BITS-Pilani Dubai Campus

I Sem 2021-22

Digital Design Laboratory / ECE/INSTR/CS F215

Submission Report

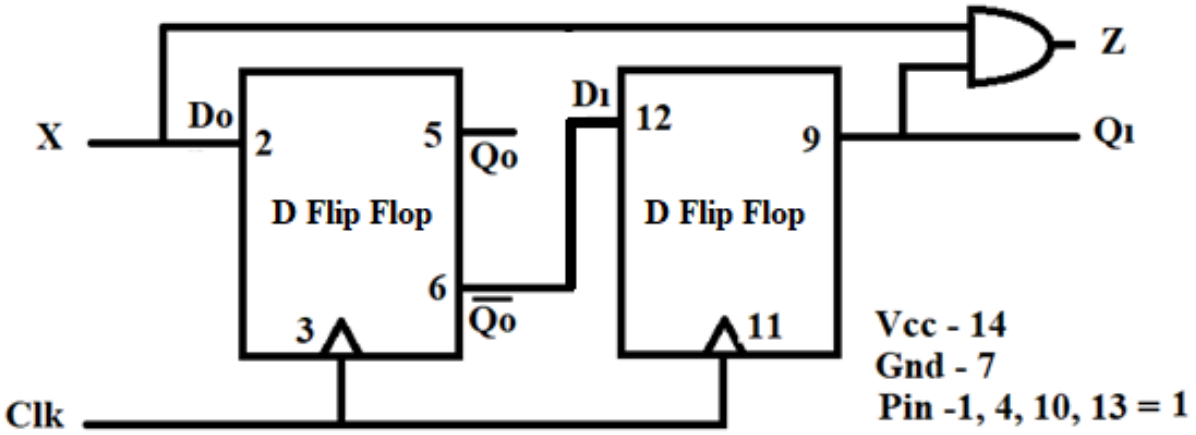
Experiment No.- 8 (Finite State Machines)

Name ……ARSHDEEP SINGH..……..… **ID Number…2020A7PS0144U….**

Hardware runs

Run 1:

Diagram



Q. Complete the state table by observing the outputs on LED panel.

State Table

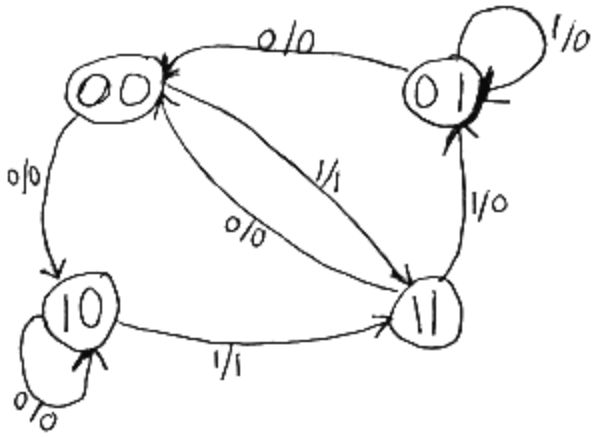
Present state			Next state		Output
Q1(t)	Q0(t)	X	Q1(t+1)	Q0(t+1)	Z
0	0	0	1	0	0
0	0	1	1	1	1
0	1	0	0	0	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	1	1	1

Experiment No.- 8 (Finite State Machines)

Name ……..…….…. ID Number....…….

