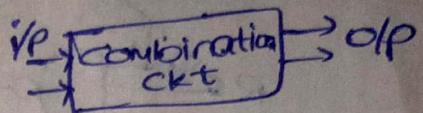
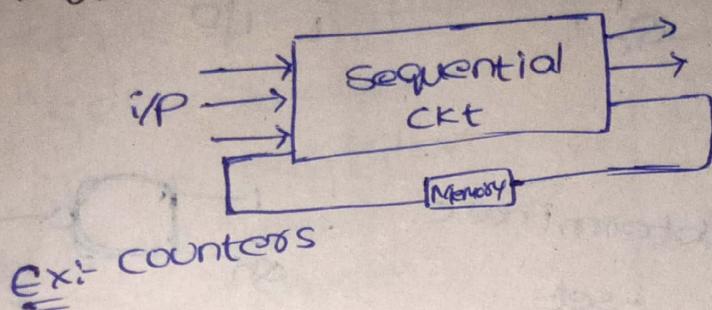


25/1/23

#### 4. Sequential Circuits.

Sequential circuit is a circuit whose output depends on present input as well as past output values.

Logical symbol:-



#### Combinational circuits

1. Combinational circuit is a circuit whose o/p depend on present i/p

2. combination circuits has no memory

3. No feedback

- A. Ex:- Adders,  
subtractors,  
Multiplexers,  
encoders.

- ⑥ There is a memory element called latch capable of storing one bit of data. latch can be represented by using universal gates (Nor, Nand).

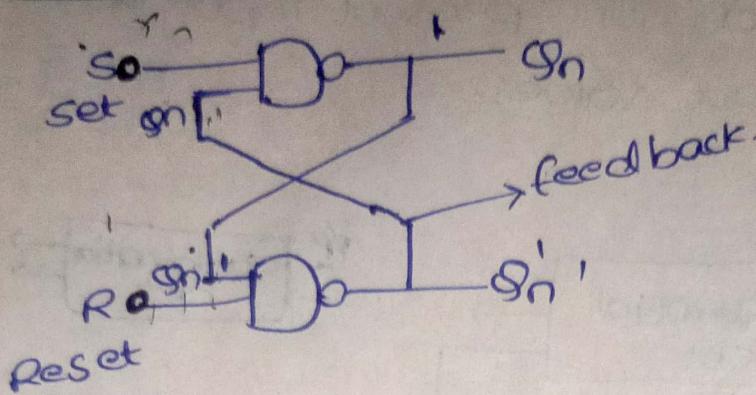
sequential circuits.

1. Sequential circuit is a circuit whose o/p depend on present i/p as well as past o/p values
2. It has memory element
3. NO clockpulse required.

- A. Ex:- Registers,  
counters,  
flipflops.

## Using NAND.

### SR latch:-



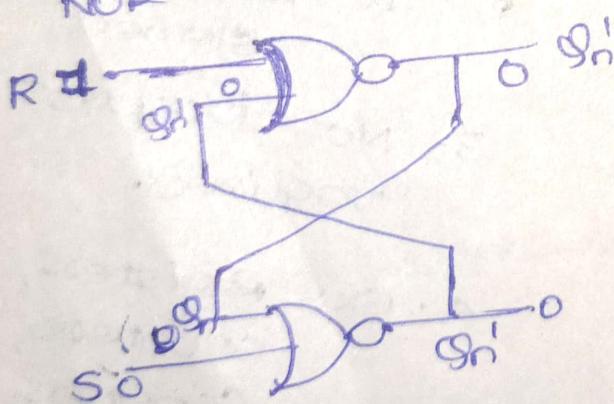
NAND	
0	1
1	0
0	0
1	1

S	R	Qn	Indeterminate
0	0	1	-set
0	1	0	-Reset
1	0	0	-
1	1	Qn	- Hold.



## Using NOR.

### NOR latch:-



S	R	Qn	Indeterminate state.
0	0	Qn	→ Hold
0	1	0	→ Reset
1	0	1	→ set
1	1		

NOR	
0	1
1	0
1	1
0	0

$$\begin{aligned}
 & (0+Qn)' \\
 & = 0' \cdot Qn' = Qn \\
 & (0+Qn')' \\
 & 1 \cdot Qn' = 1 \cdot Qn
 \end{aligned}$$

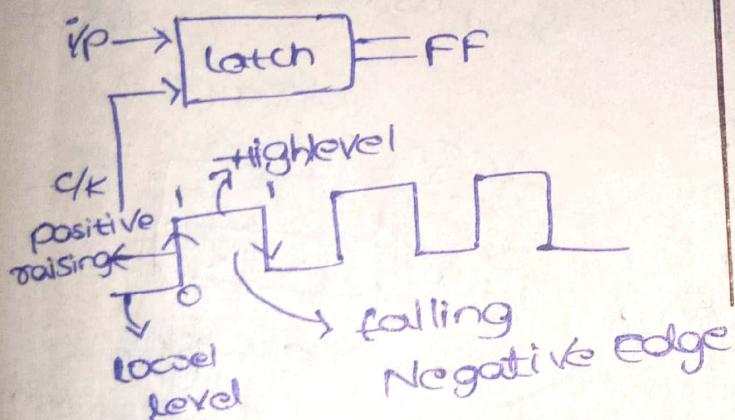
## Flip Flop:-

$$[1 \cdot Q_n] = T + Q_n'' = 0 + Q_n = Q_n.$$

If latch is connected with clock pulse then it is called as flip flop. It is also a memory which is used to store one bit of data.

