Submitted by; Submitted to;

**SHREENIDHI R S**  **1stop**

Government Engineering College Keep Growing

Hassan – 573201

Project

**TRAFFIC LIGHT CONTROLLER**

**ABSTRACT:**

The **Digital Traffic Light Controller** is designed to manage traffic flow at an intersection between a busy main street (North-South) and a less frequently used side street (East-West). The system ensures that the North-South direction remains green for a minimum of 25 seconds to accommodate the heavier traffic flow. It will continue to stay green until vehicles are detected on the East-West side. When traffic is detected, the East-West signal turns green for a maximum of 25 seconds, ensuring minimal disruption to the main street. A yellow light phase of 4 seconds is implemented for both directions before transitioning between red and green signals, ensuring safe and smooth traffic flow at all times. The controller is designed for efficiency, using sensors to monitor traffic demand and adjust signal timings dynamically.

**INTRODUCTION:**

The **Digital Traffic Light Controller** is a critical system designed to efficiently manage the flow of vehicles at intersections, particularly where a busy main street intersects with a less frequently used side street. This design focuses on optimizing traffic flow and reducing unnecessary delays. The North-South direction, representing the busier main street, will remain green for a minimum of 25 seconds, ensuring that heavier traffic is prioritized. It will continue to stay green until vehicles are detected on the East-West side street. Once traffic is detected, the East-West signal will turn green for a maximum of 25 seconds to allow cross-traffic to flow. To ensure safe transitions between the green and red signals, both directions will display a yellow light for 4 seconds before switching. This controller uses real-time traffic detection and fixed timing to provide a reliable and efficient solution for managing intersection traffic.

**DESIGN:**

1. **North-South (Main Street) Green Light**:
   * **Minimum duration**: 25 seconds.
   * Stays green as long as no vehicles are detected on East-West.
2. **East-West (Side Street) Green Light**:
   * **Maximum duration**: 25 seconds, triggered by vehicle detection.
   * Automatically switches back to North-South when no traffic is present or the time elapses.
3. **Yellow Light (Both Streets)**:
   * **Duration**: 4 seconds before switching to red.
4. **Sensors**:
   * **Vehicle Detector on East-West Street** to trigger East-West green light when a car is present.
5. **Assumptions**:
   * Initial state: North-South green, East-West red.
   * System will continuously cycle between these phases.

**CODE:**

**Traffic light controller**

`timescale 1ns / 1ps

module tlc\_fsm{

//global signals

input wire i\_clk ,

input wire i\_rst ,

//inputs

input wire i\_ew\_vd ,

//outputs

output wire o\_ns\_red ,

output wire o\_ns\_yellow ,

output wire o\_ns\_green ,

output wire o\_ew\_red ,

output wire o\_ew\_yellow ,

output wire o\_ew\_green ,

};

//state declaration

localparm NS\_GREEN = 0,

YELLOW = 1,

EW\_GREEN = 2;

reg ns\_to\_ew ;

reg[9:0] counter ;

reg[1:0] state, next\_state ;

reg ns\_red , next\_ns\_red ;

reg ns\_yellow , next\_ns\_yellow ;

reg ns\_green , next\_ns\_green ;

reg ew\_red , next\_ew\_red ;

reg ew\_yellow , next\_ew\_yellow ;

reg ew\_green , next\_ew\_green ;

//update state

always@(posedge i\_clk) begin

if (i\_rst)begin

ns\_red <= 1'b0 ;

ns\_yellow <= 1'b0 ;

ns\_green <= 1'b1 ;

ew\_red <= 1'b1 ;

ew\_yellow <= 1'b0 ;

ew\_green <= 1'b0 ;

state <= NS\_GREEN ;

end else begin

ns\_red <= next\_ns\_red ;

ns\_yellow <= next\_ns\_yellow ;

ns\_green <= next\_ns\_green ;

ew\_red <= next\_ew\_red ;

ew\_yellow <= next\_ew\_yellow ;

ew\_green <= next\_ew\_green ;

