14. Tabulate the PLA programmable table for the four Boolean functions listed below.

$$A(x,y,z) = \sum m(0,1,2,4,6)$$
$$B(x,y,z) = \sum m(0,2,6,7)$$
$$C(x,y,z) = \sum m(3,6)$$
$$D(x,y,z) = \sum m(1,3,5,7)$$

15. Tabulate the PLA programming table for the four Boolean functions listed below. Minimize the number of product terms.

$$A(x,y,z) = \sum(1,2,4,6)$$
$$B(x,y,z) = \sum(0,1,6,7)$$
$$C(x,y,z) = \sum(2,6)$$
$$D(x,y,z) = \sum(1,2,3,5,7)$$

16. Derive the PLA programming table for the combinational circuit that squares a 3-bit number. Minimize the number of product terms.
17. List the PLA programming table for the BCD to excess 3 code converter whose Boolean functions are simplified.
18. List the PAL programming table for the BCD to excess 3 code converter whose Boolean functions are simplified.
19. The following is a truth table of a 3-input, 4-output combinational circuit. Tabulate the PAL programming table for the circuit and mark the fuse map in a PAL diagram.

Inputs			Outputs			
x	y	z	A	B	C	D
0	0	0	0	1	0	0
0	0	1	1	1	1	1
0	1	0	1	0	1	1
0	1	1	0	1	0	1
1	0	0	1	0	1	0
1	0	1	0	0	0	1
1	1	0	1	1	1	0
1	1	1	0	1	1	1

Chapter 6

Synchronous Sequential Logic Circuit

6.1 Introduction

A block diagram of a sequential circuit is shown in Fig.6.1. It 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. This binary information stored in these elements at any given time define the state of the sequential circuit at that time.

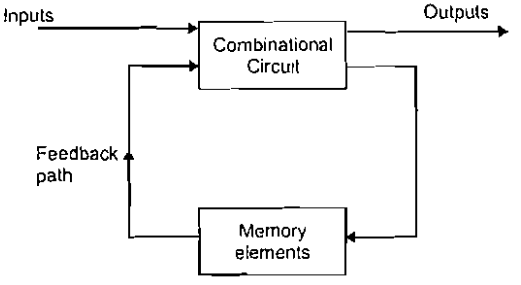


Fig 6.1 Block diagram of sequential circuits

The sequential circuit receives the binary information from external inputs. These inputs and the present state of memory elements determine the binary value of the outputs of the circuit. They also determine the condition for changing the state in memory elements. Thus, the next state of the memory elements is also a function of the external inputs and the present state.

6.1.1 Mealy Model Sequential Circuit

Fig 6.2 shows the clocked synchronous sequential Mealy machine. The output of mealy machine is the function of present inputs and present state (Flip flop outputs). If X is input, Q_n is the present state and the next state is $Q_{(n+1)}$, the output of Mealy function (Z) is given below.

$$Z = f(X, Q_n)$$

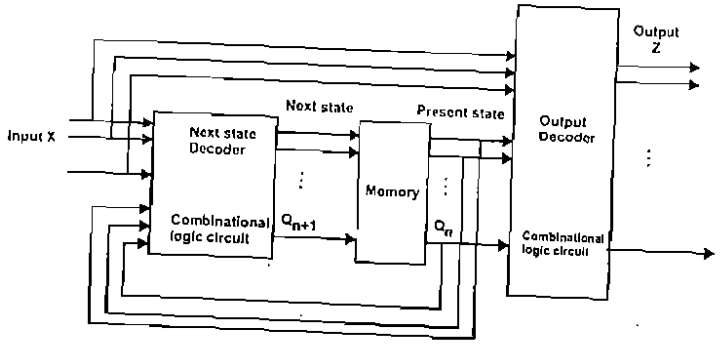


Fig 6.2 Mealy model sequential circuit

The output of memory element is conneeted to the input of output deeoder and next state decoder circuit. The output of memory element is considered as present state.

6.1.2 Moore Model Sequential Circuit

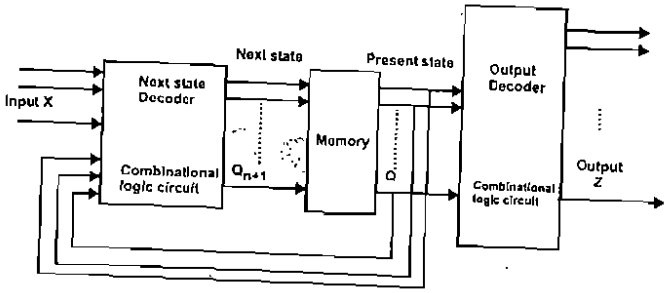


Fig 6.3 Moore model sequential circuit

Fig.6.3 shows the block diagram of a Moore machine. The output of Moore machine depends only on the present state. So the output of Moore machine is a function of its present state (Q_n). If the input is X, the next state is $Q_{(n+1)}$ and the present state is Q_n. The output of Moore machine is represented mathematically by

$$Z = f(Q_n)$$

The differences between the Moore machine and Mealy machine are tabulated as follows

S.No.	Moore machine	Mealy machine
1.	The output of this machine is the function of the present state only.	Its output is function of present input as well as present state
2.	Input changes do not affect the output	Input changes may affect the output of the circuit
3.	It requires more number of states for implementing same function	It requires less number of states for implementing same function

6.2 Analysis and Synthesis of Synchronous Sequential Circuits

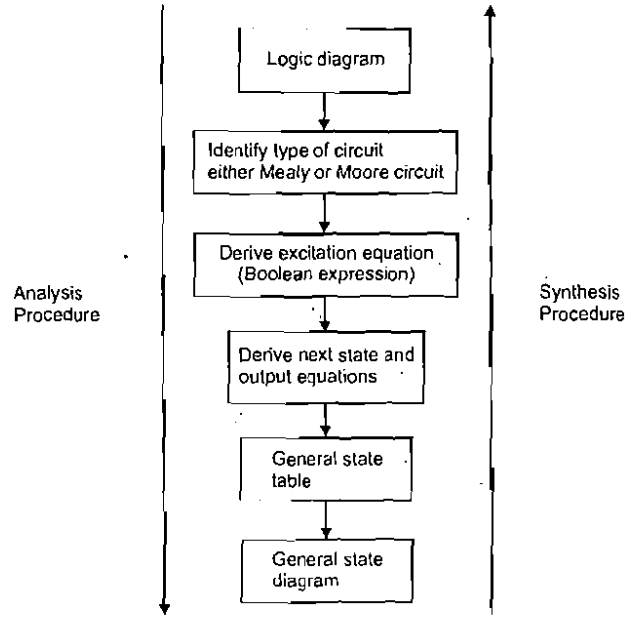


Fig 6.4 Flow chart

The behaviour of sequential circuit can be determined from the inputs, outputs and state of its flip flops. The outputs and next state are both a function of its inputs and the present state. The analysis of a sequential circuit consists of obtaining a state table or state diagram for the time sequence of inputs, outputs and internal states. The analysis of the clocked sequential circuits can be done by following the procedure as shown in Fig.6.4. The reverse process of analysis is known as synthesis of clocked sequential logic circuit.

For the analysis of sequential circuit, we start with the logic diagram. The excitation equation or Boolean expression of each flip-flop is derived from this logic diagram. Then, to obtain the next state equation, we insert the excitation equations into the characteristic equations. The output equations can be derived from the schematic. We can generate the state table using output and next state equations.

6.2.1 Analysis of Example Sequential Logic Circuit

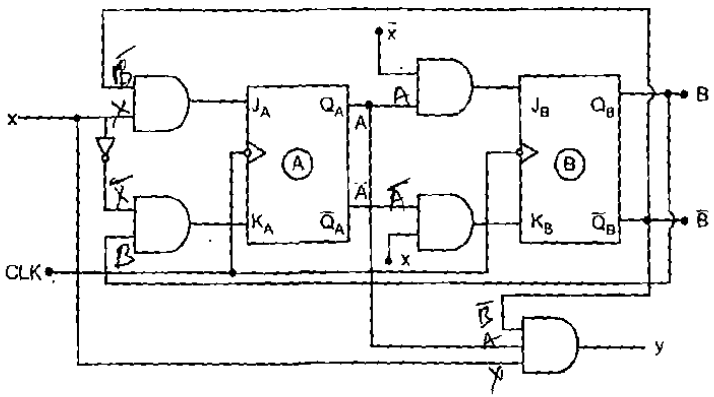


Fig 6.5 Example of sequential logic circuit

Fig. 6.5 shows a clocked sequential circuit. It has one input variable x, one output variable y and two clocked JK flip flops. The flip flops are labelled as A and B and their outputs are labelled as A and $\bar{A}$. B and $\bar{B}$ respectively.

Step 1 : Type of circuit

The output(y) of given logic circuit (Fig.6.5) depends on present input and also on present state (Flip flop outputs) of flip flops, so that the given sequential logic circuit is Mealy sequential machine.

Step 2 : Excitation equations

The excitation equations or Boolean expressions of flip flops A and B are obtained. The equations will be in the form of present states A and B and external

input x, since here are two JK flip flops which have output A and B. Therefore the excitation equation (equation formed for flip flop input)

For Flip flop - A $\quad J_A = x\bar{B}$, $\quad K_A = \bar{x}B$

For flip flop - B $\quad J_B = \bar{x}A$, $\quad K_B = x\bar{A}$

Step 3 : Next state equations The state equations can be derived directly from the logic diagram. Looking at Fig.6.5 we can see that the signal for J input of the flip flop A is generated by the function $\bar{B}x$ and the signal for input K by the function $\bar{B}x$. Substituting $J = \bar{B}x$ and $K = B\bar{x}$ into a JK flip flop characteristic equation given by

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

State equation for flip flop A

$$\begin{aligned} A_{n+1} &= (\bar{B}x)\bar{Q}_n + (B\bar{x})Q_n \qquad \text{where } Q_n = A \\ &= \bar{B}x\bar{A} + \bar{B}\bar{x}A \\ &= \bar{A}\,\bar{B}x + A(\bar{B}\bar{x}) \\ &= \bar{A}\,\bar{B}x + A(\bar{B} + x) \\ &= \bar{A}\,\bar{B}x + A\bar{B} + Ax \\ &= \overline{\bar{A}\,\bar{B}} + x(A + \overline{A\bar{B}}) \\ &= AB + x(A + \bar{B}) \qquad \because \ (A + \bar{A}\,\bar{B} = A + \bar{B}) \\ A_{n+1} &= A\bar{B} + Ax + \bar{B}x \end{aligned}$$

State equation for flip flop B

Similarly, we can find the state equation for flip flop B. $J = \bar{A}x$ and $K = \bar{A}x$.

Therefore the state equation of flip flop B is given as

$$\begin{aligned} B_{n+1} &= A\bar{x}\,\bar{B} + (\overline{\bar{A}\,x})B \\ &= A\bar{x}\bar{B} + (A + \bar{x})B \\ &= A\bar{x}\bar{B} + AB + B\bar{x} \\ &= \bar{x}(A\bar{B} + B) + AB \\ &= \bar{x}(A + B) + AB \\ B_{n+1} &= A\bar{x} + B\bar{x} + AB \end{aligned}$$

Output equation

The given sequential circuit has output y. The output equation can be found from the Fig.6.5 which is derived using three input AND gate

$$y = A\bar{B}x$$

Step 4: State table

Table 6.1 is the state table for the given sequential logic circuit. It represents the relationship between input, output and flip flop states. It consists of three columns: present state, next state and output

Present state: It specifies the state of the flip flop before occurrence of a clock pulse.

Next state: It is the state of flip flop after the application of a clock pulse.

