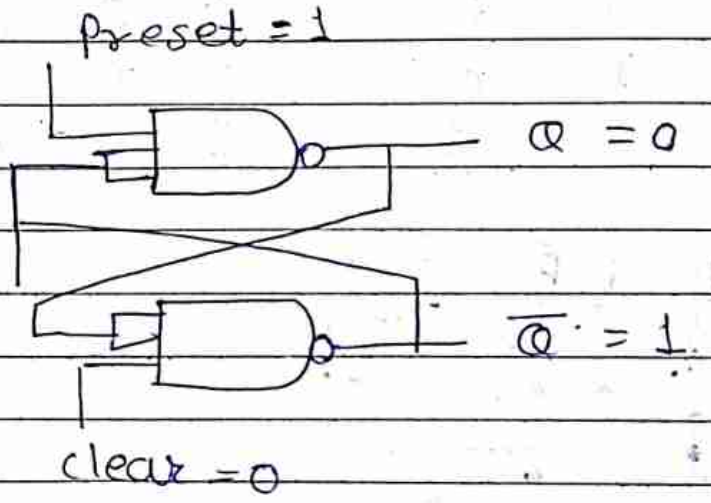
Unit-4

* Sequential ckt.

→ Types of Flip-Flops (FF)

1) SR FF (set Reset flip-flops)
2) JK FF
3) D FF (Delayed flip-flops)
4) T FF (Toggle flip-flops)

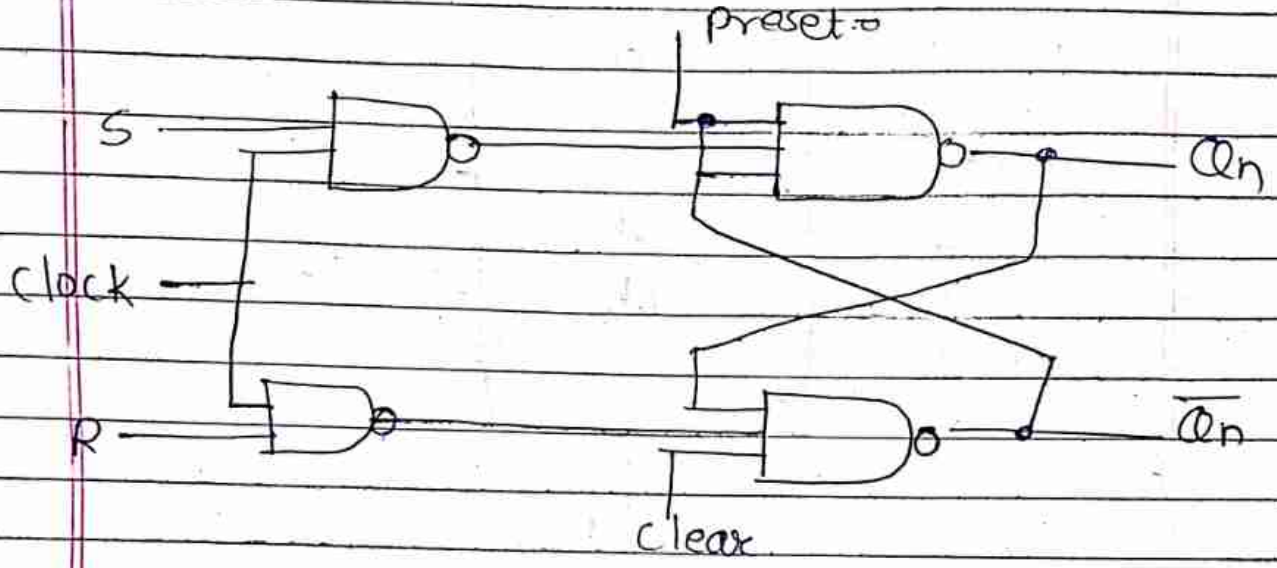
* Latch



NAND Gate truth table

Inputs		output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

* Logic diagram of SR FF

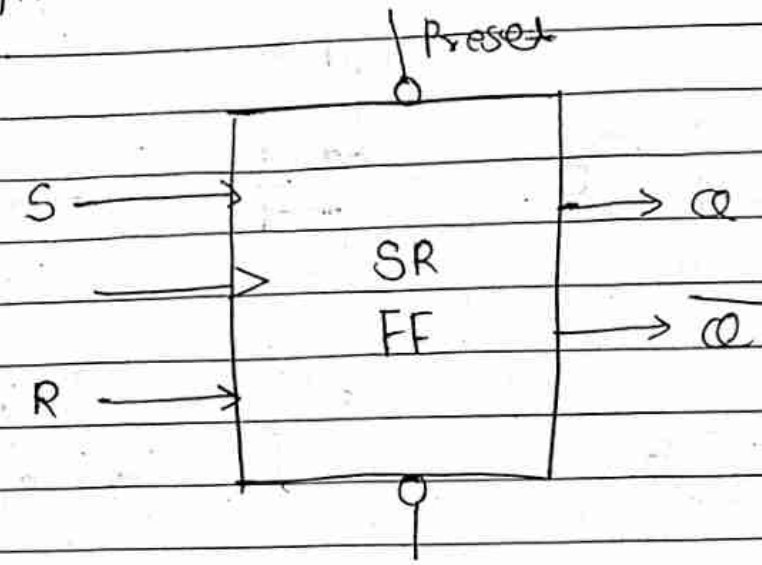


S	R	Qn	Qn+1	
0	0	0	0	Qn
0	0	1	1	
0	1	0	0	0
0	1	1	0	
1	0	0	1	1
1	0	1	1	
1	1	0	—	forbidden state.
1	1	1	—	

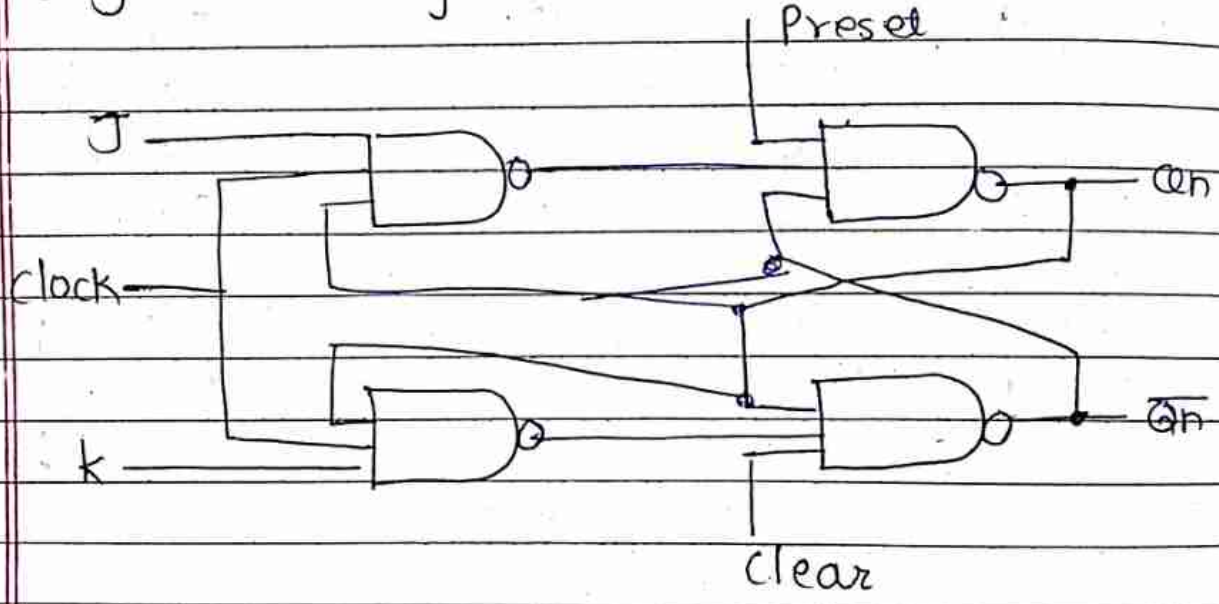
* Truth Table of SR FF.

Inputs		outputs
S	R	Qn+1
0	0	Qn
0	1	0
1	0	1
1	1	?

* Symbol for SR FF



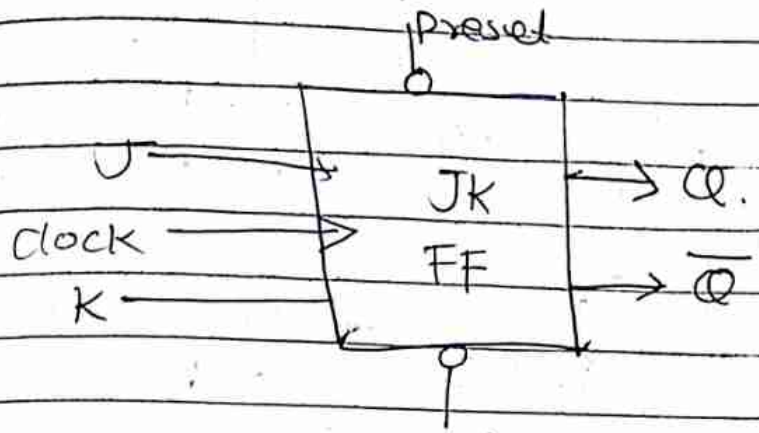
* Logic diagram of JK ff.



