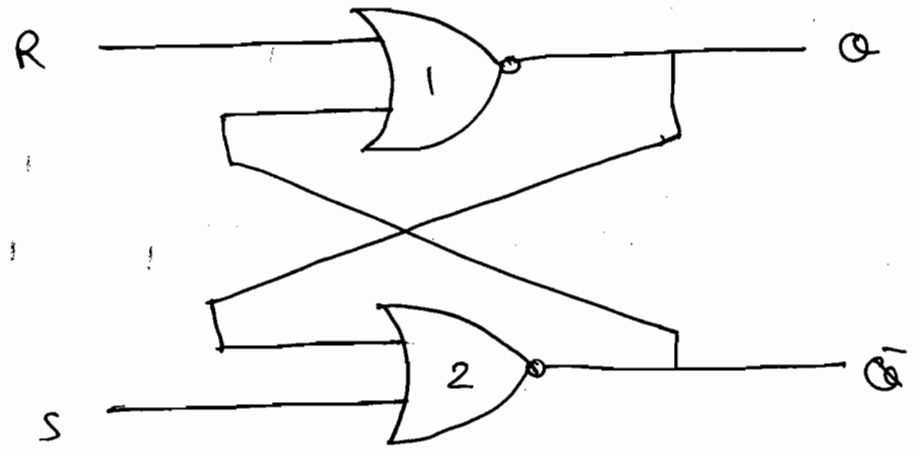
# ☆ Sequential Circuits: (1 bit memory elements)

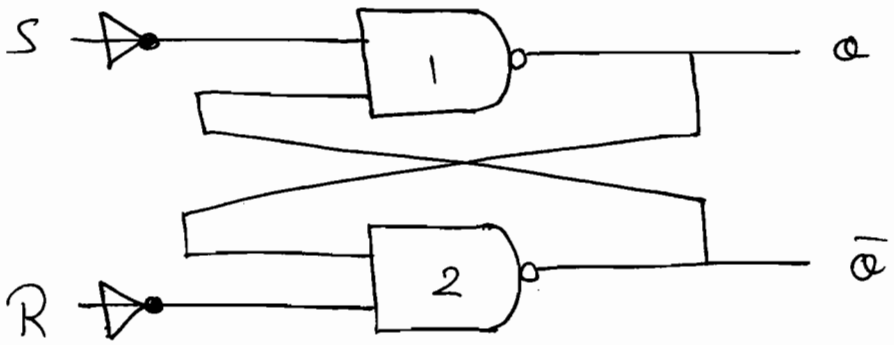
## ⇒ * S-R Latch:




| S | R | Q |
|---|---|---|
| 0 | 0 | No change |
| 0 | 1 | '0' |
| 1 | 0 | '1' |
| 1 | 1 | Impracticle state |

(∵ $Q = \bar{Q} = 0$).

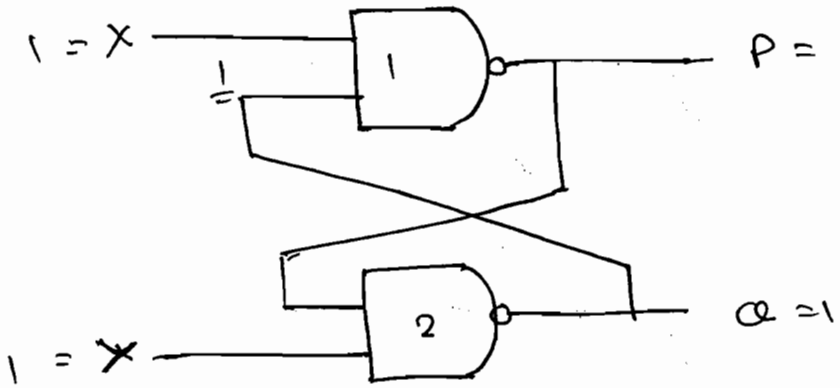
⇒




*

| S R | Q |
|---|---|
| ~~01~~ | 1 |
| 00 | 1 |

Ex-1 In the following x-y Latch initialy x=1 & y=1. Determine the outputs P & Q If the y input is change as 0,1,0,1,0,1,....

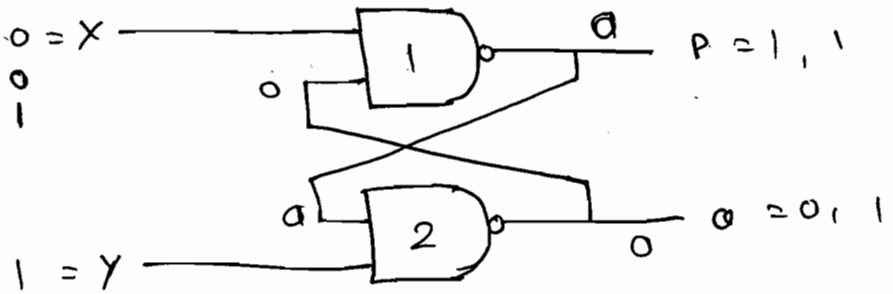



1 = X
0 = Y
1 = Y
0 = Y

So, P is fixed at P=0, Q=1.

Ex-2 In the following x-y Latch the seq. of inputs are XY = 01, 00 and 11. determine the values of P & Q.

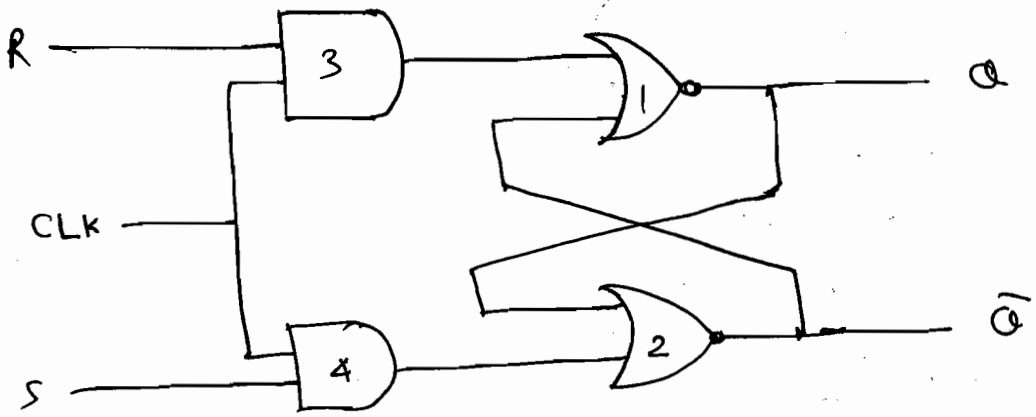
10, 1Φ, QQ.
10, 11, 0Φ.

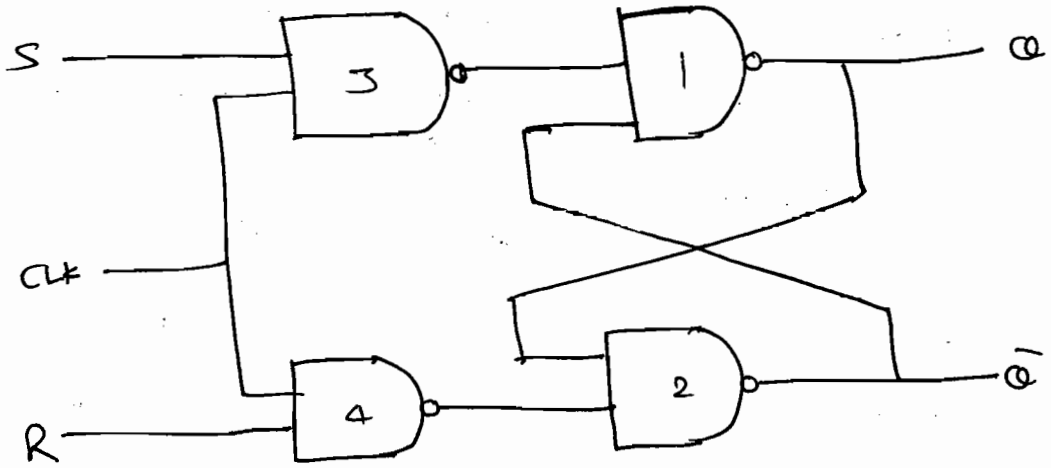


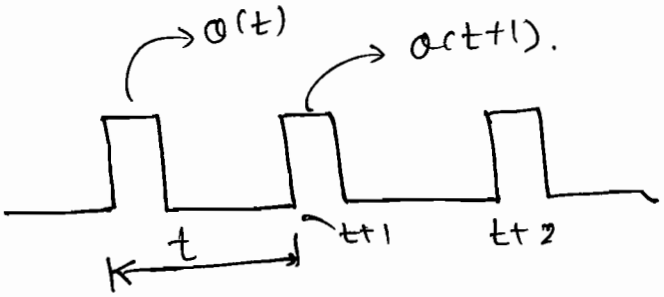

| X Y | P Q |
|---|---|
| 0 1 | 1 0 |
| 0 0 | 1 1 |
| 1 1 | 0 1 ← If Gate 1 is faster than Gate 2. |
| | (OR) |
| | Φ0 ← If Gate 2 is faster than Gate 1. |

## * Clocked S-R flip flop:

→ Synchronized latch is called Flip-flop.




T = Clock period

$\frac{1}{T} = f$ = Clock freq.