state <= next\_state ;

end

//counter logic

//reset || state change NS -EW || state change EW - NS

if(i\_rst || (counter > 25 && i\_ew\_vd && state == NS\_GREEN )|| (counter == 4 && state == YELLOW) || (counter >= 25 && (state == ~i\_ew\_vd && state == EW\_GREEN))

counter <= 1'b0 ;

else

counter <= counter +1 ;

end

//state machine logic

always@(\*)begin

//store to memory

next\_ns\_red = ns\_red ;

next\_ns\_yellow = ns\_yellow ;

next\_ns\_green = ns\_green ;

next\_ew\_red = ew\_red ;

next\_ew\_yellow = ew\_yellow ;

next\_ew\_green = ew\_green ;

//FSM

case(state)

NS\_GREEN : begin

if (counter > 25 && i\_ew\_vd) begin

next\_state = YELLOW;

ns\_to\_ew = 1'b1;

end else begin

next\_ns\_red = 1'b0 ;

next\_ns\_yellow = 1'b0 ;

next\_ns\_green = 1'b1 ;

next\_ew\_red = 1'b1 ;

next\_ew\_yellow = 1'b0 ;

next\_ew\_green = 1'b0 ;

next\_state = NS\_GREEN;

end

end

YELLOW: begin

if(counter == 4) begin

if(ns\_to\_ew)

next\_state = EW\_GREEN ;

else

next\_state = NS\_GREEN ;

end else begin

next\_ns\_red = 1'b0 ;

next\_ns\_yellow = 1'b1 ;

next\_ns\_green = 1'b0 ;

next\_ew\_red = 1'b0 ;

next\_ew\_yellow = 1'b1 ;

next\_ew\_green = 1'b0 ;

next\_state = YELLOW;

end

end

EW\_GREEN: begin

if (counter >= 25 || ~i\_ew\_vd) begin

next\_state = YELLOW;

ns\_to\_ew = 1'b0;

end else begin

next\_ns\_red = 1'b1 ;

next\_ns\_yellow = 1'b0 ;

next\_ns\_green = 1'b0 ;

next\_ew\_red = 1'b0 ;

next\_ew\_yellow = 1'b0 ;

next\_ew\_green = 1'b1 ;

next\_state = EW\_GREEN ;

end

end

default: begin

next\_ns\_red = 1'b0 ;

next\_ns\_yellow = 1'b0 ;

next\_ns\_green = 1'b1 ;

next\_ew\_red = 1'b1 ;

next\_ew\_yellow = 1'b0 ;

next\_ew\_green = 1'b0 ;

end

endcase

end

**TEST BENCH**

`timescale 1ns / 1ps

module tlc\_fsm\_tb;

//internal regs/wires

reg clock;

reg reset;

reg ew\_vehicle\_detected;

wire NS\_RED;

wire NS\_YELLOW;

wire NS\_GREEN;

wire EW\_RED;

wire EW\_YELLOW;

wire EW\_GREEN;

//gen clk

always #5 clock = ~clock;

//logic block

initial begin

clock =0;

reset =1;

ew\_vehicle\_detected=0;

#1000 ew\_vehicle\_detected=1;

#500 ew\_vehicle\_detected=0;

#500 ew\_vehicle\_detected=1;

#500 ew\_vehicle\_detected=1;

#5000 $stop;

