

## Integrated Circuits (ICs)

An IC is a small silicon semiconductor crystal which contains electrical components such as transistor, diode, capacitors, resistors etc.

The internal circuit of an IC can only be accessed through external pins. The components cannot be altered or replaced.

Benefits of ICs.

- ① Compact circuits
- ② Low cost
- ③ Reduced power consumption
- ④ High reliability
- ⑤ Higher speed

IC

|  |  |
|--|--|
| <u>linear</u><br>used for<br>conti. signals<br>and const. of<br>amplifier or<br>voltage comparators. | <u>digital</u><br>operation binary<br>signals and<br>are made<br>of interconnected<br>gates. |
|--|--|

SSI → Small Scale ICs → upto 10 gates

MSI → Medium Scale ICs → upto 100 gates

LSI → Large Scale ICs → more than 100 gates

VLSI → Very Large Scale ICs → upto 1000 gates.

## IC Logic Families

ICs are classified on the basis of the logical operation they perform as well as the specific logic-circuit family they belong to.

Each family has a basic gate using which more complex logical functions are formed. These basic gates are either NAND or NOR.

Families are named on the basis of the electronic components used to construct the basic circuits.

TTL → Transistor - Transistor logic  
CMOS → Complementary - Metal Oxide Semiconductor.

are the families which we will discuss in detail.

TTL → ① Has an extensive list of functions.  
② Most popular currently.

CMOS → Offers low power consumption.

Characteristics of Families

- TTL
1. Have LSI devices and also a large no. of MSI devices.
  2. Have a numerical designation 5400 or 7400 series. (Used in our labs).
    - 5400 → Military use
    - 7400 → Industrial use

IC's function in terms of  $H$  &  $L$  and not 0 & 1 ] example after ~~RD~~ classmate  
characteristics Date \_\_\_\_\_  
Page \_\_\_\_\_

An IC has two values high / low

1  $\rightarrow$  we can take high as 1 and low as 0.  
0  $\leftarrow$  (usually taken)

so the output will be corresponding

If  $H=1$  and  $L=0$ , it is +ve logic.

(If  $H=0$  and  $L=1$ , it is -ve logic)  
never used, only theoretical.

3. High Value = 2.4 - 5

Low Value = 0 - 0.4

Operating Voltage = 5V. ( $V_{ce}$ )

- ~~Ans~~
- ① Fan-out  $\rightarrow$  The maximum no. of inputs that can be conn. to an output of a gate, is expressed by a no. (represent gate overloading, if no. of conn. exceed fan-out, improper functioning).
  - ② Power-dissipation  $\rightarrow$  Power required to operate the gate (mW). Power delivered to a gate by the power supply.
  - ③ Propagation-delay  $\rightarrow$  average transition time in propagation of signal from i/p to o/p

④ Noise-Margin → Max noise voltage that can be added to the input signal such that it does not disrupt the functioning.

4. Basic gate → NAND.

|                 | Fan Out | Power | P.D | N.M. |
|-----------------|---------|-------|-----|------|
| 5. Standard TTL | 10      | 10    | 10  | 0.4  |

|             |    |    |   |     |
|-------------|----|----|---|-----|
| Shottky TTL | 10 | 22 | 3 | 0.4 |
|-------------|----|----|---|-----|

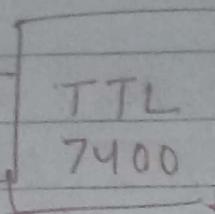
|                          |    |   |    |     |
|--------------------------|----|---|----|-----|
| low-power<br>Shottky TTL | 20 | 2 | 10 | 0.4 |
|--------------------------|----|---|----|-----|

CMOS :-

1. Has both MSI and LSI devices.
2. Has a numerical designation as 4000 series.
3. High Value =  $V_{DD}$  (3-10)  
Low Value = 0 - 0.5  
Operating voltage =  $(V_{DD})$  3-10
4. Basic Gate → INVERTER (both NOR/NAND can be derived)

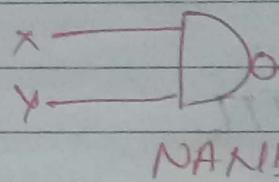
|            |       |     |      |
|------------|-------|-----|------|
| 5. Fan Out | Power | P.D | N.M. |
| 50         | 0.1   | 25  | 3.   |

| $X$ | $Y$ | $Z$ | $X$ | $Z$ |
|-----|-----|-----|-----|-----|
| L   | L   | H   |     |     |
| L   | H   | H   |     |     |
| H   | L   | H   |     |     |
| H   | H   | L   |     |     |



for +ve logic

| $X$ | $Y$ | $Z$ |
|-----|-----|-----|
| 0   | 0   | 1   |
| 0   | 1   | 1   |
| 1   | 0   | 1   |
| 1   | 1   | 0   |



for -ve logic

| $X$ | $Y$ | $Z$ |
|-----|-----|-----|
| 1   | 1   | 0   |
| 1   | 0   | 0   |
| 0   | 1   | 0   |
| 0   | 0   | 1   |

NOR

(with -ve logic)

for -ve logic  
use triangles

ROM :- Read Only Memory.

A decoder generates  $2^n$  minterms for  $n$  inputs. By inserting OR gates to these minterms, we can generate any desired combinational circuit.

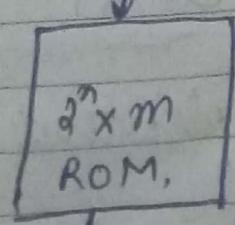
The IC of a ROM consists of both a decoder and OR gates. The conn b/w the outputs of decoder and inputs of the ~~OR~~ OR gate can be specified by programming the ROM.

ROM → used to implement complex combinational circuits in one IC package.

A Rom is a storage device in which a fixed set of binary information is stored and is ~~not~~ embedded to form unit interconnection pattern.

Once, a pattern is established, it cannot be changed and is fixed even when the power is turned ON/OFF.

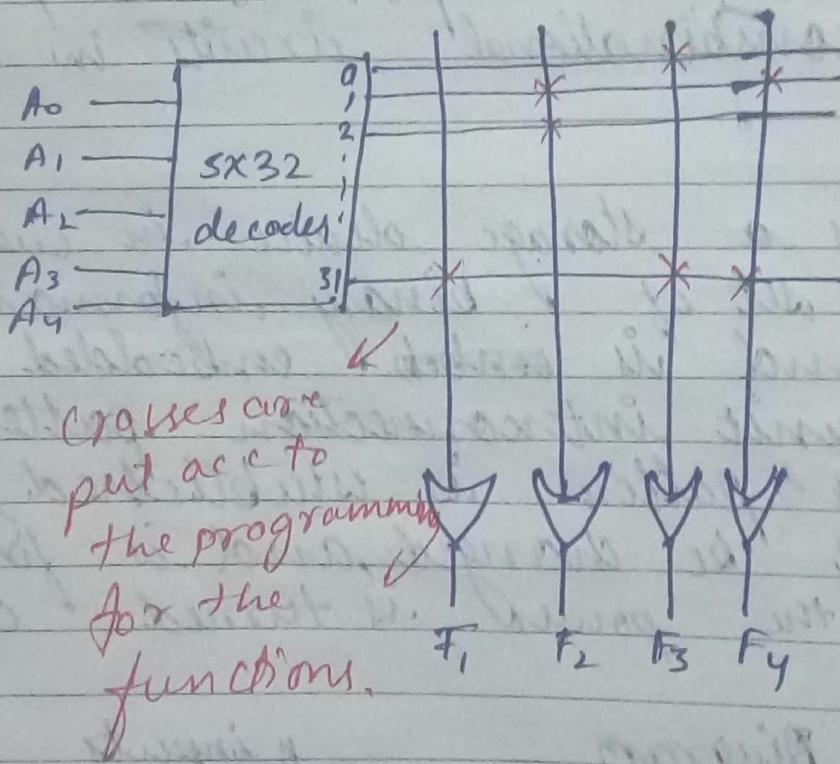
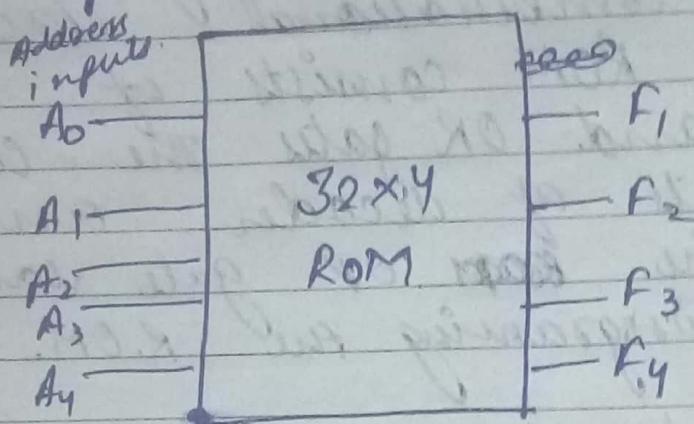
Block Diagram



$n$  inputs

$m$  outputs

A  $2^n \times m$  ROM consists of  $2^n$  words of  $m$  bits each.  
 i.e. There are  $m$  functions  
 and  $2^n$  minterms i.e.  $n$  address inputs.



32x4 ROM

Ques Implement functions.

$$F_1(A_1, A_0) = \Sigma(1, 2, 3)$$

$$F_2(A_1, A_0) = (0, 2) \quad \text{using } \underline{4 \times 2} \text{ Rom}$$

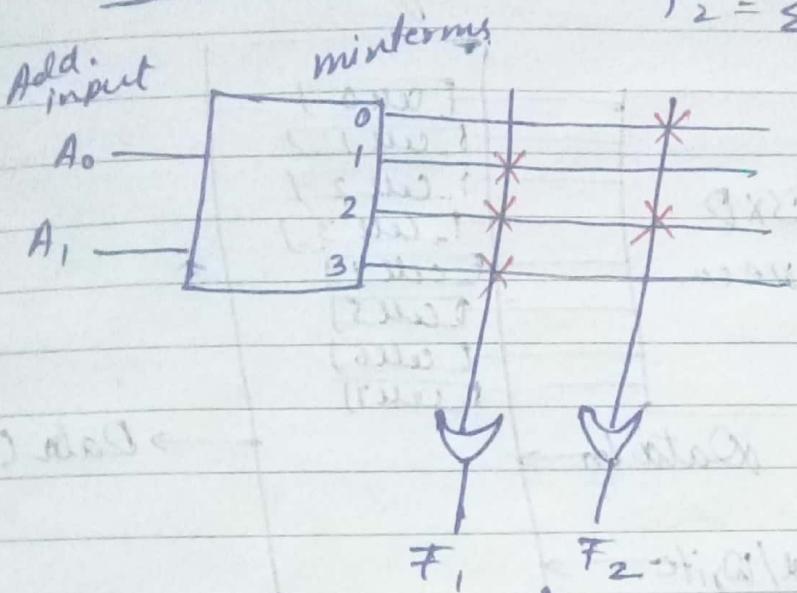
Ans:-  $4 \times 2$  rom

$$\therefore n = 2 ; m = 2$$

$$4 = 2^n$$

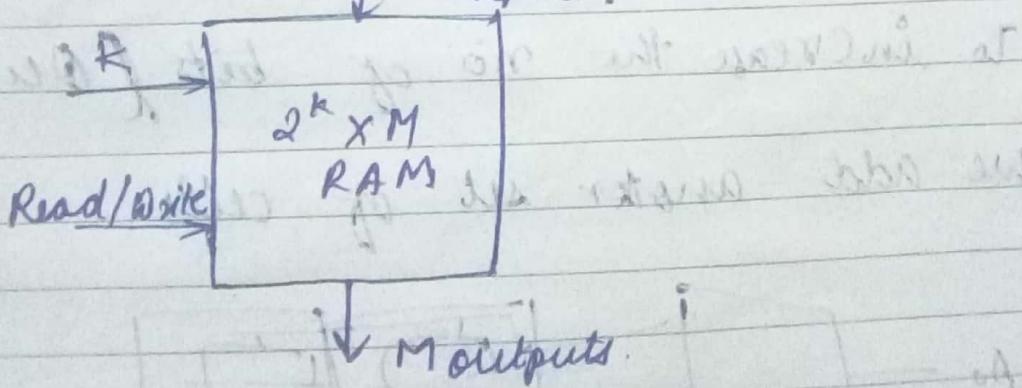
$$F_1 = \Sigma (1, 2, 3)$$

$$F_2 = \Sigma (0, 2)$$



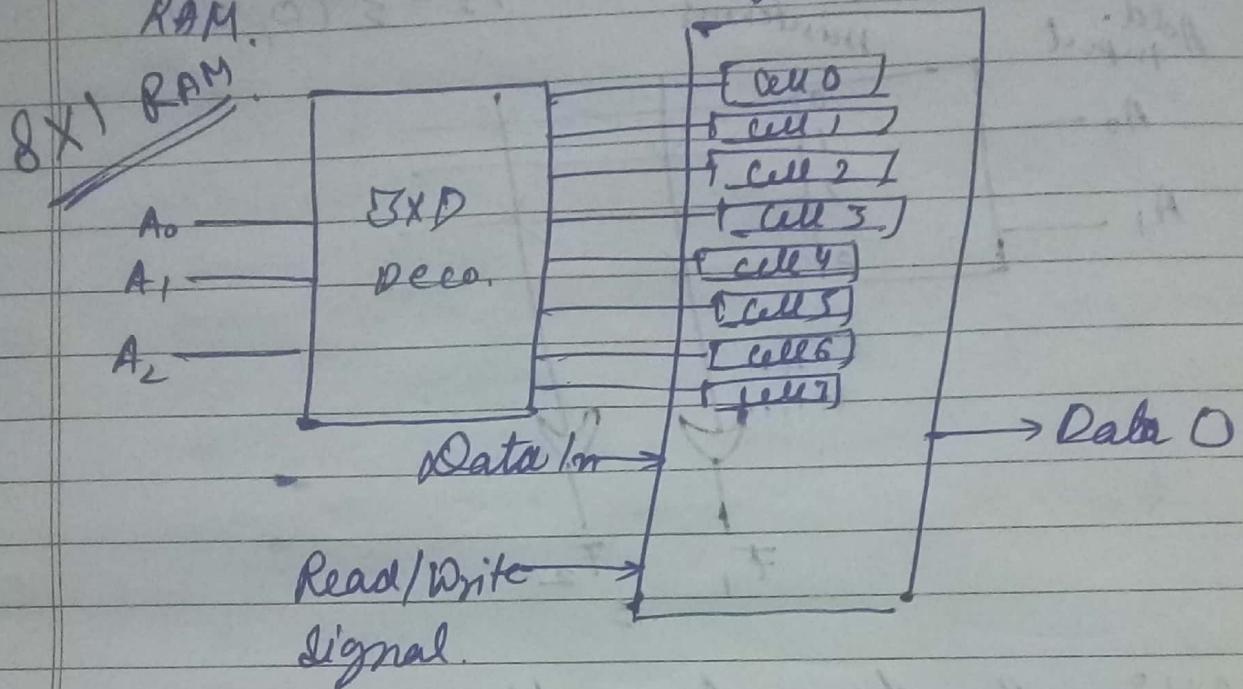
RAM:- Random Access Memory.

The fastest rewritable memory storage. A RAM is ~~reliable~~ volatile and loses data once the computer is shut.  
↓ M inputs.



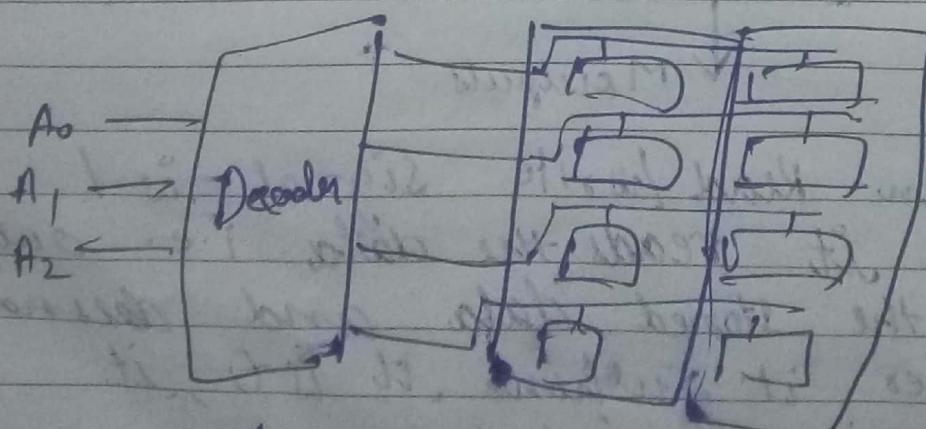
When Read/write signal is 1  
it reads the data i.e. outputs  
the stored data and does not  
alter it. When it is 0, it  
rewrites the data.