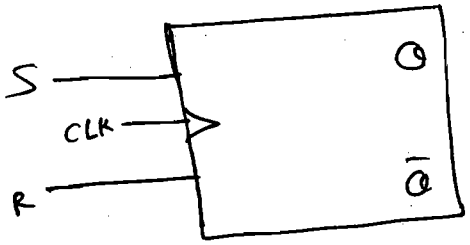
## * Truth table:

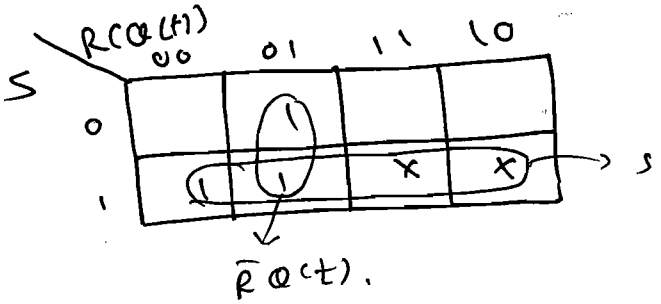
| S | R | Q(t+1) |
|---|---|---|
| 0 | 0 | Q(t) |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | Ambiguous state |

do not apply.

## * Characteristic Table:

| S | R | Q(t) | Q(t+1) |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| | | 1 | ϕ |
| 0 | 1 | 0 | 0 |
| | | 1 | 0 |
| 1 | 0 | 0 | 1 |
| | | 1 | 1 |
| 1 | 1 | 0 | X |
| | | 1 | X |



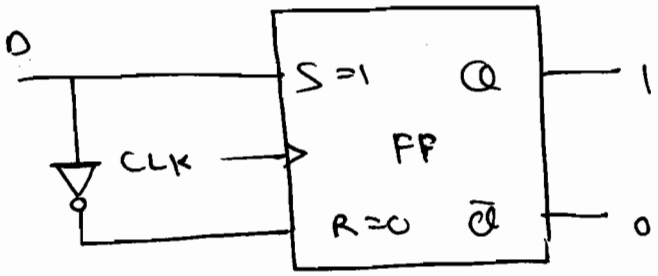




$$Q(t+1) = S + \bar{R}\,Q(t)$$

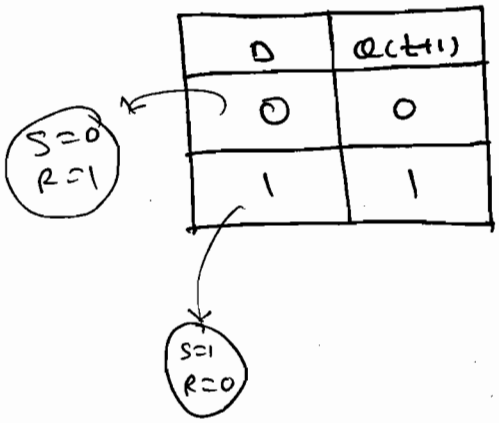
# * Clocked D-flip flop:

→ used in shift Register

→ D → Data, Delay.




## * Truth table:



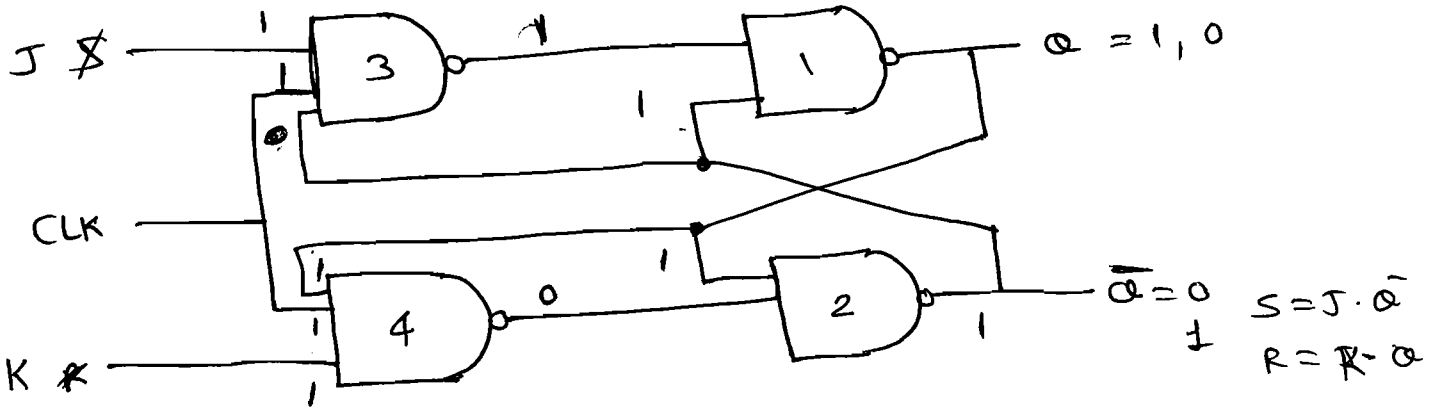

## * Characteristic table:

| D | Q(t) | Q(t+1) |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

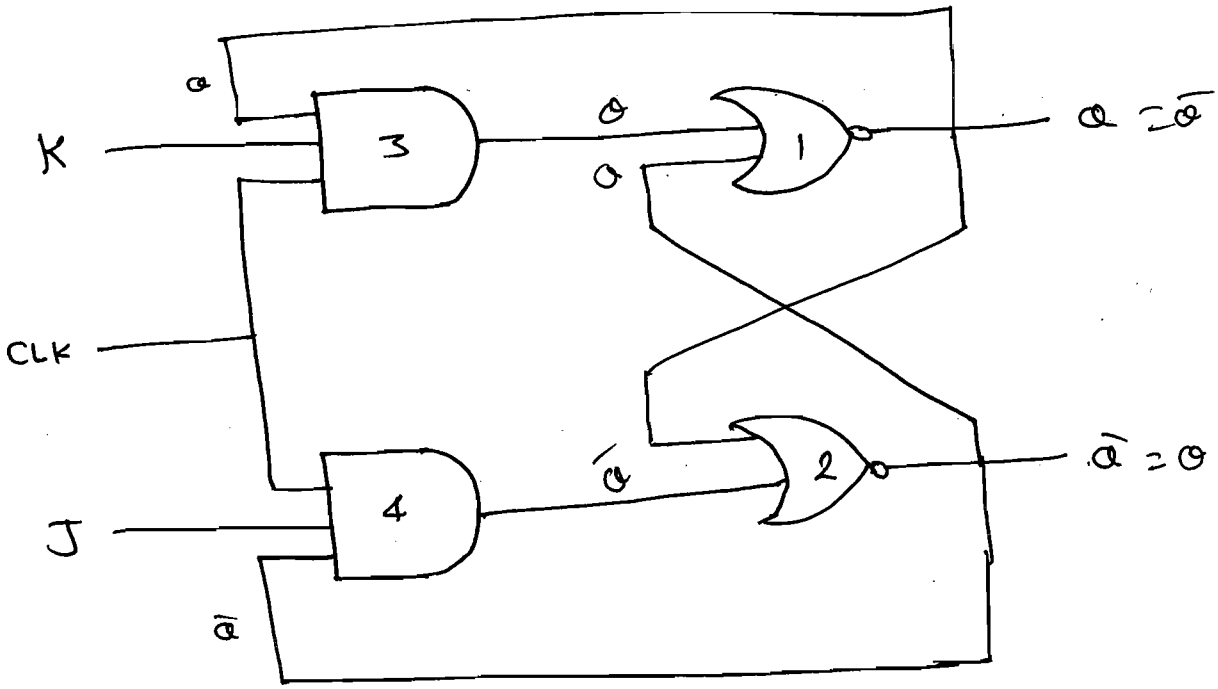
## * Characteristic equation:

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

# * Clocked J K - Flip Flop.




(or)




## * Truth table:

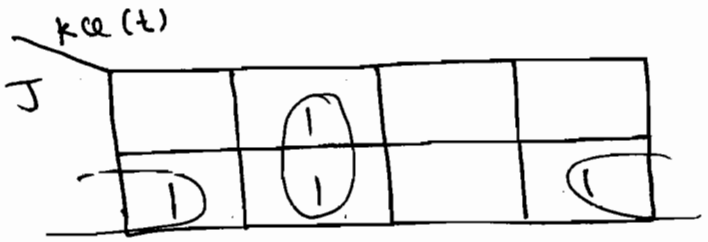
| S / J | R / K | Q (t+1) |
|---|---|---|
| 0 | 0 | Q (t) |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | $\overline{Q(t)}$ |

## * Characteristic table:

| J | K | Q(t) | Q(t+1) | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Q(t) |
| | | 1 | 1 | |
| 0 | 1 | 0 | 0 | 0 |
| | | 1 | 0 | |
| 1 | 0 | 0 | 1 | 1 |
| | | 1 | 1 | |
| 1 | 1 | 0 | 1 | $\overline{Q(t)}$ |
| | | 1 | 0 | |

