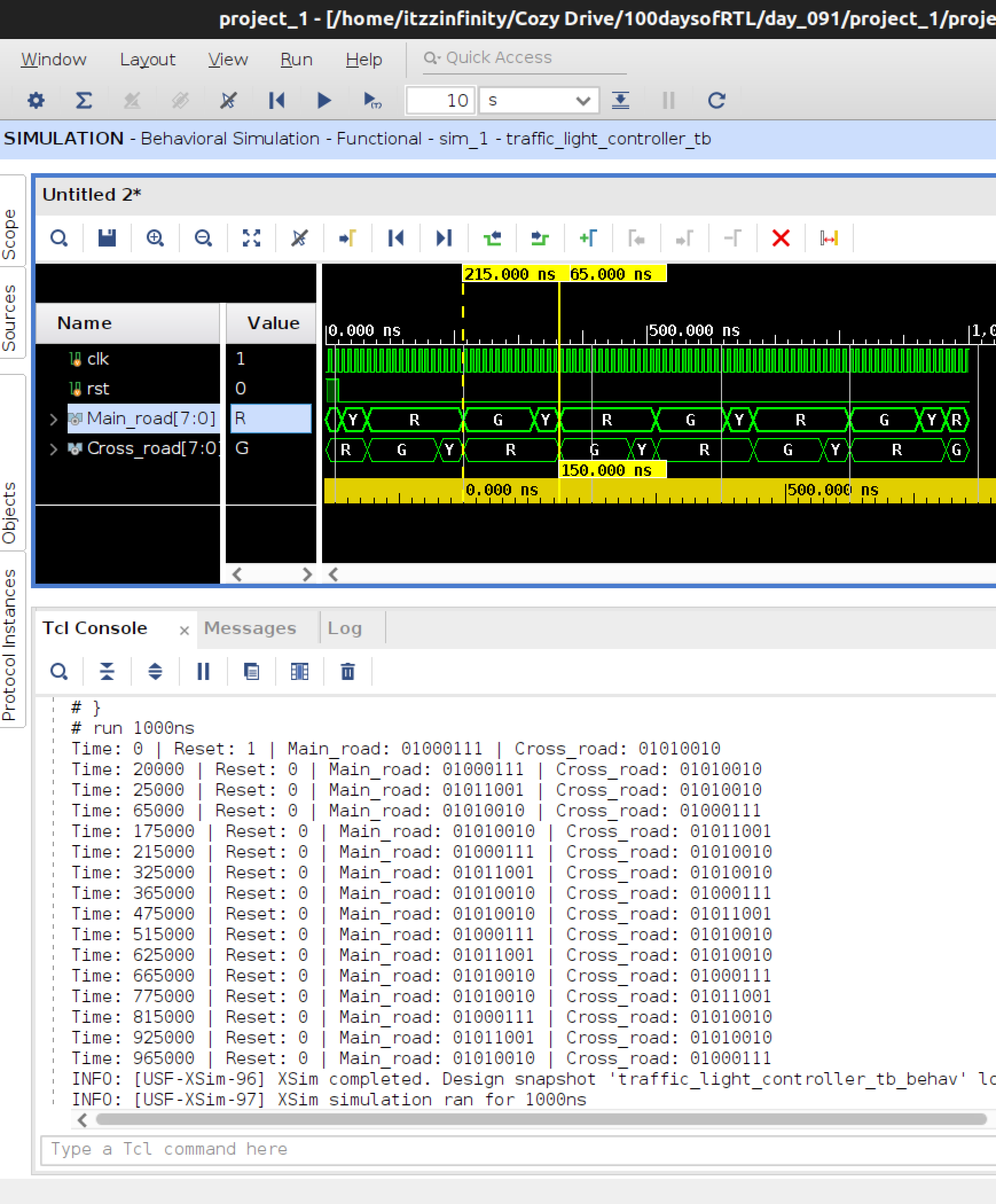
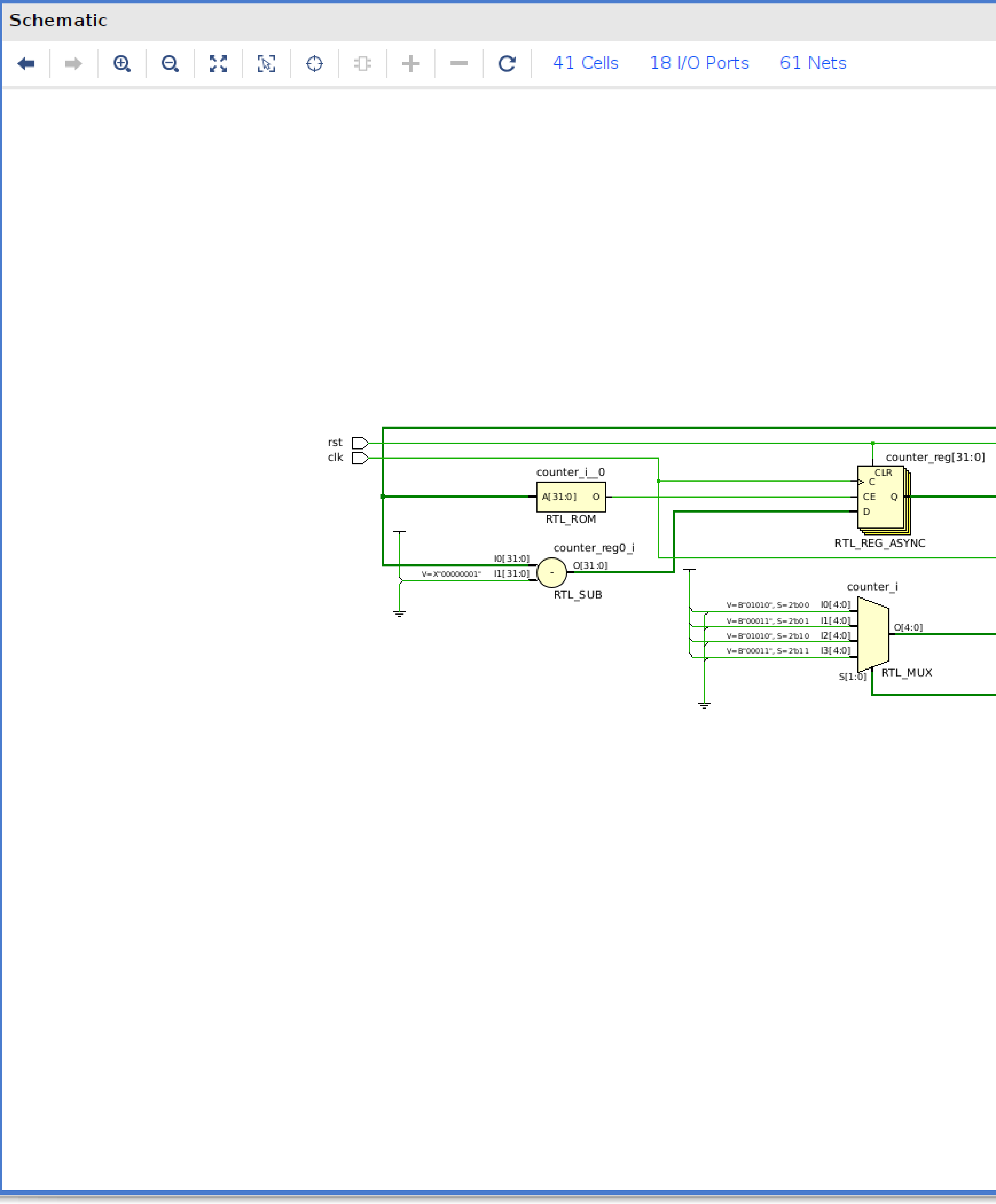
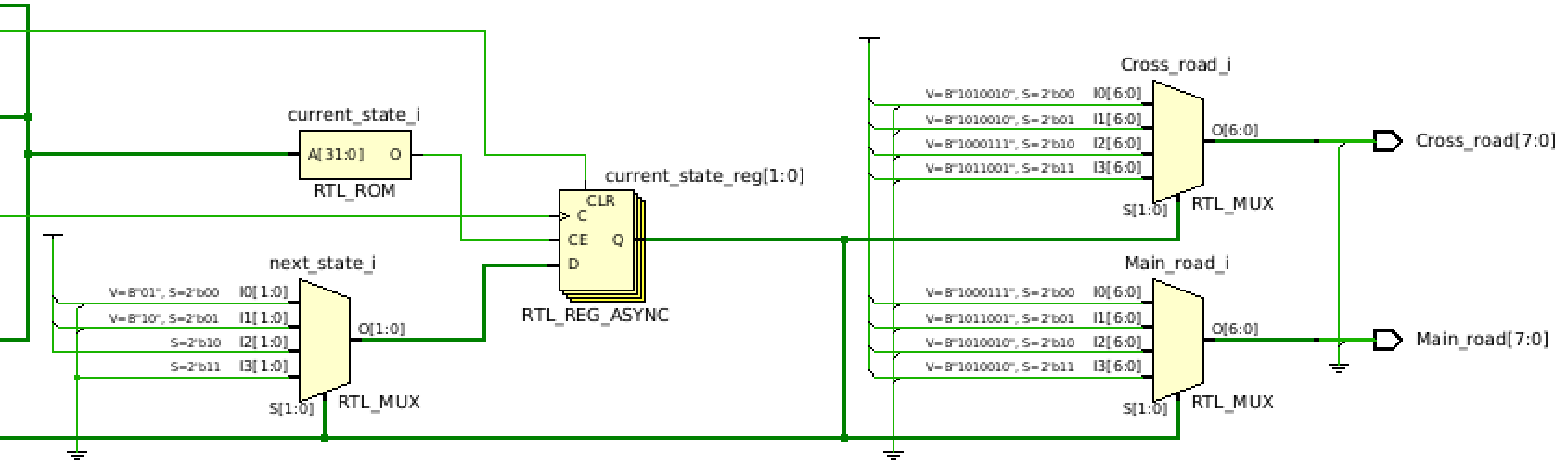


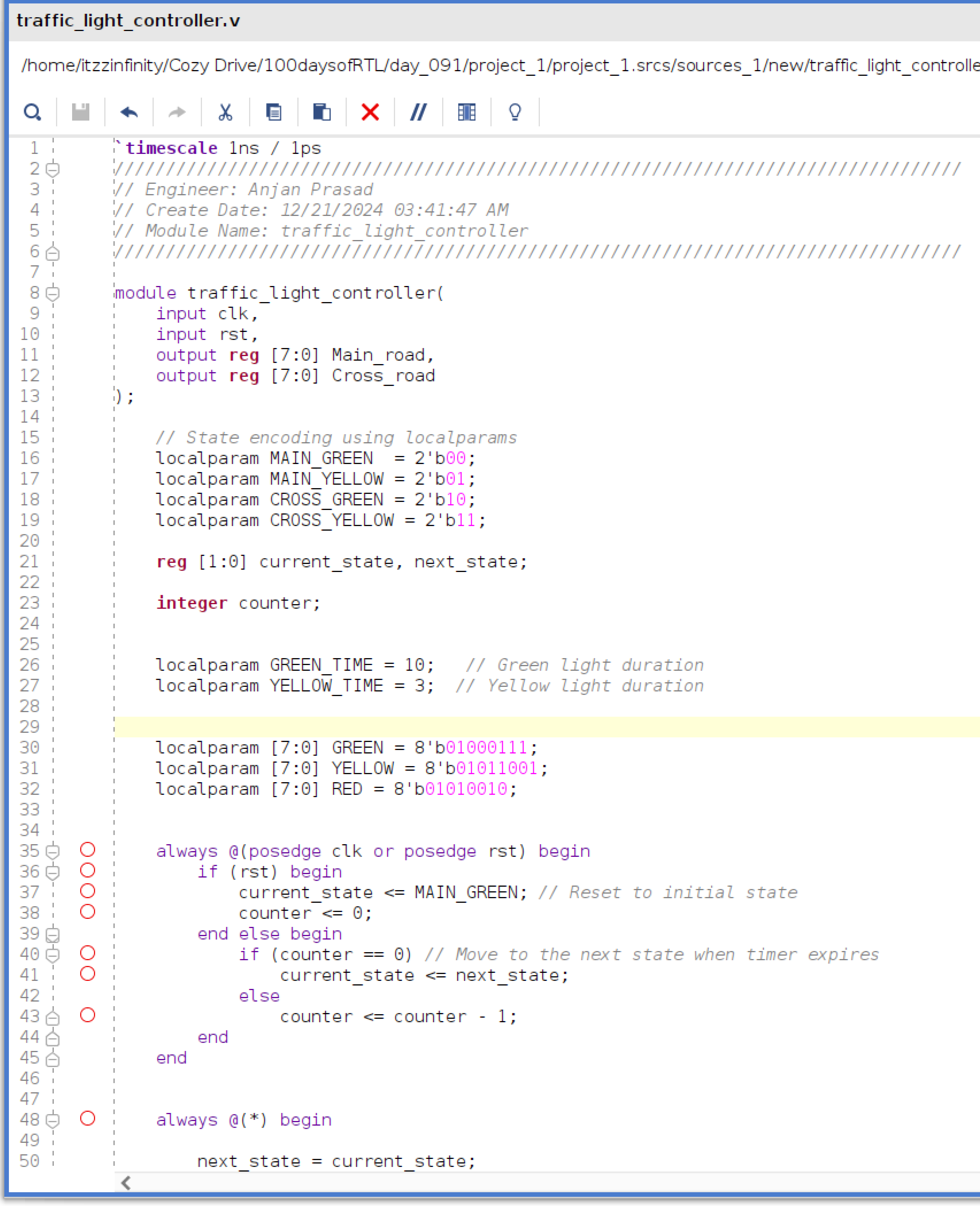
project_1 - [/home/itzzinfinity/Cozy Drive/100daysofRTL/day_091/project_1/proje

SIMULATION - Behavioral Simulation - Functional - sim_1 - traffic_light_controller_tb

```
# }
# run 1000ns
Time: 0 | Reset: 1 | Main_road: 01000111 | Cross_road: 01010010
Time: 20000 | Reset: 0 | Main_road: 01000111 | Cross_road: 01010010
Time: 25000 | Reset: 0 | Main_road: 01011001 | Cross_road: 01010010
Time: 65000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01000111
Time: 175000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01011001
Time: 215000 | Reset: 0 | Main_road: 01000111 | Cross_road: 01010010
Time: 325000 | Reset: 0 | Main_road: 01011001 | Cross_road: 01010010
Time: 365000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01000111
