EXPERIMENT 9

REALIZATION OF LOGIC GATES USING VERILOG

Aim:

1) To write the Verilog HDL code for the following logic gates using dataflow modelling and simulate it in ModelSim.
    a) NOT gate
    b) AND gate
    c) OR gate
    d) XOR gate
    e) NAND gate
2) To write the Verilog HDL code for the given Boolean function using dataflow modelling and simulate it in ModelSim.

$\mathbf{F = A'B'C + ABC' + A'BC' + ABC}$

CODE: *(ON RIGHT SIDE)*

DESIGN BLOCK:

<>

TEST BENCH:

<>

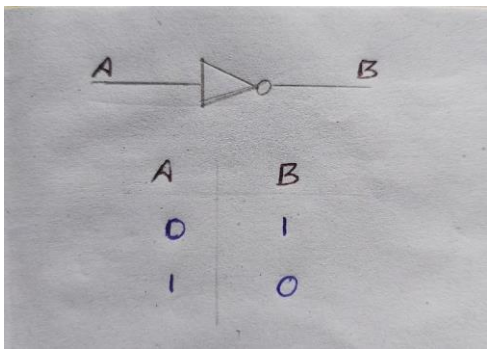
Result:

1) The Verilog HDL code for the given logic gates is simulated in ModelSim and the output is obtained and verified.
2) The Verilog HDL code for the given Boolean function is simulated in ModelSim and the output is obtained and verified.

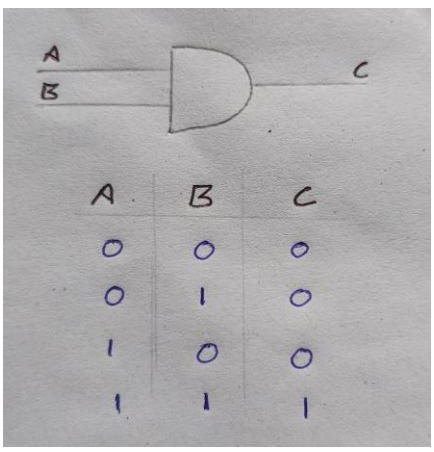
NB: Diagram, Truth table and Screenshots of output: (on left side)

NB: Change the letter notation according to your program

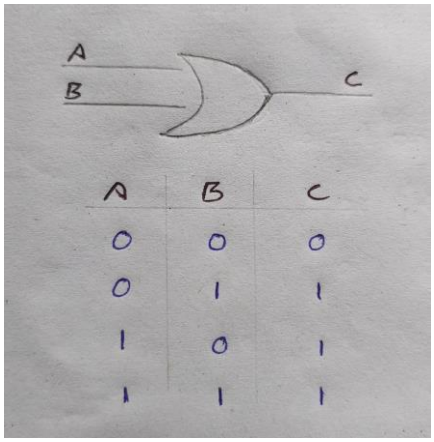
NOT gate:



