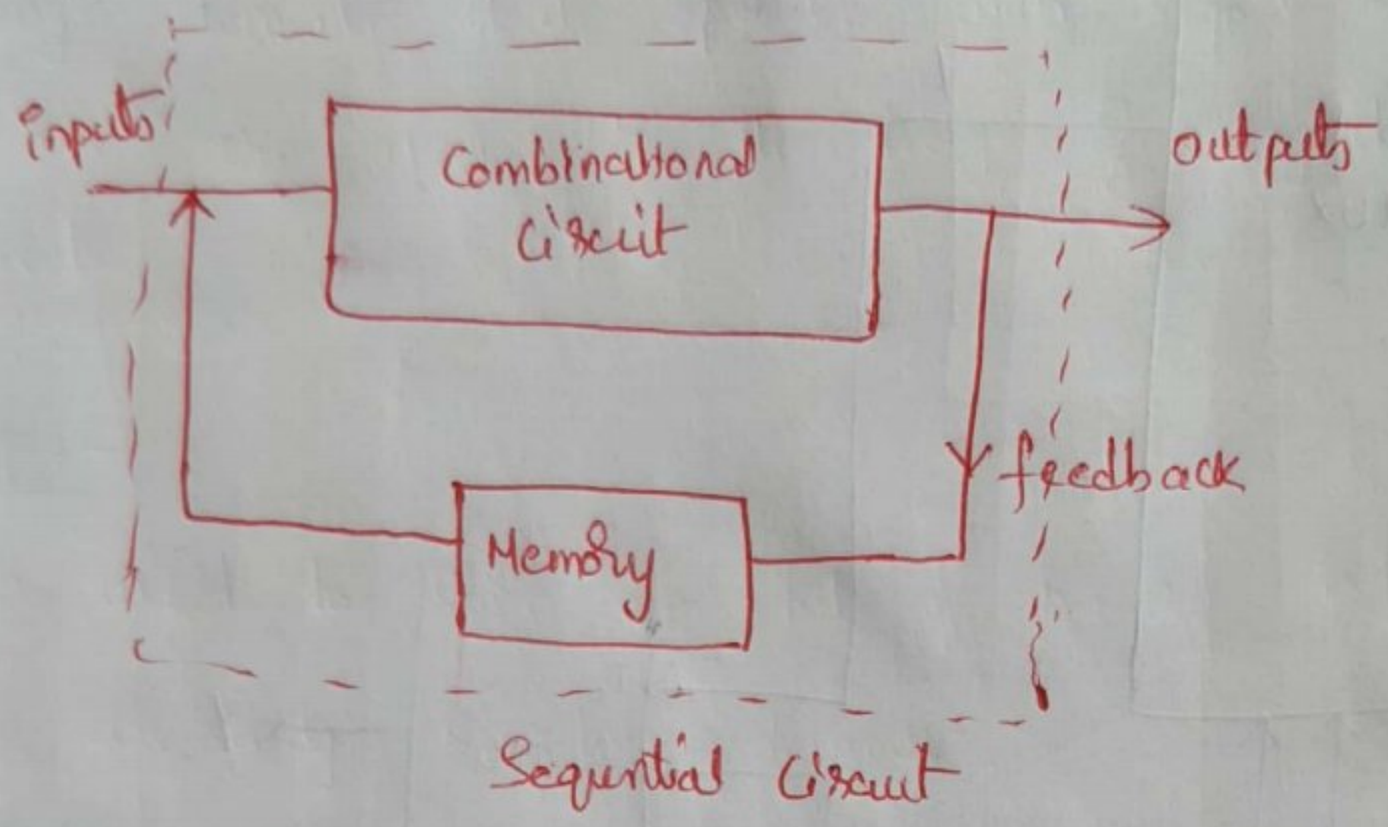
# UNIT-III SEQUENTIAL CIRCUITS

→ Sequential circuit is a combinational circuit with memory.

→ The output of sequential circuit depends upon present inputs and present state (Past outputs)




→ The information stored in sequential circuit represents present state.

→ The present state and present inputs will define output and next state.

## Types of Sequential Circuit:-

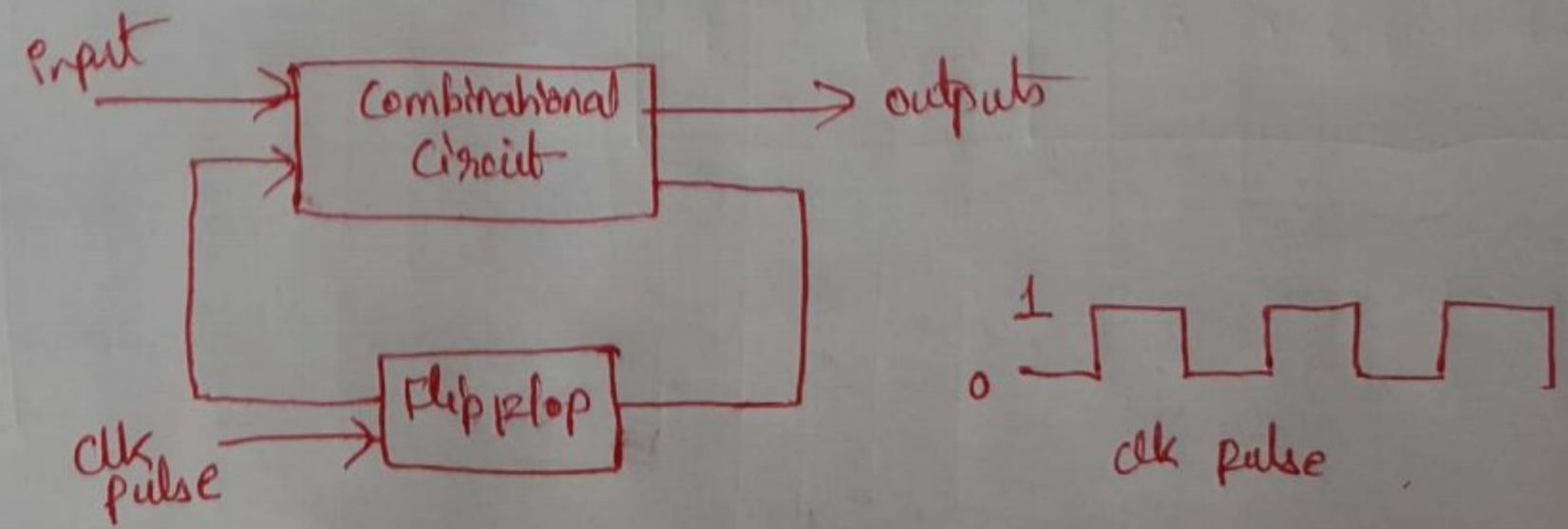
① Synchronous Sequential Circuit
② Asynchronous Sequential Circuit

① **Synchronous Sequential Circuit** :- It is a system whose behavior can be defined from the knowledge of its at desrete instant of time. It runs with same clock pulse.

→ Memory elements → Flip Flops.

→ Easy to Design.

→ operating speed depends on clock.




Scanned with CamScanner

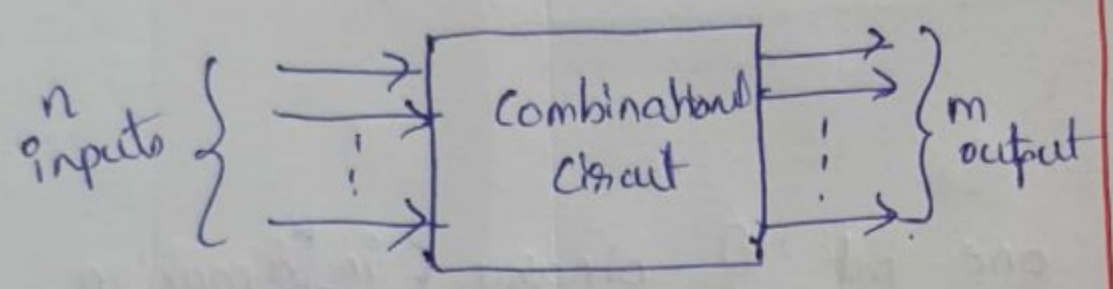
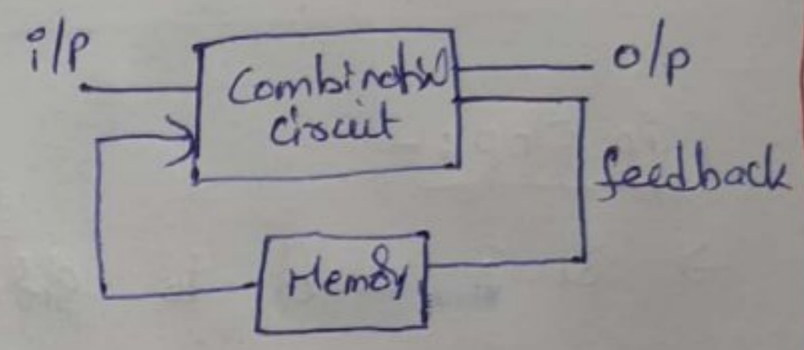
Scanned with CamScanner

Asynchronous Sequential Circuit :- It depends upon the input signals at any instant of time and the order in which the input changes.

→ The memory element used in asynchronous sequential circuit is time delay devices and Latches.

→ Faster than Synchronous → Difficult to design.

Difference between Combinational circuit and Sequential Circuit.

| | Combinational circuit | Sequential Circuit |
|---|---|---|
| 1 |   |   |
| 2 | In this the output is discovered by present state & the inputs | Here both the present input state and past output are used to identify the output. |
| 3 | It doesn't store any data | It can store a small amount of data |
| 4 | These circuit do not rely on the clock | These circuit rely on clk for triggering function. |
| 5 | It doesn't require any feedback | Here feedback is required |
| 6 | Examples :- Adders, Decoders, encoders, Multiplexers. | Example :- Latches, FlipFlops. |

Scanned with CamScanner

Scanned with CamScanner

## LATCH :-

→ It is used to store one bit of binary information.
→ These are level sensitive devices.
→ storage elements that operate with signal levels.
→ Latches are used for the design of asynchronous sequential circuits

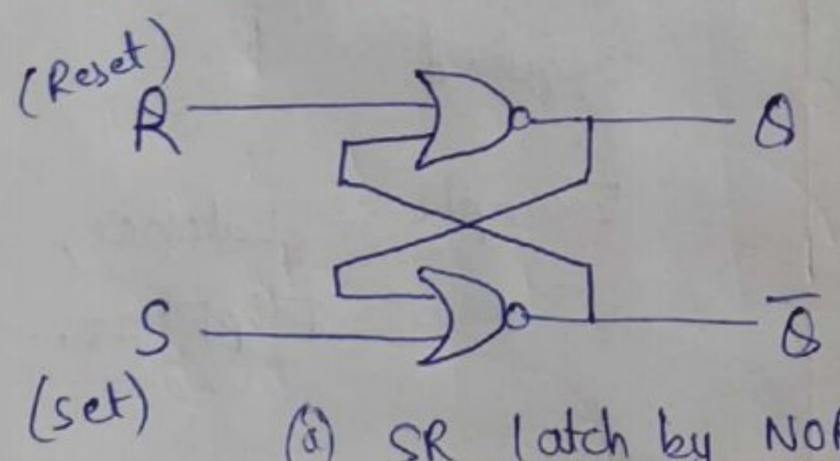
## Flip Flop :-

→ It is used to store one bit of binary information
→ These are edge sensitive devices.
→ storage elements are controlled by clock transition are called flip flops.
→ These are used in synchronous sequential circuit

## SR Latch :-

These are unclocked flip flops. It is a circuit with two cross coupled NOR gates or two NAND gates

### logic circuit

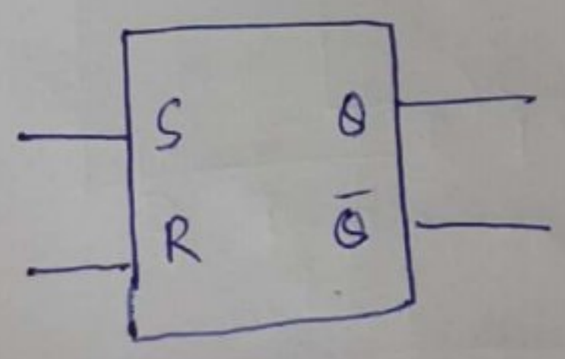



(i) SR latch by NOR gates

### Truth Table.

| S | R | Q | Q' | |
|---|---|---|---|---|
| 1 | 0 | 1 | 0 | (SET) |
| 0 | 0 | 1 | 0 | (No chang) |
| 0 | 1 | 0 | 1 | (RESET) |
| 1 | 1 | 0 | 0 | (forbidden) |

### logic Symbol :-



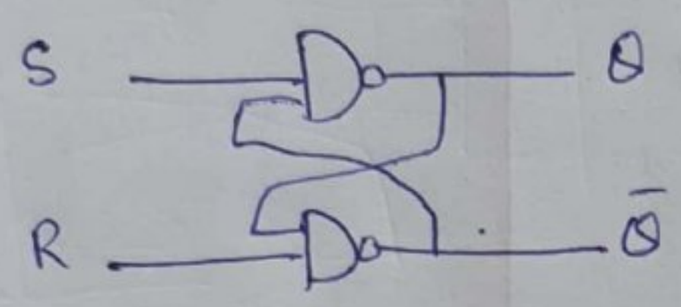

Scanned with CamScanner
Scanned with CamScanner

→ When $Q = 1$, $Q' = 0$ it is said to be SET state, when $Q = 0$, $Q' = 1$ it is reset state.

→ The o/p's $Q$ and $\overline{Q}$ are complement to each other.

→ When both the inputs are 1, 1 then both the outputs will be in Invalid state i.e 0, 0.

→ If both the inputs are switched to zero, simultaneously the device will enter in undefined state.




(b) SR latch by NAND gates.

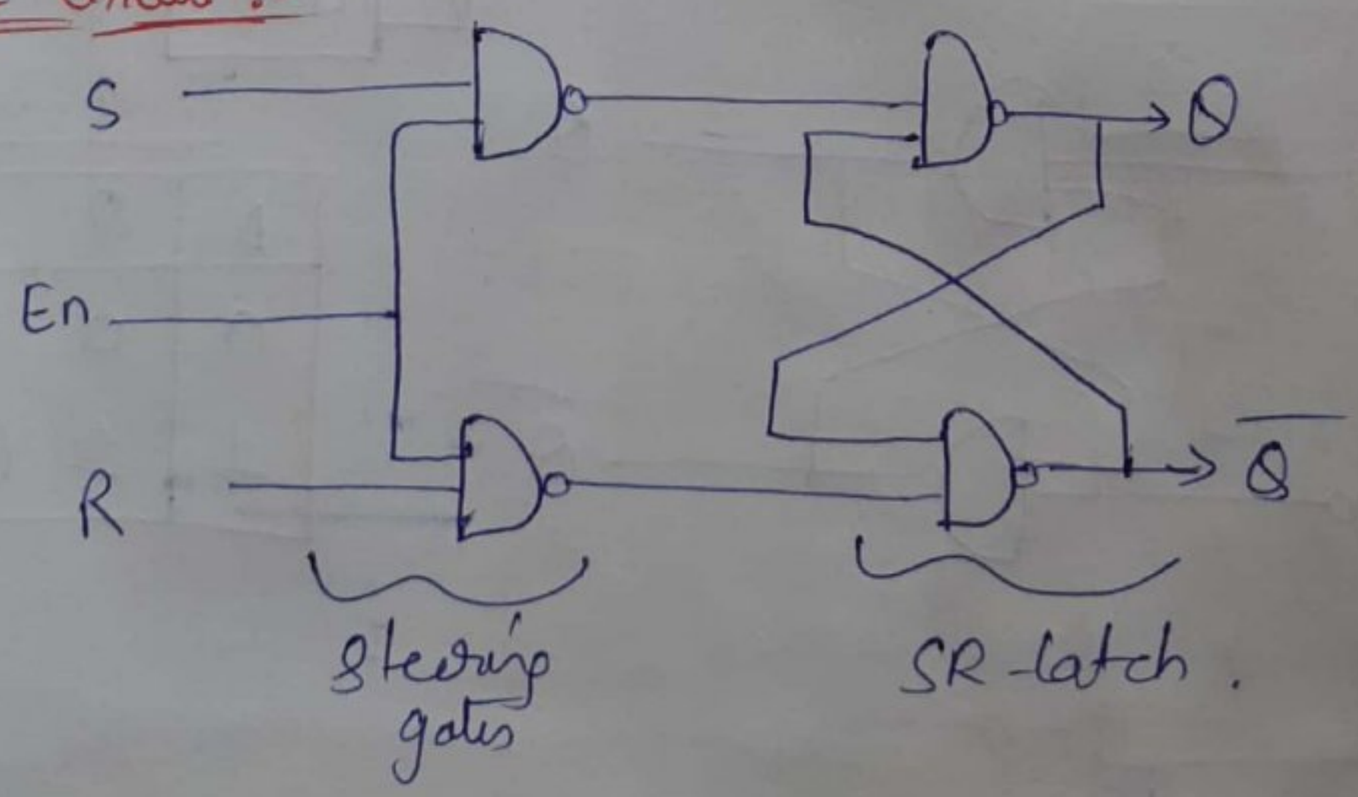
| S | R | Q | $\overline{Q}$ | |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | (Indeterminant) |
| 0 | 1 | 0 | 1 | (Reset) |
| 1 | 0 | 1 | 0 | (Set) |
| 1 | 1 | 0 | 1 | (No change) |

## SR - Latch :-

→ The basic SR latch can be modified by providing an additional input signal that determines (controls) when the state of the latch can be changed.

→ An SR latch with control input is as shown below.

**logic Circuit :-**




**logic Symbol**

Scanned with CamScanner

Scanned with CamScanner

→ Here steering gates are used to drive input to the SR - latch.

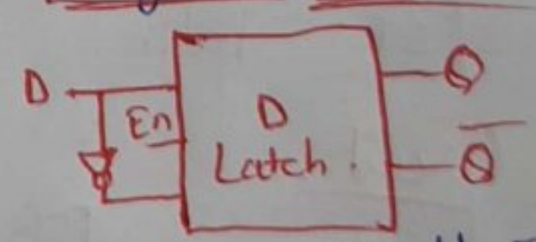
→ When En = 1, then the inputs S, R are allowed to affect the latch.

Truth Table.

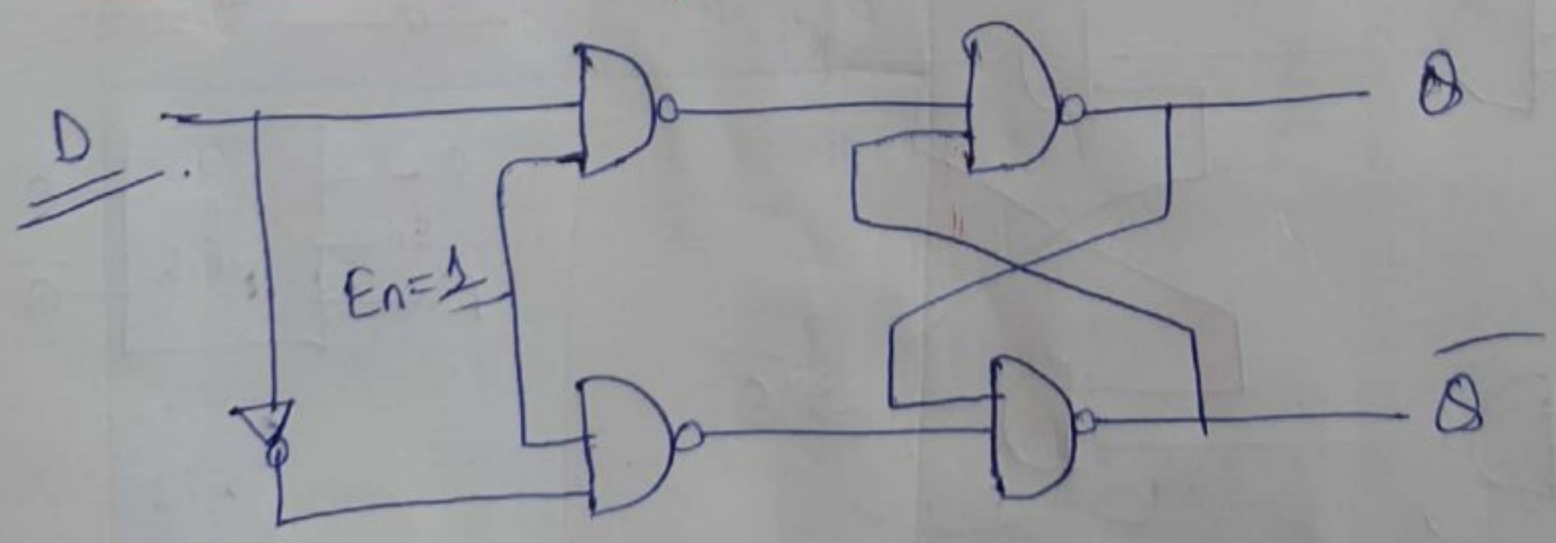
| En | S | R | Q | $\overline{Q}$ | |
|---|---|---|---|---|---|
| 0 | X | X | No change. | | |
| 1 | 0 | 0 | 0 | 1 | (No change) |
| 1 | 0 | 1 | 0 | 1 | (Reset) |
| 1 | 1 | 0 | 1 | 0 | (Set) |
| 1 | 1 | 1 | 1 | 1 | (Indeterminate State) |

D Latch :- It is a Transparent Latch. D — Data. It is also called as Data latch. Here whatever input we apply the same will be the output. The logic circuit is similar to SR latch except two inputs we connect both the inputs with an inverter in between.

logic Symbol




logic Circuit




Truth Table

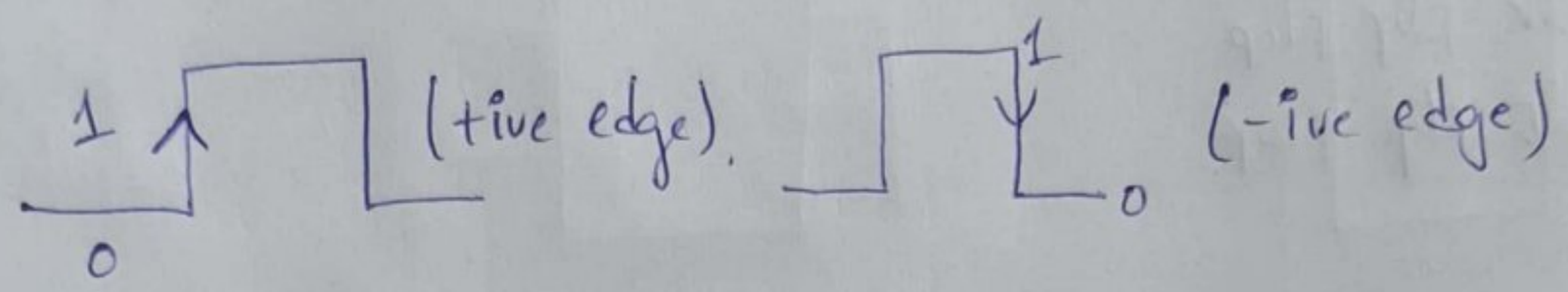
| D | Q | |
|---|---|---|
| 0 | 0 | (Reset) |
| 1 | 1 | (Set) |

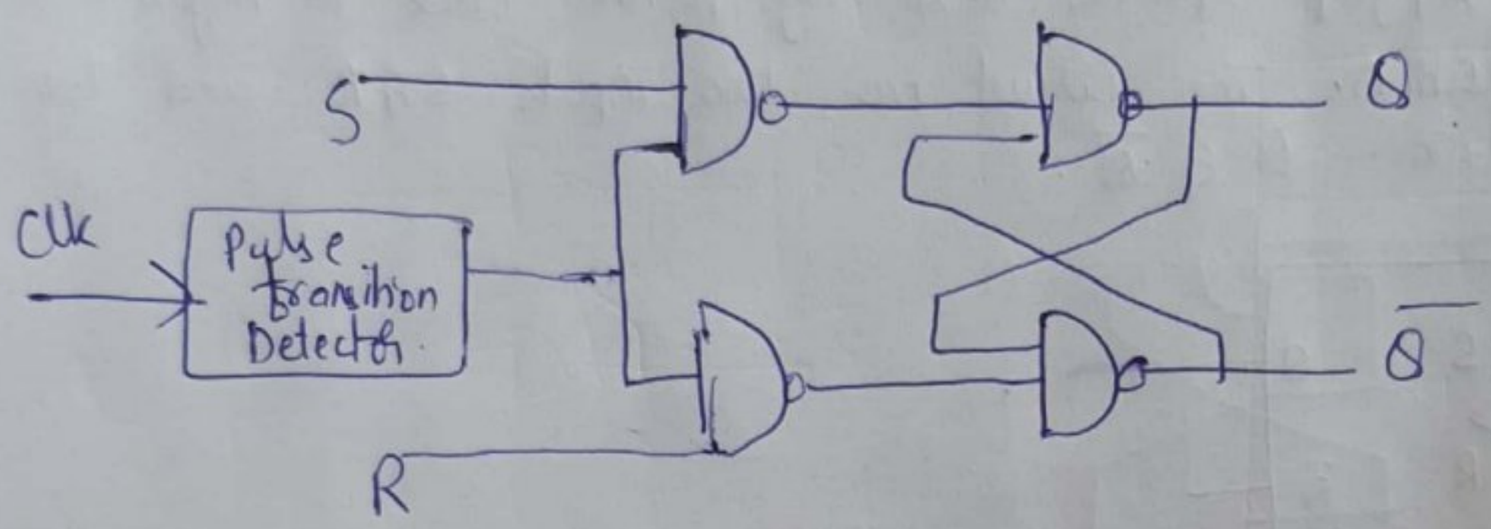
Scanned with CamScanner

Scanned with CamScanner

## Edge triggered flip flop :-

→ clocked flipflops may be either a +ive edge triggered & -ive edge triggered.

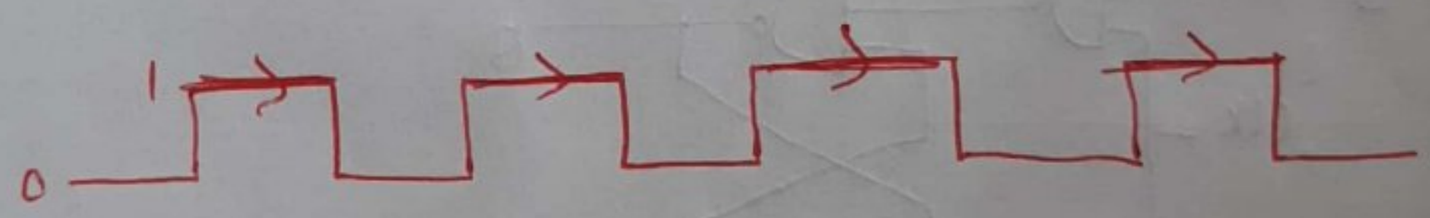


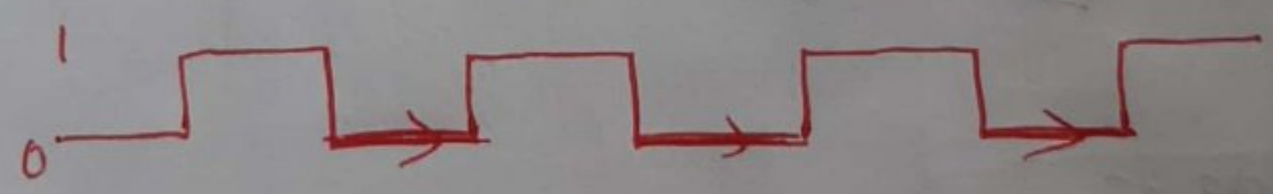

## Types of Triggering :-

① **Level trigger :-**

+ive level

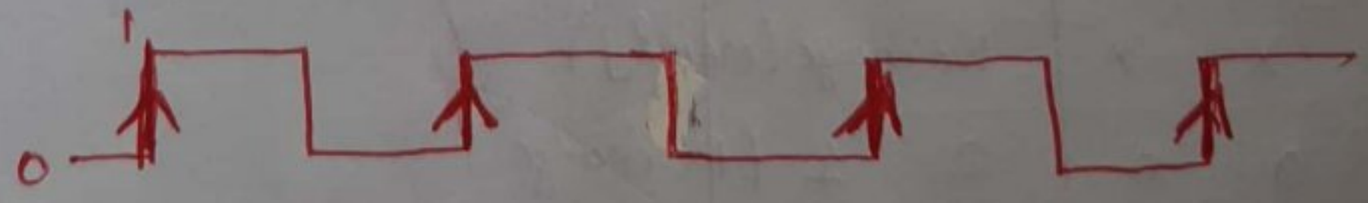



-ive level

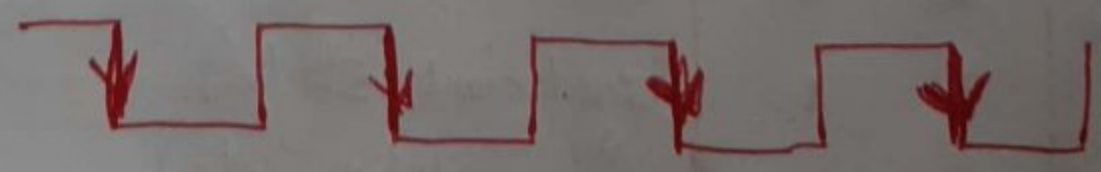



② **Edge trigger**

+ive edge




-ive edge.



Scanned with CamScanner

Scanned with CamScanner

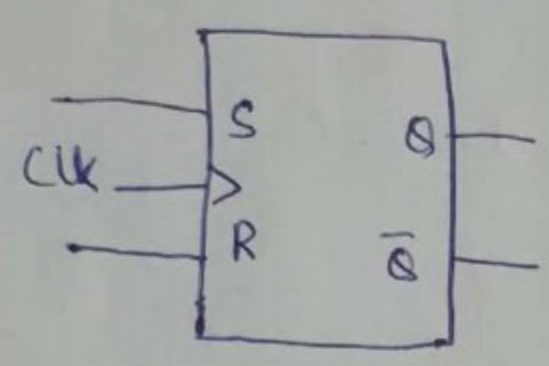
## Flip Flops :-

→ There are Four FlipFlops :-

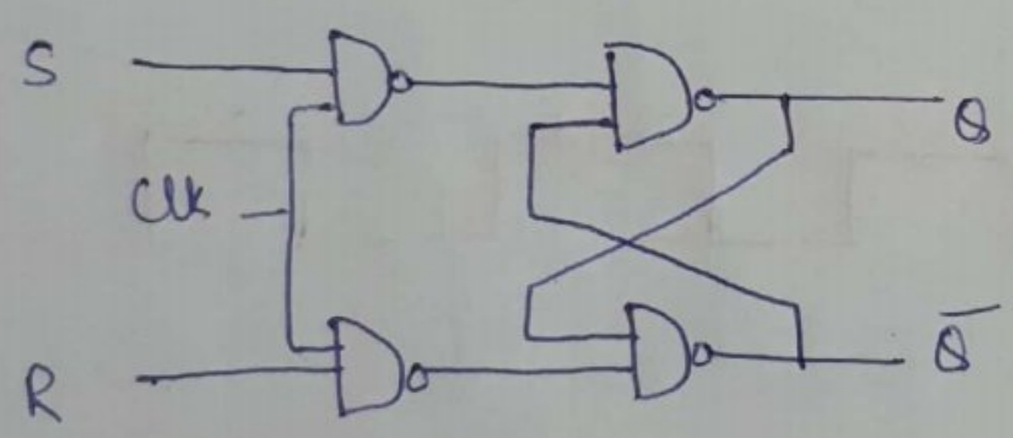
1. SR Flip Flop
2. D Flip Flop
3. JK Flip Flop
4. T Flip Flop

**SR Flip Flop :-** SR flipflop operates with only positive clock & negative clock transition. The circuit has two inputs S & R and two outputs Q & $\overline{Q}$.