A RAM consists of ~~8x8~~  $K \times 2^k$  decoder and  $2^K$  ram cells, each connected to the decoders output. They store input data given to the RAM.



The decoder helps enable one cell at a time to perform read/write operations.

To increase the no. of bits (Output/inputs) we add another set of cells.

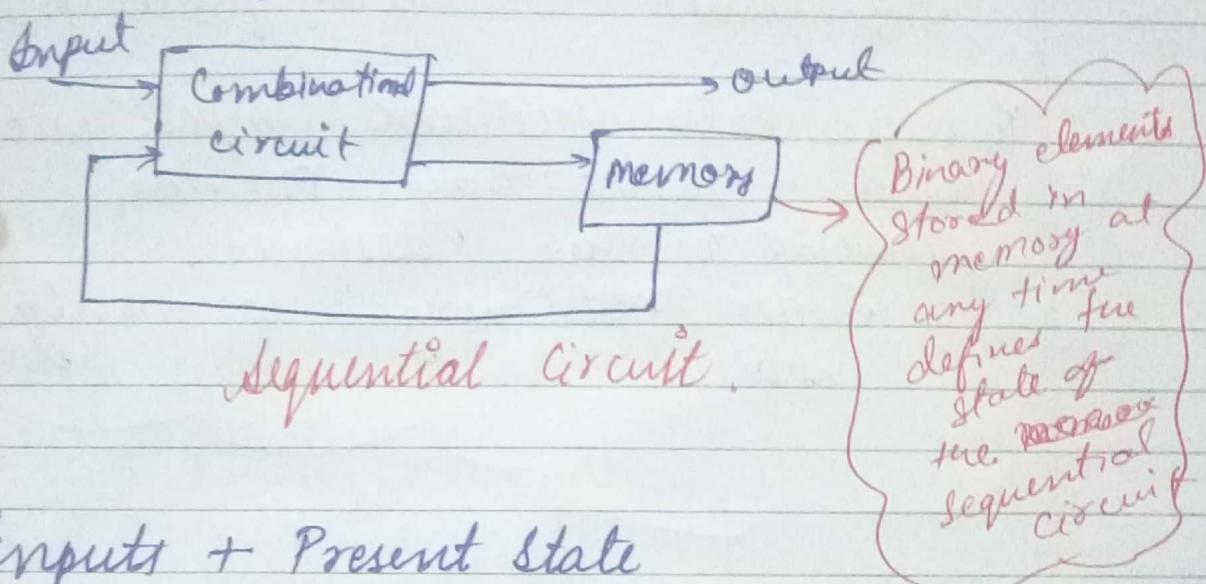


and each will have separate in/out and a Read/Write signal.

## Sequential Logical Circuits

These are logical circuits in which the output depends on the previous outputs of the device.

### Block Diagram



Inputs + Present State

↓ determine

Output + Next State.

Sequential circuit is specified by a time sequence of inputs/outputs and memory states.

Synchronous

A system whose behaviour can be defined from the knowledge of its signals at discrete instants of time.

Asynchronous

Behaviour depends on the order in ~~which~~ which input signals change, can be applied at any time.

## ~~Synchronous~~

- \* Asynchronous circuits use memory devices which are time-delay devices.
- \* In gate type asynchronous system, memory logic gates are used whose propagation provide the necessary delay.

Asynchronous  $\Rightarrow$  Combinational circuit  
Seq. Circuit with feedback

- \* logic gates feedback impose instability and hence they provide problems to the designer.  
So are not very commonly used.

## Synchronous

- \* signals affect memory states ~~every~~ only at discrete instances of time.

It is achieved by using pulses of limited duration as inputs.

(two pulses from diff sources may have diff time delays which may effect the end operation)

- \* Practical devices use fixed binary signal which are varied through clock impulses.

\* Clock pulse are implied with an ~~read AND~~ gate into the signal.

↓ use clock

∴ Clock sequential  
circuits.

↓ they use

memory element → Flip Flops.

Katchers store 1 bit of information.  
Flip-flops are a type of latch  
that work according to a  
clock circuit.

Trigger → The pulse that triggers to get  
from input to output.

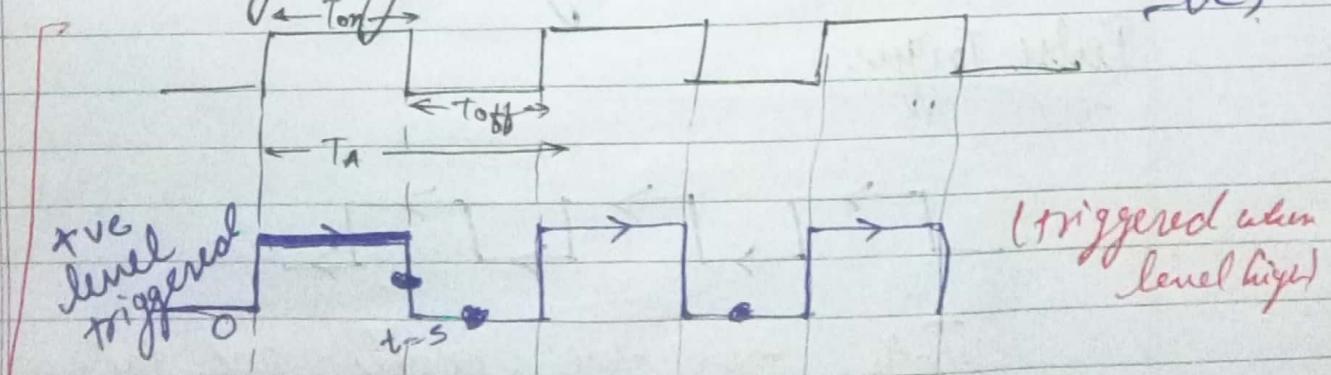
### Types of trigger

① Level → +ve  
→ -ve

② Edge → +ve  
→ -ve

③ Pulse

Level Triggering (takes note of the level +ve/-ve)



+ve level triggered

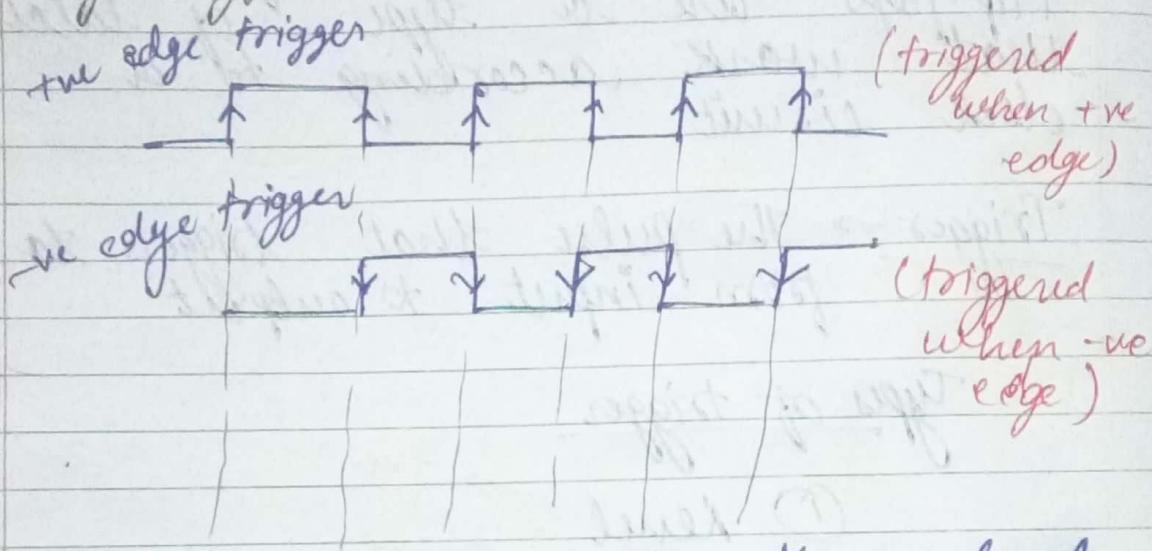
(triggered when level high)

-ve level triggered

(triggered when level low)

application → latches.

Edge Triggered (Takes note of the edge)



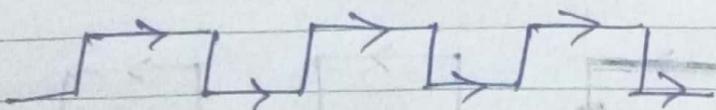
Edge triggers are better than level triggers in terms of time delay.

Application →

+ve → flipflops.

-ve → registers.

Pulse Trigger



Both +ve/-ve levels are taken into account.

application → Master Slave Flipflops

+ve level

-ve level.

Latch  $\rightarrow$  Has one input and two outputs.

SR latch  $\rightarrow$  Set/Reset latch.

D latch  $\rightarrow$  Delay/Data latch.

It has two stable states

$Q=0, \bar{Q}=1 \rightarrow$  Reset state / Logic 0

$Q=1, \bar{Q}=0 \rightarrow$  Set state / Logic 1

A latch has 1 control input and 2 inputs.

The presence of controlled input makes it a gated latch.

Control input = 1  $\rightarrow$  reset state

Control input = 0  $\rightarrow$  set state.

When level trigger is same as control I/P, it is called control circuit.

There are two types of latches.

$\rightarrow$  Active high latches.

$\rightarrow$  Active low latches.

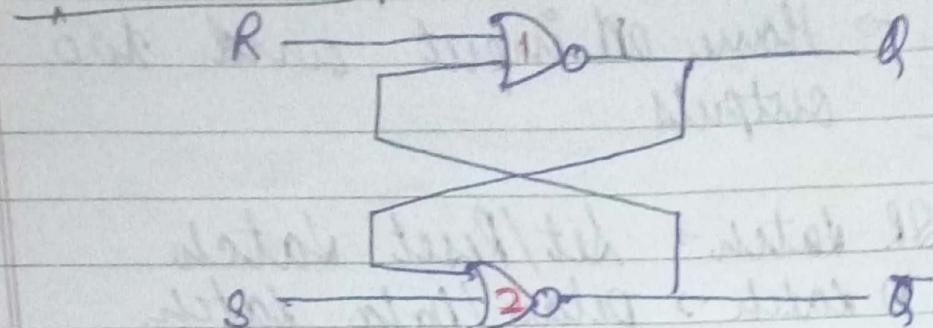
Active High Latches.

They reset at low level and set at a high level. They are

activated at a high level. Are formed by 2 cross coupled NOR gates.

high

JL active latch



| S | R | $\bar{Q}_{in}$ | $\bar{Q}_{out}$ | name          |
|---|---|----------------|-----------------|---------------|
| 0 | 1 | 0              | 1               | Reset         |
| 0 | 0 | 0              | 1               | (R) No change |
| 1 | 0 | 1              | 0               | Set           |
| 0 | 0 | 1              | 0               | (S) No change |

NOR gate

| A | B | X |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

any input is 1,  
then o/p is 0.

If  $R=1$ , input of 1 NOR gate is 1 and ~~connection~~  $\bar{Q}_{in}$  (In previous case)  
if one of the inputs is 1  
output is 0.

$$\therefore 0 = 0$$

now for 2 NOR gate

$$\bar{0} + \bar{0} = \bar{0} = 1 \\ \therefore \bar{0} = 1 \quad \text{o/p result}$$

when both are 0, (after set)

$$1 \text{ gate} \quad \bar{0} + 1 = 0$$

$$2 \text{ gate} \quad \bar{0} + \bar{0} = 1$$

$\therefore$  No change

If  $R = 0$  and  $S = 1$ .

$$Q = 1$$

$$\text{and } \bar{Q} = 0$$

as case I follows with inverted inputs.

$\therefore$  set.

If  $R = 0$  and  $S = 0$  (after set)

$$1\text{st gate} = Q + 0 = 1$$

$$2\text{nd gate} = \bar{Q} + 1 = 0$$

$\therefore$  No change.

For case II

$Q$  and  $\bar{Q}$  both will be 0, which is invalid.

$\therefore$  we eliminate such case.

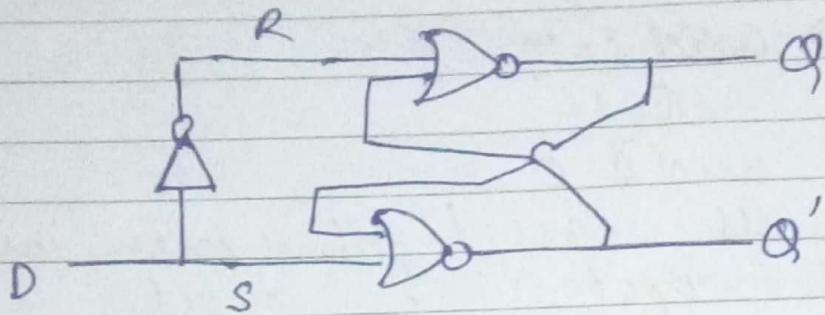
We cannot set/reset at the same time.

D-high active latch

We don't give ~~2~~ 2 separate inputs S and R instead 1-D

and if D is high  $\rightarrow$  set  
D is low  $\rightarrow$  reset.

we connect D to S and 1-D  
R is negation of S.

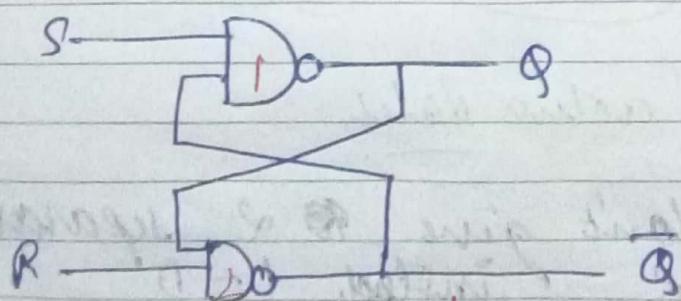


| D | $Q_{n+1}$ | $\bar{Q}_{n+1}$ | Reset |
|---|-----------|-----------------|-------|
| 0 | 0         | 1               | Set   |
| 1 | 1         | 0               |       |

### Active Low Latches

They rest at high level and are activated/set at low levels.  
They are made of cross Coupled NAND gates.

### SR low Active latch



| S | R | $Q_{n+1}$ | $\bar{Q}_{n+1}$ |
|---|---|-----------|-----------------|
| 0 | 1 | 1         | 0 (set)         |
| 0 | 0 | 1         | 0 (S) Noch.     |
| 1 | 0 | 0         | 1 (Reset)       |
| 1 | 1 | 0         | 1 (R) Noch.     |
|   |   | 1         | Invariably      |
| 0 | 0 | 1         | 1               |

NAND gate.

| A | B | X |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Any if 0  $\rightarrow$  op is 1.

$S=0$  and  $R=1$ .

Gate 1  
 $S=0$        $\bar{Q}_n = ?$  Not Known.  
 $Q_{n+1} = ?$

Gate 2  
 $R=1$        $Q = 1$

$\bar{Q}_{n+1} = 0$ .      ∴ Set

$S=1$  and  $R=1$  (after set.)

Gate 1  
 $S=1$        $\bar{Q}_n = 0$

$Q_{n+1} = 1$

Gate 2

$R=1$        $Q = 1$

$\bar{Q}_{n+1} = 0$ .

∴ No change.

$S=1$  and  $R=0$ .

Gate 1  
 $S=1$        $\bar{Q}_n = 0$

$\bar{Q} = 1$  and  $Q = 0$

as case I follows.

