Electronics Engineering

Advanced Electronics

Comprehensive Theory

with Solved Examples and Practice Questions

MADE EASY Publications

MADE EASY Publications

Corporate Office: 44-A/4, Kalu Sarai (Near Hauz Khas Metro Station), New Delhi-110016
E-mail: infomep@madeeasy.in
Contact: 011-45124660, 8860378007

Visit us at: www.madeeasypublications.org

Advanced Electronics

First Edition: 2017
Second Edition: 2018
Third Edition: 2019
Fourth Edition: 2020

Contents

Advanced Electronics

■■■

CHAPTER 5

Synthesis of Synchronous Sequential Circuits

Introduction

State Diagram (State graph)

It is a pictorial representation of the relationships between the present state, the input, the next state and the output of a sequential circuit, i.e. the state diagram is a pictorial representation of the behaviour of a sequential circuit.

State table

The state table is a tabular representation of the state diagram. Even though the behaviour of a sequential circuit can be conveniently described using a state diagram, for its implementation the information of the state diagram is to be translated into a state table. Both the state diagram and the state table contain the same information.

5.1 Finite State Machine (FSM)

- Finite state machines are sequential circuits whose past histories can affect their future behaviour in only a finite number of ways, i.e. they are machines with a fixed number of states.
- A sequential circuit is referred to as a finite state machine (FSM).
- Finite state machines are of two types. They differ in the way the output is generated. They are
  1. Mealy type model 2. Moore type model

5.1.1 Mealy Machine

In this model, the output depends on both the present state of the circuit and the present inputs.

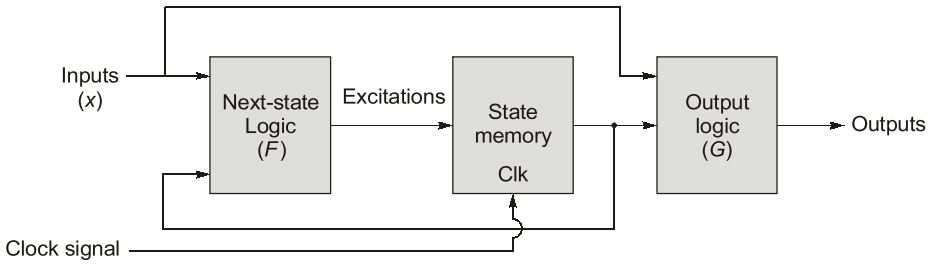
$$NS = F(PS, X)$$
$$Output = G(PS, X)$$

F and G are some logic functions.

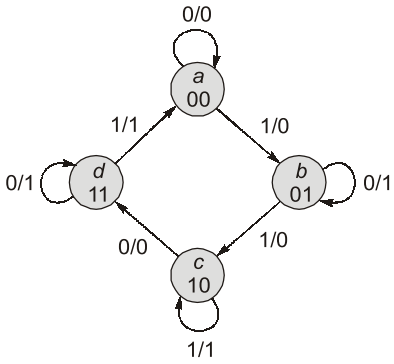
PS = Present state
NS = Next state
X = Present inputs

Input changes may affect the output of the logic circuit.

It requires less number of states for implementing a function than that of Moore model.



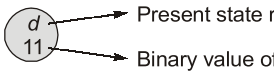
State diagram and state table for a Mealy circuit are given below.



State diagram

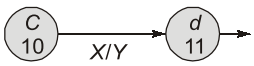
PS	NS output	
	$X = 0$	$X = 1$
a	a, 0	b, 1
b	b, 1	c, 0
c	d, 0	c, 1
d	d, 0	a, 1

Stable table



Present state represented by the circle.

Binary value of the state represented by the circle.



Always the arrow directs from the present state to the next state.

X = Input and Y = Output

In the Mealy model, the output depends on the present input. Hence both input, output are represented on the directed lines.

