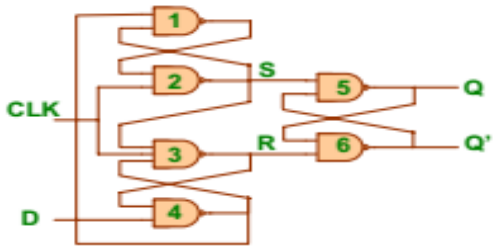
LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN

Positive Edge triggered D Flip-Flop

The positive edge triggered D flip-flop consists of three basic SR latches. This arrangement is shown in the logic diagram.

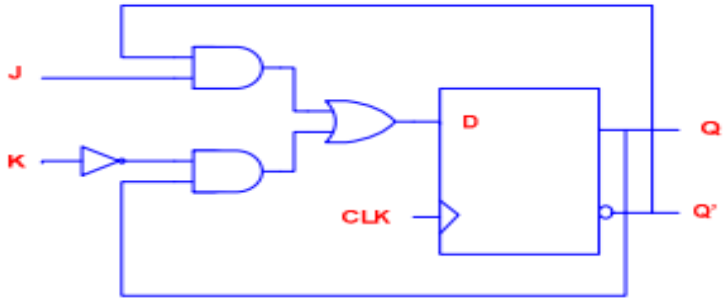


When CLK=0 and D=0, then the outputs of gates 1, 2, 3, and 4 are going to be 0111. If D=1 while CLK=0, then the outputs of these gates are going to be 1110 instead. In both cases, S=R=1, which make Q and Q' are 01 or 10. i.e. one of the two stable states.

Suppose that CLK becomes 1 while D=0. The output of gate 3 (which is R) becomes 0. this will reset the output flip-flop. Once R is 0, then D can change to 1 and R remains 0. This means that Q remains 0 while CLK is 1. No change will occur to Q until the clock returns to 0 and then goes to 1 on the next clock pulse. This scenario shows that the flip-flop is a positive edge triggered.

Similar procedure occurs if D=1 while the clock goes from 0 to 1. In this case, S=0 and Q=1.

JK Flip-Flop



The D input is given by: $D = JQ' + K'Q$

The characteristic equation of the JK flip-flop is:

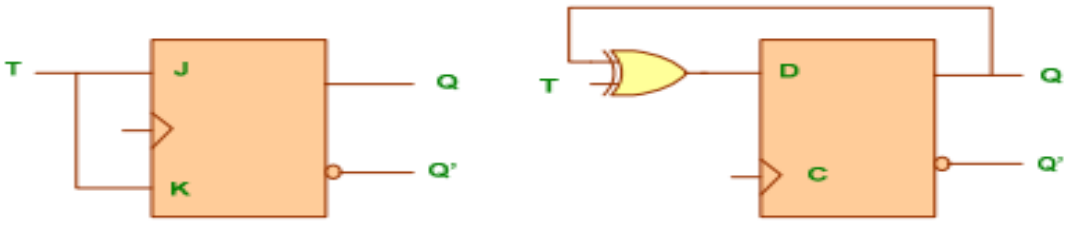
$$Q_{t+1} = JQ' + K'Q$$

The characteristic table of the JK flip-flop is given below.

J	K	Q_{t+1}	
0	0	Qt	No change
0	1	0	Reset
1	0	1	Set
1	1	Q_t'	Complement

T Flip-Flop

A T flip-flop is obtained from the JK flip-flop by connecting J and K together to form a single input T. The T flip-flop is a toggling (complementing) one if T is kept at 1. It can also be obtained from a D flip-flop and an exclusive-OR gate as shown in the Figure.



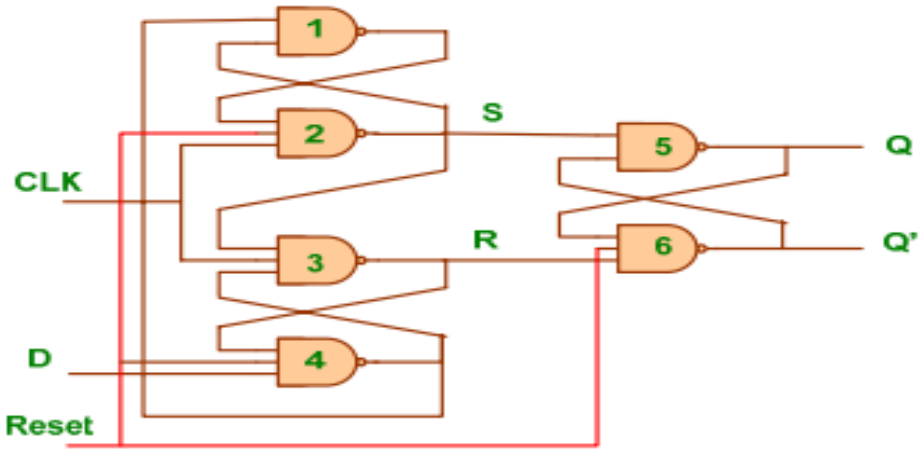
The characteristic equation of the T flip-flop is given by:

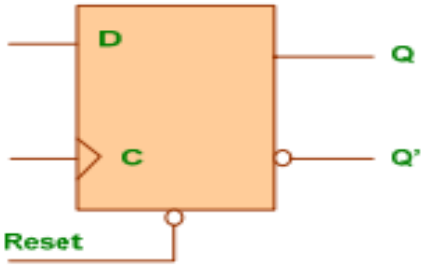
$$Q_{t+1} = T \oplus Q = TQ' + T'Q$$

Direct Inputs

These are asynchronous inputs that can reset or set the flip-flops at any time without the application of clock pulses.

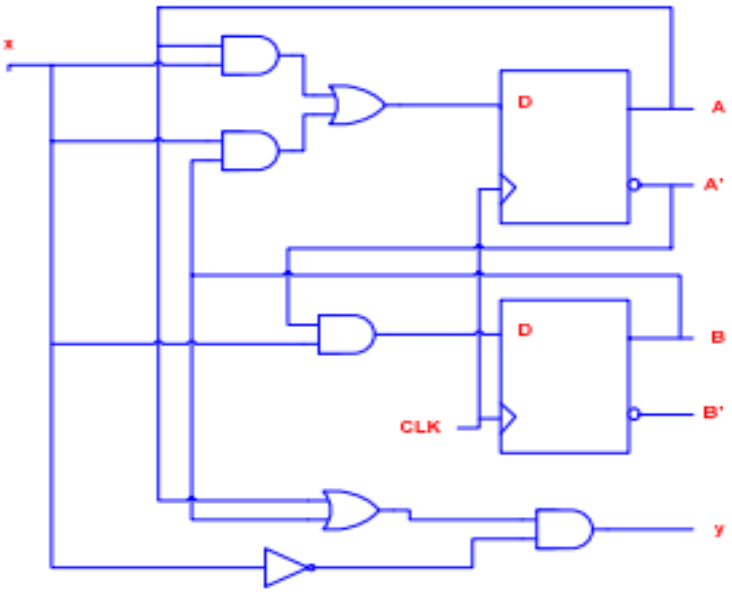
LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN





Reset	C	D	Q	Q'
0	X	X	0	1
1	↑	0	0	1
1	↑	1	1	0

Analysis of Clocked Sequential Circuits



LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN

$$A(t+1) = Ax + Bx$$
$$B(t+1) = A'x$$
$$y = (A+B)x'$$

State Table →

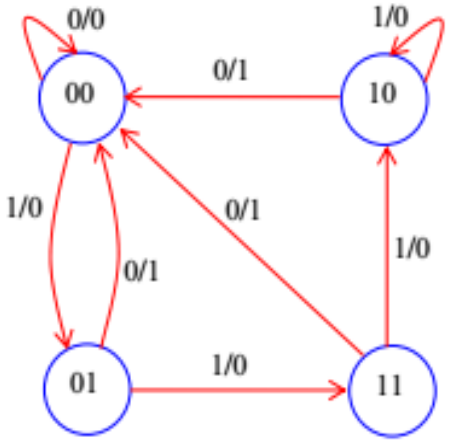
P.S.		Input	N.S.		Output
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

The state table can also be put in the following different form.

P.S.		N.S.				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	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0

State Diagram

The information in the state table can be presented pictorially in the form of a state diagram. A state is represented by a circle. A transition from present state to next state is represented by an arrow with the inputs and outputs written on that transition with a slash in between inputs and outputs. The state diagram of the previous sequential circuit is given below.



LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN

Analysis with JK flip-flops

It is easy to obtain the state table and diagram of sequential circuits with D flip-flops, because the state equation is the same as the input equation

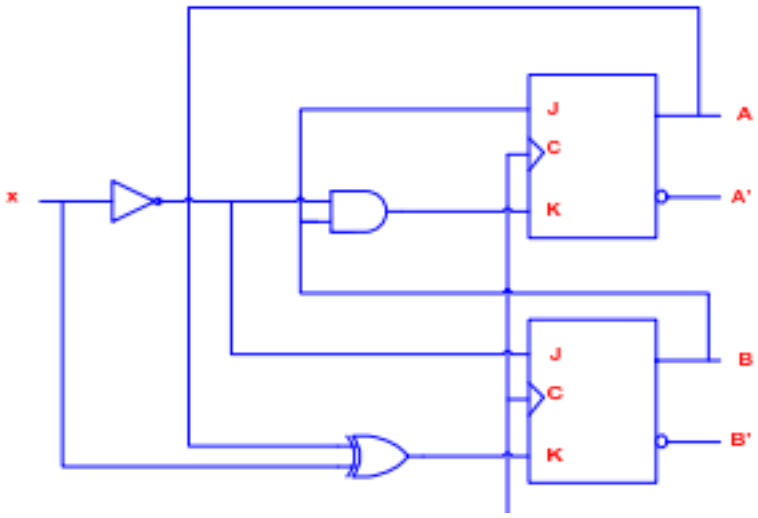
Example: $D_A = Ax + Bx$ → $A_{(t+1)} = Ax + Bx$

