Lab - 08

Name :- Yash Gupta

Roll no :- S20200010234

Experiment - 1

S.R. Latch

AIM :- To construct and verify NOR and NAND gates.

Software Used :- Logism Software

Truth Table

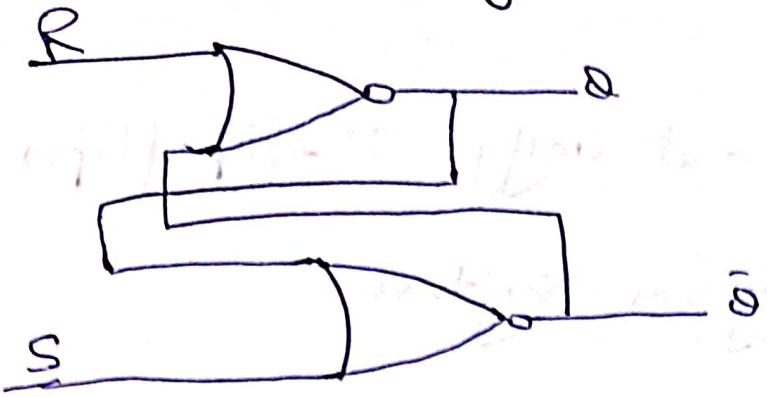
S	R	Q	$\bar{Q}$
0	0	Latched	
0	1	0	1
1	0	1	0
1	1	Illegal	

For SR latch using NOR gates

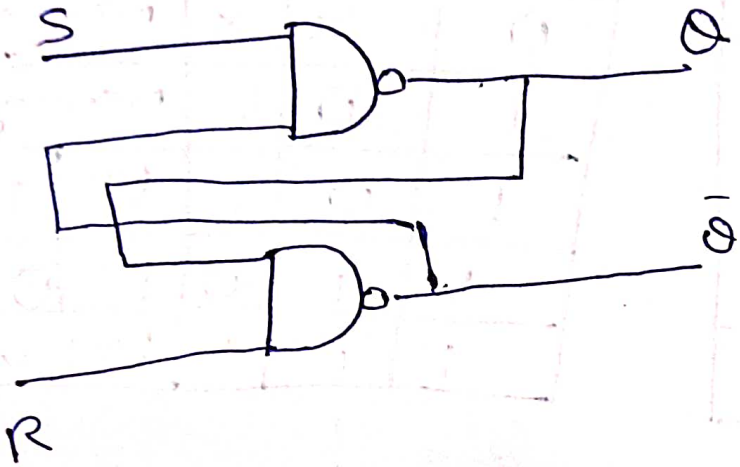
S	R	Q	$\bar{Q}$
0	0	Latched	
0	1	1	0
1	0	0	1
1	1	Illegal	

Logic diagram

Using NOR gates



Using NAND gates



Conclusion

This circuit is simulated in logisim and is verified by changing the values in the circuits

Exp-2

SR-flip-flops

AIM: To construct and verify SR-flip-flops

Software Used: Logism Software

Truth Table:

S	R	Q	$\bar{Q}$
0	0	Not used	
0	1	1	0
1	0	0	1
1	1	memory	

Clk	S	R	Q	$\bar{Q}$
0	x	x	memory	
1	0	0	memory	
1	0	1	0	1
1	1	0	1	0
1	1	1	Not used	