1	1	0	0	0	0
1	1	1	0	1	0

Q. Draw the State diagram from above state table?
Ans.



Q. Which types of Machine it is?
Ans. Finite State Machine - Mealy Model

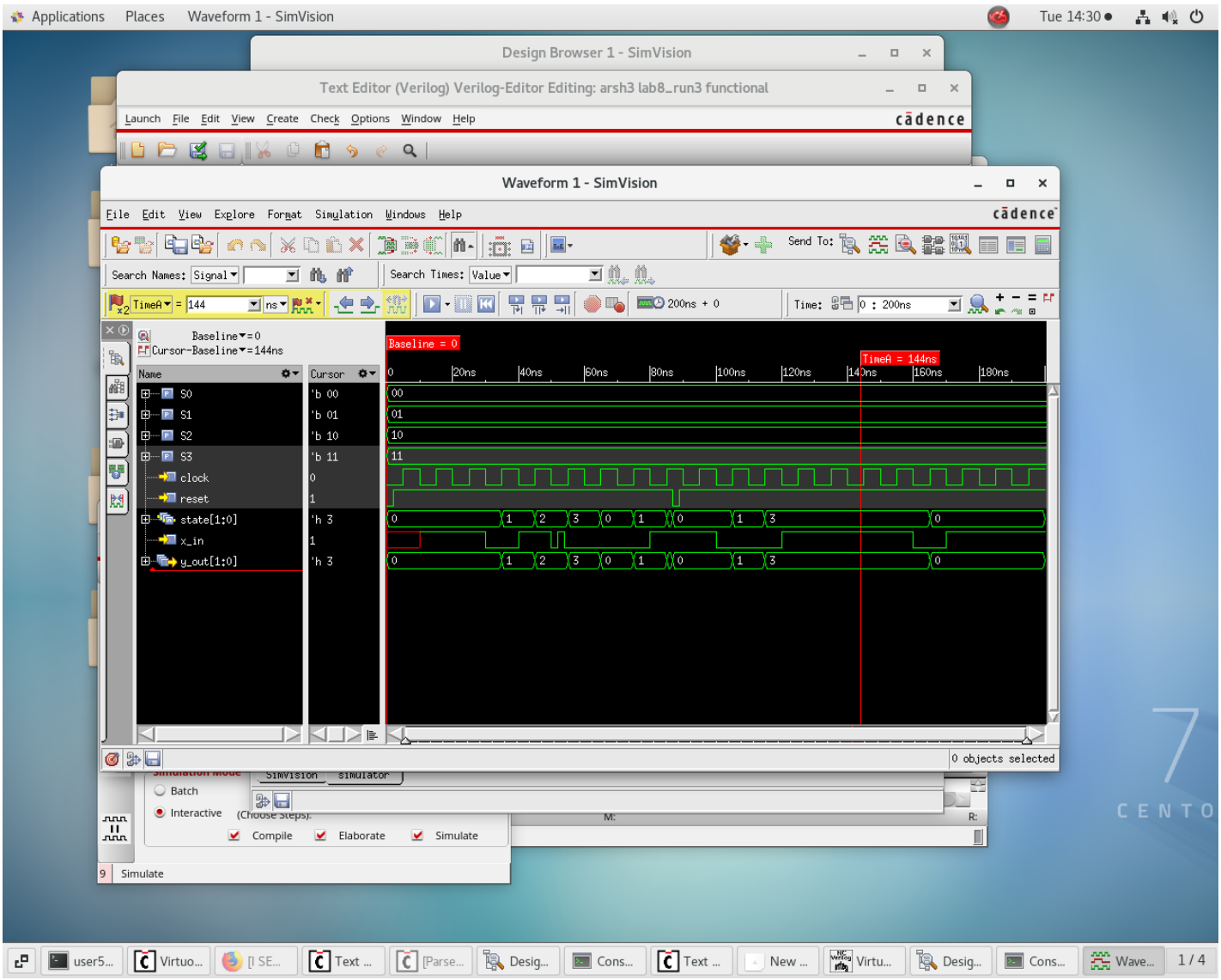
Software runs

Run 2: Mealy Machine: Zero detector after series of 1

Q: Paste the Image of your **Simvision** window where you get the waveforms for the above code.
A:

Experiment No.- 8 (Finite State Machines)

Name ……..…….. ID Number.....……



Q. Complete the state table below from the waveform observed.

Experiment No.- 8 (Finite State Machines)

Name ……..……..… ID Number....…….