5.1.2 Moore Machine

In this model, the output depends only on the present state of the circuit.

$$NS = F(PS, X)$$

$$Output = G(PS)$$

F and G are some logic functions.

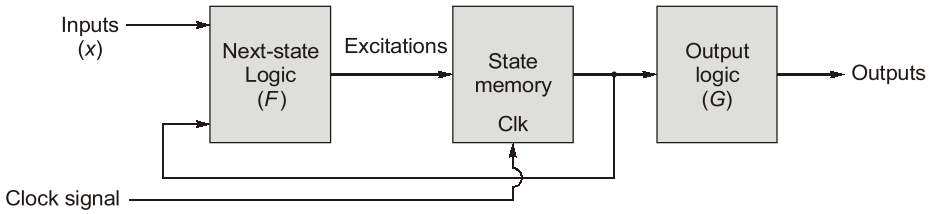
PS = Present state

NS = Next state

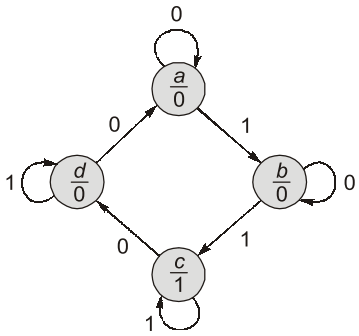
X = Present inputs

Input changes do not affect the output of the logic circuit.

It requires more number of states for implementing a function than that of Mealy model.



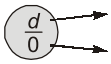
State diagram and state table for a Moore circuit are given below.



State diagram

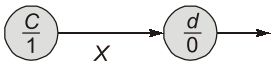
PS	NS Input		Present output
	$X = 0$	$X = 1$	
a	a	b	0
b	b	c	0
c	d	c	1
d	a	d	0

State table



Present state represented by the circle.

Output of the circuit corresponding to the present state represented by the circle.

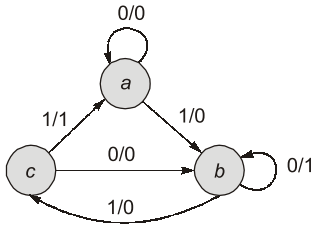


Always the line directs from the present state to the next state.

X = Input

In the Moore model, the output depends only on the present state of the circuit and it is independent of the input applied. Hence, the present output is represented inside the circle belongs to the present state and input is represented on the directed line.

Example - 5.1 Draw the equivalent Moore state diagram for the following Mealy state diagram.



Solution :

In a Moore state diagram, each state is associated with a fixed output.

States for the Moore state diagram, corresponding to the given Mealy stae diagram, can be assigned as shown below.

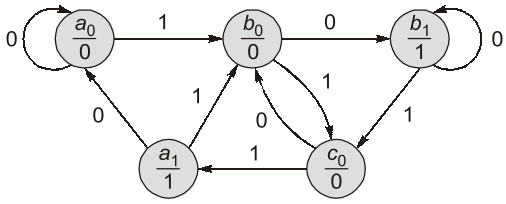
Mealy State	Equivalent Moore State
a, with output 0	a_0
a, with output 1	a_1
b, with output 0	b_0
b, with output 1	b_1
c, with output 0	c_0

In the given Mealy state diagram, there no transition that produces next state as "c" with output "1". Hence, there is no need a state equivalent to "c" with output "1", i.e. c_1 in Moore model.

The state table and state diagram of the required Moore model can be given as,

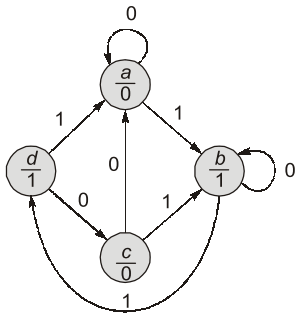
Present state	Next state (output)	
	$X = 0$	$X = 1$
a_0	$a_0(0)$	$b_0(0)$
a_1	$a_0(0)$	$b_0(0)$
b_0	$b_1(1)$	$c_0(0)$
b_1	$b_1(1)$	$c_0(0)$
c_0	$b_0(0)$	$a_1(1)$

State table



State diagram

Example - 5.2 Draw the equivalent Mealy state diagram for the following Moore state diagram.