When other types of flip-flops are used such as JK and T, then we have to use the characteristic table or characteristic equation to obtain the next state.

The analysis procedure in this case consists of:

1. Determine the flip-flop input equations in terms of the present state and input variables.
2. List the binary values of each input equation.

3. Use the corresponding flip-flop characteristic table to determine the next state values.



LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN

The flip-flop input equations are given by:

$J_A = B, \quad K_A = Bx', \quad J_B = x', \quad$ and $K_B = A \oplus x$

The flip-flop inputs are obtained first and then the next state can be found using the JK flip-flop characteristic table.

P.S.		Input	N.S.		Flip-flop Inputs			
A	B	x	A	B	J_A	K_A	J_B	K_B
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	1	1	1	1	1	0
0	1	1	1	0	1	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	1
1	1	1	1	1	1	0	0	0

The state table could also be completed by using the flip-flop input equations and substituting it in the characteristic equation for the flip-flop. Then the next state columns can be filled.

The characteristic equation of the JK flip-flop is given by:

$$Q_{t+1} = JQ' + K'Q$$

Replacing Q with A and B, produces the following two equations:

1. $A_{t+1} = J_A A' + K'_A A$

2. $B_{t+1} = J_B B' + K'_B B$

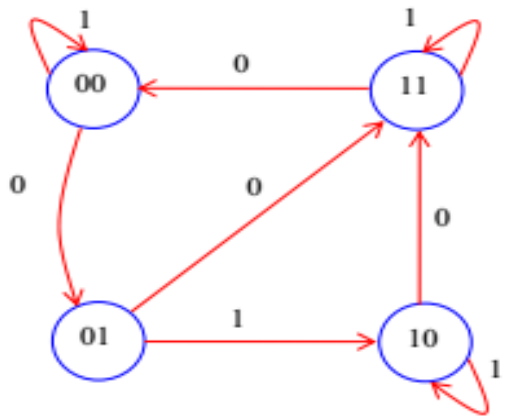
Substituting for J_A, K_A, J_B, and K_B:

$$A_{t+1} = BA' + (Bx')'A = A'B + AB' + Ax$$

and $B_{t+1} = x'B' + (A \oplus x)'B = B'x' + ABx + A'B'x'$

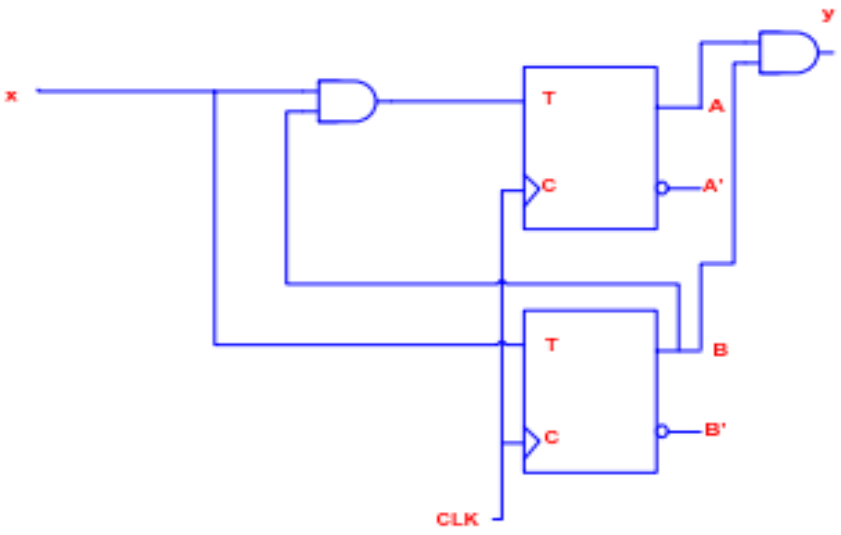
The next state columns can then be filled using these last equations.

The sate diagram of the previous circuit is given next.



