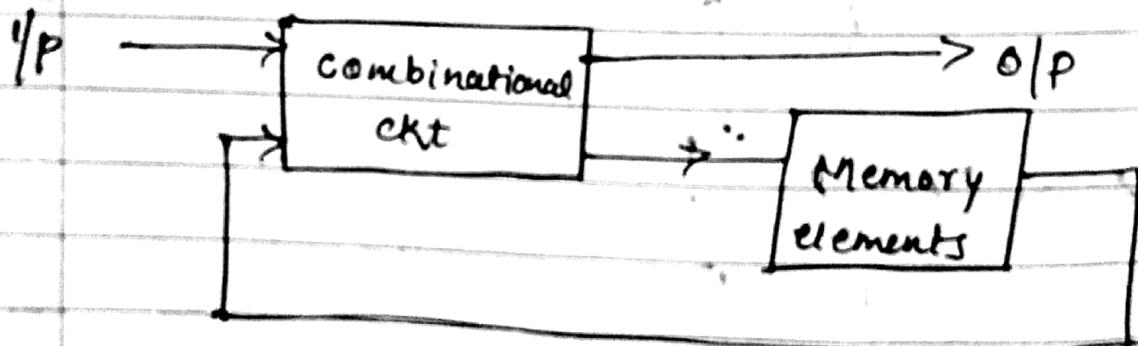


SEQUENTIAL CIRCUITS

sequential CKTS -

- Sequential ckt is a combination of a combinational ckt and a memory elements connected in feedback path.
- The memory elements are devices capable of storing binary information within them.
- The binary information stored in the memory elements at any given time defines the state of the sequential ckt.

Block diagram -



Types of sequential ckt -

- Synchronous sequential ckt is a system whose behavior can be defined from the knowledge of all signals at discrete instants of time.

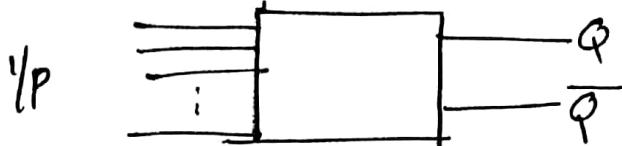
Asynchronous  $\rightarrow$  whose O/p depends on the sequence in which the i/p changes. it can

5 Synchronous  $\rightarrow$  whose O/p behaviour depends on the i/p at discrete time is called

10 Synchronous sequential ckt's that use clock pulses in the i/p of memory elements are called clocked sequential ckt's.

### Flip-flop -

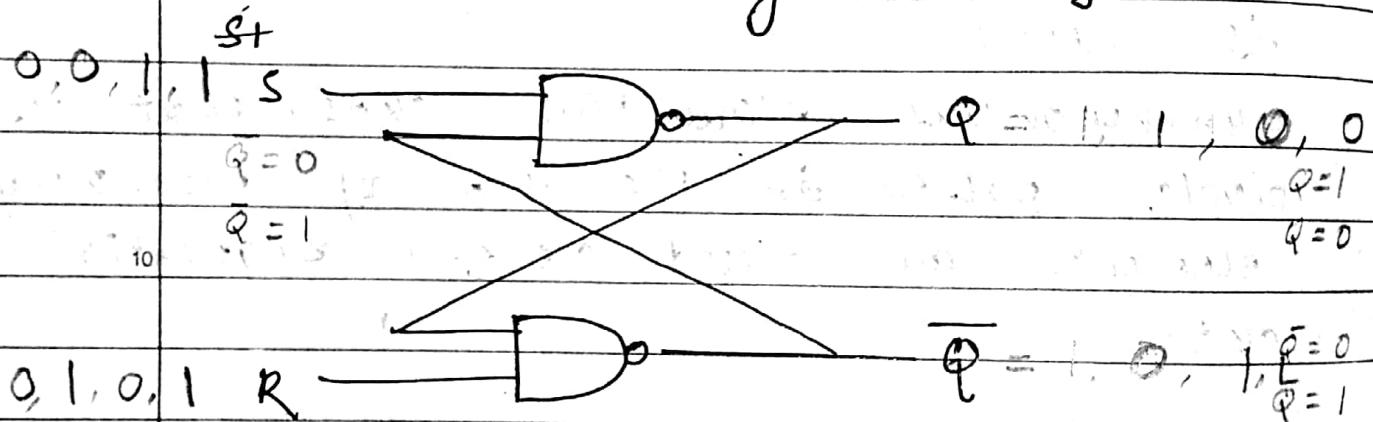
- The memory elements used in clocked sequential ckt's are called flip-flops.
- These ckt's are binary cells capable of storing one bit of information.
- F.F has 2 O/p labeled as Q and  $\bar{Q}$  that are inverse of each other.
- A ff has 2 stable state hence its known as bistable multivibrator.
- A logic gate by itself, has no storage capability, but several such logic gates can be connected together in ways that permit information to be stored.



Latch -

A latch is the most basic type of ff ckt. it can be constructed using NAND or NOR gates.

S-R latch using NAND gate -



A      B      Y

0	0	1	0
0	1	1	1
1	0	1	1
1	1	0	1

S      R      Y      Q

0      0      Invalid ( $Q = \bar{Q} = 1$ )

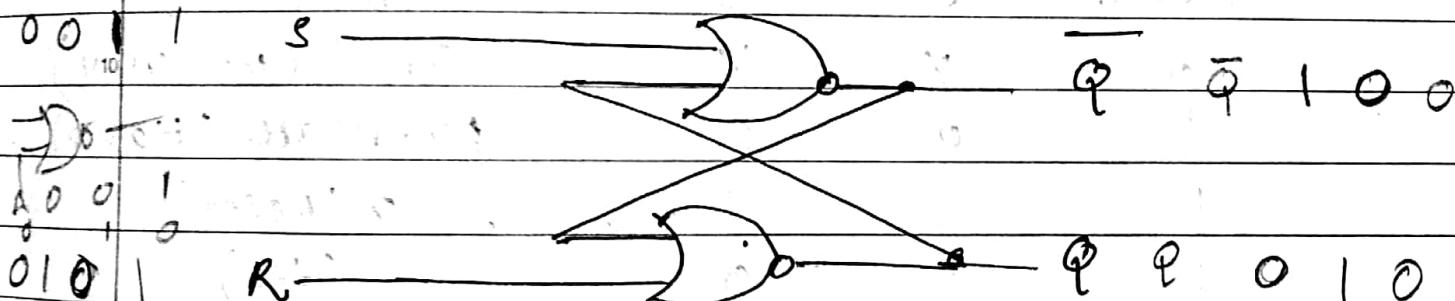
0      1      1

1      0      0

1      1      previous state

- In S-R latch if both gates are enabled, o/p will remain same as previous state if previous state is valid.
- If both gates are enabled then o/p will presenting invalid state.

S-R latch using NOR-



	A	B	Y	S	R	Q
0 0 1	0	0	1	0	0	previous
0 1 0	0	1	0	0	1	0
1 0 0	1	0	0	1	0	1
1 1 0	1	1	0	1	1	Invalid state.

- S-R latch is mostly used as basic memory element in sequential ckt.
- it can also be used as bounce elimination ckt's in keyboard or switches.

S-R ff -

0011

 01  
clock

R

0101

clock

1100

1101

 Q<sub>n</sub>

Q = 1100

Q = 0010

 Q<sub>n</sub> → present state

 Q<sub>n+1</sub> → Next state

Clock	S	R	Q <sub>n+1</sub>
0	X	X	Q <sub>n</sub> (memory)
1	0	0	per. state (Hold)
1	0	1	0 (Reset)
1	1	0	1 (Set)
1	1	1	Invalid (unused)

S	R	Q <sub>n+1</sub>	Q <sub>n</sub>	Logic symbol						
0	0	Q <sub>n+1</sub>	Q <sub>n</sub>	<table border="1"> <tr> <td>S</td> <td>Q</td> </tr> <tr> <td>clock</td> <td>Q</td> </tr> <tr> <td>R</td> <td>Q</td> </tr> </table>	S	Q	clock	Q	R	Q
S	Q									
clock	Q									
R	Q									
0	1	0	Q <sub>n</sub>							
1	0	1								
1	1	Invalid.								

characteristic table -

S	R	Q <sub>n</sub>	Q <sub>n+1</sub>
0	0	0	0 ] HOLD
0	0	1	1
0	1	0	0

0	1	1	0	Reset
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Invalid
1	1	1	1	Invalid

Characteristic table.—

S	R	$Q_n$	$Q_{n+1}$	
0	0	0	0	HOLD $Q_n$
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	Set
1	0	1	1	
1	1	0	1	Invalid
1	1	1	1	Invalid

$$Q_{n+1} = \Sigma m(1, 4, 5) + \Sigma d(6, 7)$$

S.	$RQ_n$			
	00	01	11	10
0	•	1	1	3
1	4	15	X7	X6

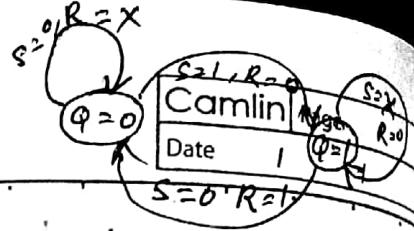
Characteristic equation —

$$Q_{n+1} = S + \bar{R} Q_n \quad (\because S, R \neq 1)$$

This equation specifies the value of the next state as a function of the present state & inputs.