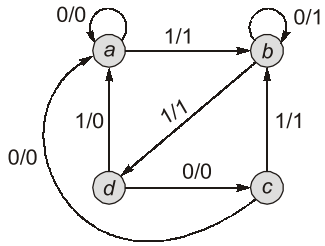
Solution :

Same state notations can be followed in the corresponding Mealy state diagram of the given Moore state diagram.

The state table and state diagram of the required Mealy model can be given as,

Present state	Next state		Output	
	$X = 0$	$X = 1$	$X = 0$	$X = 1$
a	a	b	0	1
b	b	d	1	1
c	a	b	0	1
d	c	a	0	0

State table

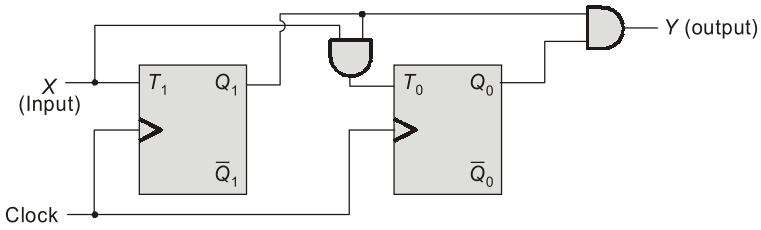


State diagram

State Reduction

- The state reduction technique basically avoids the introduction of redundant states.
- The reduction in redundant states reduces the number of flip-flops and logic gates, reducing the cost of the final circuit.
- Two states are said to be equivalent if every possible set of inputs generate exactly the same output and the same next state.
- When two state are equivalent, one of them can be removed without altering the input-output relationship.
- Let us illustrate the reduction technique with an example.

Example - 5.3 Develop a state diagram for the logic circuit shown in the figure below.



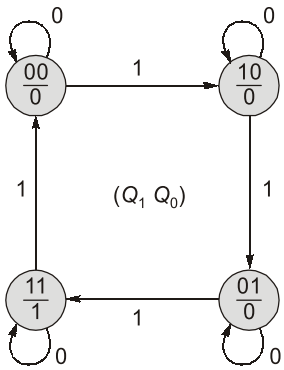
Solution :

Excitations of the flip flops are, $T_1 = X$ and $T_0 = Q_1X$

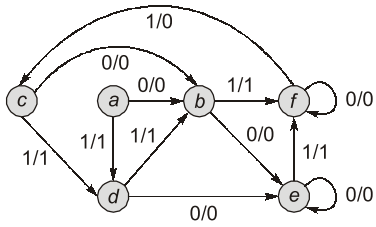
Output signal, $Y = Q_1Q_0 \Rightarrow$ Moore circuit

The state table and state diagram of the given sequential circuit can be developed as shown below.

Present state		Input	Excitations		Next state		Output
Q_1	Q_0	X	T_1	T_0	Q_1^+	Q_0^+	Y
0	0	0	0	0	0	0	0
0	0	1	1	0	1	0	0
0	1	0	0	0	0	1	0
0	1	1	1	0	1	1	1
1	0	0	0	0	1	0	0
1	0	1	1	1	0	1	0
1	1	0	0	0	1	1	1
1	1	1	1	1	0	0	0



Example - 5.4 Reduce the following state diagram:



Solution :

- The state table of the given state diagram is,

PS	NS		Output	
	$X = 0$	$X = 1$	$X = 0$	$X = 1$
a	b	d	0	1
b	e	f	0	1
c	b	d	0	1
d	e	b	0	1
e	e	f	0	1
f	f	c	0	0

"*b*" and "*e*" are said to be equivalent states. So, remove "*e*" and replace it with "*b*".