State Table

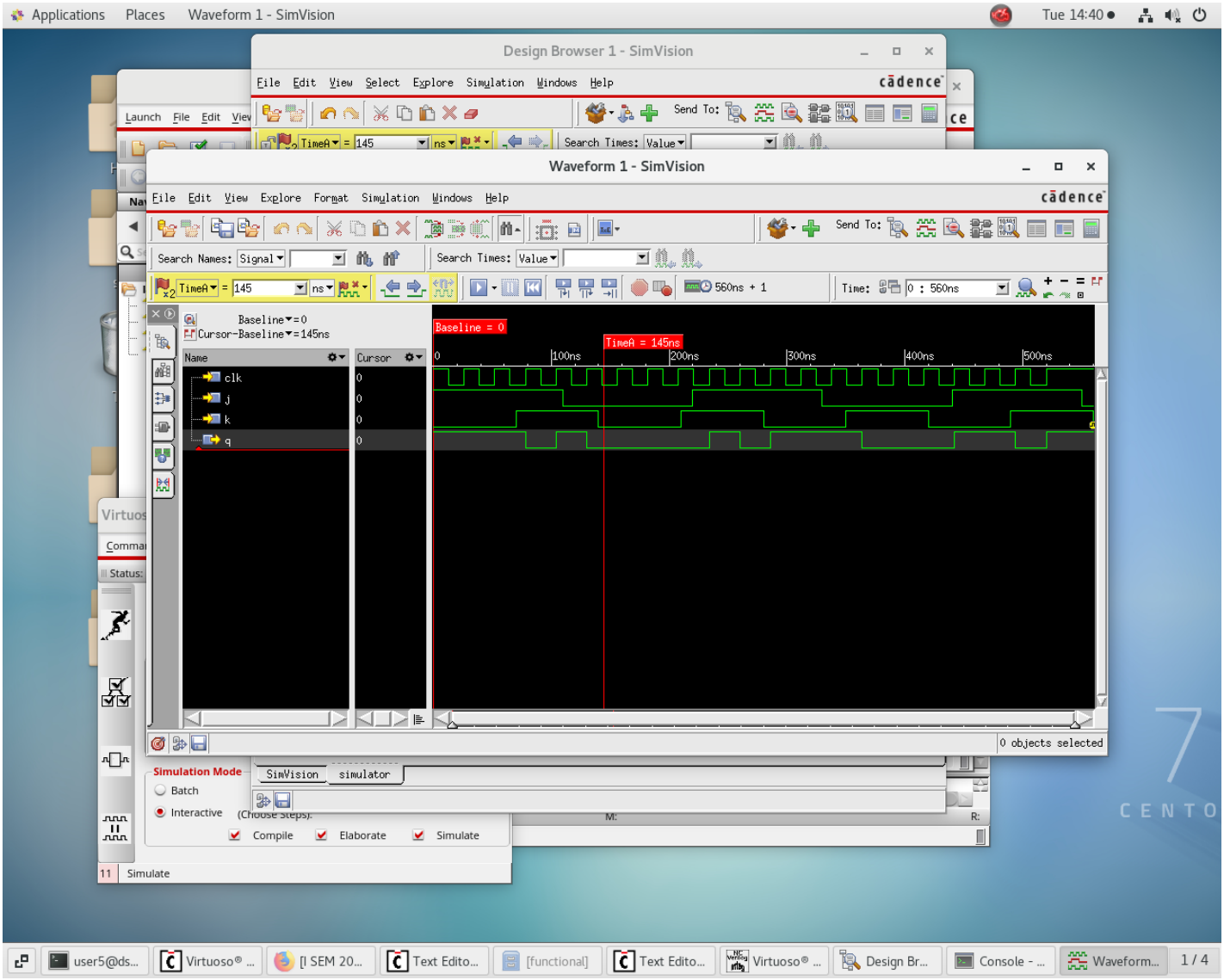
Present state		Next state	Output
state [1:0]	**t_x_in**	**next_state [1:0]**	**t_y_out**
S0	0	S0	0
S0	1	S1	0
S1	0	S0	1
S1	1	S3	0
S2	0	S0	1
S2	1	S2	0
S3	0	S0	1
S3	1	S2	0

Run 3: Moore Machine: Zero detector after series of 1

Q: Paste the Image of your **Simvision** window where you get the waveforms for the above code.
A:

Experiment No.- 8 (Finite State Machines)

Name ……..…… ID Number.…….



Q. Complete the state table below from the waveform observed.

Experiment No.- 8 (Finite State Machines)

Name ……..……..…. ID Number....…….

State Table

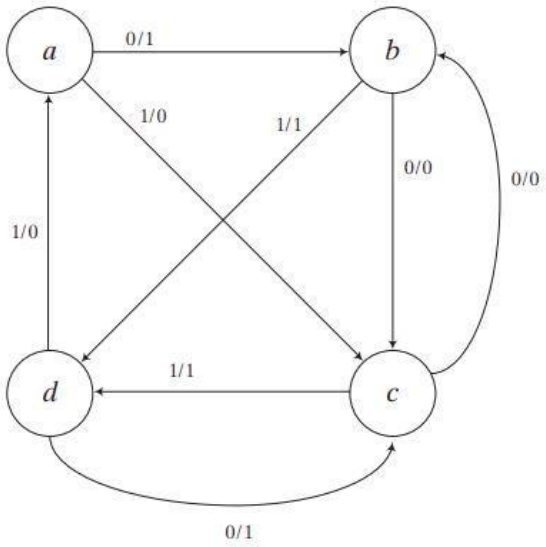
Present state		Next state	Output
state [1:0]	**t_x_in**	**next_state [1:0]**	**t_y_out**
S0 = 00	0	01	01
S0 = 00	1	00	00
S1 = 01	0	11	11
S1 = 01	1	10	10
S2 = 10	0	11	11
S2 = 10	1	10	10
S3 = 11	0	00	00
S3 = 11	1	11	11

Assignment All assignments are to be submitted strictly before start of next lab session through online only. Late assignments will not be entertained and will be awarded '0' marks.

1. Write a Verilog model of the Mealy FSM described by below the state diagram. Develop a test bench and demonstrate that the machine state transitions and output correspond to its state diagram.

