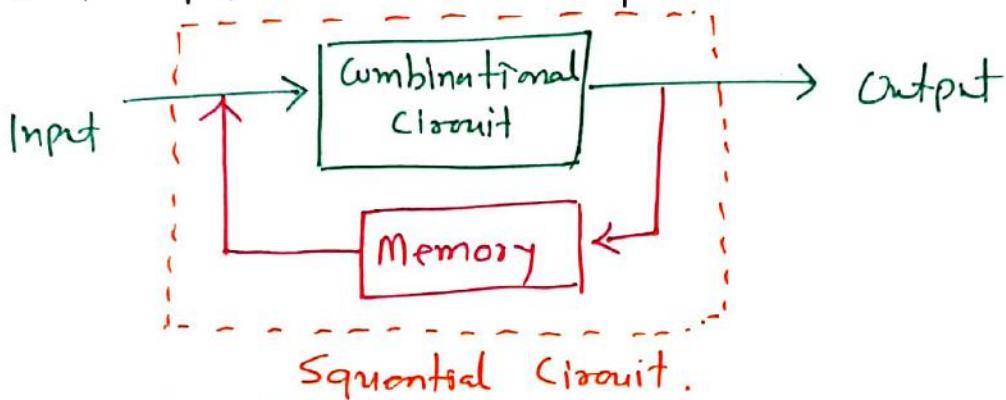


## Squential Circuit

- Sequential circuit is a combination of combinational circuit and memory.
- In Sequential Circuit, Present output is depending on Present input and Past output.



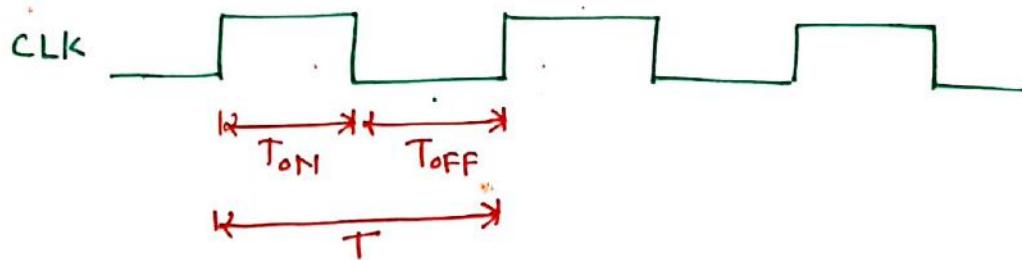
- E.g.
- Flip Flop
  - Counter
  - Register.

→ Classification of Sequential Circuit

- 1] Synchronous Sequential Circuit
  - All memory Elements work with same clock pulse
- 2] Asynchronous Sequential Circuit.
  - Memory elements are working with different clock pulse.

Clock and triggering by clock in Sequential Circuit

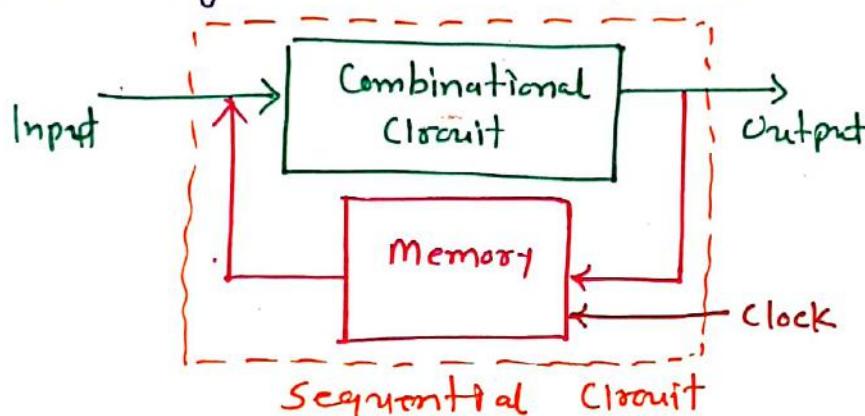
→ Clock is a signal, used in digital circuit 140



1) Duty cycle  $D = \frac{T_{on}}{T} = \left[ \frac{T/2}{T} = \frac{1}{2} = (0.5) \right]$

2) Frequency  $F = \frac{1}{T}$

→ Operational Speed and transition of state is define by clock in sequential circuit.



- O/p of circuit is defined as state.
- O/p state will change with respect to clock.

→ There are two types of clock triggering

- 1] Edge trigger clock
  - +ve Edge trigger
  - ve Edge trigger
- 2] Level trigger clock
  - +ve Level trigger
  - ve Level trigger.

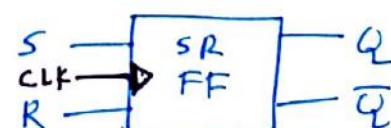


- Clock rises from Low to High, trigger recorded as +ve edge trigger.

- Clock falls from High to Low, trigger recorded as -ve edge trigger.

- During clock is high, trigger recorded as +ve level trigger.

- During clock is low, trigger recorded as -ve level trigger.



CLK → (+ve) Edge trigger

CLK → (-ve) Edge trigger

CLK → (+ve) Level trigger

CLK → (-ve) Level trigger

## Latch

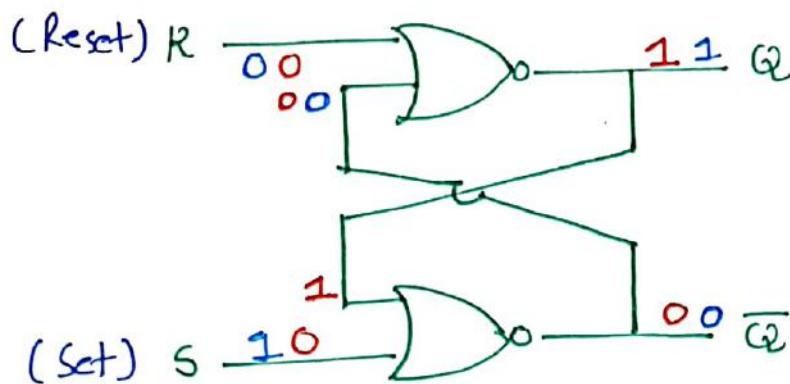
- It is structured based on logic gates blocks.
- It does not have clock in its internal circuit.
- It checks inputs continuously and changes output correspondingly.
- Latch is sensitive to input signal only.
- It is level trigger.
- It can not be used as register.
- It is Asynchronous circuit.
- It requires less power.
- It is faster.
- Designing is less complex.

## Flip Flop

- It is structured based on blocks of Latch and logic gates.
- It has a clock with its internal circuit.
- It checks input continuously and changes output with respect to clock signal.
- F.F. is sensitive to input and clock signal.
- It is edge trigger.
- It can be used as register.
- It is synchronous circuit.
- It requires more power.
- It is slower.
- Designing is more complex.

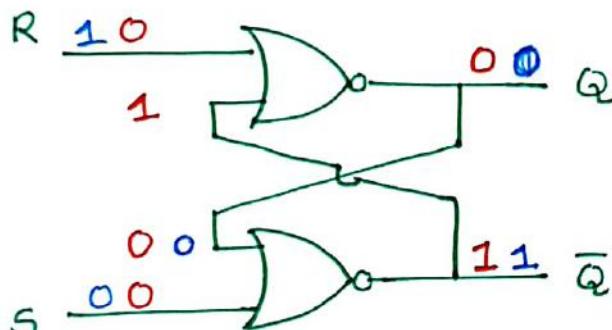
# SR Latch using NOR gates.

142



$$S = 1, R = 0, Q = \underline{1} \quad \& \quad \bar{Q} = \underline{0}$$

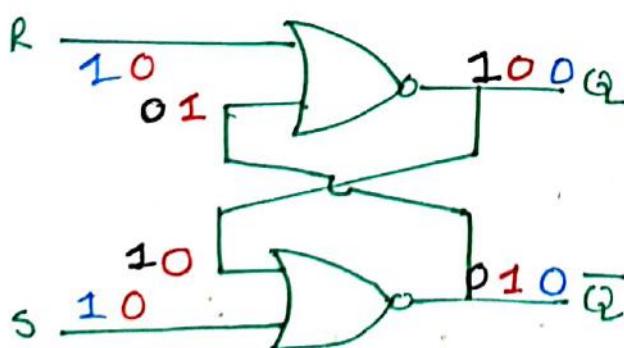
$$\boxed{S = 0, R = 0, Q = \underline{1} \quad \& \quad \bar{Q} = \underline{0}} \leftarrow \text{It acts like memory.}$$



S	R	Q	$\bar{Q}$	
0	0	memory.		
0	1	0	1	
1	0	1	0	
1	1	Invalid		

$$S = 0, R = 1, Q = 0 \quad \& \quad \bar{Q} = 1$$

$$\boxed{S = 0, R = 0, Q = 0 \quad \& \quad \bar{Q} = 1} \leftarrow \text{It acts like memory.}$$

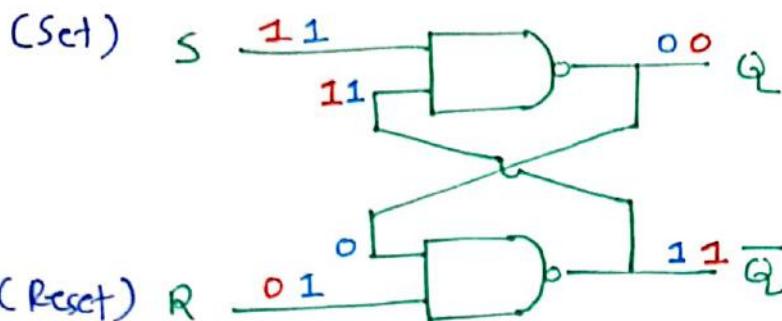


↓ Cond.<sup>n</sup> Violation.

$$S = 1, R = 1, \boxed{Q = \underline{0} \quad \& \quad \bar{Q} = \underline{0}}$$

$$S = 0, R = 0, \boxed{Q = 0 \quad \& \quad \bar{Q} = 1}$$

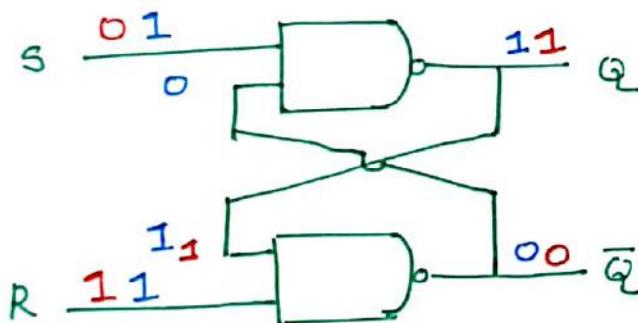
↓ Invalid O/P.



$$S = 1, R = 0, Q = \underline{0} \text{ } \& \text{ } \bar{Q} = \underline{1}$$

$$\boxed{S = 1, R = 1, Q = 0 \text{ } \& \text{ } \bar{Q} = 1}$$

← This state acts like memory state.

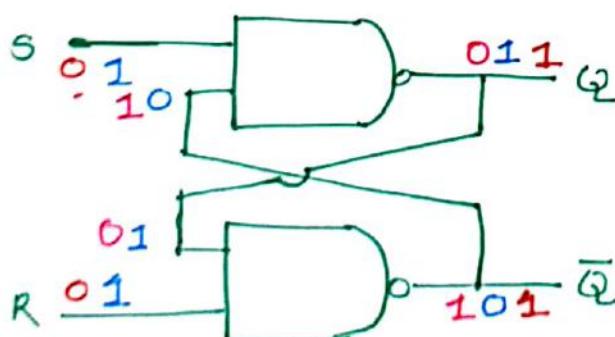


S	R	Q	$\bar{Q}$
0	0	Invalid	
0	1	1	0
1	0	0	1
1	1	Memory	

$$S = 0, R = 1, Q = \underline{1} \text{ } \& \text{ } \bar{Q} = \underline{0}$$

$$\boxed{S = 1, R = 1, Q = \underline{1} \text{ } \& \text{ } \bar{Q} = \underline{0}}$$

← This state acts like memory state.



This is Invalid state.

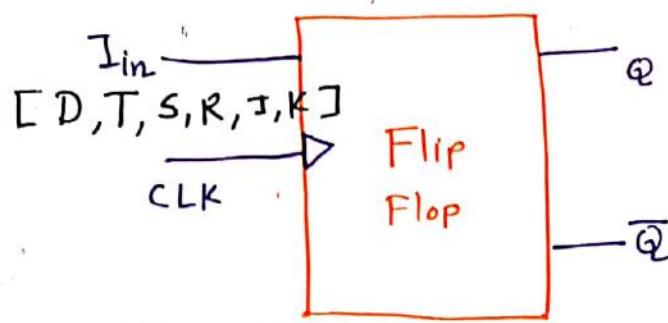
$$S = 0, R = 0, \boxed{Q = \underline{1} \text{ } \& \text{ } \bar{Q} = \underline{1}}$$

$$S = 1, R = 1, Q = 1 \text{ } \& \text{ } \bar{Q} = 0$$

$$Q = 0 \text{ } \& \text{ } \bar{Q} = 1$$

# Truth Table, Characteristics Table & Excitation Table of Flip Flop.

149



Char. Table

$Q_n$	$I_{in}$	$Q_{n+1}$
.	.	.

Truth Table

CLK	$I_{in}$	Q	$\bar{Q}$
.	.	.	.

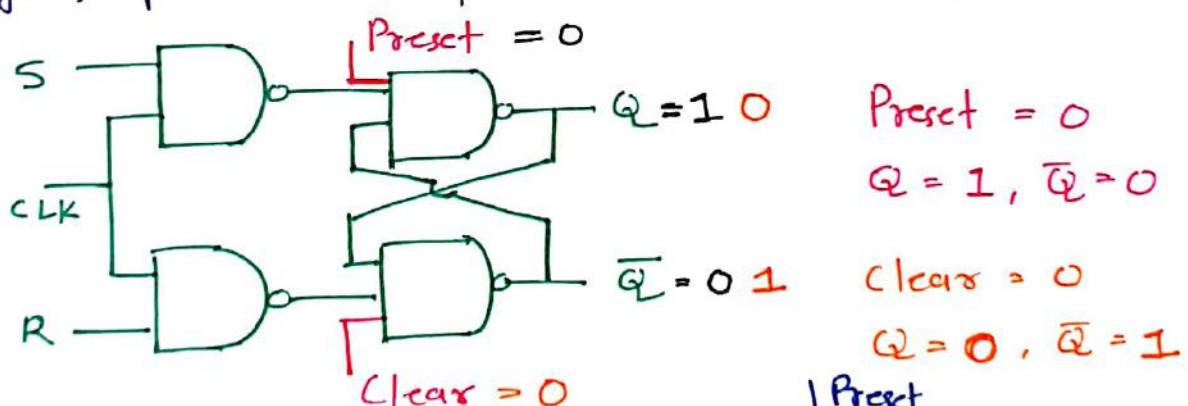
CLK	$I_{in}$	$Q_{n+1}$
.	.	.

Excitation Table

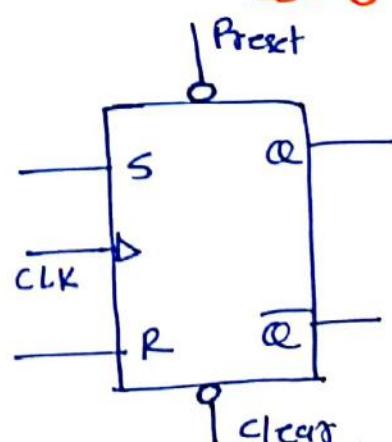
$Q_n$	$Q_{n+1}$	$I_{in}$
.	.	.

## Preset and Clear Input

- It can be given at any time without synchronization.
- It gives predefined output without bothering other inputs.

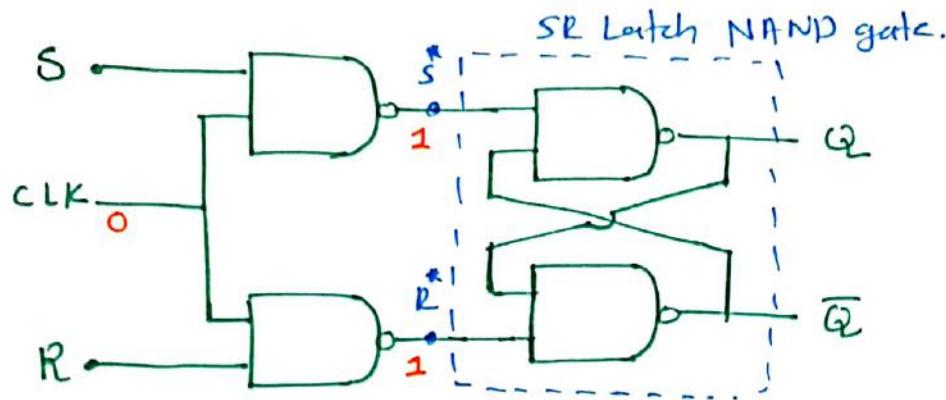


Preset	Clear	$Q_n$
0	0	Not used
0	1	1
1	0	0
1	1	FF Operation.



# SR Flip Flop

145



$S^*$	$R^*$	$Q$	$\bar{Q}$
0	0		
0	1	1	0
1	0	0	1
1	1		

Invalid  
Memory

$CLK$	$S$	$R$	$Q$	$\bar{Q}$
0	X	X		
1	0	0		
1	0	1	0	1
1	1	0	1	0
1	1	1		

Memory  
Memory  
0 1  
1 0  
Invalid

$CLK$	$S$	$R$	$Q_{n+1}$
0	X	X	$Q_n$
1	0	0	$Q_n$
1	0	1	0
1	1	0	1
1	1	1	Invalid

Truth Table.

$$S^* = \overline{S \cdot CLK} = \overline{S} + \overline{CLK}$$

$$R^* = \overline{R \cdot CLK} = \overline{R} + \overline{CLK}$$

$$S^* = \overline{S}, \quad R^* = \overline{R}$$

Char. Table

$Q_n$	$S$	$R$	$Q_{n+1}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Invalid
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Invalid.

