```
1   //ee417 Lesson 4 Assignment 3 Part 2, L4A3P2
2   //Name: Lamin Jammeh, Date: 06-09-24, Group: Kalin/Jammeh
3   //Testbench for BCD_or_2421_up_down_counter using Moore FSM with case structure
4   //can select down or up counting for 2421code & also select (2421code or BCD) counting
5   //Step1 define test bench name
6   module BCD_or_2421_up_down_counter_tb();
7   /* original module declaration
8   module BCD_or_2421_up_down_counter(output [3:0] count_out,
9                                      input clk,
10                                     input reset,
11                                     input enable,
12                                     input select_up_down,
13                                     input select_code2421_BCD );*/
14  //Step2 define inputs as registers, outputs as wires
15  reg clk, rst,enable, select_up_down, select_code2421_BCD;
16  wire [3:0] count_out;
17  //internal probe wires: observe change in state..this gives Questa error if not enough
    digits
18  wire [3:0] downCount2421;
19  wire [3:0] upCount2421;
20  wire [3:0] count2421code;
21  wire [3:0] countBCD;
22  //wire [2:0] state, next_state;
23
24  //Step3 define unit under test
25  BCD_or_2421_up_down_counter UUT (count_out,clk,rst,enable,select_up_down,
26                                   select_code2421_BCD);
27
28  //internal probes to track logic and troubleshoot
29  assign downCount2421= UUT.downCount2421;
30  assign upCount2421= UUT.upCount2421;
31  assign count2421code= UUT.count2421code;
32  assign countBCD= UUT.countBCD;
33  //assign next_state= UUT.next_state;
34
35  //Step4 open initial block, define all possible input combinations
36  // Clock generation (adjust the period as needed)
37  initial begin
38    clk=0;
39    forever
40    #5 clk = ~clk;
41  end
42
43  initial  //reset is active high, longer time to count when reset is inactive (low)
44   begin   //4 cases with two selects
45    rst  = 1'b1; //reset on
46    # 10 rst = 1'b0; enable=1'b1; select_up_down=1'b0; select_code2421_BCD=1'b0;
47    //reset off, enable on, select down, select 2421
48    #120 rst = 1'b1;  //reset on
49    # 20 rst = 1'b0; enable=1'b1; //reset off, enable on
50    # 60 enable=1'b0;  //enable off: freeze count
51
52      # 10 rst = 1'b0; enable=1'b1; select_up_down=1'b0; select_code2421_BCD=1'b1;
53    //reset off, enable on, select down, select BCD
54    #120 rst = 1'b1;  //reset on
55    # 20 rst = 1'b0; enable=1'b1; //reset off, enable on
56    # 60 enable=1'b0;  //enable off: freeze count
57
58    # 10 rst = 1'b0; enable=1'b1; select_up_down=1'b1; select_code2421_BCD=1'b0;
59    //reset off, enable on, select up, select 2421
60    #120 rst = 1'b1;  //reset on
61    # 20 rst = 1'b0; enable=1'b1; //reset off, enable on
62    # 60 enable=1'b0;  //enable off: freeze count
63
64    # 10 rst = 1'b0; enable=1'b1; select_up_down=1'b1; select_code2421_BCD=1'b1;
65    //reset off, enable on, select up, select BCD
66    #120 rst = 1'b1;  //reset on
67    # 20 rst = 1'b0; enable=1'b1; //reset off, enable on
68    # 60 enable=1'b0;  //enable off: freeze count
```

```
69
70      #100 $stop; //close debug window to view waveform viewer
71     end
72
73    //Step5 Display the results
74    initial begin //monitor counter value
75      $display("_____output_count_out = -count-");
76      $monitor("clk_in = %b: rst_in = %b:  enable_in=%b: select_up_down=%b:
      select_code2421_BCD=%b: output_count_out = %d ",
77      clk, rst, enable, select_up_down, select_code2421_BCD, count_out);
78     end
79    endmodule
80
```