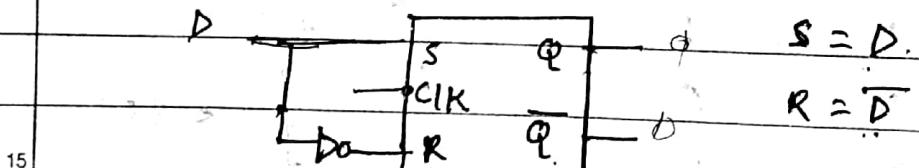
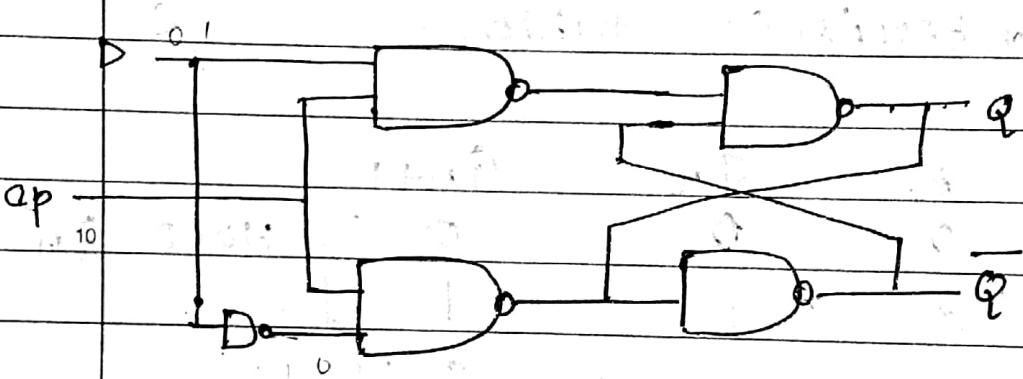
Excitation Table

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



$\rightarrow$  flip flop -

The  $\rightarrow$  flip flop shown is a modification of the clocked SR flip flop. The  $\rightarrow$  i/p goes directly to the  $S$  i/p and its complement and its complement is applied to the  $R$  i/p.



$CLK$	$D$	$Q_{n+1}$	$Q_n$	(M)
0	0	X	0	(R)
1	0	0	0	
1	1	1	1	(S)

20

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

25

Characteristic Table  $\rightarrow$

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

char. equation  $\rightarrow Q_{n+1} = D\bar{Q_n} + \bar{D}Q_n$   
 $= D(\bar{Q_n} + Q_n)$

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

excitation table -

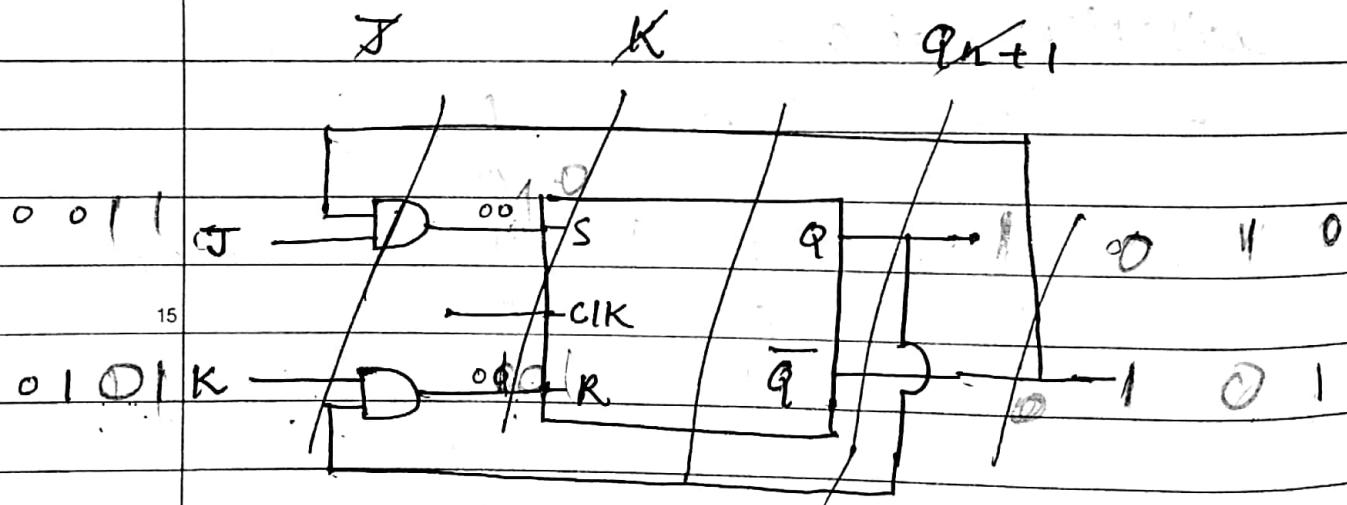
$Q_n$	$Q_{n+1}$	D
0	0	0
0	1	1
1	0	0
1	1	1

State Diagram -

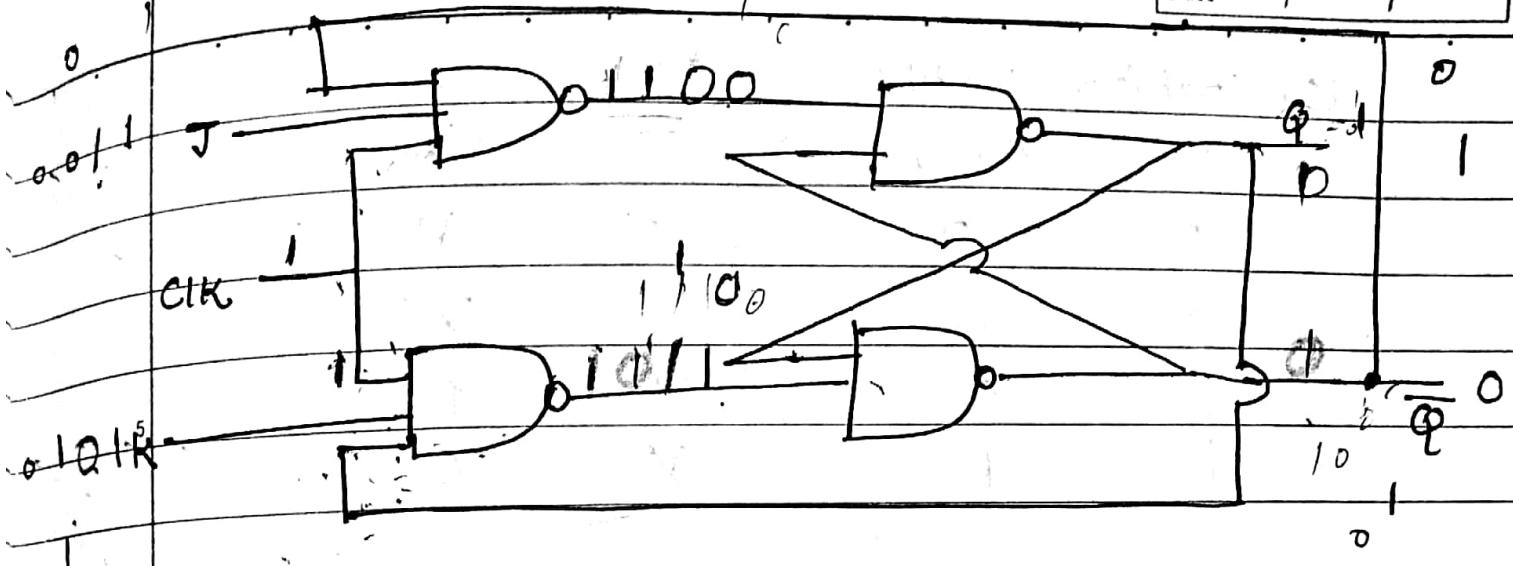
## T-ff $\leftrightarrow$

### JK flip flop -

- A JK flip flop is a refinement of the RS flip-flop in which the undeterminate state of the RS type is defined in the JK type.
- SR ff presents invalid state when both inputs are at logic 1. To overcome this problem JK ff is used.



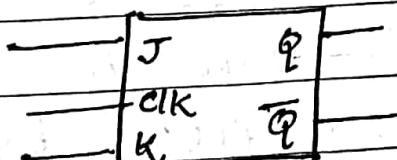
	CLK	J	K	$Q_{n+1}$
0	0	X	X	$Q_n \rightarrow M$
1	0	0	0	$Q_n \rightarrow H$
1	0	0	1	0 $\rightarrow R$
1	1	0	0	1 $\rightarrow S$
1	1	1	1	$Q_n \rightarrow \text{Toggle State}$
25	0	0	0	$Q_n$
	1	1	1	



## Characteristic Table -

	$Q_n$	J	K		$Q_{n+1}$
10	0	0	0		0
	0	0	1		0
	0	1	0		1
	0	1	1		1
	1	0	0		1
15	1	0	1		0
	1	1	0		1
	1	1	1		0
	1	1	1		1