AND gate

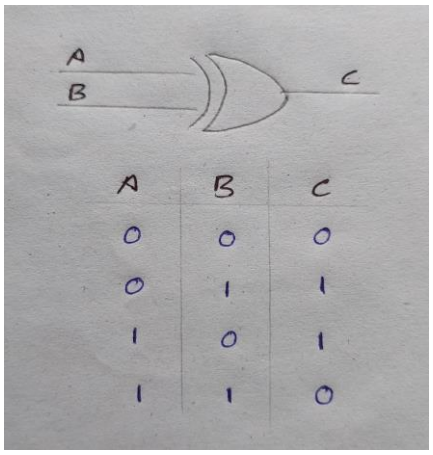


OR gate



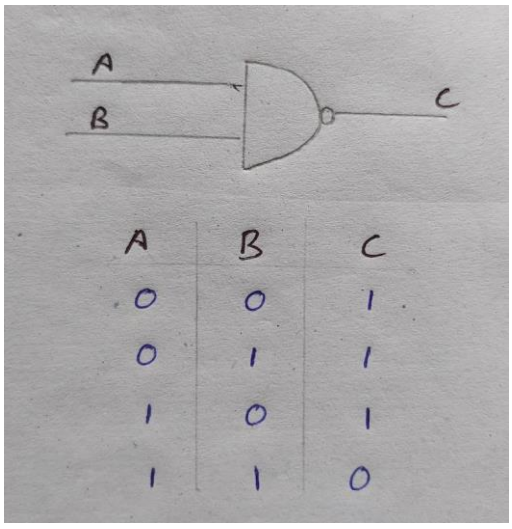
A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

XOR gate



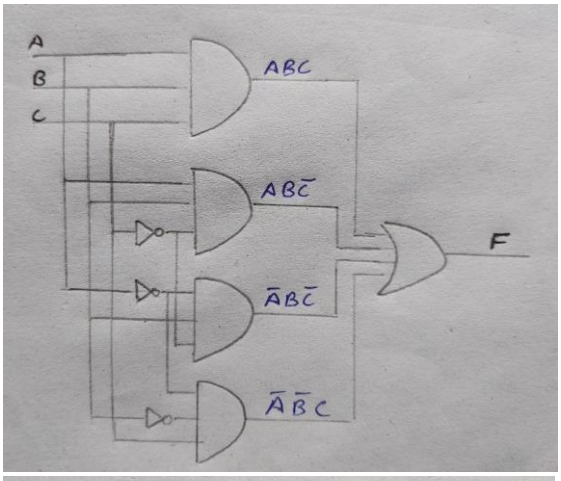
A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

NAND gate



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

F = A'B'C + ABC' + A'BC' + ABC



A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

EXPERIMENT 10

REALIZATION OF ADDER AND SUBTRACTOR CIRCUITS USING VERILOG

Aim:

1) Design Half adder, Full Adder, Half Subtractor, Full Subtractor circuits. Write the Verilog codes for the circuits and simulate them in ModelSim.
2) Write the Verilog code of Full Adder using Half Adder by using structural style of modelling and implement it in ModelSim.

CODE: *(ON RIGHT SIDE)*

DESIGN BLOCK:

<>

TEST BENCH:

<>

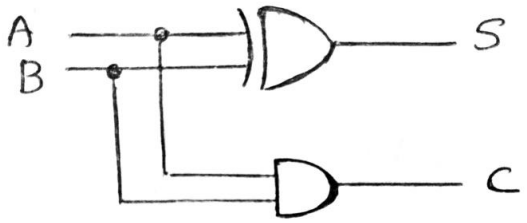
Result:

1) The Verilog HDL code for Half adder, Full Adder, Half Subtractor, Full Subtractor circuits are simulated in ModelSim and the output is obtained and verified.
2) The Verilog HDL code for Full Adder using Half Adder by using structural style of modelling is simulated in ModelSim and the output is obtained and verified.

NB: Diagram, Truth table and Screenshots of output: (on left side)

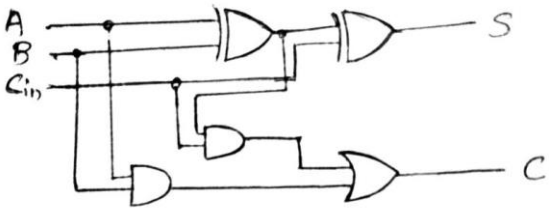
NB: Change the letter notation according to your program

HALF ADDER:



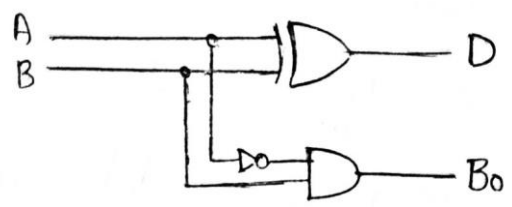
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

