Q.Simulate the UPDOWN counter counting 4 twice while down count using SV (enum datatype.

// System Verilog code…

```
module updown_count_4twice(
   input  logic       clk,
   input  logic       rst,
   input  logic       up_down,     // 1 = up, 0 = down
   output logic [3:0]  count
);
   always_ff @(posedge clk or posedge rst) begin
      if (rst)
         count <= 4'd0;
      else if (up_down)
         count <= count + 1;
      else
         count <= count - 1;
   end
endmodule
```

//Testbench for given code…

```
`timescale 1ns / 1ps
module tb_up_down_counter;
   // Clock and control signals
   logic clk;
   logic rst;
   logic up_down;
   logic [3:0] count;
```

```
// Enum for direction
typedef enum logic {
    DOWN = 1'b0,
    UP   = 1'b1
} direction_t;
direction_t dir;
// Instantiate the DUT
up_down_counter uut (
    .clk(clk),
    .rst(rst),
    .up_down(dir),
    .count(count)
);
// Clock generation (10ns period)
always #5 clk = ~clk;
// Main test process
initial begin
    $display("Starting simulation...");
    $dumpfile("counter.vcd");     // For GTKWave (optional)
    $dumpvars(0, tb_up_down_counter);
    // Init
    clk = 0;
    rst = 1;
    dir = UP;
    #12 rst = 0;         // Deassert reset
    #10;
    // Count up for 8 cycles
    dir = UP;
```

```
        repeat (8) @(posedge clk);
        // Count down and observe value 4 twice
        dir = DOWN;
        int count_4_seen = 0;
        while (count_4_seen < 2) begin
            @(posedge clk);
            if (count == 4)
                count_4_seen++;
        end
        $display("Observed value 4 twice during down count.");
        #20;
        $finish;
    end
    // Monitor values
    initial begin
        $monitor("Time=%0t | rst=%b | dir=%s | count=%0d",
                 $time, rst, (dir == UP) ? "UP  " : "DOWN", count);
    end
endmodule
```