```
1   module DE1_SoC (CLOCK_50, HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, KEY, LEDR, SW);
2      input  logic CLOCK_50; // 50MHz clock.
3      output logic [6:0] HEX0, HEX1, HEX2, HEX3, HEX4, HEX5;
4
5      output logic [9:0] LEDR;
6      input  logic [3:0] KEY; // True when not pressed, False when pressed
7      input  logic [9:0] SW;
8
9      // Hook up FSM inputs and outputs.
10     logic  reset;
11     assign reset = SW[9]; // Reset when SW[9] is switched.
12
13     logic BL, BR;
14     logic [9:1] out;
15     logic [1:0] keys1, keys2;
16
17     always_ff @(posedge CLOCK_50) begin
18        keys1[0] <= KEY[0];
19        keys1[1] <= KEY[3];
20     end
21
22     always_ff @(posedge CLOCK_50) begin
23        keys2 <= keys1;
24     end
25
26     UserIn Left    (.clk(CLOCK_50), .reset, .key(keys2[1]), .out(BL));
27     UserIn Right   (.clk(CLOCK_50), .reset, .key(keys2[0]), .out(BR));
28
29     Light one   (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[2]),
      .r_on(0),      .out(out[1]));
30     Light two   (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[3]),
      .r_on(out[1]), .out(out[2]));
31     Light three (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[4]),
      .r_on(out[2]), .out(out[3]));
32     Light four  (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[5]),
      .r_on(out[3]), .out(out[4]));
33     Light five  (.clk(CLOCK_50), .reset, .is_center(1), .l_in(BL), .r_in(BR), .l_on(out[6]),
      .r_on(out[4]), .out(out[5]));
34     Light six   (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[7]),
      .r_on(out[5]), .out(out[6]));
35     Light seven (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[8]),
      .r_on(out[6]), .out(out[7]));
36     Light eight (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(out[9]),
      .r_on(out[7]), .out(out[8]));
37     Light nine  (.clk(CLOCK_50), .reset, .is_center(0), .l_in(BL), .r_in(BR), .l_on(0),
          .r_on(out[8]), .out(out[9]));
38
39
40     assign LEDR[9:1] = out[9:1];
41
42     Victory V (.clk(CLOCK_50), .reset, .l_in(BL), .r_in(BR), .l_on(out[9]), .r_on(out[1]), .
   out(HEX0));
43
44     // Show signals on LEDRs so we can see what is happening.
45     assign LEDR[0] = reset;
46
47  endmodule
48
49  module Tug_Testbench();
50     logic CLOCK_50;
51     logic [6:0] HEX0, HEX1, HEX2, HEX3, HEX4, HEX5;
52     logic [3:0] KEY;
53     logic [9:0] LEDR;
54     logic [9:0] SW;
55
56     DE1_SoC dut(CLOCK_50, HEX0, HEX1, HEX2, HEX3, HEX4, HEX5, KEY, LEDR, SW);
57
58     parameter CLOCK_PERIOD = 100;
59     initial begin
60        CLOCK_50 <= 0;
61        forever #(CLOCK_PERIOD/2) CLOCK_50 <= ~CLOCK_50;
62     end
63
64     initial begin                                                    @(posedge CLOCK_50);
```

```
65          SW[9] <= 1;                                                 @(posedge CLOCK_50);
66          SW<= 0; KEY <= 0;                                           @(posedge CLOCK_50);
67                                                                      @(posedge CLOCK_50);
68          KEY[3] <= 1;                                                @(posedge CLOCK_50);
69          KEY[3] <= 0;                                                @(posedge CLOCK_50);
70                                                                      @(posedge CLOCK_50);
71          KEY[3] <= 1;                                                @(posedge CLOCK_50);
72          KEY[3] <= 0;                                                @(posedge CLOCK_50);
73                                                                      @(posedge CLOCK_50);
74          KEY[3] <= 1;                                                @(posedge CLOCK_50);
75          KEY[3] <= 0;                                                @(posedge CLOCK_50);
76                                                                      @(posedge CLOCK_50);
77          KEY[3] <= 1;                                                @(posedge CLOCK_50);
78          KEY[3] <= 0;                                                @(posedge CLOCK_50);
79                                                                      @(posedge CLOCK_50);
80          KEY[3] <= 1;                                                @(posedge CLOCK_50);
81          KEY[3] <= 0;                                                @(posedge CLOCK_50);
82                                                                      @(posedge CLOCK_50);
83          KEY[3] <= 1;                                                @(posedge CLOCK_50);
84          KEY[3] <= 0;                                                @(posedge CLOCK_50);
85                                                                      @(posedge CLOCK_50);
86          KEY[0] <= 1;                                                @(posedge CLOCK_50);
