

Sequential Logic Circuits

Introduction

All the digital systems designed and analyzed so far are based upon combinational logic where the output levels at any instant of time depend only upon the levels present at the inputs at that time. Such systems are said to be memoryless systems. However there are many applications in which digital outputs are required to be generated in accordance with the sequence in which the input signals are received, which cannot be accomplished using combinational circuits. These applications require outputs to be generated, they are not only dependent on the level present at the inputs at that time, but also on the state of circuit. i.e. on the prior input level conditions. The past history is provided by feedback from the output back to the input. Thus, sequential circuits are made up of combinational circuits and memory elements.

Figure (5.1) shows the block diagram of a sequential circuit. It consists of a combinational circuit, which accepts digital signals from external inputs and from the outputs of memory elements connected in feedback path and generates signals for external outputs and for the inputs to memory element. The Table (5.1) the comparison between combinational and sequential circuit.

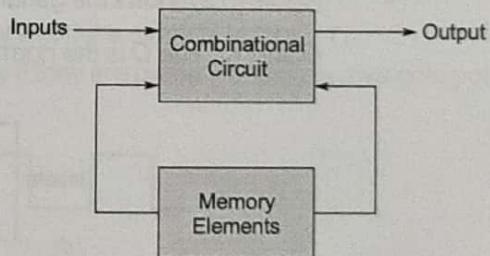


Figure-5.1 : Block diagram of a sequential circuit

Table-5.1 Comparison of combinational and sequential circuit

Combinational Circuits	Sequential Circuits
<ol style="list-style-type: none"> The output variables at any instant of time depends only on the present input variables. These circuits do not require any memory element, hence called memory less system. Combinational circuits are faster. They are easy to design. 	<ol style="list-style-type: none"> The output variables at any instant of time depends not only on the present input variables but also on the past history of the system. To store the past history of the input variables, memory unit is required. Sequential circuits are slower. They are comparatively harder to design

- The sequential circuits are classified as synchronous sequential circuits and asynchronous sequential circuits depending on the timing of their signals.

(i) Synchronous Sequential Circuit

- The change in input signals can affect memory element upon activation of clock signals.
- The maximum operational speed of clock depends on time delays involved.
- In this circuit, memory elements are "clocked flip-flops".
- It is easier to design.
- It is generally "edge triggered".

(ii) Asynchronous Sequential Circuit

- The change in input signals can affect memory element at any instant of time.
- Because of absence of clock, this circuit can operate faster than synchronous circuit.
- In this circuit, memory elements are either "unlocked flip-flops" or time delay elements.
- More difficult to design.
- It is generally "Level triggered".

NOTE: Clock is a periodically recurring pulse. It is generated by a pulse generator.

5.1 Latches and Flip-Flops

- A basic memory cell is a circuit that stores one bit of information. This one bit memory element is called flip-flop. The flip-flop is made up of an assembly of logic gates. Even though a logic gate by itself has no storage capacity, several logic gates can be connected together in such a way that permit information to be stored. Flip-flops are the basic building blocks of most sequential circuit.
- A flip-flop, known more formally as a bistable multivibrator, has two stable states. It can remain in either of the states indefinitely, and the state can be changed by applying proper triggering signal.
- Flip flop is also called a binary or one-bit memory.
- Figure (5.2) shows the general symbol used for a flip-flop. The flip-flop has two outputs, labelled as Q and \bar{Q} . The Q is the normal output of the flip-flop and \bar{Q} is the inverted output.

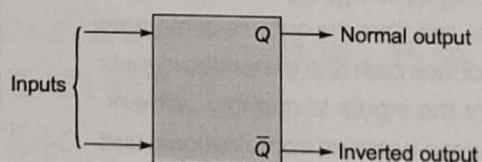


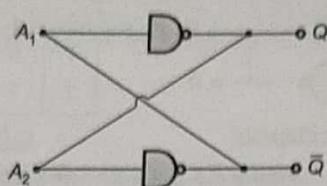
Figure-5.2 General symbol of a flip-flop

- In its simplest form, flip-flop is called a 'Latch', since it latches (or locks) data in it. In latch there is no facility to read its contents. They are temporary storage devices, ideally suited for storing information between processing units and input units. The main difference between a latch and a flip-flop is in method used for changing their state. Latches are generally unclocked.
- The below Table (5.3) lists the basic difference between flip-flop and latch as:

Table-5.2 Comparison between flip-flop and latch

Latch	Flip-Flop
1. Latches use level triggering	1. Flip-flops use edge triggering.
2. Asynchronous inputs	2. Synchronous inputs
3. The output changes as per the input till enable is high.	3. The output changes as per the input only at triggering point.

- The flip-flop can also be realized by the cross connection of NAND gates or NOR gates. The two NAND gates connected in feedback loop are shown in *Figure (5.3)*.

**Figure-5.3 Basic memory element**

where,

$$Q = \overline{A_1} = A_2$$

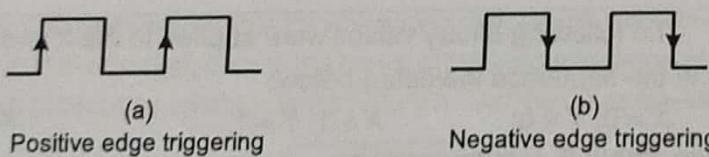
$$\bar{Q} = \overline{A_2} = A_1$$

As we know, the two outputs Q and \bar{Q} are always complement of each other. It can exist in two stable states i.e. set and reset. In set state, Q is HIGH (logic 1) and \bar{Q} is LOW (logic 0); whereas in reset state \bar{Q} is HIGH (logic 1) and Q is LOW (logic 0). For flip flop to act as a memory element it should retain the information stored in it.

5.1.1 Triggering

Triggering is used to initiate the operation of latches or flip-flops. Its main purpose is to synchronize latches or flip-flops. It is classified as:

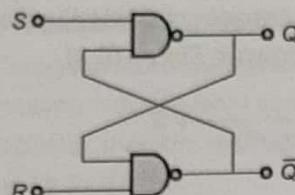
- (i) Level triggering
- (ii) Edge triggering
- In level triggering, input signals affect the flip-flop only when the clock is at logic '1'.
- In edge triggering, input signals affect the flip-flop only if they are present at the positive going or negative going edge of the clock pulse.

**Figure-5.4 Edge triggering**

- In level triggering circuit the output may change several times in a single clock, whereas in edge triggering circuit the output will change only once in a single clock.

5.1.2 The S-R Latch

- The simplest type of flip-flop is called a S-R latch. It can be constructed with two cross coupled NAND gates or two cross-coupled NOR gates and the two inputs labeled *S* for set and *R* for reset. The two outputs are Q and \bar{Q} .
- The S-R latch constructed using two cross coupled NAND gates is shown in *Figure 5.5 (a)*. Note that the output of each gate is connected to one of the inputs of other gate. The latch works as per the truth table shown in *Figure 5.5 (b)*.



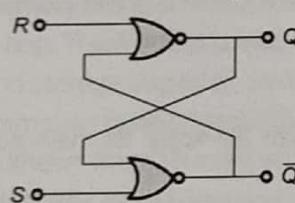
(a) Logic diagram

S	R	Q_{n+1}	State
0	0	x	Forbidden
0	1	1	Set
1	0	0	Reset
1	1	Q_n	Halt

(b) Truth table

Figure-5.5 S-R Latch with NAND gates

- The S-R latch with two cross-coupled NAND operates with both inputs normally at 1, unless the state of the latch is changed. The application of '0' to the S input causes output Q to go to '1', putting the latch in the 'Set' state. When the S input goes back to '1', the circuit remains in the set state. After both inputs go back to '1', we are allowed to change the state of the latch by placing a '0' in the R input. The action causes the circuit to go to reset and return to '1'. The condition is avoided in NAND latch when both the inputs are '0' at the same time is called invalid or forbidden state.
- Figure 5.6 (a) and (b) shows the logic diagram and the truth table for a NOR S-R latch. The operation of this latch is totally reverse of the operation of the NAND latch.



(a) Logic diagram

S	R	Q_{n+1}	State
0	0	Q_n	Halt
0	1	0	Reset
1	0	1	Set
1	1	x	Forbidden

(b) Truth table

Figure-5.6 S-R latch switch NOR gates

- If all the 0's are replaced by 1's and 1's by 0's in Figure 5.6 (b), we get the same truth table as that of NAND gate latch shown in Figure 5.5 (b).

Example-5.1 The following binary values were applied to the X and Y inputs of the NAND latch shown in the figure in the sequence indicated below:

$$X = 0, Y = 1; \quad X = 0, Y = 0; \quad X = 1, Y = 1.$$

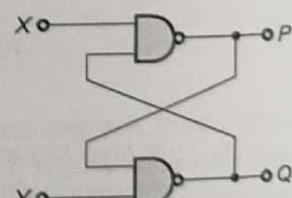
The corresponding stable P, Q outputs will be

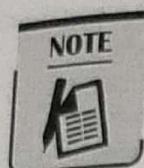
- (a) $P = 1, Q = 0; P = 1, Q = 0$ or $P = 1, Q = 0$ or $P = 0, Q = 1$
 (b) $P = 1, Q = 0; P = 0, Q = 1$ or $P = 0, Q = 1$ or $P = 0, Q = 1$
 (c) $P = 1, Q = 0; P = 1, Q = 1; P = 1, Q = 0$ or $P = 0, Q = 1$
 (d) $P = 1, Q = 0; P = 1, Q = 1$ or $P = 1, Q = 1$

Solution: (c)

As given in the truth table of NAND-gate latch circuit, we get

Inputs		Outputs	
X	Y	P	Q
0	1	1	0
0	0	1	1 (Invalid)
1	1	1	0 } (Previous)





In S-R latch if both the gates are enabled (for NAND = '1' and for NOR = '0') the output remains in previous state and when both the gates are disabled (NAND = '0' and NOR = '1') the output remains in invalid (prohibited) state.

5.1.3 Clocked Flip-Flop and Clock Signal

- Digital systems can operate either Asynchronously or Synchronously.
- In synchronous system, the exact time at which any output can change states is determined by a signal commonly known as the "Clock signal". The outputs can change states only when the clock makes a transition (also called edges).
- Clock changes from 0 to 1 → Positive going transition (PGT).
- Clock changes from 1 to 0 → Negative going transition (NGT).

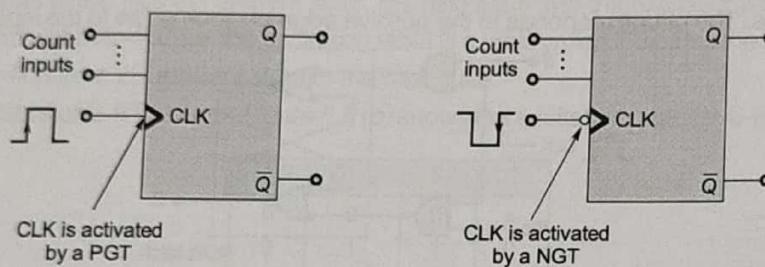


Figure-5.7: Clocked Flip-flop

- Clocked FFs have a clock input i.e. typically labelled CLK, CK or CP. This is indicated by a small triangle on the CLK input.
- Ultimately we can say that the Control Inputs get the FF outputs ready to change. While the active transition at the CLK inputs actually triggers the change i.e. the control inputs control the "What" (what state the output will go to); the CLK input determines the "When".

Setup Time (t_s)

The set-up time (t_s) is the minimum time for which the control levels need to be maintained constant on the input terminals of the flip-flop, prior to the arrival of the triggering edge of the clock pulse, in order to enable the flip-flop to respond reliably. Below Figure 5.7 (a) illustrates the set-up time for a D flip-flop.

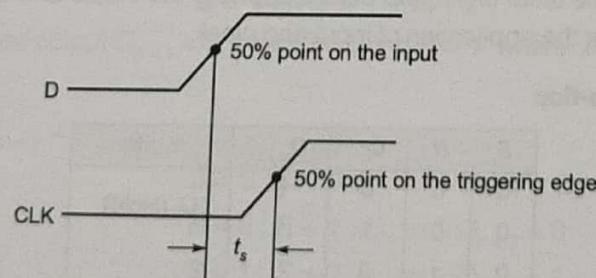


Figure-5.7(a)

Hold Time (t_h)

The hold time (t_h) is the minimum time for which the control signals need to be maintained constant at the input terminals of the flip-flop, after the arrival of the triggering edge of the clock pulse, in order to enable the flip-flop to respond reliably. Below Figure 5.7 (b) illustrates the hold time for a D flip-flop.

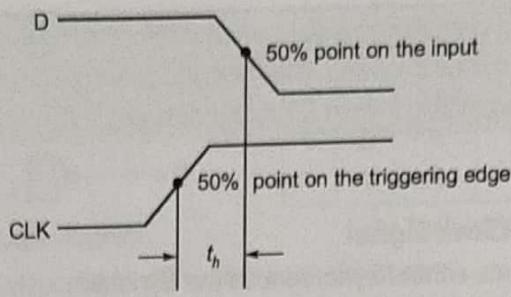


Figure-5.7(b)

NOTE: In IC-FFs the minimum allowable t_S and t_H values are in the range of nanoseconds.

5.1.4 Clocked S-R Flip-Flop

The Figure 5.8 (a) shows the clocked S-R flip-flop. The circuit is similar to S-R latch except clk signal and two AND gates. The circuit responds to the positive edge of clock pulse to the inputs S & R.

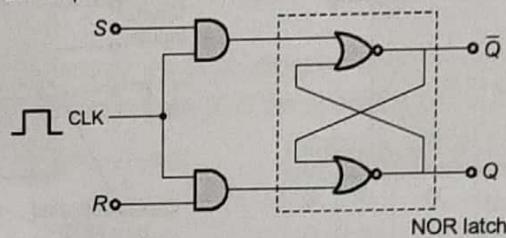


Figure-5.8 (a) Logic diagram of S-R flip-flop

Truth Table for S-R flip-flop

Clock	S	R	Q_{n+1}	State
0	x	x	Q_n	
1	0	0	Q_n	Hold
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	x	Invalid

Figure-5.8 (b) Truth table for S-R flip-flop

Here, Q_n represents the state of flip-flop before applying the inputs and Q_{n+1} represents the state of flip-flop output 'Q' after the application of input and clock.

Characteristic Table of S-R Flip-flop

S	R	Q_n	Q_{n+1}	State
0	0	0	0	Q_n (Hold)
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	x	Forbidden/
1	1	1	x	Invalid

Figure-5.8 (c) Characteristic table of S-R flip-flop

Characteristic Equation of S-R Flip-flop

The characteristic equation is an algebraic expression for the binary information of the characteristic table. It specifies the value of next state of a flip-flop in terms of its present state and present excitation.

To obtain the characteristic equation of S-R flip-flop the K-map for the next state Q_{n+1} in terms of its present state and inputs is shown in Figure 5.8 (d).

∴ Characteristic equation of S-R flip-flop is $Q_{n+1} = S + \bar{R}Q_n$

$S \backslash RQ_n$	00	01	11	10
0		1		
1	1	1	X	X

Figure-5.8 (d) K-map

Excitation Table

The truth table of flip-flop refers to the operation characteristic of the flip-flop, but in the designing of sequential circuits, we often face the situations where the present state and the next state of the flip-flop is specified and we have to find out the input conditions that must prevail for the desired output condition. Thus, the table which lists the present state, the next state and the excitations of a flip-flop called the excitation table of a flip-flop i.e. the excitation table is a table which indicates the excitations required to take the flip-flop from the present state to the next state.

Excitation table of S-R flip-flop: Figure 5.8 (e) shows the excitation table for S-R flip-flop.

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

Figure-5.8 (e) Excitation table of S-R flip-flop

The description for excitation table is as follows:

Case-A:

when, $Q_n = 0$ and $Q_{n+1} = 0$

For

$$S = 0 ; R = 0 \Rightarrow Q_{n+1} = Q_n = 0$$

$$S = 0 ; R = 1 \Rightarrow Q_{n+1} = 0$$

$$S = 1 ; R = 0 \Rightarrow Q_{n+1} = 1$$

$$S = 1 ; R = 1 \Rightarrow x$$

Therefore, the desired output $Q_{n+1} = 0$, when $S = 0$ and $R = 'x'$ where 'x' is a don't care.

Case-B:

when, $Q_n = 0$ and $Q_{n+1} = 1$

For

$$S = 0 ; R = 0 \Rightarrow Q_{n+1} = Q_n = 0$$

$$S = 0 ; R = 1 \Rightarrow Q_{n+1} = 0$$

$$S = 1 ; R = 0 \Rightarrow Q_{n+1} = 1$$

$$S = 1 ; R = 1 \Rightarrow Q_{n+1} = x$$

Therefore, desired output $Q_{n+1} = 0$, when $S = 1$ and $R = 0$.

Case-C:

when, $Q_n = 1$ and $Q_{n+1} = 0$

For

$$S = 0 ; R = 0 \Rightarrow Q_{n+1} = Q_n = 1$$

$$\begin{aligned} S = 0; R = 1 &\Rightarrow Q_{n+1} = 0 \\ S = 1; R = 0 &\Rightarrow Q_{n+1} = 1 \\ S = 1; R = 1 &\Rightarrow Q_{n+1} = x \end{aligned}$$

So, desired output $Q_{n+1} = 0$, when $S = 0$ and $R = 1$.

