Unit 7: Sequential Logic

Presented By: Er. Ganesh Kumal

Contents

- Event driven model and state diagram
- Flip-flops and their types
- Analysis of clocked sequential circuits
- Decoder as memory devices
- State reduction and assignment
- Synchronous and asynchronous logic
- Edge triggering devices
- Master slave flip-flops
- JK and T flip-flops

Sequential circuit

- Combinational circuit with feedback element (memory).
- Outputs depend upon present inputs and previous outputs (present states).

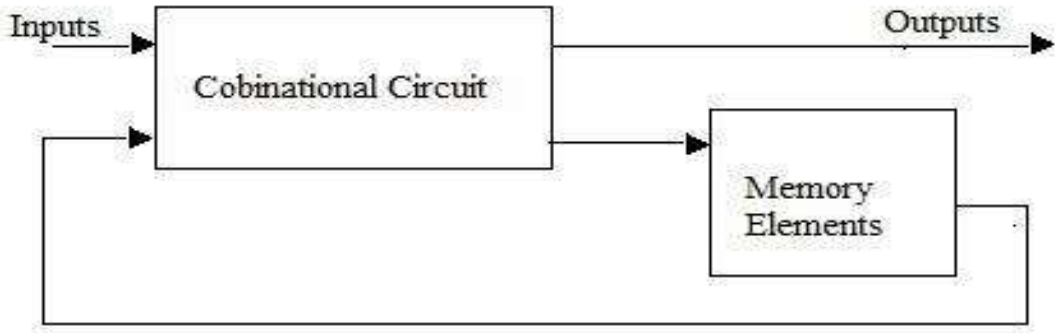


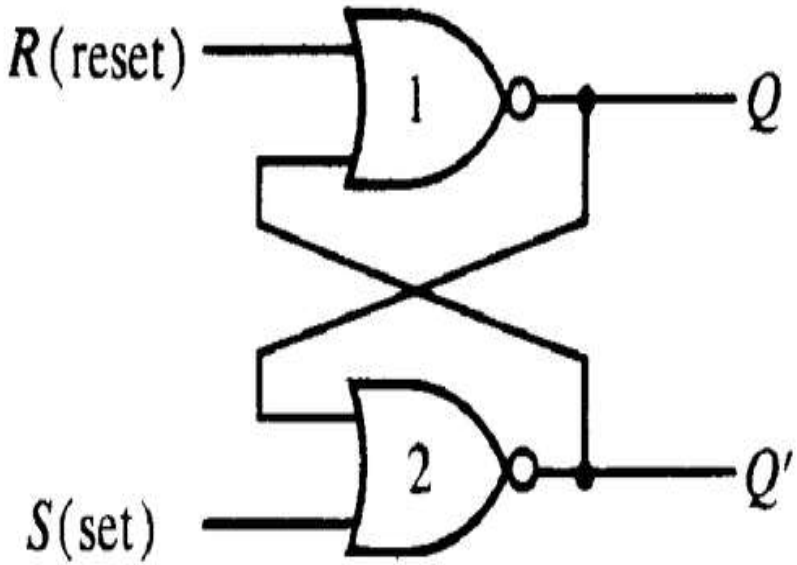
Figure 6.1: Block diagram of sequential circuit.

Memory elements are devices capable of storing binary information. The binary information stored in the memory elements at any given time defines the state of the sequential circuit.

Basic Flip-Flop (Latch)

- Single bit storage device (memory unit).
- It latches ‘0’ or ‘1’.
- Basic Flip-flop circuit can be constructed from two NAND gates or two NOR gates.
- The cross-coupled connection from the output of one gate to the input of the other gate constitutes a feedback path.
- Each latch has two outputs (Q and Q’) and two inputs Set (S) and Reset (R).
- This type of flip-flop is sometimes called a direct-coupled RS flip-flop, or RS latch.

Basic Flip-Flop with NOR Gates



(a) Logic diagram

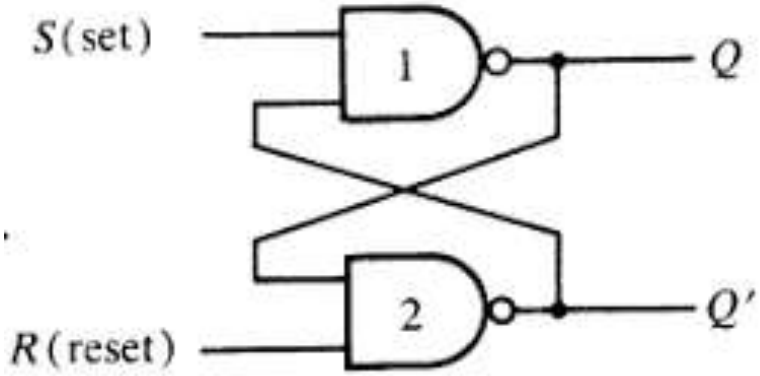
S	R	Q	Q′	
1	0	1	0	
0	0	1	0	(after $S = 1, R = 0$)
0	1	0	1	
0	0	0	1	(after $S = 0, R = 1$)
1	1	0	0	

(b) Truth table

When,
S=0, R=0 ; No change state
S=0, R=1 ; Reset
S=1, R=0 ; Set
S=1, R=1; Invalid

Figure 6.2: NOR Latch

Basic Flip-Flop with NAND Gates



(a) Logic diagram

S	R	Q	Q'	
1	0	0	1	
1	1	0	1	(after $S = 1, R = 0$)
0	1	1	0	
1	1	1	0	(after $S = 0, R = 1$)
0	0	1	1	

(b) Truth table

Figure 6.3: NAND latch

Clock Pulse

- In the latches and flip-flips, we use the additional signal called clock.
- Digital circuits are invariably controlled by a ***clock*** and events take place at discrete points in time.
- The clock is a circuit that provides a sequence of pulses to trigger each internal operation.

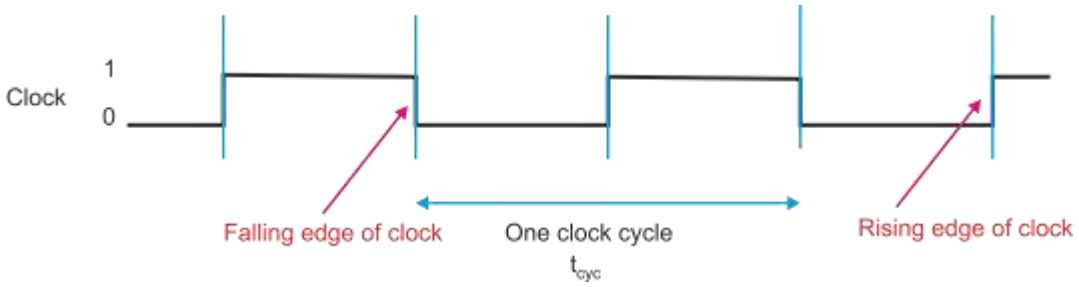
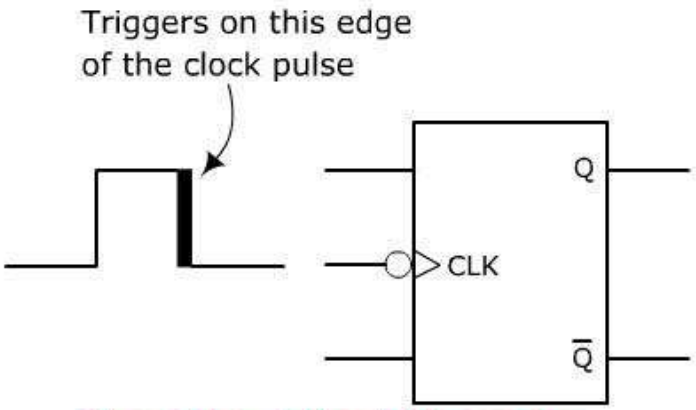


Figure 6.11: Perfect idealized clock

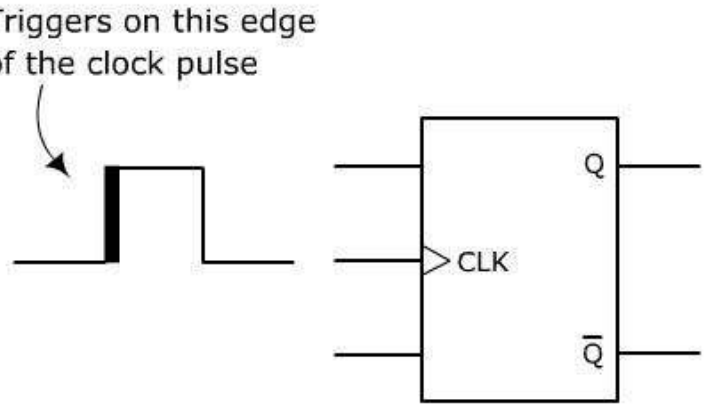
Positive and Negative Edge Triggering

- Depending on which portion of the clock signal the flip-flop responds to, we can classify them into two types:
  1. Level Triggering
     i. Positive level triggering
     ii. Negative level triggering
  2. Edge triggering
     i. Positive (rising) edge triggering
     ii. Negative (falling) edge triggering