Output: This section gives the value of the output variables during the present state. Both next state and output section have two columns representing two possible input conditions $x = 0$ and $x = 1$.

Table 6.1

Present state	Next state		Output	
	AB	AB	y	
AB	$x=0$	$x=1$	$x=0$	$x=1$
00	00	10	0	0
01	01	00	0	0
10	11	10	0	1
11	01	11	0	1

We can derive the state table as follows

(i) If present state $AB = 00, x = 0$

When a present state is 00 i.e. $A = 0$ and $B = 0$ and input $x = 0$, the next state is obtained by using next state equation

Next state for flip flop A

$$\begin{aligned} A_{n+1} &= A\bar{B} + Ax + \bar{B}x \\ &= 01 + 0.0 + 1.0 \\ &= 0 \end{aligned}$$

Next state for flip flop B

$$\begin{aligned} B_{n+1} &= A\bar{x} + B\bar{x} + AB \\ &= 0.1 + 0.1 + 0.0 \\ &= 0 \end{aligned}$$

Next state for this case $AB = 00$

(ii) If present state $AB = 00, x = 1$

Next state for flip-flop A

$$\begin{aligned} A_{n+1} &= A\bar{B} + Ax + \bar{B}x \\ &= 0.1 + 0.1 + 1.1 \\ &= 1 \end{aligned}$$

Next state flip flop B

$$\begin{aligned} B_{n+1} &= A\bar{x} + B\bar{x} + AB \\ &= 0.0 + 0.0 + 0.0 \\ &= 0 \end{aligned}$$

Next state for this case $AB = 10$

Similarly we can obtain next state for all these different cases as shown in the table.

(iii) Determine the entries in the output section. For this, we have to examine AND gate for all possible present states and input.

(a) If a present state $AB = 00, \quad x = 0$

$$\begin{aligned} \text{output } y &= A\bar{B}x \\ &= 0.10 \end{aligned}$$

$$y = 0$$

(b) If a present state $AB = 00, \quad x = 1$

$$\begin{aligned} \text{output } y &= A\bar{B}x \\ &= 0.11 \end{aligned}$$

$$y = 0$$

Thus, the state table of any sequential circuit can be obtained by the same procedure used in the above example. This example contains 2 flip flops and one input, and one output, producing four rows, two columns in the next state and output sections. In general, a sequential circuit with m flip-flops and n-input variables produces 2^m rows and one for each state and 2^n columns, one for each input combination in the next state and output sections of the state table.

Step 5 State diagram

State diagram is a graphical representation of a state table. Fig.6.6 shows the state diagram for sequential circuit. Here each state is represented by a circle, and transition between states is indicated by directed lines connecting the circles. The binary number inside each circle identifies the state represented by the circle. The directed lines are labelled with two binary numbers separated by a symbol '/' (slash). The input value that causes the state transition is labelled first and output value is next.

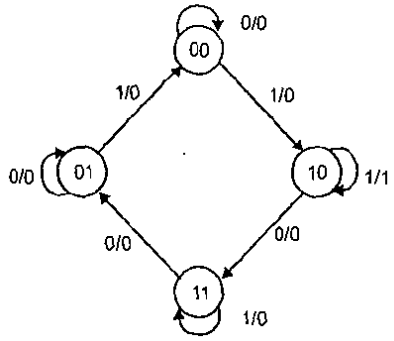


Fig 6.6 State diagram of Fig.6.5

Example 6.1 Derive the state table and state diagram for the sequential circuit shown in Fig.6.7(a).

□ **Solution**

Step 1: Type of circuit

The output y of given sequential circuit (Fig.6.7) depends on the present input and also present state (flip flop output) of flip flops, so the given sequential logic circuit is Mealy sequential machine.

Step 2: Excitation equation

For flip flop A

$$J_A = x + B$$
$$K_A = A$$

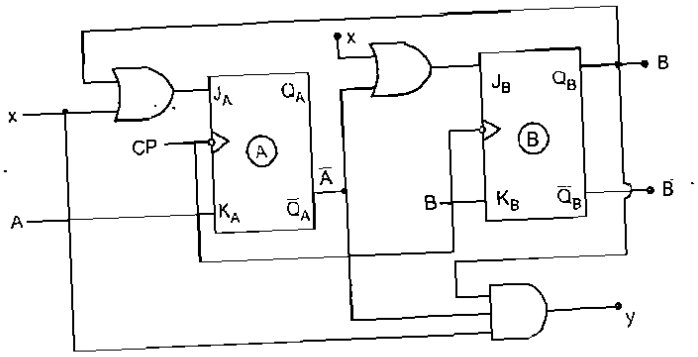


Fig 6.7

For flip flop B

$$J_B = \overline{A} + x$$
$$K_B = B$$

Step 3

We know that characteristic equation of JK flip flop

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

State equation for flip flop A

$$A_{n+1} = (x + B)\overline{Q}_n + AQ_n \qquad \text{(where } Q_n = A \text{ for flip flop } A\text{)}$$
$$= (x + B)\overline{A} + \overline{A}A$$
$$= \overline{A}x + \overline{A}B + 0$$
$$A_{n+1} = \overline{A}x + \overline{A}B$$

State equation for flip flop B

$$B_{n+1} = (\overline{A} + x)\overline{Q}_n + \overline{B}Q_n \qquad \text{(where } Q_n = B \text{ for flip flop } B\text{)}$$
$$= (\overline{A} + x)\overline{B} + \overline{B}B$$
$$B_{n+1} = \overline{A}\,\overline{B} + \overline{B}x$$

Output equation

$$y = \overline{A}Bx$$

Step 4 : State table

Present state	Next state		Output	
	AB	$\overline{AB}$	y	
AB	$x=0$	$x=1$	$x=0$	$x=1$
00	00	11	0	0
01	10	10	0	1
10	01	00	0	0
11	00	00	0	0

Step 5 : State diagram

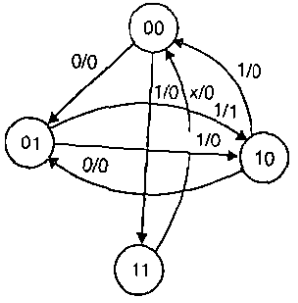


Fig 6.8

Example 6.2 Derive the state table and state diagram for the sequential circuit shown in Fig 6.9

□ **Solution**

Step 1 : Type of circuit

The output of a given circuit(See Fig.6.9) depends on present input and also on present states. so the given sequential logic circuit is Mealy machines

Step 2: Excitation Equation

For flip flop A $\quad D_A = Ax + B\bar{x}$

For Flip flip B $\quad D_B = \bar{A}x$

For Output $\quad y = AB + \bar{x}$

Step 3:

We know that characteristic equation of D flip flop (next state depends on input D)

$$A_{n+1} = D_A$$

$$\boxed{A_{n+1} = Ax + B\bar{x}}$$

$$B_{n+1} = D_B$$

$$\boxed{B_{n+1} = \bar{A}x}$$

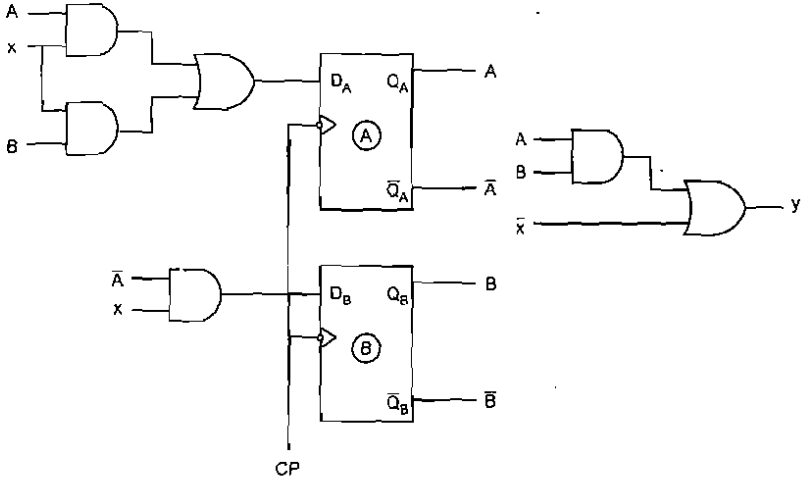


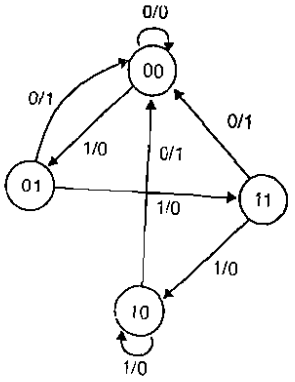
Fig 6.9

Step 4: State table

The state table contains four rows and three columns. The next state and output have two sub columns.

Present state	Next state		Output	
	AB	$\overline{AB}$	y	
AB	$x=0$	$x=1$	$x=0$	$x=1$
00	00	01	0	0
01	00	11	1	0
10	00	10	1	0
11	00	10	1	0

Step 5: State Diagram



Example 6.3 Derive the state table and state diagram for sequential circuit shown in Fig.6.10

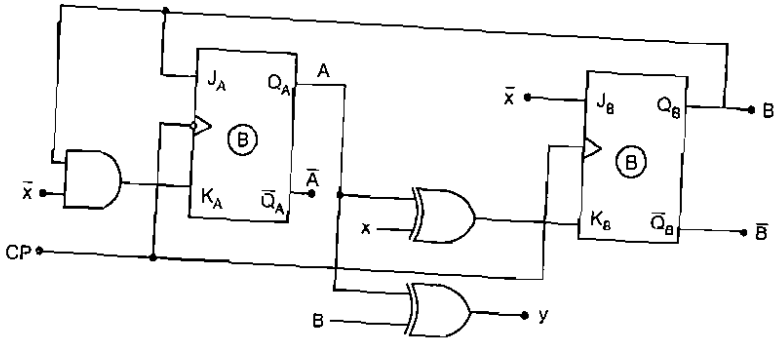


Fig 6.10

Solution

***Step 1* : Type of circuit**

The output y of the sequential circuit depends on present state only, so the given logic circuit is the Moore type circuit.

Step 2: Excitation equations

For flip flop A

$$J_A = B$$

$$K_A = B\bar{x}$$

For flip flop B

$$J_B = \bar{x}$$

$$K_B = A \oplus x$$

$$y = A \oplus B$$

Step 3 :

We know that characteristics equation of JK flip flop

$$A_{n+1} = J\overline{Q}_n + \overline{K}Q_n$$

State equation for flip flop A : $A_{n+1} = B\overline{A} + (\overline{B\bar{x}})\,A$ $\quad (\because \quad Q_n = A)$

$$= B\overline{A} + A(\overline{B} + x)$$

$$= B\overline{A} + A\overline{B} + x$$

$$A_{n+1} = (A \oplus B) + x$$

State equation for flip flop B : $B_{n+1} = \bar{x}\overline{B} + (\overline{A \oplus x})B$

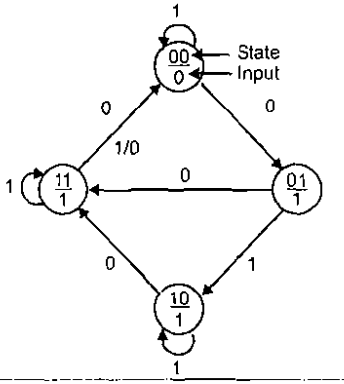
$$= \bar{x}\,\overline{B} + (Ax + \overline{A}\,\bar{x})B$$

$$B_{n+1} = \bar{x}\overline{B} + AxB + \overline{A}\bar{x}B$$

Step 4: State table

Present state	Next state		Output
	$x = 0$	$x = 1$	y
AB	AB	AB	
00	01	00	0
01	11	10	1
10	11	10	1
11	00	11	0

State diagram



Note: The state diagram for Moore machine is different from Mealy machine. Here each circle is coded with state binary number/output.