There are different types of flip flops

1. SR - Set Reset
2. JK - Jack Kilby
3. D - Data
4. T - Toggle



Triggering -

Activation of

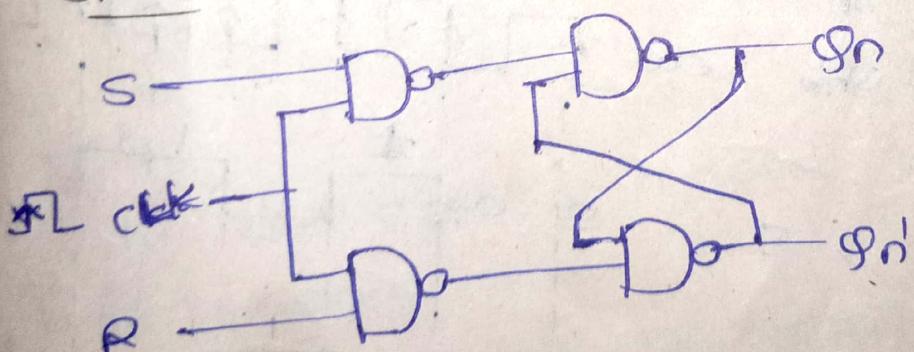
Flipflop:

Sinusoidal signals

Triangular signals

Rectangular signals

## SR flip flop:-



## Truth table:-

S	R	$Q_{n+1}$
0	0	$Q_n$ - Hold
0	1	0 - Reset
1	0	1 - Set
1	1	Indeterminate

# Characteristic Table:

S	R	$Q_n$	$Q_{n+1}$	
0	0	0	0	
1	0	0	1	1
2	0	1	0	
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	x
7	1	1	1	x

Hold  
Reset  
Set  
Indeterminate.

# Excitation Table:-

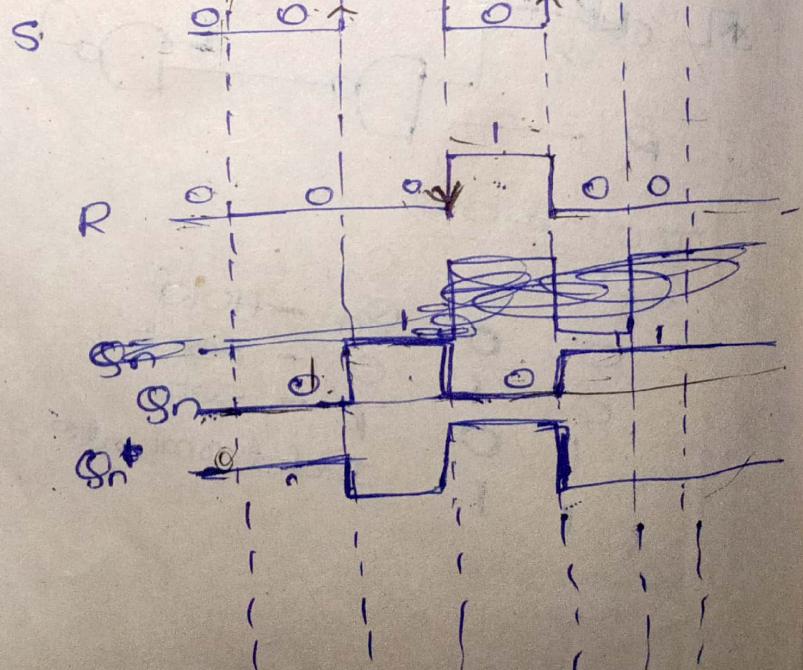
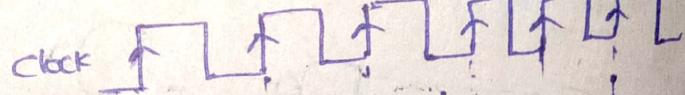
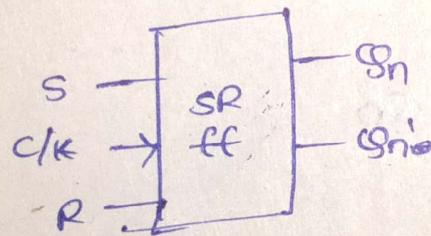
$Q_n$	$Q_{n+1}$	S	R
0	0	0	x
0	1	1	0
1	0	0	1

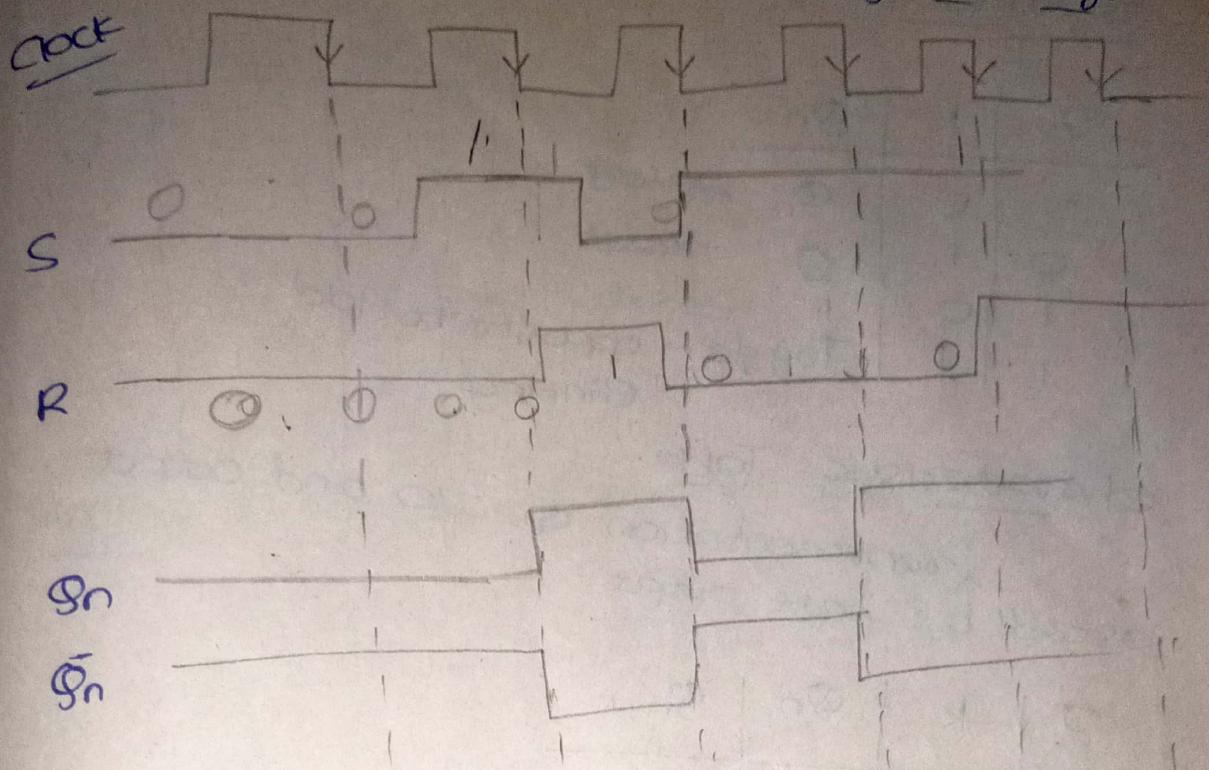
$$Q_{n+1}(S, R, Q_n) = \Sigma m(1, 4, 5) + \Sigma m(6, 7).$$

