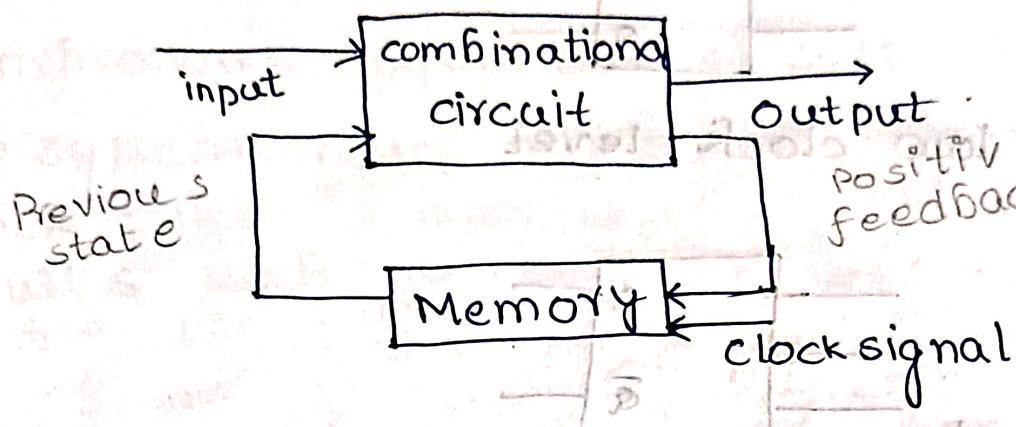


710125

Sequential circuits:-

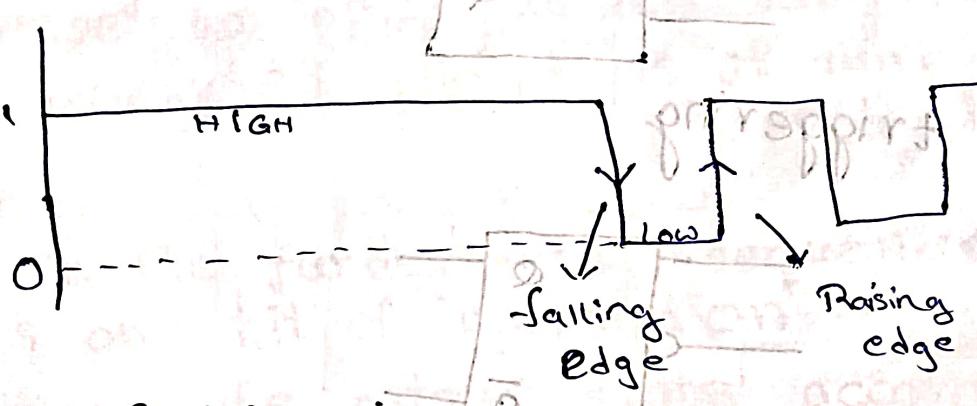
In sequential circuit the output depends on present inputs as well as present inputs and previous output.



Need of triggering:-

* A clock pulse activates the circuit and then it changes the state based on input and previous output.

* Triggering ensures the precise timing and synchronization of state changes which is important for reliable operation.



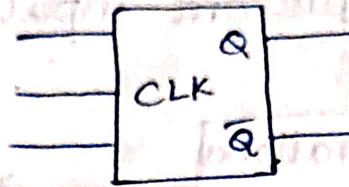
Types of triggering:

There are two types of triggering

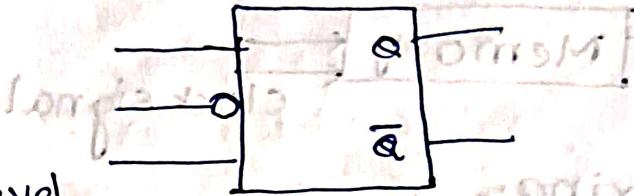
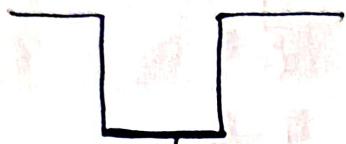
1. Pulse triggering (or) Level triggering.
2. Edge triggering

Pulse triggering.

a. High level triggering.

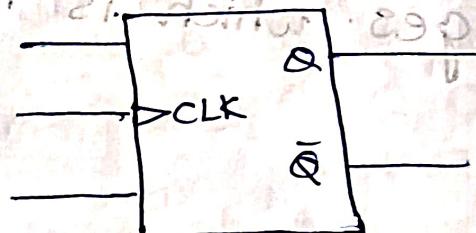


b. Triggers on low clock level

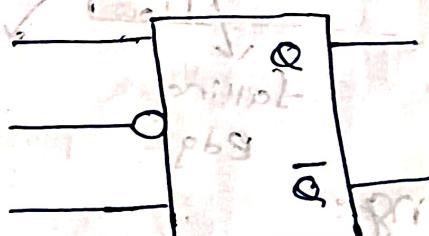
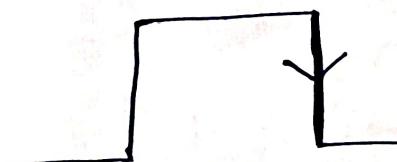


c. Edge triggered

a. Positive edge triggering



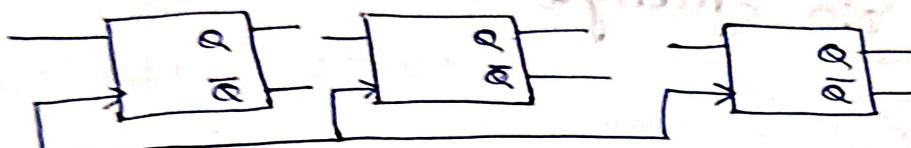
b. Negative edge triggering



Types of sequential -
sequential circuits are classified into 2 types
1. synchronous sequential circuit
2. asynchronous sequential circuit

Synchronous sequential circuit

In synchronous sequential circuits common clock pulse is applied among all stages which results uniform state changes.



A clock pulse is a regular, repeating signal used in digital circuit

Asynchronous sequential circuit:-

Asynchronous sequential circuit does not rely on global clock signal for its operation.

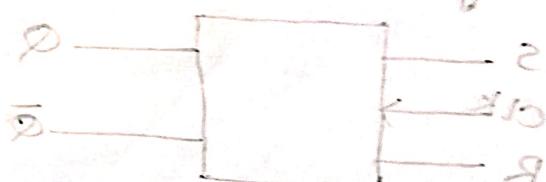
It changes the state and produce outputs in direct response to input changes using memory elements like unclocked flip flops or time delays.

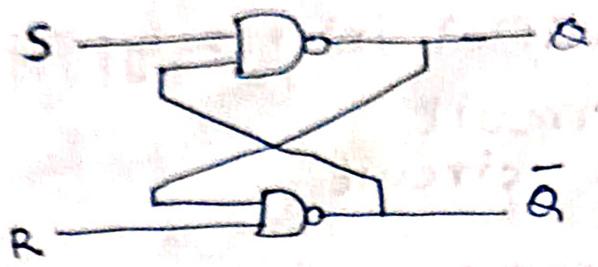
Latch:

Latch is the fundamental sequential circuit that stores one bit of information

It changes the output state according to the input instantaneously

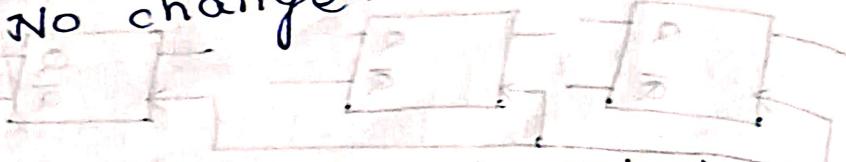
SR latch (Set Reset latch) :-





Truth table :-

S	R	Q	\bar{Q}
0	0	Not used	
0	1	0	1
1	0	1	0
1	1	No change	



Flip-flop :-

Flip-flop is a sequential circuit that stores 1 bit of data (0 or 1) and it has 2 stable states. It retains its state until an input signal synchronizes with a clock pulse.

Types of flip flops:-

There are 4 different types of flip flops in sequential circuit.

