(86)

DIGITAL LOGIC DESIGN LAB (EET1211)

LAB VII: CONSTRUCT, TEST AND INVESTIGATE THE OPERATION OF VARIOUS FLIP-FLOP CIRCUITS

Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar

Branch: CSIT		Section: E	
S. No.	**Name**	**Registration No.**	**Signature**
1.	SHIPRA TANVI	2141020004	Shipra Tanvi
2.	SASHIKANT TRIPATHY	2141020030	
3.	DEBAPRIYA SAHOO	2140194947	
4.	PRAYAS KU. SETHI	2140194945	

Marks: _____/10

Remarks:

14/01/23

Teacher's Signature

(07)

I. OBJECTIVE

1. Design and test of SR latch using universal logic gates.

2. Design and test of SR flip-flop using universal logic gates.

3. Design and test of JK flip-flop using universal logic gates.

4. Investigation of the logic behavior of various flip flop ICs:
   7474 D flip-flop IC
   7476 JK flip-flop IC

5. Realize the function of T flip flop using the JK flip-flop IC.

II. PRE-LAB

For Obj. 1:

a. Obtain the characteristic table for the SR latch.
b. Draw the logic diagram for SR latch using NAND gates.
c. Write HDL code for SR latch.

For Obj. 2:

a. Obtain the characteristic table for the SR flip flop.
b. Draw the logic diagram for SR flip flop using two cross coupled NAND gates.
c. Write HDL code for SR flip flop.

For Obj. 3:

a. Obtain the characteristic table for the JK flip flop.
b. Draw the logic diagram for JK flip flop using two cross coupled NAND gates.
c. Write HDL code for JK flip flop.

For Obj. 4:

a. Write the characteristic table for D flip-flop.
b. Write the characteristic table of JK flip-flop.
c. Write the characteristic table of T flip-flop.

For Obj. 5:

a. Draw the logic diagram for T Flip flop using JK flip-flop.
b. Write HDL code for T flip flop.

endmodule.

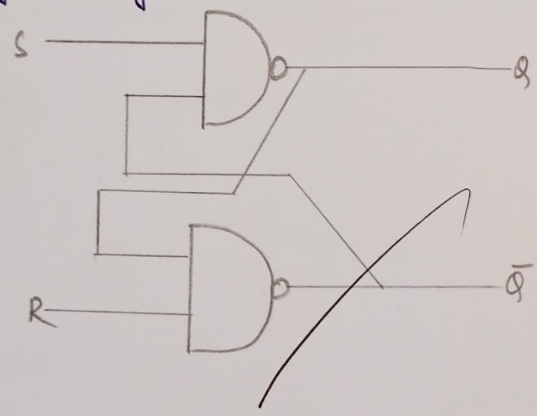
II PRE LAB

OBJECTIVE 1

a) Characteristic Table for the SR Latch

S	R	Q	$\bar{Q}$	Comment
1	0	0	1	Reset
1	1	0	1	No change
0	1	1	0	Set
1	1	1	0	No change
0	0	1	1	Invalid

b) Logic Diagram.



c) HDL Code

```
module sr_latch(
  input S, R,
  output Q, Qn
  );
wire Q_int, Qn_int;
assign #1 Q_int = ~(S & Qn_int);
assign #1 Qn_int = ~(R & Q_int);
 assign Q = Q_int;
 assign Qn = Qn_int;
```