Case-D:

when, $Q_n = 1$ and $Q_{n+1} = 1$

For

$$\begin{aligned} S = 0; R = 0 &\Rightarrow Q_{n+1} = Q_n = 1 \\ S = 0; R = 1 &\Rightarrow Q_{n+1} = 0 \\ S = 1; R = 0 &\Rightarrow Q_{n+1} = 1 \\ S = 1; R = 1 &\Rightarrow Q_{n+1} = x \end{aligned}$$

The desired output $Q_{n+1} = 1$, when $R = 0$ and $S = x$.

Graphical symbol of S-R latch

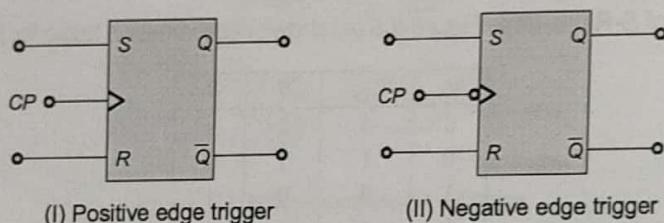


Figure 5.8 (f) Graphical symbol of S-R latch

- Disadvantages:** Invalid states are present when both the inputs 'S' and 'R' are made HIGH (logic 1). To avoid this difficulty we use J-K flip-flop.

DO YOU KNOW? Truth table of S-R flip-flop using NAND gates = S-R latch using NOR gates.

5.1.5 J-K Flip-Flop

The J-K flip-flop is a refinement of the S-R flip-flop in which the indeterminate (invalid) state of the S-R type is defined. In the Figure 5.9 (a) shows the logic diagram of J-K flip-flop with data input J and K ANDed with \bar{Q} and Q respectively to obtain S and R inputs i.e.

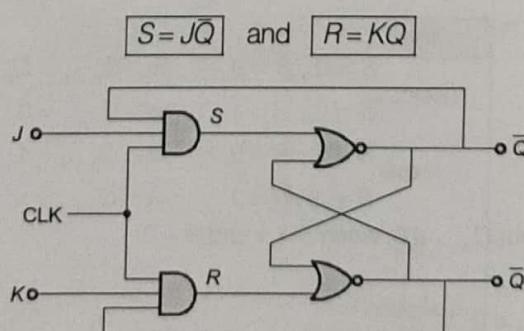


Figure 5.9 (a) Logic diagram of J-K flip-flop

Characteristics Table of J-K Flip-flop

J	K	Q_n	Q_{n+1}	State
0	0	0	0	Q_n (Hold)
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

Figure-5.9 (b) Characteristic table of J-K flip-flop

Truth Table of J-K Flip-flop

Clock	J	K	Q_{n+1}	State
0	x	x	Q_n	Hold
1	0	0	Q_n	
1	0	1	0	Reset
1	1	0	1	
1	1	1	\bar{Q}_n	Toggle

Figure-5.9 (c)

Characteristic Equation of J-K Flip-flop

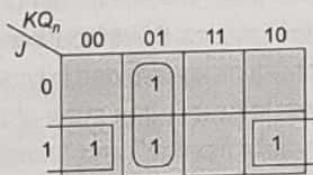


Figure-5.9 (d) K-map

∴ Characteristic equation is $Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n$

Excitation Table of J-K Flip-flop

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

Figure-5.9 (e) Excitation table of J-K flip-flop

Case-A

when, $Q_n = 0$ and $Q_{n+1} = 0$

This condition can happen with either $J = 0$ and $K = 0$ or $J = 0$ and $K = 1$ [Characteristic table].

Therefore, the desired output $Q_{n+1} = 0$ is obtained when $J = 0$ and $K = X$ (don't care).

Case-B

when, $Q_n = 0$ and $Q_{n+1} = 1$

This can happen with either $J = 1$ and $K = 0$ or $J = 1$ and $K = 1$ (toggle condition).

Therefore the desired output $Q_{n+1} = 1$ is obtained when, $J = 1$ and $K = X$.

Case-C

when, $Q_n = 1$ and $Q_{n+1} = 0$

This can happen with either $J = 0$ and $K = 1$ or $J = 1$ and $K = 1$.

Therefore, the desired output $Q_{n+1} = 0$ is obtained when $J = X$ and $K = 1$.

Case-D

when, $Q_n = 1$ and $Q_{n+1} = 1$

This condition can happen with either $J = 0$ and $K = 0$ or $J = 1$ and $K = 0$.

Thus, the desired output $Q_{n+1} = 1$ is obtained with $J = X$ and $K = 0$.

Graphical Symbol

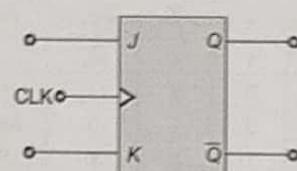


Figure-5.9 (f) Graphical symbol of J-K flip-flop

5.1.6 D Flip-Flop

- From the truth table of S-R flip flop it is clear that the output of S-R flip flop is in unpredictable state when the inputs are same (i.e. when $S = R = 0$ then $Q = \text{Halt}$ and when $S = R = 1$; $Q = \text{invalid}$). Therefore in many practical applications, these input conditions are not required. Thus the modified S-R flip flop in which such conditions are avoided is known as D flip flop, shown in Figure 5.10 (a).
- It is a flip-flop with delay equal to exactly one cycle of CLK.
- It is also called "data transmission flip-flop", and "transparent latch".

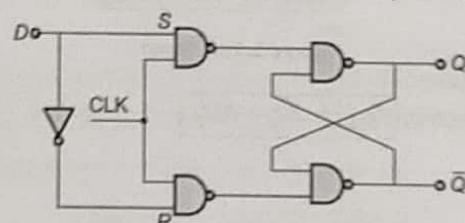


Figure-5.10 (a) Logic diagram of D flip-flop

where,

$$S = D \quad \text{and} \quad R = \bar{D}$$

Truth Table of D Flip-flop

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

Figure-5.10 (b) Truth table of D flip-flop

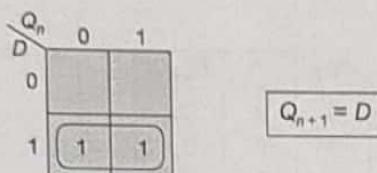


Characteristic Table of D Flip-flop

D	Q_n	Q_{n+1}	State
0	0	0	Reset
0	1	0	Reset
1	0	1	Set
1	1	1	Set

Figure-5.10 (c) Characteristic table of D flip-flop

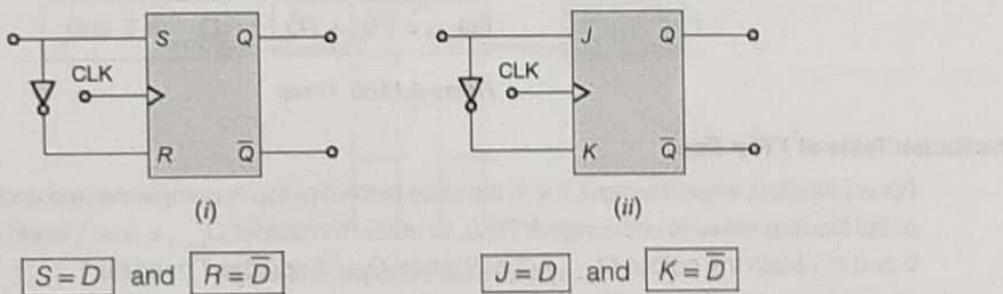
The characteristic equation of D flip-flop is

**Figure-5.10 (d) K-map****The excitation Table of D Flip-flop**

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

Figure-5.10 (e) Excitation table of D flip-flop

For a D flip-flop, the next state is always equal to the D input and it is independent of the present state. Therefore, D must be '0' if Q_{n+1} has to be 0, and 1 if Q_{n+1} has to be 1 regardless of the value of Q_n .

Graphical Symbol**Figure-5.10 (f) Graphical symbol of D flip-flop****5.1.7 T Flip-Flop**

- Figure 5.11 (a) shows the logical diagram of T flip-flop. T flip-flop is a single input version of J-K flip-flop and can be obtained from J-K flip-flop if J and K inputs are tied together.
- The designation 'T' comes from the ability of the flip-flop to 'Toggle' or 'Change' state.

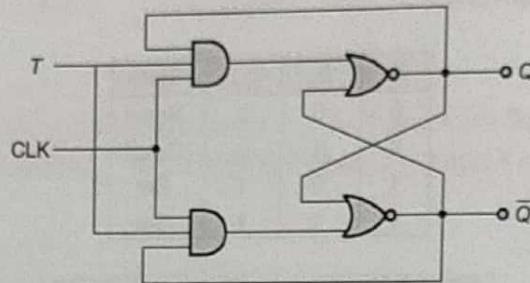


Figure-5.11 (a) Logic diagram T flip-flop

Truth Table of T Flip-flop

CLK	T	Q_{n+1}
0	X	Q_n
1	0	Q_n → Hold
1	1	Q_n → Toggle

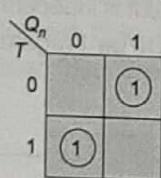
Figure-5.11 (b) Truth table

The Characteristic Table for T Flip-flop

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

Figure-5.11 (c) Characteristic table of T flip-flop

The characteristic equation of T flip-flop is



$$Q_{n+1} = \bar{T}Q_n + T\bar{Q}_n \quad \text{or} \quad Q_{n+1} = T \oplus Q_n$$

Figure-5.11 (d) K-map

Excitation Table of T Flip-flop

For a T flip-flop, when the input $T = 1$, the state of the flip-flop is complemented and when $T = 0$, the state of the flip-flop remains unchanged. Thus, to obtain the output $Q_{n+1} = 0$ or 1 when $Q_n = 0$ or 1 T must be 0 and to obtain the output $Q_{n+1} = 1$ or 0 when $Q_n = 1$ or 0 the T must be 1. Therefore, the excitation table,

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

Figure-5.11 (e) Excitation table of T flip-flop



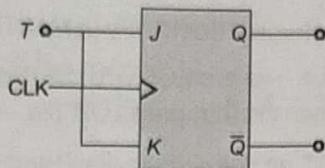
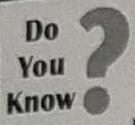
Graphical Symbol

Figure-5.11 (f) Graphical symbol of T flip-flop



Using J-K flip-flop all other flip-flops can be designed thus it is known as universal flip-flop.

J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	\bar{Q}_n

SR flip-flop

D flip flop T flip flop

5.2 Race Around Condition

- The difficulty of both the inputs to be '1' in case of S-R of invalid state is eliminated by a J-K flip-flop by using feedback connections. From output to the input as shown in Figure 5.9 (a). However the condition when (level triggered) $J = K = 1$ is not yet perfect.
- Consider $J = K = 1$ and $Q_n = 0$ and a clock (CLK) is applied. After a propagation delay time t_{pd} through two NAND gates, the output will toggle to $Q_n = 1$. Since this is feedback to the inputs, the output will toggle back to $Q_n = 0$ after another delay of $t_{pd(FF)}$.
- Thus, as long as the clock pulse is present (t_{pw}), the output will toggle at every $t_{pd(FF)}$ and at the end of the clock pulse, the value of Q_n is uncertain. This situation will continue as long as the clock pulse width t_{pw} is longer than the propagation delay (t_{pd}) of the flip-flop. Such situation is referred to as the "race around condition".

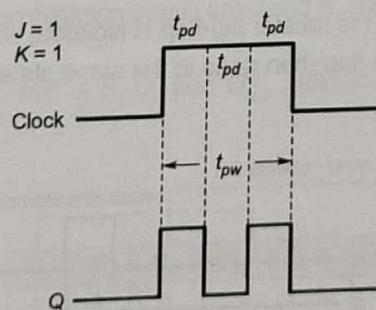


Figure-5.12 Waveform of J-K flip-flop

Thus, the "Race around condition" will occur when

- $J = K = 1$
 - When $t_{pd(FF)} < t_{pw}$
 - When level trigger is applied.
- One way to avoid this problem is to maintain $t_{pw} < t_{pd(FF)} < T$. A most practical method for overcoming this problem is use of the "Master slave configuration".

5.2.1 Master-Slave Flip-Flop

- A "M-S FF" is basically constructed from 2 FFs (a MASTER and a SLAVE) and an 'INVERTER'.
- On the rising edge of CLK (i.e. +ve edge CLK PULSE) the control inputs are used to determine the output of the "MASTER". When the CLK goes LOW (i.e. -ve edge CLK PULSE), the state of Master is transferred to the "SLAVE", whose outputs are Q and \bar{Q} .
- In the "M-S FF", output is fully dependent upon the output of SLAVE-FF.

Logic Diagram of M-S FF

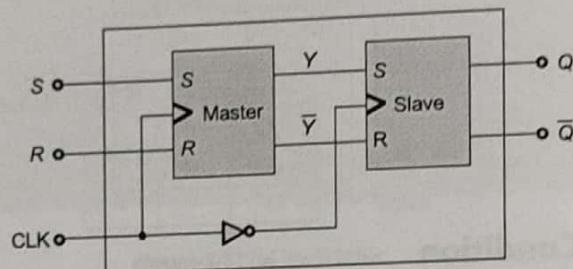


Figure-5.13 (a) Master-slave flip-flop

Operation

- When clock pulse CLK is 0, the output of the inverter is 1. Since the clock input of the slave is 1, the flip-flop is enabled and output Q is equal to Y , while \bar{Q} is equal to \bar{Y} . The master flip-flop is disabled because $CLK = 0$.
- When the pulse becomes 1, the information at the external R and S inputs is transmitted to the master flip-flop. The slave flip-flop, clock is zero because the inverter output is zero i.e. is slave flip-flop is isolated.
- When pulse returns to 0, the master flip-flop is isolated, which prevents the external inputs from affecting it. The slave flip-flop then goes to the same state as the master flip-flop.

Timing diagram in M-S FF

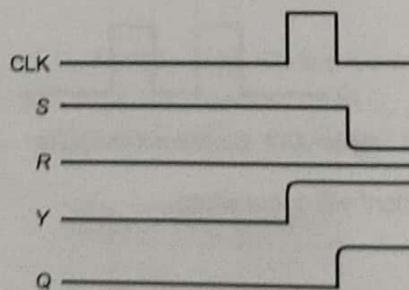


Figure-5.13 (b)

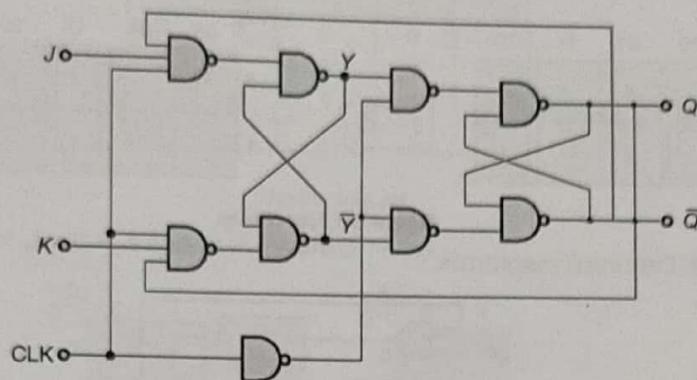
Master-Slave J-K Flip-Flop by using only NAND gates

Figure-5.13 (c)

5.3 Conversion of Flip-Flops

In order to convert one type of flip-flop into another type of flip-flop, the following steps must be considered:

- Draw the generalized block diagram as shown in *Figure 5.14*. Here, the inputs of the desired flip-flop are fed as inputs to the combinational circuit and the output of the combinational circuit is connected to the input of the available flip-flop, then the output of the available flip-flop (given) is the output of the desired flip-flop.

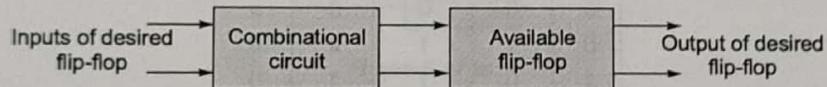


Figure-5.14 Block diagram for conversion of flip-flop

- Write the characteristic table of desired flip-flop followed by excitation table of given flip-flop.
- Using K-map drive the expressions for available flip-flop in terms of desired flip-flop.

5.3.1 Conversion of S-R Flip-Flop to J-K Flip-Flop

- Here the inputs to the combination circuit are J and K . S and R are the output of the available flip-flop (i.e. S-R FF).
- The characteristic table with J , K , Q_n and Q_{n+1} , and the excitation table for S & R is shown in *Figure 5.15 (a)*.

