**Test bench**

`timescale 1ns / 1ps

////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 10:41:40 06/21/2017

// Design Name: counter

// Module Name: E:/project\_students/pd mux/framepdmux/counterr\_test.v

// Project Name: framepdmux

// Target Device:

// Tool versions:

// Description:

//

// Verilog Test Fixture created by ISE for module: counter

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

////////////////////////////////////////////////////////////////////////////////

module counterr\_test;

// Inputs

reg [7:0]tsin1;

reg [7:0]tsin2;

reg [7:0]tsin3;

// Outputs

wire out;

wire frame;

//reg [3:0]out;

reg clk; reg load; reg rst,temp,i,p;

// Instantiate the Unit Under Test (UUT)

counter uut (

.out(out),

.frame(frame),

.tsin1(tsin1),

.tsin2(tsin2),

.tsin3(tsin3),

.load(load),

.clk(clk),

.rst(rst)

);

always #36 clk=~clk;

always begin

#29 load=1'b1;

#10 load=1'b0; end

initial begin

// Initialize Inputs

clk=0; rst=0; #36 rst=1; #36 rst=0;

// Wait 100 ns for global reset to finish

tsin1=8'b11111111; tsin2=8'b10101010;

tsin3=8'b01010101;

#100 tsin1=8'b10101010;

#100 tsin1=8'b01010101;

// Add stimulus here

/\*$display("temp value =%b",temp);

$display("temp value =%b",i);

$display("temp value =%b",p);\*/

end

endmodule

**Code**

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 13:23:39 06/20/2017

// Design Name:

// Module Name: counter

// Project Name:

// Target Devices:

// Tool versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module counter(

output out,output frame,

input tsin1,

input tsin2,

input tsin3,

input load,

input clk,

input rst

);

//----------Output Ports--------------

//------------Input Ports--------------

//------------Internal Variables--------

reg [3:0] out;

//integer i;

reg [7:0]temp;

reg frame;

wire [7:0]tsin1;

wire [7:0]tsin2;

wire [7:0]tsin3;

reg p;

//-------------Code Starts Here-------

always @(posedge clk)

begin:shifting

/\*integer i;

for(i=0;i<3;i=i+1)

begin\*/

fork

begin

/\*if(i==0) \*/p<=tsin1;

/\*else if(i==1) p<=tsin2;

else if(i==2) p<=tsin3;\*/

//anush a1(frame,load,clk,rst,p);

begin

if (rst==1)

temp <= 1;

else if(load)

begin

temp <= tsin1;

end

else begin

frame <= temp[7];

temp <= {temp[6:0],1'b0};

end

end

end

begin

if (rst)

begin

out <= 4'b0 ;

end

else if(out<4'b0111)

begin

out <= out + 1;

end

end

begin

if(out==4'b0111)

begin

//i=i+1;

out=4'b0000;

