

$$\begin{aligned}
 &= (\bar{x}yz + \bar{x}y\bar{z} + \bar{x}yz + \bar{x}\bar{y}z) + (xyz + xy\bar{z} + \bar{x}yz + \bar{x}\bar{y}z) \\
 &= \bar{x}yz + \bar{x}y\bar{z} + \bar{x}yz + \bar{x}\bar{y}z + xyz + xy\bar{z} \\
 &= m_3 + m_2 + m_1 + m_0 + m_7 + m_6 \\
 \therefore F_2 &= \Sigma_m(0, 1, 2, 3, 6, 7)
 \end{aligned}$$

$$\begin{aligned}
 F_3 &= xy + \bar{x}\bar{y} \\
 &= xy(z + \bar{z}) + \bar{x}\bar{y}(z + \bar{z}) \\
 &= xyz + xy\bar{z} + \bar{x}\bar{y}z + \bar{x}\bar{y}\bar{z} \\
 &= m_7 + m_6 + m_1 + m_0 \\
 \therefore F_3 &= \Sigma_m(0, 1, 6, 7)
 \end{aligned}$$

Since, no. of input variables = 3

\therefore Required decoder = 3×8 decoder

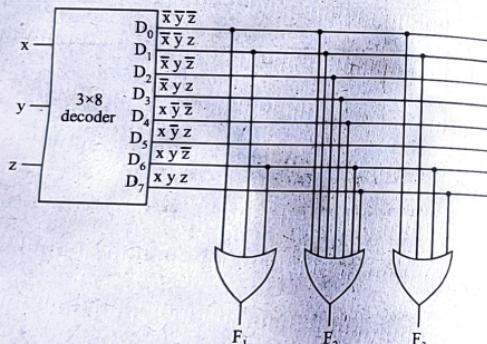


Fig.: Implementation of given functions using decoder and external gates

SEQUENTIAL LOGIC

7.1 Sequential Circuit

The logic circuits whose outputs at any instant of time depend not only on the present inputs but also on past outputs are called sequential circuits. A sequential circuit consists of a combinational circuit to which storage elements are connected to form a feedback path. The storage elements are devices capable of storing binary information.

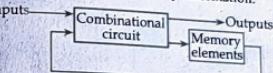


Fig.: Block diagram of a sequential circuit

Sequential circuit is slower in operation than combinational circuit. It may or may not contain clock input.

There are two types of sequential circuits:

i. Synchronous sequential circuit

It is a system whose behavior can be defined from the knowledge of its signals at discrete instant of time (i.e., by the clock signal parallelly driven by one clock signal).

ii. Asynchronous sequential circuit

It is a system whose behavior depends in the order in which its input signals change and can be affected at any instant of time (one's output is given to clock of another).

7.2 Event Driven Model and State Diagram

There are two distinct models by which asynchronous sequential logic circuit can be designed:

- Moore Model
- Mealy Model

Moore Model	Mealy Model
i. The output depends only on present state and not on input.	i. The output depends on present state as well as input.
ii. It requires more no. of states and thereby more hardware.	ii. It requires less no. of states and thereby less hardware to solve any problem.
iii. Output is generated one clock cycle later than Mealy.	iii. Output is generated one clock cycle earlier than Moore.
iv. The output remains stable over entire clock period and changes only when there occurs a state change.	iv. The output does not remain stable over entire clock period and changes over the same state.

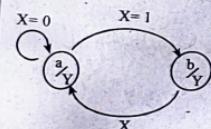


Fig.: State diagram representation

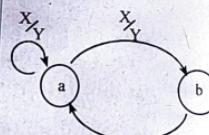


Fig.: State diagram representation

7.3 Flip flops and Their Types

A flip flop is a sequential circuit which is popularly known as a basic digital memory circuit. A flip flop stores 1 bit, therefore, it is called as 1-bit memory cell. It has two stable states: logic 1 and logic 0. It can flip from one state to another and then flop back, so it is called a flip flop and also known as a bistable multivibrator. The outputs of circuit (Q and \bar{Q}) will always be complementary. This means that if $Q = 0$, then $\bar{Q} = 1$ and vice-versa. They will never be equal; $Q = \bar{Q} = 0$ or 1 is an invalid state. If $Q = 1$, $\bar{Q} = 0$, it is called 1 state or SET state. If $Q = 0$, $\bar{Q} = 1$, it is called 0 state or RESET state. If the circuit is in reset state, then it continues to be in reset state and if it is in set state, then it continues to be in set state. This characteristic of the circuit illustrates that it can store 1 bit of digital information.

Application of flip flops:

1. As a memory element
2. In various types of registers
3. In counters/timers
4. As a delay element

The different types of flip flop are explained below:

- i. SR flip flop (set reset flip flop)
- ii. D flip flop (delay or data flip flop)
- iii. JK flip flop
- iv. T flip flop (toggle flip flop)
- v. Master-slave flip flop

7.3.1 SR Flip flop

SR flip flop has two inputs namely SET (S) and RESET (R) and two outputs Q and \bar{Q} . It can be implemented using NOR and NAND gates. The flip flop is often called a *latch* since it will hold, or latch, in either stable state.

NOR-based SR latch

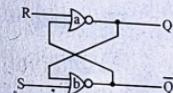


Fig.: NOR latch
Truth Table

S	R	Q	\bar{Q}	Remarks
1	0	1	0	Set
0	0	1	0	No change
0	1	0	1	Reset
0	0	0	1	No change
1	1	?	?	Invalid/ Forbidden

Case I: $S = 1, R = 0$

As $S = 1$, output of NOR gate b i.e., $\bar{Q} = 0$. This makes both inputs of NOR gate a at 0, therefore $Q = 1$. Now, both inputs of NOR gate b are 1. Hence, for $S = 1, R = 0$, $Q = 1$ and $\bar{Q} = 0$ (Set condition).

Case II: $S = 0, R = 1$

As $R = 1$, the output of NOR gate a i.e., $Q = 0$. This makes both inputs of NOR gate b are 0, therefore $\bar{Q} = 1$. Now, both inputs of NOR gate a are 1. Therefore, for $S = 0, R = 1, Q = 0, \bar{Q} = 1$ (Reset condition).

Case III: $S = 0, R = 0$

We know that the output of NOR gate a i.e., $Q = \overline{R + \bar{Q}}$

$$\text{As } R = 0, Q = \overline{0 + \bar{Q}} = \bar{Q} = Q$$

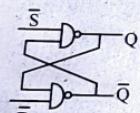
$$\text{Again the output of NOR gate b i.e., } \bar{Q} = \overline{S + Q}$$

$$\text{As } S = 0, \bar{Q} = \overline{0 + Q} = \bar{Q}$$

Therefore, for $S = 0, R = 0, Q$ and \bar{Q} do not change their state.

Case IV: $S = 1, R = 1$

If $S = 1, R = 1$ then the outputs Q and \bar{Q} both are forced to 0. Actually this is indeterminate state which must be avoided (since $Q = \bar{Q}$ which violates the basic requirements of flip-flop i.e., Q and \bar{Q} must be complement of each other).

NAND-based SR latch**Fig.: NAND latch****Case I: $\bar{S} = 0, \bar{R} = 0$**

When any input of NAND gate is 0, the output is forced to 1. Here $S = R = 0$, therefore, Q and \bar{Q} are forced to be equal to 1 which is indeterminate and must be avoided.