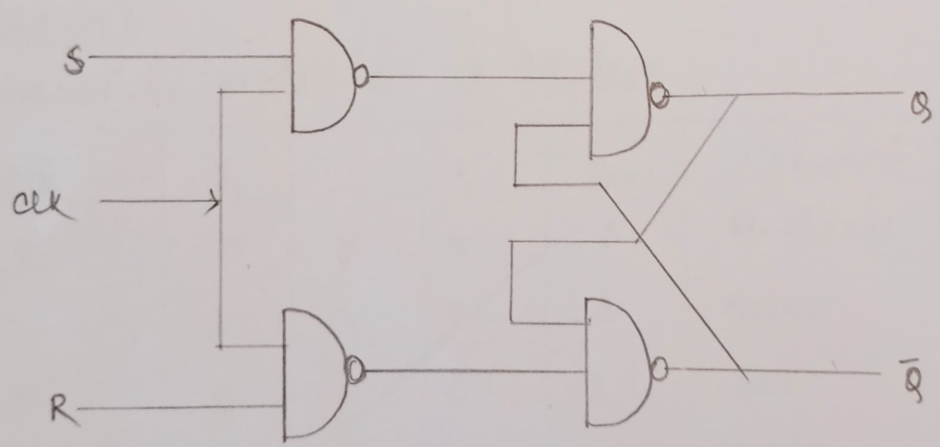
endmodule.

OBJECTIVE2.

a) Characteristic Table for the SR flipflop.

CLK	Q_t	S	R	Q_{t+1}	Comment
↑	0	0	0	0	No change
↑	0	0	1	0	Reset
↑	0	1	0	1	Set
↑	0	1	1	X	Race
↑	1	0	0	1	No change
↑	1	0	1	0	Reset
↑	1	1	0	1	Set
↑	1	1	1	X	Race.

b) Logic Diagram.



c) HDL code

```
module SR_FF (s, r, clk, reset, q, q_bar);
  input s, r, clk, reset;
  output q, q_bar;
```

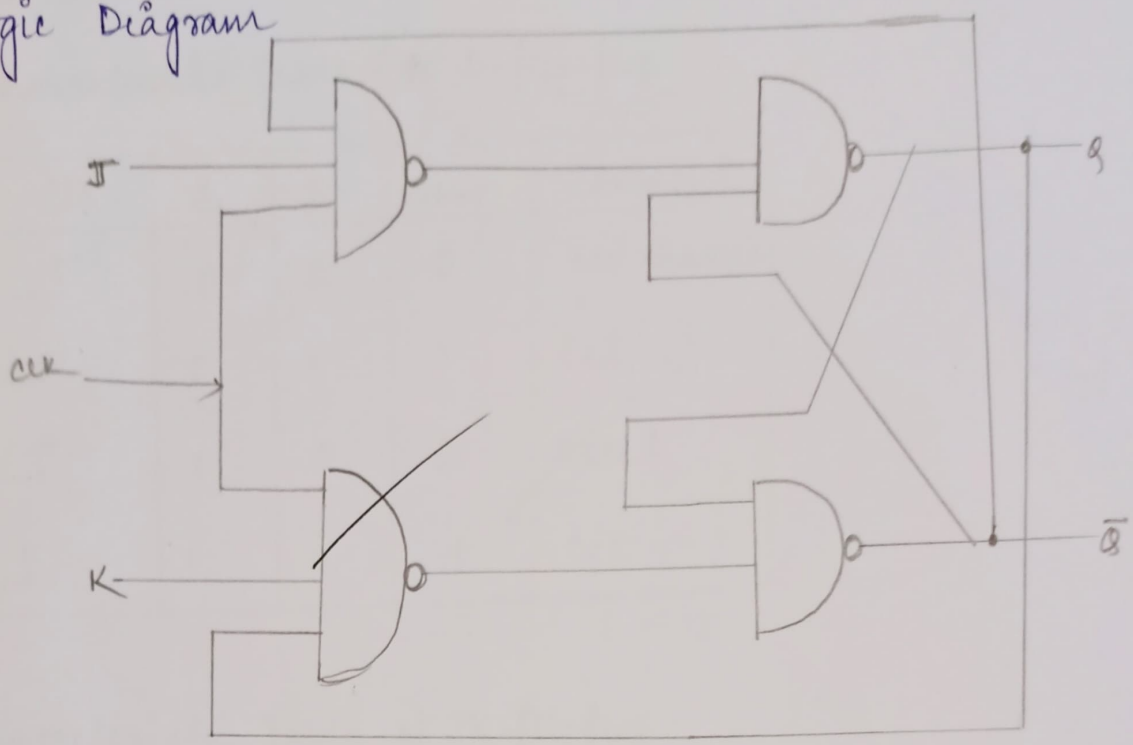
```
wire s, r, clk;
reg q, q_bar
always @(posedge clk) begin
  if(reset) begin
    q = 1'b0;
  q_bar = 1'b1;
  end else begin
  case({s,r})
{1'b0, 1'b0}: begin q = q; q_bar = q_bar; end
{1'b0, 1'b1}: begin q = 1'b0; q_bar = 1'b1; end
{1'b1, 1'b0}: begin q = 1'b1; q_bar = 1'b0; end
{1'b1, 1'b1}: begin q = 1'bx; q_bar = 1'bx; end

  endcase
  end
  end
  endmodule
```