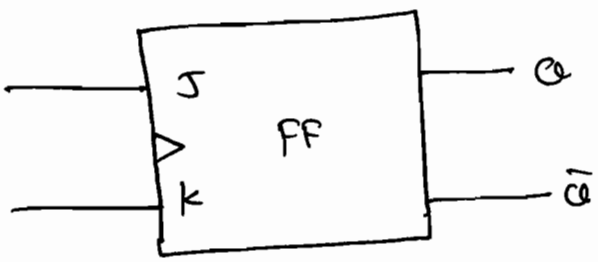
toggle

## * Characteristic equation:

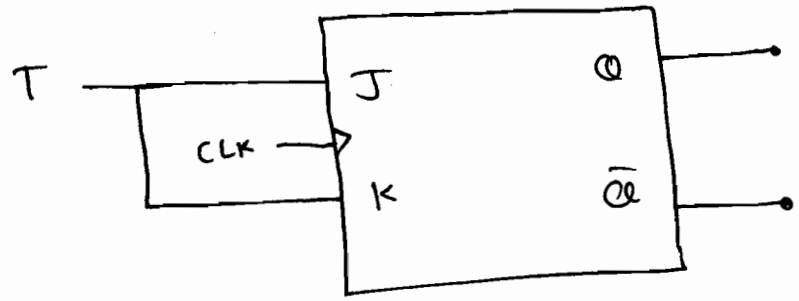



$$\therefore \quad Q(t+1) = J \cdot \overline{Q(t)} + \bar{K} \cdot Q(t)$$

*



## * Clocked T - Flip-Flop:




## * Truth Table:

| T | Q(t+1) |
|---|---|
| 0 | Q(t) |
| 1 | $\overline{Q(t)}$ |

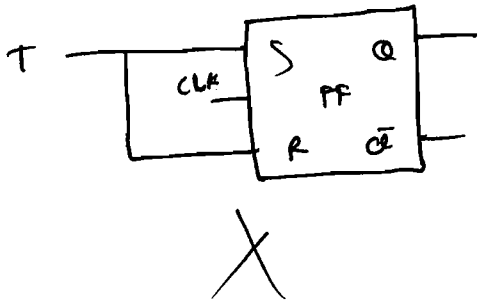
## * Characteristic Table:

| T | Q(t) | Q(t+1) | |
|---|---|---|---|
| 0 | 0 | 0 | } Q(t) |
| 0 | 1 | 1 | |
| 1 | 0 | 1 | } $\overline{Q(t)}$ |
| 1 | 1 | 0 | |

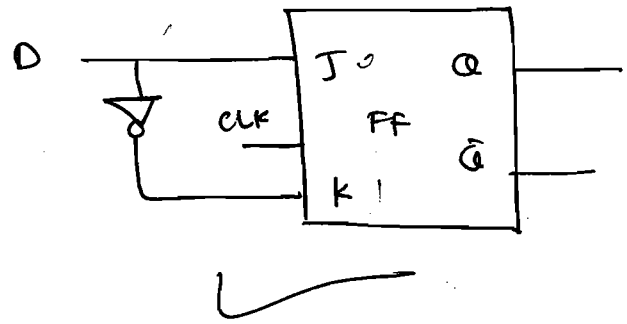
## * Characteristic eqn:

$$Q(t+1) = T \oplus Q(t)$$

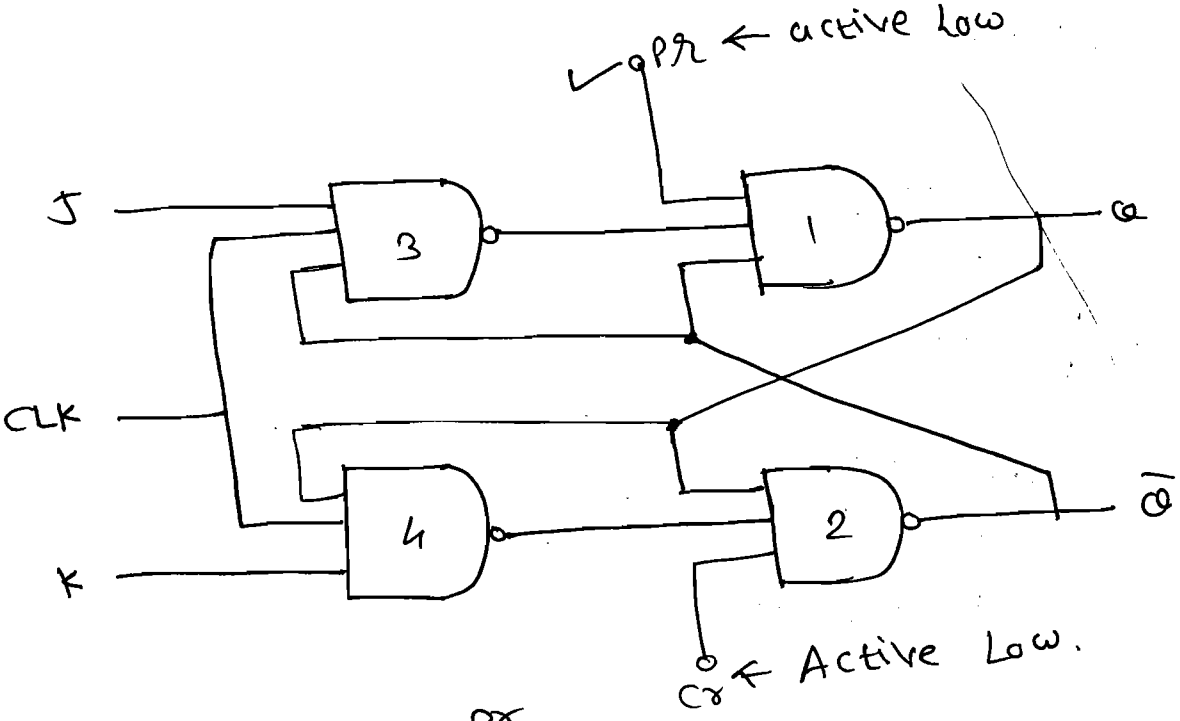
(a)

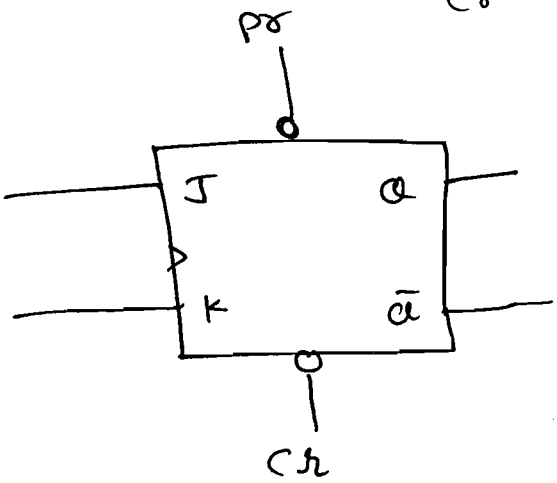



(b)




* Asynchronous (or) Direct Inputs → preset (Pr), clear (Cr).




| CLK | Pr | Cr | Q |
|---|---|---|---|
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| ⎍⎍⎍ | 1 | 1 | depends on FF I/P. |

## * Setup, Hold times of FF.

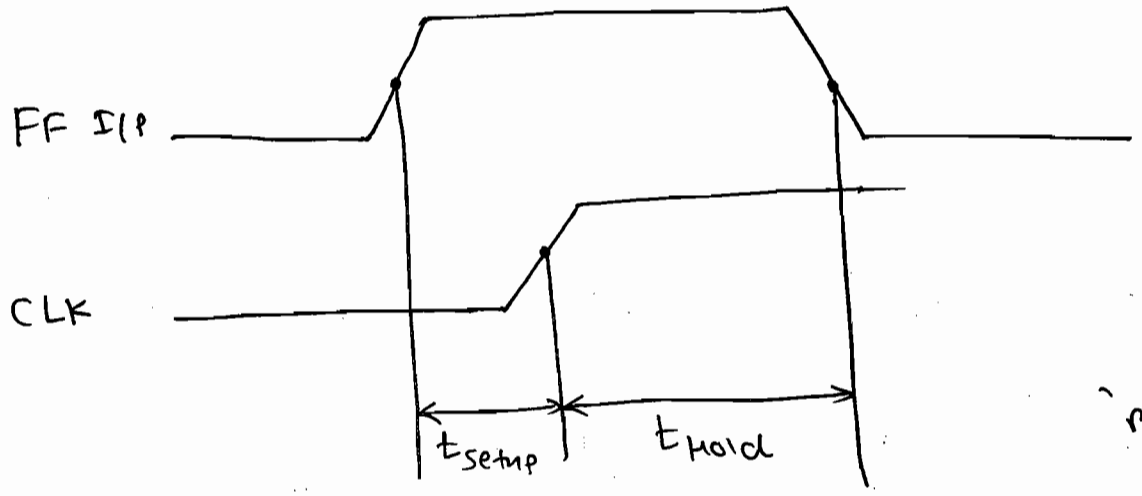
### (1) Setup time:

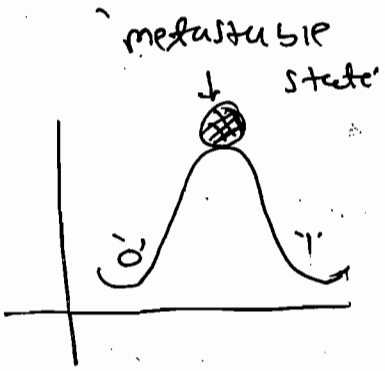
→ It is the minimum time by which the Input should come ahead of the clocked input.

### (2) Hold time:

→ It is the minimum time for which the input should be maintained constant after applying the clock pulse.

→ If setup time and hold times are not satisfies then the FF enters into metastable state i.e. neither zero nor '1' output.