6.3 State Reduction

Any logic design process *must* consider the problem of minimizing the cost of the final circuit. One way to reduce the cost is by reducing the number of flip flops, *i.e.* by reducing the number of states. The state reduction technique basically avoids the introduction of redundant equivalent states. The reduction of redundant states reduces the number of flip flops and logic gates required, thus reducing the cost of the final circuit. Two states are said to be redundant or equivalent, if every possible set of inputs generate exactly the same outputs and the same next states. When two states are equivalent, one of them can be removed without altering input-output relationship. Let us consider the state diagram shown in Fig 6.11. The states are denoted by letter symbols instead of their binary values because in state reduction technique internal states are also important, but input output sequences are more important. The procedure contains two steps.

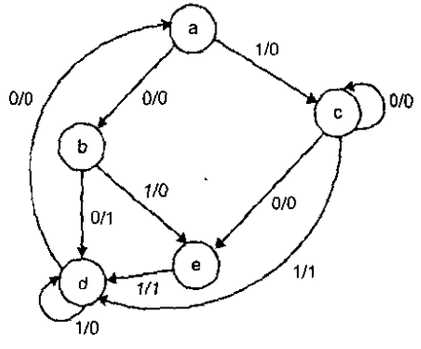


Fig 6.11

Step 1: Finding the state table for the given state diagram

First the given state diagram is converted into a state table. Fig.6.11 shows the example of state diagram.

Present state	Next state		Output	
	x=0	x=1	x=0	x=1
a	b	c	0	0
b	d	e	1	0
c	c	d	0	1
d	a	d	0	0
e	c	d	0	1

Both are equivalenl slates because of state c and e having same next state and same output

Step 2: Finding equivalent states

The two present states go to the same next state and have the same output for both the input combinations. We can easily find this from the state table. states

c and e are equivalent. This is because both c and e states go to states c and d outputs of 0 and 1 for $x = 0$, $x = 1$ respectively. Therefore, the state e can be removed and replaced by c. The final reduced table and state diagram are given in the table 6.2 and Fig.6.12. The second row have e state for the input $x = 1$, it replaced by c because the states c and e are equivalent.

Table 6.2 Reduced state table

Present state	Next state		Output y	
	AB	AB		
AB	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	b	c	0	0
b	d	c	1	0
c	c	d	0	1
d	a	d	0	0

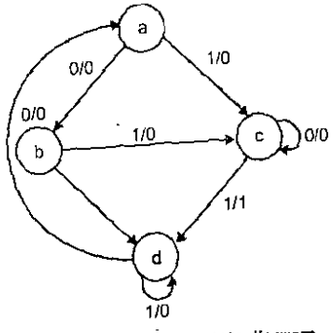


Fig. 6.12 Reduced state diagram

Example 6.4 Obtain the reduced state table and reduced state diagram for a sequential circuit whose state diagram is shown in Fig.6.13.

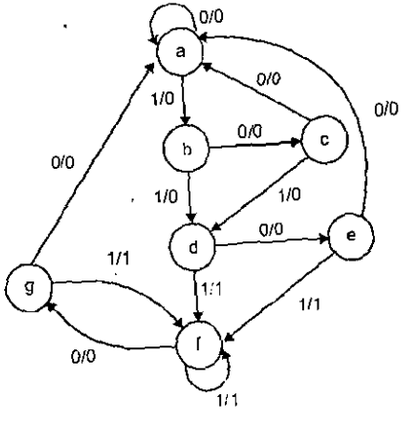


Fig 6.13

☐ Solution

The given diagram has seven states, one input and one output. As per the step 1, the given state diagram is converted to a state table.

State table

Table 6.4(a)

Present state	Next state		Output	
	x=0	*x*=1	*x*=0	*x*=1
a	*a*	*b*	0	0
b	*c*	*d*	0	0
c	*a*	*d*	0	0
d	*e*	*f*	0	1
e	*a*	*f*	0	1
f	*g*	*f*	0	1
g	*a*	*f*	0	1

Both are equivalent states because of state *e* and *g* having same next state and same output

From the above state table, it is clear that states *e* and *g* are equivalent. So the state *g* is replaced by state *e*. The reduced state table is shown in Ex.6.4

Reduced state Table

Table 6.4(b)

Present state	Next state		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
a	*a*	*b*	0	0
b	*c*	*d*	0	0
c	*a*	*d*	0	0
d	*e*	*f*	0	1
e	*a*	*f*	0	1
f	*e*	*f*	0	1

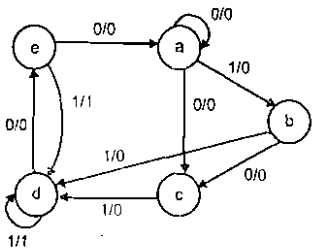
Both are equivalent states

From the above reduced table, states *d* and *f* are equivalent, hence '*f*' can be replaced by *d* and it can be removed. Then finally the reduced state table is shown in Table 6.4(c)

Final reduced table

The state diagram of the reduced state Table is shown in Fig.6.4(b).

Present state	Next state		Output	
	AB	*AB*		
AB	$x=0$	$x=1$	$x=0$	$x=1$
a	*a*	*b*	0	0
b	*c*	*d*	0	0
c	*a*	*d*	0	0
d	*e*	*d*	0	1
e	*a*	*d*	0	1



6.4 State Assignment

In sequential circuits we know that the behaviour of the circuit is defined in terms of its inputs, present state, next state and outputs. To generate the desired next state at particular present state and inputs, it is necessary to have specific flip flop inputs. These flip flop inputs are described by a set of Boolean functions called flip flop input functions. To determine the flip flop input functions, it is necessary to represent states in the state diagram using binary values instead of alphabets. This procedure is known as state assignment. The following rules are used in state assignment.

Rule 1. States having the same next states for a given input condition should have assignments which can be grouped into logically adjacent cells in a K-map. (Fig.6.15)

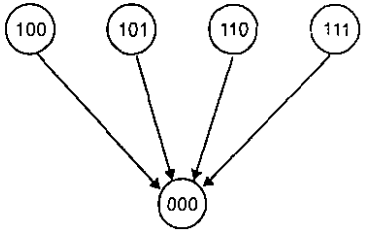


Fig.6.15

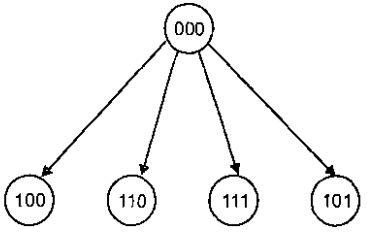


Fig.6.16

Rule 2. States having different next states should have assignment which can be grouped into logically adjacent cells in K-map. (Fig.6.16)

Example 6.5 Design a sequential circuit using D flip flop for a state diagram given below. Use state assignment rules for assigning states and compare the required combinational circuit with random state

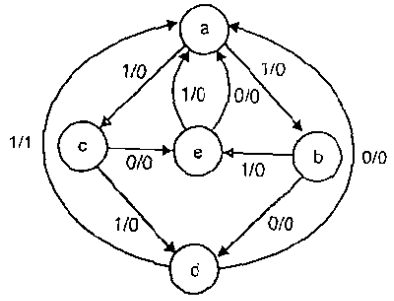


Fig 6.17

☐ Solution

The states are a, b, c, d and e. Each state is randomly assigned.

$a = 000$, $b = 001$, $c = 010$, $d = 011$, $e = 100$. The remaining combinations are considered as don't care conditions.

Excitation table

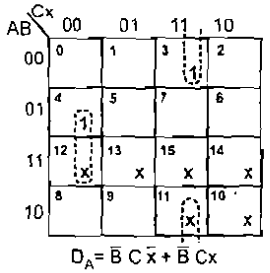
Present state			Input	Next State			Output
A	B	C	X	A_{n+1}	B_{n+1}	C_{n+1}	Z
0	0	0	0	0	0	1	0
0	0	0	1	0	1	0	0
0	0	1	0	0	1	1	0
0	0	1	1	1	0	0	0
0	1	0	0	1	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	0	0	0
0	1	1	1	0	0	0	1
1	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0

Present state			Input	Next State			Output
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

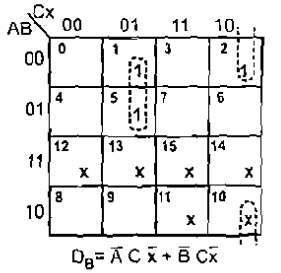
K-map simplification

The D flip flop input is equal to next state and the flip flop expression is obtained directly.

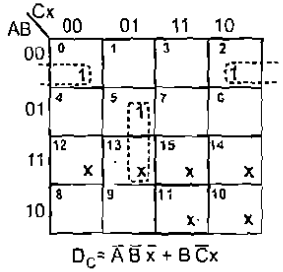
Expression for flip flop input D_A



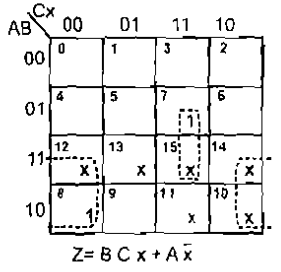
Expression for flip flop input D_B



Expression for flip flop input D_C



Expression for flip flop input D_D



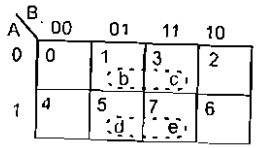
The random assignment requires

7 three input AND gates
1 two input AND gates
4 two input OR gates

Total 12 gates with 31 inputs and 3 flip flops are required to construct the sequential logic circuit. Now we apply state assignment rules, then follow the above steps.

From Rule 1, The states e and d must be adjacent

From Rule 2, states *b* and *c* must be adjacent. We form the adjacent cells in the 3 variable K-map

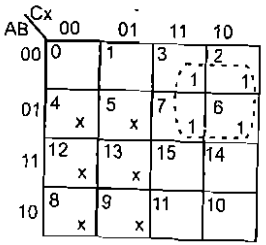


Excitation table

Present state			Input	Next State			Output
A	B	C	X	A_{n+1}	B_{n+1}	C_{n+1}	Z
0	0	0	0	0	0	1	0
0	0	0	1	0	1	1	0
0	0	1	0	1	0	1	0
0	0	1	1	1	1	1	0
0	1	0	0	X	X	X	X
0	1	0	1	X	X	X	X
0	1	1	0	1	1	1	0
0	1	1	1	1	0	1	0
1	0	0	0	X	X	X	X
1	0	0	1	X	X	X	X
1	0	1	0	0	0	0	0
1	0	1	1	0	0	0	1
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	0

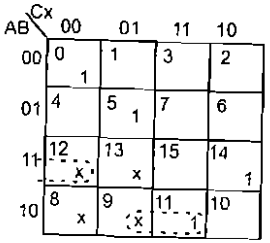
***K*- Map simplification**

Expression for flip flop input D_A



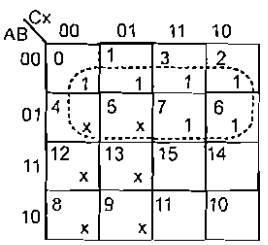
$D_A = \bar{A}C$

Expression for flip flop input D_B



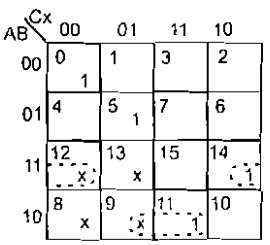
$D_B = A\bar{B}\,x + \bar{A}\,B\,\bar{x}$

Expression for flip flop input D_C



$D_C = \bar{A}$

Expression for flip flop input D_D



$D_D = AB\,x + A\,B\,x$

Under the state assignment rules, we require

4 three input AND gates

1 two input AND gate

2 two input OR gates

A total of 7 gates with 18 inputs and 3 flip flops are required to construct the sequential logic circuit based on the state assignment rules.