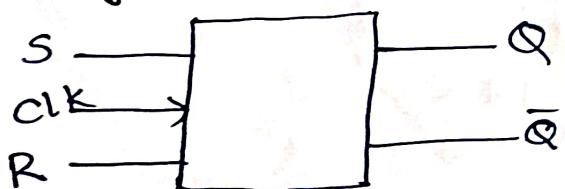
- 1. SR flip-flop
- 2. JK flip-flop

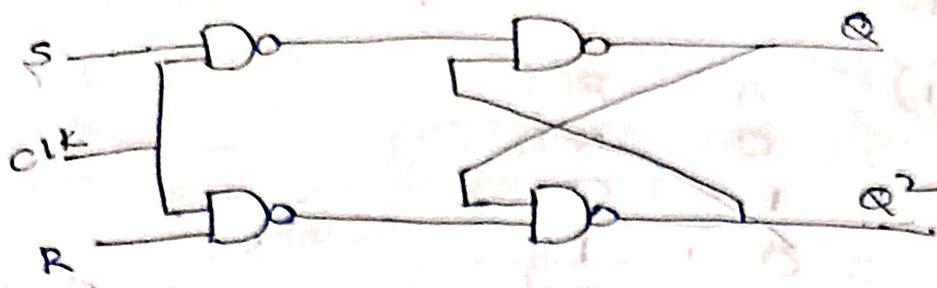
3. D flip flop

4. T flip-flop

SR flip-flop :-

Logic symbol :-





Truth table:

S	R	Q	\bar{Q}	Notes
0	0	0	1	No change
0	1	0	1	
1	0	1	0	
1	1	X	X	

S	R	Q(t+1)	Q(t)	Notes
0	0	0	0	
0	1	0	1	
1	0	1	0	
1	1	1	1	Invalid

Characteristic table:-

S	R	Q(t)	Q(t+1)
0	0	0	0
0	1	0	1
1	0	1	0
1	1	0	0
1	1	1	1
1	1	1	0
1	1	1	1

Invalid
Invalid

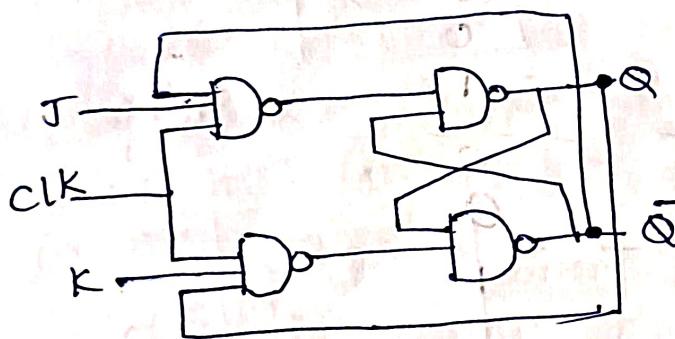
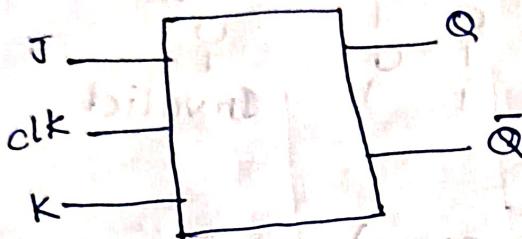
Characteristic equation:-

S	RQ	$\bar{R}\bar{Q}$	$\bar{R}Q$	$R\bar{Q}$
\bar{S}	0	1	3	0
S	1	5	X	7

$$S + \overline{R}Q(t)$$

$Q(t)$	$Q(t+1)$	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

JK flip-flop:-



Truth table:

JK	$Q(t+1)$
	$Q(t)$
0 0	0
0 1	1
1 0	1
1 1	<u>$Q(t)$</u>

characteristic equation $Q(t+1) = \bar{K}Q + JQ(t)$

J	K	$Q(t)$	$Q(t+1)$
0	0	0	0
0	0	1	1
0	-1	0	0
0	1	1	0
0	1	0	1
1	0	1	1
0	1	0	1
0	1	1	0

characteristic equation $Q(t+1) = \bar{K}Q + JQ(t)$

		$\bar{K}Q$	$\bar{J}Q$	KQ	$\bar{K}Q + \bar{J}Q$
		0	1	1	0
		1	0	0	1
J	0	0	1	1	0
J	1	1	0	0	1
K	0	0	1	1	0
K	1	1	0	0	1

$Q(t+1) = \bar{K}Q + JQ(t)$

Excitation tables-

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0



Drawback JK flip-flop:-

In JK flip flop when $J=K=1$ and if clock = 1 for a long period of time then output will be toggled as long as clock remains high. This makes the output unstable or uncertain.

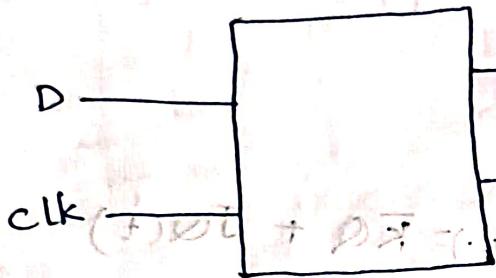
This condition is called as ~~rays~~ around condition in JK flip-flop.

We can overcome this problem by making clock is equal to one for very less duration.

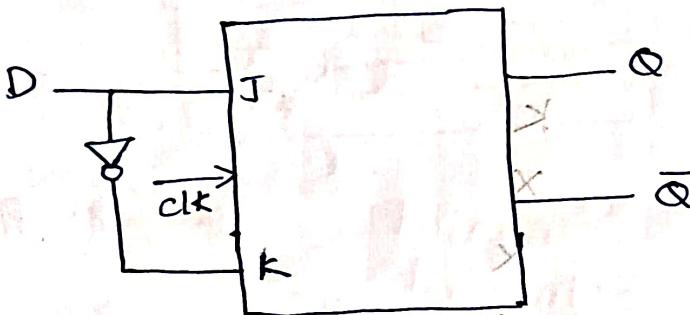
1410125

D-flip-flop

D flip-flop is known as data flip-flop or delay flip-flop.



Logic diagram:



Truth table:

D	Q(t+1)
0	0

D	$Q(t)$	$Q(t+1)$
0	0	0
0	1	0
1	0	1
1	1	1

Expressions:-

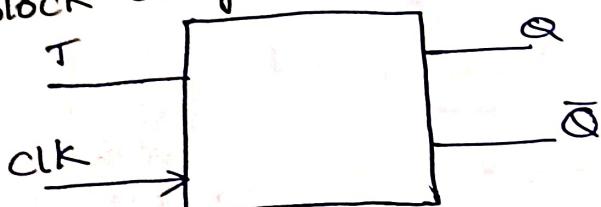
$$Q(t+1) = D \bar{Q}(t) + \bar{D}Q(t) = D(\bar{Q}(t) + Q(t)) = D.$$

Excitation table:

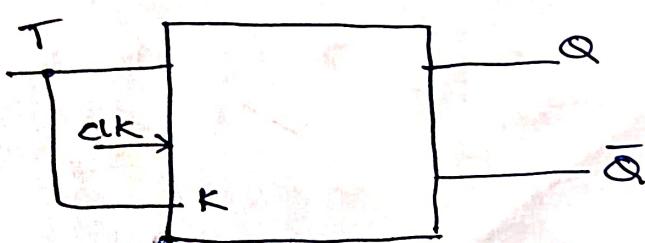
$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

T-flip flop:-

T-flip flop is also called as toggle
block diagram



logic diagram:



T	$Q(t+1)$
O	$Q(t)$
I	$\bar{Q}(t)$

Characteristic table:-

T	$Q(t)$	$Q(t+1)$
O	0	0
I	1	1
T	0	1
I	1	0

for $T=0$ $Q(t+1)=\bar{Q}(t)$

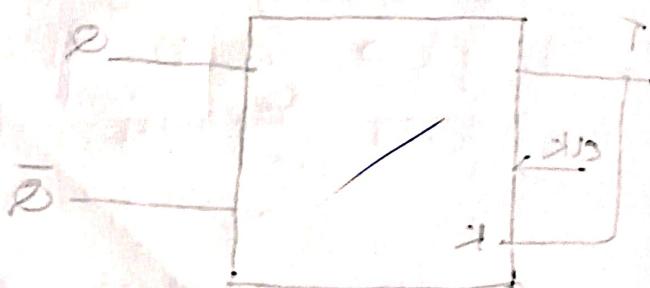
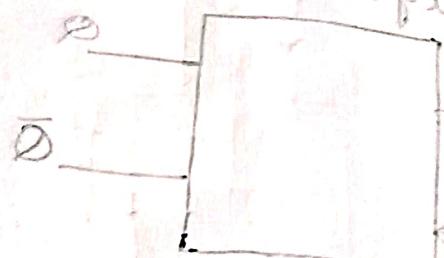
for $T=1$ $Q(t+1)=\bar{Q}(t)$