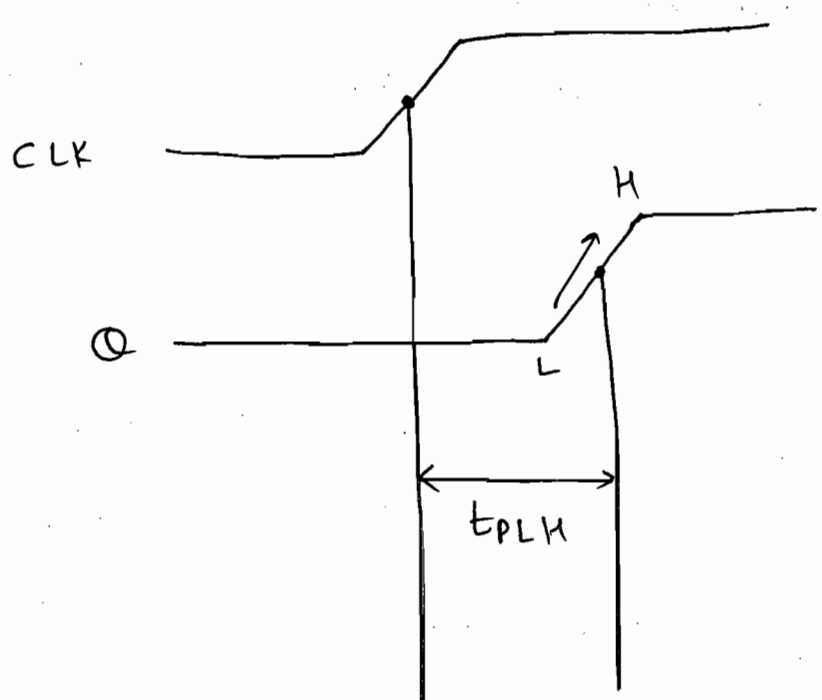


## * Propogation Delay for FF.

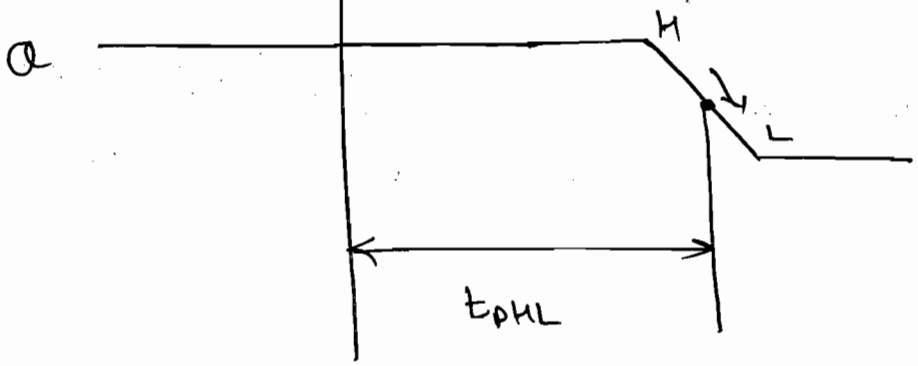
→ It is the time taken by the flip flop to change its state. (i.e. from L → H (or) H → L.).

(a)

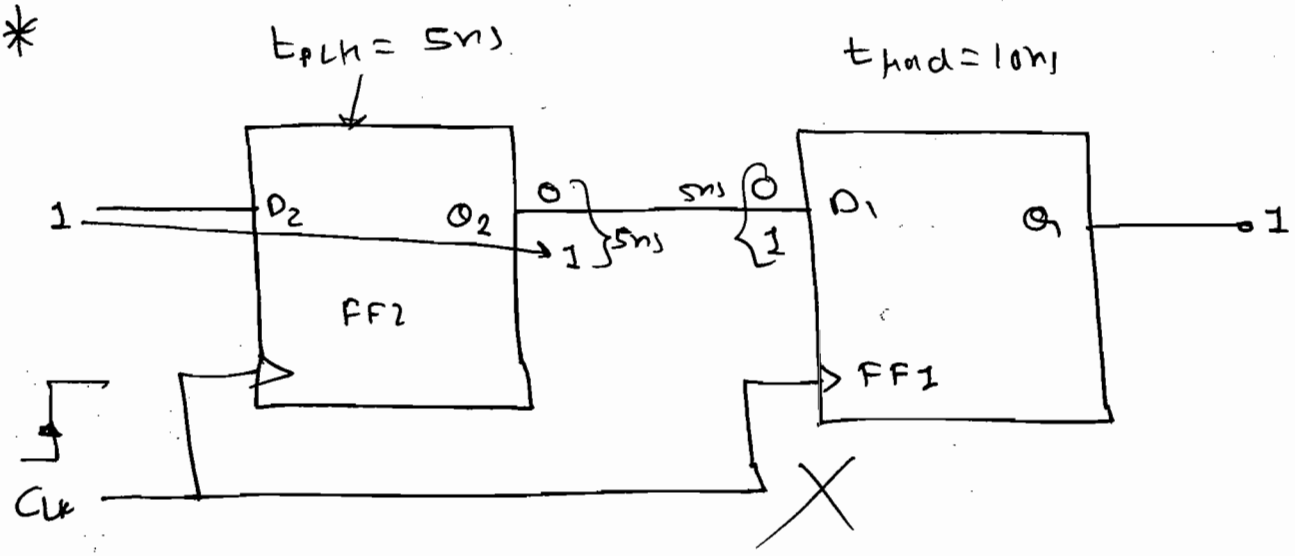



$$\therefore t_p = \frac{t_{PLH} + t_{PHL}}{2}$$

(b)



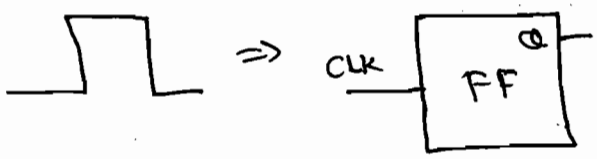

*




→ For proper FF operation $t_{Hold} < t_{PLH}, t_{PHL}$.

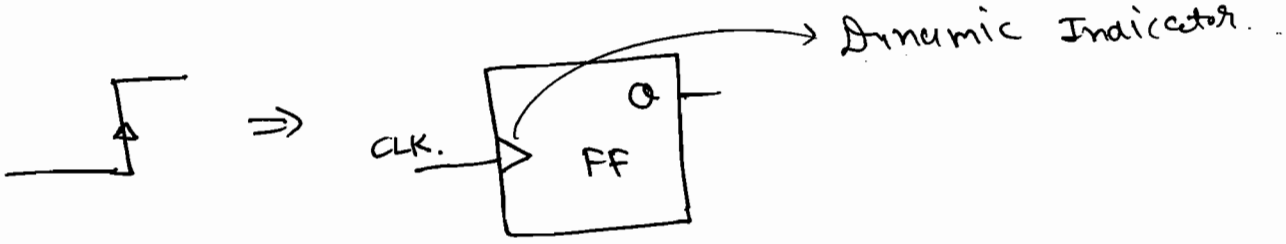
# * Types of Triggering:

## 1) Level Triggered FF

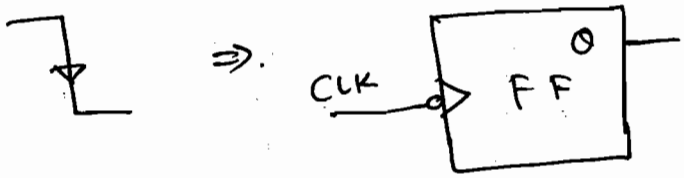



## 2) Edge Triggered FF:

### (a) Positive Edge Triggered FF (Leading Edge Triggred FF).

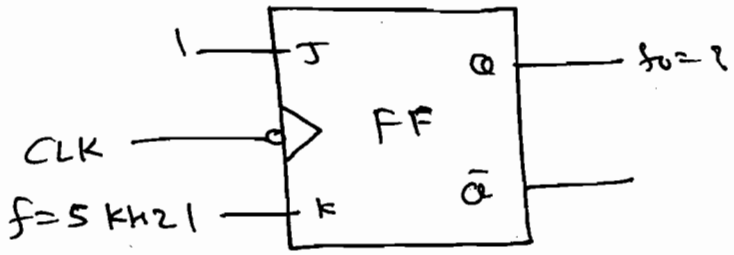



### (b) Negative Edge Triggered FF (Trailing Edge Triggered FF).



* Determine the o/p freq. of following FF if the CLK freq is 5 KHz.

(a)



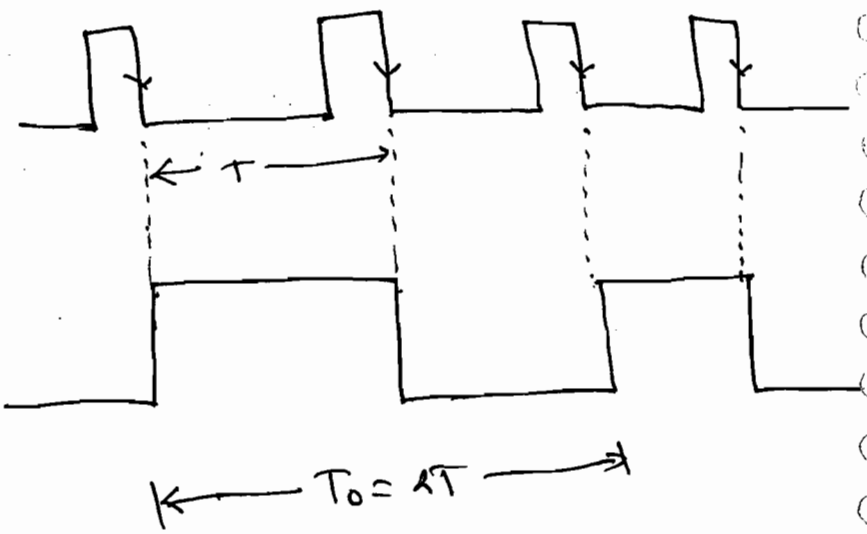

NOTE:

→ If a FF is in toggle mode the o/p freq is half of the CLK freq. with 50% duty cycle. (i.e) A toggle mode FF will act as freq. divider by 2.

→ $J = K = 1 \Rightarrow$ FF is in Toggle.