6.5 Design Procedure

The following steps are followed to design the clocked sequential logic circuit.

1. Obtain the state table from the given circuit information such as a state diagram, a timing diagram or description.
2. The number of states may be reduced by state reduction technique.
3. Assign binary values to each state in the state table.
4. Determine the number of flip flops required and assign a letter symbol to each flip flop.
5. Choose the flip flop type to be used according to the application.
6. Derive the excitation table from the reduced state table.
7. Derive the expression for flip flop inputs and outputs using k-map simplification (The present state and inputs are considered for k-map simplification) and draw logic circuit using flip flops and gates

6.6 Synthesis of Clocked Sequential Logic Circuits

Synthesis means that, it is the reverse process of analysing a sequential logic circuit. In this synthesis, we get a logic circuit from the information of state diagram, word description etc. The detailed steps are given in the example. Now we will see the detailed description of each step.

A state diagram is obtained from the word description, timing diagram or other pertinent information. From this state diagram, we can form a state table.

The reduction of number of states and binary value assignment to each state gives the reduction in combinational circuit requirement. The number of flip flops required to design any sequential logic circuit depends on the number of states.

Example 6.6 A sequential circuit has one input and one output and its state diagram is shown in Fig.6.18(a).Design the sequential circuit using D flip flop

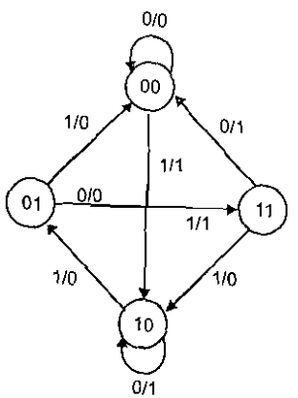


Fig 6.18(a)

☐ Solution

The given state diagram consists of four states. It hat one input (x) and one output (y). The state table for the given state diagram is shown in Table.6.6(a). It is clear that there are no equivalent states. Therefore, there is no reduction in the state diagram. As the state diagram contains 4-states, it requires 2 flip-flops which are named as A and B.

Table 6.6 (a)

Present state	Next state		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
AB	$A\bar{B}$	$A\bar{B}$	y	y
00	00	10	0	1
01	11	00	0	0
10	10	01	1	0
11	00	10	1	0

Design using *D*-flip flop

For the design of circuit using D flip flop (or any flip flop), we need the excitation table. Table Ex 6.6(b) shows the excitation table of D flip flop from which we can develop excitation table for the required circuit as shown in table 6.6(c).

Table 6.6 (b) Excitation table for D-flip flop

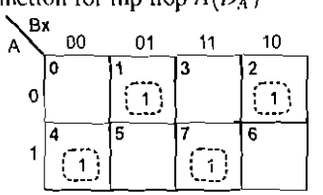
Present state	Next state	Flip flop input
Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Excitation table

Present state		Input	Next state		Flip flop input		Output
A	B	x	A	B	D_A	D_B	y
0	0	0	0	0	0	0	0
0	0	1	1	0	1	0	1
0	1	0	1	1	1	1	0
0	1	1	0	0	0	0	0
1	0	0	1	0	1	0	1
1	0	1	0	1	0	1	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

The flip-flop input function and the circuit output function are obtained by using K-map simplification.

Input equation (or) function for flip flop $A(D_A)$



$$D_A = \bar{A}\,\bar{B}\,x + \bar{A}\,B\bar{x} + A\bar{B}\,\bar{x} + ABx$$

$$= \bar{A}(\bar{B}x + B\bar{x}) + A(\bar{B}\,\bar{x} + Bx)$$

Let us consider $z = \bar{B}x + B\bar{x}$. then $\bar{B}\,\bar{x} + Bx = \bar{z}$. Simplify the above equation

$$DA = \bar{A}z + A\bar{z}$$

$$= A \oplus z$$

Substitute $z = \bar{B}x + B\bar{x} = B \oplus x$ in the above equation

$$\boxed{D_A = A \oplus B \oplus x}$$

Input equation for flip flop $B(D_B)$

A \ Bx	00	01	11	10
0	0	1	3	2 1
1	4	5 1	7	6

$$D_B = \bar{A}B\bar{x} + A\bar{B}x$$

Output function y

A \ Bx	00	01	11	10
0	0	1 1	3	2 1
1	4 1	5	7	6 1

$$y = A\bar{x} + \bar{A}\bar{B}x$$

The input equation for flip and output equation are simulated as follows

$$DA = A \oplus B \oplus x$$

$$DB = \bar{A}\bar{B}x + A\bar{B}x$$

$$y = A\bar{x} + \bar{A}\bar{B}x$$

A sequential circuit using D flip flop is obtained by using the above equations as shown in fig 6.18(b).

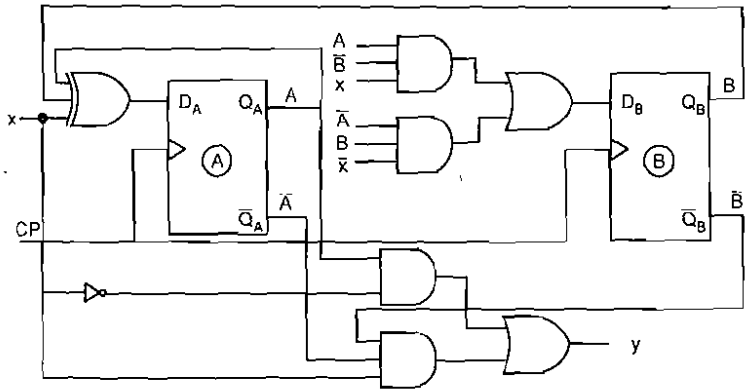


Fig 6.18(b)

6.7 Sequence Generator

A sequential circuit which generates a prescribed sequence of bits, synchronous with the clock, is referred to as a sequence generator. We can construct sequence generators by two ways

1. Sequence generators using counters
2. Sequence generators using shift registers

6.7.1 Sequence Generator using Counters

Fig.6.19 shows the block diagram of a sequence generator using counters. It contains two stages

1. counter, and
2. next state decoder.

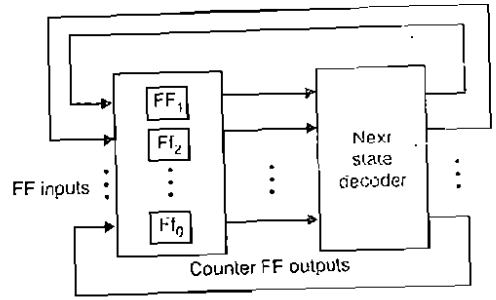


Fig 6.19

Design Procedure

Step 1 : Determine the number of flip-flops required

The number of flip-flops required to generate a particular sequence can be determined as follows.

(a) Find the number of 1's in the sequence.

(b) Find the number of 0's in the sequence.

(c) Take the maximum value from both. If 'n' is the required number of flip-flops, choose minimum value of 'n' to satisfy the following condition.

$$\max(0's, 1's) \leq 2^{n-1}$$

Step 2 : State assignment

Once the number of flip-flops is decided, we have to assign unique states corresponding to each bit in the given sequence such that the flip-flop representing least significant bit generates the given sequence (the output of the flip-flop which represents the least significant bit is used to represent the given sequence)

Step 3 : Draw the state diagram from the above state assignment and obtain the excitation table from the state diagram.

Step 4 : Find the Boolean expression for each flip-flop input by using k-map and draw the logic diagram for this Boolean expression

Example 6.7 Find the number of flip-flops required to generate the sequence 10110110.

Solution

In the given sequence, the number of 0's are 3 and number of 1's are 5.

$$\max(3,5) \leq 2^{n-1}$$

$$5 \leq 2^{n-1}$$

$$\boxed{n = 4}$$

Example 6.8 Design a sequence generator using *JK* flip-flop to generate the sequence 1101011.

Solution

Step 1: Number of flip-flops required

Number of 0's in the sequence = 2

Number of 1's in the sequence = 5

Hence $\max(2,5) \leq 2^{n-1}$

$$5 \leq 2^{n-1}$$

$$\boxed{n = 4}$$

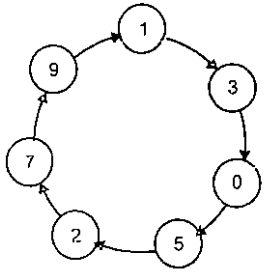
We need four flip-flops named as A, B, C and D. The desired sequence is generated by the D flip-flops output

Step 2: State assignment

Decimal equivalent	*A*	*B*	*C*	*D*	
1	0	0	0	1	← Given sequence, first enter this column
3	0	0	1	1	
0	0	0	0	0	
5	0	1	0	1	
2	0	0	1	0	
7	0	1	1	1	
9	1	0	0	1	

Assign binary value based on non-repeated states

Step 3: State diagram

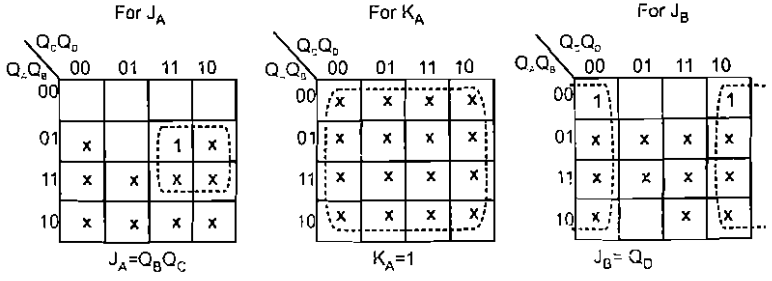


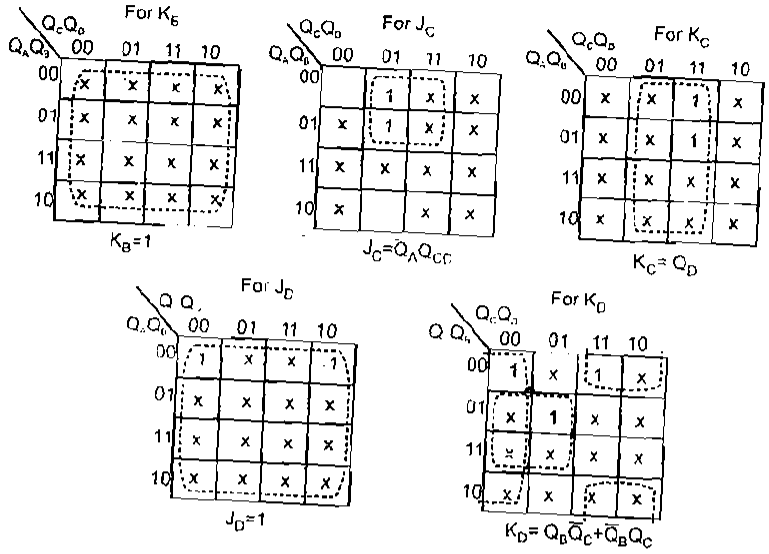
Excitation table

Present state				Next state				Flip-flop inputs							
Q_A	Q_B	Q_C	Q_D	Q_A	Q_B	Q_C	Q_D	J_A	K_A	J_B	K_B	J_C	K_C	J_D	K_D
0	0	0	0	0	1	0	1	0	X	1	X	X	1	1	X
0	0	0	1	0	0	1	1	0	X	0	X	1	X	X	0
0	0	1	0	0	1	1	1	0	X	1	X	X	0	1	X
0	0	1	1	0	0	0	0	0	X	0	X	X	1	X	1
0	1	0	0	X	X	X	X	X	X	X	X	X	X	X	X
0	1	0	1	0	0	1	0	0	X	X	1	1	X	X	1
0	1	1	0	X	X	X	X	X	X	X	X	X	X	X	X
0	1	1	1	1	0	0	1	1	X	X	1	X	1	X	0
1	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X
1	0	0	1	0	0	0	1	X	1	0	X	0	X	X	0
1	0	1	0	X	X	X	X	X	X	X	X	X	X	X	X
1	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X
1	1	0	0	X	X	X	X	X	X	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X	X	X	X	X	X	X
1	1	1	0	X	X	X	X	X	X	X	X	X	X	X	X
1	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X

Note : The unused states 4, 6, 8, 10, 11, 12, 13, 14, and 15 are considered as *X*

***K*-map simplification**





Logic diagram

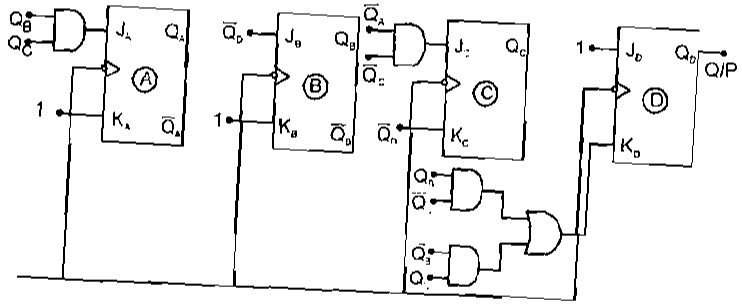
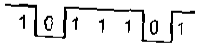


Fig 6.20

Example 6.9 Design a pulse train generator for the waveform shown below.