Excitation table					
Characteristic table					
J	K	Q_n	Q_{n+1}	S	R
0	0	0	0	0	x
0	0	1	1	x	0
0	1	0	0	0	x
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	x	0
1	1	0	1	1	0
1	1	1	0	0	1

Figure-5.15 (a) Conversion table

- The K-map for S and R is shown in Figure 5.15 (b) as

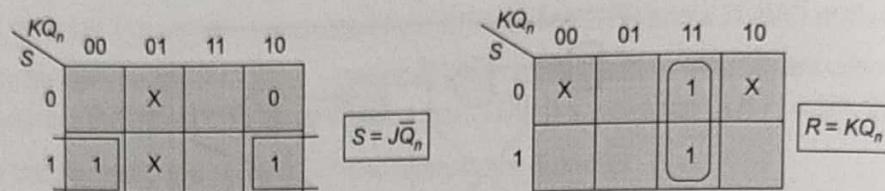


Figure-5.15 (b) K-map

- Therefore, the circuit diagram is

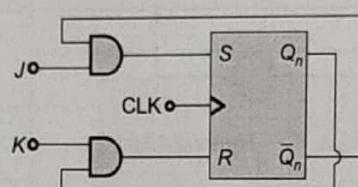


Figure-5.15 (c) Circuit diagram

5.3.2 Conversion of S-R Flip-Flop into T Flip-Flop

- Figure 5.16 (a) shows the characteristic table of T flip-flop and the excitation table of S-R flip-flop.

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

Figure-5.16 (a)

- Using K-map, we get the expression of S & R in terms of T

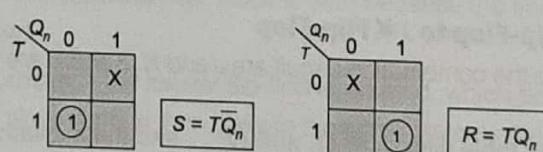


Figure-5.16 (b)

- Therefore, the circuit diagram for conversion of S-R flip-flop into T flip-flop is

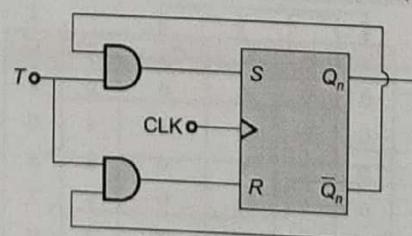


Figure-5.16 (c)

5.3.3 Conversion of J-K Flip-Flop into D Flip-Flop

- Here the available flip-flop is J-K FF and the desired flip-flop is D FF. The Figure 5.17(a) shows the characteristic table of D flip-flop with excitation table of J-K flip-flop.

D	Q_n	Q_{n+1}	J	K
0	0	0	0	x
0	1	0	x	1
1	0	1	1	x
1	1	1	x	0

Figure-5.17(a)

- The K-map for J and K is shown in Figure 5.17 (b).

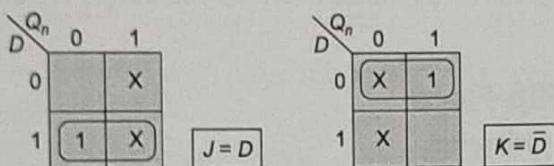


Figure-5.17(b)

- The circuit diagram of the conversion of J-K flip-flop into D flip-flop is shown in Figure 5.17 (c).

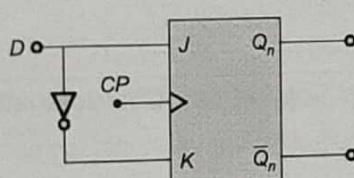


Figure-5.17(c)

- Similarly the conversion of S-R flip-flop to D flip-flop is obtained with

$$S = D \quad \text{and} \quad R = \bar{D}$$

5.3.4 Conversion of J-K Flip Flop into T Flip Flop

- The Figure 5.18 (a) shows the characteristic table of T flip-flop (desired) and the excitation table of J-K flip-flop (available).

T	Q_n	Q_{n+1}	J	K
0	0	0	0	x
0	1	1	x	0
1	0	1	1	x
1	1	0	x	1

Figure-5.18(a)

- The K-map for J and K is shown in Figure 5.18 (b).

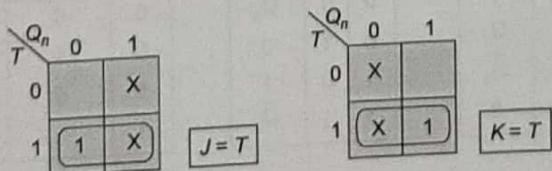


Figure-5.18(b)

- The circuit diagram for the conversion of J-K flip-flop to T flip-flop is shown in Figure 5.18 (c).

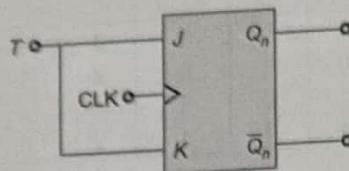


Figure-5.18 (c)

- Similarly D flip-flop can be converted into T flip-flop as: $D = T \oplus Q_n$

5.4 Applications of Flip-Flops

- As bounce elimination switch
- Register
- Counters
- Memory etc.
- Frequency division

Example-5.2 An X-Y flip flop, whose Characteristic Table is given below is to be implemented using a J-K flip flop

X	Y	Q_{n+1}
0	0	1
0	1	Q_n
1	0	\bar{Q}_n
1	1	0

This can be done by making

- $J = X, K = \bar{Y}$
- $J = \bar{X}, K = Y$
- $J = Y, K = \bar{X}$
- $J = \bar{Y}, K = X$

Solution: (d)

X-Y Truth Table

X	Y	Q_{n+1}
0	0	1
0	1	Q_n
1	0	\bar{Q}_n
1	1	0

J-K Truth Table

J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	\bar{Q}_n

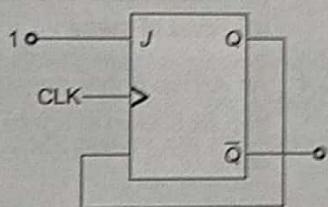
Excitation table

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

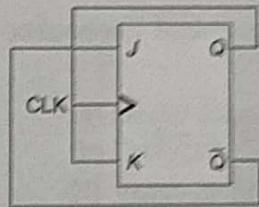
To make (X-Y) FF using (J-K) FF, (J) should be (\bar{Y}) and (K) should be (X).

Summary

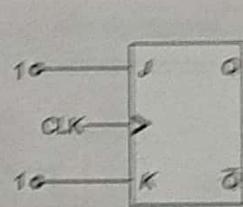
- "Reset" and "Clear" are one and the same. It is used for setting the contents of FF or memory to ZERO.
- "Preset" is the state of a FF when $Q = 1$ and $\bar{Q} = 0$.
- "Set" is the state of a FF when $Q = 0$.
- Race-around condition occurs in J-K-FF to store 1-bit of information.
- Race-around condition always arises in "Asynchronous circuits".
- A Master-slave FF consists of an S-R-FF followed by a T-FF.
- The frequency is always halved at the output of any FFs whose behaviour is same as TOGGLED-FF.
- The time required to change the voltage level from 10% to 90% is known as rise time (τ_r) whereas the time required to change the voltage level from 90% to 10% is known as fall time (τ_f).
- The pulse width on a clock is always measured at its 50% voltage level.
- The phase shift between rectangular clock waveform is referred as 'skew' and the time delay between the two clock pulse is called 'clock skew'.
- The toggle mode operation of various flip-flops are:



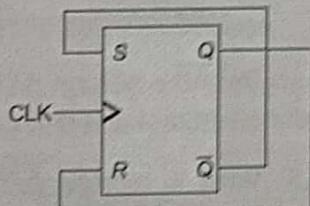
(a)



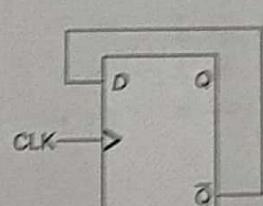
(b)



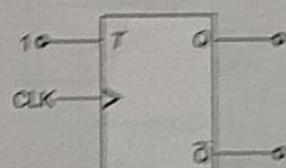
(c)



(d)



(e)



(f)



Student's Assignments

1

Q.1 What do you mean by Latch? How it is different from flip-flops? Make the logical diagram of S-R Latch by using NAND and NOR gates with appropriate Truth Table.

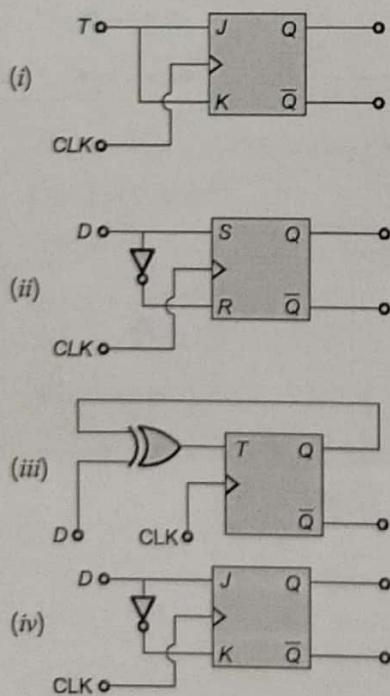
- Q.2** An AB flip-flop has four operations as:
Clear to '0', No change, Compliment and
Set to '1', when the inputs A and B are 00, 01, 10
and 11 respectively.
- Tabulate the characteristic table and drive the characteristic equation.
 - Show that the AB flip-flop can be converted to D flip-flop.



Student's Assignments

2

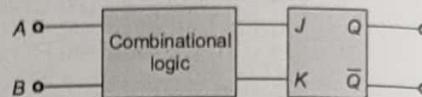
Q.1 Which of the following circuit connection represents the D-FF?



- (a) (ii) and (iii) (b) (ii), (iii) and (iv)
(c) (i), (ii) and (iii) (d) (ii) and (iii)

Q.2 Consider a circuit realisation of a combinational logic block as shown in figure to obtain the following truth table:

A	B	Q_{n+1}
0	0	\bar{Q}_n
0	1	1
1	0	Q_n
1	1	0



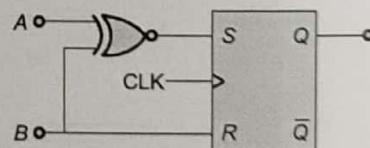
The combinational logic involves

- Only EX-OR gates
- AND and NOT gates
- NOT and EX-NOR gates
- OR and NAND gates

Q.3 Which of the following flip-flop can not be converted to D-type (delay) flip-flop.

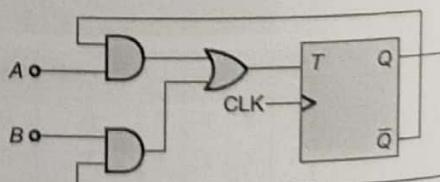
- S-R flip-flop
- J-K flip-flop
- Master slave flip-flop
- none of these

Q.4 An AB flip-flop is constructed from an S-R Flip-flop as shown in figure. The expression for next state Q^+ is



- (a) $\overline{A}\overline{B} + AQ$ (b) $\overline{A}\overline{B} + \overline{B}Q$
(c) Both A and B (d) None of above

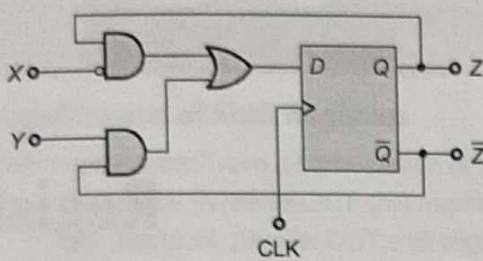
Q.5 What is represented by digital circuit given below?



- (a) A J-K flip-flop with $A = K$ and $B = J$
(b) A J-K flip-flop with $A = J$ and $B = K$
(c) An S-R flip-flop with $A = R$ and $B = S$
(d) None of the above



- Q.6 A sequential circuit using D flip-flop and logic gates is shown in the figure where X and Y are the inputs and Z is the output. The circuit is



- (a) S-R FF with input $X = R$ & $Y = S$
- (b) S-R FF with input $X = S$ & $Y = R$
- (c) J-K FF with input $X = J$ & $Y = K$
- (d) J-K FF with input $X = K$ & $Y = J$

- Q.7 For a flip-flop with provisions of preset and clear
- (a) preset and clear operations are performed simultaneously.
 - (b) while clearing, preset is disabled.
 - (c) while presetting, clear is disabled.
 - (d) both (b) and (c) are correct.

- Q.8 The outputs Q and \bar{Q} of a master-slave flip-flop are connected to its R and S inputs respectively. Its output Q when clock pulses are applied will be
- (a) fixed 0 or 1
 - (b) permanently 0
 - (c) permanently 1
 - (d) complementing with every clock pulse

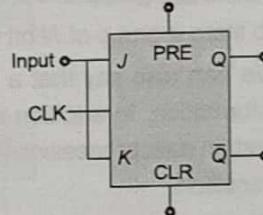
- Q.9 The output Q_n of a J-K flip-flop is 1. It changes to 0 when a clock pulse is applied. The inputs J and K are respectively
- (a) 0 and x
 - (b) 1 and x
 - (c) x and 0
 - (d) x and 1

- Q.10 The state table of a latch with inputs L and M is given below. The expression for the next state Q^+ is

L	M	Q	Q^+
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

- (a) $\bar{L}M + MQ$
- (b) $L\bar{M} + LQ$
- (c) $\bar{L}M + M\bar{Q}$
- (d) $L\bar{M} + L\bar{Q}$

- Q.11 The following configuration provides



- (a) Master slave action
- (b) Delay gate
- (c) Toggle switch
- (d) None of these

Answer Key:

- | | | | | |
|--------|--------|--------|--------|---------|
| 1. (b) | 2. (c) | 3. (d) | 4. (c) | 5. (b) |
| 6. (d) | 7. (d) | 8. (d) | 9. (d) | 10. (d) |

11. (c)



Registers

Introduction

- A register is a digital circuit with two basic functions i.e. data storage and data movement. It is basically a group of flip-flops logically connected to perform various functions.
- To store a group of N -bit word, the number of flip-flops required is ' N ' (one for each bit).
- We can also say that a register is a group of binary storage cells suitable for holding binary information. In addition to the flip-flops, a register may have combinational gates that perform certain data processing tasks. Thus a register consists of a group of FFs and gates that effect their transition.
- The simplest possible register is the one that consists of only flip-flop without any external gate, as shown in *Figure (6.1)*.

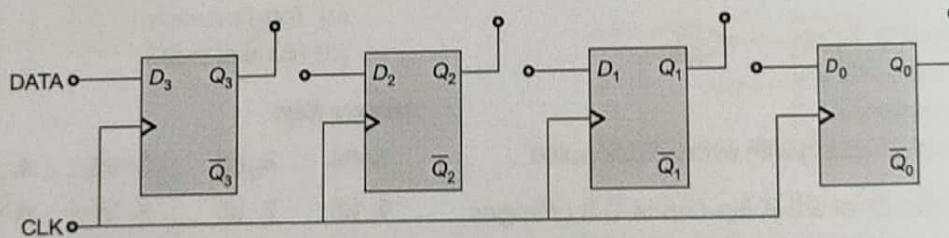


Figure-6.1 General block diagram of a register

- The data can be entered in SERIAL (one bit at a time) or in PARALLEL form (all the bits simultaneously) and can be retrieved in the serial or parallel form.

Remember: Data in serial form is called temporal code & data in parallel form is called spatial code.

6.1 Shift Register

- A register capable of shifting the binary information entered into it from an external binary source is called the "shift register".
- It is a sequential circuit mainly used to store or shift binary data either to the right (called right shift register) or to the left (called left shift register).

- The logical configuration of a shift register consists of a chain of flip-flops connected in cascade, with the output of one flip-flop, connected to the input of next flip-flop.
- All flip-flops receive a common CLK pulse which causes shift from one state to the next.
- In shift register each CLK pulse shifts the contents of register one bit position to the RIGHT or LEFT.

6.1.1 Classification of Shift Registers

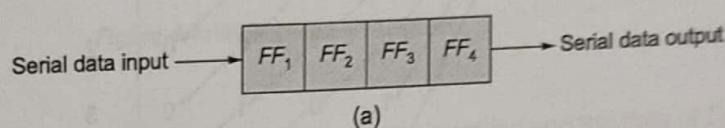
Shift registers can have a combination of serial and parallel shifted inputs and outputs including

- Serial IN, serial OUT shift register (SISO)
- Serial IN, parallel OUT shift register (SIPO)
- Parallel IN, serial OUT shift register (PISO)
- Parallel IN, parallel OUT shift register (PIPO)

The 'serial input' determines, what goes into the left flip-flop during the shift, whereas the 'serial output' is taken from the output of the right most FF prior to the application of a pulse.

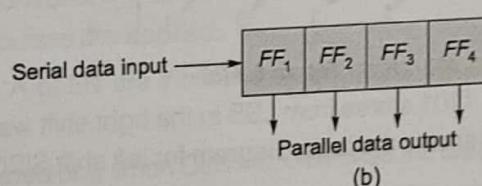
Figure (6.2) shows the classification of various registers.

