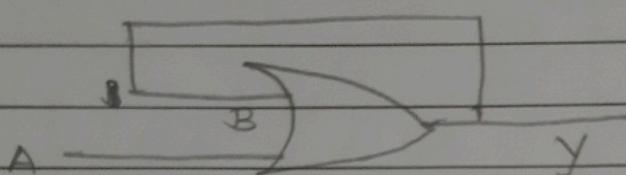


* Sequential Circuits. *

- Sequential circuit is type of logic circuits whose output depends not only on present value of its input signals but on sequence of past inputs; input history as well.
- This is contrast to combination logic.

* Consider OR Gate with feedback.

IMP



A	Y
0	B
1	1

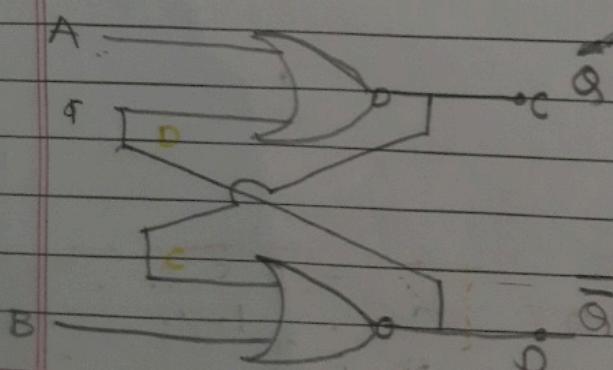
- For OR gate if A is 1 then Y 1 but if A = 0 then output depends on other OR input.
- Due to direct feedback $A=0$ then $Y=B$ then directly $B=Y$

A	Y	$Y = 0 \text{ or } 1$
0	Y	
1	1	

Sequential circuits allow multiple outputs combining same input combi. 1 to many combi.

* FLIP-FLOP

- It uses two NOR gate and cross coupling.
- There are two stable states and can be used to store state info. - a bistable multivibrator.
- The circuit can be made to change state by signal applied to one or more control inputs & will have one ~~two~~ output.



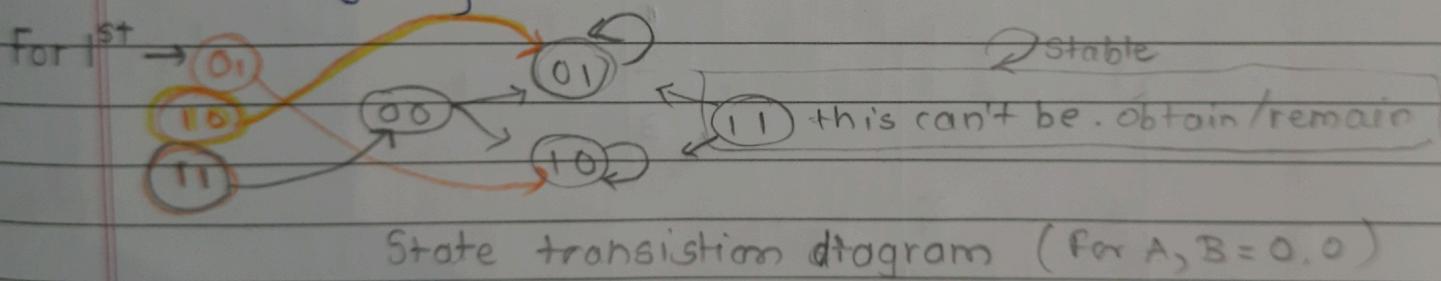
	A	B	C	D
1	0	0	0	1
2	0	1	1	0
3	1	0	0	1
4	1	1	0	0

- color coding

- In nor gate here, if A is 1 then output C = 0 and if B is 1 then output D always = 0.
- For NOR gate.
- IF one input (A) is 0 then output is complement of other input.
i.e when A is 0 then C is \bar{D}
& when B is 0 then D is \bar{C}
- Two variable and 1 eqn (Nor or eqns < Nor. variable them multiple soln)
As to $C = \bar{D}$ or $D = \bar{C}$
- ∴ By putting $C = 0$ & $D = 1$
& $C = 1$ & $D = 0$

∴ There are two values.