87          KEY[0] <= 0;                                                @(posedge CLOCK_50);
88                                                                      @(posedge CLOCK_50);
89          SW[9] <= 1;                                                 @(posedge CLOCK_50);
90          SW<= 0; KEY <= 0;                                           @(posedge CLOCK_50);
91                                                                      @(posedge CLOCK_50);
92          KEY[0] <= 1;                                                @(posedge CLOCK_50);
93          KEY[0] <= 0;                                                @(posedge CLOCK_50);
94                                                                      @(posedge CLOCK_50);
95          KEY[0] <= 1;                                                @(posedge CLOCK_50);
96          KEY[0] <= 0;                                                @(posedge CLOCK_50);
97                                                                      @(posedge CLOCK_50);
98          KEY[0] <= 1;                                                @(posedge CLOCK_50);
99          KEY[0] <= 0;                                                @(posedge CLOCK_50);
100                                                                     @(posedge CLOCK_50);
101         KEY[0] <= 1;                                                @(posedge CLOCK_50);
102         KEY[0] <= 0;                                                @(posedge CLOCK_50);
103                                                                     @(posedge CLOCK_50);
104         KEY[0] <= 1;                                                @(posedge CLOCK_50);
105         KEY[0] <= 0;                                                @(posedge CLOCK_50);
106                                                                     @(posedge CLOCK_50);
107         KEY[0] <= 1;                                                @(posedge CLOCK_50);
108         KEY[0] <= 0;                                                @(posedge CLOCK_50);
109                                                                     @(posedge CLOCK_50);
110         KEY[3] <= 1;                                                @(posedge CLOCK_50);
111         KEY[3] <= 0;                                                @(posedge CLOCK_50);
112                                                                     @(posedge CLOCK_50);
113         SW[9] <= 1;                                                 @(posedge CLOCK_50);
114         SW<= 0; KEY <= 0;                                           @(posedge CLOCK_50);
115                                                                     @(posedge CLOCK_50);
116                                                                     @(posedge CLOCK_50);
117         KEY[0] <= 1; KEY[3] <= 1;                                   @(posedge CLOCK_50);
118         KEY[0] <= 0;                                                @(posedge CLOCK_50);
119                                                                     @(posedge CLOCK_50);
120         KEY[0] <= 1; KEY[3] <= 0;                                   @(posedge CLOCK_50);
121         KEY[0] <= 0;                                                @(posedge CLOCK_50);
122                                                                     @(posedge CLOCK_50);
123         KEY[0] <= 1;                                                @(posedge CLOCK_50);
124         KEY[0] <= 0;                                                @(posedge CLOCK_50);
125                                                                     @(posedge CLOCK_50);
126         KEY[3] <= 1;                                                @(posedge CLOCK_50);
127         KEY[3] <= 0;                                                @(posedge CLOCK_50);
128                                                                     @(posedge CLOCK_50);
129         KEY[3] <= 1; KEY[0] <= 1;                                   @(posedge CLOCK_50);
130         KEY[3] <= 0; KEY[0] <= 0;                                   @(posedge CLOCK_50);
131                                                                     @(posedge CLOCK_50);
132         KEY[0] <= 1;                                                @(posedge CLOCK_50);
133         KEY[0] <= 0;                                                @(posedge CLOCK_50);
134                                                                     @(posedge CLOCK_50);
135         KEY[3] <= 1;                                                @(posedge CLOCK_50);
136         KEY[3] <= 0;                                                @(posedge CLOCK_50);
137                                                                     @(posedge CLOCK_50);
138         $stop;
```

```
139       end
140   endmodule
```