Case II: $S = 0, R = 1$ (i.e., $\bar{S} = 1, \bar{R} = 0$)

As $\bar{R} = 0$, output of NAND gate b i.e., $\bar{Q} = 1$. This makes both inputs of NAND gate a at 1, therefore, $Q = 0$. Hence, for $S = 0, R = 1$, the output $Q = 0$ and $\bar{Q} = 1$ (Reset condition).

Case III: $S = 1, R = 0$ (i.e., $\bar{S} = 0, \bar{R} = 1$)

As $\bar{S} = 0$, output of NAND gate a i.e., $Q = 1$. This makes both inputs of NAND gate b at 1, therefore, $\bar{Q} = 0$. Hence, for $S = 1, R = 0$, the output $Q = 1$ and $\bar{Q} = 0$ (Set condition).

Case IV: $S = 0, R = 0$, (i.e., $\bar{S} = 1, \bar{R} = 1$)

The output of NAND gate a, $Q = \overline{\bar{S} \cdot \bar{R}} = \overline{1 \cdot 1} = \bar{Q} = Q$

The output of NAND gate b, $\bar{Q} = \overline{Q \cdot \bar{R}} = \overline{1 \cdot 1} = \bar{Q}$

Thus, there is no change in the output if $\bar{S} = \bar{R} = 1$.

7.3.2 Gated Flipflop**Gated RS Flipflop**

The addition of two AND gates at the R and S inputs will result in a flipflop that can be enabled or disabled. When the ENABLE input is HIGH, information at the R and S will be transmitted directly to the outputs. The latch is said to be enabled. When the ENABLE input is LOW, the AND gate outputs must both be low and changes in neither R nor S will have any effect on the flip flop output Q. The latch is said to be disabled.

Truth Table

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

Fig.: Gated RS flipflop (NOR)

When two inputs signal R and S are applied to the circuit, there is a time delay between these two signals, which may lead to a wrong output result. In order to overcome this problem of unequal propagation/delay time of input signals, the gated flipflop is used.

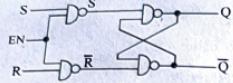


Fig.: Gated RS flipflop (NAND)

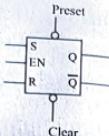


Fig.: Logic symbol

The characteristic table of gated RS flipflop is given below.

Q_n	S	R	Q_{n+1}	Remarks
0	0	0	0	No change
0	0	1	0	Reset
0	1	0	1	Set
0	1	1	x	Invalid
1	0	0	1	No change
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	x	Invalid



$$Q_n + 1 = S + Q_n R$$

This is the characteristic equation.

2. Gated D Flipflop (Delay or Data Flipflop)

The data flipflop has only one input called data (D) input and two outputs Q and \bar{Q} . It can be constructed from SR flipflop by inserting an inverter between S and R and assigning the symbol D to S input. The output Q is same as D input.

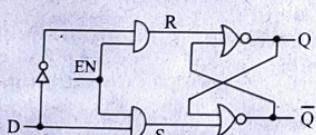


Fig.: Logic diagram of a gated D flipflop

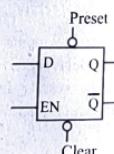


Fig.: Logic symbol

Characteristics table

Q_n	D	Q_{n+1}	Remarks
0	0	0	same as D
0	1	1	same as D
1	0	0	same as D
1	1	1	same as D



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

Advantages of D flipflop over RS flipflop:

In some events, both inputs become high in RS flipflop which is undesirable condition. This drawback of RS flipflop is overcome in D flipflop. There is only one input i.e., D which drive the flipflop.

3. Gated JK Flipflop

It is the refinement of RS flipflop as the indeterminate state (or invalid state) of RS is defined as 'Toggle' in JK. A JK flipflop can be constructed using two cross coupled NOR gates and two AND gates as shown in below figure.

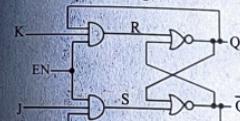


Fig.: JK flip - flop

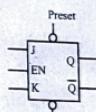


Fig.: Logic Symbol

Case I: $J = 0, K = 0$

If $J = 0, K = 0$, the output of two AND gates i.e., R and S are equal to 0.

We know that if $R = S = 0$, the output Q and \bar{Q} i.e., $Q_{n+1} = Q_n$ and

$$\overline{Q_{n+1}} = \overline{Q_n}$$

Case II: $J = 0, K = 1$

(i) Provided that $Q = 1$ and $\bar{Q} = 0$

$$\begin{aligned} S &= EN \cdot J \cdot \bar{Q} \\ &= 1 \cdot 0 \cdot 0 \\ &= 0 \end{aligned} \quad \begin{aligned} R &= EN \cdot K \cdot Q \\ &= 1 \cdot 1 \cdot 1 \\ &= 1 \end{aligned}$$

$$\text{i.e., } Q_{n+1} = 0$$

(ii) Provided that, $Q = 0$ and $\bar{Q} = 1$

$$\begin{aligned} S &= EN \cdot J \cdot \bar{Q} & R &= EN \cdot K \cdot Q \\ &= 1 \cdot 1 \cdot 0 & &= 1 \cdot 1 \cdot 0 \\ &= 0 & &= 0 \end{aligned}$$

i.e., remain in reset state, $Q_{n+1} = 0$

Case II: $J = 1, K = 0$

(i) Provided that $Q = 1$ and $\bar{Q} = 0$

$$\begin{aligned} S &= EN \cdot J \cdot \bar{Q} & R &= EN \cdot K \cdot Q \\ &= 1 \cdot 1 \cdot 0 & &= 1 \cdot 0 \cdot 1 \\ &= 0 & &= 0 \end{aligned}$$

i.e., $Q_{n+1} = 1$

(ii) Provided that $Q = 0$ and $\bar{Q} = 1$

$$\begin{aligned} S &= EN \cdot J \cdot \bar{Q} & R &= EN \cdot K \cdot Q \\ &= 1 \cdot 1 \cdot 1 & &= 1 \cdot 0 \cdot 0 \\ &= 1 & &= 0 \end{aligned}$$

i.e., $Q_{n+1} = 1$

Case IV: $J = 1, K = 1$

(i) Provided that $Q = 1$ and $\bar{Q} = 0$

$$\begin{aligned} S &= EN \cdot J \cdot \bar{Q} & R &= EN \cdot K \cdot Q \\ &= 1 \cdot 1 \cdot 0 & &= 1 \cdot 1 \cdot 1 \\ &= 0 & &= 1 \end{aligned}$$

i.e., $Q_{n+1} = 0$

(ii) Provided that $Q = 0$ and $\bar{Q} = 1$

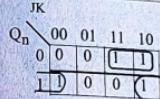
$$\begin{aligned} S &= EN \cdot J \cdot \bar{Q} & R &= EN \cdot K \cdot Q \\ &= 1 \cdot 1 \cdot 1 & &= 1 \cdot 1 \cdot 0 \\ &= 1 & &= 0 \end{aligned}$$

i.e., $Q_{n+1} = 1$

The outputs are inverted i.e., toggle from reset to set [$Q = 0$ to $Q = 1$]

Characteristics table

Q_n	J	K	Q_{n+1}	Remarks
0	0	0	0	No change
0	0	1	0	Reset
0	1	0	1	Set
0	1	1	1	Toggle
1	0	0	1	No change
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	0	Toggle



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

Gated Toggle Flipflop (T flipflop)

It is made by shorting the J and K input of JK flipflop to single input T. The output toggles each time for $T = 1$.