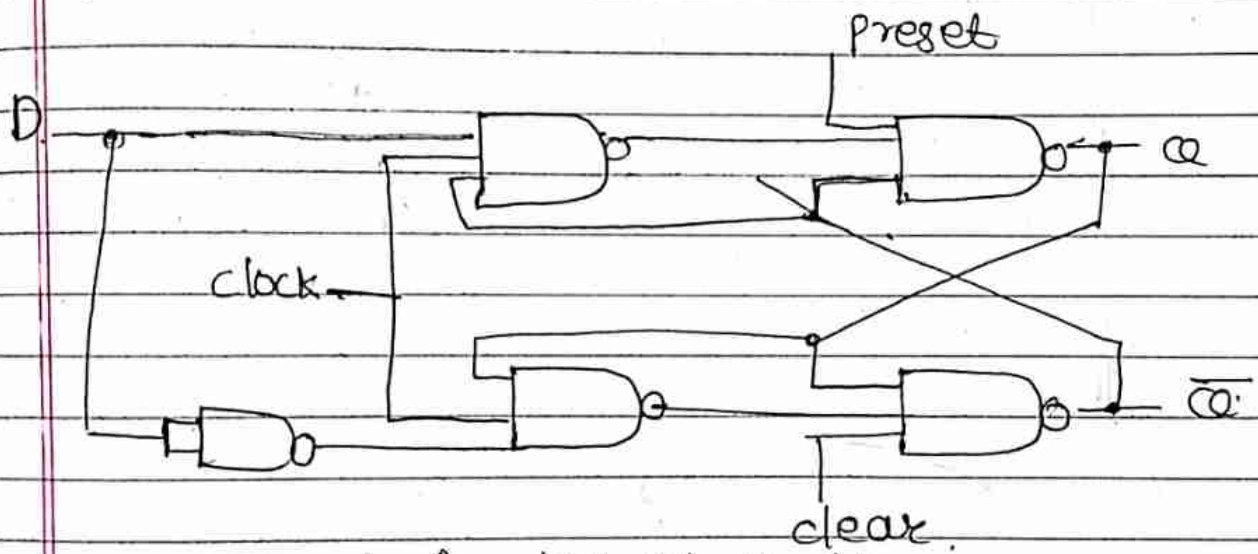
- Truth table of JK ff.

Inputs		outputs
J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$

* Symbol for JK FF



* D-FF

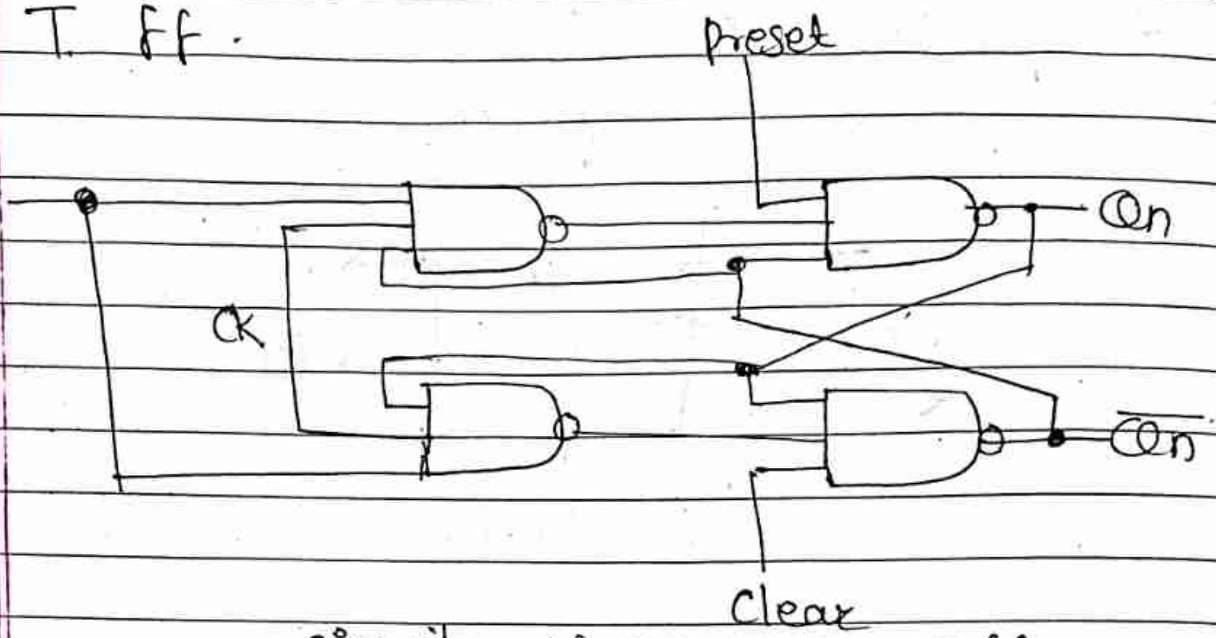


logic diag. of D-FF

Truth Table :

Input	output
D	Q_{n+1}
0	0
1	1

* T ff.



Circuit diagram for T-ff

Truth Table for T-ff

T	Qn+1
0	Qn
1	$\overline{Qn}$

* Excitation Table : Of Flip-flops

In excitation table we know present output state and next output state from that we have to find possible combination of input required.

present state	Next state	SR ff		JK ff		T ff	D ff
		S	R	J	K	T	D
0	0	0	X	0	X	0	0
0	1	1	0	1	X	1	1
1	0	0	1	X	1	1	0
1	1	X	0	X	0	0	1

* Conversion of flip flop

Que: convert D type flip flop into T flip flop.

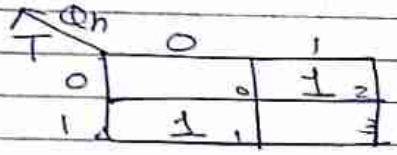
Sol ⇒

Q_n	Q_{n+1}	SR FF		JK FF		T FF	D FF
		S	R	J	K	T	D
0	0	0	X	0	X	0	0
0	1	1	0	1	X	1	1
1	0	0	1	X	1	1	0
1	1	X	0	X	0	0	1

Excitation Table for the conversion

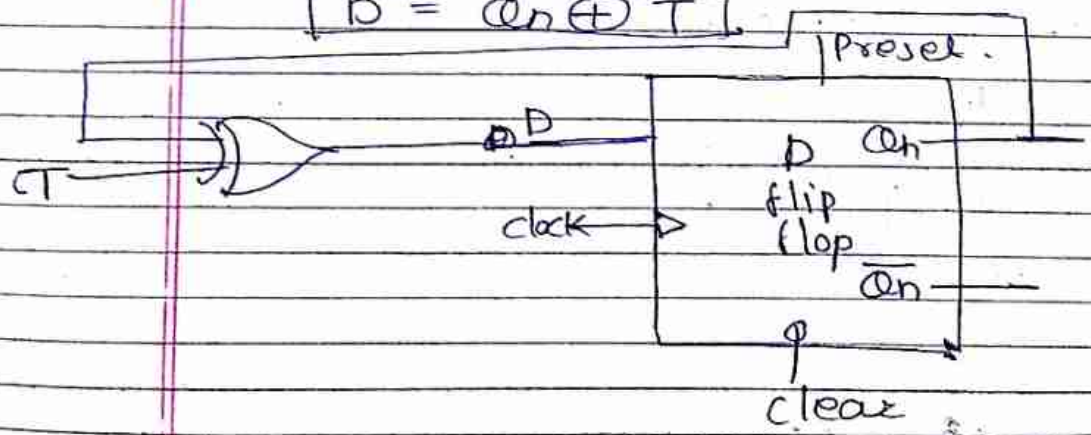
	Present state Q_n	Next state Q_{n+1} (T)	(D)
0	0	0	0
1	0	1	1
3	1	1	0
2	1	0	1

$D = \Sigma m(1,2)$



$D = \overline{Q_n} T + Q_n \overline{T}$

$D = Q_n \oplus T$



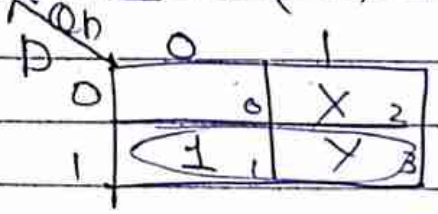
Ckt dia of T-FF using D-FF.

que: Convert JK flip flop to D flip flop.

→ Excitation Table for conversion

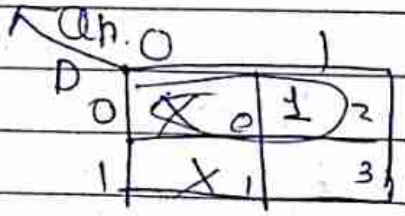
Present state	Next state	Inputs	
Q_n	D	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

$J = \Sigma m(1) + d(2,3)$

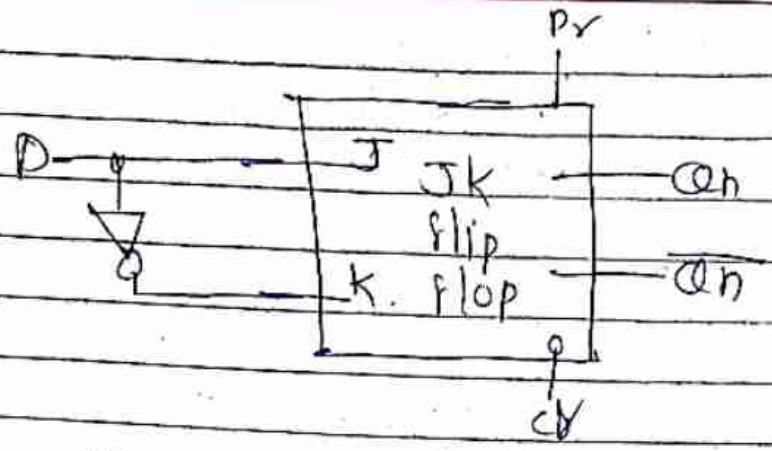


$J = D$

$K = \Sigma m(2) + d(0,1)$



$\boxed{K = \overline{D}}$



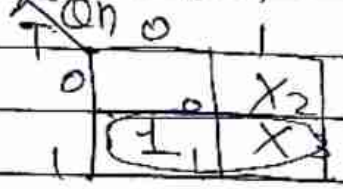
Ckt diag of D-ff using Jk ff.