**Logic Symbol :-**




**logic Circuit :-**




**Truth Table :-**

| CLK | S | R | Q | $\overline{Q}$ | |
|---|---|---|---|---|---|
| 0 | X | X | No change (memory) | | |
| 1 | 0 | 0 | 0 | 1 | (No change) |
| 1 | 0 | 1 | 0 | 1 | (Reset) |
| 1 | 1 | 0 | 1 | 0 | (Set) |
| 1 | 1 | 1 | 1 | 1 | (Indeterminant state & Invalid state) |

Scanned with CamScanner

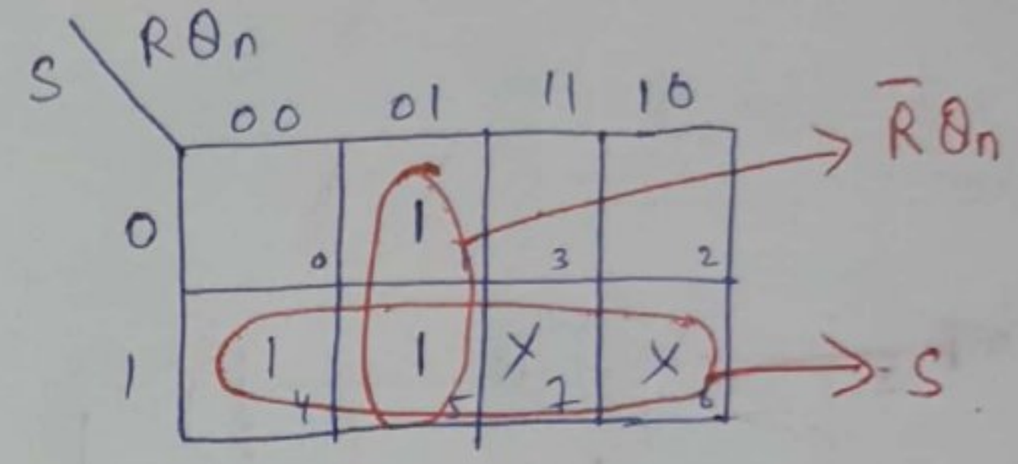
Scanned with CamScanner

## Characteristic Table :-

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

Here $Q_n$ = Present State
$Q_{n+1}$ = Next State

## Characteristic Equation



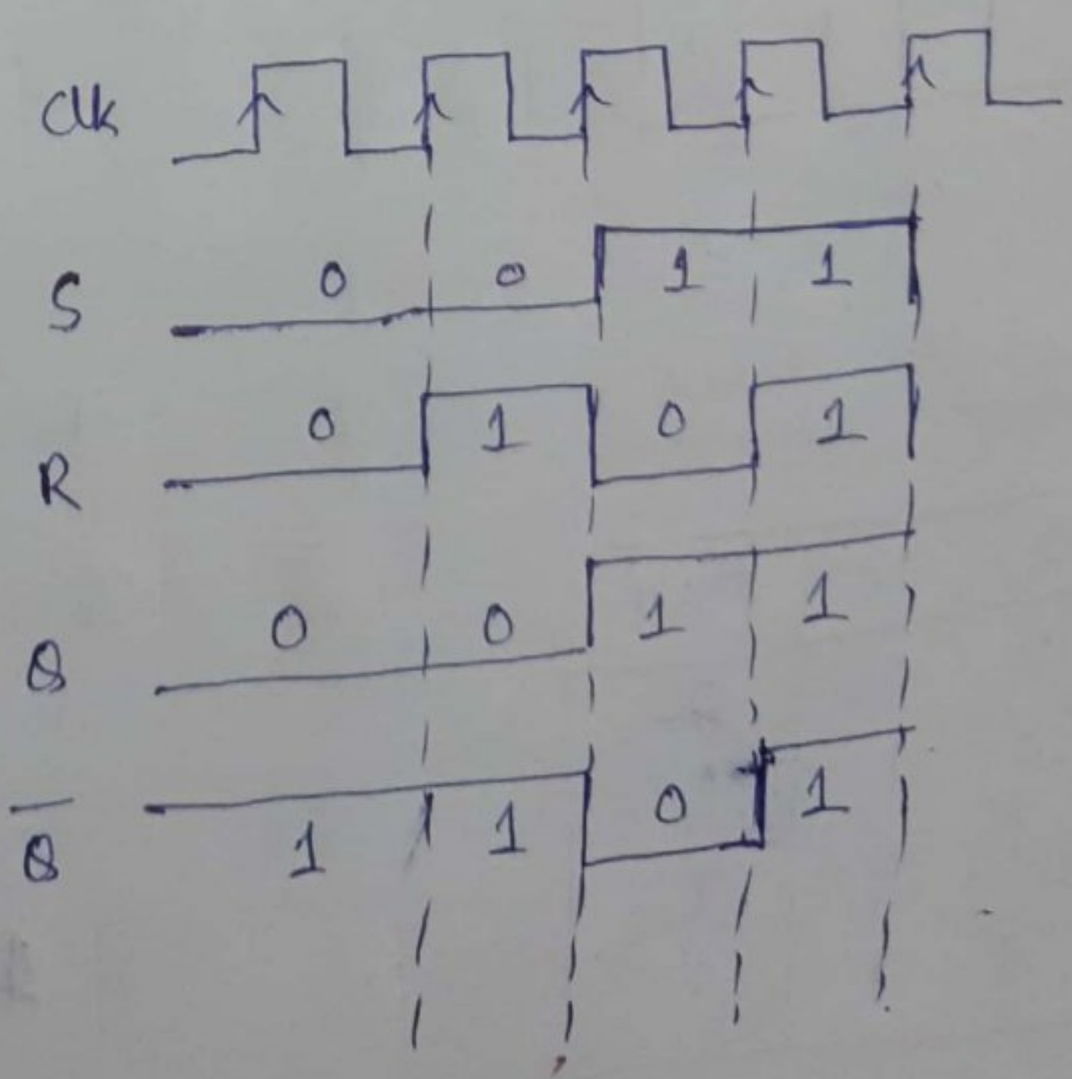

$$Q_{n+1} \Rightarrow S + \bar{R} Q_n$$

## Excitation Table :-

It takes the present state $Q_n$, Next state $Q_{n+1}$ as the inputs and S, R as outputs.

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

## Waveforms :-

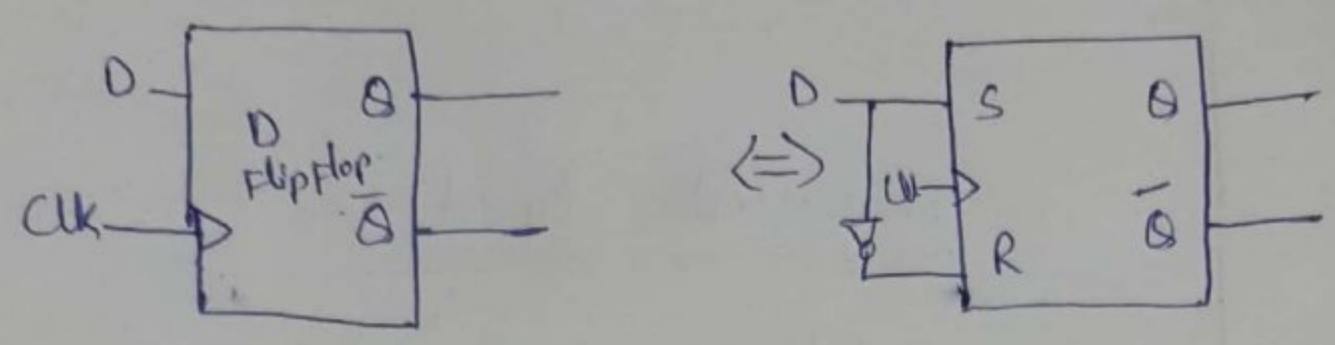



Scanned with CamScanner

Scanned with CamScanner

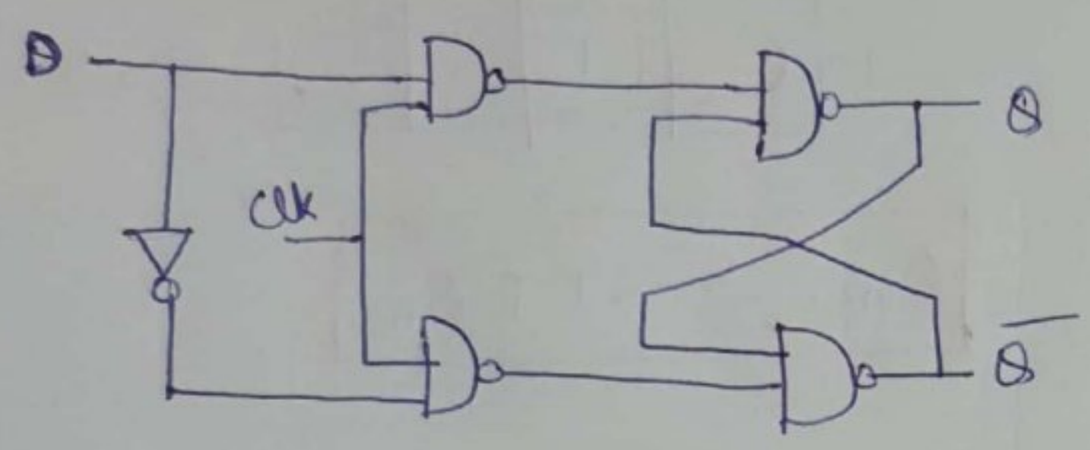
**D Flip Flop :-** D flip flop operates with only positive clock transition & negative clock transition. The output of D flip flop is insensitive to the changes in input D except at active transition of clock.

**logic Symbol :-**




→ It has one input D & two outputs Q & $\overline{Q}$

**logic circuit :-**



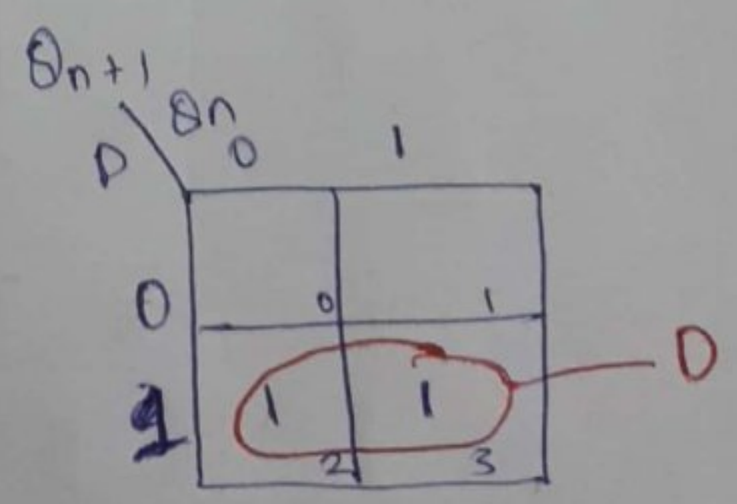

**Truth Table :-**

| clk | D | Q | $\overline{Q}$ |
|---|---|---|---|
| 0 | X | N.C | NC |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

**characteristic Table :-**

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

Characteristic equation.




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

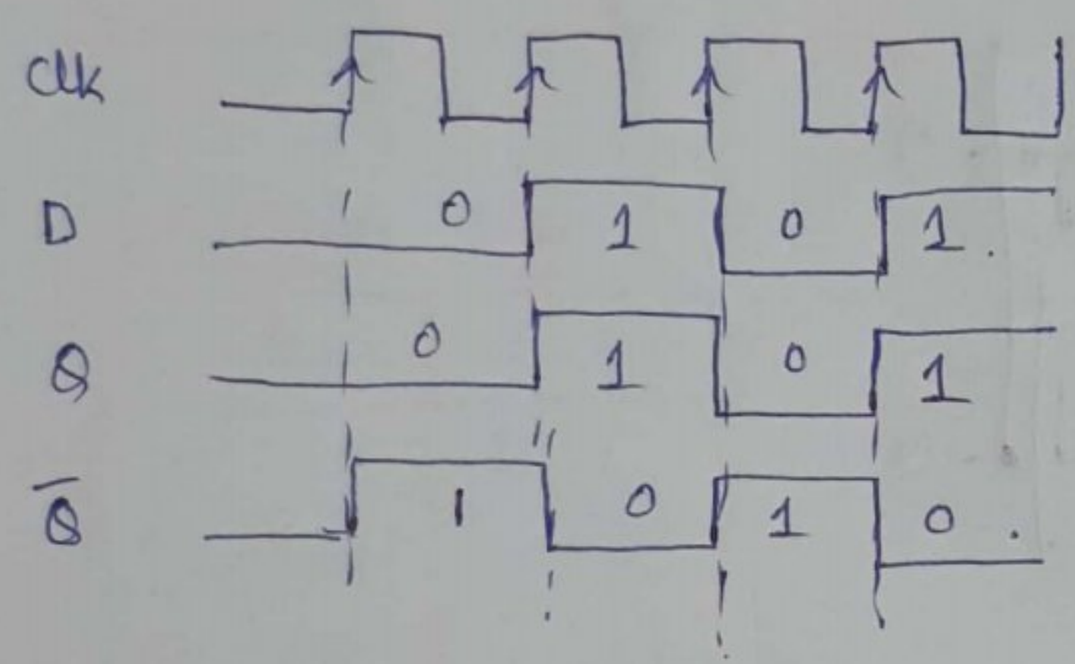
Scanned with CamScanner

Scanned with CamScanner

## Excitation table :-

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

## Waveforms :-

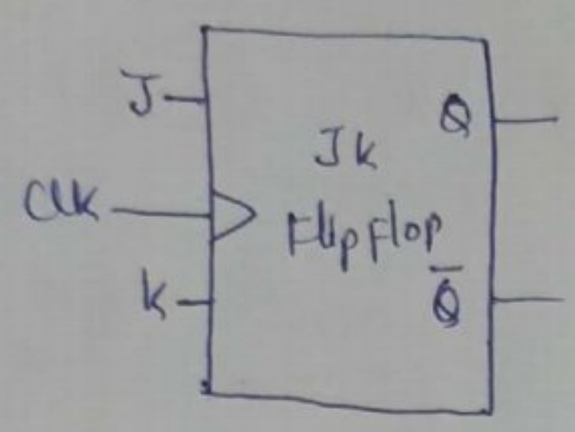



Scanned with CamScanner

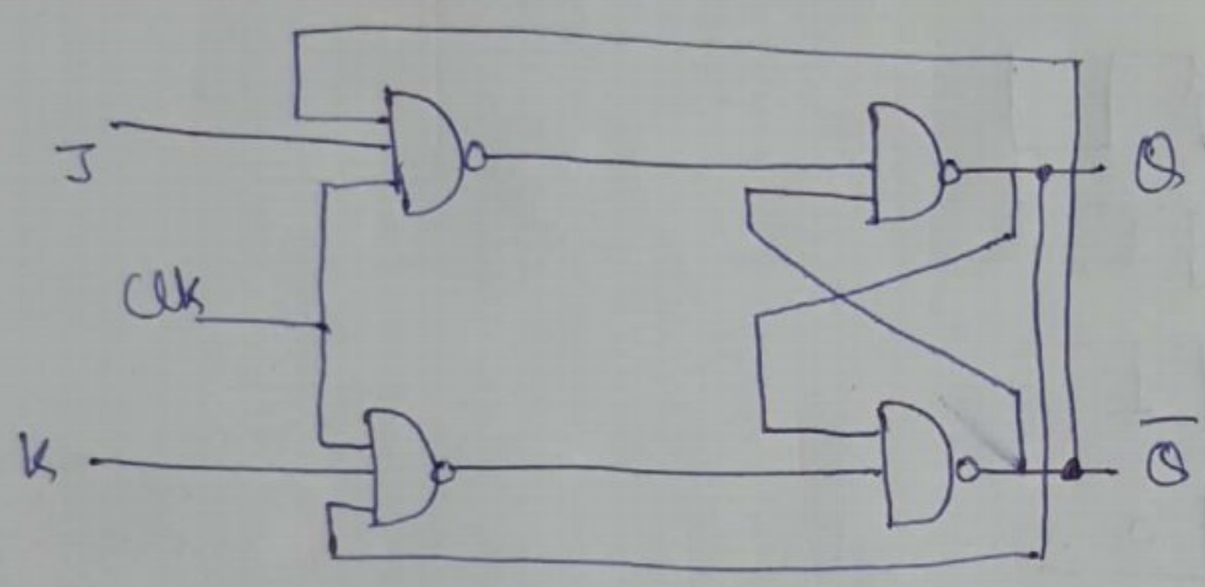
Scanned with CamScanner

**JK Flip Flop :-** It is the modified Version of SR flip flop. It operates only with positive clock & negative clock transition. The Circuit has two inputs J & K, two outputs Q and $\overline{Q}$.

logic Symbol :-




**logic Circuit**




**Truth Table.**

| Clk | J | K | Q | $\overline{Q}$ | |
|---|---|---|---|---|---|
| 0 | x | x | memory (No change) | | |
| 1 | 0 | 0 | 0 | 1 | (memory) |
| 1 | 0 | 1 | 0 | 1 | (Reset) |
| 1 | 1 | 0 | 1 | 0 | (Set) |
| 1 | 1 | 1 | Toggle | | |

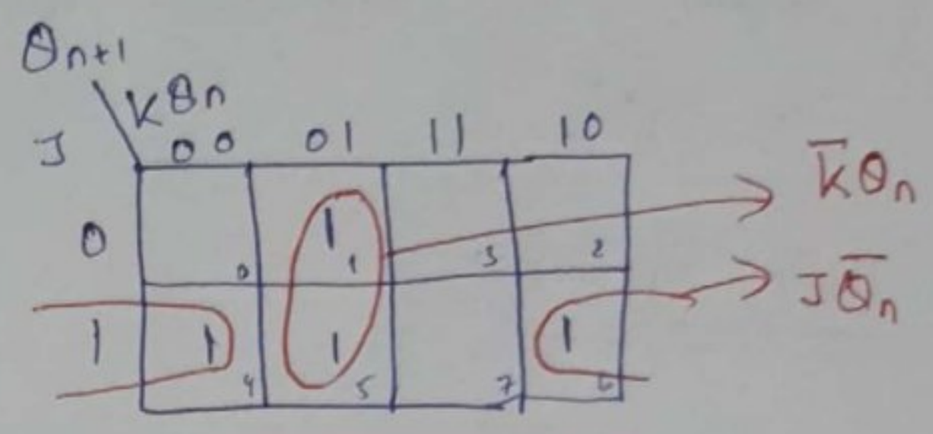
Scanned with CamScanner

Scanned with CamScanner

## Characteristic Table.

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

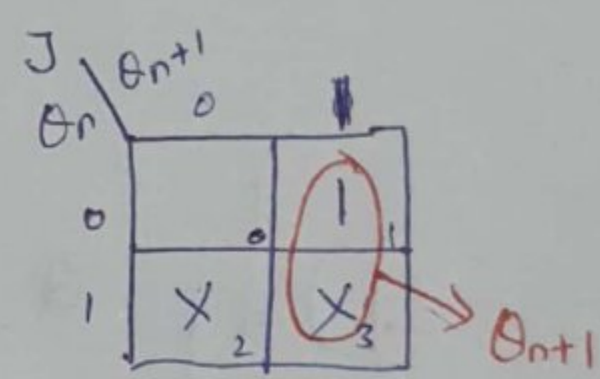
## Characteristic equation.

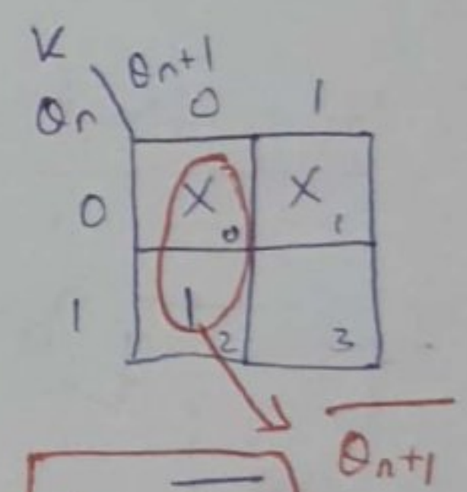



$$Q_{n+1} \Rightarrow J\bar{Q}_n + \bar{K}Q_n$$

## 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 |

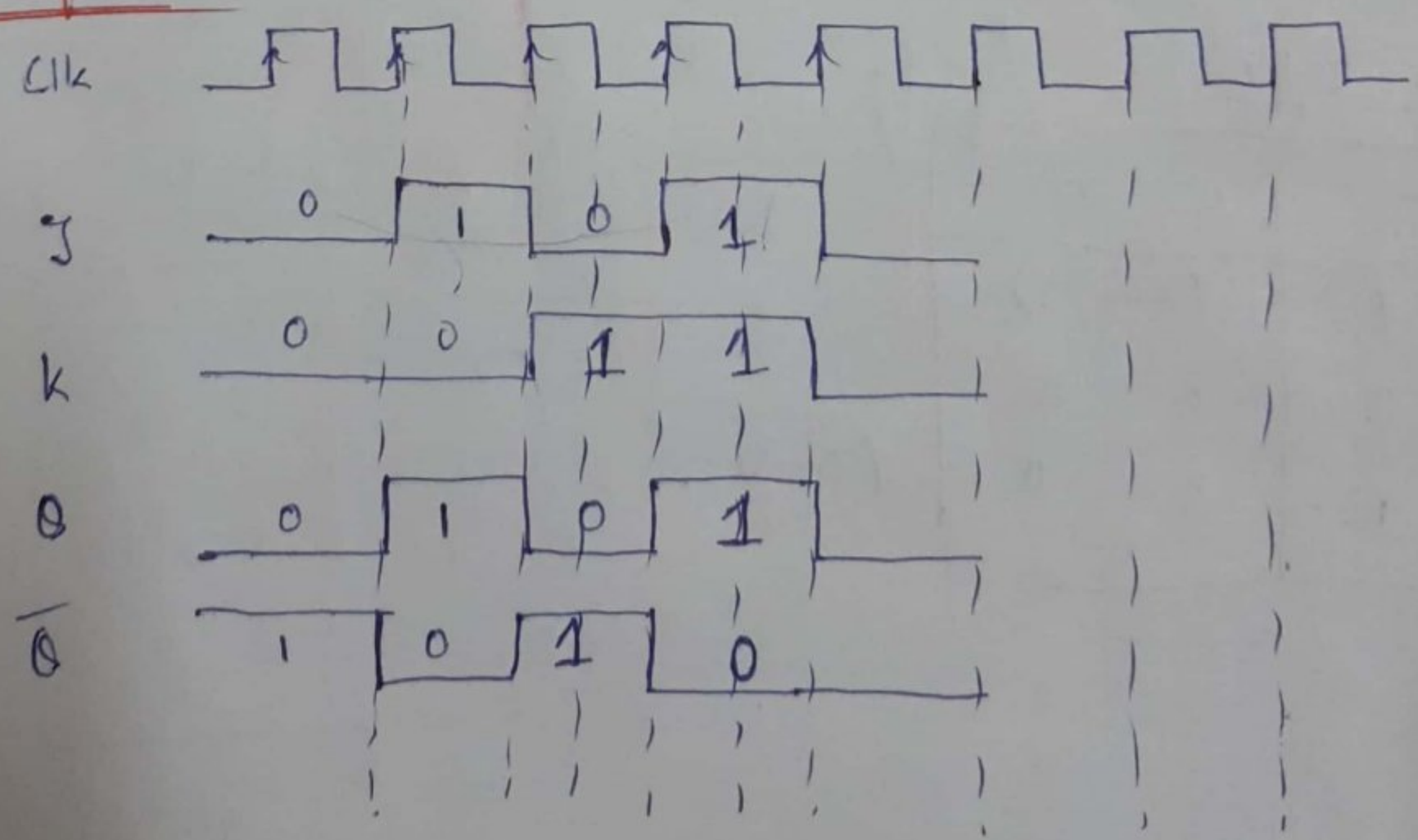



$$J = Q_{n+1}$$

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

## Waveform



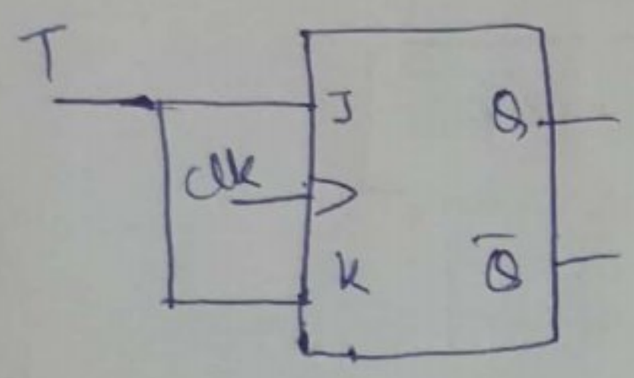

Scanned with CamScanner

Scanned with CamScanner

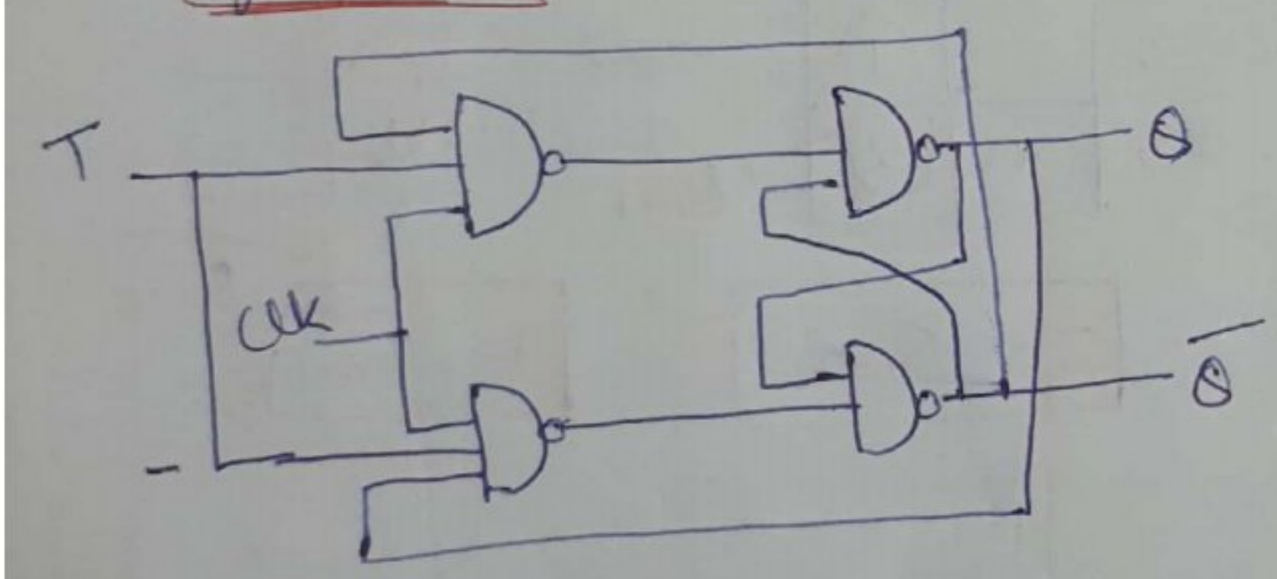
**T-Flip Flop:-** T flip flop is an edge triggered device. i.e its low to high or high to low transition on input clock signal will cause a change in output state of flipflop.

→ T flip flop is constructed from a JK flip flop. Here J, K inputs are connected together and a single input T is given as input.

**logic Symbol:-**




**logic Circuit:-**




**Truth Table**

| Clk | T | Q | $\overline{Q}$ |
|---|---|---|---|
| 0 | x | (memory) | |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 → (Toggle). |

Scanned with CamScanner

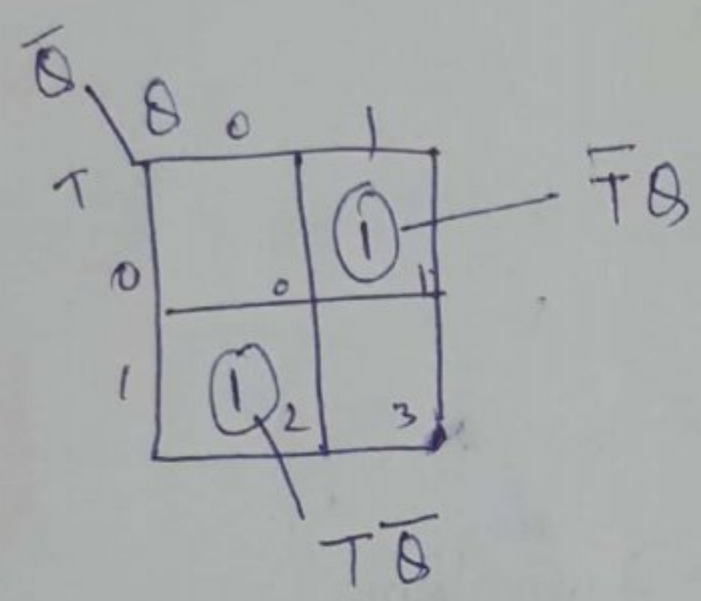
Scanned with CamScanner

Characteristic Table.

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

} Toggle).

## Characteristic Equation:-



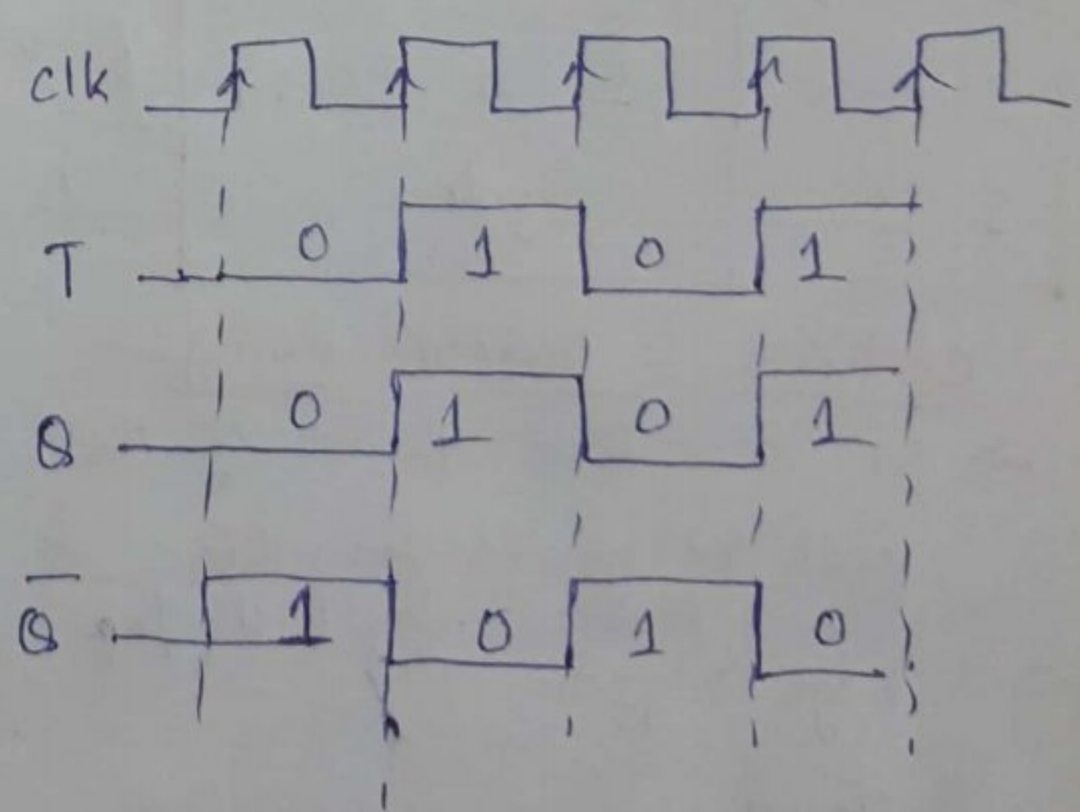

$$Q_{n+1} = T\overline{Q} + \overline{T}Q$$

$$\Rightarrow T \oplus Q$$

## Excitation Table:-

| Q | $\overline{Q}$ | T |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

## Waveforms.



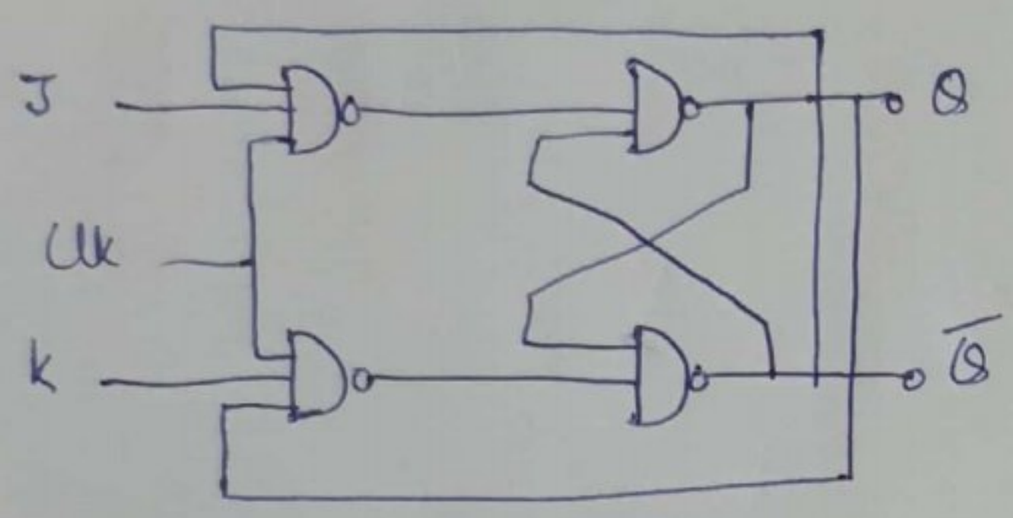

Scanned with CamScanner

Scanned with CamScanner

## Race Around Condition :-

→ For J-K flip flop; if J = K = 1 & if clock is too long then the state of flip flop keep on Toggle which leads to uncertainity in determining output state of flipflop. This problem is called Race around condition.