Characteristic expression:-

$$\begin{aligned} Q(t+1) &= \bar{T} Q(t) + T \bar{Q}(t) \\ &= T \oplus Q(t) \end{aligned}$$

Excitation table:-

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0



flip-flop conversions:-
to D flip flop

flip-flop conversion:-

1. J-K to D flip flop conversion
 step 1: Draw the characteristic table of D-flip flop
 and excitation table of J-K flip flop

Char	table	$Q(t)$	$Q(t+1)$
D		0	0
O		1	0
O		0	1
I		1	1

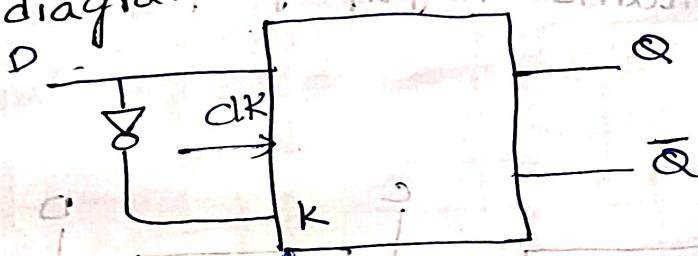
excitation table		J	K
Q(t)	Q(t+1)		
0	0	0	x
0	1	1	x
1	0	x	-
1	1	x	0

15/10/29

D	D̄	O	Ō	X
D	D̄	O	Ō	X ₁
D̄	D	Ō	O	X ₂
D̄	D̄	D̄	D	X ₃

A hand-drawn logic diagram of a 2-to-4 decoder. The inputs are labeled D_1 and D_2 . The output D is shown as a box containing a switch. The switch has two positions: one connected to ground and one connected to the output line. The output line also connects to a label \bar{D} . The output D is labeled Q . The enable input $Q\bar{Q}$ is connected to the switch. The switch is controlled by the expression $\bar{Q}Q + Q\bar{Q}$.

logic



J-K to T flip flop conversion
characteristics of T=K

2. J-K to excitation table of J-K.

Characteristic table of T

EXCITATION		J	K
$Q(t)$	$Q(t+1)$		
0	0	0	X
0	-	-	X
-	0	X	-1
-	1	X	0

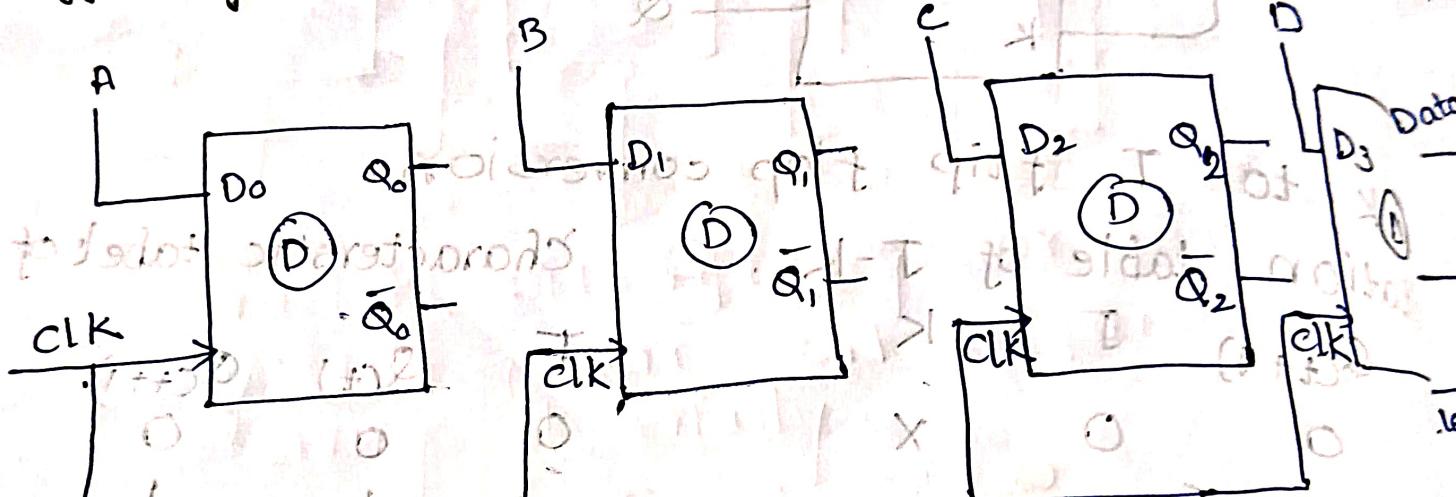
A shift linear f
together one to
Direction types

→ Ring

~~Shift regt~~ Register:-

- * A group of flip-flops can be used to store a ~~bi~~ bidirectional serial (n no. of bits) which is called register.
- * An n bit register contains n flip-flops

Buffer register



~~shift registers:-~~

A shift register provides data shifting function from one flip flop to another.

A shift register is a group of flip-flops set up in a linear fashion with their inputs and outputs connected together in such a way that data is shifted from one to another when clock input is applied.

~~Directions~~
~~Types of shifting operations in shift registers:-~~

Right shift

Bidirectional shift

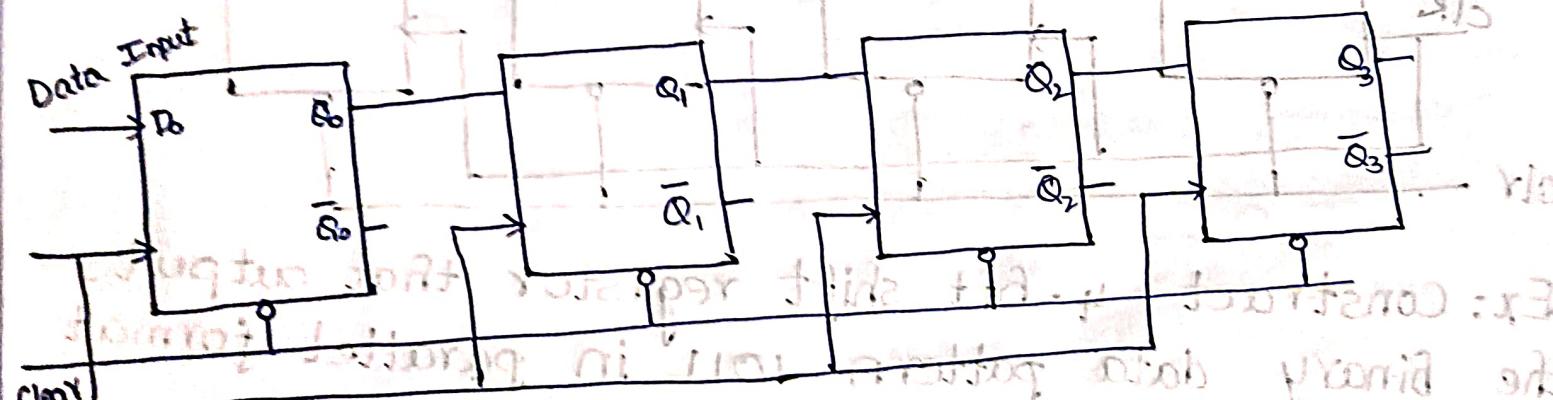
1. Serial in and serial out shift register