FULL ADDER



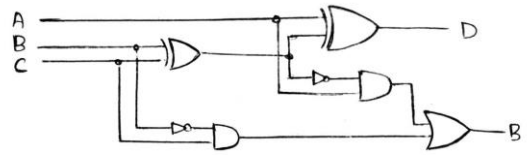
Cin	B	A	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

HALF SUBTRACTOR



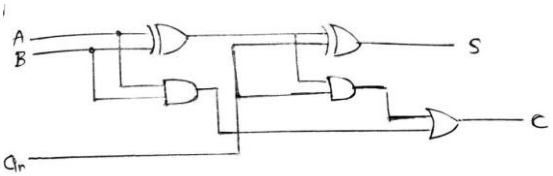
A	B	D	Bo
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

FULL SUBTRACTOR



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

FULL ADDER USING HALF ADDER BY USING STRUCTURAL STYLE OF MODELING



A	B	Cin	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

EXPERIMENT 11

REALIZATION OF 4:1 MULTIPLEXER AND 1:4 DEMULTIPLEXER CIRCUITS USING VERILOG

Aim:

1) Design a 4:1 Multiplexer circuit. Write the Verilog codes for the circuits and simulate them in ModelSim.
2) Design a 1:4 Demultiplexer circuit. Write the Verilog codes for the circuits and simulate them in ModelSim.

CODE: *(ON RIGHT SIDE)*

DESIGN BLOCK:

<>

TEST BENCH:

<>

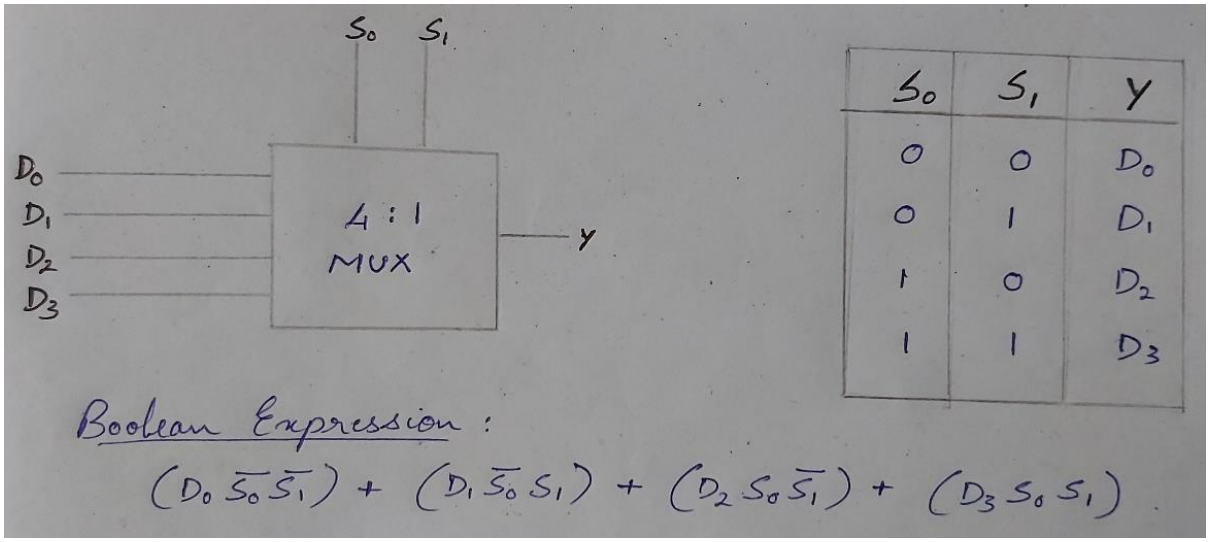
Result:

1) The Verilog 4:1 Multiplexer circuit is simulated in ModelSim and the output is obtained and verified.
2) The Verilog 1:4 Demultiplexer circuit is simulated in ModelSim and the output is obtained and verified.