Boolean Expression

$$S^* = \overline{S \cdot clock} \qquad R^* = \overline{R \cdot clock}$$

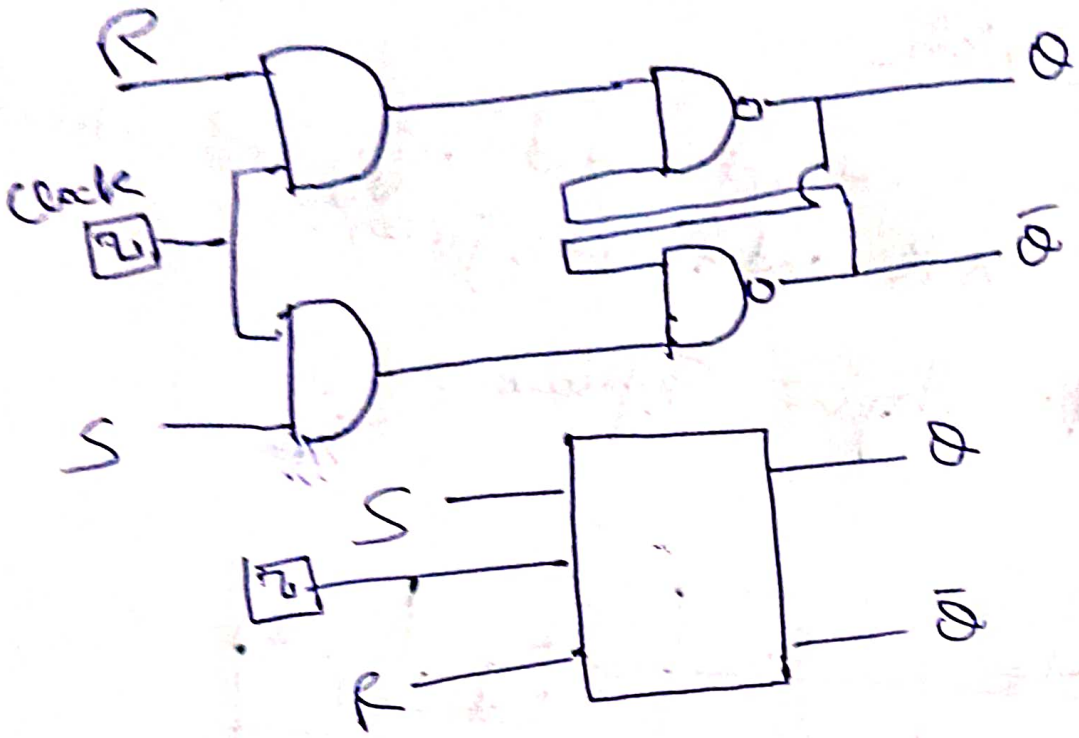
If clock = 0

$S^* = 1, \quad R^* = 1$

If clock = 1

$S^* = \bar{S}, \quad R^* = \bar{R}$

Logic Diagrams



Conclusion:

This circuit is simulated in logism and is verified by changing the values in the circuit.

Exp-3

D-latch

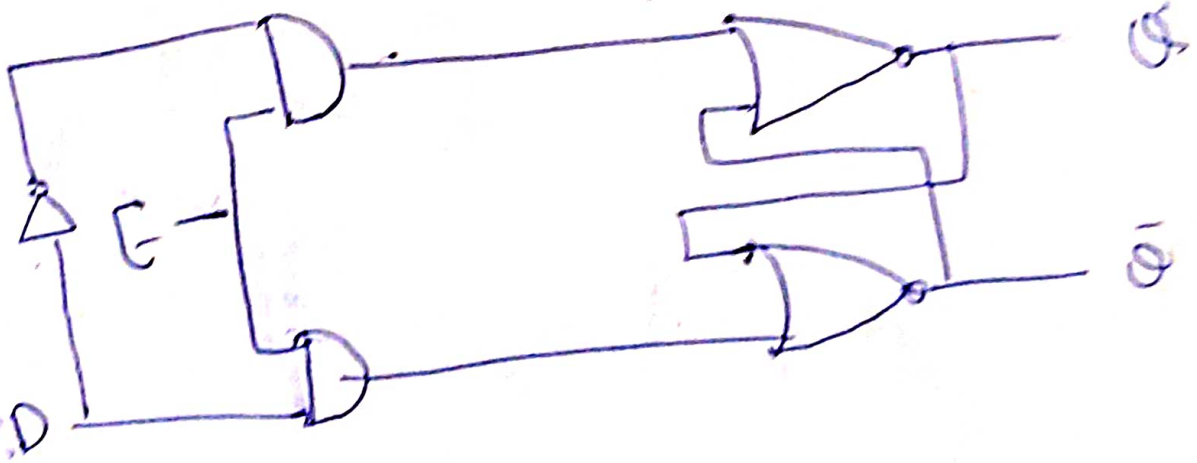
Aim: To construct and verify D latch using NOR and AND gates.

Software Used: Logisim Software.