2. Serial in, parallel out shift registers

3. Parallel in serial out shift registers

4. Parallel in parallel out shift register

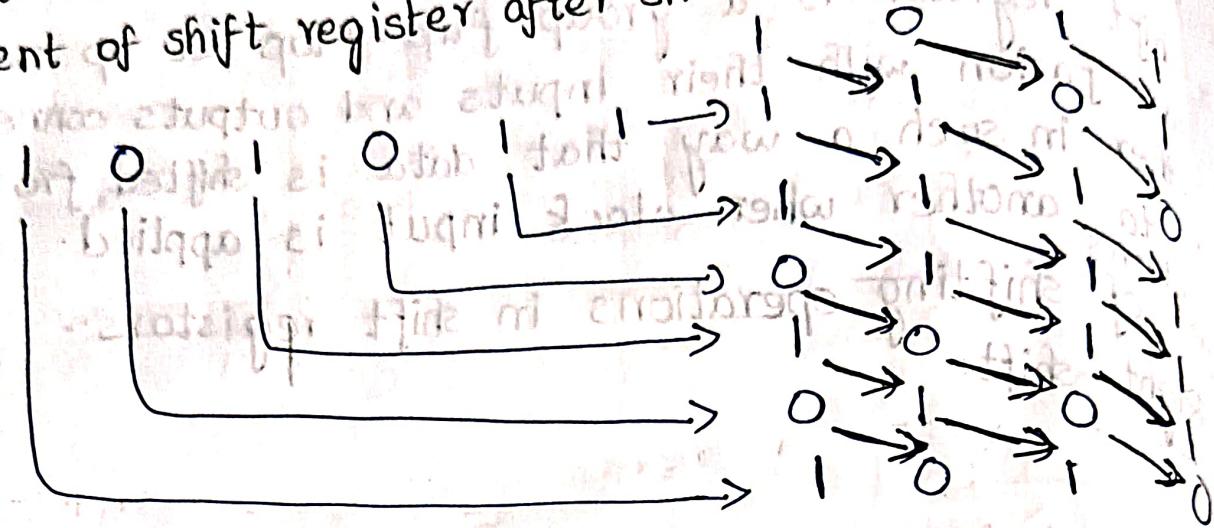
→ Serial in and serial out shift register



Serial in serial out shift register stores and shifts the data in serial manner.

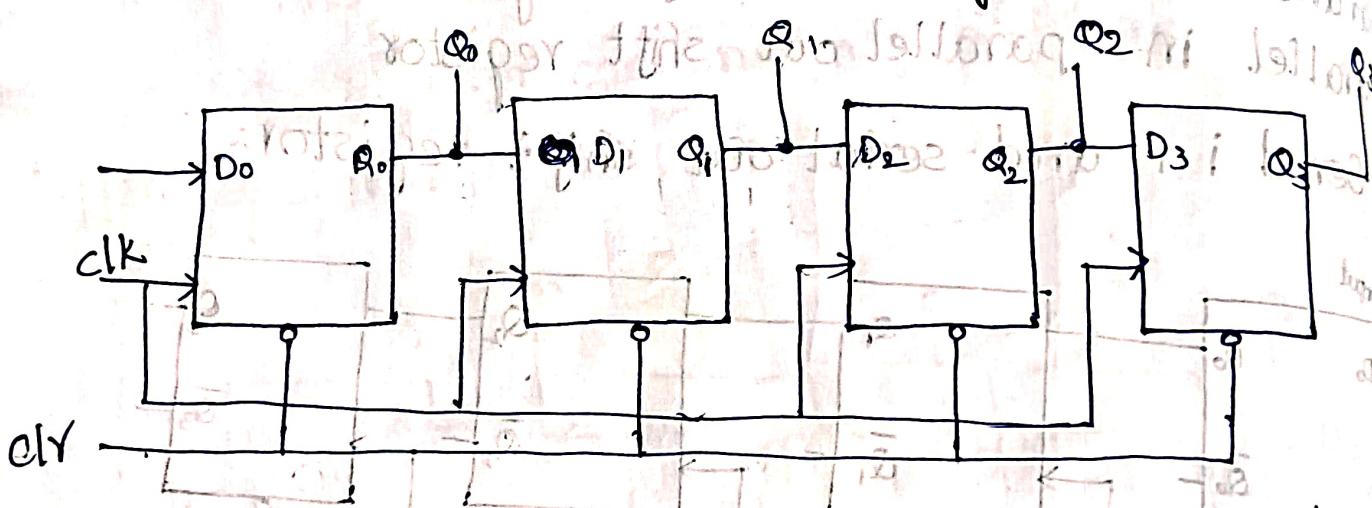
The input data is applied to first flip-flop in series with a clock pulse.

Ex: A 4 bit shift register is shifted right with the serial input being 101011. What is the content of shift register after sixth shift.