Que: convert Jk flip flop to T flip flop

→ Excitation Table for conversion

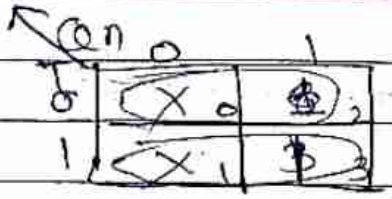
	present state Qn	Next state T	Inputs J	K
0	0	0	0	X
1	0	1	1	X
3	1	1	X	0
2	1	0	X	1

$J = \sum m(1) + d(2,3)$

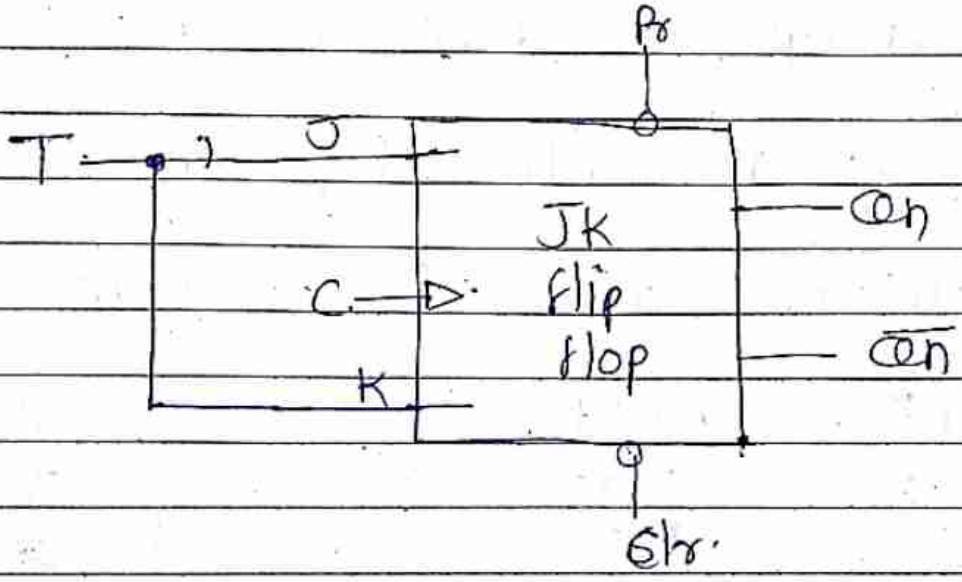


$J = T$

$K = \sum m(3) + d(0,1)$



$K = T$



Ckt diagram for JKff to T flip flop.

Que: Convert SR ff to JK ff

→ Excitation Table for conversion.

	Present state	Next state		Inputs	
	Q_n	J	K	S	R
0, 1	0	0	X	0	X
2, 3	0	1	X	1	0
4, 6	1	X	0	0	1
5, 7	1	X	0	X	0

	Present state	Next state		Inputs	
	Q_n	J	K	S	R
0	0	0	0	0	X
1	0	0	1	0	X
2	0	1	0	1	0
3	0	1	1	1	0
5	1	0	1	0	1
7	1	1	1	0	1
4	1	0	0	X	0
6	1	1	0	X	0

$S = \Sigma m(2,3) + d(4,6)$

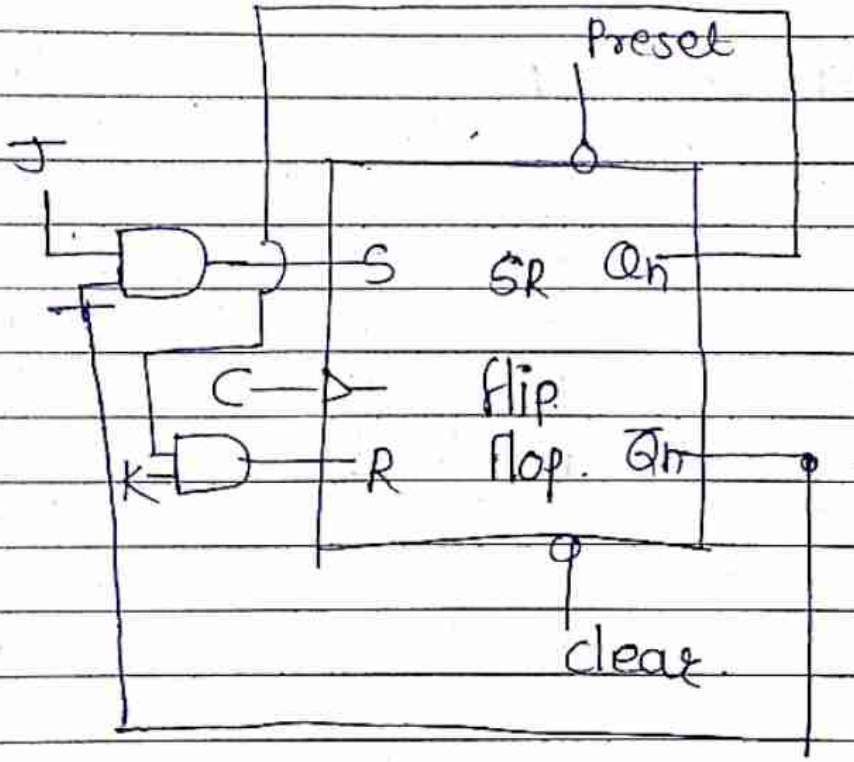
$R = \Sigma m(5,7) + d(0,1)$

$Q_n J$ / K	00	01	11	10
0	0	1 (2)	X 6	X 4
1	1	1 (3)	7	5

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

$Q_n J$ / K	00	01	11	10
0	X 0	2	6	4
1	X 1	3	1 7	1 5

$$R = Q_n K$$



ckt diagram for JK flip flop using SR flip flop.

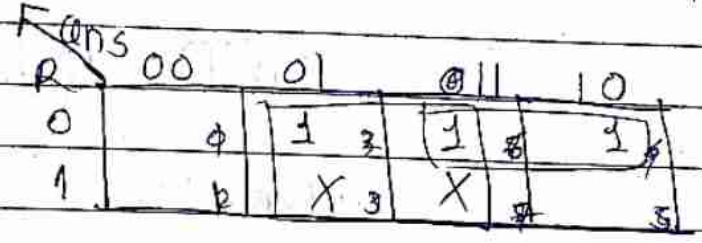
Que 1: Convert JK ff to D ff

Que 2: convert D ff to SR ff.

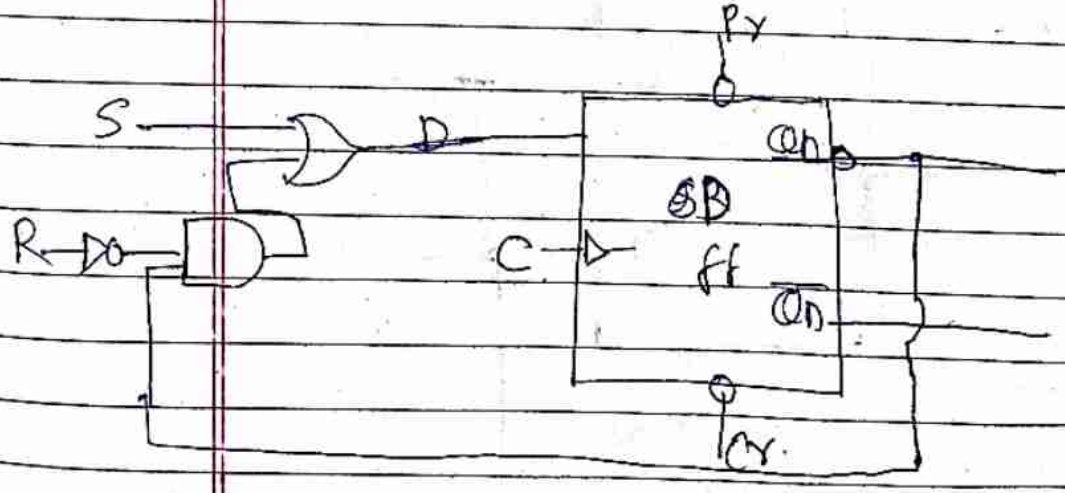
→ Exciation Table for conversion:

	present state	Next state		Input
	Q_n	S	R	D
0, 1	0	0	X	0
2,	0	1	0	1
5,	1	0	1	0
4, 6	1	X	0	1

$$D = \Sigma m(2,4,6) + d(3,7)$$



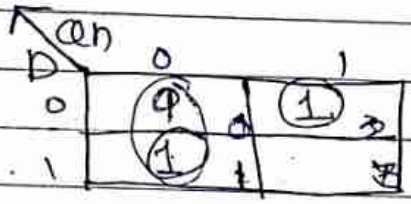
$$D = S + Q_n\overline{R}$$



→ Excitation Table for conversion.

