**Name: Laiba Fatima Khan (04067)**

**CA LAB 01**

**Run**

#compiling design modules

vlog tb.v ripple\_carry\_counter.v T\_FF.v D\_FF.v

#no optimization

vsim -novopt work.tb

#view waves

view wave

#adding waves

add wave sim:/tb/clk

add wave sim:/tb/reset

add wave sim:/tb/q

run 250ns

**Test Bench**

module tb

(

);

reg clk;

reg reset;

wire[3:0] q;

ripple\_carry\_counter r1

(

.clk(clk),

.reset(reset),

.q(q)

);

initial

clk = 1'b0;

always

#5 clk = ~clk;

initial

begin

reset = 1'b1;

#15 reset = 1'b0;

#180 reset = 1'b1;

end

initial

$monitor("Time = ", $time, "---> Output = %d", q);

Endmodule

**T-FF**

module T\_FF

(

input c, r,

output q

);

wire d;

//Instantiates D\_FF

D\_FF dff0

(

.c(c),

.r(r),

.q(q),

.d(d)

);

not n1(d,q);

endmodule

**D-FF**

module D\_FF

(

input c,r,

input d,

output reg q

);

always @(posedge r or negedge c)

begin

if(r)

q<=1'b0;

else

q<=d;

end

endmodule

**Ripple Carry Counter**

module ripple\_carry\_counter

(

input clk,reset,

output[3:0] q

);

//Instantiates T-FF modules

T\_FF tff0

(

.c(clk),

.r(reset),

.q(q[0])

);

T\_FF tff1

(

.c(q[0]),

.r(reset),

.q(q[1])

);

T\_FF tff2

(

.c(q[1]),

.r(reset),

.q(q[2])

);

T\_FF tff3

(

.c(q[2]),

.r(reset),

.q(q[3])

);

endmodule

**EXERCISES**

**Exercise 1:**

To make ripple-carry-counter to count till 31, we need at least 5 bits. Hence, we change the output q of Ripple Carry Counter from 4-bit value to 5-bit value.

**RIPPLE CARRY COUNTER:**

module ripple\_carry\_counter

(

input clk,reset,

output[4:0] q

);

//Instantiates T-FF modules

T\_FF tff0

(

.c(clk),

.r(reset),

.q(q[0])

);

T\_FF tff1

(

.c(q[0]),

.r(reset),

.q(q[1])

);

T\_FF tff2

(

.c(q[1]),

.r(reset),

.q(q[2])

);

T\_FF tff3

(

.c(q[2]),

.r(reset),

.q(q[3])

);

T\_FF tff4

(

.c(q[3]),

.r(reset),

.q(q[4])

);

endmodule

**TEST BENCH:**

module tb

(

);

reg clk;

reg reset;

wire[4:0] q;

ripple\_carry\_counter r1

(

.clk(clk),

.reset(reset),

.q(q)

);

initial

clk = 1'b0;

always

#5 clk = ~clk;

initial

begin

reset = 1'b1;

#15 reset = 1'b0;

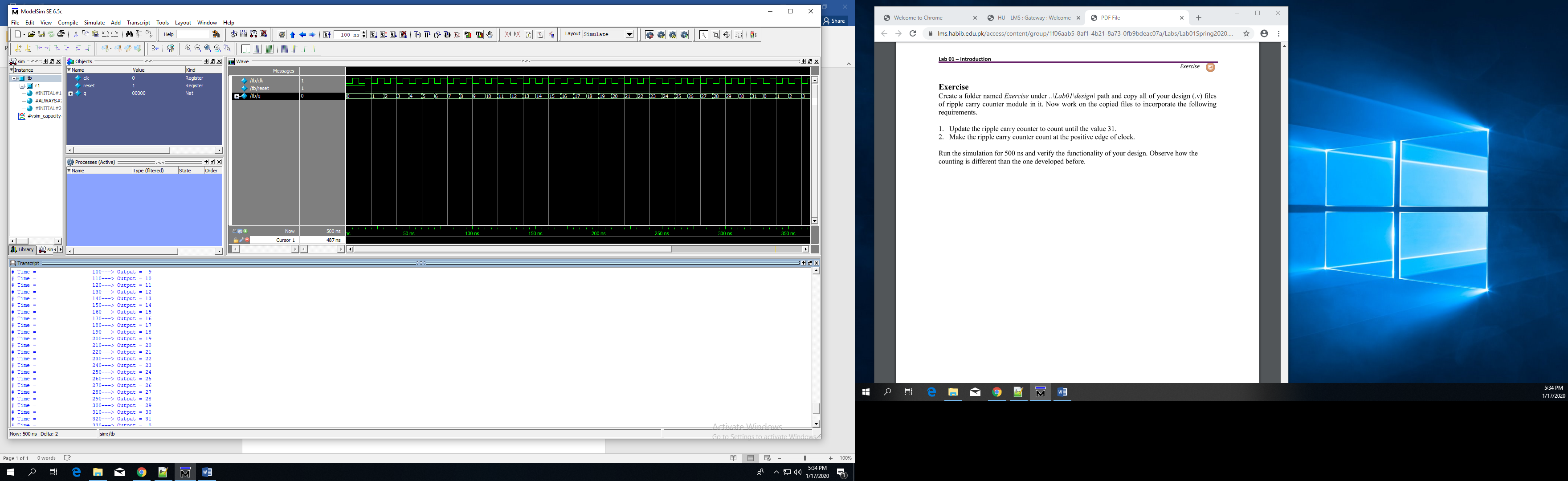
#400 reset = 1'b1;