- * IF both output taken 0 in 1st. $C = 0$ and $D = 0$
Both the gates trying to make 1 but the first gate becomes 1 and 0,0 state is self destroying state. (state transition)



State transition diagram (for A, B = 0, 0)

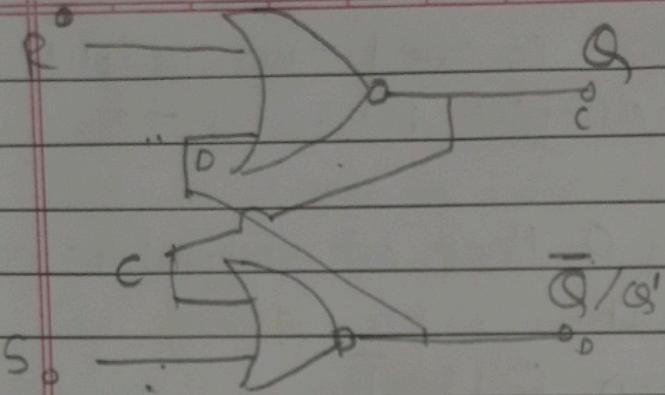
- Here, if current state input is 0,0 then (A, B) then it depends on previous state if it pre.state is A, B (0, 1) that that time output for them is 1, 0 then output for AB (0, 0) is 1, 0.
- IF previous state is AB, (1, 0) the output for AB (0, 0) then output for AB (0, 0) is (0, 1)

Last

- For A, B (0, 0) last input state is (1, 1) and output state is 0, 0 then it not maintain.
- * If (A, B) (1, 1) is banned (forbidden) the (0, 0) not occurs. Last state is written because (0, 0) at last is banned.

A	B	C	D
0	0	0	0
0	1	1	0
1	0	0	1
1	1	0	0

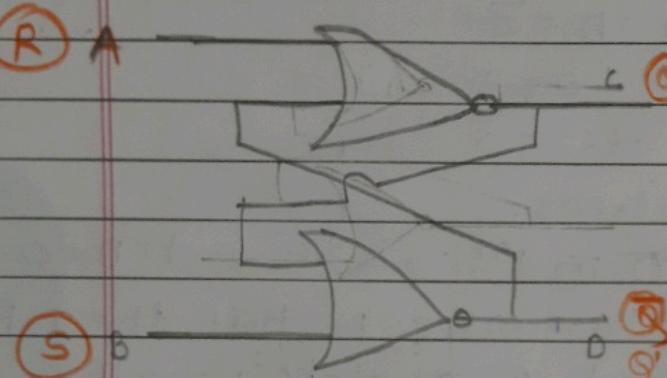
Banned.



R	S	Q	\bar{Q}
0	0	0	1
0	1	1	0
1	0	0	1
1	1	0	0

Banana

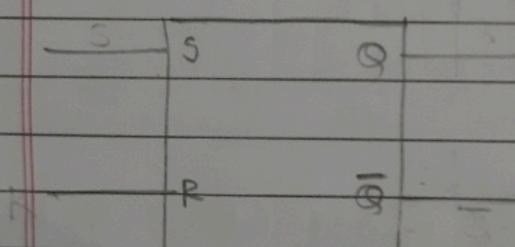
(RS) * Flip-Flops.



R S Q \bar{Q}/\bar{Q}'