CONT…

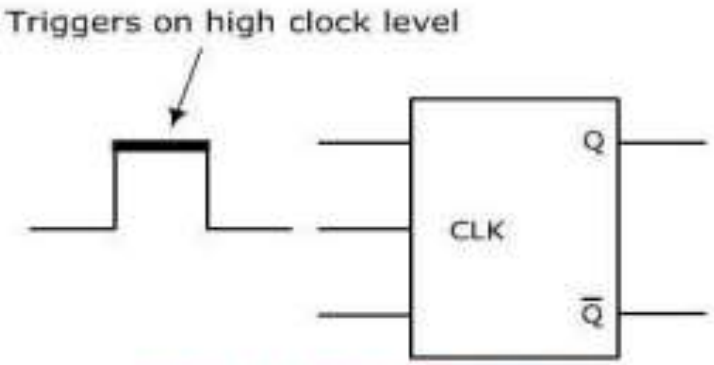


Negative Edge Triggering

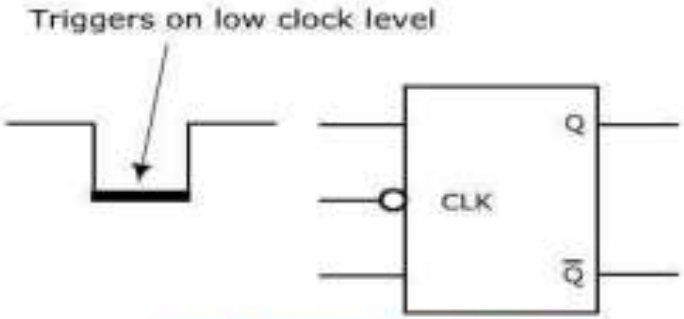


Positive Edge Triggering

Edge Triggering



High Level Triggering



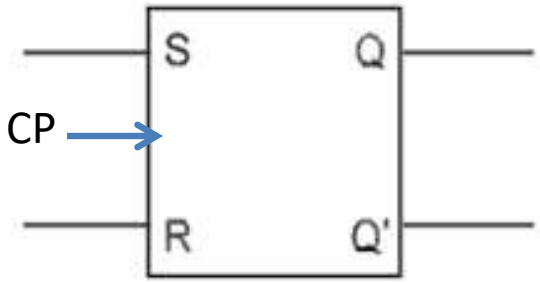
Low Level Triggering

Level Triggering

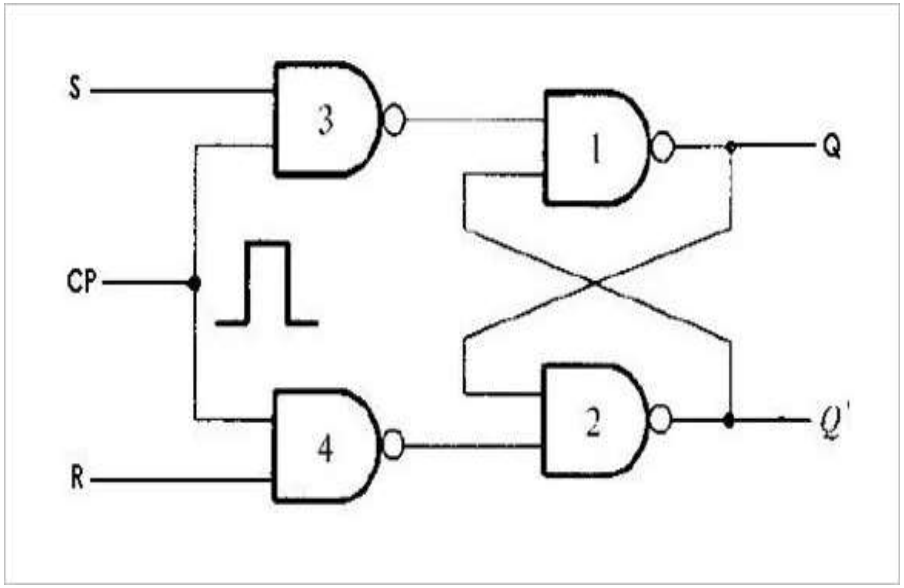
Flip-Flop

- Sequential circuit.
- Single bit memory element (storage device).
- Bi-stable device (either ‘1’ or ‘0’).
- Example: S-R flip-flop, D-flip-flop, J-K flip-flop, Master-Slave Flip-Flop and T-flip-flop.

Clocked RS Flip-Flop



(a) Block diagram



(b) Logic diagram

Inputs			Outputs	
CP	S	R	Q	Q’
X	0	0	No change	
1	0	1	0	1
1	1	0	1	0
1	1	1	Invalid	

(c) Truth table

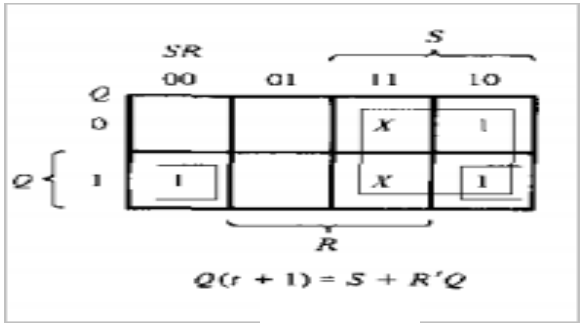
Figure 6.4: RS Flip-Flop

Operation of S-R flip-flop

- When CP=0, flip-flop remains in previous state
- When CP=1 and
  - i. S = 0, R = 0 ; flip-flop holds the data (No change).
  - ii. S = 0, R = 1 ; flip-flop reset irrespective of Q
  - iii. S = 1, R = 0; flip-flop set irrespective of Q .
  - iv. S =1, R =1; flip-flop is invalid state

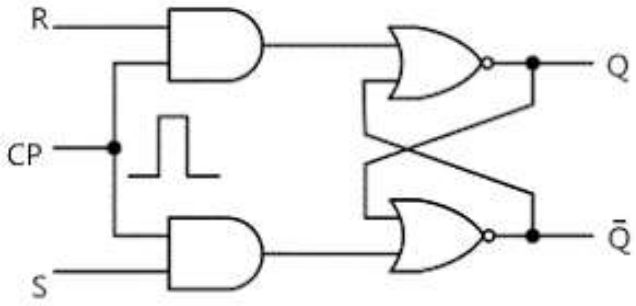
S	R	Q(t)	Q (t +1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X (Indeterminate)
1	1	1	X (Indeterminate)

Figure: Characteristic table

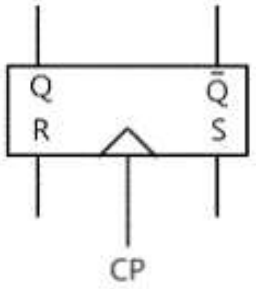


Characteristic equation

Clocked RS Flip-Flop



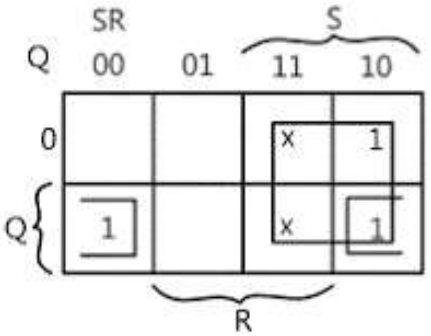
(a) Logic Circuit



(b) Symbol

Q	S	R	Q(t+1)	
0	0	0	0	
0	0	1	0	
0	1	0	1	
0	1	1		Indeterminate
1	0	0	1	
1	0	1	0	
1	1	0	1	
1	1	1		Indeterminate

(c) Characteristic Table



$Q(t+1)=S+\bar{R}Q$

(b) Characteristic Equation

Figure 6.5: RS flip-flop by NOR latch

Timing diagram of S-R F/F

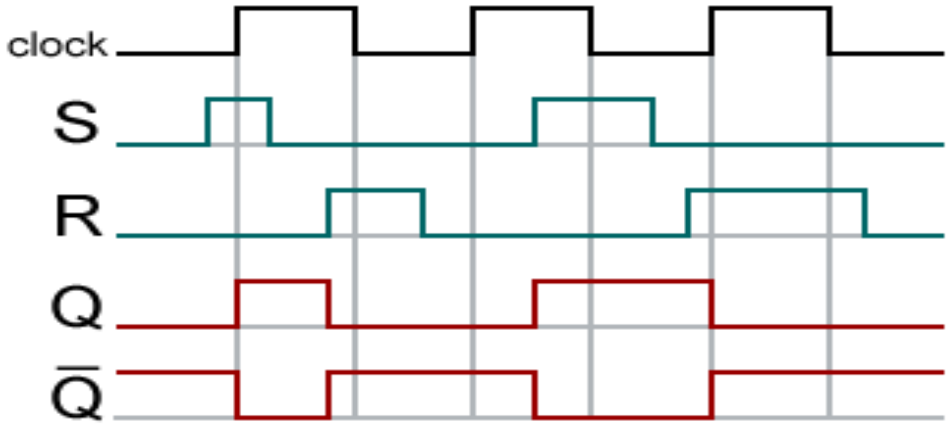


Figure: Level Triggering

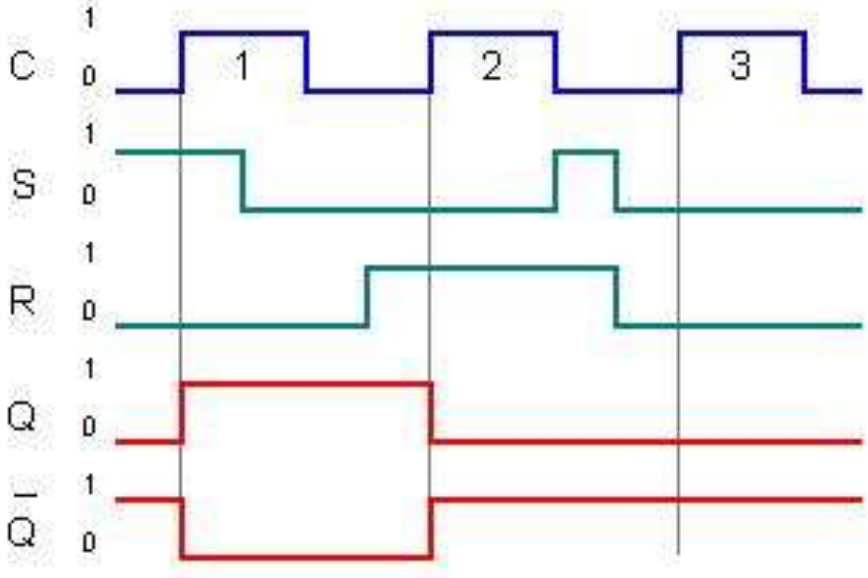


Figure: Edge Triggering

D Flip-Flop

- One way to eliminate the undesired condition of the indeterminate state in the RS flip-flop is to ensure that input S and R are never equal to 1 at the same time.
- This is done in the D flip-flop as shown in figure 6.6.
- S input is inverted and given to R input to make D flip-flop from S-R flip-flop.
- Used in registers, shift registers and some of the counters.