end

//instantiations

tlc\_fam u\_tlc\_fsm{

.i\_clk (clock),

.i\_rst (reset),

.i\_ew\_vd (ew\_vehicle\_detected),

.o\_ns\_red (NS\_RED),

.o\_ns\_yellow(NS\_YELLOW),

.o\_ns\_green (NS\_GREEN),

.o\_ew\_red (EW\_RED),

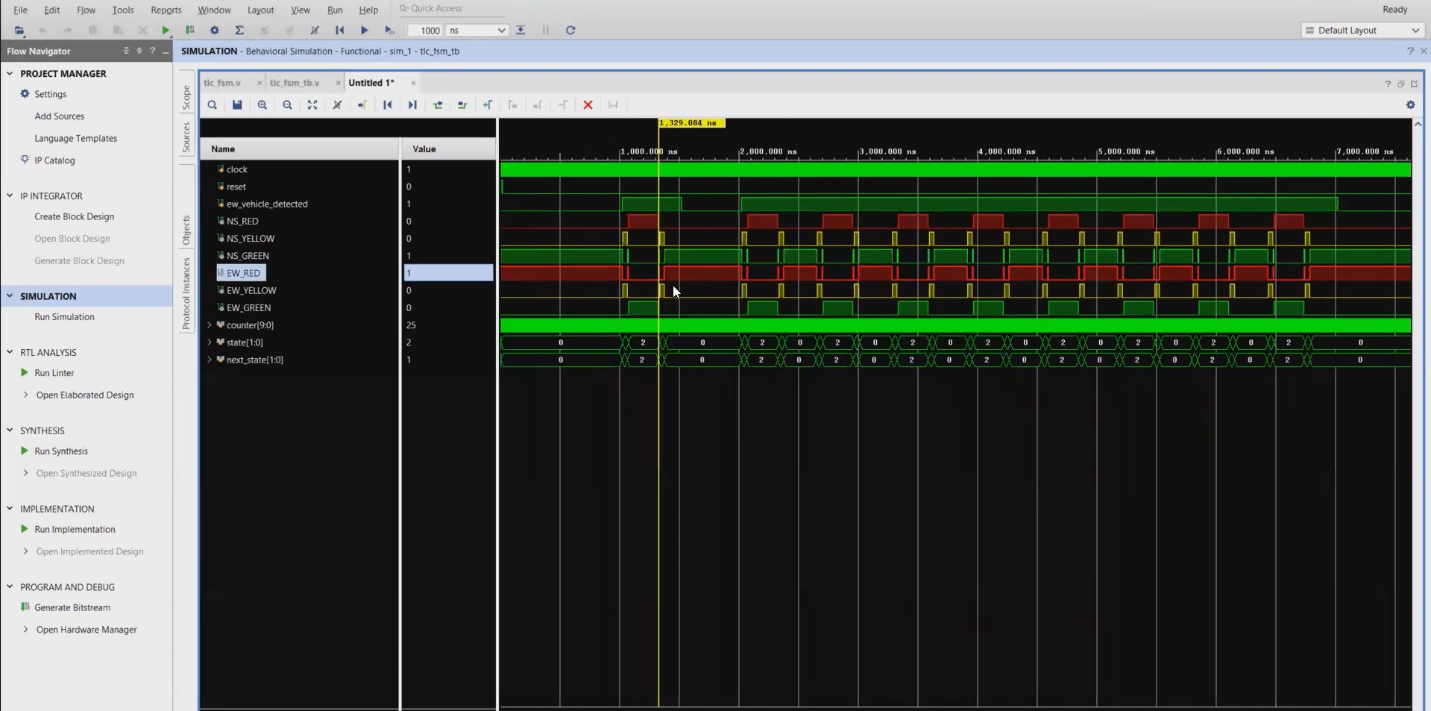
.o\_ew\_yellow(EW\_YELLOW),

.o\_ew\_green (EW\_GREEN)

};

Endmodule

**WAVEFORM**

****

**CONCLUSION**

The **Digital Traffic Light Controller** designed for the intersection of a busy Main Street (North-South) and an occasionally used side street (East-West) provides an efficient, responsive, and safe traffic management solution. By ensuring that the North-South direction stays green for a minimum of 25 seconds and remains green until traffic is detected on the East-West side, the system prioritizes the heavier traffic flow while still accommodating cross-traffic efficiently. The East-West direction is granted a maximum of 25 seconds of green light, balancing the need for minimal delays. A 4-second yellow light interval for both directions ensures smooth and safe transitions between light changes. This design improves traffic flow and reduces wait times at intersections, making it a reliable solution for urban traffic control.