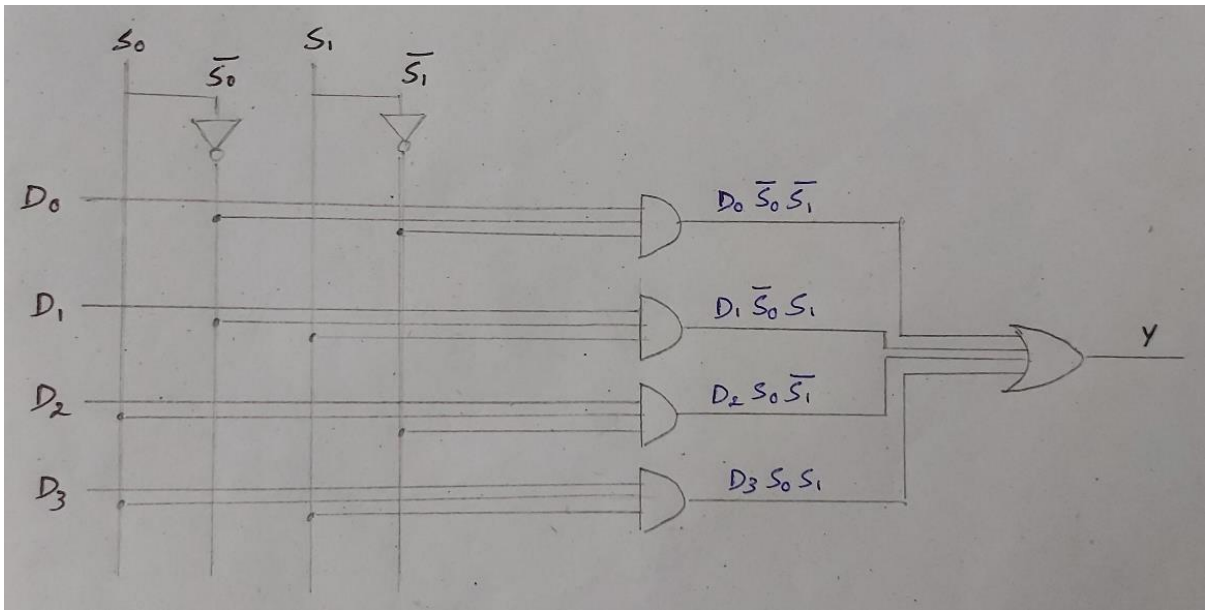
NB: Diagram, Truth table and Screenshots of output: (on left side)

NB: Change the letter notation according to your program

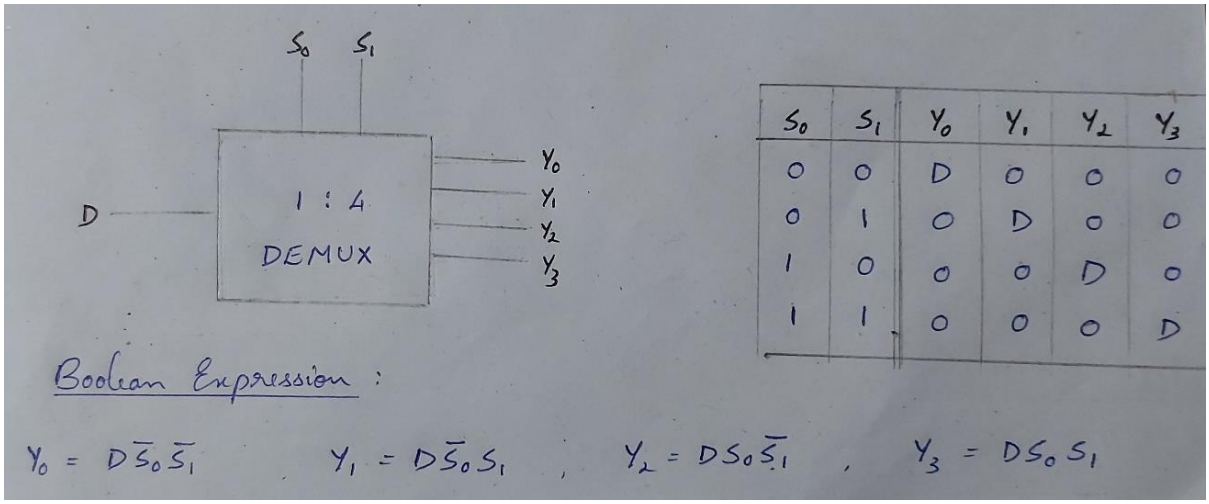
4:1 MULTIPLEXER:



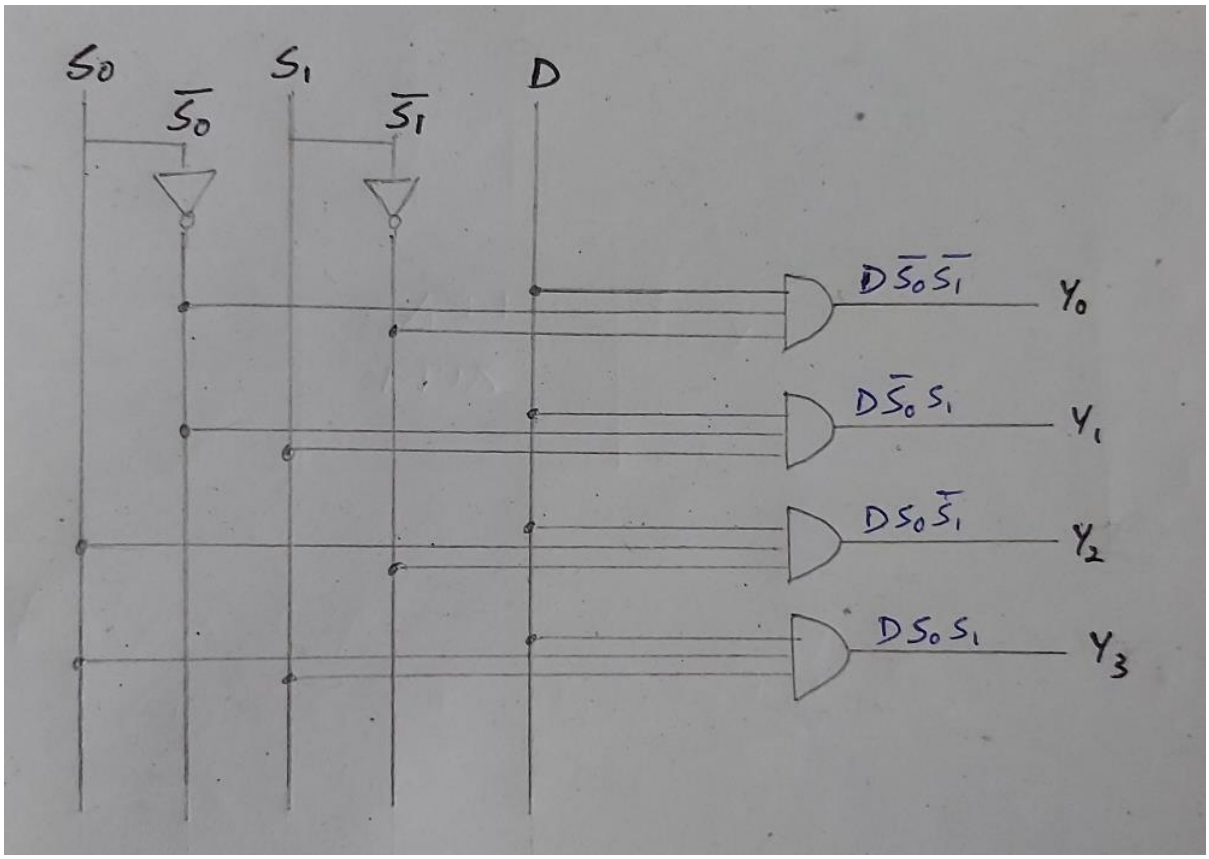
S_0	S_1	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3