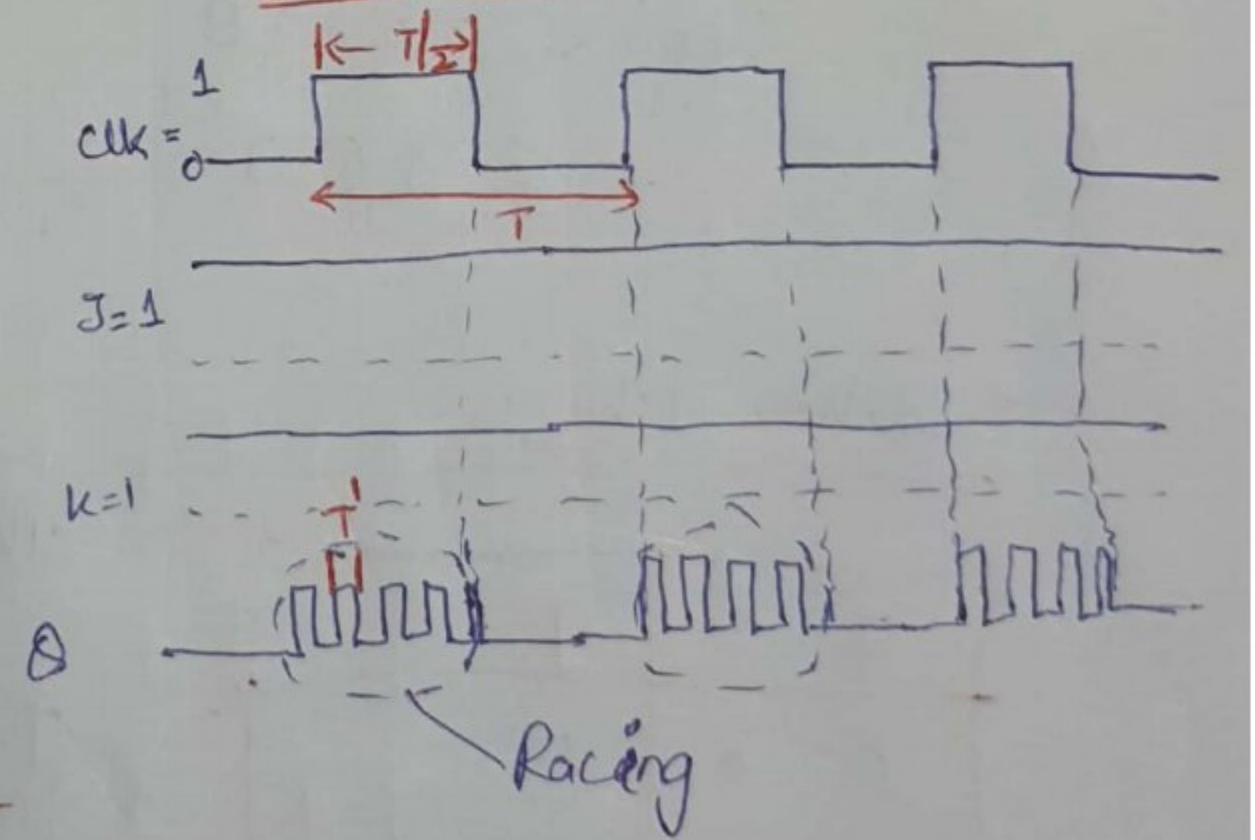
## J K flip flop :-




## Truth Table.

| Clk | J | K | Q | $\overline{Q}$ |
|---|---|---|---|---|
| 0 | X | X | memory | |
| 1 | 0 | 0 | 0 | 1 (memory) |
| 1 | 0 | 1 | 0 | 1 (Reset) |
| 1 | 1 | 0 | 1 | 0 (SET) |
| 1 | 1 | 1 | Toggle | |

## Waveform.




## Conditions to Overcome Racing :-

→ Let the Time period of clock is T; the half time period must be greater than delay of flip flop. i.e ($T/2$ > delay)

→ $T/2$ < produced delay of flip flop ($T' > T/2$).

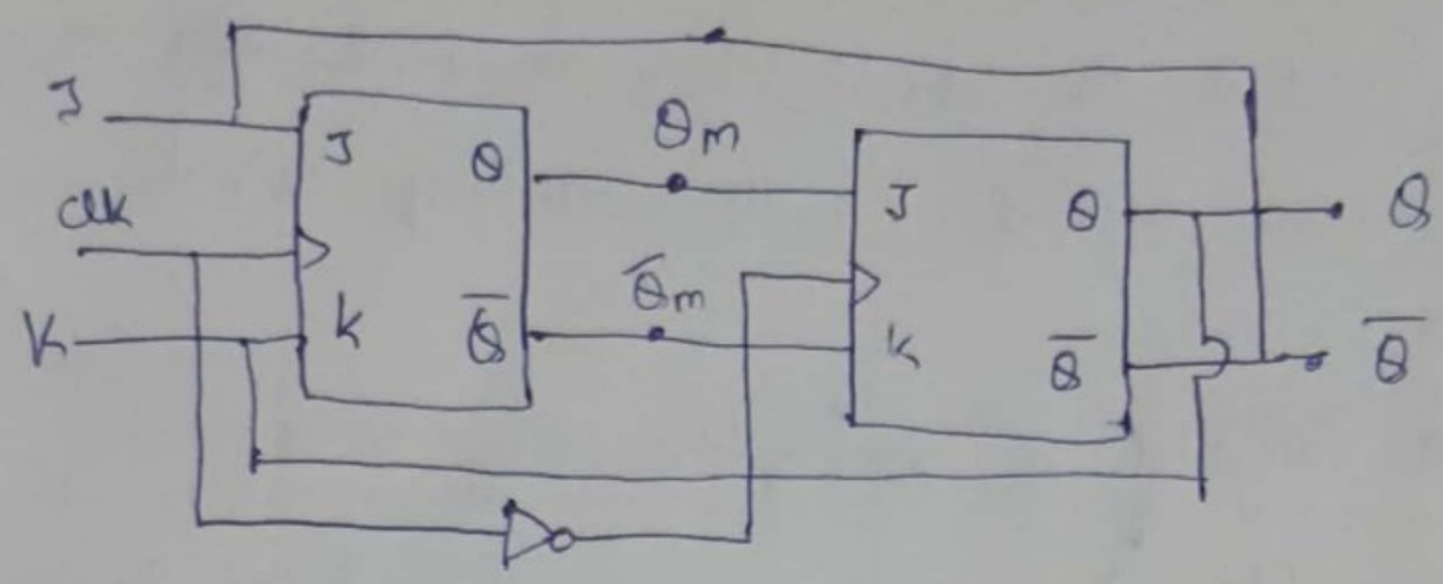
→ Edge triggering

→ Master - Slave Configuration.

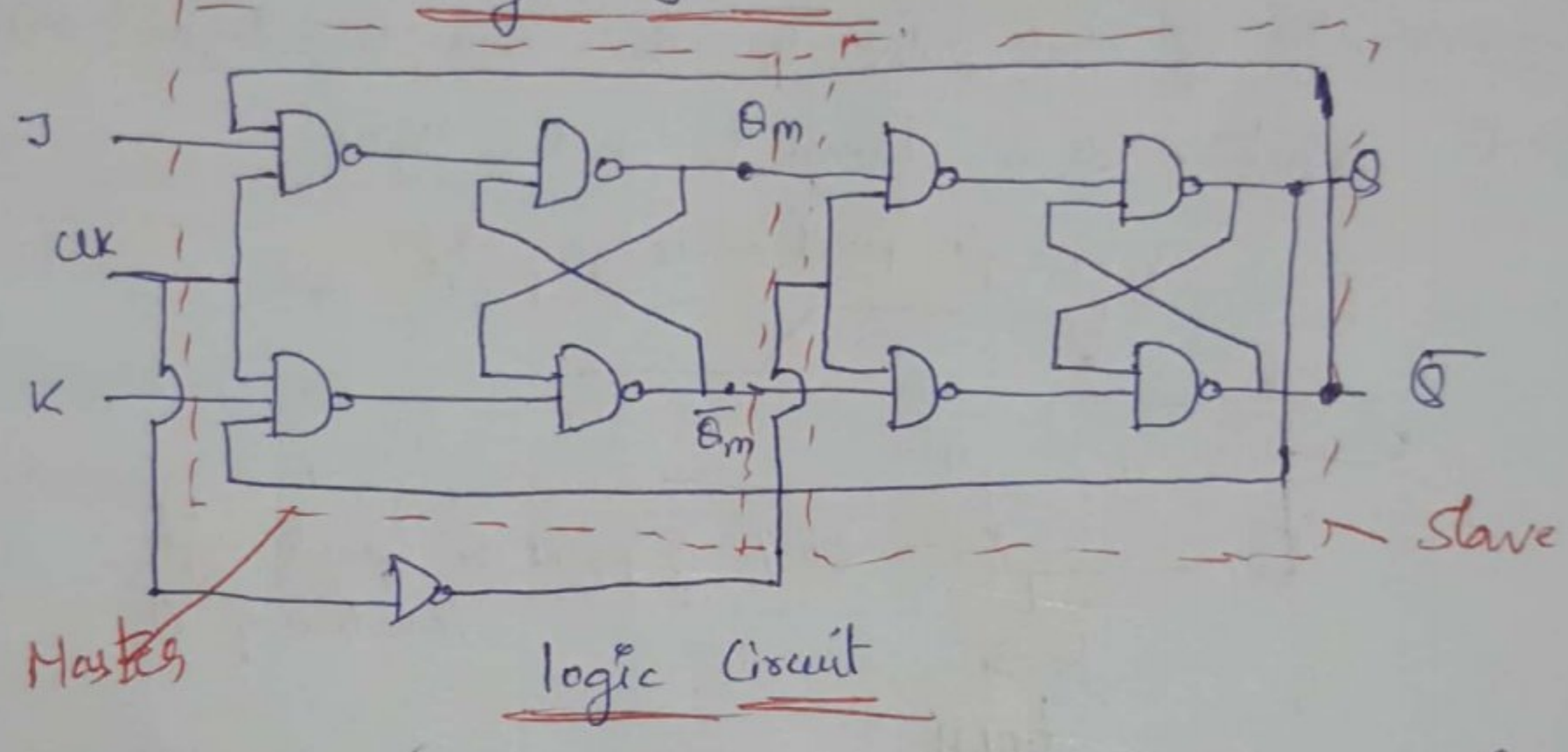
Scanned with CamScanner

Scanned with CamScanner

## Master slave Configuration.




logic Symbol:-

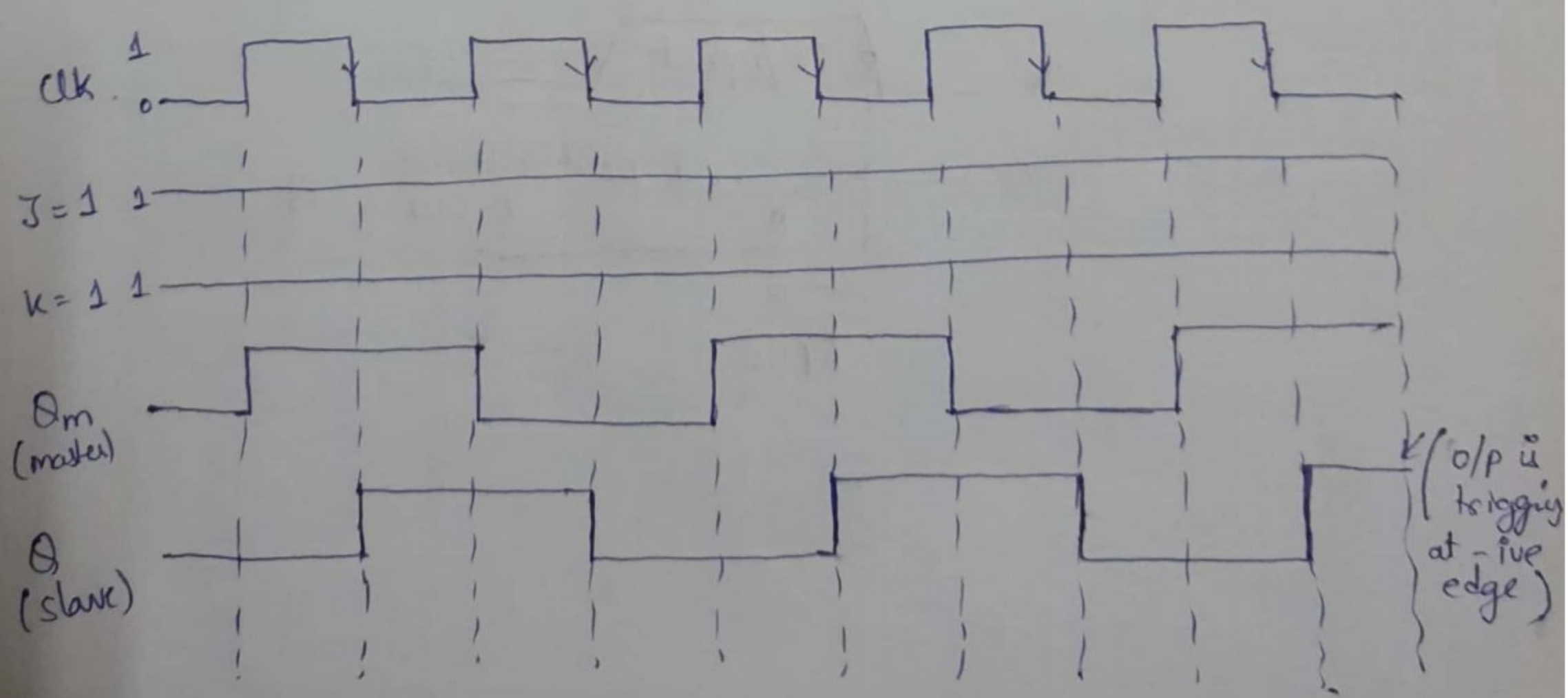



logic Circuit

→ If J = 1 ; k = 1; clk=1; $Q_m$ = active ; Q = memory. (Master is active)

If J = 1 ; k = 1; clk = 0 ; $Q_m$ = memory ; Q = Active (slave is active)

Waveforms.




Scanned with CamScanner

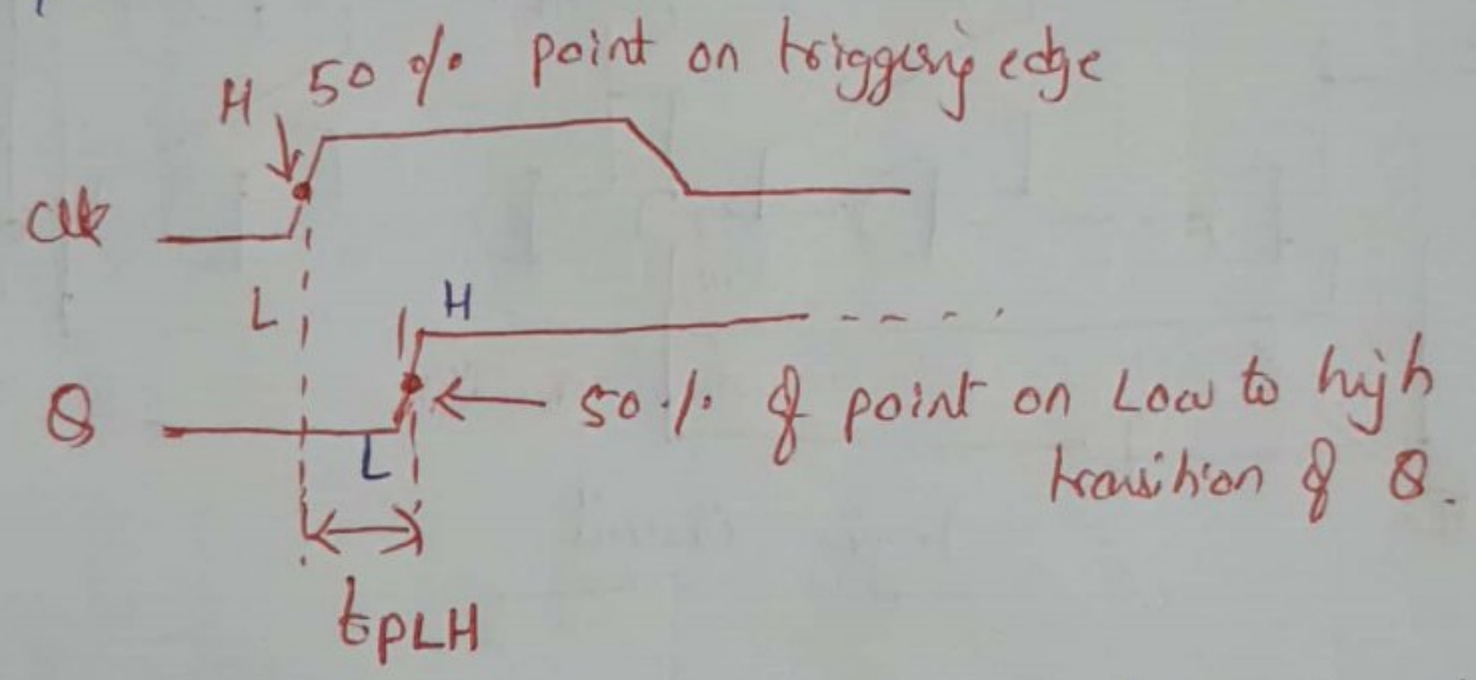
Scanned with CamScanner

## Flip Flop operating characteristics:-

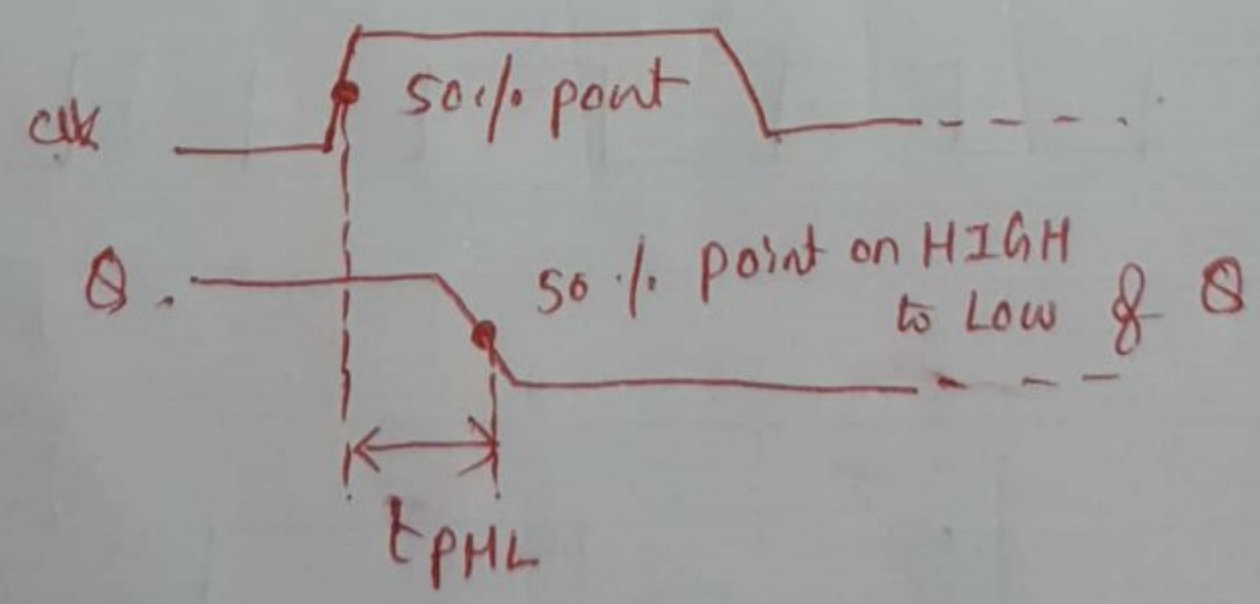
① **Propogation delay :-** It is the interval of time required after an input signal has been applied to the resulting output change to occur.

→ There are four categories of propogation delay times :-

ⓐ **Propogation delay ($t_{PLH}$) :-** It is measured from the triggering edge of clock pulse to the Low-to High transition of the output. It is shown in below figure.




ⓑ **($t_{PHL}$) :-** It is measured from the triggering edge of the clock pulse to High to LOW transition of output. It is shown in below figure




Scanned with CamScanner

Scanned with CamScanner

② Set up time :- ($t_s$) :- It is the minimum interval required for logic levels to be maintained constantly on the inputs (JK & D) prior to the triggering edge of clock pulse in order for the levels to be reliable clocked into the flip flop.

③ Hold time ($t_H$) :- It is the minimum interval required for the logic levels to remain on the inputs after the triggering edge of clock pulse in order for the levels to be reliably clocked into the flip flop.

④ Maximum clock frequency :- ($f_{max}$) :- It is highest rate at which a flip-flop can be reliably triggered. At clock frequencies above the maximum, the flip flop would be unable to respond quickly enough, and its operation would be ~~impaired~~ impaired.

⑤ Clock Skew :-

One of the most common timing problem in synchronous circuit is clock skew.

→ The clock signal which is applied simultaneously to all the flip flops in a synchronous system may undergo varying degrees of delay caused by wiring between components and arrive at the clock inputs of different flip flops at different times. This delay is called clock skew.

Scanned with CamScanner

Scanned with CamScanner

## Asynchronous Inputs :-

→ For clocked Flip Flop :- S, R, J, K, D, T are called Synchronous Inputs, because their effect on Flip Flop output is synchronized with the clock input.

→ These synchronous input should be used in conjunction with clock signal to trigger the flipflop.

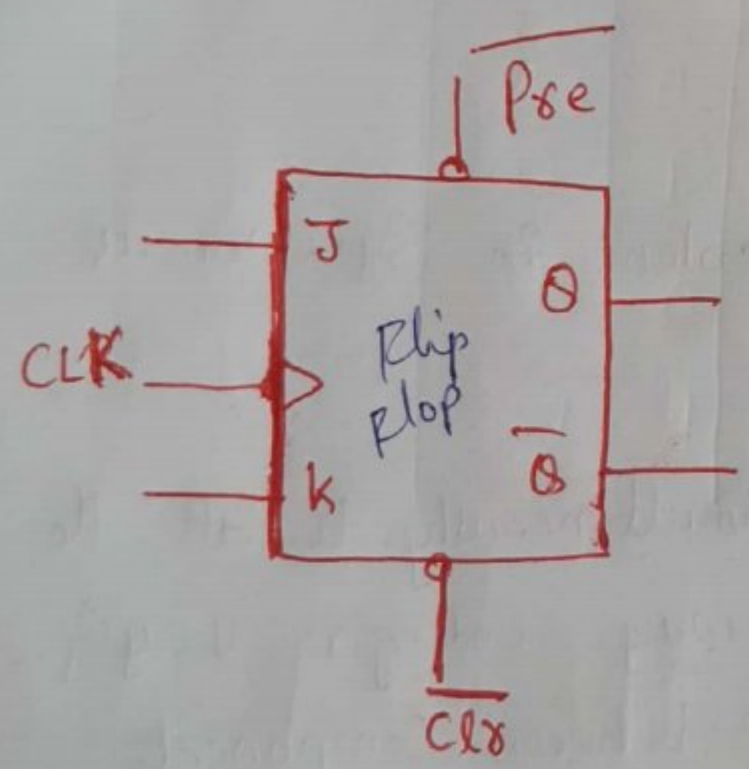
→ Most IC's flip flop have one or more asynchronous inputs.

→ These affect the flipflop output independently of the synchronous inputs and the clock input.

→ SET & RESET

→ PRESET (PRE) & Direct SET ($S_D$) & DC SET.

→ CLEAR (CLR) & Direct reset ($R_D$) & DC CLEAR.

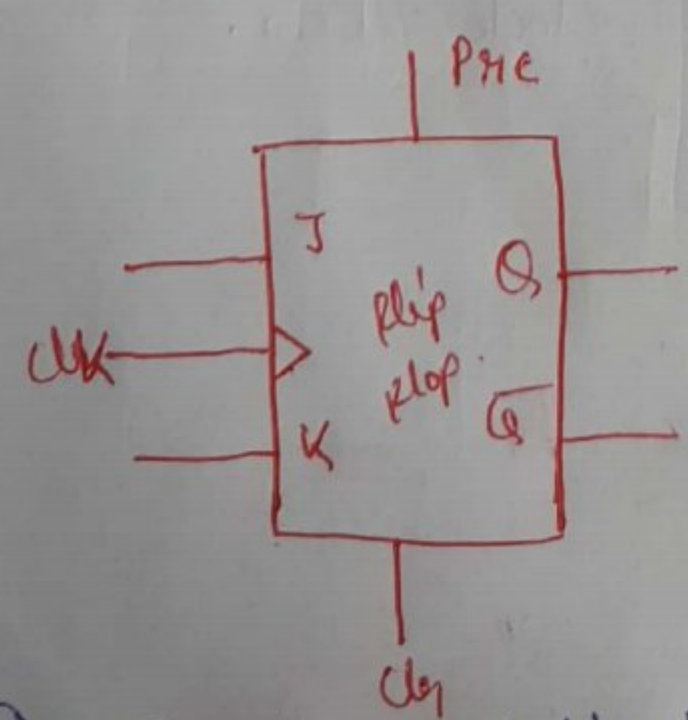



Preset = 0
Q = 1
Clear = 0
Q = 0

Active low

| $\overline{Pre}$ | $\overline{clr}$ | output (Q) |
|---|---|---|
| 0 | 0 | Not used (Invalid) |
| 0 | 1 | Q = 1 |
| 1 | 0 | Q = 0 |
| 1 | 1 | Clocked operation |

(a) JK flip flop Active low




Preset = 1
Q = 1
Clr = 1
Q = 0

Active high.

| Pre | clr | output |
|---|---|---|
| 0 | 0 | Clocked operation |
| 0 | 1 | Q = 0 |
| 1 | 0 | Q = 1 |
| 1 | 1 | Not used (Invalid) |

(b) JK flip flop Active High.

Scanned with CamScanner

Scanned with CamScanner

## Flip Flop Conversions :-

Steps :-

1. Identify available and required FlipFlop.
2. Make characteristic Table for required flipflop.
3. Make Excitation Table for available flipflop.
4. Write boolean expression for available flipflop.
5. Draw the logic symbol.

### Q. Convert JK Flip Flop to D Flip Flop ?

Ans Available Flip Flop → JK Flip Flop.
Required FlipFlop → D FlipFlop.

characteristic Table of required flip flop. (D F/F)

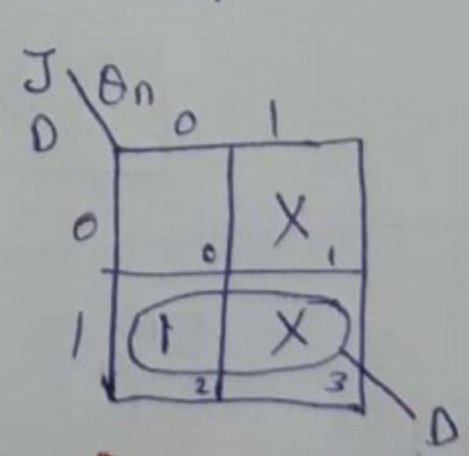
| D | $Q_n$ | $Q_{n+1}$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Excitation Table of available Flip flop (JK Flip flop)

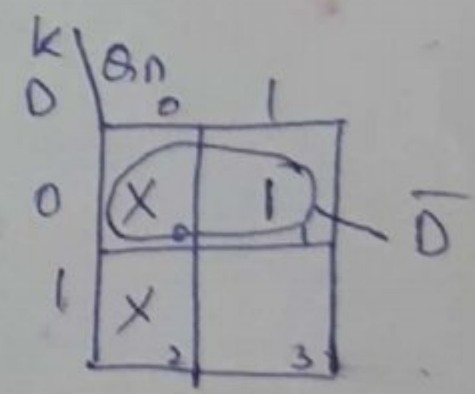
| $Q_n$ | $Q_{n+1}$ | J | K |
|---|---|---|---|
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | X |
| 1 | 0 | X | 1 |
| 1 | 1 | X | 0 |

| 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 |

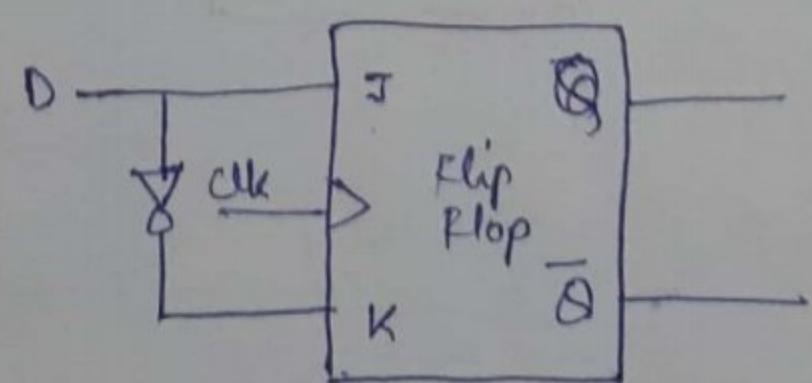
k-map.




$J = D$




$K = \overline{D}$




Scanned with CamScanner

Scanned with CamScanner

② Convert T flip flop to D flip flop.

Available Flip Flop → T flip flop

Required flip flop → D flip flop.

Characteristic Table of Required flip flop (D F/F)

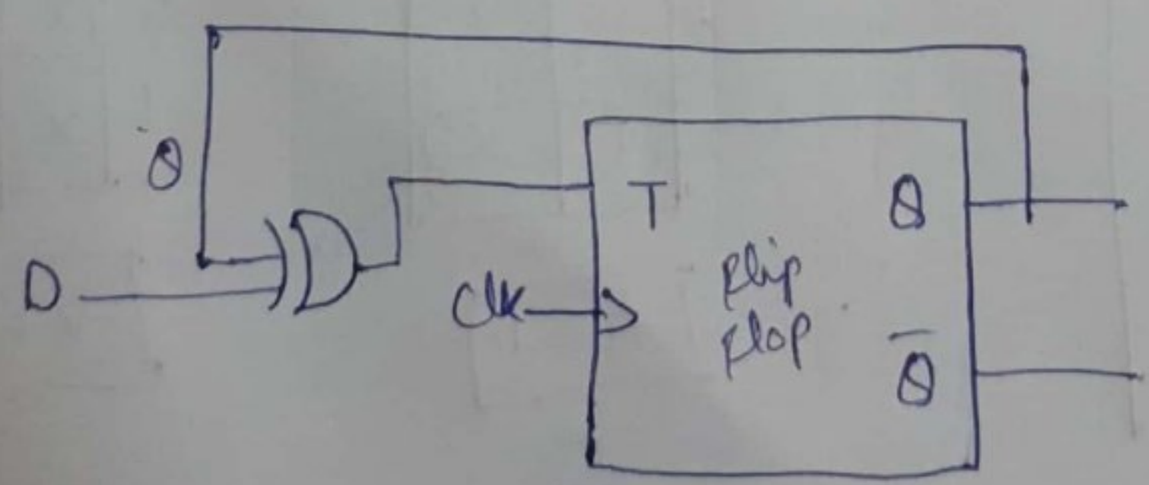
| D | $Q_n$ | $Q_{n+1}$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Excitation Table of Available flip flop (T F/F)

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

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

$$\boxed{T \Rightarrow D \oplus Q_n}$$




Scanned with CamScanner

Scanned with CamScanner

③ Convert SR Flip Flop to JK Flip Flop.

Ans Available is SR Flip Flop

Required is JK flip Flop.

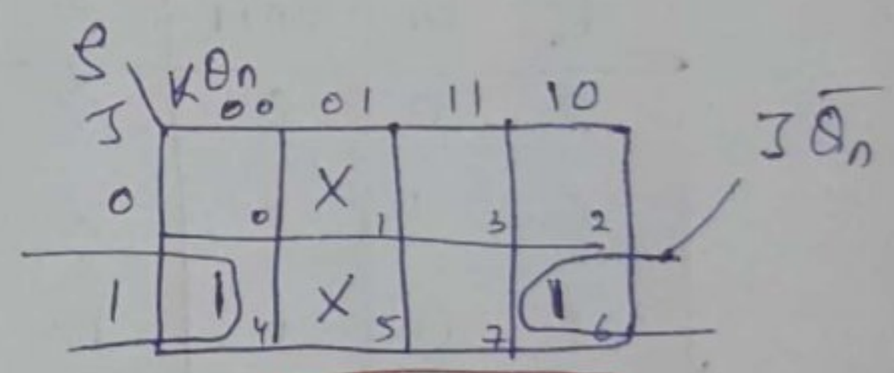
Characteristic Table of Required Flip Flop (JK - Flip Flop)

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

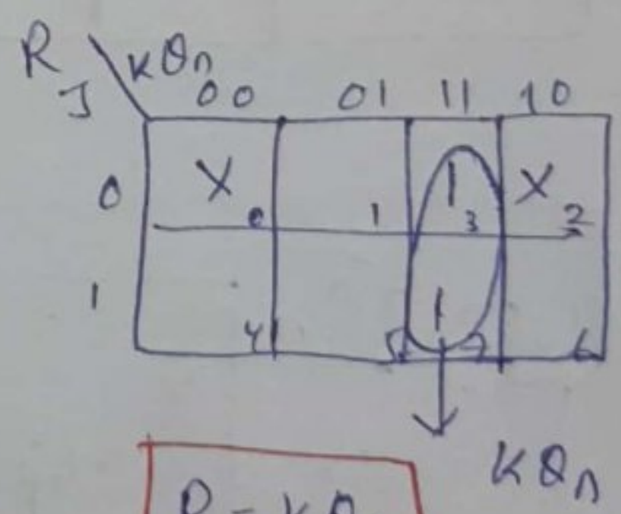
Excitation Table of available Flip Flop (SR - Flip Flop)

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

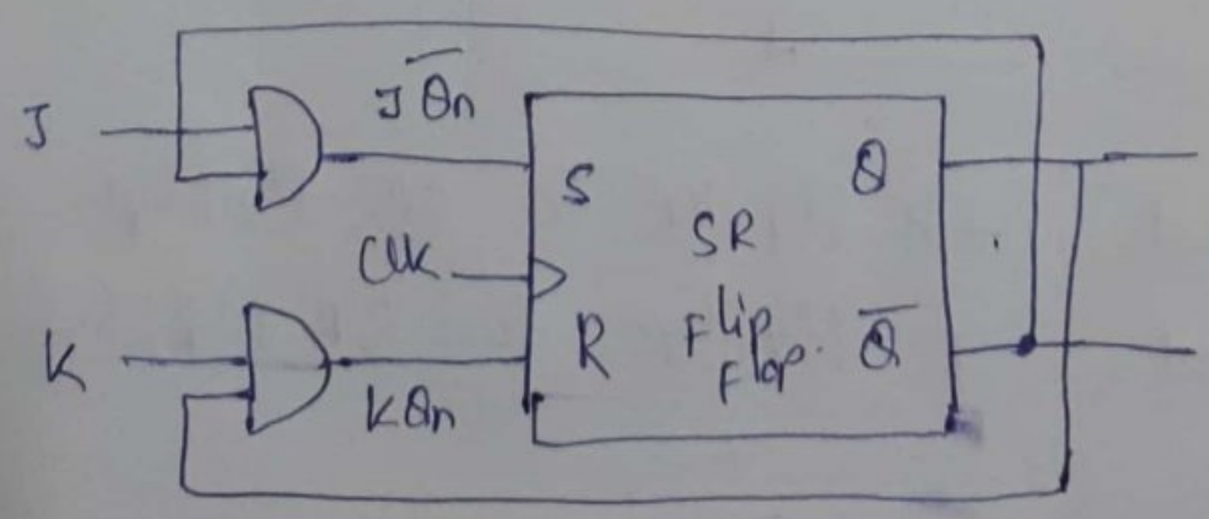
| J | K | $Q_n$ | $Q_{n+1}$ | S | R |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | X |
| 0 | 0 | 1 | 1 | X | 0 |
| 0 | 1 | 0 | 0 | 0 | X |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | X | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 |




$S = J\overline{Q_n}$




$R = KQ_n$.