	Present state Q_n	Next state D	Inputs T
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

~~D = Σm(~~ $T = \Sigma m(1, 2)$



$T = \overline{Q_n} D + Q_n \overline{D}$

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

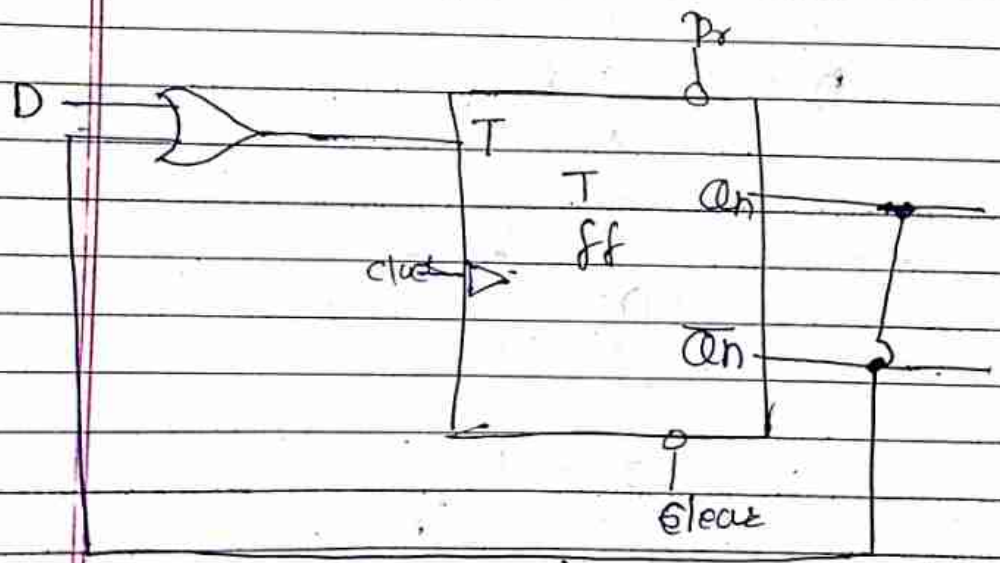


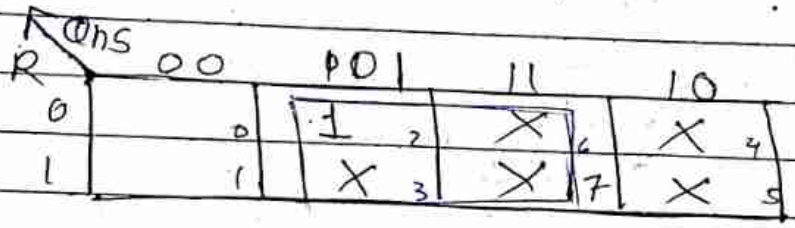
figure ckt diagram of Dff using Tff

Que° Convert JK-ff to SR-ff.

→ Exciation Table for conversion.

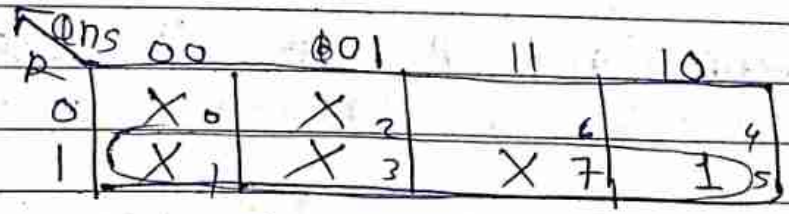
	Present state Qn	Next state		Inputs	
		S	R	J	K.
0, 1	0	0	X	0	X
2,	0	1	0	1	X
5,	1	0	1	X	1
4, 6,	1	X	0	X	0

$J = \Sigma m(2) + d(3, 4, 5, 6, 7)$

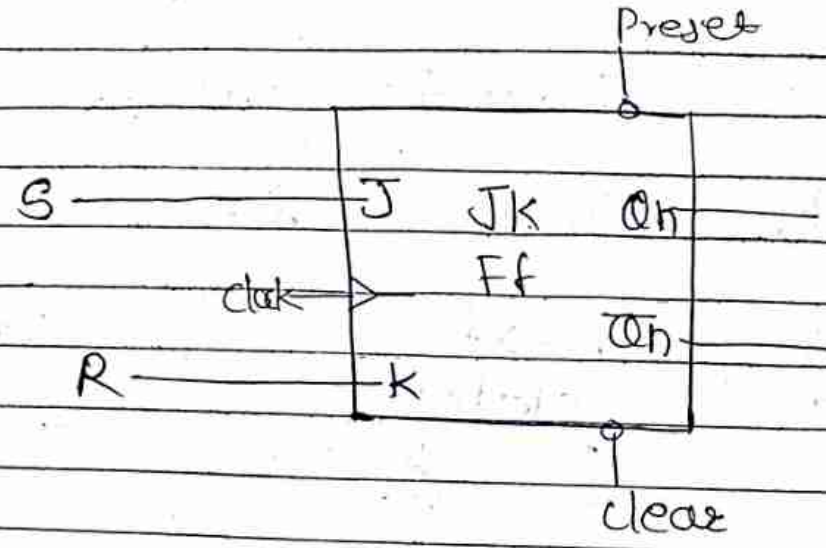


$J = S$

$K = \Sigma m(5) + d(0, 1, 2, 3, 7)$



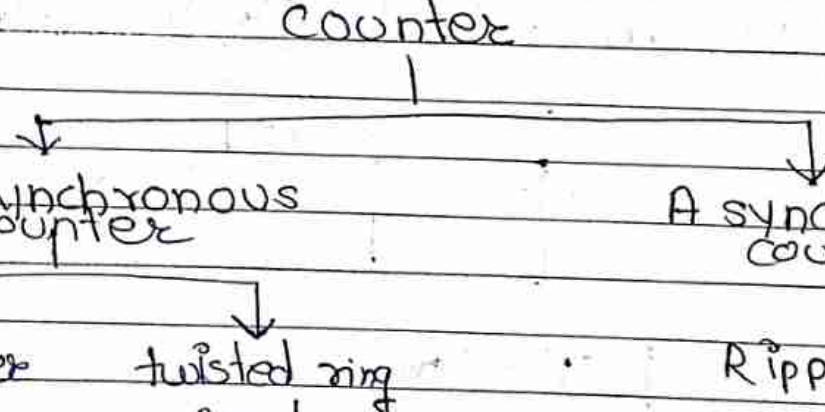
$K = R$



ckt diagram for SR-ff using J-ff

Counters

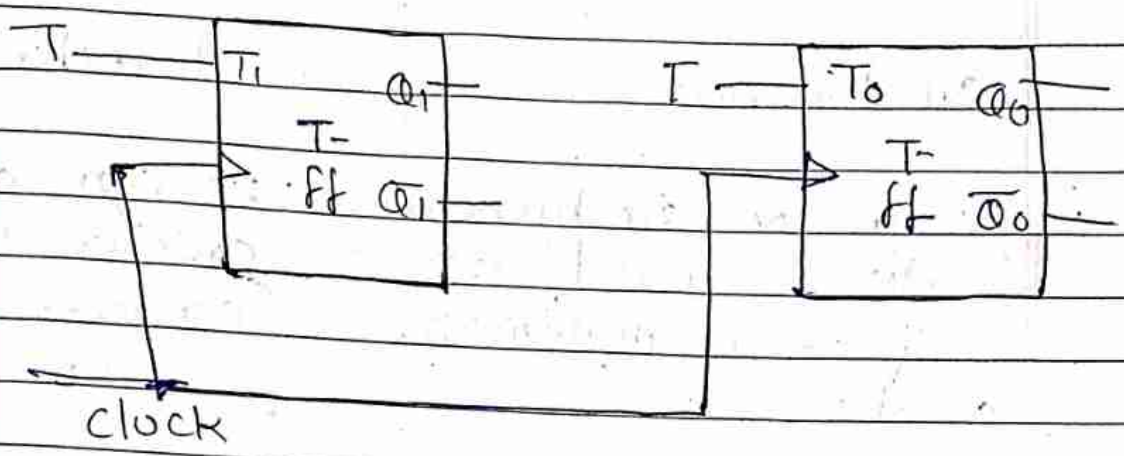
→ Types



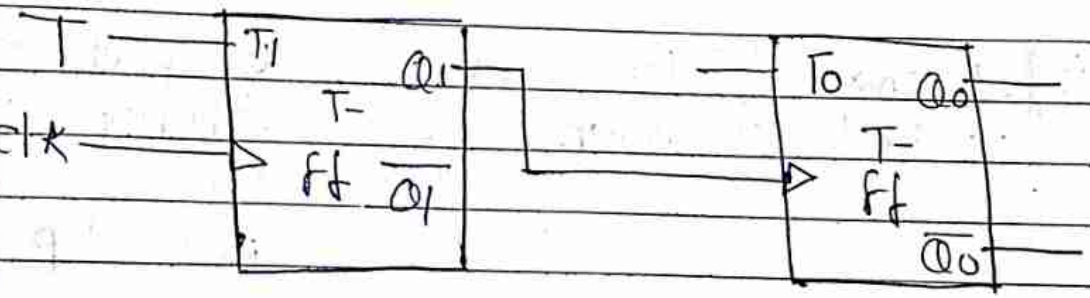
Defination : A circuit use for counting pulses is known as counter

Modulo (Mod) indicates number of states in the counter, example 3 bit counter is also called as mod 8 counter

* Synchronous:



* Asynchronous



W-19
S-19

* Differentiate Between synchronous and Asynchronous counter

	Synchronous	Asynchronous
i]	Design of synchronous counter is 'hard' as compare to Asynchrous counter	Design of Asynchrous counter is easy as compare to Synchrous counter
ii]	Speed of operation is fast as compared to Asynchronous	Speed of operation is low as compared to synchronous counter
iii]	All the flip-flops are clocked simultensouly	Output of one flip-flop is given as clock to the next flip-flop
iv]	Counter can be designed for straight binary sequence or random sequence	Counter can be designed for only straight binary sequence (Assending or decending)
v]		

* Synchronous counter Designing steps:

1] Decide how many number of flip-flop required

2] Draw the state diagram for the counter flow

3] Draw the excitation table for the counter designing

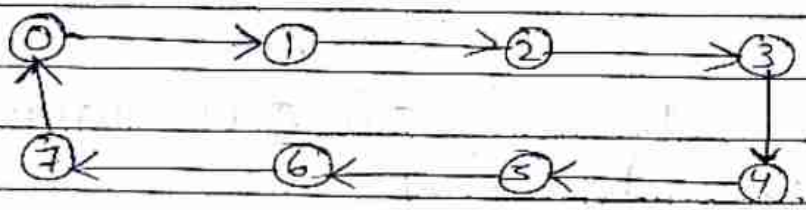
4] Find the Equation form the excitation table and solve using k-map.