S	$Q_n$	01	11	10
0	0	1	0	1
1	1	1	1	0

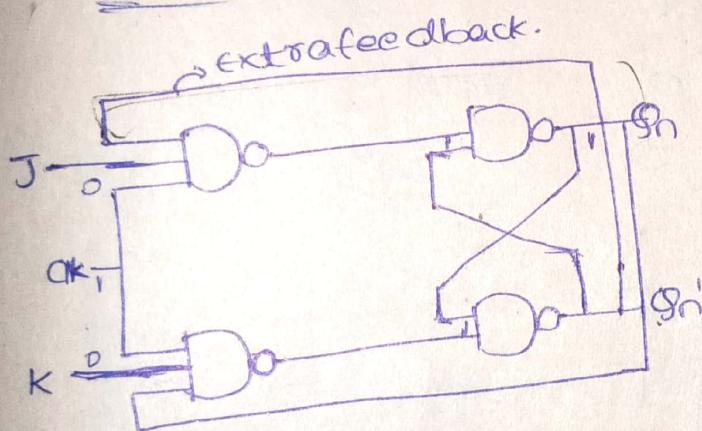
$$Q_{n+1} = S + \bar{R}Q_n$$

Timing diagram. for positive edge.





### JK- flipflop.



### NAND.

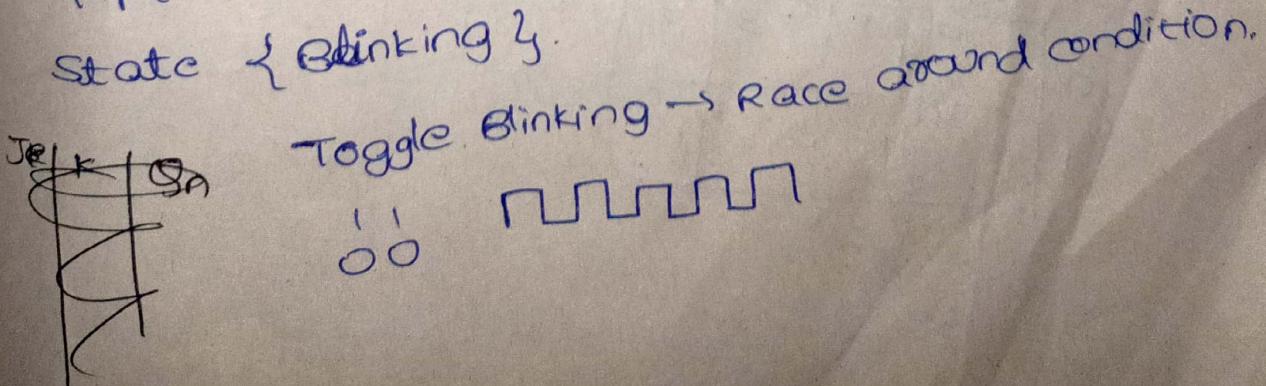
00		1
01		1
10		1
11		0

$$(1 \cdot Q_n) \cdot (1 - Q_n) = 1$$

$$(Q_n) - Q_n \cdot Q_n = 0$$

Note:- The only difference between SR and JK is

- 1) The extra feedback circuits.
- 2) JK flipflop works as similar as SR flipflop but when  $J=K=1$  it goes into Toggle state {Blinking}.



Truth table:-

J	K	$Q_n$	
0	0	$Q_n$	→ Hold
0	1	0	→ Reset
1	0	1	→ Set
1	1		Toggle → opposite to Hold Blinking

Characteristic Table.

representation of I/P and output  
as well as past output.

