BITS-Pilani Dubai Campus
I Sem 2021-22
Digital Design Laboratory / ECE/INSTR/CS F215
Submission Report
Experiment No.- 10 (Counters)

Name: V N Suchir Vangaveeti **ID Number: 2020A7PS0018U**

Hardware runs

Run 1: Mod -16 Counter

Truth Table

No. of Clk pulses(n)	Count- $Q_AQ_BQ_CQ_D$	No. of Clk pulses(n)	Count- $Q_AQ_BQ_CQ_D$
0	0000	9	1001
1	0001	10	1010
2	0010	11	1011
3	0011	12	1100
4	0100	13	1101
5	0101	14	1110
6	0110	15	1111
7	0111	16	0000
8	1000	17	0001

Run 2: BCD counter (Mod 10)

Truth Table

No. of Clk pulses(n)	Count- $Q_AQ_BQ_CQ_D$	No. of Clk pulses(n)	Count- $Q_AQ_BQ_CQ_D$
0	0000	6	0110
1	0001	7	0111
2	0010	8	1000
3	0011	9	1001
4	0100	10	0000
5	0101	11	0001

Experiment No.- 10 (Counters)

Name: V N Suchir Vangaveeti **ID Number: 2020A7PS0018U**

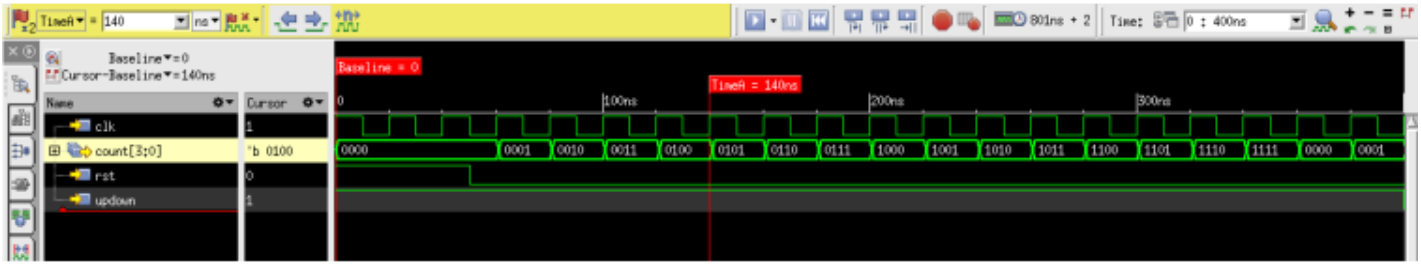
Software runs

Run 3: Write verilog code and testbench for 4-bit synchronous UP/DOWN counter.

```
module counter_1 (input clk, rst, updown, output reg [3:0] count);
        always @ (posedge clk, posedge rst) begin
        if (rst)
        count <= 4'b0000;
        else if (updown)
        count <= #1 count + 1'b1;
        else
        count <= #1 count - 1'b1;
        end
endmodule

// Test Bench
        module counter_tb;
        reg clk,rst,updown;
        wire [3:0]count;
                initial begin
                clk = 1'b1;
                repeat (80) # 10 clk = ~clk;
                $stop;
                end
                initial begin
                rst=1'b1;
                repeat (1) # 50 rst = ~rst;
                end
                initial begin
                #00 updown =1'b1;
                #400 updown =1'b0;
                end
        counter_1 ud (clk,rst,updown,count);
        endmodule
```

Q: Paste the Image of your **Simvision** window where you get the waveforms for the above code.

A:



Experiment No.- 10 (Counters)

Name: V N Suchir Vangaveeti **ID Number: 2020A7PS0018U**

Q. Complete the truth table below from the waveform observed.

Truth Table

Up/ Down	No. of Clk pulses(n)	Count [3:0]	Up/ Down	No. of Clk pulses(n)	Count [3:0]
1	0	0000	1	11	1001
1	1	0000	1	12	1010
1	2	0000	1	13	1011
1	3	0001	1	14	1100
1	4	0010	1	15	1101
1	5	0011	1	16	1110
1	6	0100	1	17	1111
1	7	0101	1	18	0000
1	8	0110	1	19	0001
1	9	0111	0	20	0000
1	10	1000	0	21	1111
1	11	1001	0	22	1110

Run 4: Write Verilog code and testbench for even counter.