Excitation Table

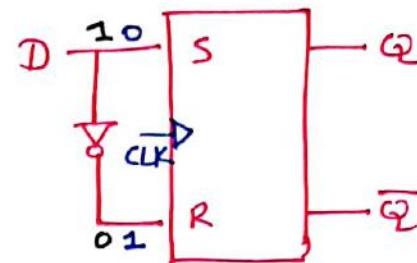
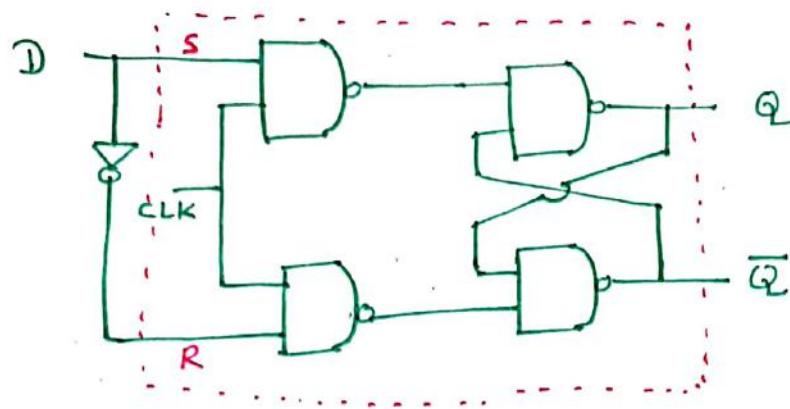
$Q_n$	$Q_{n+1}$	$S$	$R$
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

$$Q_{n+1} = S + Q_n \overline{R}$$

$Q_n$	SR	00	01	11	10
0	00	0	0	X	1
1	1	0	X	X	1

# D Flip Flop [ Data Flip Flop ]

146



char. Table

CLK	S	R	$Q_{n+1}$
0	X	X	$Q_n$
1	0	0	$Q_n$
1	0	1	0
1	1	0	1
1	1	1	Invalid

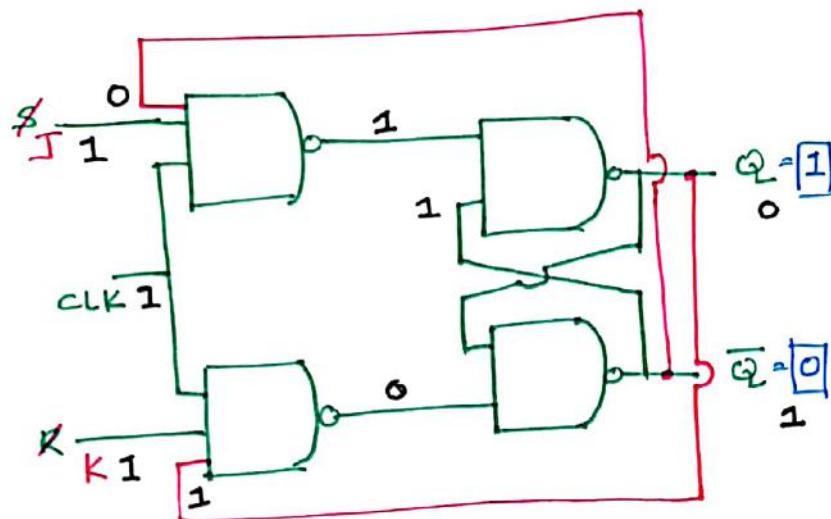
CLK	D	$Q_{n+1}$
0	X	$Q_n$
1	0	0
1	1	1

$Q_n$	D	$Q_{n+1}$
0	0	0
0	1	1
1	0	0
1	1	1

$$Q_{n+1} = D$$

Excitation Table

$Q_n$	$Q_{n+1}$	D
0	0	0
0	1	1
1	0	0
1	1	1



CLK	S	R	$Q = 1$	$\bar{Q} = 0$
0	x	x	Memory	
1	0	0	Memory	
1	0	1	0 1	
1	1	0	1 0	
1	1	1	Memory	

CLK	J	K	$Q = 1$	$\bar{Q} = 0$
0	x	x	Memory	
1	0	0	Memory	
1	0	1	0 1	
1	1	0	1 0	
1	1	1	Memory	

Char. Table

$Q_n$	J	K	$Q_{n+1}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

$Q_n$	$(Q_{n+1} = \bar{Q}_n J + Q_n K)$			
$Q_n$	00	01	11	10
0	0	0	1 1	
1	1	0	0	1

$CLK=1, J=1, K=1, Q=0 \text{ & } \bar{Q}=1$

Truth Table

CLK	J	K	$Q_{n+1}$
0	x	x	$Q_n$
1	0	0	$Q_n$
1	0	1	0
1	1	0	1
1	1	1	$\bar{Q}_n$

Excitation Table

$Q_n$	$Q_{n+1}$	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

J	$Q_{n+1}$
0	0
1	x

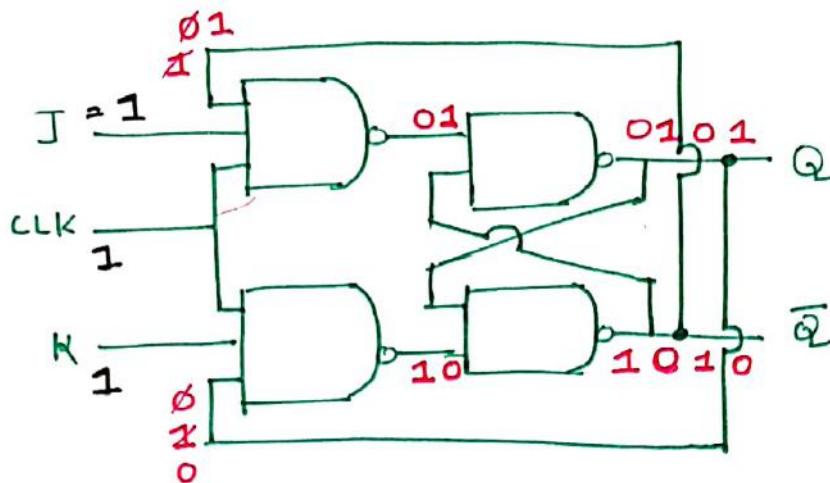
$$J = Q_{n+1}$$

K	$Q_{n+1}$
0	x
1	0

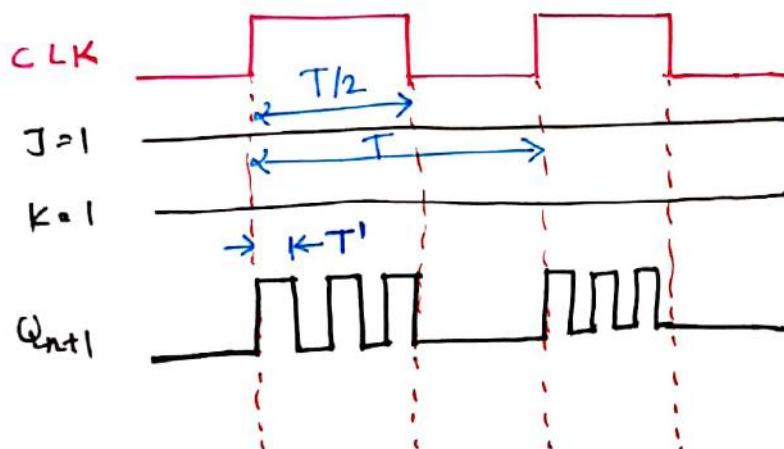
$$K = \bar{Q}_{n+1}$$

# Race Around Condition

148



CLK	J	K	Q	$\bar{Q}$
0	x	x	Memory	
1	0	0	Memory	
1	0	1	0	1
1	1	0	1	0
1	1	1	Memory	



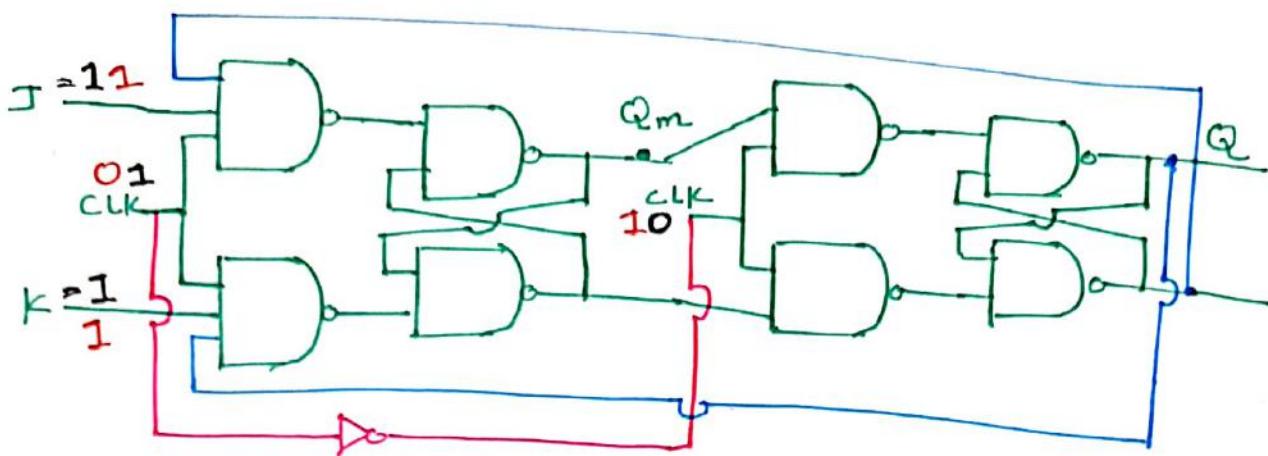
Sol. 2

- 1]  $T' > T/2$  X  
Propagation delay greater than  $T/2$ .

- 2] Edge Trigger. ✓  
3] Master Slave Flip Flop.  
(-ve Edge Trigger).

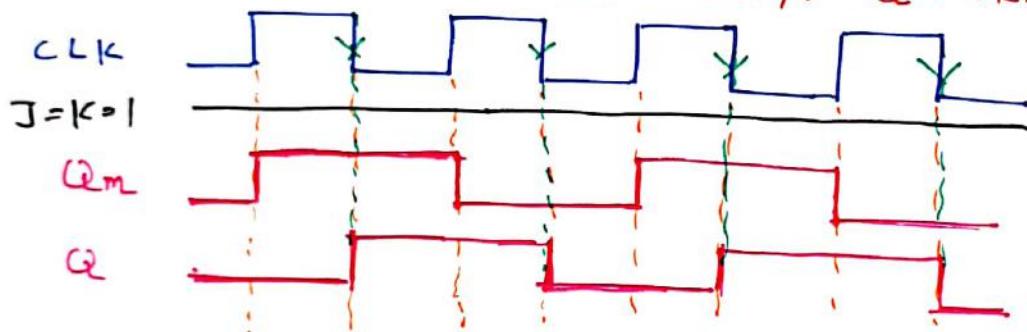
# Master Slave Flip Flop

149



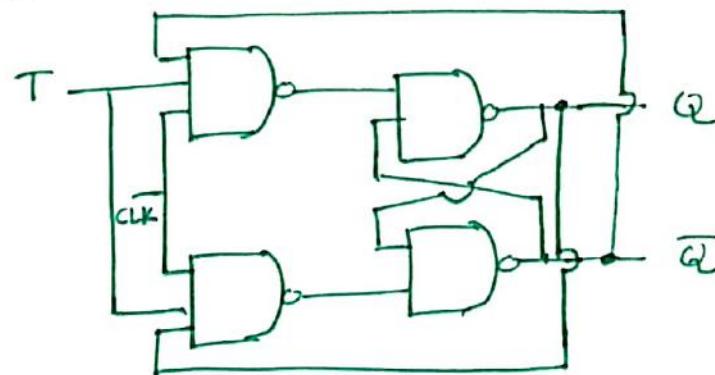
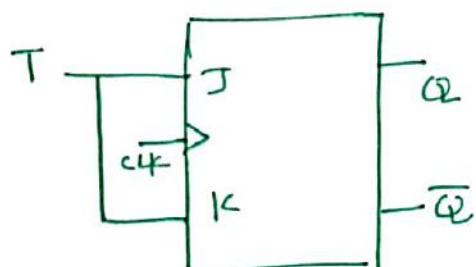
$J = K = 1, \text{ } CLK = 1, \text{ } Q_m = \text{Active}, \text{ } Q = \text{Memory}$

$J = K = 1, \text{ } CLK = 0, \text{ } Q_m = \text{Memory}, \text{ } Q = \text{Active}.$



# T Flip Flop [Toggle Flip Flop]

156



<u>CLK</u>	<u>J</u>	<u>K</u>	<u>Q</u>	<u>Q̄</u>
0	x	x	m	m
1	0	0	m	m
1	0	1	0	1
1	1	0	1	0
1	1	1	m	m

Truth Table

<u>CLK</u>	<u>T</u>	<u>Q</u>	<u>Q̄</u>
0	x	m	m
1	0	m	m
1	1	m	m

<u>CLK</u>	<u>T</u>	<u>Q<sub>n+1</sub></u>
0	x	Q <sub>n</sub>
1	0	Q <sub>n</sub>
1	1	Q̄ <sub>n</sub>

Char. Table

<u>Q<sub>n</sub></u>	<u>T</u>	<u>Q<sub>n+1</sub></u>
0	0	0
0	1	1
1	0	1
1	1	0

$$Q_{n+1} = Q_n \oplus T$$

Excitation Table

<u>Q<sub>n</sub></u>	<u>Q<sub>n+1</sub></u>	<u>T</u>
0	0	0
0	1	1
1	0	1
1	1	0

$$T = Q_n \oplus Q_{n+1}$$

# SR to D Flip Flop Conversion

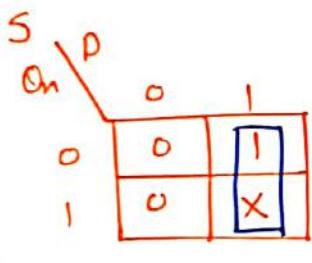
152

→ Available = SR FF

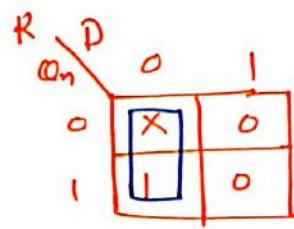
Req'd = D FF

$Q_n$	$D$	$Q_{n+1}$	S R
0 0	0	0	0 X
0 1	1	1	1 0
1 0	0	0	0 1
1 1	1	X	0 0

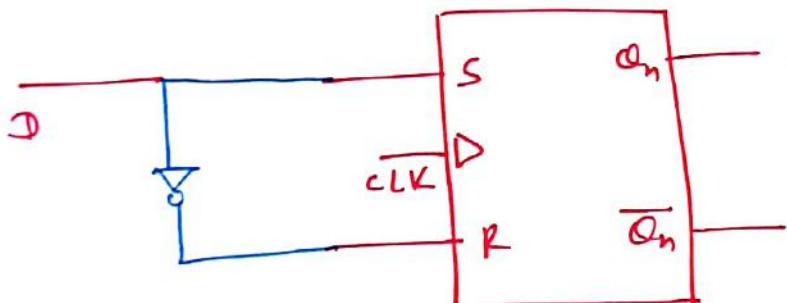
$Q_n$	$Q_{n+1}$	S	R
0 0	0	0	X
0 1	1	1	0
1 0	0	0	1
1 1	X	X	0



$$S = D$$



$$R = \overline{D}$$



# SR to D Flip Flop Conversion

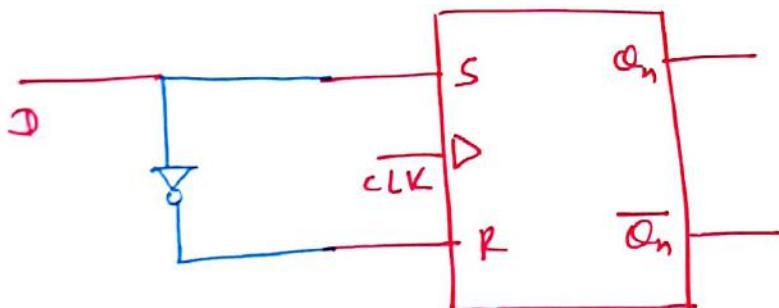
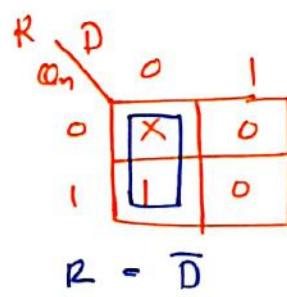
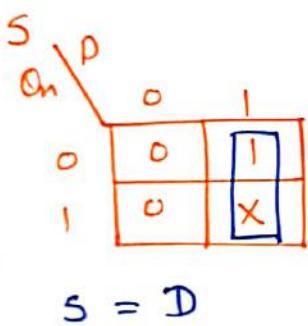
152

→ Available = SR FF

Req.<sup>ed</sup> = D FF

$Q_n$	D	$Q_{n+1}$	S R
0 0	0	0	0 X
0 1	1	1	1 0
1 0	0	0	0 1
1 1	1	X	X 0

$Q_n$	$Q_{n+1}$	S	R
0 0	0	0	X
0 1	1	1	0
1 0	0	0	1
1 1	X	X	0



# SR to JK Flip Flop Conversion

153

→ Available = SR FF

Req<sup>ed</sup> = JK FF

$Q_n$	J	K	$Q_{n+1}$	S	R
0	0	0	0	0	X
0	0	1	0	0	X
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	1	X	0
1	0	1	0	0	1
1	1	0	1	X	0
1	1	1	0	0	1

$Q_n$	$Q_{n+1}$	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

$S$  JK

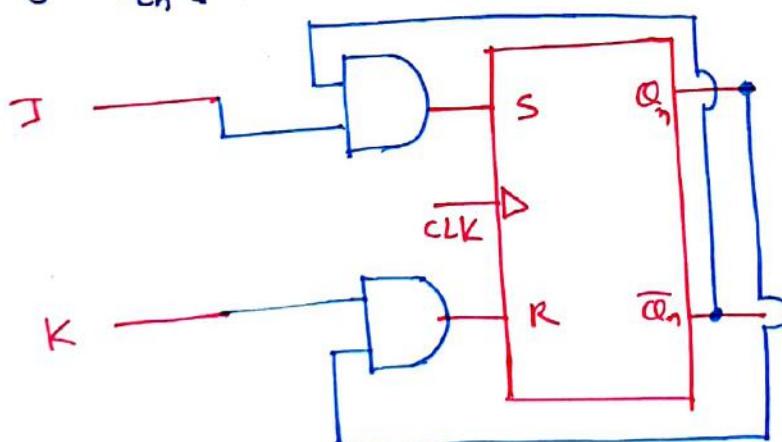
$Q_n$	00	01	11	10
0	0	0	1	1
1	X	0	0	X

$S = \bar{Q}_n J$

$R$  JK

$Q_n$	00	01	11	10
0	X	X	0	0
1	0	1	1	0

$R = Q_n K$



# SR to T Flip Flop Conversion

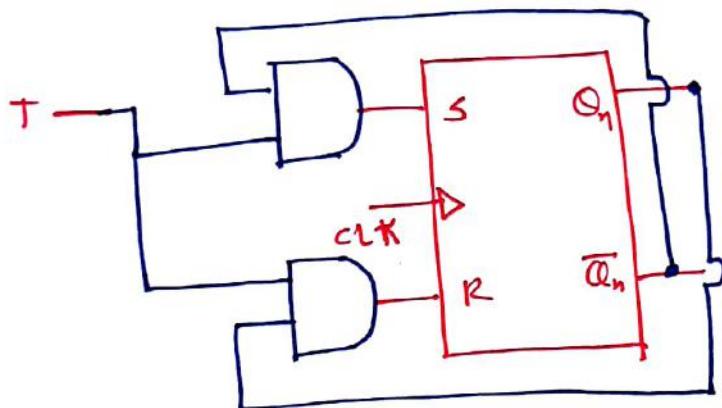
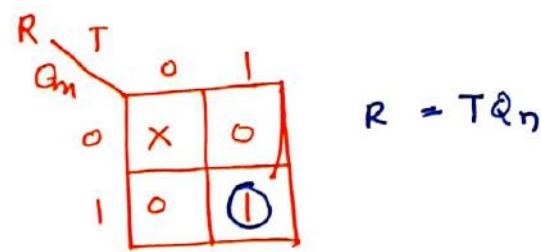
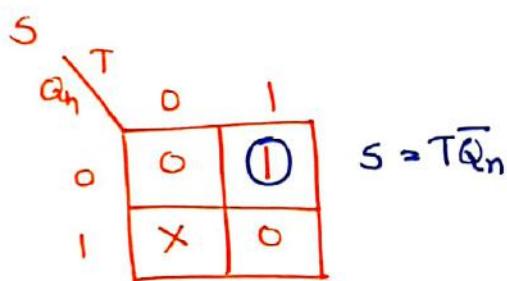
154

→ Available FF = SR FF

Req'd FF = T FF

$Q_n$	T	$Q_{n+1}$	S	R
0	0	0	0	X
0	1	1	1	0
1	0	1	X	0
1	1	0	0	1

$Q_n$	$Q_{n+1}$	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0



# $JK$ to $D$ Flip Flop Conversion

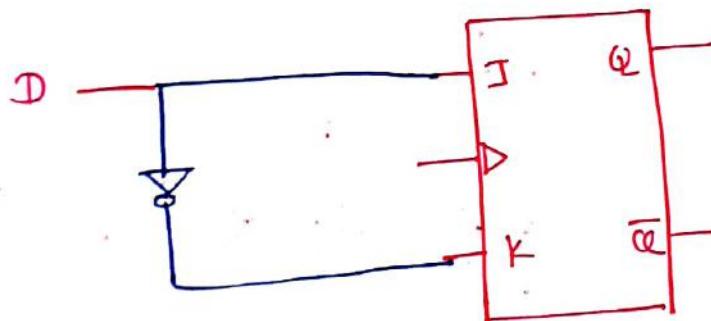
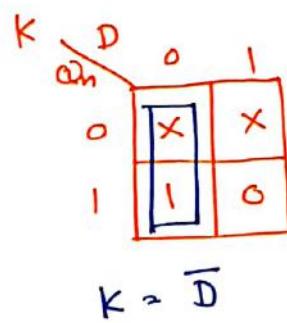
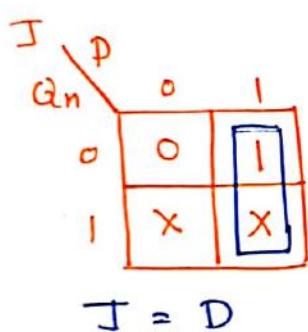
155

↳ Available =  $JK$  FF

$R_{cd.}^{ed}$  =  $D$  FF.