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

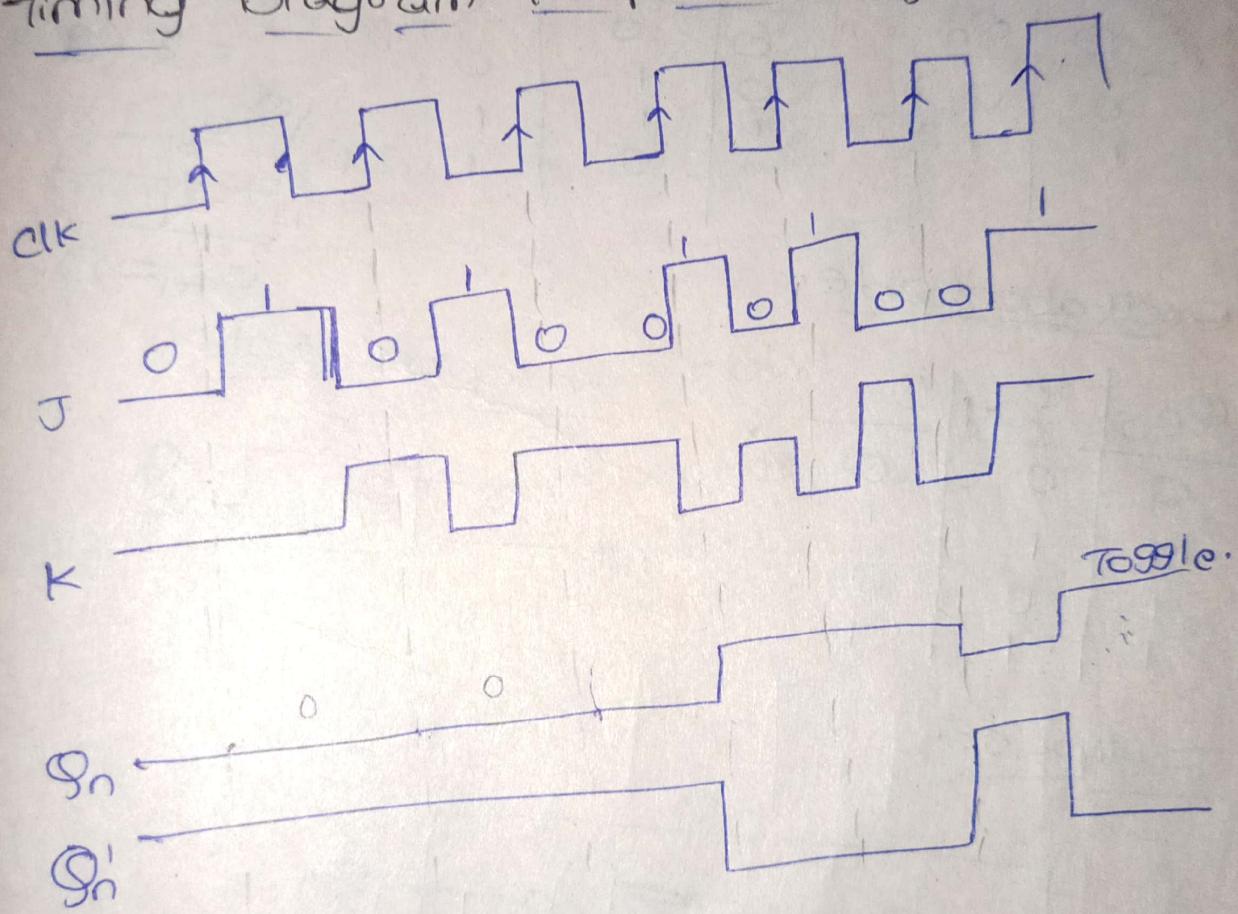
Characteristic Equation:-  $Q_{n+1} = JQ_n + K'Q_n$

$$Q_{n+1}(J, K, Q_n) = \Sigma (1, 4, 5, 6)$$

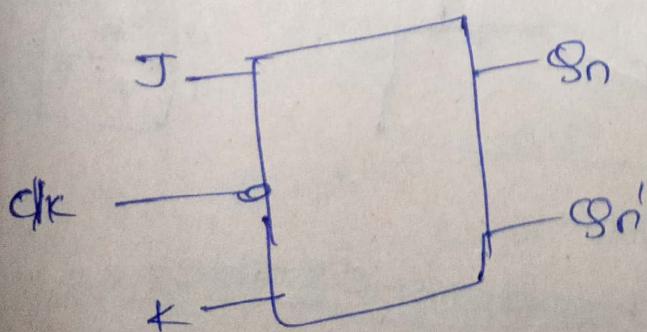
J	$KQ_n$			
	00	01	10	11
0	0	1	3	2
1	4	5	7	6

$S_n$	$S_{n+1}$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

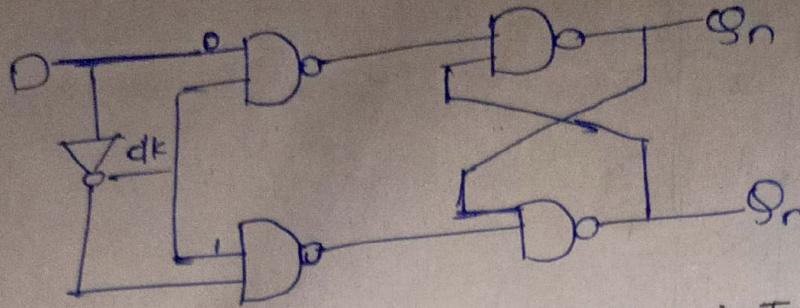
Timing Diagram for positive Edge.



IC representation:



D-flipflop. (Data flipflop)



NAND

## Tooth table.

D	$\otimes_n$
0	0

## Characteristic Table

D	$g_n$	$g_{n+1}$
0	0	0
0	-	0
0	-	1
-	0	-
-	0	-

characteristics

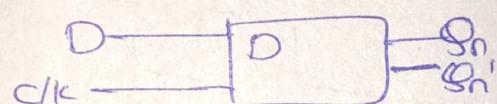
D	1	0	0
0	0	1	1
1	2	1	3

Anti FD.