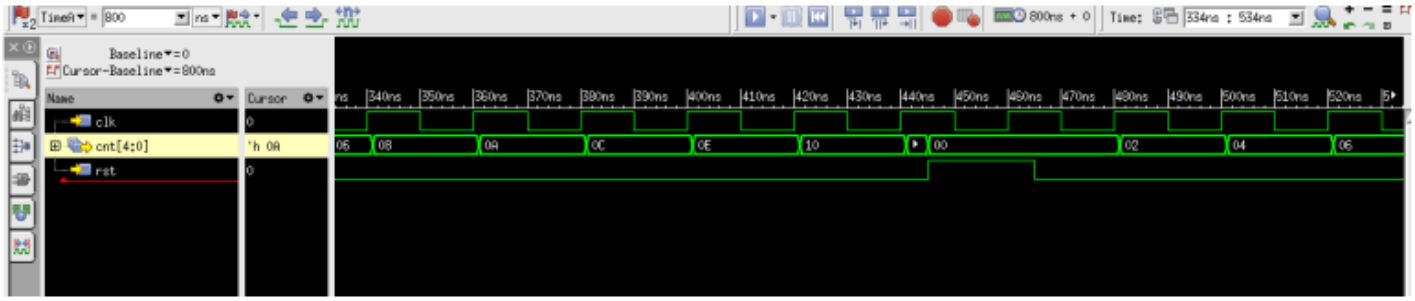
```
module counter_2 (input clk, rst, output reg [4:0] cnt);
        always @ (posedge clk, posedge rst) begin
        if (rst)
        cnt <= 5'b00000;
        else if (cnt == 5'd20)
        cnt <= #1 5'b00000;
        else
        cnt <= #1 cnt + 5'b00010;
        end
endmodule

//Test Bench
 module counter_tb;
  reg clk,rst;
  wire [4:0]count;
        initial begin
        clk = 1'b1;
        repeat (80) # 10 clk = ~clk;
        $stop;
        end
        initial begin
        rst=1'b1;
        repeat (1) # 50 rst = ~rst;
        #395 rst = 1'b1;
        #20 rst = 1'b0;
```

Experiment No.- 10 (Counters)

Name: V N Suchir Vangaveeti **ID Number: 2020A7PS0018U**

```
        end
counter_2 even (clk,rst,count);
endmodule
```

Q: Paste the Image of your **Simvision** window where you get the waveforms for the above code.
A:



Q. Complete the truth table below from the waveform observed

Truth Table

No. Of pulses(n)	rst	Count (in hexadecimal)	No. Of pulses(n)	rst	Count (in hexadecimal)
0	1	00	11	0	12
1	1	00	12	0	14
2	1	00	13	0	00
3	0	02	14	0	02
4	0	04	15	0	04
5	0	06	16	0	06
6	0	08	17	0	08
7	0	0A	18	0	0A
8	0	0C	19	0	0C
9	0	0E	20	0	0E
10	0	10	21	0	10

Experiment No.- 10 (Counters)

Name: V N Suchir Vangaveeti **ID Number: 2020A7PS0018U**

Assignment All assignments are to be submitted strictly before start of next lab session through online only. Late assignments will not be entertained and will be awarded '0' marks.

1. Below is the code for counter which counts first 5 values mod-3 values. Write the testbench for the same and show the proper output waveforms and understand the code.

```
module countseq (input clk, rst, output reg [2:0] modfive, output [1:0] cnt);
wire s1;
wire [2:0] b1, b2;
wire [1:0] b3, b4;
assign s1 = (modfive ==5);
assign b1 = modfive +1;
assign b2 = s1 ? 1 : b1;
always @ ( posedge clk, posedge rst) begin
if (rst)
modfive <= 1;
else modfive <= b2;
end
```

Ans: Link1: https://edaplayground.com/x/CFbQ

2. Write Verilog code and testbench for 4-bit ring counter.

Ans: Link2: https://edaplayground.com/x/SfTa

Self-Practice and self-evaluation

1. Write Verilog code and testbench for 4-bit gray counter.
2. Write Verilog code and testbench for 4-bit down counter.