TRAFFIC LIGHT CONTROLLER USING VERILOG

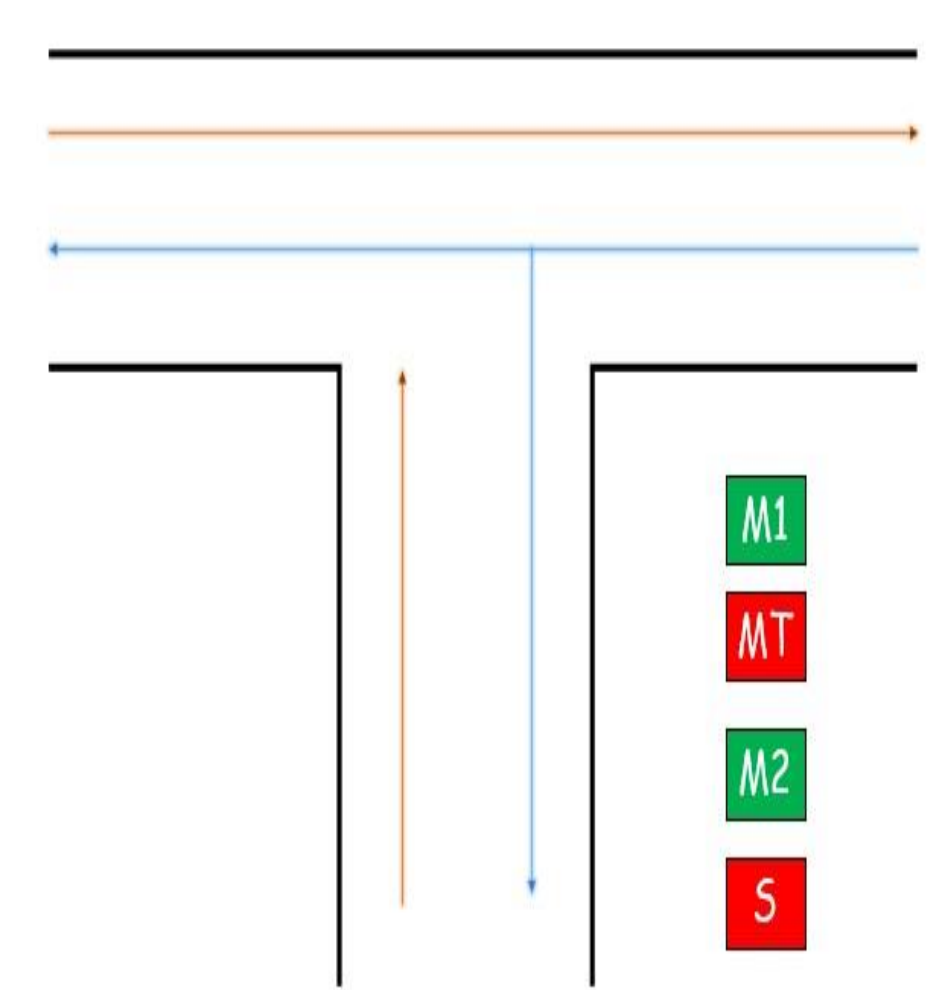
PRESENTED BY:

AMAAN AHMAD (2K19/EE/031)