Scanned with CamScanner

Scanned with CamScanner

Q) Convert SR flip flop to T - flip flop.

S

Available flip flop → SR

Required flip flop → T

Required flip flop - characteristic Table.

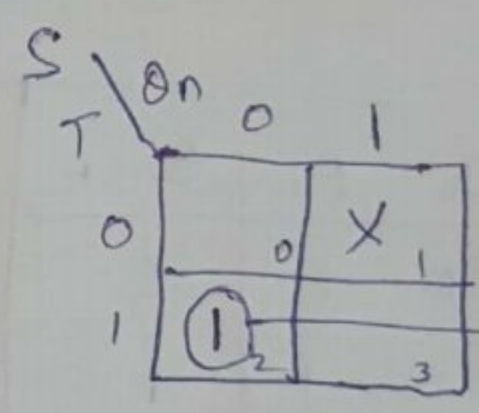
T - flip flop.

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

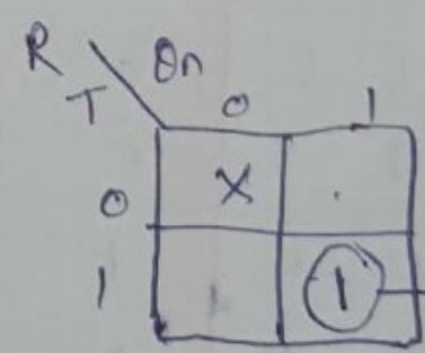
Available flip flop excitation Table. (SR - flip flop)

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

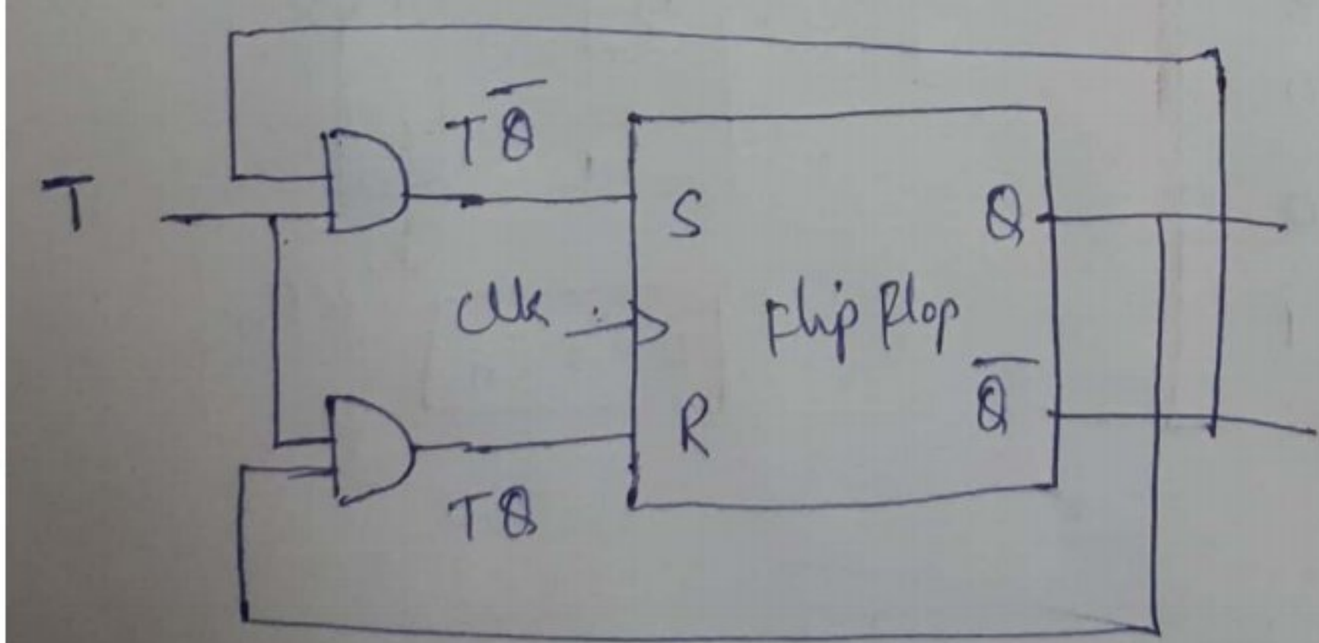
| T | $Q_n$ | $Q_{n+1}$ | S | R |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | X |
| 0 | 1 | 1 | X | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |




$S = T\overline{Q_n}$




$R = TQ_n$




H.W

1. Convert JK flip flop to SR flip flop
2. Convert T flip flop to SR flip flop.

Scanned with CamScanner

Scanned with CamScanner

# SHIFT REGISTERS

→ A Register is a device which is used to store n bits of data & information. It is a group of flipflops connected in series used to store multiple bits of data.

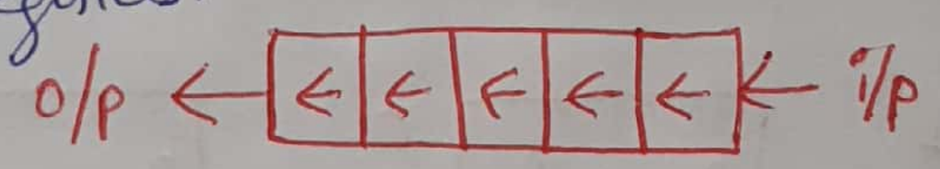
→ The information stored within this registers can be transferred with the help of registers called as shift registers.

→ A shift register is a group of flipflops used to store multiple bits of data.

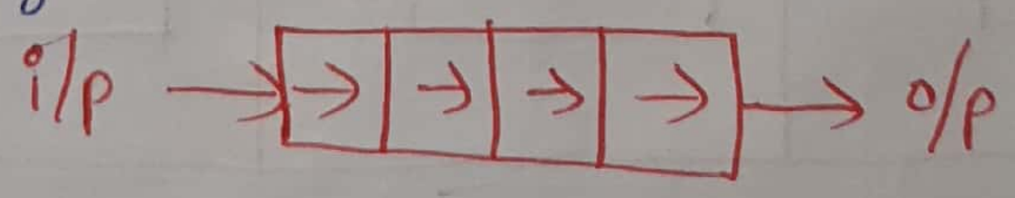
→ The bits stored in such registers can be made to move within the registers and in/out of the register by applying clock pulse.

→ An n-bit shift register can be formed by connecting n flipflop where each flipflop stores a single bit of data.

→ The registers which will shift the bits to left are called shift left registers.




→ The register which will shift the bits to right are called right shift registers.



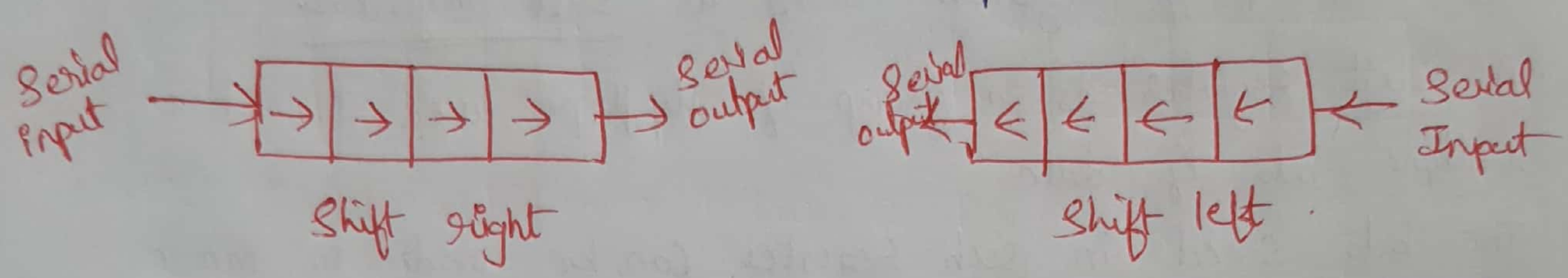

→ Shift registers are basically of four types :-

1. Serial in serial out shift register
2. Serial in parallel out shift register
3. Parallel in serial out shift register
4. Parallel in parallel out shift register.

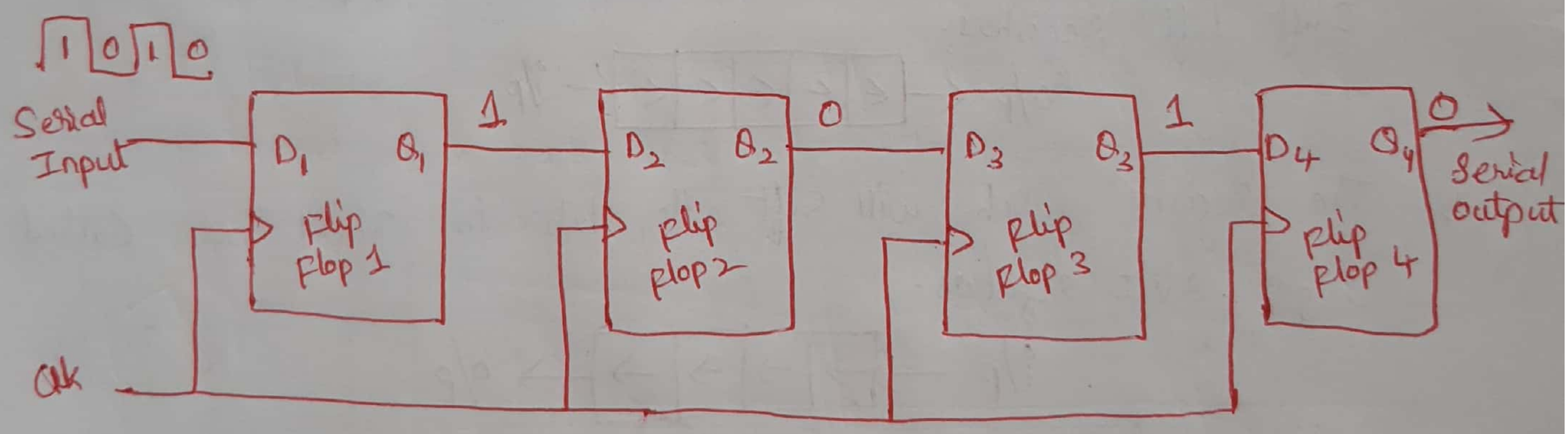
## ① Serial - In Serial - out Shift Register :- ( SISO)

→ The shift register, that allows serial input (one bit after the other through a single data line) and produces a serial output is known as Serial - In Serial - out shift register.

→ Here there is only one output, the data leaves the shift register one bit at a time in a serial pattern.




→ The logic circuit is shown below. The circuit consist of four D - flip flops that are connected in a serial manner. All these flip flops are synchronous with each other since the same clock signal is applied to each flip flop.



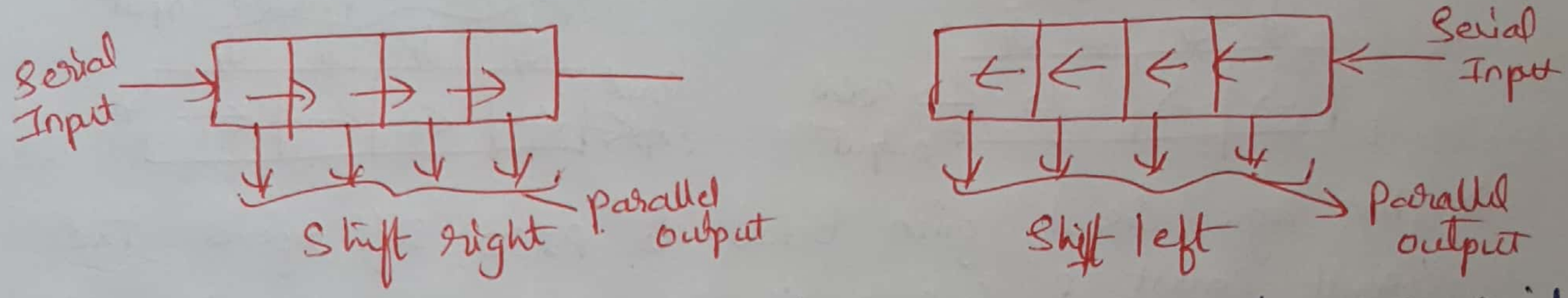

→ The above circuit is an example of shift right register, taking the serial data input from left side of the flip flop.

→ The main use of SISO is to act as a delay element.

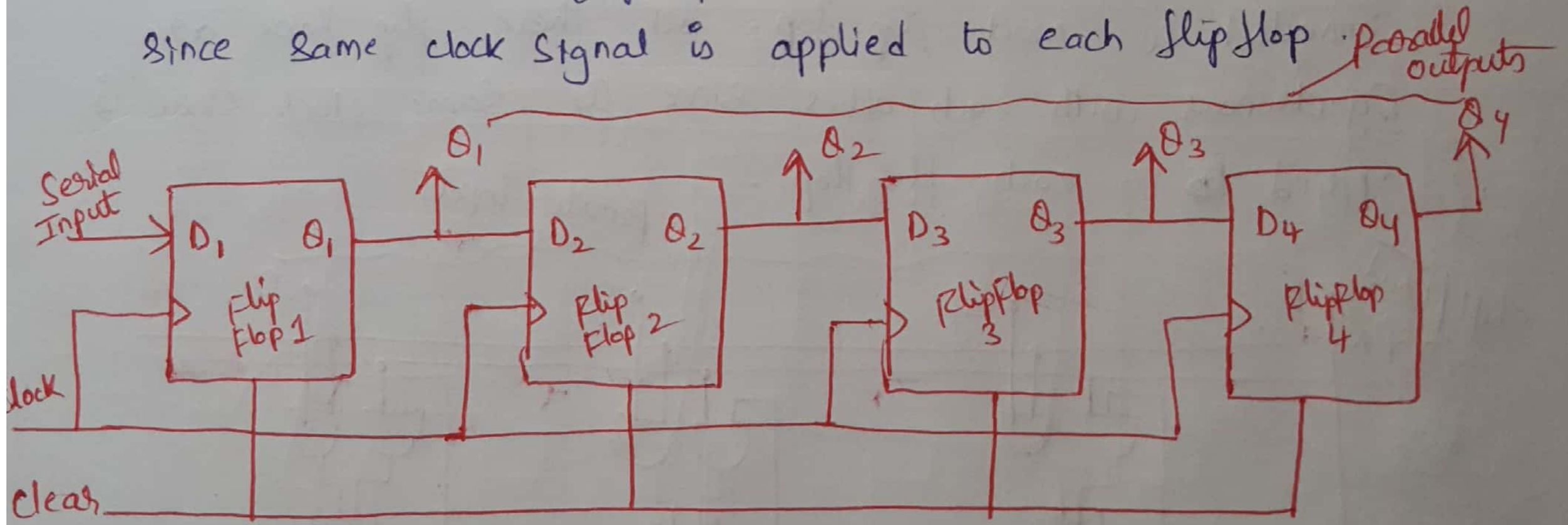
## ② Serial-In Parallel-out Shift register (SIPO) :-

→ The shift register that allows serial input (one bit after the other through a single data line) and produces a parallel output is known as Serial-In Parallel-out Shift register.

→ The shift registers for shift left and shift right are shown below




→ The logic circuit given below shows a SIPO register. The circuit consists of four D-flipflops that are connected. The clear (CLR) signal is connected in addition to clock signal to all the four flip flops in order to RESET them. The output of first flipflop is connected to the input of next flipflop and so-on. All these flipflops are synchronous with each other since same clock signal is applied to each flip flop.



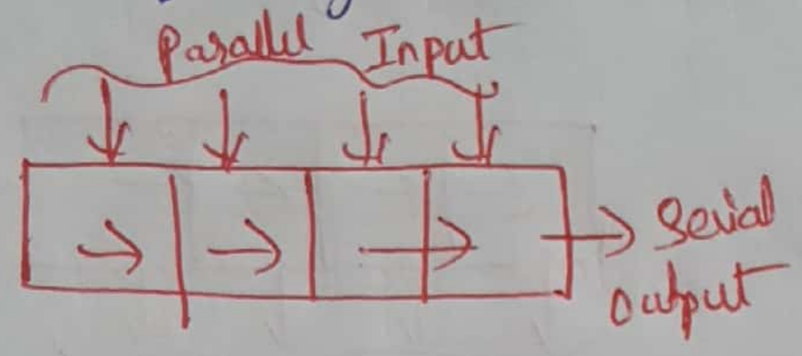

→ The above circuit is an example of shift right register, taking the serial data input from left side of the flipflop and producing a parallel output.

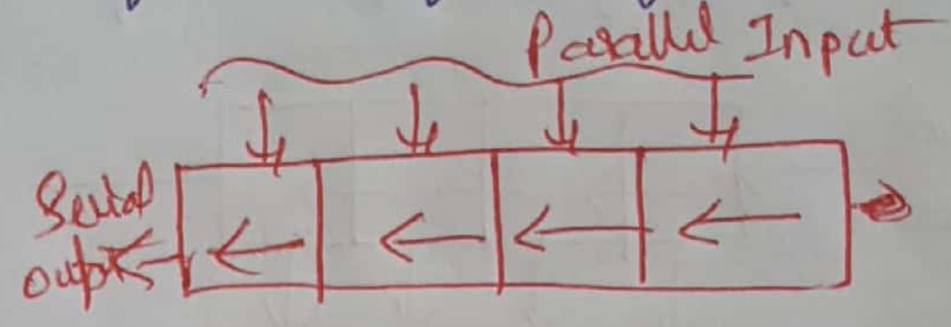
→ They are used in communication lines where demultiplexing of data line into several parallel lines is required.

## ③ Parallel - In Serial out Shift register (PISO)

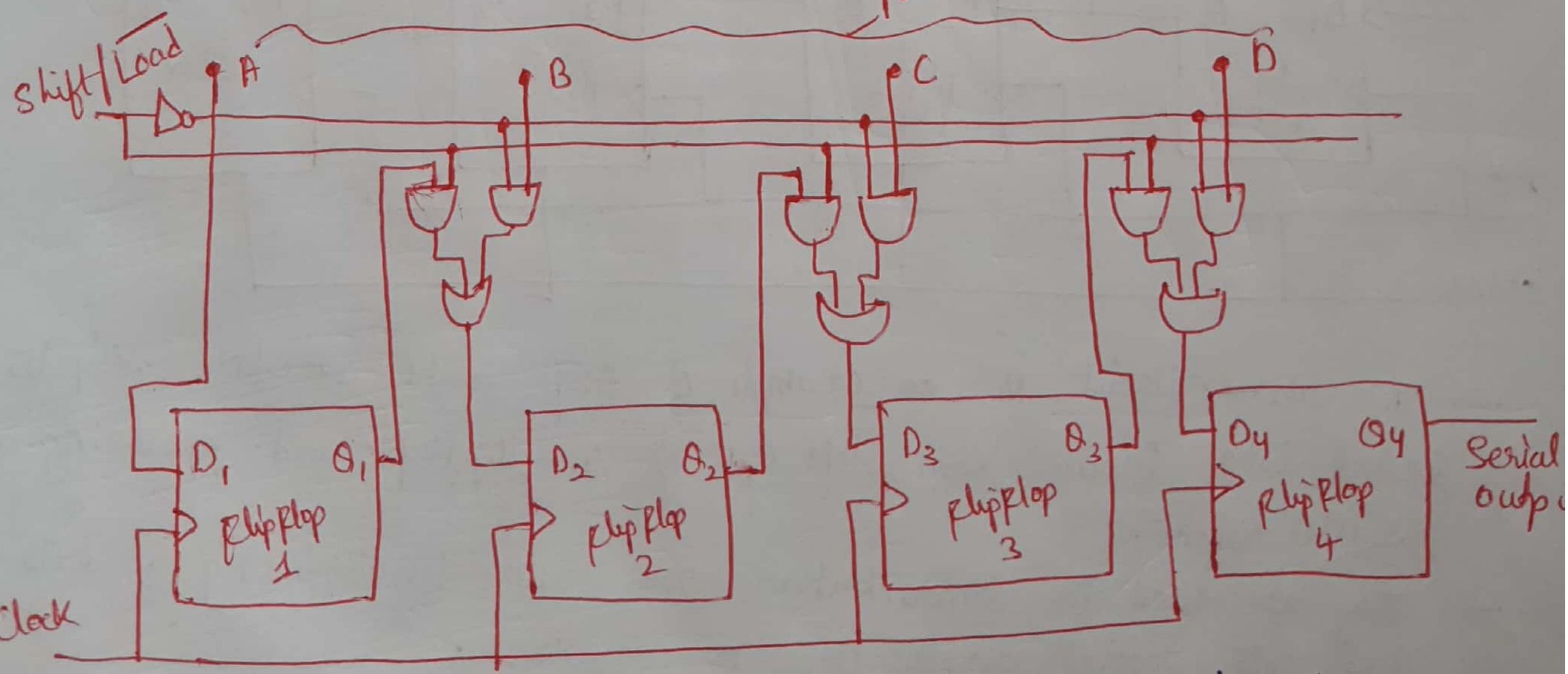
→ The shift register, that allows parallel input (data is given seperately to each flip flop and in simultaneously manner) and produces a serial output is known as PISO shift register.

→ The shift register for right shift and left shift are shown below:




→ The logic circuit given below is PISO shift register. The circuit consist of four D-flip flops that are connected. The clock input is directly connected to all the flip flops but the input data is connected individually to each flip flop through a multiplexer at the input of each flip flop. The output of previous flip flop and parallel data input are connected to the input of the MUX and the output of MUX is connected to next flip flop. All these flip flops are synchronous with each other since the same clock signal is applied to each flip flop.

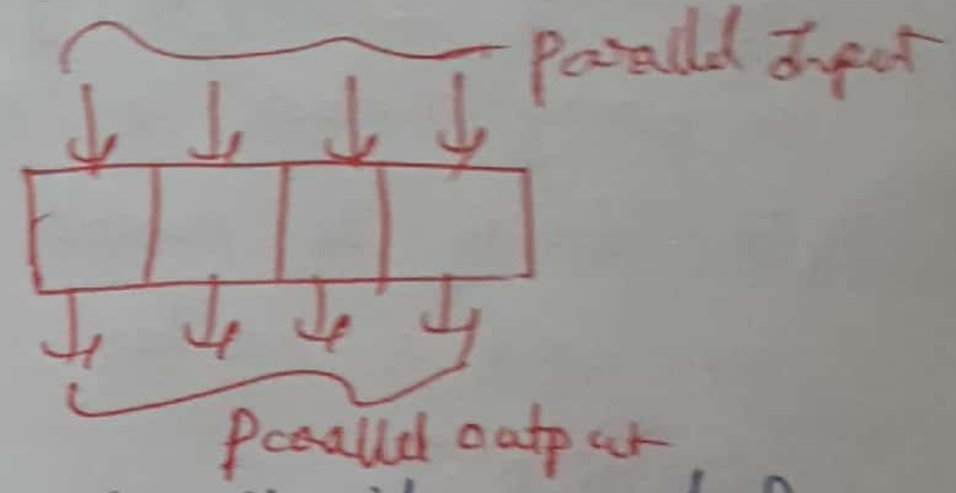



→ A PISO shift register is used to convert parallel data to serial data.

## (4) Parallel-In Parallel-out Shift register (PIPO):

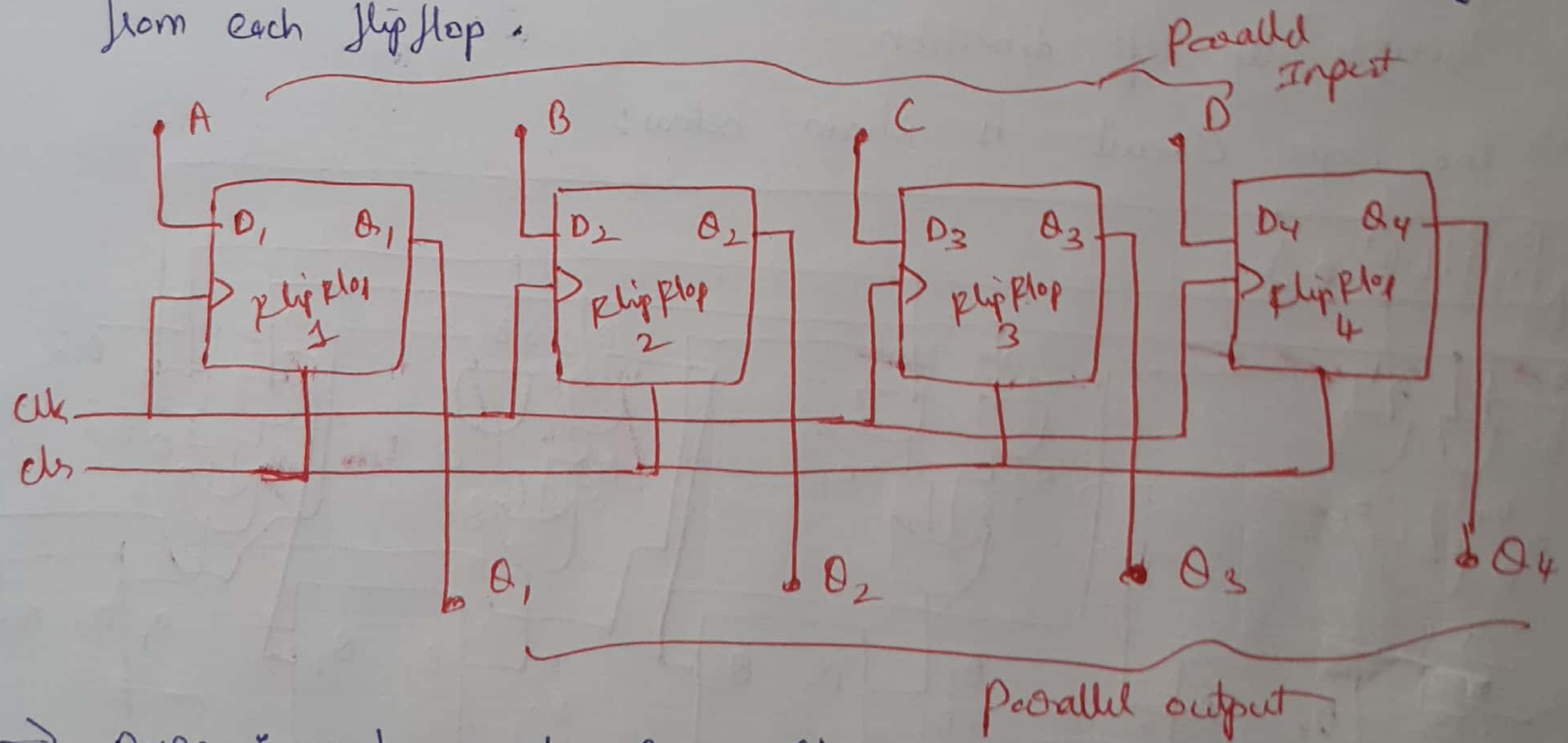
→ The shift register that allows parallel input (data is given seperately to each flip flop and in a simultaneous manner) and also produces a parallel output is known as PIPO.

→ The shift register for PIPO is




→ The logic circuit below shows PIPO. The circuit consist of four D-flip flops that are connected. The clear (CLR) signal and clock signals are connected to all 4 flip flops. In this type of register, there are no interconnections between individual flipflops, since no serial shifting of data is required.

→ Data is given as input seperately for each flip flop and in the same way, output also collected individually from each flip flop.




→ PIPO is used as temporary storage device and it also act as delay element.

## Bidirectional Shift Register :-

→ If we shift a binary number to the left by one position, it is equivalent to multiplying the number by 2, and if we shift a binary number to right by one position, it is equivalent to dividing the number by 2. To perform these operations we need a register that can shift the data in either direction.

→ Bidirectional shift register are the registers that are capable of shifting the data either right or left depending on the mode selected.

→ If the mode selected is high i.e (1), the data will be shifted towards the right direction.

→ If the mode selected is Low i.e (0), the data will be shifted towards left direction.

→ The Bidirectional Shift register consist of four D flipflops that are connected together. The input data is connected at two ends of the circuit.

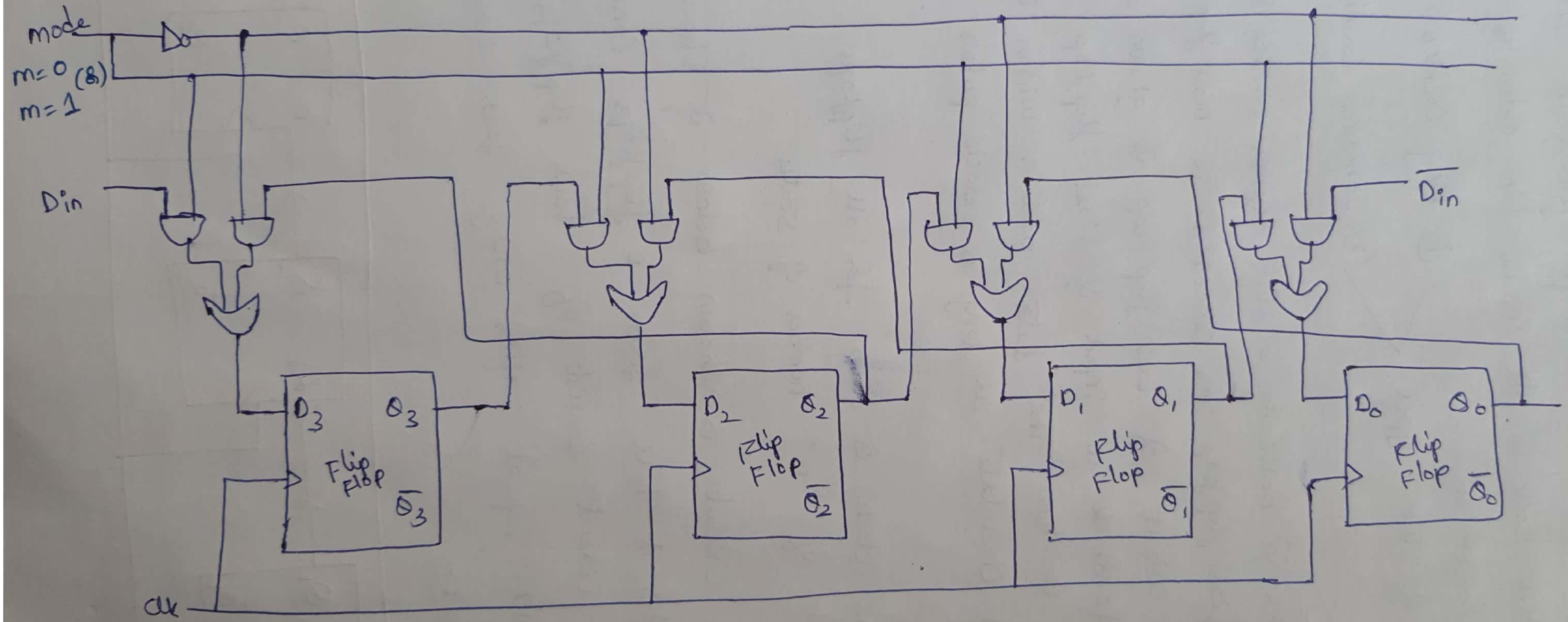
→ Depending on the mode selected only one AND gate will be in active state

→ The logic circuit is shown below :-

→ Based on mode the Shifting operation will be decided.

→ when $m=0$ ; the Left Shift operation will happen $o/p = (Q_2\ Q_1\ Q_0\ \overline{D_{in}})$

→ when $m=1$ ; the right Shift operation will happen $o/p = (D_{in}\ Q_3\ Q_2\ Q_1)$




Scanned with CamScanner

## Shift Register Counters :-

→ Shift register counters are the Shift registers in which the outputs are connected back to the inputs in order to produce particular sequences.