A	B	C D
0	0	Last state
0	1	1 - 0
1	0	0 - 1
1	1	0 - 0

Forbidden

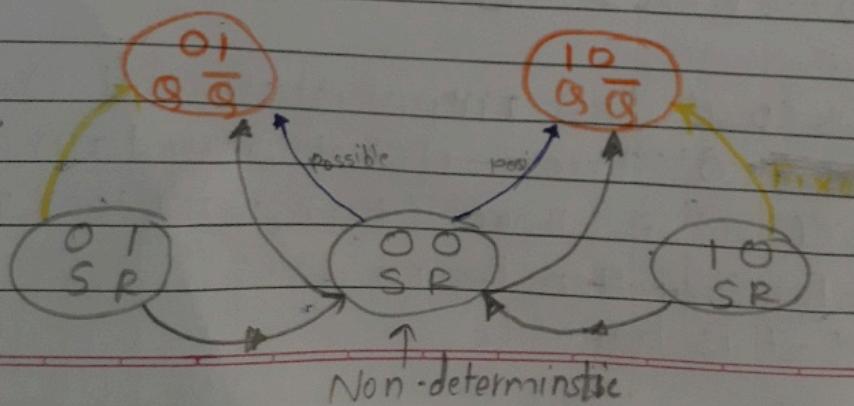


S	R	Q	\bar{Q}
0	0	0	1
0	1	0	1
1	0	1	0

Forbidden

Block diagram.

- * For 2nd & 3rd combi. output is unique fixed.
- * For 1st output depends on how you arrived for input

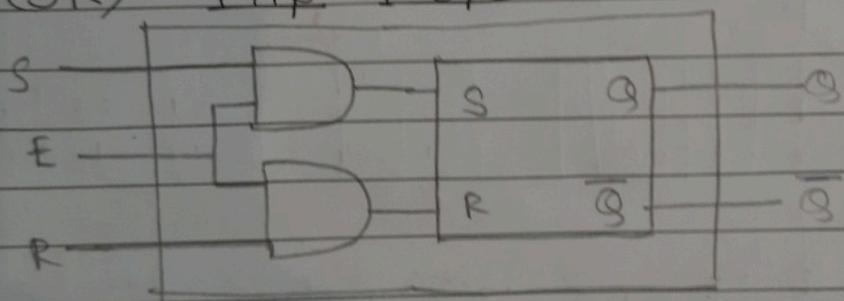


- ① When it is $(0,0)$ from $(0,1)$ i.e. From deterministic input $(0,1)$ to non-deterministic $(0,0)$ then the output for 1st combi. is (01) is remained.
- ② When ^{enter} input $(1,0)$ to the $(0,0)$ then the output is $(1,0)$ is remained.
- How you arrived — Memory.

S	R	Q	\bar{Q}	
1	0	1	0	Fixed, from previous.
0	0	1	0	←
0	1	0	1	Fixed, from previous.
0	0	0	1	← From previous.

= D AND

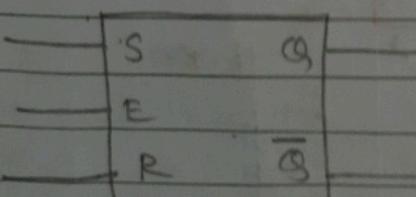
* (SR) Flip - Flops.



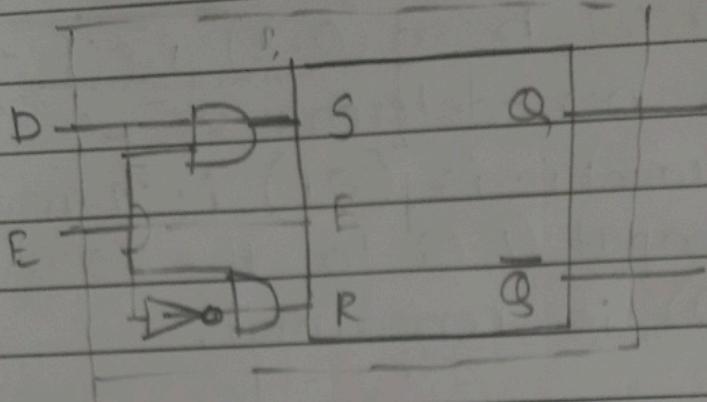
A flip - Flop with overriding input called Enable (E) input.

E	S	R	Q	\bar{Q}	
0	X	X			Last state
1	0	0			Last state
1	0	1	0	1	
1	1	0	1	0	
1	1	1	0	0	← forbidden.

