USE OF TIMER IN TRAFFIC LIGHT CONTROLLER

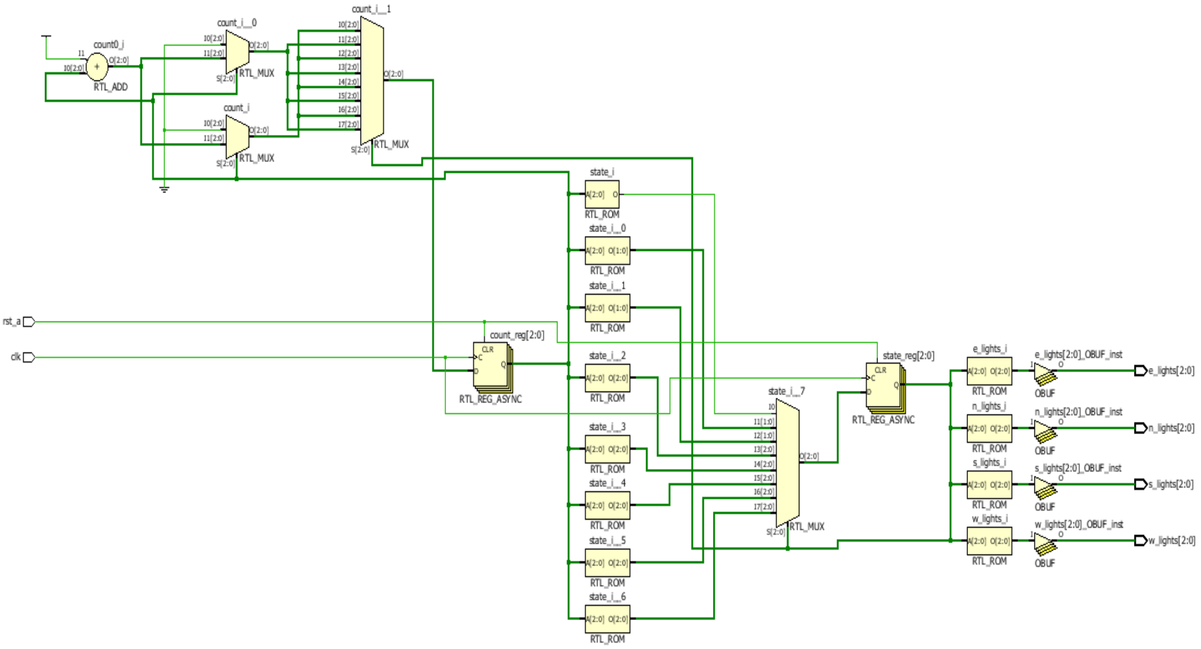
AIM:

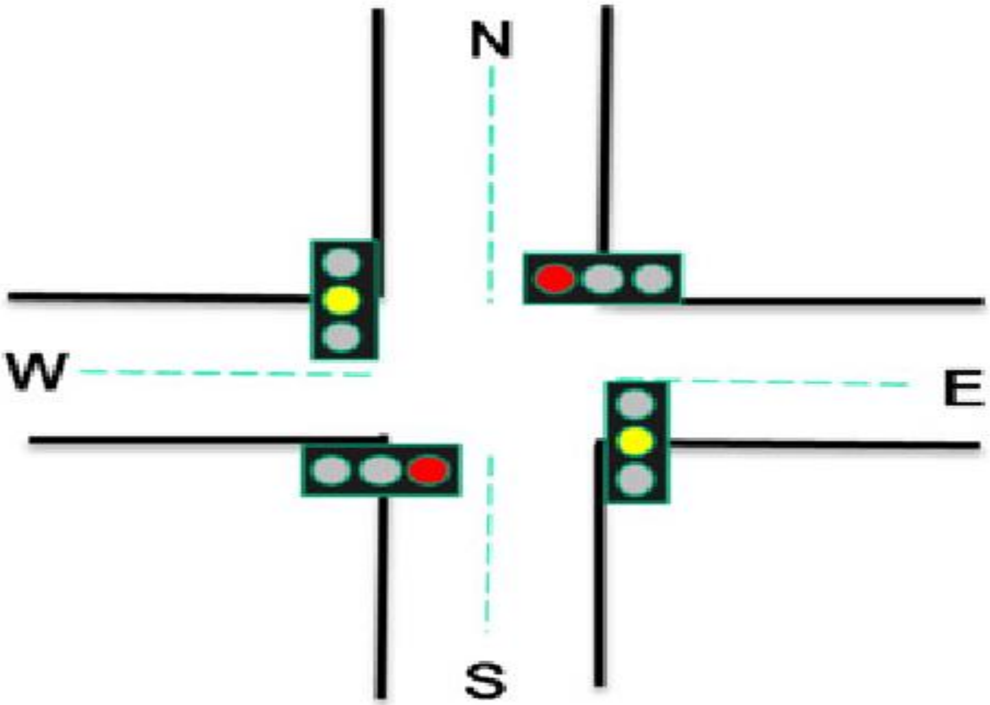
To design Traffic Controller using Timer in ModelSim.

