```
module Light(clk, reset, is_center, l_in, r_in, l_on, r_on, out);
   input logic    clk, reset, is_center, l_in, r_in, l_on, r_on;
   logic [3:0]    in;
   assign in[0] = r_on;
   assign in[1] = l_on;
   assign in[2] = r_in;
   assign in[3] = l_in;
   output logic   out;

   typedef enum logic {ON, OFF} state;
   state ps, ns;

   always_comb begin
      case (ps)
         ON:   if ((in == 4'b0100) | (in == 4'b1000)) ns = OFF;
               else                                   ns = ON;
         OFF:  if ((in == 4'b0110) | (in == 4'b1001)) ns = ON;
               else                                   ns = OFF;
      endcase
   end

   assign out = (ps == ON);

   always_ff @(posedge clk) begin
      if (reset)
         if (is_center) ps <= ON;
         else           ps <= OFF;
      else
         ps <= ns;
   end

endmodule

module CenterLight_testbench();

   logic       clk, reset, is_center, l_in, r_in, l_on, r_on;
   logic       out;

   assign is_center = 1;

   Light dut(clk, reset, is_center, l_in, r_in, l_on, r_on, out);

   parameter CLOCK_PERIOD=100;
   initial begin
      clk <= 0;
      forever #(CLOCK_PERIOD/2) clk <= ~clk;
   end

   initial begin                                                    @(posedge clk);
         reset <= 1;                                                @(posedge clk);
         reset <= 0; l_in <= 0; r_in <= 0; l_on <= 0; r_on <= 0;    @(posedge clk);
                                                                    @(posedge clk);
                                                                    @(posedge clk);
         r_in <= 1;                                                 @(posedge clk);
         r_in <= 0; r_on <= 1;                                      @(posedge clk);
                                                                    @(posedge clk);
                                                                    @(posedge clk);
         r_in <= 1;                                                 @(posedge clk);
         r_in <= 0; r_on <= 0;                                      @(posedge clk);
                                                                    @(posedge clk);
                                                                    @(posedge clk);
         l_in <= 1;                                                 @(posedge clk);
         l_in <= 0; r_on <= 1;                                      @(posedge clk);
                                                                    @(posedge clk);
                                                                    @(posedge clk);
         l_in <= 1;                                                 @(posedge clk);
         l_in <= 0; r_on <= 0;                                      @(posedge clk);
                                                                    @(posedge clk);
                                                                    @(posedge clk);
         l_in <= 1;                                                 @(posedge clk);
         l_in <= 0; l_on <= 1;                                      @(posedge clk);
                                                                    @(posedge clk);
                                                                    @(posedge clk);
         l_in <= 1;                                                 @(posedge clk);
```

```
75            l_in <= 0; l_on <= 0;                                          @(posedge clk);
76                                                                           @(posedge clk);
77                                                                           @(posedge clk);
78            r_in <= 1;                                                     @(posedge clk);
79            r_in <= 0; l_on <= 1;                                          @(posedge clk);
80                                                                           @(posedge clk);
81                                                                           @(posedge clk);
82            r_in <= 1;                                                     @(posedge clk);
83            r_in <= 0; l_on <= 0;                                          @(posedge clk);
84                                                                           @(posedge clk);
85                                                                           @(posedge clk);
86            reset <= 1;                                                    @(posedge clk);
87            reset <= 0;                                                    @(posedge clk);
88         $stop;
89      end
90   endmodule
91
92   module NormalLight_testbench();
93
94      logic       clk, reset, is_center, l_in, r_in, l_on, r_on;
95      logic       out;
96
97      assign is_center = 0;
98
99      Light dut(clk, reset, is_center, l_in, r_in, l_on, r_on, out);
100
101     parameter CLOCK_PERIOD=100;
102     initial begin
103        clk <= 0;
104        forever #(CLOCK_PERIOD/2) clk <= ~clk;
105     end
106
107     initial begin                                 @(posedge clk);
108           reset <= 1;                                                    @(posedge clk);
109           reset <= 0; l_in <= 0; r_in <= 0; l_on <= 0; r_on <= 0;        @(posedge clk);
110                                                                          @(posedge clk);
111                                                                          @(posedge clk);
112           r_in <= 1;                                                     @(posedge clk);
113           r_in <= 0; l_on <= 1;                                          @(posedge clk);
114                                                                          @(posedge clk);
115                                                                          @(posedge clk);
116           r_in <= 1;                                                     @(posedge clk);
117           r_in <= 0; l_on <= 0;                                          @(posedge clk);
118                                                                          @(posedge clk);
119                                                                          @(posedge clk);
120           r_in <= 1;                                                     @(posedge clk);
121           r_in <= 0; r_on <= 1;                                          @(posedge clk);
122                                                                          @(posedge clk);
123                                                                          @(posedge clk);
124           r_in <= 1;                                                     @(posedge clk);
125           r_in <= 0; r_on <= 0;                                          @(posedge clk);
126                                                                          @(posedge clk);
127                                                                          @(posedge clk);
128           l_in <= 1;                                                     @(posedge clk);
129           l_in <= 0; r_on <= 1;                                          @(posedge clk);
130                                                                          @(posedge clk);
131                                                                          @(posedge clk);
132           l_in <= 1;                                                     @(posedge clk);
133           l_in <= 0; r_on <= 0;                                          @(posedge clk);
134                                                                          @(posedge clk);
135                                                                          @(posedge clk);
136           l_in <= 1;                                                     @(posedge clk);
137           l_in <= 0; l_on <= 1;                                          @(posedge clk);
138                                                                          @(posedge clk);
139                                                                          @(posedge clk);
140           l_in <= 1;                                                     @(posedge clk);
141           l_in <= 0; l_on <= 0;                                          @(posedge clk);
142                                                                          @(posedge clk);
143                                                                          @(posedge clk);
144           reset <= 1;                                                    @(posedge clk);
145           reset <= 0;                                                    @(posedge clk);
146        $stop;
147     end
148  endmodule
```