end

end

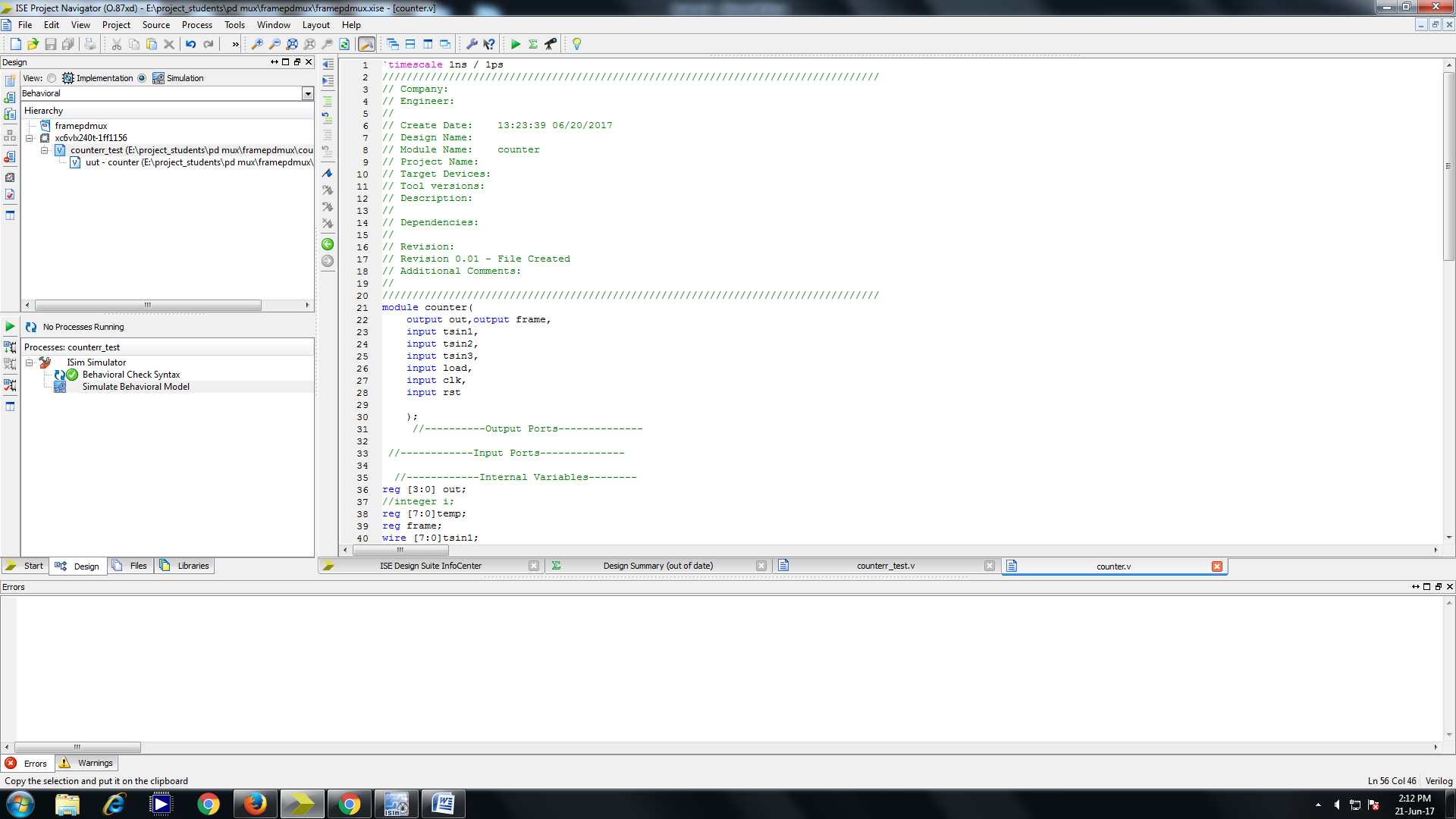
join