By packing



Next



Compressed

E	S	D	R	Q	Q-bar
0	X				Last S
I	0	0	0	0	1
I	1	1	1	1	0
I					

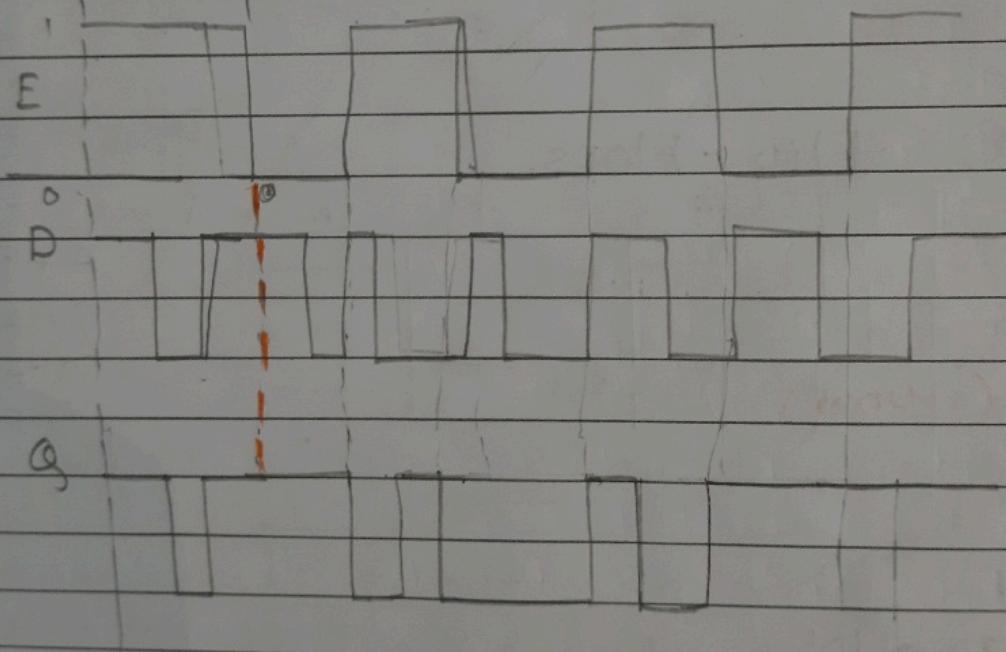
If E is 0 whatever may be D, then S,R is 0,0

Further,

E	Q	Q-bar
0	LS	
1	D	D

Q Follows E when E=1

D Flip-Flop



Whenever E is 1 $Q = D$ when E is 0 the Q is old Q or Q is D at time E is 1.

* Flip-Flops \Rightarrow CLOCK

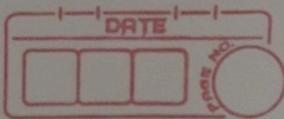
S	E	D	S	Q	CLK	S	R	Q	Q-bar
0	0	0	0	0	0	X	X	LS	
1	0	0	0	0	1	X	X	LS	
0	1	0	0	0	0	X	X	LS	
0	1	1	1	1	1	0	0	LS	
1	1	1	1	1	1	0	1	01	
1	1	1	1	0	1	1	0	10	
1	1	1	1	0	1	1	0	10	
1	1	1	1	0	1	1	0	10	
1	1	1	1	0	1	1	0	10	

CLK basicff

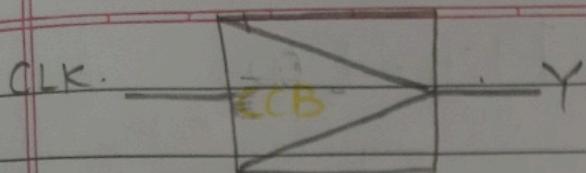
█ = clock circuit block (CCCB)