## Logic Symbol -



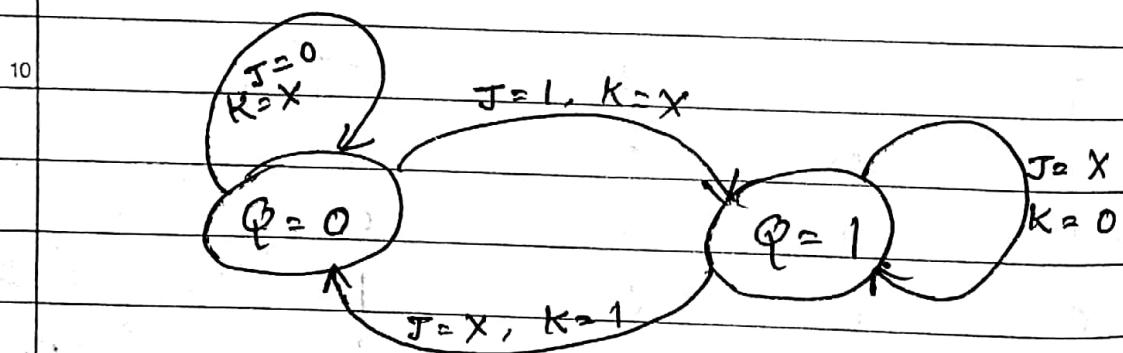
$$q_{n+1} = \text{em}(1, 4, 5, 6)$$

$$Q_{n+1} = J \overline{Q_n} + K Q_n$$

## 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

## State diagram -

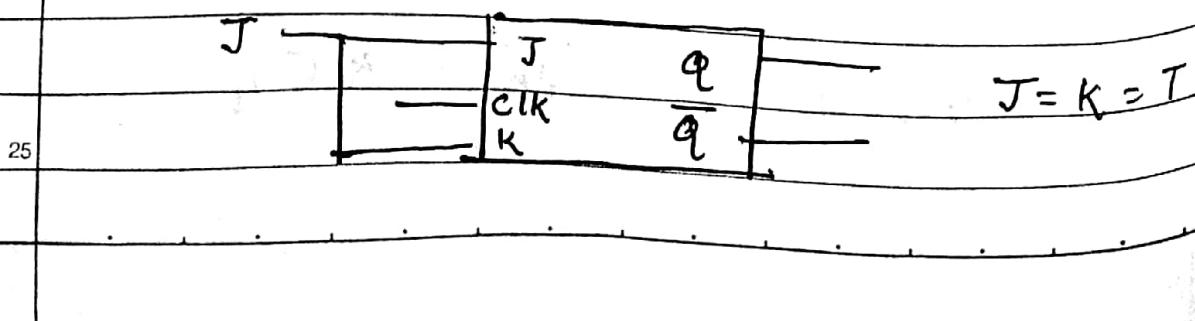


## T- flip flop -

The T flip flop is a single i/p version of the JK flip flop.

It acts as a toggle switch.

it can be realised from JK flip flop by making  $J = K = T$   
ie. tied together.



$Q_{n+1}$

0 X

$Q_n \rightarrow \text{Memory}$

1 0

$Q_n \rightarrow \text{Hold}$

1 1

$\overline{Q_n} \rightarrow \text{Toggle}$

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

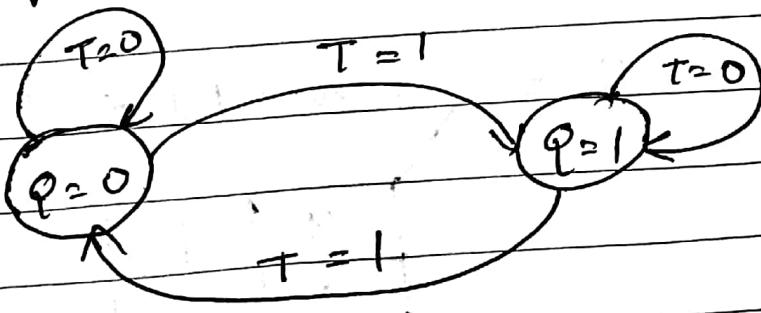
$$= T \oplus Q_n$$

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

Excitation table -

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

State diagram -



conversion from one type of flip flop to another type -

JK ff to D ff  $\rightarrow$

excitation table of combine characteristic table of D ff with

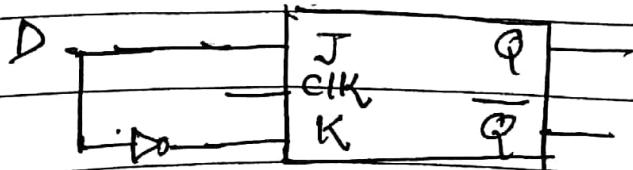
D	$Q_n$	$Q_{n+1}$	J	K
0	0	0	0	X
0	1	0	X	1
1	0	1	1	X
1	1	1	X	0

$$J = \sum m(2) + \sum d(1, 3)$$

D	$Q_n$	0	1	J = D
0			X	
1		1	X	

$$K = \sum m(1) + \sum d(0, 2)$$

	X	1	1	K = $\overline{D}$
20	X			



JK TO T.

T TO SR

JK TO SR

T TO D

SR TO JK

SR TO D

SR TO T

D TO SR

D TO JK.

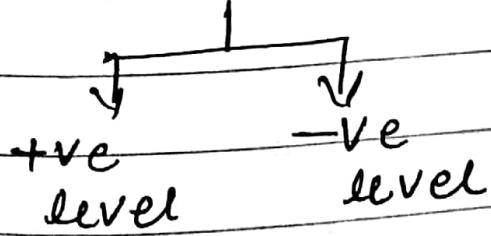
D TO T

synchronous  $\rightarrow$  T TO JK

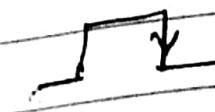
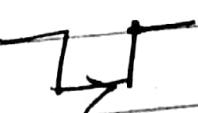
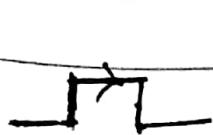
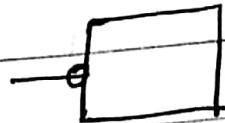
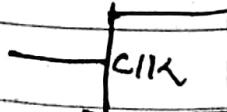
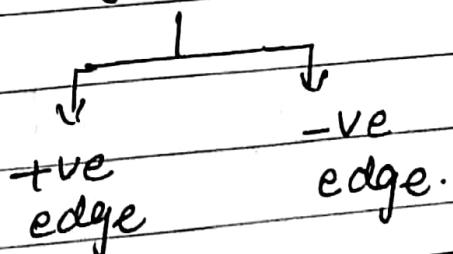
This momentary change is called a trigger and the transition it causes is said to trigger the ff.

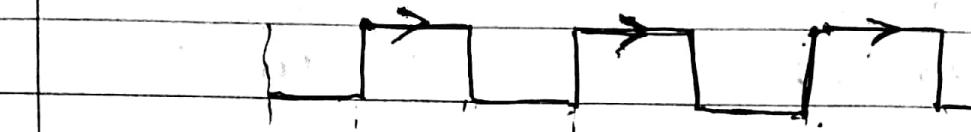
Triggering of flip flops - The state of a flip flop is switched by a momentary change in the sig. signal. [CLOCK]

↓  
level trigger



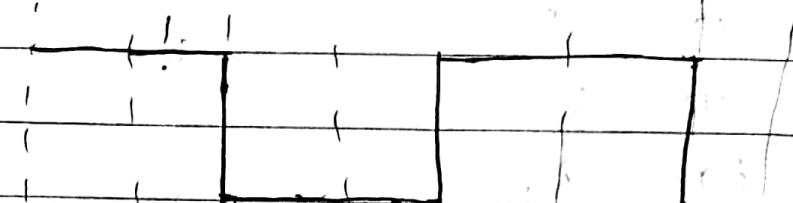
↓  
edge trigger.