$S=1$  and  $R=1$  (after set)

Gate 1  
 $S=1$        $\bar{Q} = 1$

$R=1$        $Q = 0$

$\bar{Q} = 1$

∴ no change.

For case 00

$Q = 1$  and  $\bar{Q} = 1$

which is not possible

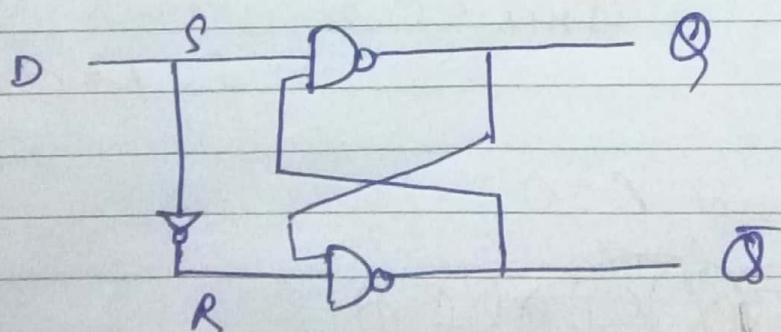
∴ ignore this

case.

Resetting & Setting cannot occur together.

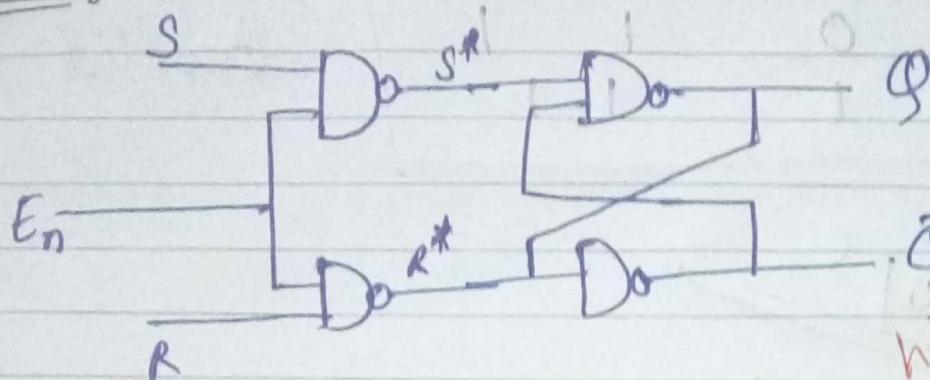
## D - how active latch

Connect D to S and R to negation of Q.



| D | $Q_{n+1}$ | $\bar{Q}_{n+1}$ | Reset. | Set. |
|---|-----------|-----------------|--------|------|
| 1 | 0         | 1               | 1      | 0    |

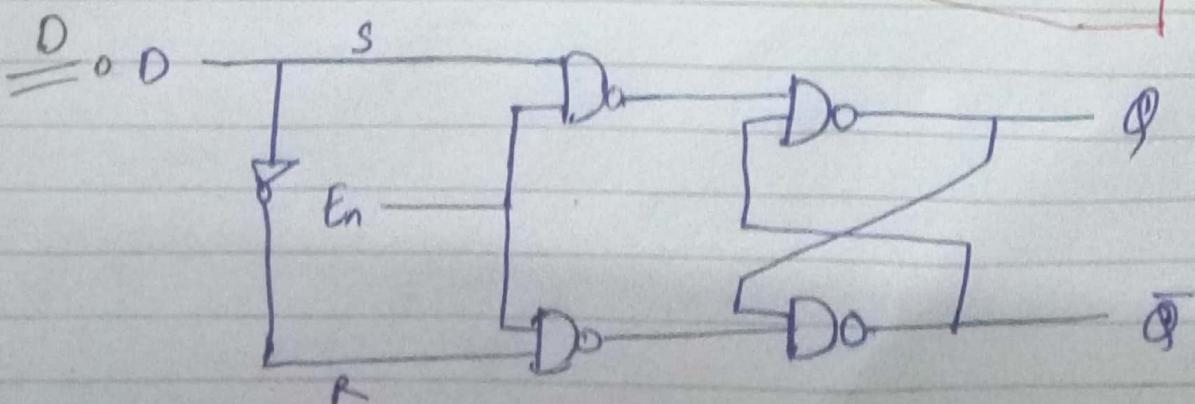
Gated latch.  $\rightarrow$  Latch with control input  $E_n$ .

SR

$R=1 \rightarrow$  Reset  
 $S=1 \rightarrow$  Set

High Active Control  $\downarrow$   
Control latch

| $E_n$ | S | R | $S^*$ | $R^*$ | $Q$ | $\bar{Q}$ |               |
|-------|---|---|-------|-------|-----|-----------|---------------|
| 1     | 0 | 1 | 1     | 0     | 0   | 1         | (Reset.)      |
| 1     | 0 | 0 | 1     | 1     | 0   | 1         | (R) No change |
| 1     | 1 | 0 | 0     | 1     | 1   | 0         | (Set)         |
| 1     | 0 | 0 | 1     | 1     | 1   | 0         | No change     |
| 1     | 1 | 1 | 0     | 0     | 1   | 1         |               |
| 0     | 0 | 1 | 1     | 1     | 1   | 1         |               |
| 0     | 0 | 0 | 1     | 1     | 1   | 1         |               |
| 0     | 1 | 0 | 1     | 1     | 1   | 1         |               |
| 0     | 0 | 0 | 1     | 1     | 1   | 1         |               |
| 0     | 1 | 1 | 1     | 1     | 1   | 1         | Invalid.      |

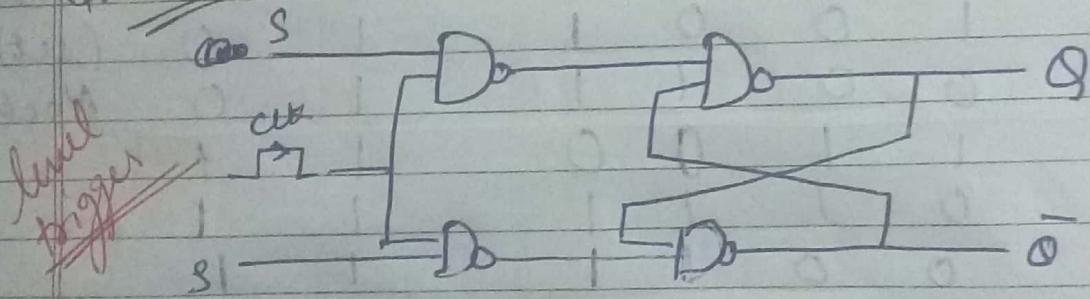


| En | D | Q | $\bar{Q}$ |         |
|----|---|---|-----------|---------|
| 1  | 0 | 0 | 1         | Reset   |
| 1  | 1 | 1 | 0         | Set     |
| 0  | 0 | 1 | 1         | Invalid |
| 0  | 1 | 1 | 1         |         |

~~Flip Flops~~

Flip Flops are gated latches with control signal as a clk.

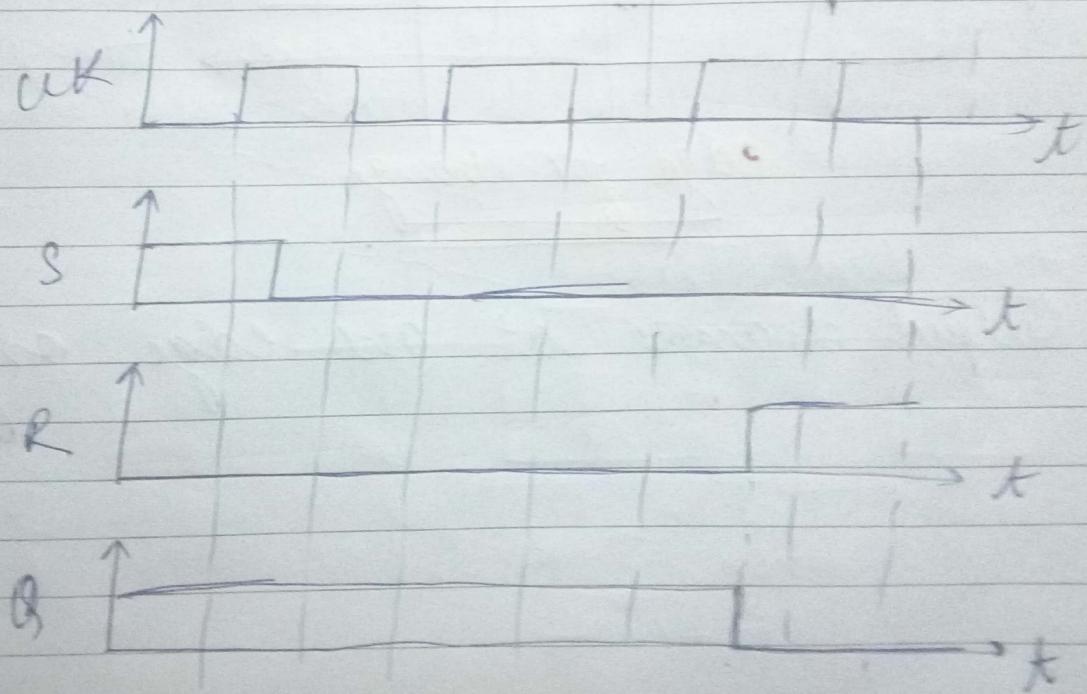
~~SR - Flip flops~~



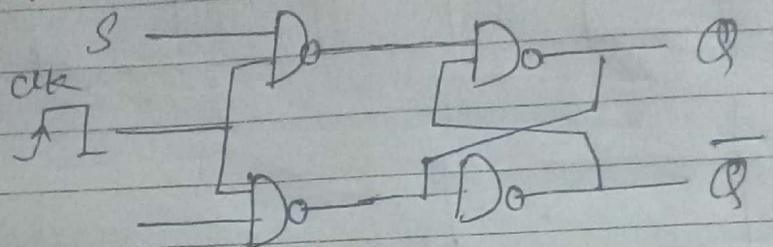
| Clk | S | R | Q | $\bar{Q}$ |               |
|-----|---|---|---|-----------|---------------|
| ↑   | 1 | 0 | 1 | 0         | Set           |
| ↓   | 0 | 0 | 1 | 0         | (S) No change |
| ↑   | 0 | 1 | 0 | 1         | Reset         |
| ↓   | 0 | 0 | 0 | 1         | (R) No change |
| ↑   | 1 | 1 | 1 | 1         | X Invalid     |

Changes occur only when clk is 1.  
Or else no change.

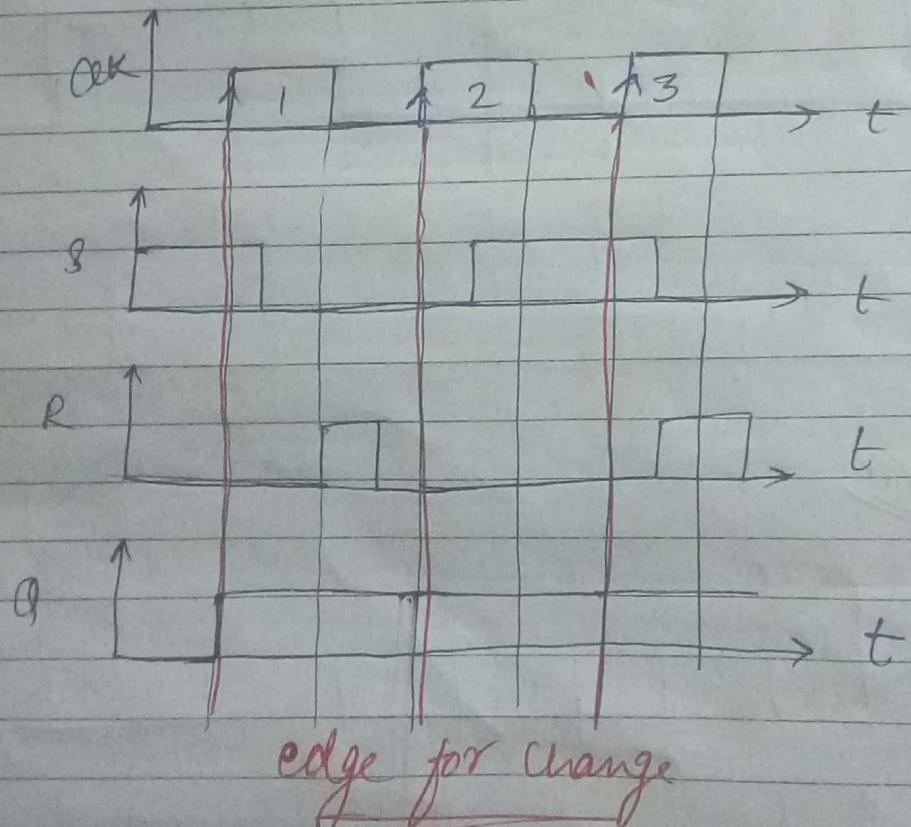
Wave Form.



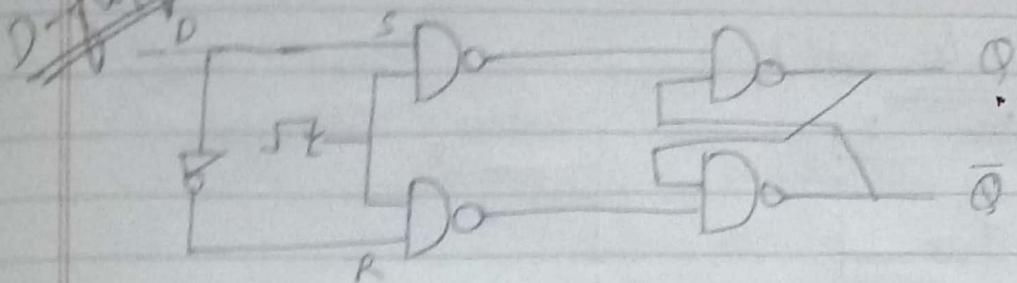
~~edge trigger~~ Output will change at one edge.



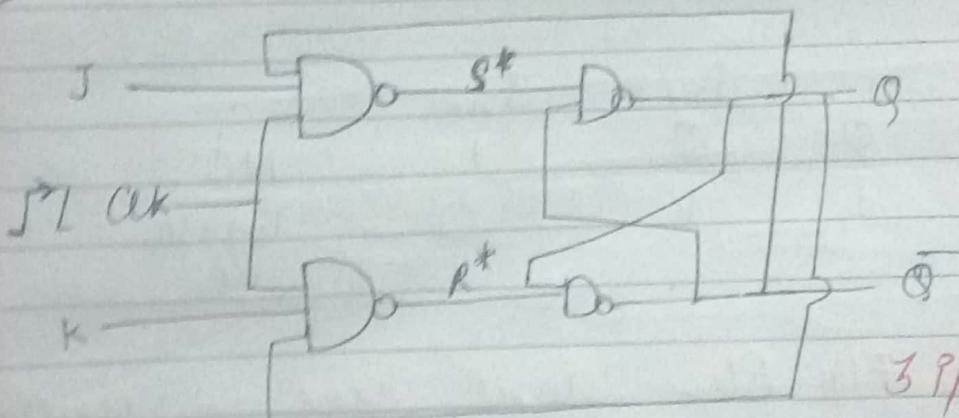
~~wave form~~



Output changes at one edge only.

