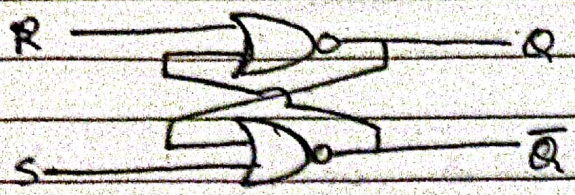
SR Latch

- Storage element

Nor Version

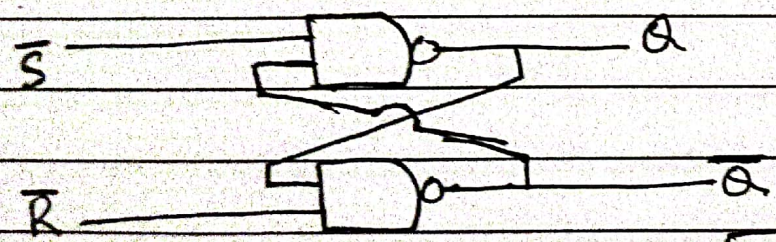


R : Reset | if(R=1) Q=0
S : Set | if(S=1) Q=1

T T

S	R	Q	$\bar{Q}$
0	0	Previous Output	
0	1	0	1
1	0	1	0
1	1	Invalid	

Nand Version



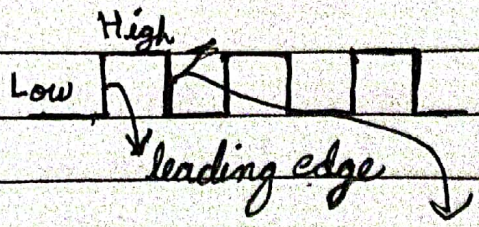
It is called $\bar{S}\bar{R}$ as a way to reconcile it with the Nor version Truth Table.

T T

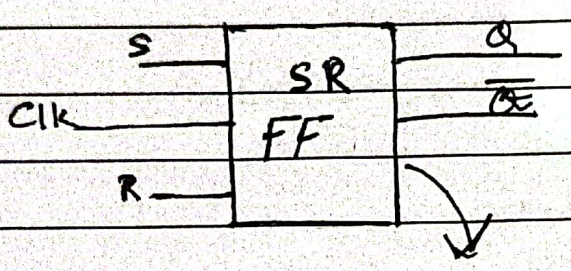
$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$
0	0	Invalid	
0	1	1	0
1	0	0	1
1	1	Previous Output	

Scanned with CamScanner

Clock



Clock is something that alternates between 1 and 0 repeatedly.



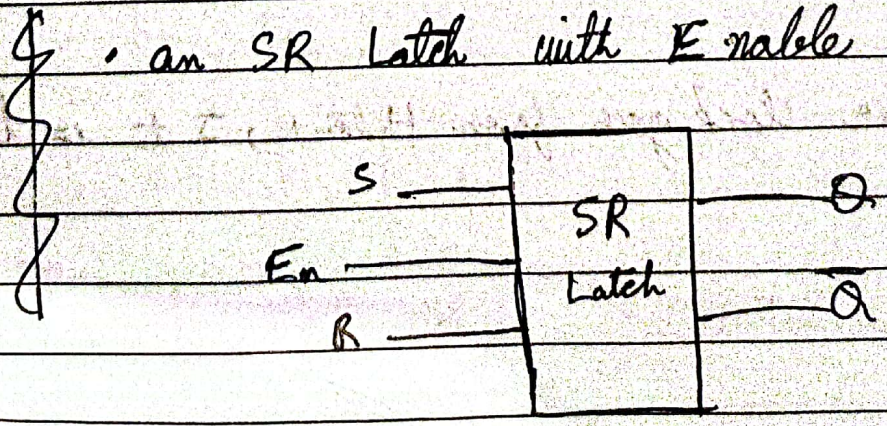
This will only work when the clock is high.