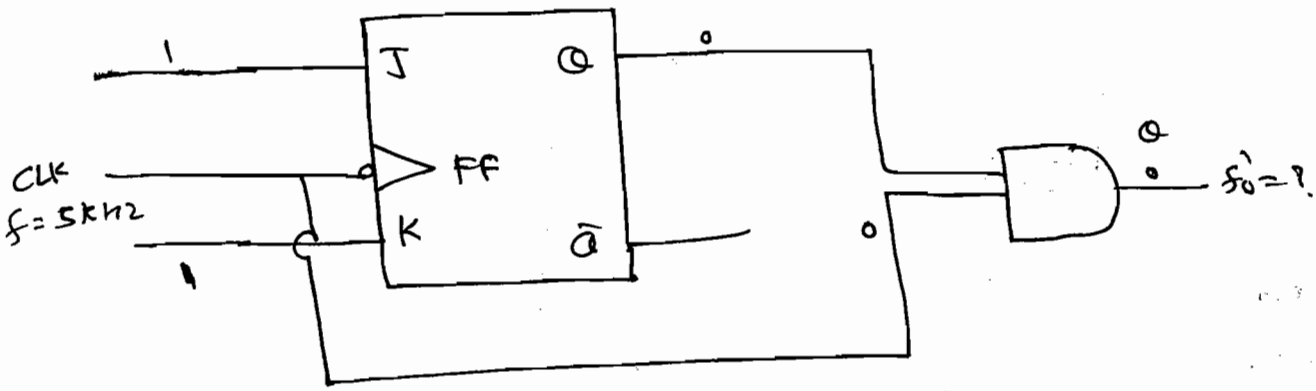
i.e. $\boxed{Q(t+1) = \overline{Q(t)}}$.

| CLK | Q |
|---|---|
| 0 | 0 |
| 1 | ↓ 1 |
| 2 | ↓ 0 |
| 3 | ↓ 1 |
| 4 | ↓ 0 |

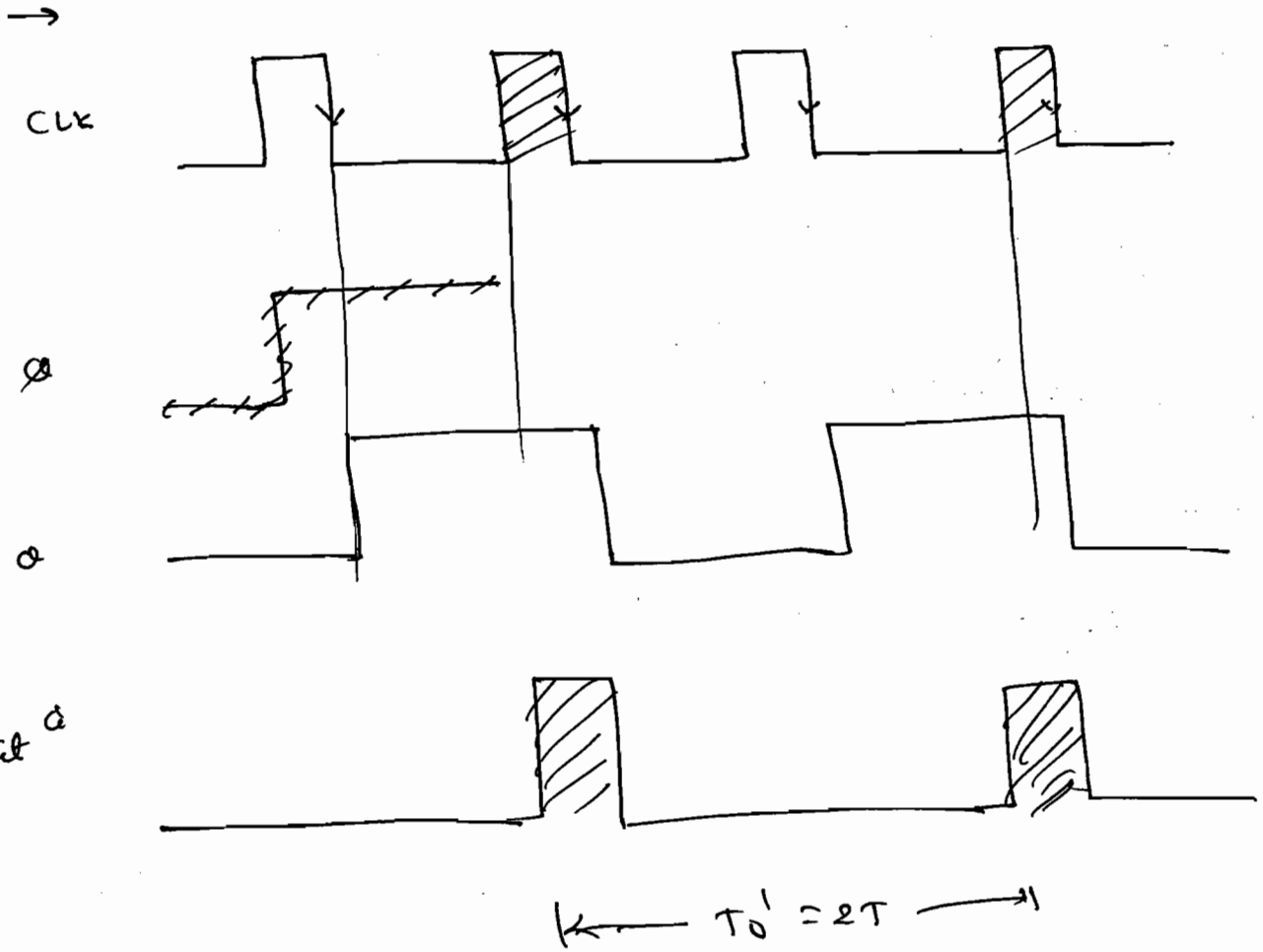



$\frac{1}{T_0} = \frac{1}{2T} \Rightarrow \boxed{f_0 = \frac{f}{2}}$ $\therefore f_0 = 2.5$ KHz.

(b)



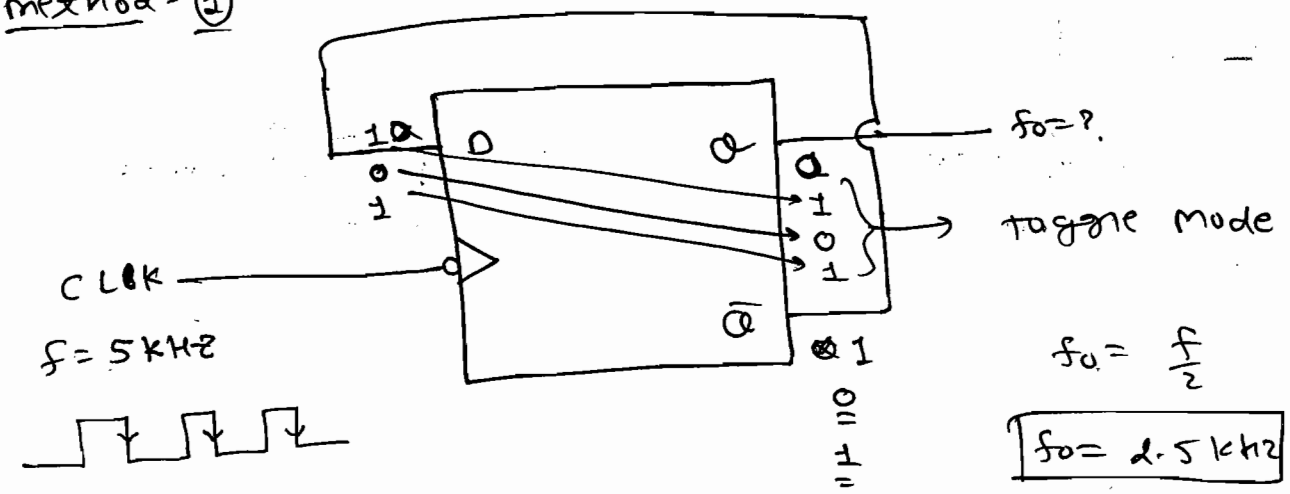

| CLK | O/P at Q |
|---|---|
| 0 | 1 |
| 1 | 0 |
| ↓ 2 | 1 |
| 3 | 0 |
| ↓ 4 | 1 |




$\therefore$ $\boxed{f_0 = 2.5\text{KHz}}$

(c)

* method - (1)




→

* method - (2)

→ D - FF Char. eqn ⇒ $Q(t+1) = D$. —(1)

But $D = \bar{Q}(t)$. —(2)

Put (2) in (1)

∴ $\boxed{Q(t+1) = \bar{Q}(t).}$

FF is in "toggle mode"