~~D-flipflop~~

| Clk    | D | $Q_{n+1}$ | $\bar{Q}_{n+1}$ |           |
|--------|---|-----------|-----------------|-----------|
| ↑      | 0 | 0         | 1               | Reset     |
| ↑      | 1 | 1         | 0               | Set       |
| No Clk | X | No change | No change       | No change |

~~JK FLIP-FLOP~~

3 P/p NAND.

x y z o/p

| Clk       | J | K | S*        | R* | $Q_{n+1}$ | $\bar{Q}_{n+1}$ |   |
|-----------|---|---|-----------|----|-----------|-----------------|---|
| Reset     | ↑ | 0 | 1         | 1  | 0         | 0               | 1 |
| No Change | ↑ | 0 | 0         | 1  | 1         | 0               | 1 |
| Set       | ↑ | 1 | 0         | 0  | 1         | 0               | 1 |
| No Change | ↑ | 0 | 0         | 1  | 1         | 0               | 1 |
| Off       | ↑ | 1 | 1         | 1  | 0         | 1               | 1 |
|           | ↑ | 1 | 1         | 0  | 1         | 0               | 0 |
| No Clk    | X | X | No Change | —  | + + +     | 1               | 0 |

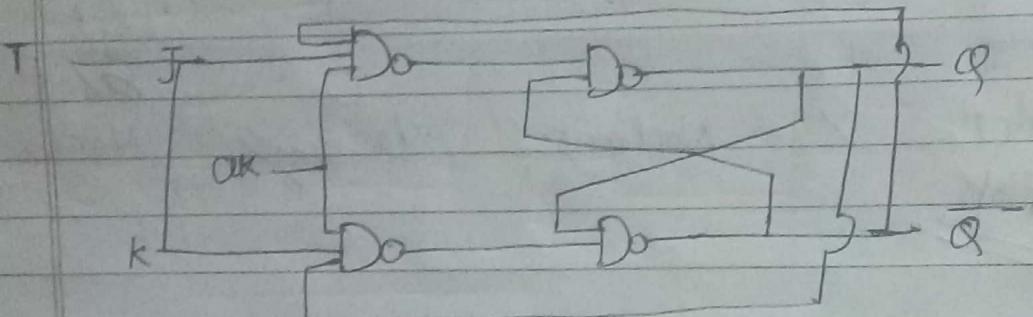
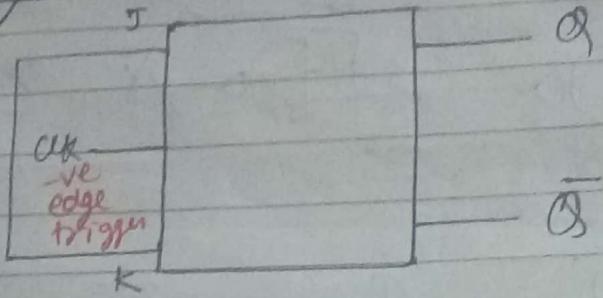
any 1 i/p is 0  
then, 1.

when J, K are 11

 $Q \leftrightarrow \bar{Q}$  Interchanges.

~~T-Flip Flops.~~

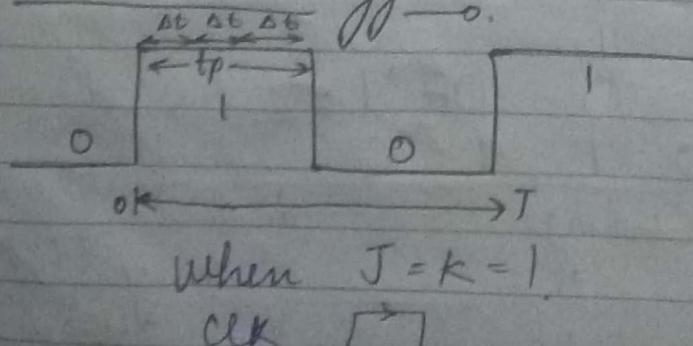
Toggle  $\rightarrow T$ .



| clk | T | $Q_{n+1}$ | $Q_{n+1}'$ |         |
|-----|---|-----------|------------|---------|
| 0   | 0 | 0         | 1          | NC      |
| 1   | 1 | 1         | 0          | Toggle. |

\* JK Flip-Flops don't have an unstable state. They don't have any invalid inputs such cases are replaced by Toggle.

+ve level Trigger.



Let  $st$  = delay time for NAND gate in series.  
if it is connected 5 or 8 times,  
when  $J=K=1$ , when  $t_p$  time is completed  
then output is in dilemma.  
This condition is called race around  
condition and occurs in JK flip flops with  
level triggering.

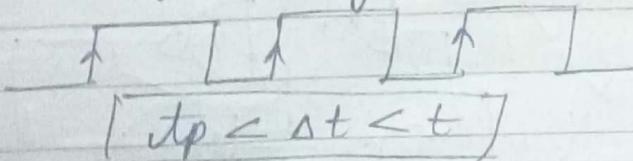
$t_p \rightarrow$  pulse width.

$$[st < t_p < T]$$

$T \rightarrow$  Time period of clock pulse.

To avoid race around.

- ① Use edge triggered i/p.



- ② Use master-slave flip-flops.  
(They use pulse trigger).

Master Slave flip-flops can be SR, JK, D  
or T.

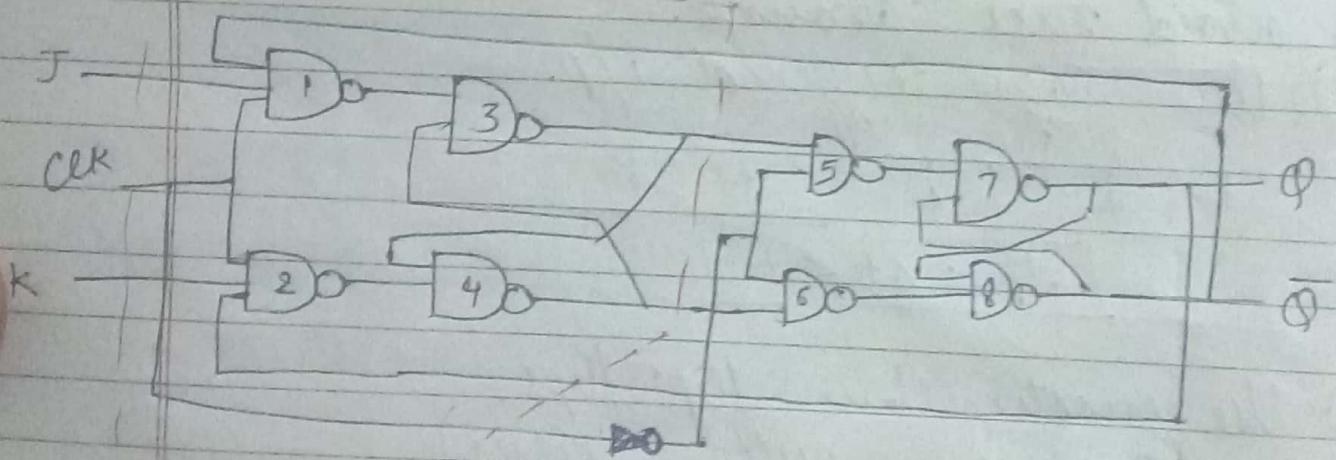
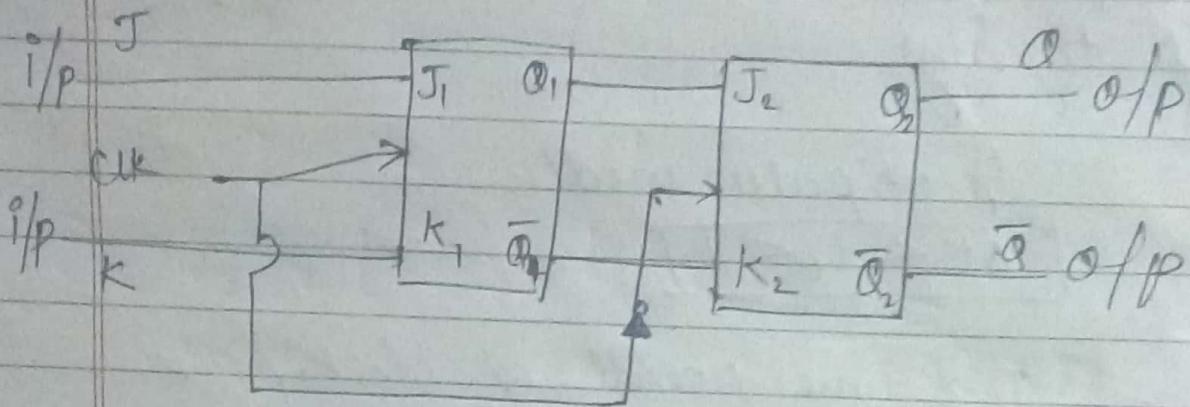
### # Master Slave (MS)- JK flip flops

Here 2 circuits of JK flip flops are used  
where the Slave circuit is following the  
Master circuit with a clk.

The o/p of the master circuit is fed to  
the i/p of the slave circuit with a

clock passing through the not gate.  
 The o/p of the slave circuit is  
 o/p of the flip/flop.

## Block Diagram



master

slave

| Clk | J, K, | Clk | $Q_1 = J_2 \bar{Q}_1$ | $\bar{Q}_1 = K_2$ | $Q_2 = Q$ | $\bar{Q}_2 = \bar{Q}$ | comm   |
|-----|-------|-----|-----------------------|-------------------|-----------|-----------------------|--------|
| 1   | 0 1   | 1   | 0                     | 1                 | 0         | 1                     | Reset  |
| 1   | 0 0   | 1   | 0                     | 0                 | 0         | 1                     | NC     |
| 1   | 1 0   | 1   | 1                     | 0                 | 1         | 0                     | Set    |
| 1   | 0 0   | 1   | 1                     | 0                 | 1         | 0                     | NC     |
| 1   | 1 1   | 1   | 0                     | 1                 | 0         | 1                     | Toggle |

\* Master-Slave Flip-flop with the level triggered works as an -ve edge triggered flip flop.

Race around condition is not working as at the level triggering the value of NAND gate 3 & 4 is unchanged as the slave works at -ve edge level of the clock pulse.

Output of the master flip flop is the output of the slave flip flop.

## Conversion b/w Flip Flops.

$SR \rightarrow JK$  (follow the steps for any conversion).

① Draw excitation table of SR flip flop. (i.e. table with  $S, R, Q_n$  and  $Q_{n+1}$ ). (follow below Active)

|                   |   | PS           | NS    |
|-------------------|---|--------------|-------|
| no change/reset   | g | $X \oplus 0$ | $Q_n$ |
| set               | 1 | 0            | 0     |
| reset             | 0 | 1            | 1     |
| no change/<br>set | X | 0 $\oplus 0$ | 1     |

Keeping PS and NS in mind  
Find S and R for each case.

② Draw excitation table for JK Flip Flop.

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

③ Relation b/w S R and JK Q

| Given S R | Reg. | <del>Output</del> |           |
|-----------|------|-------------------|-----------|
|           | J K  | $Q_n$             | $Q_{n+1}$ |
| 0 X       | 0 0  | 0                 | 0         |
| X 0       | 0 0  | 1                 | 1         |
| 0 X       | 0 1  | 0                 | 0         |
| 0 1       | 0 1  | 1                 | 0         |
| 1 0       | 1 0  | 0                 | 1         |
| X 0       | 1 0  | 1                 | 1         |
| 1 0       | 1 1  | 0                 | 1         |
| 0 1       | 1 1  | 1                 | 0         |

↓  
write other  
cases acc to

this and

comp the table.

Use K-map to find.

S in terms of J, K and  $Q_n$   
R in terms of J, K and  $Q_n$

$J =$

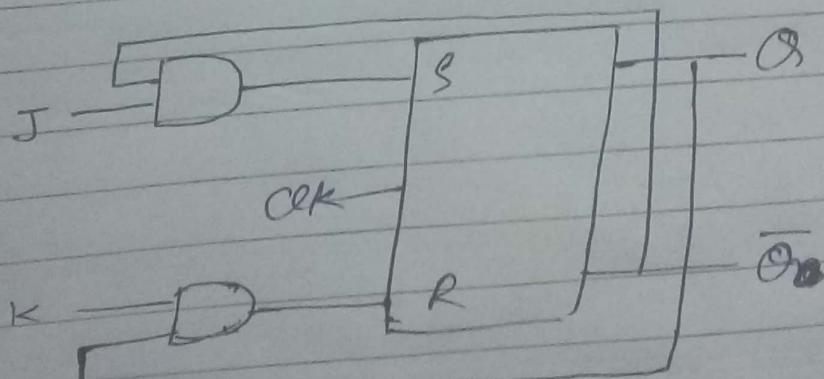
| $\bar{J}$ | $\bar{K}Q_n$   | $\bar{K}Q_n$   | $KQ_n$ | $K\bar{Q}_n$ |
|-----------|----------------|----------------|--------|--------------|
| $\bar{J}$ | 0              | X <sub>1</sub> | 3      | 2            |
| J         | 1 <sub>4</sub> | X <sub>5</sub> | 7      | 6            |

$$S = J \bar{Q}_n$$

R

| $\bar{J}$ | $\bar{K}Q_n$ | $\bar{K}Q_n$ | $KQ_n$ | $K\bar{Q}_n$ |
|-----------|--------------|--------------|--------|--------------|
| $\bar{J}$ | $x_0$        | 1            | (1)    | $x_2$        |
| J         | 4            | 5            | 7      | 6            |

$$R = K Q_n$$



Counter: A sequential counter which counts a value can count from any no to any no depending on our design.

0, 1, 2, ... 10 or 2, 3, 4, ... 6

or 9, 10, 11, ...

~~Binary Counter~~ → 0, 1, 2, 3 ...

~~Binary Asynchronous Counter~~ Bits of counting come out at different times. Also known as ripple counters.

① An Asynchronous Counter is built using a D flip flop.

② O/p of one D flip flop is the i/p of another.

③ n bit binary counter consists of n flip-flops and can count from 0 to  $a^n - 1$ .

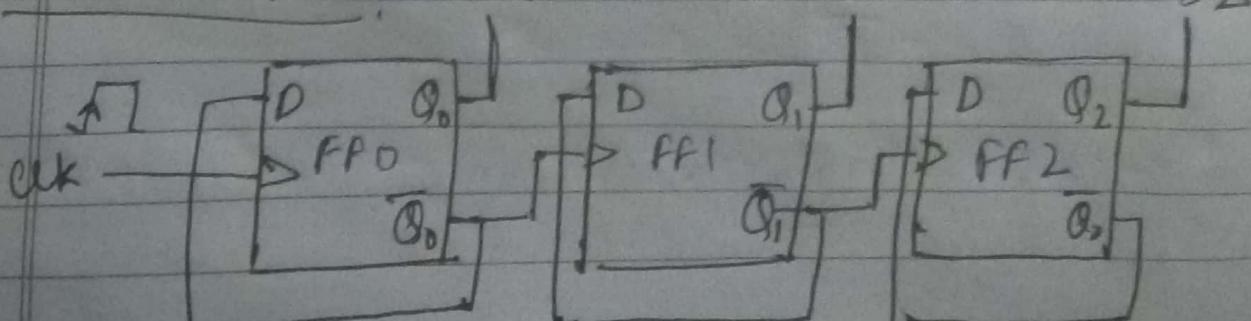
④ The Clock is given as inp to first counter. The rest of the flip/flops have the output of the previous flip flop as an input.

3 bit Counter

B<sub>0</sub>

B<sub>1</sub>

B<sub>2</sub>



| Clk | $Q_0$ | $\bar{Q}_0$ | (clk) $Q_1$ | $\bar{Q}_1$ | (clk) $Q_2$ | $\bar{Q}_2$ |
|-----|-------|-------------|-------------|-------------|-------------|-------------|
| 0   | 0     | 1           | 0           | 1           | 0           | 1           |
| 1   | 1     | 0           | 0           | 1           | 0           | 1           |
| 2   | 0     | 1           | 1           | 0           | 0           | 1           |
| 3   | 1     | 0           | 1           | 0           | 0           | 1           |
| 4   | 0     | 1           | 1           | 0           | 1           | 0           |
| 5   | 1     | 0           | 0           | 1           | 1           | 0           |
| 6   | 0     | 1           | 1           | 0           | 1           | 0           |
| 7   | 1     | 0           | 1           | 0           | 1           | 0           |

| Clk | $B_2$ | $\overbrace{B_1}$ | $B_0$ |
|-----|-------|-------------------|-------|
| 0   | 0     | 0                 | 0     |
| 1   | 0     | 0                 | 1     |
| 2   | 0     | 1                 | 0     |
| 3   | 0     | 1                 | 1     |
| 4   | 1     | 0                 | 0     |
| 5   | 1     | 0                 | 1     |
| 6   | 1     | 1                 | 0     |
| 7   | 1     | 1                 | 1     |

when 1st clock pulse arrives,

$D = 1 \therefore (Q_0)_{n+1} = 1$  as D takes up  $(\bar{Q}_0)_n$  which was 1.

however, there is no change in clock pulse for FF2 and FF1  $\therefore$  No change.