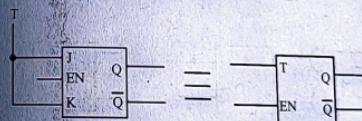


Fig.: Logic symbol

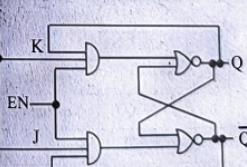


Fig.: Logic diagram of T flipflop

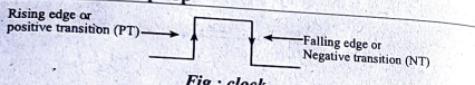
Characteristic table:

Q_n	T	Q_{n+1}	Remarks
0	0	0	No change
0	1	1	Toggle
1	0	1	No change
1	1	0	Toggle



$$\begin{aligned} Q_{n+1} &= \bar{Q}_n T + Q_n \bar{T} \\ &= (Q_n \oplus T) \end{aligned}$$

7.3.3 Edge Triggered Flipflop



Everything is same as gated (or level triggered), only the difference is the clock pulse type which makes the flip-flop enable.

1. Edge Triggered RS Flipflop

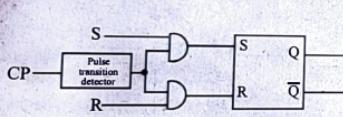


Fig.: Logic diagram of RS flipflop

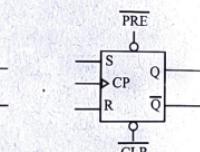


Fig.: Logic symbol of positive edge triggered RS flipflop

Truth Table

CP	S	R	Q_{n+1}	Remarks
↑	0	0	Q_n	No change
↑	0	1	0	Reset
↑	1	0	1	Set
↑	1	1	?	Invalid

Similarly, for negative edge triggered RS flipflop, the logic symbol is

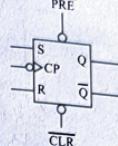
Fig.: Logic symbol for negative edge triggered RS flipflop
Edge Triggered D Flipflop

Fig.: Logic diagram of D flipflop

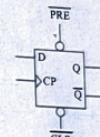


Fig.: Logic symbol of positive edge triggered D flipflop

Truth Table

CP	D	Q_{n+1}
0	x	Q_n
↑	0	0
↑	1	1

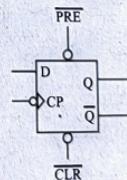


Fig.: Logic symbol of negative edge triggered D flipflop

3. Edge Triggered JK Flipflop

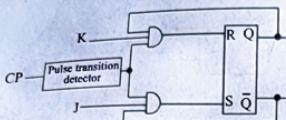


Fig.: Logic diagram of JK flipflop

CP	J	K	Q_{n+1}
↑	0	0	Q_n (Last state)
↑	0	1	0 (Reset)
↑	1	0	1 (Set)
↑	1	1	\bar{Q} (Toggle)

Fig.: Logic symbol of positive edge triggered JK flipflop

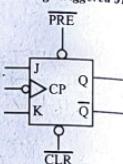


Fig.: Logic symbol of negative edge triggered JK flipflop

Race around condition: In JK flipflop, if the duration of clock pulse is greater than delay between input and output, the output may come back as the input to the flipflop and may result indeterminate outputs within the same clock pulse (instead of a single state output). This problem is known as "race around condition" in JK flipflop.

4. Master Slave Flipflop

Master slave flipflop is required to remove the race around condition. Master slave flipflop consists of two flipflops; one serve as master and another as slave. Master is positive-edge triggered and slave is negative-edge triggered.

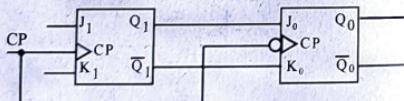


Fig.: Logic symbol of master slave JK flipflop

Master

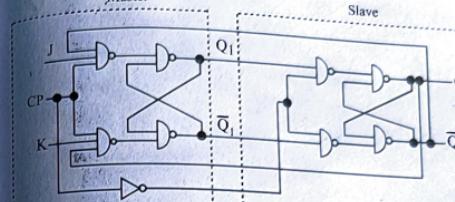


Fig.: Logic diagram of JK master slave flipflop

Initially, the master is positive-edge triggered and the slave is negative-edge triggered. Therefore, the master responds to its J and K inputs before the slave. If J=1 and K=0, the master sets. The high Q output of the master drives the J input of the slave, so on negative edge of the clock (this makes slave positive-edge triggered), the slave sets copying the action of the master. If J=0, K=1, the master resets when it is positive-edge triggered.

The high Q output of the master goes to the K input of the slave. So, on negative level of the clock, slave resets. Again, the slave has copied the master. If J=1, K=1, first master toggles its output and later slave does the same. If J=K=0, the flip-flop is disabled and Q remains unchanged.

7.3.4 Asynchronous Inputs (Direct Inputs)

PRESET and CLEAR are called asynchronous inputs because they activate the flipflop independent of clock. It may be active low or high.

If CLR is LOW, then the output is reset. If PRE is LOW, then the output is set.

The block diagram of edge triggered JK flipflop along with asynchronous inputs is shown below.

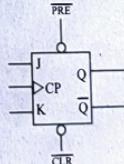


Fig.: Asynchronous inputs in JK flipflop

7.4 Various Representation of Flip flops

- Characteristic equation of FF
- FF as finite state machine
- FF excitation table

1) Characteristic equations of FF

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

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

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

$$T : Q_{n+1} = \bar{Q}_n T + Q_n \bar{T}$$

2) FF as finite state machine

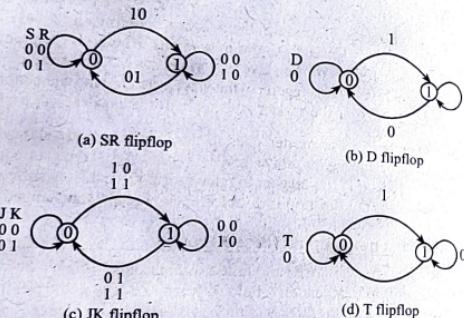


Fig: State transition diagram

3) Flip flop excitation table

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

Excitation table of flipflops

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

EXAMPLE:

- Convert SR flipflop to JK flipflop.

Step 1:

First we generate the truth table for JK flip-flop i.e., note $Q_n \rightarrow Q_{n+1}$ transition for a given combination of JK input.

Step 2:

Next on the same table for $Q_n \rightarrow Q_{n+1}$ transition identify the excitation table for SR flip-flop such tables are known as synthesis table.

J_n	K_n	Q_n	Q_{n+1}	S_n	R_n
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

Step 3:

Write SR inputs as a function JK inputs and present state Q_n . It is done through K-map.

For S_n

$K_n Q_n$		00	01	11	10
J_n	Q_n	0	0	x	0
J_n	Q_n	1	1	x	0
J_n	Q_n	1	1	x	1

$$S_n = J_n \bar{Q}_n$$

For R_n

$K_n Q_n$		00	01	11	10
J_n	Q_n	0	x	x	x
J_n	Q_n	1	0	0	0
J_n	Q_n	1	0	0	0

$$R_n = K_n Q_n$$

Step 4:

Design the circuit using SR flipflop and input JK.