5] Design the circuit using above generated equation.

Que:- Design 3 bit Synchronous Counter using JK flip-flop

⟹ 3 no. of JK-ff required for design.

State diagram



Excitation

	Present State			Next State			Inputs					
	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_0	K_0	J_1	K_1	J_2	K_2
0	0	0	0	0	0	1	1	X	0	X	0	X
1	0	0	1	0	1	0	X	1	1	X	0	X
2	0	1	0	0	1	1	1	X	X	0	0	X
3	0	1	1	1	0	0	X	1	X	1	1	X
4	1	0	0	1	0	1	1	X	0	X	X	0
5	1	0	1	1	1	0	X	1	1	X	X	0
6	1	1	0	1	1	1	1	X	X	0	X	0
7	1	1	1	0	0	0	X	1	X	1	X	1

$J_0 = \Sigma m(0,2,4,6) + d(1,3,5,7)$

$K_0 = \Sigma m(1,3,5,7) + d(0,2,4,6)$

$J_1 = \Sigma m(1,5) + d(2,3,6,7)$

$K_1 = \Sigma m(3,7) + d(0,1,4,5)$

$J_2 = \Sigma m(3) + d(4,5,6,7)$

$K_2 = \Sigma m(7) + d(0,1,2,3)$

Q_2Q_1 / Q_0	00	01	11	10
0	1 (0)	1 (2)	1 (6)	1 (4)
1	X (1)	X (3)	X (7)	X (5)

$J_0 = 1$

Q_2Q_1 / Q_0	00	01	11	10
0	X (0)	X (2)	X (6)	X (4)
1	1 (1)	1 (3)	1 (7)	1 (5)

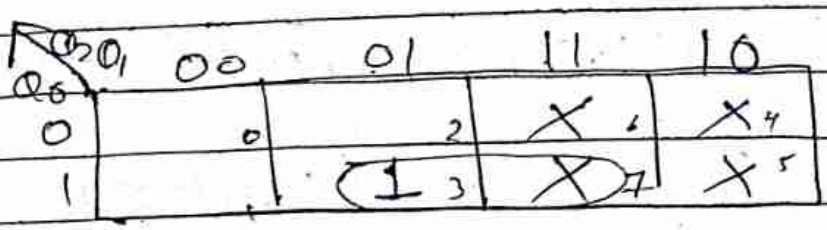
$K_0 = 1$

Q_2Q_1 / Q_0	00	01	11	10
0	(0)	X (2)	X (6)	(4)
1	1 (1)	X (3)	X (7)	1 (5)

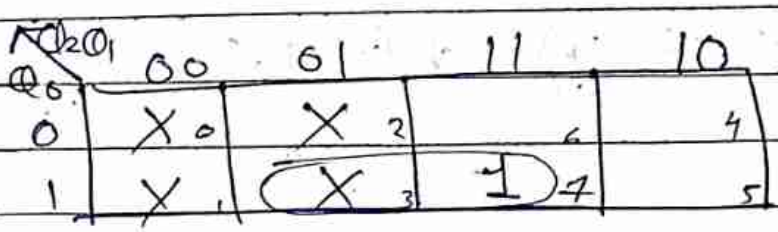
$J_1 = Q_0$

Q_2Q_1 / Q_0	00	01	11	10
0	X (0)	(2)	(6)	X (4)
1	X (1)	1 (3)	1 (7)	X (5)

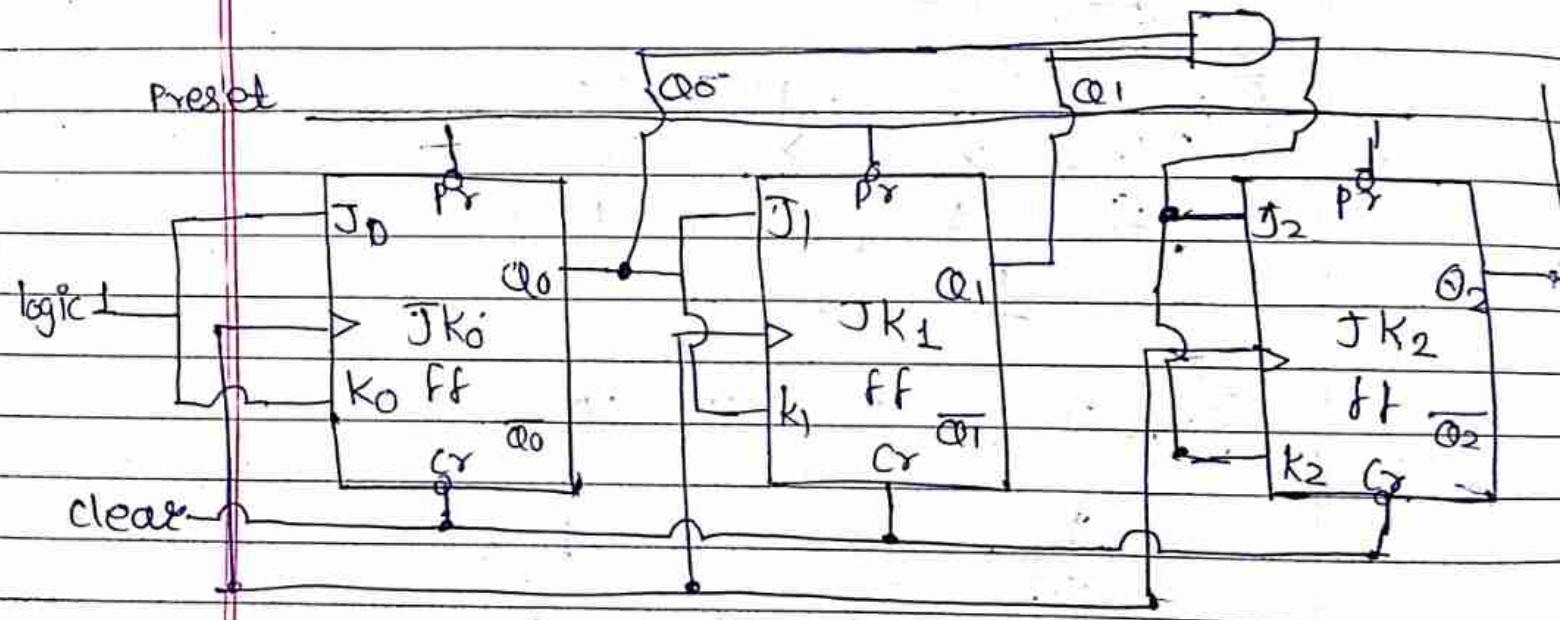
$K_1 = Q_0$



$J_2 = Q_1 Q_0$



$K_2 = Q_1 Q_0$



3-bit synchronous counter using JK-ff

Que :- Design a synchronous counter for the following sequence

4 → 6 → 7 → 3 → 1

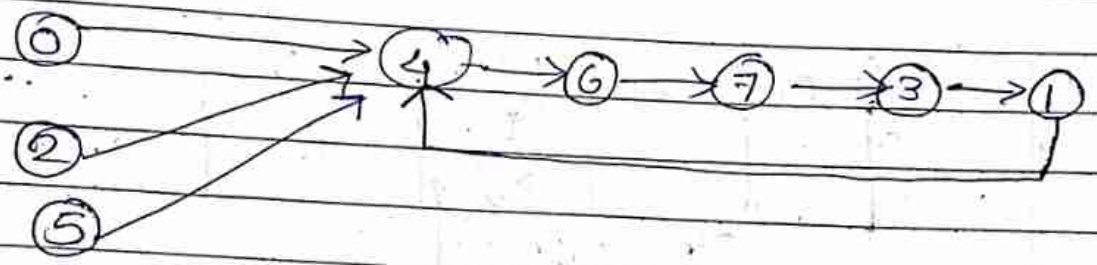
4 marks
or 18

avoid lock out condition. use JK-ff for design

→

To design the above counter we required 3 JK-ff

State diagram



→ Excitation Table :

	Present state			Next state			Inputs :		
	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_0K_0	J_1K_1	J_2K_2
4	1	0	0	1	1	0	0 X	1 X	X 0
6	1	1	0	1	1	1	1 X	X 0	X 0
7	1	1	1	0	1	1	X 0	X 0	X 1
3	0	1	1	0	0	1	X 0	X 1	0 X
1	0	0	1	1	0	0	X 0	0 X	1 X
0	0	0	0	1	0	0	0 X	0 X	1 X
2	0	1	0	1	0	0	0 X	X 1	1 X
5.	0	1	0	1	0	0.	X 1	0 X	X 0