Solution

Step 1 : The pulse is repeated for every 4-bit sequence 0111. Therefore the required number of flip flop is determined as follows.

(i) number of 0's = 1
(ii) number of 1's = 3

$$\text{Hence} \quad \max(1,3) \le 2^{n-1}$$
$$3 \le 2^{n-1}$$
$$\boxed{n = 3}$$

We need three flip-flop named as A, B and C. The desired sequence is generated by the C flip-flop.

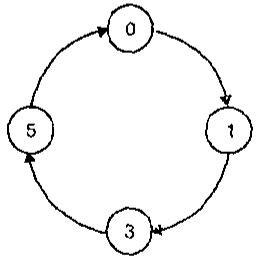
Step 2 : State assignment

Decimal equivalent	A	B	C
0	0	0	0
1	0	0	1
3	0	1	1
5	1	0	1

↑ Given sequence

Note : The unused states are 2,4,6 and 7. Consider the don't care (X) for these states in the K-map simplification.

Step 3 : State diagram

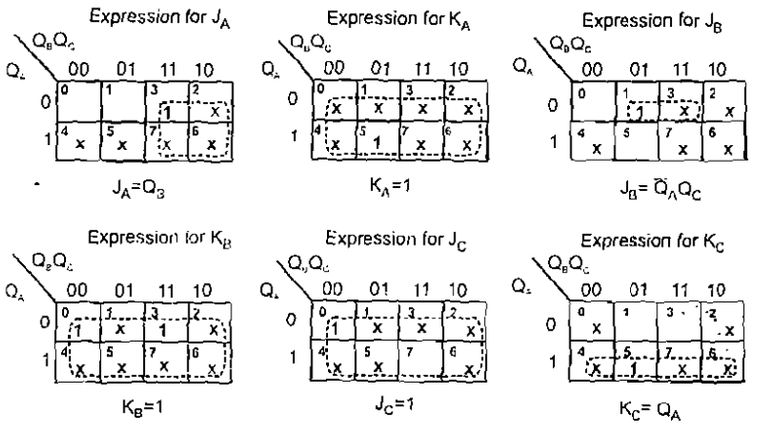


Excitation table

Present state			Next state			Flip-flop inputs					
Q_A	Q_B	Q_C	Q_A	Q_B	Q_C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	1	0	X	1	X	X	0
0	1	1	1	0	1	1	X	X	1	X	0
1	0	1	0	0	0	X	1	0	X	X	1

Note : Unused states 2, 4, 6 and 7 are considered as X

K-map simplification



This minimal expression form is *K*-map simplification

$$J_A = Q_B \qquad J_B = \bar{Q}_A Q_C \qquad J_C = 1$$
$$K_A = 1 \qquad K_B = 1 \qquad K_C = Q_A$$

Logic diagram

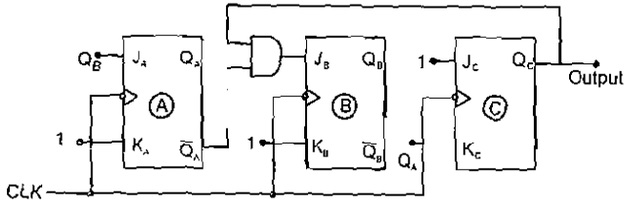


Fig 6.21

6.7.2 Sequence Generator using Shift Registers

This is another method for designing sequence generator. In this method shift registers with next state decoder logic are used. Fig.6.22 shows the block diagram of sequence generator using shift registers.

From this Fig.6.22.we see that the output of next state decoder is a function of Q_A. Q_B. ... Q_n. The next state decoder is a logic circuit which decodes the output of shift register and generates (input to get desired sequence from flip flop *A* (least significant bit).

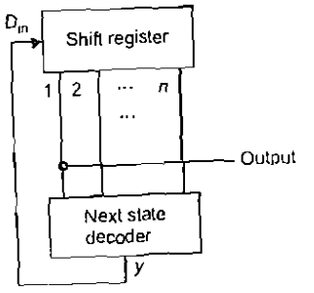


Fig 6.22 Block diagram of a sequence generator using shift registers

Example 6.10 Design a sequence generator to generate the sequence 1101011 by shift register method.

In this approach, the minimum number of flip-flops *n*, required to generate a sequence of length N is given by

$$N \leq 2^{n-1}$$

In this example $N = 7$ and therefore, the minimum value of *n*, which may generate the sequence is

$$7 \leq 2^n - 1$$
$$\boxed{n = 3}$$

With the three flip-flops. the sequence generation is shown in Table.6.7 The state diagram is shown In Fig.6.23

Table 6.7 State assignment

Flip-flop outputs			D_{in}	States
Q_A	Q_B	Q_C		
1	0	0	1	4
1	0	1	0	5
0	0	0	1	0
1	1	0	0	6
0	1	0	1	2
1	1	1	1	7

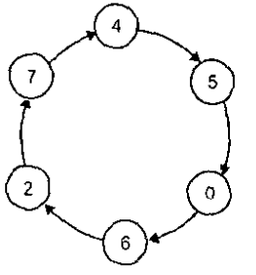
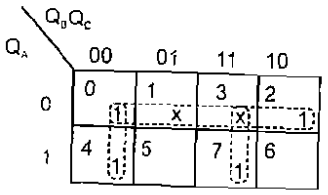


Fig.6.23

***K*-map simplification for D_{in}**



$$D_{in} = \overline{Q}_A + \overline{Q}_B\overline{Q}_C + Q_BQ_C$$
$$= \overline{Q}_A + Q_B \odot Q_C$$

The logic diagram is shown in Fig.6.24.The initial state is 100. So the input for D_A, D_B and D_C are 100 respectively.

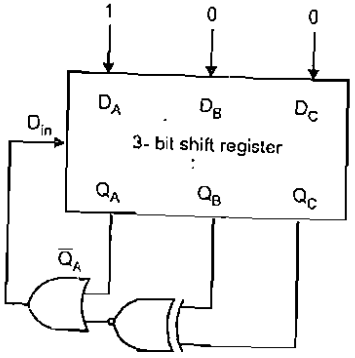


Fig 6.24

6.8 Sequence Detector

A sequence detector is a sequential logic circuit that can be used to detect whether a given sequence of bits has been received or not. We can draw the state diagram when we know the sequence and then follow the steps to design sequential logic circuit to obtain the sequence detector sequential logic circuit.

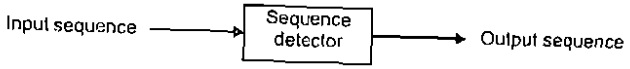


Fig 6.25

Generally sequence detector produces an output = 1, whenever it detects the desired input sequence and '0' for other cases. There are two types of detectors:

1. A detector which detects overlapping input sequence, and
2. A detector which detects non-overlapping input sequence.

Example 6.11 Design a sequence detector which detects the sequence 100011.

□ Solution

In general, the number of states in the state diagram is equal to the number of bits in the sequence. Once the number of states is known, one has to draw the directed lines with inputs and outputs as weighed between the two states. Let us start to draw state diagram, assuming initial state is A.

State A: In this state, the detector may receive either an input 0 or 1. Based on these inputs, a sequence detector is either in same state or move on to the next state as shown in Fig.6.26.

When input is 1, we have detected first bit in the sequence, hence we have to go to next state (B) to detect the next bit in the sequence

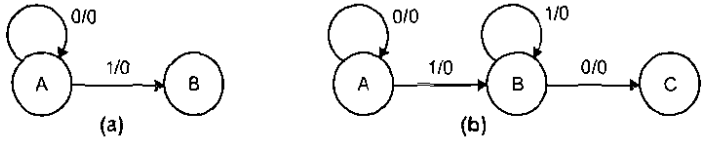


Fig 6.26

When input is 0, we have to remain in state A, because bit '0' is not the first bit in the sequence

State B: When input is 0, we have detected the second bit in the sequence. Hence we have to go to Next state (C) to detect the next bit in the sequence [See Fig.6.26(b)]

When input is 1, we have to remain in the state B, because 1 which we have detected may start the sequence output which is still zero for both cases

State C: When input is 0, we have detected the third bit in the sequence, hence we have to go next state (D) to detect the next bit in the sequence

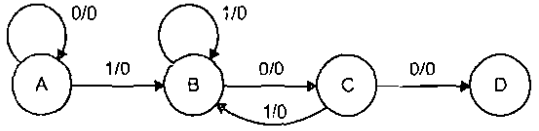


Fig 6.26(c)

When input is 1, we have to go to state B, because 1 which we have detected may not be in the sequence to be detected but it may be the start bit of the sequence, hence we can move to state B. The output is still zero (See Fig.6.26(c))

State D to State F: As explained for state A, state B and state C, if the desired bit is detected, we have to go for the next state otherwise we have to go to the previous state from where we can continue the desired sequence. When complete sequence is detected, we have to make output 1 and go to the initial state. The complete state diagram is shown in Fig.6.26(d).

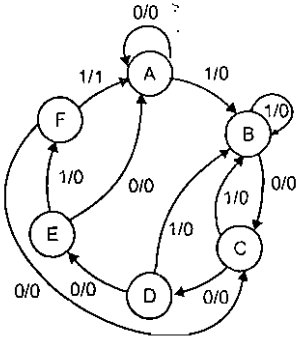


Fig 6.26(d)

State assignment

Assume that for state assignment, we need 3 flip flops to construct the sequence detector circuit. The sequence detector has a total of 6 states. Two flip flops are enough for less than or equal to 4 states. Hence $3(2^3 = 8)$ flip flops are required to construct the sequence detector. We choose JK flip flop and the flip flops are labeled as A, B and C, assuming state assignments as $A = 000, B = 001, C = 010, D = 011, E = 100$ and $F = 101$.

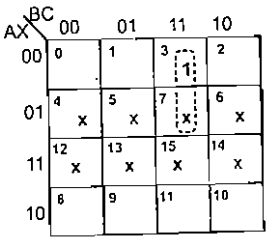
Excitation table

We can easily write the excitation table from the state diagram.