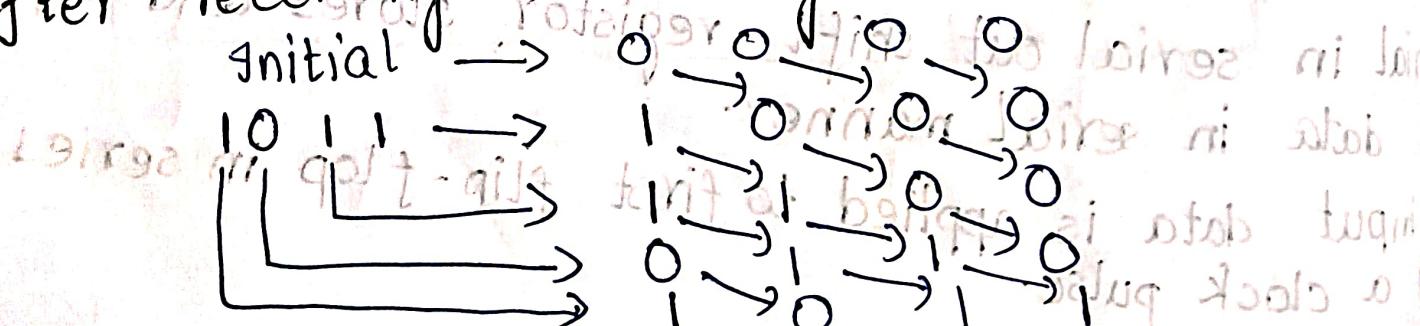


After 6th clock - data shift = 101011

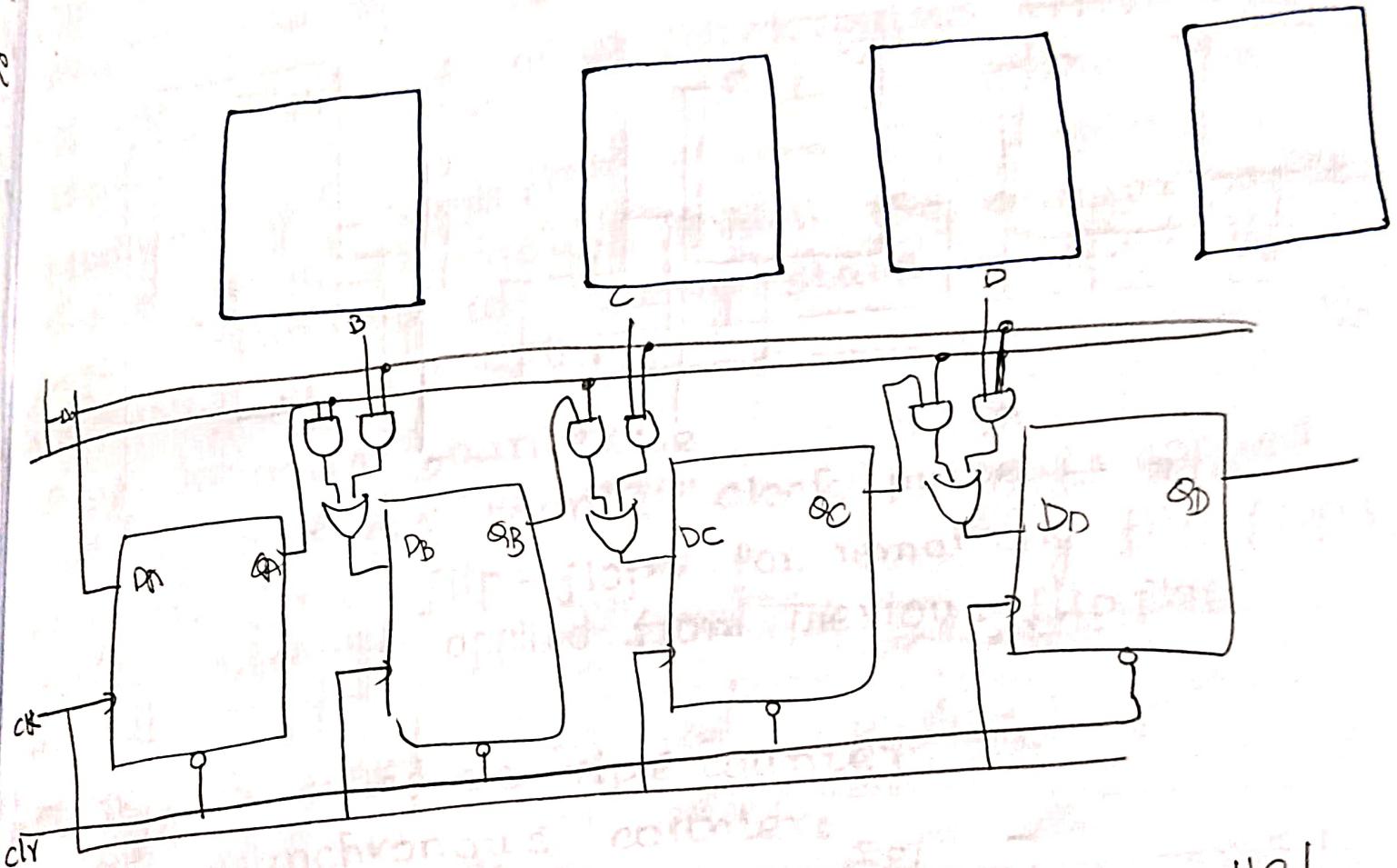
Serial in parallel out shift register:-



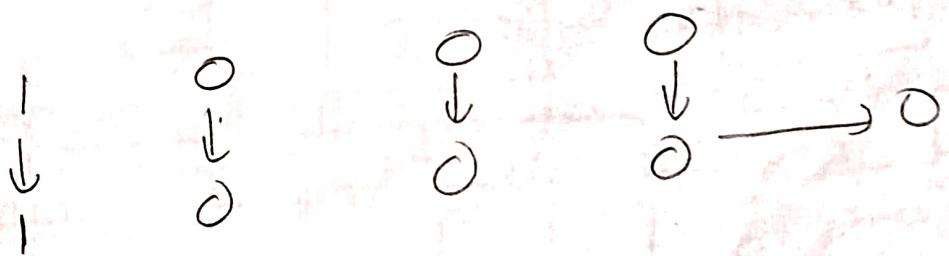
Ex: Construct 4-bit shift register that outputs the binary data pattern 1011 in parallel form after receiving it serially.



parallel in serial out shift register:-

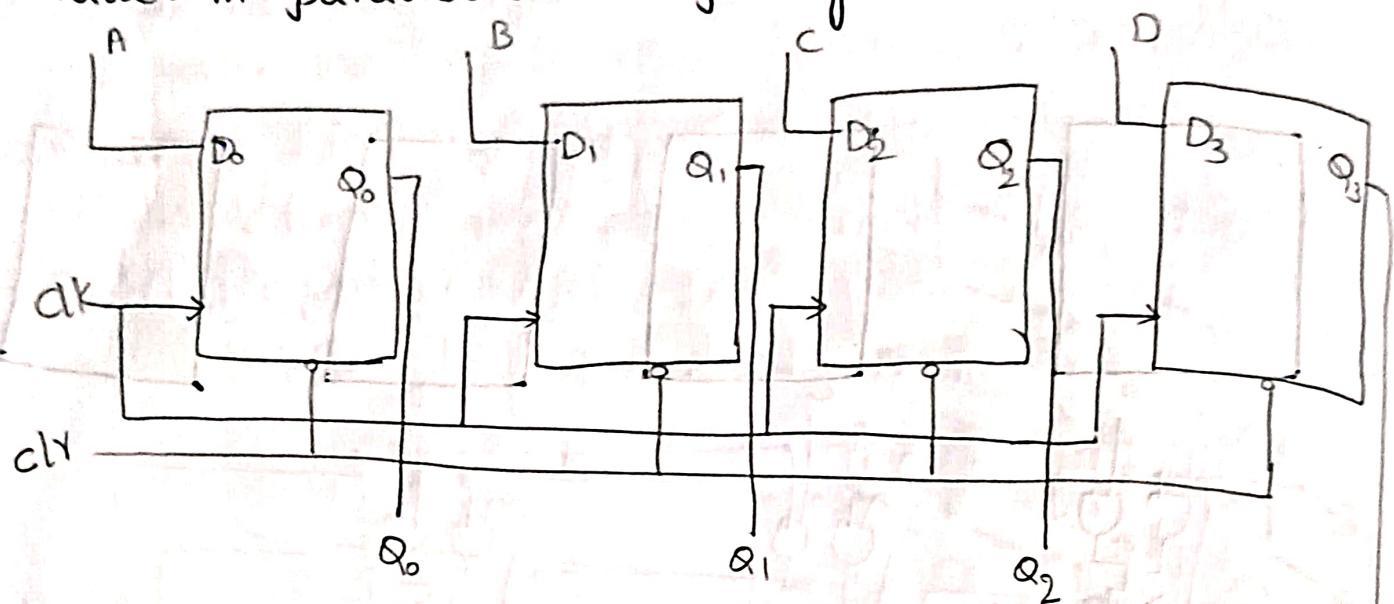


Design 4-bit shift register that converts parallel input into serial output specifically output in binary data as 1000



22/10/25

Parallel in parallel out shift registers:-



counter.

A counter is a set of flip flops whose state changes in response to the clock pulses applied at the input. It can count the no. of clock pulses arriving at its input.

Modules of a counter:-

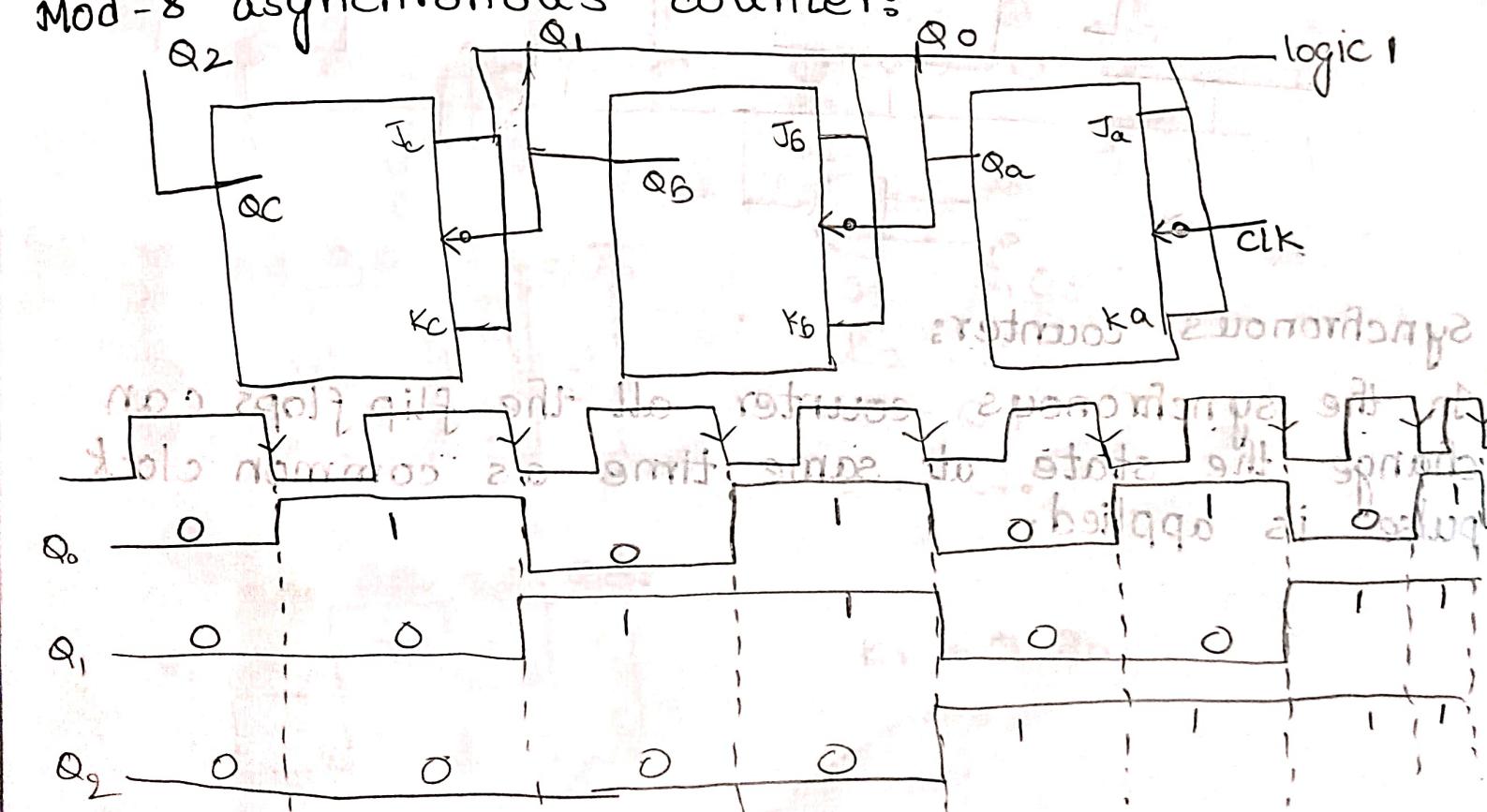
The no. of states through which the counter passes before returning to starting state is called the modulus of a counter.