- SISO



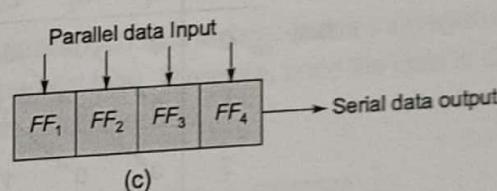
(a)

- SIPO



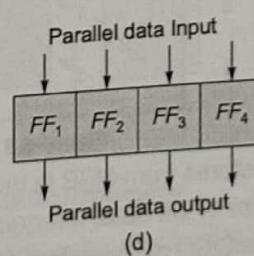
(b)

- PISO



(c)

- PIPO



(d)

Figure-6.2 Classification of various shift registers

Serial IN Serial OUT Shift Register (SISO)

- The serial IN serial OUT shift register accepts the data serially, one bit at a time on a single input line. It produces the stored binary information on its single output line in serial form.
- Figure 6.3 (a) shows the logic diagram of a 4 bit SISO register using D flip-flops. Let the binary information 1101 is applied to the input.

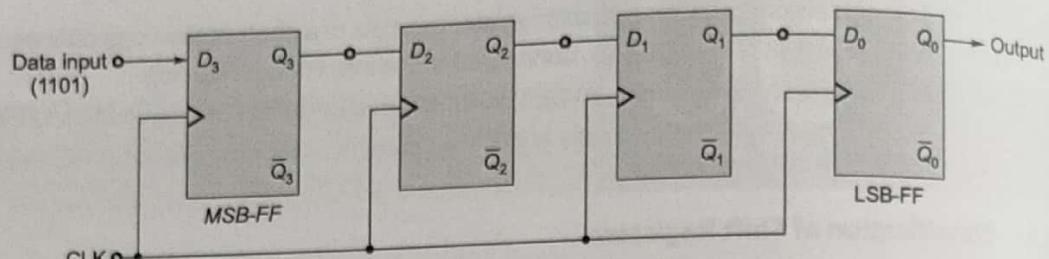
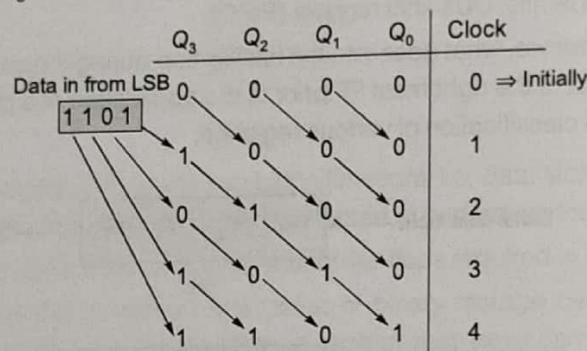


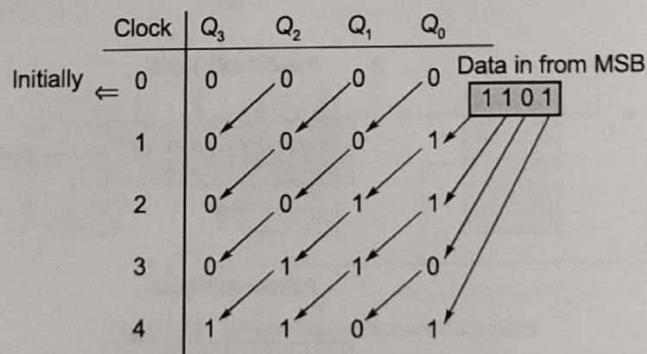
Figure 6.3 (a) Right shift SISO register

- Since shift register is reset, so initially all the flip-flop outputs are zero i.e. 0000.
- Data 1101 is applied to the input line. Therefore in the right shift SISO register, LSB data is applied at the MSB FF(D_3).



It is clear that the data 1101 stores from LSB in the right shift way after 4 clock pulses.

- Similarly Figure 6.3 (c) shows the logic diagram for left shift SISO shift register.



It is clear that the data 1101 stores from MSB in the left shift way after 4 clock pulses.

- In ' n ' bit register, to enter ' n ' bit data, it requires ' n ' clock pulses in serial form.
- If ' n ' bit data is stored in SISO register then output is taken serially for this it requires $(n-1)$ clock pulses.
- SISO register is used to provide ' n ' clock pulses delay to the input data.
- If ' T ' is the time period of clock pulse, then delay provided by SISO is nT .

Bi-Directional Shift Register

Such register which are capable of shifting the information (data) to right and left both is called "Bi-directional shift register".

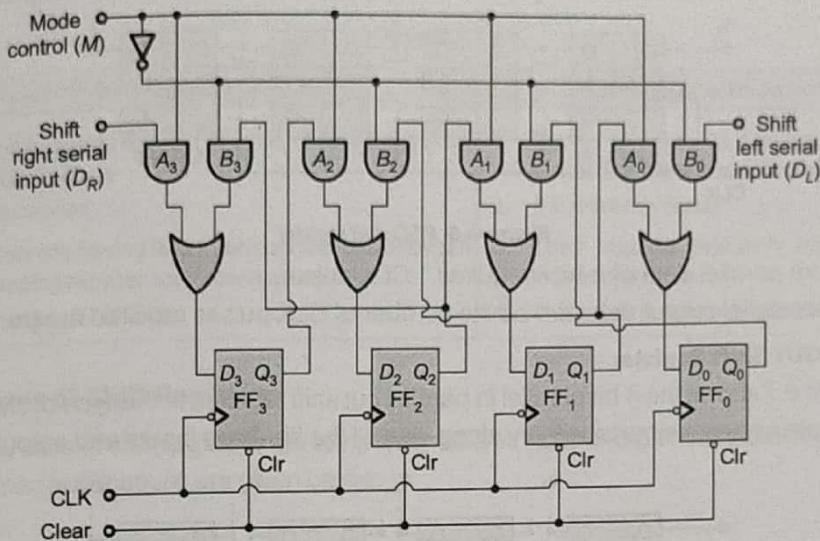


Figure-6.4: Bi-directional shift register

Operation

- When the Mode control $M = 1$, all the 'A' AND gates are enabled and the data at D_R is shifted to the right when clock pulses are applied.
- When $M = 0$, the A gates are inhibited and B gates are enabled allowing the data at D_L to be shifted to the left.
- M should be changed only when $CLK = 0$, otherwise the data stored in the register may be altered.

Serial IN Parallel OUT Shift Register (SIPO)

- Figure 6.5(a) shows, a 4 bit serial IN parallel OUT shift register. In this shift register, data is entered in serial form and output is taken in parallel form. Therefore, once the data is stored each bit is available on its respective output lines simultaneously.

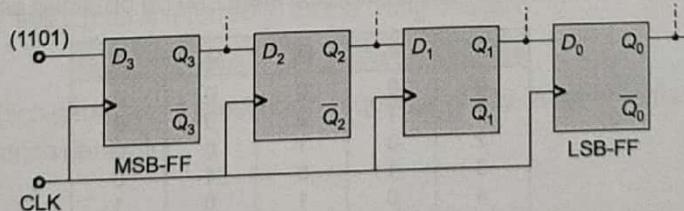


Figure-6.5 SIPO shift register

- For parallel out data the number of clock pulse required is zero.
- For storing n -bit serial input data number of CLK pulses required = n .

Parallel IN Serial OUT Shift Register (PISO)

- In PISO shift register the data bits are entered simultaneously into the respective flip-flops and the shifted data is available at the output serially.
- The Figure 6.6 shows the 4 bit PISO shift register. Consider 4-bit binary data b_3, b_2, b_1 and b_0 .

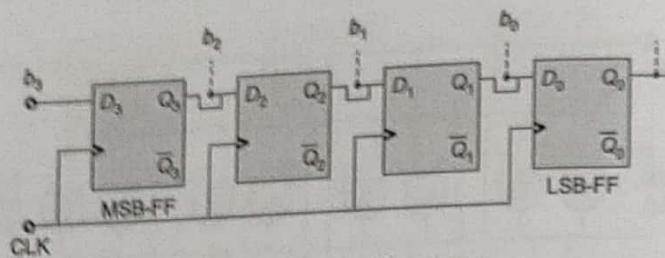


Figure-6.6 PISO shift register

- To store parallel data of n -bit it requires 1 CLK pulse.
- To store serial output data of n -bit the number of CLK pulses required are $(n - 1)$.

Parallel IN Parallel OUT Shift Register

- Figure 6.7 shows the 4 bit parallel in parallel out shift register. In this type of shift register, whole of the data appears simultaneously along with all the flip-flops inputs and outputs.

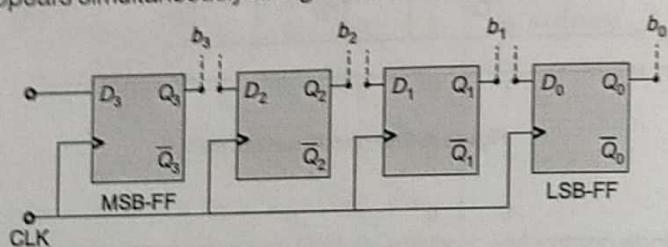
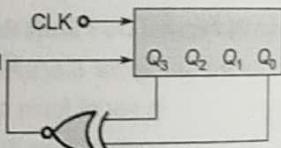


Figure-6.7 PIPO shift register

- For parallel IN data the number of clock pulse required is 1.
- For parallel OUT data the number of CLK pulse required is 0.

Example-6.1 A four bit serial in parallel out shift register is used with a feedback as shown in below figure. The shifting sequence is $Q_3 \rightarrow Q_2 \rightarrow Q_1 \rightarrow Q_0$. If the output is initially 0000, the number of clock pulses after which the output will repeat itself is

**Solution:**

From the given serial IN parallel OUT circuit the next state can be obtained as

CP	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	0	1	0
4	0	1	0	1
5	0	0	1	0
6	1	0	0	1
7	1	1	0	0
8	0	1	1	0
9	1	0	1	1
10	1	1	0	1
11	1	1	1	0
12	0	1	1	1
13	0	0	1	1
14	0	0	0	1
15	0	0	0	0

Thus, the number of clock pulses required is = 15.

6.1.2 Difference Between Serial Data Transfer and Parallel Data Transfer

Table-6.1

Serial Data Transfer	Parallel Data Transfer
<ol style="list-style-type: none"> For complete transfer of n bits of data it requires n CLK pulses. It is slower. They are having less inter-connections between sending register and receiving register. The circuit is simple. 	<ol style="list-style-type: none"> During a single CLK pulse, all the data is transferred simultaneously. It is relatively faster. They require relatively more interconnections. The circuit is relatively complex.

6.1.3 Applications of Shift Registers

The primary uses of shift registers are temporary data storage and bit manipulations. In addition to this, other common applications are given below:

Time-Delay

- A SISO shift register may be used to introduce time delay " Δt " in digital signals given by:

$$\Delta t = N \times T = N \times \frac{1}{f_c}$$

where,

N = Number of FFs

T = Time period of CLK pulse

f_c = CLK frequency

- The amount of delay can be controlled by the " f_c " or number of FFs in the SR.

Data Conversion

- Data in the serial form can be converted into parallel form by using SIPO-SR.
- Data in the parallel form can be converted into serial form by using PISO-SR.

Ring Counter

- If the output of the shift register is connected back to the input then an injected pulse will keep circulating, this circuit is referred to as the "Ring counter".

Sequence Generator

- A circuit which generates a prescribed sequence of bits, in synchronism with a CLK, is referred to as a "sequence generator".

Arithmetic Operation

- Shift registers are used to perform various arithmetic operations e.g. "serial adder" adds two binary numbers.

Example-6.2 Which of the following capabilities are available in a Universal Shift Register?

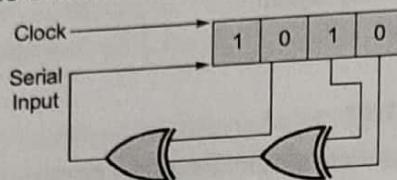
- Shift left
 - Shift right
 - Parallel load
 - Serial add
- Select the correct answer from the codes given below:
- (a) 2 and 4 only (b) 1, 2 and 3 (c) 1, 2 and 4 (d) 1, 3 and 4

Solution : (b)

Universal shift register features parallel inputs, parallel outputs. It performs both functions shift left and shift right.

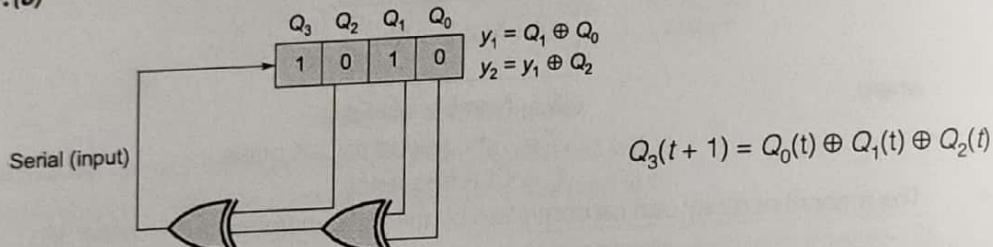
Example - 6.3 The shift register shown in figure is initially loaded with the bit pattern 1010.

Subsequently the shift register is clocked, and with each clock pulse the pattern gets shifted by one bit position to the right. With each shift, the bit at the serial input is pushed to the left most position (MSB). After how many clock pulses will the content of the shift register become 1010 again?



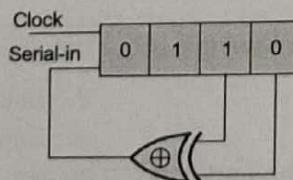
- (a) 3
(c) 11

- (b) 7
(d) 15

Solution : (b)

CLK pulse	Q ₃ Q ₂ Q ₁ Q ₀
1	1 0 1 0
2	1 1 0 1
3	0 1 1 0
4	0 0 1 1
5	0 0 0 1
6	1 0 0 0
7	0 1 0 0
	1 0 1 0

Example - 6.4 The initial contents of the 4-bit serial-in-parallel-out, right shift, shift register shown in the given figure is 0110. After three clock pulses are applied, the contents of the shift register will be



- (a) 0000
(c) 1010

- (b) 0101
(d) 1111

Solution: (c)

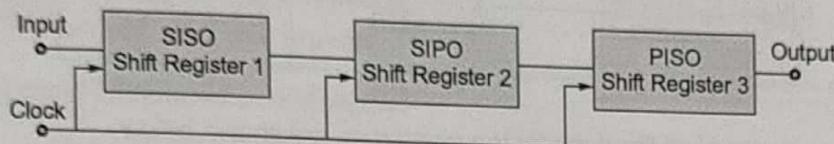
After 1 clock pulse, contents are 1011

After 2 clock pulses, contents are 0101.

After 3 clock pulses, contents are 1010.

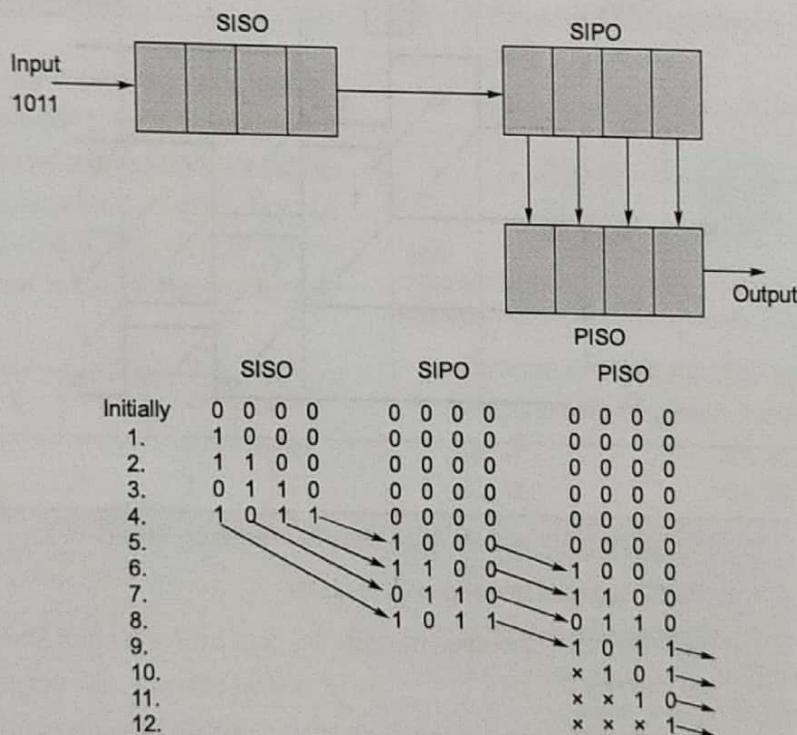
Example - 6.5

Example-6.5 Three 4 bit shift registers are connected in cascade as shown in figure below. Each register is applied with



A 4 bit data 1011 is applied to the shift register 1. The minimum number of clockpulses required, to get same output data as the input data with the same clock pulse, will be

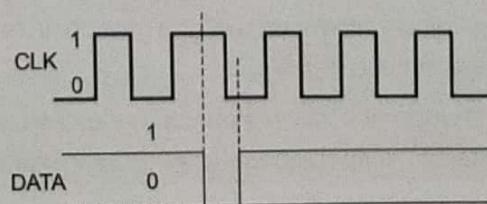
Solution : (b)



Minimum number of clock pulses required = 12.