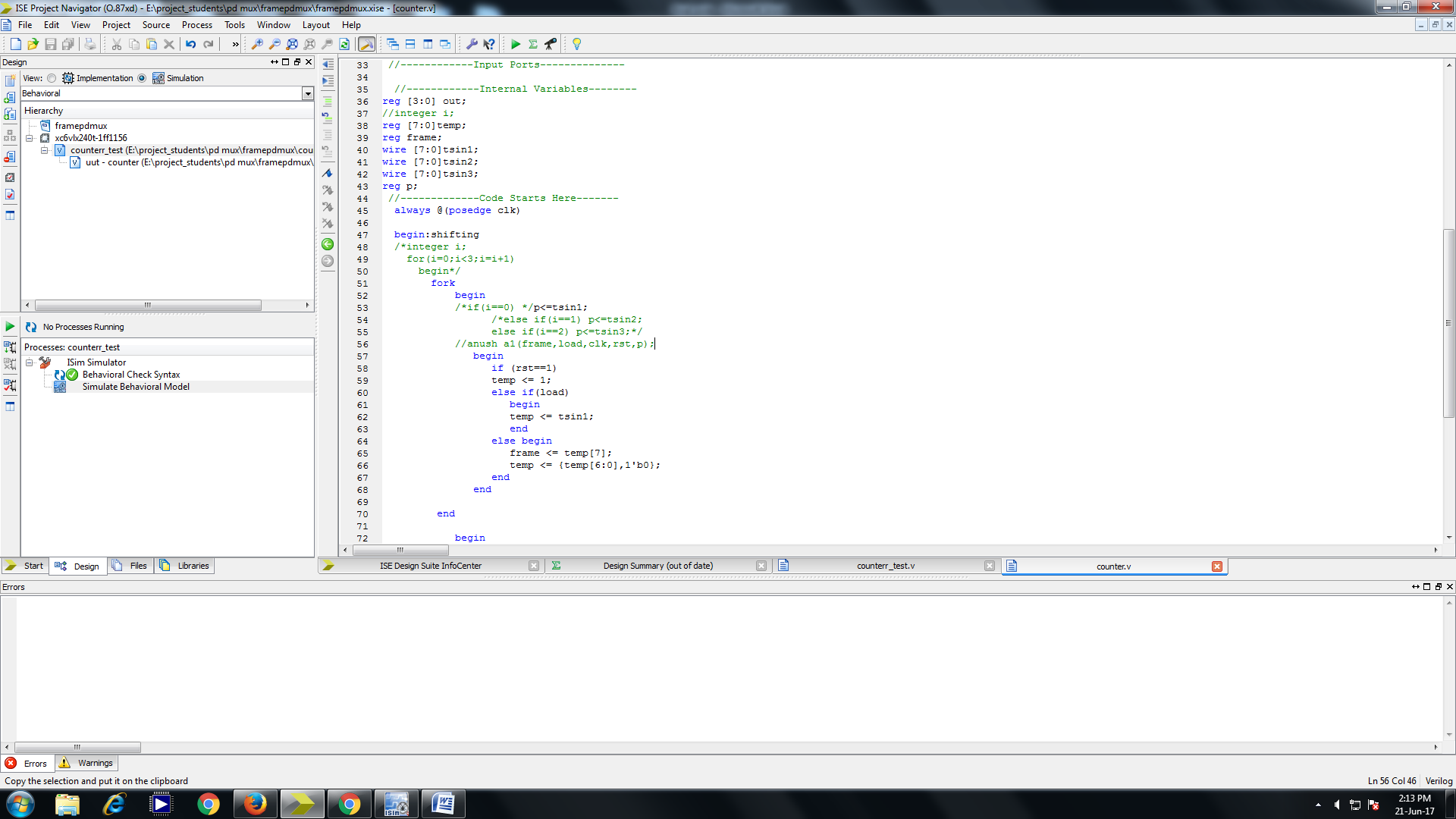
//end