D Flip-Flop

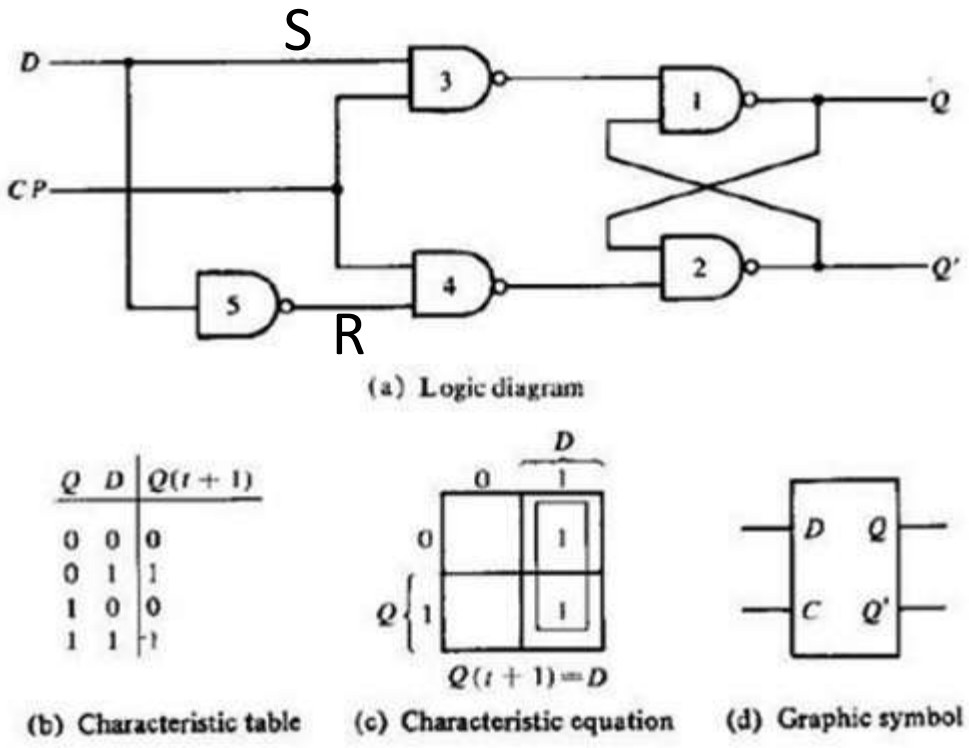


Figure 6.6: D Flip-Flop

Operation:

- As long as CP=0, the output of gate 3 and 4 are high and the circuit cannot change the state regardless of the value of D.
- When CP=1, next state of D flip-flop is always equal to data input, D.

Invalid state is removed in D flip-flop.

Advantage: Invalid state is never happened.

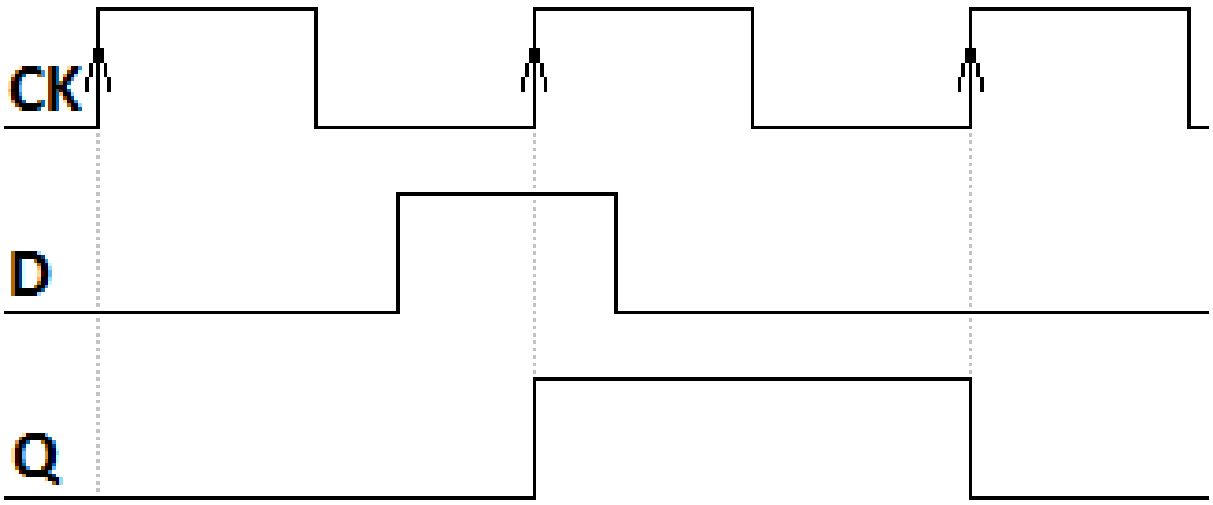
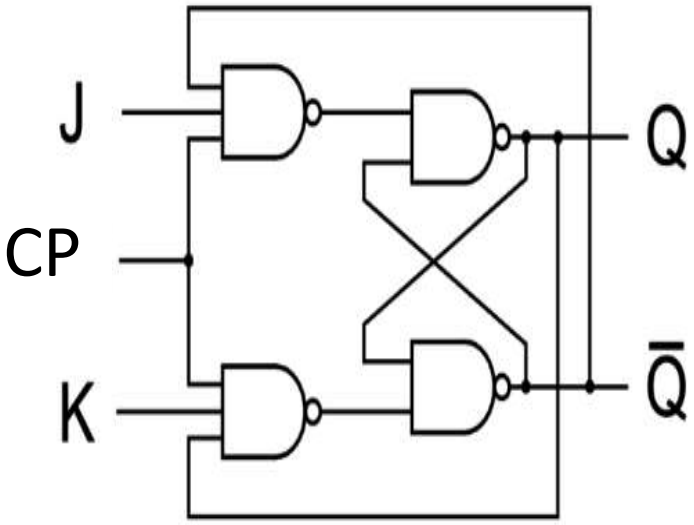


Figure: Timing diagram of D flip-flop (positive edge triggering)

Clocked J-K Flip-Flop

- J-K flip-flop is a refinement of the R-S flip-flop.
- The indeterminate state of the R-S flip-flop is defined in the J-K flip-flop (when S=1 and R= 1).
- We consider the inputs of S-R flip-flop as S = JQ’ and R = KQ.



(a) Logic diagram

Inputs			Outputs	
CP	J	K	Q	Q’
X	0	0	No change	
1	**0**	**1**	**0**	**1**
1	1	0	1	0
1	1	1	Toggle	