5

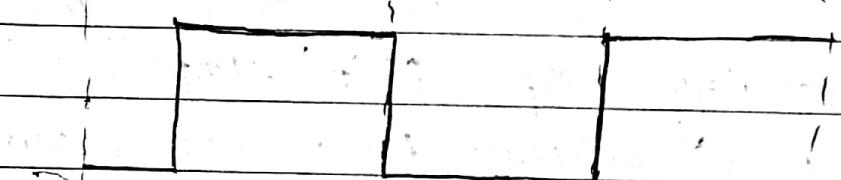
S



R

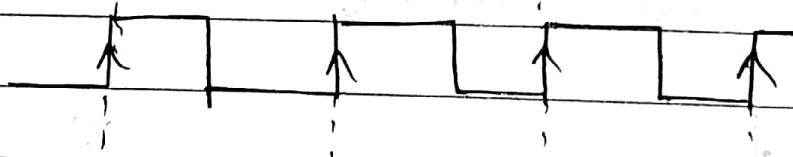
O

10

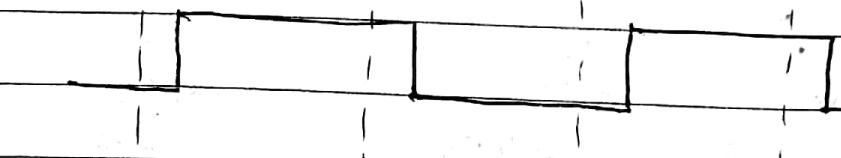


15

S

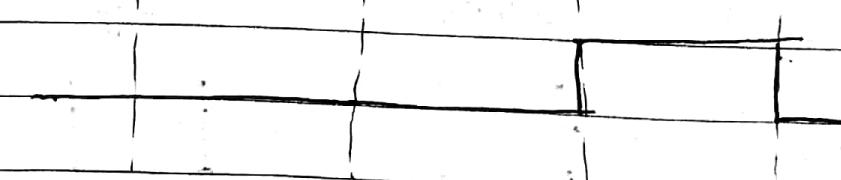


R



20

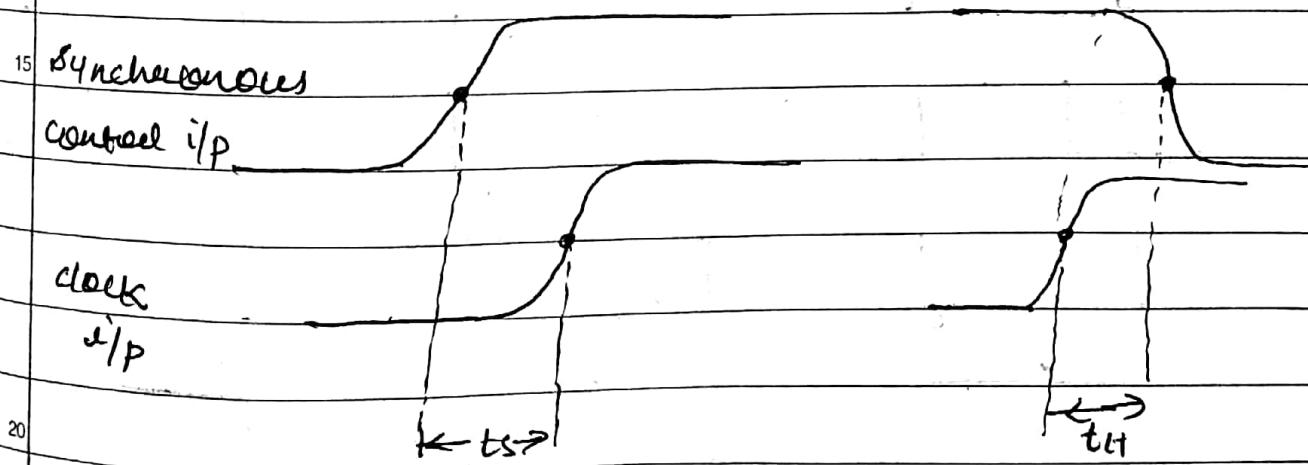
Q →



25

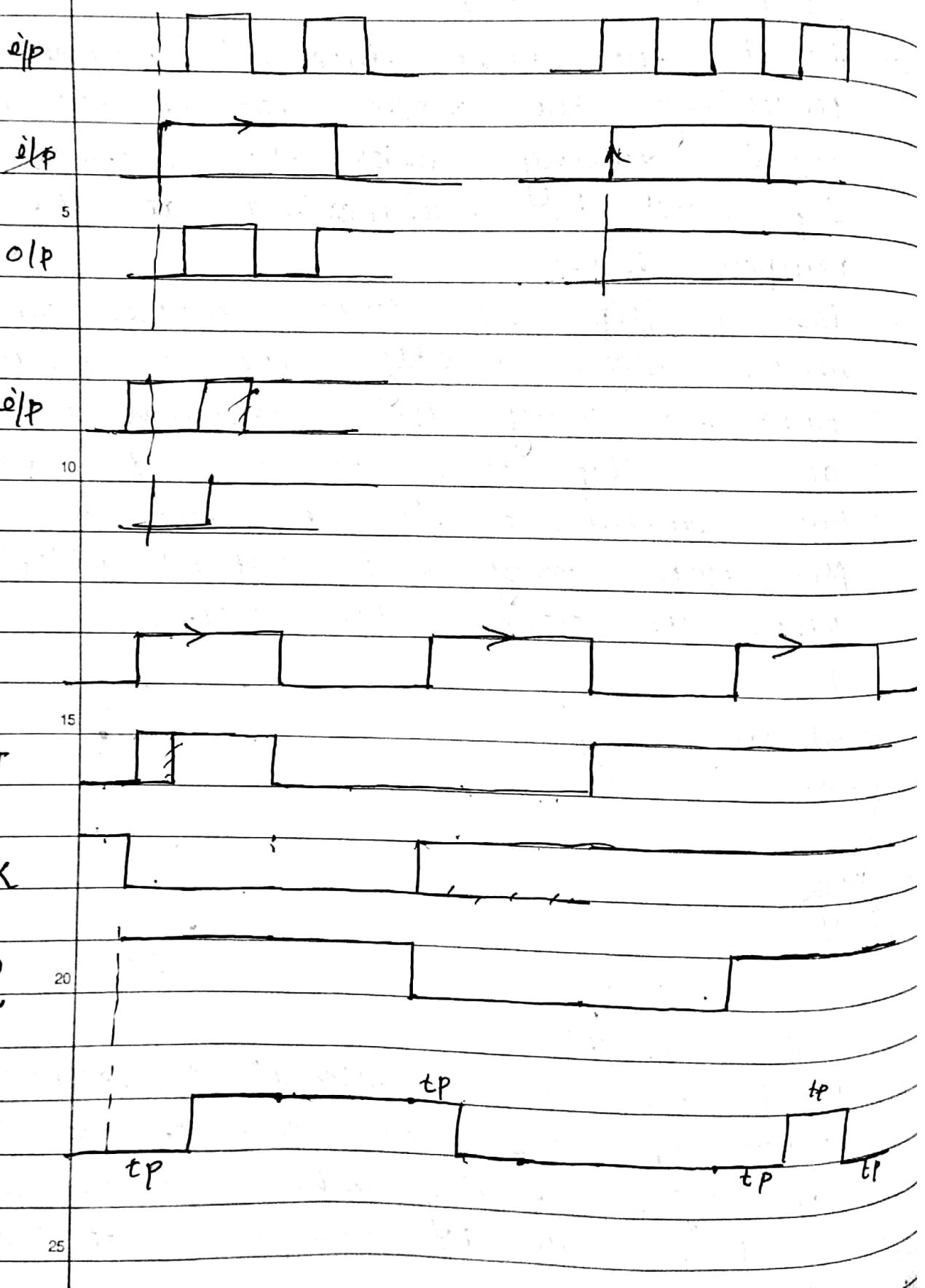
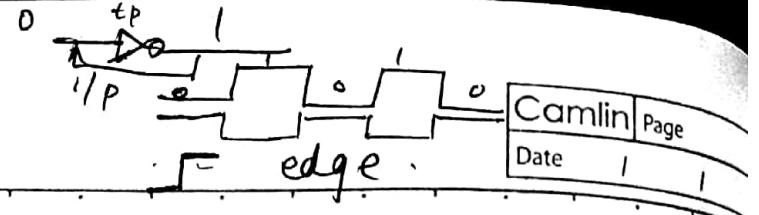
### Setup time - ( $t_s$ )

- it is the time interval immediately preceding the active edge of the clock signal during which the control i/p must be maintained at the proper level.
- for a flip flop that triggers on the rising edge. usually, the ic manufacturers specify the minimum allowable setup time,  $t_s(\text{min})$ . if this time requirement is not met, the flip flop may not respond reliably when the clock edge occurs.



### Hold time ( $t_H$ )

- It is the time interval immediately following the active edge of the clock signal during which the synchronous control i/p must be maintained at the proper level.

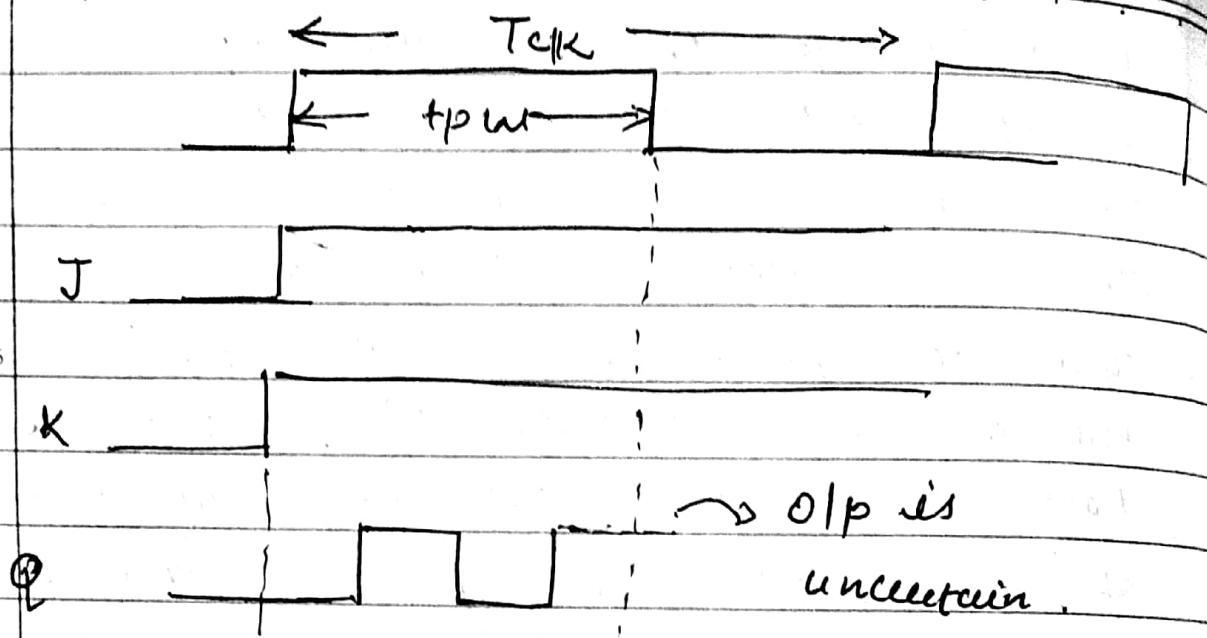


Disadvantage of JK flip flop -

if inputs of JK flip flop are  $J=K=1$  and  $Q=0$  and a clock pulse as shown, after a time interval  $t_p$  equal to propagation delay of NAND gates, the o/p will change  $Q=1$ . Now we have  $J=1$ ,  $K=1$  and  $Q=1$ . if duration of clock pulse ( $T$ ) is greater than propagation delay  $t_p$ , after another time interval of  $t_p$ , the o/p will change back to  $Q=0$ . hence, the o/p will oscillate back & forth b/w 0 & 1. The o/p is uncertain at the end of clock pulse if flip flop is level triggered.