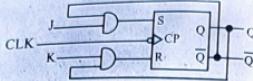


Fig.: JK flipflop from SR flipflop

2. Convert JK flipflop to SR flipflop.

=

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

for, J

S	00	01	11	10
0	0	x	x	0
1	x	x	x	x

$J = S$

S	00	01	11	10
0	x	0	1	x
1	x	0	x	x

$K = R$



Fig.: SR flipflop from JK flipflop

3. Convert SR flipflop to D flipflop.

=

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

D	Q _n	Q _{n+1}
0	0	0
1	0	1

D	Q _n	Q _{n+1}
0	0	0
1	1	0

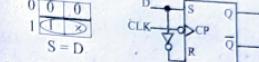


Fig.: SR flipflop to D flipflop

7.5 Analysis of Clocked Sequential Circuit

The steps to be followed for designing are:

- 1) Choose the model to be implemented (Moore or Mealy).
- 2) Draw the state transition diagram from word description problem.
- 3) Draw the next state table from the given information.
- 4) The number of states may be reduced by state reduction method if necessary.
- 5) Assign the states with binary value.
- 6) Choose the type of flipflop to be used, determine number of flipflops to be used based on states, and derive the excitation table based on present state (Q_n) to next state (Q_{n+1}) transition and output table.
- 7) Using K-map, simplify each input of flipflop and circuit output.
- 8) Draw the logic diagram.

EXAMPLE:

A synchronous machine has one bit serial input 'X'. The output 'Z' of a machine is to be set high when the input contains the message '100'. Draw the state diagram, derive the transition table (state table), excitation table, and design the circuit diagram.

Solution:

Using Mealy Model

Step 1: We choose Mealy model

Step 2: State diagram

Circuit is initialized with state a.

If $X = 0$, then it remains in same state a. If $X = 1$, then it moves to state b (because first bit is detected).

At state b, if $X = 0$, the first two pattern is detected, so moves to state c. But if $X = 1$, it remains in same state b because state b is the state which have detected 1.

At state c, if $X = 0$, the input stream is 100, so the output is high and it goes to state a since it has to detect another 1 from starting bit of sequence 100. But if $X = 1$, it moves to state b because state b has detected 1.

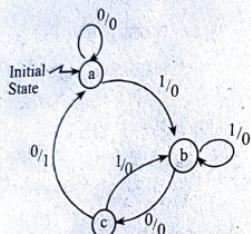


Fig.: State transition diagram

Step 3: Next state table (transition table)

Previous State	Next State		Output	
	$X = 0$	$X = 1$	$X = 0$	$X = 1$
A	a	b	0	0
B	c	b	0	0
C	a	b	1	0

Step 4: No reduction required.

Step 5: Binary assignment

Let $a = 00$, $b = 01$, $c = 10$

Step 6: No. of flipflops = 2. We design using JK flipflop.

Present State	Present Input	Next State	Output	Flipflop Excitation					
B_n	A_n	X	B_{n+1}	A_{n+1}	X	J_B	K_B	J_A	K_A
0	0	0	0	0	0	0	x	0	x
0	0	1	0	1	0	0	x	0	x
0	1	0	1	0	0	0	x	1	x
0	1	1	0	1	0	1	x	x	1
1	0	0	0	0	1	x	1	0	x
1	0	1	0	1	0	x	1	1	x

Here, 11 is the unused table, so values corresponding to it are don't care conditions.

Step 7:

I_B		A_n				
B_n	00 01 11 10	0	0	0	1	
0	x x x x	0	0	0	1	
1	x x	1	1	1	x	x

$$J_B = A_n \bar{X}$$

I_K		A_n				
B_n	00 01 11 10	0	x x x	x x	x x	
0	x x x x	0	x x x	x x	x x	
1	x x	1	1	x x	x x	

$$K_n = 1$$

I_J		$A_n X$				
B_n	00 01 11 10	0	0	1	x	x
0	x x x x	0	0	1	x	x
1	x x	1	1	x x	x x	

$$J_A = X$$

I_Y		$A_n X$				
B_n	00 01 11 10	0	0	0	0	
0	x x x x	0	0	0	0	
1	x x	1	0	1	x x	

$$Y = B_n X$$

I_K_A		$A_n X$				
B_n	00 01 11 10	0	x	x	0	1
0	x x x x	0	x	x	0	1
1	x x	1	x	x	x	x

$$K_A = \bar{X}$$

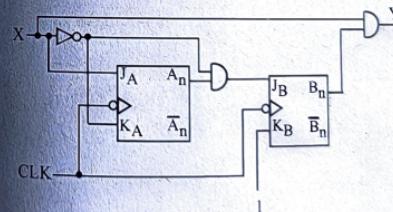


Fig.: Logic diagram of '100' sequence detector

7.6 Decoder as Memory Device

For n-variable function, sum of minterms form can be implemented using a single n to 2^n decoder to generate minterms and an OR gate to form the sum.

Any combinational circuit with 'n' inputs and 'm' outputs can be implemented with n to 2^m decoder with 'm' OR gates.

EXAMPLE: Implement the following Boolean expression using decoder.

$$F_1 = \Sigma m(0, 4, 6)$$

$$F_2 = \Sigma m(0, 5)$$

$$F_3 = \Sigma m(1, 2, 3, 7) \text{ where } F \text{ is a function of } A, B, \text{ and } C$$

$$\Rightarrow \text{No. of inputs} = 3$$

$$\text{Decoder size} = 3 \cdot 2^3 = 3 \cdot 8$$

$$\text{no. of outputs} = 3 (F_1, F_2, F_3)$$

$$\text{no. of OR gates} = 3$$

$$F_1(A, B, C) = \Sigma m(0, 4, 6)$$

$$= m_0 + m_4 + m_6$$

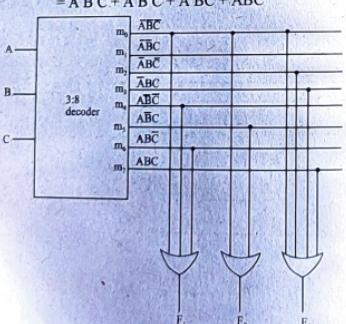
$$= \bar{A} \bar{B} \bar{C} + A \bar{B} \bar{C} + A B \bar{C}$$

$$F_2(A, B, C) = m_0 + m_5$$

$$= \bar{A} B \bar{C} + A \bar{B} C$$

$$F_3(A, B, C) = m_1 + m_2 + m_3 + m_7$$

$$= \bar{A} B C + \bar{A} B \bar{C} + \bar{A} \bar{B} C + A B C$$



EXAMPLE: Implement full adder using decoder.

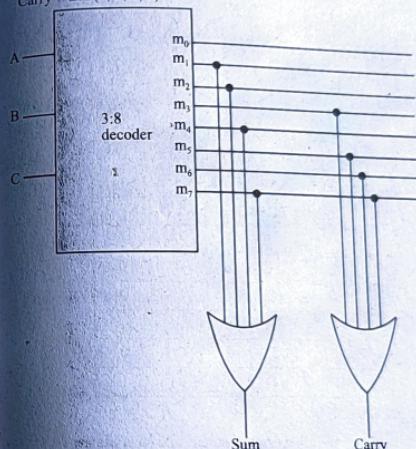
For full adder, no. of inputs = 3 (A, B, C)

Decoder size = 3:8

No. of outputs = 2 (sum, carry)

Sum = $\Sigma m(1, 2, 4, 7)$

Carry = $\Sigma m(3, 5, 6, 7)$



7.7 State Reduction and Assignment

Redundant States

Two states are equivalent if they have same next state and output conditions. If they are equivalent, one of them is redundant i.e., can be eliminated or replaced. The necessary condition for equivalent is "output must be same".