(b) Truth table

Figure 6.7: J- K Flip - Flop

J-K Flip-Flop

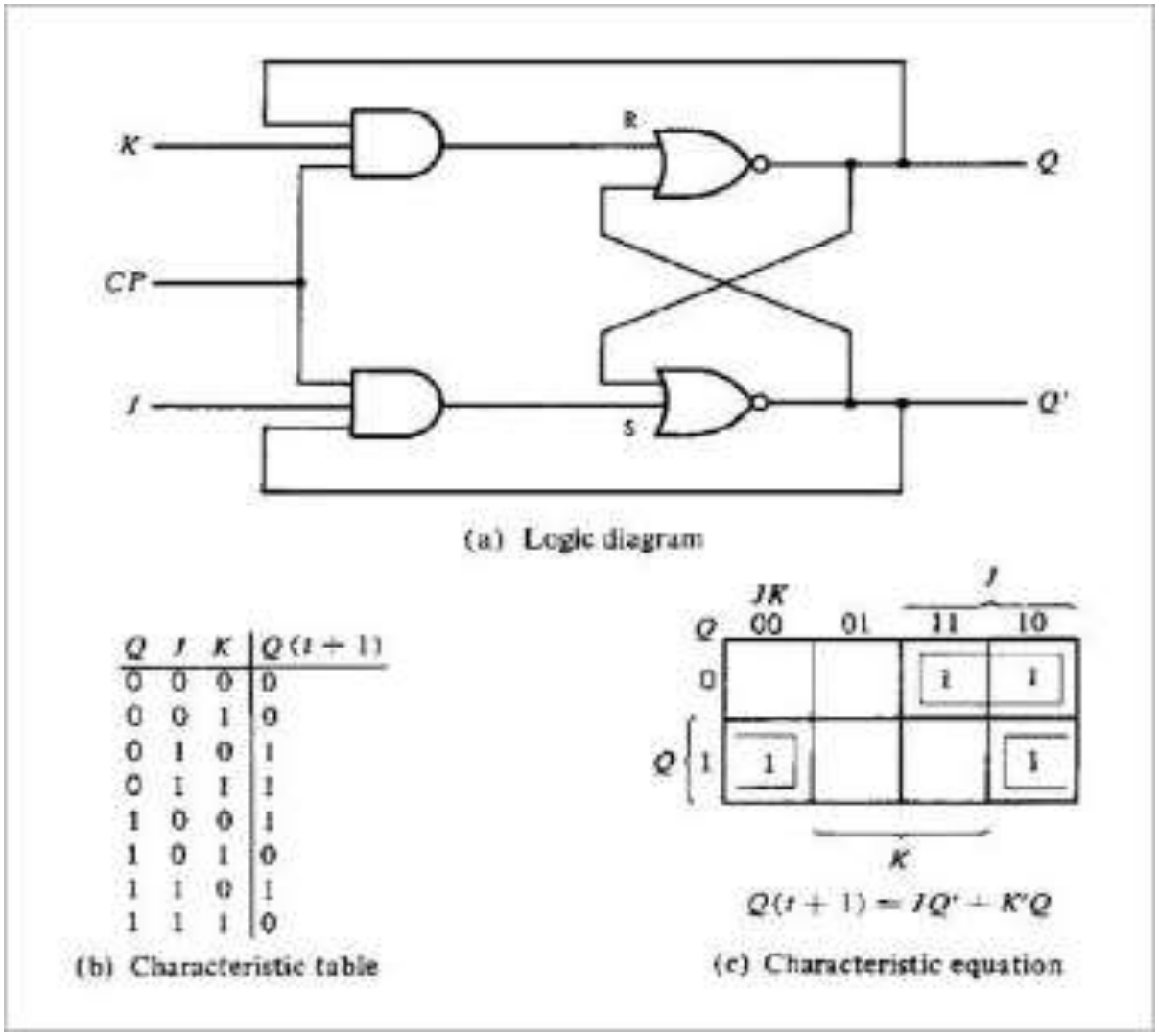


Figure 6.8: J-K Flip-Flop

Race around condition

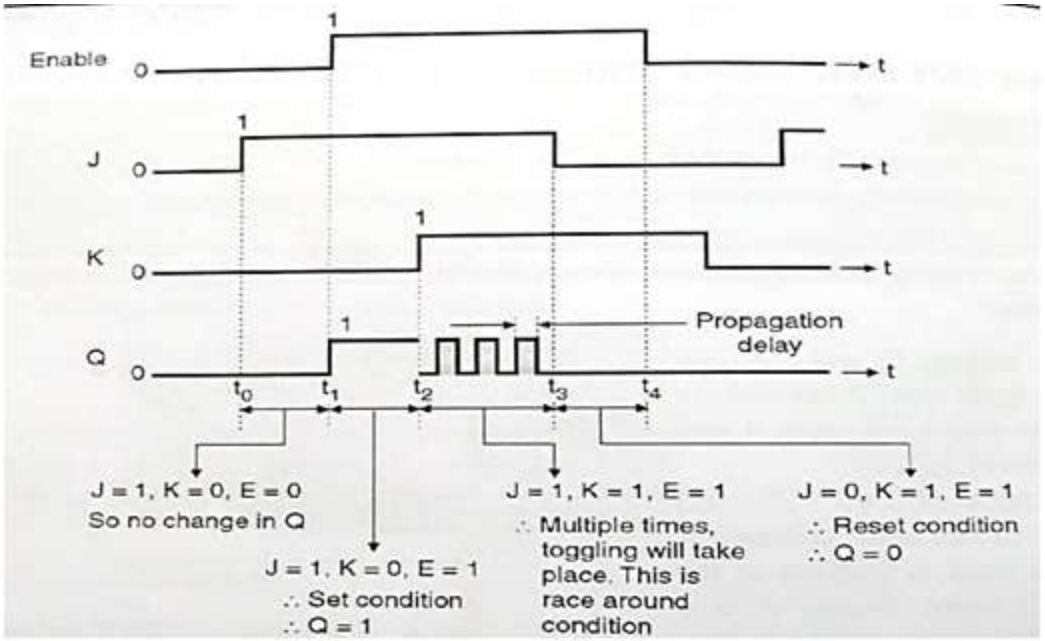
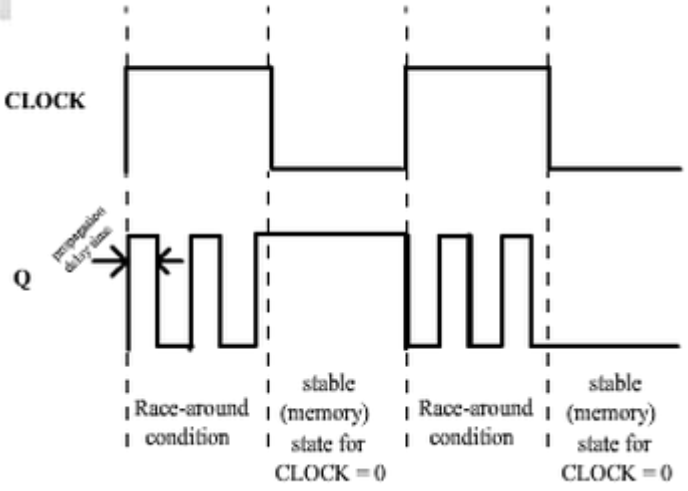


Fig5. Waveform showing Race around condition.



How to avoid race around condition??

i. Using Master-Slave J-K flip-flop.
ii. Using the Edge triggering flip-flop
iii. T/2 < propagation delay for level triggering

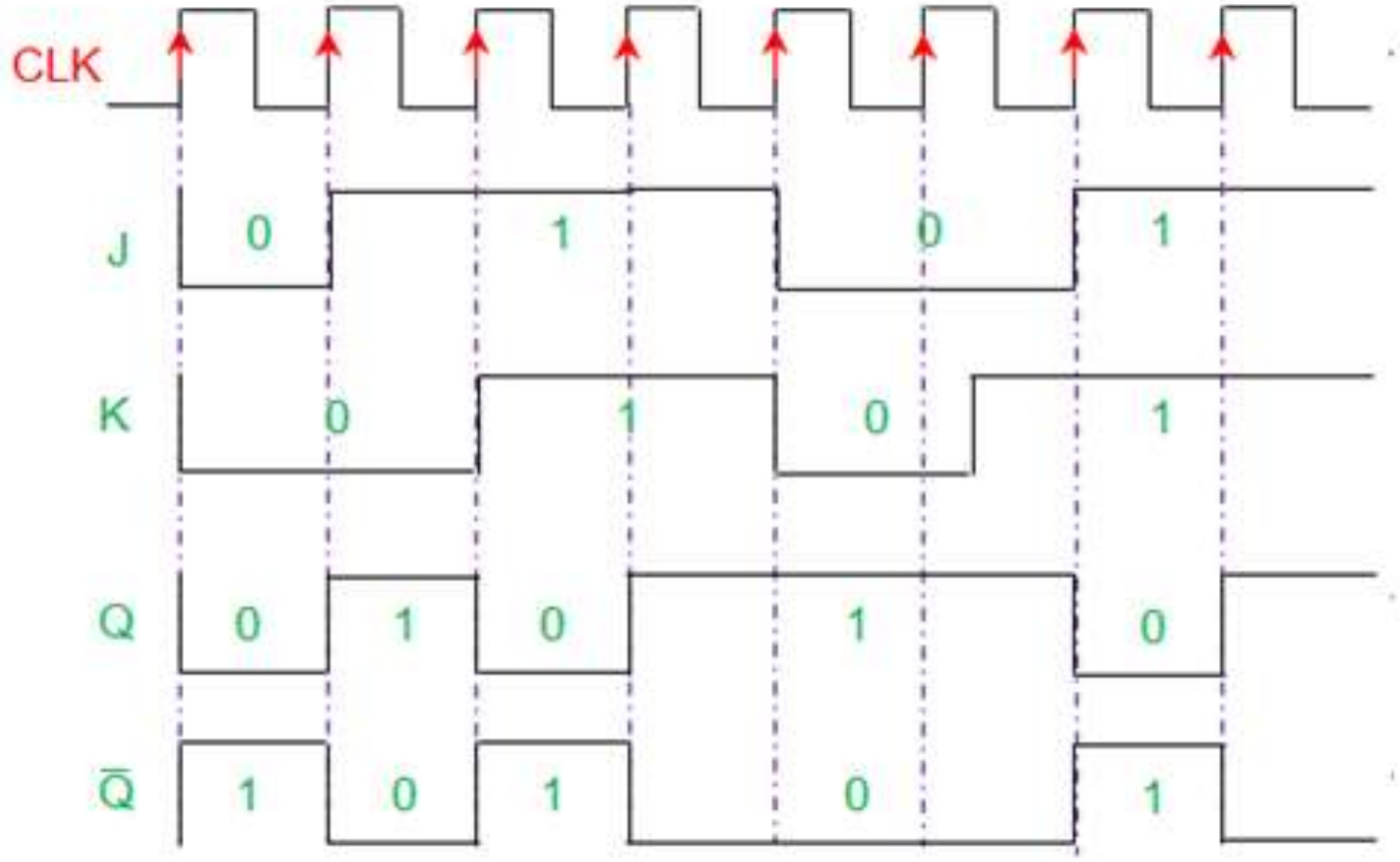
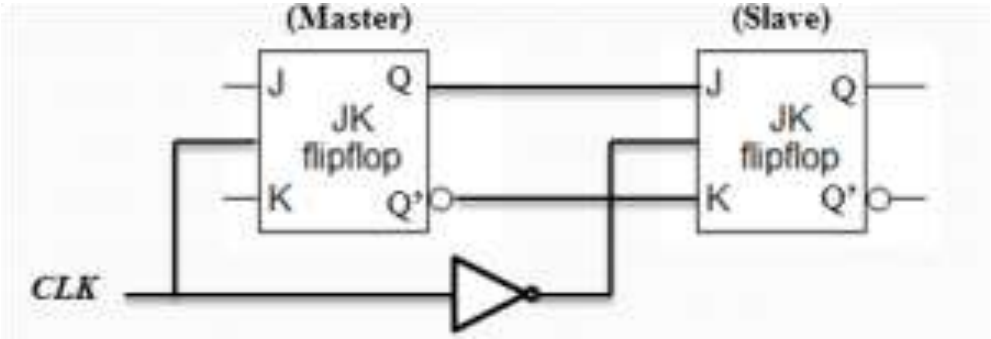