Truth Table

Enable	D	Q	$\bar{Q}$
0	0	Latched	
0	1	Latched	
1	0	0	1
1	1	1	0

Logic Diagram



Conclusion:

This circuit is simulated in logisim and is verified by changing the value in the circuit.

Exp. 4

D flip flop

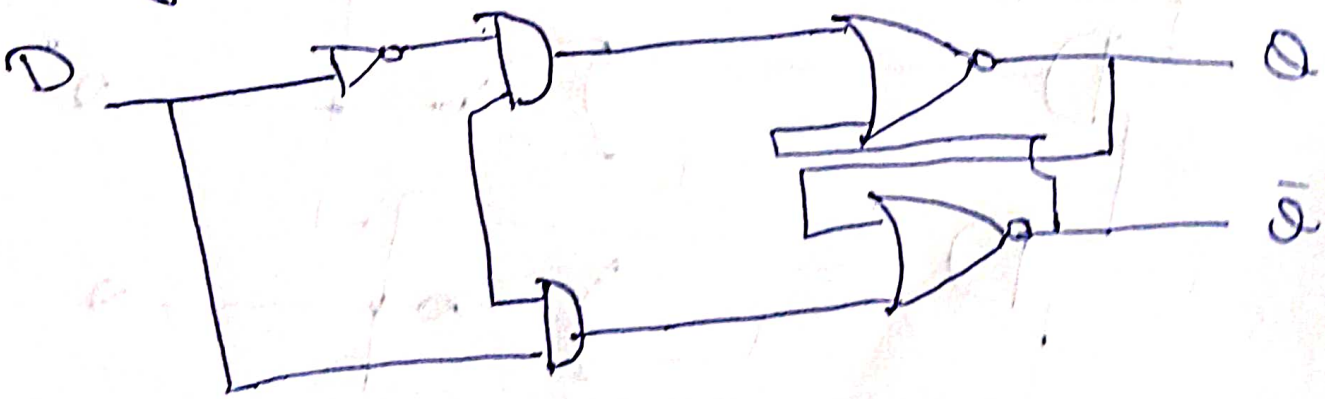
AIM: To construct and verify D-flip flop

Software used: Logisim Software

Truth table

Clock	D	Q	$\bar{Q}$
0	X	memory	
1	1	1	0
1	0	0	1
1			

Logic Diagram



Conclusion:-

This circuit is simulated in logisim and is verified by changing the values in the circuit.

Experiment - 5

JK - flip-flops

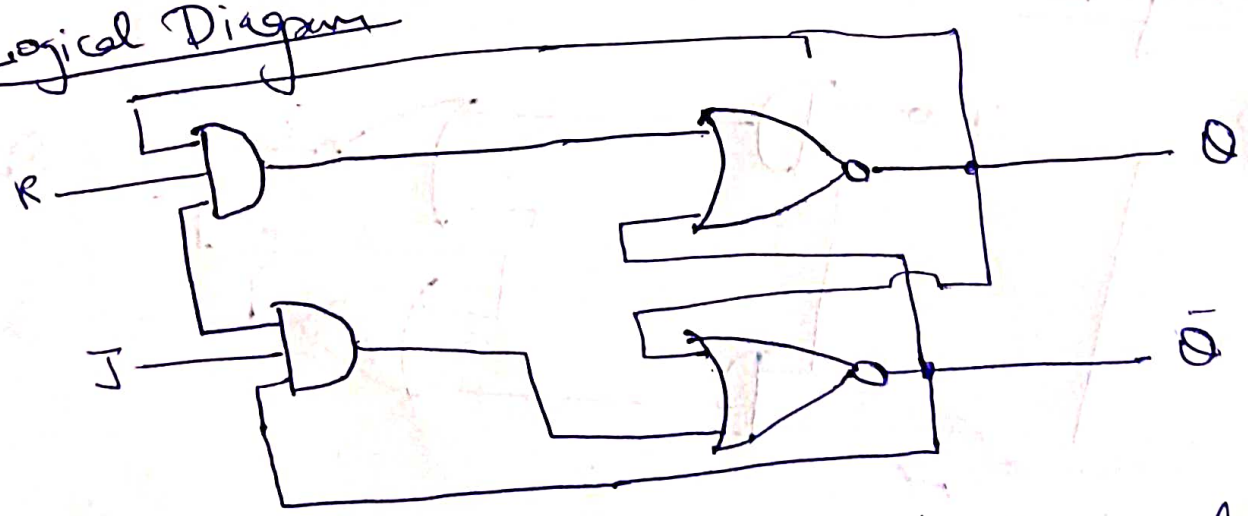
AIM: To construct and verify JK flip flop