Duty cycle = $\frac{\text{Ration of time the signal is high.}}{\text{total time.}}$

$= \frac{t/2}{t} = \frac{1}{2}$ = 50% of the time

gated SR Latch

- an SR Latch with Enable



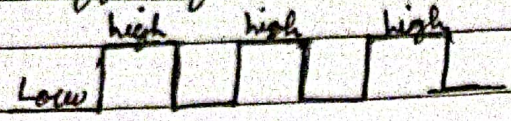
Scanned with CamScanner

Date:

Sun Mon Tue Wed Thu Fri Sat

Triggering Methods

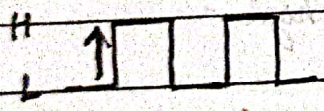
Level Triggering :-



When ever the clock is high, the mechanism is triggered.

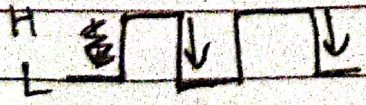
Edge Triggering :-

① +ve edge



Whenever clock goes from L to H, It is triggered.

② -ve edge



Whenever clock goes from H to L, It is triggered

Scanned with CamScanner

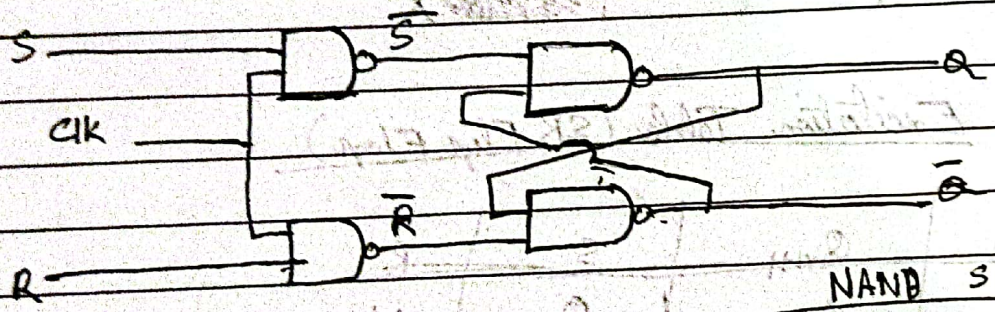
Date:.....

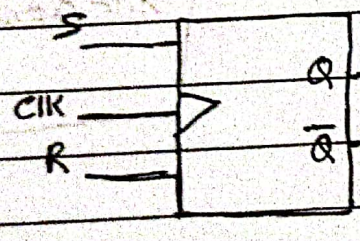
Sun Mon Tue Wed Thu Fri Sat

Difference between Latch and flip flop.

- Latch changes when the ~~ip~~ input changes while it is enabled.
- In flip flop, the ~~input~~ input will be checked only when the clock goes ~~from L to H or H to L~~ ~~is high~~ goes from either higher to low or Low to high.

SR Flip Flop





NAND SR Latch

$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$
0	0	Invalid	
0	1	1	0
1	0	0	1
1	1	Previous	

CLK	S	R	Q	$\bar{Q}$
0	X	X	Previous output	
1	0	0	Previous output	
1	0	1	0	1
1	1	0	1	0
1	1	1	Invalid	

Imp The disadvantage of SR flip flop is the fact that there is an Invalid state.

COPY

Scanned with CamScanner

Date:.....

Sun Mon Tue Wed Thu Fri Sat

Characteristic Table (SR Flip Flop)

Clk=1

Q_n	S	R	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	X
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	X

Q_n \ SR	00	01	11	10
0	0	0	X	1
1	1	0	X	1

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

Excitation Table (SR Flip Flop)

Q_n	Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

X :- don't care

for S :-

Q_n \ Q_{n+1}	0	1
0	0	1
1	0	X

$S = Q_{n+1}$

for R :-

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

Q_n \ Q_{n+1}	0	1
0	X	0
1	1	0

Scanned with CamScanner

Date:.....

Sun Mon Tue Wed Thu Fri Sat

D Flip Flop

Q_n = Present output

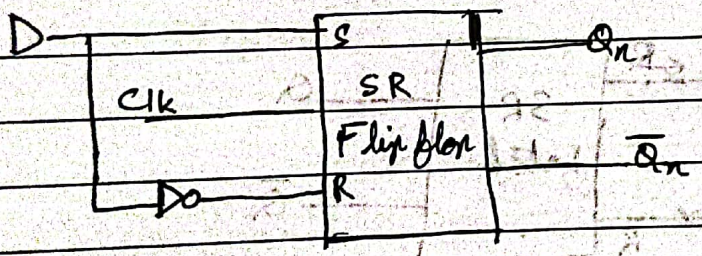
T T SR Flip flop

Clk	S	R	Q_{n+1}
0	X	X	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	invalid

Change only occurs when S and R are opposite.