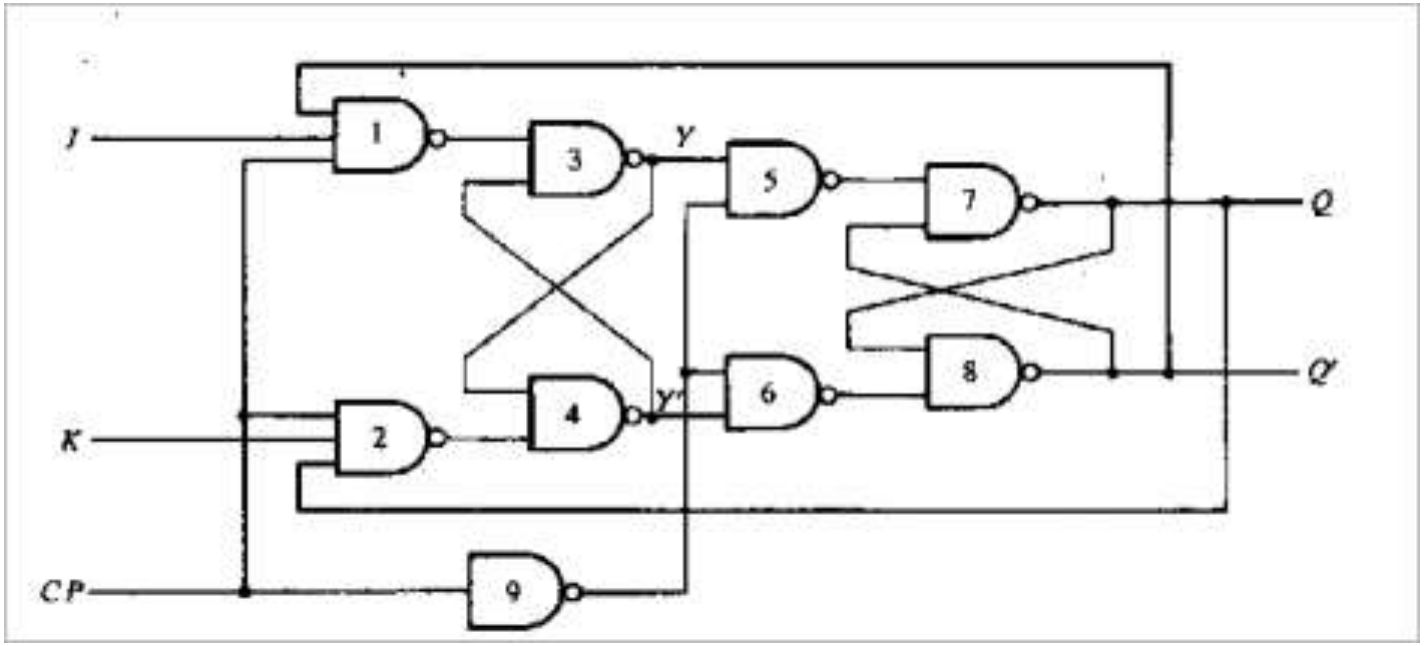
Figure: Time diagram of Edge triggered J-K flip-flop

J-K Master Slave F/F

- Master-Slave J-K flip-flop is constructed from two separate flip-flops. One circuit serves as a master and the other as a slave and the overall circuit is referred to as a master slave flip-flop.
- The master section is basically a gated latch and slave is also the same except that it is clocked an inverted clock pulse and is controlled by outputs of master section rather than by external J-K inputs.
- Characteristic table is same as J-K Flip-flop.



(a) Block diagram

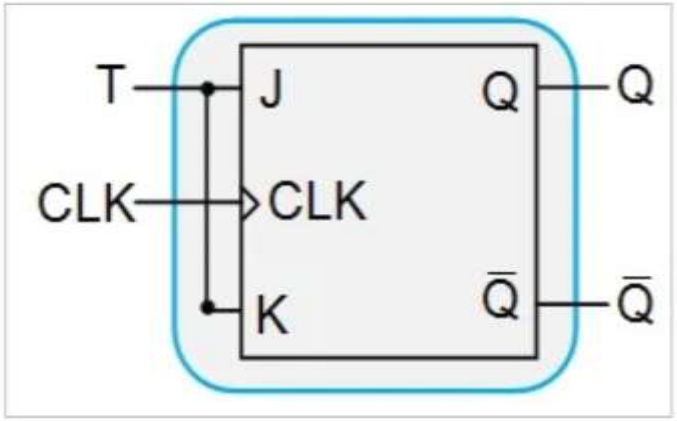


(b) Logic diagram

Figure 6.9: Master Slave J-K flip-flop.

T Flip-Flop

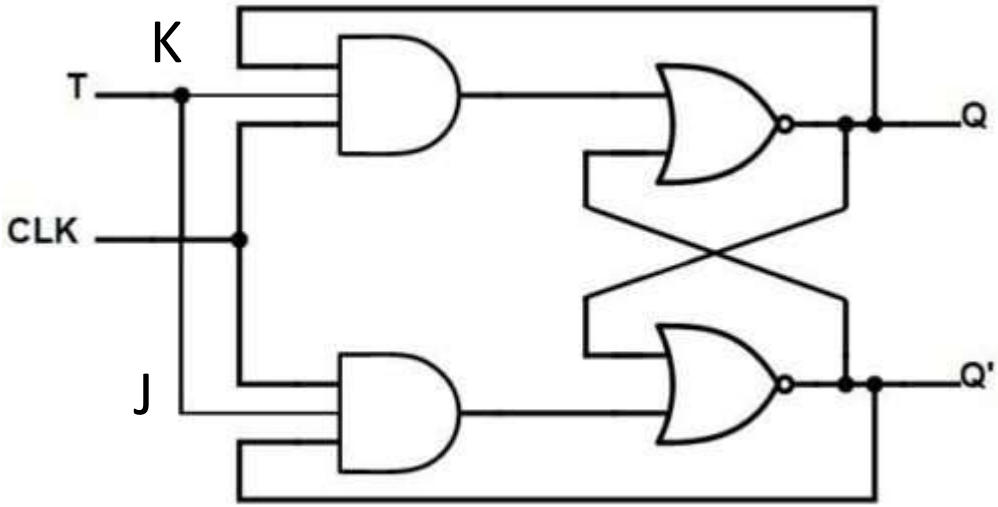
- It is obtained by connecting J and K inputs together; so only one input denoted by 'T'.
- There is only two input conditions
  J = K = 1
  J = K = 0
- T flip-flop is used for counter.



(a) Block diagram

Q(t)	T	Q(t + 1)
0	0	0
0	1	1
1	0	1
1	1	0

(C) Characteristic table



(b) Logic diagram

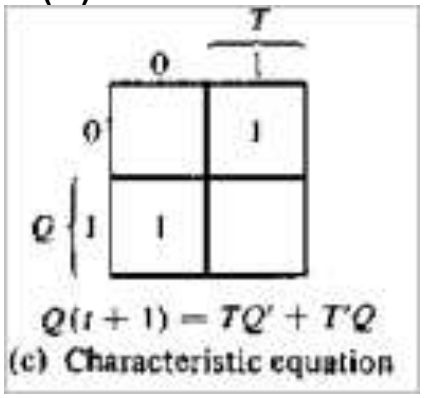


Figure 6.10: T Flip-Flop

Characteristic Table

(a) *JK* Flip-Flop

J	K	$Q(t+1)$	Operation
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	$\overline{Q}(t)$	Complement

(b) *SR* Flip-Flop

S	R	$Q(t+1)$	Operation
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	?	Undefined

(c) *D* Flip-Flop

D	$Q(t+1)$	Operation
0	0	Reset
1	1	Set

(d) *T* Flip-Flop

T	$Q(t+1)$	Operation
0	$Q(t)$	No change
1	$\overline{Q}(t)$	Complement

Excitation Table

A table that lists required inputs for a given change of state (present to next state) is called an excitation table.

Excitation Table for 4 Flip-Flops

SR flip-flop

$Q(t)$	$Q(t+1)$	S	R
0	0	0	×
0	1	1	0
1	0	0	1
1	1	×	0

D flip-flop

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

JK flip-flop

$Q(t)$	$Q(t+1)$	J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

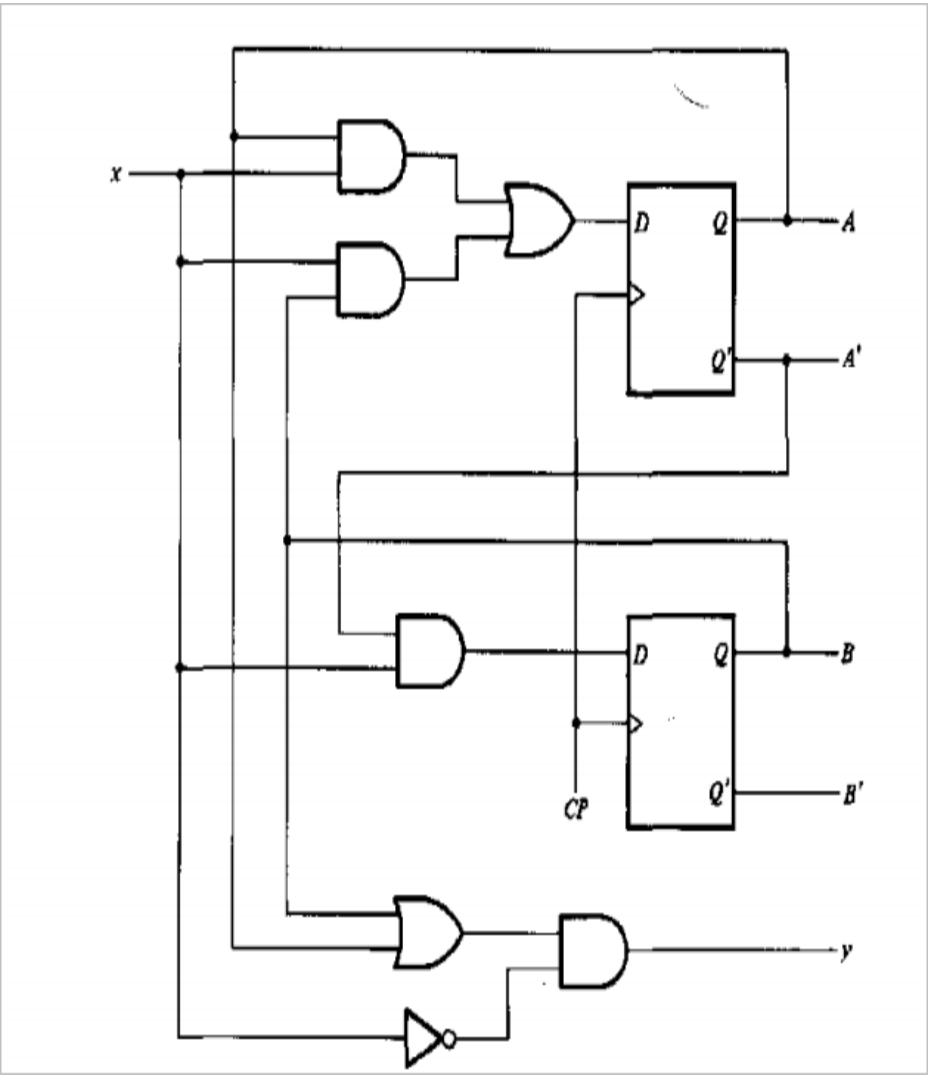
T flip-flop

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

Analysis of Clocked Sequential Circuit

- The behavior of a sequential circuit is determined from the inputs, the outputs, and the state of its flip-flops.
- The outputs and the next state are both a function of the inputs and the present state.
- The **analysis of a sequential circuit** consists of obtaining a table or a diagram for the time sequence of inputs, outputs, and internal states. It is also possible to write Boolean expressions that describe the behavior of the sequential circuit.
- A logic diagram is recognized as a clocked sequential circuit if it includes flip-flops. The flip-flops may be of any type and the logic diagram may or may not include combinational circuit gates.