1:4 DEMULTIPLEXER



S_0	S_1	Y_0	Y_1	Y_2	Y_3
0	0	D	0	0	0
0	1	0	D	0	0
1	0	0	0	D	0
1	1	0	0	0	D


S_0
$\overline{S_0}$
S_1
$\overline{S_1}$
D
$D\overline{S_0}\overline{S_1}$
Y_0
$D\overline{S_0}S_1$
Y_1
$DS_0\overline{S_1}$
Y_2
DS_0S_1
Y_3

EXPERIMENT 12

REALIZATION OF D, T, SR, JK FLIP FLOPS AND 4-BIT UP COUNTER USING STRUCTURAL STYLE OF MODELING USING VERILOG

Aim:

1) Design and implement the Verilog code of D, T, SR & JK flip flops and stimulate them in ModelSim.

2) Design and implement the Verilog code for 4-bit up counter using structural style of modelling and simulate it in ModelSim.

CODE: *(ON RIGHT SIDE)*

DESIGN BLOCK:

<>

TEST BENCH:

<>

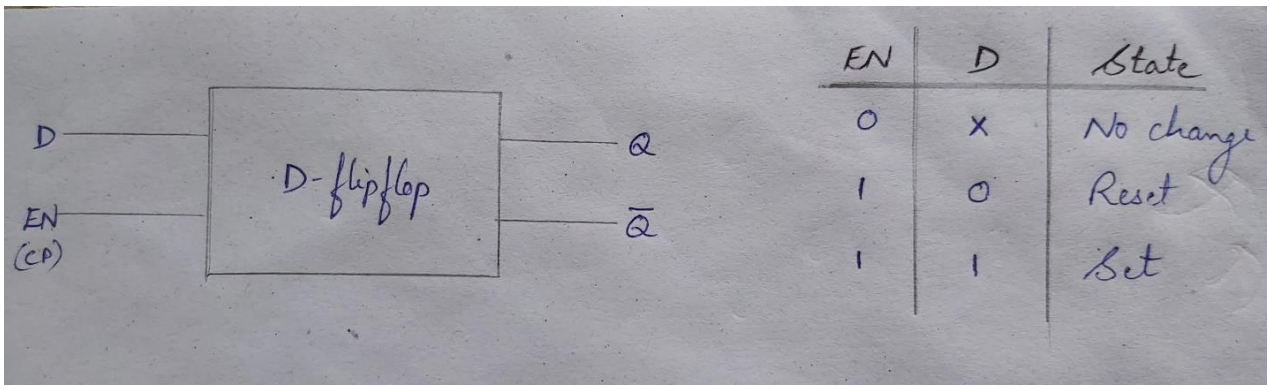
Result:

1) The Verilog HDL code for the given logic gates is simulated in ModelSim and the output is obtained and verified.
2) The Verilog HDL code for the given Boolean function is simulated in ModelSim and the output is obtained and verified.