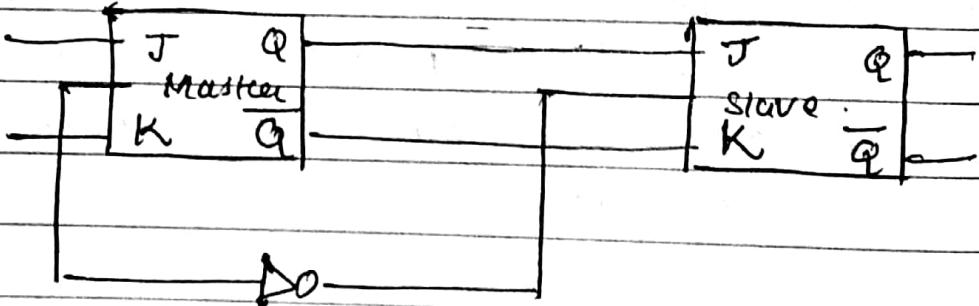
- Race around condition will occur when  $J$  &  $K$  inputs are at logic 1 &  $t_{pdff} < t_{pw}$
- During race around o/p will changes many times in a single clock pulse when inputs are constant.

$$t_{pw} < t_{pdff} \leq T_{clk}$$



10 Master slave flip flop -

A



11 master slave

→ In this ckt master is applied with  $\phi$ p clock & slave is applied with inverted clock. due to this master & slave will not change at a time.

→ In master slave there is no race around conditn.

25

## Registers -

Registers are used to store group of bits.

To store n-bits, n-FF are used in register.

Depending on S/P and O/P register the way in which data are loaded & retrieved, registers are classified in 4 types -

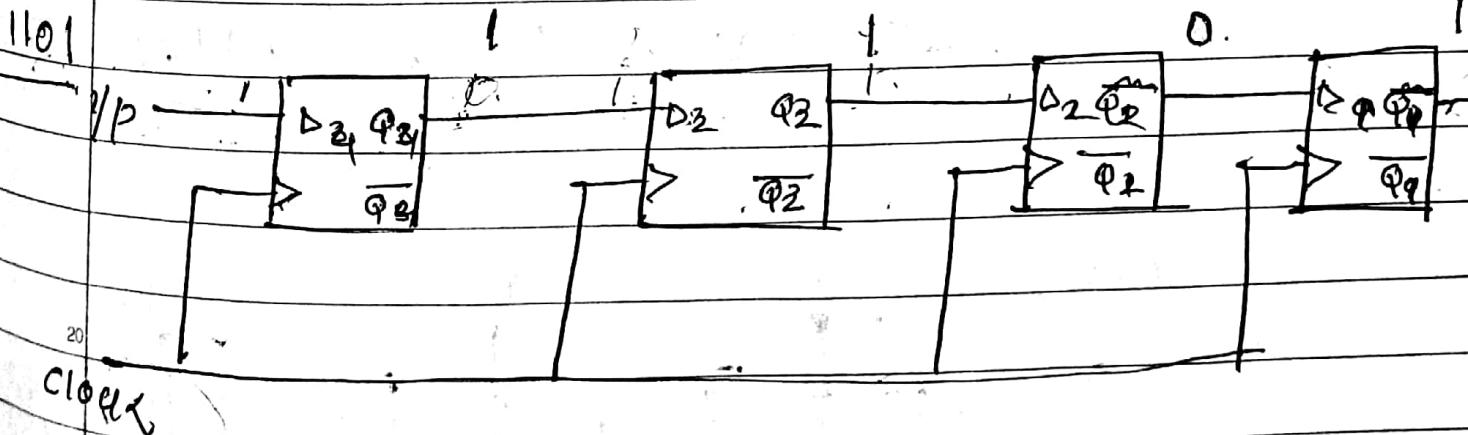
SISO

SIPD

PISO

PIPO

SISO



The O/P of each FF is connected to the S/P of the next flip flop at its right. Each clock pulse

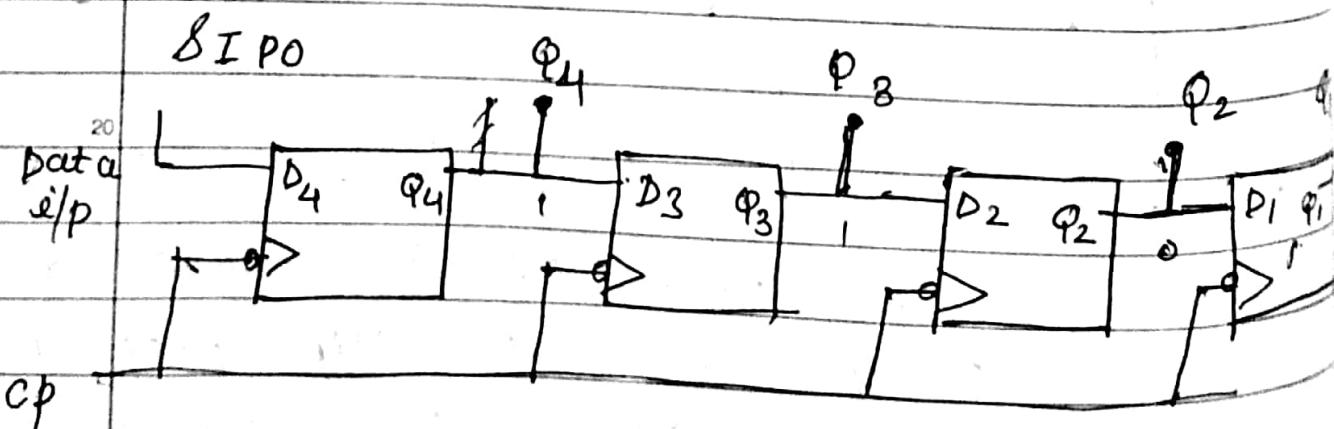
shifts the contents of the register one bit position to the right.

Clock	Data	$Q_4$	$Q_3$	$Q_2$	$Q_1$
-------	------	-------	-------	-------	-------

0	1101	0	0	0	0
1	110	1	0	0	0
n	2	11	0	1	0
3	1	1	0	1	0
4	.	1	1	0	1

↑  
Serial loading of info.

CLOCK	$Q_4$	$Q_3$	$Q_2$	$Q_1$	O/P
1	1	1	0	1	
5	X	1	1	01	
n-1	X	X	1	101	01
6	X	X	1	101	01
15	X	X	X	1101	101
7	X	X	X	1101	101
8	X	X	X	X	1101



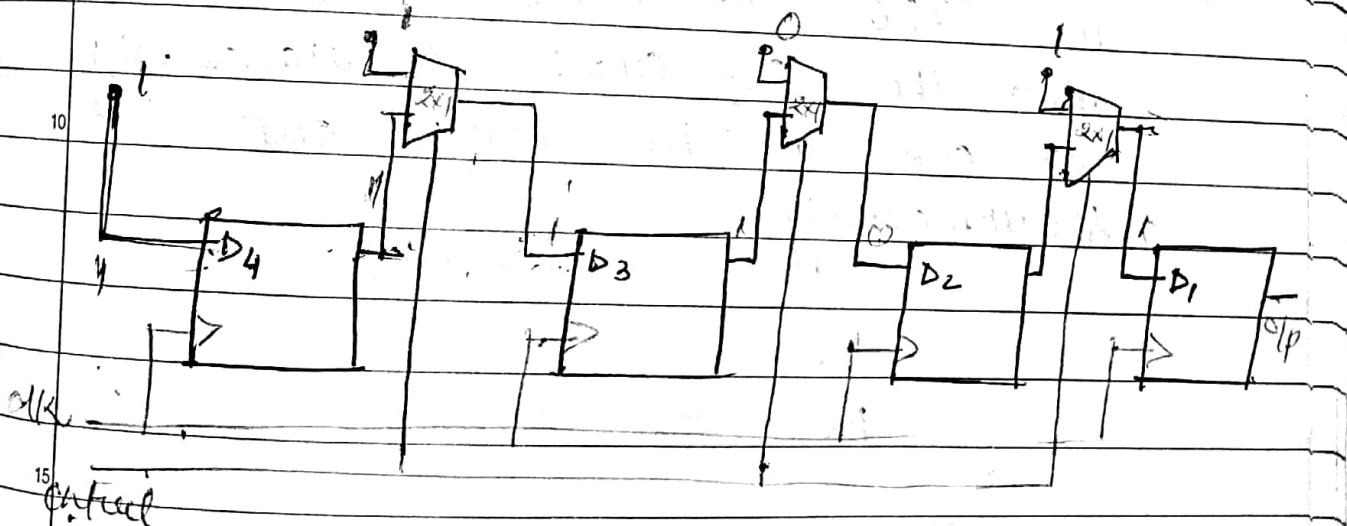
In SIPO data is loaded serially one bit at a time, but the

CamScanner  
Data

data stored can be read simultaneously  
The n bit input would becomes installed  
in the register after the nth clock pulse.

Serial i/p  $\rightarrow$  n,  
parallel o/p  $\rightarrow$  0

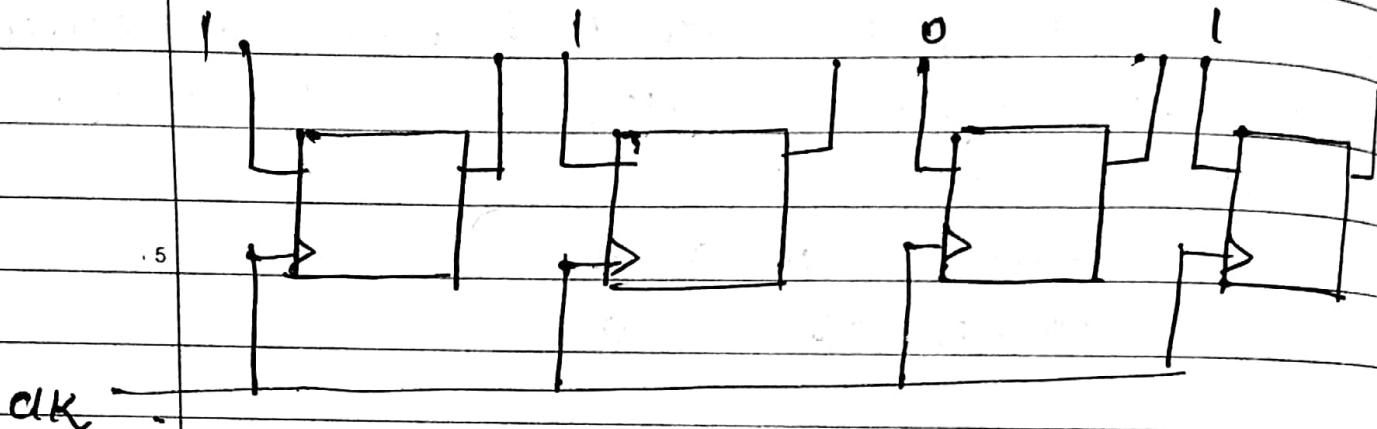
PISO  $\rightarrow$



Since the data can be loaded into the flip flops simultaneously and can be read from the register one bit at a time by clock pulse, this shift register is a parallel to serial converter.