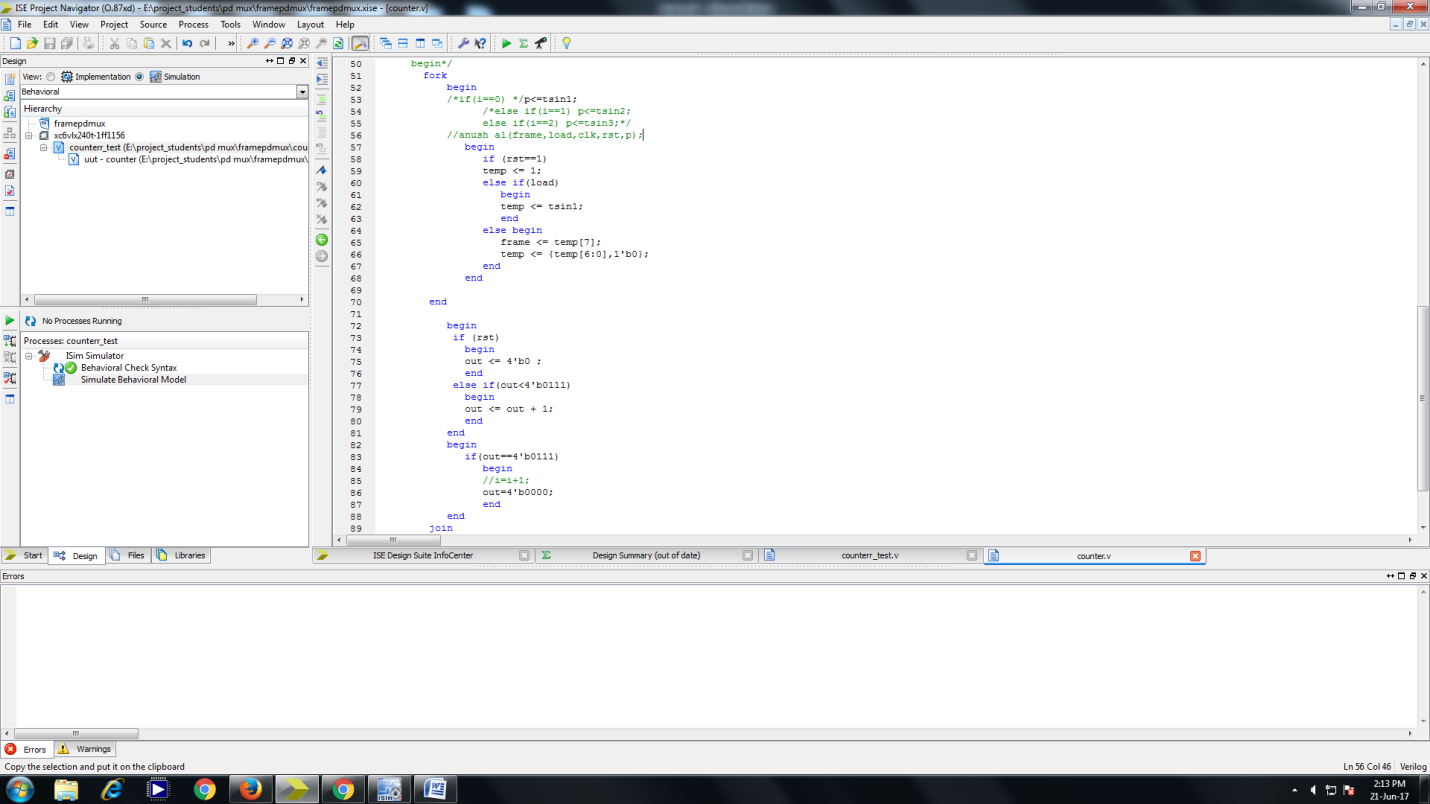
end

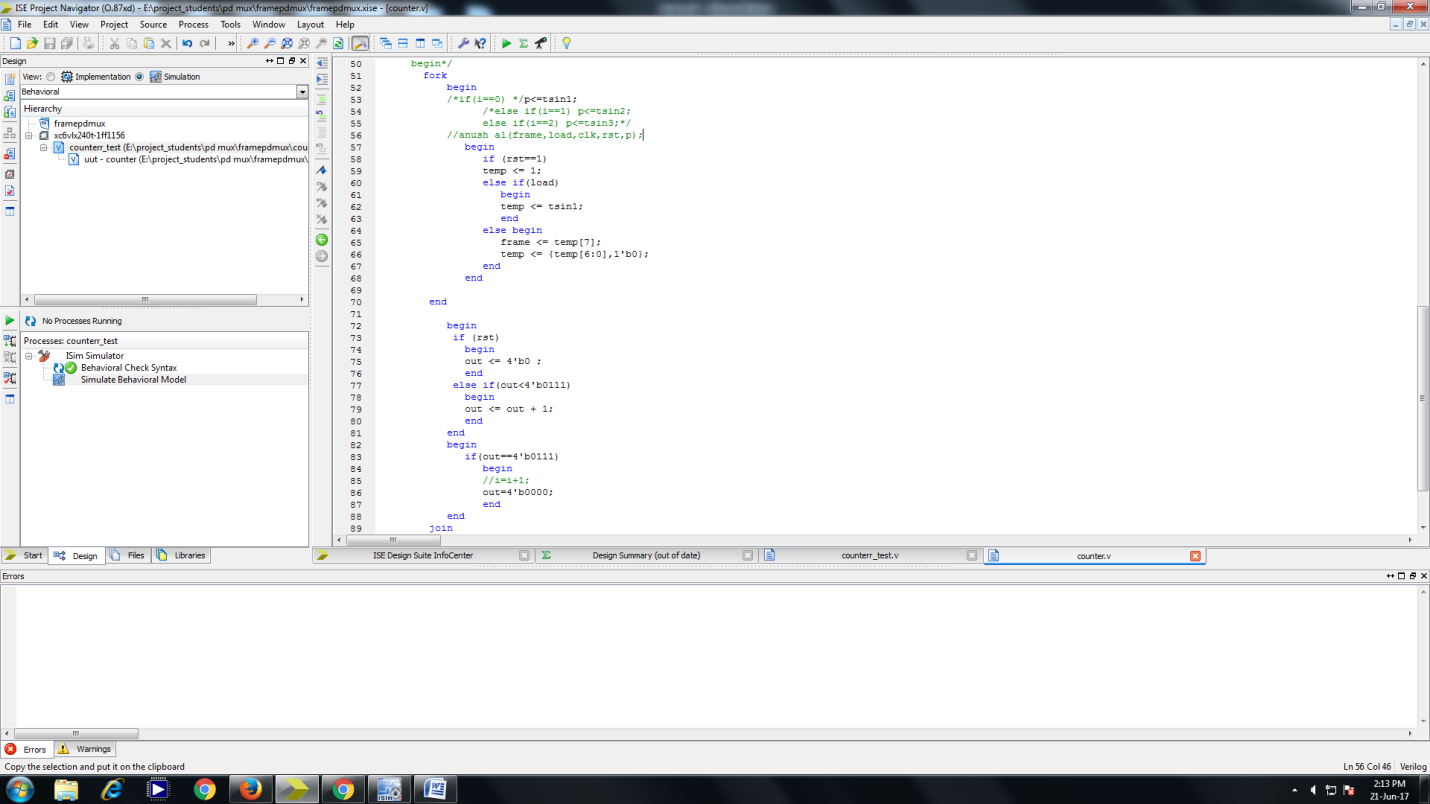
