Digital Design and Computer Organization Laboratory

3rd Semester, Academic Year 2024

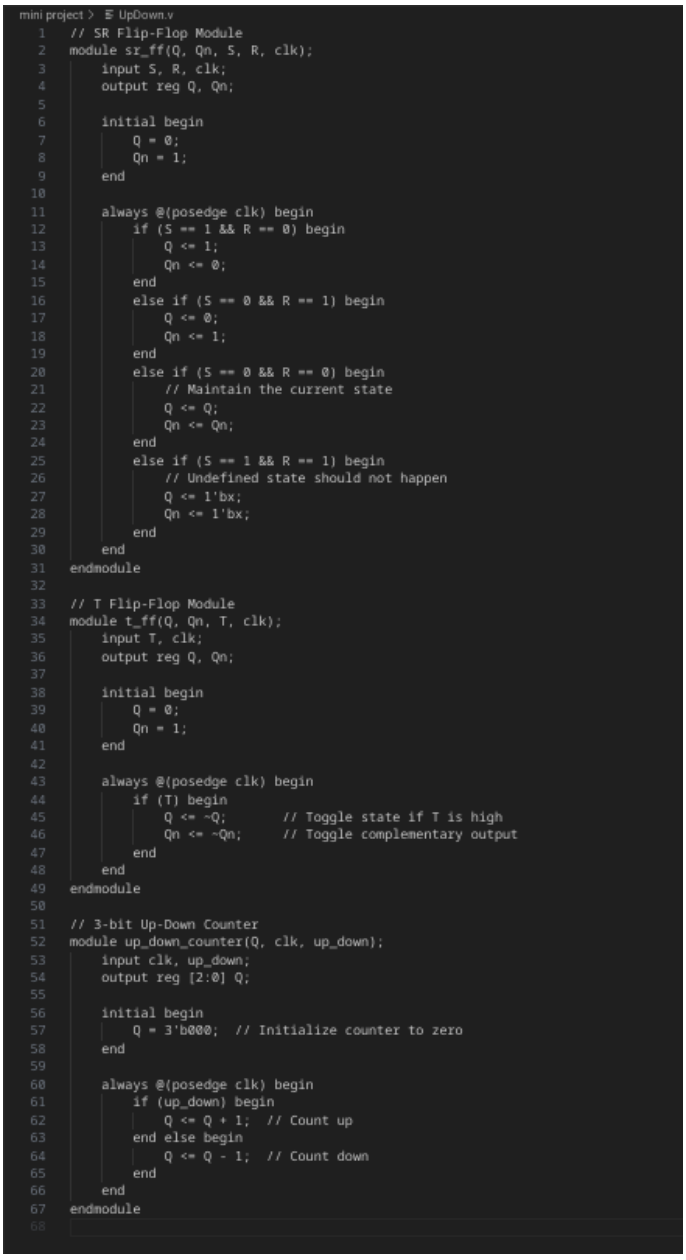
Date: 20/10/2024

Team members:	SRN:	Section: H
Rithvik Rajesh Matta	PES2UG23CS485	
Rishi A Sheth	PES2UG23CS479	
Rohan	PES2UG23CS489	
Ritesh M	PES2UG23CS483	

Aim of the Experiment:

TO DESIGN AND IMPLEMENT A 3-BIT UP/DOWN COUNTER THAT INCREMENTS OR DECREMENTS ITS COUNT VALUE BASED ON CONTROL SIGNALS, FOLLOWED BY GENERATING THE VERILOG VVP OUTPUT AND SIMULATION WAVEFORM USING GTK WAVE, AND VERIFYING THE OUTPUT AND WAVEFORM AGAINST THE TRUTH TABLE.

I. Verilog Code Screenshot



mini project > UpDown.v

```
// SR Flip-Flop Module
module sr_ff(Q, Qn, S, R, clk);
    input S, R, clk;
    output reg Q, Qn;

    initial begin
        Q = 0;
        Qn = 1;
    end

    always @(posedge clk) begin
        if (S == 1 && R == 0) begin
            Q <= 1;
            Qn <= 0;
        end
        else if (S == 0 && R == 1) begin
            Q <= 0;
            Qn <= 1;
        end
        else if (S == 0 && R == 0) begin
            // Maintain the current state
            Q <= Q;
            Qn <= Qn;
        end
        else if (S == 1 && R == 1) begin
            // Undefined state should not happen
            Q <= 1'bx;
            Qn <= 1'bx;
        end
    end
endmodule

// T Flip-Flop Module
module t_ff(Q, Qn, T, clk);
    input T, clk;
    output reg Q, Qn;

    initial begin
        Q = 0;
        Qn = 1;
    end

    always @(posedge clk) begin
        if (T) begin
            Q <= ~Q;        // Toggle state if T is high
            Qn <= ~Qn;      // Toggle complementary output
        end
    end
endmodule

// 3-bit Up-Down Counter
module up_down_counter(Q, clk, up_down);
    input clk, up_down;
    output reg [2:0] Q;

    initial begin
        Q = 3'b000;  // Initialize counter to zero
    end

    always @(posedge clk) begin
        if (up_down) begin
            Q <= Q + 1;  // Count up
        end else begin
            Q <= Q - 1;  // Count down
        end
    end
endmodule
```