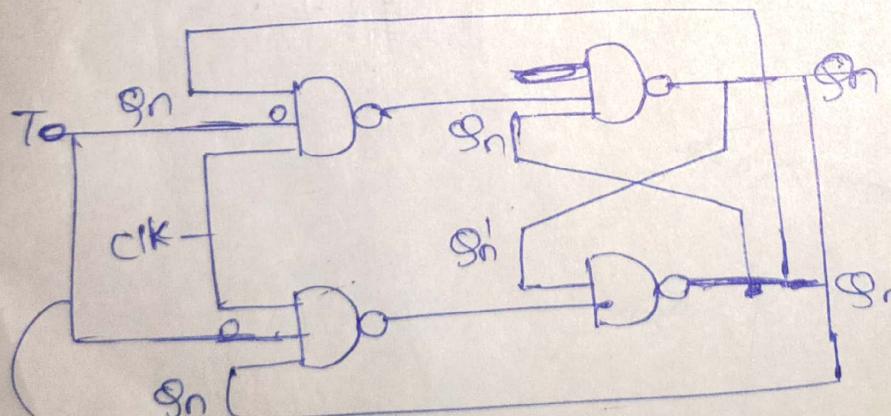
## excitation table

$g_n$	$g_{n+1}$	$D$
0	0	0
0	-1	-1
-1	0	0
1	1	-1

## IC representation



## T-flipflop



→ ~~costliest~~ shooting of inputs.

## Truth table

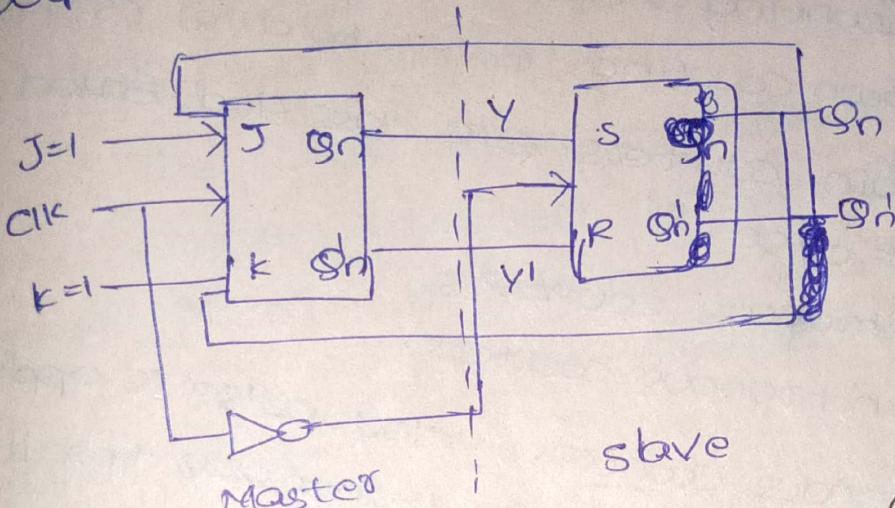
T  $\otimes_n$   
O  $\otimes_n$  - Hold  
I  $\otimes_n$  - toggle

## Characteristic table

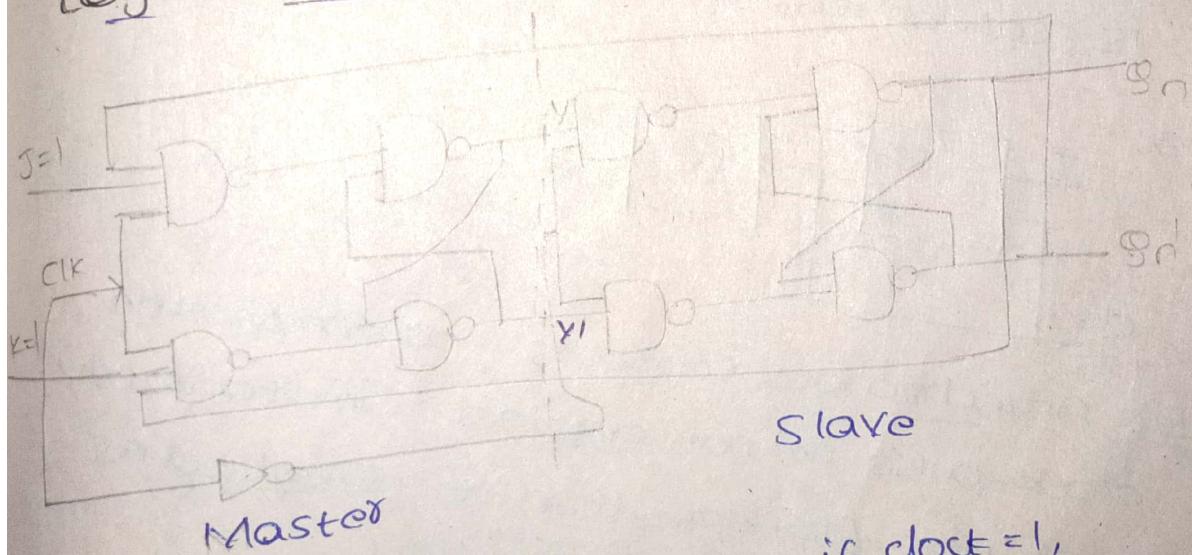
T	$q_n$	$q_{n+1}$	
0	0	0	$y_{q_n}$ hold
0	1	1	$y_{q_n}$ toggle
1	0	1	$y_{q_n}$ toggle

Master Slave flip flop.

To eliminate the drawback in J-K flip flop, whenever  $J=1$  &  $K=1$  there occurs a condition called as race around condition. In this condition flip flop keep on Toggle, which leads to uncertainty in determining the output state of J-K flip flop.



Logic diagram for Master slave flip flop.



1) In Master slave flip flop if  $clock = 1$ , master is active and it produces  $Y$  &  $Y'$  and slave will be inactive by using NOT gate.