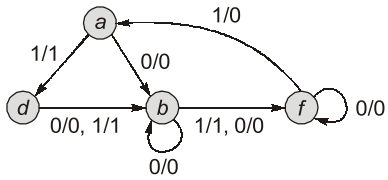
"*a*" and "*c*" are said to be equivalent states. So, remove "*c*" and replace it with "*a*".

- So, the reduced state table is,

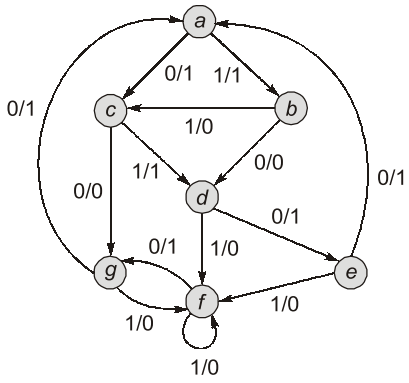
PS	NS		Output	
	$X=0$	$X=1$	$X=0$	$X=1$
a	*b*	*d*	0	1
b	*b*	*f*	0	1
d	*b*	*b*	0	1
f	*f*	*a*	0	0

No further equivalent states are found. So, it is not possible to reduce further.

- Now, the reduced state diagram can be drawn from the reduced state table as shown below:



Example - 5.5 Reduce the following state diagram:



Solution :

- The state table of the given state diagram is,

PS	NS		Output	
	$X=0$	$X=1$	$X=0$	$X=1$
a	*c*	*b*	1	1
b	*d*	*c*	0	0
c	*g*	*d*	0	1
d	*e*	*f*	1	0
e	*a*	*f*	1	0
f	*g*	*f*	1	0
g	*a*	*f*	1	0

"*g*" and "*e*" are said to be equivalent states. So, remove "*g*" and replace it with "*e*".

- The reduced state table is,

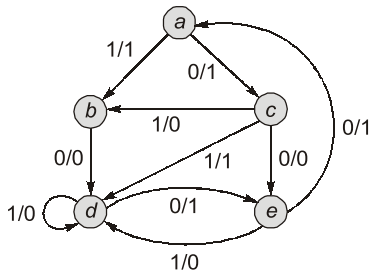
PS	NS		Output	
	$X=0$	$X=1$	$X=0$	$X=1$
a	c	b	1	1
b	d	c	0	0
c	e	d	0	1
d	e	f	1	0
e	a	f	1	0
f	e	f	1	0

"e" and "f" are said to be equivalent states. So, remove "f" and replace it with "e".

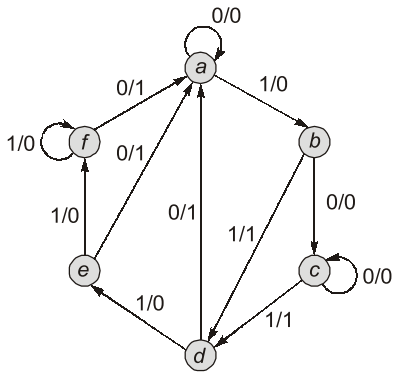
- The reduced state table is,

PS	NS		Output	
	$X=0$	$X=1$	$X=0$	$X=1$
a	c	b	1	1
b	d	c	0	0
c	e	d	0	1
d	e	d	1	0
e	a	d	1	0

- No further reduction is possible, hence the reduced state diagram is as shown below:



Example - 5.6 Reduce the following state diagram:



Solution :

- The state table of the state diagram can be given as follows:

Present State	Next State $X=0$	Next State $X=1$	Output $X=0$	Output $X=1$
a	a	b	0	0
b	c	d	0	1
c	c	d	0	1
d	a	e	1	0
e	a	f	1	0
f	a	f	1	0

X = Input

States "b" and "c" have same next state and output for a given value of input. So, these two states are said to be equal and we can replace the state "c" with state "b".

States "e" and "f" have same next state and output for a given value of input. So, these two states are said to be equal and we can replace the state "f" with state "e".