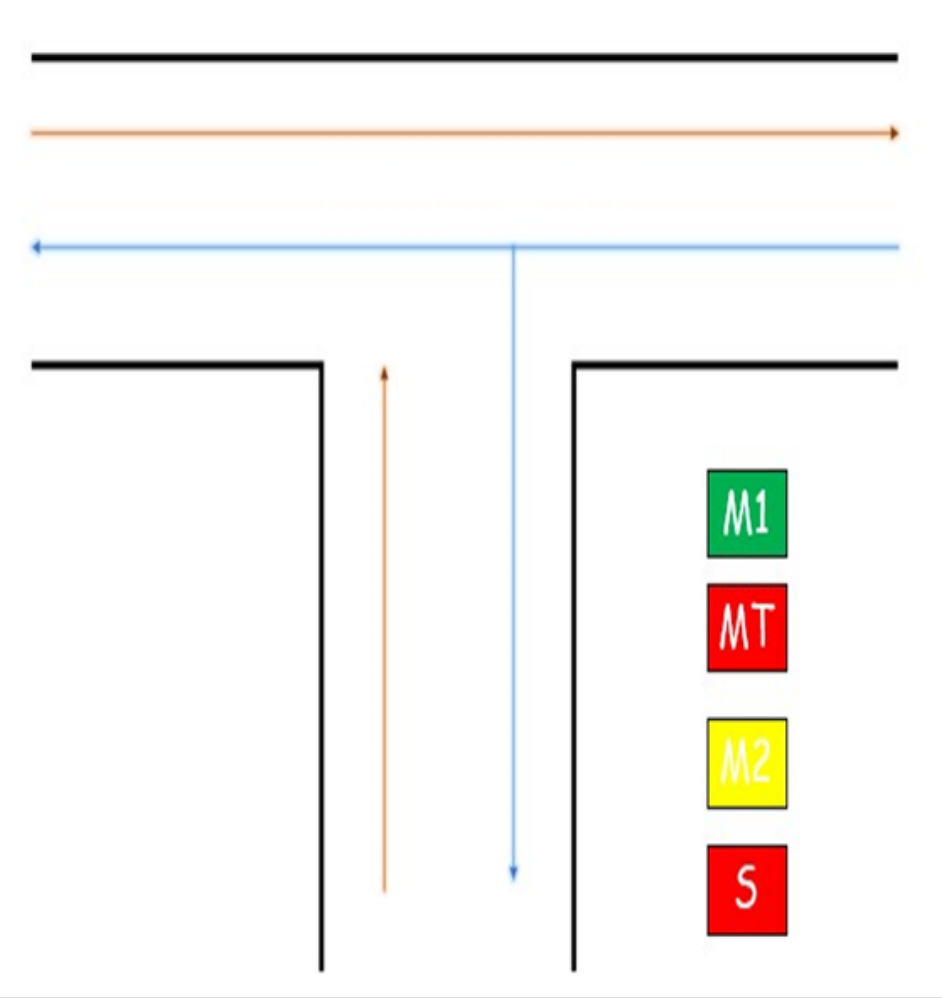
ANIRUDH SINGH (2K19/EE/042)

INTRODUCTION

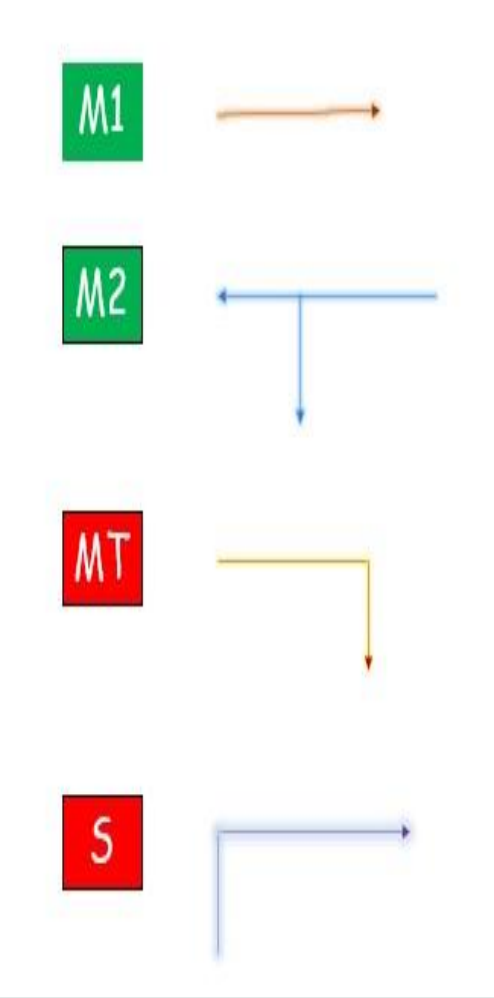
In this project our aim is to simulate a traffic light controller at a T point. We are going to use Verilog to simulate this controller so that it can change the traffic light after a pre-defined time interval so that the traffic can be regulated.