→ They are of two types
- ① Ring Counter
- ② Johnson Counter.

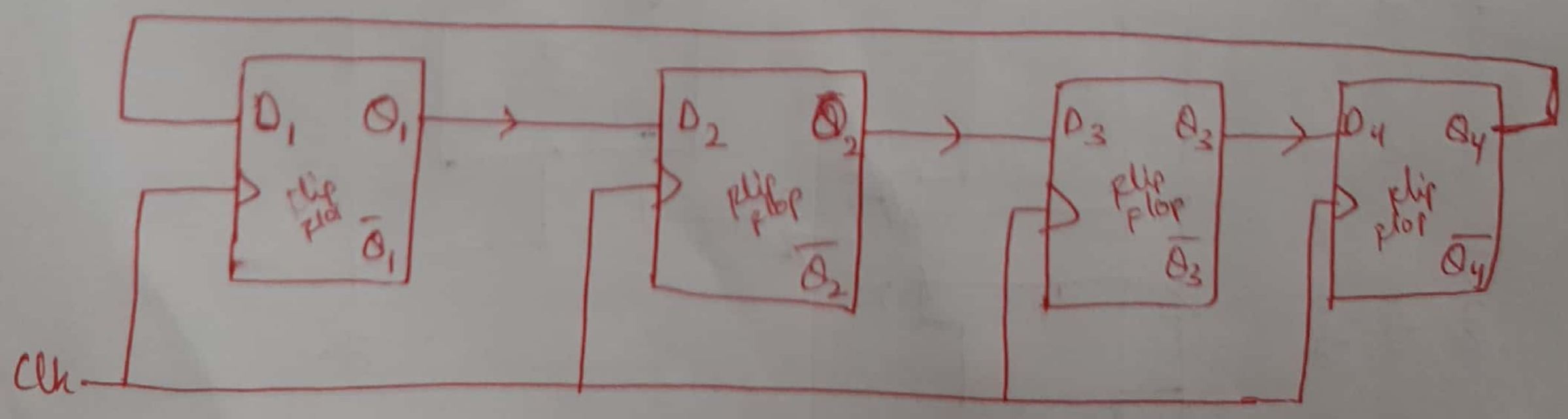
### ① Ring Counter:-

→ A ring counter is basically a shift register counter in which the output of first flip flop is connected to next flip flop and so on. and output of last flip flop is given as the input via feedback to input of first flipflop. Thus the name ring counter. The data pattern within the shift register will circulate as long as clock pulses are applied.

→ Here same clock is given for all flipflops.

→ It consist of 'n' number of states.

→ The logic circuit is shown below it shows a ring counter it has four D flipflops connected together. Since the circuit consist of four flip flops the data pattern will repeat after every four clock pulses as shown :-

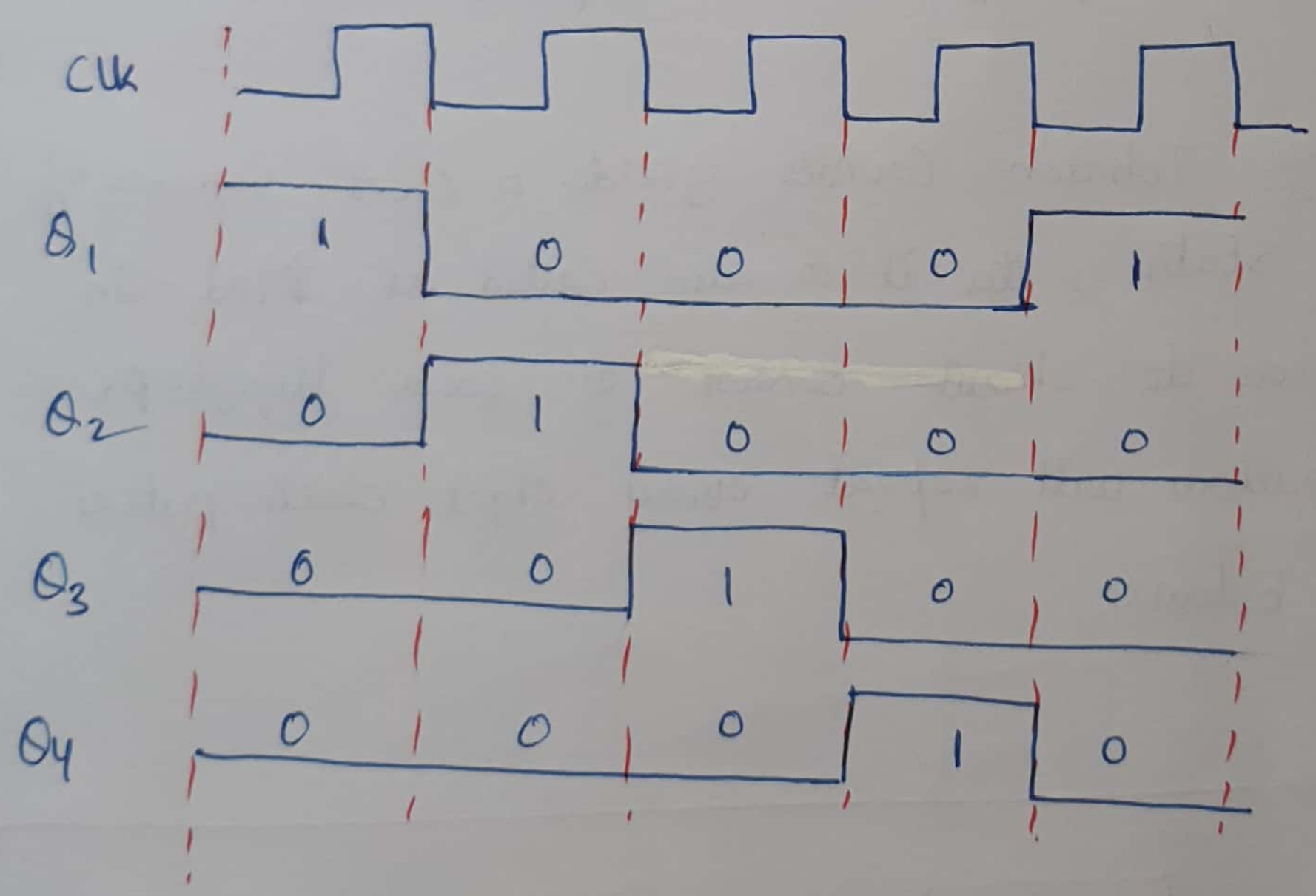


Truth Table.

| Clk pulse | $Q_1$ | $Q_2$ | $Q_3$ | $Q_4$ |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 0 | 1 |
| 4 | 1 | 0 | 0 | 0 |

(repeated)

Waveforms.




→ The ring counter is a self decoding circuit

→ No extra decoding circuit is needed to determine what state the counter is in.

## Johnson Counter:-

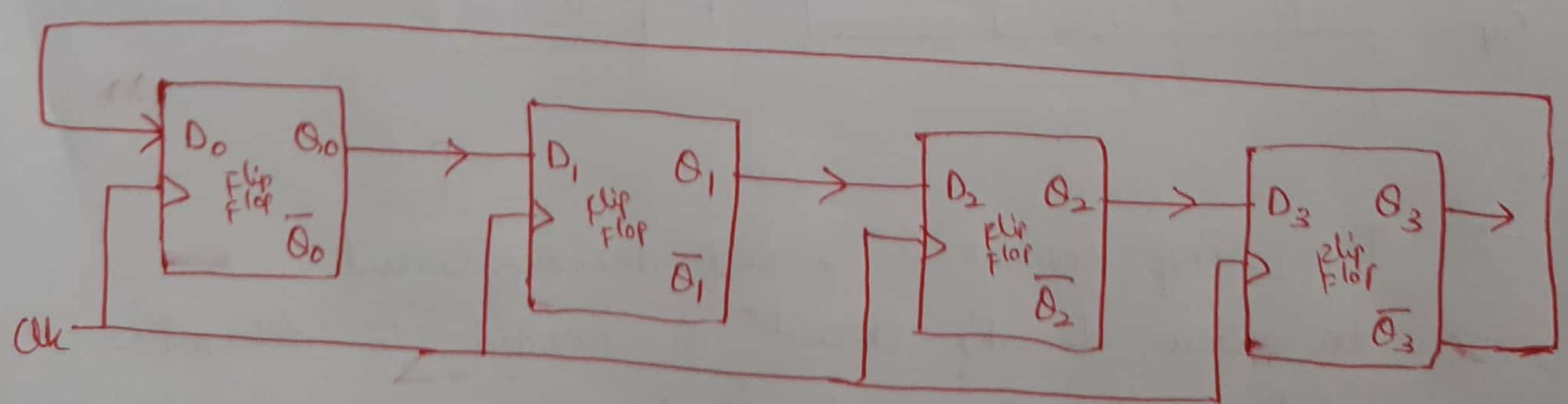
→ It is basically a shift register counter in which the output of the first flip is connected to next flip flop and so on. and the inverted output of last flip flop is again fed back to the input of first flip flop. They are also called a Johnson ring counter / Twisted ring counter.

→ It is one of the application of shift register.

→ Total no of states = 2 × no of flipflops. (or) 2 × no of bits

→ The logic circuit shown below is a Johnson counter. The circuit consist of of four D-flipflops connected together.

→ An n-stage Johnson counter yields a count sequence of 2n different states, thus it is also called as Mod-2n counter. Since the circuit consist of four flipflops the data pattern will repeat every eight clock pulses as shown below:-



## Truth Table :-

| Clk pulse | $Q_0$ | $Q_1$ | $Q_2$ | $Q_3$ |
|---|---|---|---|---|
| X | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 1 | 1 | 0 | 0 |
| 3 | 1 | 1 | 1 | 0 |
| 4 | 1 | 1 | 1 | 1 |
| 5 | 0 | 1 | 1 | 1 |
| 6 | 0 | 0 | 1 | 1 |
| 7 | 0 | 0 | 0 | 1 |
| 8 | 0 | 0 | 0 | 0 |

Inv

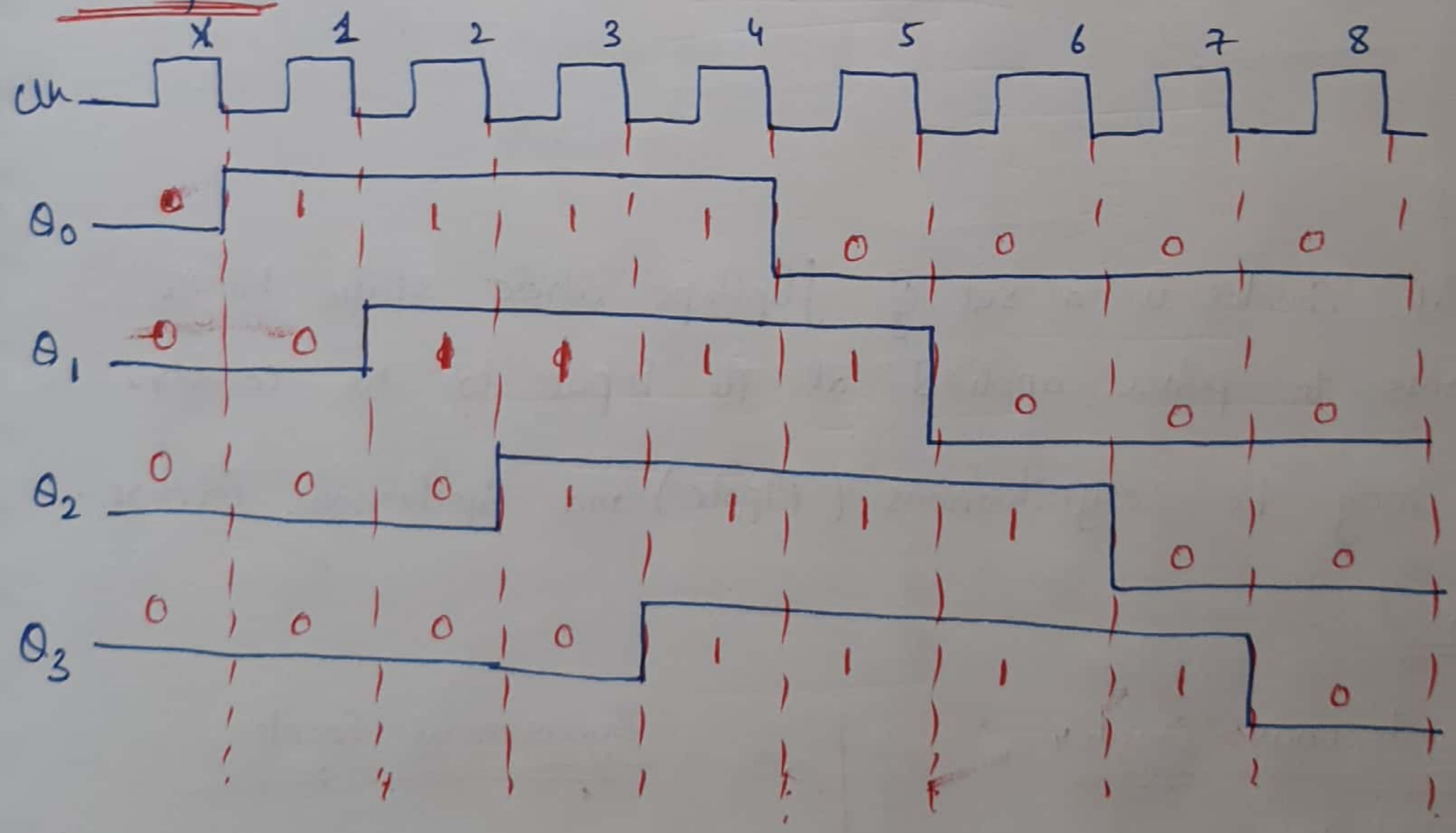
Inv

⇒ 2N States

⇒ 2×4

⇒ 8 States

## Waveform :-




→ The main advantage of Johnson Counter is that it only needs "n" number of flip flops compared to the ring counter to circulate a given data to generate a sequence of 2n states.

## Applications of Shift registers:-

→ The shift register are used for temporary data storage.

→ The shift registers are also used for data transfer and data manipulation.

→ The Serial-in Serial-out and parallel-in Parallel-out shift registers are used to produce time delay to digital circuits

→ The Serial in parallel out shift register is used to convert serial data into parallel data thus they are used in communication lines when demultiplexing of a data line into several parallel line is required

→ A parallel in serial out shift register used to convert parallel data to serial data.

## Counters:-

→ A digital counter is a set of flipflops whose status changes in response to pulses applied at the input to the counter.

→ Counters may be asynchronous (Ripple) and synchronous counters.

### Differences:-

| Asynchronous Counter | Synchronous Counter |
| --- | --- |
| ① In this flipflop are connected in such a way that output of first flipflop drives the clock of second flipflop, the output of second to the clock of third flipflop and so on. | → In this type of counter there is no connection between the output of first flipflop and clock input of next flipflop and so on. |

Scanned with CamScanner

| | |
|---|---|
| ② All the flipflop are not clocked simultaneously | ② All flipflop are clocked simultaneously. |
| ③ Design and implementation is very simple | ③ It is complex as the number of states increases. |
| ④ Main drawback is low speed. | ④ They are faster. |

→ A counter may be of two types ① up counter
② Down counter.

① **Up-counter** :- A counter that counts in upward direction.
(0, 1, 2, 3, --- N-1, N)

② **Down-counter** :- A counter that counts in downward direction
(N, N-1, --- 3, 2, 1, 0)

**Modulus of the counter** :-

→ The number of states through which the counter passes before returning to the starting or initial state.

For ex :- Mod-4 :- 0 to 3 states
return to 0

→ A 2 bit counter has $2^n \rightarrow 2^2 = 4$ states, called as Mod-4 counter.

→ It divides the input clock frequency by 4, therefore it is called divide by 4 counter.

→ It requires 2 flipflops. (no of states = no of flipflops)

|||ly ⇒ A 3 bit counter has $2^n \rightarrow 2^3 \Rightarrow 8$ states, called as Mod-8 counter (8) divide by 8 counter. It require 3 flipflops.

Scanned with CamScanner

**Full modulus Counter:-**

→ A counter which goes through all the possible states before restarting.

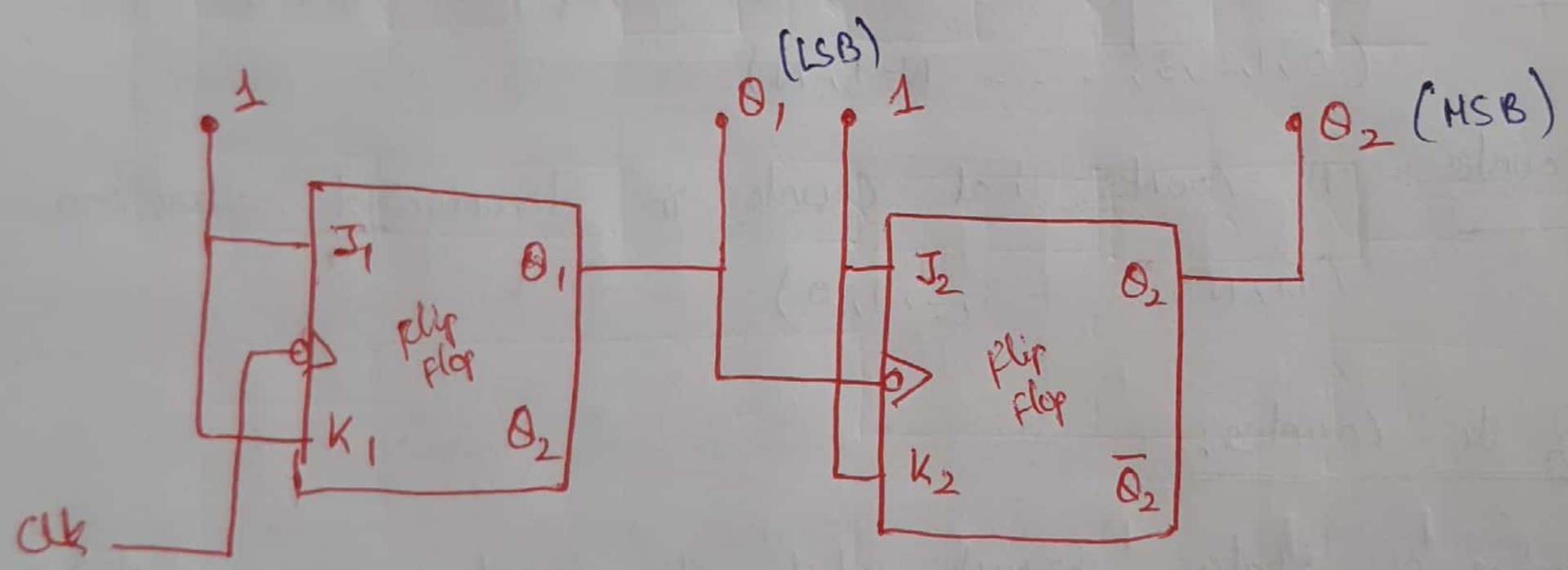
**Variable Modulus Counter:-**

→ A Counter in which maximum number of states can be changed.

**Terminal Count :-** It is the final state of counter sequence.

**Asynchronous Counter :-**

**Design of 2- bit ripple up Counter using -ive edge triggered flipflop.**




→ Asynchronous Counter means clock is not given simultaneously to all flipflops, i.e the o/p of first flipflop act as clock input to next flipflop

→ As it is upcounter it will count from low to high at each clock pulse.

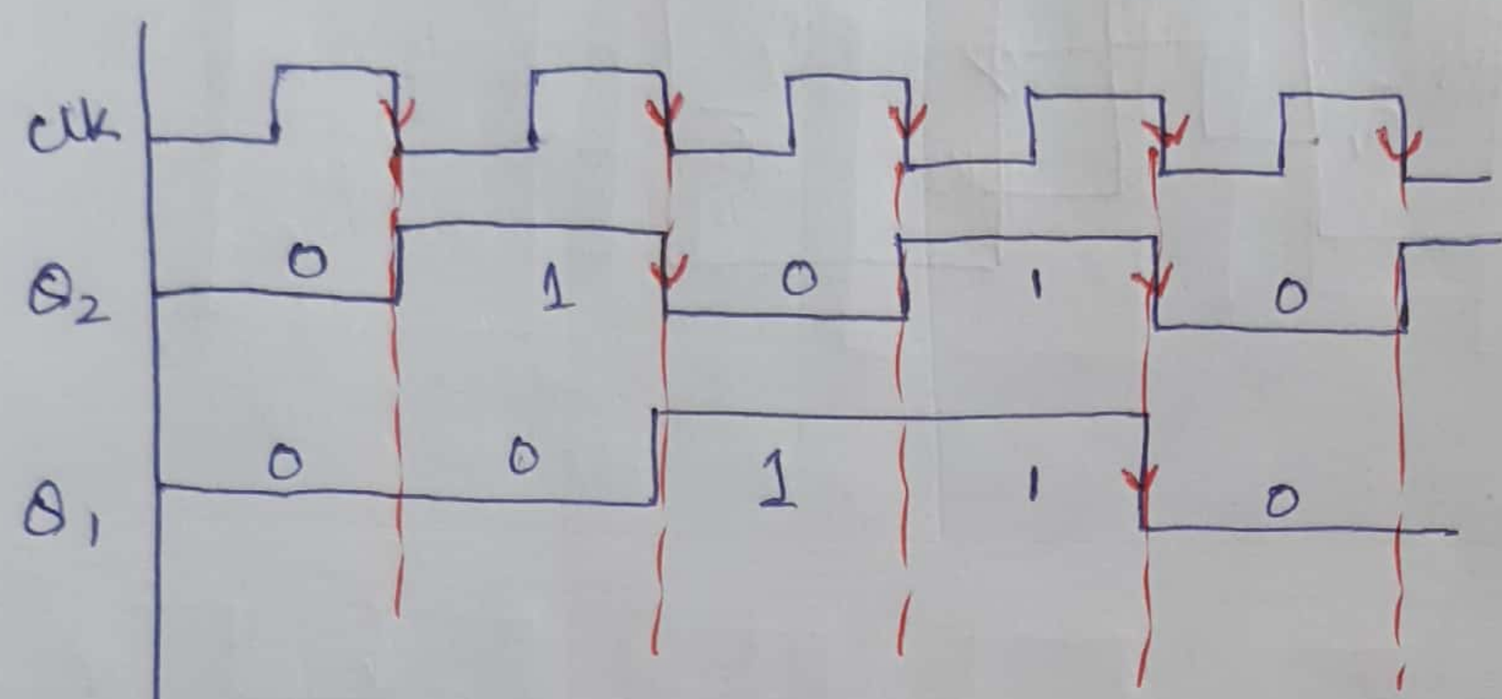
→ As it is 2 bit up counter it count from 00 to 11.

→ As it is 2 bit, it has 2 flip flops, external clock is given to first flip flop, o/p of 1st flipflop is given as clock input to 2nd flip flop.

Scanned with CamScanner

→ A toggling is required hence all the inputs are applied to logic 1.
→ Here a -ive edge triggering is used, All changes in flipflop occurs at falling edge.

$J = K = 1 \ (Q_{n+1} = \bar{Q}_n)$




Truth Table.

| Clock | $Q_1$ | $Q_2$ | Decimal eq. |
|---|---|---|---|
| Initially | 0 | 0 | 0 |
| 1st (↓) | 0 | 1 | 1 |
| 2nd (↓) | 1 | 0 | 2 |
| 3rd (↓) | 1 | 1 | 3 |
| 4th (↓) | 0 | 0 | 0 |

States

$2^n \Rightarrow 2^2 = 4$

Maximum count

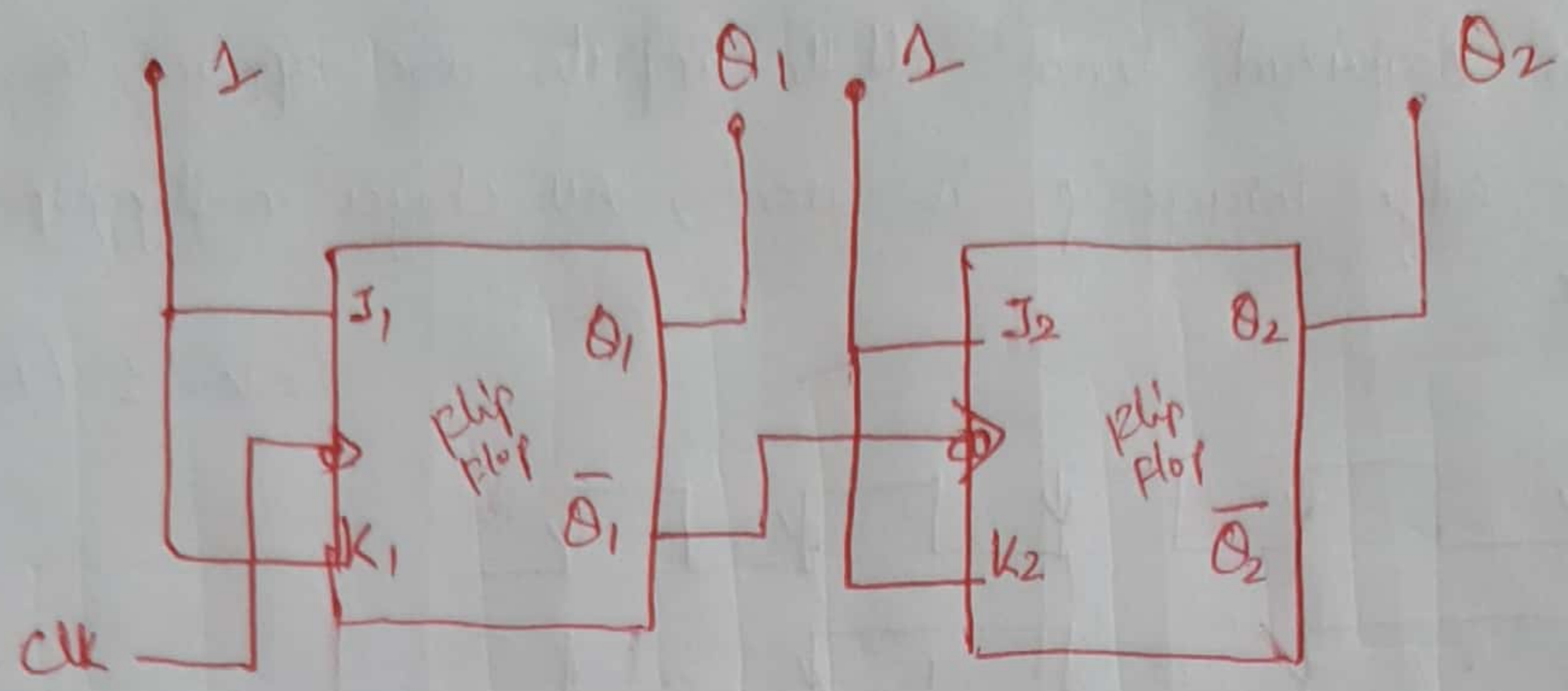
$2^n - 1 \Rightarrow 2^2 - 1$
$\Rightarrow 4 - 1$
$\Rightarrow 3$.

(8) Design of 2 bit ripple down counter using -ive edge triggered flip flop?

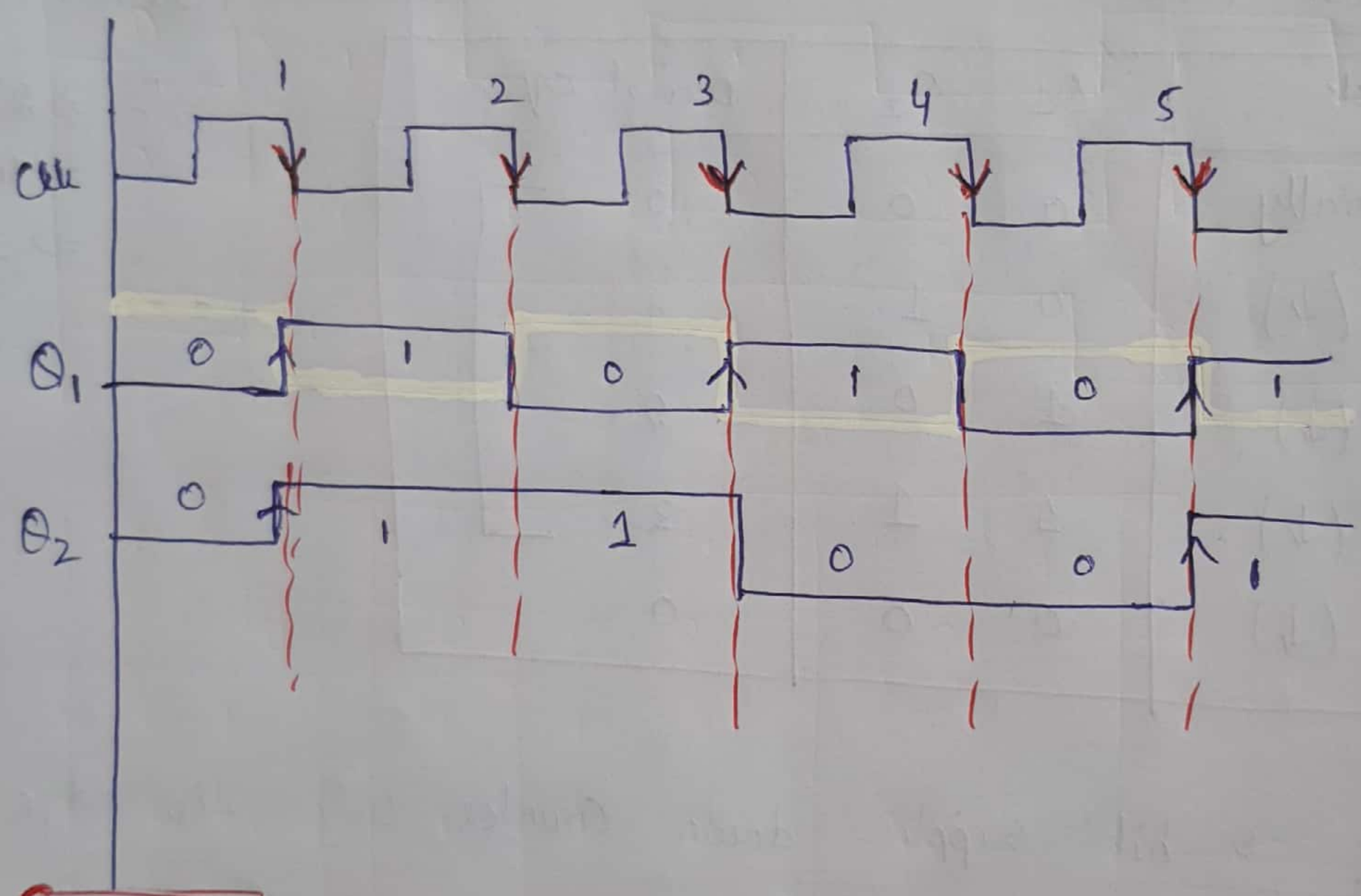
→ A downcounter will count from high to low at each clock pulse

→ As it is 2 bit down counter it count from 11 to 00.

→ As it is 2 bit, it has 2 flipflops, external clock is given to first flipflop and complented output of 1st flip flop is given as clock input to 2nd flipflop.