Sequential Circuit Example



State Equation

- It is an algebraic expression that specifies the condition for a flip-flop state transition.
- The circuit consists of two *D* flip-flops *A* and *B,* an input *x,* and an output *y.*
- Since the *D* inputs determine the flip-flops' next state, it is possible to write a set of next-state equations for the circuit:

$$A(t + 1) = A(t)x(t) + B(t)x(t)$$

$$B(t + 1) = A'(t)x(t)$$

- The previous equations can be expressed in more compact form as follows:

$$A(t + 1) = Ax + Bx$$

$$B(t + 1) = A'x$$

Figure 6.17: Sequential circuit example

Similarly, the present-state value of the output can be expressed algebraically as follows:

$y(t) = [A(t) + B(t)]x'(t)$

Removing the symbol *(t)* for the present state. we obtain the output Boolean function:

$y = (A + B)x'$

State Table

The time sequence of inputs, outputs, and flip-flop states (present state and next state) can be enumerated in a *state table.*

The table consists of four sections:

Present state: Shows the states of flip-flops A and B at any given time t.

Input: Gives a value of X for each possible present state.

Output: Gives the value of Y for each present state

Next state: Shows the state of flip-flops after one clock pulse

Next state and output column is derived from the **state equations**.

$A(t + 1) = AX + BX$

$B(t + 1) = A'X$

$y = (A + B)X'$

State table for the circuit of figure 6-17

Present State		Input	Next State		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

Second form of the state table

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

State Diagram

- The information available in a state table can be represented graphically in a state diagram(or vice-versa).
- state diagram is a pictorial representation of the relationship between present state, input, next state and output of sequential circuit.
- **In** this type of diagram, a state is represented by a circle, and the transition between states is indicated by directed lines connecting the circles.

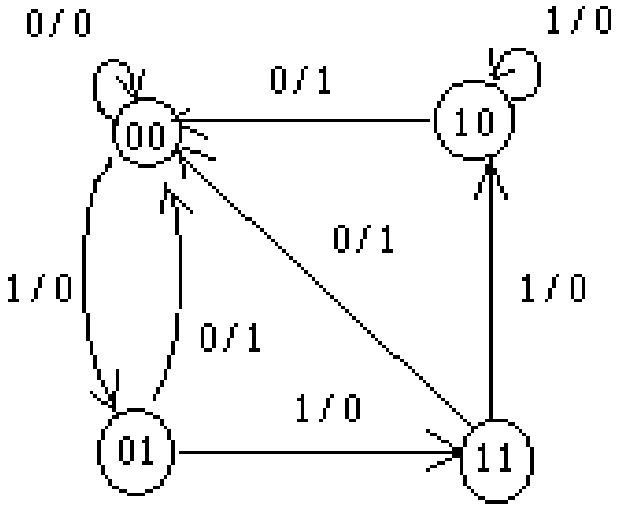


Figure: State diagram of circuit of figure 6.17

State Reduction and Assignment

- State Reduction
  - Reduction on the number of flip-flops and the number of gates.
  - Reduction in the number of states may result in a reduction in the number of flip-flops.
  - State-reduction algorithms are concerned with procedures for reducing the number of states in a state-table while keeping the external input-output requirements unchanged.

Example:

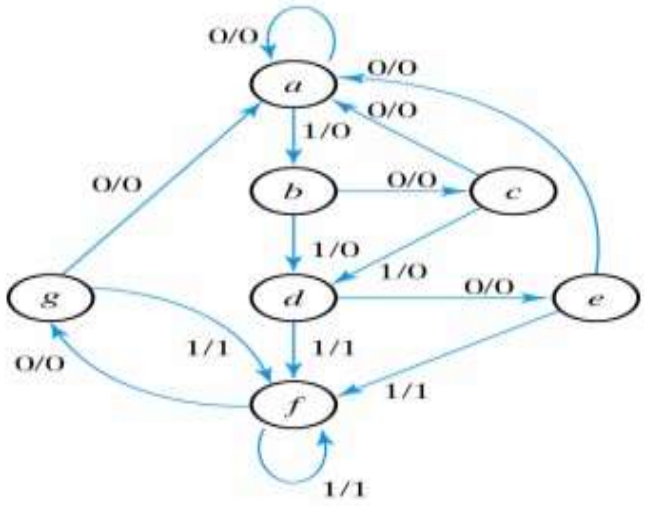


Figure: State diagram

State table from the state diagram

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	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

Same next state and same output at e and g state, so e= g and state g is reduction.

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	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	e	f	0	1

Equivalent state

- Two states are said to be equivalent
  - For each member of the set of inputs, they give exactly the same output and send the circuit to the same state.
  - One of them can be removed

Reduced state table

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	d	0	1
e	a	d	0	1

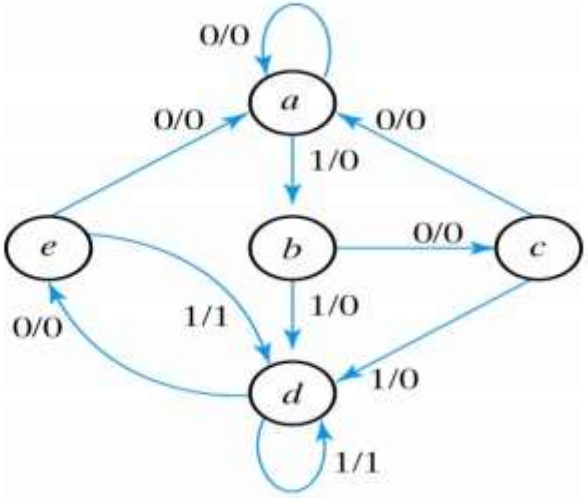


Figure: Reduced state diagram

State assignment:

a = 000

b = 001

c = 010

d = 011

e = 100

Characteristic Table

(a) *JK* Flip-Flop

J	K	$Q(t+1)$	Operation
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	$\overline{Q}(t)$	Complement

(b) *SR* Flip-Flop

S	R	$Q(t+1)$	Operation
0	0	$Q(t)$	No change
0	1	0	Reset
1	0	1	Set
1	1	?	Undefined

(c) *D* Flip-Flop

D	$Q(t+1)$	Operation
0	0	Reset
1	1	Set

(d) *T* Flip-Flop

T	$Q(t+1)$	Operation
0	$Q(t)$	No change
1	$\overline{Q}(t)$	Complement

Excitation Table

A table that lists required inputs for a given change of state (present to next state) is called an excitation table.

Excitation Table for 4 Flip-Flops

SR flip-flop

$Q(t)$	$Q(t+1)$	S	R
0	0	0	×
0	1	1	0
1	0	0	1
1	1	×	0

D flip-flop

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

JK flip-flop

$Q(t)$	$Q(t+1)$	J	K
0	0	0	×
0	1	1	×
1	0	×	1
1	1	×	0

T flip-flop

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

Design Procedure

i. From the given information (state diagram or word description of the circuit behavior) obtain the state table.
ii. The number of states can be reduced by state reduction method.
iii. Assign binary values to each state.
iv. Determine the no. of flip-flops needed and assign a letter symbol to each.
v. Choose type of flip-flop to be used
vi. From the state table, derive the circuit excitation and output table.
vii. Derive the output functions and input functions using K-map.
viii. Draw the logic diagram.

Example: Design synchronous sequential circuit for the given state diagram using J-K F/F

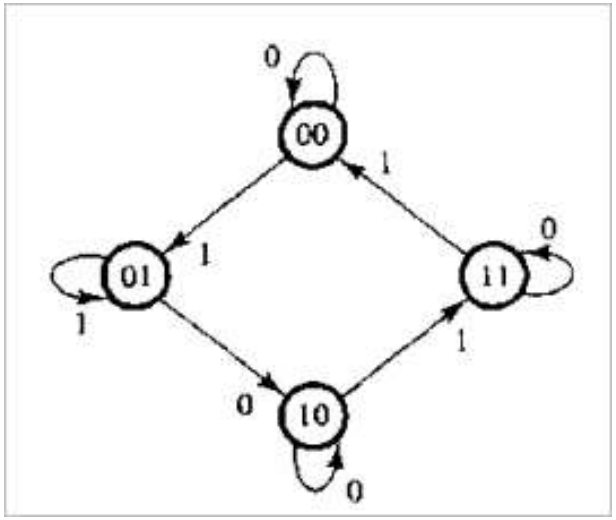


Figure: State diagram for design example

- The state diagram consists of four states with binary values already assigned.
- Directed lines contain single binary digit without a slash, so no output variables.
- The two flip-flops needed to represent the four states are designated A and B.
- The input variable is designated X.