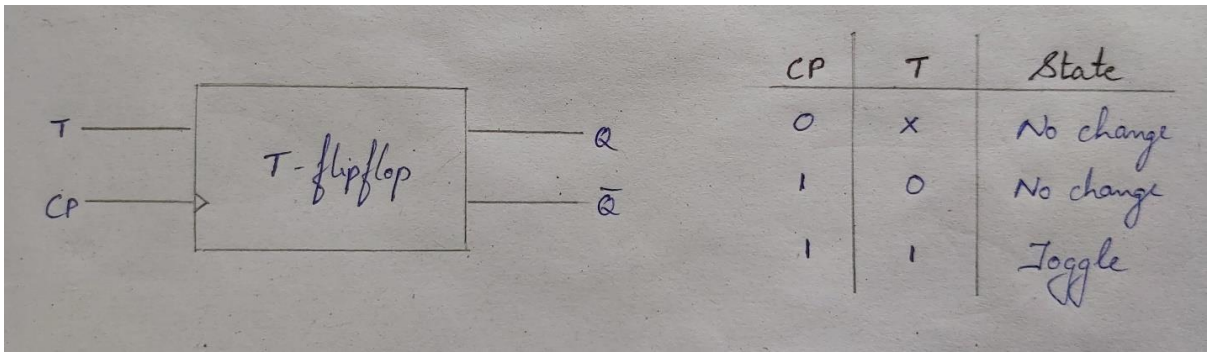
NB: Diagram, Truth table and Screenshots of output: (on left side)

NB: Change the letter notation according to your program

D FLIP FLOP:

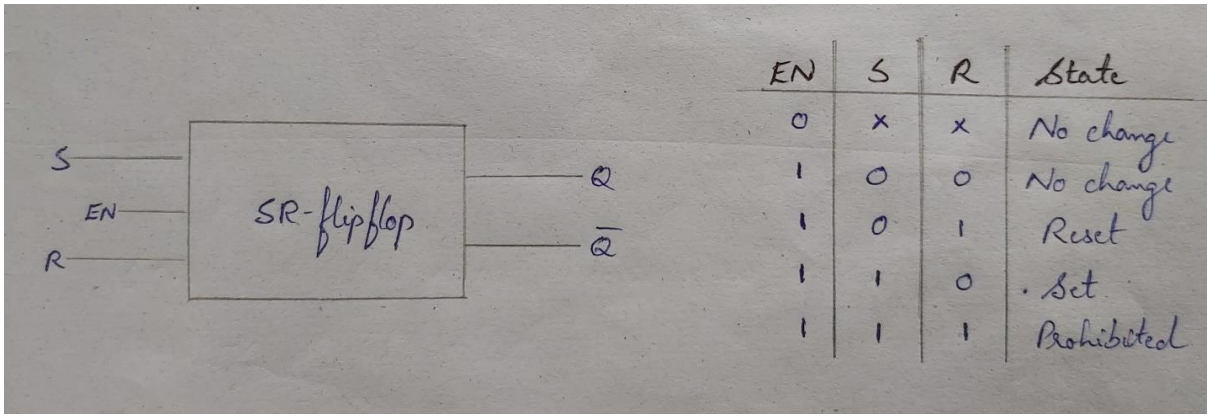


T FLIP FLOP



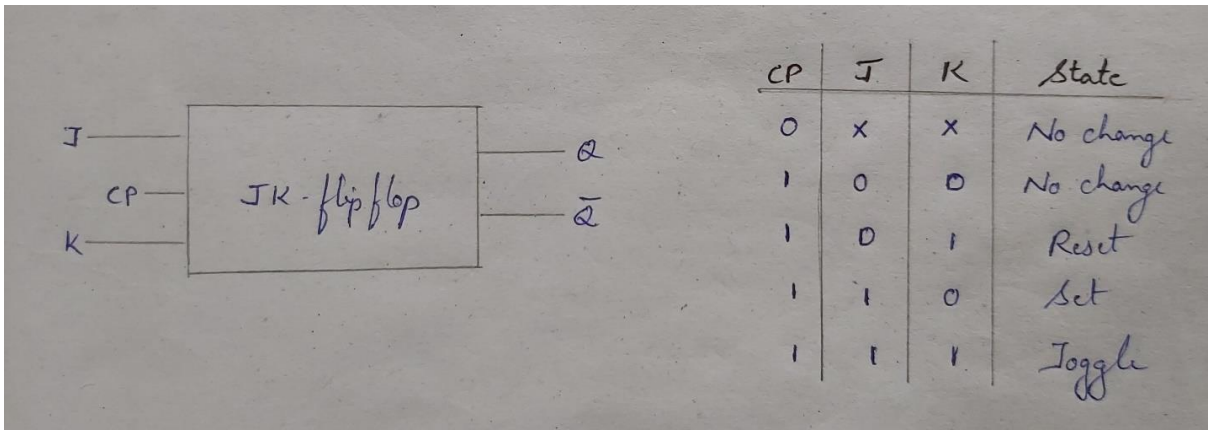
CP	T	State
0	x	No change
1	0	No change
1	1	Toggle