→ This is ignored as we do not use this.

So,



T T

Clk	D	Q_{n+1}
0	X	Q_n
1	0	0
1	1	1

Imp | An advantage of D flip flop is that there is no invalid state.

COPY

Scanned with CamScanner

Date:.....

Sun Mon Tue Wed Thu Fri Sat

D Latch

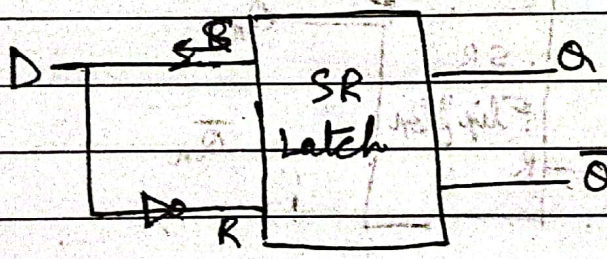
II SR Latch

S	R	Q	$\bar{Q}$
0	0	Previous	
0	1	0	1
1	0	1	0
1	1	invalid	

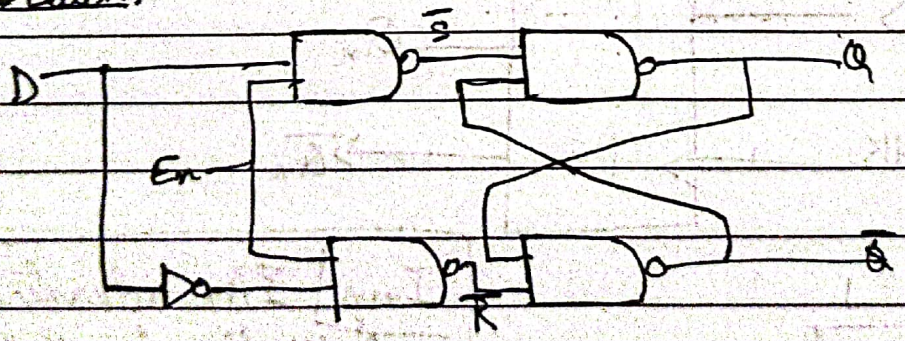
output changes only when S and R are complements of each other.

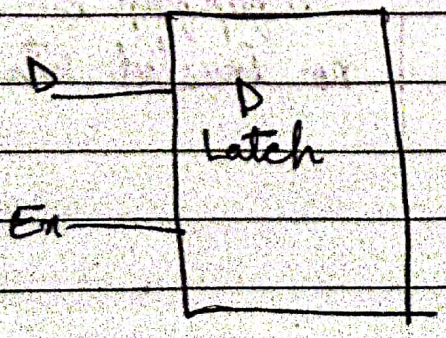
This is ignored because we don't use it.

So,



Gated D Latch:-





Scanned with CamScanner

Date:

Sun Mon Tue Wed Thu Fri Sat

Characteristic Table (D Flip flop)

CLK	D	Q_{n+1}
0	X	Q_n
1	0	0
1	1	1

Char Table :-

Q_n	D	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

$Q_{n+1} = D$

Excitation Table (D flip flop)

Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

for D :-

Q_n \ Q_{n+1}	0	1
0	0	1
1	0	1

$D = Q_{n+1}$

COPY

Scanned with CamScanner

Date:.....