PI  $\rightarrow$  1 clock  
Serial out  $\rightarrow$  n-1 clock

parallel in parallel out -



clk -

In PIPO, the data can be loaded into the flip-flops simultaneously and can also be taken out simultaneously.

PI  $\rightarrow$  1.

PO  $\rightarrow$  0

counters -

Counters are digital circuit which is used to count the clock pulses and to generate sequence of states, where state of counter is set of O/P of the flip flops.

In counter  $n$  ff  $\rightarrow$  states  $\rightarrow 2^n$  st

BCD counter  $\rightarrow$  4 ff  $\rightarrow 2^4 = 16 \rightarrow 0-9$  state

Depending upon the manner in which flip flops are triggered, counters can be classified into 2 major groups

### Asynchronous

if all clock i/p of each ff are synchronized by the external clock by single clock.

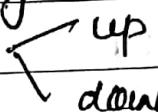
### Asynchronous

The 1st ff is clocked pulse and then each successive ff is clocked by the o/p of previous ff.

faster

slower

No decoding requires  
Ring, Johnson

Decoding requires  
Ripple 

### Modulus of the counter

Modulus of the counter is the no. of different states before the sequence repeats.

$$M \leq 2^N$$

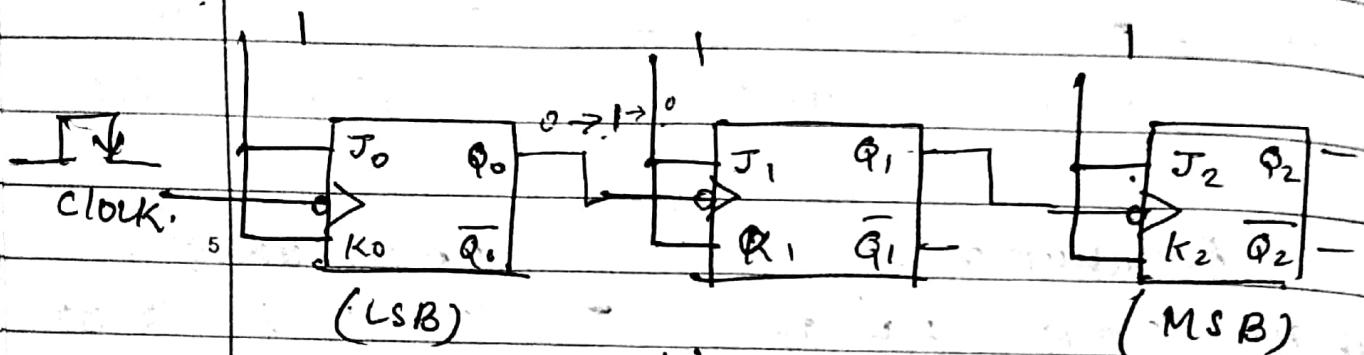
### Applications

frequency dividers  
timers

Range/distance measurement

frequency/pulse width measurement

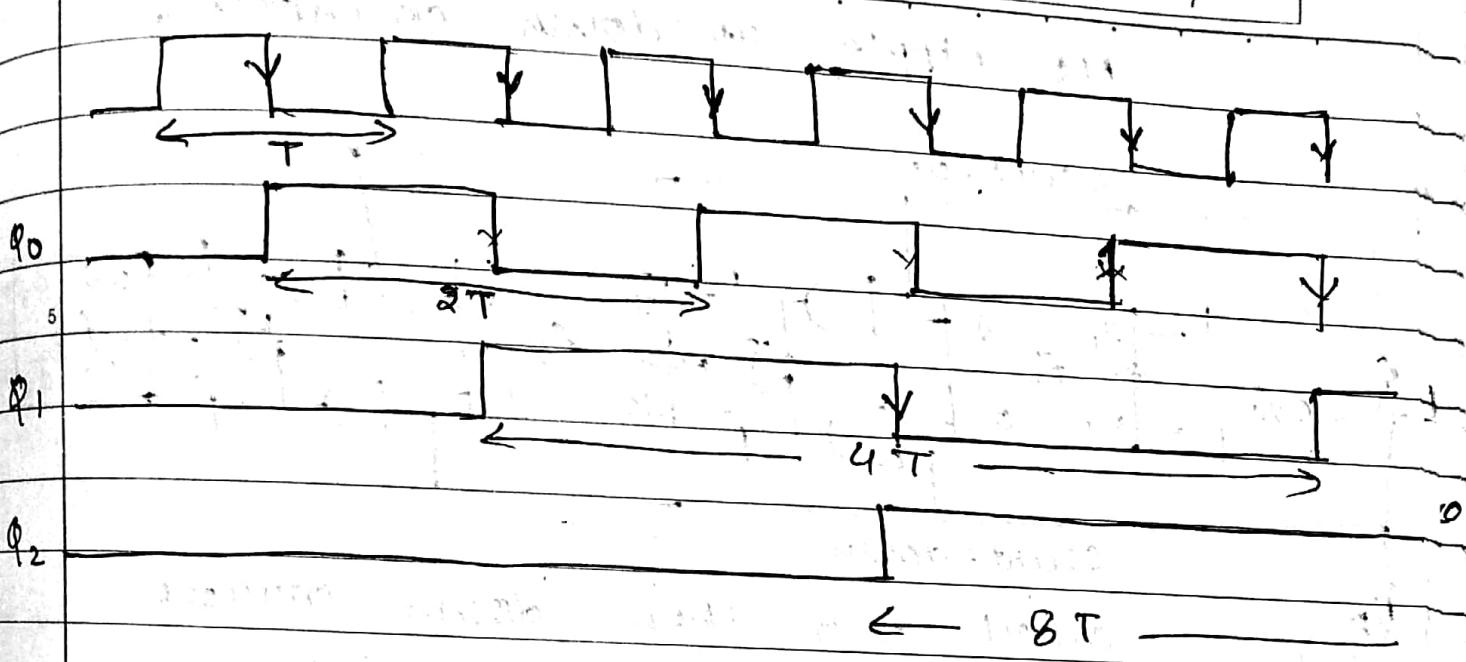
# Asynchronous Ripple counter.



3 bit Ripple counter

- 1) In this ff are operate in toggle mode
- 2) different ff are applied with diff. clock pulse.
- 3) ff @/p  $Q_0, Q_1$  and  $Q_2$  represents a 3-bit binary no. with  $Q_0$  as the LSB &  $Q_2$  as the MSB.
- 4)

CK	$Q_2$	$Q_1$	$Q_0$
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0



$$f_{max} = \frac{1}{8T}$$

$$\text{Q/p frequency} = \frac{1}{8T}$$

$f = \frac{1}{8T}$    
  $f$  = 1/p frequency  
 8 = Mod of counter.

Mod-8 ripple up counter.

-ve edge ]

$Q$  as clock ]

+ve edge ]

$\bar{Q}$  as clock ]

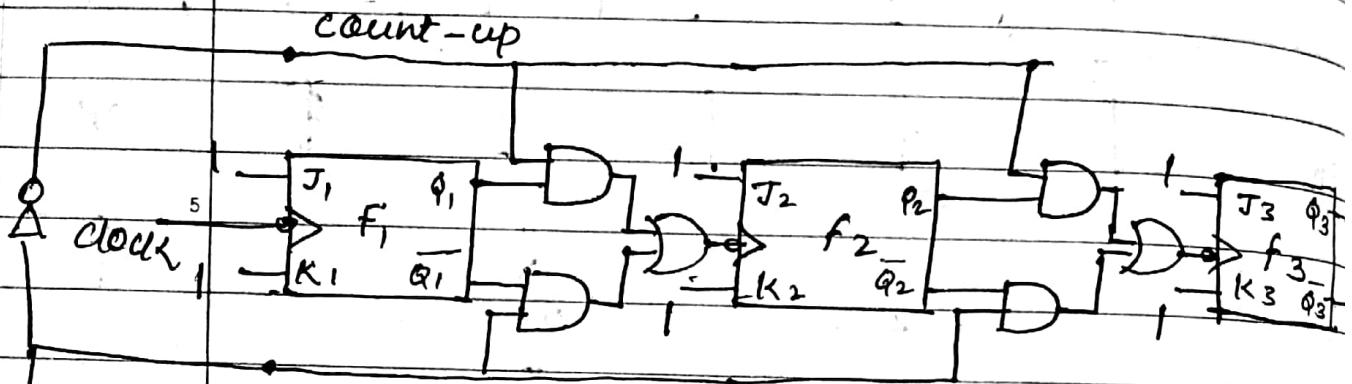
+ve edge ] down

$Q$  as clock ]

-ve edge ] down

$\bar{Q}$  as clock ]