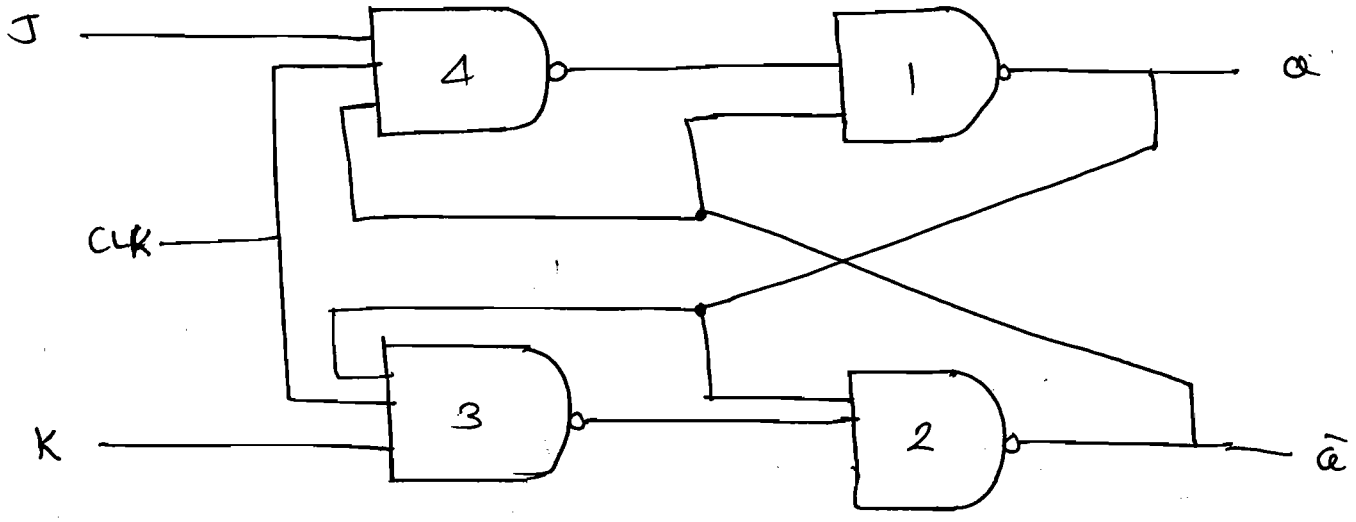
Hence $f_0 = \frac{f}{2}$

∴ $\boxed{f_0 = 2.5 \text{ kHz}}$

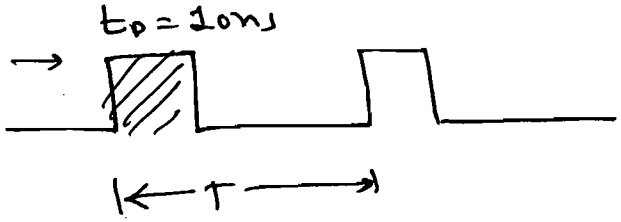
## * Race Around Condition: (RAC).

→ RACE AROUND CONDITION occurs in Level trigger flip flop and doesn't occur in edge triggered FFs.

→ $\Delta t$ = FF prop. delay = 2ns




→




⇒ "RAC" is when $J=K=1$ and $t_p \gg \Delta t$.

→ Output Toggels many times instead of once.

* How to avoid RAC (in level Triggered FF).

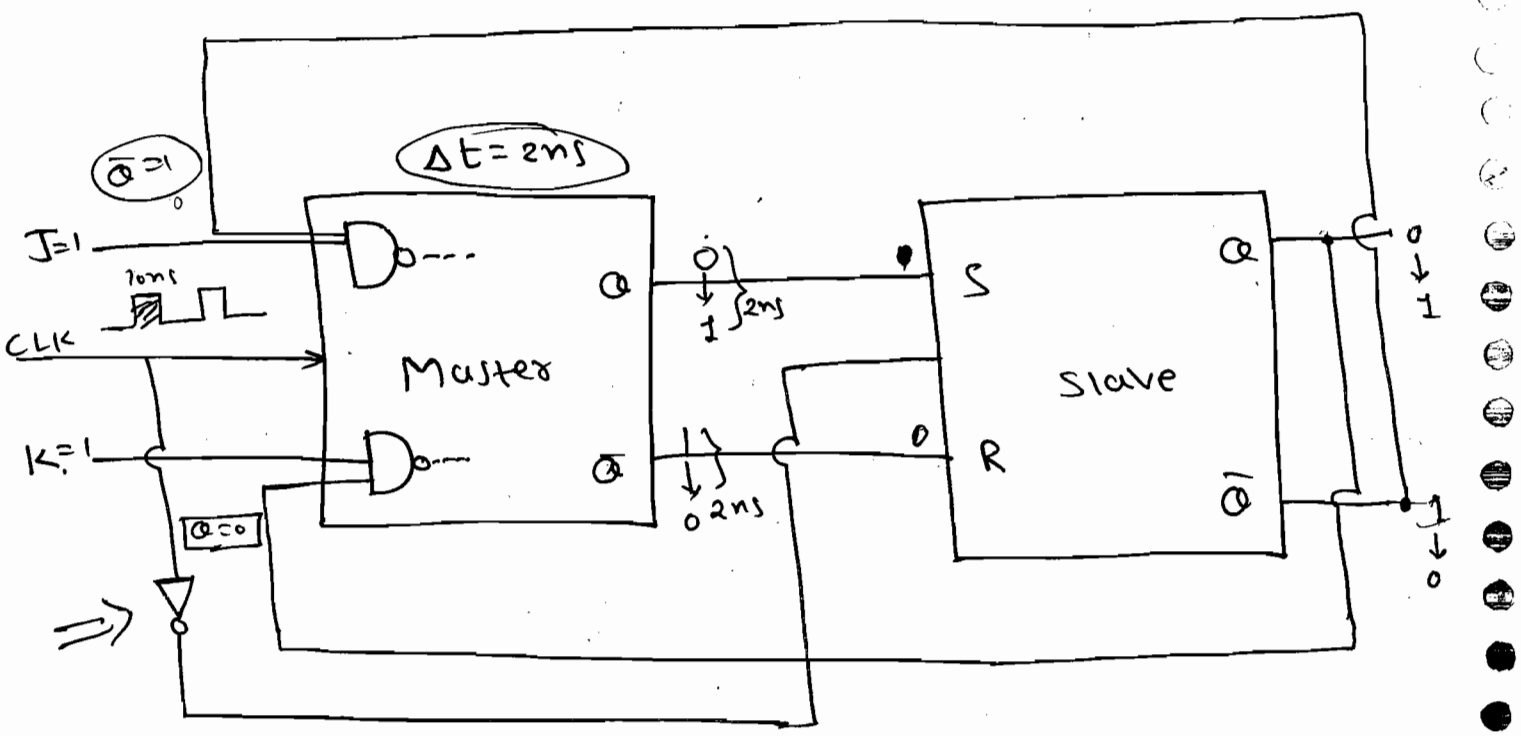
① Choose FF prop delay '$\Delta t$' such that.

$$t_p < \Delta t < T.$$

② Master-Slave JK FF.

## * Master Slave JK FF

→ In master slave JK FF the feedback values Q and $\bar{Q}$ do not change during the clock pulse. eventhough the output changes. Hence, 'RAC' doesn't occur.

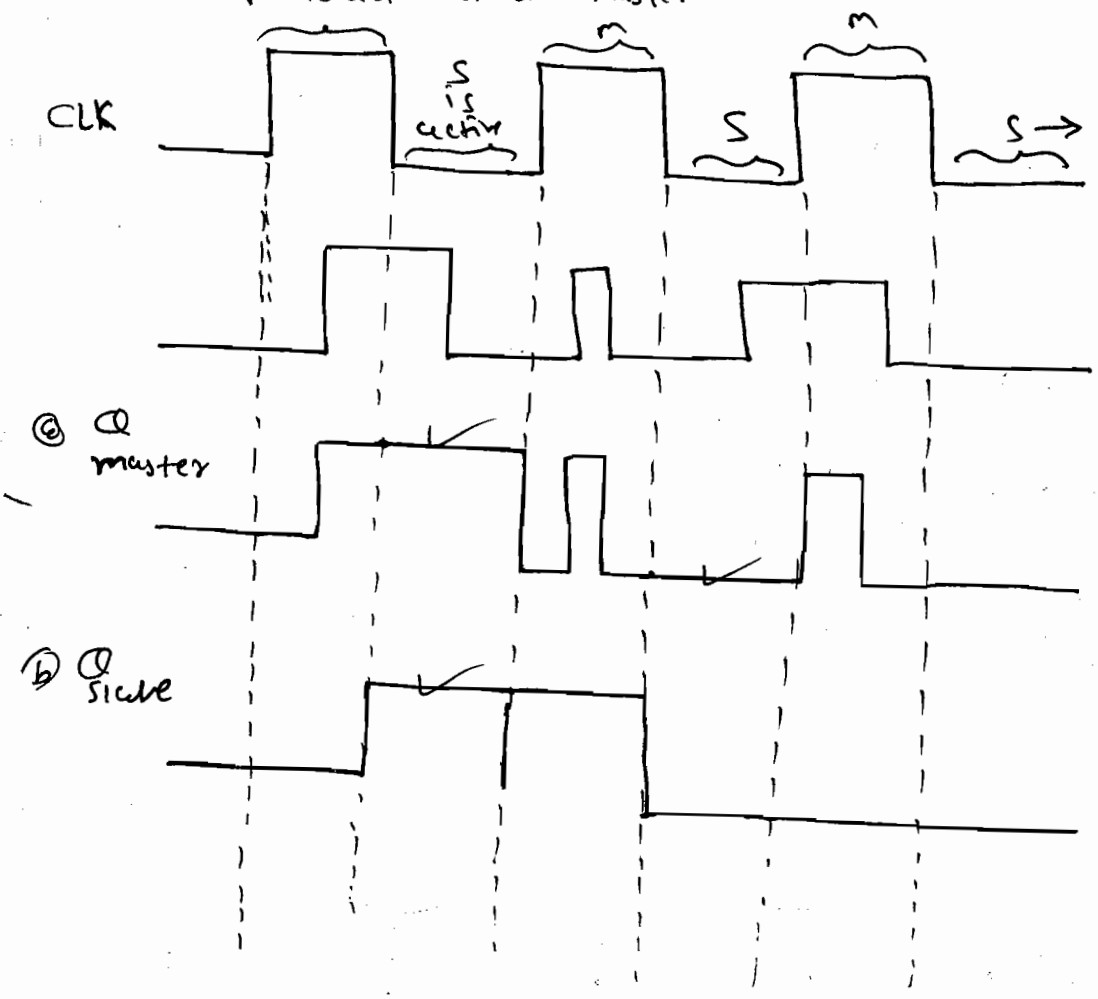



→ In master slave J-k flip flop the feedback values Q and $\bar{Q}$ do not change during the clock pulse because they are taken from inactive slave FF. Hence 'RAC' doesn't occured.