Forbidden

Extra



CCB



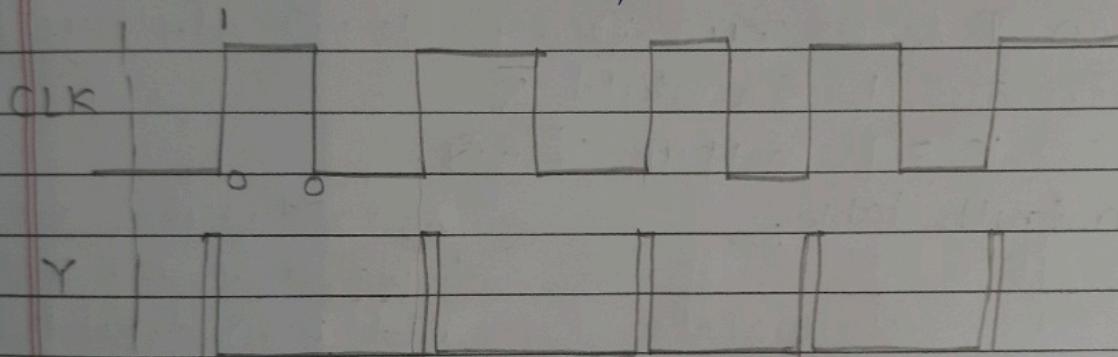
it has one input & output(Y)

→ ↓ \Rightarrow CLK or input changing
from 1 to 0

$\uparrow \Rightarrow$ CLK or \rightarrow \uparrow ①
0 to 1

CLK	Y
0	0
1	0
	↓ ①
	↑ ①
	1

* Period of output of edges of clock = 10ns.
arrow $\downarrow \uparrow$ is 10ns period after change over.

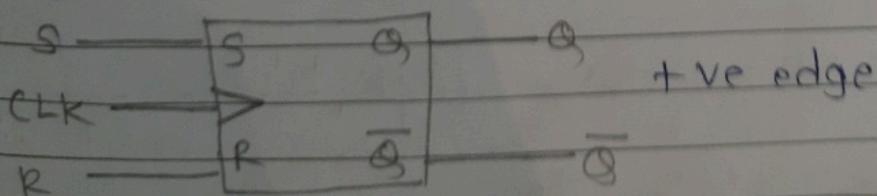


Description.

When input is 0 becomes 1 from 0 output becomes 1 for only 10ns.

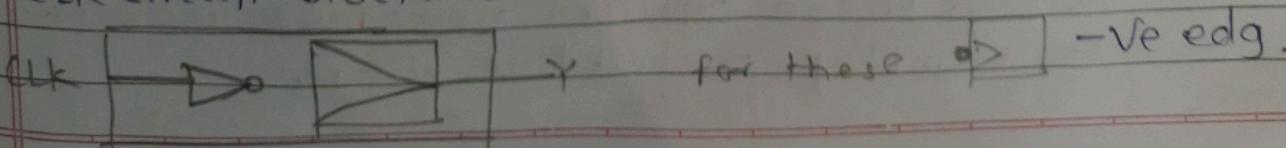
- clock circuit block is single input, output digital circuit where output is normally 0 except for small period 10ns after every +ve rising edge. for only 10ns otherwise last state.

* Symbol for clocked SR flip flop.