$Q_n$	$D$	$Q_{n+1}$	$J$	$K$
0	0	0	0	x
0	1	1	1	x
1	0	0	x	1
1	1	1	x	0

$Q_n$	$Q_{n+1}$	$J$	$K$
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0



# JK to T Flip Flop Conversion

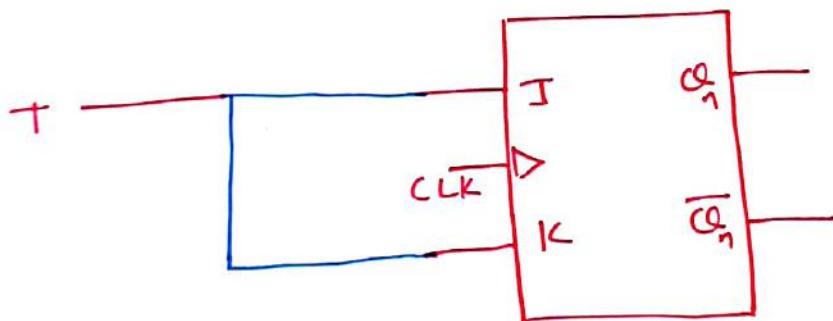
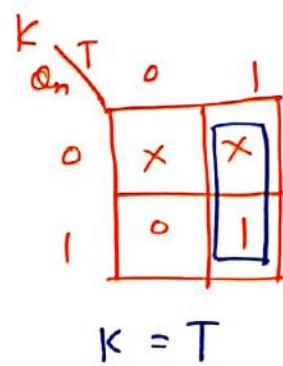
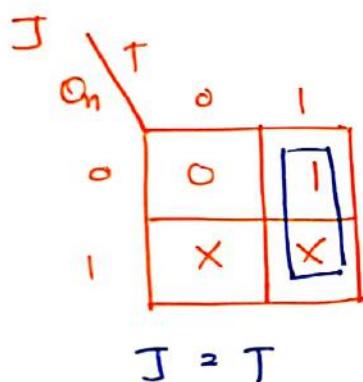
156

→ Available = JK FF

Req'd = T FF

$Q_n$	T	$Q_{n+1}$	J	K
0 0	0	0	0	X
0 1	1	1	1	X
1 0	1	X	0	0
1 1	0	X	1	1

$Q_n$	$Q_{n+1}$	J	K
0 0	0	0	X
0 1	1	1	X
1 0	X	X	1
1 1	X	X	0



# D to T Flip Flop Conversion

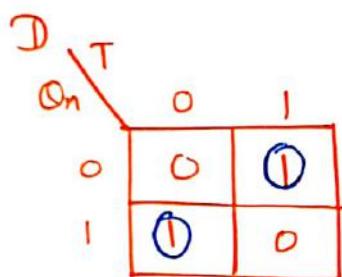
157

→ Available = D FF

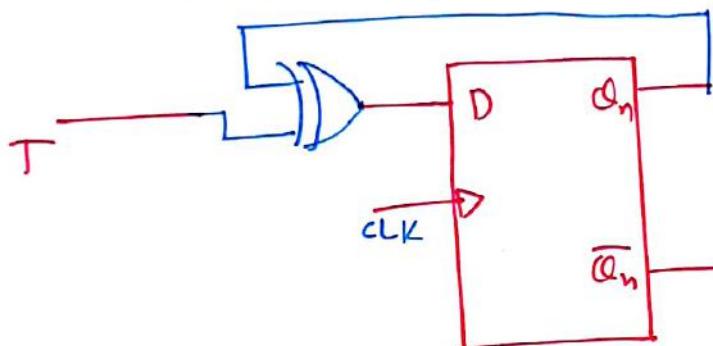
Req'd = T FF

$Q_n$	T	$Q_{n+1}$	D
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

$Q_n$	$Q_{n+1}$	D
0	0	0
0	1	1
1	0	0
1	1	1



$$\begin{aligned} D &= T\bar{Q}_n + \bar{Q}_n\bar{T} \\ &= T \oplus Q_n \end{aligned}$$



# D to JK Flip Flop Conversion

158

→ Available FF = D FF

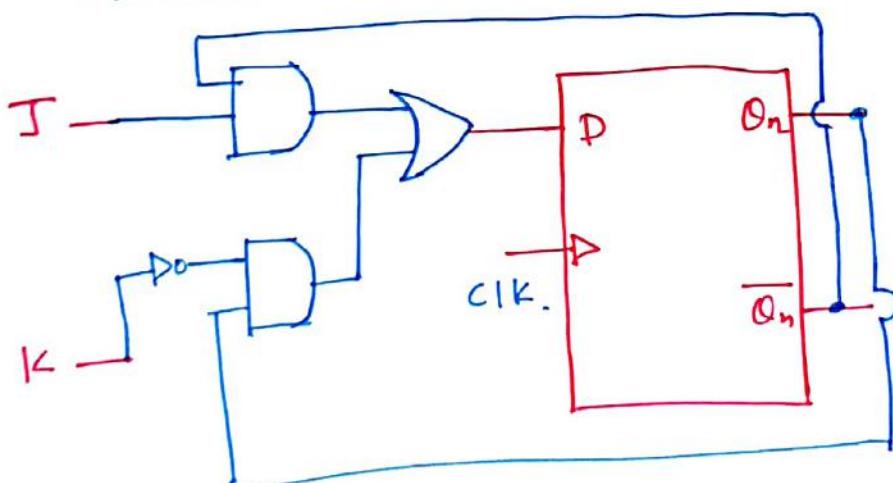
$R_{eq.}^d$  FF = JK FF

$Q_n$	J	K	$Q_{n+1}$	D
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	0	0
1	1	0	1	1
1	1	1	0	0

$Q_n$	$Q_{n+1}$	D
0	0	0
0	1	1
1	0	0
1	1	1

$Q_n$	JK	00	01	11	10
0		0	0	1	1
1		1	0	0	1

$$D = Q_n \bar{K} + \bar{Q}_n J$$



# T to D Flip Flop Conversion

159

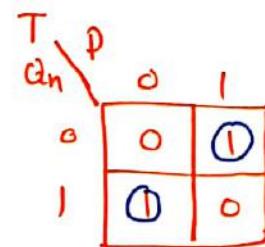
→ Available = T FF

Req'd = D FF

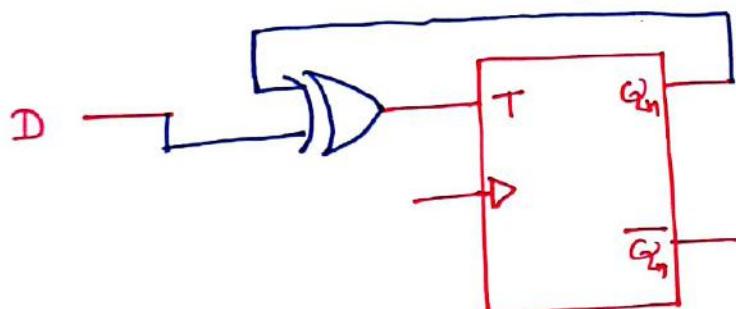
$Q_n$	D	$Q_{n+1}$	T
0	0	0	0
0	1	1	1
1	0	0	1
1	1	1	0

$Q_n$	$Q_{n+1}$	T
0	0	0
0	1	1
1	0	1
1	1	0

$$T = Q_n \oplus D$$



$$\begin{aligned} T &= D\bar{Q}_n + Q_n\bar{D} \\ &= Q_n \oplus D \end{aligned}$$



# T to JK Flip Flop Conversion

160

→ Available FF = T FF

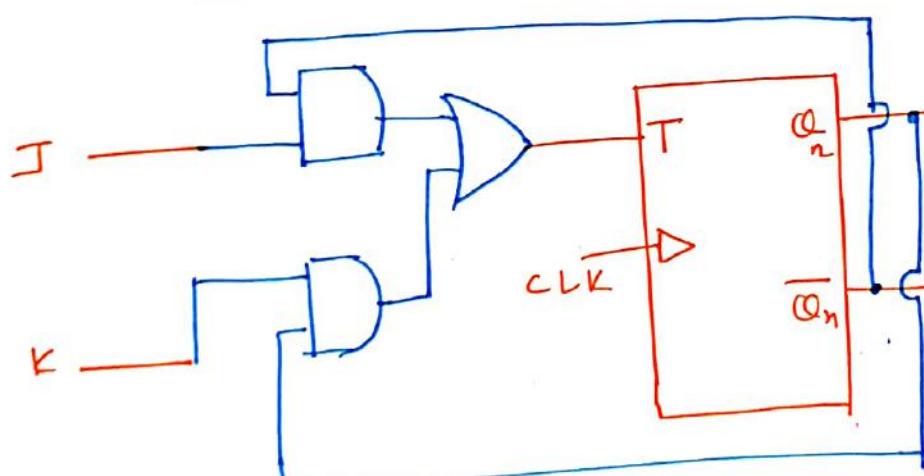
Req'd FF = JK FF

$Q_n$	JK	$Q_{n+1}$	T
0 0 0	0	0	0
0 0 1	0	0	0
0 1 0	1	1	1
0 1 1	1	1	1
1 0 0	1	0	0
1 0 1	0	1	1
1 1 0	1	0	0
1 1 1	0	1	1

$Q_n$	$Q_{n+1}$	T
0 0	0	0
0 1	1	1
1 0	1	1
1 1	0	0

$Q_n$	00	01	11	10
0	0	0	1	1
1	0	1	1	0

$$T = Q_n K + \bar{Q}_n J$$



JK to SR, T to SR and D to SR Flip Flop 161

-  $S = 1, R = 1, Q = \underline{\text{Invalid}}$   
 $\bar{Q} = \underline{\text{Invalid}}$

- we can not form SR

JK to SR, T to SR and D to SR Flip Flop conversion.

-  $S = 1, R = 1, Q_{n+1} = \underline{\text{Invalid}}$

- JK, T & D,  $Q_{n+1} = \text{It will never go in Invalid state.}$

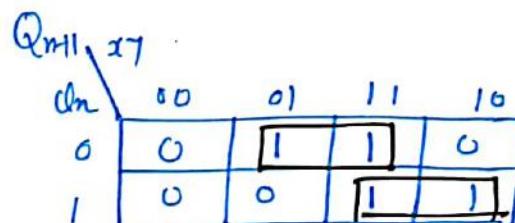
## Examples on Flip flop 162

1) A new Flip flop is having behaviour as described below. It has two inputs  $X$  and  $Y$  and when both inputs are same and they are 1, the flip flop is going to set else flip flop resets. If both inputs are different and they are 0,1 the flip flop complements itself otherwise it is going to retain the last state. Which of the following representation is the characteristics expression of the new flip flop?

- A)  $XQ + Y\bar{Q}$  B)  $X\bar{Q} + YQ$  C)  $X\bar{Q} + Y\bar{Q}$  D) None

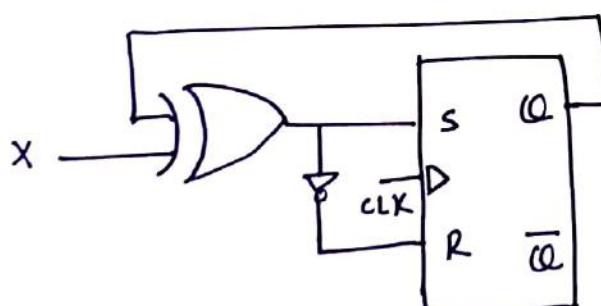
X	Y	Q
0	0	0
0	1	$\bar{Q}$
1	0	$Q$
1	1	1

$Q_n$	X	Y	$Q_{n+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



$$Q_{n+1} = \bar{Q}_n Y + Q_n X$$

2) Make expression of  $Q^+$  for given circuit

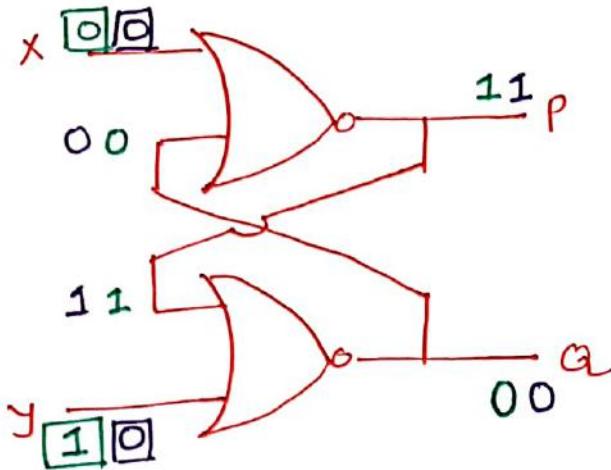


- A)  $XQ$   
B)  $X\bar{Q}$   
C)  $X \oplus Q$   
D)  $X \oplus \bar{Q}$

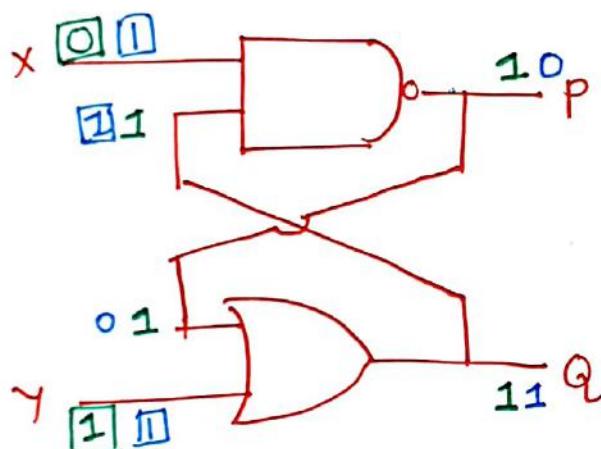
X	Q	$Q^+$
0	0	0
0	1	1
1	0	1
1	1	0

$$\boxed{Q^+ = X \oplus Q}$$

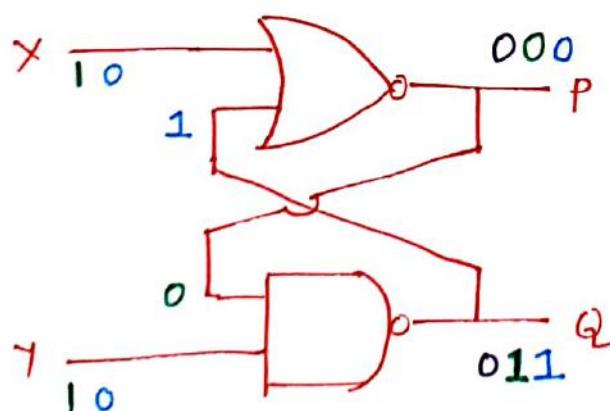
# Examples on Latch 163



→ If Input  $x_1$  changes from 01 to 00 then Output PQ will change from 10 to 10



→ If Input XY changes from 01 to 11 then PQ will change from 11 to 01



→ If Input  $x_2$  changes from 11 to 00 then PQ will change from 01 to 01  
[Initially PQ is 00]

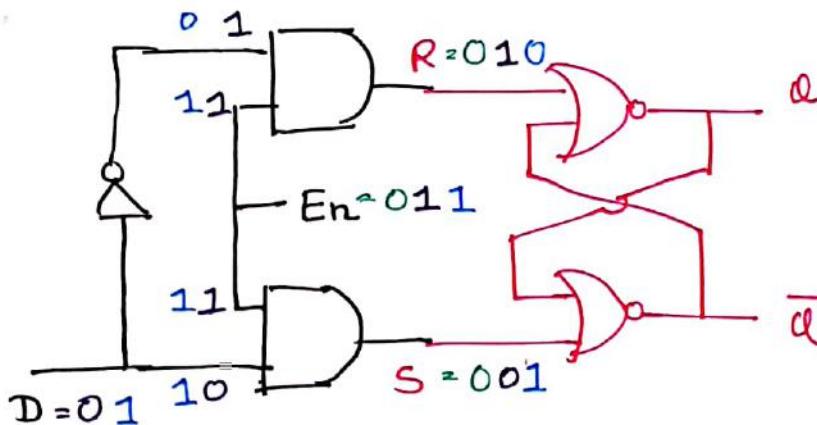
# D Latch

164

→ Truth Table of SR Latch using NOR gate.

→ O/p = V<sub>P</sub>, E<sub>n</sub> = 1

O/p = M<sub>mm</sub>, E<sub>n</sub> = 0



S	R	Q	$\bar{Q}$
0	0	M <sub>mm</sub>	
0	1	0	1
1	0	1	0
1	1	Invalid.	

→ E<sub>n</sub> = 0, D = X, Q = M<sub>mm</sub>

~~XXXXX0XXXXXX, X1XXXXX0XXXXXX~~

→ E<sub>n</sub> = 1, D = 0, Q = 0

→ E<sub>n</sub> = 1, D = 1, Q = 1

E <sub>n</sub>	D	Q
0	X	M <sub>mm</sub>
1	0	0
1	1	1

# Counter

165

→ Counter is used for counting pulses and also we can use it as frequency divider.

→ There are two types of counter

1] Up Counter [0, 1, 2, ..., N]

2] Down Counter [N, N-1, ..., 1, 0]

→ Classifications of Counter.

1] Synchronous Counter

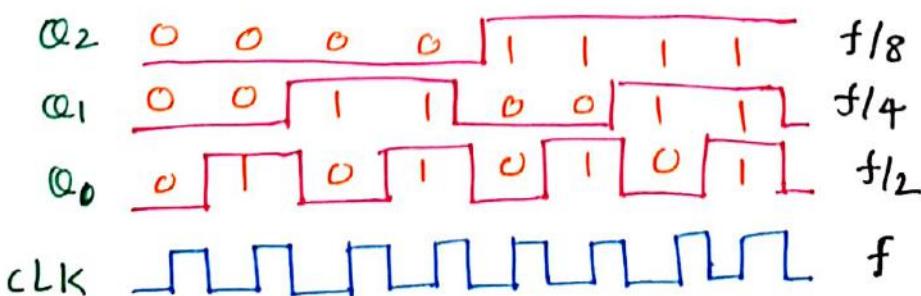
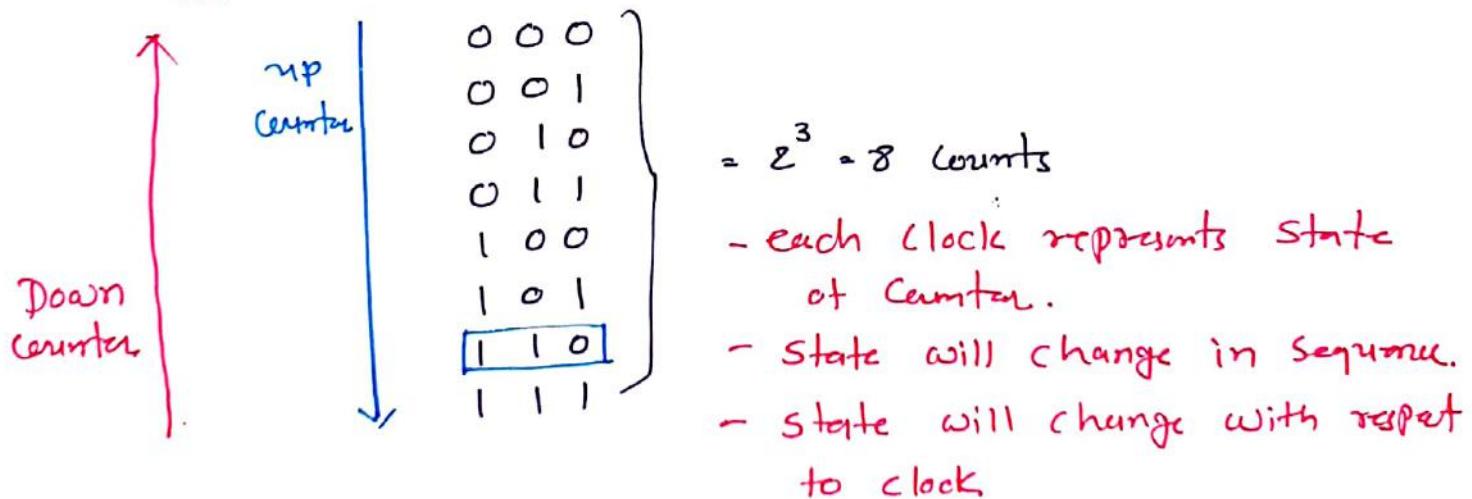
- memory elements are having same clock pulse.

2] Asynchronous Counter.