2) If  $clock = 0$ , Master becomes inactive and slave becomes active.

Master becomes inactive  
and slave becomes active.

3/11/23

## 5. Counters.

Counter is also a sequential circuit which is used to count <sup>no. of</sup> pulses.

→ Depending upon the process of counting

It can be divided into 2 types.

1) Upcounting -  $\alpha$ -elections (0, 1, 2, 3, 4, 5)

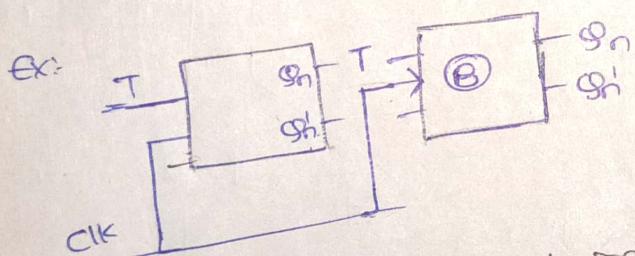
2) Down counting - (satellite launching). (5, 4, 3, 2, 1, 0)

Again counters are classified based on clock noise.

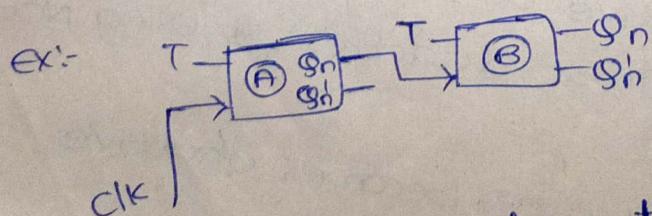
1) Synchronous counters.

2) Asynchronous counters.

1) Synchronous Counters - clock pulse is applied to all the flipflops simultaneously then it is called as synchronous counter.



2) Asynchronous Counters - whenever the clock-pulse is not applied simultaneously to all the flipflops then it is called as asynchronous counters.



clock pulse is applied to the first flop and remaining flipflops gets their clock pulse from the preceding flop output.

	$S_n^+$	$S_n'$
+ve	Down	Up
-ve	Up	Down

same - Down counting  
different - up counting

Counters are constructed by using Toggle state flipflops only. (JK flipflop & T flipflop)

Modulus of a Counter.

The No. of states available to count in a counter is called Modulus (Q) "Mod".  
Where  $N = 2^n$        $N = \text{mod} (\text{No. of states})$   
 $n = \text{no. of flipflops required.}$

MOD 3 - counter.

i)  $0, 1, 2.$  = no. of states.

$3 = 2^n$   
choose the max (Q) as 10. 2 flipflops are used

also,  $\log_2 N.$

$$\log_2 (3) = 1.5 \approx 2$$

$$\log_2 (78) = 7.8 \approx 8.$$

ii)  $0-1-8-5-3.$

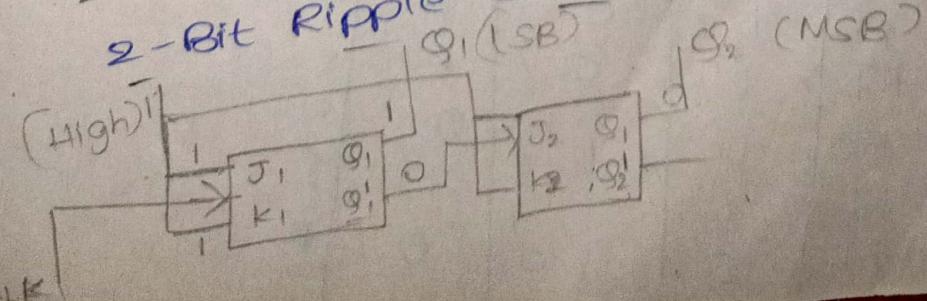
max - 8 - 1000 - 4 flipflops used.

2-Bit Asynchronous upcounter.  $N = 2^n = 2^2 = 4.$

2-Bit Asynchronous upcounter. (Q)

Mod 4 ASynchronous upcounter. (Q)

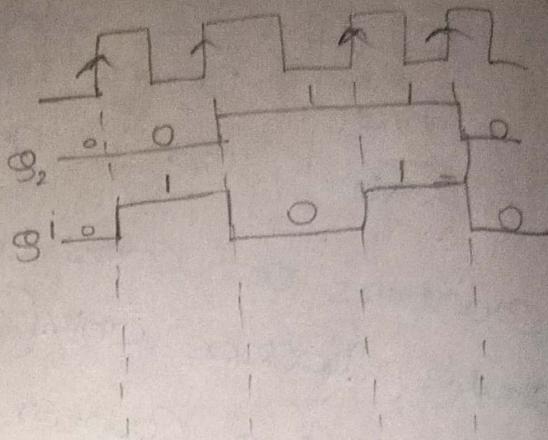
2-Bit Ripple counter.