Row elimination method:

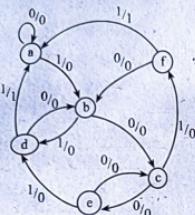


Fig.: State diagram to be reduced

Let's suppose the sequential machine state diagram as follows:

Present State	Next State		Present Output	
	X = 0	X = 1	X = 0	X = 1
A	A	b	0	0
B	C	d	0	0
C	E	f	0	0
D	B	a	0	1
E	C	d	0	0
F	B	a	0	1

a. **Original table**

In table (a), b and e have same output and next states, so they are equivalent. Therefore e row is eliminated and substituted e by b.

Present State	Next State		Present Output	
	X = 0	X = 1	X = 0	X = 1
A	a	b	0	0
B	c	d	0	0
C	b	f	0	0
D	b	a	0	1
F	b	a	0	1

b. **After one row elimination**

d and f states also have same output and same next states, so they are also equivalent. Therefore, f row is eliminated and substituted f by d.

Present State	Next State		Present Output	
	X = 0	X = 1	X = 0	X = 1
A	a	b	0	0
B	c	d	0	0
C	b	d	0	0
D	b	a	0	1

c. **After two row elimination**

Here, for X = 0, b → c and c → b, the states c and d are equivalent

Present State	Next State		Present O/P	
	X = 0	X = 1	X = 0	X = 1
A	a	b	0	0
B	b	d	0	0
D	b	a	0	1

d. **Final reduced table after three row elimination**

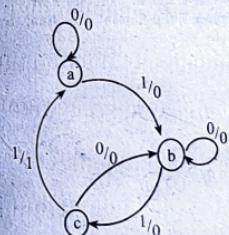


Fig.: Reduced state diagram

7.8 Synchronous and Asynchronous Logic

A sequential logic circuit contains flipflops as memory elements and also contains logic gates as combinational circuits. Analysis of a circuit helps to explain its performance.

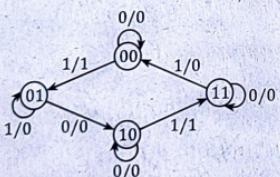
Sequential circuits may be classified into following two categories:

- Synchronous sequential circuits
- Asynchronous sequential circuits

Synchronous sequential circuit	Asynchronous sequential circuit
1. These circuits are easy to design.	1. Difficult to design.
2. A clocked flipflop acts as a memory element.	2. Unlocked flipflop or time delay element is used as memory element.
3. These circuits are slower because the delays correspond to those of the memory element.	3. Faster as the clock is not present.
4. The status of memory element is affected only at active edge of clock if the input is changed.	4. The status of memory element will change any time as soon as the input is changed.

SOLUTION TO IMPORTANT AND EXAM QUESTIONS

1. Realize the following state diagram into a circuit using T flip-flop.
How can you replace T flip-flops of your final circuit with JK flip-flops?
[2021 Fall]



Solution:

Excitation table of T flipflop

Q _n	Q _{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

Next state table

Present state	A _n	Next state		Output (Y)	
		at X = 0	at X = 1	at X = 0	at X = 1
0	0	0	0	1	0
0	1	1	0	0	0
1	0	1	0	1	0
1	1	1	1	0	0

Synthesis table

Present state	Input	X	B _{n+1}	A _{n+1}	Y	T _B	T _A
0	0	0	0	0	0	0	0
0	0	1	0	1	1	0	1
0	1	0	1	0	0	1	1
0	1	1	0	1	0	0	0
1	0	0	1	0	0	0	0
1	0	1	1	1	1	0	1
1	1	0	1	1	0	0	0
1	1	1	0	0	0	1	1

Using K-map,

For T_B,

BX		AB			
A	B	00	01	11	10
0	0	0	0	0	①
1	0	0	①	0	0

$$T_B = AB\bar{X} + A\bar{B}X$$

$$= B(A\bar{X} + \bar{A}X)$$

$$= B(A \oplus X)$$

For T_A :

A	BX	00	01	11	10
0	0	0	1	0	1
1	0	1	1	0	0

$$T_A = \bar{B}X + ABX + \bar{A}B\bar{X}$$

$$= \bar{B}X + B(AX + A\bar{X}) = \bar{B}X + B(A \oplus X)$$

For output Y,

A	BX	00	01	11	10
0	0	0	1	0	0
1	0	1	1	0	0

$$Y = \bar{B}X$$

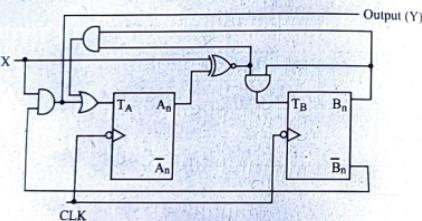
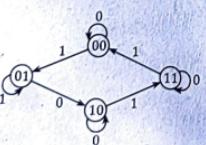


Fig.: Logic diagram of given state diagram

T flip-flops can be replaced in final circuit with JK flip-flops by making inputs J and K and making it single input by making both inputs same.

2. Design synchronous sequential circuit for the given state diagram using T flip-flop.
[Fall 2020]



Solution:

First, we make synthesis table according to given state diagram.

Excitation table of T flipflop

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

Synthesis Table

Q_{1n}	Q_{0n}	X	Present state		Next state		FF Excitation	
			Q_{1n+1}	Q_{0n+1}	Q_{1n+1}	Q_{0n+1}	T_1	T_0
0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1	1
0	1	1	1	0	1	0	0	0
1	0	0	0	1	0	0	0	0
1	0	1	1	1	1	0	0	1
1	1	1	0	1	1	1	0	0
1	1	1	1	0	0	0	1	1

For T_1 :

$Q_{0n}X$	00	01	11	10
Q _{1n}	0	0	0	1
	0	0	0	0

$$T_1 = \bar{Q}_{1n}Q_{0n}\bar{X} + Q_{1n}Q_{0n}X$$

$$= Q_{0n}(\bar{Q}_{1n}\bar{X} + Q_{1n}X)$$

$$= Q_{0n}(\overline{Q_{1n} \oplus X})$$

For T_0 ,

$Q_{0n}X'$	00	01	11	10
0	0	1	0	1
1	0	1	1	0

$$T_0 = \bar{Q}_{0n}X + \bar{Q}_{1n}Q_{0n}\bar{X} + Q_{1n}X$$

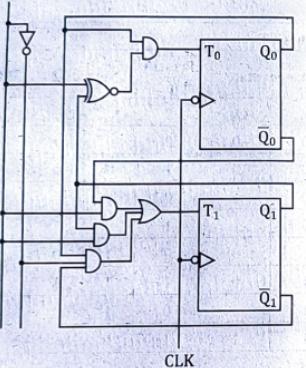
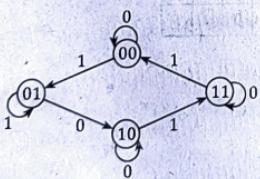


Fig.: Circuit diagram

3. Realize the following state diagram into a circuit using SR flipflop.
[Spring 2020]



Solution:

Excitation table of SR flipflop

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

Synthesis table

Present state	Present input	Next state		FF Excitation						
		Q_{1n}	Q_{0n}	X	Q_{1n+1}	Q_{0n+1}	S_0	R_0	S_1	R_1
0	0	0	0	0	0	0	0	x	0	x
0	0	1	0	0	1	0	0	x	1	0
0	1	0	0	1	0	1	1	0	0	1
0	1	1	0	0	1	0	0	x	x	0
1	0	0	0	1	0	0	x	0	0	x
1	0	1	1	1	1	1	x	0	0	1
1	1	0	0	1	1	1	x	0	0	1
1	1	1	1	0	0	0	0	1	x	0

Using K-map,

For S_0 ,

$Q_{0n}X'$	00	01	11	10
0	0	0	0	1
1	x	x	1	x

$$\therefore S_0 = Q_{0n}\bar{X}$$

For R_0 ,

$Q_{0n}X'$	00	01	11	10
0	x	x	x	0
1	0	0	1	0

$$\therefore R_0 = Q_{0n}X$$

For S_1 ,

Q_{in}	00	01	11	10
0	0	1	x	0
1	0	1	x	0

$$\therefore S_1 = X$$

For R_1 ,

Q_{in}	00	01	11	10
0	x	0	0	1
1	x	0	0	1

$$\therefore R_1 = \bar{X}$$

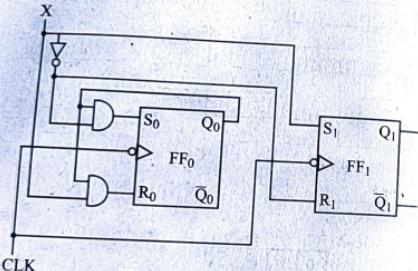
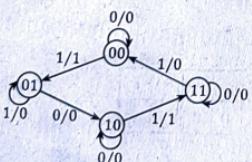


Fig.: Circuit diagram using SR flipflop

4. Design a sequential circuit using JK flipflop for the following state diagram.
[Fall 2019]



Solution:

Excitation table of JK flipflop

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

Synthesis table

Present state	Present input	Next state		Output	FF Excitation			
		Q_{in+1}	Q_{0n+1}		J_1	K_1	J_0	K_0
0	0	0	0	0	0	x	0	x
0	0	1	0	1	1	0	x	1
0	1	0	1	0	0	1	x	x
0	1	1	0	1	0	0	x	x
1	0	0	1	0	0	x	0	x
1	0	1	1	1	1	x	0	1
1	1	0	1	1	0	x	0	1
1	1	1	1	0	0	x	1	x

Using K-map,

For J_1 ,

Q_{in}	00	01	11	10
0	0	0	0	1
1	x	x	x	x

$$\therefore J_1 = Q_{0n} \bar{X}$$

For K_1 ,

Q_{in}	00	01	11	10
0	x	x	x	x
1	0	0	1	0

$$K_1 = Q_{0n} X$$

Q_{in}	00	01	11	10
0	0	1	x	x
1	0	1	x	x

$$\therefore J_0 = X$$

For K_0 ,

Q_{in}	00	01	11	10
0	x	x	0	1
1	x	x	1	0

$$\therefore K_0 = \bar{Q}_{in}X + Q_{in}X = Q_{in} \oplus X$$

For output Y ,

Q_{in}	00	01	11	10
0	0	1	0	0
1	0	1	0	0

$$\therefore Y = \bar{Q}_{in}X$$

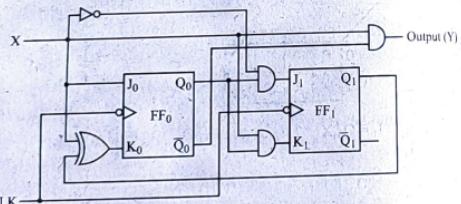
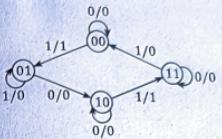


Fig.: Circuit diagram using JK flip flop

5. Design a sequential circuit corresponding to the given state diagram using SR flip flop for the following state diagram.
[Fall 2018]



Solution:

Excitation table of SR 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

Synthesis table

Present state Q_n	Present input (X)	Next state		Output (Y)	FF excitation			
		Q_{n+1}	Q_{n+1}		S_1	R_1	S_0	R_0
00	0	00	00	0	0	x	0	x
00	1	01	01	1	0	x	1	0
01	0	10	01	0	1	0	0	1
01	1	11	01	0	0	x	x	0
10	0	01	10	0	x	0	0	x
10	1	11	10	1	x	0	1	0
11	0	00	11	0	x	0	x	0
11	1	01	11	0	0	0	1	0

Using K-map,

For S_1 ,

Q_{in}	00	01	11	10
0	0	0	0	1
1	x	x	1	x

$$S_1 = Q_{in}\bar{X}$$

For R_1 ,

$Q_{1n}X$	00	01	11	10	
Q_{0n}	0	x	x	(1)	0
0	0	0	0	(1)	0

$$R_1 = Q_{1n}X$$

For S_0 ,

$Q_{1n}X$	00	01	11	10	
Q_{0n}	0	0	(1)	x	0
1	0	(1)	0	x	

$$\therefore S_0 = \bar{Q}_{1n}X$$

For R_0 ,

$Q_{1n}X$	00	01	11	10
Q_{0n}	0	x	0	(1)
1	x	0	(1)	0

$$\therefore R_0 = Q_{0n}Q_{1n}X + \bar{Q}_{0n}\bar{X}$$

For output Y ,

$Q_{1n}X$	00	01	11	10
Q_{0n}	0	0	(1)	0
1	0	(1)	0	0

$$\therefore Y = \bar{Q}_{1n}X$$

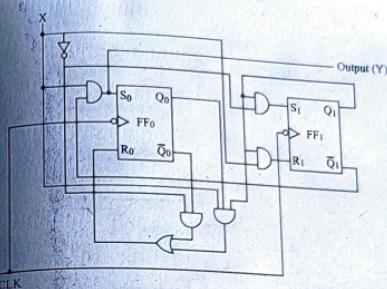


Fig.: Circuit diagram of sequential circuit using SR flipflop

6. Explain the operation of RS flipflop with the help of characteristics table. How can it be converted into T flipflop?
[Spring 2017]

Solution:

RS flipflop to T flipflop

Step 1: First, we generate truth table of T flipflop i.e. note $Q_n \rightarrow Q_{n+1}$ transition for a given combination of T flipflop.

Step 2: On the same table for $Q_n \rightarrow Q_{n+1}$ transition, we identify excitation table for each SR flipflop (synthesis table)

T_n	Q_n	Q_{n+1}	S_n	R_n
0	0	0	0	x
0	1	1	x	0
1	0	1	1	0
1	1	0	0	1

Step 3: Using K-map, find expression for S_n, R_n .

