Assignment – 5

Implementation of Up/Down Counter for 4bits

```
//Verilog Code
module up_down_counter (
    input wire clk,
    input wire reset,
    input wire up_down,
    output reg [3:0] count);
always @(posedge clk) begin
    if (reset)
        count <= 4'b0000;
    else if (up_down)
        count <= count + 1;
    else
        count <= count - 1;
end
endmodule

//TestBench
timescale 1ns / 1ps
module tb_up_down_counter;
    reg clk;
    reg reset;
    reg up_down;
    wire [3:0] count;
    up_down_counter uut (
        .clk(clk),
```

```
        .reset(reset),
        .up_down(up_down),
        .count(count)
    );
    always #5 clk = ~clk;
    initial begin
        $dumpfile("up_down_counter.vcd");
        $dumpvars(0, tb_up_down_counter);
        clk = 0;
        reset = 1;
        up_down = 1;
        #10 reset = 0;
        #50 up_down = 0;
        #40 up_down = 1;
        #30 reset = 1;
        #10 reset = 0;
        #20 $finish;
    end
endmodule
```