Time: 475000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01011001
Time: 515000 | Reset: 0 | Main_road: 01000111 | Cross_road: 01010010
Time: 625000 | Reset: 0 | Main_road: 01011001 | Cross_road: 01010010
Time: 665000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01000111
Time: 775000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01011001
Time: 815000 | Reset: 0 | Main_road: 01000111 | Cross_road: 01010010
Time: 925000 | Reset: 0 | Main_road: 01011001 | Cross_road: 01010010
Time: 965000 | Reset: 0 | Main_road: 01010010 | Cross_road: 01000111
INFO: [USF-XSim-96] XSim completed. Design snapshot 'traffic_light_controller_tb_behav' lo
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
```


Schematic
41 Cells
18 I/O Ports
61 Nets
rst
clk
counter_i__0
A[31:0]
O
RTL_ROM
counter_reg[31:0]
CLR
C
CE
Q
D
RTL_REG_ASYNC
counter_reg0_i
I0[31:0]
V=X"00000001"
I1[31:0]
-
O[31:0]
RTL_SUB
counter_i
V=B"01010", S=2'b00
I0[4:0]
V=B"00011", S=2'b01
I1[4:0]
V=B"01010", S=2'b10
I2[4:0]
V=B"00011", S=2'b11
I3[4:0]
O[4:0]
S[1:0]
RTL_MUX

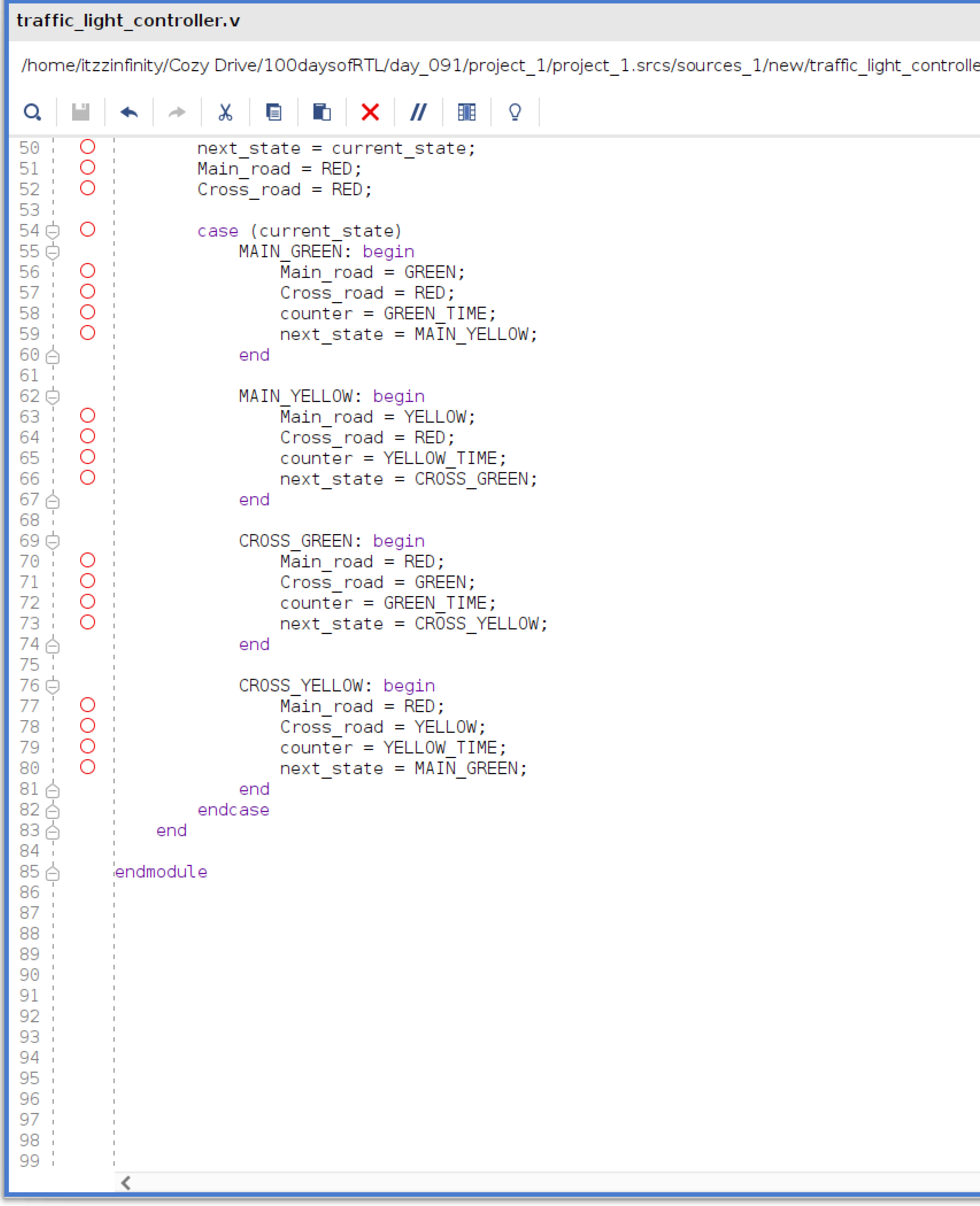

current_state_i
A[31:0]
O
RTL_ROM
current_state_reg[1:0]
CLR
C
CE
Q
D
RTL_REG_ASYNC
next_state_i
V=8"01", S=2'b00
I0[1:0]
V=8"10", S=2'b01
I1[1:0]
S=2'b10
I2[1:0]
S=2'b11
I3[1:0]
O[1:0]
S[1:0]
RTL_MUX
Cross_road_i
V=8"1010010", S=2'b00
I0[6:0]
V=8"1010010", S=2'b01
I1[6:0]
V=8"1000111", S=2'b10
I2[6:0]
V=8"1011001", S=2'b11
I3[6:0]
O[6:0]
S[1:0]
RTL_MUX
Cross_road[7:0]
Main_road_i
V=8"1000111", S=2'b00
I0[6:0]
V=8"1011001", S=2'b01
I1[6:0]
V=8"1010010", S=2'b10
I2[6:0]
V=8"1010010", S=2'b11
I3[6:0]
O[6:0]
S[1:0]
RTL_MUX
Main_road[7:0]



traffic_light_controller.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_091/project_1/project_1.srcs/sources_1/new/traffic_light_controlle

