

```

1  // Eugene Ngo
2  // 12/2/2022
3  // EE 271
4  // Lab 5
5
6  // userInput module takes clk, reset, and button as 1-Bit inputs and returns 1-Bit out
7  // This module instatiates the buttons for user input.
8
9  module userInput(out, clk, reset, button);
10     output logic out;
11     input logic clk, reset, button;
12
13     enum {on, off} ps, ns;
14
15     always_comb begin
16         case(ps)
17             on:   if(button) ns = on;
18                  else ns = off;
19
20             off:  if(button) ns = on;
21                  else ns = off;
22
23         endcase
24     end
25
26     // same as Lab 6 behavior
27     assign out = (ps == on & ns == off);
28
29     always_ff @(posedge clk) begin
30         if(reset)
31             ps <= off;
32         else
33             ps <= ns;
34     end
35 endmodule
36
37 // userInput_testbench tests all expected, unexpected, and edgecase behaviors
38
39 module userInput_testbench();
40     logic clk, reset;
41     logic button;
42     logic out;
43
44     userInput dut (.out, .clk, .reset, .button);
45
46     parameter CLOCK_PERIOD = 100;
47     initial begin
48         clk <= 0;
49         forever #(CLOCK_PERIOD / 2)
50             clk <= ~clk;
51     end
52
53     initial begin
54         reset <= 1;          @(posedge clk);
55                             @(posedge clk);
56         reset <= 0;          @(posedge clk);
57                             @(posedge clk);
58         button <= 1;         @(posedge clk);
59                             @(posedge clk);
60                             @(posedge clk);
61         button <= 0;         @(posedge clk);
62                             @(posedge clk);
63                             @(posedge clk);
64         button <= 1;         @(posedge clk);
65                             @(posedge clk);
66                             @(posedge clk);
67         button <= 0;         @(posedge clk);
68                             @(posedge clk);
69                             @(posedge clk);
70         button <= 1;         @(posedge clk);
71                             @(posedge clk);
72                             @(posedge clk);
73         button <= 0;         @(posedge clk);

```

```
74      @(posedge clk);
75      @(posedge clk);
76      button <= 1;
77      @(posedge clk);
78      $stop;
79      end
80  endmodule
```