→ Master slave JK FF output is similar to the negative edge triggered JK flip flop output.

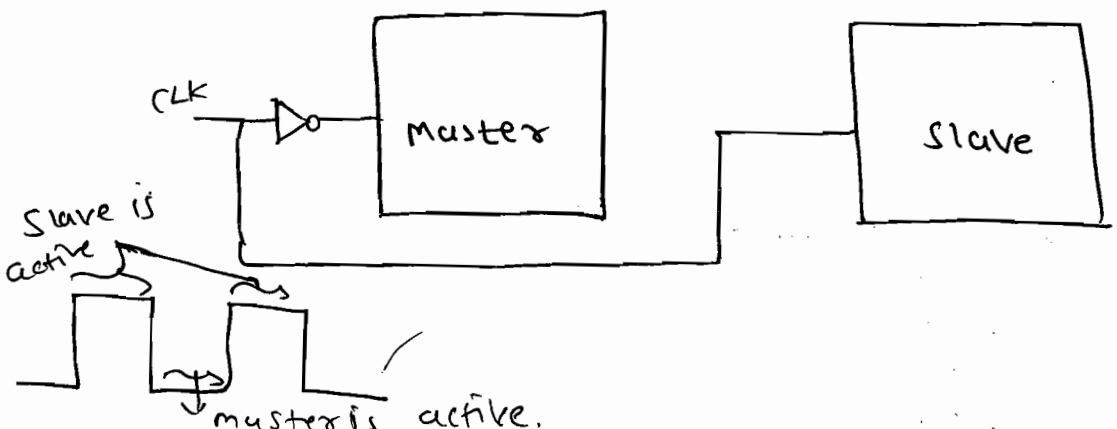
Ex-1 In a master slave D flip flop draw the o/p of master and slave if the clock and D inputs are as given below.

→ M is active and $Q_{master} = D$




→ Slave o/p is similar to -ve Edge Triggered FF o/p.

NOTE: Master slave JK FF as JKFF as positive Edge triggered FF.



→ In slave we can use J-K flip flop instead of S-R flip flop. But is more costly than S-R.

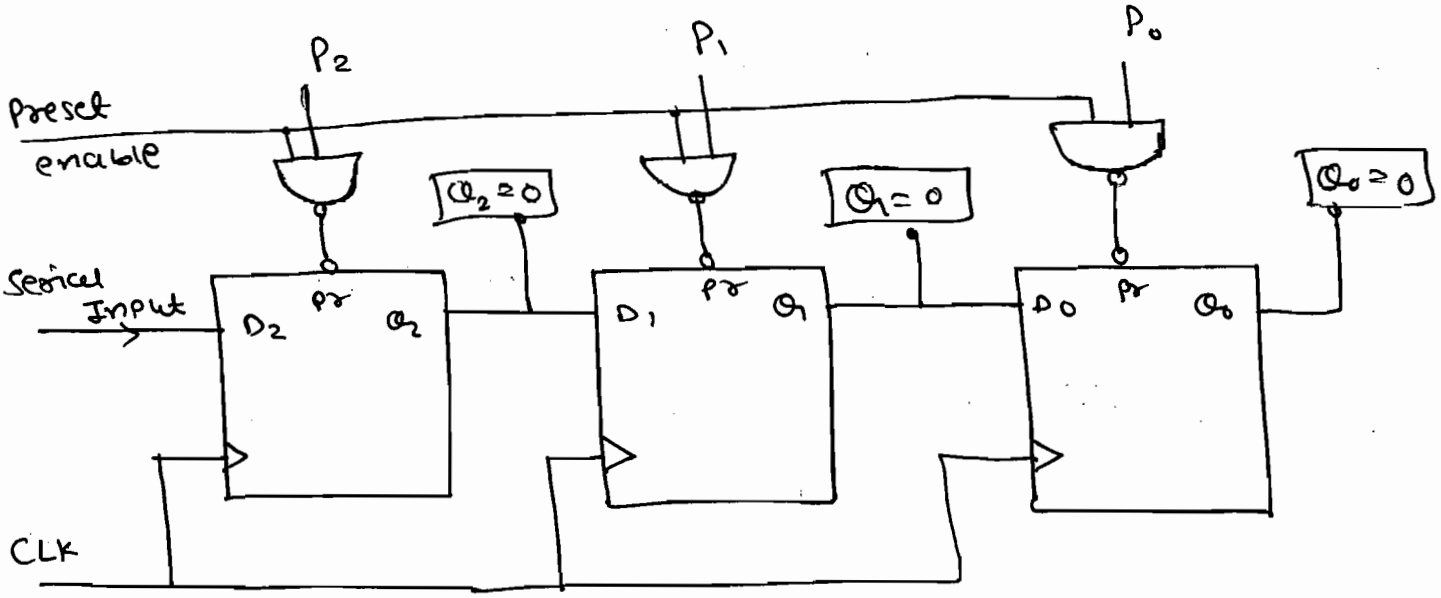
① Shift Register → "Sequential memory".

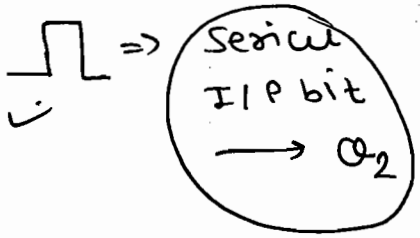
② Counters

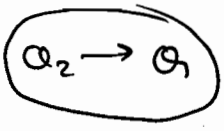
ⓐ to count the no. of pulses.

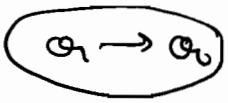
ⓑ Freqency divider.

## * 3-Bit Shift Register:

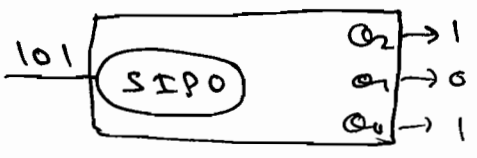



① Serial in parallel out =



| CLK | Serial Input | $Q_2$ | $Q_1$ | $Q_0$ |
|---|---|---|---|---|
| 0 | – | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 1 | 1 | 0 | 1 |

MSB LSB
101

LSB

serial

parallel

* 'N' Bit shift Register.

'N' clock pulses

time = N·T secs.

## ② Serial in serial out :

101 → SISO → 101.

| CLK | Serial Input | $Q_2$ | $Q_1$ | $Q_0$ |
|---|---|---|---|---|
| 0 | – | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 1 | 1 | 0 | 1 |
| 4 | – | – | 1 | 0 |
| 5 | – | – | – | 1 |

LSB

Serial Input

serial o/p.

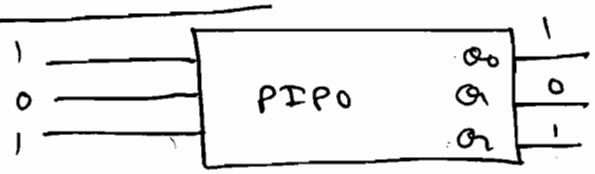
$N + (N-1) = (2N-1)$ CLK periods.

time = $(2N-1)T$.

N = No. of FFs.

## ③ Parallel in parallel out:

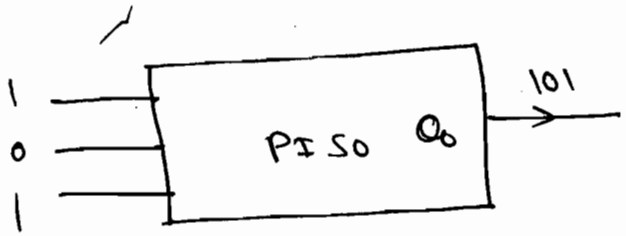
$P_2 P_1 P_0$ = Parallel input




→ No clock pulse is required.

→ Preset is asynchronous input therefore no clock pulse is required.

## ④ Parallel in Serial out:




⇒ **Time = (N-1) T**

Input

| $P_2$ | $P_1$ | $P_0$ |
|---|---|---|
| 1 | 0 | 1 |

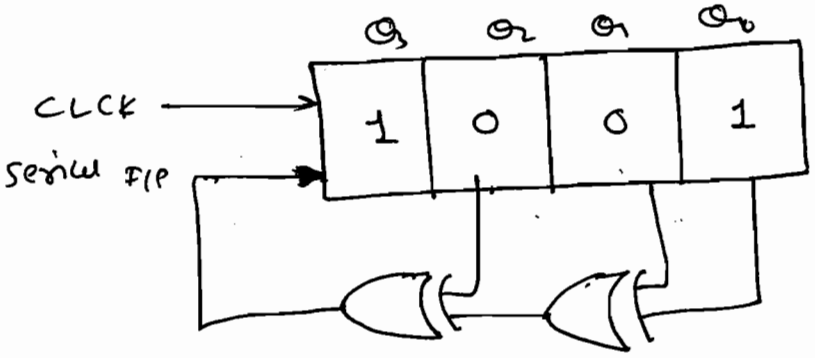
$Q_2 = 1$ $Q_1 = 0$ $Q_0 = \phi$

| $Q_2$ | $Q_1$ | $Q_0$ |
|---|---|---|
| 1 | 0 | 1 |

after 1 CLK.
$Q_2 = Q_1 = 1$, $Q_0 = 0$
after 2 CLK
$Q_1 = -$, $Q_0 = 1$

NOTE: In Parallel in serial out operation preset enable is disabled after obtaining the input data at $Q_2, Q_1, Q_0$.