### Truth Table.

	$Q_2$	$Q_1$	Decimal
①	0	0	0
②	0	1	1
③	1	0	2
④	1	1	3
⑤	0	0	0

### Timing Diagram.



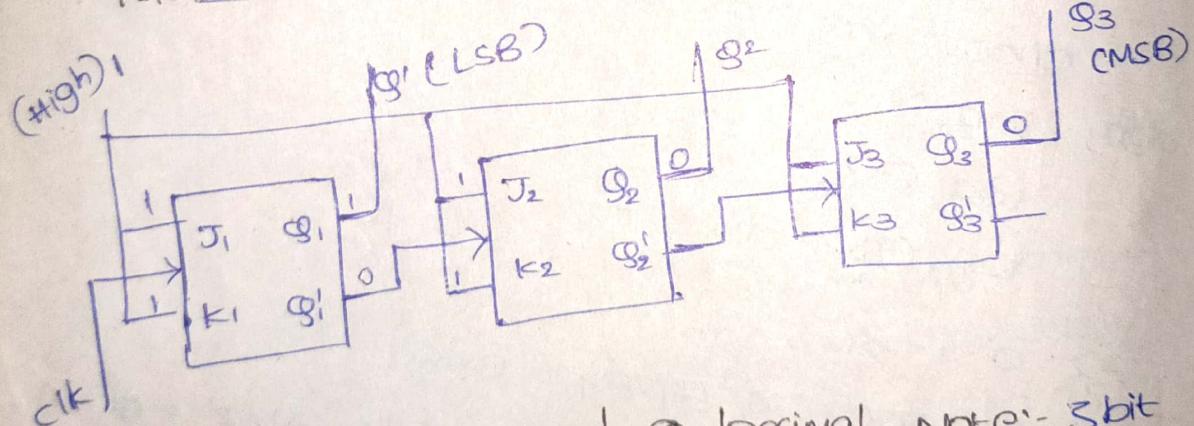
Note:-  
Mod 4 ripple counter needs 4 clock cycles  
(N clock cycles)

The same circuit is designed for downcounting  
when clock pulse is connected from  $Q_1$ ,  
it becomes upcounter. (Q<sup>1</sup>)

3-bit Asynchronous Upcounter (Q<sup>1</sup>)

3-bit ripple counter (Q<sup>1</sup>)

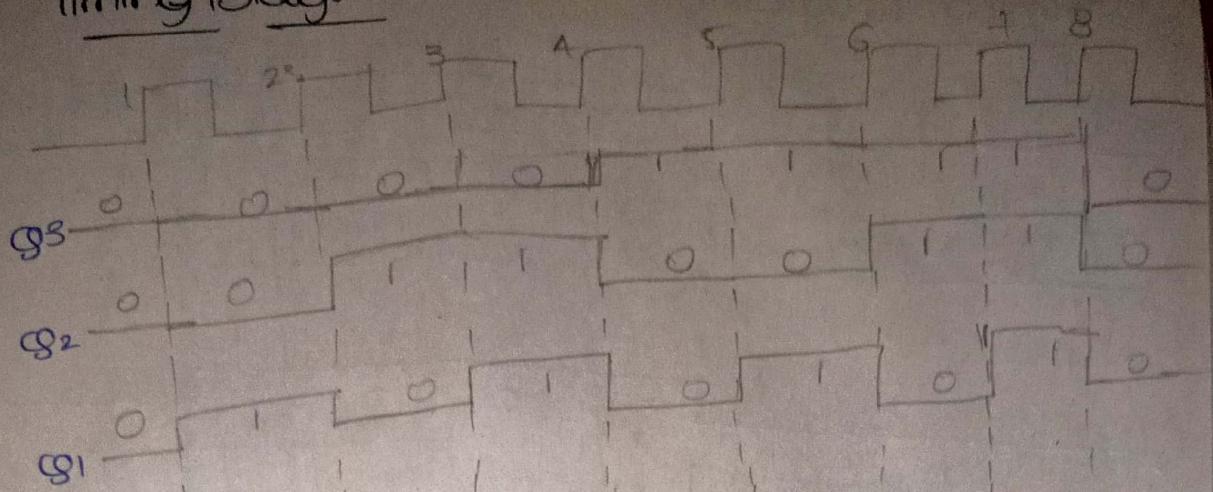
Mod-8 Asynchronous Upcounter



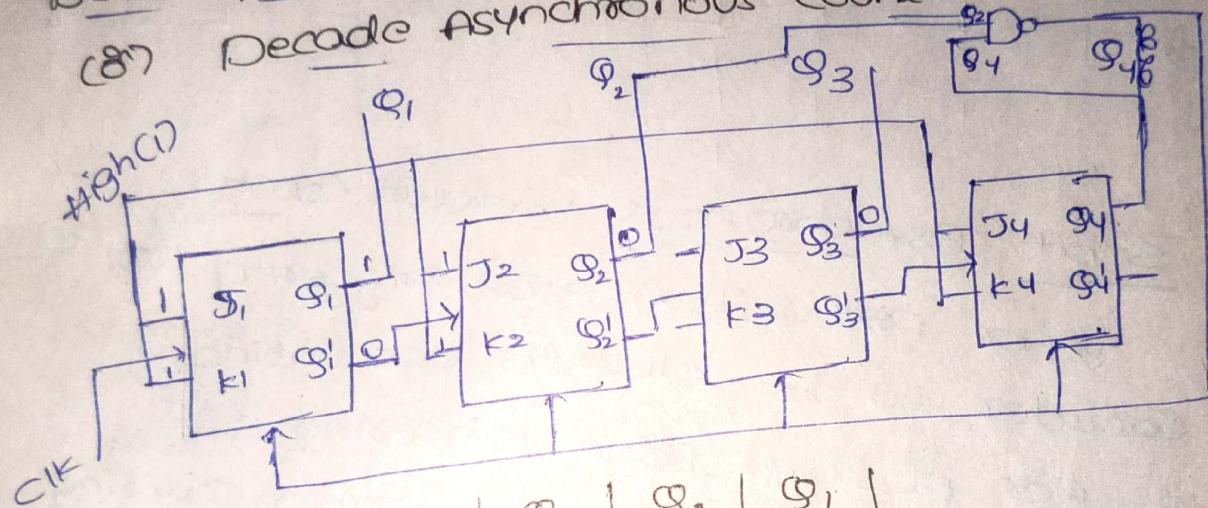
clk	$Q_3$	$Q_2$	$Q_1$	Decimal
1)	0	0	0	0
2)	0	0	1	1
3)	0	1	0	2
4)	1	0	0	3
5)	1	0	1	4
6)	1	1	0	5
7)	1	1	1	6
8)	0	0	0	7

Note:- 3 bit  
ripple counter  
needs  
8 clock pulses!