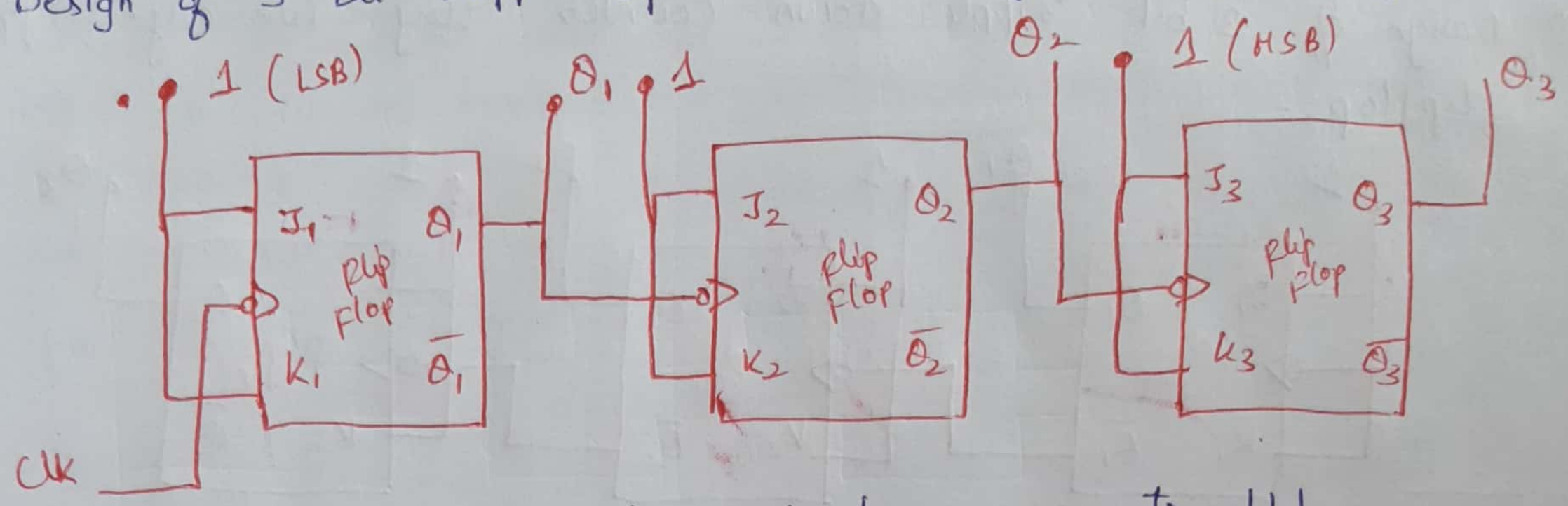

Waveform:-



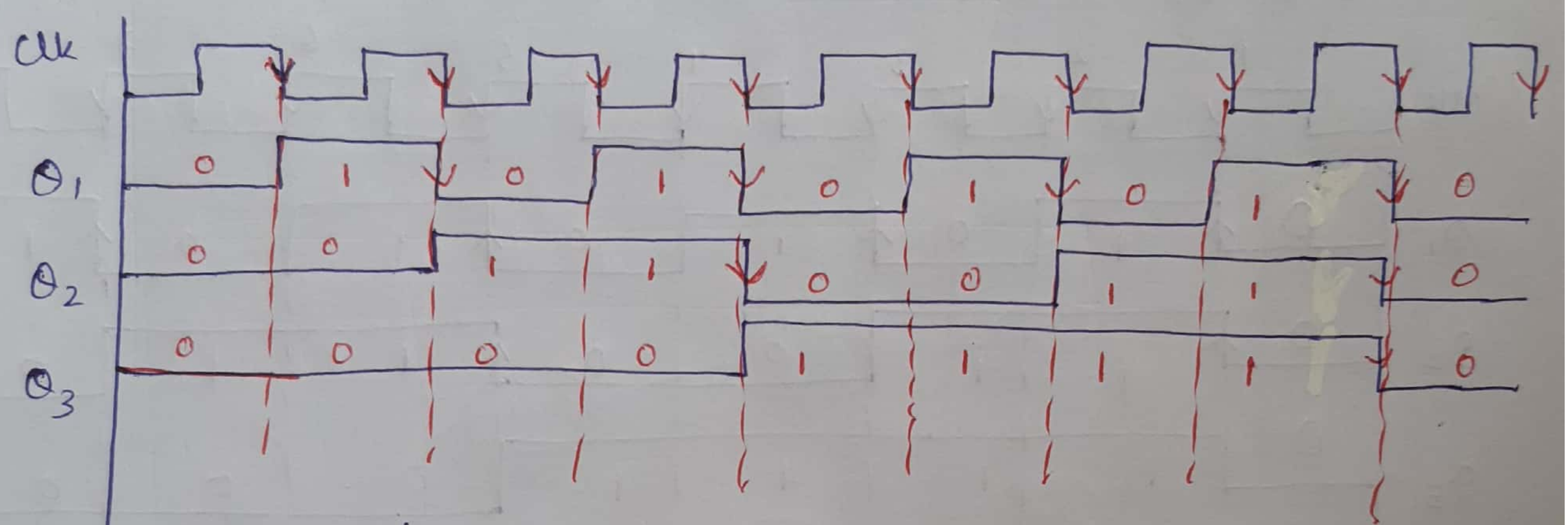

| clock | $Q_2$ | $Q_1$ | Decimal eq |
|---|---|---|---|
| Initially | 0 | 0 | 0 |
| 1st (↓) | 1 | 1 | 3 |
| ↓ | 1 | 0 | 2 |
| ↓ | 0 | 1 | 1 |
| ↓ | 0 | 0 | 0 |
| ↓ | 1 | 1 | 3 |

Scanned with CamScanner

Q Design of 3 bit ripple up counter using -ive edge triggered flipflop




→ 3 bit up counter → it count from 000 to 111.

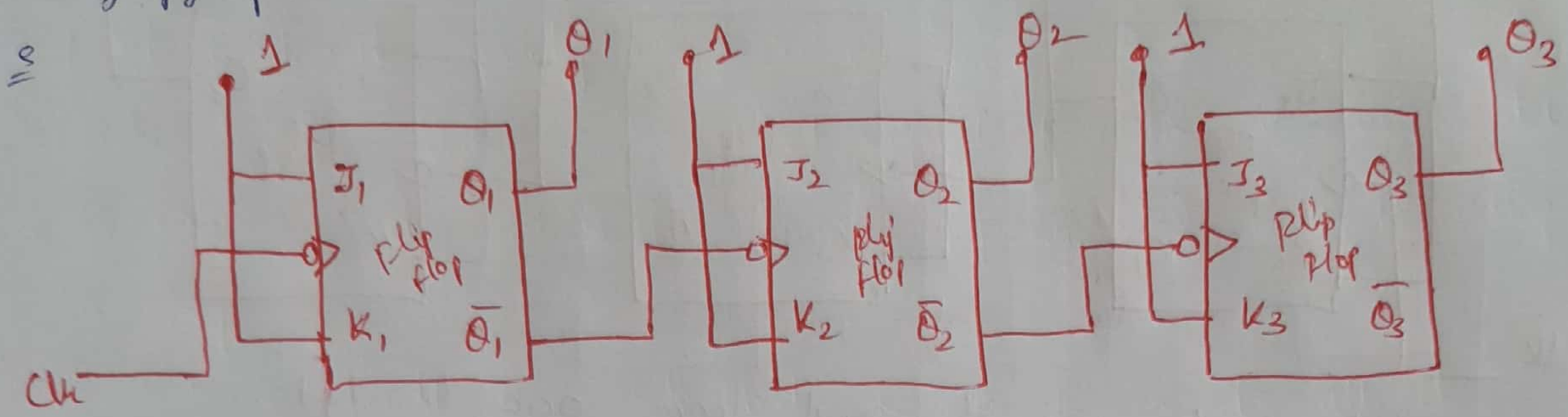



Truth Table.

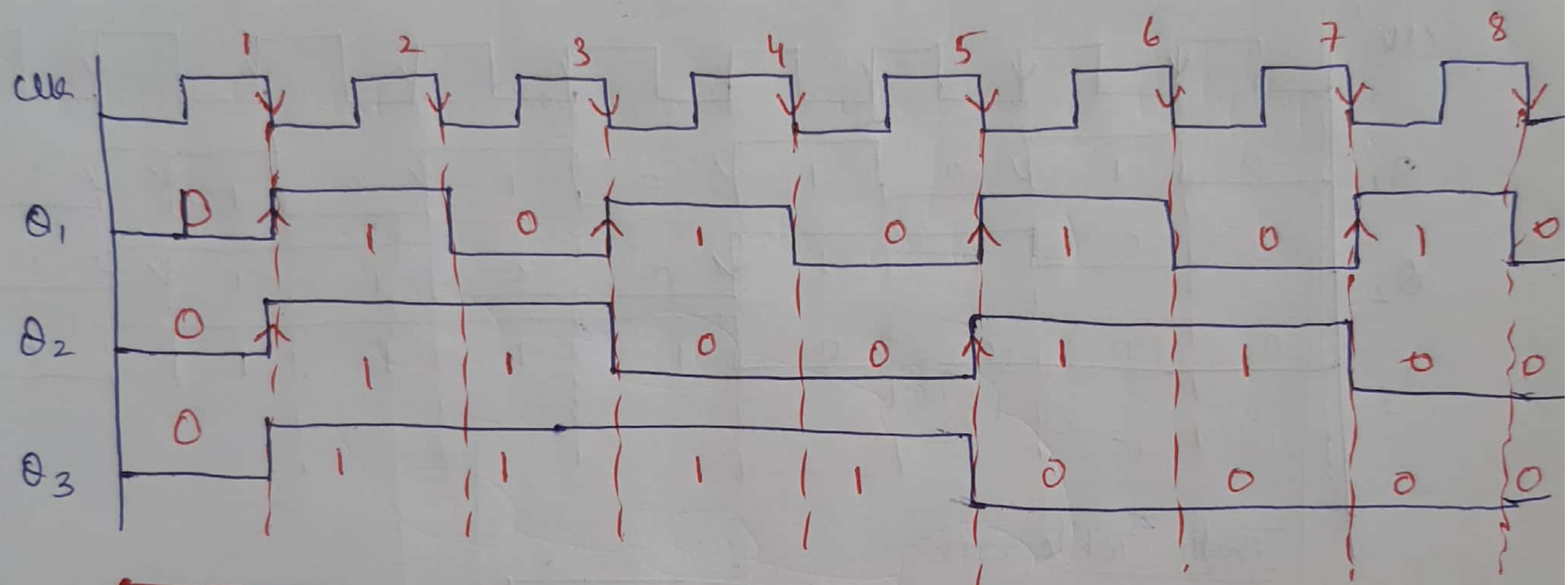
| Clock | $Q_3$ | $Q_2$ | $Q_1$ | Decimal eq |
|---|---|---|---|---|
| Initially | 0 | 0 | 0 | 0 |
| ↓ | 0 | 0 | 1 | 1 |
| ↓ | 0 | 1 | 0 | 2 |
| ↓ | 0 | 1 | 1 | 3 |
| ↓ | 1 | 0 | 0 | 4 |
| ↓ | 1 | 0 | 1 | 5 |
| ↓ | 1 | 1 | 0 | 6 |
| ↓ | 1 | 1 | 1 | 7 |
| ↓ | 0 | 0 | 0 | 0 |

Scanned with CamScanner

Ⓐ Design of 3 bit ripple down counter using -ive edge triggered flipflop:-



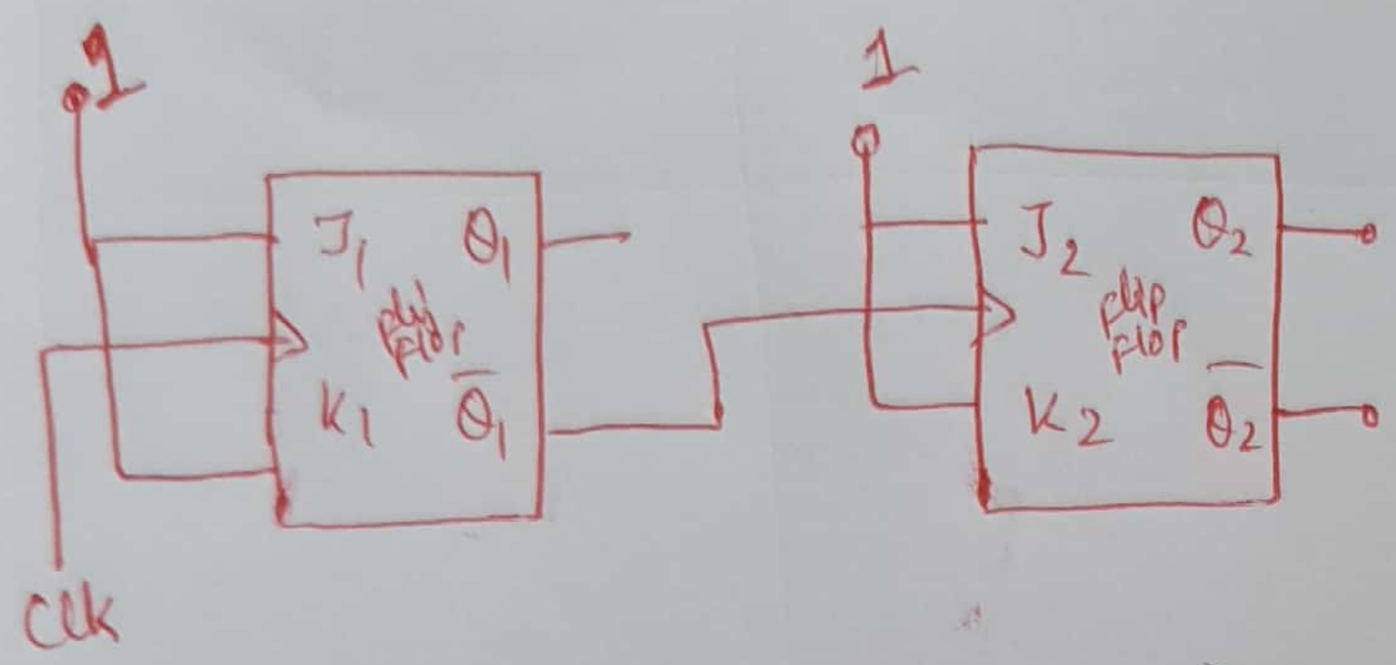

→ 3 bit down counter counts from 111 to 000.

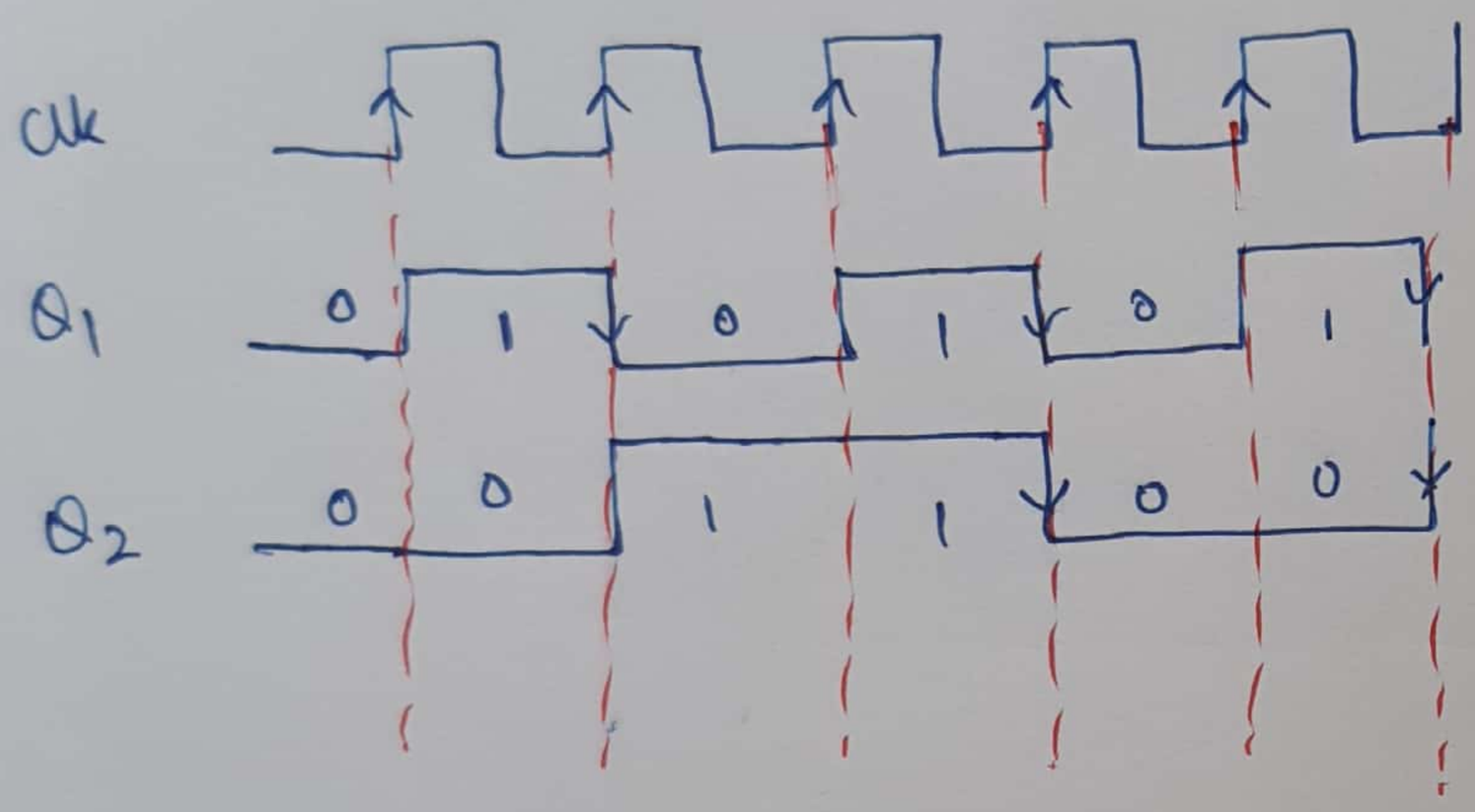



| Clk | $Q_3$ | $Q_2$ | $Q_1$ | Decimal eq. |
|---|---|---|---|---|
| Initially | 0 | 0 | 0 | 0 |
| 1st ↓ | 1 | 1 | 1 | 7 |
| ↓ | 1 | 1 | 0 | 6 |
| ↓ | 1 | 0 | 1 | 5 |
| ↓ | 1 | 0 | 0 | 4 |
| ↓ | 0 | 1 | 1 | 3 |
| ↓ | 0 | 1 | 0 | 2 |
| ↓ | 0 | 0 | 1 | 1 |
| ↓ | 0 | 0 | 0 | 0 |

Scanned with CamScanner

## Design of 2 bit up-Counter using +ve edge triggering

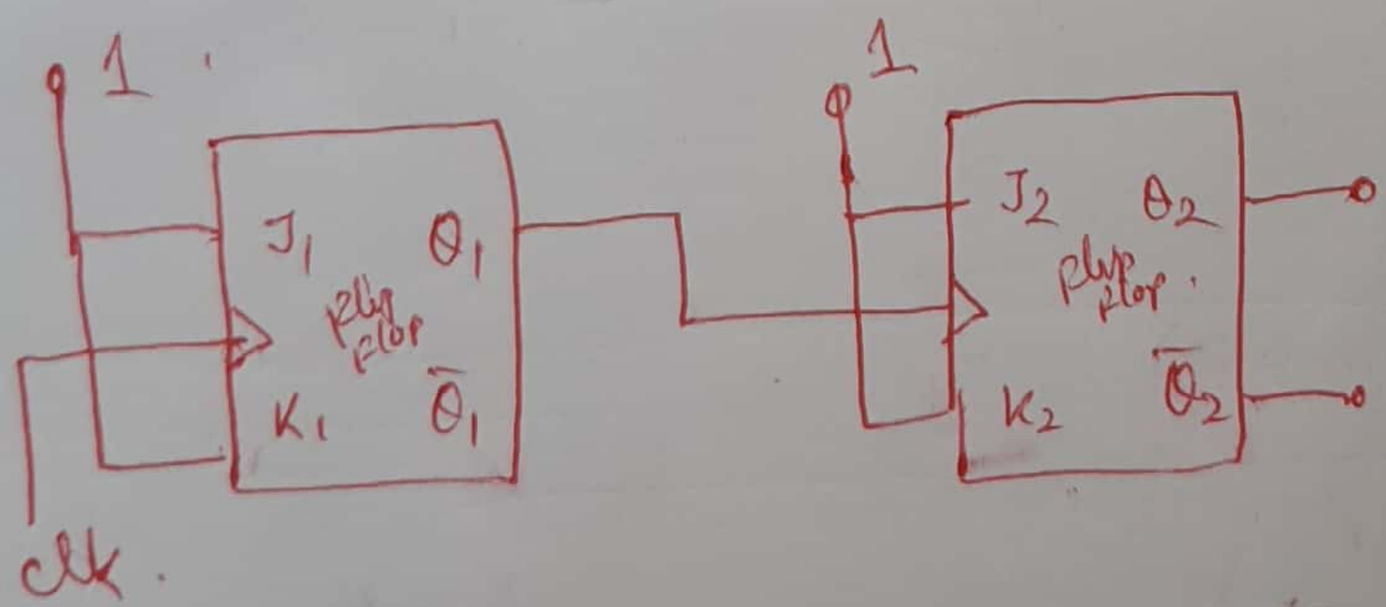


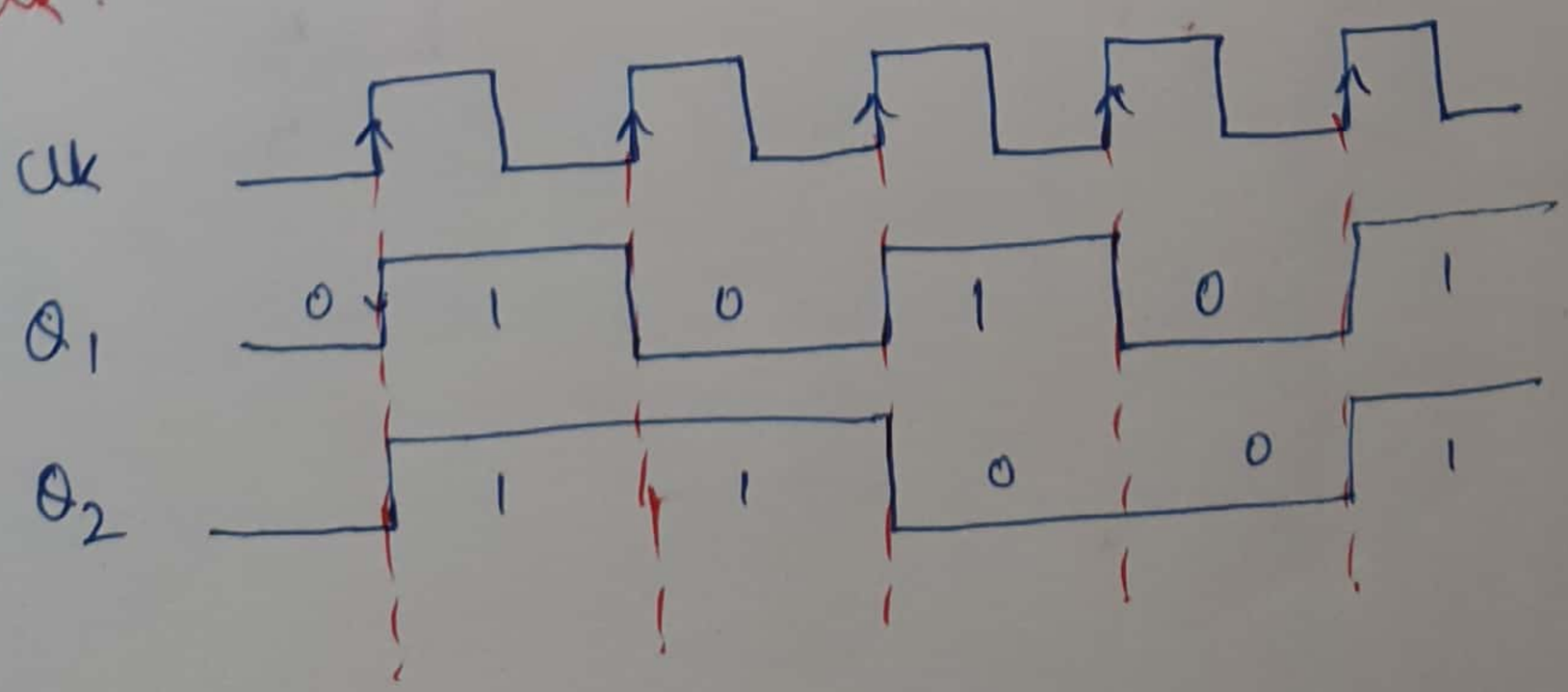




Truth Table

| Clk | $Q_2$ | $Q_1$ | Decimal eq |
|---|---|---|---|
| ↑ | 0 | 0 | 0 |
| ↑ | 0 | 1 | 1 |
| ↑ | 1 | 0 | 2 |
| ↑ | 1 | 1 | 3 |
| ↑ | 0 | 0 | 0 |

## Design of 2 bit down-Counter using +ive edge triggering




Truth Table

| Clk | $Q_2$ | $Q_1$ | Decimal eq |
|---|---|---|---|
| ↑ | 1 | 1 | 3 |
| ↑ | 1 | 0 | 2 |
| ↑ | 0 | 1 | 1 |
| ↑ | 0 | 0 | 0 |
| ↑ | 1 | 1 | 3 |

Scanned with CamScanner

## Q Design a 3 bit up / Down Ripple Counter?

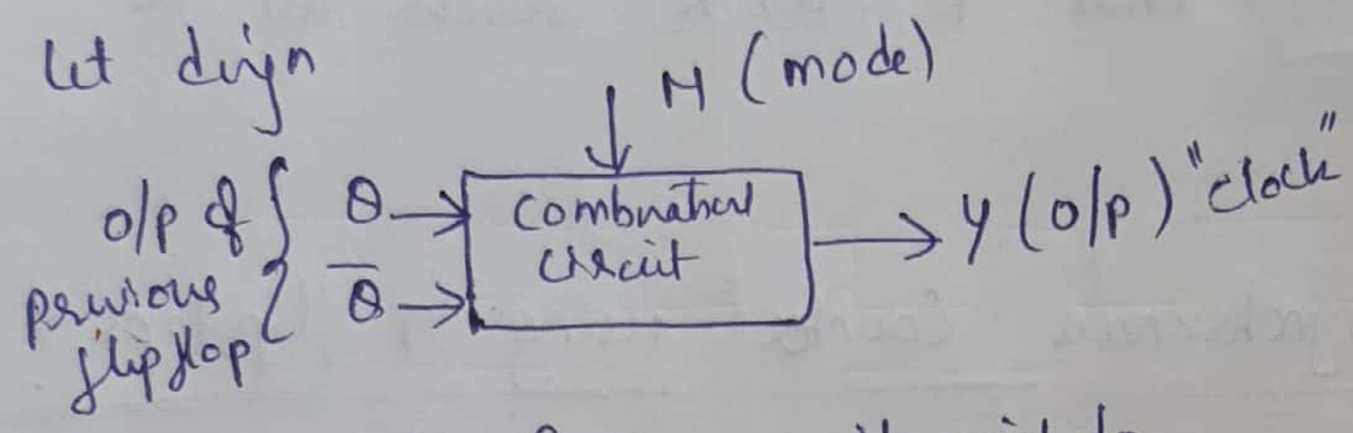
S → Till now we have designed up counters and down counters separately

→ In practice we need to combine both up and down counters.

→ Thus we use a mode control input (M) that selects either a up-counter & down counter.

→ A combinational circuit is required b/w each pair of flipflops.

→ Let design




→ whenever value of M = 0, it will do up counting

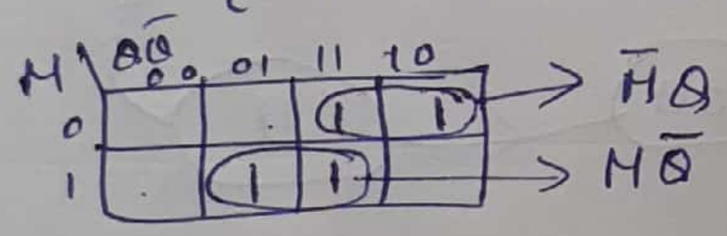
→ when value of M = 1, it will do down counting

→ use this logic $\left\{ \begin{array}{l} M=0 \; ; \; Q = Clk \\ M=1 \; ; \; \bar{Q} = Clk \end{array} \right\}$.

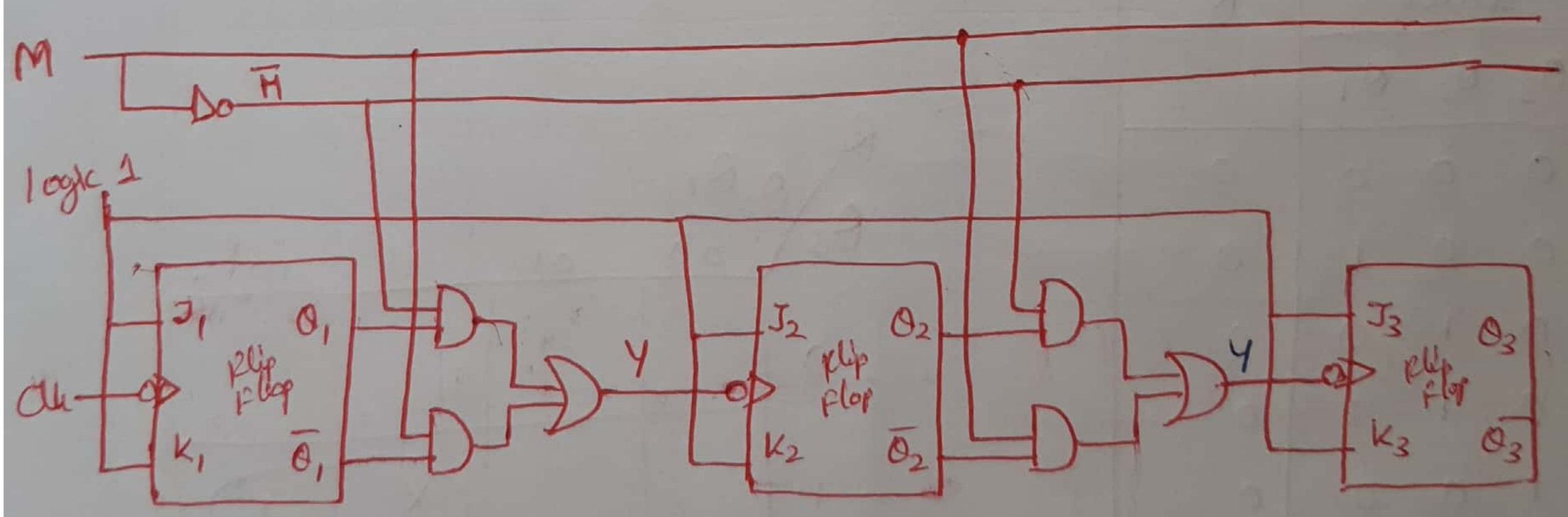
Truth Table

| M | Q | $\bar{Q}$ | Y |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

Solve by K-map




∴ $\boxed{Y = \bar{M}Q + M\bar{Q}}$




→ If M = 1, $\bar{M} = 0$ ; then $Y = \bar{Q}_1$, $Y = \bar{Q}_2$ hence it will perform down counting operation.

→ If M = 0, $\bar{M} = 1$ ; then $Y = Q_1$, $Y = Q_2$ hence it will perform up counting operation.

## Design of Asynchronous Counters:-

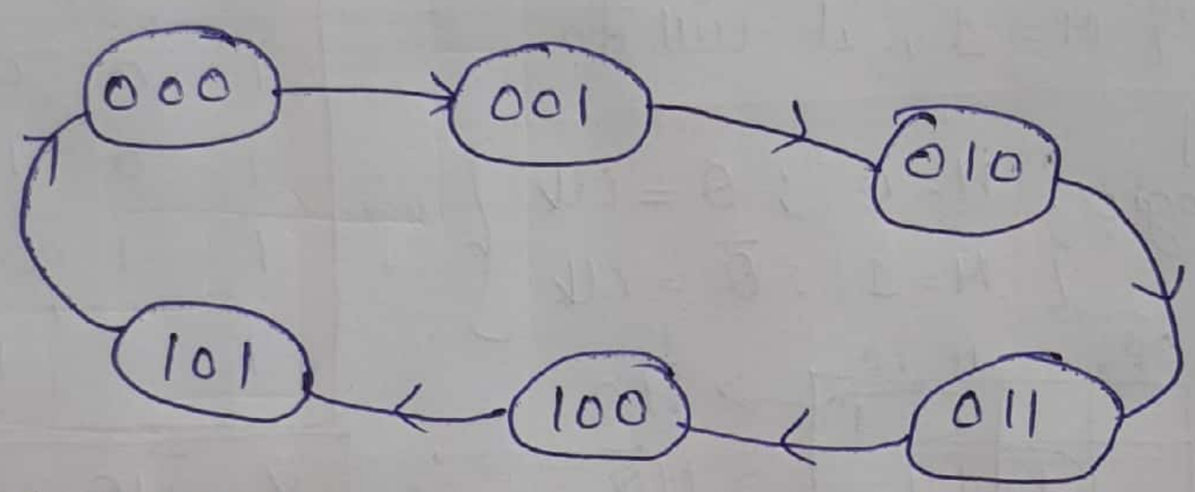
→ First write the counting sequence.

→ Tabulate the value of reset signal R for various states of counters and obtain the minimal expression of R & $\overline{R}$ using K-map.

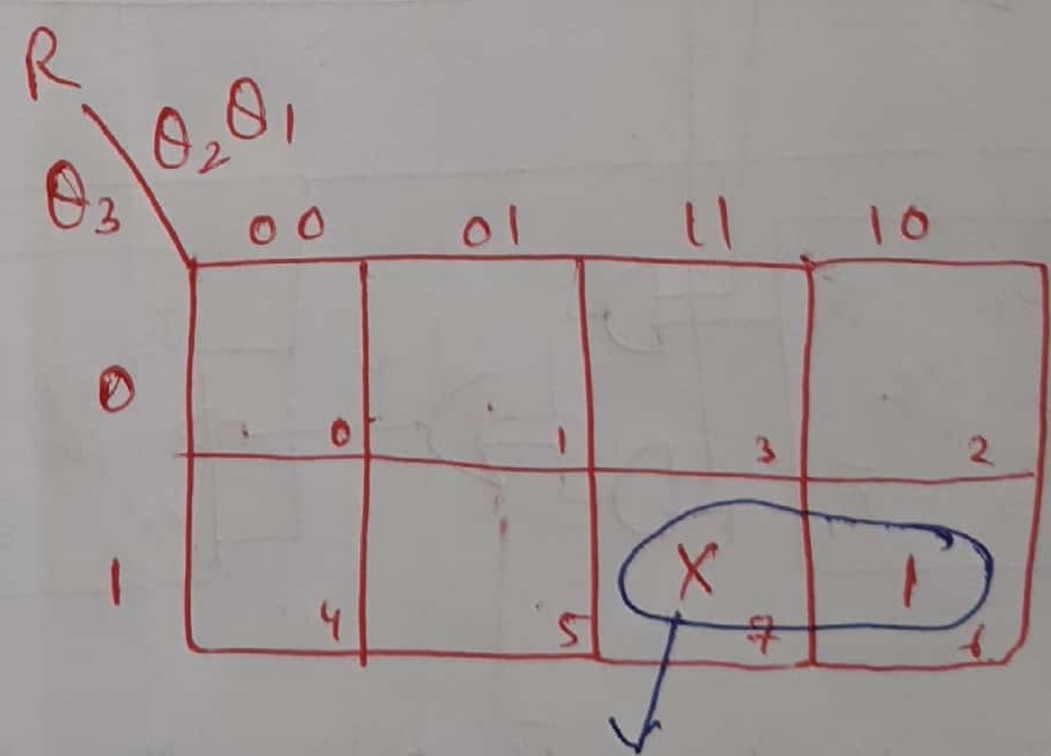
→ provide a feedback such that R (or) $\overline{R}$ resets all the flipflops.