Input	Present state			Next state			Output	Flip flop Inputs					
X	A	B	C	A_{n+1}	B_{n+1}	C_{n+1}	Y	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	0	0	0	0	0	X	0	X	0	X
0	0	0	1	0	1	0	0	0	X	1	X	X	1
0	0	1	0	0	1	1	0	0	X	X	0	1	X
0	0	1	1	1	0	0	0	1	X	X	1	X	1
0	1	0	0	0	0	0	0	X	1	0	X	0	X
0	1	0	1	1	1	0	0	X	1	1	X	X	1
1	0	0	0	0	0	1	0	0	X	0	X	1	X
1	0	0	1	0	0	1	0	0	X	0	X	X	0
1	0	1	0	0	0	1	0	0	X	X	1	1	X
1	0	1	1	0	0	1	0	0	X	X	1	X	0
1	1	0	0	1	0	1	0	X	0	0	X	1	X
1	1	0	1	0	0	1	1	X	1	0	X	X	1

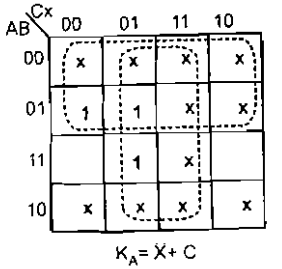
***K*-map simplification**

Expression for flip flop input J_A



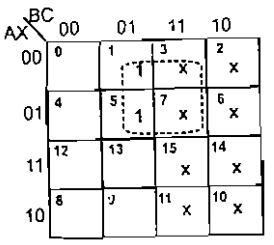
$J_A = \bar{X}\,BC$

Expression for flip flop input K_A



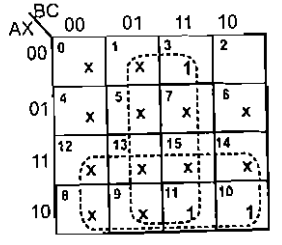
$K_A = \bar{X} + C$

Expression for flip flop input J_B



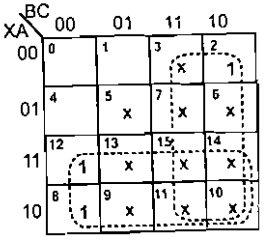
$J_B = \bar{X}\,C$

Expression for flip flop input K_B



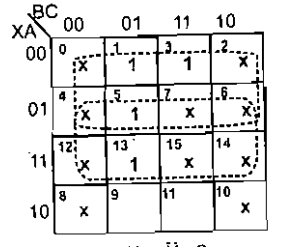
$K_B = \bar{X} + C$

Expression for flip flop input J_C



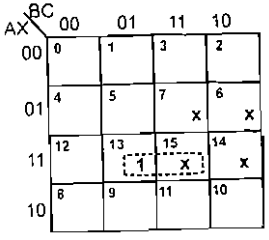
$J_C = X + B$

Expression for flip flop input K_C



$K_C = \bar{X} + C$

Expression for Output Y



$Y = XAC$

Logic diagram for sequence detector

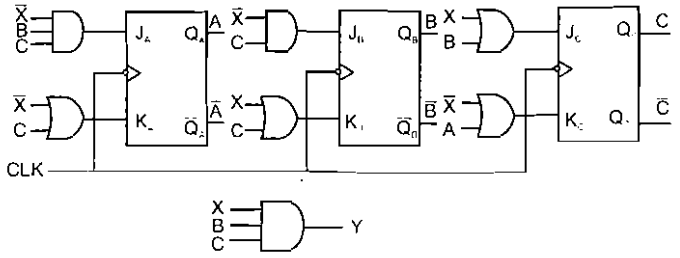


Fig 6.27 Logic diagram for sequence detector

Example 6.12 Design a sequence detector to detect the sequence 101 from 10101.

□ **Solution**

We have only 3 states because we have to detect the sequence 101 from the given number 10101. This circuit is allowed repetition.

Initially, we assume that the circuit is in its reset state, state *a*. With a 1 coming in as first bit in the valid sequence, it will go from state *a* to state *b* with an output as 0 because we have not yet detected all the bits in the sequence. When input is 0, we detect second valid bit in the sequence so it will go from state *b* to state *c*, otherwise state *b* remains same. When the input is 1, we detect third bit in the sequence, which will go from state *c* to state *b* with the output as 1 because we are yet to detect all bits in the sequence. If the input is 0, it will go from state *c* to state a with output 0.

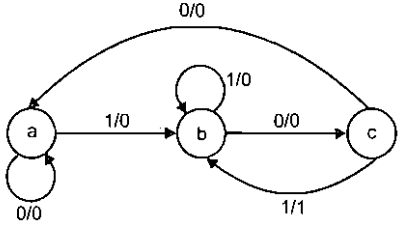


Fig 6.28 State diagram

The binary values are assigned to state *a*, *b* and *c*. Only two flip flops are enough ($2^2 = 4$)to design the sequence detector sequential logic circuit, by using *T* flip flops.

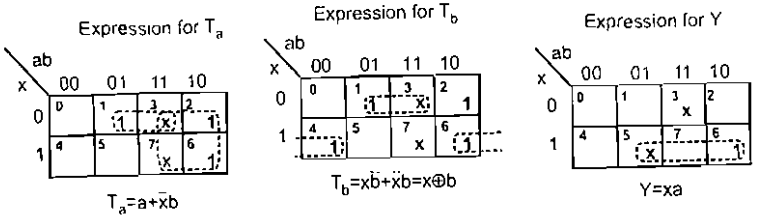
Assume the state assignment

$$a = 00, \qquad b = 01, \qquad c = 10$$

Consider the input is *X* and the output is *Y*.

Input	Present state		Next state		Flip flop Inputs		Output
X	*a*	*b*	a_{n+1}	b_{n+1}	T_a	T_b	*Y*
0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0
0	1	0	0	0	1	0	0
0	1	1	X	X	X	X	X
1	0	0	0	1	0	1	0
1	0	1	0	1	0	0	0
1	1	0	0	1	1	1	1
1	1	1	X	X	X	X	X

***K*-map simplification**



Logic diagram for sequence detector

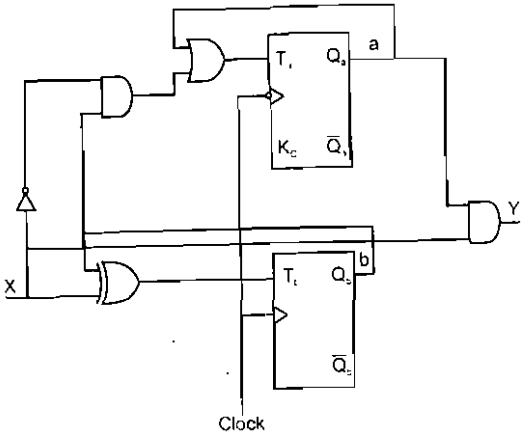


Fig 6.29 Logic diagram for sequence detector

Example 6.13 A sequential circuit with two *D* flip flops *A* and *B* and input *X* and output *Y* is specified by the following next state and output equations

$$A(t+1) = AX + BX$$
$$B(t+1) = A'X$$
$$Y = (A+B)X'$$

(a) Draw the logic diagram of the circuit

(b) Derive the state table

(c) Derive the state diagram

□ **Solution**

(a)

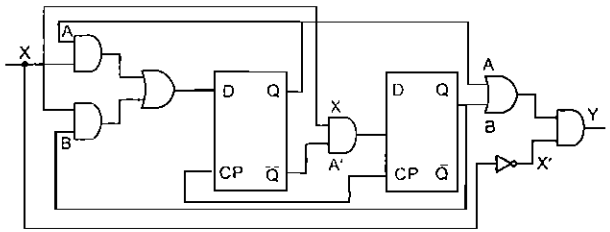


Fig 6.30

(b) **State Table**

Present state		Input	Next state		Flip flop Inputs		Output
A	B	X	A^+	B^+	D_A	D_B	Y
0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0
0	0	1	0	1	0	1	0
0	1	1	1	1	1	1	0
1	0	1	1	0	1	0	0
1	1	1	1	0	1	0	0

(c) **State diagram**

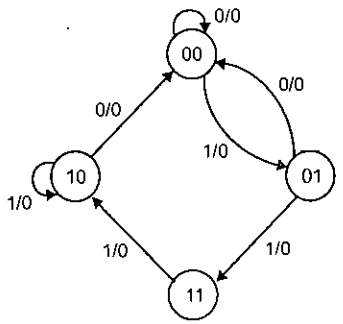


Fig 6.31

Example 6.14 Reduce the number of states in the following state table and tabulate the reduced state table.

Present state	Next state		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
a	f	b	0	0
b	d	c	0	0
c	f	e	0	0
d	g	a	1	0
e	d	c	0	0
f	f	b	1	1
g	g	h	0	1
h	g	a	1	0

□ **Solution**

Two states are said to be equivalent if their next states and outputs are equal.

Compare state 'a' with other states. The next state of 'a' and 'f' are equal for the inputs 0 and I but their outputs are not equal. Hence $a \neq f$. Similarly comparing other states it is found that $b \equiv e$ as their next state and outputs are equal and $d = h$. Hence the state table can be reduced as shown. The row e is removed from the table and if any previous row containing e, it is replaced by b and the rows are compared. If any state has the same next state and outputs replace one row, continue this process to obtain reduced state table.

[The strike out lines are kept to illustrate reduction process]

Present state	Next state		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
a	f	b	0	0
b	d	~~e~~a	0	0
~~c~~	~~f~~	~~b~~	~~0~~	~~0~~
d	g	a	1	0
~~e~~	~~d~~	~~c~~	~~0~~	~~0~~
f	f	b	1	1
g	g	~~h~~d	0	1
~~h~~	~~g~~	~~a~~	~~1~~	~~0~~

After the first comparison.we find that $a \equiv c$ hence the reduced state table is as shown below.

Present state	Next state		Output	
	$x=0$	$x=1$	$x=0$	$x=1$
a	f	b	0	0
b	d	a	0	0
d	g	a	1	0
f	f	b	1	1
g	g	d	0	1

■

Example 6.15 Design a sequential circuit with two D flip flops, A and B, and one input x. When $x = 0$, the state of the circuit remains the same. When $x = 1$, the circuit passes through the state transitions from 00 to 01 to 11 to 10 and back to 00 and repeats.

□ **Solution**

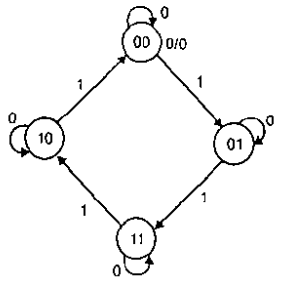


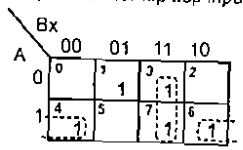
Fig 6.32

State table

Present state		Input	Next state		Output	
A	B	X	A^+	B^+	D_A	D_B
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	1
0	1	1	1	1	1	1
1	0	0	1	0	1	0
1	0	1	0	0	0	0
1	1	0	1	1	1	1
1	1	1	1	0	1	0

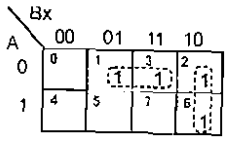
***K*-map simplification**

Expression for flip flop input D_A



$D_A = A\bar{x} + Bx$

Expression for flip flop input D_B



$D_A = \bar{A}x + B\bar{x}$

Circuit diagram

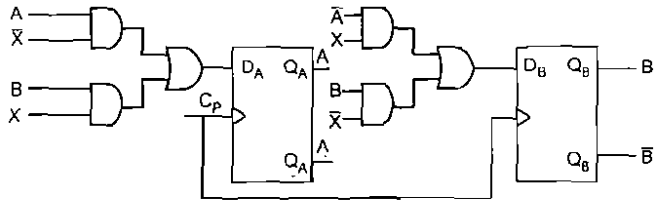


Fig 6.33

■

Example 6.16 Design a sequential circuit that three flip flops A, B, C, one input x and one output y. The state diagram is shown in the fig 6.34. The circuit is to be designed by treating the unused states as don't care conditions. Use JK flip flops in the design.