SR FLIP FLOP



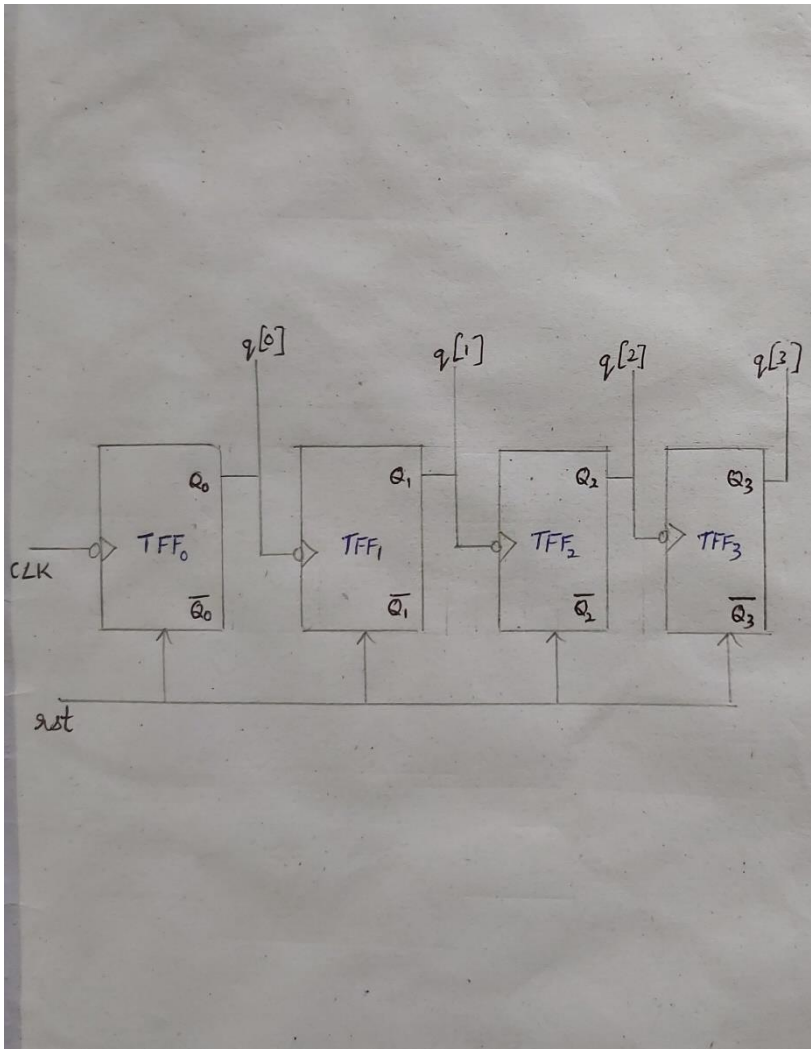
EN	S	R	State
0	x	x	No change
1	0	0	No change
1	0	1	Reset
1	1	0	Set
1	1	1	Prohibited

JK FLIP FLOP



CP	J	K	State
0	x	x	No change
1	0	0	No change
1	0	1	Reset
1	1	0	Set
1	1	1	Toggle

4-BIT UP COUNTER USING STRUCTURAL STYLE OF MODELING



CLK	Q_3	Q_2	Q_1	Q_0
	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0