OBJECTIVE 3

a) Characteristic table for JK flipflop

clk	Q_t	J	K	Q_{t+1}	Comment
↑	0	0	0	0	No change
↑	0	0	1	0	Reset
↑	0	1	0	1	Set
↑	0	1	1	1	Toggle
↑	1	0	0	1	No change
↑	1	0	1	0	Reset
↑	1	1	0	1	Set
↑	1	1	1	0	Toggle

b) Logic Diagram



c) HDL Code.

```
module JK_FF (input j, K, clk, reset, output Q);
    reg Q;
    always @ (posedge(clk))
    begin
    if (reset) begin
      Q = 1'b0
      end
      else
      begin
    case ({j,K})
    2'b00 : Q <= Q;
    2'b01 : Q <= 0;
    2'b10 : Q <= 1;
    2'b11 : Q = ~Q;
    endcase
    end
    end
endmodule.
```

OBJECTIVE 4

a) Characteristic table for D-flip-flop.

Clk	Q_t	D	Q_{t+1}	Comment
↑	0	0	0	No change
↑	0	1	1	Set
↑	1	0	0	Reset
↑	1	1	1	Set

b) Characteristic Table of JK flipflop.

Clk	J	K	Q_n	Q_{n+1}	Comment
↑	0	0	0	0	Memory
↑	0	0	1	1	Memory
↑	0	1	0	0	Reset
↑	0	1	1	0	Reset
↑	1	0	0	1	Set
↑	1	0	1	1	Set
↑	1	1	0	1	Toggle
↑	1	1	1	0	Toggle