SOFTWARE REQUIRED:

ModelSim

BLOCK DIAGRAM:





PROGRAM:

```
module traffic_control(n_lights,s_lights,e_lights,w_lights,clk,rst_a);

  output reg [2:0] n_lights,s_lights,e_lights,w_lights;
  input      clk;
  input      rst_a;

  reg [2:0] state;

  parameter [2:0] north=3'b000;
  parameter [2:0] north_y=3'b001;
  parameter [2:0] south=3'b010;
  parameter [2:0] south_y=3'b011;
  parameter [2:0] east=3'b100;
  parameter [2:0] east_y=3'b101;
  parameter [2:0] west=3'b110;
  parameter [2:0] west_y=3'b111;

  reg [2:0] count;


  always @(posedge clk, posedge rst_a)
   begin
     if (rst_a)
        begin
            state=north;
```

```
      count =3'b000;
   end
else
   begin
      case (state)
      north :
         begin
            if (count==3'b111)
               begin
               count=3'b000;
               state=north_y;
               end
            else
               begin
               count=count+3'b001;
               state=north;
               end
         end

      north_y :
         begin
            if (count==3'b011)
               begin
               count=3'b000;
               state=south;
               end
            else
               begin
               count=count+3'b001;
               state=north_y;
            end
         end

      south :
         begin
            if (count==3'b111)
               begin
               count=3'b0;
               state=south_y;
               end
            else
               begin
               count=count+3'b001;
               state=south;
            end
         end
```

```
south_y :
  begin
    if (count==3'b011)
      begin
      count=3'b0;
      state=east;
      end
    else
      begin
      count=count+3'b001;
      state=south_y;
      end
    end

east :
  begin
    if (count==3'b111)
      begin
      count=3'b0;
      state=east_y;
      end
    else
      begin
      count=count+3'b001;
      state=east;
      end
    end

east_y :
  begin
    if (count==3'b011)
      begin
      count=3'b0;
      state=west;
      end
    else
      begin
      count=count+3'b001;
      state=east_y;
      end
    end

west :
  begin
    if (count==3'b111)
      begin
      state=west_y;
```

```
                count=3'b0;
                end
              else
                begin
                count=count+3'b001;
                state=west;
                end
              end

          west_y :
            begin
              if (count==3'b011)
                begin
                state=north;
                count=3'b0;
                end
              else
                begin
                count=count+3'b001;
                state=west_y;
                end
              end
          endcase // case (state)
        end // always @ (state)
      end


    always @(state)
       begin
         case (state)
          north :
            begin
              n_lights = 3'b001;
              s_lights = 3'b100;
              e_lights = 3'b100;
              w_lights = 3'b100;
            end // case: north

          north_y :
            begin
              n_lights = 3'b010;
              s_lights = 3'b100;
              e_lights = 3'b100;
              w_lights = 3'b100;
            end // case: north_y

          south :
```