For S_n ,

T_n	Q_n	0	1
0	0	x	
1	(1)	0	0

$$\therefore S_n = T_n \bar{Q}_n$$

For R_n ,

T_n	Q _n	0	1
0	x	0	
1	0	(1)	

$$\therefore R_n = T_n Q_n$$

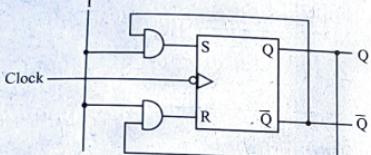
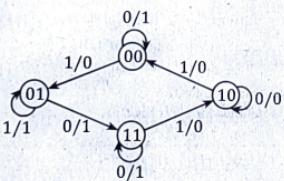


Fig.: T flip-flop using SR flipflop

7. Realize the following function state diagram into the circuit using SR flipflop.

[Spring 2016]



Solution:

Excitation table of SR flipflop

Q _n	Q _{n+1}	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

Synthesis table

Present state Q_{in}	Q_{in}	Present input (X)		Next state		Output (Y)	FF excitation			
		Q_{n+1}	Q_{n+1}	S_1	R_1		S_0	R_0	S_1	R_1
0	0	0	0	1	0	x	0	x	0	x
0	0	1	0	1	0	0	0	x	1	0
0	1	0	1	1	1	1	0	x	0	0
0	1	1	0	1	1	1	0	x	x	0
1	0	0	1	0	0	0	x	0	0	x
1	0	1	0	0	0	0	0	1	-Q	x
1	1	0	1	1	1	x	0	x	0	0
1	1	1	1	0	0	0	x	0	0	1

Using K-map,

For S_1 ,

$Q_{in} X$	00	01	11	10
Q_{in}	0	0	0	1
	0	x	0	x

$$S_1 = Q_{in} \bar{X}$$

For R_1 ,

$Q_{in} X$	00	01	11	10
Q_{in}	0	x	x	0
	1	0	1	0

$$R_1 = \bar{Q}_{in} X$$

For S_0 ,

$Q_{in} X$	00	01	11	10
Q_{in}	0	0	1	x
	1	0	0	x

$$S_0 = \bar{Q}_{in} X$$

For R_0 ,

$Q_{0n} X$	00	01	11	10
0	x	0	0	0
1	x	(x)	1	0

$$R_0 = Q_{0n} X$$

For output Y,

$Q_{0n} X$	00	01	11	10
0	1	0	(1)	1
1	0	0	0	1

$$Y = \bar{Q}_{1n} \bar{X} + \bar{Q}_{1n} Q_{0n} + Q_{0n} \bar{X}$$

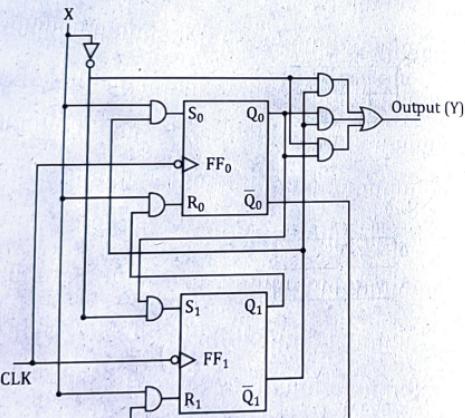


Fig.: Detailed circuit diagram

8. Design a counter with the following binary sequence: 0, 1, 3, 2, 6, 4, 5, 7 and repeat. Use T flip-flop.
[Fall 2016]

Solution:

Given sequence: 0, 1, 3, 2, 6, 4, 5, 7
Now, the state diagram will be

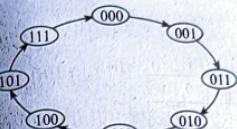


Fig.: State diagram

Excitation table of T flip-flop

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

Synthesis table

Present state	Next state					Excitation table			
	Q_{2n}	Q_{1n}	Q_{0n}	Q_{2n+1}	Q_{1n+1}	Q_{0n+1}	T_2	T_1	T_0
0 0 0	0	0	0	0	0	1	0	0	1
0 0 1	0	1	0	0	1	1	0	1	0
0 1 0	1	0	0	1	1	0	1	0	0
0 1 1	1	1	0	1	0	1	0	0	1
1 0 0	0	0	1	0	1	0	0	0	1
1 0 1	0	1	1	1	1	1	0	1	0
1 1 0	1	0	0	1	0	0	0	1	0
1 1 1	1	1	0	0	0	0	1	1	1

Using K-map,

For T_2 ,

$Q_{2n} Q_{0n}$	00	01	11	10
0	0	0	0	(1)
1	0	0	(1)	0

$$\begin{aligned}\therefore T_2 &= Q_{2n}Q_{1n}Q_{0n} + \bar{Q}_{2n}Q_{1n}\bar{Q}_{0n} \\ &= Q_{1n}(Q_{2n}Q_{0n} + \bar{Q}_{2n}Q_{0n}) \\ &= Q_{1n}(Q_{0n} \odot Q_{2n})\end{aligned}$$

For T_1 ,

$Q_{2n}Q_{0n}$	00	01	11	10
0	0	1	0	0
1	0	1	1	1

$$\begin{aligned}\therefore T_1 &= \bar{Q}_{1n}Q_{0n} + Q_{2n}Q_{0n} \\ &= Q_{0n}(\bar{Q}_{1n} + Q_{2n})\end{aligned}$$

For T_0 ,

$Q_{2n}Q_{0n}$	00	01	11	10
0	1	0	1	0
1	1	0	1	0

$$\begin{aligned}\therefore T_0 &= \bar{Q}_{2n}\bar{Q}_{0n} + Q_{2n}Q_{0n} \\ &= Q_{0n} \odot Q_{2n}\end{aligned}$$

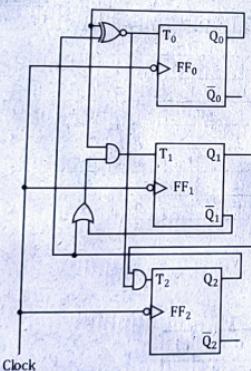
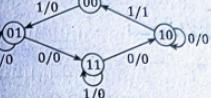


Fig.: Detailed circuit diagram of counter using T flipflop

Design a sequential circuit corresponding to the given state diagram using JK flipflop.
[Spring 2015]



Solution:

Excitation table of JK flipflop

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

Synthesis table

Present state	Present input (X)	Next state		Output (Y)	FF Excitation			
		Q_{1n+1}	Q_{0n+1}		J_1	K_1	J_0	K_0
0 0	0	0	0	0	0	x	0	x
0 0	1	1	0	1	0	0	x	1
0 1	0	0	1	1	0	1	x	x
0 1	1	1	0	1	0	0	x	x
1 0	0	1	0	0	0	x	0	x
1 0	1	0	0	1	x	1	0	x
1 1	0	1	0	0	0	x	0	x
1 1	1	1	1	0	x	0	x	0

Using K-map,

$Q_{0n}X$	00	01	11	10
0	0	0	0	(1)
1	x	x	x	x

$$J_1 = Q_{0n}\bar{X}$$

For K_1 ,

$Q_{0n}X$	00	01	11	10
0	x	x	x	x
1	0	1	0	0

$$K_1 = \bar{Q}_{0n}X$$

For J_0 ,

$Q_{0n}X$	00	01	11	10
0	0	1	x	x
1	0	0	x	x

$$J_0 = \bar{Q}_{1n}X$$

For K_0 ,

$Q_{0n}X$	00	01	11	10
0	x	x	0	0
1	x	x	0	1

$$K_0 = Q_{1n}\bar{X}$$

For output Y,

$Q_{0n}X$	00	01	11	10
0	0	0	0	0
1	0	1	0	0

$$Y = Q_{1n}\bar{Q}_{0n}X$$

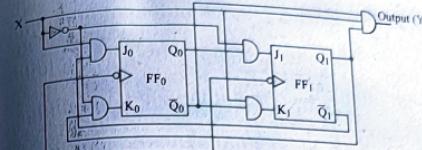
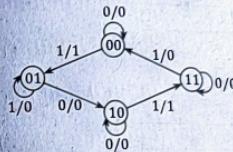