endmodule

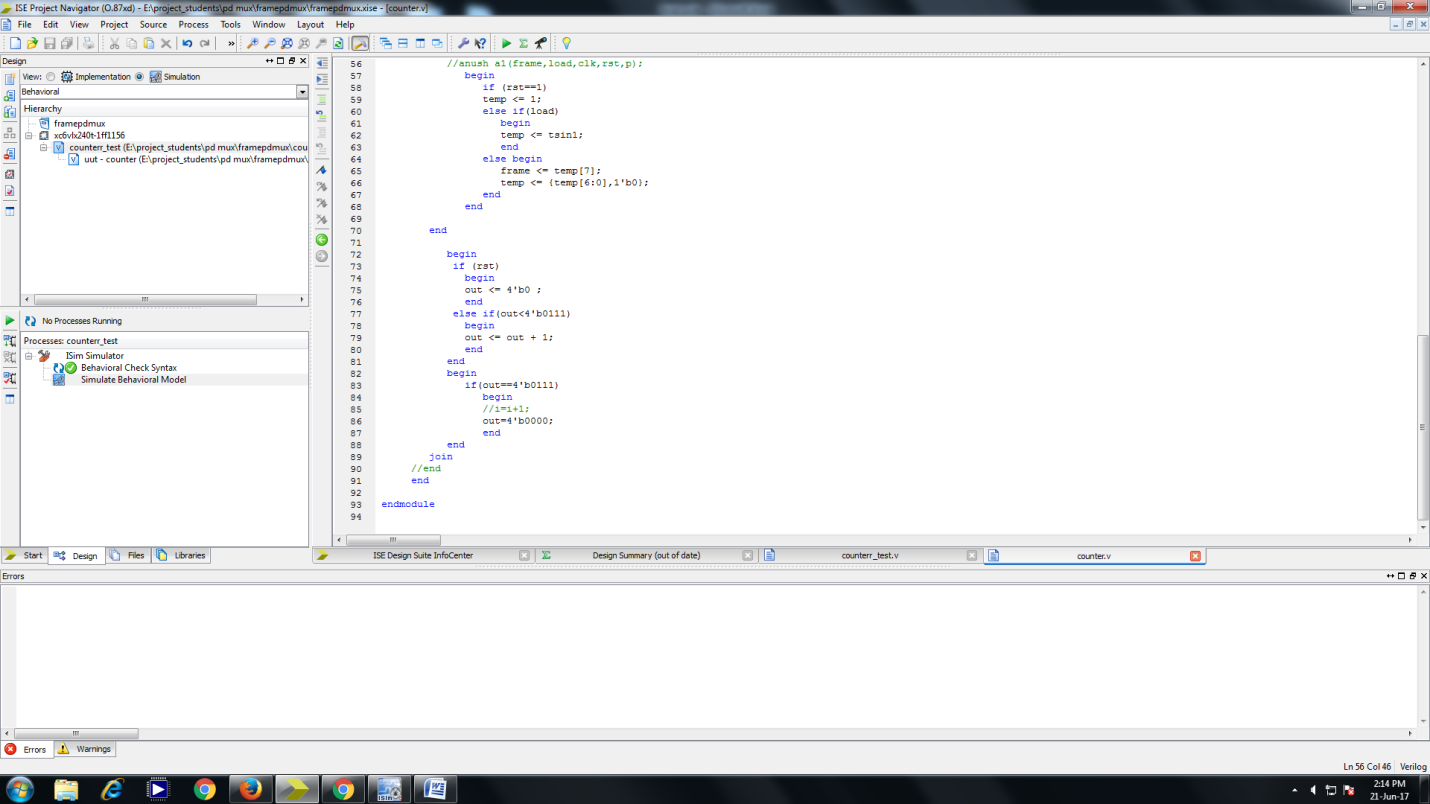