■ **Solution**

State diagram

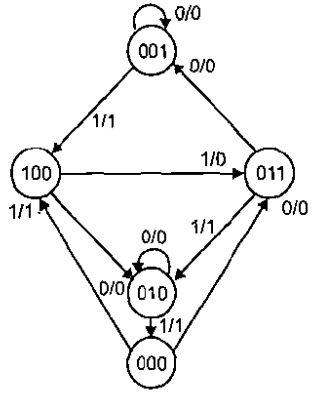


Fig 6.34

State table for the circuit

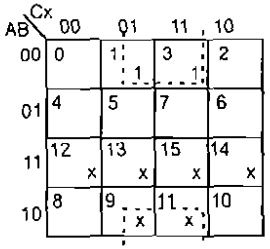
Present State			Next state						Output y	
			$x=0$			$x=1$				
A	B	C	A^+	B^+	C^+	A^+	B^+	C^+	$x=0$	$x=1$
0	0	1	0	0	1	1	0	0	0	1
1	0	0	0	1	0	0	1	1	0	0
0	1	0	0	1	0	0	0	0	0	1
0	1	1	0	0	1	0	1	0	0	1
0	0	0	0	1	1	1	0	0	0	1

Excitation table for the circuit

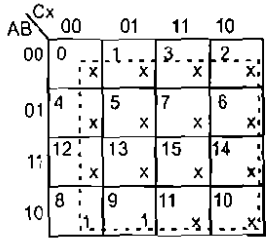
Present state			Input	Next state			Flip flop Inputs						Output
A	B	C	x	A	B	C	J_A	K_A	J_B	K_B	J_C	K_C	y
0	0	0	0	0	1	1	0	X	1	X	1	X	0
0	0	0	1	1	0	0	1	X	0	X	0	X	1
0	0	1	0	0	0	1	0	X	0	X	X	0	0
0	0	1	1	1	0	0	1	X	0	X	X	1	1
0	1	0	0	0	1	0	0	X	X	0	0	X	0
0	1	0	1	0	0	0	0	X	X	1	0	X	1
0	1	1	0	0	0	1	0	X	X	1	X	0	0
0	1	1	1	0	1	0	0	X	X	0	X	1	1
1	0	0	0	0	1	0	X	1	1	X	0	X	0
1	0	0	1	0	1	1	X	1	1	X	1	X	0

***K*-map simplification**

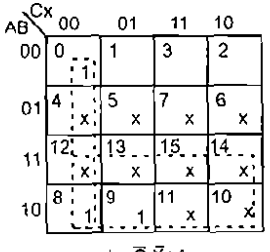
Note : minterms 10, 11, 12, 13, 14 and 15 are don't care



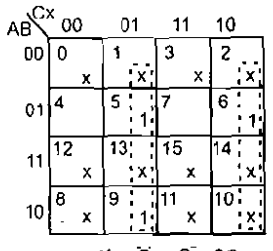
$J_A = \bar{B}x$



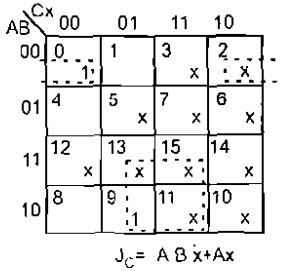
K_A=1(since all the cells are grouped)



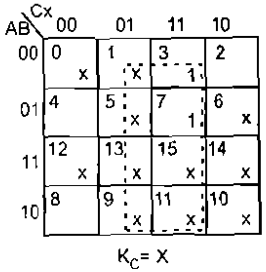
$J_B = \bar{C}\,\bar{X} + A$



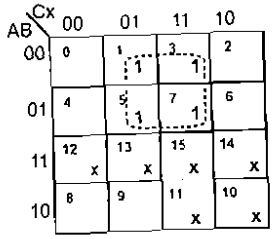
$K_B = \bar{C}x + C\bar{x} = C \oplus x$



$J_C = A\,B\,\bar{x} + Ax$



$K_C = X$



$Y = Ax$

Circuit Diagram

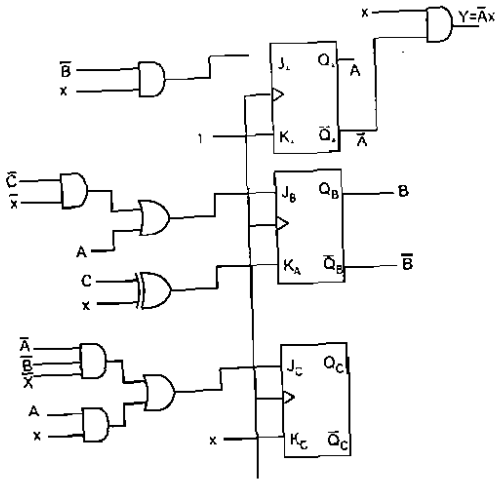


Fig 6.35

■

Example 6.17 Derive the state table and state diagram for the sequential circuit shown Fig 6.36.

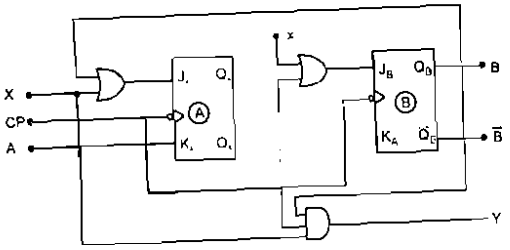


Fig 6.36

☐ Solution

The given circuit has two flip flops, A and B, one input X and one input Y. So, it produces a state table with 4 rows as shown below

Present State		Next state		Output y	
		$x=0$	$x=1$	$x=0$	$x=1$
A	B	$A\bar{B}$	AB	y	y
0	0	01	11	0	1
0	1	10	10	0	1
1	0	01	00	0	0
1	1	00	00	0	0

The state diagram for the sequential circuit is shown in Fig 6.37.

State diagram

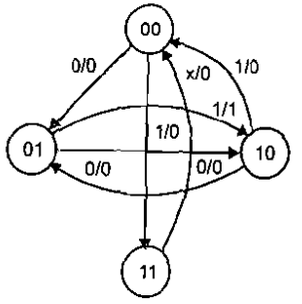


Fig 6.37

■

Example 6.18 A sequential circuit with two D flip flops, A and B; two inputs x and y; and one output z, is specified by the following next state and output equations:

$$A(t+1) = \bar{x}y + xA$$

$$B(t+1) = \bar{x}\bar{B} + xA$$

$$z = B$$

(a) Draw the logic diagram of the circuit

(b) List the state table for sequential circuit

(c) Draw the corresponding state diagram

☐ Solution

(a) Logic diagram of the sequential logic circuit

A sequential logic circuit is drawn using given equations as shown in Fig.6.38.

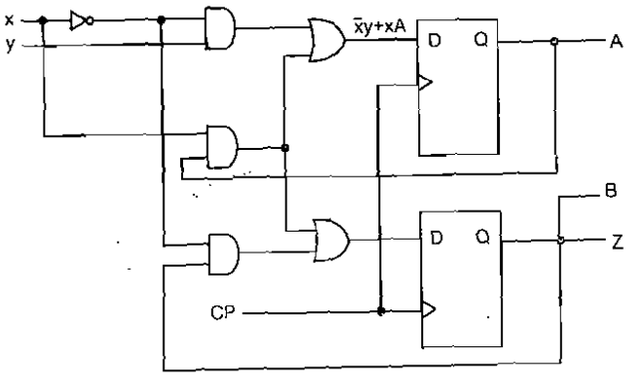


Fig 6.38

(b) State table

Present State		Inputs		Next state		Output
A	B	x	y	A	B	z
0	0	0	0	0	0	0
0	0	0	1	1	0	0
0	0	1	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	1	1
0	1	0	1	1	1	1
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	0	0	0
1	0	0	1	1	0	0
1	0	1	0	1	1	0
1	0	1	1	1	1	0
1	1	0	0	0	1	1
1	1	0	1	1	1	1
1	1	1	0	1	1	1
1	1	1	1	1	1	1

(c) State diagram

The state diagram is drawn as shown in Fig.6.39 with the help of the above table

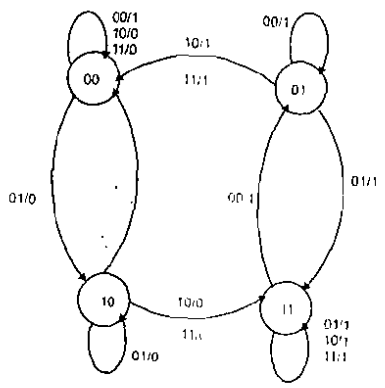


Fig 6.39

Example 6.19 Design a sequential circuit with two flip-flops A and B, and one input x. When $x = 0$, the state of the circuit remains same. When $x = 0$, the circuit goes through the state transitions tram 00 to 01 to 11 to 1 0 back to 00, and repeats.

□ **Solution**

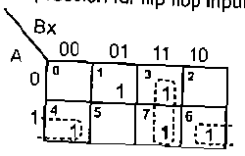
State table

Present state		Input	Next state		FF inputs	
A	B	x	A	B	D_A	D_B
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	1
0	1	1	1	0	1	0
1	0	0	0	1	0	1
1	0	1	1	0	1	0
1	1	0	1	0	1	0
1	1	1	1	1	1	1

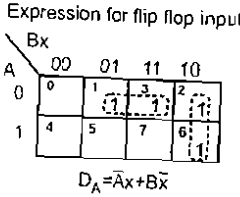
Note : the FF inputs of the D FF is same as the next state

***K*-map simplification**

Expression for flip flop input D_A



Expression for flip flop input D_B



Logic Diagram

By using the above Boolean Expression, the diagram is constructed as shown in Fig.6.40.

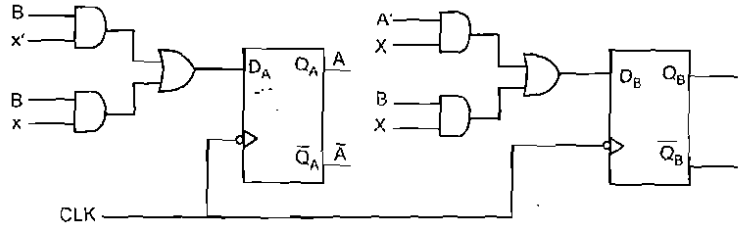


Fig 6.40

Example 6.20 Design a sequential circuit with two JK flip-flops A and B and two inputs E and x. If $E = 0$, the circuit remains in the same state regardless of the value of x. When $E = 1$ and $x = 1$, the circuit goes through the state transitions from 00 to 0 1 to 10 to 11 back to 00, and repeats. When $E = 1$ and $x = 0$, the circuit goes through the state transitions from 00 to 11 to 10 to 0 1 back to 00, and repeats.

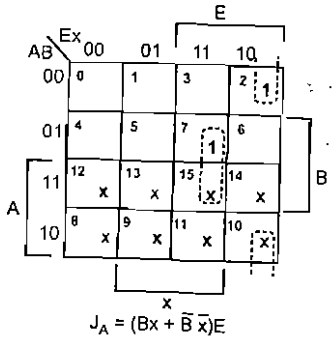
□ **Solution**

The excitation table is derived directly from the given specification.