Fig.: Circuit diagram for the sequential circuit

10. Design a sequential circuit corresponding to the given state diagram using D flipflop for the following state diagram.
[Fall 2015]



Solution:

Excitation table of D flipflop		
Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Synthesis table

Present state Q_{1n}	Present state Q_{0n}	Present input (X)	Present	Next state Q_{1n+1}	Next state Q_{0n+1}	Output (Y)	FF Excitation	
							D_1	D_0
0	0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0	1
0	1	0	0	1	0	0	1	0
0	1	1	1	0	1	0	0	1
1	0	0	0	1	0	0	1	0
1	0	1	1	1	1	1	1	1
1	1	0	0	1	1	0	1	1
1	1	1	1	0	0	0	0	0

Using K-map,

For D_1 ,

	$\bar{Q}_{0n}X$	00	01	11	10
\bar{Q}_{1n}	0	0	0	1	
1	1	1	0	1	

$$D_1 = Q_{1n}\bar{Q}_{0n}$$

For D_0 ,

	$\bar{Q}_{0n}X$	00	01	11	10
\bar{Q}_{1n}	0	0	1	0	
1	0	1	0	1	

$$D_0 = \bar{Q}_{0n}X + \bar{Q}_{1n}X + Q_{1n}Q_{0n}\bar{X}$$

For output Y,

	$\bar{Q}_{0n}X$	00	01	11	10
\bar{Q}_{1n}	0	0	1	0	0
1	0	1	0	0	0

$$Y = \bar{Q}_{0n}X$$

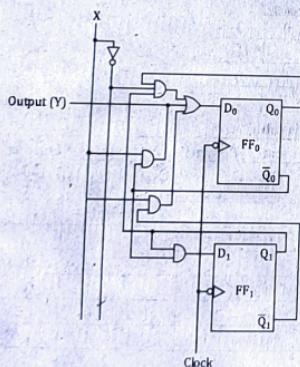
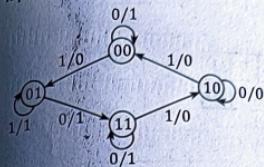


Fig.: Circuit diagram for given sequential circuit using D flipflop

11. Realize the following state diagram into a circuit using JK flipflop.



[Spring 2014]

Solution:

Excitation table of JK flipflop

Q_i	Q_{i+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

Synthesis table

Present state Q_{in}	Present input (X)	Next state		Output (Y)	FF excitation			
		Q_{in+1}	Q_{0n+1}		J_1	K_1	J_0	K_0
0 0	0	0	0	1	0	x	0	x
0 0	1	0	1	0	0	0	x	1
0 1	0	1	1	1	1	1	x	x
0 1	1	0	1	1	1	0	x	x
1 0	0	1	0	0	x	0	0	x
1 0	1	0	0	0	x	1	0	x
1 1	0	1	1	1	x	0	x	0
1 1	1	1	0	0	x	0	x	1

Using K-map,

For J_1 ,

	$Q_{0n}X$	00	01	11	10
Q_{1n}		0	0	0	1
0		0	x	x	x
1		x	x	x	x

$$\therefore J_1 = Q_{0n}\bar{X}$$

For K_1 ,

	$Q_{0n}X$	00	01	11	10
Q_{1n}		0	x	x	x
0		0	x	0	0
1		0	0	0	0

$$\therefore K_1 = \bar{Q}_{0n}X$$

For J_0 ,

	$Q_{0n}X$	00	01	11	10
Q_{1n}		0	0	1	x
0		0	0	x	x
1		0	0	x	x

$$\therefore J_0 = \bar{Q}_{1n}X$$

For K_0 ,

	$Q_{0n}X$	00	01	11	10
Q_{1n}		x	x	0	0
0		x	x	1	0
1		x	x	1	0

$$\therefore K_0 = Q_{1n}X$$

For output Y,

	$Q_{0n}X$	00	01	11	10
Q_{1n}		0	1	0	1
0		0	0	1	0
1		0	0	0	1

$$\therefore Y = \bar{Q}_{1n}\bar{Q}_{0n}X + \bar{Q}_{1n}Q_{0n} + Q_{0n}\bar{X}$$

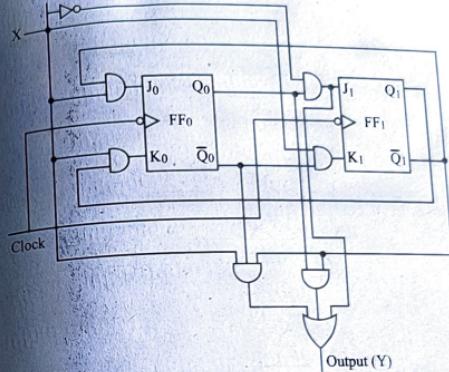
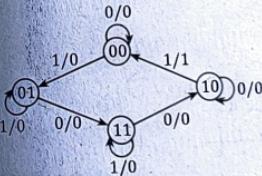


Fig.: Circuit diagram using JK flipflop

12. Design a sequential circuit corresponding to the given state diagram using SR flipflop.
[Fall 2014]



Solution:

Excitation table of SR flipflop

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

Synthesis table

Present state	Present input (X)	Next state		Output Y	FF excitation			
		Q_{1n}	Q_{0n}		S_1	R_1	S_0	R_0
0 0	0	0	0	0	0	x	0	x
0 0	1	0	1	0	0	x	0	x
0 1	0	1	1	0	1	0	x	0
0 1	1	0	1	0	0	x	x	0
1 0	0	1	0	0	x	0	0	x
1 0	1	0	0	1	0	1	0	x
1 1	0	1	0	0	x	0	0	1
1 1	1	1	1	0	x	0	x	0

Using K-map,

For S_1 ,

$Q_{0n}X$	00	01	11	10
Q_{1n}	0	0	0	0
	0	x	0	x
1	x	0	x	x

$$S_1 = Q_{0n}\bar{X}$$

For R_1 ,

$Q_{0n}X$	00	01	11	10
Q_{1n}	0	x	x	0
	1	0	1	0

$$R_1 = \bar{Q}_{0n}X$$

For S_0 ,

$Q_{0n}X$	00	01	11	10
Q_{1n}	0	0	1	x
	1	0	0	x

$$S_0 = \bar{Q}_{1n}X$$

For R_0 ,

$Q_{0n}X$	00	01	11	10
Q_{1n}	0	x	0	0
	1	x	x	1

$$R_0 = Q_{1n}\bar{X}$$

For output Y,

$Q_{0n}X$	00	01	11	10
Q_{1n}	0	0	0	0
	1	0	①	0

$$Y = Q_{1n}\bar{Q}_{0n}X$$

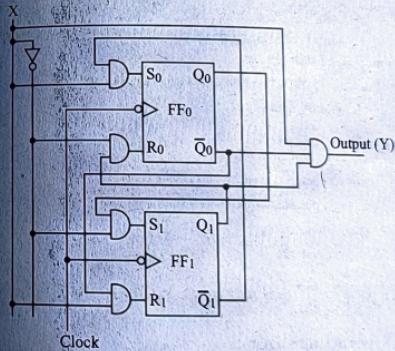


Fig.: Detailed circuit diagram using SR flipflop