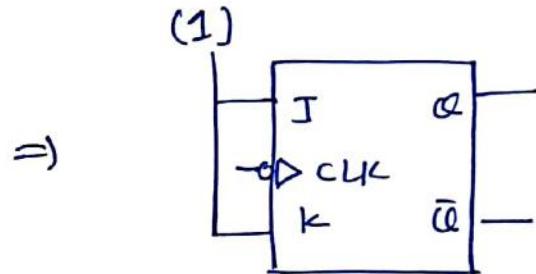
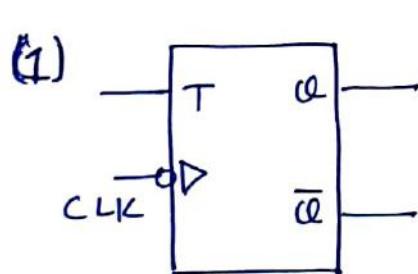
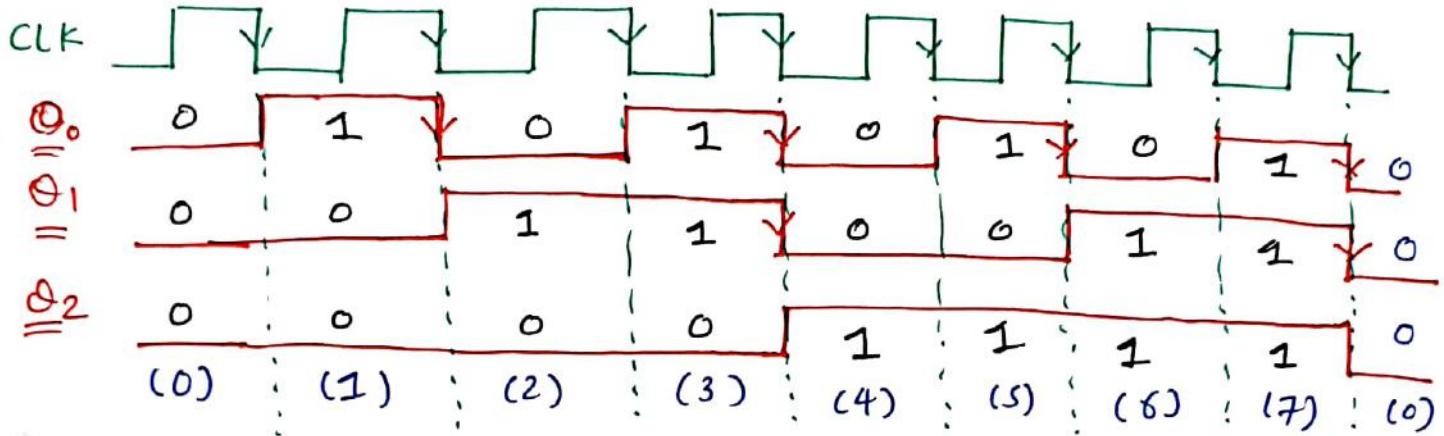
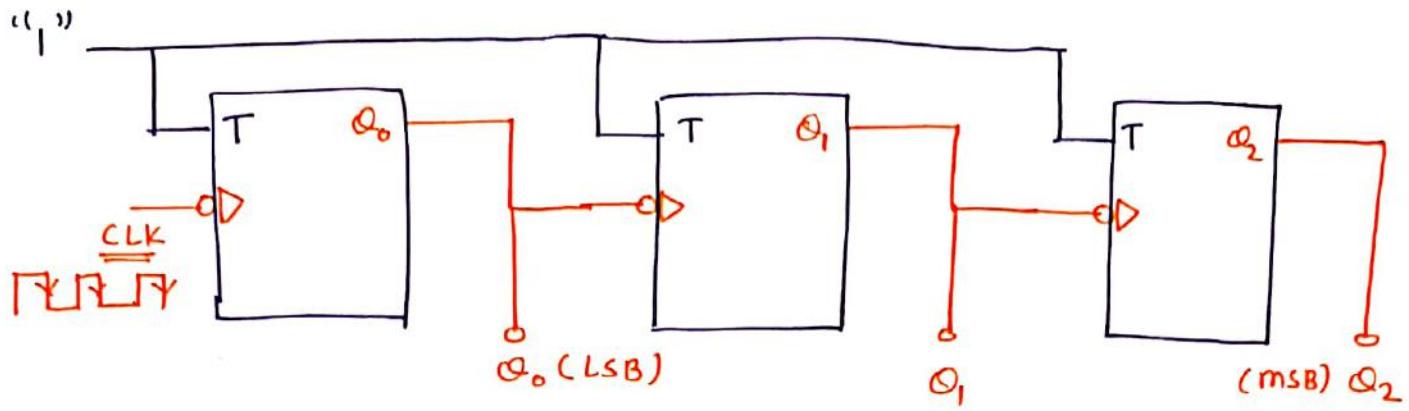
- memory elements are having different clock pulse.

→ For n bits Counter, Max counts =  $2^n$

e.g. For 3 bits Counter.

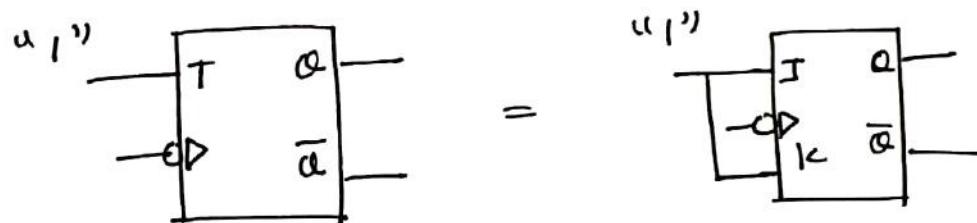
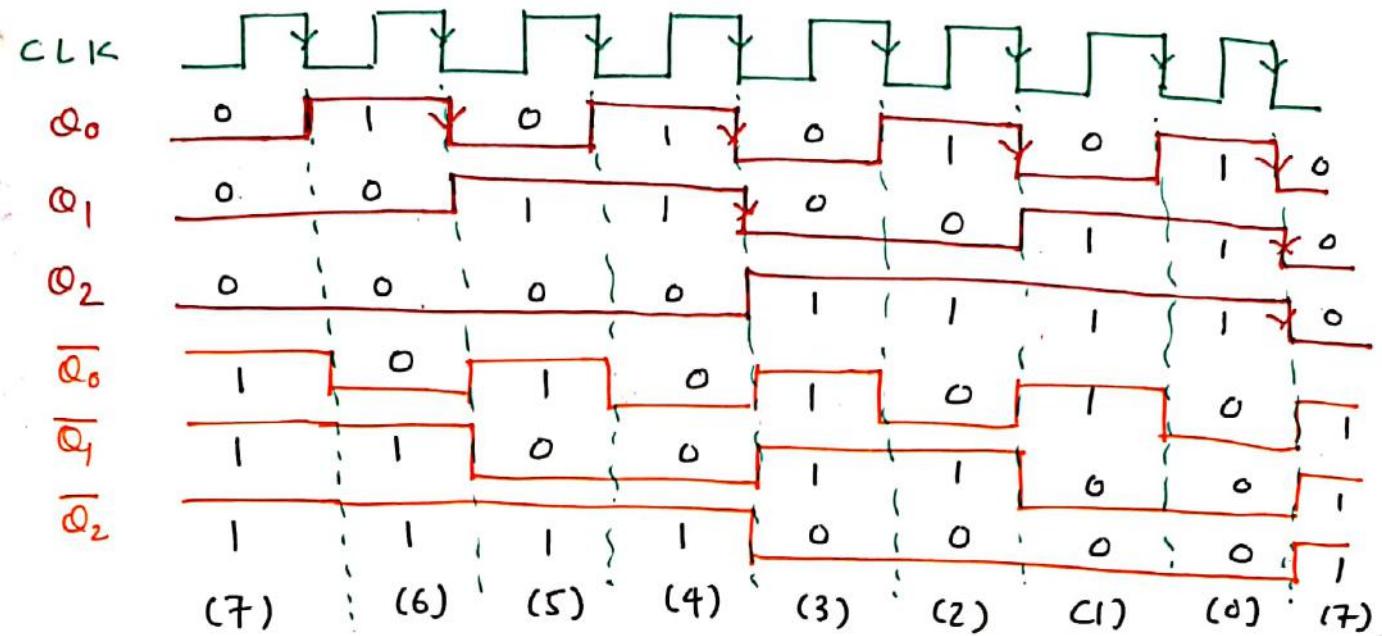
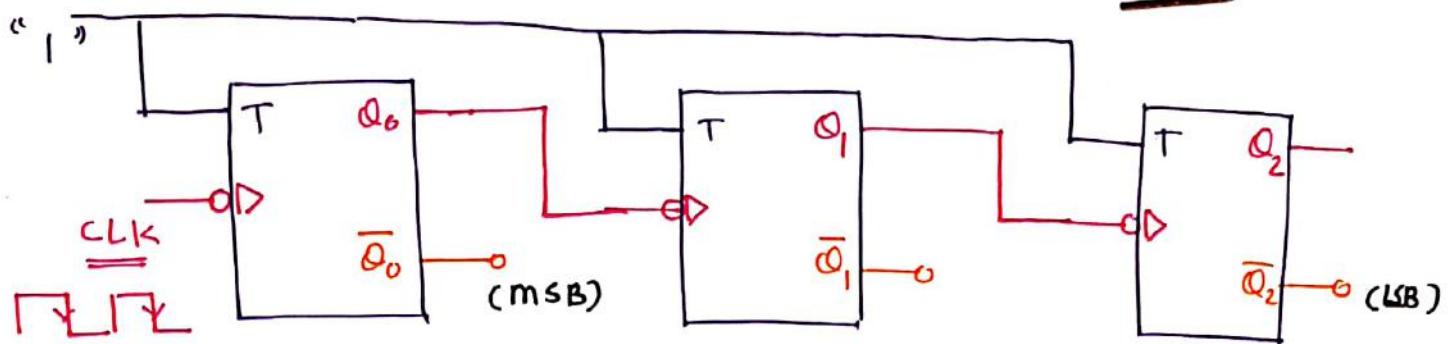


# Asynchronous (Ripple) Up Counter 166



# Asynchronous ( Ripple ) down Counter

167

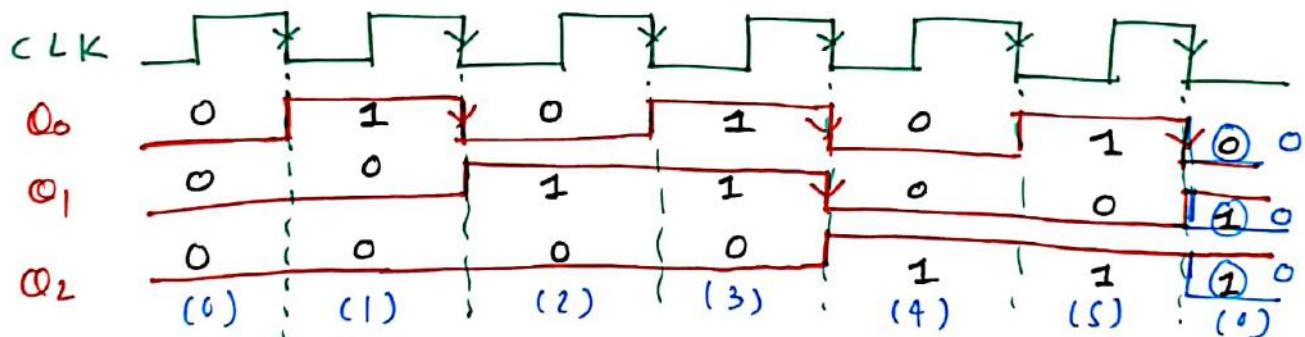
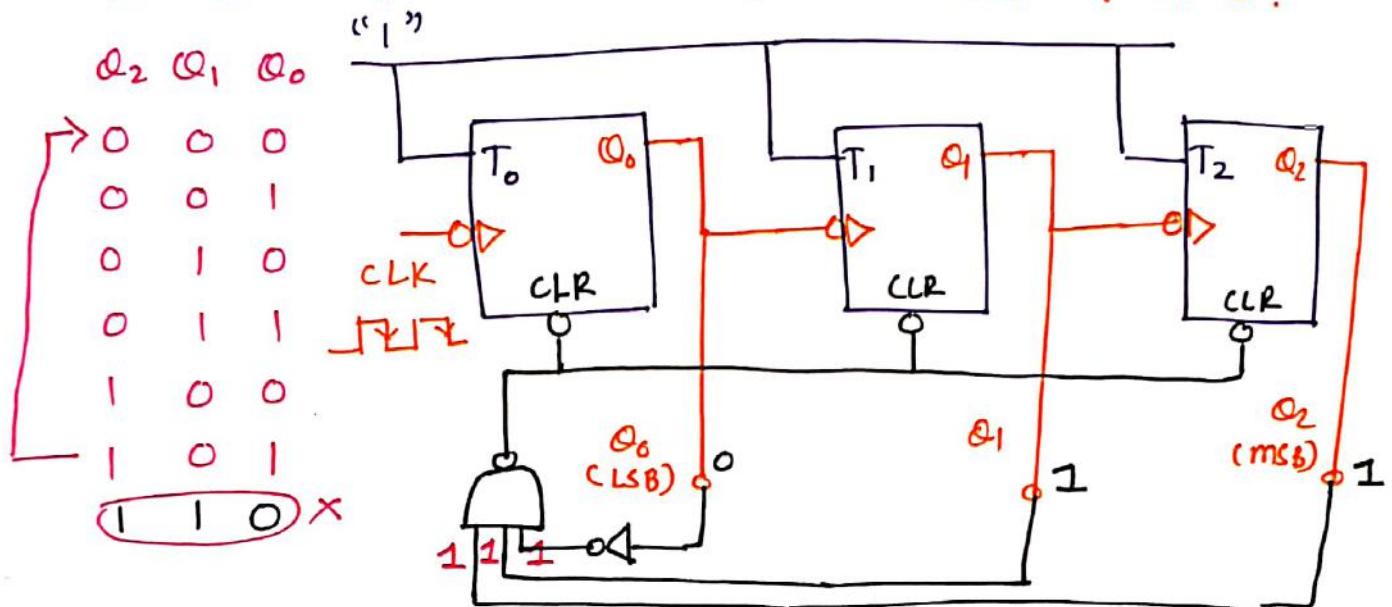


# Modulo Counter by Asynchronous Counter 168

- By 2 bits counter, At max we can have Mod 4 counter
- By 3 bits counter, At max we can have Mod 8 counter
- By 4 bits counter, At max we can have mod 16 counter.

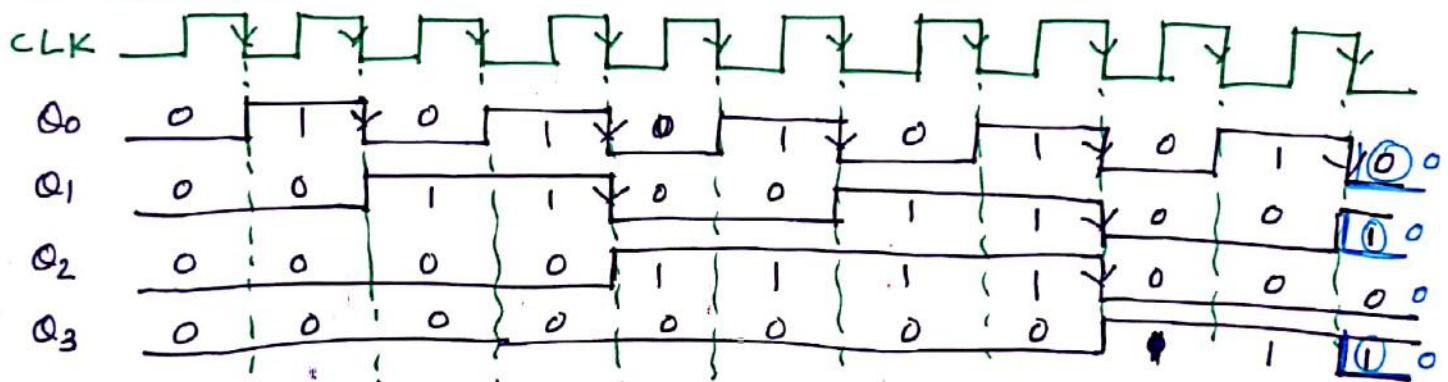
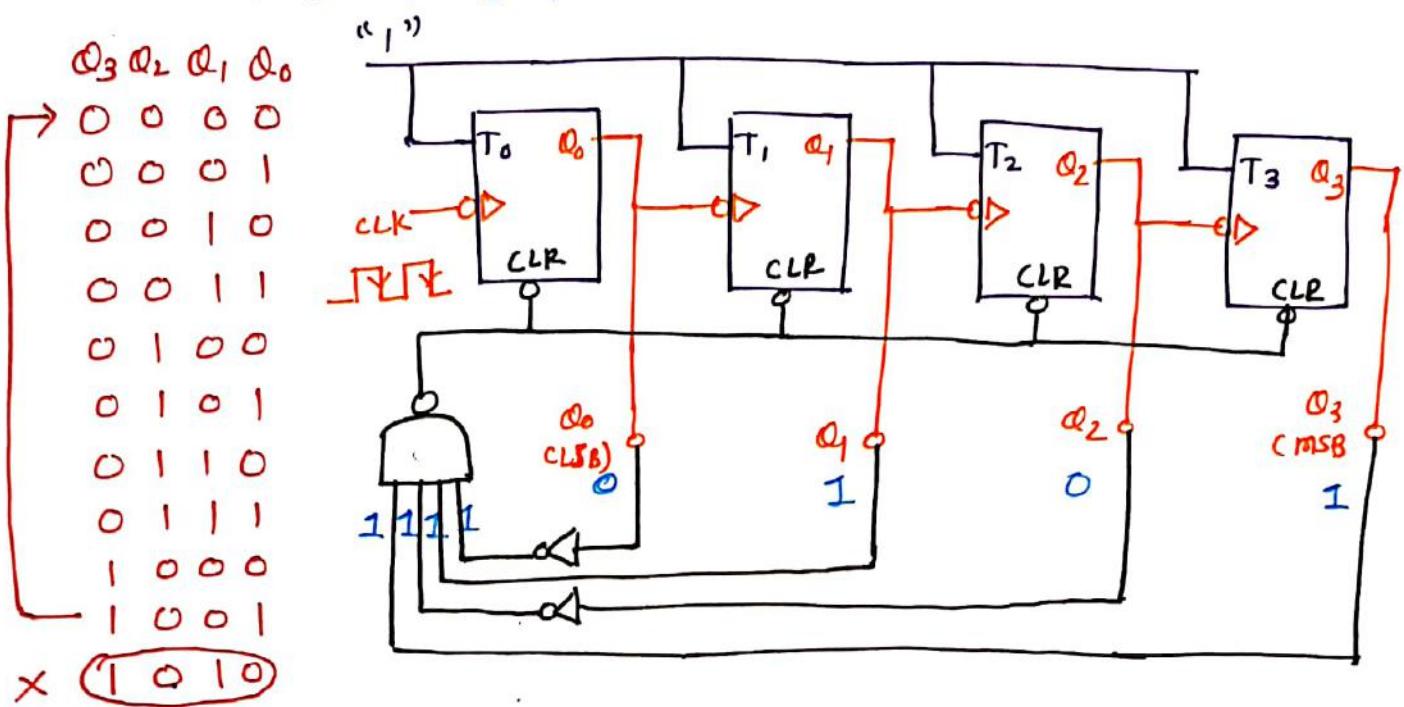
→ E.g. Modulo 6 counter.

It starts from 0 and it will count up to 5.



# BCD [Decade] Counter 169

- In BCD Counter, it counts (0 to 9).
- To count (0 to 9), it takes 4 bits.



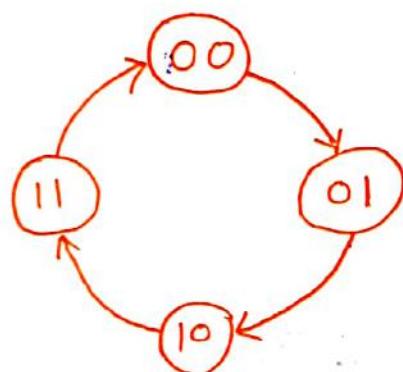
2 bits Synchronous Counter by JK Flip Flop. 170

Step 1 ;  $n = 2$  bits, flip flop = JK Flip Flop.

Step 2 Excitation Table of JK Flip Flop.

$Q_n Q_{n+1}$	J	K
0 0	0	x
0 1	1	x
1 0	x	1
1 1	x	0

Step 3 State Diagram & State table



$Q_1 Q_0$	$Q_1^+ Q_0^+$	$J_1 K_1$	$J_0 K_0$
0 0	0 1	0 x	1 x
0 1	1 0	1 x	x 1
1 0	1 1	x 0	1 x
1 1	0 0	x 1	x 1

Step 4 Find Boolean expression

$J_1$	$Q_0$	.
0	0	1
1	x	x

$$J_1 = Q_0$$

$K_1$	$Q_0$	.
0	x	x
1	0	1

$$K_1 = \bar{Q}_0$$

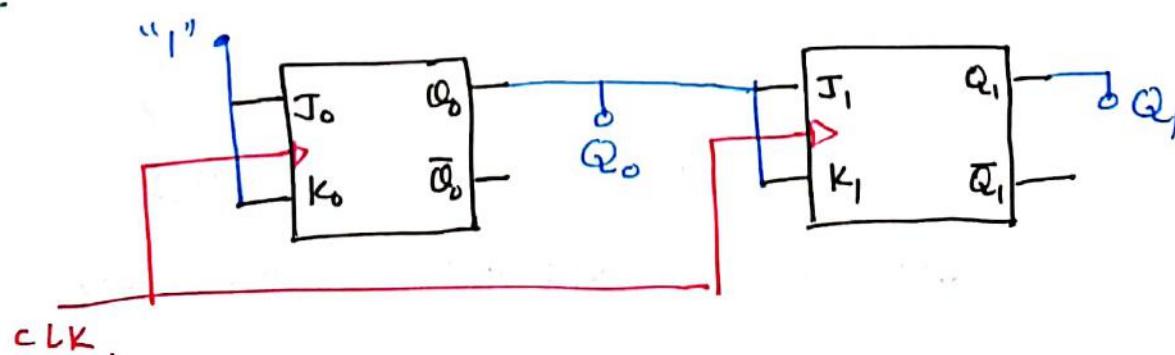
$J_0$	$Q_0$	.
0	1	x
1	1	x

$$J_0 = 1$$

$K_0$	$Q_0$	.
0	x	1
1	x	1

$$K_0 = 1$$

Step 5 Make circuit



CLK.