mini project > UpDown_tb.v

```
`timescale 1ns / 1ps

module tb_up_down_counter;

// Inputs
reg clk;
reg up_down;

// Outputs
wire [2:0] Q;

// Instantiate the Unit Under Test (UUT)
up_down_counter uut (
    .Q(Q),
    .clk(clk),
    .up_down(up_down)
);

// Clock Generation: Toggle every 10 ns
initial begin
    clk = 0;
    forever #5 clk = ~clk;  // Toggle clock every 5 ns (10 ns period)
end

// Test stimulus
initial begin
    // Initialize Inputs
    up_down = 1; // Start by counting up

    // Print the state every clock cycle
    forever begin
        @(posedge clk);  // Wait for the rising edge of clk
        $display("At time %t: clk = %b, up_down = %b, Q = %b", $time, clk, up_down, Q);
    end
end

// Additional stimulus
initial begin
    // Run for some time with 'up_down = 1' (UP)
    #80;  // Run for 80 ns (8 clock cycles)

    // Change to 'up_down = 0' (DOWN)
    up_down = 0;

    // Run for another 80 ns (8 clock cycles)
    #80;

    // Finish the simulation
    $finish;
end

// Generate VCD file for GTKWave
initial begin
    $dumpfile("up_down_counter.vcd"); // VCD output file
    $dumpvars(0, tb_up_down_counter);  // Dump all variables
end

endmodule
```

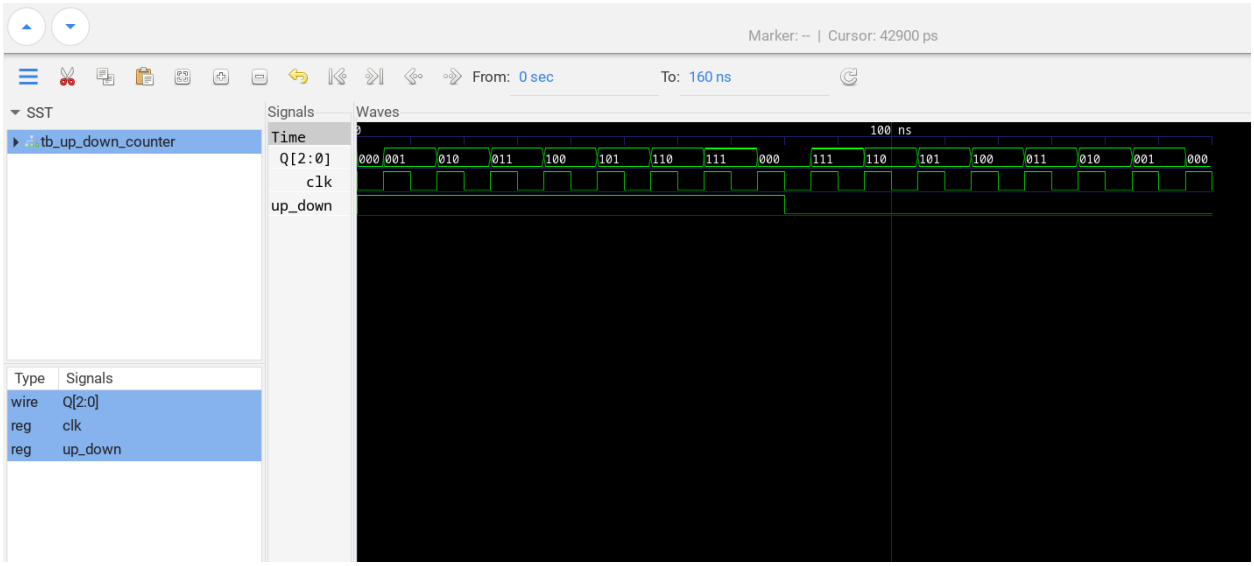
II. Verilog VVP Output Screen Shot

```
rithvikmatta@penguin:~/PES/DDCO-sem-3/mini project$ iverilog -o proj UpDown.v 'UpDown_tb.v'
rithvikmatta@penguin:~/PES/DDCO-sem-3/mini project$ vvp proj
VCD info: dumpfile up_down_counter.vcd opened for output.
At time                 5000: clk = 1, up_down = 1, Q = 000
At time                15000: clk = 1, up_down = 1, Q = 001
At time                25000: clk = 1, up_down = 1, Q = 010
At time                35000: clk = 1, up_down = 1, Q = 011
At time                45000: clk = 1, up_down = 1, Q = 100
At time                55000: clk = 1, up_down = 1, Q = 101
At time                65000: clk = 1, up_down = 1, Q = 110
At time                75000: clk = 1, up_down = 1, Q = 111
At time                85000: clk = 1, up_down = 0, Q = 000
At time                95000: clk = 1, up_down = 0, Q = 111
At time               105000: clk = 1, up_down = 0, Q = 110
At time               115000: clk = 1, up_down = 0, Q = 101
At time               125000: clk = 1, up_down = 0, Q = 100
At time               135000: clk = 1, up_down = 0, Q = 011
At time               145000: clk = 1, up_down = 0, Q = 010
At time               155000: clk = 1, up_down = 0, Q = 001
rithvikmatta@penguin:~/PES/DDCO-sem-3/mini project$ gtkwave up_down_counter.vcd

GTKWave Analyzer v3.3.118 (w)1999-2023 BSI

[0] start time.
[160000] end time.
GTKWAVE | Touch screen detected, enabling gestures.
Exiting.
rithvikmatta@penguin:~/PES/DDCO-sem-3/mini project$
```

III. GTKWAVE Screenshot



IV. Output Table (Truth Table)

M	Q3	Q2	Q1

Up counter

0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
0	0	0	0

Down counter

1	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	0	1	0