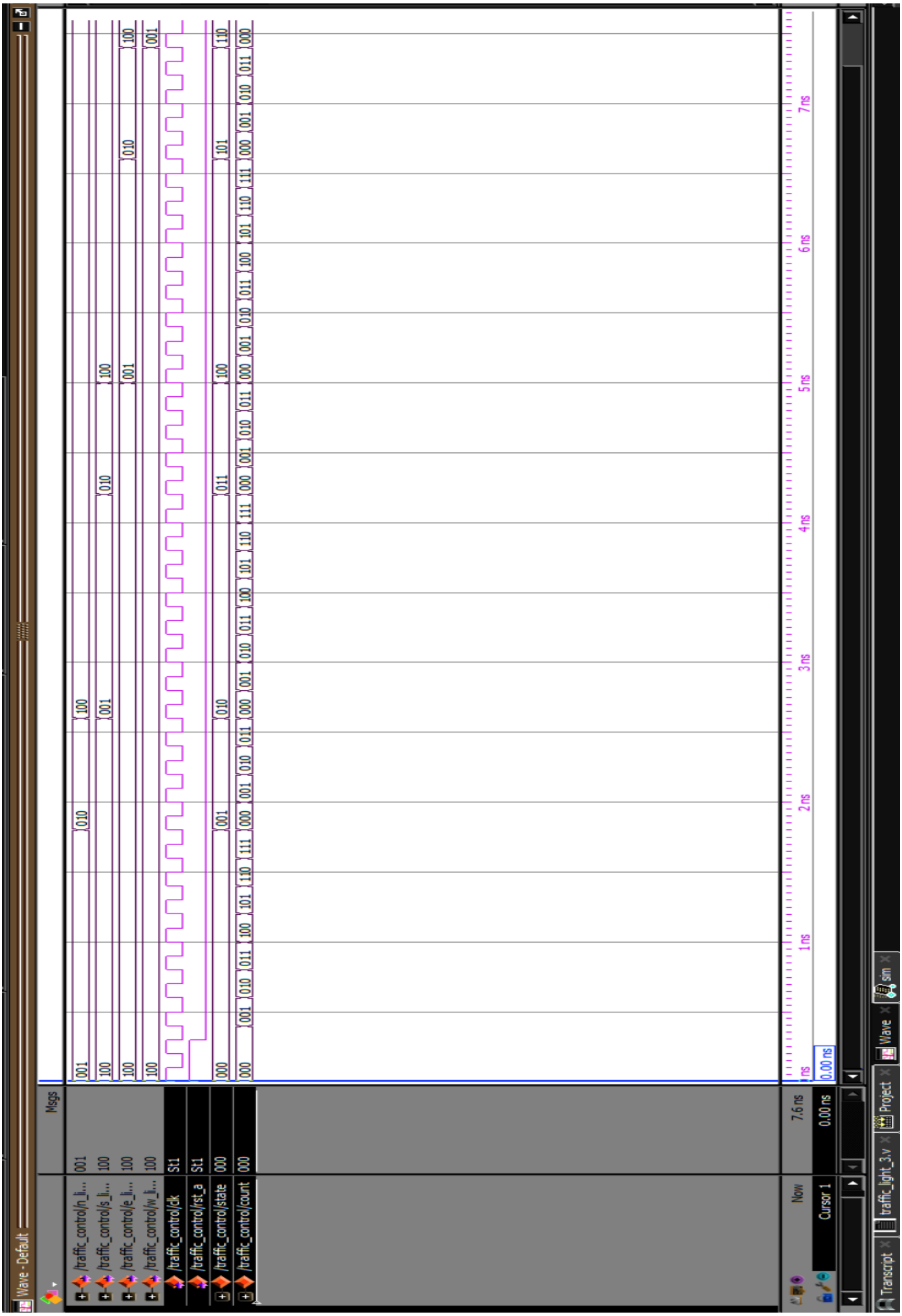
```
        begin
          n_lights = 3'b100;
          s_lights = 3'b001;
          e_lights = 3'b100;
          w_lights = 3'b100;
        end // case: south

      south_y :
        begin
          n_lights = 3'b100;
          s_lights = 3'b010;
          e_lights = 3'b100;
          w_lights = 3'b100;
        end // case: south_y

      west :
        begin
          n_lights = 3'b100;
          s_lights = 3'b100;
          e_lights = 3'b100;
          w_lights = 3'b001;
        end // case: west

      west_y :
        begin
          n_lights = 3'b100;
          s_lights = 3'b100;
          e_lights = 3'b100;
          w_lights = 3'b010;
        end // case: west_y
      east :
        begin
          n_lights = 3'b100;
          s_lights = 3'b100;
          e_lights = 3'b001;
          w_lights = 3'b100;
        end // case: east
      east_y :
        begin
          n_lights = 3'b100;
          s_lights = 3'b100;
          e_lights = 3'b010;
          w_lights = 3'b100;
        end // case: east_y
      endcase // case (state)
  end // always @ (state)
endmodule
```

OUTPUT:


Wave - Default
Msgs
/traffic_control/n_li... 001
/traffic_control/s_li... 100
/traffic_control/e_li... 100
/traffic_control/w_li... 100
/traffic_control/clk St1
/traffic_control/rst_a St1
/traffic_control/state 000
/traffic_control/count 000
Now 7.6 ns
Cursor 1 0.00 ns
0.00 ns
1 ns
2 ns
3 ns
4 ns
5 ns
6 ns
7 ns
Transcript
traffic_light_3.v
Project
Wave
sim