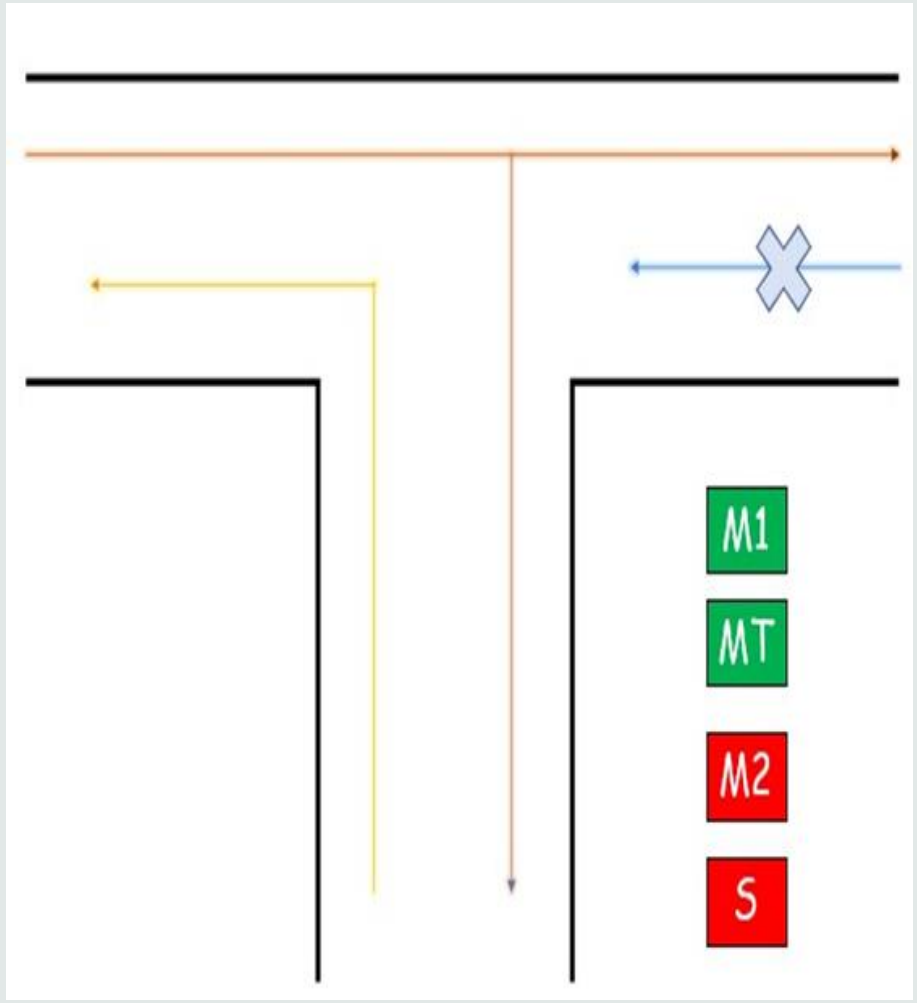


STATE-1

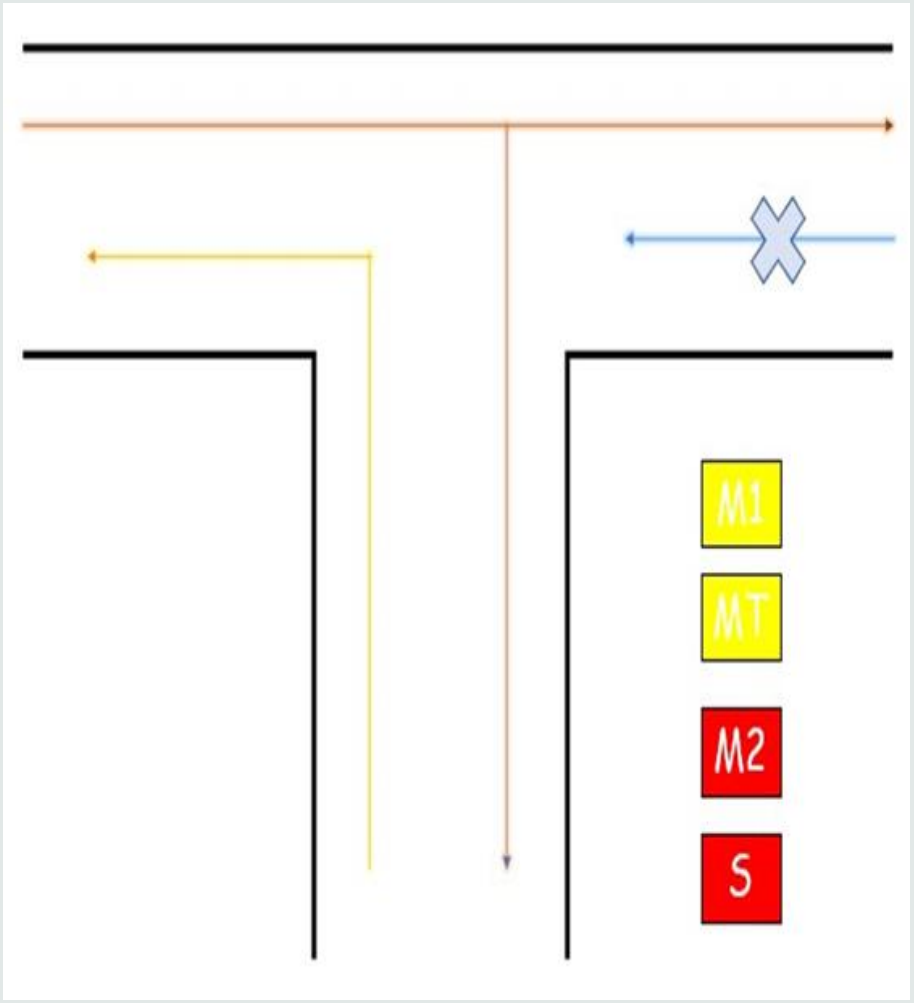


STATE-2

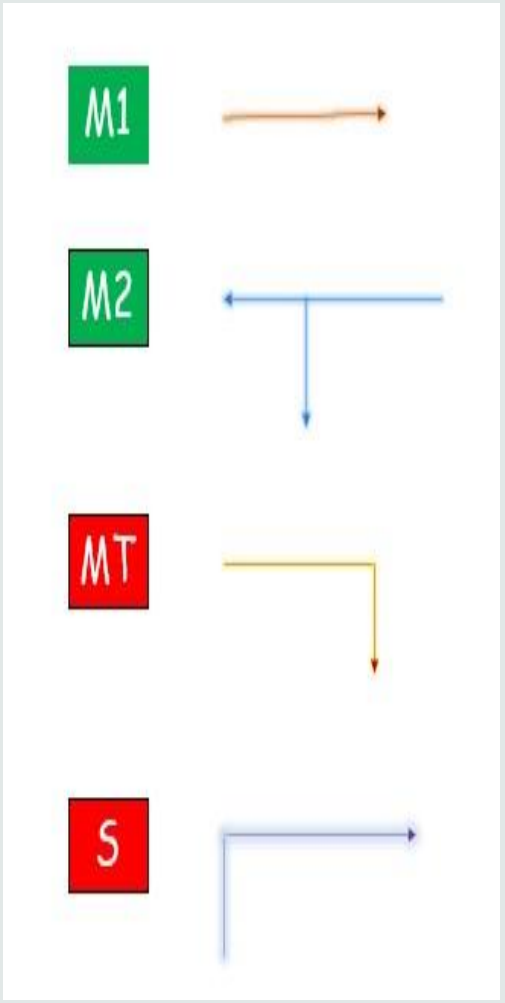


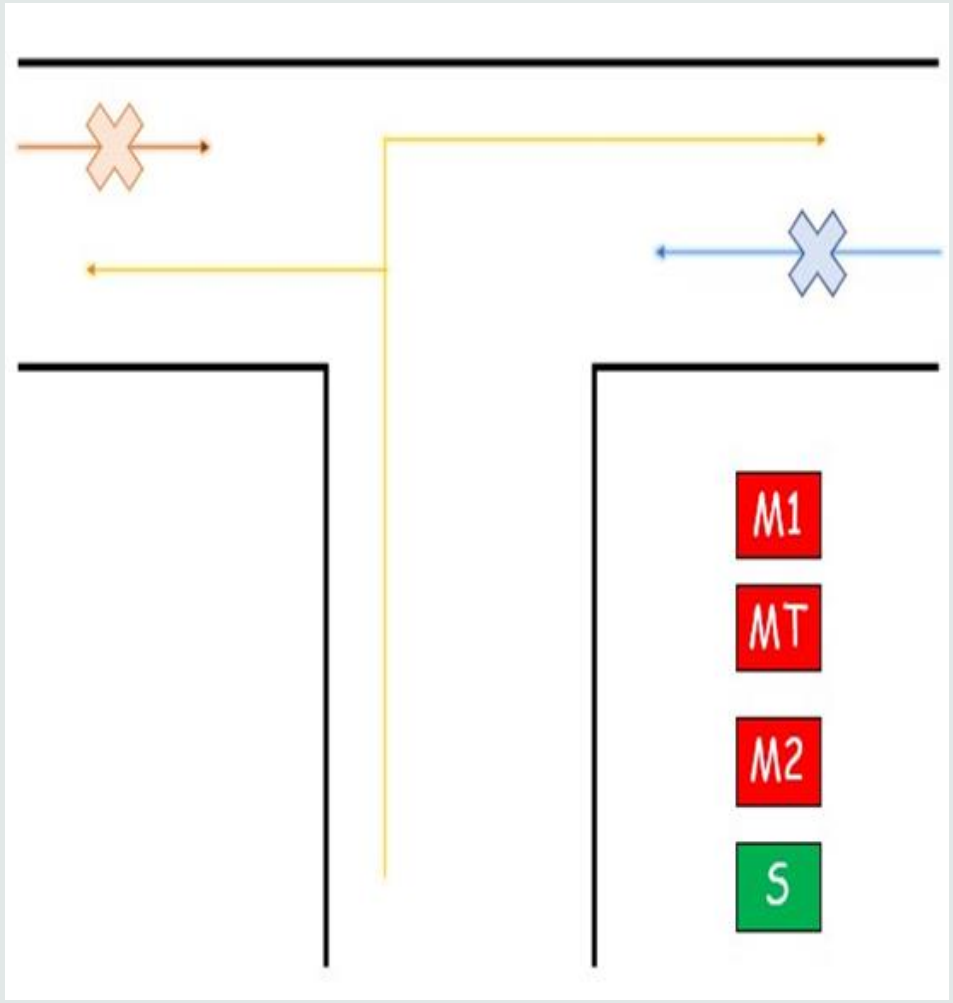


STATE-3

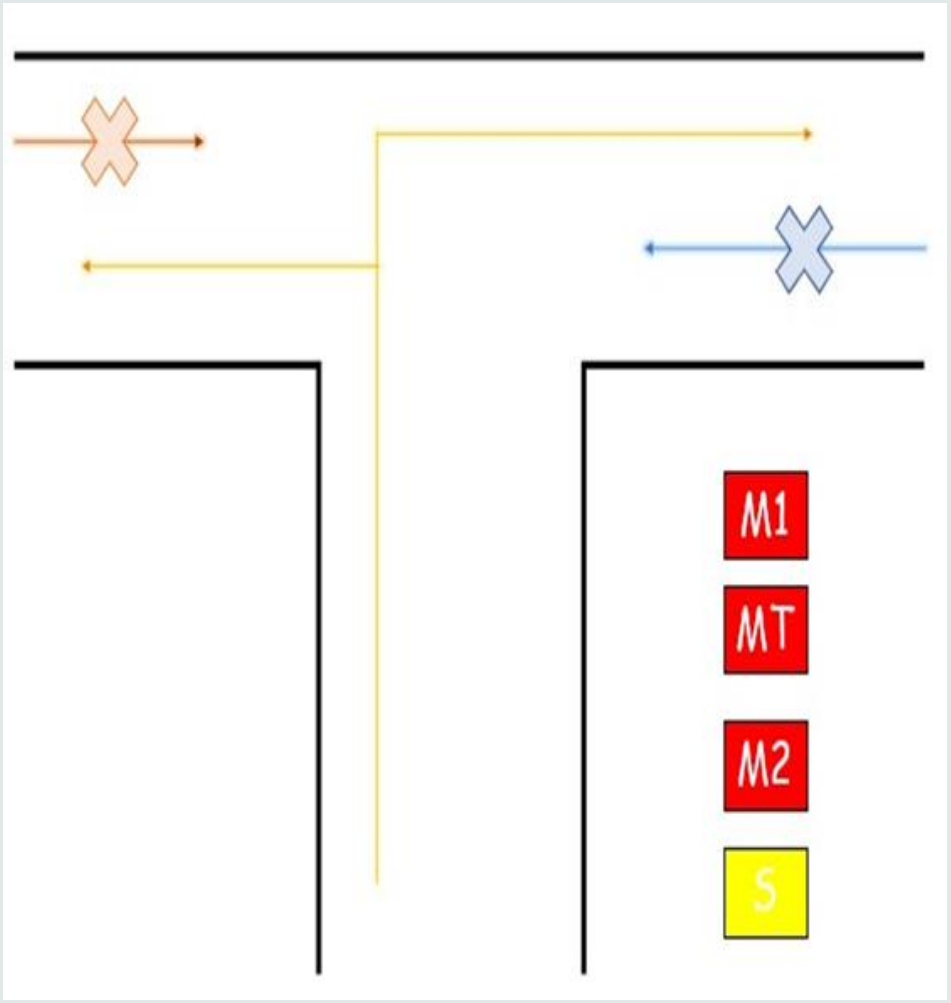


STATE-4

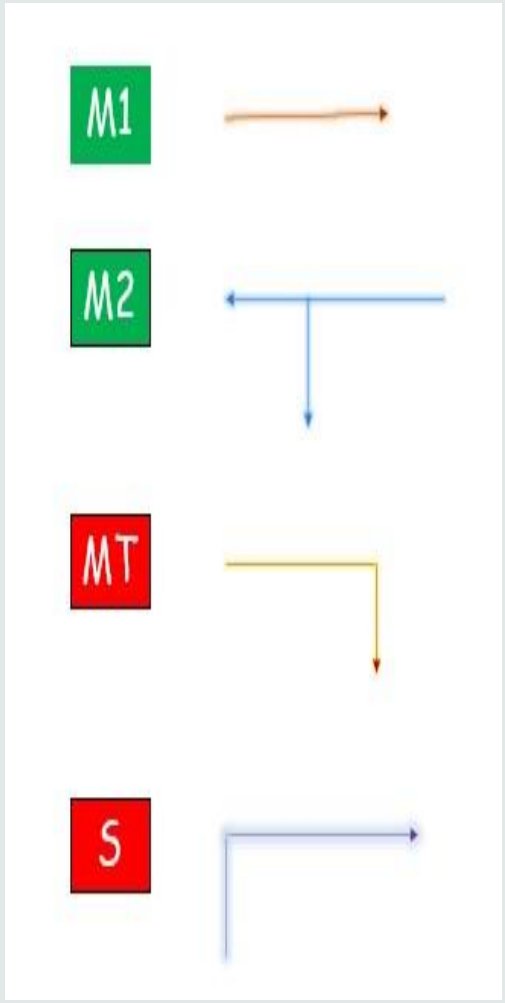


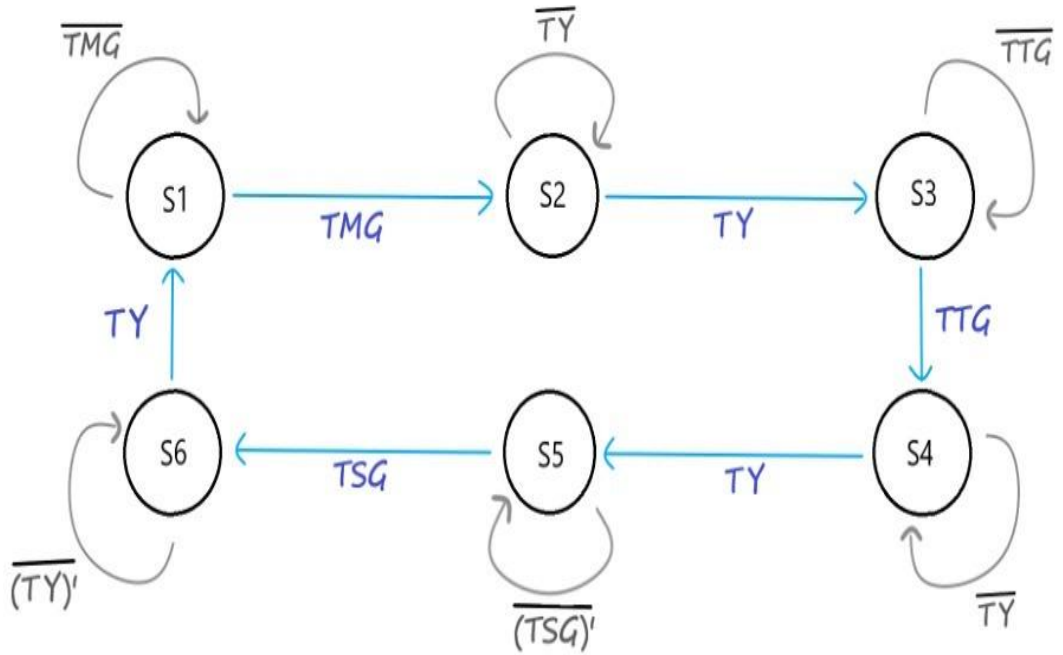


STATE-5



STATE-6





STATE DIAGRAM

STATE ASSIGNMENT:

- S1= 000
- S2= 001
- S3= 010
- S4= 011
- S5= 100
- S6= 101

STATE TABLE

