

```

1  //Traffic light controller Engine
2  module traffic_light_ctrl_eng(
3  //port declarations
4  //Inputs
5  input wire [4:0]    i_NS_Count      ,
6  input wire [3:0]    i_EW_Count      ,
7  input wire [1:0]    yellow_count    ,
8  input wire          NS_vehicle_detect ,
9  input wire          EW_vehicle_detect ,
10 //Outputs
11 output reg          NS_red           ,
12 output reg          NS_yellow        ,
13 output reg          NS_green         ,
14 output reg          EW_red           ,
15 output reg          EW_yellow        ,
16 output reg          EW_green         ,
17 );
18
19 //Instalizacion
20 initial begin
21
22     NS_red      <= 1'b0;
23     NS_yellow   <= 1'b0;
24     NS_green    <= 1'b1;
25     EW_red      <= 1'b1;
26     EW_yellow   <= 1'b0;
27     EW_green    <= 1'b0;
28
29 end
30
31 //NS_controller
32 always@(i_NS_Count)begin
33
34     if(i_NS_Count == 31 & EW_vehicle_detect & NS_green)begin
35
36         NS_red      <= 1'b0;
37         NS_yellow   <= 1'b1;
38         NS_green    <= 1'b0;
39         EW_red      <= 1'b1;
40         EW_yellow   <= 1'b0;
41         EW_green    <= 1'b0;
42
43     end
44
45 end
46
47 //EW_controller
48 always@(i_EW_Count)begin
49
50     if(i_EW_Count == 15 & NS_vehicle_detect & EW_green)begin
51
52         NS_red      <= 1'b1;
53         NS_yellow   <= 1'b0;
54         NS_green    <= 1'b0;
55         EW_red      <= 1'b0;
56         EW_yellow   <= 1'b1;
57         EW_green    <= 1'b0;
58
59     end
60
61 end
62
63 //Yellow_controller
64 always@(yellow_count)begin
65
66     if(yellow_count == 3 & NS_yellow)begin
67
68         NS_red      <= 1'b1;
69         NS_yellow   <= 1'b0;

```

```

70     NS_green    <= 1'b0;
71     EW_red      <= 1'b0;
72     EW_yellow   <= 1'b0;
73     EW_green    <= 1'b1;
74
75     end
76
77     if(yellow_count == 3 & EW_yellow)begin
78
79         NS_red    <= 1'b0;
80         NS_yellow  <= 1'b0;
81         NS_green   <= 1'b1;
82         EW_red     <= 1'b1;
83         EW_yellow  <= 1'b0;
84         EW_green   <= 1'b0;
85
86     end
87
88 end
89
90 endmodule
91
92 //NS counter
93 module NS_Count(
94 //port declarations
95 input wire      i_clk    , //Input clock signal
96 output reg [4:0] o_count  //Output counter
97 );
98
99 //initialization
100 initial
101     o_count = 0;
102
103 always@(negedge i_clk)
104     o_count[0] <= ~o_count[0];
105
106 always@(negedge o_count[0])
107     o_count[1] <= ~o_count[1];
108
109 always@(negedge o_count[1])
110     o_count[2] <= ~o_count[2];
111
112 always@(negedge o_count[2])
113     o_count[3] <= ~o_count[3];
114
115 always@(negedge o_count[4])
116     o_count[4] <= ~o_count[4];
117
118 endmodule
119
120 //EW counter
121 module EW_Count(
122 //port declarations
123 input wire      i_clk    , //Input clock signal
124 output reg [3:0] o_count  //Output counter
125 );
126
127 //initialization
128 initial
129     o_count = 0;
130
131 always@(negedge i_clk)
132     o_count[0] <= ~o_count[0];
133
134 always@(negedge o_count[0])
135     o_count[1] <= ~o_count[1];
136
137 always@(negedge o_count[1])
138     o_count[2] <= ~o_count[2];