Asynchronous counter:-

In asynchronous counter clock pulse is applied only at first flip-flops for remaining flip-flops clock pulse is applied from previous flip flop output.

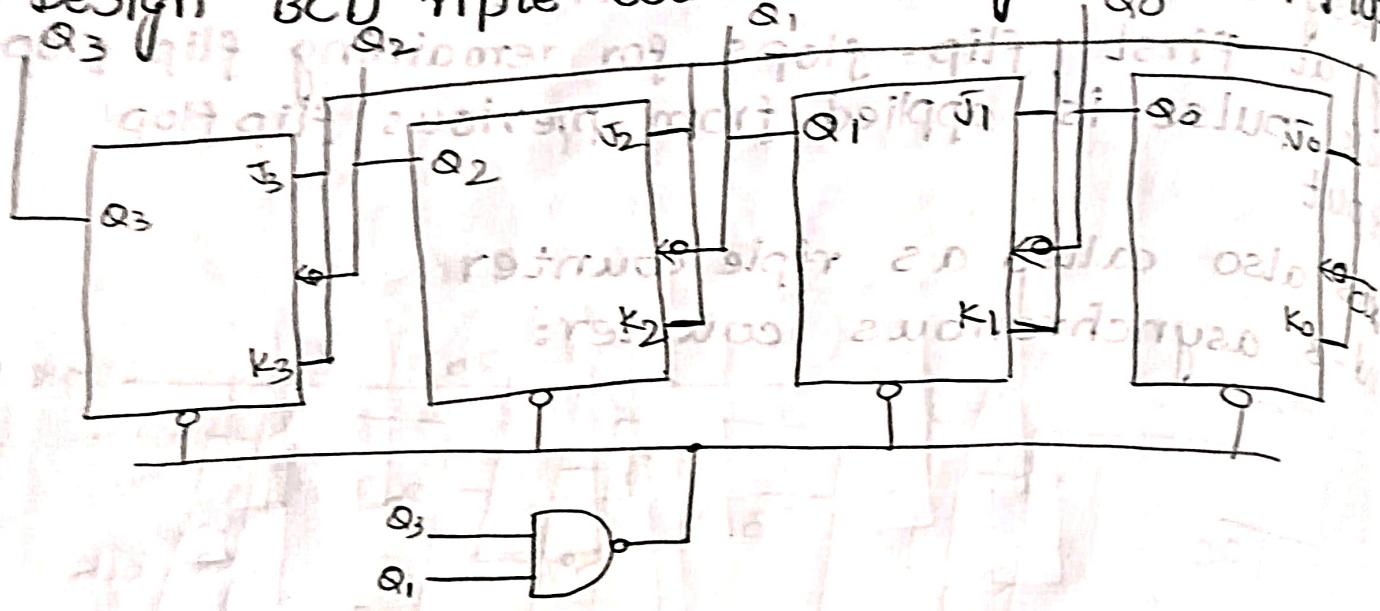
* It is also called as ripple counter

Mod-8 asynchronous counter:



Q_0	Q_1	Q_2
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

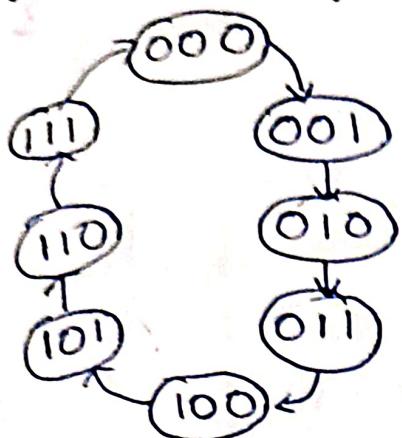
Q. Design BCD triple counter using J-K flip flop.



Synchronous counters:

In the synchronous counter all the flip flops can change the state at same time as common clock pulse is applied.

Mod - 8 synchronous counter using J-K flip-flop



Excitation table

$Q(t)$	$Q(t+1)$	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

Q_2^+	Q_1^+	Q_0^+	Q_2^+	Q_1^+	Q_0^+	J_2, K_2	J, 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	1, x
0	1	0	0	1	0	0, x	x, 1	x, 1
0	1	1	1	0	0	1, x	x, 1	1, x
1	0	0	1	0	1	x, 0	0, x	x, 1
1	0	1	1	0	0	x, 0	1, x	1, x
1	1	0	1	1	0	x, 0	x, 1	x, 1
1	1	1	0	0	0	x, 0	x, 0	x, 1

$Q_2^{(0)}$	$\bar{Q}_1 Q_0$	$\bar{Q}_1 \bar{Q}_0$	$Q_1 Q_0$	$Q_1 \bar{Q}_0$
Q_2	0	1	1	2
Q_2	X ₄	X ₅	X ₇	X ₆