PRESENT STATE	INPUT	NEXT STATE	M1 R Y G	M2 R Y G	MT R Y G	S R Y G
0 0 0	T(MG)′ TMG	0 0 0 0 0 1	0 0 1	0 0 1	1 0 0	1 0 0
0 0 1	TY′ TY	0 0 1 0 1 0	0 0 1	0 1 0	1 0 0	1 0 0
0 1 0	T(TG)′ TTG	0 1 0 0 1 1	0 0 1	1 0 0	0 0 1	1 0 0
0 1 1	TY′ TY	0 1 1 1 0 0	0 1 0	1 0 0	0 1 0	1 0 0
1 0 0	T(SG)′ TSG	1 0 0 1 0 1	1 0 0	1 0 0	1 0 0	0 0 1
1 0 1	TY′ TY	1 0 1 0 0 0	1 0 0	1 0 0	1 0 0	0 1 0

VERILOG CODE FOR TRAFFIC LIGHT CONTROLLER

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date:    21:09:25 03/22/2021
7  // Design Name:
8  // Module Name:    DCSdraft_1
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 module DCSdraft_1(
22     input clk,
23     input rst,
24     output reg [2:0] light_M1,
25     output reg [2:0] light_S,
26     output reg [2:0] light_MT,
27     output reg [2:0] light_M2
28     );
29      parameter  S1=0, S2=1, S3 =2, S4=3, S5=4,S6=5;
30     reg [3:0]count;
31     reg[2:0] ps;
32     parameter  sec7=7,sec5=5,sec2=2,sec3=3;
33
34
```

```
35
36      always@(posedge clk or posedge rst)
37          begin
38          if(rst==1)
39          begin
40          ps<=S1;
41          count<=0;
42          end
43          else
44
45
46              case(ps)
47                  S1: if(count<sec7)
48                          begin
49                          ps<=S1;
50                          count<=count+1;
51                          end
52                      else
53                          begin
54                          ps<=S2;
55                          count<=0;
56                          end
57                  S2: if(count<sec2)
58                          begin
59                          ps<=S2;
60                          count<=count+1;
61                          end
62
63                      else
64                          begin
65                          ps<=S3;
66                          count<=0;
67                          end
```