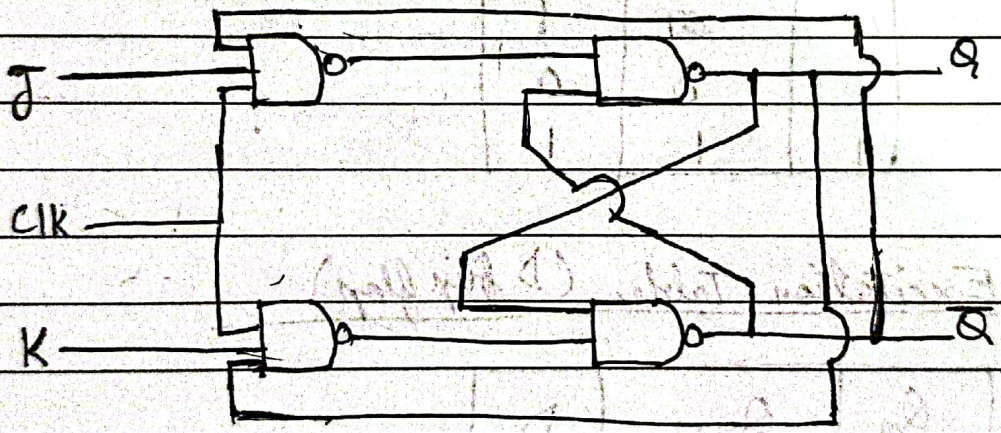
Sun Mon Tue Wed Thu Fri Sat

J K flip flop

T.T (SR FF)

Clk	S	R	Q_{n+1}
0	X	X	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	Invalid

JK flip flop



T.T

Clk	J	k	Q_{n+1}	$\bar{Q}_{n+1}$
0	X	X	Previous	
1	0	0	Previous	
1	0	1	0	1
1	1	0	1	0
1	1	1	?	?
1	1	1	$\bar{Q}_n$	Q_n

Imp In JK flip flop, for $J=1, K=1$, The output is (assuming starting $Q_n=0, \bar{Q}=1$).

$Q = 0\,1\,0\,1$

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

So,

$Q_{n+1} = \bar{Q}_n$

This is called toggling.

Scanned with CamScanner

Date:.....

Sun Mon Tue Wed Thu Fri Sat

T.T (JK ff)

Clk	J	K	Q_{n+1}
0	X	X	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	$\overline{Q_n}$

Excitation Table

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Characteristic Table

Q_n	J	K	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

JK

Q_n	00	01	11	10
0	0	0	1	1
1	1	0	0	1

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

for J:-

Q_n \ Q_{n+1}	0	1
0	0	1
1	X	X

$J = Q_{n+1}$

for K:-

Q_n \ Q_{n+1}	0	1
0	X	X
1	1	0

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

Scanned with CamScanner

Date:

Sun Mon Tue Wed Thu Fri Sat

Race Around Condition (JK ff)

- It is the name of the toggling condition that occurs in JK flip flop, when J=1, K=1 but multiple times in one cycle.
- A way to solve this problem is that the delay of change be greater than half of the time period.

  delay > T/2

- The second way is edge triggering.
- The third way is master-slave flip flop.

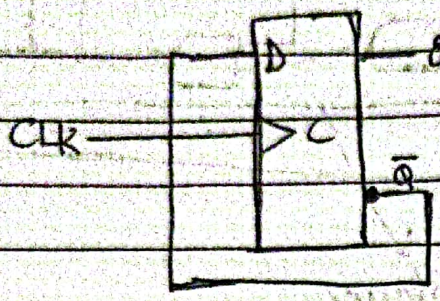
Scanned with CamScanner

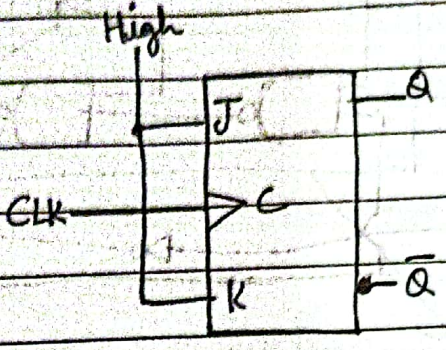
Date:.....

Sun Mon Tue Wed Thu Fri Sat

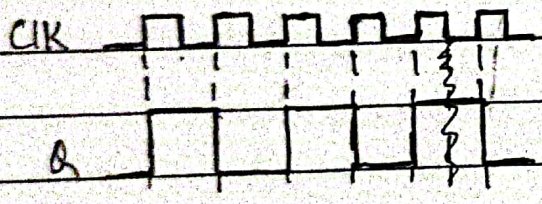
Frequency Division

- Another application of flip flop is dividing the frequency of a periodic wave.