Present state		Input		Next state		Flip flop Inputs			
A	B	E	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	0	0	0	X	0	X
0	0	0	1	0	0	0	X	0	X
0	0	1	0	1	1	1	X	1	X
0	0	1	1	0	1	0	X	1	X
0	1	0	0	0	1	0	X	X	0
0	1	0	1	0	1	0	X	X	0
0	1	1	0	0	0	0	X	X	1
0	1	1	1	1	0	1	X	X	1
1	0	0	0	1	0	X	0	0	X
1	0	0	1	1	0	X	0	0	X
1	0	1	0	0	1	X	1	1	X
1	0	1	1	1	1	X	0	1	X
1	1	0	0	1	0	X	0	X	0
1	1	0	1	1	1	X	0	X	0
1	1	1	0	1	0	X	0	X	1
1	1	1	1	0	0	X	1	X	1

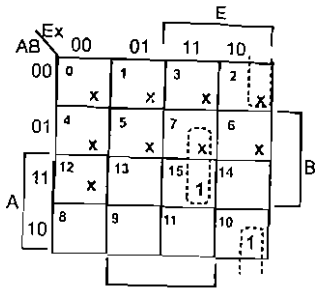
K-map simplification

Expression for J_A



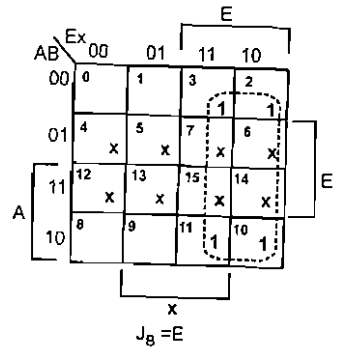
$J_A = (Bx + \bar{B}\,\bar{x})E$

Expression for K_A



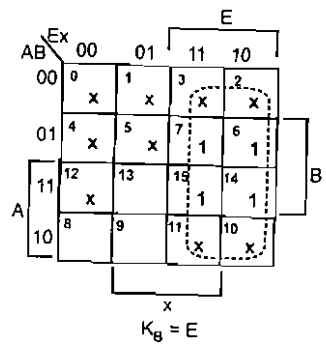
$K_A = BEx + \bar{B}\,\bar{E}\bar{x} = E(Bx + \bar{B}\bar{x})$

Expression for J_B



$J_B = E$

Expression for K_B



$K_B = E$

Logic diagram

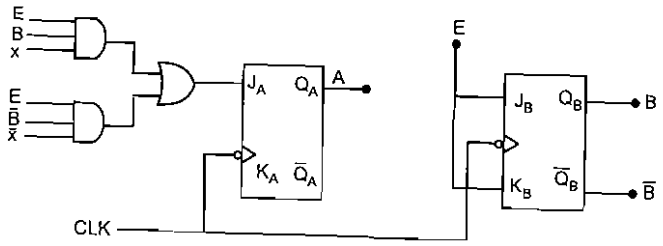


Fig 6.41

Short Questions and Answer

1. ***Mention the steps involved in the analysis of a sequential circuit.***

   The analysis of a sequential circuit consists of:

   (a) Obtaining a table or a diagram for the time sequence of inputs, outputs and internal states.
   (b) Writing Boolean expressions, which include the necessary time sequence, either directly or indirectly.

2. ***What is a state-equation?***

   A state-equation (also called transition equation) specifies the next state as a function of the present state and inputs.

3. ***What is a state-table?***

   A state-table contains data such as the time sequence of inputs, outputs and flip-flop states. The table consists of four sections labelled present state, input, next state and output state.

4. ***What is a state diagram?***

   A state diagram is a graphical representation of the information available in a state table. In the diagram, a state is represented by a circle and the transitions between states are indicated by directed lines connecting the circles.

5. ***What are the informations obtained from a state diagram?***

   The informations obtained are as follows :

   The state of the flip-flops are identified by the binary number inside the circle.

   A directed line connecting a circle which indicates that change of state occurs.

6. ***What are input and output equations?***

   The part of the circuit that generates the inputs to flipflops is described algebraically by a set of Boolean functions called input equations.

7. ***How are the next-state values of a sequential circuit that uses JK or T type flip-flops derived?***

   The next-state values of a sequential circuit that uses JK or T type flip-flops are derived using the following procedure:

   (a) Determine the flip flop input equations in terms of the present state and input variables.

(b) List the binary values of each input equation.
(c) Use the corresponding flip flop characteristic table to determine the next state value in the state table.

8. ***How are the next-state values obtained from the characteristic equation?***

The next state values can be obtained by evaluating the state equations from the characteristic equation. This is done by the following procedure:

(a) Determine the flip-flop input equations in terms of the present state and input variables.
(b) Substitute the input equations into the flip-flop characteristic equation to obtain the state equations.
(c) Use the corresponding state equations to determine the next state value in the state table.

9. ***What are the two models of sequential circuits?***

The two models of sequential circuits are:

(a) Mealy model, and
(b) Moore model.

10. ***Compare Mealy and Moore machine.***

Mealy machine	Moore machine
i. The output is a funetion of both the present state and input.	i. The output is a function of the present state only.
ii. The outputs may change if the inputs change during the clock cycle.	ii. The outputs are synchronized with the clock, because they depend only on flip-flop outputs that are synchronized with the clock.

11. ***What is the use of initial statement?***

The initial statement is used for generating input signals to simulate a design. In simulating a sequential circuit, it is necessary to generate a clock source for triggering the flip-flops.

12. ***How can the always statement be controlled?***

The always statement could be controlled by delays that wait for a certain time or by certain conditions to become true or by events to occur.

13. ***What is a sensitivity list?***

A sensitivity list specifies the events that must occur to initiate the execution of the procedural statements in the always block. Statements within the block execute sequentially and the execution suspends after the last statement has been executed.

14. ***What are the two kinds of procedural assignments?***

(a) The two kinds of procedural assignments are:

Blocking assignments

Blocking assignment statements are executed sequentially in the order, they are listed in a sequential block.

Blocking assignments use the symbol (=) as the assignment operator.

Example for procedural blocking assignments:

$$B = A$$
$$C = B + 1$$

(b) Non-blocking assignments:

It evaluates the expressions on the right hand side, but do not make the assignment to the left hand side, until all expressions are evaluated.

It uses the (<=) as the operator.

Example for non-blocking assignments: $B \Leftarrow A$

$$C \Leftarrow B + 1$$

15. ***How can we determine the behavior of a clocked sequential circuit?***

The behavior of a clocked sequential circuit could be determined from the inputs, outputs and the state of its flip-flops.

16. ***What are clocked sequential circuits?***

Synchronous sequential circuits that use clock pulses in the inputs of storage elements are called clocked sequential circuits.

17. ***How can we describe the structure of a sequential circuit?***

The sequential circuit is made up of flip-flops and gates and so its structure can be described by a combination of data flow and behavioral statements.

Short Answer Questions

1. Differentiate synchronous and asynchronous sequential logic circuits.
2. What are the classification of sequential machine ?
3. Define Mealy and Moore Machines.
4. Define state assignment.
5. When are two states said to be equivalent states ?

Review Questions

1. What are the steps for the design of an asynchronous sequential circuit ?
2. What is the significance of state assignment ?
3. List the different techniques used for state assignment.
4. Write a short note on

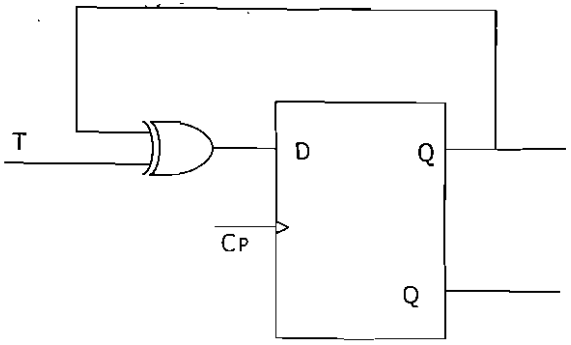
   (a) Shared row state assignment.
   (b) One hot state assignment.

Exercises

1. A sequential circuit with two D flip flops, A and B, two inputs x and y, and one input z is specified by the following next state and output equations.

$$A(t+1) = x'y + xA$$
$$B(t+1) = x'B + xA$$
$$Z = B$$

   (a) Draw the logic diagram of the circuit
   (b) Derive the state table.



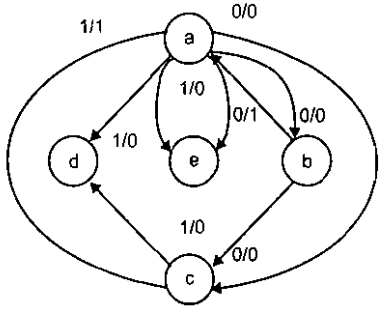
2. A sequential circuit has one flip flop Q, two inputs x and y: and one output S. It consists of a full-adder circuit connected to a D flip flop as shown in following fig.1. Derive the state table and state diagram of the sequential circuit.

3. Analyze the circuit in the following Fig.2 and prove that it is equivalent to a T flip flop.

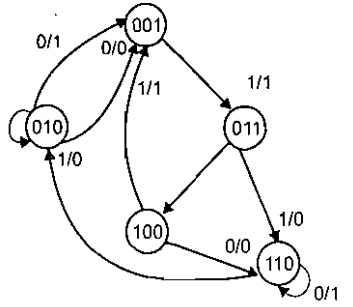
4. A sequential circuit has two JK flip flops, one input x and one output y. Following is the logic diagram of the circuit. Derive the state table and state diagram 7. Design a sequential circuit for the given state diagram by using JK flip flop of the circuit.

5. Design a sequential circuit with two JK flip flops, A and B and two inputs E and X. If E = 0, the circuit remains in the same state regardless of the value of x. When E = 1 and X = 1, the circuit goes through the state transitions from 00 to 01 to 10 to 11 back to 00 and repeats. When E = 1 and X = 0, the circuit goes through the state transitions from 00 to 11 to 10 to 01 back to 00 and repeats.

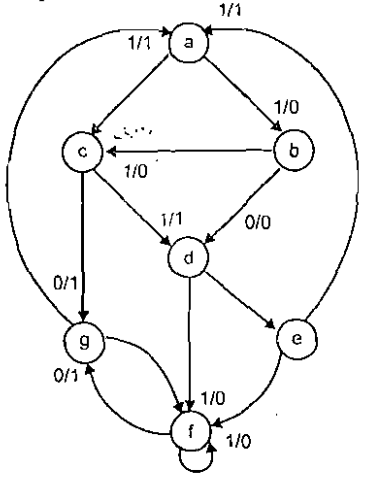
6. Design a sequential circuit for the given state diagram.



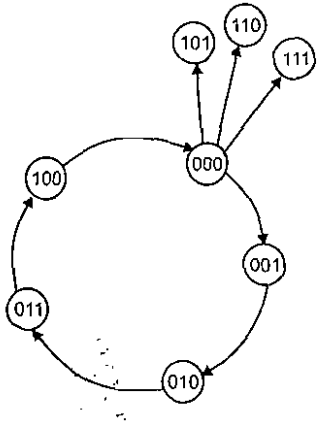
7. Design a sequential circuit for the given state diagram in Fig. P5.



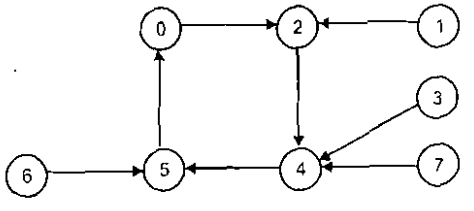
8. Design a clocked sequential circuit for the given state diagram in Fig. P6.



9. Design a sequential circuit for the following state diagram.



10. Design a sequential circuit for the given state diagram using JK flip flop.



11. Obtain the state table for the given state diagram and reduce it.