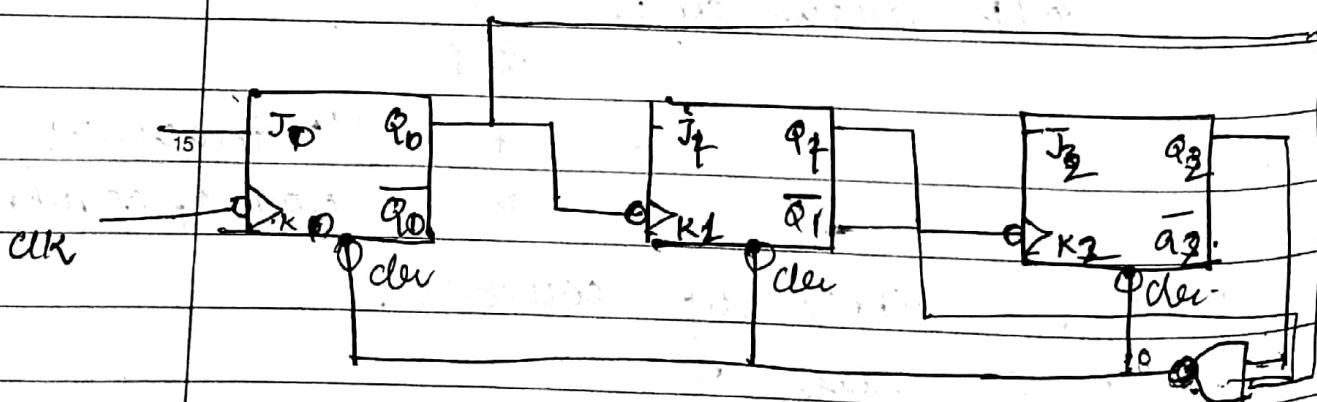
## Mod Ripple up down counter -



control → 0 then down counter

1 then up counter

## Mod 5 ripple up counter -



CLK      Q<sub>2</sub>      Q<sub>1</sub>      Q<sub>0</sub>

0      0      0      0

1      0      0      1

2      0      1      0

3      0      0      1

4      0      0      0

5      0      0      0

101

Q<sub>2</sub> Q<sub>1</sub> Q<sub>0</sub>

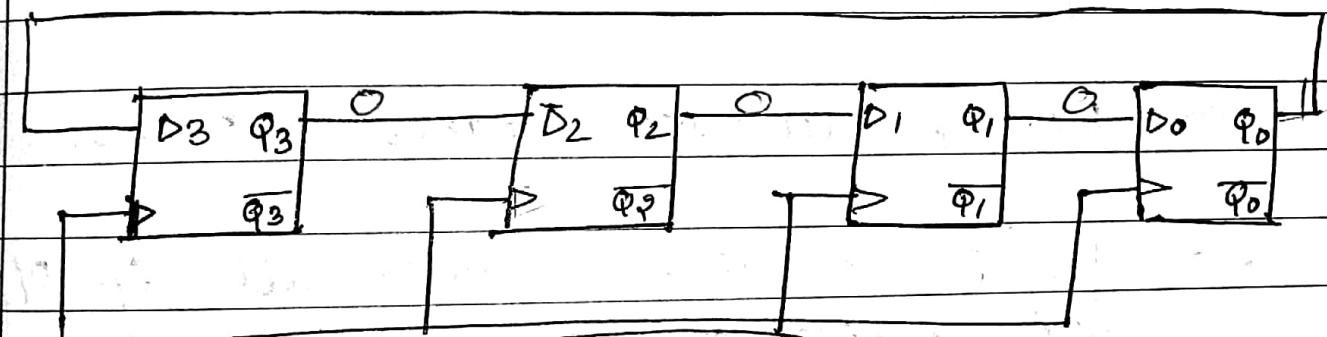
Q<sub>2</sub> Q<sub>1</sub> Q<sub>0</sub>

101

## Synchronous counters -

Ring counter -

it is constructed from DLSO Reg.  
with feedback  $Q_0$ .



CK |  $Q_3$     $Q_2$     $Q_1$     $Q_0$

0	0	0	0	1
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1
9	1	0	0	0

In Ring counter only 1 bit changes at logic 1. Other bits are at.

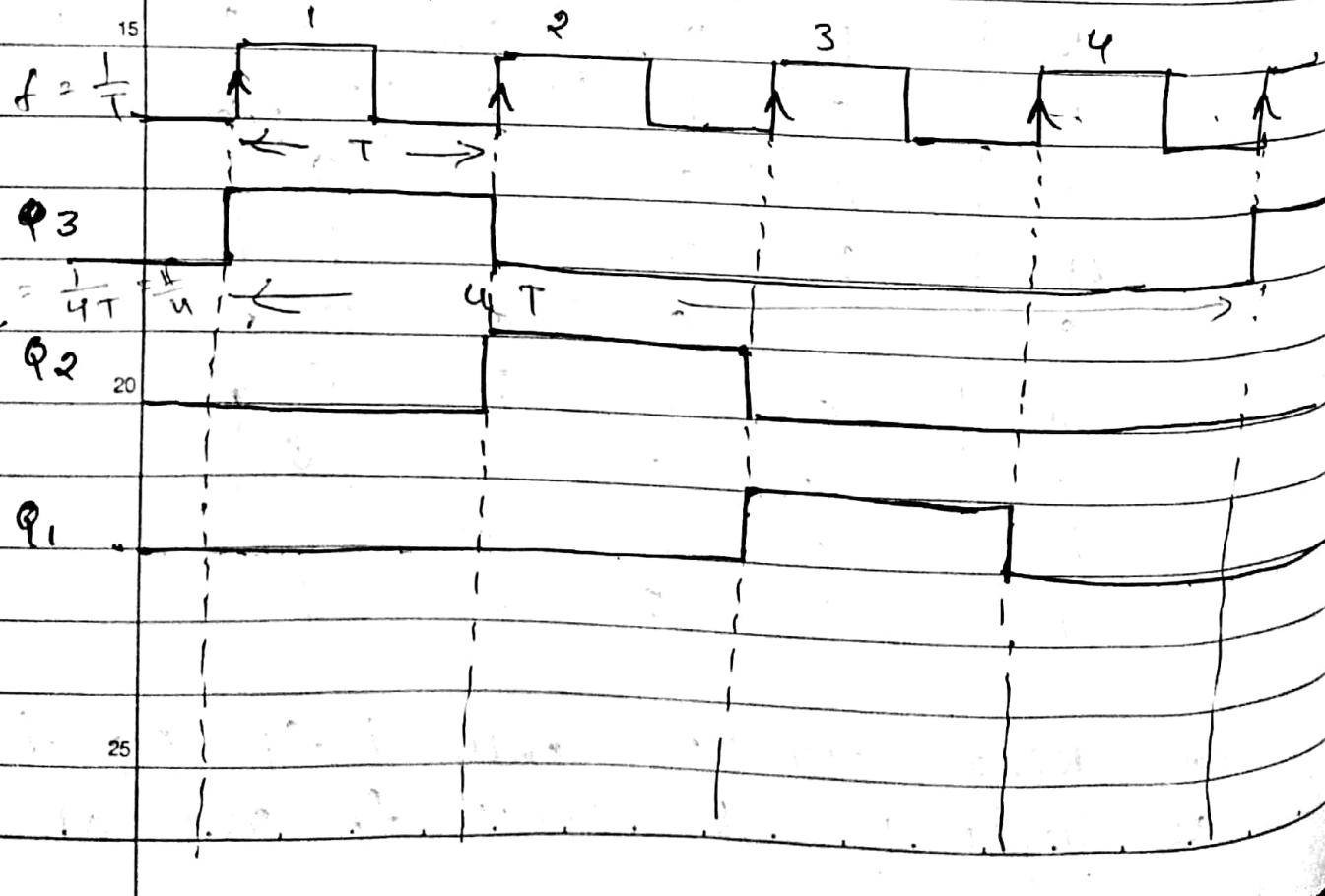
logic 0. When clock pulse is applied, logic 1 will rotate to other ff.

→ In Ring counter with n-ff no. of states used - n states.

→ No. of unused states -  $2^n - n$

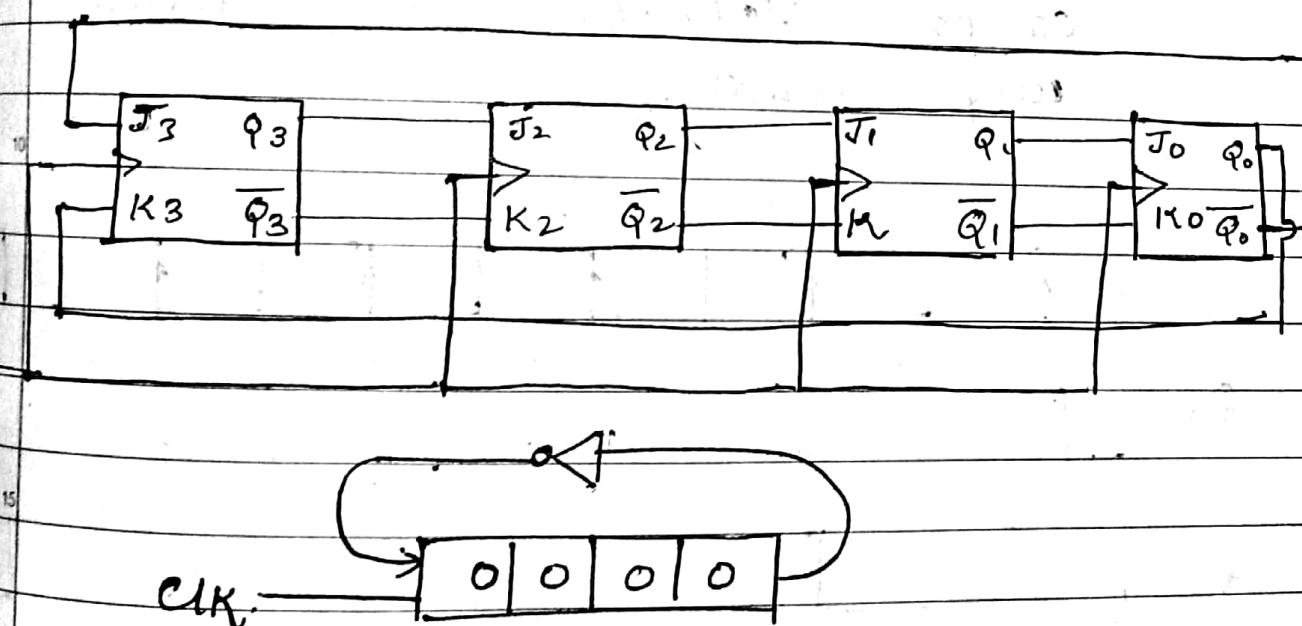
→ In Synchronous counter all ff o/p are available at a time hence no transition & decoding errors.