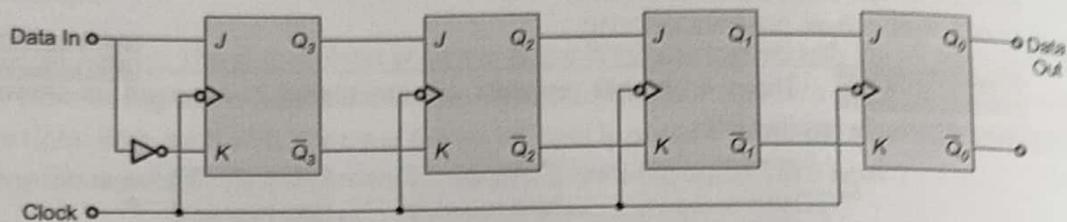
Example - 6.6

Example-6.6 Draw the diagram of a 4-bit shift register using JK flip-flops. Each flip-flop triggers on the negative going transition. Draw the output waveforms for all flip-flops when the input data and clock signals are as shown.

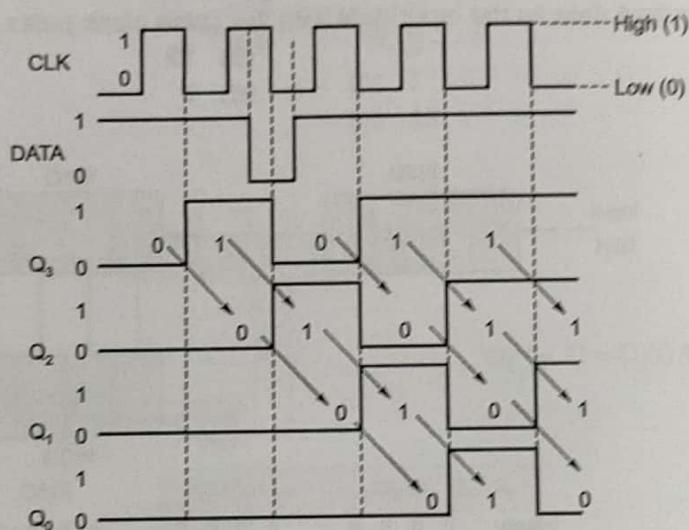


Solution:

The diagram of a 4-bit shift register using JK flip-flops is shown below:



The output waveforms of all flip-flops for the given data are shown below

**Summary**

- "PIPO" register is a storage register made up with D-FFs.
- "PIPO" register is not a shift register.
- Each Shift Left operation multiply the data by 2 and each Shift Right operation divides the data by 2.
- The transfer of new information (data) into a register is referred to as "Loading" the register.
- A group of FFs sensitive to the pulse duration is usually called "gated latch" and a group of FFs sensitive to the pulse transition is called a 'register'.
- In storage registers mostly D flip-flops are used.
- The FFs hold binary information and the gate controls WHEN and HOW, new information is transferred into the register.
- To convert temporal code into spatial code, we use SIPO register, while to convert spatial code into temporal code use PISO register.

Student's
Assignments

1

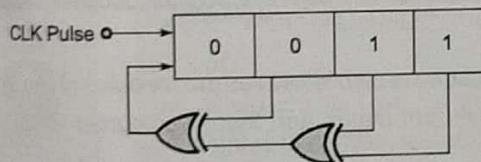
Q.1 What do you understand by the Bi-directional shift register and universal shift register? Give its function table and draw the logical circuit diagram of a 4-bit bi-directional shift register.

Q.2 What is the difference between serial and parallel transfer?
Explain how to convert serial data to parallel data and vice-versa.

Student's
Assignments

2

Q.1 The shift register shown below is initially loaded with binary bit stream 0011. If clock pulses are applied continuously and after some clock pulses the loaded data is repeated in shift register. During this, which pulse is responsible for the highest decimal data to be stored in the shift register?

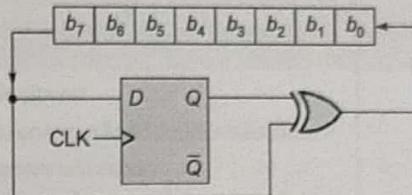


- (a) 4th
(b) 5th
(c) 6th
(d) 2nd

Q.2 Data can be changed from spatial code to temporal code and viceversa by using
(a) ADCs and DACs
(b) Shift registers
(c) Synchronous counter
(d) Timers

Q.3 Shifting a register to the left by one bit position is equivalent to
(a) division by 2
(b) multiplication by 2
(c) addition of 2
(d) subtraction of 2

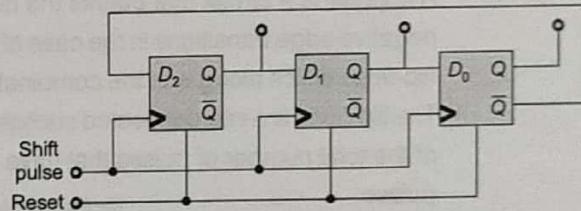
Q.4 An 8-bit left shift register and D-flip-flop shown in figure is synchronized with same clock. The D flip-flop is initially cleared.



The circuit acts as

- (a) Binary to 2's complement converter
(b) Binary to Gray code converter
(c) Binary to 1's complement converter
(d) Binary to Excess- 3 code converter

Q.5 A three-bit shift register is shown in the given figure.



To have the content '000' again, the number of clock pulses required should be

- (a) 3
(b) 6
(c) 8
(d) 16

Q.6 A universal register

- (a) accepts serial input
(b) accepts parallel input
(c) gives serial and parallel outputs
(d) is capable of all of the above

Answer Key:

1. (b) 2. (b) 3. (b) 4. (b)
5. (b) 6. (d)



Introduction

- A counter is a circuit that counts the number of occurrences of an input (in terms of positive or negative edge transitions in the case of binary input). Counter circuits are primarily constituted of flip-flops which along with the combinational elements are used for generation of control signals. The flip-flops are interconnected such that their combined state at any time is the binary equivalent of the total number of pulses that have occurred upto that time. Thus a counter is used to count pulses.
- Each count, a binary number is called a state of a counter. Hence, a counter counting in terms of n -bits (n flip-flops) has 2^n different states.
- The number of different states in counting sequence is also known as the modulus of the counter. Thus, for ' n ' flip-flops counter will have ' 2^n ' different states and then the counter is said to be "MOD- 2^n " counter.

The MOD number represents the frequency division obtained from the last flip-flop. It would be capable of counting upto $(2^n - 1)$ before returning to zero state.

Therefore, If

$$M = \text{Total number of states}$$

and

$$n = \text{Total number of FFs}$$

Then,

$$M \leq 2^n$$

If

$$M = 2^n; \text{ Binary counter}$$

$$M < 2^n; \text{ Non-binary counter}$$

- Depending upon the manner in which the flip-flops are triggered, counters can be divided into two major categories:
 - (i) Asynchronous counter (Ripple/series counter).
 - (ii) Synchronous counter (Parallel counter).

The comparison between synchronous and asynchronous counter is listed in Table (7.1).

Table-7.1

Synchronous Counter	Asynchronous Counter
<ol style="list-style-type: none"> All the flip-flops are triggered simultaneously with the same clock. Operation is faster. Any required sequence can be designed. No decoding error occurs. Designing is complex as the number of states increases. e.g. Ring counter, Johnson counter. 	<ol style="list-style-type: none"> No common clock for all the flip-flops. Operation is slower. Will operate only in a fixed count sequence. Decoding errors will occur. Designing is easy even for more number of states. e.g. Ripple UP counter, Ripple DOWN counter.

- Up/Down Counter: If a counter counts in such a way that the decimal equivalent of the output increases with successive clock pulses, is called 'UP counter' and if it decreases, it is called 'Down counter'. An Up/Down counter can also be designed which can count in any direction depending upon the control input.

Application of Counters

- To count the number of CLK Pulses.
- Works as a "Frequency divider".
- Used in time measurement.
- For distance measurement in RADAR system.
- In Analog to Digital converter (ADC).
- In measurement of PRI (Pulse Repetition Interference).

Remember



- In "MOD-N Counter", if applied input frequency is "f", then the output frequency is f/N .
- If two counters are cascaded with MOD-M followed by MOD-N, then number of overall states of combined counter is $(M \times N)$ and counter is called "MOD-MN" counter.

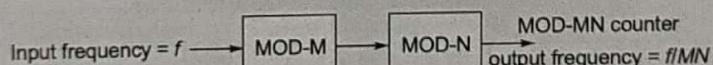


Figure-7.1

Example-7.1 A certain J-K flip-flop has $t_{pd} = 12 \text{ ns}$, the largest 'MOD' counter that can be constructed from cascading these FFs and still operating upto 10 MHz is

Solution:

$$\text{Time period of clock pulse} = \frac{1}{10 \text{ MHz}} = 10^{-7} \text{ sec.}$$

$$\text{Propagation delay of each FF} = 12 \times 10^{-9} \text{ sec.}$$

$$\therefore \text{Number of FF required} = \frac{t_{CLK}}{t_{pd}} = \frac{10^{-7}}{12 \times 10^{-9}} = 8.33$$

Therefore, the number of FF required = 8

and Modulus of counter = $2^8 = 256$

(As MOD $\leq 2^8$)

7.1 Asynchronous/Ripple Counters

- In a ripple counter, there is no clock or source of synchronizing the pulses, however, the state change still occurs due to pulses at clock input of the flip-flops.
- Here, the LSB flip-flop is clocked by external clock pulse and each successive flip-flop is clocked from the previous LSB flip-flop.

7.1.1 3-Bit Ripple Counter

- Figure 7.2 (a) shows a 3-bit binary ripple counter which consists of a series connection of complementing J-K flip-flops, with the output of each flip-flop connected to the clock pulse (CP) input of the next higher order flip-flop.
- The flip-flop holding the *LSB* receives the incoming clock pulses.

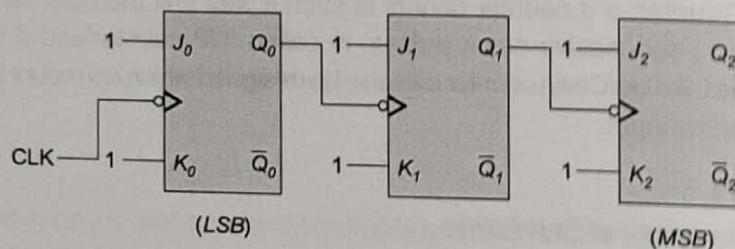


Figure-7.2 (a) Logic diagram of a 3-bit ripple counter

Operation

- The flip-flops change their state on the negative going edge of clock pulse.
- All the flip-flops are connected in toggled mode.
- Q_0 will change its state in every clock pulse.
- Q_1 will change its state when Q_0 changes from 1 to 0.
- Q_2 will change its state when Q_1 changes from 1 to 0.
- Depending upon the operation, the transition table is shown in Table 7.2.

Table-7.2 Transition table/truth table

CLK	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

- Initially all FFs are set to zero.
- The maximum possible states = 8 (from 0 to 7).
- The timing diagram of 3 bit ripple counter is shown in Figure 7.2 (b).

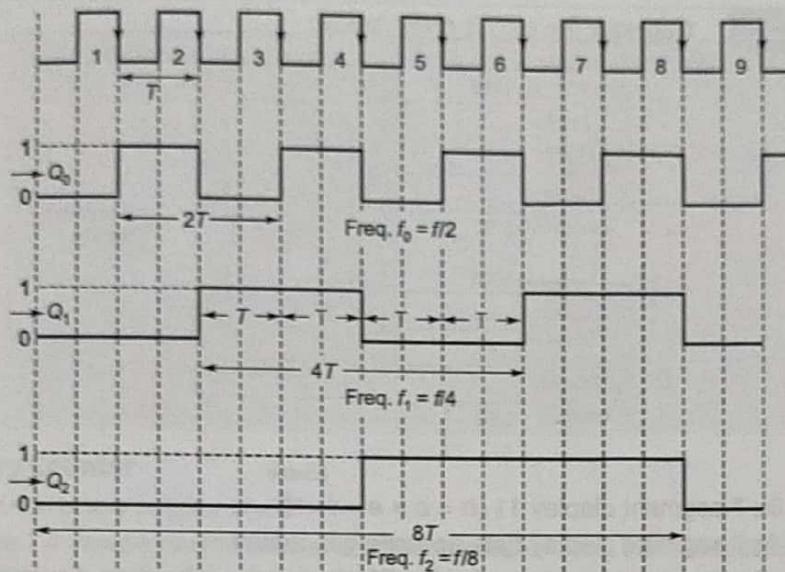


Figure-7.2(b)

- If the input clock frequency is 'f' then the output frequency at $Q_2 = f/8$.
- For an 'N' bit ripple counter, if the propagation delay of each flip-flop is t_{pd} , then the period of clock is given by

$$T_{CLK} \geq N \cdot t_{pd}$$

or

$$f_{CLK} \leq \frac{1}{Nt_{pd}}$$

and

$$f_{max} = \frac{1}{Nt_{pd}}$$

- For determination of Up/Down counter:

Trigger Applied	CLK is given by	Access as
Negative edge	Q	Up counter
Positive edge	Q	Down counter
Negative edge	\bar{Q}	Down counter
Positive edge	\bar{Q}	Up counter

- The major disadvantage of a ripple counter is the error due to the propagation delay of flip-flops i.e. t_{pd} which, is known as decoding error or transient state. To overcome the decoding error in ripple counter we may use "strobe signal".

Thus, with strobe signal

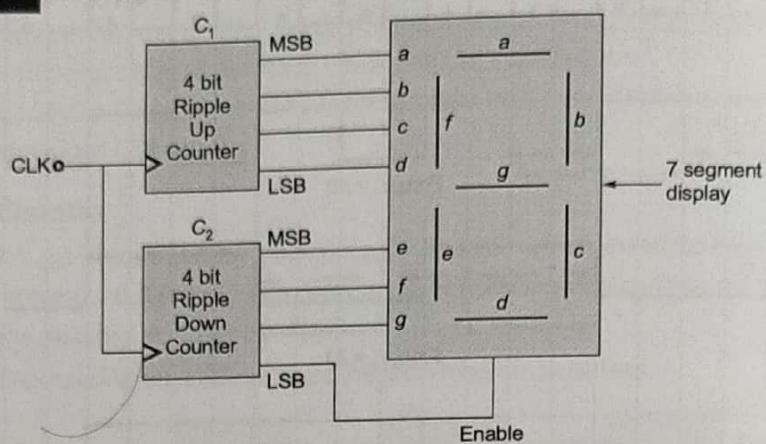
$$T_{CLK} \geq Nt_{pd} + T_s$$

where, T_s = Strobe pulse width

- In the counter discussed above all the states are used and therefore it is called as a binary ripple counter.

Example-7.2

Consider the circuit given below:



If $\text{Enable} = 0$; 7 segment display 11 ($b = c = e = f = 1$)

$\text{Enable} = 1$; 7 segment display data according to Inputs

Initially both the counter were cleared. After 78 clock pulses the data displayed on the 7 segment display is ____.

Solution:

After 78 clock pulse

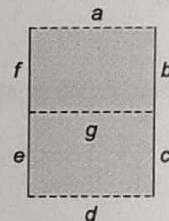
$$\text{Output of counter } C_1 = (1110)_2 = (14)_{10}$$

$$\text{Output of counter } C_2 = (0010)_2 = (2)_{10}$$

7 segment display

a	b	c	d	e	f	g	Enable
1	1	1	0	0	0	1	0

⇒ Data displayed on 7 segment is '11'



7.1.2 Asynchronous Inputs

- In the clocked flip-flops discussed in previous chapters the inputs to S-R, J-K, D and T are called synchronous inputs because their effect on the output of the flip-flop is synchronized with the clock input.
- There are most IC flip-flops available, which have one or more asynchronous inputs. These asynchronous inputs affect the flip-flop output independently from the synchronous inputs and clock input.
- These inputs can be used to SET the flip-flop (1) or RESET the flip-flop (0) at any instant of time regardless of the conditions at other inputs.
- There are mainly two types of asynchronous inputs labelled-PRESET (PRE) and CLEAR (CLR).
- An active HIGH level on PRESET input will set the FF and active HIGH level on CLEAR input will RESET the FF. Similarly an active LOW level on PRESET input will SET the FF and on CLEAR input will RESET the FF as shown in Figure 7.3.

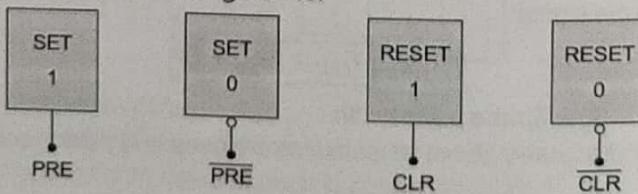
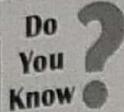
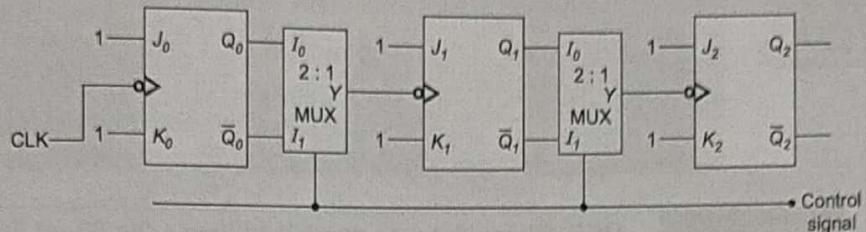


Figure-7.3



A ripple counter can be made to work as an up/down counter using 2 : 1 MUX and a control (signal) as shown in figure.