# Timing Diagram.

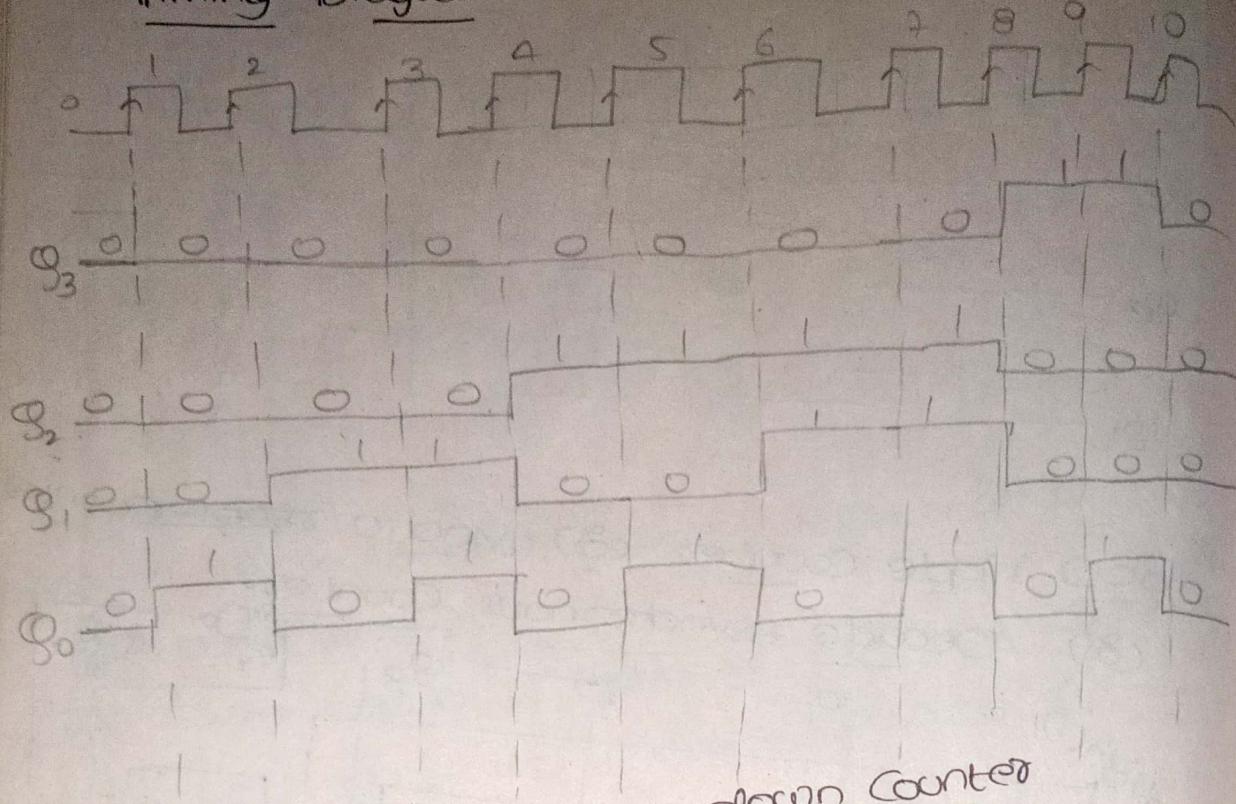


\* BCD ripple counter (8) Mod 10 Ripple Counter  
 (8) Decade Asynchronous Counter.



CLK	Q <sub>4</sub>	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	
0	0	0	0	0	0
1	FL	0	0	1	1
2	FL	0	0	0	2
3	FL	0	1	1	3
4	FL	0	0	0	4
5	FL	0	1	1	5
6	FL	0	0	0	6
7	FL	0	1	1	7
8	FL	1	0	0	8
9	FL	1	0	0	9
10	FL	0	0	0	10
					11
					12
					13
					14
					15

## Timing Diagram.



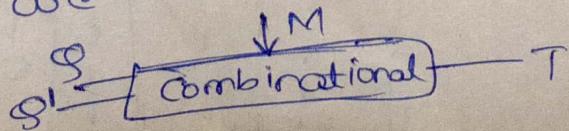
### 3-bit Asynchronous updown Counter

- i) for positive (tve) edge triggering  $S_1'$   
consider to be clock for upcounting.
- ii) for positive edge triggering  $S_1$   
considered to be clock for down counting.
- iii) for negative edge triggering  $S_1$   
considered to be clock for upcounting.
- iv) For negative Edge triggering  $S_1'$   
considered to be clock for down counting.
- To decide Upcounting & downcounting  
we need a controlling input called mode Control (M)

If  $M=0$  it performs upcounting

$M=1$  downcounting,

for this, we are designing a combinational circuit



# Tooth Table.

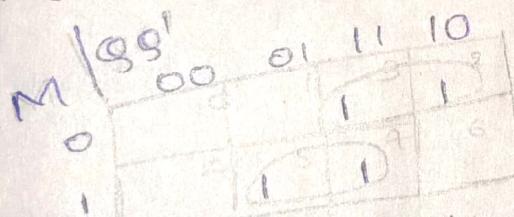
	M	$Q^0$	$Q^1$	Y or P
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

up counting  $Q^n$

down counting  $Q^n$

using K-map

$$Y(M, Q, Q^1) = \sum(2, 3, 5, 7)$$



$$Y = M Q^1 + M' Q.$$

$$Y = M \oplus Q.$$

Circuit

Negative-edge

