Digital Electronics

State Equations & Diagrams

A state euqtion is an algebraic expression that specifies the condition for a flip-flop state transition. The left side of the equation denotes the next state of the flip-flop and the right side of the equation is a Boolean expression that specifies the present state and input conditions that make the next state equal to 1.

$$A(t+1) = Ax + Bx; \qquad B(t+1) = A'x$$

State Table:

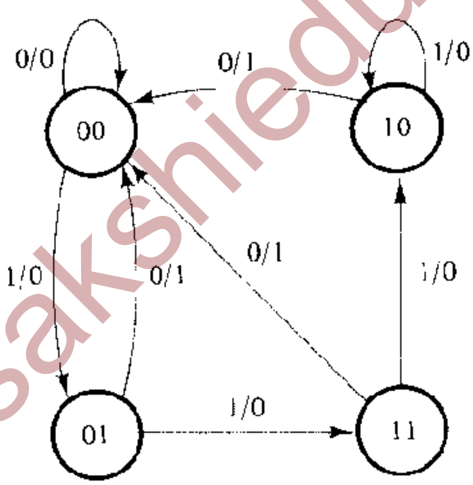
The State Table consists of four sections labeled present state, input, next state, and output. The present state column shows the states of the flip-flops A and B at any given time t. The next state column shows the states of the flip-flops one clock period later at time $t+1$, for given value of x. The output section gives the value of y for each present state. Both the next state and output sections will have two sub columns, one for $x=0$ and the other for $x=1$.

Present State (PS)		**Next State (NS)**				**Output**	
		$x=0$		$x=1$		$x=0$	$x=1$
A	B	A	B	A	B	y	y
0	0	0	0	0	1	0	0
0	1	1	1	0	1	0	0
1	0	1	0	0	0	0	1
1	1	1	0	1	1	0	0

In general, a sequential circuit with m flip-flops, and n input variables, will have 2^m rows, one for each state. The next state and ouput columns will have 2^n columns one for each input combination.

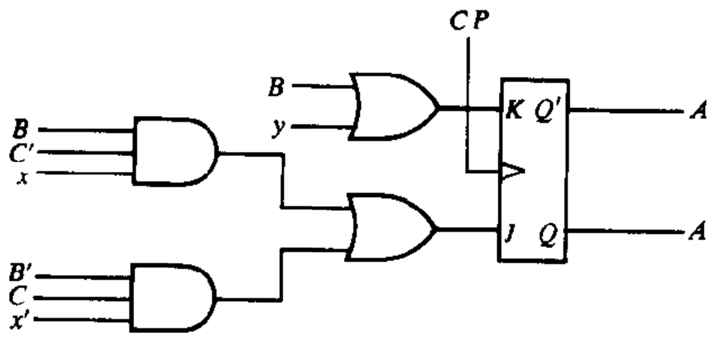
State Diagram:

The information available in a State Table can be represented graphically in a state diagram. In this, a state is represented by a circle, and the transition between states is indicated by directed lines connecting the circles. The state diagram provides the same information as the State Table. The binary number inside each circle represent the state of the flip-flops. The directed lines are labeled with two binary numbers separated by a slash. The input value during the present state is labeled first and th number after the slash gives the output during the present state. A directed line connecting a circle with itself indicates that no change of state occurs.



Flip-flop input functions:

The logic diagram of a sequential circuit consists of flip-flops and gates. The knowledge of the type of flip-flops and a list of the Boolean functions of the combinational circuit provide all the information needed to draw the logic diagram. The part of the combinational circuit that generates external outputs is described algebraically by the circuit output functions. The part of the circuit that generates the inputs to flip-flops are described algebraically by a set of Boolean functions called flip-flop input functions.

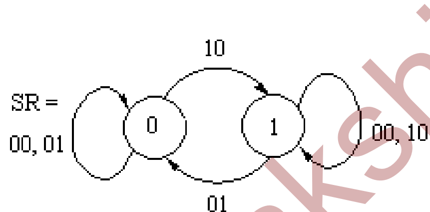


The flip-flop input functions for the above circuit are

$$J_A = BC'x + B'Cx'; \quad K_A = B + y$$

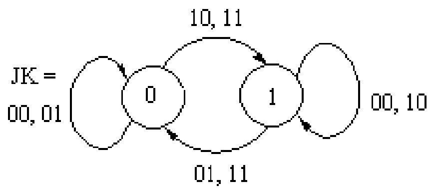
State Diagrams and State Tables of flip-flops:

SR Flip-flop:



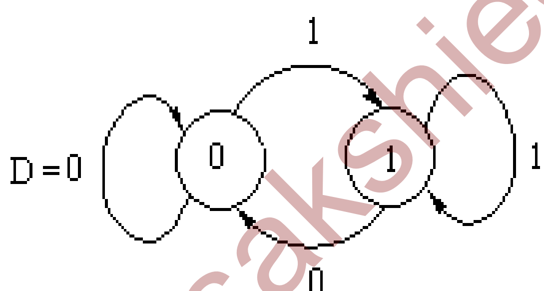
Present State (PS)	Next State (NS)			
	$SR = 00$	$SR = 01$	$SR = 10$	$SR = 11$
0	0	0	1	X
1	1	0	1	X

JK Flip-flop:



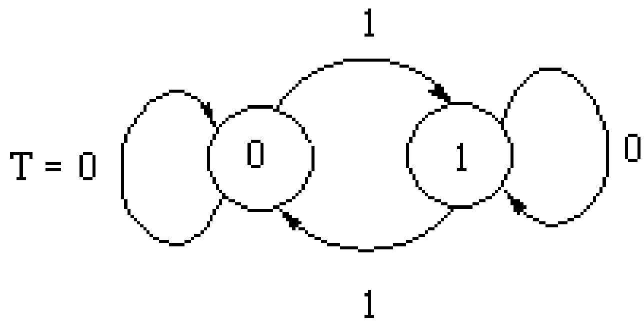
Present State (PS)	Next State (NS)			
	***JK* = 00**	***JK* = 01**	***JK* = 10**	***JK* = 11**
0	0	0	1	1
1	1	0	1	1

D Flip-flop:



Present State (PS)	Next State (NS)	
	***D* = 0**	***D* = 1**
0	0	1
1	0	1

T Flip-flop:



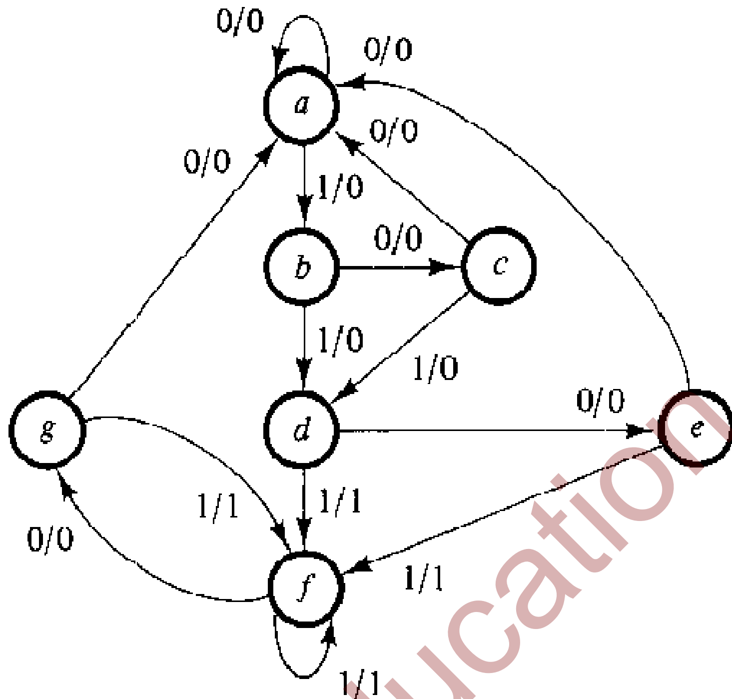
Present State (PS)	Next State (NS)	
	$T = 0$	$T = 1$
0	0	1
1	1	0

State Reduction and Assignment:

Any design process must consider the problem of minimizing the cost of the final circuit. The two most obvious cost reductions are reductions in the number of flip-flops and the number of gates. The reduction of the number of flip-flops in a sequential circuit is referred to as the state reduction problem. Since m flip-flops produce 2^m states, a reduction in the number of states may result in a reduction in the number of flip-flops.

"Two states are said to be equivalent if, for they give exactly the same output and send the circuit either to the same state or to an equivalent state." When two states are equivalent, one of them can be removed.

Example:



State Table for the given state diagram:

PS	Next State (NS)		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

In the above State Table, *e* and *g* are equivalent and hence *g* can be removed and in place of *g*, *e* is placed. Then, again we look for equivalent states and we see that *d* and *f* are equivalent. Hence, the reduced Table will consists of 5 states and requires 3 flip-flops again.

PS	Next State (NS)		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*	*d*	0	1
e	*a*	*d*	0	1

State Assignment:

State assignment procedures are concerned with methods for assigning binary values to states in such a way as to reduce the cost of the combinational circuit that drives the flip-flops.

Three possible binary State Assignments			
State	*Assignment 1*	*Assignment 2*	*Assignment 3*
a	001	000	000
b	010	010	100
c	011	011	010
d	100	101	101
e	101	111	111

Reduced State Table with Binary Assignment 1				
Present State	*Next State*		*Output*	
	x=0	***x=1***	***x=0***	***x=1***
001	001	010	0	0
010	011	100	0	0
011	001	100	0	0
100	101	100	0	1
101	001	100	0	1