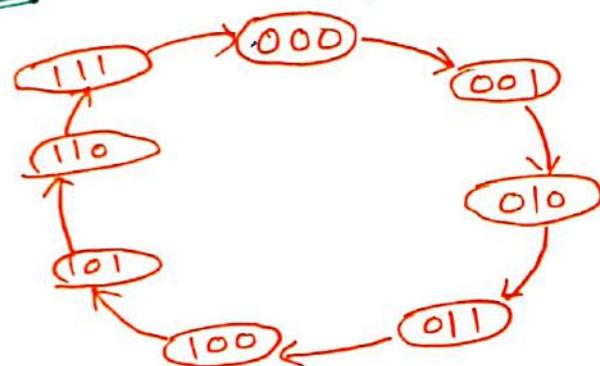
# 3 bits Synchronous Counter by T Flip Flop 171

Step-1 -  $n = 3$  bits , Flip Flop = T Flip Flop.

Step-2 - Write Excitation table of T flip flop.

$Q_n$	$Q_{n+1}$	T
0	0	0
0	1	1
1	0	1
1	1	0

Step-3 - State diagram & State Table



$Q_2 \ Q_1 \ Q_0$	$Q'_2 \ Q'_1 \ Q'_0$	$T_2 \ T_1 \ T_0$
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 0
1 0 1	1 1 0	0 1 1
1 1 0	0 0 0	1 1 1
1 1 1	0 0 0	1 1 1

Step-4 Find boolean expression

$T_2$	$Q_1 \ Q_0$
0	00 01 11 10
1	00 01 11 10

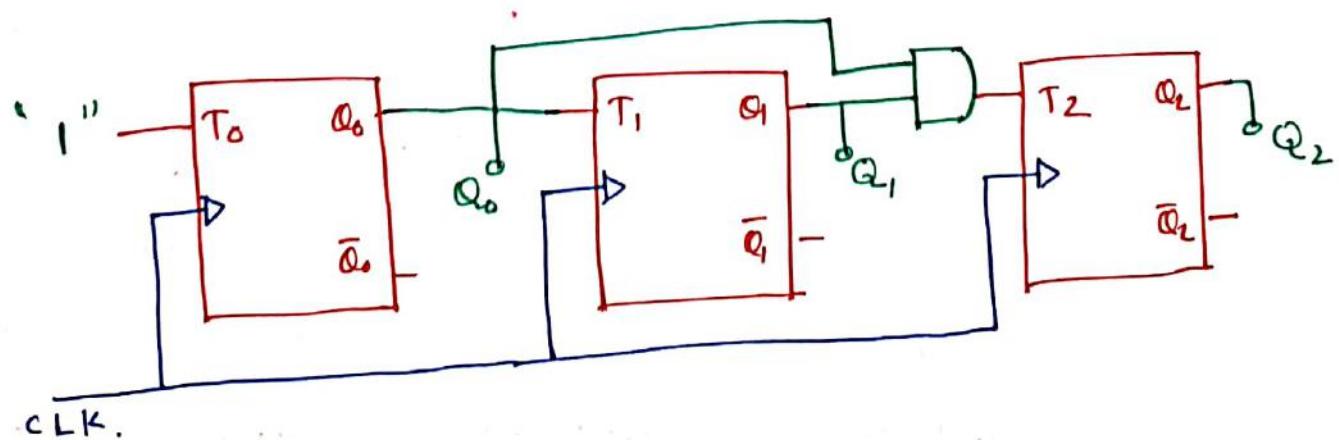
$T_2 = Q_1 Q_0$

$T_1$	$Q_1 \ Q_0$
0	00 01 11 10
1	00 01 11 10

$T_1 = Q_0$

$$T_0 = 1$$

$$T_0 = 1$$



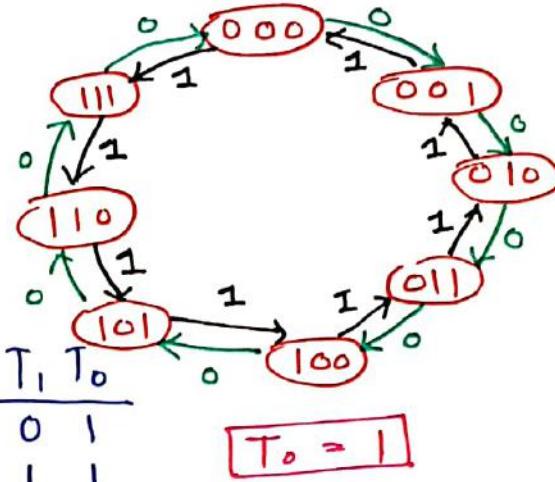
# 3 bits Synchronous up/down Counter 172

Step 1  $n = 3$  bits, Flip Flop = T flip flop.

Step 2 Excitation table of T flip flop.

$Q_n Q_{n+1}$	T
0 0	0
0 1	1
1 0	1
1 1	0

$m = 0$ , up counter  
 $m = 1$ , Down counter.



Step 3 State Diagram & State Table.

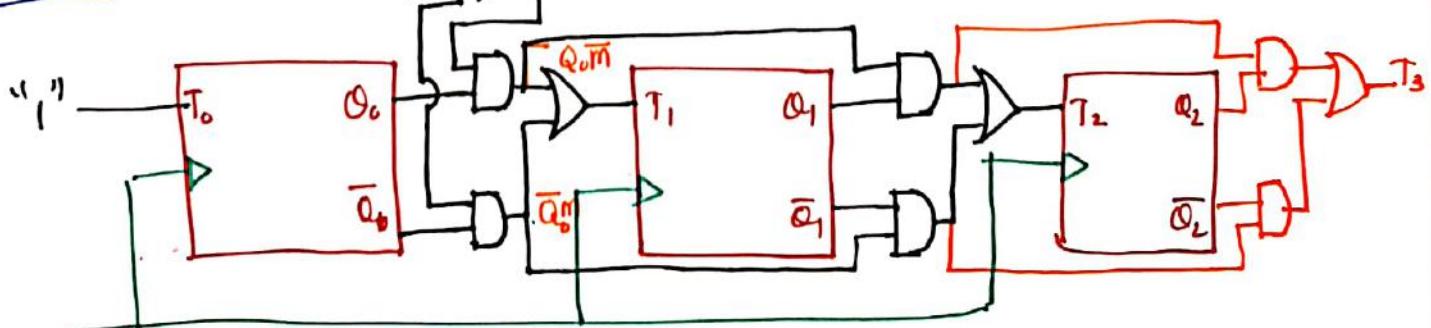
m	$Q_2 Q_1 Q_0$				$Q_2^+ Q_1^+ Q_0^+$			$T_2 \quad T_1 \quad T_0$		
	0	0	0	0	0	0	1	0	0	1
0	0	0	1	0	0	1	0	0	1	1
0	0	1	0	0	0	1	1	0	0	1
0	0	1	1	0	1	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	1
0	1	0	1	0	1	1	0	1	1	1
0	1	1	0	0	1	1	1	1	1	1
0	1	1	1	0	0	0	0	0	1	1
1	0	0	0	0	1	1	1	1	1	1
1	0	0	1	0	0	0	0	0	0	0
1	0	1	0	0	0	0	1	0	0	1
1	0	1	1	0	0	1	1	0	0	1
1	1	0	0	0	0	1	0	0	1	1
1	1	0	1	0	0	1	0	1	1	1
1	1	1	0	0	1	1	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

$$T_1 = m Q_2 \quad T_1 = Q_0 \bar{m} + \bar{Q}_0 m$$

$Q_1 Q_0$	00	01	11	10
00	0	0	1	1
01	1	1	0	0
11	1	1	0	0
10	0	0	1	1

$$T_2 = m Q_2 \quad T_2 = Q_1 Q_0 \bar{m} + \bar{Q}_1 \bar{Q}_0 m$$

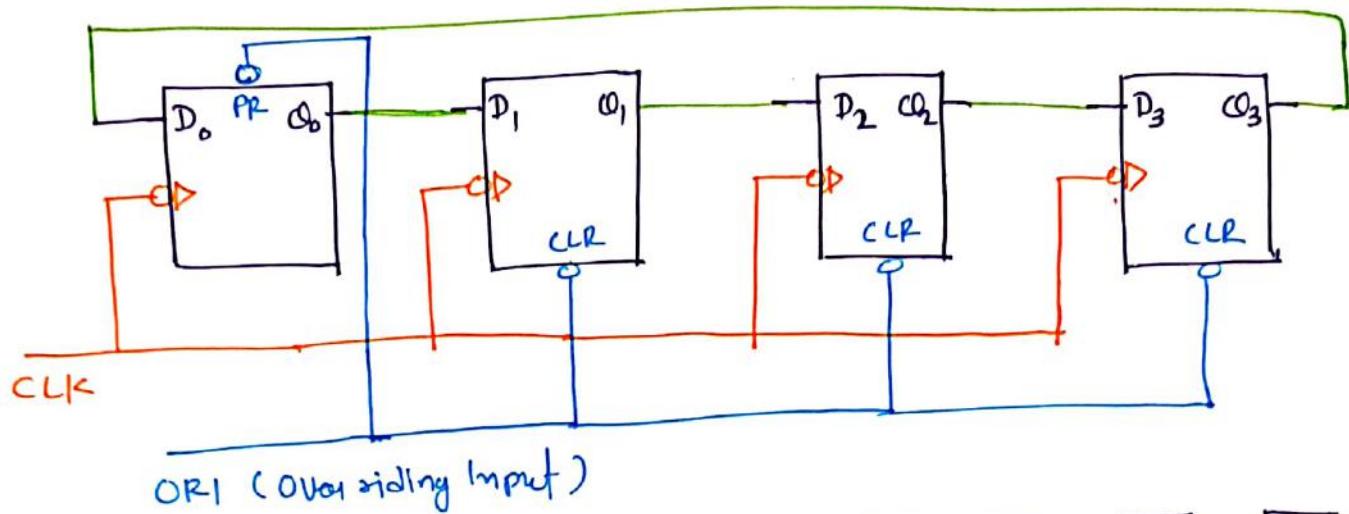
$Q_1 Q_0$	00	01	11	10
00	0	0	1	1
01	0	0	0	0
11	1	1	0	0
10	0	0	0	0



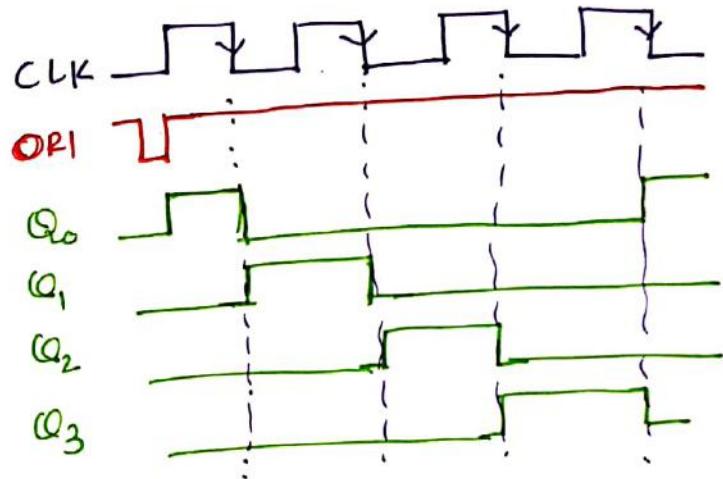
# Ring Counter 173

→ It is application of shift register.

→ No of states = no of bits  
= no of Flip Flop.



ORI	CLK	$Q_0$	$Q_1$	$Q_2$	$Q_3$
X	X	1	0	0	0
1	↓	0	1	0	0
1	↓	0	0	1	0
1	↓	0	0	0	1
1	↓	1	0	0	0



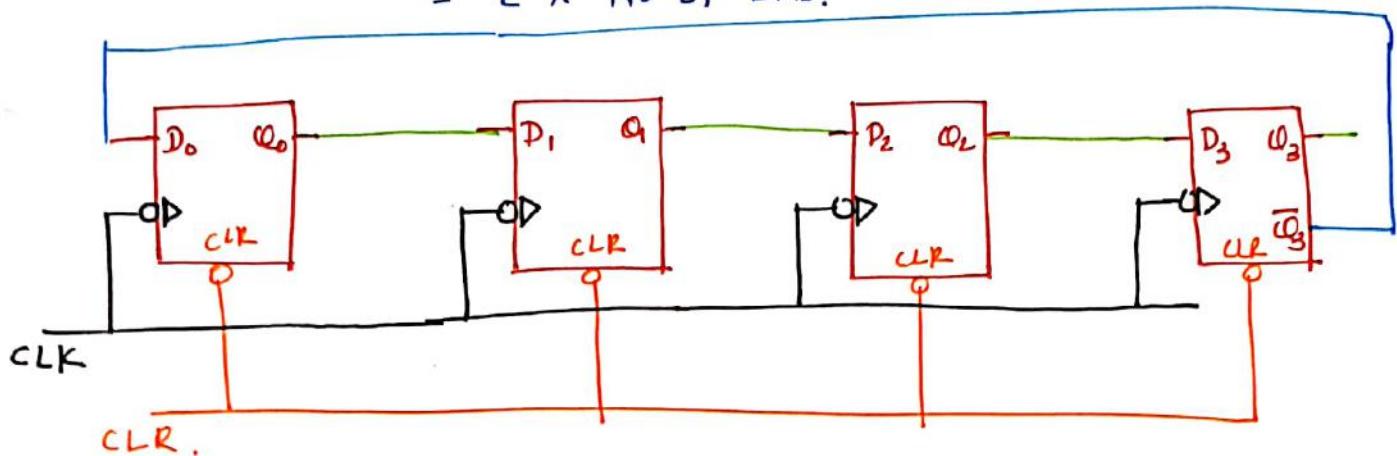
Johnson's Counter / Twisted Ring Counter / Switch Tail

→ It is application of shift register.

→ No of states =  $2 \times$  No of flip flop.  
 $= 2 \times$  No of bits.

174

Ring Counter



CLR	CLK	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>	CLK	CLR	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>
X		0	0	0	0						
1	↓	1	0	0	0						
1	↓	1	1	0	0			Q <sub>0</sub>			
1	↓	1	1	1	0			Q <sub>1</sub>			
1	↓	1	1	1	1			Q <sub>2</sub>			
1	↓	0	1	1	1			Q <sub>3</sub>			
1	↓	0	0	1	1						
1	↓	0	0	0	1						
1	↓	0	0	0	0						

→ Make a counter for given sequence 0, 1, 3, 5, 6.

Step-1

Total bits = Total Flip flop = 3

Flip flop  $\Rightarrow$  T flip flop.

Step-2

Excitation Table of T flip flop.

$Q_n$	$Q_{n+1}$	T
0	0	0
0	1	1
1	0	1
1	1	0

Step-3

State Table. (0, 1, 3, 5, 6)

$Q_A$	$Q_B$	$Q_C$	$Q_A^+$	$Q_B^+$	$Q_C^+$	$T_A$	$T_B$	$T_C$
0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	0	1	0
0	1	0	X	X	X	X	X	X
0	1	1	1	0	1	1	1	0
1	0	0	X	X	X	X	X	X
1	0	1	1	1	0	0	1	1
1	1	0	0	0	0	1	1	0
1	1	1	X	X	X	X	X	X

$Q_C$	00	01	11	10
0	0	X	1	X
1	0	1	X	0

$$T_A = Q_B$$

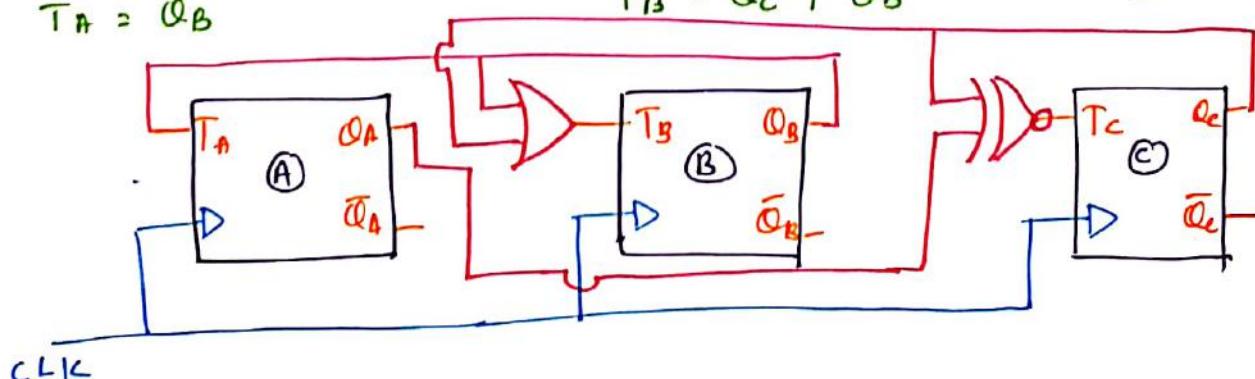
$Q_C$	00	01	11	10
0	0	X	1	X
1	1	1	X	1

$$T_B = Q_C + Q_B$$

$Q_A$	00	01	11	10
0	1	X	0	X
1	0	0	X	1

$$T_C = Q_C Q_A + \bar{Q}_C \bar{Q}_A$$

$$= Q_A \bar{Q}_C Q_C$$



# Sequence Counter. 178

→ Make a given arbitrary sequence counter which counts from 7, 4, 1, 6, 2, 5,

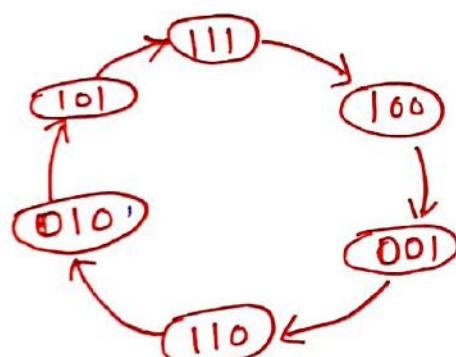
Step-1 No of Flip Flop = 3

Flip Flop = T flip flop.

Step-2 Excitation Table of T flip flop

$Q_n$	$Q_{n+1}$	T
0	0	0
0	1	1
1	0	1
1	1	0

Step 3 [ 7, 4, 1, 6, 2, 5 ]



$Q_A$	$Q_B$	$Q_C$	$Q_A^+$	$Q_B^+$	$Q_C^+$	$T_A$	$T_B$	$T_C$
0	0	0	x	x	x	x	x	x
0	0	1	1	1	0	1	1	1
0	1	0	1	0	1	1	1	1
0	1	1	x	x	x	x	x	x
1	0	0	0	0	1	1	0	1
1	0	1	1	1	1	0	1	0
1	1	0	0	1	0	1	0	0
1	1	1	1	0	0	0	1	1

$T_A$	$Q_A Q_B$
$\bar{Q}_A + \bar{Q}_C$	00 01 11 10
0 1 X X 0 0	X 1 1 0 0

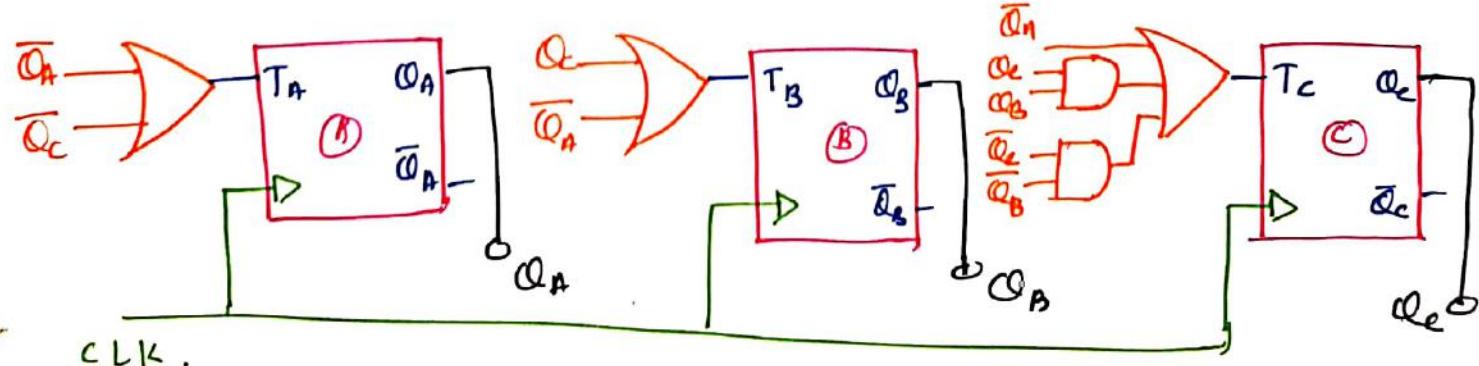
$T_B$	$Q_A Q_B$
$Q_C + \bar{Q}_A$	00 01 11 10
0 1 X X 0 0	X 1 0 0 0

$T_C$	$Q_A Q_B$
$\bar{Q}_A + Q_C Q_B + \bar{Q}_C \bar{Q}_B$	00 01 11 10
0 1 X X 0 0	X 1 0 1 0

$$T_A = \bar{Q}_A + \bar{Q}_C$$

$$T_B = Q_C + \bar{Q}_A$$

$$T_C = \bar{Q}_A + Q_C Q_B + \bar{Q}_C \bar{Q}_B$$



# Sequence Generator 177

→ Generate Sequence of 10110.....