```
68                  S3: if(count<sec5)
69                          begin
70                          ps<=S3;
71                          count<=count+1;
72                          end
73
74                      else
75                          begin
76                          ps<=S4;
77                          count<=0;
78                          end
79                  S4:if(count<sec2)
80                          begin
81                          ps<=S4;
82                          count<=count+1;
83                          end
84
85                      else
86                          begin
87                          ps<=S5;
88                          count<=0;
89                          end
90                  S5:if(count<sec3)
91                          begin
92                          ps<=S5;
93                          count<=count+1;
94                          end
95
96                      else
97                          begin
98                          ps<=S6;
99                          count<=0;
100                         end
101
```

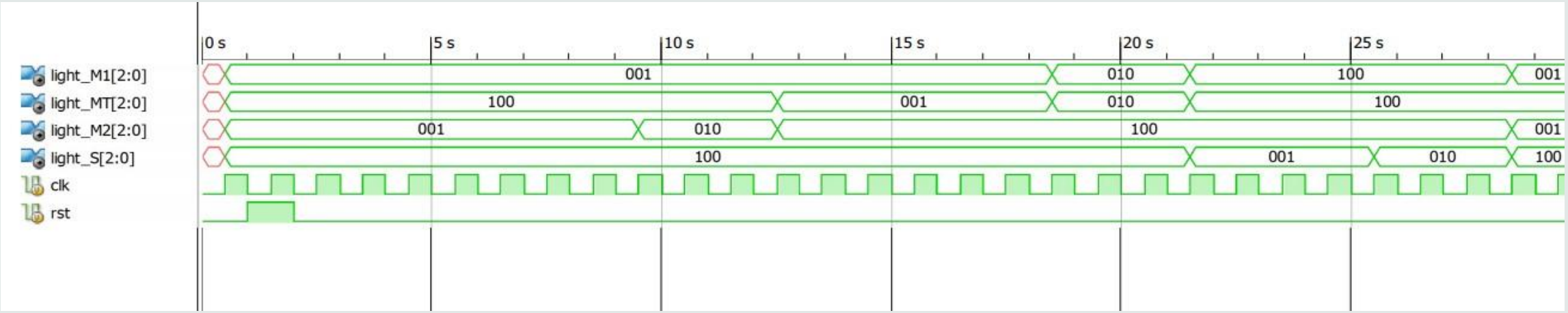
```
102                 S6:if(count<sec2)
103                         begin
104                         ps<=S6;
105                         count<=count+1;
106                         end
107
108                     else
109                         begin
110                         ps<=S1;
111                         count<=0;
112                         end
113                 default: ps<=S1;
114                 endcase
115             end
116
117             always@(ps)
118             begin
119
120                 case(ps)
121
122                     S1:
123                     begin
124                        light_M1<=3'b001;
125                        light_M2<=3'b001;
126                        light_MT<=3'b100;
127                        light_S<=3'b100;
128                     end
129                     S2:
130                     begin
131                        light_M1<=3'b001;
132                        light_M2<=3'b010;
133                        light_MT<=3'b100;
134                        light_S<=3'b100;
135                     end
```

```
136                     S3:
137                     begin
138                        light_M1<=3'b001;
139                        light_M2<=3'b100;
140                        light_MT<=3'b001;
141                        light_S<=3'b100;
142                     end
143                     S4:
144                     begin
145                        light_M1<=3'b010;
146                        light_M2<=3'b100;
147                        light_MT<=3'b010;
148                        light_S<=3'b100;
149                     end
150                     S5:
151                     begin
152                        light_M1<=3'b100;
153                        light_M2<=3'b100;
154                        light_MT<=3'b100;
155                        light_S<=3'b001;
156                     end
157                     S6:
158                     begin
159                        light_M1<=3'b100;
160                        light_M2<=3'b100;
161                        light_MT<=3'b100;
162                        light_S<=3'b010;
163                     end
```

```
164                     default:
165                     begin
166                         light_M1<=3'b000;
167                         light_M2<=3'b000;
168                         light_MT<=3'b000;
169                         light_S<=3'b000;
170                     end
171                     endcase
172             end
173
174
175 endmodule
176
```

```v
`timescale 1ns / 1ps

////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date:   21:13:16 03/22/2021
// Design Name:   DCSdraft_1
// Module Name:   D:/Xilinx/DCSdraft/DCSdraft_test.v
// Project Name:  DCSdraft
// Target Device:
// Tool versions:
// Description:
//
// Verilog Test Fixture created by ISE for module: DCSdraft_1
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
////////////////////////////////////////////////////////////////////////////////

module DCSdraft_test;

    // Inputs
    reg clk;
    reg rst;

    // Outputs
    wire [2:0] light_M1;
    wire [2:0] light_S;
    wire [2:0] light_MT;
    wire [2:0] light_M2;

    // Instantiate the Unit Under Test (UUT)
    DCSdraft_1 uut (
        .clk(clk),
        .rst(rst),
        .light_M1(light_M1),
        .light_S(light_S),
        .light_MT(light_MT),
        .light_M2(light_M2)
    );

initial
begin
    clk=1'b0;
    forever #(1000000000/2) clk=~clk;
end
//    initial
//    $stop;//to add ps


initial
begin
    rst=0;
    #1000000000;
    rst=1;
    #1000000000;
    rst=0;
    #(1000000000*200);
    $finish;
    end

endmodule
```

TEST BENCH CODE OF TRAFFIC LIGHT CONTROLLER



SIMULATION RESULT

THANK YOU