- by setting $D = \bar{Q}$ in D flip flop or $(J = K = 1)$, the output would be a periodic wave with half the frequency.



- This happens because the flip flops are edge triggered, In the above diagram, they are +ve edge triggered.

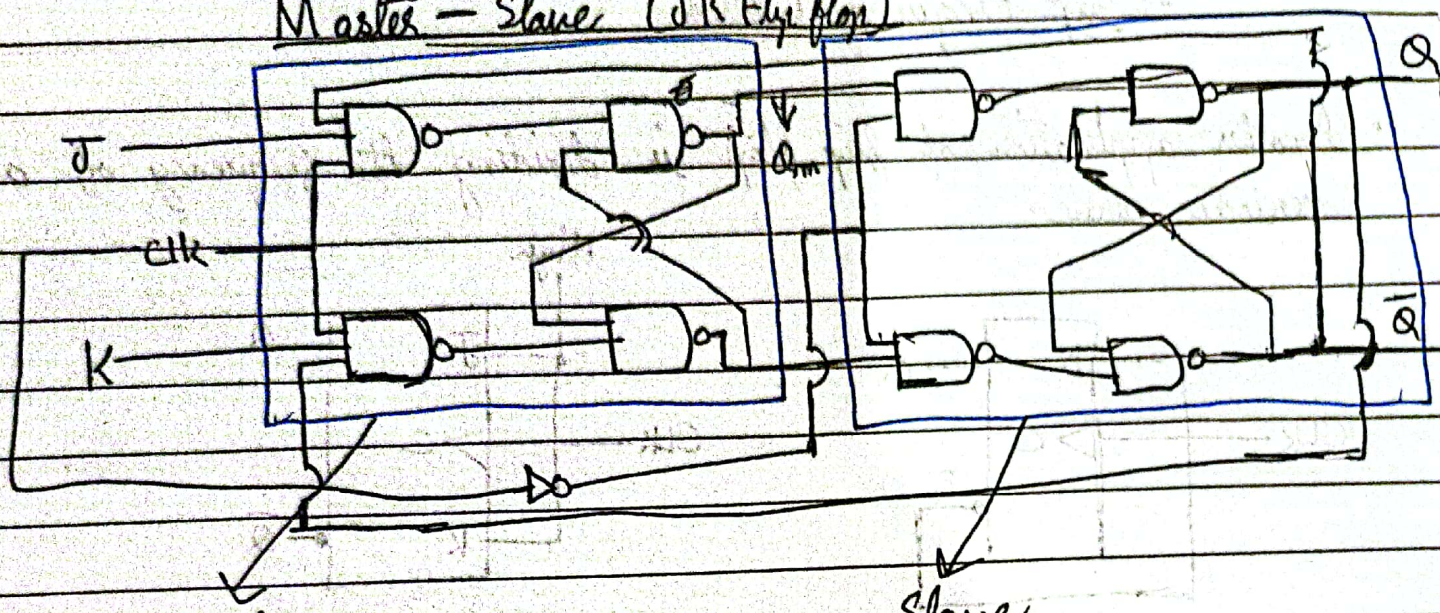
COPY

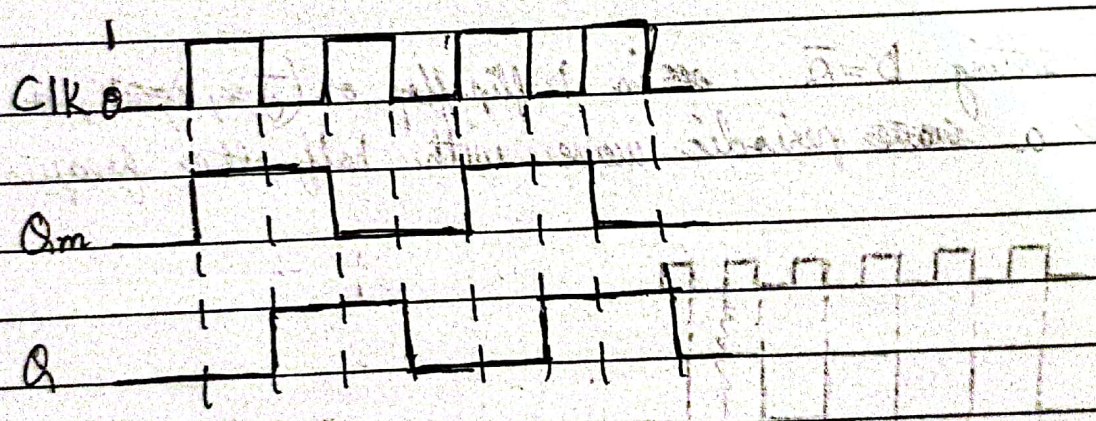
Scanned with CamScanner

Date:.....