+ve edge

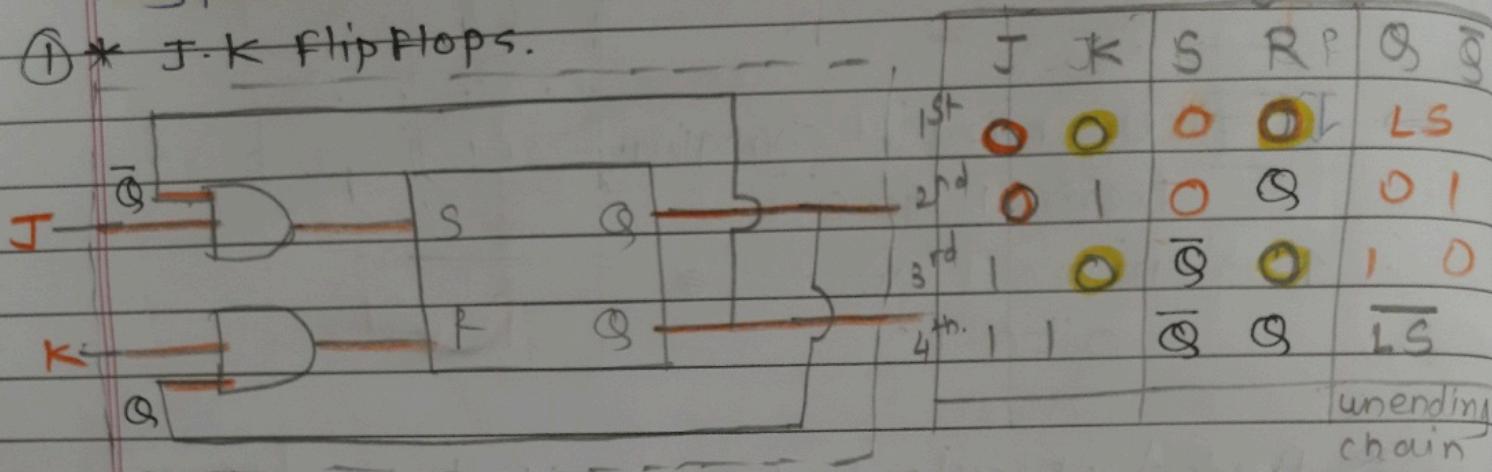
* CLK circuit block working on falling edge.



for these \Rightarrow -ve edg.

* Types of SR (or RS) Flip-Flops

① * J-K flip-flops.



It have feedback.

- When J is 0 & S is 0 & K=0 always R=0
- When J = 1 S would \bar{Q}
- When K = 1 output R=Q.

Then truth table.

	S	R	Q	\bar{Q}
1 st	0	0	0	1
2 nd	0	0	0	1
	0	1	0	1

Last state.

As 2nd state, J=0, R=1
 $\therefore S=0 \& R=\bar{Q}$

But R=Q depends on the current output.

hence if R=1 (S=0)
 as if Q=0,1 then $\bar{Q}, \bar{\bar{Q}} = 0, 1$

(S=0, R=1) But if R=0 (S=0) then there is no last state.

IF last state Q=0 & Q=1 then leads (S,R)=0,0
 leads last state but last state is 0,1

	S	R	Q	\bar{Q}
3 rd	0	0	1	0
	1	0	1	0

IF $\bar{Q}=0 \& Q=1$ Last state.

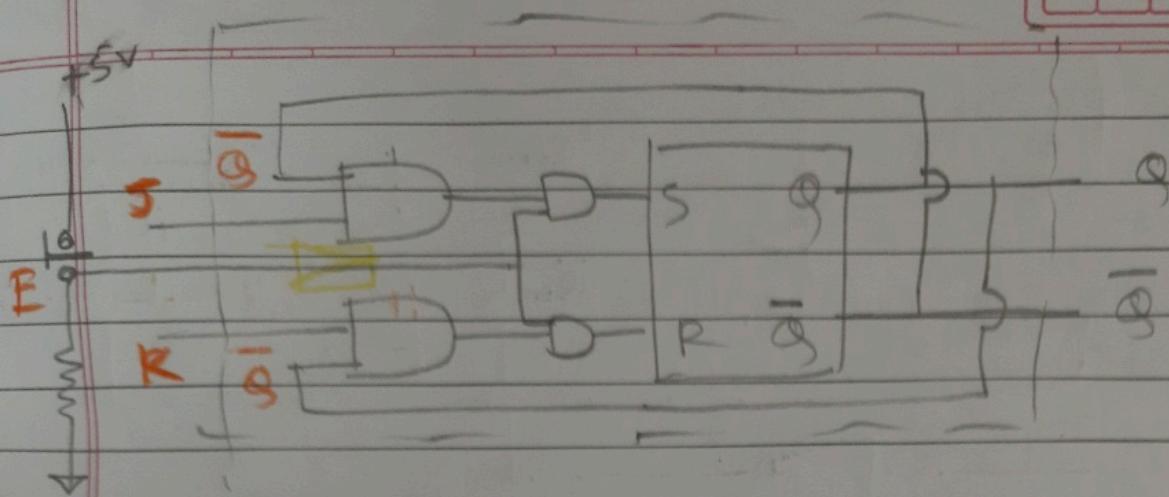
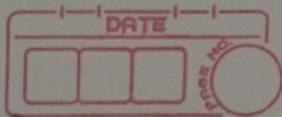
IF $\bar{Q}=1$ the SR=1,0, the output 1,0