If, Control = 0 ; clock applied with Q \therefore Up counter

If, Control = 1 ; clock applied with \bar{Q} \therefore Down counter

7.1.3 Non Binary Counter

- For a non binary counter $M < 2^n$.
- Figure 7.4 shows a mod-10 counter or a **decade** counter. Here the total number of flip-flops required is 4 thus the number of used state = 10 and the number of unused state = 6.

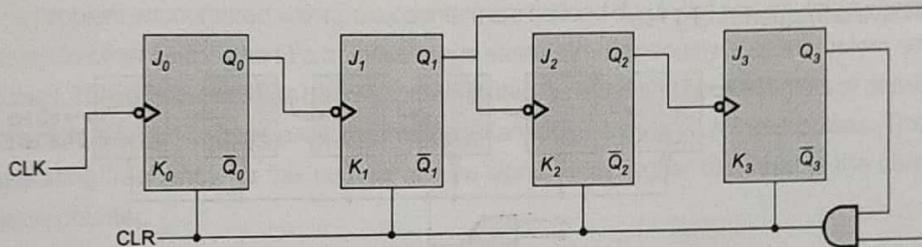


Figure-7.4 Decade counter

- In order to design a non binary decade counter a logic gate is used which detects 10 stage from 0000 to 1001 and as soon as 1010 appears it clears all the flip-flops as shown in Figure 7.4.
- Table 7.3 shows the transition table of a decade counter.

Table-7.3 Truth table

CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0

- It is clear from the truth table that as soon as 1010 appears, inputs to the AND gate becomes 11 which resets the flip-flop with 'CLR' = 1.
- The output frequency of MOD-10 counter = $f/10$.
- If there is no feedback present at Q_3 then the output frequency = $f/16$.
- When decade counter counts from 0 to 9 then it is known BCD counter.

NOTE

For making non binary counter, if 'CLR' is present and CLK is connected with output Q, then we use AND gate, (Figure 7.4). Similarly

$$\text{CLR} \rightarrow Q \rightarrow \text{AND Gate}$$

$$\text{CLR} \rightarrow \bar{Q} \rightarrow \text{NOR Gate}$$

$$\overline{\text{CLR}} \rightarrow Q \Rightarrow \text{NAND Gate}$$

$$\overline{\text{CLR}} \rightarrow \bar{Q} \Rightarrow \text{OR Gate}$$

Example - 7.3

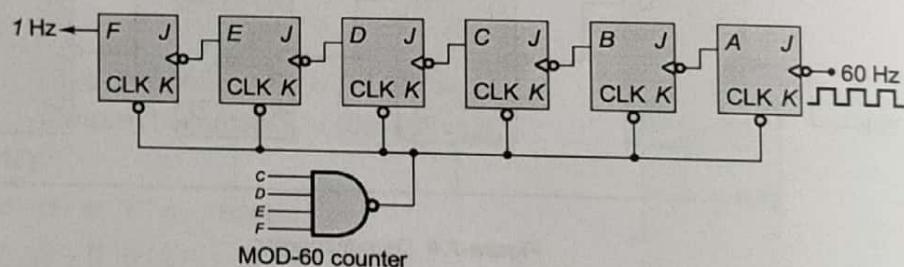
MOD-60 counter was needed to divide the 60-Hz line frequency down to 1 Hz. Construct an appropriate MOD-60 counter.

Solution:

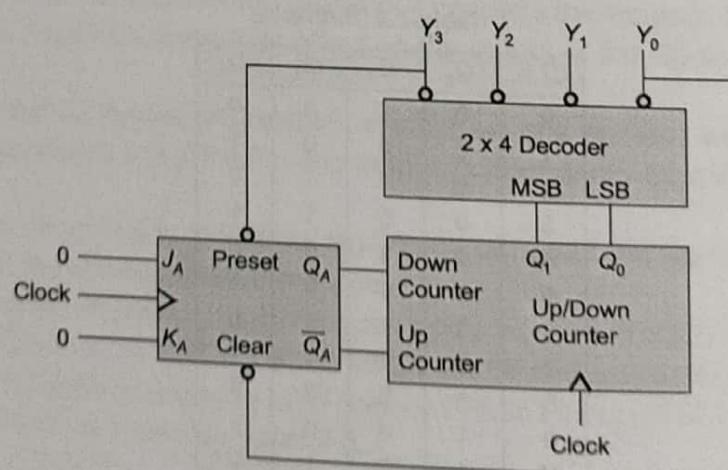
$$\text{MOD} \leq 2^N$$

$$60 \leq 2^N; \quad \text{so, } N = 6$$

Hence, we need six FFs. The counter is to be cleared when it reaches the count of sixty (111100). Thus, the outputs of flip-flops C, D, E and F must be connected to the NAND gate. The output of flip-flop F will have a frequency of 1 Hz.

**Example - 7.4**

Consider the circuit given below:



Assuming the initial value of counter output (Q_1, Q_0) as zero, the counter output for 8 clock pulses in decimal form is

- (a) 0, 3, 2, 1, 0, 1, 2, 3
- (c) 0, 1, 2, 3, 3, 2, 1, 0

- (b) 0, 1, 2, 3, 2, 1, 0, 1
- (d) 0, 3, 2, 1, 0, 0, 1, 2

Solution : (b)

Initially,

$$Q_0 = Q_1 = 0$$

⇒

$$Y_0 = 0$$

⇒ J-K flip-flop is cleared

$$Q_A = 0$$

$$\bar{Q}_A = 1$$

As clock pulse is applied counter starts up counting

As counter reaches $Q_1 = 1$, $Q_0 = 1$ after 3 clock pulses, J-K flip flop is preset.

$$\Rightarrow Q_A = 1$$

$$\bar{Q}_A = 0$$

⇒ Counter starts down counting until Y_0 is low and this repeats.So, Output $Q_1 Q_0$ in decimal form is

0, 1, 2, 3, 2, 1, 0, 1....

7.2 Synchronous Counters

- The problem encountered with ripple counters are caused due to the accumulated FFs propagation delay. In other words, the FFs do not change states simultaneously in synchronism with the input pulses. These limitations can be overcome with the use of synchronous counters or parallel counters in which, all the flip-flops are triggered simultaneously by the CLK input pulses. The maximum operating frequency for this counter will be significantly higher than that of the corresponding ripple counter.
- The synchronous counters are classified as:
 - (i) Shift register counters
 - 1. Ring counter
 - 2. Twisted ring counter/Johnson counter
 - (ii) Series carry counter
 - (iii) Parallel carry counter

7.2.1 Shift Register Counters

- One of the applications of shift register is that they can be arranged to form different types of counters. Shift register counters are obtained from serial-in serial-out (SISO) shift register by providing feedback from the output of the LSB FF to the input of the MSB FF.
- The most widely used shift register counters are:
 - 1. Ring counter.
 - 2. Twisted ring counter.

Ring Counters

- It is the simplest shift register counter.
- It is also called "end carry counter".
- The Figure 7.5 (a) shows the logic diagram of a four bit ring counter using DFF. Its state diagram and transition table is shown in Figure 7.5 (b) and (c) respectively.

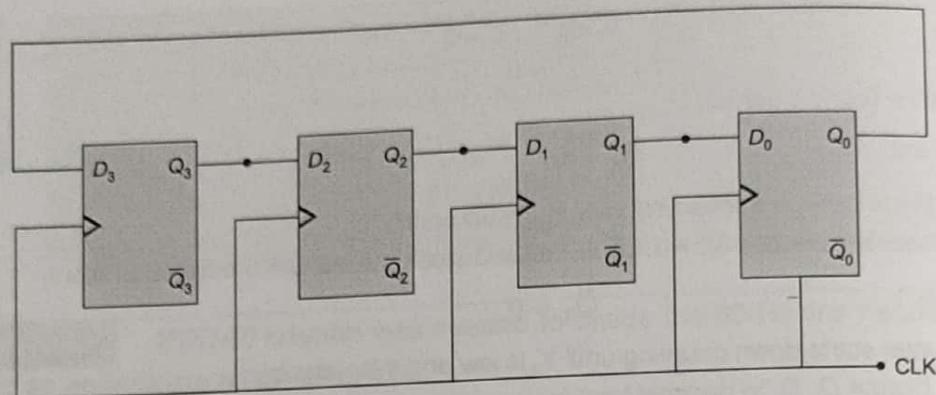
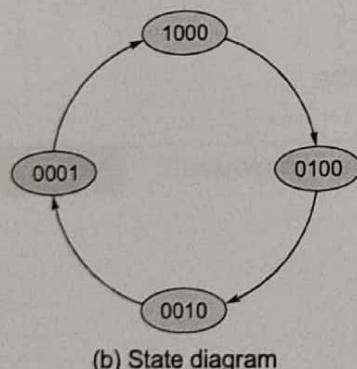


Figure-7.5 (a) Logic diagram of a ring counter

- It is clear from the logic diagram that the Q output of each state is connected to D input of the next state. However the output Q_0 of LSB FF is connected back to the input D_3 of MSB FF such that an array of FFs is arranged in a ring and, hence, the name 'ring counter'.
- In this only one bit is high and is made to circulate around the register as long as clock pulses are applied.



CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	1
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1
9	1	0	0	0

(c) Truth table

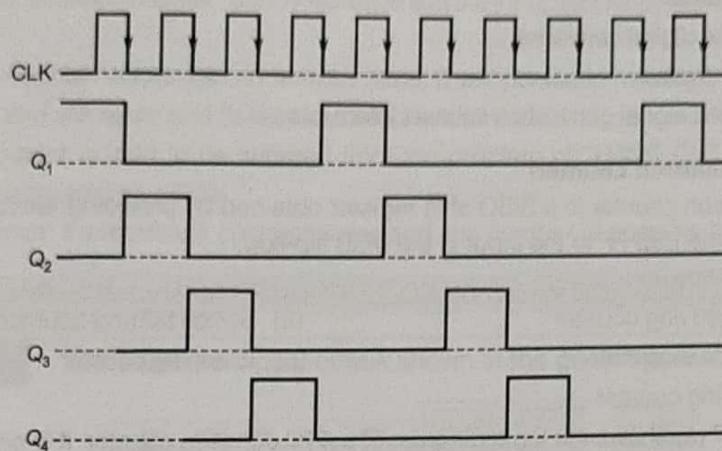
Figure-7.5 (b) and (c)

- As the sequence repeats itself after four clock pulses, thus, the number of distinct states in the ring counter is equal to the number of FFs used in the counter.
- Therefore, with n flip-flops, there are n -states present in ring counter.

Hence,

$$\text{MOD-M} = n$$

- With ' n ' flip-flops, maximum count possible in ring counter is (2^{n-1}) .
- The timing diagram is shown in Figure 7.5 (d).

**Figure-7.5(d)** Timing diagram of a 4-bit ring counter

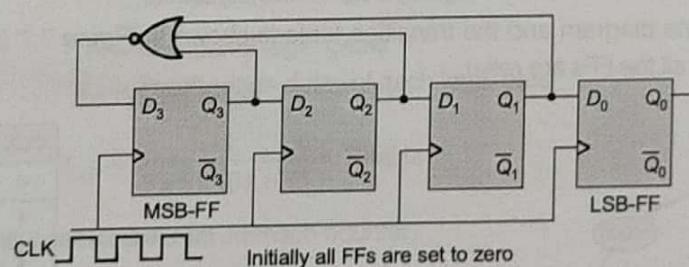
- Thus, the number of unused states in ring counter is $(2^n - n)$.

Advantages

Decoding is very easy in ring counter, because there is no aid of extra circuits.

Disadvantages

Outputs are not symmetric.

Self-Starting Ring Counter**Figure-7.6 (a)** Selfstarting ring counter

Clock	Q_3	Q_2	Q_1	Q_0	
0	0	0	0	0	
1	1	0	0	0	
2	0	1	0	0	
3	0	0	1	0	4-states
4	0	0	0	1	
5	1	0	0	0	
6	0	1	0	0	
7	0	0	1	0	4-states
8	0	0	0	1	
9	1	0	0	0	

Figure-7.6 (b)

- In 4-bit ring counter the used states = 4 and the unused input = $2^4 - 4 = 12$.
In any counter if CLK frequency is "f" the FFs output frequency is " f/N " (where N = No. of states).

Applications of Ring Counter

- In analog to digital converter
- In stepper motors
- In controlled signal generation such as interrupts

Twisted Ring Counter (Johnson Counter)

- The Johnson counter is a SISO shift register obtained by providing feedback from the inverted output of the LSB FF to the input of the MSB flip-flop.
- It is also known as
 - (i) Twisted ring counter
 - (ii) Switch tail ring counter
 - (iii) Moebius counter
 - (iv) Creeping counter
 - (v) Walking counter
- Figure 7.7 (a) shows the logic diagram of a 4-bit Johnson counter. It is clear that the Q output of each stage is connected to the D input of the next stage, however the output \bar{Q}_0 is connected to the D input of the MSB flip-flop, therefore, the name 'twisted ring counter'.

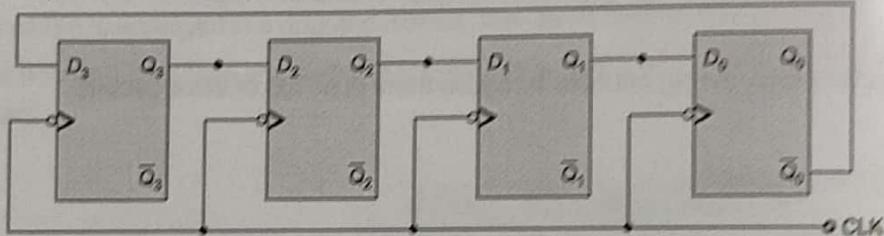
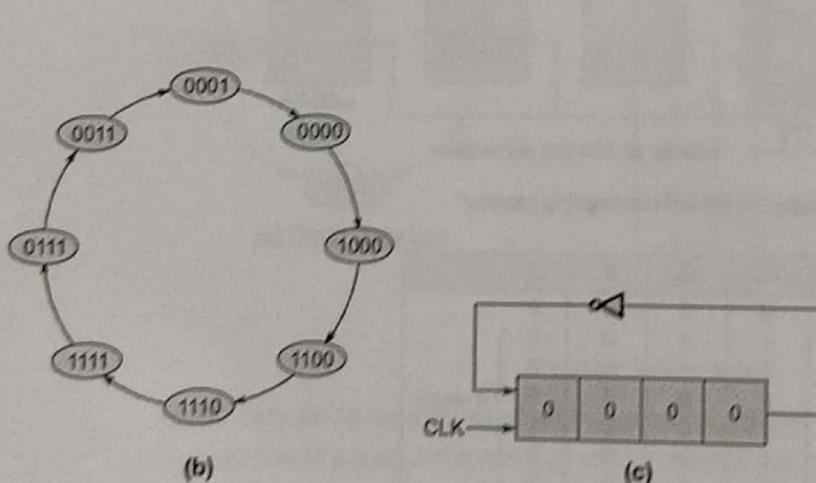


Figure-7.7 (a) Twisted ring counter

- The state diagram and the transition table is shown in Figure 7.7 (b) and (d) respectively. Let initially all the FFs are reset.



CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0
9	1	0	0	0

Figure-7.7 (b), (c) and (d)

- It is clear from the truth table, that the sequence is repeated after every eighth clock pulse. Thus for ' n ' FFs there are 2^n possible states.

$$\text{MOD-M} = 2^n$$

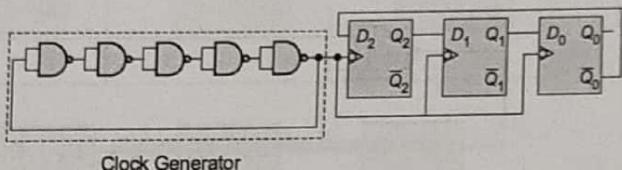
- The total number of unused state = $(2^n - 2n)$.

- In normal "Johnson counter" with 'n' flip-flops and the input frequency is 'f' then the output frequency is $f/2^n$.
- When a counter enters into an unused state, it will persist in moving from one unused state to another and will never find its way to a used state.
- Thus, counter is said to be suffered from the problem of "LOCK OUT". This problem can be overcome by adding a Gate.
- In a 'counter' if a feedback connection is used, the number of possible states will decrease.

NOTE: In Johnson counter to decode each state two input AND gate or NOR gate is used.

Example-7.5

Consider the digital circuit shown in the given figure



The average propagation delay of each NAND gate in the clock generator circuit is 10 ns. Calculate the frequency of the signal at Q_0 .