→ No of flip flop can be identify by

$$\Rightarrow L \leq 2^n - 1$$

where,  $L = \text{Length of Sequence}$

$n = \text{Number of Flip Flop (Minimum)}$ .

$$\Rightarrow 5 \leq 2^n - 1$$

$$\Rightarrow 5 + 1 \leq 2^n$$

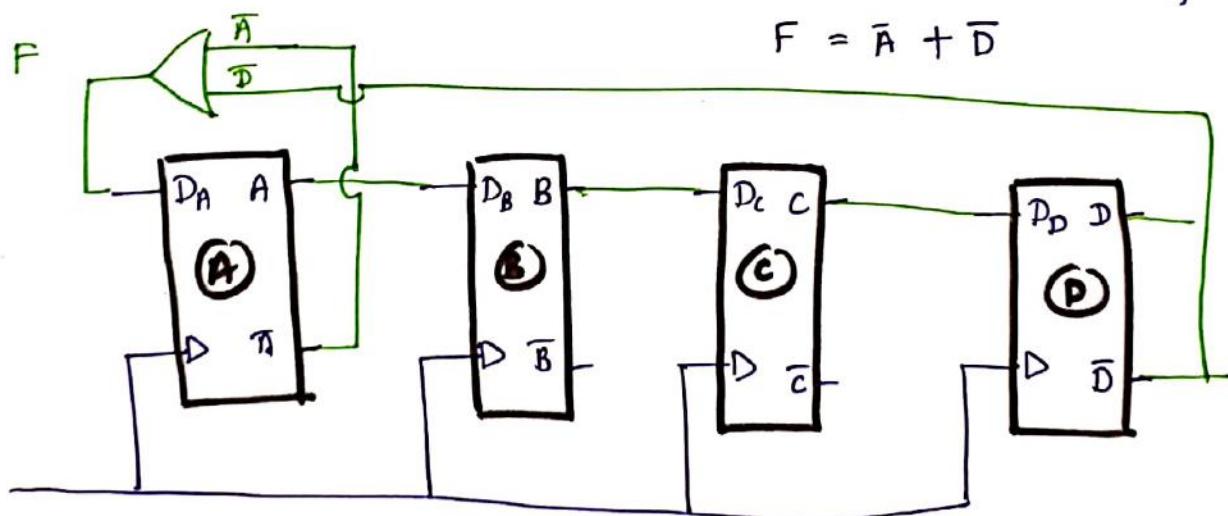
$$\Rightarrow 6 \leq 2^n$$

$$\Rightarrow n = 3$$

A	B	C	D	F (Feedback)
-1	0	1	1	0
0	1	0	1	1
-1	0	1	0	1
1	1	0	1	0
0	1	1	0	1

AB	00	01	11	10
CD	X	X	X	X
00	X	1	0	X
01	X	1	0	X
11	X	X	X	0
10	X	1	X	1

$$F = \bar{A} + \bar{D}$$



# Sequence Generator 178

→ Generate Sequence of 110010.....

→ No of flip flop can be identified by.

$$L \leq 2^n - 1$$

where,

$L$  = Length of Sequence

$n$  = Number of Flip flop (Minimum).

$$\Rightarrow 6 \leq 2^n - 1$$

$$\Rightarrow 6 + 1 \leq 2^n$$

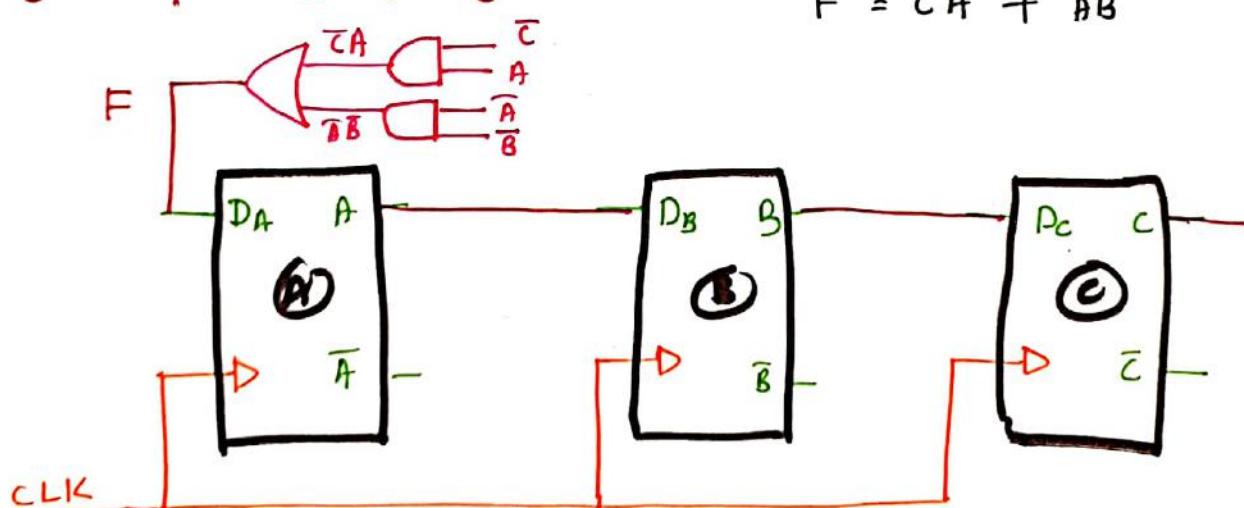
$$\Rightarrow 7 \leq 2^n$$

$$\Rightarrow n = 3$$

A	B	C	F (Feed back)
-1	0	1	0
-1	1	0	1
-0	1	1	0
-0	0	1	1
-1	0	0	1
-0	1	0	0

		AB	00	01	11	10
		C	0	0	1	1
F	A	0	X	0	1	1
		1	1	0	X	0

$$F = \overline{C}A + \overline{A}\overline{B}$$



# Sequence Detector 179

→ Design Circuit to detect 110 from given Input data stream.

→ If given Input data is

$$\begin{array}{l} X \quad 0111011010110 \dots \\ Y \quad 0000100100001 \end{array}$$

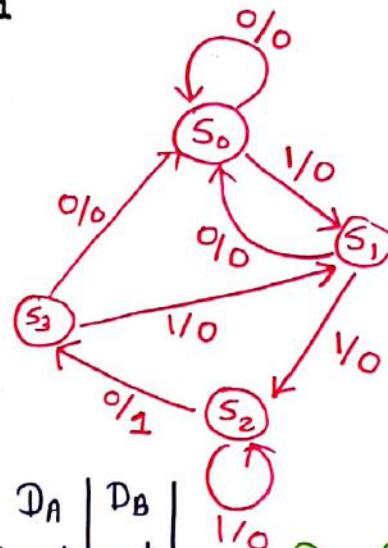
$$S_0 = 0 \text{ [ Reset ]}$$

$$S_1 = 1$$

$$S_2 = 11$$

$$S_3 = 110$$

	$Q_A$	$Q_B$
$S_0$	0	0
$S_1$	0	1
$S_2$	1	0
$S_3$	1	1



$D_A$	$Q_A Q_B$	$x$	$Q_A^+$	$Q_B^+$	$\bar{x}$
0	00	0	0	0	0
0	01	0	0	1	0
0	11	0	0	0	0
1	10	1	1	0	0
0	00	1	0	0	0
0	01	1	0	1	0
0	11	1	0	0	0
1	10	1	1	1	1
0	00	0	1	0	0
0	01	0	1	1	0
0	11	0	0	0	0
1	10	1	0	1	1

$D_B$	$Q_A Q_B$	$x$	00	01	11	10
0	00	0	0	0	0	1
1	01	0	1	0	1	0
0	11	0	0	0	0	0
1	10	1	0	1	0	0

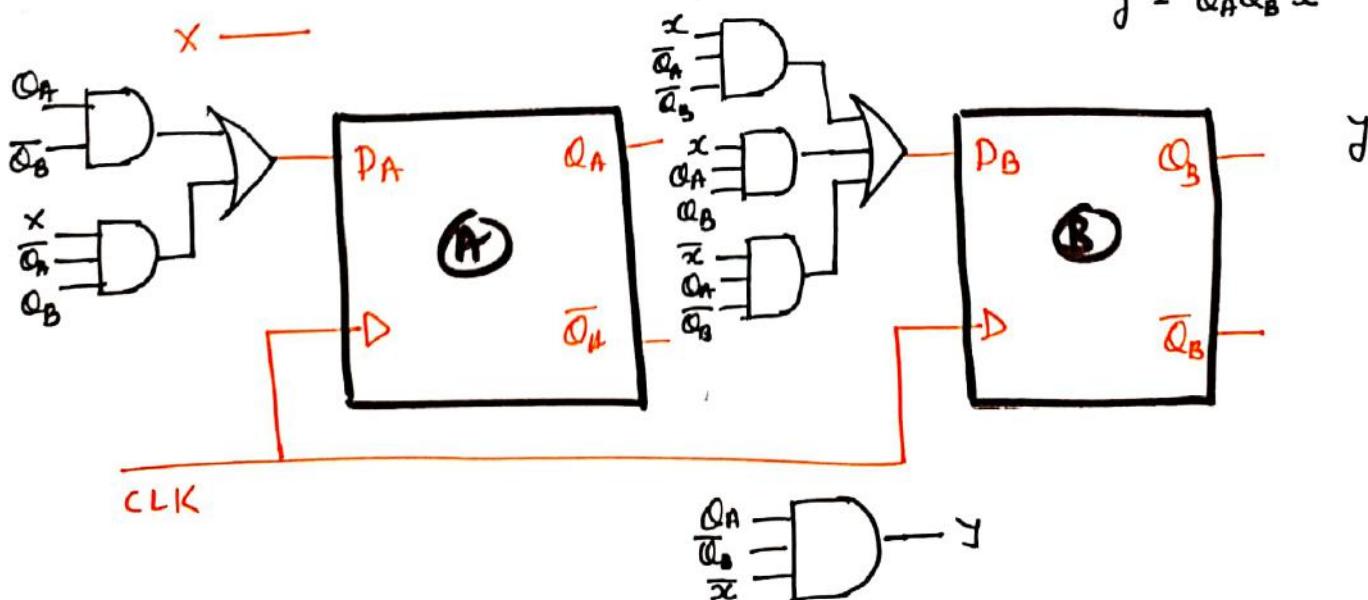
$$D_B = x\bar{Q}_A\bar{Q}_B + \bar{x}Q_AQ_B$$

$D$	$Q_A Q_B$	$x$	00	01	11	10
0	00	0	0	0	0	1
1	01	0	1	0	1	0
0	11	0	0	0	0	0
1	10	1	0	1	0	0

$$D = Q_A Q_B \bar{x}$$

$D_A$	$Q_A Q_B$	$x$	00	01	11	10
0	00	0	0	0	0	1
0	01	0	1	0	1	0
0	11	0	0	0	0	0
1	10	1	0	1	0	0

$$D_A = Q_A \bar{Q}_B + x\bar{Q}_A Q_B$$



# Sequence Detector 180

→ Make a Circuit which detects 111 from given Input data.

$$X = 101111011111$$

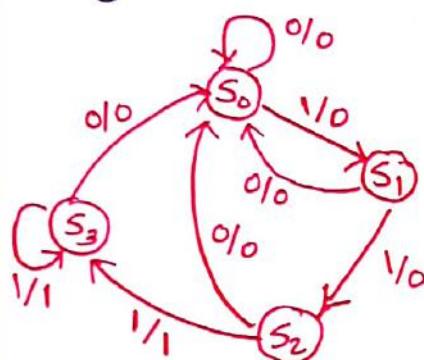
$$Y = 000011000111$$

$$S_0 = 0 \quad [\text{Reset}] \quad [00]$$

$$S_1 = 1 \quad [01]$$

$$S_2 = 11 \quad [10]$$

$$S_3 = 111 \quad [11] \quad Q_A Q_B$$



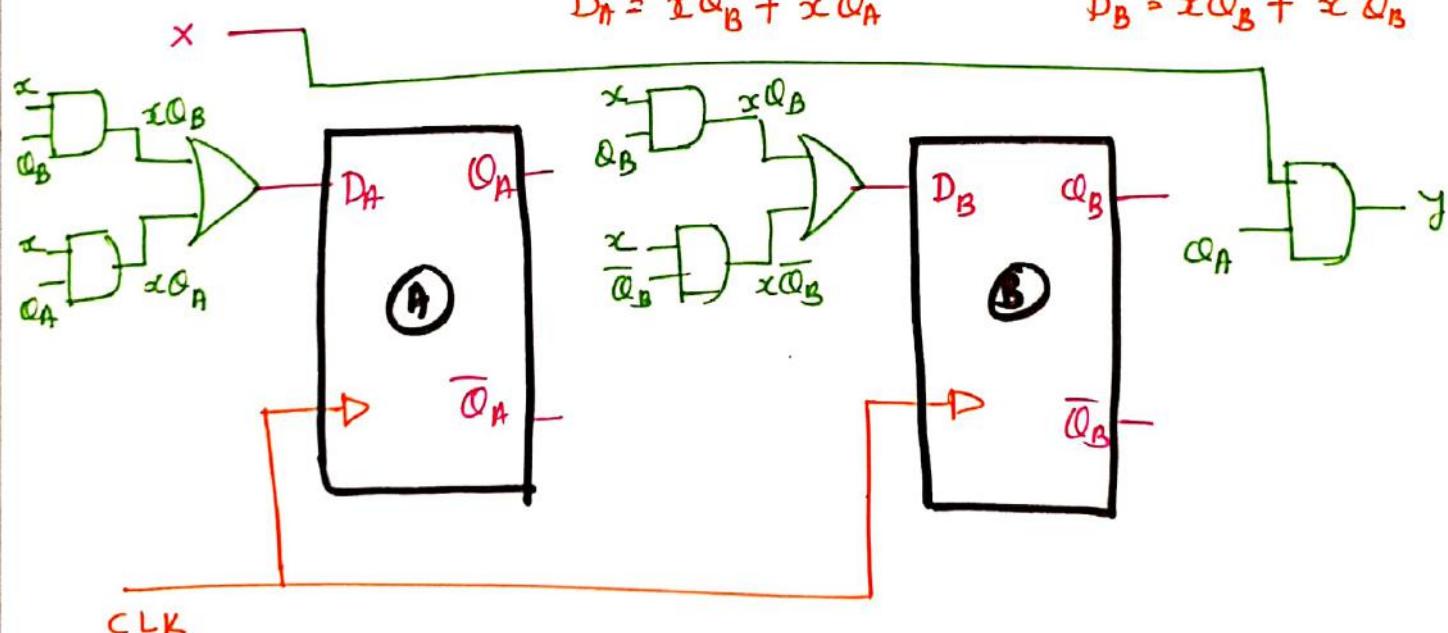
$Q_A$	$Q_B$	$x$	$D_A = Q_A^+$	$D_B = Q_B^+$	$y$
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0	0	1	1
1	1	1	1	1	1

$D_A \ Q_A Q_B$	00	01	11	10
0	0	0	0	0
1	0	11	11	11

$$D_A = x Q_B + \bar{x} Q_A$$

$D_B \ Q_A Q_B$	00	01	11	10
0	0	0	0	0
1	1	0	11	11

$$D_B = x Q_B + \bar{x} \bar{Q}_A$$



## Examples on Counter. 18)

1] If Counter is having 10 FF. It is initially 0, what Count will it hold after 2060 pulses.

→ 10 FF Counter means, It is 10 bits Counter.

→ For 10 bits Counter total Count =  $2^{10}$

$$= 1024.$$

→ after 1024 Pulses, It holds 0 Count.

→ after 1024 (2048) pulses It holds 0 Count.

→ so after 2060 pulses =  $2060 - 2048$   
= 12

→ 0000001100 ← State of Counter after 2060 pulses.

2] A 4 bits Mod 16 Counter is made by JK Flip Flop. If propagation delay of each flip flop is 50 nsec. The maximum Clock freq. can be used.

- ① 4 MHz     ② 5 MHz    ③ 10 MHz    ④ 20 MHz.

- Total propagation delay =  $n \times t_{pd}$   
=  $4 \times 50$   
= 200 nsec.

- Max clock freq =  $\frac{1}{P.d} = \frac{1}{200 \text{ nsec}} = \frac{10^9}{200} = 5 \text{ MHz.}$

3] A certain JK Flip Flop has propagation delay of  $t_{pd} = 12 \text{ nsec}$ . The largest ripple counter that can operate at 10 MHz is

- ① mod 16    ② mod 8    ③ mod 256    ④ mod 128

→ Total Propagation delay =  $\frac{1}{f} = \frac{1}{10 \text{ MHz}} = 100 \text{ nsec.}$

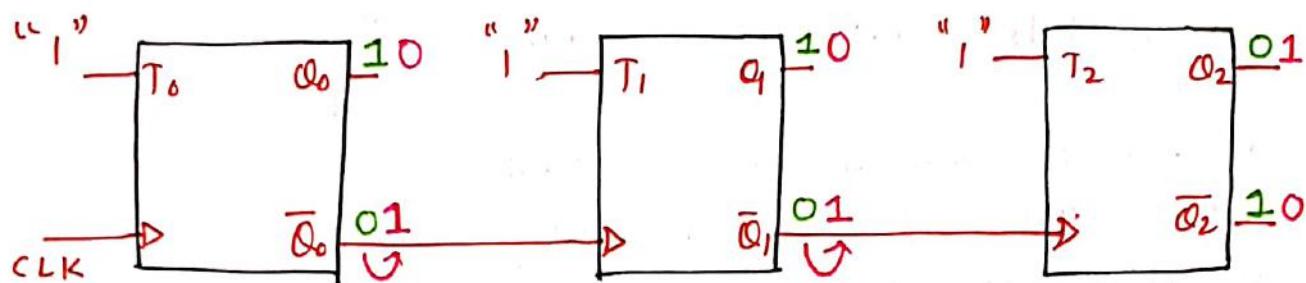
→  $n = \frac{\text{total Propagation delay}}{t_{pd}}$   
=  $\frac{100}{12} \approx 8 \text{ bits}$

→  $\text{Mod} = 2^n$   
=  $2^8$   
= 256

→ mod 256 Counter

4) A given figure shows ripple counter with positive edge trigger clock. If present state of counter is given by  $Q_2 Q_1 Q_0 = 011$ , then next state will be

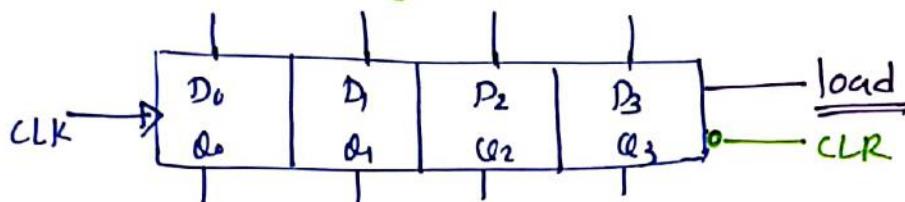
- (a) 010    (b) 100    (c) 111    (d) 101



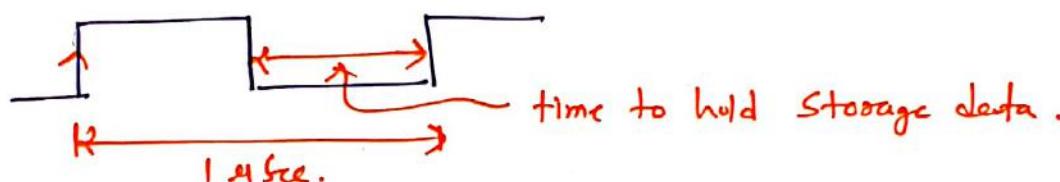
$$Q_2 Q_1 Q_0 = \boxed{100}$$

# Register 182

- Flip Flop is used to store 1 bit data.
- To increase storage capacity, we use group of flip flop, which is referred as register.
- By n number of flip flop, we can store n bits data of register which can store n bits word.
- E.g. 4 bits register.



$$\rightarrow f = 1 \text{ MHz} \rightarrow T = \frac{1}{f} = \frac{1}{1 \text{ MHz}} = 1 \mu\text{sec}$$



- load
  - Synchronous load
    - Register is operational when load & clock high.
  - Asynchronous load.
    - Register is operational when load is high.

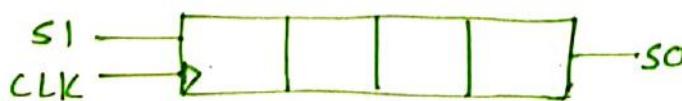
## Data Format.