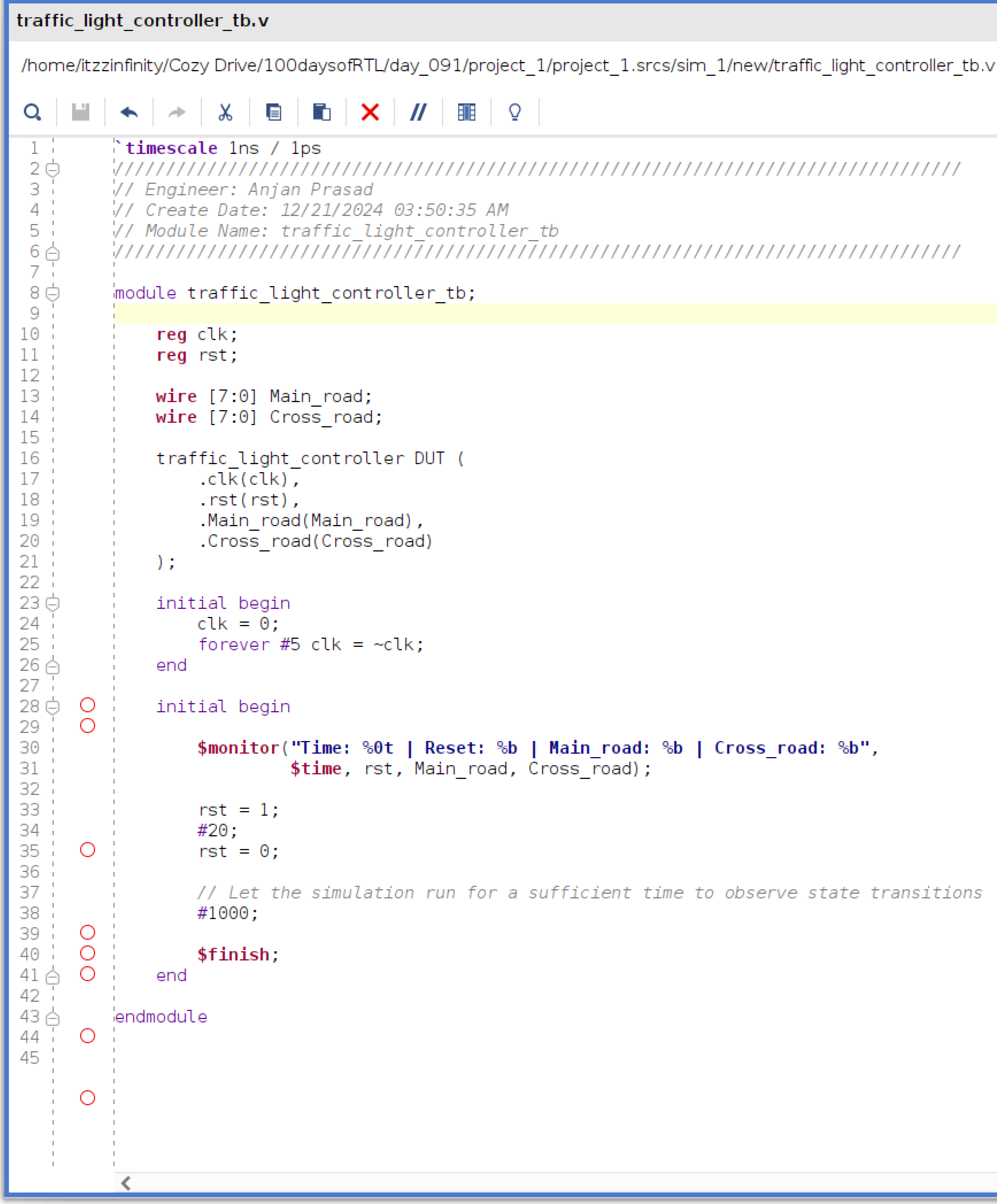
```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 12/21/2024 03:41:47 AM
// Module Name: traffic_light_controller
//////////////////////////////////////////////////////////////////////////////////

module traffic_light_controller(
    input clk,
    input rst,
    output reg [7:0] Main_road,
    output reg [7:0] Cross_road
);

    // State encoding using localparams
    localparam MAIN_GREEN  = 2'b00;
    localparam MAIN_YELLOW = 2'b01;
    localparam CROSS_GREEN = 2'b10;
    localparam CROSS_YELLOW = 2'b11;

    reg [1:0] current_state, next_state;

    integer counter;


    localparam GREEN_TIME = 10;   // Green light duration
    localparam YELLOW_TIME = 3;  // Yellow light duration


    localparam [7:0] GREEN = 8'b01000111;
    localparam [7:0] YELLOW = 8'b01011001;