→ If up frequency  $f$ ; then o/p frequency  $f/n$ .



- All O/P freq. remain same but they have diff. phase shift.
- phase shift b/w successive waveform are  $360^\circ/n$ .
- Duty cycle of O/P waveform is,  $100/n$ .

### Johnson Ring counter -



AND gate

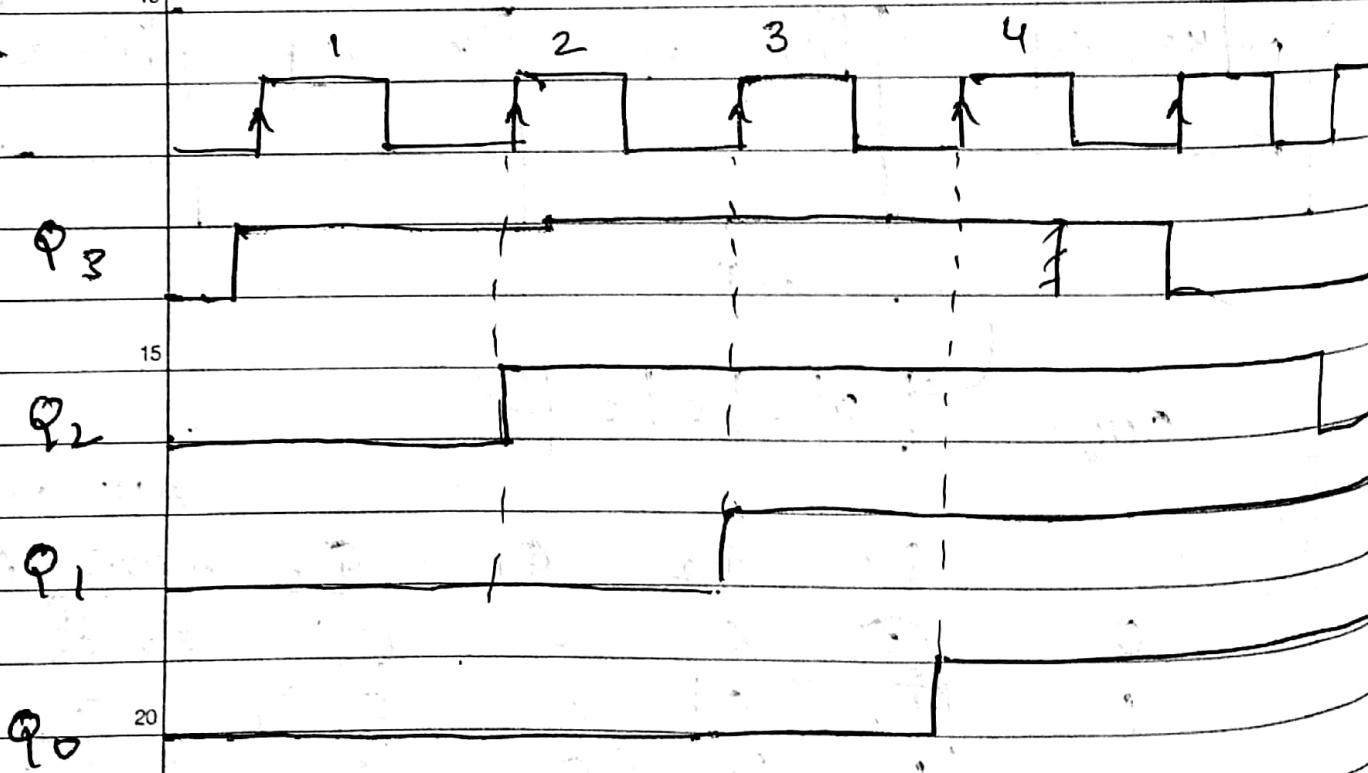
	$Q_3$	$Q_2$	$Q_1$	$Q_0$	Required for O/P
0	0	0	0	0	$\bar{Q}_3 \bar{Q}_0$
1	1	0	0	0	$Q_3 \bar{Q}_2$
2	1	1	0	0	$\bar{Q}_2 \bar{Q}_1$
3	1	1	1	0	$Q_1 \bar{Q}_0$
4	0	1	1	1	$f/8 \quad Q_3 Q_0$
5	1	0	1	1	$\bar{Q}_3 \bar{Q}_2$
6	0	1	0	1	$\bar{Q}_2 \bar{Q}_1$
7	0	0	1	1	$Q_1 \bar{Q}_0$
8	0	0	0	0	
9	1	0	0	0	

→ no. of used states  $\rightarrow 8 = \frac{2^n}{2}$

→ no. of unused states  $\rightarrow 2^n - 8$

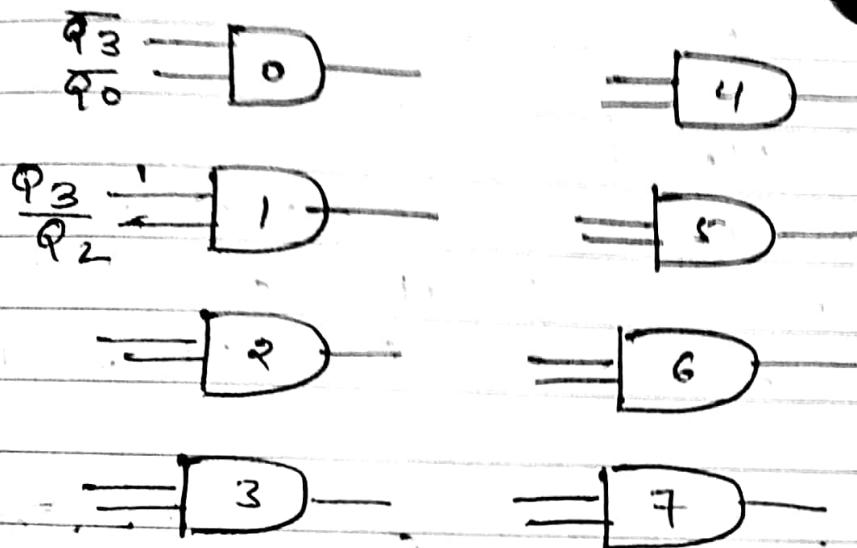
→ O/P frequency  $\rightarrow f/2^n$

→ The ring counter can be modified slightly to produce another type of shift register counter called Johnson counter.



Decoding a Johnson shift counter -  
A Johnson counter requires only half the no. of ff that a ring counter requires. However a Johnson counter requires decoding gates whereas a ring counter does not.

Decoding logic for four M.O.  
shift counter.



It requires fewer f.f than using but more than a binary.

It has more decoding logic complexity than a ring counter but less than a binary counter.

So thus it sometimes represents a logical choice for certain app.

Synchronous Sequential ckt Models -

Synch. sequential ckt is also known as a finite state machine.

The most general model of a sequential ckt has i/p, o/p & internal states.

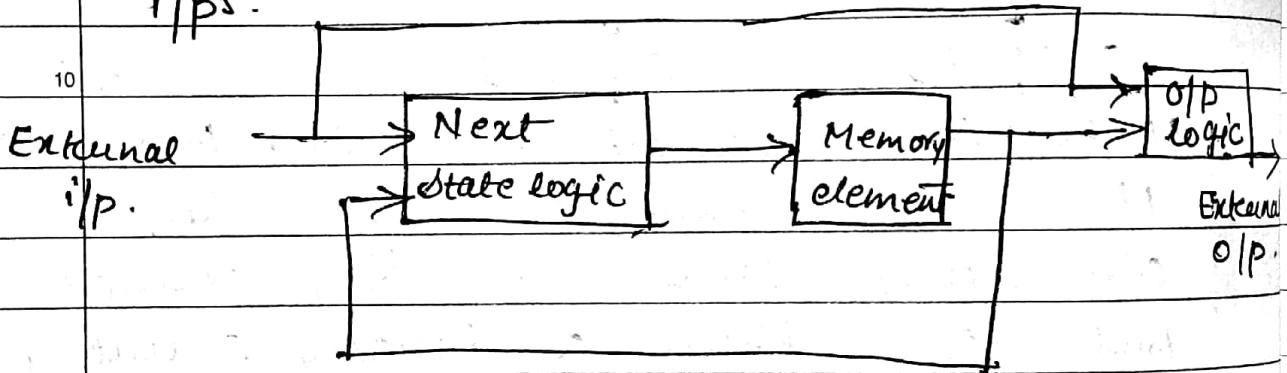
Depending upon the way the external o/p are obtained from the ckt there are 2 diff. models of

sequential dts.

- 1) Mealy Model
- 2) Moore Model.

5 Mealy Model -

In mealy model the next state is a functn of the present state and the present i/p's.



15 Moore Model -

In this model the next state is the functn of the present state and present i/p. The o/p functn of the Moore model is only the functn of the present state and independent of the i/p.

Camlin	Page
Date	/ /