Solution:

Here the clock generator is a ring oscillator circuit

$$\therefore f_{CLK} = \frac{1}{2Nt_{pd}}$$

N = Number of logic gates

t_{pd} = Propagation delay of each logic gate

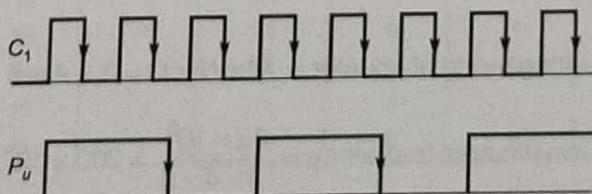
$$f_{CLK} = \frac{1}{2 \times 5 \times 10 \times 10^{-9}} = 10 \text{ MHz}$$

The clocked sequential circuit is a 3-bit Johnson counter

$$\therefore f_{Q_0} = \frac{f_{CLK}}{2 \times 3} = \frac{10 \text{ MHz}}{6} = 1.66 \text{ MHz}$$

Example-7.6

The square wave C_1 shown in figure is given to the clock input of a 4-bit binary Up/Down counter whose Up/Down input is fed with the pulse train P_u . The counter is a negative edge triggered one. The counter starts with 0000 and will reach 0000 again at the



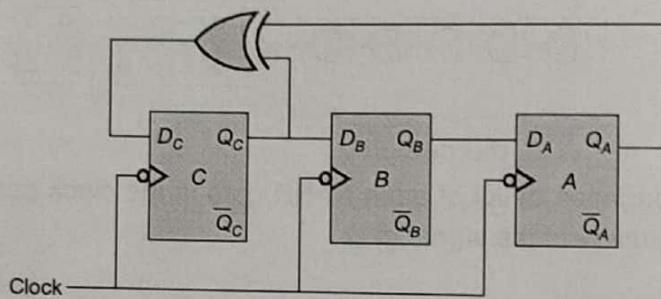
(a) 15th clock pulse
(c) 44th clock pulse

(b) 16th clock pulse
(d) 48th clock pulse

Solution : (c)

Here in every P_u pulse there are 3 clock pulses. For initial two clock pulses counter will work as up counter as P_u is high and for 3rd clock pulse it will work as down counter so in every three clock pulse it will be incremented by only 1 (2 increment and one decrement), so after $14 \times 3 = 42$ clock pulses counter goes to 14. After that P_u goes high for two clock pulses and counter will be incremented by 2 is $14 + 2 = 16$ i.e. equivalent to 0000. So after $42 + 2 = 44$ clock pulses counter will be go to 0000 again.

Example-7.7 A digital circuit is designed with three D-flip flops and an EX-OR gate as shown in below figure. If the initial value of $Q_A Q_B Q_C$ was 110 then the minimum number of clock pulses required to get $Q_A Q_B Q_C$ as 011 is ____.

**Solution :**

	Q_A	Q_B	Q_C	$Q_A \oplus Q_C$
1	1	1	0	1
2	1	0	1	0
3	0	1	0	0
4	1	0	0	1
5	0	0	1	1
6	0	1	1	1

After 5 clock pulse $Q_A Q_B Q_C$ will be 011.

Example-7.8 12 MHz clock frequency is applied to a cascaded counter of modulus-3 counter, modulus-4 counter and modulus-5 counter. What are the lowest output frequency and the overall modulus, respectively?

- (a) 200 kHz, 60
- (b) 1 MHz, 60
- (c) 3 MHz, 12
- (d) 4 MHz, 12

Solution : (a)

In cascade the resulting mode of counter = $M \times N \times O = 3 \times 4 \times 5$

$$\text{Lowest output frequency} = \frac{12 \times 10^6}{3 \times 4 \times 5} = 200 \times 10^3$$

$$= 200 \text{ kHz}$$

$$\text{Overall modulus} = 3 \times 4 \times 5 = 60$$

7.2.2 Synchronous Series Carry Counter

- A 4-bit (MOD-16) series carry counter is shown in Figure 7.8 (a). The basic principle of operation of this synchronous counter is inputs of the flip-flops are connected in such a manner that only those FFs that are supposed to toggle on a given CLK will have $T = 1$.

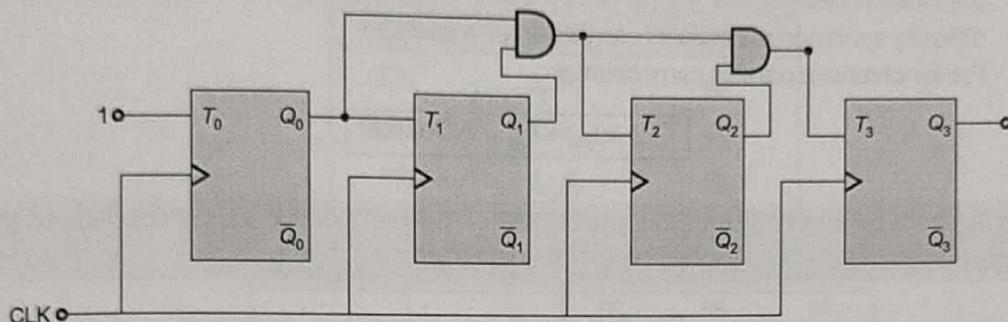


Figure-7.8 (a) Series carry counter

- The circuit shown above is a synchronous counter. In this counter
 - Q_0 toggles for every CLK pulse is applied.
 - Q_1 toggles when $Q_0 = 1$ and CLK pulse is applied.
 - Q_2 toggles when $Q_1 = 1$ and $Q_0 = 1$ and CLK pulse is applied.
 - Q_3 toggles when $Q_2 = Q_1 = Q_0 = 1$ and CLK pulse is applied.
- The truth table, depending upon the operation is shown in Table 7.4.

Table-7.4 Truth table for series carry count

CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0

- In order to get synchronous down counter, \bar{Q} output of each stage is connected to the input of the next stage.
- Most important advantage of this counter is that it reduces the 'decoding error'.

- The total delay for synchronous series counter is

$$T_{CLK} \geq t_{pd} + (n - 2)t_{pd} \text{ (AND-Gate)}$$

As the number of bit increases, the propagation delay of FFs and the propagation delay of AND gate also increases, this will increase the total delay of the circuit. Thus, in order to overcome this difficulty synchronous parallel carry counter is used.

For synchronous parallel carry counter,

$$T_{CLK} \geq t_{pd} + t_{pd} \text{ (AND-Gate)}$$

Remember: Speed of synchronous parallel carry counter > speed of synchronous series carry counter > speed of ripple carry counter.

Example-7.9

(i) Determine f_{max} if t_{pd} for each FF is 50ns and t_{pd} for each AND gate is 20 ns. Compare this with f_{max} for a MOD-16 ripple counter.

(ii) What has to be done to convert this counter to MOD-37?

(iii) Determine f_{max} for the MOD-32 parallel counter.

Solution:

(i) The total delay that must be allowed between input clock pulses is equal to FF t_{pd} + AND gate t_{pd} . Thus, $T_{clock} \geq 50 + 20 = 70$ ns, and so the parallel counter has

$$f_{max} = \frac{1}{70\text{ns}} = 14.3 \text{ MHz (parallel counter)}$$

A MOD-16 ripple counter uses four FFs $t_{pd} = 50$ ns. Thus, f_{max} for the ripple counter is

$$f_{max} = \frac{1}{4 \times 50\text{ns}} = 5 \text{ MHz (ripple counter)}$$

- (ii) A fifth FF must be added, since $2^5 = 32$. The CLK input of this FF is also tied to the input pulses. Its J and K inputs are fed by the output of a four-input AND gate whose inputs are A, B, C and D.
- (iii) f_{max} is still determined as in (a) regardless of the number of FFs in the parallel counter. Thus, f_{max} is still 14.3 MHz.

7.3 Synchronous Counter Design

Synchronous counters for any given count sequence and modulus can be designed in the following way.

- Identify the number of flip-flops, number of inputs and number of outputs required.
- Construct the state table with the help of present state, next state and excitation table of the FFs.
- Prepare K-map for each flip-flop input in terms of flip-flop outputs as the input variable, obtain and minimize the logical expressions.
- Connect the circuit using flip-flops and other gates corresponding to the minimized expressions.

Example-7.10

Design a 3-bit synchronous counter using J-K Flip-flops.

Solution:

The number of Flip-flops required is 3. Let the flip-flops be FF_0 , FF_1 and FF_2 and their inputs and outputs are given below:

Flip-flop	Input	Output
FF_0	J_0, K_0	Q_0
FF_1	J_1, K_1	Q_1
FF_2	J_2, K_2	Q_2

Present state	Next state	Flip-flop inputs								
		FF_0	FF_1	FF_2	J_0	K_0	J_1	K_1	J_2	K_2
0 0 0	0 0 1	0	0	1	1	x	0	x	0	x
0 0 0	0 1 0	0	1	0	x	1	1	x	0	x
0 1 0	0 1 1	0	1	1	1	x	x	0	0	x
0 1 1	1 0 0	1	0	0	x	1	x	1	1	x
1 0 0	1 0 1	1	0	1	1	x	0	x	x	0
1 0 1	1 1 0	1	1	0	x	1	1	x	x	0
1 1 0	1 1 1	1	1	1	1	x	x	0	x	0
1 1 1	0 0 0	0	0	0	x	1	x	1	x	1
0 0 0	0 0 0									

The count sequence and the required inputs of flip-flops are given in table. The inputs to the flip-flops are determined by using K-MAP.

$Q_2 Q_1$	00	01	11	10
Q_0	0	1	1	1
	1	x	x	x

$J_0 = 1$

$Q_2 Q_1$	00	01	11	10
Q_0	0	x	x	x
	1	1	1	1

$K_0 = 1$

$Q_2 Q_1$	00	01	11	10
Q_0	0	0	x	0
	1	1	x	1

$J_1 = Q_0$

$Q_2 Q_1$	00	01	11	10
Q_0	0	x	0	x
	1	x	1	x

$K_1 = Q_0$

$Q_2 Q_1$	00	01	11	10
Q_0	0	0	0	0
	1	0	1	x

$J_2 = Q_0 Q_1$

$Q_2 Q_1$	00	01	11	10
Q_0	0	x	x	0
	1	x	x	0

$K_2 = Q_0 Q_1$

Consider one column of the counter state at a time and start from the first row.

- ⇒ We consider Q_0 . Before the first pulse is applied, $Q_0 = 0$ and it is required to be 1 at the end of the first clock pulse.
- ⇒ Therefore, to achieve this condition, the values of J_0 and K_0 are 1 and x respectively (from the excitation table). These are entered in the table in the row corresponding to 0 pulse.
- ⇒ When the second clock pulse is applied Q_0 is to change from 1 to 0, therefore, the required inputs are

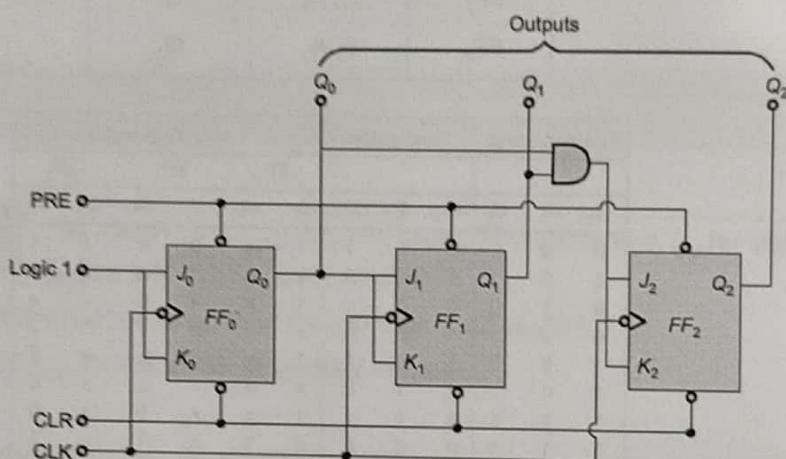
$$J_0 = x, K_0 = 1$$

⇒ In a similar manner inputs of each flip-flop are determined.

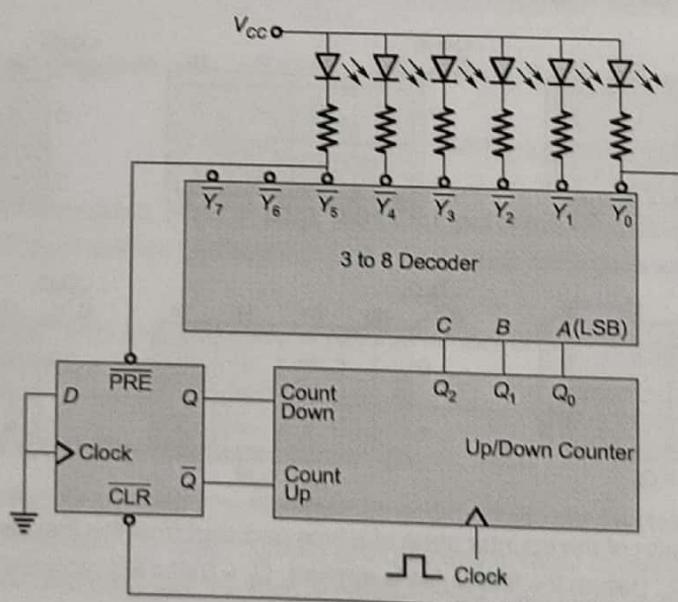
⇒ Now, we prepare the K-maps with Q_2 , Q_1 and Q_0 as input variables and flip-flop inputs are output variables as:

$$\begin{aligned} J_0 &= 1, & K_0 &= 1 \\ J_1 &= Q_0, & K_1 &= Q_0 \\ J_2 &= Q_0 Q_1, & K_2 &= Q_0 Q_1 \end{aligned}$$

Now the final circuit of 3-bit synchronous counter is,



Example-7.11 The circuit below shows an up/down counter working with a decoder and a flip-flop. Preset (PRE) and Clear (CLR) of the flip-flop are asynchronous active-low inputs.



Assuming that the initial value of counter output ($Q_2 Q_1 Q_0$) as zero, the counter outputs in decimal for 12 clock cycles are ($1 \rightarrow$ Up, $0 \rightarrow$ Down)

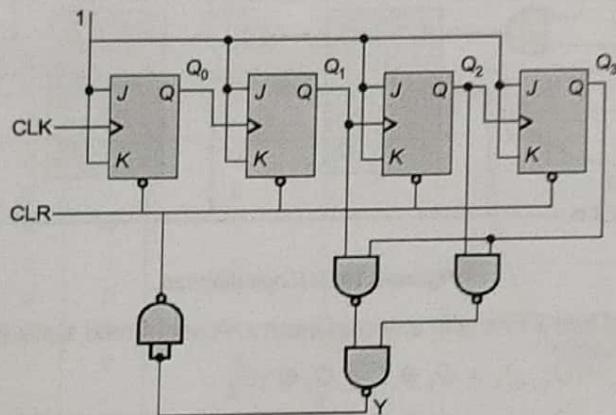
- | | |
|---|---|
| (a) 0, 1, 2, 3, 4, 4, 3, 2, 1, 1, 2, 3, 4 | (b) 0, 1, 2, 3, 4, 5, 0, 1, 2, 3, 4, 5, 0 |
| (c) 0, 1, 2, 3, 4, 5, 5, 4, 3, 2, 1, 0, 1 | (d) 0, 1, 2, 3, 4, 5, 4, 3, 2, 1, 0, 1, 2 |

Solution : (d)

Initially $Q = 0$ and count up ($\bar{Q} = 1$) is active so it started counting up and when it reaches to 5 then decoder output at pin 5 becomes 0 and preset will be active and it will set Q and it will make the counter mode down and count becomes 4, then 3 then 2 then 1 then 0, as soon as it reaches 0, decoder output at pin 0 is low and clear is active and Q goes to 0 and $\bar{Q} = 1$ so up is active and it counts 1, 2,..... so sequence is 0, 1, 2, 3, 4, 5, 4, 3, 2, 1, 0, 1, 2.

Example - 7.12

Consider the counter circuit shown below:

(i) In the above figure, Y can be expressed as

- | | |
|---------------------------------|--------------------------------|
| (a) $Q_3(Q_2 + Q_1)$ | (b) $Q_3 + Q_2 Q_1$ |
| (c) $\overline{Q_3(Q_2 + Q_1)}$ | (d) $\overline{Q_3 + Q_2 Q_1}$ |

(ii) The above circuit is a

- | | |
|--------------------|--------------------|
| (a) Mod-8 counter | (b) Mod-9 counter |
| (c) Mod-10 counter | (d) Mod-11 counter |

Solution:

(i) (a)

$$Y = \overline{(Q_1 Q_3)} \cdot \overline{(Q_2 Q_3)} = \overline{(Q_1 Q_3)} + \overline{(Q_2 Q_3)} = Q_1 Q_3 + Q_2 Q_3 = Q_3(Q_1 + Q_2)$$

(ii) (c)

To reset the counter output Y must be one.

$$Y = Q_3(Q_1 + Q_2)$$

Q_3	Q_2	Q_1	Q_0
1	0	1	0
1	1	0	0
1	1	1	0

When the counter output $Q_3 Q_2 Q_1 Q_0 = 1010$ then the output Y will give output 1 and it will be given to inverter, so it will reset the counter and again counting will start and it will not go for further combinations. So, MOD of the counter is '10'.