$J_0 = \Sigma m(6) + d(1,3,5,7)$

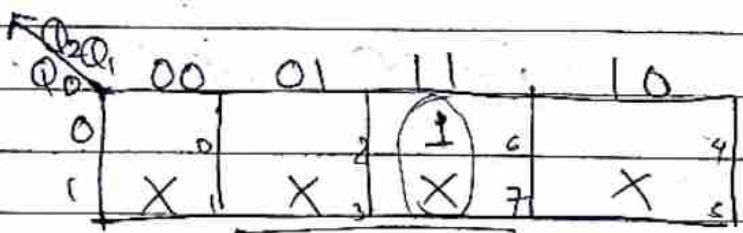
$K_0 = \Sigma m(1,5) + d(0,2,4,6)$

$J_1 = \Sigma m(4) + d(3,6,7,2)$

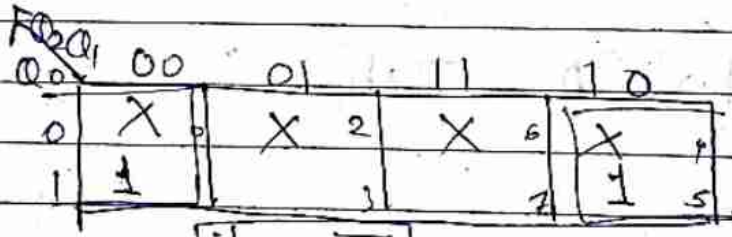
$K_1 = \Sigma m(2,3) + d(4,1,05)$

$J_2 = \Sigma m(0,1,2) + d(4,6,7,5)$

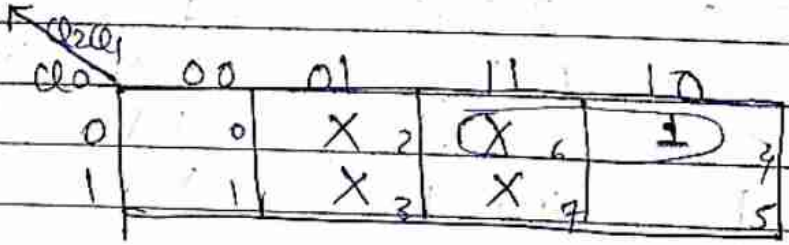
$K_2 = \Sigma m(7) + d(0,1,2,3)$



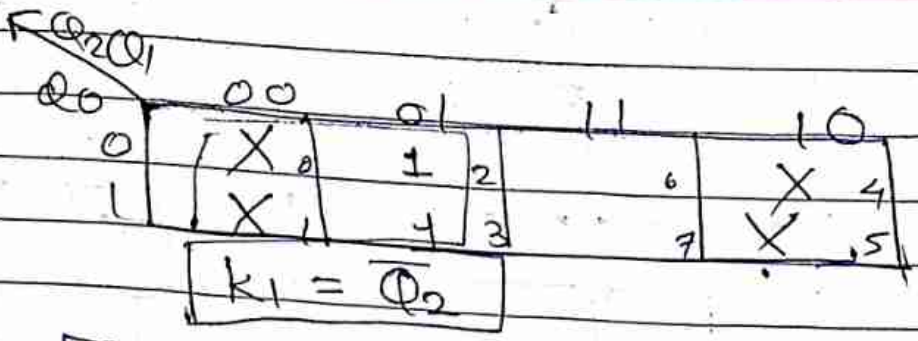
$J_0 = Q_2 Q_1$



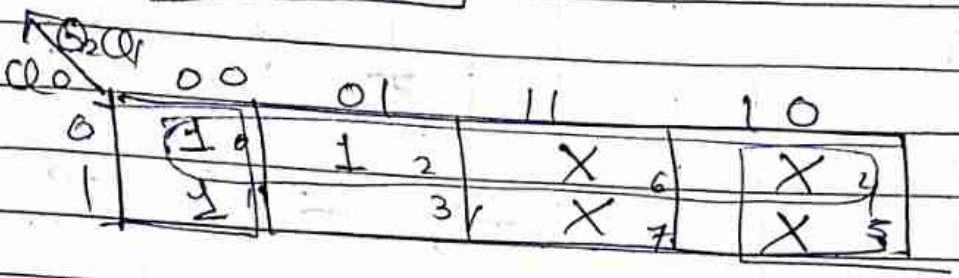
$K_0 = \overline{Q_1}$



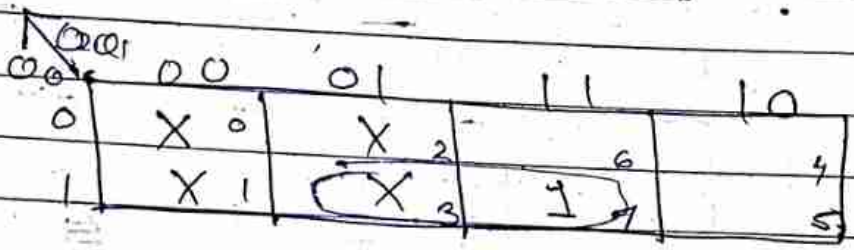
$J_1 = Q_2 \overline{Q_0}$



$$K_1 = \overline{Q_2}$$

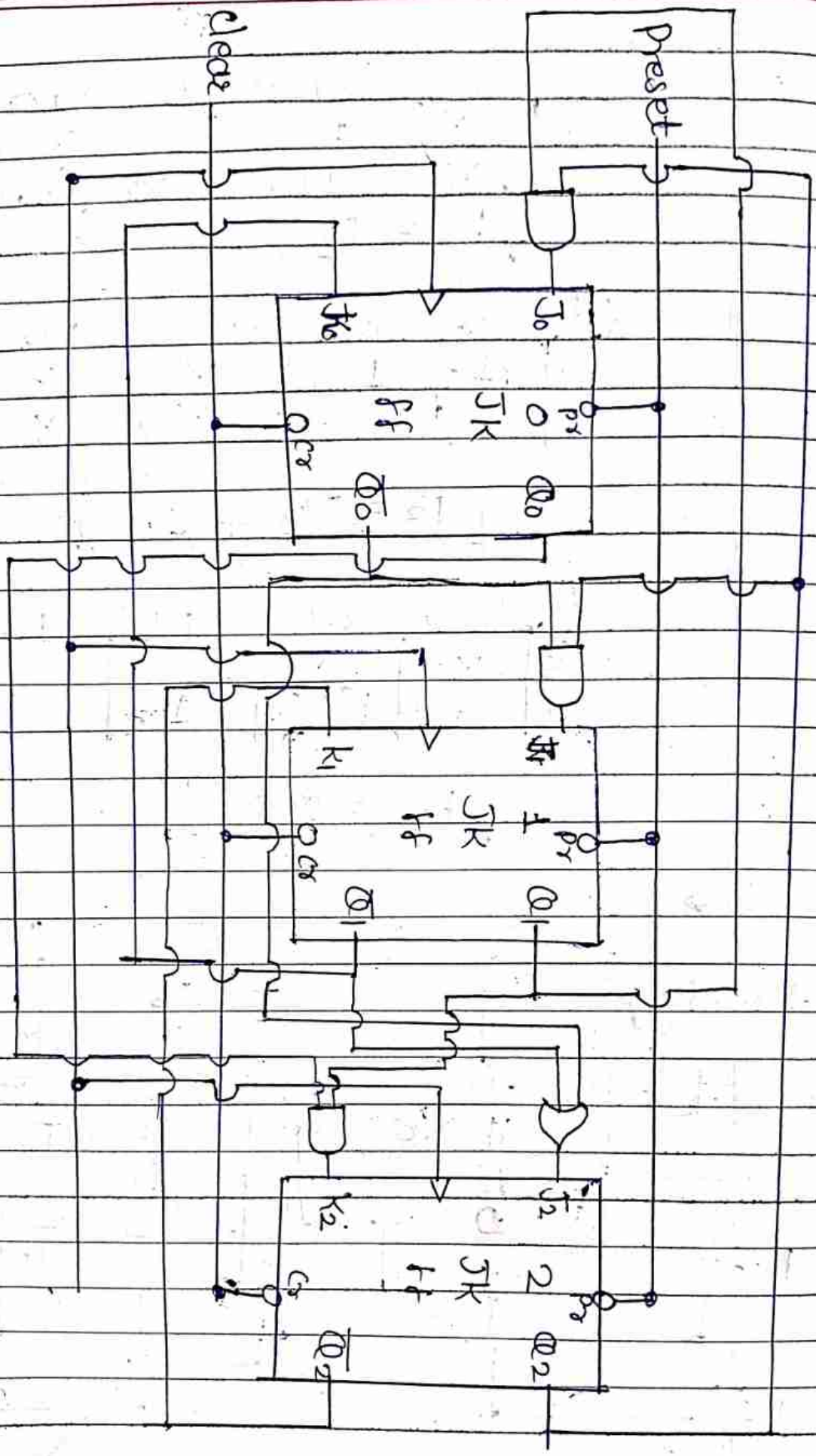


$$J_2 = \overline{Q_0} + \overline{Q_1}$$



$$K_2 = Q_1Q_0$$

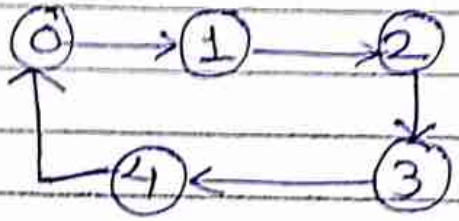
Design of lock out condition using JK-FF



Ques: Design mode-5 synchronous counter using T-ff.