Ex-1 In following shift register determine no. of CLK pulses required to bring the shift register to the initial value of 1001.



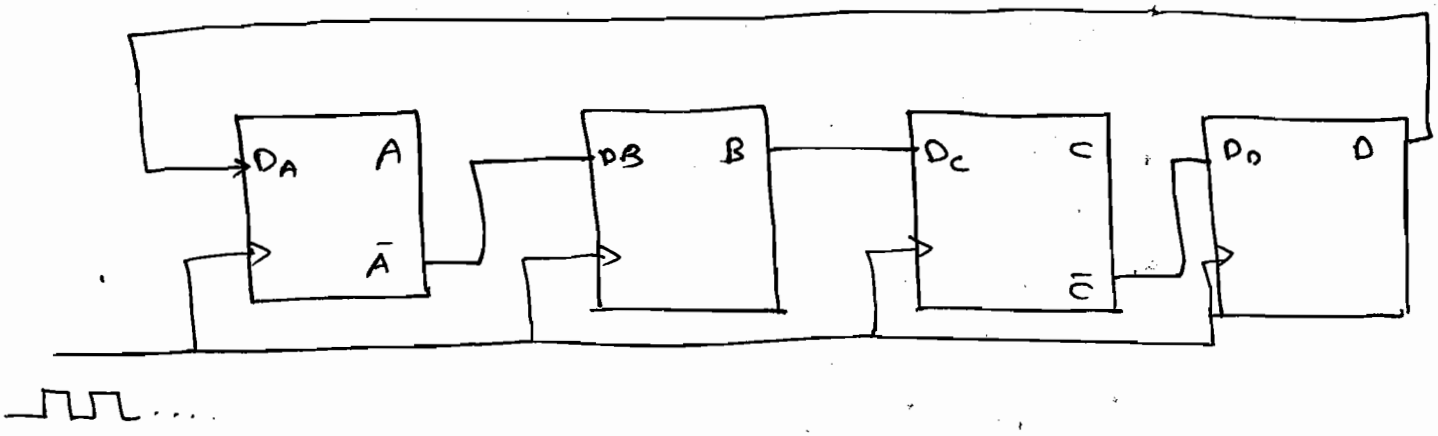

Sr. Input = $Q_2 \oplus Q_1 \oplus Q_0$.

| CLK | Serial Input $Q_2 \oplus Q_1 \oplus Q_0$ | $Q_3$ $Q_2$ $Q_1$ $Q_0$ |
|---|---|---|
| | | 1 0 0 1 → 1 |
| 0 | – | |
| 1 | 1 | 1 1 0 0 → 1 |
| 2 | 1 | 1 1 1 0 → 0 |
| 3 | 0 | 0 1 1 1 → 1 |
| 4 | 1 | 1 0 1 1 → 0 |
| 5 | 0 | 0 1 0 1 → 0 |
| 6 | 0 | 0 0 1 0 → 1 |
| 7 | 1 | 1 0 0 1 |

So, 7 CLOCK pulse is required.

Ex-2

Initial Value: 0000



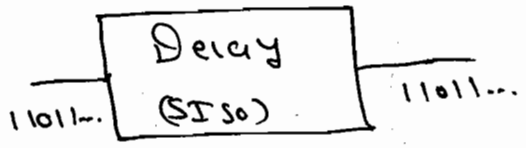

| Clock Pulse | A = D | B = $\bar{A}$ | C = B | D = $\bar{C}$ |
|---|---|---|---|---|
| – | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 2 | 1 | 1 | 1 | 1 |
| 3 | 1 | 0 | 1 | 0 |
| 4 | 0 | 0 | 0 | 0 |

So, After 4 Clock Pulse.

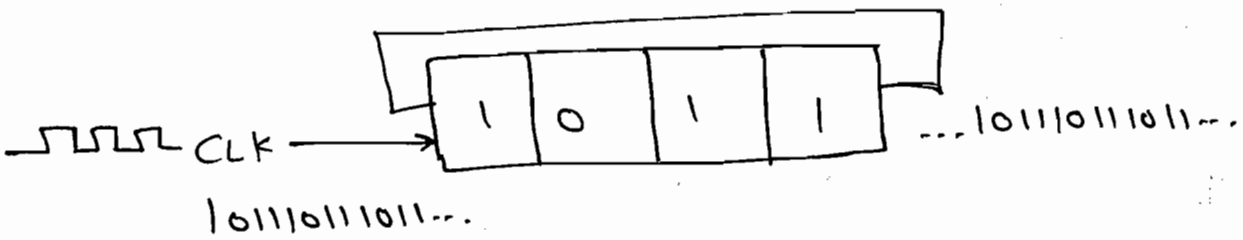
## * Shift Register Application:

① Serial to parallel | parallel to serial data converter.

② Time delay




③ Seaence Crenerator:




1011101110 11...

④ To generate PN (Pseudo Number) seaence.

⑤ Counter
- ⓐ Ring Counter
- ⓑ Johnson Counter.

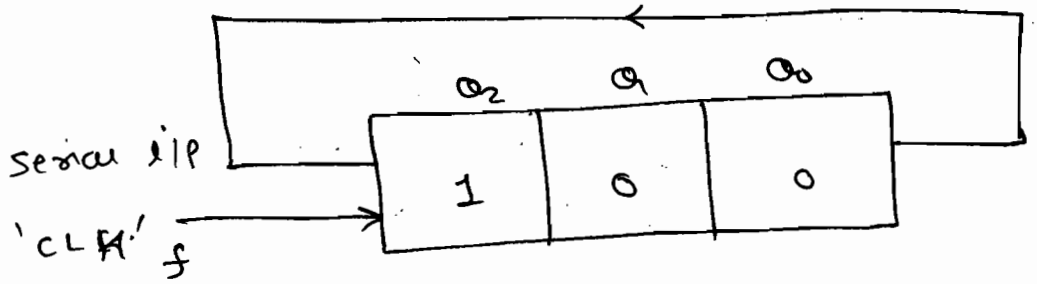
NOTE:

→ Shift Register is converted to ring counter by making two changes.

① $Q_0$ is connected to the serial input.

② one of the FF is reset.

# * 3 - Bit Ring Counter.




3:1 Counter → Can Count 3 CLK Pulses.

| CLK | serial i/p | $Q_2$ $Q_1$ $Q_0$ |
|---|---|---|
| 0 | — | 1 0 0 |
| 1 | 0 | 0 1 0 |
| 2 | 0 | 0 0 1 |
| 3 | 1 | 1 0 0 |

→ Counting:

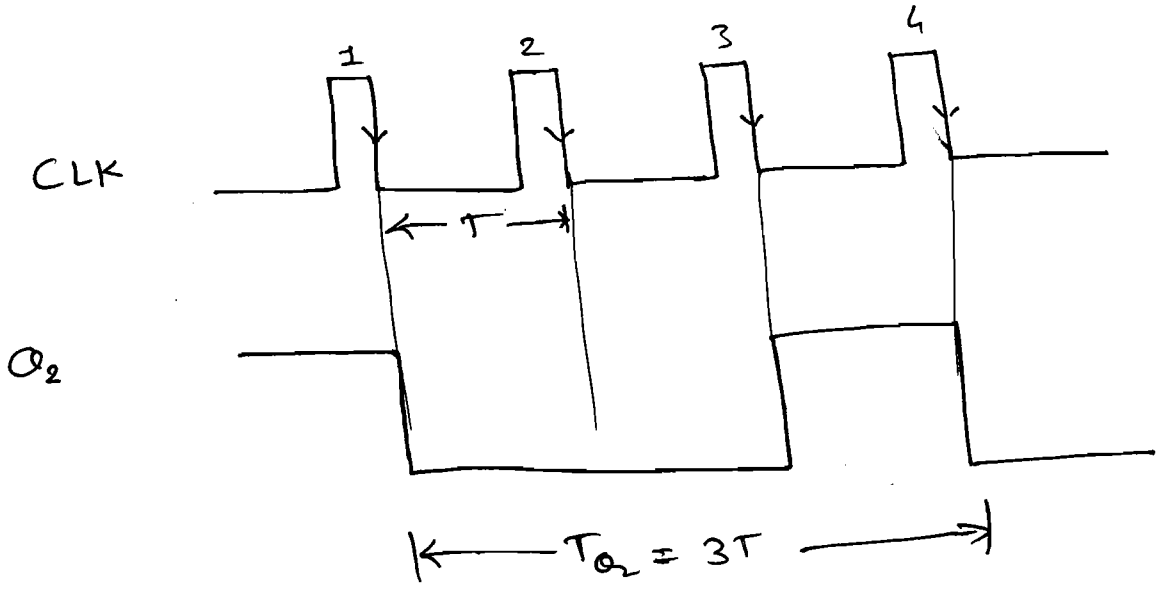
100 ← initial value

(a) 001 → Decode → 2 CLOCK Pulse.

(b) 100 → Decode → 3 CLOCK Pulse.

(c) 010 → Decode → 1 CLOCK Pulse.

⇒ Freq division:




$\therefore \quad T_{Q_2} = 3T$

$\therefore \quad \boxed{f_{Q_2} = f/3}$

similarly, $\boxed{f_{Q_1} = f/3}$

$\boxed{f_{Q_0} = f/3}$

⇒ N-bit Ring counter:

→ N:1 Counter

→ Can Count N CLOCK PULSE.

→ Freq. of each FF $O/P = f/N$.