### Q) Design a Mod-6 asynchronous Counter using T flipflops?

A: Mod-6 Counter has 6 states (0 to 5)




| States | | | R |
|---|---|---|---|
| $Q_3$ | $Q_2$ | $Q_1$ | |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | X |

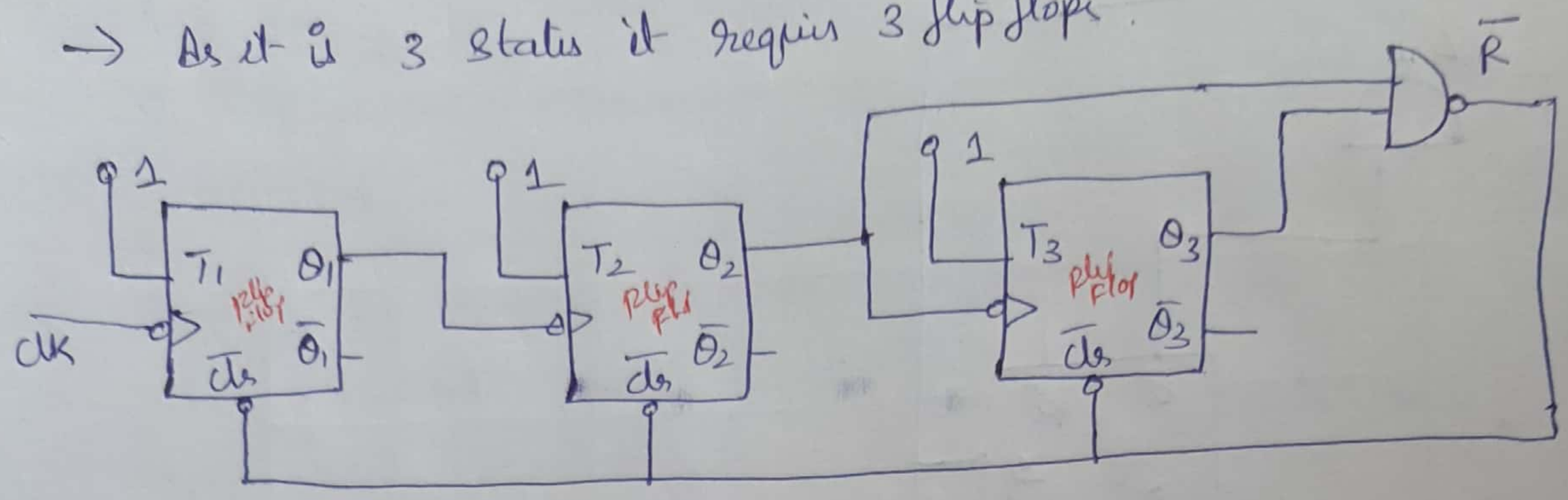



$R = Q_3 Q_2$.

for active low $\overline{R}$ is used.

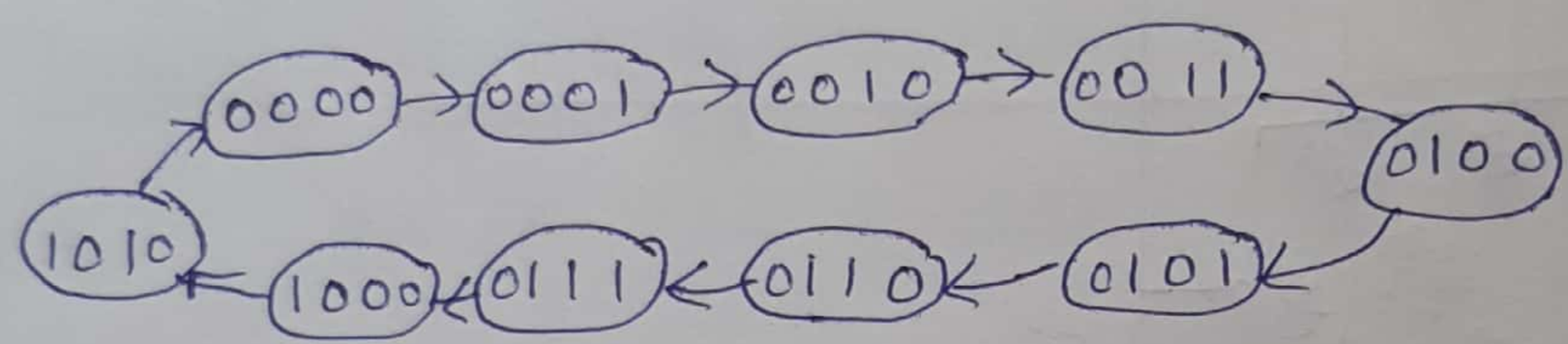
$\overline{R} = \overline{Q_3 Q_2}$

→ As it is 3 states it requires 3 flip flops.

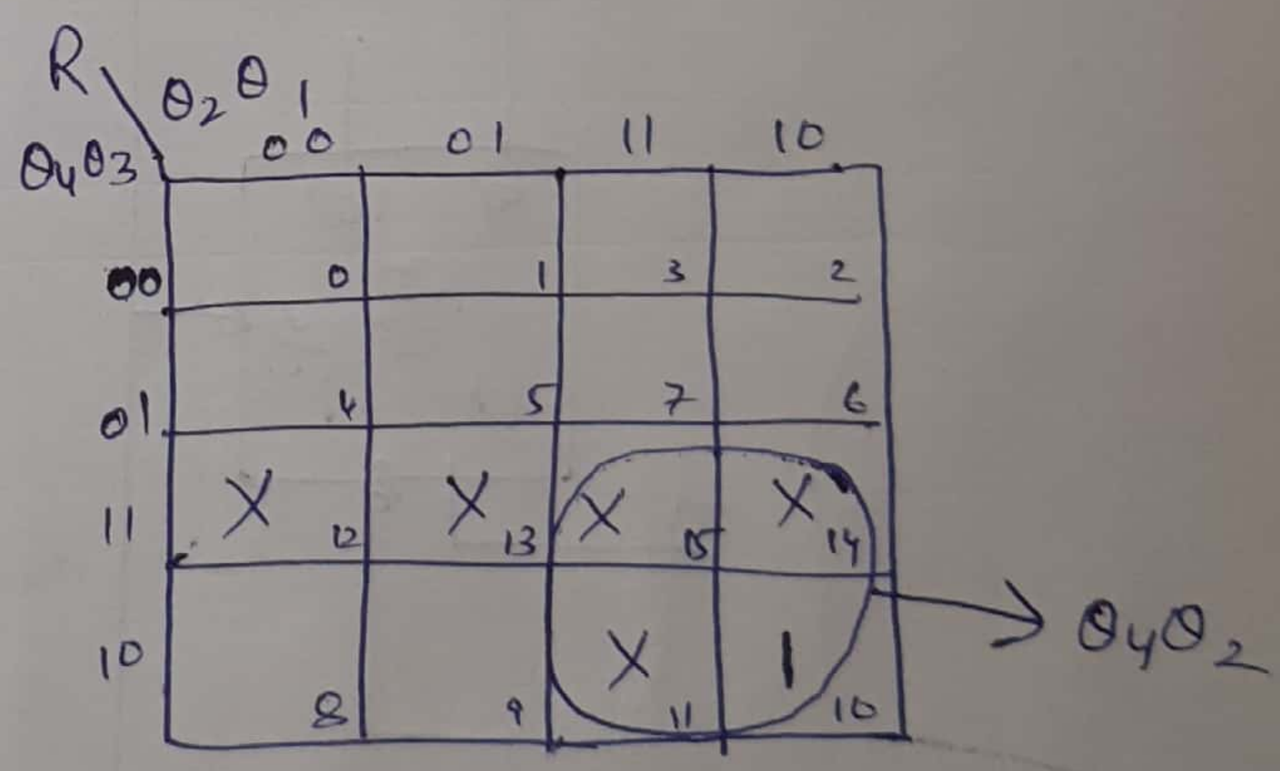



## ⑥ Design a Mod-10 asynchronous Counter using T flipflops.

Mod-10 Counter has 10 states (0 to 9)




| States $Q_4$ $Q_3$ $Q_2$ $Q_1$ | R |
|---|---|
| 0 0 0 0 | 0 |
| 0 0 0 1 | 0 |
| 0 0 1 0 | 0 |
| 0 0 1 1 | 0 |
| 0 1 0 0 | 0 |
| 0 1 0 1 | 0 |
| 0 1 1 0 | 0 |
| 0 1 1 1 | 0 |
| 1 0 0 0 | 0 |
| 1 0 0 1 | 0 |
| 1 0 1 0 | 1 |
| 1 0 1 1 | X |
| 1 1 0 0 | X |
| 1 1 0 1 | X |
| 1 1 1 0 | X |
| 1 1 1 1 | X |

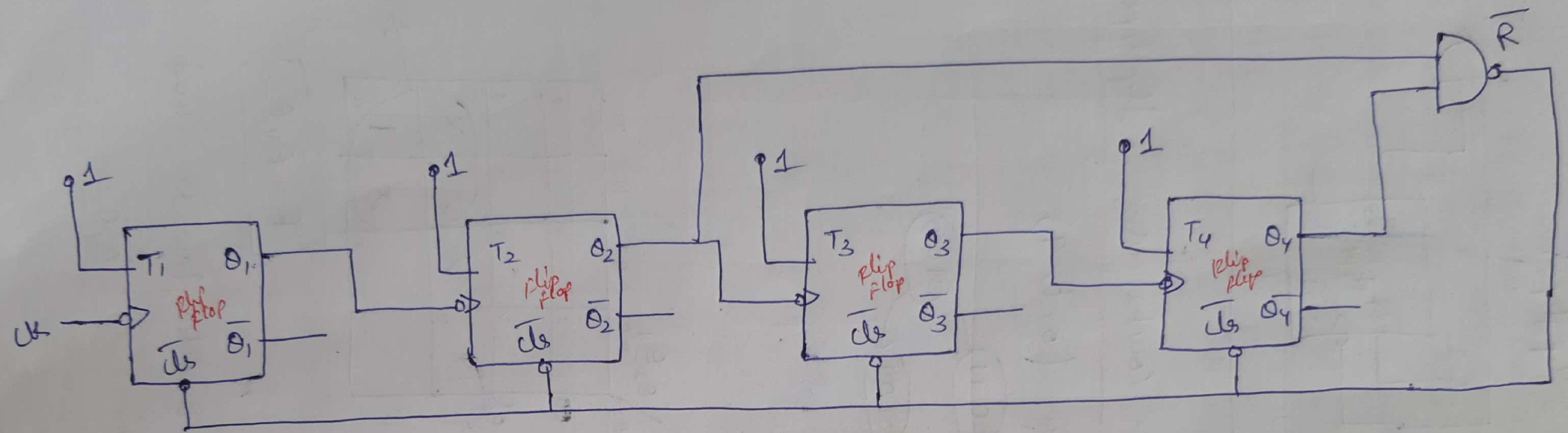



$$\boxed{R = Q_4 Q_2}$$

for active low $\overline{R}$ is used

$$\overline{R} = \overline{Q_4 Q_2}$$

$\overline{R} = \overline{Q_4 Q_2}$




Scanned with CamScanner

# Design of Synchronous Counters :-

## Steps to Design Synchronous Counter :-

1. Identify the number of bits and flip Flop
2. Write Excitation Table of flip flop
3. Make State diagram and State Table
4. Solve Boolean Expression.
5. Make / Draw the circuit.

## ① Design a 2-Bit Synchronous Counter by JK Flip Flop.

Step 1 :- Here 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 :- Make State diagram. and State table.

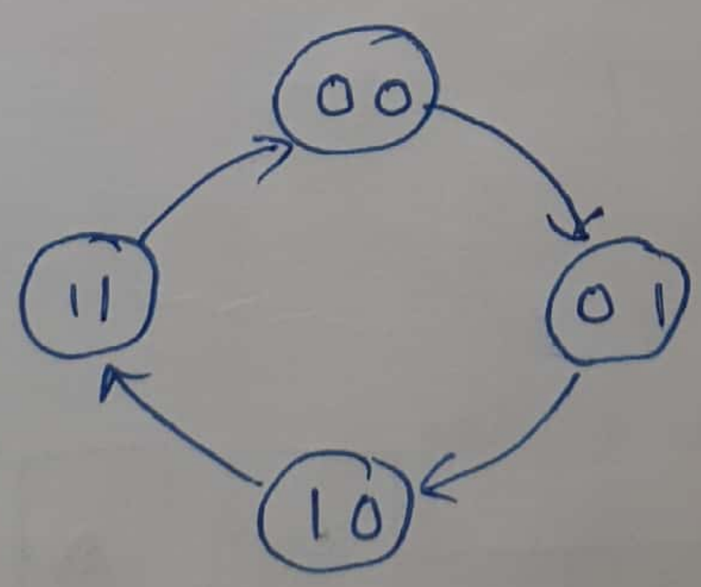
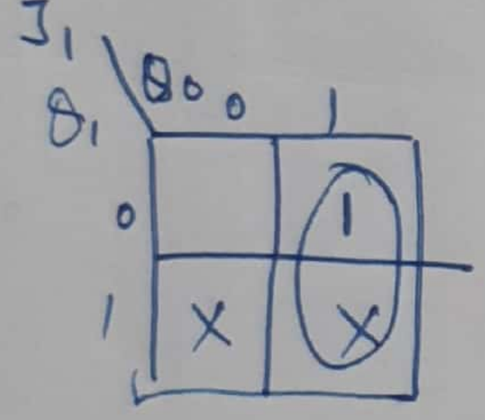



fig: State diagram

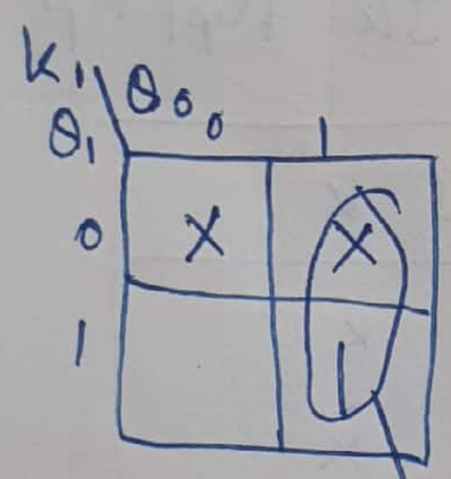
State Table :- has present state, Next state, inputs of flip flop.

| Present state $Q_1$ $Q_0$ | Next state $Q_1^+$ $Q_0^+$ | $J_1$ $K_1$ | $J_0$ $K_0$ |
|---|---|---|---|
| 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 |

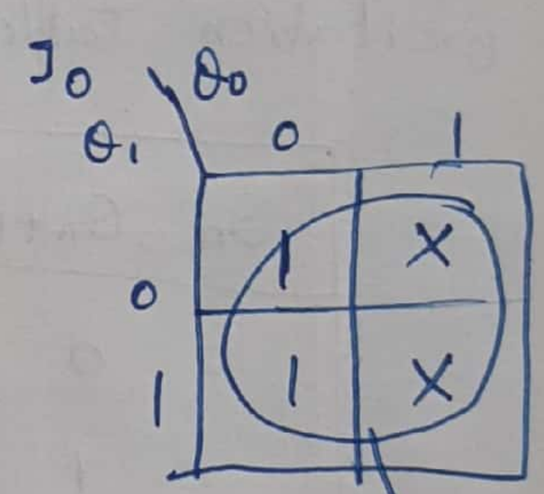
Step 4 :- find boolean expression



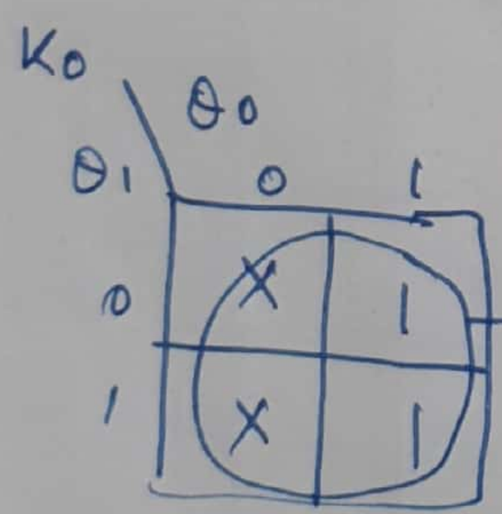

$J_1 = Q_0$




$K_1 = Q_0$

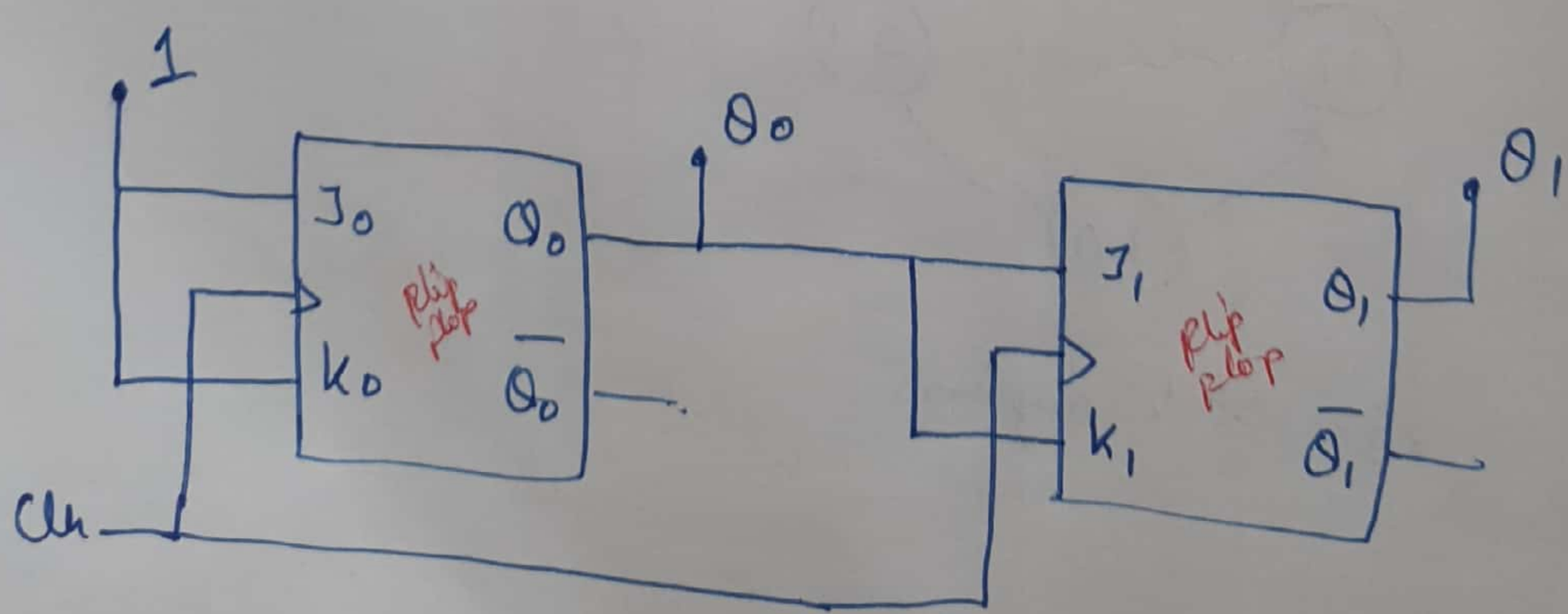



$J_0 = 1$




$K_0 = 1$

Step 5 :- Make Circuit



## (8) Design of 3 bit Synchronous Counter using T flipflops.

**Step 1 :-** Here n = 3 bits, Flip Flop = T flipflop.

**Step 2 :-** Excitation Table of T flipflop.

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

**Step 3 :-** Drawing State diagram & State Table.

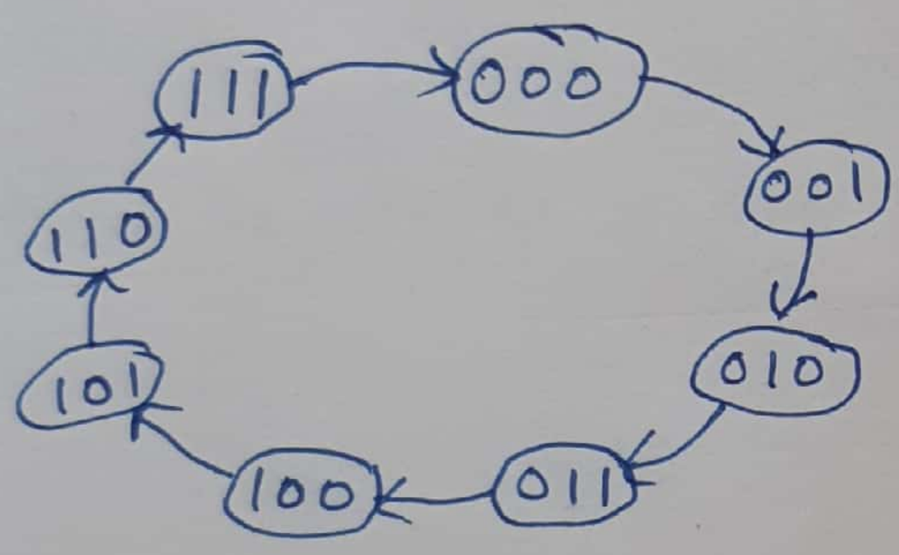


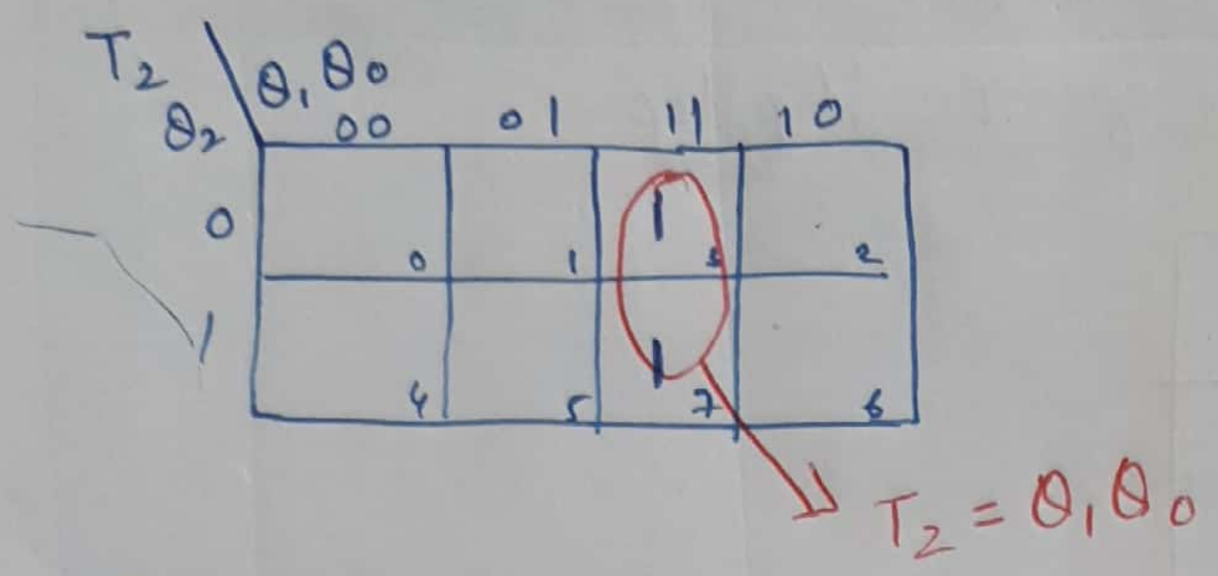

fig :- State diagram

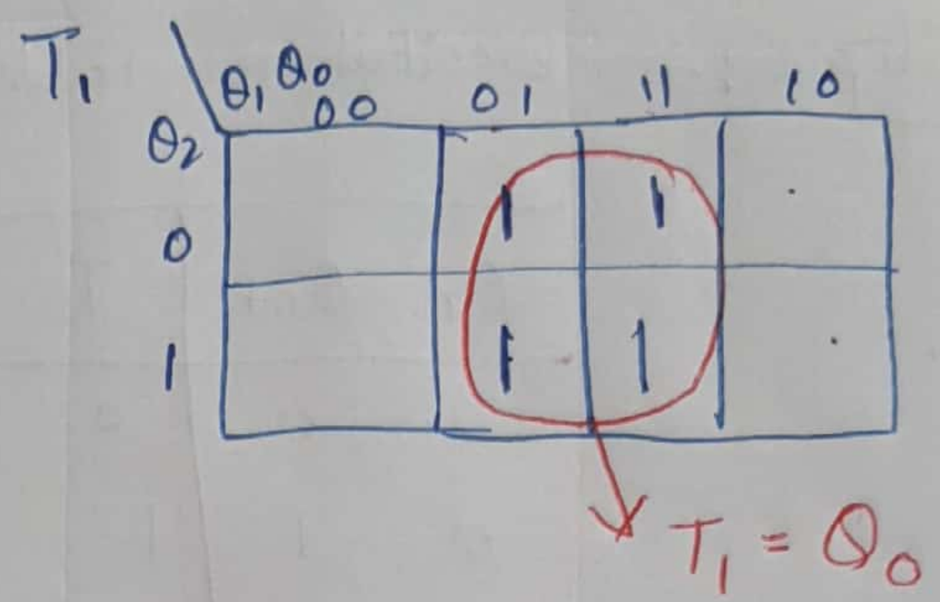
**State Table.**

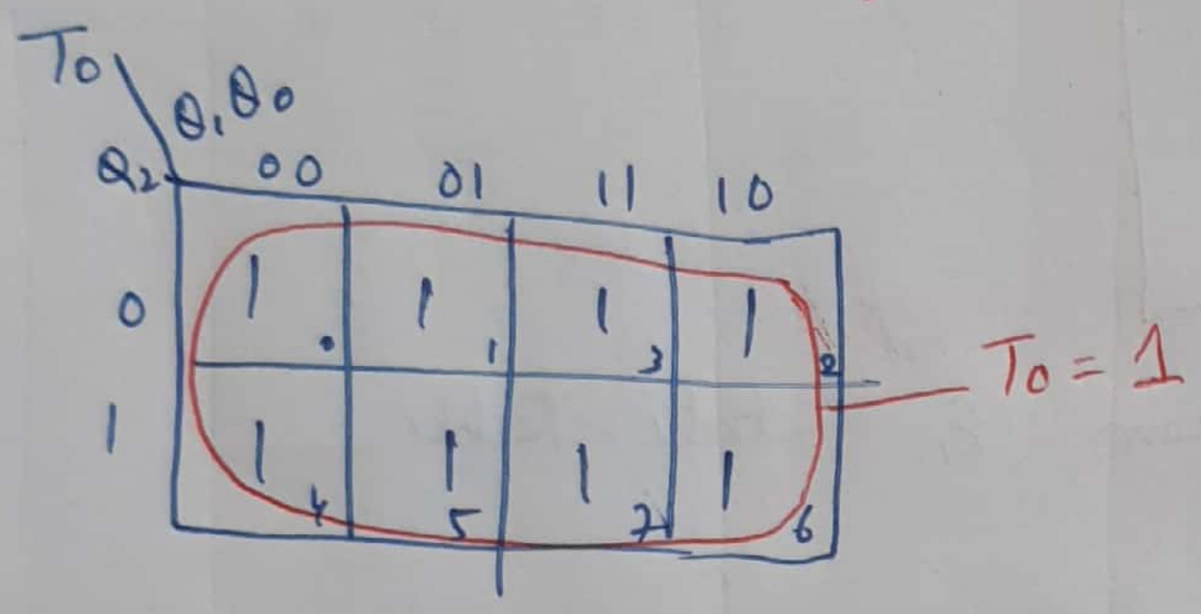
| Present State | | | Next State | | | Inputs | | |
|---|---|---|---|---|---|---|---|---|
| $Q_2$ | $Q_1$ | $Q_0$ | $Q_2^+$ | $Q_1^+$ | $Q_0^+$ | $T_2$ | $T_1$ | $T_0$ |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

Scanned with CamScanner

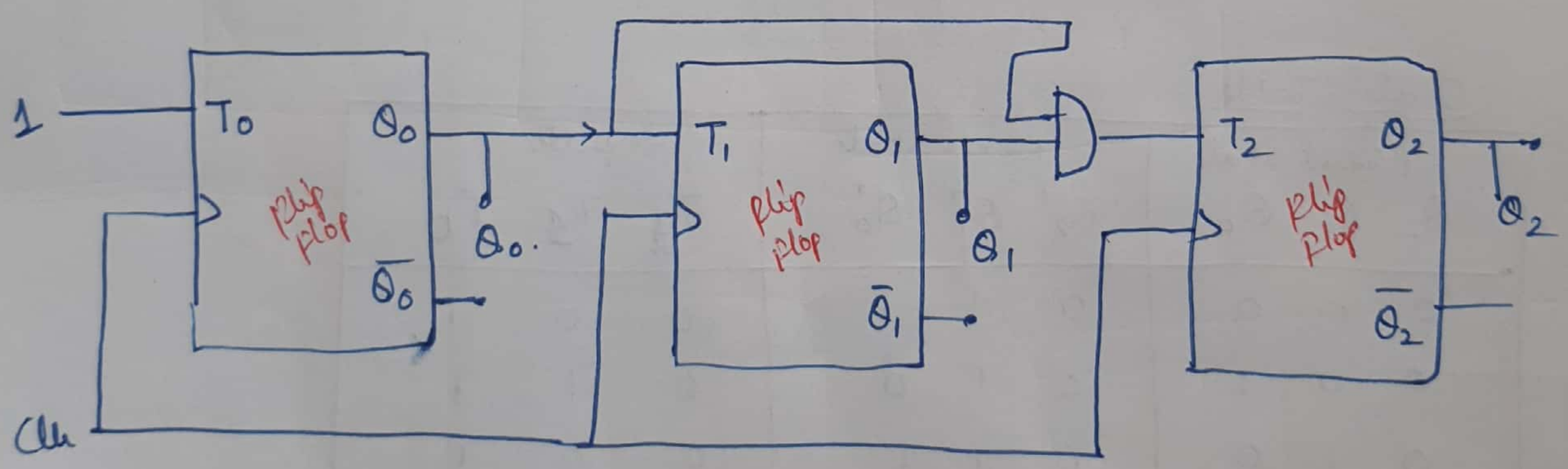
## Step 4 :- find boolean expression




## Step 5 :- Make circuit



## Design of 3 bits Synchronous up/down counter:-

Let us use a "T" flip flop for designing the Synchronous up/down counter.

**Step 1:** $n = 3$ ; flip flop = "T"

**Step 2:-** Excitation Table of T flip flop.

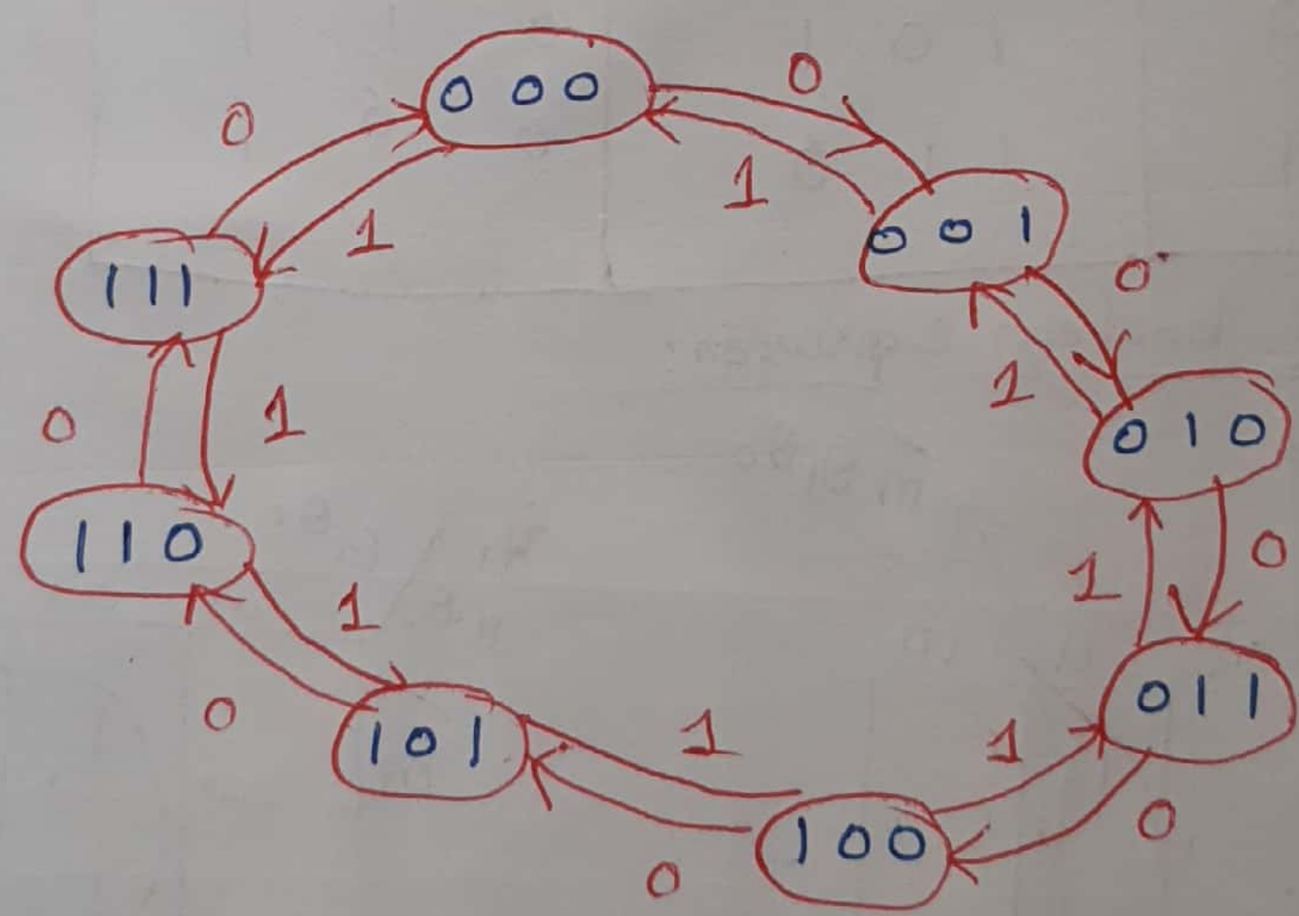
| $Q_n$ | $Q_{n+1}$ | T |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**Step 3:-** Drawing the State diagram & State Table.