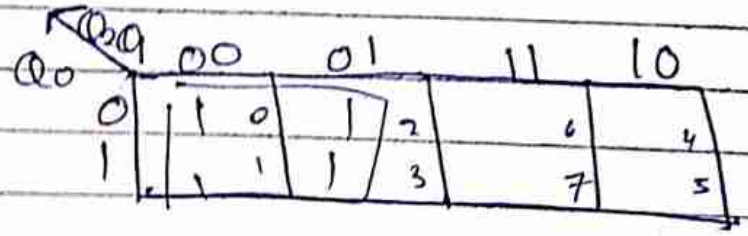
→ 3 JK-ff required for the counter design

State diagram:

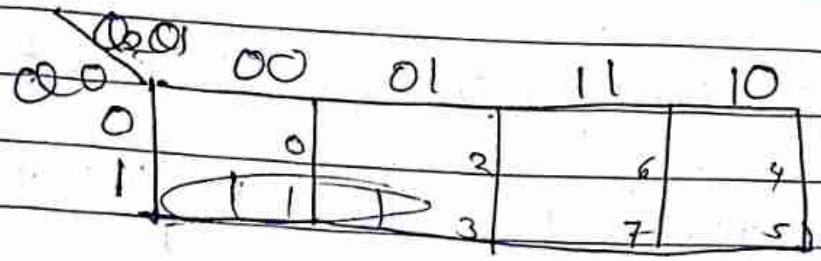


Excitation Table:

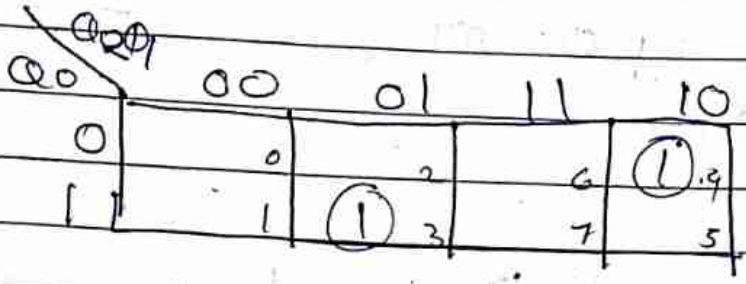
	Present state			Next state			Inputs		
	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	T_0	T_1	T_2
0	0	0	0	0	0	1	1	0	0
1	0	0	1	0	1	0	1	1	0
2	0	1	0	0	1	1	1	0	0
3	0	1	1	1	0	0	1	1	1
4	1	0	0	0	0	0	0	0	1



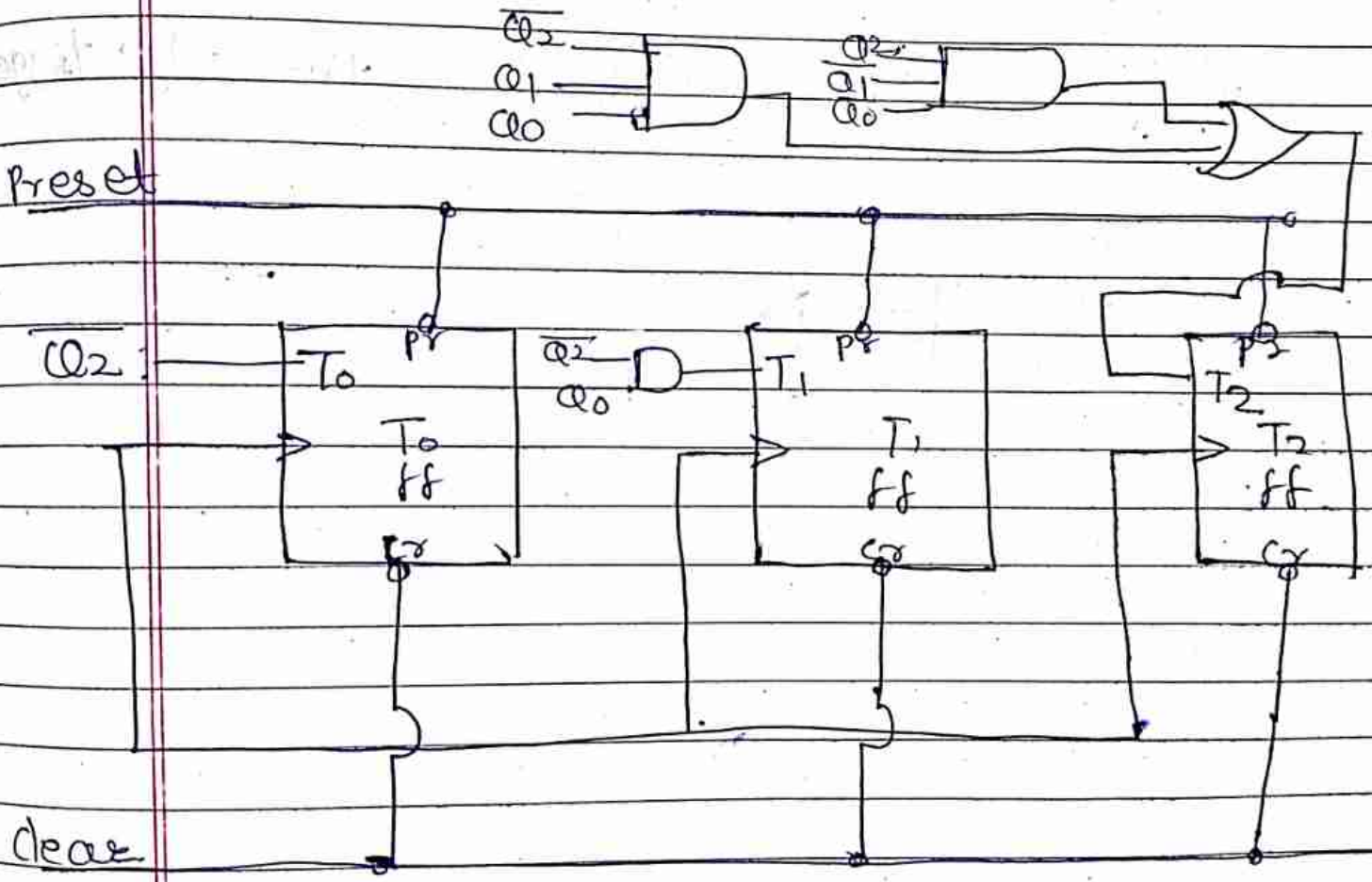
$T_0 = \overline{Q_2}$



$T_1 = \overline{Q_2}\, Q_0$



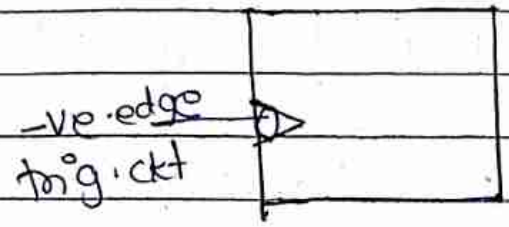
$T_2 = \overline{Q_2}\, Q_1 Q_0 + Q_2\, \overline{Q_1}\, Q_0$



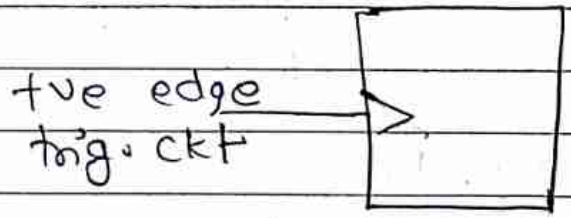
Asynchronous Counter design.