~~when 1st clock pulse arrives~~

$$D = 0 \therefore Q_0 = 0$$

$$\text{and } \overline{Q}_0 = 1$$

$\overline{Q}_0$  changes from  $0 \rightarrow 1$

i.e. +ve edge triggered

So, ~~Q<sub>1</sub> = 0~~ FF1 becomes

$$\text{active and } Q_1 = 1$$

$$\overline{Q}_1 = 0$$

But no change for FF2

~~when 3rd pulse arriving~~

$$D = 1 \therefore Q_0 = 1$$

$$\overline{Q}_0 = 0$$

$\overline{Q}_0$  changes from  $1 \rightarrow 0$

Hence, FF1 is not triggered.  
Same for FF2.

~~when 4th clock pulse arrives~~

$$D = 0 \therefore Q_0 = 0$$

$$\overline{Q}_0 = 1$$

$\overline{Q}_0$  changes from  $0 \rightarrow 1$

Hence, FF1 is triggered

$$\text{So, } Q_1 = 0 \quad \overline{Q}_1 = 1$$

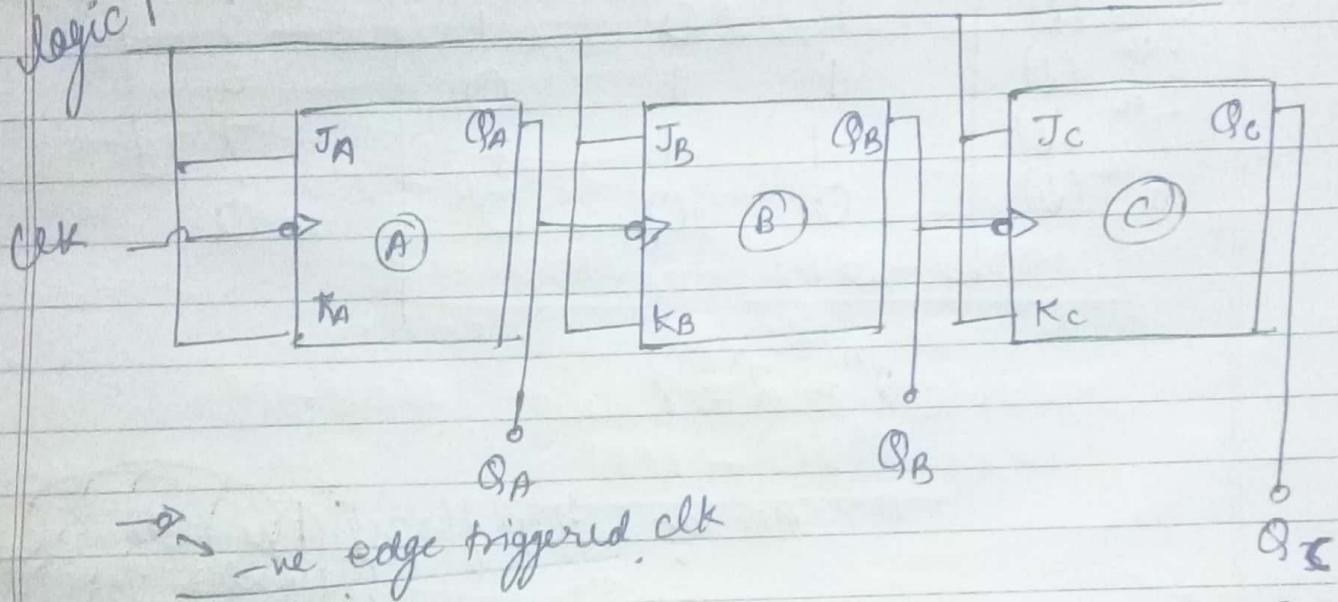
$\overline{Q}_1$  changes from  $0 \rightarrow 1$

Hence, FF2 is triggered

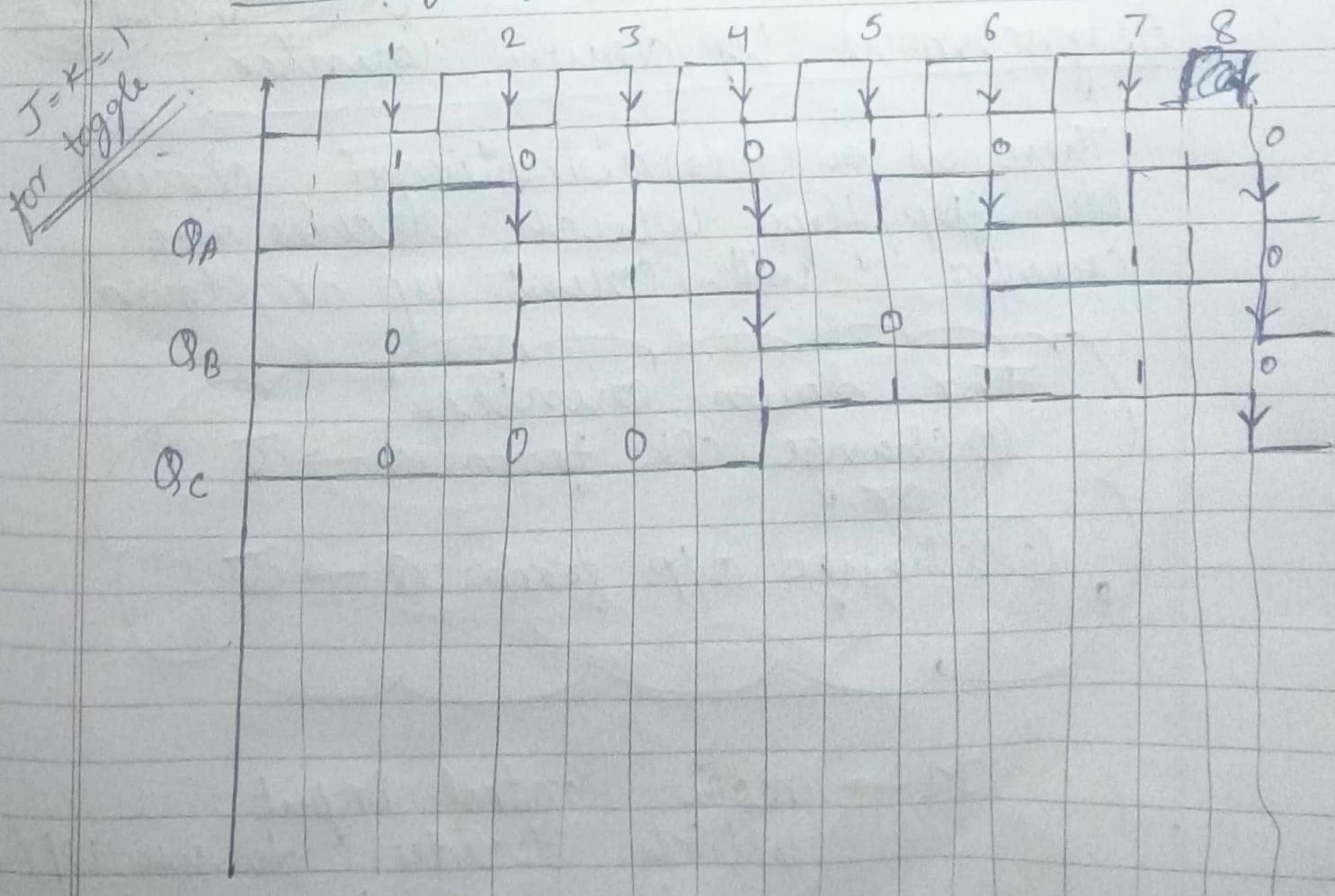
$$\text{and } Q_2 = 1.$$

# Binary Asynchronous Counter using J-K flip flop

logic



→ -ve edge triggered clk



| Clock               | $Q_A$ | $Q_B$ | $Q_C$ | Beg. eq. |
|---------------------|-------|-------|-------|----------|
| initial             | 0     | 0     | 0     | 0        |
| 1 <sup>st</sup> (↓) | 0     | 0     | 1     | 1        |
| 2 <sup>nd</sup> (↓) | 0     | 1     | 0     | 2        |
| 3 <sup>rd</sup> (↓) | 0     | 1     | 1     | 3        |
| 4 <sup>th</sup> (↓) | 1     | 0     | 0     | 4        |
| 5 <sup>th</sup> (↓) | 1     | 0     | 1     | 5        |
| 6 <sup>th</sup> (↓) | 1     | 1     | 0     | 6        |
| 7 <sup>th</sup> (↓) | 1     | 1     | 1     | 7        |
| 8 <sup>th</sup> (↓) | 0     | 0     | 0     | 0        |

$$2^n = 2^3 = 8$$

$n \rightarrow$  no. of flip flops

### Asynchronous Up/Down Counter

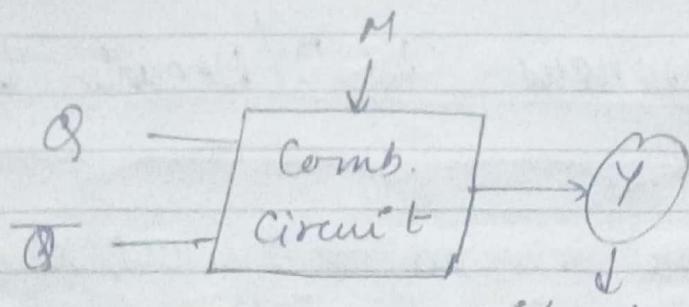
There is a combinational circuit b/w flip flops which decides the counter will count up or down.

For down counter.

① Change clk from  $Q \rightarrow \bar{Q}$   
or

② Change o/p from  $Q \rightarrow \bar{Q}$

M → mode control input  
which decides to use up/  
down counter.



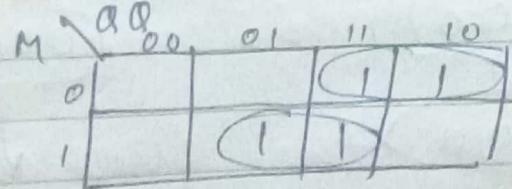
i/p for next  
FF.

$M = 0 \rightarrow \text{up}$

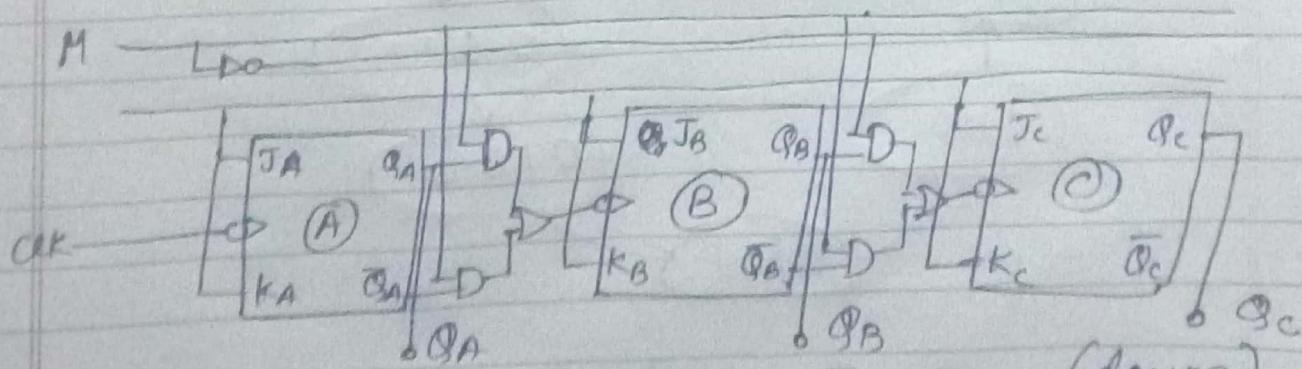
$M = 1 \rightarrow \text{down.}$

$M = 0 \quad \text{clk} = Q$   
 $M = 1 \quad \text{clk} = \bar{Q}$

| 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 |



$$Y = \bar{M}\bar{Q} + M\bar{Q}$$



$M = 1 \rightarrow \text{Counts from } 7 \rightarrow 0 \text{ (down)}$

$M = 0 \rightarrow \text{Counts from } 0 \rightarrow 7 \text{ (up).}$

# Asynchronous BCD/Decade Counter

Types

- ve edge triggered  $\rightarrow Q$  is clock  $\rightarrow$  UP counter
- ve edge triggered  $\rightarrow \bar{Q}$  is clock  $\rightarrow$  DOWN counter
- +ve edge triggered  $\rightarrow \bar{Q}$  is clock  $\rightarrow$  UP Counter
- +ve edge triggered  $\rightarrow Q$  is clock  $\rightarrow$  DOWN counter

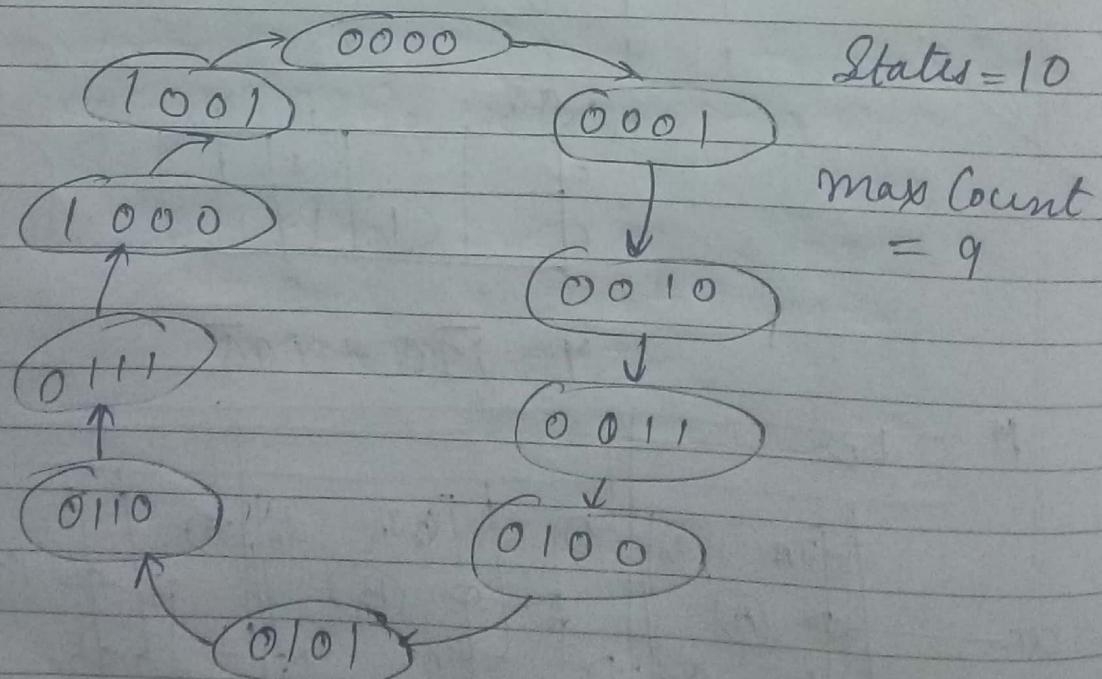
Modulus  $\rightarrow$  MOD  $\rightarrow$  No. of states.