Sun Mon Tue Wed Thu Fri Sat

Master – Slave (JK Flip flop)





Scanned with CamScanner

JK to D flip flop

① Identify available and required flip flop

av. ff = JK

req. ff = D

② Make characteristic table for required flip flop.

Q_n	D	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

$Q_{n+1} = D$

③ Make excitation table for available flip flop.

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

④ Write boolean expression for available ff.

Q_n	D	Q_{n+1}	J	K
0	0	0	0	X
0	1	1	1	X
1	0	0	X	1
1	1	1	X	0

for J :-

Q_n \ D	0	1
0	0	1
1	X	X

Scanned with CamScanner

for J:-

Q_n \ D	0	1
0	0	1
1	x	x

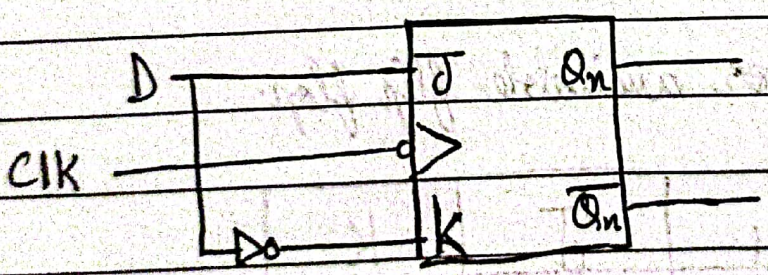
for K:-

Q_n \ D	0	1
0	x	x
1	1	0

$J = D$

$K = \overline{D}$

(5) Draw the Circuit



Scanned with CamScanner

Date:.....

Sun Mon Tue Wed Thu Fri Sat

SR ff to JK flip flop

avail = SR

req = JK

Q_n	S J	R K	Q_{n+1}	S 変	R 変
0	0	0	0	0	X
0	0	1	0	0	X
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	1	X	0
1	0	1	0	0	1
1	1	0	1	X	0
1	1	1	0	0	1

for S:-

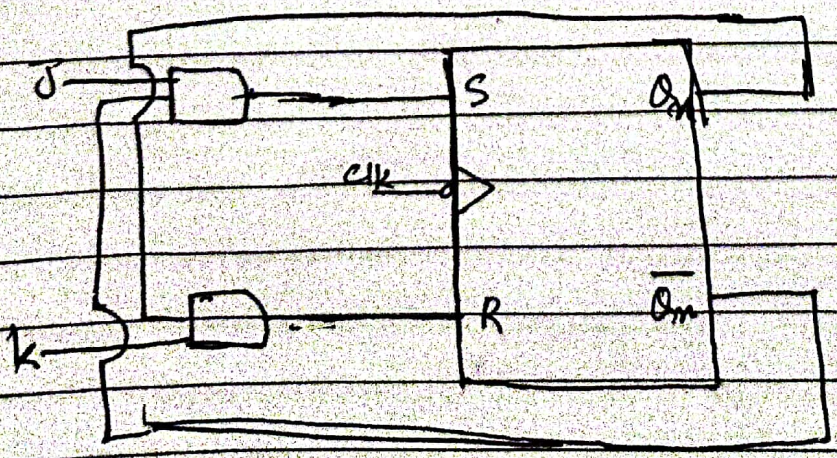
Q_n \ JK	00	01	11	10
0	0	0	1	1
1	X	0	0	X

for R:-

Q_n \ JK	00	01	11	10
0	X	X	0	0
1	0	1	1	0

$S = \overline{Q_n} J$

$R = Q_n K$



Scanned with CamScanner