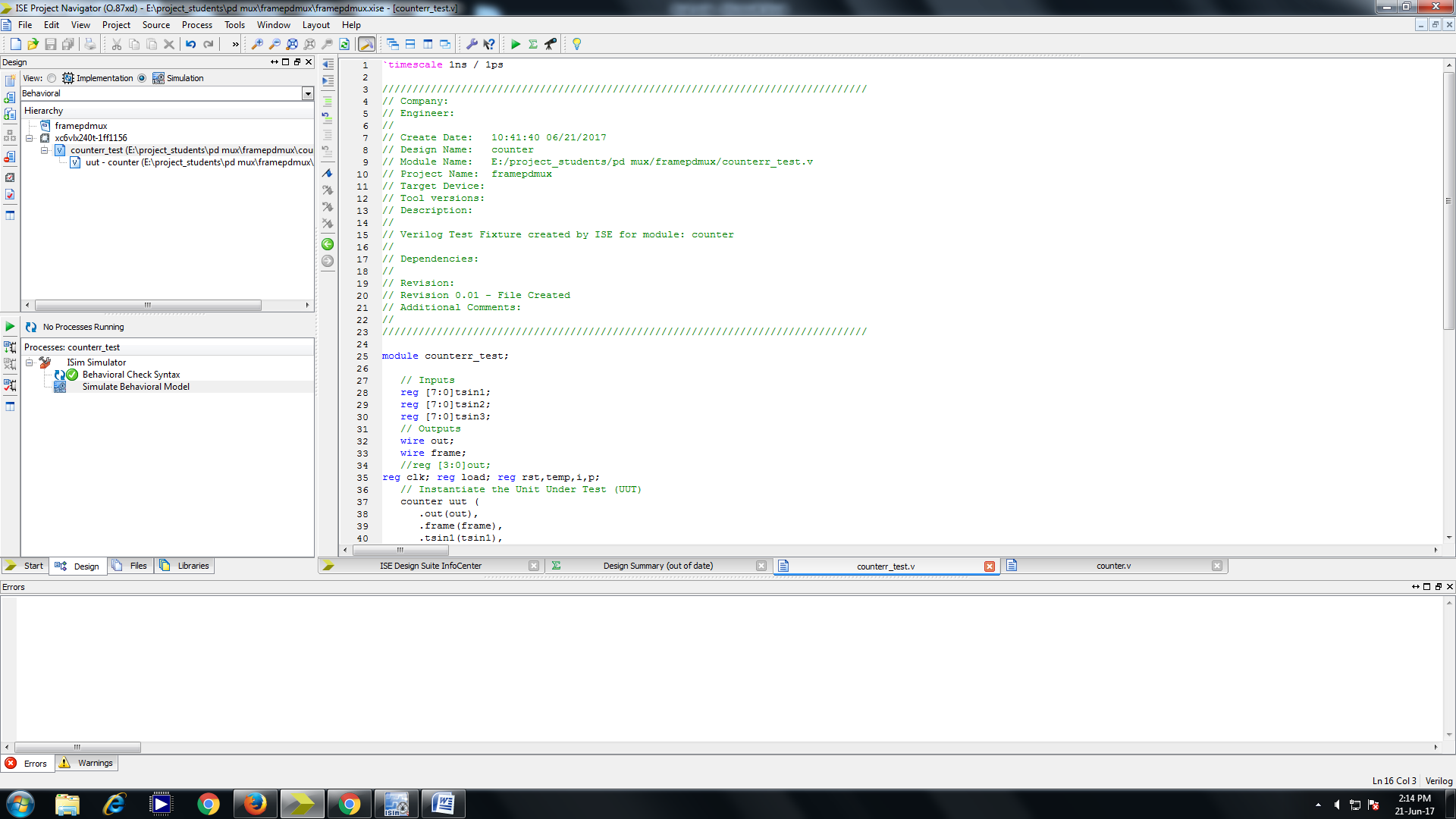