- Serial [we can store one bit at a time]
- Parallel [we can store all bits at a time]

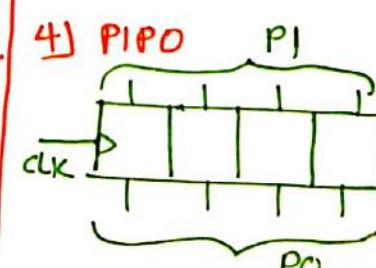
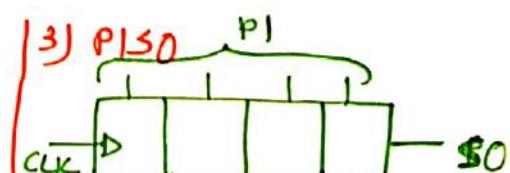
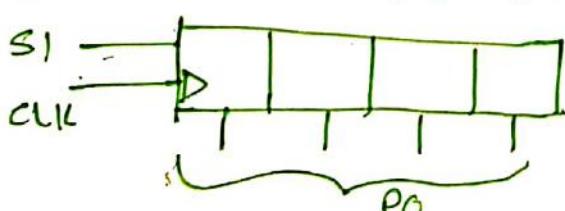
## Classification of Register

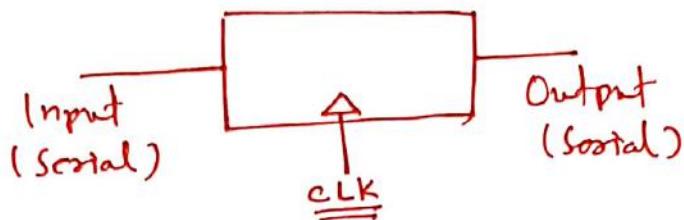
- we can classify Register based on Input and output.

1) SISO [Serial Input & Serial Output].



2) SIPO [Serial Input & Parallel Output].

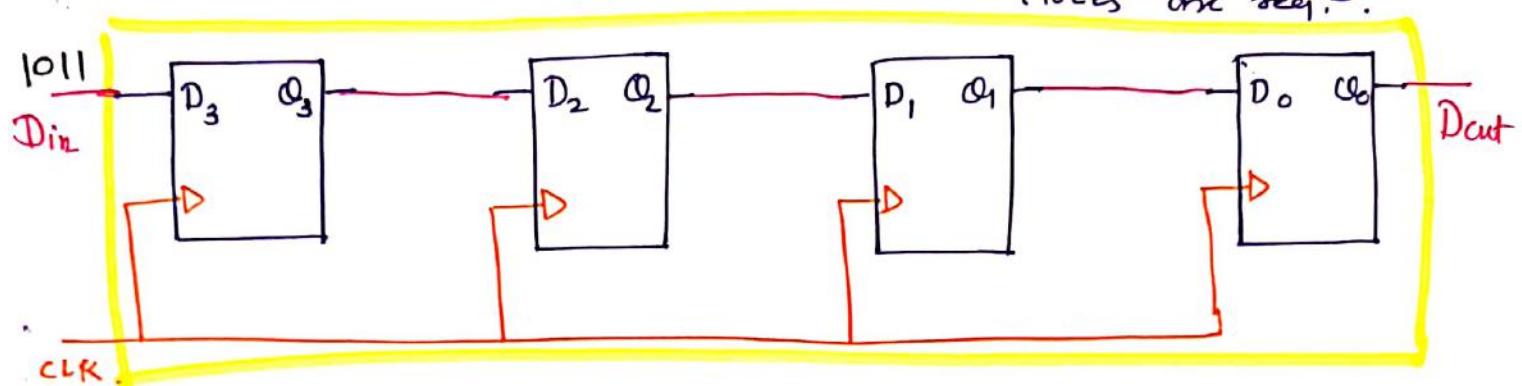




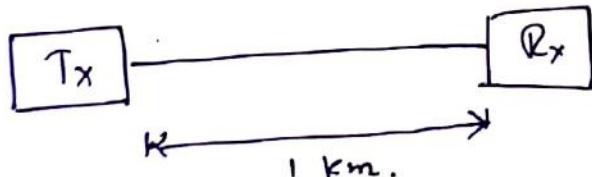
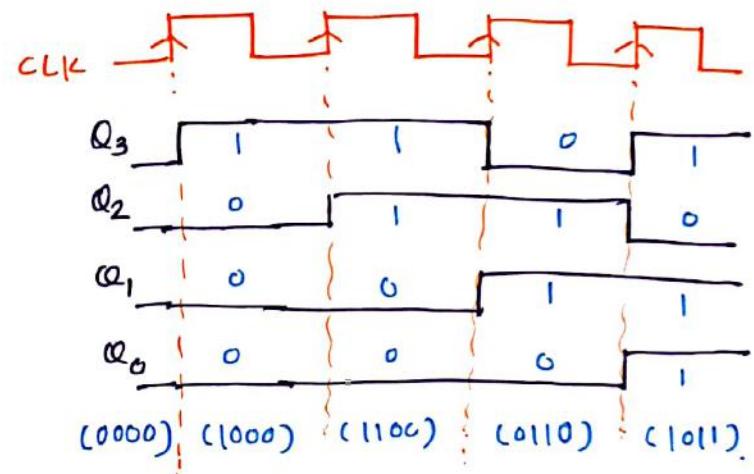
- To store data  $n$  bits  
are req'd.

- To fetch data  $(n-1)$  bits  
are req'd

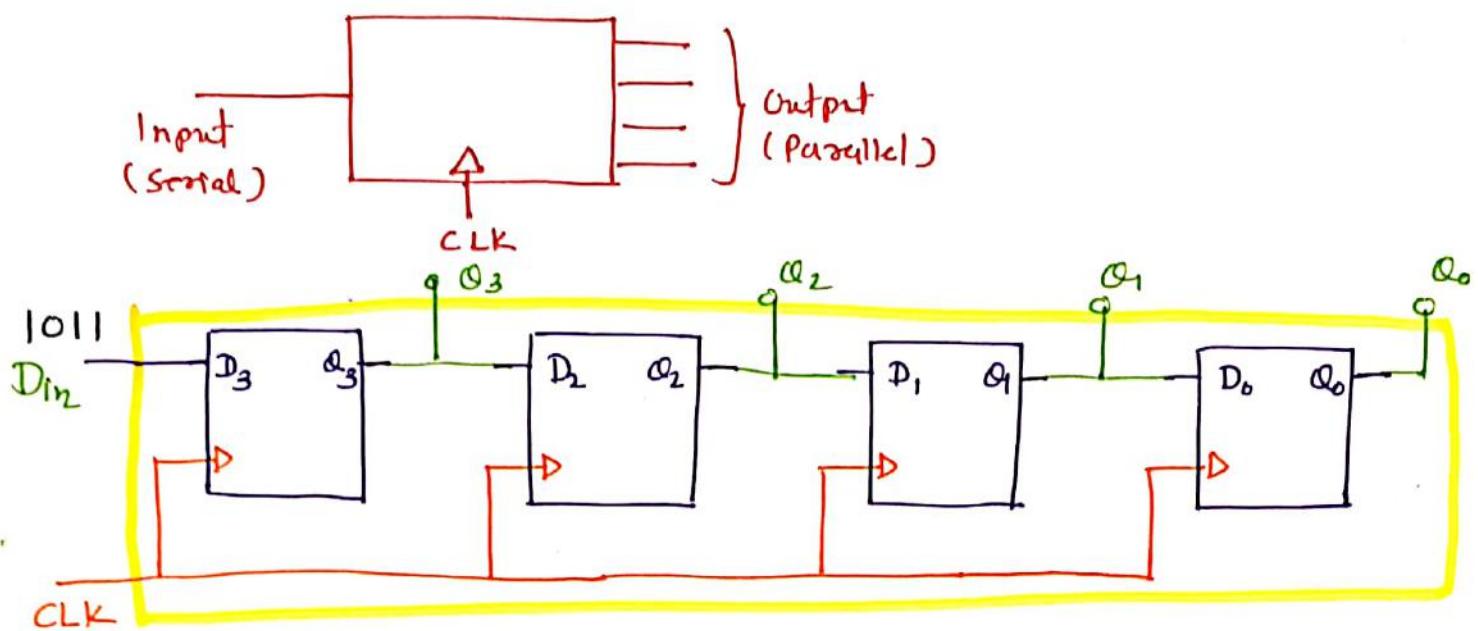
- For total processing  $2^{n-1}$   
(clocks) are req'd.



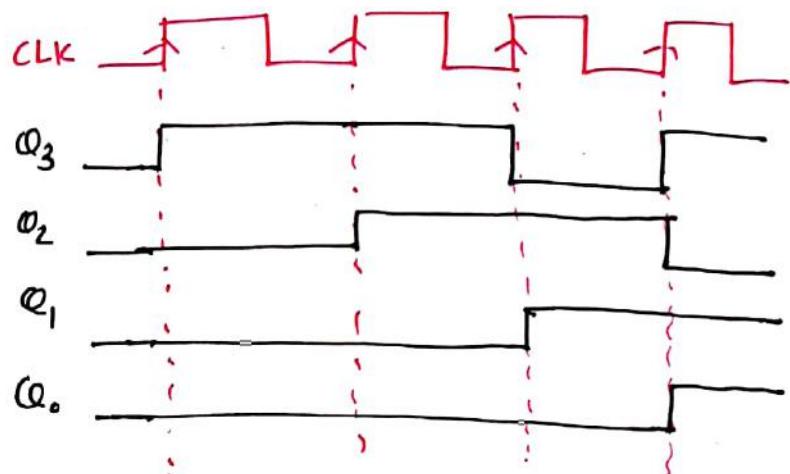
CLK	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
Initially	0	0	0	0
↑	1	0	0	0
↑	1	1	0	0
↑	0	1	1	0
↑	1	0	1	1



# S1PO Shift Register 184



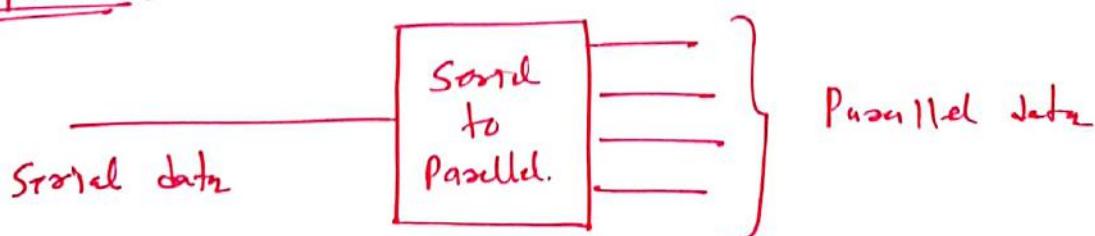
	CLK	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
Initially		0	0	0	0
↑		1	0	0	0
↑		1	1	0	0
↑		0	1	1	0
	↑	1	0	1	1



→ [ n clock for storing data.  
0 clock for use of data.  
n clock for total processing.]

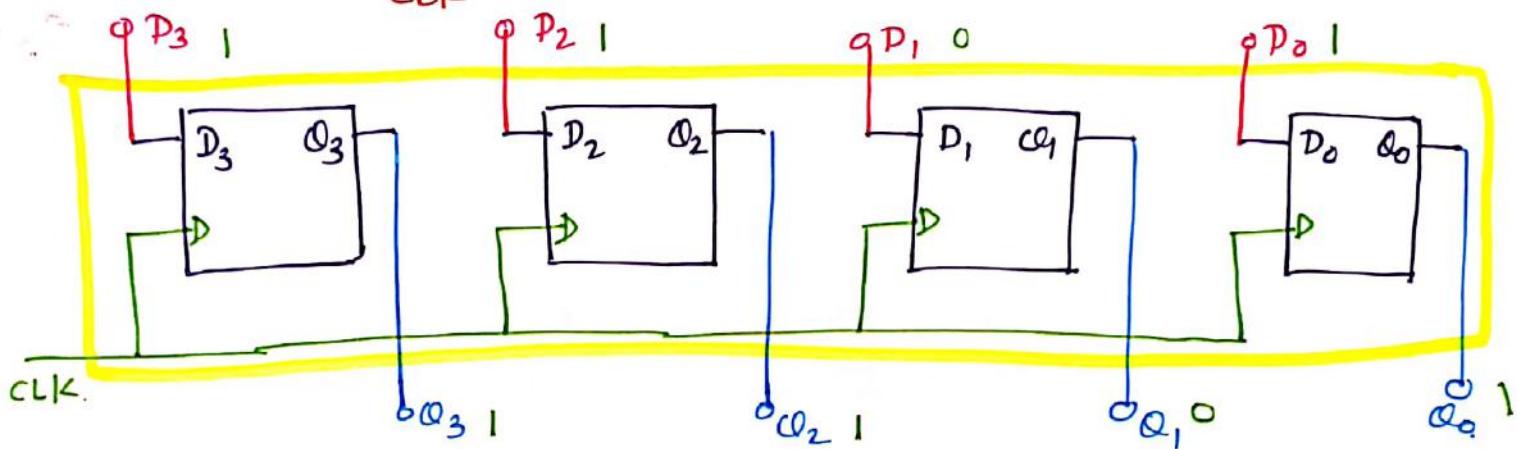
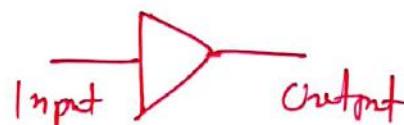
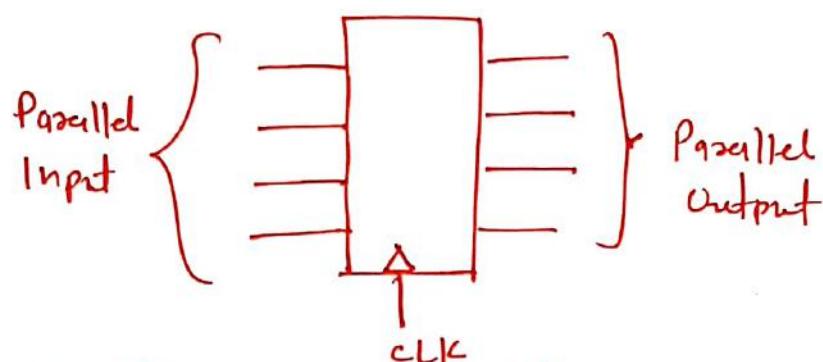
S150  
- n for storing data  
- (n-1) for use of data  
- 2n-1 clock for total processing.

## Application

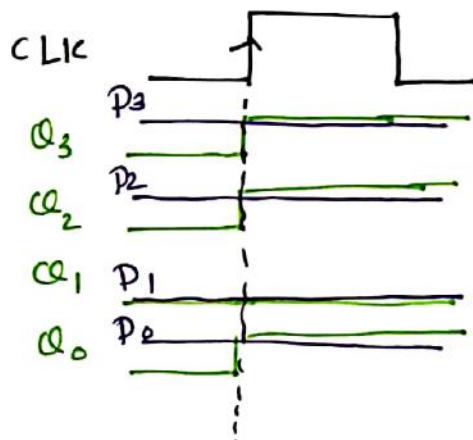


# PIPO Register / Storage Register / Buffer Register

185

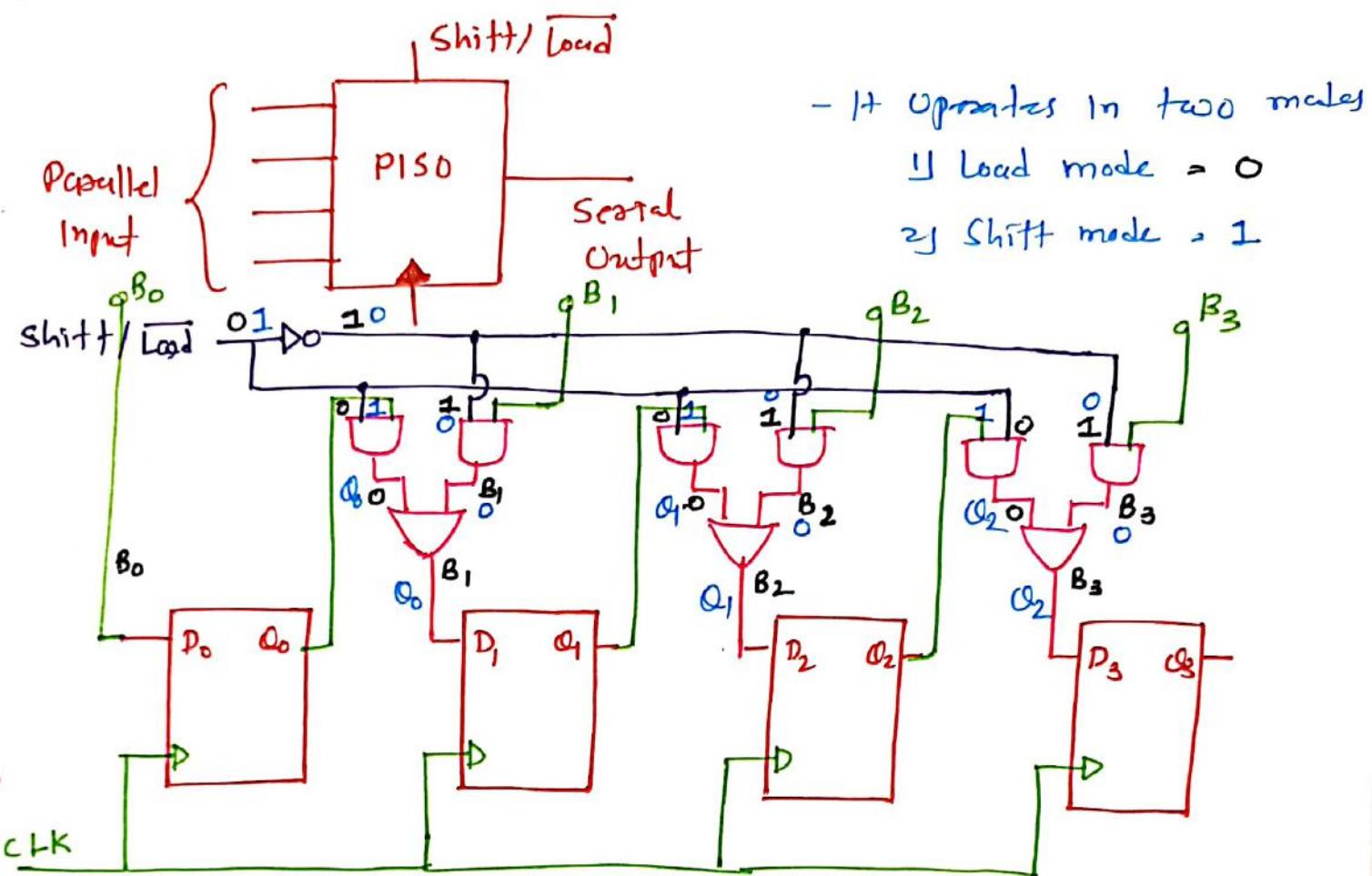


- We need 1 clock to store data.
- we need 0 clock to read the data.
- Total processing acc. to 1 clock.



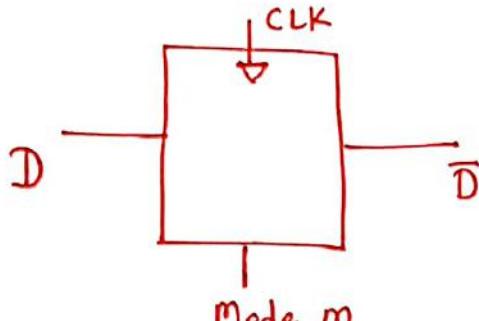
# PISO Shift Register

186

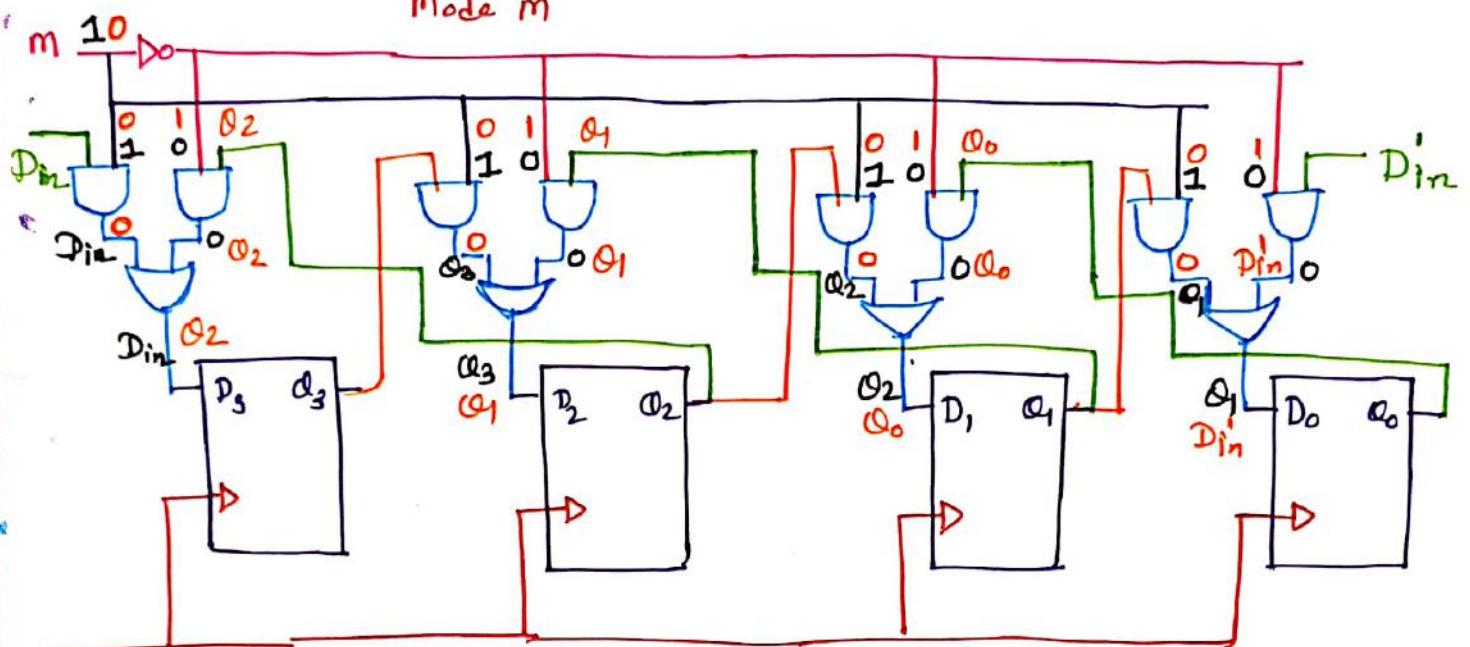


- It operates in two modes
  - 1) Load mode = 0
  - 2) Shift mode = 1

- 1 clock to Load data.
- $(n-1)$  clock to use data.
- $n$  clock is used for total processing.