1) Use T-ff, provide logic 1 as i/p to each ff

2) for up counting use -ve edge triggered clock

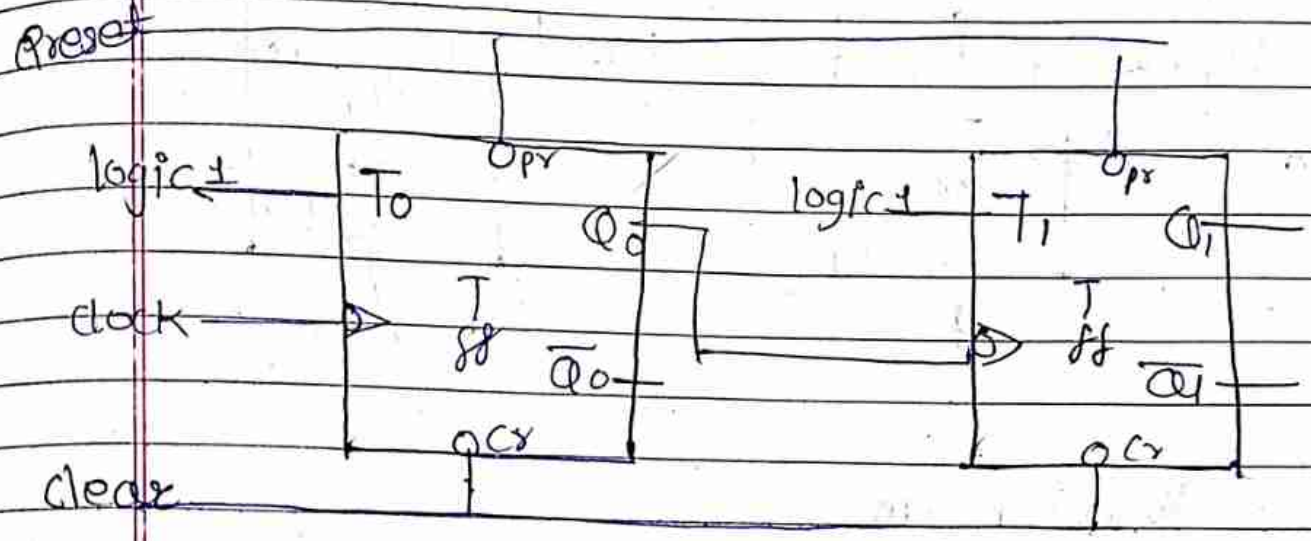


3) For down counting use +ve edge triggered clock

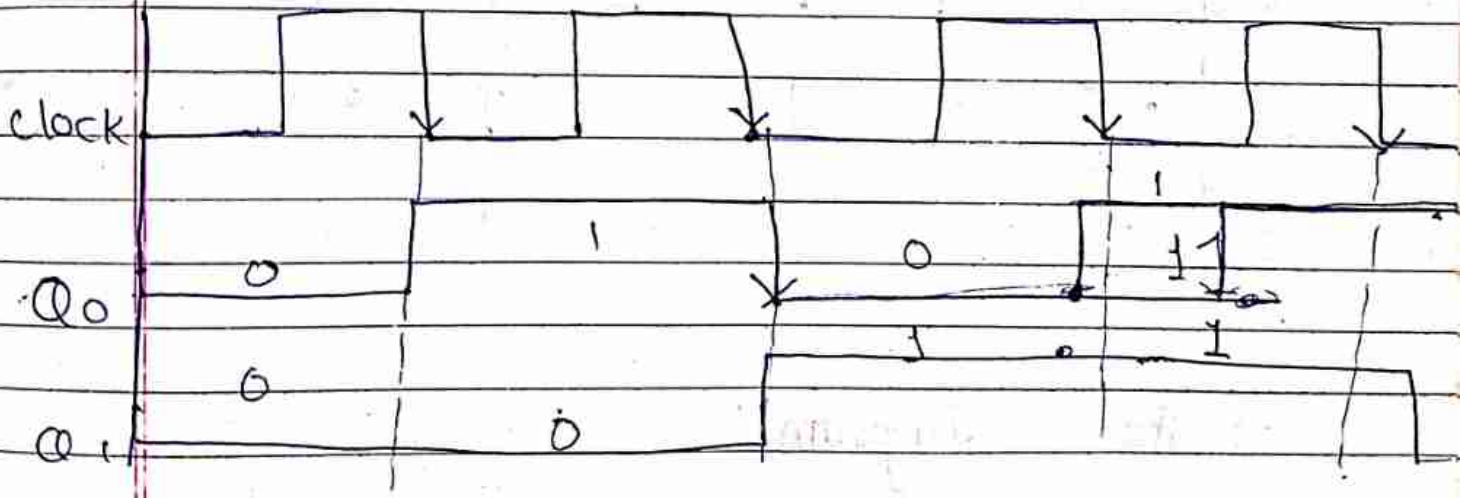


Que : Design : 2 bit Asynchronous up counter

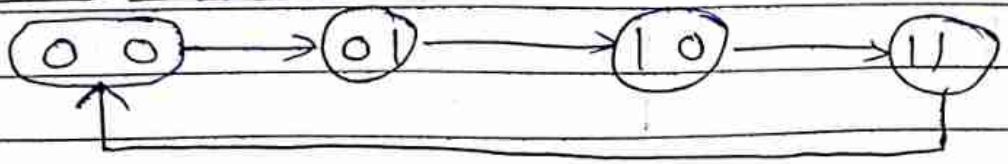
→ we required 2 - T-ff for this design.
for up counting we have to use negative edge triggered clock.



Wave forms.

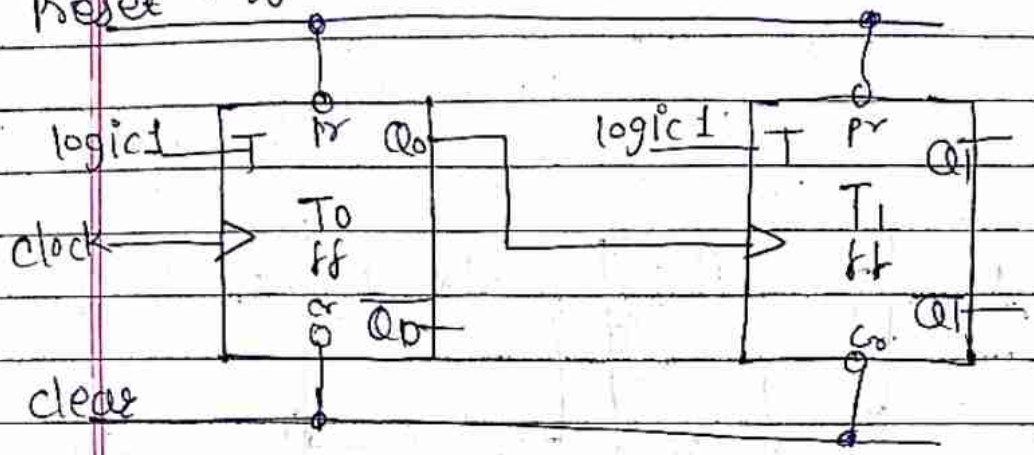


State diagram

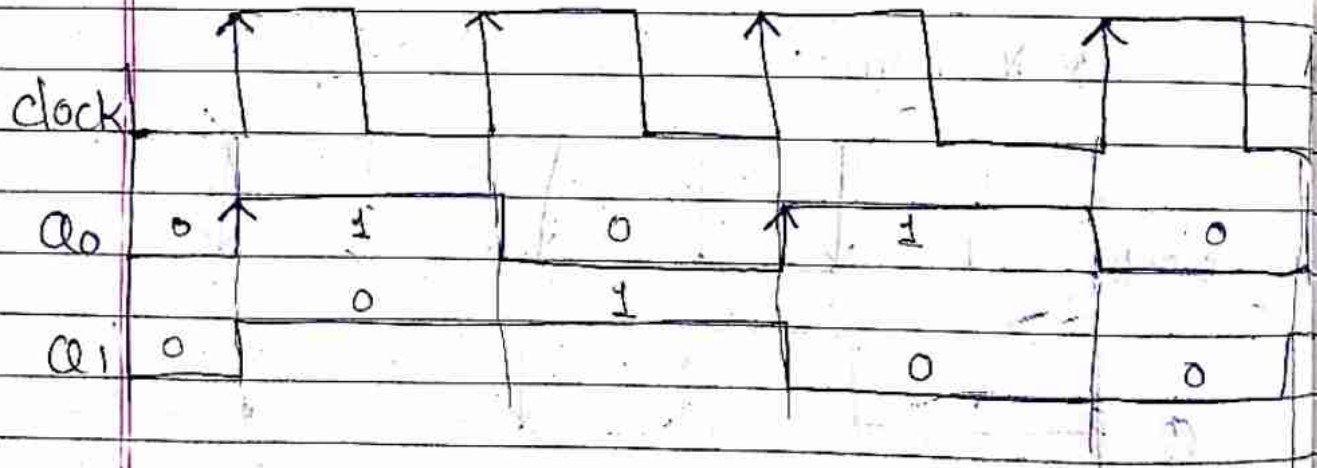


Que: Design two bit Asynchronous down counter

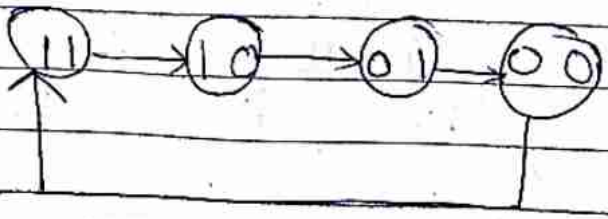
→ For the design we required 2 T-ff for down counting we are using positive edge trigger clock



* Wave form



* State diagram:



7.4.1 The Race-Around Condition

The difficulty of both inputs 1($S = R = 1$) being not allowed in an S–R FLIP-FLOP is eliminated in a J–K FLIP-FLOP by using the feedback connection from outputs to the inputs of the gates G_3 and G_4 (Fig. 7.9). Table 7.3 assumes that the inputs do not change during the clock pulse ($CK = 1$), which is not true because of the feedback connections. Consider, for example, that the inputs are $J = K = 1$ and $Q = 0$, and a pulse as shown in Fig. 7.11 is applied at the clock input. After a time interval Δt equal to the propagation delay through two NAND

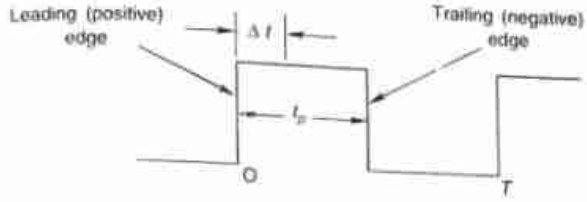


Fig. 7.11
A clock pulse.

gates in series, the output will change to $Q = 1$ (see fourth row of Table 7.3b). Now we have $J = K = 1$ and $Q = 1$ and after another time interval of Δt the output will change back to $Q = 0$. Hence, we conclude that for the duration t_p of the clock pulse, the output will oscillate back and forth between 0 and 1. At the end of the clock pulse, the value of Q is uncertain. This situation is referred to as the *race-around condition*.

The race-around condition can be avoided if $t_p < \Delta t < T$. However, it may be difficult to satisfy this inequality because of very small propagation delays in ICs. A more practical method for overcoming this difficulty is the use of the master–slave (M–S) configuration discussed below.

7.4.2 The Master–Slave J–K FLIP-FLOP

A master-slave J–K FLIP-FLOP is a cascade of two S–R FLIP-FLOPs, with feedback from the outputs of the second to the inputs of the first as illustrated in Fig. 7.12. Positive clock pulses are applied to the first FLIP-FLOP and the clock pulses are inverted before these are applied to the second FLIP-FLOP.

When $CK = 1$, the first FLIP-FLOP is enabled and the outputs Q_M and $\bar{Q}_M$ respond to the inputs J and K according to Table 7.3. At this time, the second FLIP-FLOP is inhibited

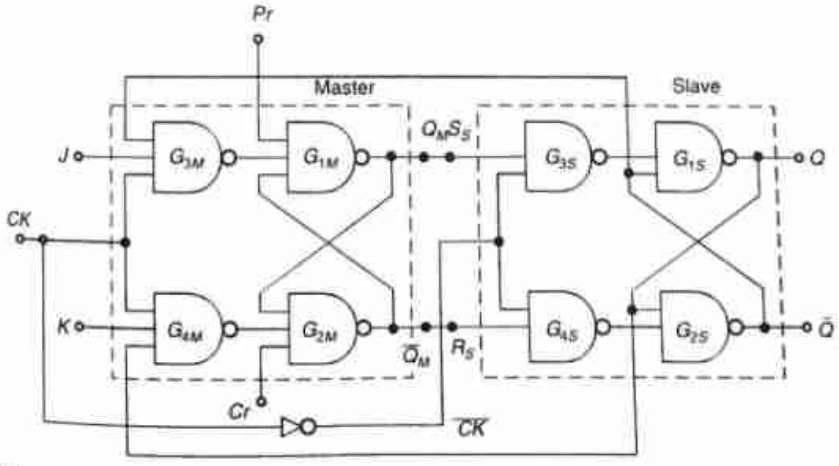


Fig. 7.12