MOD - M — MOD - N

↓  
MOD - MN

MOD - M

0, 1, 2, ..., M-1



## 4 Bit Counter

| clk | $Q_0$ | $Q_c$ | $Q_B$ | $Q_A$ |
|-----|-------|-------|-------|-------|
| 1   | 0     | 0     | 0     | 1     |
| 2   | 0     | 0     | 1     | 0     |
| 3   | 0     | 0     | 1     | 1     |
| 4   | 1     | 1     | 1     | 1     |
| 5   | 1     | 1     | 1     | 1     |
| 6   | 0     | 1     | 1     | 1     |
| 7   | 1     | 0     | 1     | 1     |
| 8   | 1     | 0     | 0     | 1     |
| 9   | 1     | 0     | 0     | 1     |
| 10  | 1     | 0     | 1     | 0     |
| 11  | 1     | 1     | 1     | 1     |
| 12  | 1     | 1     | 1     | 1     |
| 13  | 1     | 1     | 1     | 1     |
| 14  | 1     | 1     | 1     | 1     |
| 15  | 1     | 1     | 1     | 1     |

JK Flip flops has two values.

$$PST = 0 \quad CLR = 0$$

$$Q = 1 (\text{SET}) \quad Q = 0 (\text{RESET})$$

Set  $PST = 1$  as we don't need to set.

$$clr = 0 \text{ at } Q_0 = 1 \quad Q_c = 0 \\ Q_B = 1 \quad Q_A = 0$$

$$clr = \bar{Q}_0 + \bar{Q}_B + Q_c + Q_A$$

$$clr = (\bar{Q}_A \cdot Q_B \cdot \bar{Q}_c \cdot \bar{Q}_0)'$$

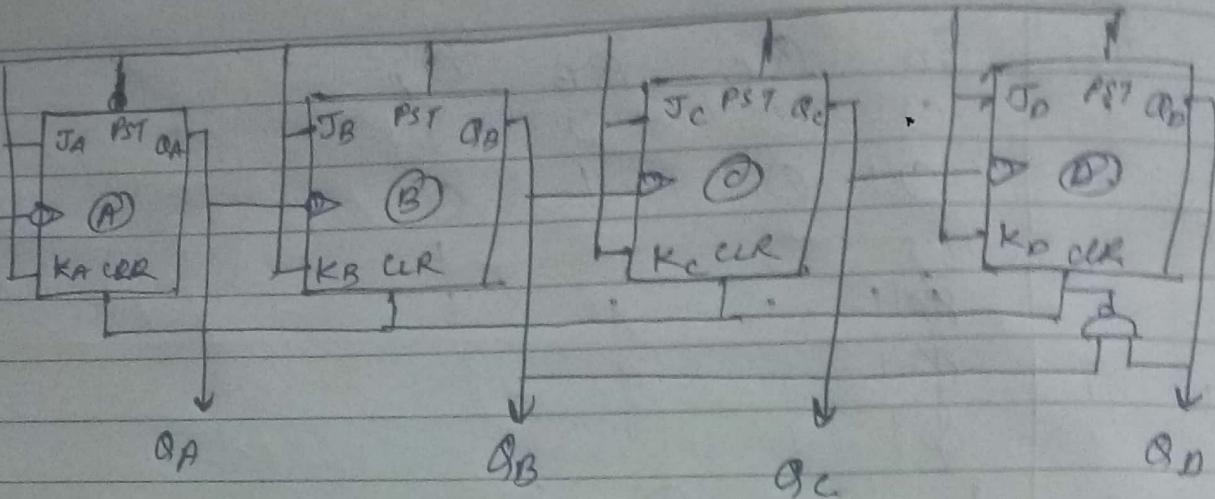
For 0-9 no case has

$$Q_0 = Q_B = 1 \text{ other than 10}$$

∴ we can eliminate  $Q_2$  and  $Q_c$  as  $Q_0$  and  $Q_B$  are enough

$$clr = (Q_0 \cdot Q_B)$$

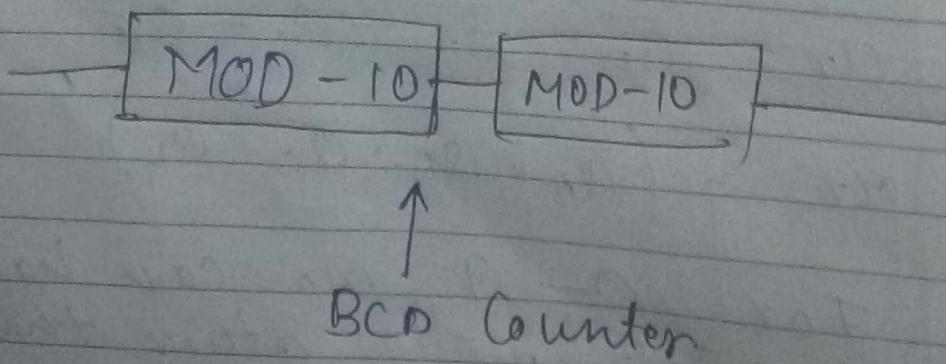
logic



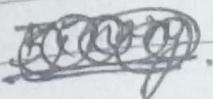
BCD Counter is a MOD-10 counter made from a MOD-16 counter.

Imp Ques can come from  
Converting one counter to another.  
eg → MOD6 using MOD8

use the similar steps.  
clr and preset with  
AND gates.



# Synchronous Counters



~~Final~~ Steps

- ① Decide no of Flip Flops
- ② Excitation table of flip flop.
- ③ State diagram and circuit excitation table.
- ④ Expression using K-Maps.
- ⑤ Logic Diagram.

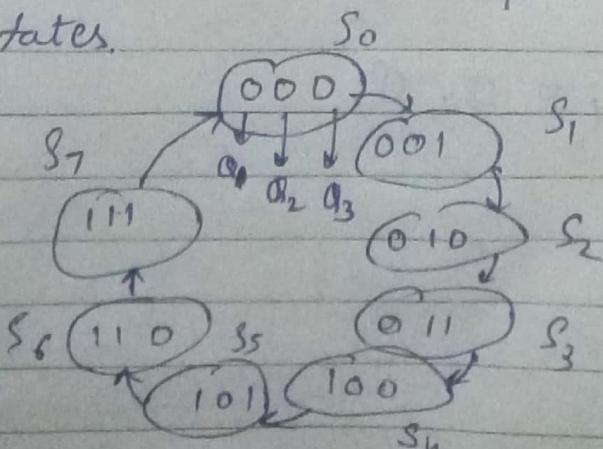
Binary (3-bit)  $\rightarrow$  up

① no of bits = no of flip flops.

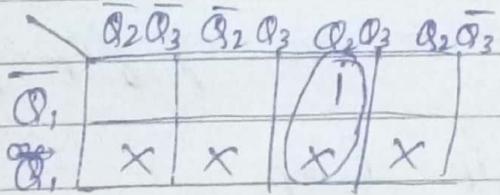
② JK Flip Flop.

| Q <sub>n</sub> | Q <sub>n+1</sub> | J | K |
|----------------|------------------|---|---|
| 0              | 0                | 0 | x |
| 0              | 1                | 1 | x |
| 1              | 0                | x | 1 |
| 1              | 1                | x | 0 |

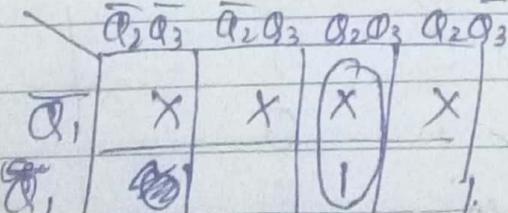
③  $2^3 = 8$  states.



| $\Phi_1$ | $\Phi_2$ | $\Phi_3$ | $\Phi_1^*$ | $\Phi_2^*$ | $\Phi_3^*$ | $J_1, K_1$ | $J_2, K_2$ | $J_3, K_3$ |
|----------|----------|----------|------------|------------|------------|------------|------------|------------|
| 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        | 1          | 1          | 0          | X 0        | 1 X        | X 1        |
| 1        | 1        | 0        | 1          | 1          | 1          | X 0        | X 0        | 1 X        |
| 1        | 1        | 1        | 0          | 0          | 0          | X 1        | X 1        | X 1        |

Q.  $J_1 =$  

$$J_1 = \Phi_2 \Phi_3$$

$K_1 =$  

$$K_1 = \Phi_2 \Phi_3$$

|             | $\bar{Q}_2 \bar{Q}_3$ | $\bar{Q}_2 Q_3$ | $Q_2 \bar{Q}_3$ | $Q_2 Q_3$ |  |
|-------------|-----------------------|-----------------|-----------------|-----------|--|
| $\bar{Q}_1$ | 1                     | X               | X               |           |  |
| $Q_1$       | 1                     | X               | X               |           |  |

$$J_2 = Q_3$$

|             | $\bar{Q}_2 \bar{Q}_3$ | $\bar{Q}_2 Q_3$ | $Q_2 \bar{Q}_3$ | $Q_2 Q_3$ |  |
|-------------|-----------------------|-----------------|-----------------|-----------|--|
| $\bar{Q}_1$ | X                     | X               | 1               |           |  |
| $Q_1$       | X                     | X               | 1               |           |  |

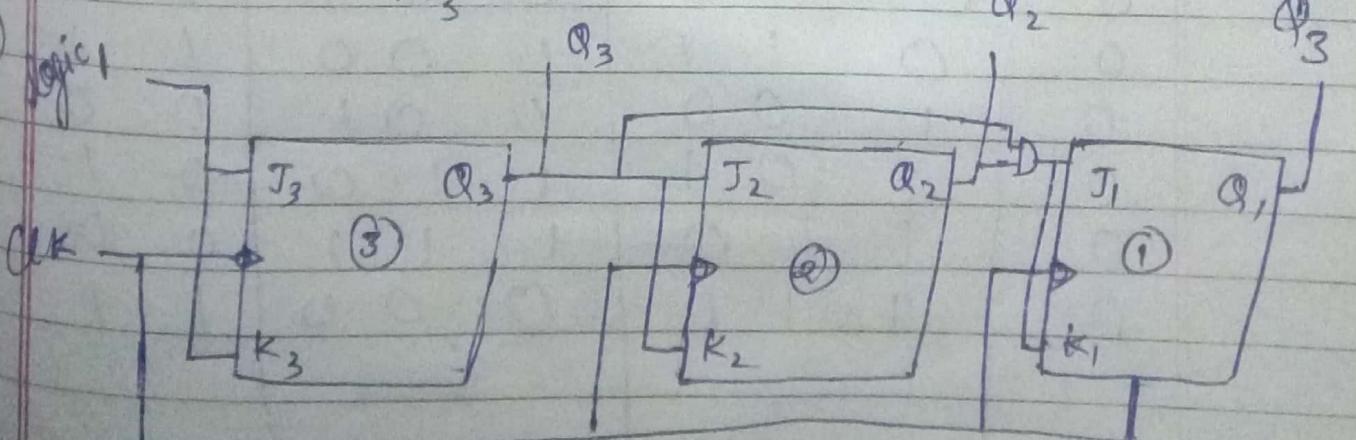
$$K_2 = Q_3$$

|             | $\bar{Q}_2 \bar{Q}_3$ | $\bar{Q}_2 Q_3$ | $Q_2 \bar{Q}_3$ | $Q_2 Q_3$ |  |
|-------------|-----------------------|-----------------|-----------------|-----------|--|
| $\bar{Q}_1$ | 1                     | X               | X               | 1         |  |
| $Q_1$       | 1                     | X               | X               | 1         |  |

$$J_3 = 1$$

|  | X | 1 | 1 | X |
|--|---|---|---|---|
|  | X | 1 | 1 | X |

$$K_3 = 1$$



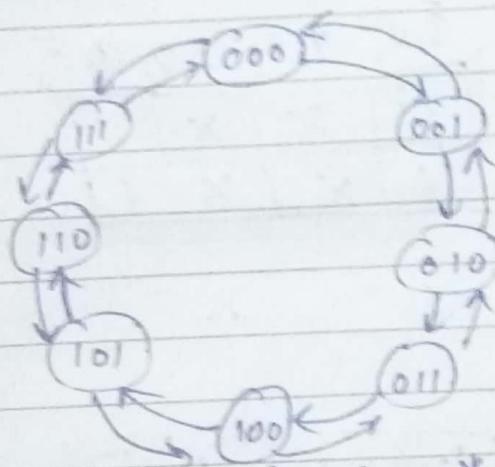
# Up / Down Counter

Using T-flip flops.

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

$M=0 \rightarrow \text{up}$

$M=1 \rightarrow \text{Down}$



| M | $Q_C$ | $Q_B$ | $Q_A$ | $Q_C^*$ | $Q_B^*$ | $Q_A^*$ | $T_C$ | $T_B$ | $T_A$ |
|---|-------|-------|-------|---------|---------|---------|-------|-------|-------|
| 0 | 0     | 0     | 0     | 0       | 0       | 1       | 0     | 0     | 1     |
| 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     |

| M | $Q_C$ | $Q_B$ | $Q_A$ | $Q_C^*$ | $Q_B^*$ | $Q_A^*$ | $T_C$ | $T_B$ | $T_A$ |
|---|-------|-------|-------|---------|---------|---------|-------|-------|-------|
| 1 | 0     | 0     | 0     | 1       | 1       | 1       | 1     | 1     | 1     |
| 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     | 0     | 1     |

$$T_A = 1.$$

for  $T_B$   $M Q_C$

|               | $\bar{Q}_B Q_A$ | $\bar{Q}_A \bar{Q}_B$ | $\bar{Q}_A Q_A$ | $Q_B Q_B$ | $Q_B \bar{Q}_A$ |
|---------------|-----------------|-----------------------|-----------------|-----------|-----------------|
| $M Q_C$       | 1               | 1                     | 1               | 1         | 1               |
| $M \bar{Q}_C$ | 1               | 1                     | 1               | 1         | 1               |
| $M Q_C$       | 1               | 1                     | 1               | 1         | 1               |
| $M \bar{Q}_C$ | 1               | 1                     | 1               | 1         | 1               |

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

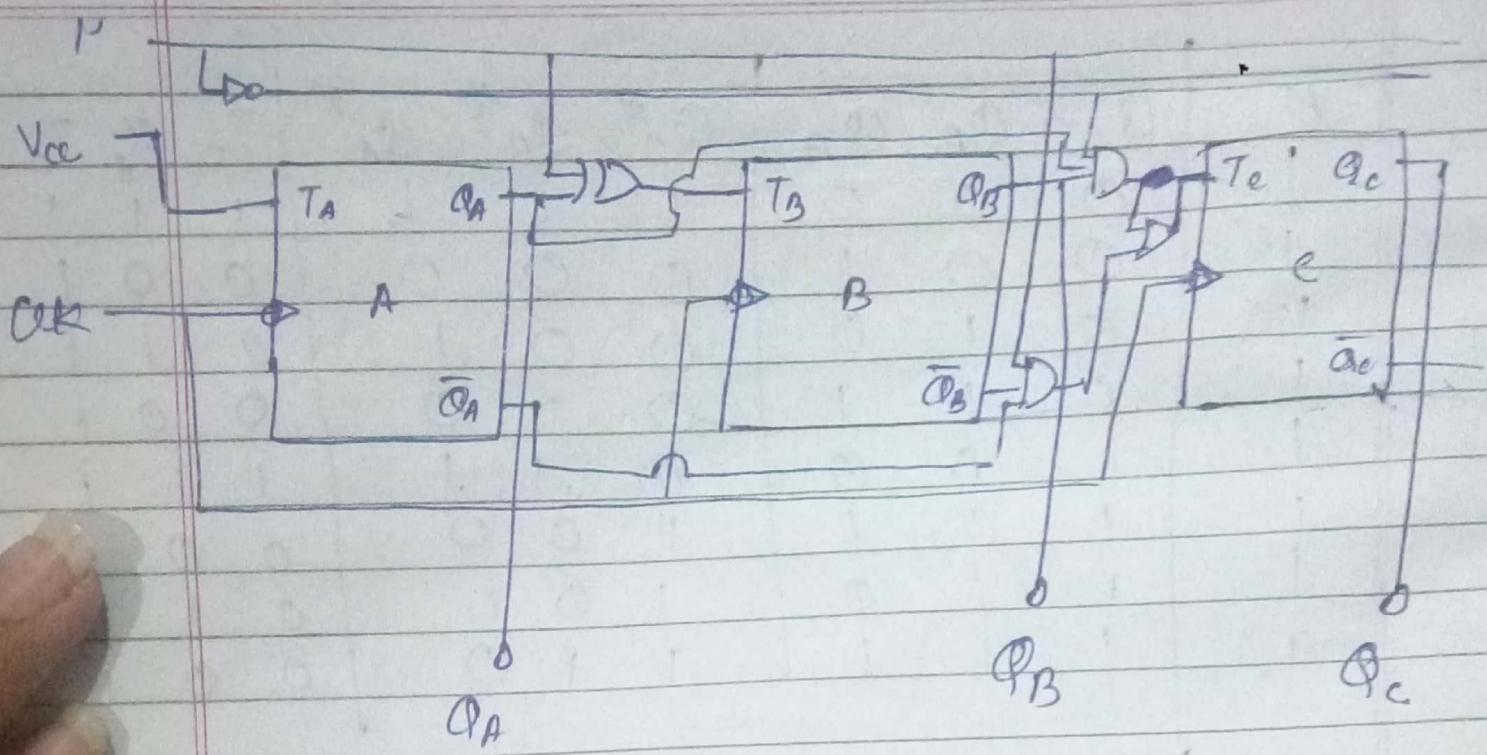
$$= M (\bar{Q}_A Q_A)$$

$T_C =$

|               | $\bar{Q}_B \bar{Q}_A$ | $\bar{Q}_B Q_A$ | $Q_B Q_A$ | $Q_B \bar{Q}_A$ |
|---------------|-----------------------|-----------------|-----------|-----------------|
| $M \bar{Q}_C$ | 1                     | 1               | 1         | 1               |
| $M Q_C$       | 1                     | 1               | 1         | 1               |
| $M Q_C$       | 1                     | 1               | 1         | 1               |
| $M \bar{Q}_C$ | 1                     | 1               | 1         | 1               |