	1	0	1	0
4 th	0	1	0	1

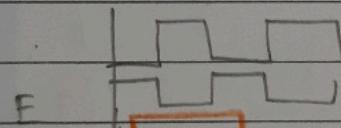
IF last state,
 $Q=1 \& \bar{Q}=0$ then S,R=1,0
 (inputs)

toggling if $Q=0 \& \bar{Q}=1$ SR=0,1

When J, & K=1,1 the get square wave at Q, \bar{Q}
 To stop these square wave use enable
 SR flip flop instead of simple.



CLK	E	J	K	S	R	Q	\bar{Q}	Notes
0	↓	0	X	X	0	0	LS	
1	1	0	0	0	0	LS		
		-0	1	0	0	Q	01	Normally E low.
		1	1	0	\bar{Q}	0	10	
		1	1	1	\bar{Q}	Q	LS	



If we make $E = 1$ for small time, then $E = 0$ hence the toggling is stopped.

IF CLK circuit block is placed at rising edge.
E becomes 0 for small time (10ns) Last case.

Hence if propagation delay is 7ns the 1 toggling.
compliment of last state depends on toggling.
it is X. (problem)
Multiple toggling useless.
∴ it is problem of single stage J-K flip-flops

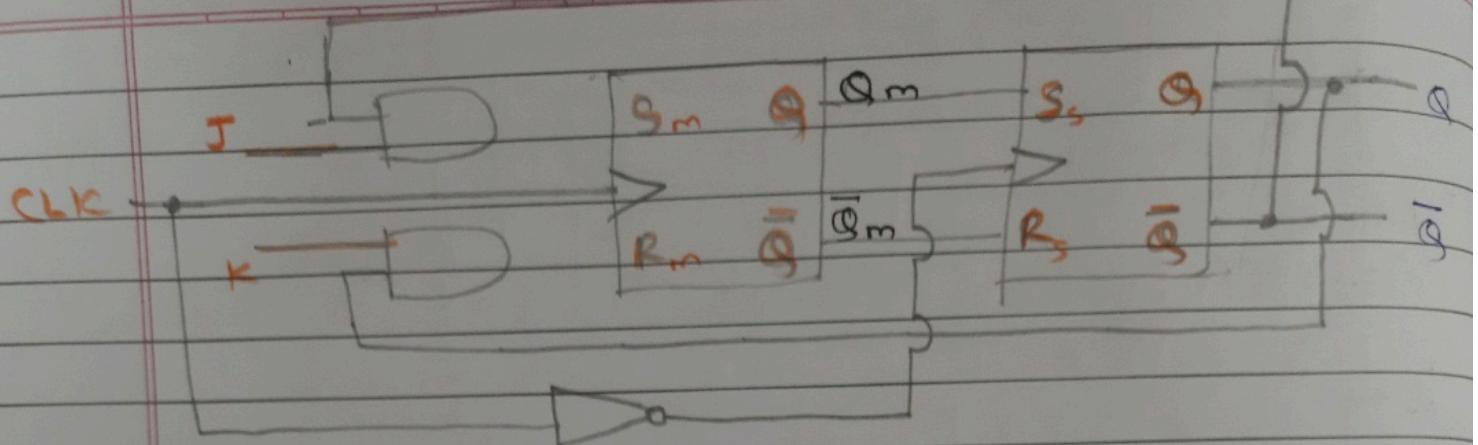
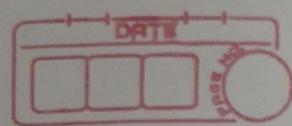
* Master slave J-K Flip Flops.

It is basic building blocks of digital circuits.

