## IMPLEMENT A TRAFFIC LIGHT CONTROLLER USING



**FSM LAB # 12**

**Spring 2024**

## CSE-308L

### Digital System Design Lab

Submitted by: **Naveed Ahmad** Registration No.: **22PWCSE2165**

Class Section: **B**

“I affirm that I have completed this work with integrity”

Student Signature:

Submitted to:

### Engr. Shah Zada Fahim Jan

Sunday, June 1, 2025

Department of Computer Systems Engineering University of Engineering and Technology, Peshawar

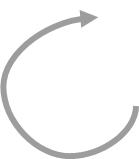
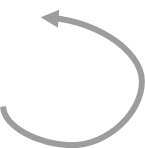
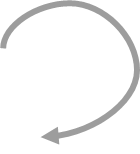
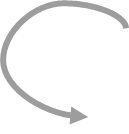
**Implement A Traffic Light Controller Using FSM**

# Objectives:

* Continue the introduction to FSMs
* Mealy machine

# Lab Tasks:

**Task 1**:



HG\_FR

V = 1

HR\_FY

H=001 F=100

H=100 F=010

D = 3 sec

D = 3 sec

HY\_FR

HR\_FG

H=010 F=100

D = 10 sec

H=100 F=001

#### V = 0

D < 3 sec

D < 3sec

D < 10 sec

***Verilog Code:***

module Traffic\_lights(clk,rst,v,out\_farm,out\_highway); input clk,rst,v;

output reg [2:0] out\_farm, out\_highway; reg [1:0] PS,NS;

parameter [1:0] FR\_HG = 0, FY\_HR = 1, FG\_HR = 2, FR\_HY = 3;

parameter [2:0] red = 100, yellow = 010, green = 001; wire out\_clk;

clk\_divider d1(clk,out\_clk,rst); always @(posedge out\_clk)

if(rst==0) begin

PS = FR\_HG;

end else

#### PS = NS;

always @(PS or v or rst) case(PS)

#### FR\_HG:

begin

end

NS = v?FY\_HR: FR\_HG;

out\_highway = v?red:green; out\_farm = v?yellow:red;

#### FY\_HR:

begin

end

#### NS = FG\_HR;

out\_highway = red; out\_farm = green;

#### FG\_HR:

begin

end

#### NS = FR\_HY;

out\_highway = yellow; out\_farm = red;

#### FR\_HY:

begin

endmodule

end endcase

#### NS = FR\_HG;

out\_highway = green; out\_farm = red;

Clock Divider Code:

module clk\_divider(input in\_clk,output reg out\_clk, input rst); reg [100:0] count;

always @(posedge in\_clk) if(rst==0)

begin

end else begin

out\_clk = 0;

count = 0;

count = count+1; if(count==3\*100000000) begin

endmodule

end

end

out\_clk = ~out\_clk; count = 0;

***UCF File:***

UCF file:

net "rst" LOC =K18 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST

| PULLUP;

net "v" LOC =F17 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST | PULLUP;

net "clk" LOC =V10 | IOSTANDARD = LVCMOS33 | period = 100MHz;

net "out\_highway[0]" LOC = N16 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST;

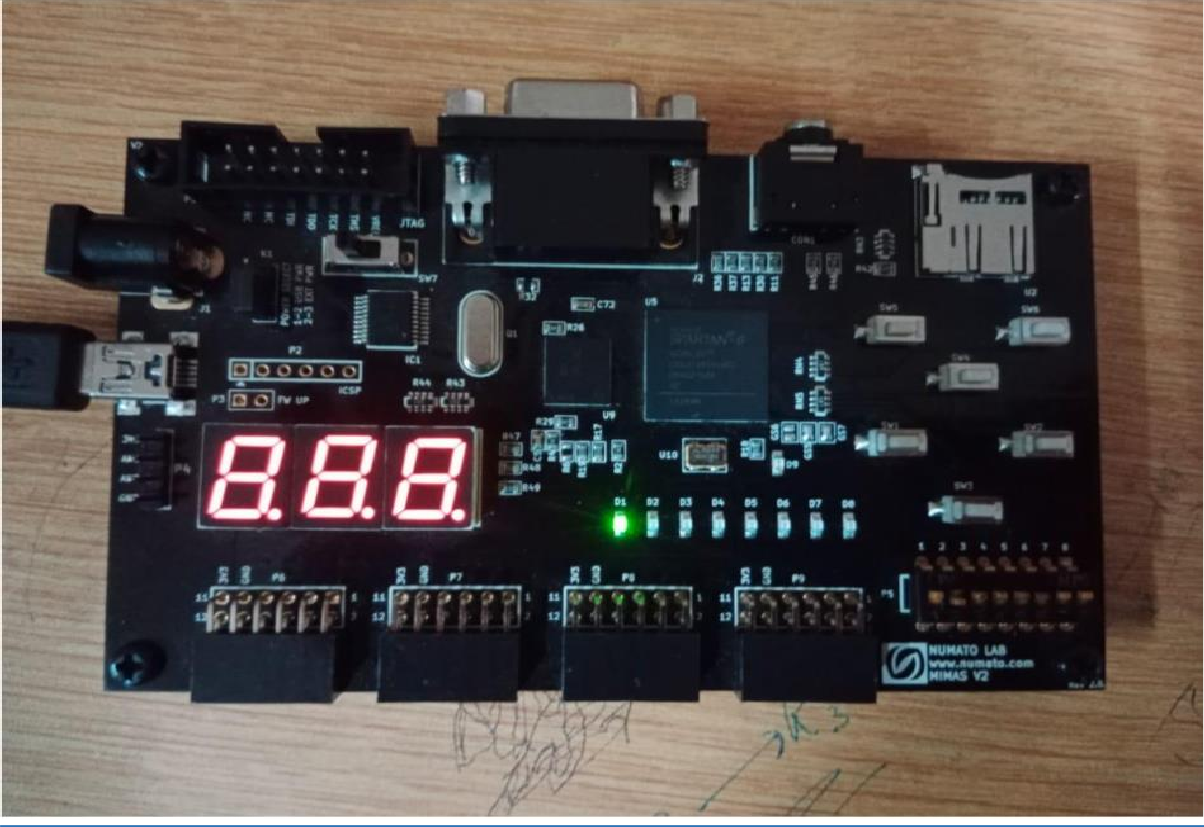
net "out\_highway[1]" LOC = U17 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST;

net "out\_highway[2]" LOC = U18 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST;

net "out\_farm[0]" LOC = P15 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST;

net "out\_farm[1]" LOC = P16 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST;

net "out\_farm[2]" LOC = N15 | IOSTANDARD = LVCMOS33 | DRIVE = 8 | SLEW = FAST;

***Output:***

# References:

* To view my lab codes, please refer to the [**DCSE**](https://github.com/aimalexe/DCSE/)repository on [**my GitHub Account**](https://github.com/aimalexe/): [https://github.com/aimalexe/DCSE/tree/main/semester\_6\_(spring-](https://github.com/aimalexe/DCSE/tree/main/semester_6_(spring-24)/digital_system_design_lab/lab_reports) [24)/digital\_system\_design\_lab/lab\_reports](https://github.com/aimalexe/DCSE/tree/main/semester_6_(spring-24)/digital_system_design_lab/lab_reports)
* And so on…
* ​

# Conclusion:

In summary, our Digital Systems Design lab has been a success, enabling us to achieve our objectives through hands-on tasks and theoretical exploration. This experience has enhanced my understanding of fundamental concepts in Verilog and their practical applications. I am excited to apply this knowledge in various contexts moving forward.

The End.