Experiment No.- 8 (Finite State Machines)

Name ……..…….…. ID Number....……



Ans: Link1: https://www.edaplayground.com/x/MVf5

2. Draw the state diagram of the machine described by the Verilog model given below.

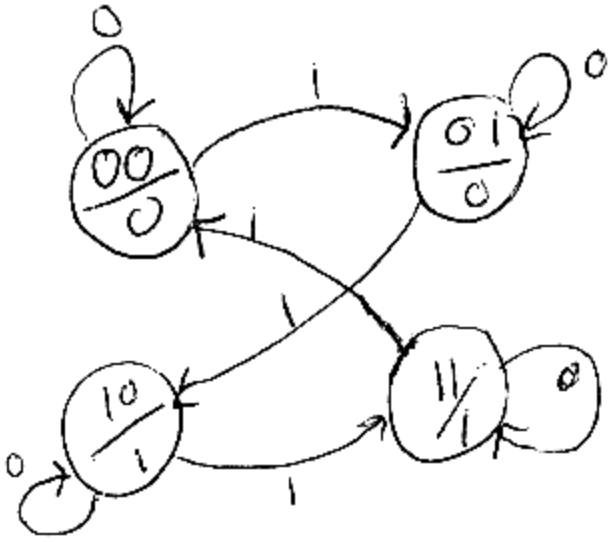
```
module Prob_1 ( output reg y_out, input x_in, clk, reset);
parameter s0 = 2'b00, s1 = 2'b01, s2 = 2'b10, s3 = 2'b11;
reg [1:0] state, next_state;
always @ ( posedge clk, negedge reset) begin
if (reset == 1'b0) state <= s0;
else
 state <= next_state;
always @(state, x_in) begin
y_out = 0;
next_state = s0;
case (state)
s0: if x_in = 1 begin y_out = 0; if (x_in) next_state = s1; else next_state = s0; end
s1: if x_in = 1 begin y_out = 0; if (x_in) next_state = s2; else next_state = s1; end
s2: if x_in = 1 begin y_out = 1; if (x_in) next_state = s3; else next_state = s2; end
s3: if x_in = 1 begin y_out = 1; if (x_in) next_state = s0; else next_state = s3; end
default: next_state = s0;
endcase
end
endmodule
```

Ans: Link2: https://www.edaplayground.com/x/j6pW

State Diagram:

Experiment No.- 8 (Finite State Machines)

Name ……..…….…. ID Number….…….

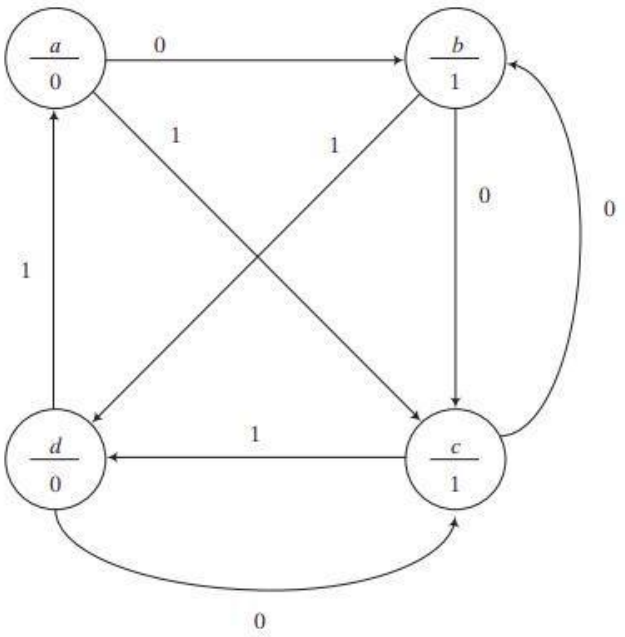


Self-Practice and self-evaluation

1. Write a Verilog model of the Moore FSM described by the below state diagram. Develop a test bench and demonstrate that the machine's state transitions and output correspond to its state diagram.

Experiment No.- 8 (Finite State Machines)

Name ……..…….…. ID Number….……



2. Draw the state diagram of the machine described by the Verilog model given below.

```
module Prob_2 ( output reg y_out, input x_in, clk, reset);
parameter s0 = 2'b00, s1 = 2'b01, s2 = 2'b10, s3 = 2'b11;
reg [1:0] state, next_state;
always @ ( posedge clk, negedge reset) begin
if (reset == 1'b0) state <= s0;
else
state <= next_state;
always @(state, x_in) begin
y_out = 0;
next_state = s0;
case (state)
s0: if x_in = 1 begin y_out = 0; if (x_in) next_state = s1; else next_state = s0; end
s1: if x_in = 1 begin y_out = 0; if (x_in) next_state = s2; else next_state = s1; end
s2: if x_in = 1 if (x_in) begin next_state = s3; y_out = 0; else begin next_state = s2; y_out = 1; end
s3: if x_in = 1 begin y_out = 1; if (x_in) next_state = s0; else next_state = s3; end
default: next_state = s0;
endcase
end
endmodule
```