

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