```
1   /* Name: Eugene Ngo
2      Date: 3/7/2023
3      Class: EE 371
4      Lab 6: Parking Lot 3D Simulation
5   */
6
7   // hourCount takes 2 inputs (reset, incr) and outputs
8   // the hours counted thus far to progress the day in the parking lot system
9   `timescale 1 ps / 1 ps
10  module hourCount (inc, clk, reset, out);
11
12     input logic inc, clk, reset;
13     output logic [3:0]  out;
14
15     // Sequential logic for counting up and counting down depending on the input.
16     always_ff @(posedge clk) begin
17        if (reset) begin
18           out <= 4'b0000;
19        end
20        else if (inc & out < 4'b1000) begin //increment when not at max
21           out <= out + 4'b0001;
22        end
23        else
24           out <= out; // hold value otherwise
25     end // always_ff
26
27  endmodule
28
29  // hourCount_testbench tests all expected, unexpected, and edgecase behaviors
30  module hourCount_testbench();
31     logic inc, clk, reset;
32     logic [3:0] out;
33     logic CLOCK_50;
34
35     hourCount dut (.inc, .clk(CLOCK_50), .reset, .out);
36
37     // Setting up the clock.
38     parameter CLOCK_PERIOD = 100;
39     initial begin
40        CLOCK_50 <= 0;
41        forever #(CLOCK_PERIOD/2) CLOCK_50 <= ~CLOCK_50; // toggle the clock forever
42     end // initial
43
44     initial begin
45        reset <= 1;                @(posedge CLOCK_50); // reset
46        reset <= 0;                @(posedge CLOCK_50); // inc past max limit
47        inc <= 0;                  repeat(7) @(posedge CLOCK_50); // dec past min limit
48        inc <= 1;                  repeat(7) @(posedge CLOCK_50); // dec past min limit
49        $stop;
50     end
51  endmodule // counter_testbench
```