→ Let us consider a mode input say "m".

where $m = 0 \rightarrow$ up counter

$m = 1 \rightarrow$ Down counter.

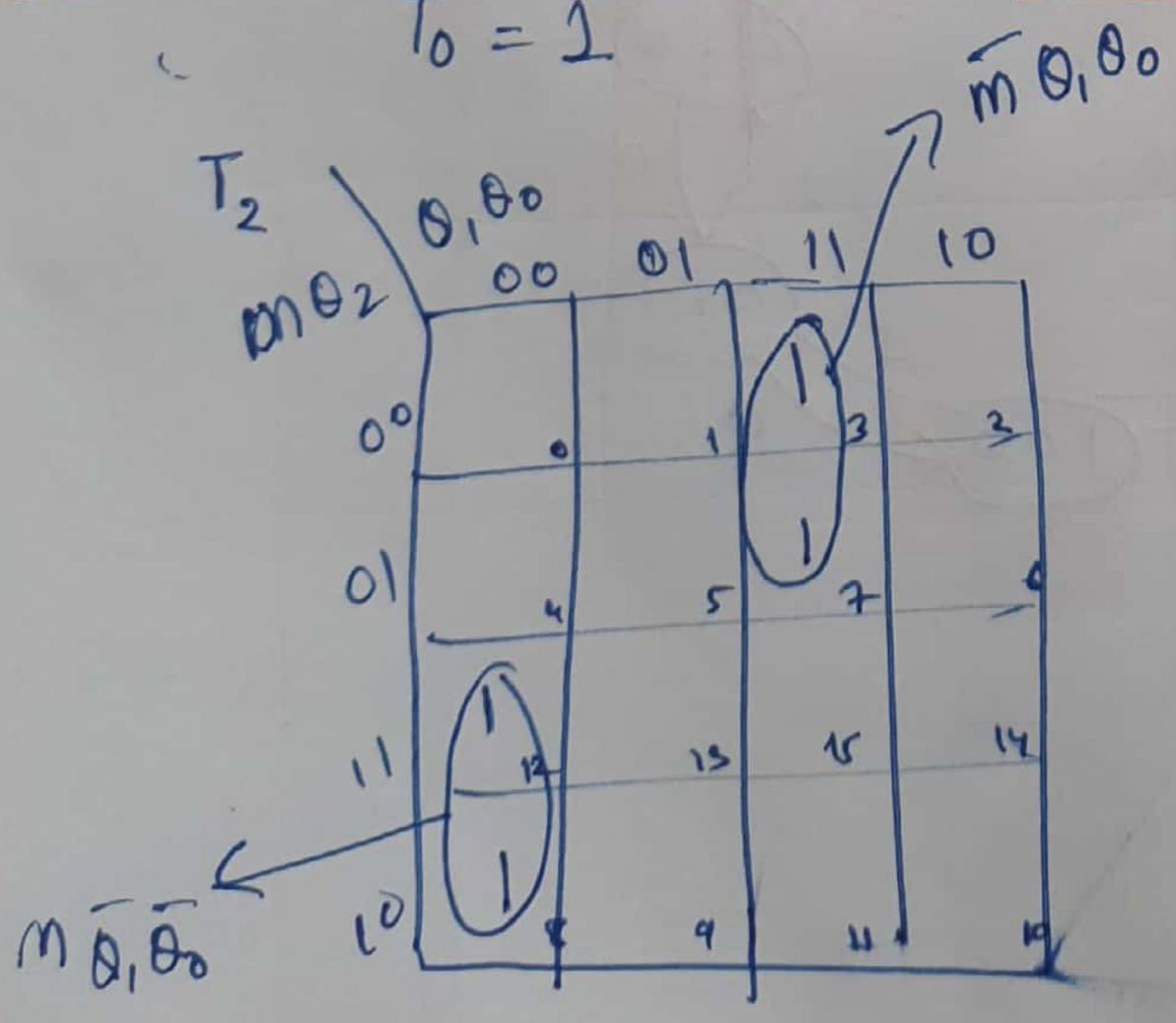


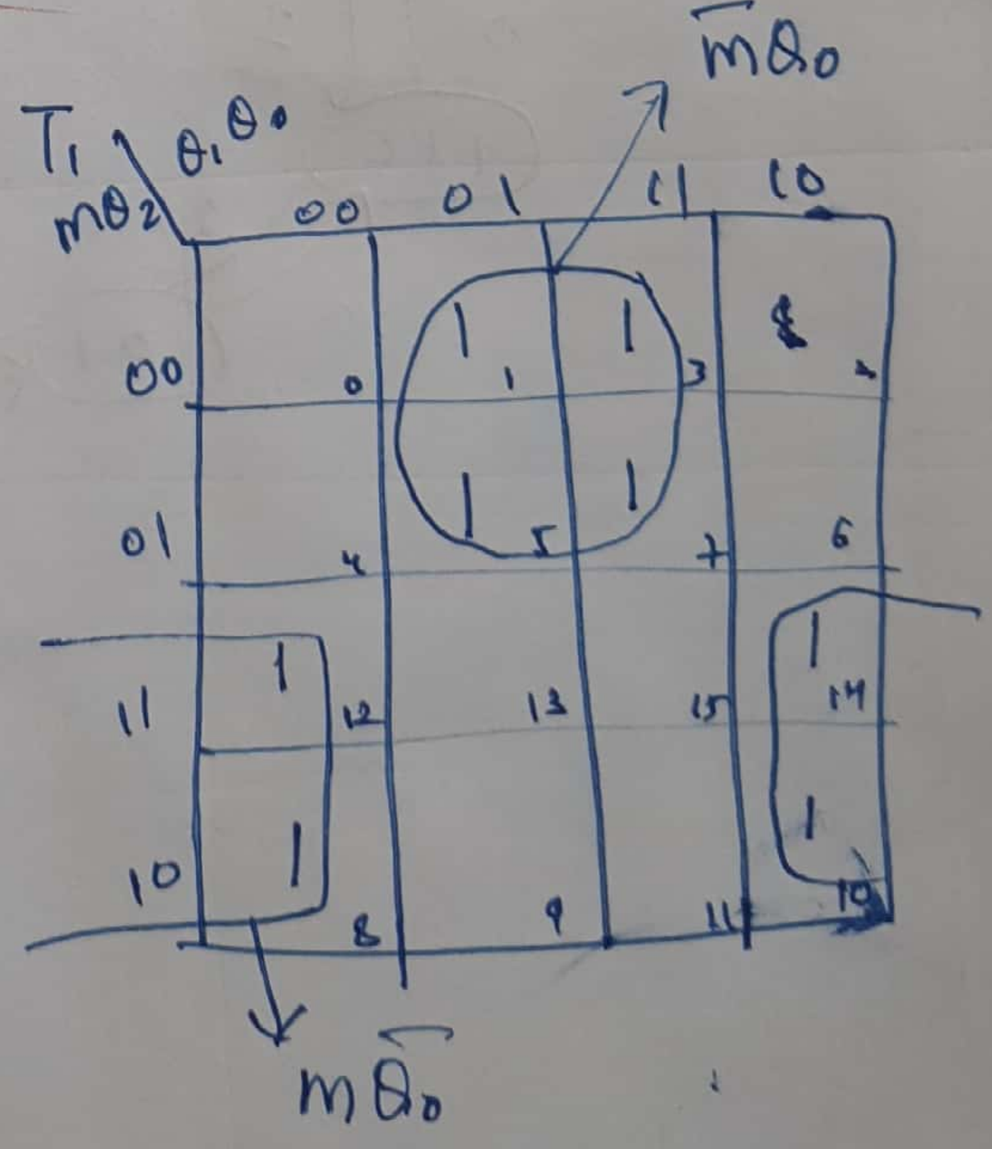
State Table

| mode | Present State | | | Next State | | | Inputs | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $m$ | $Q_2$ | $Q_1$ | $Q_0$ | $Q_2^+$ | $Q_1^+$ | $Q_0^+$ | $T_2$ | $T_1$ | $T_0$ | |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | Up counting |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | Down counting |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 6 | 1 | |

Step 4:- Realize the boolean expression:

$T_0 = 1$

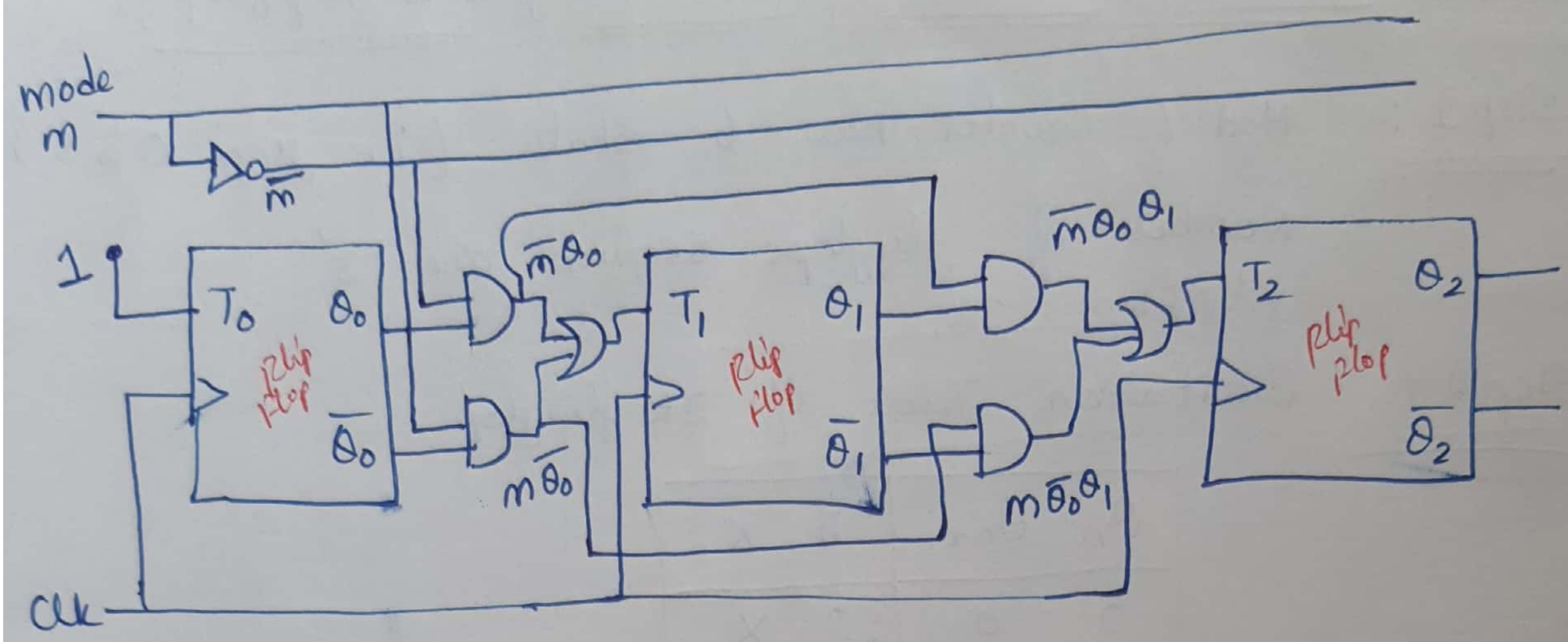



$$T_2 = \bar{m}\,Q_1 Q_0 + m\,\bar{Q}_1 \bar{Q}_0$$

$$\boxed{T_1 = m\,\bar{Q}_0 + \bar{m}\,Q_0}$$

Step 5:- make the circuit.



Q) Design a Synchronous MOD-6 counter using J-K flipflop?

Step 1 :- Mod-6 counter has 6 states (i.e from 0 to 5)

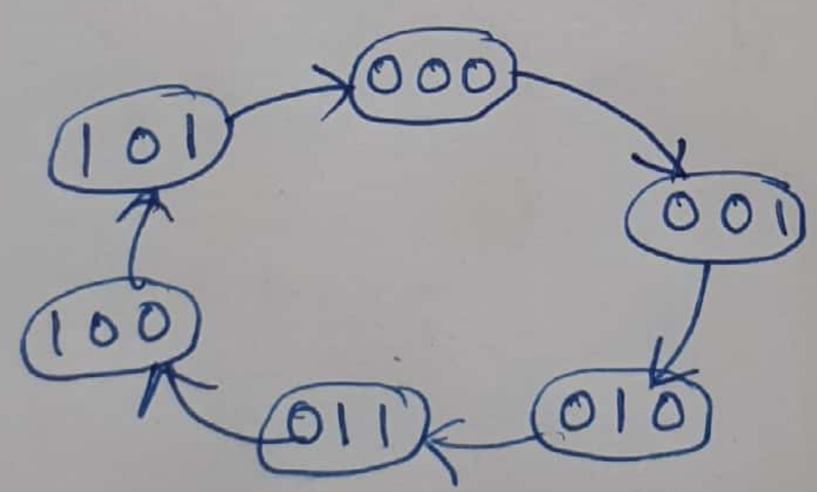
Number of flipflops required are '3'.

FlipFlop : JK

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:- Draw state diagram & State Table




State Table.

| Present State $Q_2$ | $Q_1$ | $Q_0$ | Next State $Q_2^+$ | $Q_1^+$ | $Q_0^+$ | Inputs $J_2$ | $K_2$ | $J_1$ | $K_1$ | $J_0$ | $K_0$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | X | 0 | X | 1 | X |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | X | 1 | X | X | 1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | X | X | 0 | 1 | X |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | X | X | 1 | X | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | X | 0 | 0 | X | 1 | X |
| 1 | 0 | 1 | 0 | 0 | 0 | X | 1 | 0 | X | X | 1 |
| 1 | 1 | 0 | X | X | X | X | X | X | X | X | X |
| 1 | 1 | 1 | X | X | X | X | X | X | X | X | X |

Scanned with CamScanner

## Step 4 :- Realize the boolean expression.

$J_2$

| $Q_2 \backslash Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | | | 1 | |
| 1 | X | X | X | X |

$Q_1Q_0$

$$J_2 = Q_1 Q_0$$

$K_2$

| $Q_2 \backslash Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | X | X | X | X |
| 1 | | 1 | X | X |

$Q_0$

$$K_2 = Q_0$$

$J_1$

| $Q_2 \backslash Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | | 1 | X | X |
| 1 | | | X | X |

$\overline{Q_2}\, Q_0$

$$J_1 = \overline{Q_2}\, Q_0$$

$K_1$

| $Q_2 \backslash Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | X | X | 1 | |
| 1 | X | X | X | X |

$Q_0$

$$K_1 = Q_0$$

$J_0$

| $Q_2 \backslash Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 1 | X | X | 1 |
| 1 | 1 | X | X | X |

1

$$J_0 = 1$$

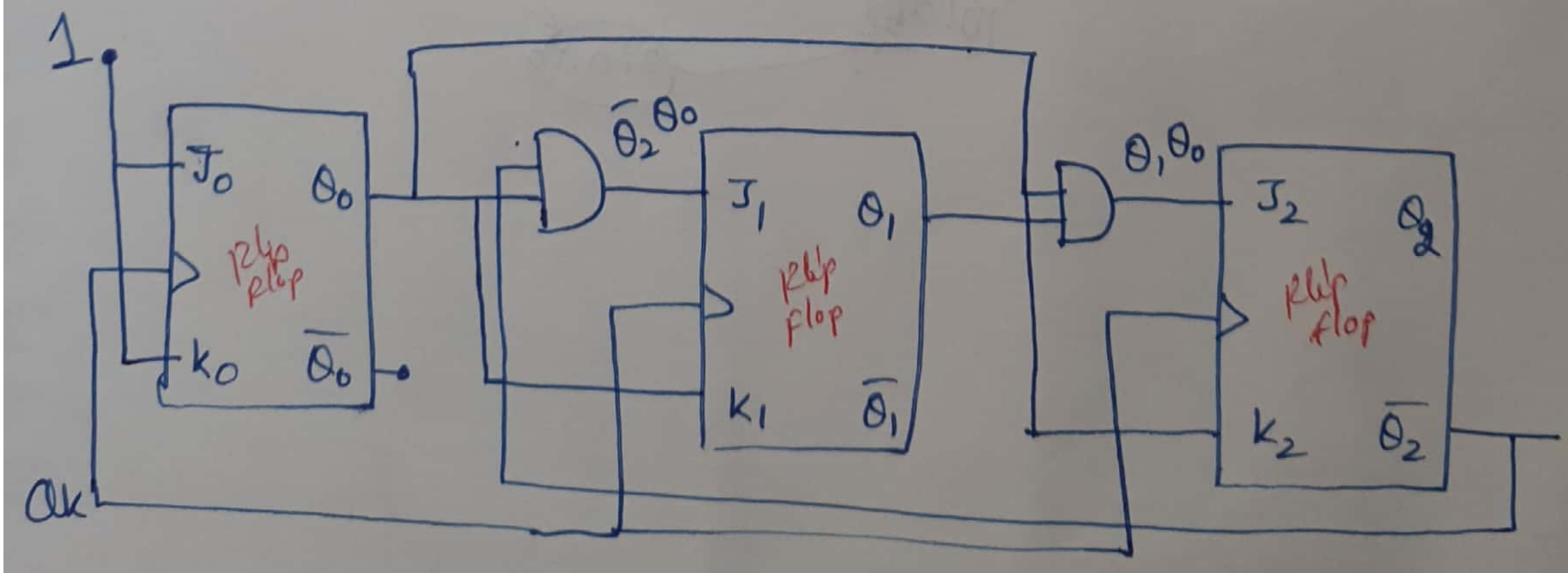
$K_0$

| $Q_2 \backslash Q_1Q_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | X | 1 | 1 | X |
| 1 | X | 1 | X | X |

1

$$K_0 = 1$$

## Step 5 :- Draw the circuit



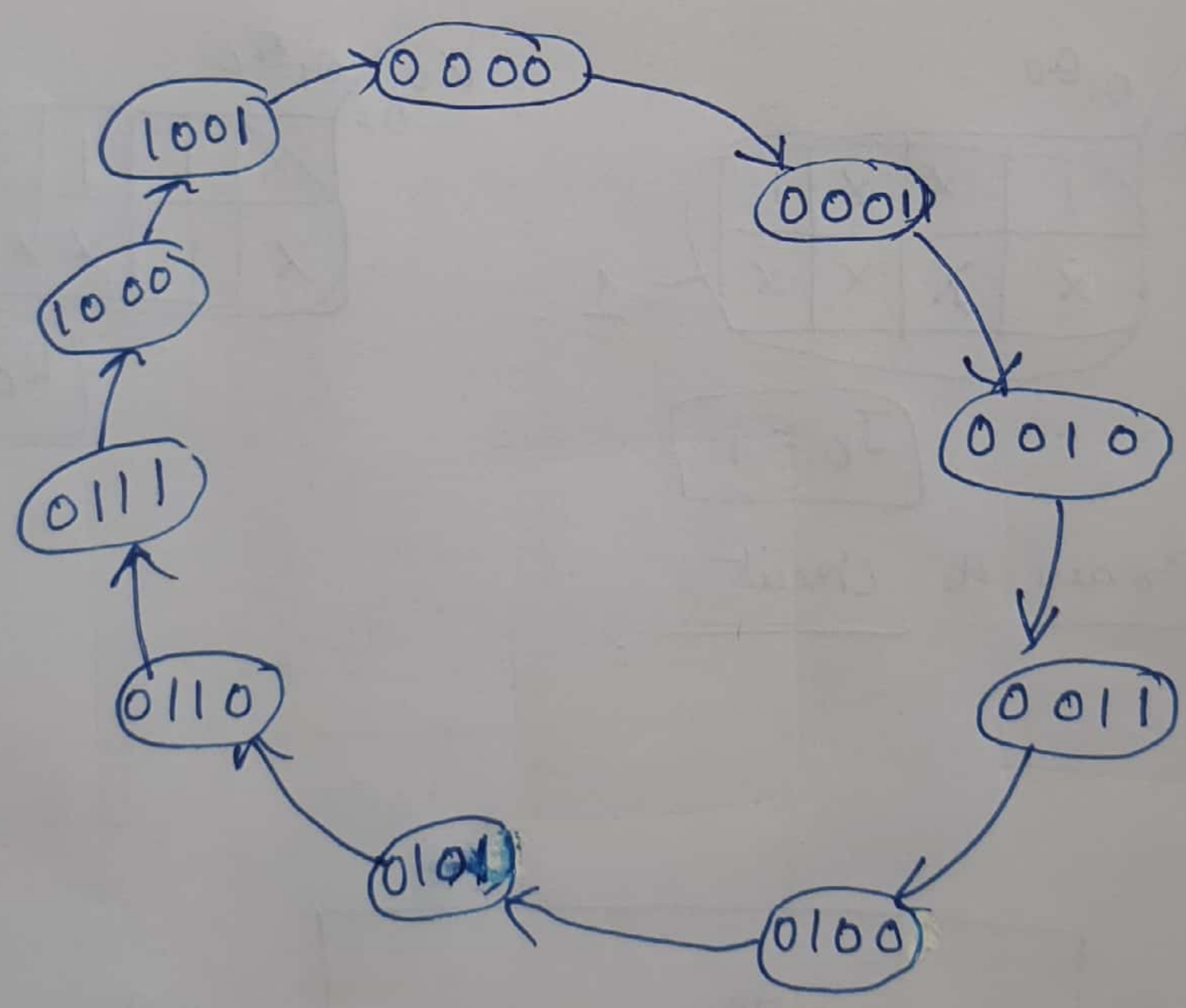
## ⑧ Design a Synchronous MOD-10 Counter using T flip flop ?

**Step 1 :-** MOD-10 counter has 10 states (0 - 9)
numbers of flipflops required are '4'
Flip Flop - "T"

**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 |

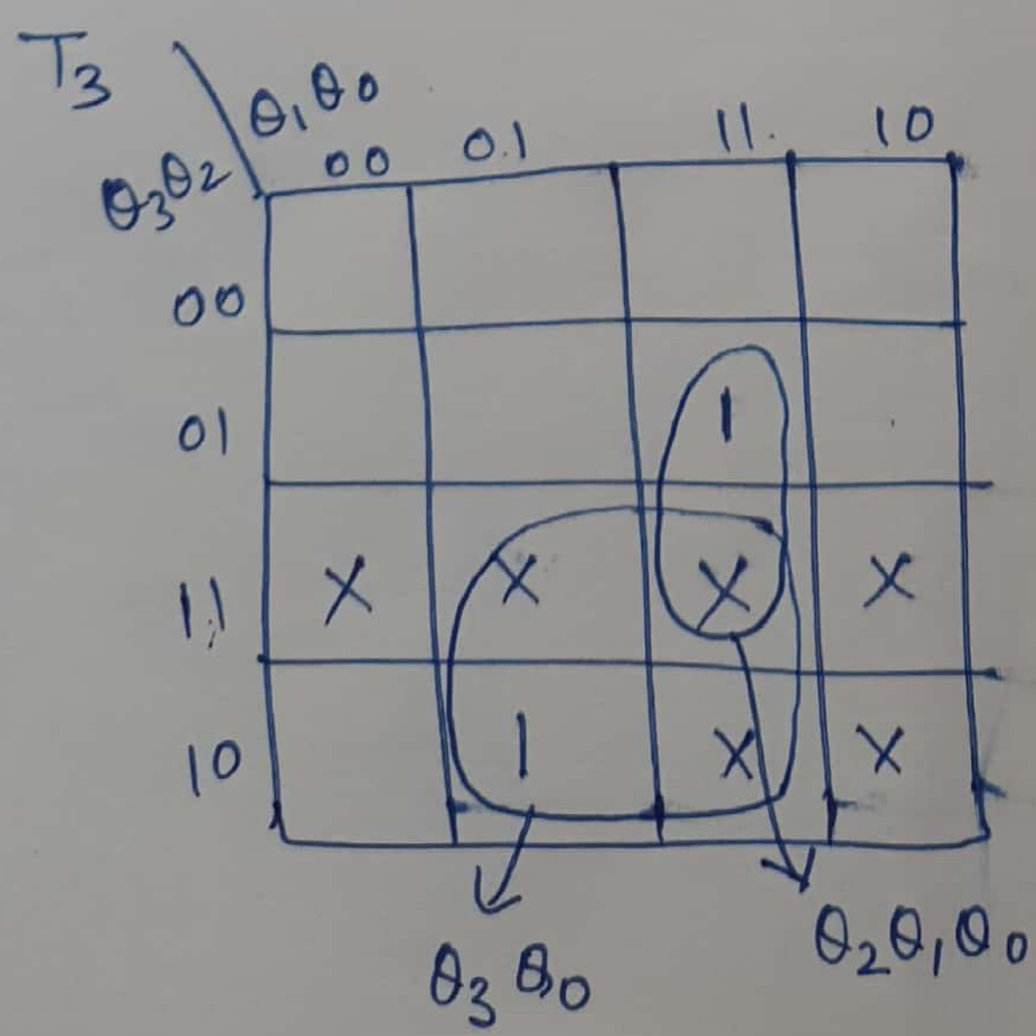
**Step 3 :-** Drawing State Diagram & State Table.

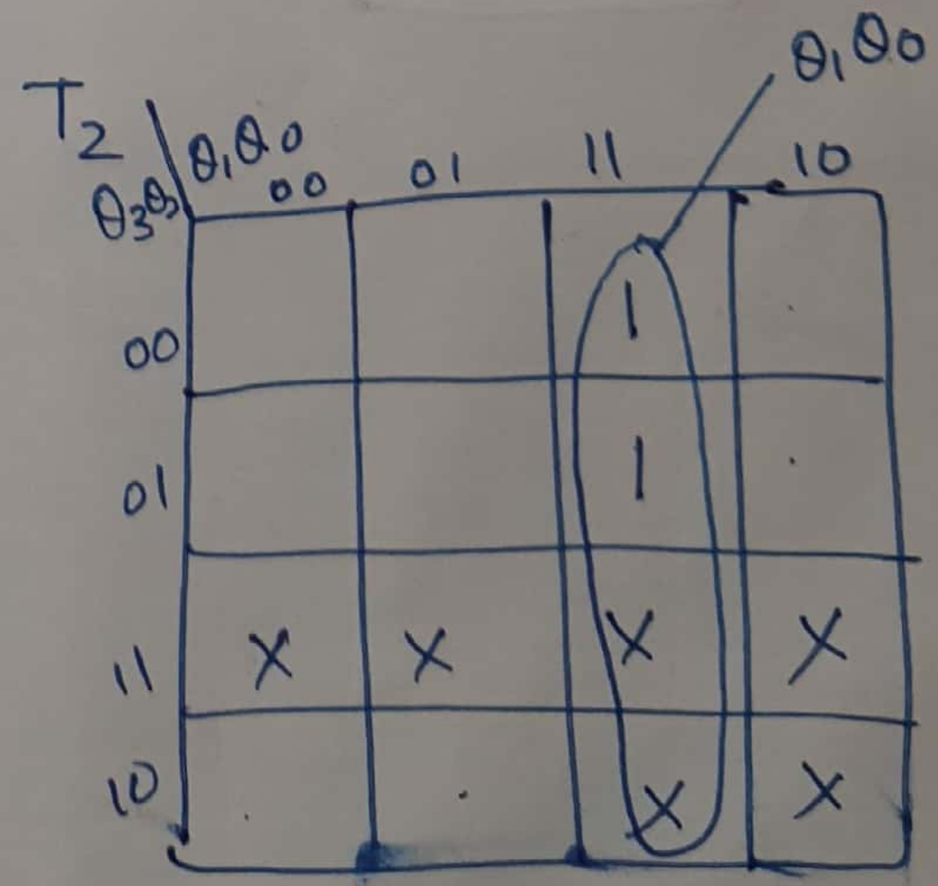


State Table :-

| Present State | | | | Next State | | | | Inputs | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $Q_3$ | $Q_2$ | $Q_1$ | $Q_0$ | $Q_3^+$ | $Q_2^+$ | $Q_1^+$ | $Q_0^+$ | $T_3$ | $T_2$ | $T_1$ | $T_0$ |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | X | X | X | X | X | X | X | X |
| 1 | 0 | 1 | 1 | X | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 0 | X | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 1 | X | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 0 | X | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 1 | X | X | X | X | X | X | X | X |

Step 4 :- Solve boolean expression.




$\therefore$ $T_3 = Q_3Q_0 + Q_2Q_1Q_0$

$T_2 = Q_1Q_0$

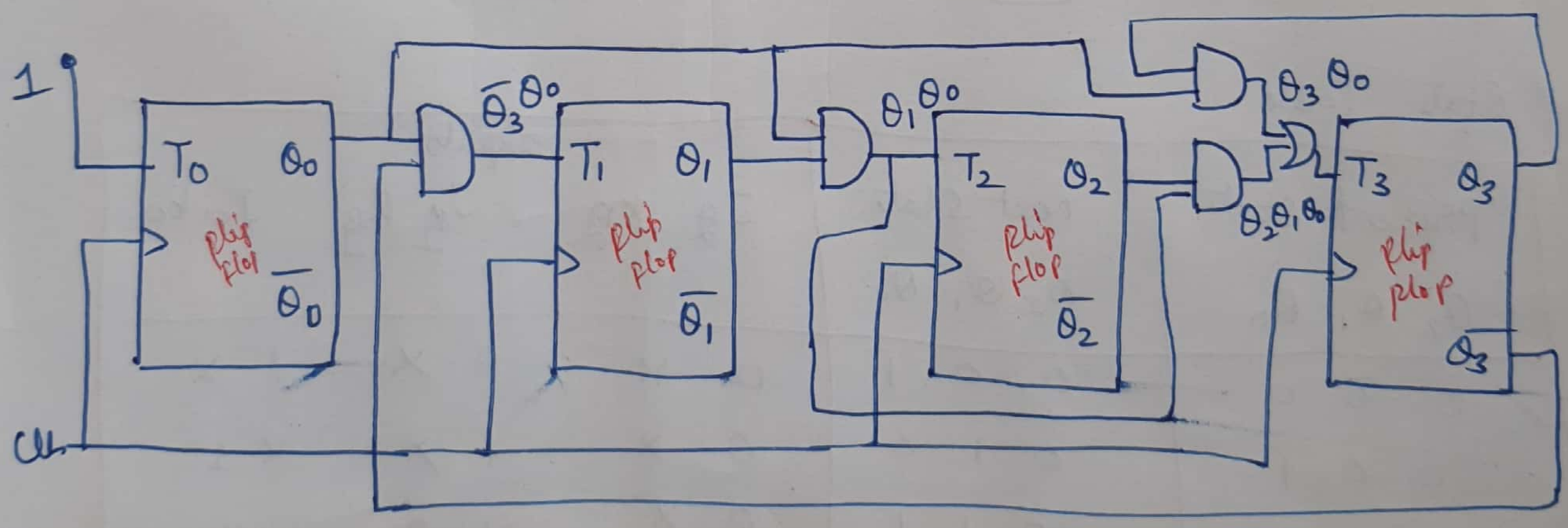
| $T_1$ | $\theta_1\theta_0$ = 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| $\theta_3\theta_2$ = 00 | | 1 | 1 | |
| 01 | | 1 | 1 | |
| 11 | X | X | X | X |
| 10 | | | X | X |

$\bar{\theta}_3\theta_0$

$$\boxed{T_1 = \bar{\theta}_3\theta_0}$$

| $T_0$ | $\theta_1\theta_0$ = 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| $\theta_3\theta_2$ = 00 | 1 | 1 | 1 | 1 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

1

$$\boxed{T_0 = 1.}$$

Step 5:- Draw Circuit