Present State		Next State			
		$x = 0$		$x = 1$	
A	B	A	B	A	B
0	0	0	0	0	1
0	1	1	0	0	1
1	0	1	0	1	1
1	1	1	1	0	0

Table: State table

The state table for this circuit, derived from the state diagram. Note that there is no output section for this circuit

Table
Excitation table

Inputs of Combinational Circuit			Next State		Outputs of Combinational Circuit			
Present State		Input			Flip-Flop Inputs			
A	B	x	A	B	JA	KA	JB	KB
0	0	0	0	0	0	X	0	X
0	0	1	0	1	0	X	1	X
0	1	0	1	0	1	X	X	1
0	1	1	0	1	0	X	X	0
1	0	0	1	0	X	0	0	X
1	0	1	1	1	X	0	1	X
1	1	0	1	1	X	0	X	0
1	1	1	0	0	X	1	X	1

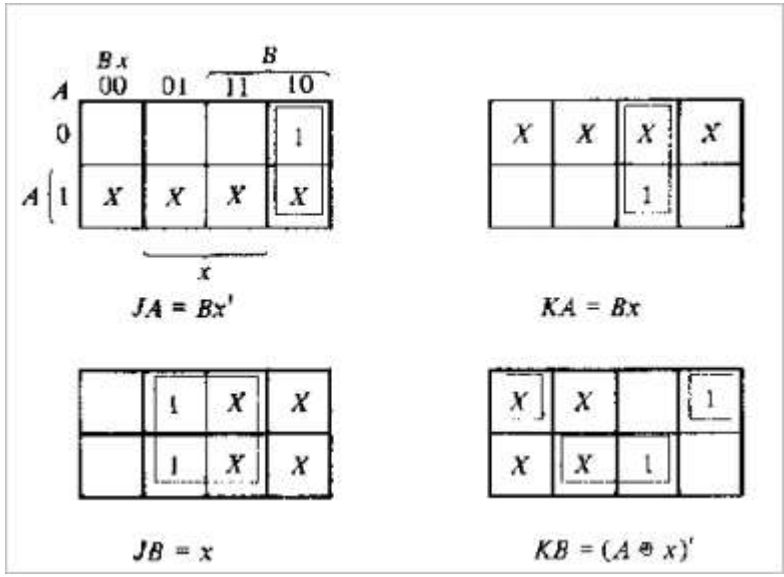


Figure: Maps for combinational circuit

Since J-K flip-flops are used we need columns for the J and K inputs of flip-flops A (denoted by JA and KA) and B (denoted by JB and KB).

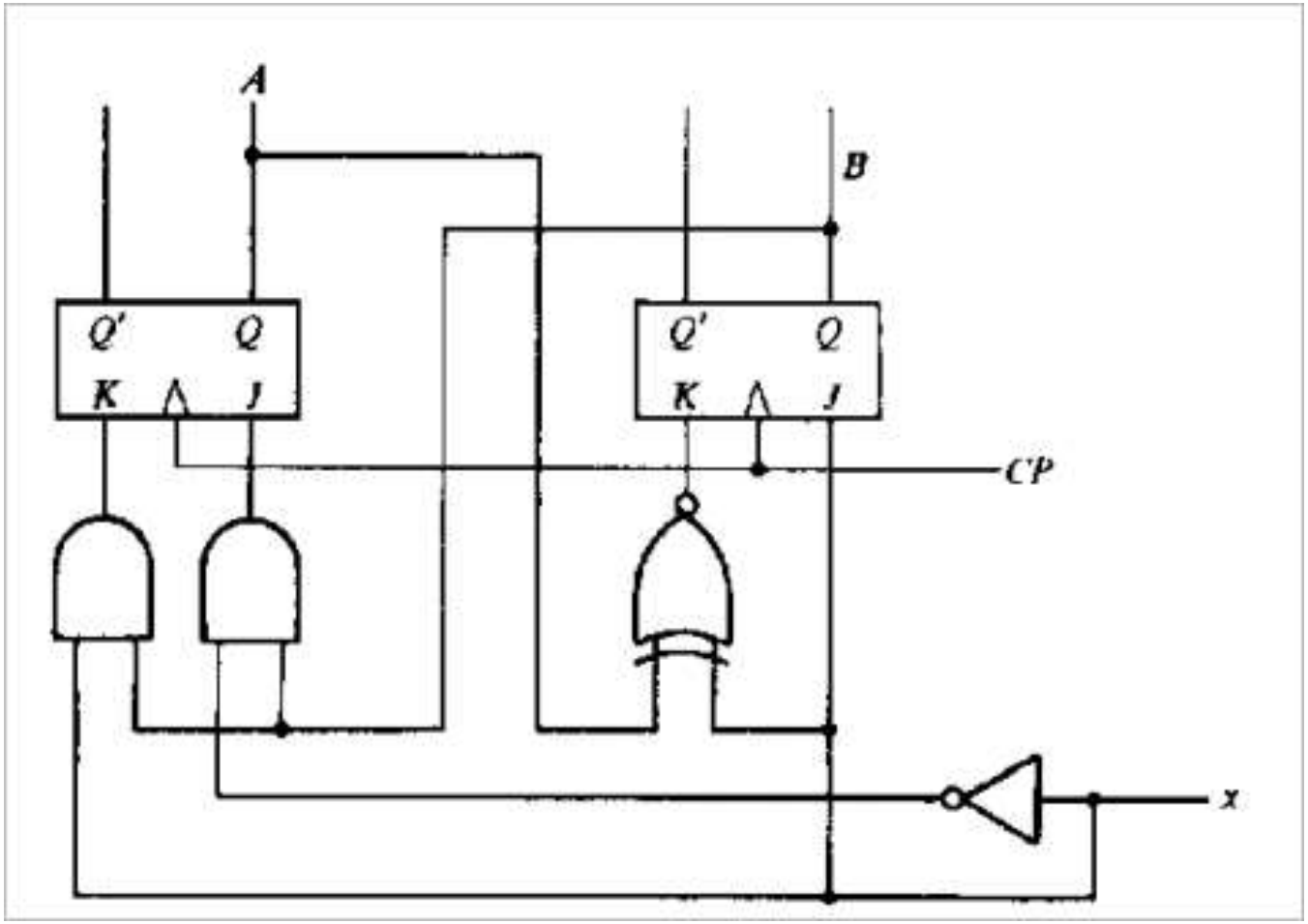


Figure: Logic diagram of sequential circuit

Example: Design sequential circuit with D flip-flops having following next states and output columns.

DA (A, B, X) = Σ m (2, 4, 5, 6)

DB (A, B, X) = Σ m (1, 3, 5, 6)

Y (A, B, X) = Σ m (1, 5)

State Table for Design with D Flip-Flops

Present State		Input	Next State		Output
A	B	x	A	B	y
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	0	0	0

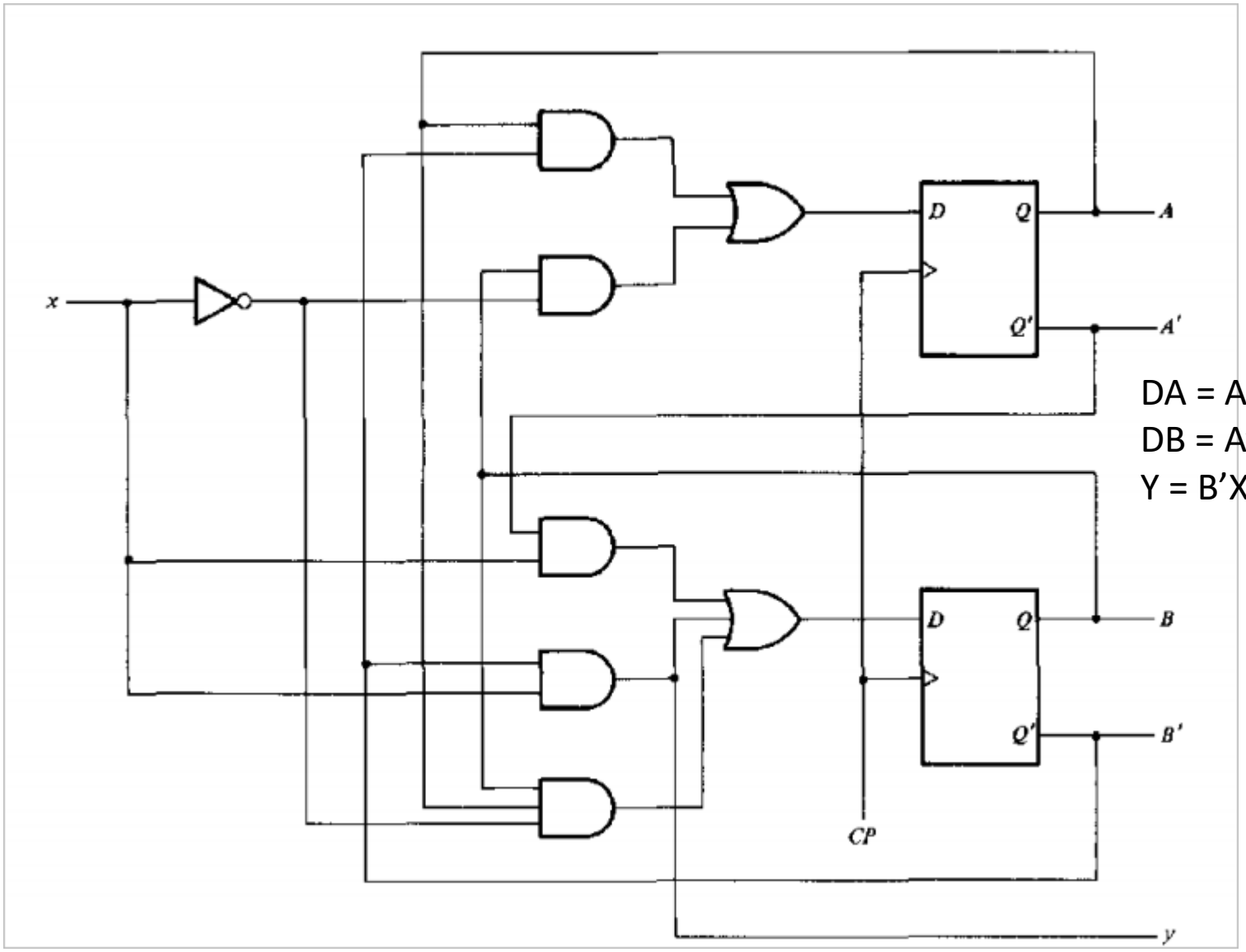
Figure: state diagram

From the K-map simplification

DA = AB' + BX'

DB = A'X + B'X + ABX'

Y = B'X



DA = AB’ + BX’

DB = A’X + B’X + ABX’

Y = B’X

Figure: Logic diagram of sequential circuit with D flip-flop

Design Counter

- A sequential circuit that goes through a prescribed sequence of states upon the application of input pulses is called a *counter.*
- **In** a counter, the sequence of states may follow a binary count or any other sequence of states.
- A counter that follows the binary sequence is called a *binary counter.*
- An n-bit binary counter consists of *n* flip-flops and can count in binary from 0 to 2n - 1.