```

```

139
140 always@(negedge o_count[2])
141     o_count[3] <= ~o_count[3];
142
143 endmodule
144
145 //Yellow counter
146 module yellow_count(
147     //port declarations
148     input wire      i_clk      , //Input clock signal
149     output reg [1:0] o_count    //Output counter
150 );
151
152 //initialization
153 initial
154     o_count = 0;
155
156 always@(negedge i_clk)
157     o_count[0] <= ~o_count[0];
158
159 always@(negedge o_count[0])
160     o_count[1] <= ~o_count[1];
161
162 endmodule

```

```

1  `timescale 1ns / 1ps
2  ///////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date: 19.05.2024 13:25:13
7  // Design Name:
8  // Module Name: traffic_tb
9  // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 ///////////////////////////////////////////////////////////////////
21
22
23 module traffic_tb;
24
25 //Internal wires/regs
26 wire [4:0] i_NS_Count    ;
27 wire [3:0] i_EW_Count    ;
28 wire [1:0] yellow_count  ;
29 reg        CLK           ;
30
31 //INPUTS
32 reg        NS_vehicle_detect  ;
33 reg        EW_vehicle_detect  ;
34
35 //OUTPUTS
36 wire       NS_red            ;
37 wire       NS_yellow        ;
38 wire       NS_green         ;
39 wire       EW_red           ;
40 wire       EW_yellow        ;
41 wire       EW_green         ;
42
43
44 //Initial Block
45 initial begin
46
47     CLK                = 1'b0  ;
48     NS_vehicle_detect  = 1'b1  ;
49     EW_vehicle_detect  = 1'b0  ;
50
51     $display("    NS      |      EW  ");
52     $display("    R    Y    G    R    Y    G  ");
53     $monitor(" %h %h %h %h %h %h ",NS_red, NS_yellow, NS_green, EW_red, EW_yellow,
54     EW_green);
55
56     #1000 $finish;
57
58 end
59
60 //Clock Generator
61 always
62     #5 CLK                = ~CLK  ;
63
64 //Test case 2
65 //always@(CLK) begin
66 //    if($time % 21 == 0) begin
67 //        NS_vehicle_detect = ~NS_vehicle_detect;
68 //        EW_vehicle_detect = ~EW_vehicle_detect;
69 //    end

```

```

69 //end
70
71 //end of test case 2
72
73 //Test case 2
74 always@(CLK) begin
75     if($time % 6 == 0)
76         NS_vehicle_detect = ~NS_vehicle_detect;
77     if($time % 15 == 0)
78         EW_vehicle_detect = ~EW_vehicle_detect;
79 end
80
81 //end of test case 3
82 //Instantiations
83
84 //TRAFFIC CORE
85 traffic_light_ctrl_eng CORE(
86 //inputs
87 .i_NS_Count          (i_NS_Count),
88 .i_EW_Count          (i_EW_Count),
89 .yellow_count        (yellow_count),
90 .NS_vehicle_detect   (NS_vehicle_detect),
91 .EW_vehicle_detect   (EW_vehicle_detect),
92 //outputs
93 .NS_red              (NS_red),
94 .NS_yellow           (NS_yellow),
95 .NS_green            (NS_green),
96 .EW_red              (EW_red),
97 .EW_yellow           (EW_yellow),
98 .EW_green            (EW_green)
99 ) ;
100
101 //NORTH SOUTH COUNTER
102 NS_Count i_NS_Count_0 (
103     .i_clk              (CLK),
104     .o_count            (i_NS_Count)
105 ) ;
106
107 //EAST WEST COUNTER
108 EW_Count i_EW_Count_0 (
109     .i_clk              (CLK),
110     .o_count            (i_EW_Count)
111 ) ;
112
113 //YELLOW LIGHT COUNTER
114 yellow_count i_yellow_count_0 (
115     .i_clk              (CLK),
116     .o_count            (yellow_count)
117 ) ;
118 endmodule
119

```