$$J_2 = Q_1 Q_0$$

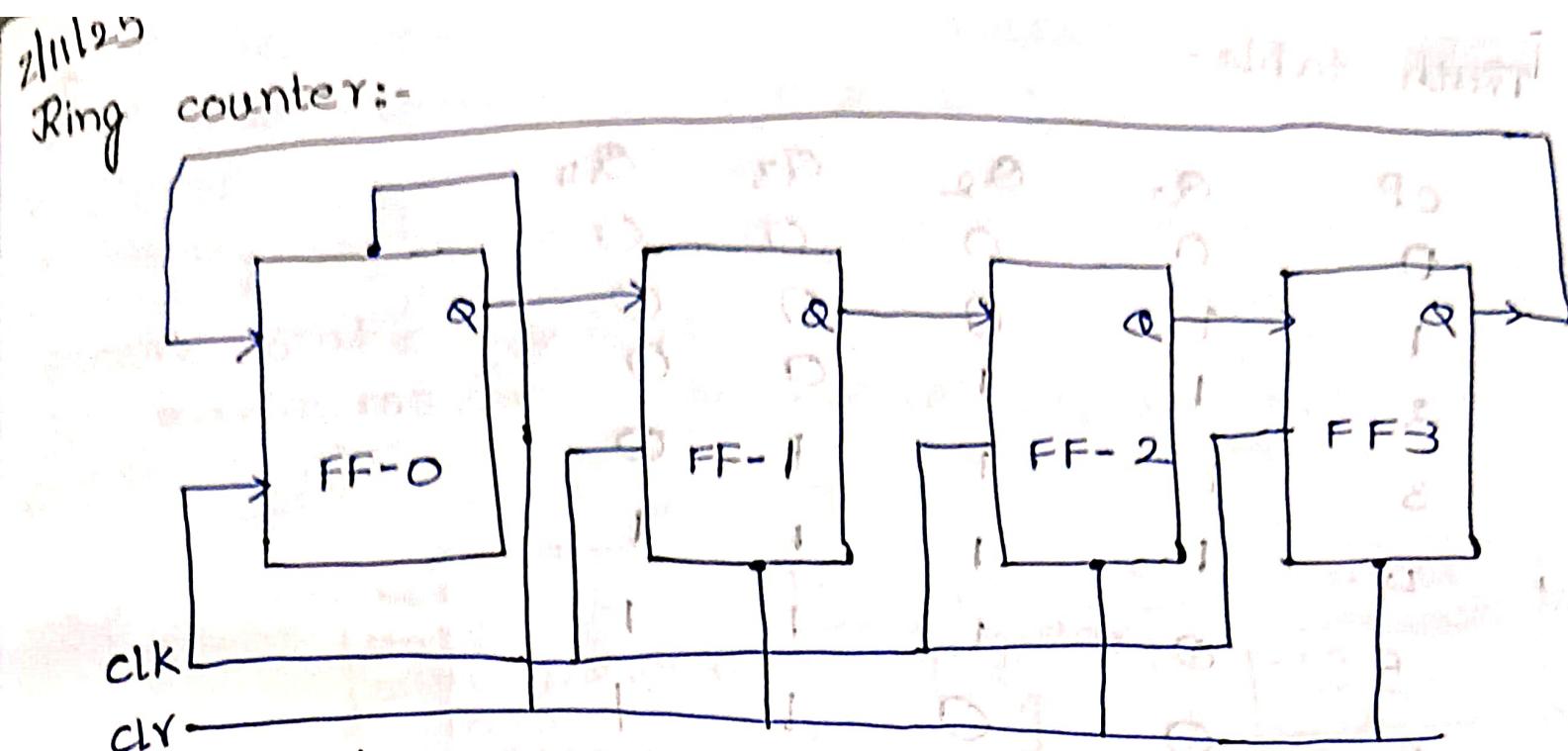
$$K_2 = \bar{Q}_1 \bar{Q}_0$$

$$J_1 = Q_0$$

$$K_1 = \bar{Q}_0$$

$Q_2^{(0)}$	$\bar{Q}_1 Q_0$	$\bar{Q}_1 \bar{Q}_0$	$Q_1 Q_0$	$Q_1 \bar{Q}_0$
Q_2	0	1	1	2
Q_2	X ₀	X ₁	X ₃	X ₂

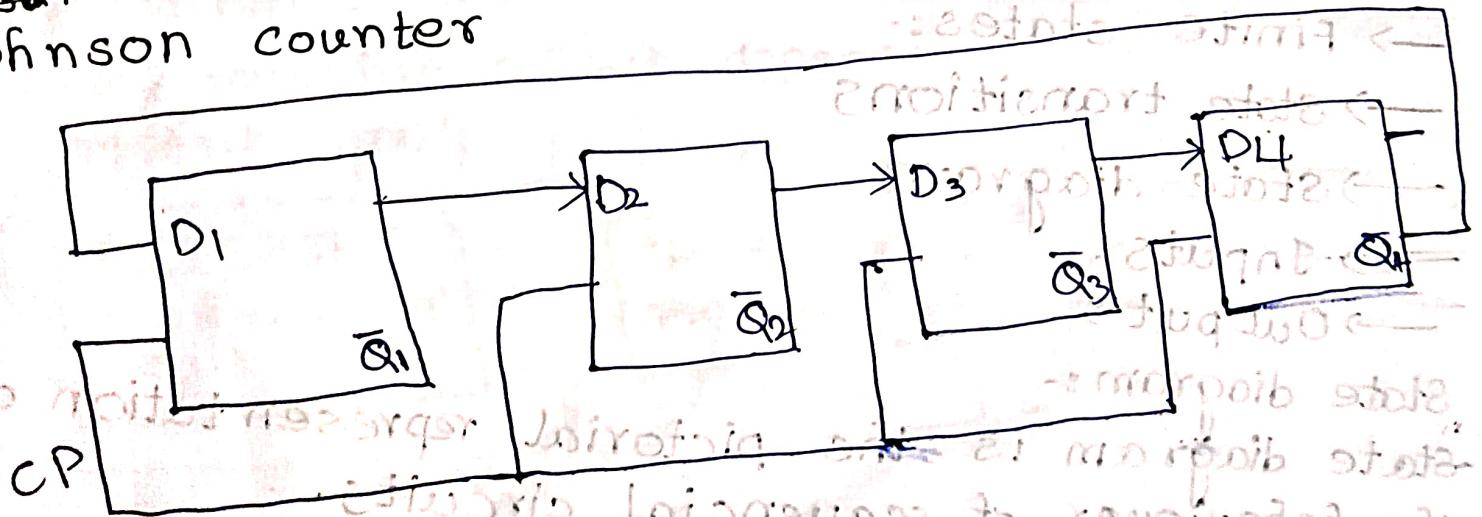
$$K_2 = \bar{Q}_1 \bar{Q}_0$$



Truth table

OR1	clk	Q_0	Q_1	Q_2	Q_3
low	0 X	1	0	0	0
high	high	0	0	0	0
high	high	0	1	0	0
high	high	1	1	0	0
high	high	1	0	0	0

~~Johnson~~ Johnson counter



Truth table:-

CP	Q ₁	Q ₂	Q ₃	Q ₄
0	0	0	0	0
1	1	0	0	0
2	1	1	1	0
3	1	1	1	1
4	1	1	1	1
5	0	0	1	1
6	0	0	0	1
7	0	0	0	0
8	0	0	0	0

Finite state machine (FSM) :-

Finite state machine (FSM) is a mathematical model used to understand the behaviour of digital systems.

Components of FSM :-

- Finite states :-
- State transitions
- State diagram
- Inputs
- Outputs

State diagram :-

State diagram is the pictorial representation of the behaviour of sequential circuits.

types of FSM:-

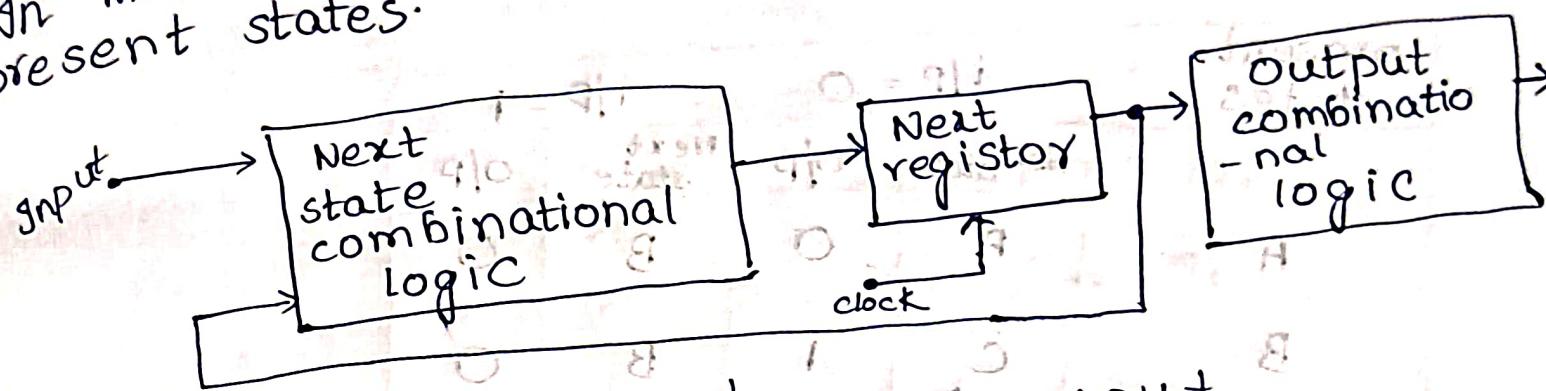
FSM are classified into 2 types

* Moore state machine

* Mealy state machine

Moore state machine:-

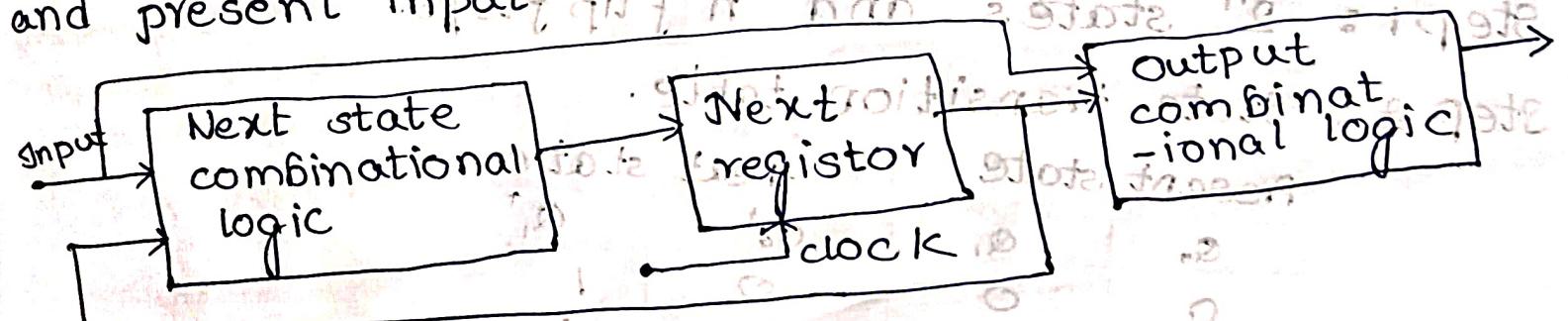
In moore machine the output depends only on present states.