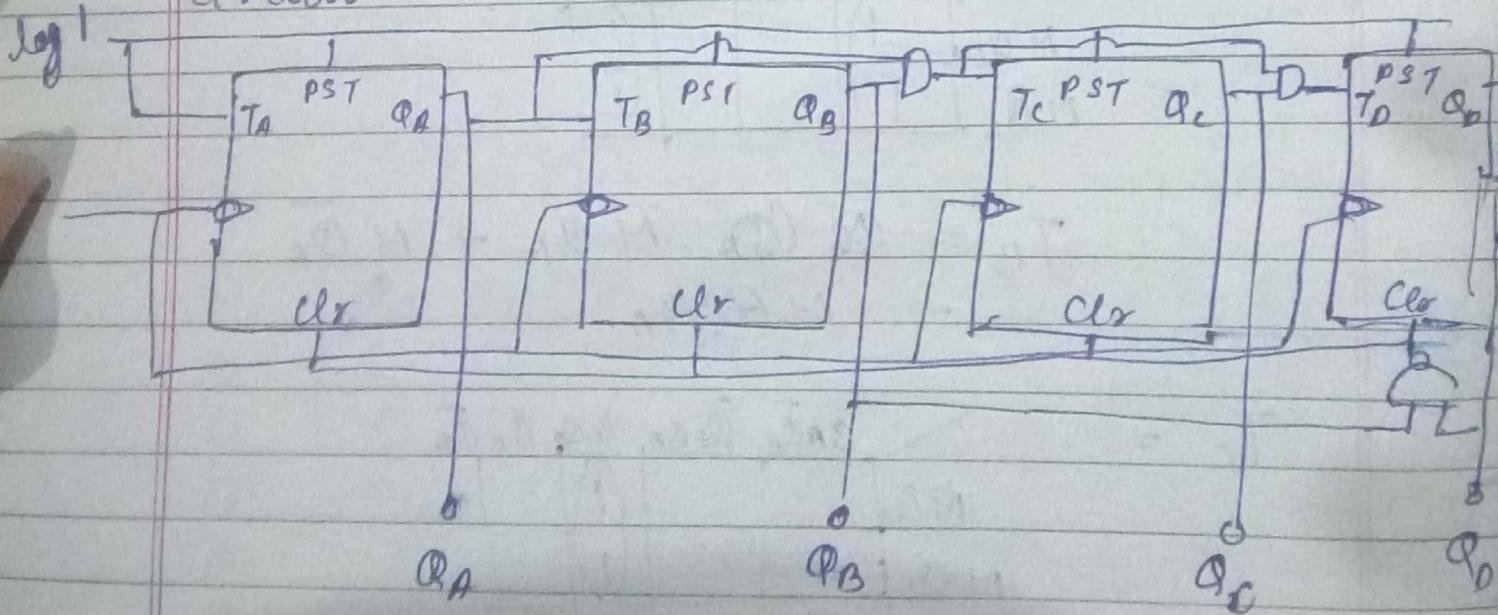
$$T_C = M \bar{Q}_B \bar{Q}_A + M Q_A Q_B$$

~~error~~



BCD Counter Can be designed similarly.

Circuit



BCD Counter

## Registers

A register is a group of flip flops to store n-bit of information.

It is a synchronous sequential circuit. It requires synchronise clock pulse and can transfer the information.

A register stores and shifts the information. It is of two types.

- ① Serial →  $\overset{\text{In}}{\rightarrow}$   
                →  $\overset{\text{Out}}{\rightarrow}$
- ② Parallel →  $\overset{\text{In}}{\rightarrow}$   
                →  $\overset{\text{Out}}{\rightarrow}$

When Shifting right → Right Shift

When Shifting left → left shift

\* n bit register has n-flip flops.

⇒ have no specified sequence of state ~~e.g.~~ except in a specialised application.

# Types

$S_1, S_0$   
Series in  
Series out

$S_1, P_0$

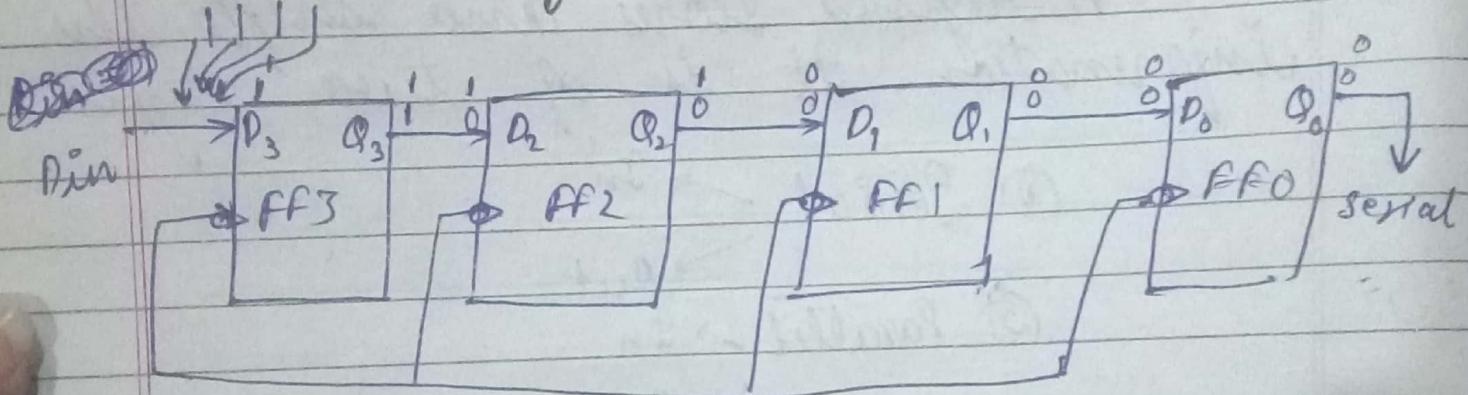
Series In  
Parallel out  
Parallel out Series out

$P_1, S_0$

$P_1, P_0$   
Parallel in  
Parallel out.

$S_1 S_0$

Shift Right



CK      D       $Q_{n+1}$

0      x       $Q_n$

1      0      0

1      1      1

Clk       $Q_3$        $Q_2$        $Q_1$        $Q_0$

initially      0      0      0      0

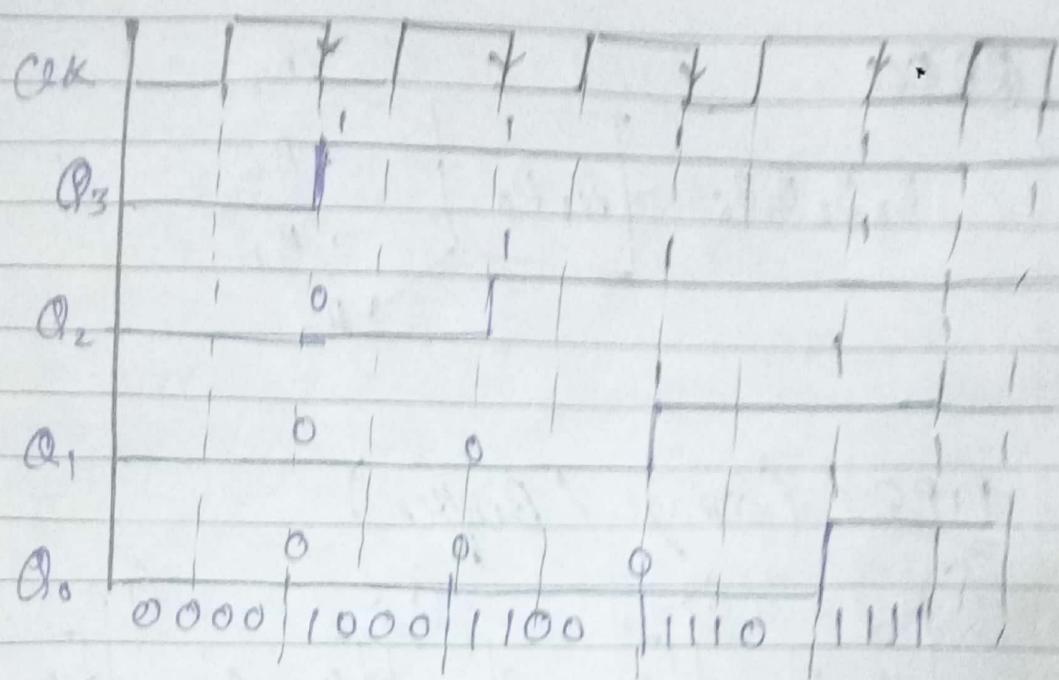
↓      1      0      0      0

↓      1      1      0      0

↓      1      1      1      0

↓      1      1      1      1

4 Clock pulses required to store the data.

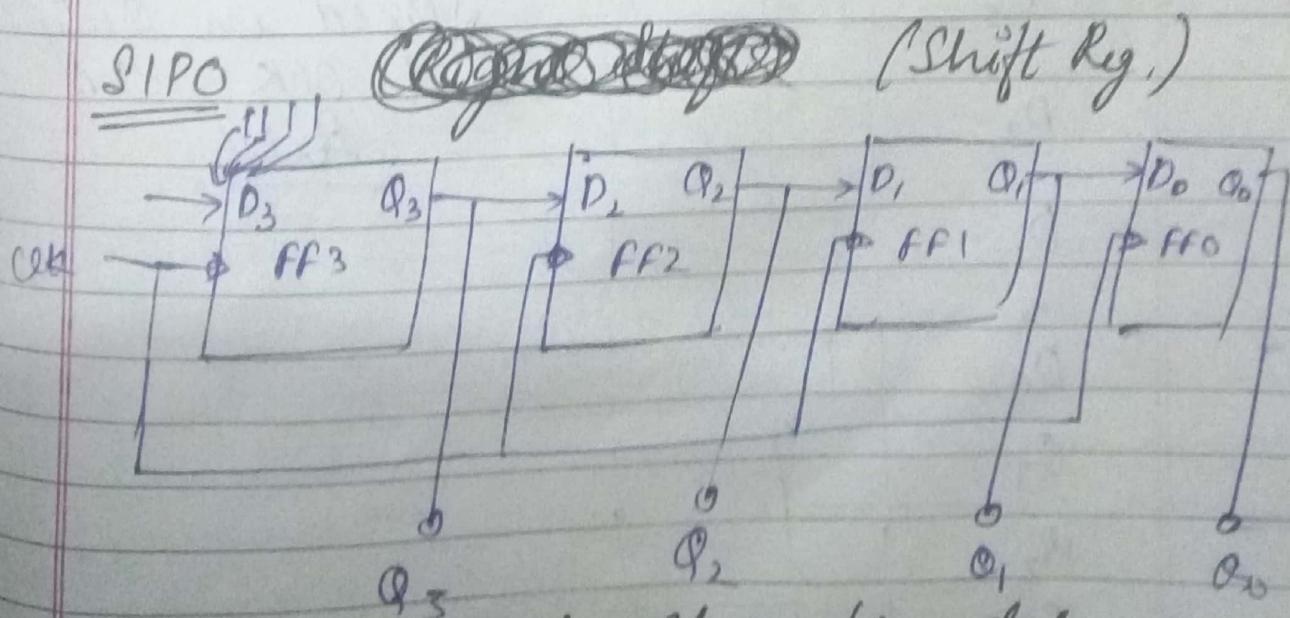


left Shift can be made similarly.

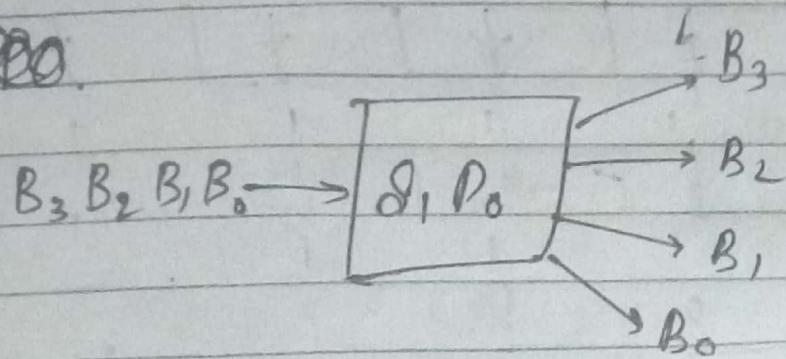
~~8000~~ output → 1111b000

need 8 clock pulses  
to get the final output.

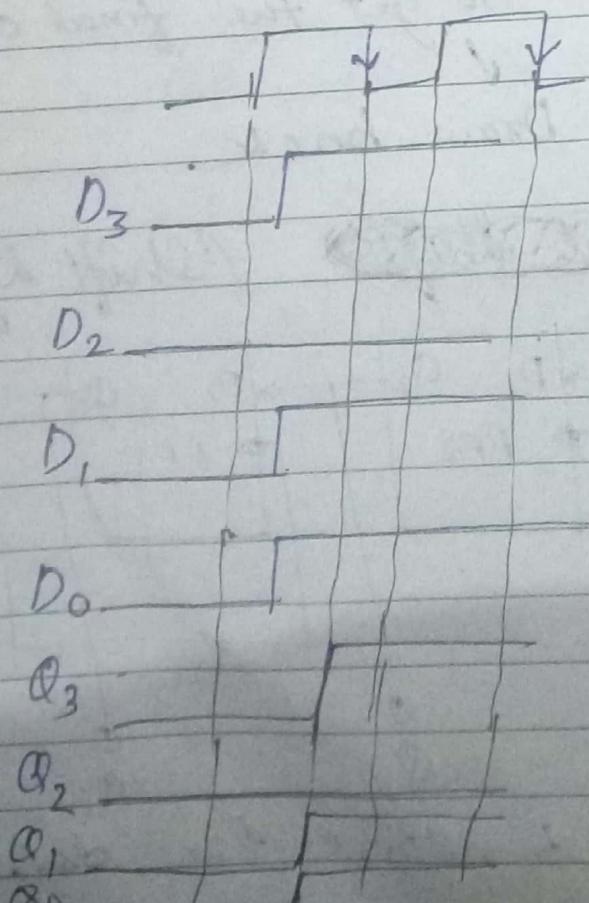
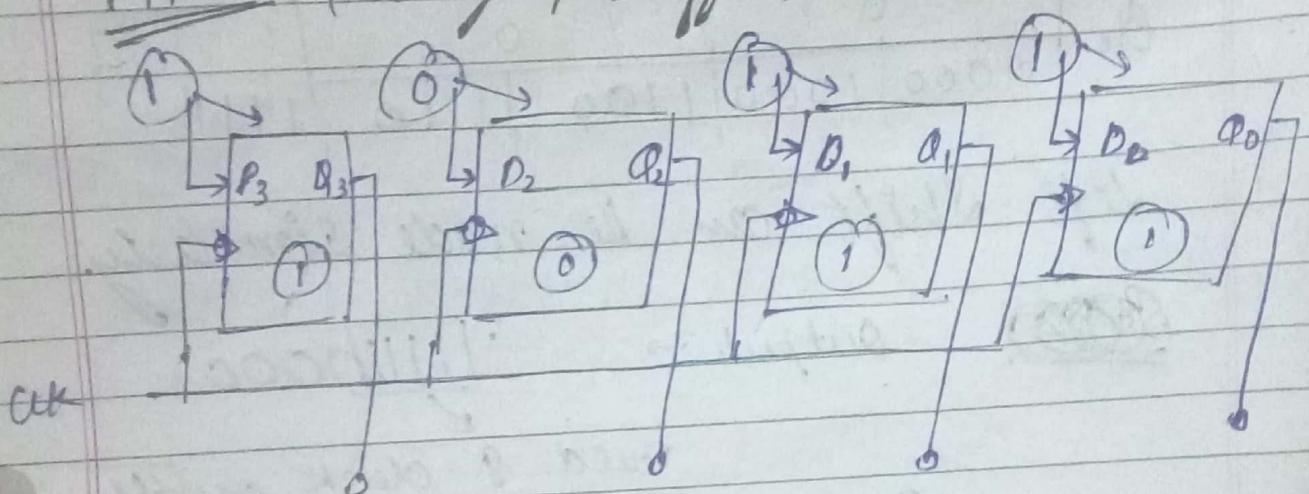
Draw back.