- After replacing states "c" and "f" with states "b" and "e" respectively, the resultant state table is as follows:

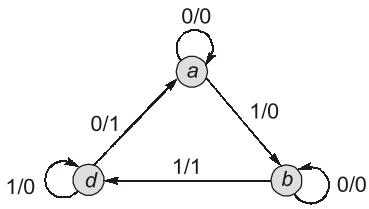
Present State	Next State $X=0$	Next State $X=1$	Output $X=0$	Output $X=1$
a	a	b	0	0
b	b	d	0	1
d	a	e	1	0
e	a	e	1	0

States "d" and "e" have same next state and output for a given value of input. So, these two states are said to be equal and we can replace the state "e" with state "d".

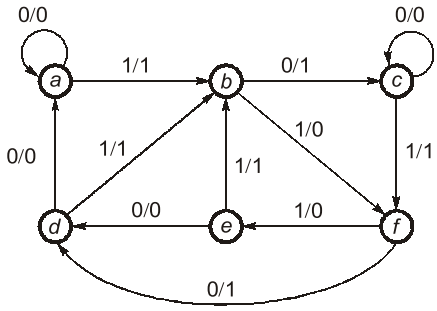
- After replacing state "e" with state "d", the resultant state table is as follows:

Present State	Next State $X=0$	Next State $X=1$	Output $X=0$	Output $X=1$
a	a	b	0	0
b	b	d	0	1
d	a	d	1	0

- In the above state table, no two states have same next state and output for a given value of input. Hence there are no equal states and it is not possible to reduce the state table further.
- From the above table, the reduced state diagram of the given state diagram can be drawn as follows:



Example - 5.7 Reduce the following state diagram:



Solution :

- Let us assume that, the input variable of the circuit as "X" and the output variable of the circuit as "Z".
- The stable table for the given mealy type state diagram is as given below:

Present State	Next State		Output (z)	
	$X=0$	$X=1$	$X=0$	$X=1$
a	a	b	0	1
b	c	f	1	0
c	c	f	0	1
d	a	b	0	1
e	d	b	0	1
f	d	e	1	0

For states "a" and "d", next state and output are same for an applied input. Hence, states "a" and "d" are said to be equal and the state "d" can be replaced with state "a".

- After replacing state "d" with state "a", the resultant state table is as given below:

Present State	Next State		Output (z)	
	$X=0$	$X=1$	$X=0$	$X=1$
a	a	b	0	1
b	c	f	1	0
c	c	f	0	1
e	a	b	0	1
f	a	e	1	0

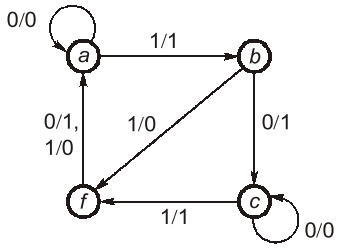
For states "a" and "e", next state and output are same for an applied input. Hence, states "a" and "e" are said to be equal and the state "e" can be replaced with state "a".

- After replacing state "e" with state "a", the resultant state table is as given below:

Present State	Next State		Output (z)	
	$X=0$	$X=1$	$X=0$	$X=1$
a	a	b	0	1
b	c	f	1	0
c	c	f	0	1
f	a	a	1	0

In the above state table, no two states have same next state and output for an applied input. Hence, it is not possible to reduce the state table further.

- So, the corresponding reduced state diagram is as given below:



State Assignment

The process of assigning binary values to the states of the sequential machine is known as "state assignment". The binary values are to be assigned to the states in such a way that it is possible to implement flip-flop input functions using minimum logic gates. The output values of the memory devices are referred to as state variable.

Here in our discussion memory elements are nothing but flip-flops. So, the output of a flip-flop is known as a state variable of the finite state machine. Follow some basic rules while assigning the binary values to the states.