Software Used: Logisim Software

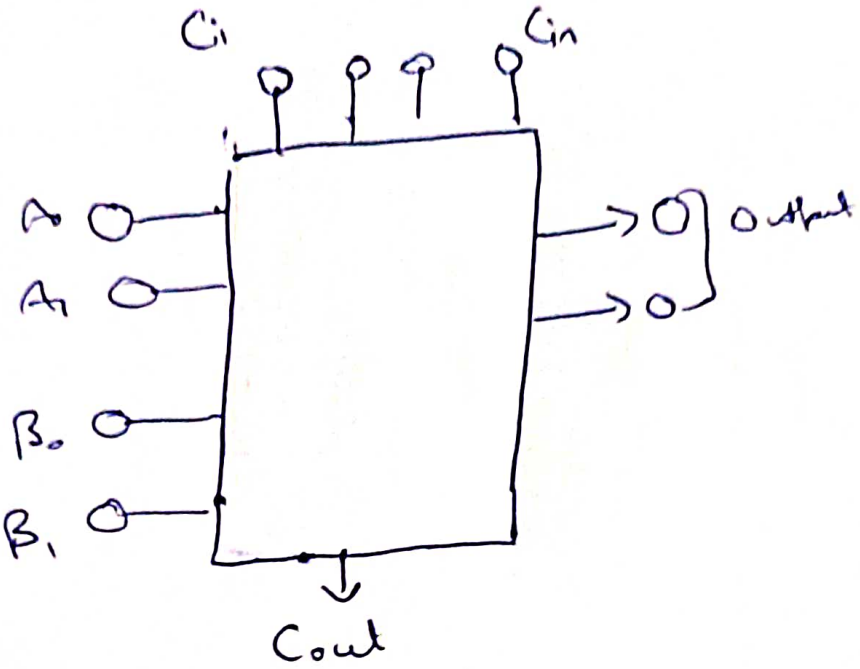
Truth Table:

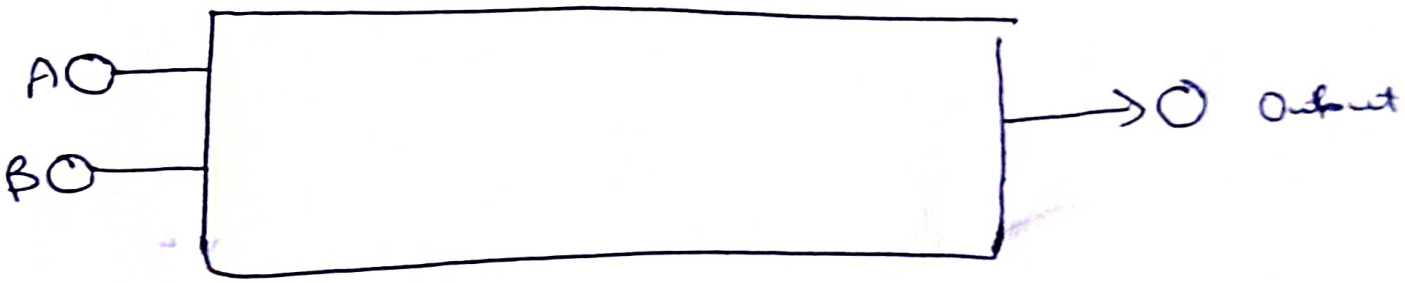
clock	J	K	Q	$\bar{Q}$
0	0	0	No change	
1	0	1	0	1
1	1	0	1	0
1	1	1	Toggle	

Logical Diagram



Conclusion This circuit is simulated in logisim and is verified by changing the values of the circuit.


Ci
Cin
A0
A1
B0
B1
Output
Cout


A
B
Output