Design 3-bit counter

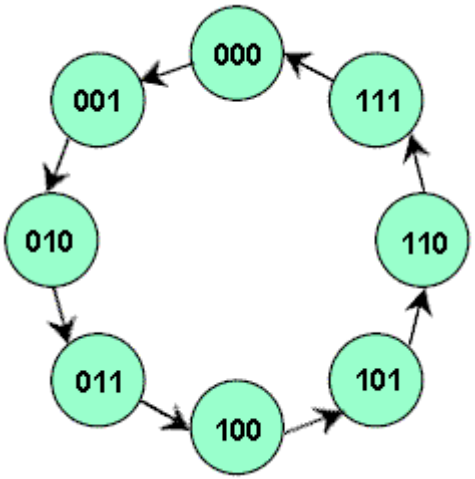
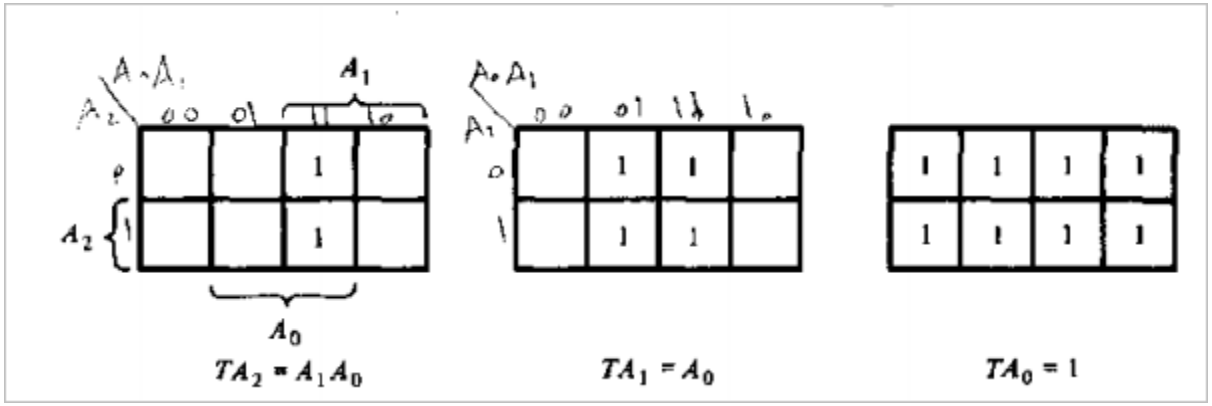


Figure: State diagram

Excitation Table for 3-Bit Counter

Present State			Next State			Flip-Flop Inputs		
A_2	A_1	A_0	A_2	A_1	A_0	TA_2	TA_1	TA_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1



Maps for 3-bit binary counter

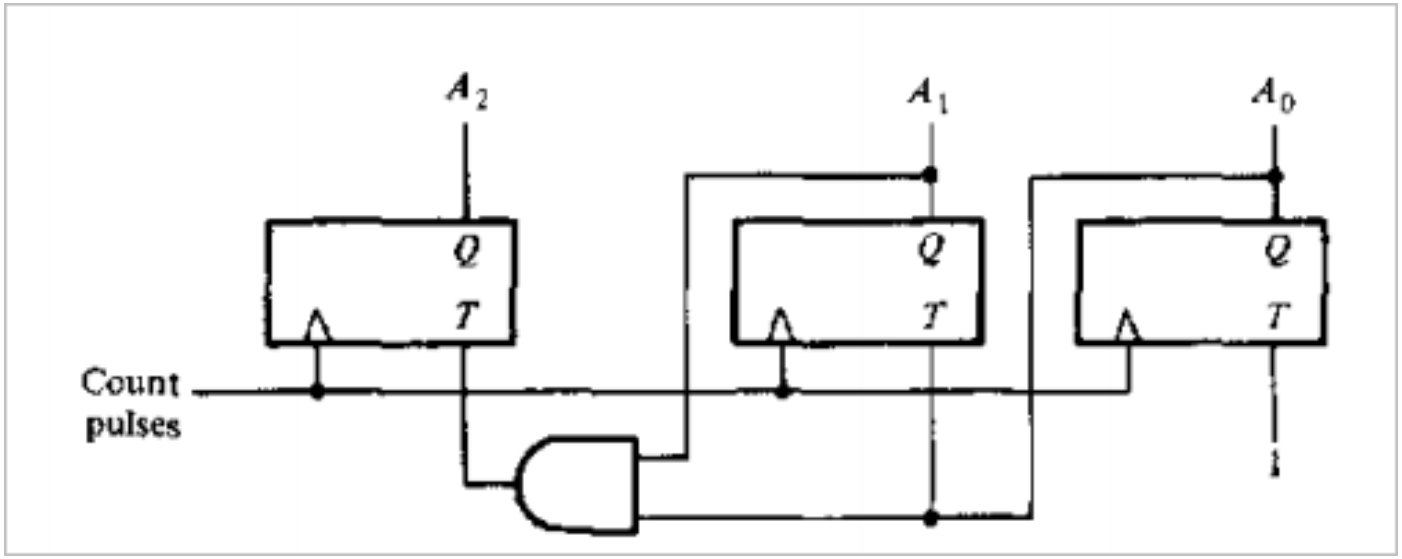


Figure: Logic diagram for 3-bit binary counter

Counter with Non-binary Sequence

Design a counter as shown in state diagram below.

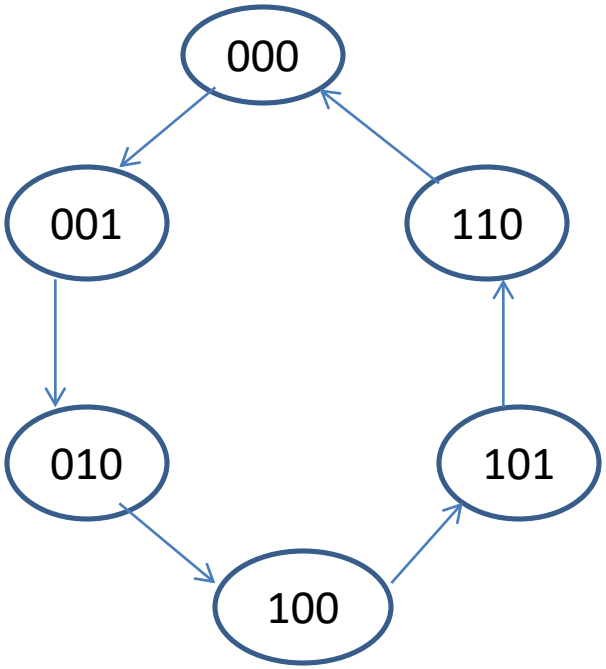


Figure: State diagram

Excitation Table for Counter

Present State			Next State			Flip-Flop Inputs					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	1	0	0	1	X	X	1	0	X
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	0	0	0	X	1	X	1	0	X

Inputs *KB* and *KC* have only I's and X's in their columns, so these inputs are always equal to I. The other flip-flop input functions can be simplified using min-terms 3 and 7 as don't-care conditions. The simplified functions are:

JA = *B* KA = B
JB = C KB = 1
JC = B' KC = 1

You have to show the K-map for simplification

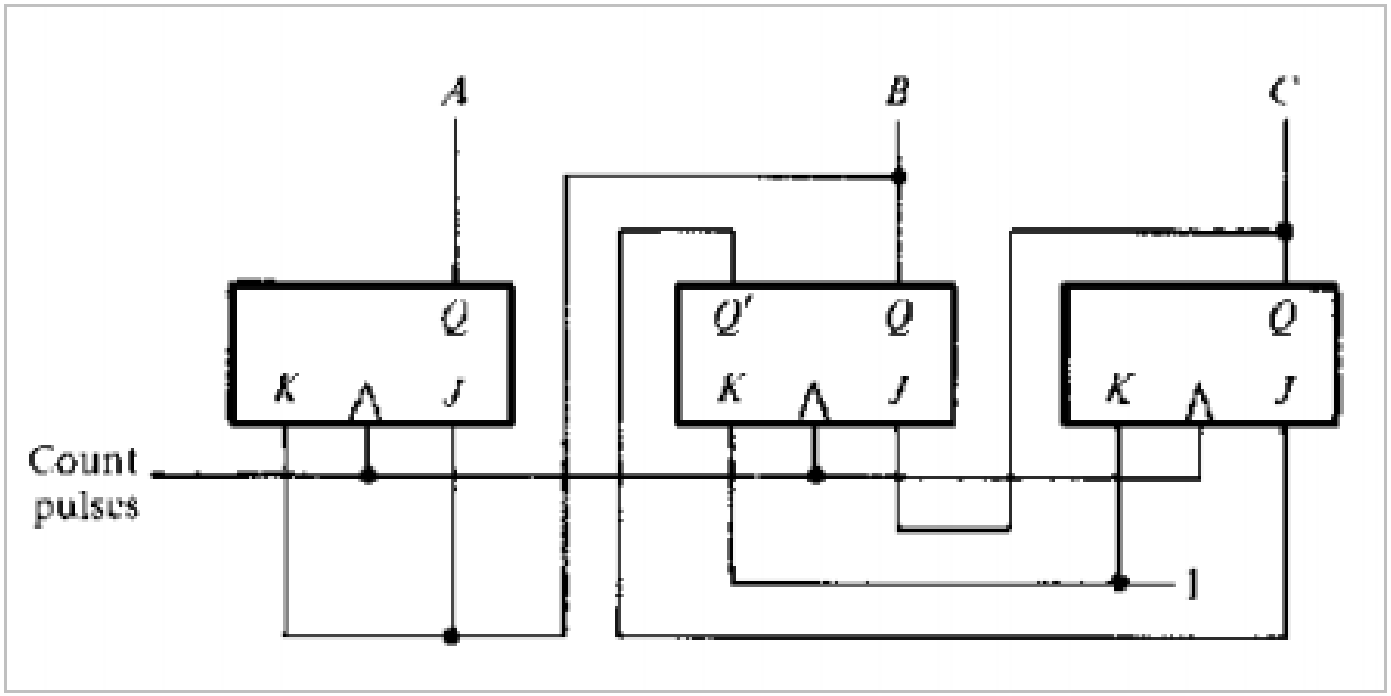


Figure: Logic diagram of counter