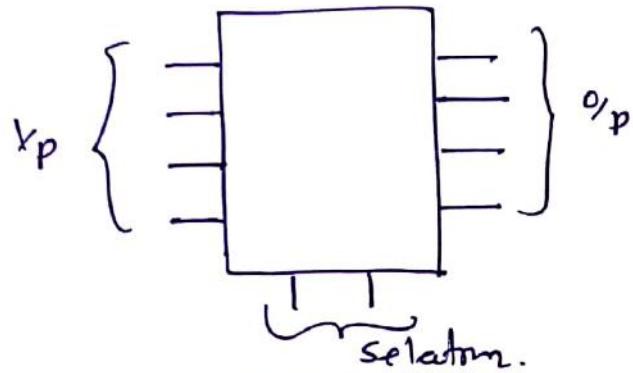
$m = 1$ , Shift Right  
 $m = 0$ , Shift Left.



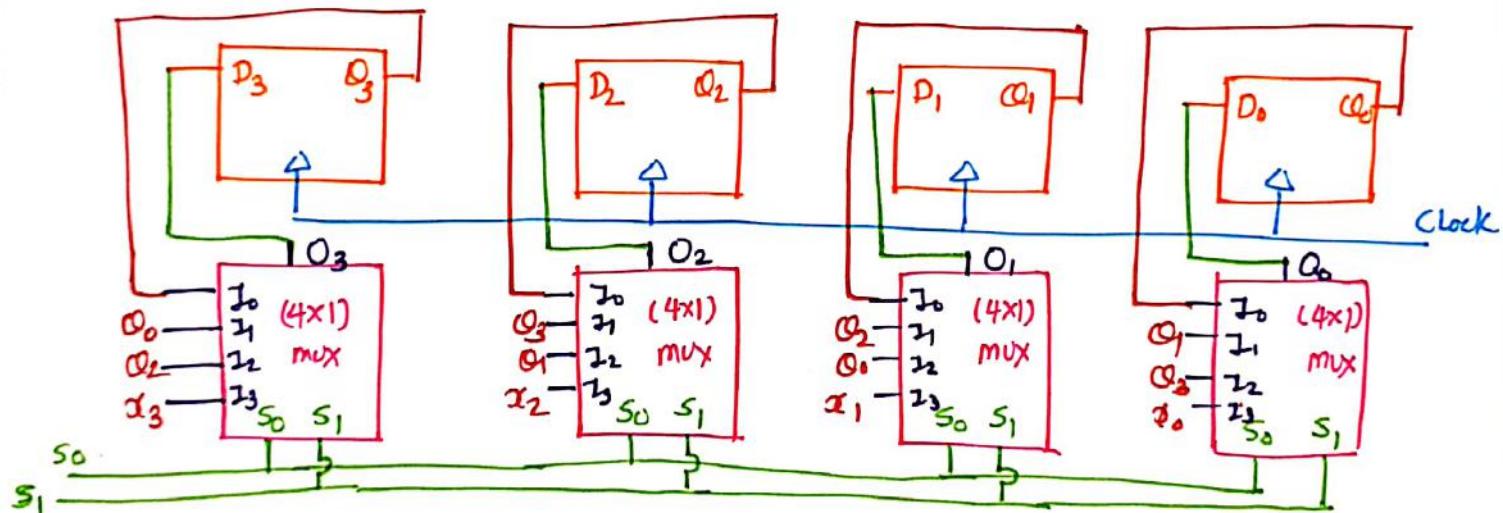
# Universal Shift Register 188

- Universal Shift Register has following function

- 1] Shift Left
- 2] Shift Right
- 3] PIPD
- 4] Hold Data



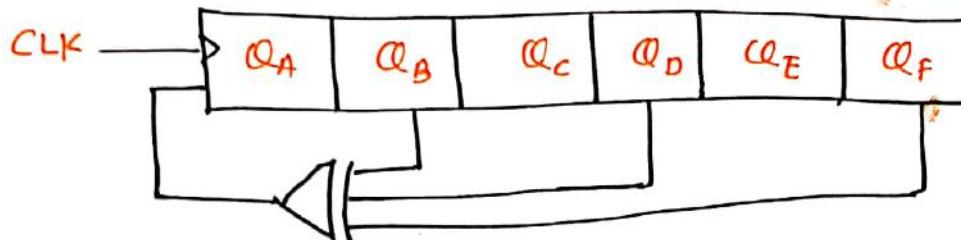
$S_0$	$S_1$	function
0	0	Hold
0	1	Right shift
1	0	Left shift
1	1	PIPO



# Examples on Register

189

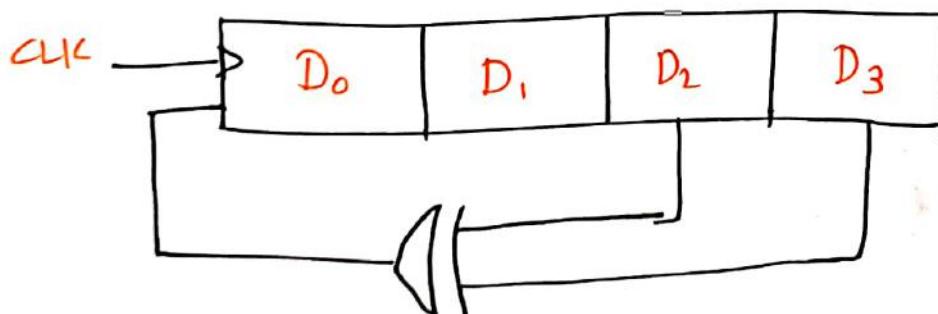
- 1] 6 bits serial in Parallel Out Shift register is shown in figure. If initial data of shift register is 110101. Then find output of shift register after three clock.



CLK	$V_p$	$Q_A$	<u><math>Q_B</math></u>	$Q_C$	<u><math>Q_D</math></u>	$Q_E$	<u><math>Q_F</math></u>	
0	-	1	1	0	1	0	1	
1	1	1	1	1	0	1	0	11110
2	1	1	1	1	1	0	1	
3	1	1	1	1	1	1	0	

- 2] The initial content of 4 bits serial in parallel out shift register is shown in figure is 0110. After four clock pulses. Content of shift register is 1101.

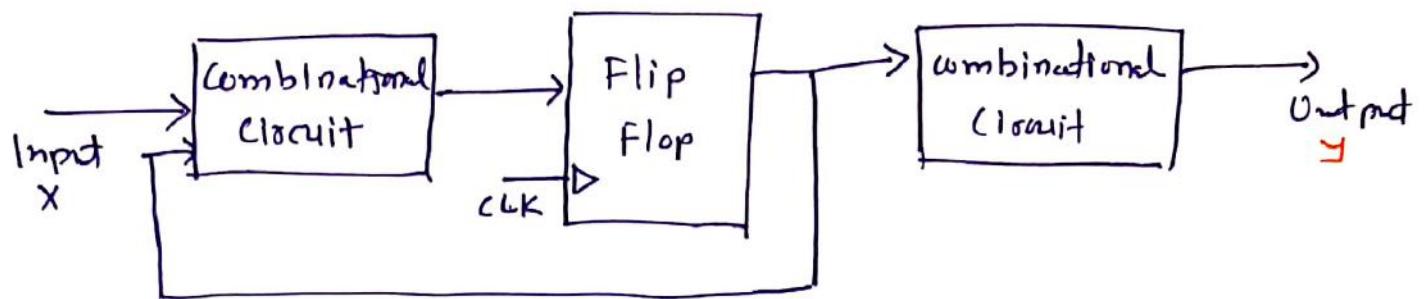
- (A) 0101     (B) 1101    (C) 1110    (D) 1011



CLK	In	$D_0$	$D_1$	$D_2$	<u><math>D_3</math></u>	
0	-	0	1	1	0	
1	1	1	0	1	1	
2	0	0	1	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	1101

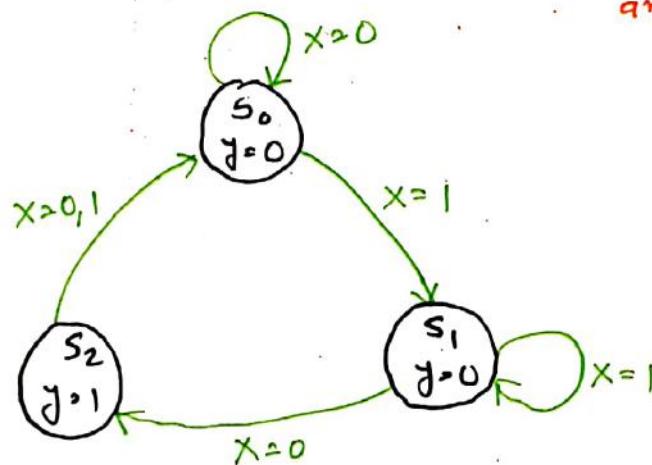
# Moore State Machine and Example on Moore State Machine

190

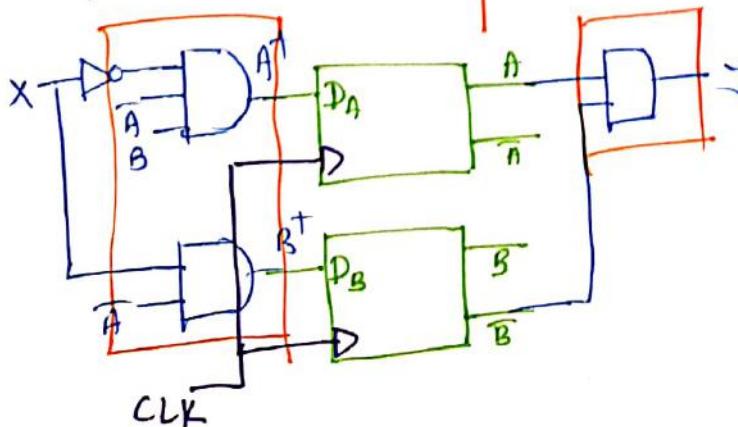


- Output  $y$  Only depends on present state of machine.
- Next state of moore machine depends on present state and Input  $x$ .

$$\begin{aligned} S_0 &= 00 \\ S_1 &= 01 \\ S_2 &= 10 \end{aligned}$$



Present state	Input	A	B	Next state output		
				$A^+(D_A)$	$B^+(D_B)$	$y$
0 0	0	0	0	0	0	0
0 0	1	0	0	0	1	0
0 1	0	1	0	1	0	0
0 1	1	1	0	0	1	0
1 0	0	0	0	0	0	1
1 0	1	0	0	0	0	1
1 1	0	1	0	1	1	1
1 1	1	1	1	1	1	1

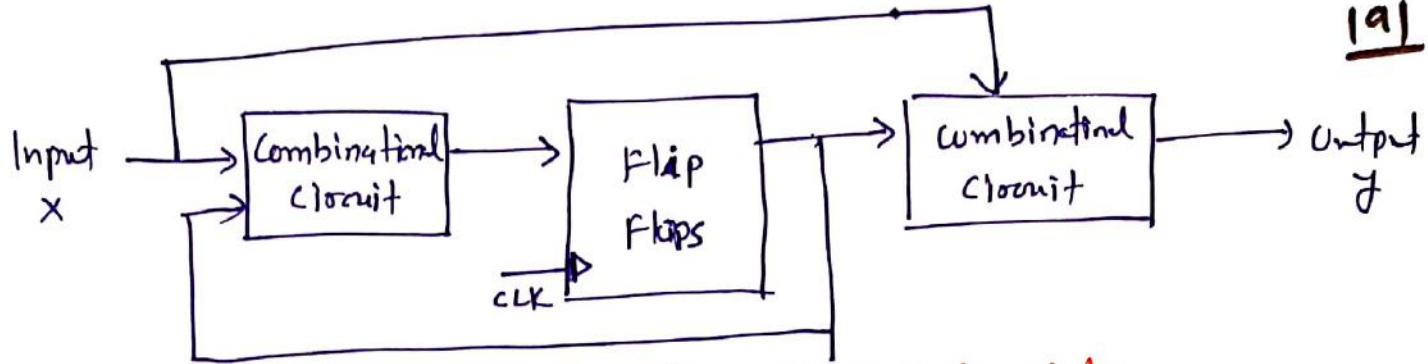


$$\begin{array}{c}
 \begin{array}{ccccc}
 A^+ & & & & \\
 AB & \times & 0 & 1 & \\
 \hline
 00 & 0 & 0 & 0 & \\
 01 & 0 & 0 & 1 & \textcircled{1} \\
 11 & - & - & - & \\
 10 & 0 & 0 & 0 &
 \end{array} & 
 \begin{array}{ccccc}
 B^+ & & & & \\
 AB & \times & 0 & 1 & \\
 \hline
 00 & 0 & 0 & 1 & \boxed{1} \\
 01 & 0 & 1 & 0 & \\
 11 & - & - & - & \\
 10 & 0 & 0 & 0 &
 \end{array} \\
 A^+ = \bar{A}B\bar{X} & B^+ = \bar{A}X
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{ccccc}
 Y & & & & \\
 AB & \times & 0 & 1 & \\
 \hline
 00 & 0 & 0 & 0 & \\
 01 & 0 & 0 & 0 & \\
 11 & - & - & - & \\
 10 & 1 & 1 & 1 & \boxed{1}
 \end{array} \\
 Y = \bar{A}\bar{B}
 \end{array}$$

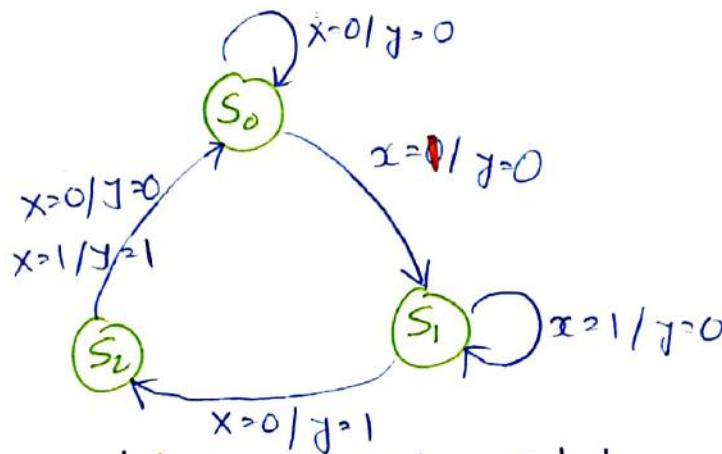
# Mealy State Machine and Example on mealy State machine

19



- Output  $y$  depends on Input and present state
- Next state depends on present state and Input  $x$ .

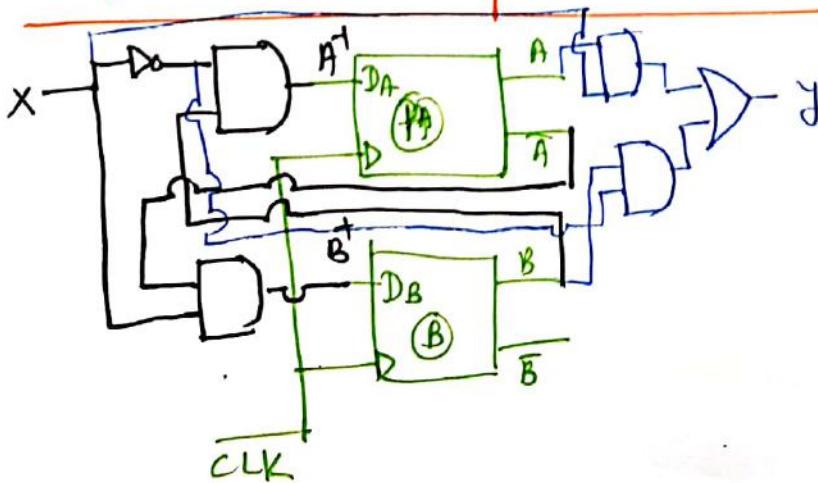
$$\begin{aligned} S_0 &= 00 \\ S_1 &= 01 \\ S_2 &= 10 \end{aligned}$$



Present State	Input	Next state			Output $y$
		$A^+(D_A)$	$B^+(D_B)$		
A	B	x			
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	1
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	0	0	1
1	1	0	X	X	X
1	1	1	X	X	X

$A^+$	$x$	$B^+$
AB	0 1	0 1
00	0 0	0 0
01	1 0	0 1
11	X X	X X
10	0 0	0 0

$$A^+ = \bar{X}B$$

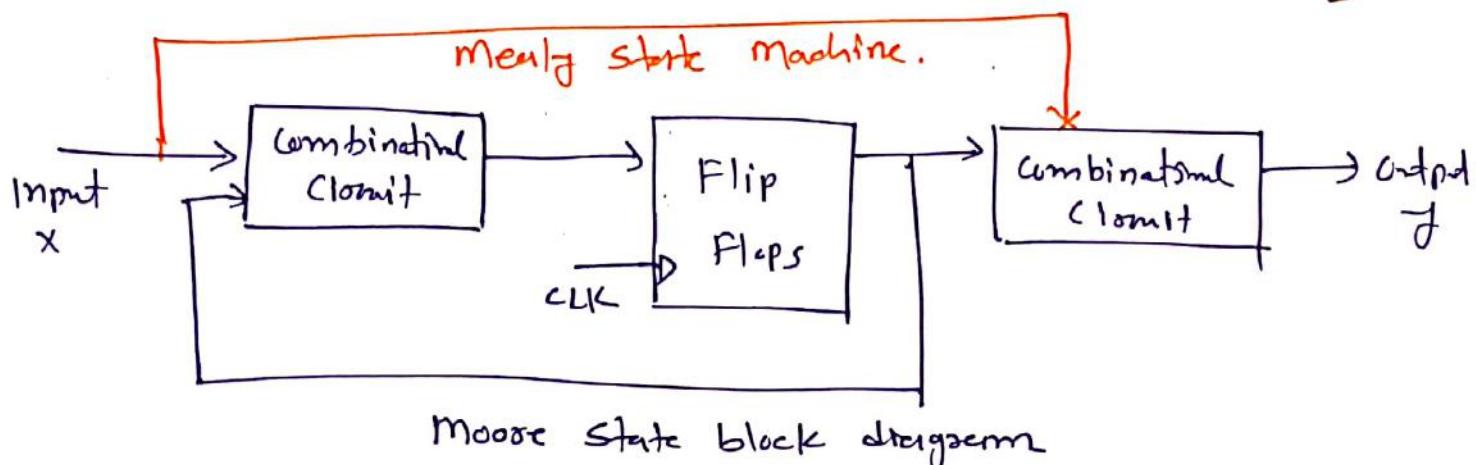
$$B^+ = X\bar{A}$$


$A^+$	$x$	$B^+$
AB	0 1	0 1
00	0 0	0 0
01	1 0	0 1
11	X X	X X
10	0 1	0 1

$$y = B\bar{X} + XA$$

# Comparison of Moore and Mealy State Machine.

192



- Output of Mealy State Machine depends on Input  $x$  + present state. While in Moore State Machine output only depends on present state.
- Moore State machine is faster than Mealy State machine.  
↳ Output of Moore State machine changes with clock while with Mealy State output will change with respect to Input.
- Mealy State machine are faster compared to Moore State machine.

How to choose between Mealy and Moore State Machine.

Q - Does we need Synchronous or Asynchronous machines (Moore) (Mealy).

Q - Speed?

Q - Are inputs and present state is readily available?