7.4 State Diagram and State Table

- A state diagram is a pictorial representation of the relationship between the present state, the input, the next state, and the output of a sequential circuit. However, for the implementation of sequential circuit, the information contained in the state diagram is to be translated into state table. Thus, the state table is a tabular representation of the state diagram.
- Consider the sequential circuit shown in *Figure 7.10 (a)*.

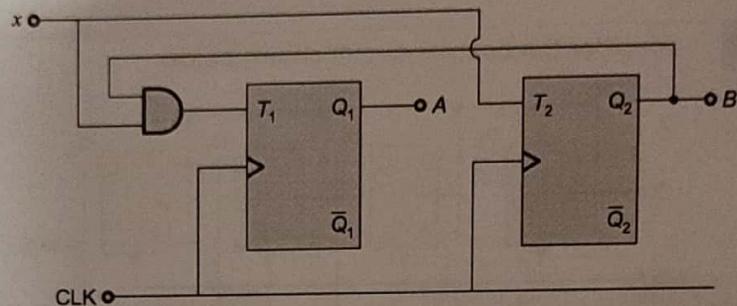


Figure-7.10 (a) Logic diagram

The circuit consists of two T FFs with external input x . A set of next state equation can be written as

$$(Q_{n+1})_1 = Q_1 \oplus T_1 = Q_1 \oplus xQ_2$$

$$(Q_{n+1})_2 = Q_2 \oplus T_2 = Q_2 \oplus x$$

- The corresponding state table is shown *Table 7.5*.

Table-7.5 State table

Present State		Input	Next state		Required excitation	
Q_1	Q_2	x	$(Q_{n+1})_1$	$(Q_{n+1})_2$	T_1	T_2
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	0
0	1	1	1	0	1	1
1	0	0	1	0	0	0
1	0	1	1	1	0	1
1	1	0	1	1	0	0
1	1	1	0	0	1	1

- From the state table, the state diagram may be drawn as shown in *Figure 7.10 (b)*.

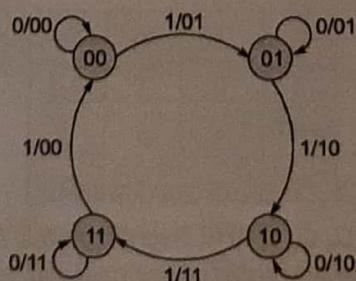


Figure-7.10 (b) State diagram

- Here, each encircled binary number represents a particular state of the flip-flop outputs. Each directed line which indicates the transition between the two state is labelled with two binary numbers separated by a 'slash'. The first number represent the input during present state and the second number represents the next state.

7.4.1 State Diagram of Various Flip-Flops

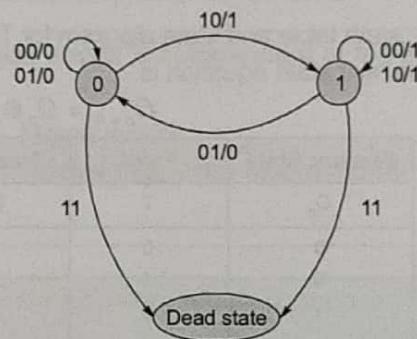
S-R Flip-Flop

The state table and state diagram of the SR flip-flop are shown in Figure 7.11 (a), and (b) respectively. For SR flip-flop the next state equation is

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

Present State	Input		Next state
Q_n	S	R	$(Q_{n+1})_1$
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

(a) State Table



(b) State Diagram

Figure-7.11 (a) and (b)

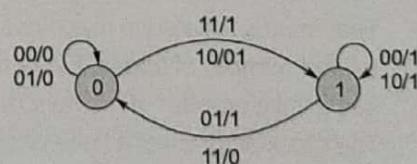
J-K Flip-Flop

The state table and state diagram for JK flip-flop shown in Figure 7.12 (a) and (b) respectively. For JK flip-flop the next state equation is

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n$$

Present State	Input		Next state
Q_n	J	K	$(Q_{n+1})_1$
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

(a) State Table



(b) State Diagram

Figure-7.12 (a) and (b)

D Flip-Flop

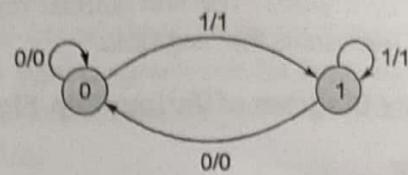
The state table and state diagram for D flip-flop are shown in Figure 7.13 (a) and (b) respectively.

For D flip-flop the next state equation is

$$Q_{n+1} = D$$

Present State	Input	Next state
Q_n	D	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

(a) State Table



(b) State Diagram

Figure-7.13 (a) and (b)

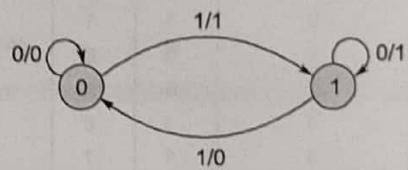
T Flip-Flop

The state table and state diagram for T flip-flop are shown in Figure 7.14 (a) and (b) respectively. For T flip-flop the next state equation is

$$Q_{n+1} = Q_n \oplus T$$

Present State	Input	Next state
Q_n	T	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

(a) State Table



(b) State Diagram

Figure-7.14 (a) and (b)

7.5 Finite State Model/Machine

- The finite state machine is an abstract model that describes the synchronous sequential machine.
- In a sequential circuit, the output depends upon the present input as well as past history and therefore it would need an infinite capacity for storing them. Since it is impossible to implement machines with infinite storage capacity, finite state machines are used. Thus finite state machines are sequential circuits whose past history can affect their future behaviour in only a finite number of ways.
- That means finite state machines are machines with a fixed number of states. Every finite state machine has a finite number of memory devices.
- The block diagram of a finite state model is shown in Figure 7.15. Here $x_1, x_2 \dots x_n$ are the inputs $y_1, y_2 \dots y_m$ are the outputs, $z_1, z_2 \dots z_k$ are state variables and $Z_1, Z_2 \dots Z_K$ represents the next state.
- With an n -state machine, we can generate a periodic sequence of less than or equal to n -states.
- Finite machines are of two types. The major difference between them is the way the output is generated. They are
 - Moore type model.
 - Mealy type model.

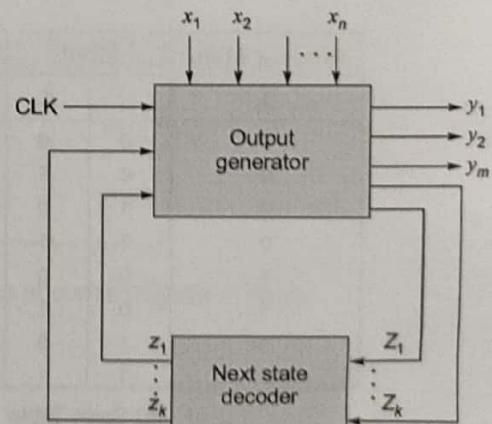


Figure-7.15 Block diagram of a state model

- Table (7.6) shows a comparison between the Moore machine and the mealy machine.

Table-7.6 Comparison between Moore and Mealy model

Moore Machine	Mealy Machine
1. The output is a function of present state only.	1. The output is a function of present state as well as present input.
2. The change in input does not affect the output of the circuit.	2. The change in input may affect the output of the circuit.
3. For implementing same function, it requires more number of states.	3. It requires less number of states for implementing same function.
4. The designing of Moore model is easy.	4. The designing of Mealy model is complex.

Example-7.13 The output of a clocked sequential circuit is independent of the input. The circuit can be represented by

Solution : (b)

Summary



- The ring counter and Johnson counter are not used for counting purpose.
 - In ring counter the phase shift between the generated waveform is $\frac{360^\circ}{n}$, where 'n' is the number of bits.
 - "Lock out" problem occurs in non-binary counter.
 - J, K, S, R, D and T are called synchronous input.
 - Reset and Set are called asynchronous input.
 - "Preset" always make the output to '1'.
 - "Clear" always make the output to '0'.
 - "Glitch" is an unwanted spike in the signal.
 - For a MOD-10 Counter (DECADE), Johnson counter uses 5 FFs, ring counter uses 10 FFs and ripple counter uses 4 FFs.
 - Synchronous counter is used for "High Frequency Application" and Asynchronous counter is used for "Low Frequency Application".
 - MOD-16 counter could also be called a "divide-by-16 counter".
 - Johnson counter represents a middle ground between ring counter and binary counter.
 - A "ring counter" is a circular shift register with only one FF being set at any particular time; all others are cleared.
 - Minimum number of J-K-FFs needed to construct a BCD counter is 4.
 - The overall MOD of a counter with MOD M is connected in cascade with another counter of MOD N is $(M \times N)$.
 - In "MOORE state machines" present output depends only on present state.
 - In "MEALY state machines" present output depends on present state and present external inputs.



Student's Assignments

1

- Q.1** What is the delay time of 14 stage ripple counter and that of N stage synchronous counter? Draw the logic diagram of a counter which can count both in up and down directions.

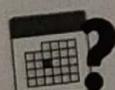
Q.2 Draw the state diagram, logic diagram and timing diagram for a decimal BCD ripple counter. Explain the operation mentioning the conditions for state transition of each flip-flop.

Q.3 A flip-flop has a 5 ns delay from the time the clock edge occurs to the time the output is complemented. What will be the maximum delay in 10 bit binary ripple counter that uses this FF. What is the maximum frequency at which the counter can operate reliably?

Q.4 How many flip-flop will be complemented in a 10 bit binary ripple counter to reach the next count after 1111000111?

Answer : (Conventional)

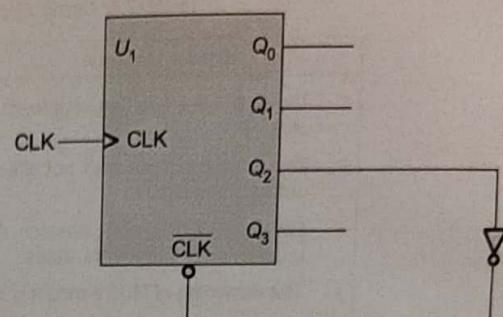
3. (50 ns; 20 MHz)
4. 4



Student's Assignments

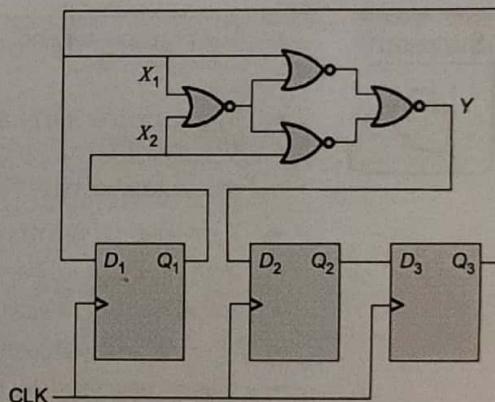
2

Circuit shown represents a



- (a) MOD-2 Counter
 - (b) MOD-3 Counter
 - (c) MOD-4 Counter
 - (d) MOD-5 Counter

- Q.3** For the circuit shown below:



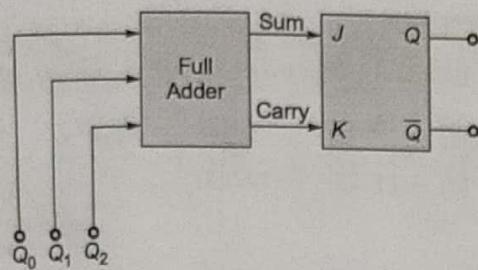
- (i) The correct input-output relationship between Y and (X_1, X_2) is

 - $Y = X_1 + X_2$
 - $Y = X_1 X_2$
 - $Y = X_1 \oplus X_2$
 - $Y = \overline{X_1 \oplus X_2}$

(ii) The D flip-flops are initialized to $Q_1 Q_2 Q_3 = 000$. After 1 clock cycle, $Q_1 Q_2 Q_3$ is equal to

 - 011
 - 010
 - 100
 - 101

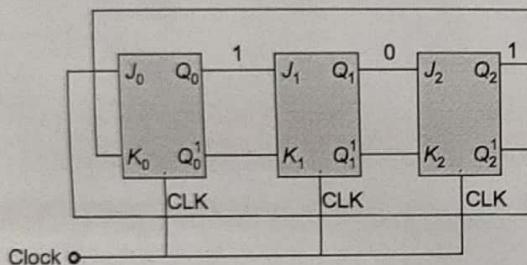
- Q.4** Just before the flip-flops of the counter are cleared (that is, when the counter is in the last used state), the flip flop outputs are made to pass through a full adder. The output of the full adder is then passed through a JK flip flop as shown below.



The JK flip flop enters into the

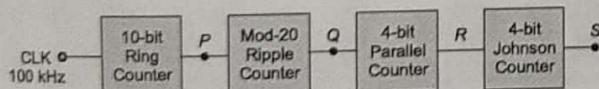
- (a) Hold state
- (b) Set state
- (c) Reset state
- (d) Toggle state

- Q.5 The three state Johnson-ring counter as shown above is clocked at constant frequency of f_c from the starting state of $Q_0 Q_1 Q_2 = 101$. The frequency of outputs $Q_0 Q_1 Q_2$ will be



- (a) $\frac{f_c}{8}$
- (b) $\frac{f_c}{6}$
- (c) $\frac{f_c}{3}$
- (d) $\frac{f_c}{2}$

- Q.6 Consider a cascaded network having 4-sequential circuit block with input clock frequency is 100 kHz. The frequencies of the pulses at the points P, Q, R and S respectively are



- (a) 10 kHz, 0.5 kHz, 31.25 kHz, 390 Hz
- (b) 10 kHz, 5 kHz, 1.25 kHz, 390 Hz
- (c) 10 kHz, 0.5 kHz, 31.25 Hz, 7.8 Hz
- (d) 10 kHz, 0.5 kHz, 31.25 kHz, 3.9 Hz

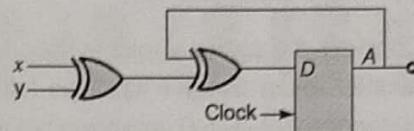
- Q.7 Which of the following measurements can be done using a counter?

1. Pulse duration
2. Interval between two pulses
3. Amplitude of the pulse
4. Rise time of a pulse

Select the correct answer from the codes given below:

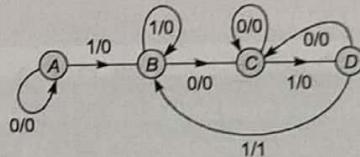
- (a) 1 and 2
- (b) 2 and 3
- (c) 1 and 4
- (d) 2 and 4

- Q.8 A finite state machine is given below. Here x, y are inputs and 'A' is the output. The FSM is



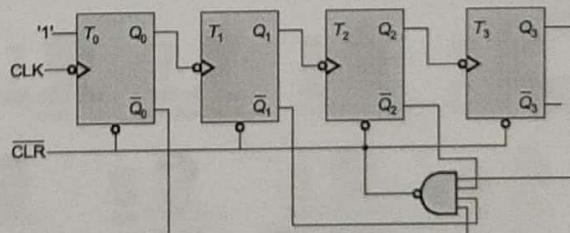
- (a) a Moore machine with $A(t+1) = D$
- (b) a Mealey machine with $A(t+1) = A \oplus x \oplus y$
- (c) neither moore nor mealey machine
- (d) a logarithmic state machine

- Q.9 Which of the following is correct regarding the above transition diagram?



- (a) detects the sequence 1010 and is overlapping detector.
- (b) detects the sequence 1011 and is non-overlapping detector.
- (c) detects the sequence 1011 and is overlapping detector.
- (d) detects the sequence 1010 and is non-overlapping.

- Q.10 If the clock frequency is 70 MHz, then the output frequency at Q_3 of the above counter is



- (a) 7.77 MHz
- (b) 11.66 MHz
- (c) 10 MHz
- (d) 8.75 MHz

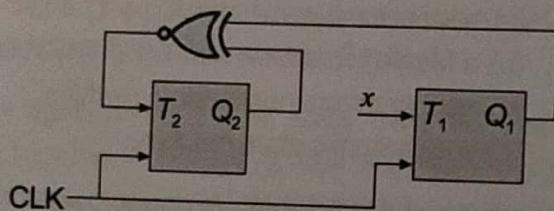
- Q.11 In a non-binary ripple counter, by using negative edge-triggered Toggled FFs, the input clock pulse is applied to
- T*-inputs of all FFs
 - T*-inputs of one FF
 - Clock input of one FF
 - Clock input of all FFs

Answer Key:

- | | | | |
|---------|---------|-----------------------|--------|
| 1. (d) | 2. (c) | 3. (i) (d) & (ii) (b) | 4. (d) |
| 5. (b) | 6. (d) | 7. (a) | 8. (a) |
| 10. (d) | 11. (c) | 12. (d) | |



- Q.12 Consider the partial implementation of a 2 bit counter using using T flip-flops following the sequence 0-2-3-1-0, as shown below:



To complete the circuit the input x should be

- Q_2'
- $Q_2 + Q_1$
- $(Q_1 \oplus Q_2)'$
- $Q_1 \oplus Q_2$