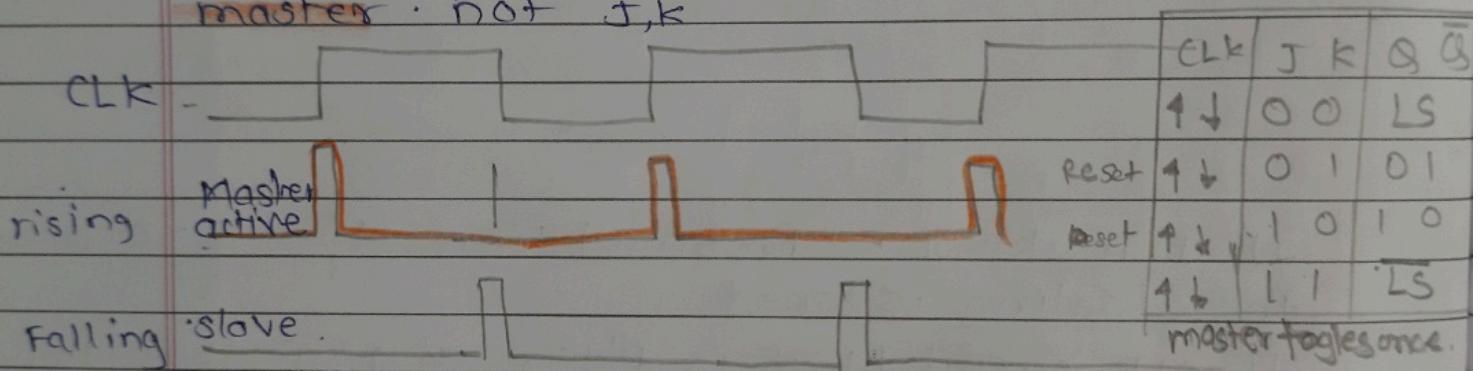
Universal flip flop, store, compliment, encounter.

- It has two SR flip-flops.
- Reverse Feedback from slave.

$J, K = 1, 1$
Flip-flop toggles.



- Active state is for 1ns at rising edge master sets output depend upon J, K .
- And at time of falling edge slaves follows master · not J, K



- At time of rising edge overall Q & \bar{Q} depend on J, K decide Q_m & \bar{Q}_m but don't change Q & \bar{Q}
 - A falling edge slaves copy intermediate output master is inactive.
- ↓ decide final output

CLK	J	K	S_m	R_m	Q_m	\bar{Q}_m	$S_s + R_s$	Q_s	\bar{Q}_s
0, 1	X	X	X	X	Last state		X	X	Last state
↑	0	0	0	0	LS		X	X	LS
↑	0	1	0	0	0	1	X	X	1
↑	1	0	0	0	1	0	X	X	0
↑	1	1	0	0	0	1	0	0	1

don't change

Master output are slaves output complement. happens only once.



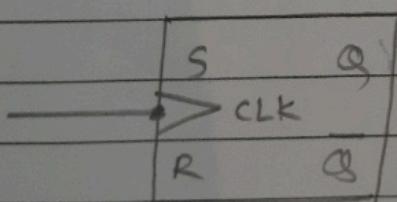
Q9

- IF previous state of actual output is 0,1,
New state of intermediate output is 1,0
- If $\overbrace{\text{---}}^{\text{---}} \rightarrow \overbrace{\text{---}}^{\text{---}}$ is 1,0
 $\downarrow \quad \downarrow$ is 1,0
- ∴ The actual previous output is inverted and remains as intermediate output that doesn't remains actual output again hence multiple toggling is completely avoided

CLK	J	K	S_m	R_m	$Q_m \bar{Q}_m$	S_s	R_s	Q, \bar{Q}
+	X	X	X	X	LS	$\bar{Q}_m Q_m$	Q, \bar{Q}	Q, \bar{Q}

At falling edge S_s & R_s are ~~not~~ of slave are intermediate output of master.

J,K at time of rising edge decide Q, \bar{Q} decide at subsequent falling edge.



→ Final output at falling edge

CLE	JK	Q, \bar{Q}
↓	00	LS
↓	01	01
↓	10	10
↓	11	LS Toggling