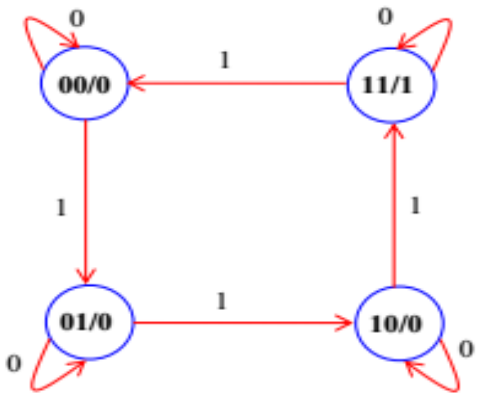
Analysis with T flip-flops

The analysis of a sequential circuit with T flip-flop follows the same procedure as for circuits with JK flip-flops.



$T_A = Bx$, $T_B = x$, and $y = AB$

LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN



Mealy and Moore Models

The Mealy model is characterized by the outputs being a function of the present state and the inputs. The first circuit we analyzed with D flip-flops is an example of a Mealy model. On the other hand, the Moore model is characterized by the outputs being a function of the present state only. Examples are the circuits we analyzed with JK and T flip-flops.

The state table for the previous circuit can be obtained either by using the characteristic table or writing the characteristic equations for the T flip flops.

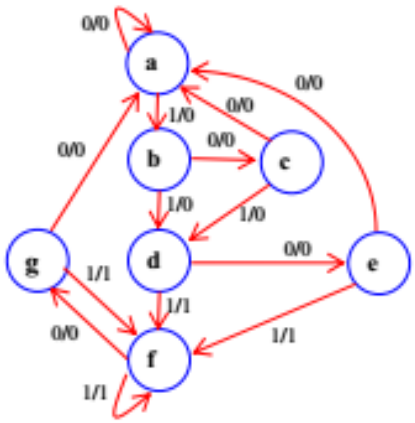
P.S.		Input	N.S.		Output	f/f Inputs	
A	B	x	A	B	y	TA	TB
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	1
1	0	0	1	0	0	0	0
1	0	1	1	1	0	0	1
1	1	0	1	1	1	0	0
1	1	1	0	0	1	1	1

LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN

State reduction

In the design of sequential circuits, we need to reduce the number of flip flops and the number of logic gates used in the combinational circuit part. Reduction of the number of flip-flops may result from the reduction of the number of states in the circuit. This is possible if we are interested in the input output relationship of the circuit and not in the outputs of the flip-flops.

The state reduction procedure will be illustrated with an example. Consider the given state diagram



Suppose that the input sequence to the circuit is 01010110100 starting from the initial state a. Then the output and next state will be as follows:

State	a	a	b	c	d	e	f	f	g	f	g	a
Input	0	1	0	1	0	1	1	0	1	0	0	
Output	0	0	0	0	0	1	1	0	1	0	0	

The pattern is as shown here: → P.S. Input Output → N.S.

LECTURE 11: DIGITAL LOGIC CIRCUIT DESIGN

The state table can be arranged as follows:

P.S.	N.S.		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

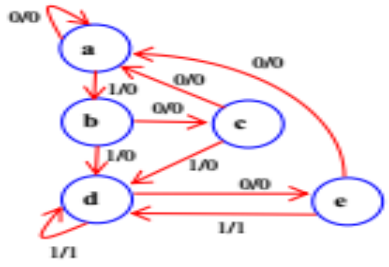
States g and e are identical then we can cancel state g and substitute e for g in the table. This will reduce the states from 7 to 6. Next, we find that states d and f are identical. Then we cancel f and replace it in the table with d. No further reduction is possible.

The number of states has been reduced from 7 to 5. No reduction in the number of flip-flops is possible, because we need 3 flip-flops for 5 states, which is he same number of flip-flops for 7 states.

The reduced state table will be as follows:

P.S.	N.S.		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

And the reduced state diagram will be as follows:



The input sequence will result in the same output sequence using the reduced state table or diagram as shown in the following table.

State	a	a	b	c	d	e	d	d	e	d	e	a
Input	0	1	0	1	0	1	1	0	1	0	0	
Output	0	0	0	0	0	1	1	0	1	0	0	

The reduction of the number of states could lead to reduction of the number of flip-flops or it might not lead to it. WSe started in the previous example with seven states, which need 3 flip-flops in the design of the circuit. We managed to reduce the number of states to five. The number of flip-flops needed is still 3.

State assignment

We need to assign coded binary values to the different states in the state table. In the previous example, we have 5 states and three flip-flops. We must choose five combinations out of the total of 8 possible combinations. 3 combinations will not be used and can be considered don't care conditions. Examples of state assignments are given in the following table.

State	Assign.1	Assign. 2	Assign.3
a	000	000	011
b	001	001	100
c	010	011	101
d	011	010	110
e	100	110	111