4 clock pulses to store the data.

R0000

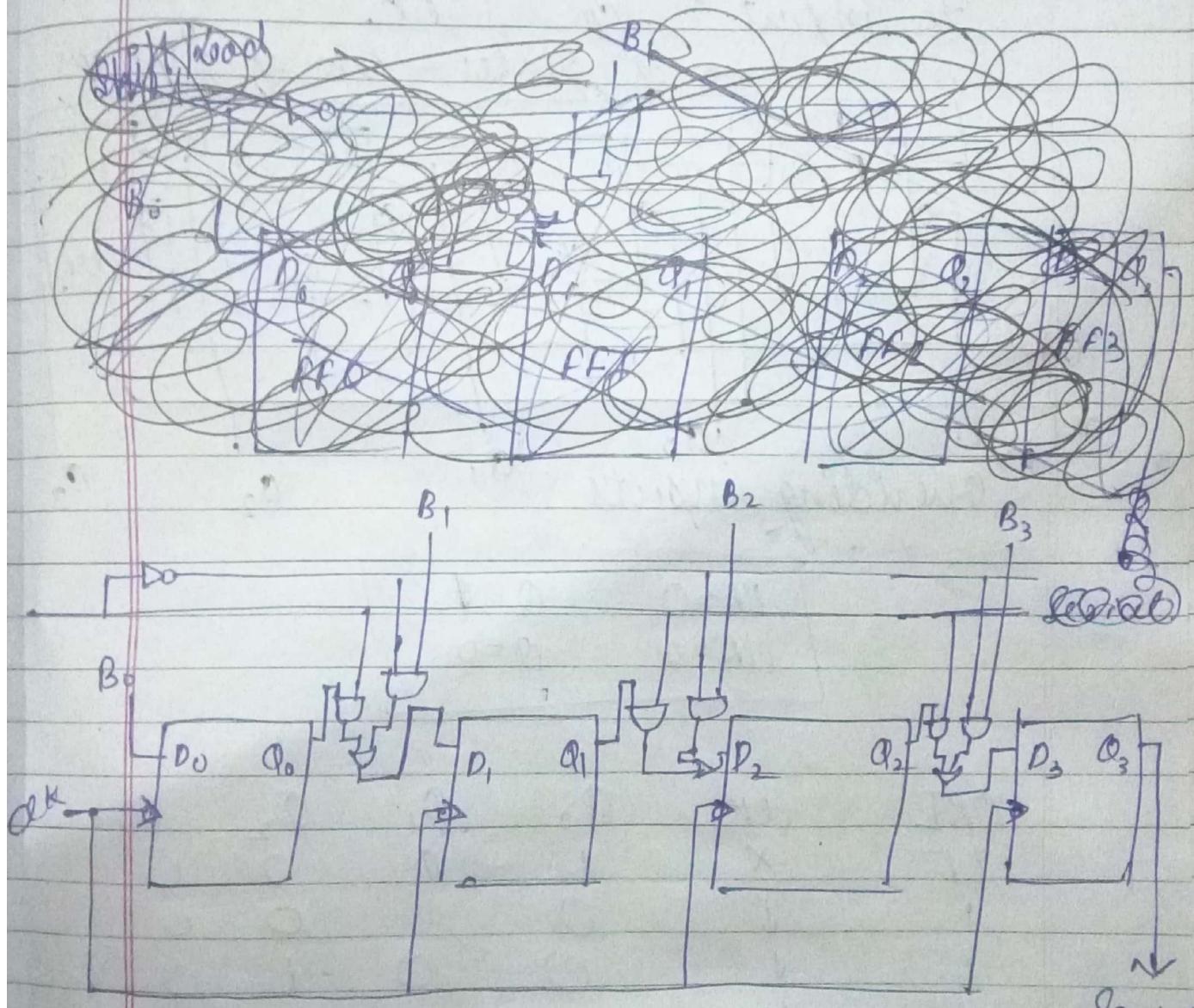
P1PO (Storage / Buffer)



Stored in only  
i- CLK  
Pulse.

# P1 So (Shift Register)

- 1) Load Mode at 0
- 2) Shift Mode at 1



If  $S = 1$ .

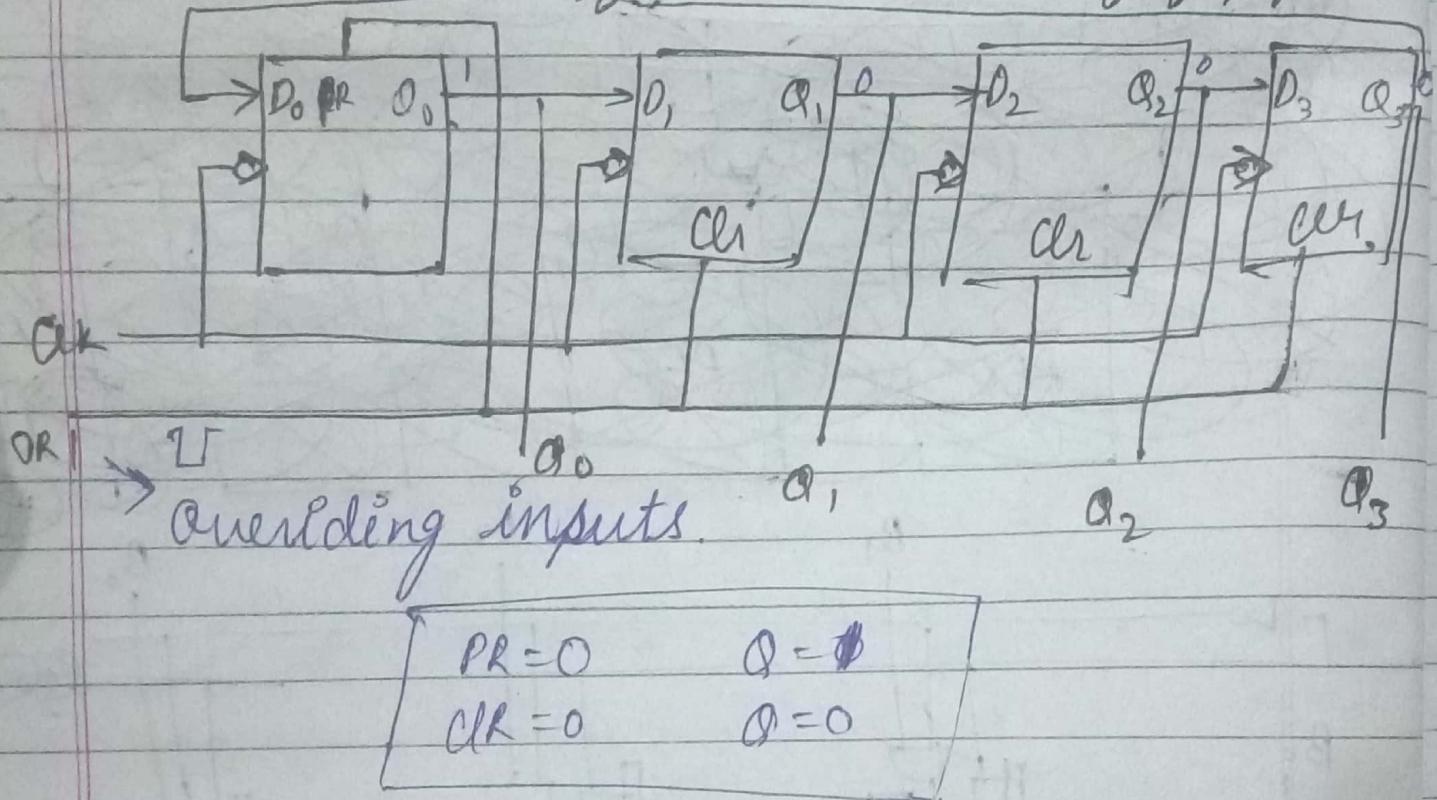
$D_1 = Q_0 ; D_2 = \emptyset , D_3 = Q_2$   
(Shifting.)

If  $S = 0$  (Loading)

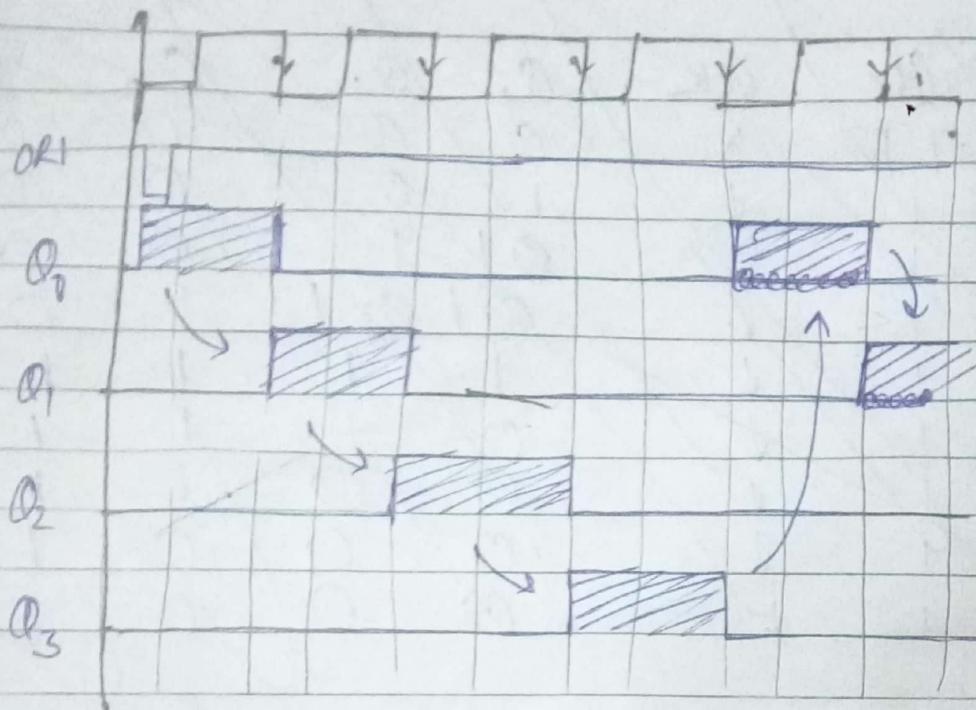
$D_0 = B_0 ; D_1 = B_1 , D_2 = B_2 , D_3 = B_3$

# Ring Counter

- ① Application of shift register.
- ② Output of last FF is conn to input of 1<sup>st</sup> FF.  
no of states = no of flip flops



| ORI | CK | $Q_0$ | $Q_1$ | $Q_2$ | $Q_3$ |
|-----|----|-------|-------|-------|-------|
| 75  | X  | 1     | 0     | 0     | 0     |
| 1   | ↓  | 0     | 1     | 0     | 0     |
| 1   | ↓  | 0     | 0     | 1     | 0     |
| 1   | ↓  | 0     | 0     | 0     | 1     |
| 1   | ↓  | 1     | 0     | 0     | 0     |



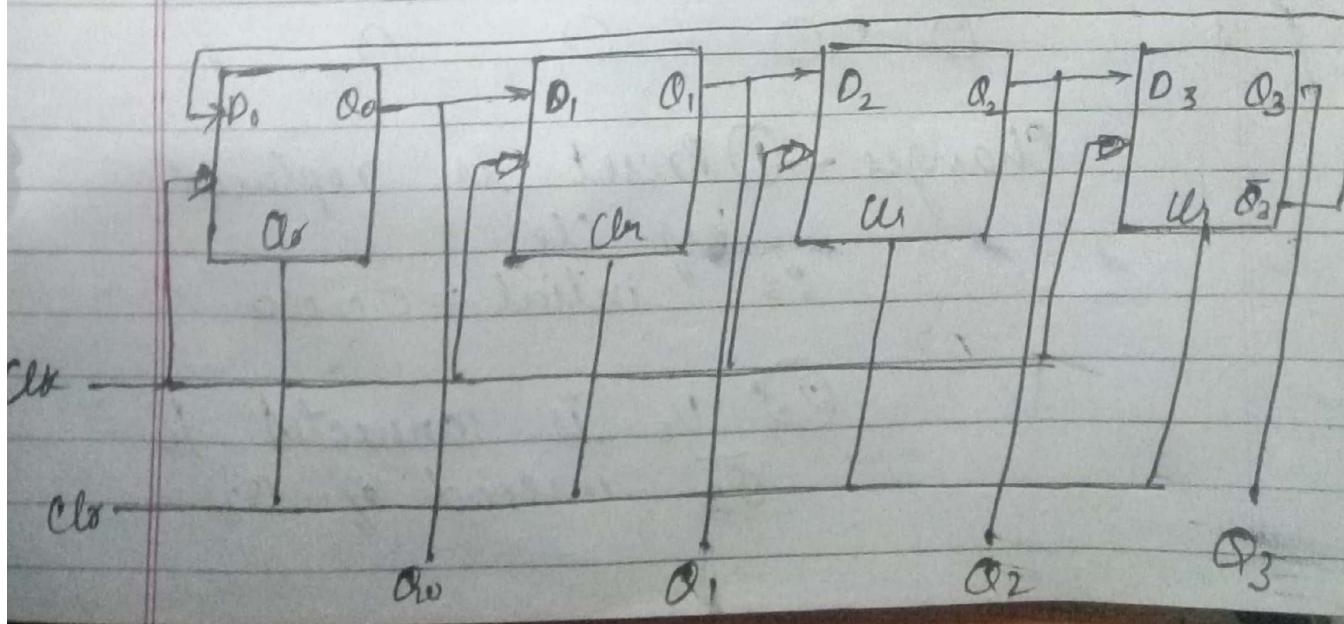
1 shift 1 bit per clk pulse.

### Johnson Counter

[Twisted / Switched tail Ring Counter]

$$\text{no of states} = 2^n$$

$n = \text{no of flip flops.}$



| Clr | QK | $Q_0$ | $Q_1$ | $Q_2$ | $Q_3$ |   |
|-----|----|-------|-------|-------|-------|---|
| 15  | X  | 0     | 0     | 0     | 0     | 0 |
| 1   | ↓  | 1     | 0     | 0     | 0     | 1 |
| 1   | ↓  | 1     | 1     | 0     | 0     | 2 |
| 1   | ↓  | 1     | 1     | 1     | 0     | 3 |
| 1   | ↓  | 1     | 1     | 1     | 1     | 4 |
| 1   | ↓  | 0     | 1     | 1     | 1     | 5 |
| 1   | ↓  | 0     | 0     | 1     | 1     | 6 |
| 1   | ↓  | 0     | 0     | 0     | 1     | 7 |
| 1   | ↓  | 0     | 0     | 0     | 0     |   |

Changes → ① Preset is replaced by Clr.  
 $\therefore$  initial = 0000

②  $Q_0$  is connected to  $\bar{Q}_3$  instead of  $Q_3$