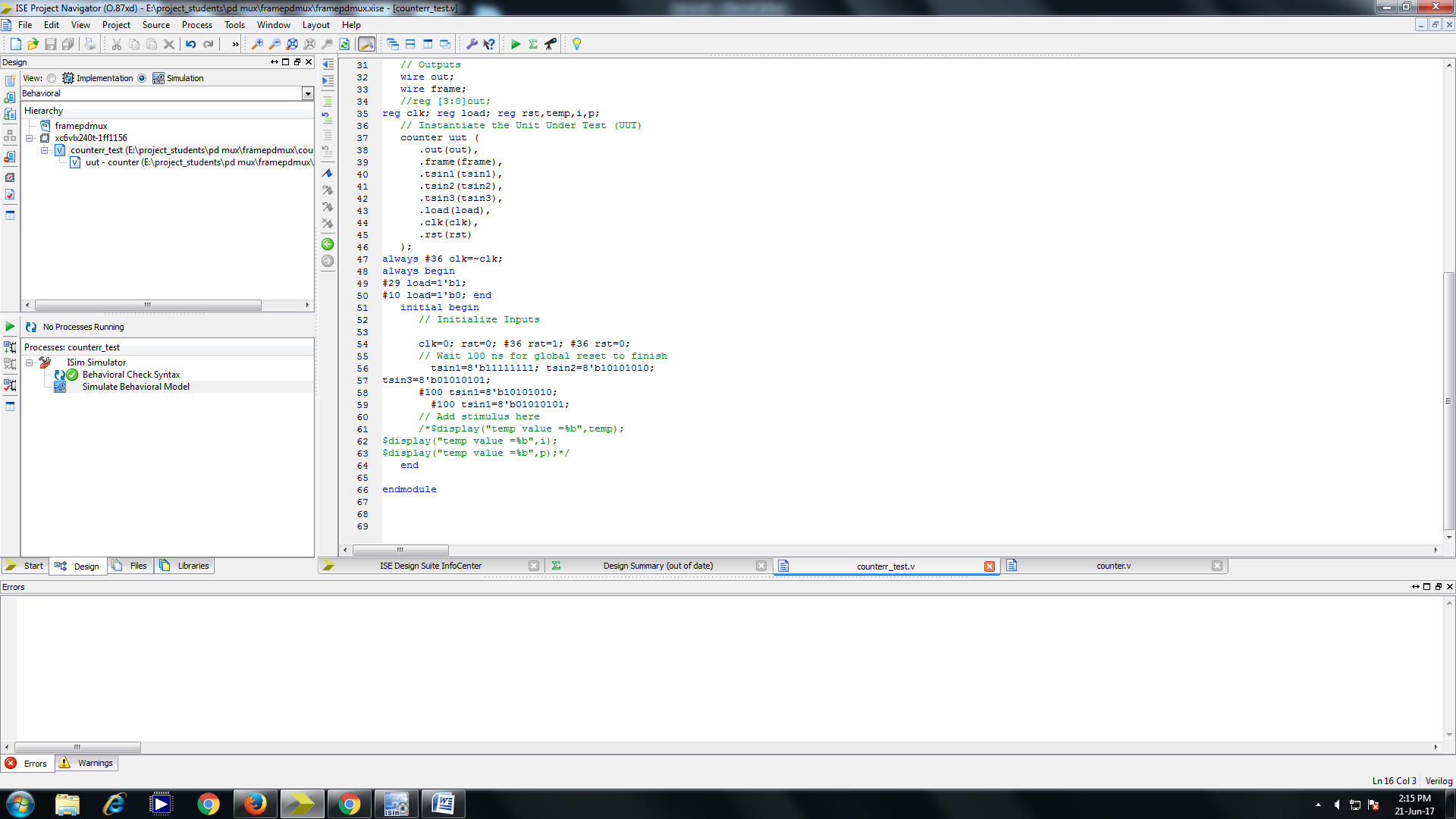
**screen shots**