    localparam [7:0] RED = 8'b01010010;


    always @(posedge clk or posedge rst) begin
        if (rst) begin
            current_state <= MAIN_GREEN; // Reset to initial state
            counter <= 0;
        end else begin
            if (counter == 0) // Move to the next state when timer expires
                current_state <= next_state;
            else
                counter <= counter - 1;
        end
    end


    always @(*) begin

        next_state = current_state;
```



traffic_light_controller.v

/home/itzzinfinity/Cozy Drive/100daysofRTL/day_091/project_1/project_1.srcs/sources_1/new/traffic_light_controlle

```
        next_state = current_state;
        Main_road = RED;
        Cross_road = RED;

        case (current_state)
            MAIN_GREEN: begin
                Main_road = GREEN;
                Cross_road = RED;
                counter = GREEN_TIME;
                next_state = MAIN_YELLOW;
            end

            MAIN_YELLOW: begin
                Main_road = YELLOW;
                Cross_road = RED;
                counter = YELLOW_TIME;
                next_state = CROSS_GREEN;
            end

            CROSS_GREEN: begin
                Main_road = RED;
                Cross_road = GREEN;
                counter = GREEN_TIME;
                next_state = CROSS_YELLOW;
            end

            CROSS_YELLOW: begin
                Main_road = RED;
                Cross_road = YELLOW;
                counter = YELLOW_TIME;
                next_state = MAIN_GREEN;
            end
        endcase
    end

endmodule
```


traffic_light_controller_tb.v
/home/itzzinfinity/Cozy Drive/100daysofRTL/day_091/project_1/project_1.srcs/sim_1/new/traffic_light_controller_tb.v

```verilog
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Engineer: Anjan Prasad
// Create Date: 12/21/2024 03:50:35 AM
// Module Name: traffic_light_controller_tb
//////////////////////////////////////////////////////////////////////////////////

module traffic_light_controller_tb;

    reg clk;
    reg rst;

    wire [7:0] Main_road;
    wire [7:0] Cross_road;

    traffic_light_controller DUT (
        .clk(clk),
        .rst(rst),
        .Main_road(Main_road),
        .Cross_road(Cross_road)
    );

    initial begin
        clk = 0;
        forever #5 clk = ~clk;
    end

    initial begin

        $monitor("Time: %0t | Reset: %b | Main_road: %b | Cross_road: %b",
                 $time, rst, Main_road, Cross_road);

        rst = 1;
        #20;
        rst = 0;

        // Let the simulation run for a sufficient time to observe state transitions
        #1000;

        $finish;
    end

endmodule
```