

```

1  module UserInput(clk, reset, key, out);
2
3      input logic clk, reset, key;
4      output logic out;
5
6      enum {on, off} ps, ns;
7
8      always_comb begin
9          case(ps)
10             on:      if(key) ns = on;
11                     else ns = off;
12
13             off: if(key) ns = on;
14                  else ns = off;
15          endcase
16      end
17
18      assign out = (ps == off & key == 1);
19
20      always_ff @(posedge clk) begin
21          if(reset)
22              ps <= off;
23          else
24              ps <= ns;
25      end
26  endmodule
27
28  module UserInput_testbench();
29      logic clk, reset, key;
30      logic out;
31
32      UserInput dut (.clk, .reset, .key, .out);
33
34      parameter CLOCK_PERIOD = 100;
35      initial begin
36          clk <= 0;
37          forever #(CLOCK_PERIOD / 2)
38              clk <= ~clk;
39      end
40
41      initial begin
42          reset <= 1;      repeat(2) @(posedge clk);
43          reset <= 0;      repeat(2) @(posedge clk);
44          key <= 1;        repeat(3) @(posedge clk);
45          key <= 0;        repeat(3) @(posedge clk);
46          key <= 1;        repeat(3) @(posedge clk);
47          key <= 0;        repeat(3) @(posedge clk);
48          key <= 1;        repeat(3) @(posedge clk);
49          key <= 0;        repeat(3) @(posedge clk);
50          key <= 1;        repeat(3) @(posedge clk);
51          $stop;
52      end
53  endmodule

```