1. $N \le 2^n \Rightarrow N$ = Number of unique states of the system
   n = Number of state variables (or) number of flip-flops required
2. States having the same NEXT STATE for a given input condition should have assignments which can be grouped into logically adjacent cells in a *K*-map.
3. States that are the next states of a single state should have assignment which can be grouped into logically adjacent cells in a *K*-map.
4. Resultant binary states must be unique.

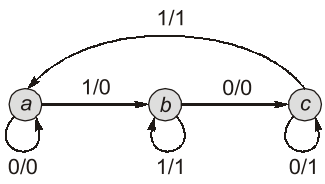
Transition and Output Table

- The transition and output table can be obtained from the state table by modifying the entries of the state table to correspond to the states of the machine in accordance with the selected state assignment.
- In this table, the next state and output entries are separated into two sections.
- The next state part of the state table is called the transition table.

Excitation Table

- The excitation table of a sequential machine gives information about the excitations (or) inputs required to be applied to the memory elements (flip-flops) in the sequential circuit to bring the sequential machine from the present state to the next state.
- It also gives information about the outputs of the machine after application of the present inputs.

Example - 5.8 Assign binary values to the states, write transition and output table, excitation table (with *D*-FFs) and obtain the minimal expressions for the state diagram shown below.



PS	NS		Output	
	$X=0$	$X=1$	$X=0$	$X=1$
a	a	b	0	0
b	c	b	0	1
c	c	a	1	1

Solution :

State assignment:

3 unique states (a, b, c) ⇒ So, two states variable (or) two flip-flops are required.

$a \rightarrow 00$
$b \rightarrow 01$ } randomly assigned three unique binary states.
$c \rightarrow 10$

Transition and output table:

	PS	NS		Output (f)	
		$X=0$	$X=1$	$X=0$	$X=1$
a →	00	00	01	0	0
b →	01	10	01	0	1
c →	10	10	00	1	1

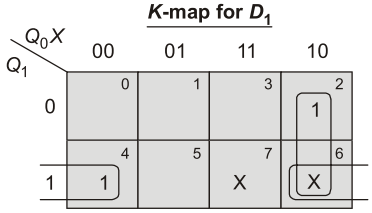
Excitation table:

PS			NS				
Q_1	Q_0	X	Q_1	Q_0	D_1	D_2	F (output)
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0
0	1	0	1	0	1	0	0
0	1	1	0	1	0	1	1
1	0	0	1	0	1	0	1
1	0	1	0	0	0	1	1

Q_1, Q_0 are called as state variables which are the outputs of the flip-flops.

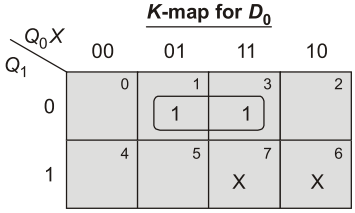
Minimization:

K-map for D_1



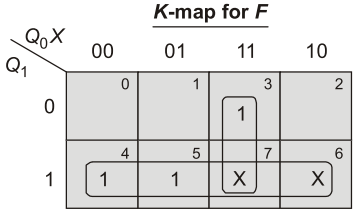
$D_1 = Q_1\overline{X} + Q_0\overline{X}$

K-map for D_0



$D_0 = Q_1\overline{X}$

K-map for F



$F = Q_1 + Q_0X$

5.2 Design of a Sequential Circuit or Finite State Machine

Step 1 : State the purpose of the machine in simple words.

Step 2 : Draw the state diagram which gives the complete information about the circuit working.

Step 3 : Write the state table for the state diagram obtained in step-2.

Step 4 : Convert the state table into reduced standard form state table by removing the redundant states.

Step 5 : Assign the binary values to the states (state assignment).

Step 6 : Write the transition and output table.

Step 7 : Choose the type of flip-flops and form the excitation table.

Step 8 : Obtain the minimal expressions flip-flops and the output (based on the contents of the excitation table) using *K*-maps.

Step 9 : Draw the logic diagram using minimal expression obtained in the step 9.