation of feedback makes the combinational circuit less complex as compared to sequential circuit.	However on other hand implementation of feedback makes sequential circuit more complex as compared to combinational circuit.
Elementary building blocks for combinational circuit are logic gates.	On other hand building blocks for sequential circuit are flip flops..
Combinational circuit are mainly used for arithmetic as well as Boolean operations.	On other hand Sequential circuit is mainly used for storing data.

3. Draw symbol and truth-table for S-R flip-flop.

Ans:



a) Logic diagram

fig: Clocked SR flip flop

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

b) Truth table

4. What is the meaning of 'toggle'?

Ans: Toggling means when output changes to its complement on applying Clock signal. So if you assume the initial output to be Q(0 or 1) then after toggle the output state will be Q' (1 or 0 respectively).

Conditions for toggle in JK- flip flop:

1. Both J and K should be 1.
2. Clock should be present(for edge triggered, ‘edge’ should be present, and for level, proper 'level' should be present)

5. What is race around condition? How this problem can be avoided?

Ans:
Race around condition means toggling is happening at the output many a time within a single clock period.

In the case of JK flip flop (master slave) when our input is J=K=1, then flip flop is to be called in toggle state which means it will switch the output only one time in every clock cycle from one state to another state.

But for the clocked latch JK when input is J=K=1, then clocked latch is to be called in toggle state with race around condition (*provided clock period is greater than the signal time period*) which means our output is toggling many a time within a single clock period.

To avoid the race around condition we use JK Flip flop (master slave) in place of clocked latch.

6. Write a short note on flip-flops.

Ans: Flip-flop is a basic digital memory circuit, which stores one bit of information. Flip-flops are the fundamental blocks of most sequential circuits. It is also known as a bistable multivibrator or a binary or one-bit memory. Flip-flops are used as memory elements in sequential circuit.

The output is obtained in a sequential circuit from combinational circuit or flip-flop or both. The state of flip-flop changes at active state of clock pulses and remains unaffected when the clock pulse is not active. In particular, clocked flip flops serve as memory elements in synchronous sequential Circuits and unclocked flip-flops (i.e., latches) serve as memory elements in asynchronous sequential circuits.

7. List out the applications of flip-flop.

Ans:

Applications of Flop-Flops:

a. Bounce elimination switch
b. Parallel Data Storage in Registers
c. Transfer of Data from one bit to another.
d. Counters
e. Frequency Division

8. With relevant diagram explain the working of JK flip flop.

Ans:

A J-K flip-flop can be constructed from the clocked S-R flip-flop by adding two AND gate as shown in figure 3.4. The input $J.\overline{Q}$ is applied to input S and K.Q is applied to input R of S-R flip-flop. The graphic symbol of J-K flip-flop is shown in figure 3.4 b.

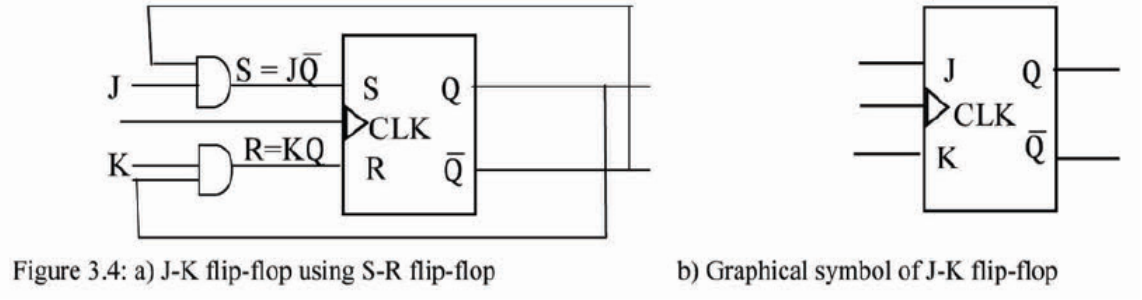


Figure 3.4: a) J-K flip-flop using S-R flip-flop b) Graphical symbol of J-K flip-flop

The working of J-K flip-flop is described below:

1. When J and K both are 0, the input of the basic S-R flip-flop are S = 0 and R = 0. Under this condition the flip-flop remains in the same state i.e. Q_n.

2. When J = 0 and K = 1 and if S-R flip-flop is in SET state i.e. $Q_n = 1$ and $\overline{Q}_n = 0$. The J-K flip - flop resets on the application of clock pulse i.e. $Q_n = 0$ and $\overline{Q}_n = 1$. If the previous state of S-R flip-flop is already RESET i.e. $Q_n = 0$ and $\overline{Q}_n = 1$, In this case flip-flop will remain in the same state.

3. When J = 1 and K = 0 and if the previous state of S-R flip-flop is SET $Q_n = 1$ and $\overline{Q}_n = 0$, the output of J-K flip-flop will be in SET state on application of clock pulse. If the previous state of S-R flip-flop is RESET i.e. $Q_n = 0$ and $\overline{Q}_n = 1$, the output of J-K flip-flop will also remain in SET condition.

4. The condition J = 1, K = 1 need to be discussed carefully. When J = 1, K = 1, and previous state is a SET state i.e. $Q_n = 1$ and $\overline{Q}_n = 0$, then $S = J.\overline{Q}_n = 0$ and $R = K.Q_n = 1$. Since S = 0 and R = 1, this resets the flip-flop on the application of a clock pulse i.e. flip-flop toggles form SET to RESET state. Toggle means to switch to the opposite state.

   Similarly when J = 1, K = 1, if the previous state is a RESET state, flip-flop toggles form RESET to SET state on the application of a clock pulse. Hence when J = 1, K = 1 flip-flop toggles on application of next clock pulse.

The characteristic equation of J-K flip-flop is $Q_n+1 = J.\overline{Q}_n + K.Q_n$

The truth table of J-K flip-flop is shown in table 3.3.

Table 3.3 Truth table of J K Flip-flop

CLK	Input		Output	Action
	J	K	Q_n+1	
X	0	0	Q_n	No Change
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	$\overline{Q}_n$	Toggle

9. State few characteristics of sequential circuits.

Ans:

- In this output depends upon present as well as past input.
- Speed is slow.
- It is designed tough as compared to combinational circuits.
- There exists a feedback path between input and output.
- This is time dependent.
- Elementary building blocks: Flip-flops
- Mainly used for storing data.
- Sequential circuits have capability to store any state or to retain earlier state.
- As sequential circuits are clock dependent they need triggering.
- These circuits have memory element.
- It is not easy to use and handle.