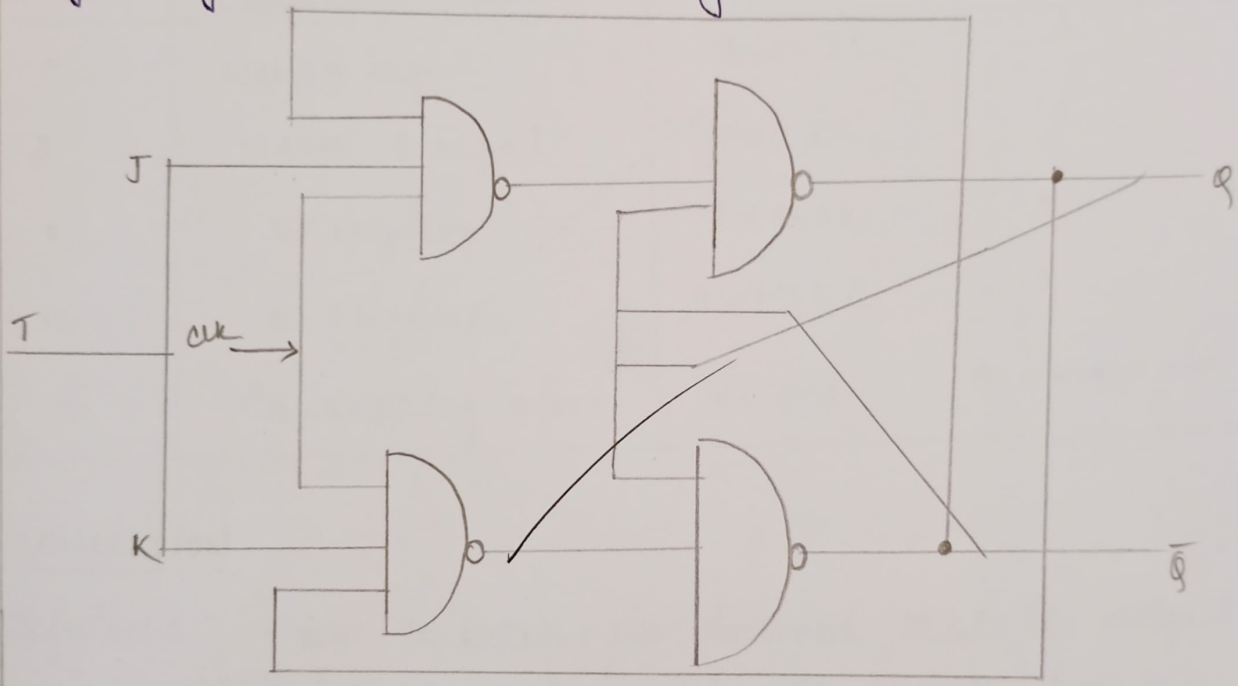
c) Characteristic Table of T flip-flop.

(a3)

Clk	T	Qn	Qn+1	Comment
↑	0	0	0	Qn
↑	0	1	1	Qn
↑	1	0	1	$\bar{Q}n$
↑	1	1	0	$\bar{Q}n$

OBJECTIVE 5

a) Logic diagram for T flipflop using JK flipflop



b) HDL Code for T flip-flop.

```
module T_ff(input T, clk, reset, output Q);
  reg Q;
always @(T, posedge(clk))
  begin
if (reset) begin
  Q = 1'b 0;
    end
```

```
else
begin
Q<= T^Q;
 end
 end
endmodule
```

III LAB

Components Required

S.No.	Components Required	Specification	Quantity
1	NAND 2-input	7400 IC	1.
2	NAND 3-input	7410 IC	1.
3.	JK Flipflop	74761C	1.
4.	D-Flipflop	7474 IC	1.
5.	Connecting wires	23SW4	As required

CONCLUSION

Objective 1: From SR Latch, we observed that its output sets for 10, resets for 01, no change for 00 and invalid ~~f~~ in case of 11 for respective input of JK.

Objective 2: In case of SR flipflop, we took positive edge triggered. Our next state is set for 10 input reset for 01 input, no change for 00 input and race for 11 input in J and K respectively for each case.

Objective 3: In case of JK-flipflop, we took positive edge triggered. Our next step shows set in case of 10 input, reset for 01 input, no change for 00 input and toggle in case of 11 input into J & K input respectively.

Objective 4: In designing T-flipflop, we observed, we can short J & K inputs to give one input & hence JK-flipflop is converted to T-flipflop.

IV POST LAB

Q1. Differentiate between a latch and a flip flop.

Ans:

LATCHES	FLIP-FLOPS.
→ It is level triggered device.	→ It is edge triggered device
→ They don't have clock signals.	→ They have clock signals
→ There is no such classification in latches.	→ They are classified into synchronous and asynchronous.
→ Less power (less gates)	→ More power (more gates).

Q2. If both inputs of a SR NOR Latch are low, what will happen to the output?.

Ans- The output will not be changed.

Q3. If both inputs of a SR NAND Latch are low, what will happen to the output?

Ans- The output will become unpredictable.

Q4. Which of the following describes the operation of a positive edge-triggered D-type flip-flop?

Ans- The output will follow the input on the leading edge of the clock.

J. 14/01/23