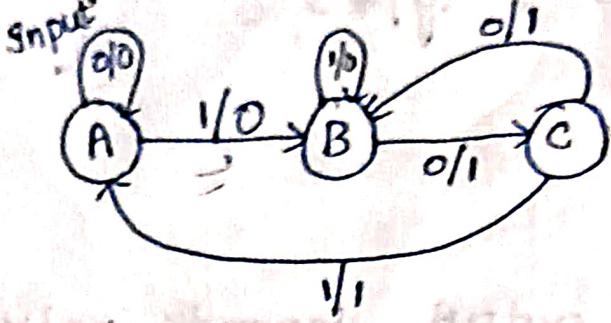
Current state	Next state		Output
	q_0	q_1	
q_0	1	0	1
q_1	0	1	0
q_2	1	1	1

Mealy Machine:-

In mealy machine output depends on present state and present input.



State diagram:



state table:

present states	$i/P = 0$		$i/P = 1$	
	Next state	O/P	Next state	O/P
A	A	0	B	0
B	C	1	B	0
C	B	1	A	1

Sequence generator:-

Sequence generator is a set of digital circuits designed to produce specific bit sequence at the output.

Design a sequence generator using D-flip flops to generate the bit sequence 0, 1, 3, 2.

Step 1: 2^n states and n flip flops.

Step 2: States transition table.

Present state	next state			
	Q_0	Q_1	Q_0^+	Q_1^+
0	0	0	0	1
0	1	1	1	1
1	0	1	0	0
1	0	0	0	0

Step 3: Extend state with excitation table of flip-flops

Present state

Q_1	Q_0
0	0
0	1
1	1
1	0

Next states

$$Q_1^+ \quad Q_0^+$$

Input of flip-flop

$$D_1 \quad D_0$$

$$0 \quad 1$$

$$0 \quad 0$$

$$1$$

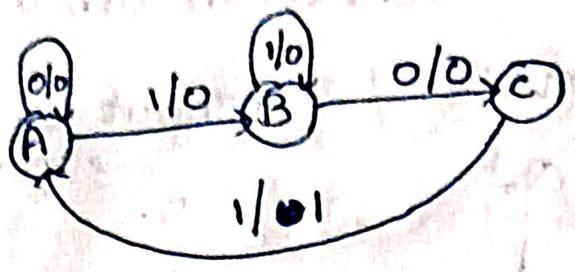
$$1 \quad 1$$

$$1 \quad 0$$

$$0$$

$$0 \quad 0$$

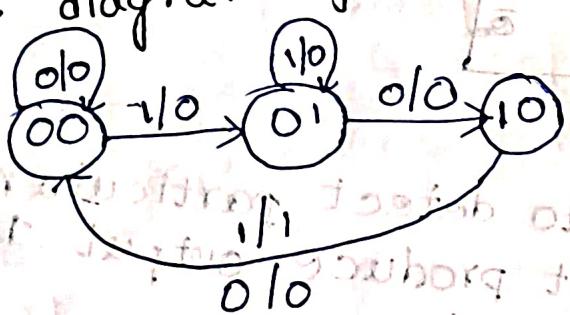
$$0 \quad 0$$



5/11/25
state table:-

Present state	i/p	Next state	output
A	0	A	0
A	1	B	0
B	0	C	0
B	1	B	0
C	0	A	0
C	1	A	1

State diagram after state assignment:-



present state	input	Next state	output	D ₁	D ₀
Q ₁ , Q ₀	x	Q ₁ ⁺ , Q ₀ ⁺	z		
0, 0	0	0, 0	0	0	0
0, 0	1	0, 1	0	0	1
0, 1	0	1, 0	0	1	0
0, 1	1	0, 1	0	0	1
1, 0	0	0, 0	1	0	0
1, 0	1	0, 0	x	x	x
1, 1	0	x, x	x	x	x
1, 1	1	x, x	x	x	x

D_1

	$Q_1 \bar{x}$	$\bar{Q}_1 x$	$Q_0 \bar{x}$	$\bar{Q}_0 x$
D_1	0	1	3	1
\bar{D}_1	4	5	X	X
Q_1				

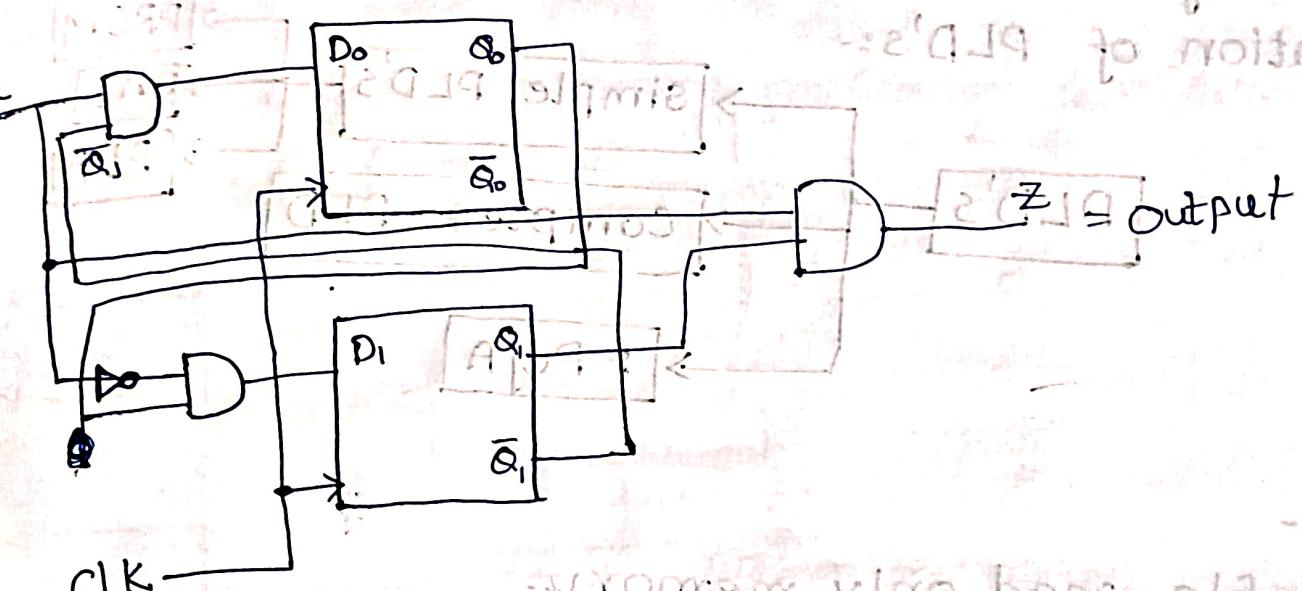
$$D_1 = Q_0 \bar{x}$$

D_0

	$Q_1 \bar{x}$	$\bar{Q}_1 x$	$Q_0 \bar{x}$	$\bar{Q}_0 x$
D_0	0	1	1	2
\bar{D}_0	4	5	X	X
Q_1				

$$D_0 = \bar{Q}_1 x$$

logic diagram:-



-:MORP

behit and font ssivsb oipol eldomorropq n si MORP
yorro R0 . eldomorropq bho gondina