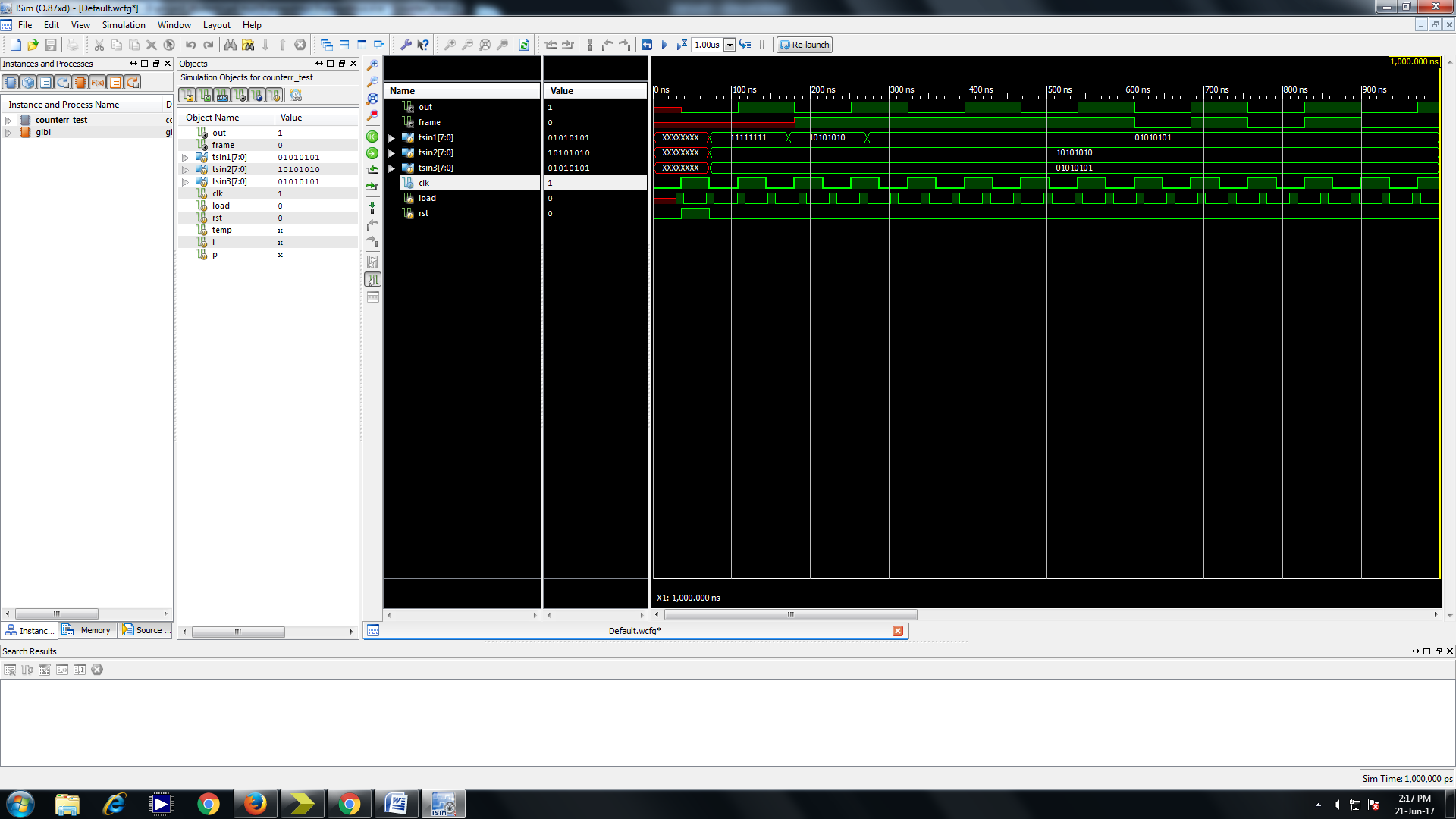






**Output**

****