end

initial

$monitor("Time = ", $time, "---> Output = %d", q);

endmodule



**Exercise 2:**

To make Ripple Carry Counter count at positive edge of clock, we can change the always block of D Flip Flop so that it activates on positive edge of clock.

**D – FLIP FLOP**

module D\_FF

(

input c,r,

input d,

output reg q

);

always @(posedge r or posedge c)

begin

if(r)

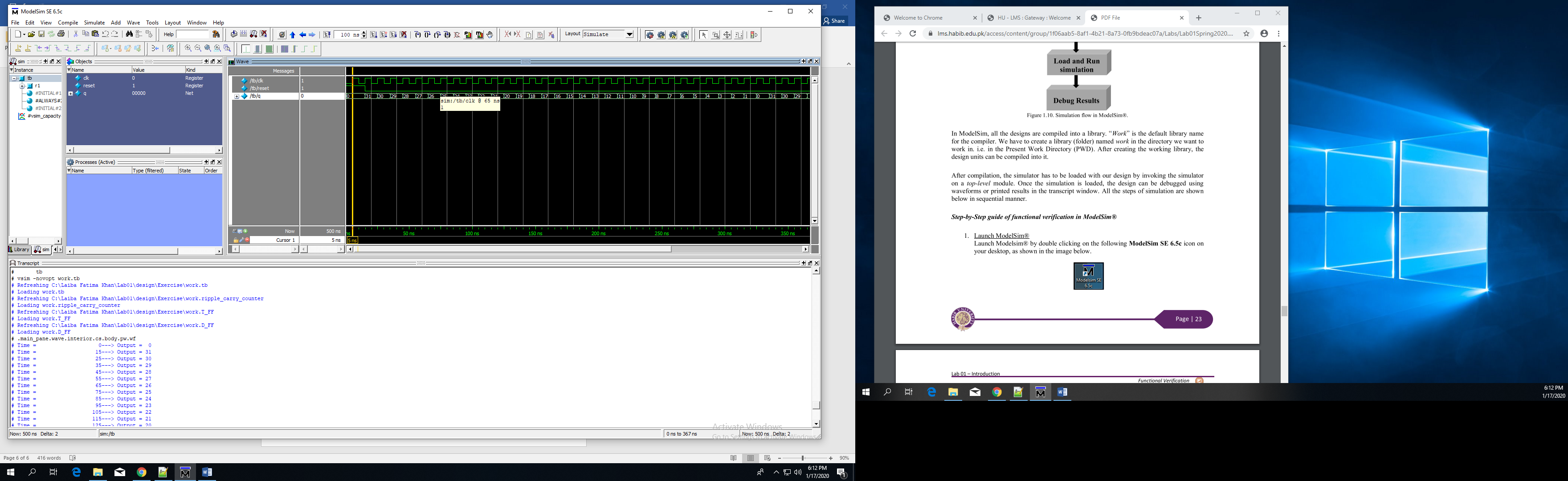
q<=1'b0;

else

q<=d;

end

endmodule



**RUN.do for Exercises:**

vlib work

#compiling design modules

vlog tb.v ripple\_carry\_counter.v T\_FF.v D\_FF.v

#no optimization

vsim -novopt work.tb

#view waves

view wave

#adding waves

add wave sim:/tb/clk

add wave sim:/tb/reset

add wave sim:/tb/q

run 500ns