INDRAPRASTHA INSTITUTE of INFORMATION TECHNOLOGY DELHI

Department
of
Electronics & Communication Engineering

Embedded Logic Design(ECE270)

Dr. Sumit J Darak

Lab_5: Design and implement a GCD Module

Mohammad Shariq
2020220

25-10-2021

OBJECTIVE:

- GCD code implementation on Vivado

Theory:

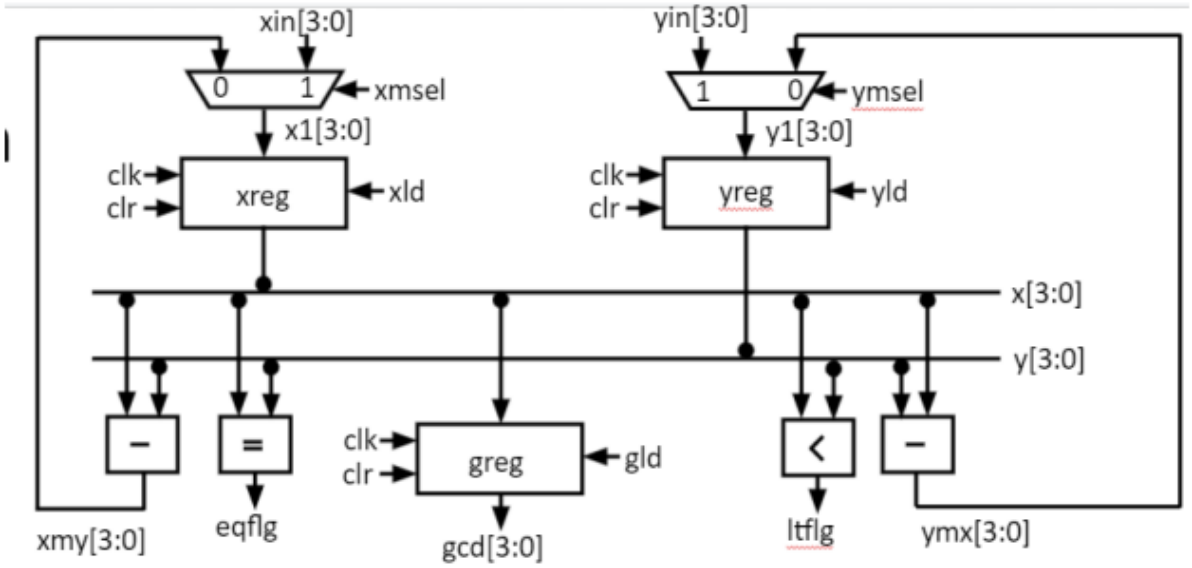
GCD Explanation : GCD stands for Greater common divisor. Example : GCD(12,8), GCD(35,49) = 7

GCD calculation is done by subtracting larger numbers with smaller ones and replacing the larger number with the result of subtraction. Repeat this and stop when both the numbers become equal.

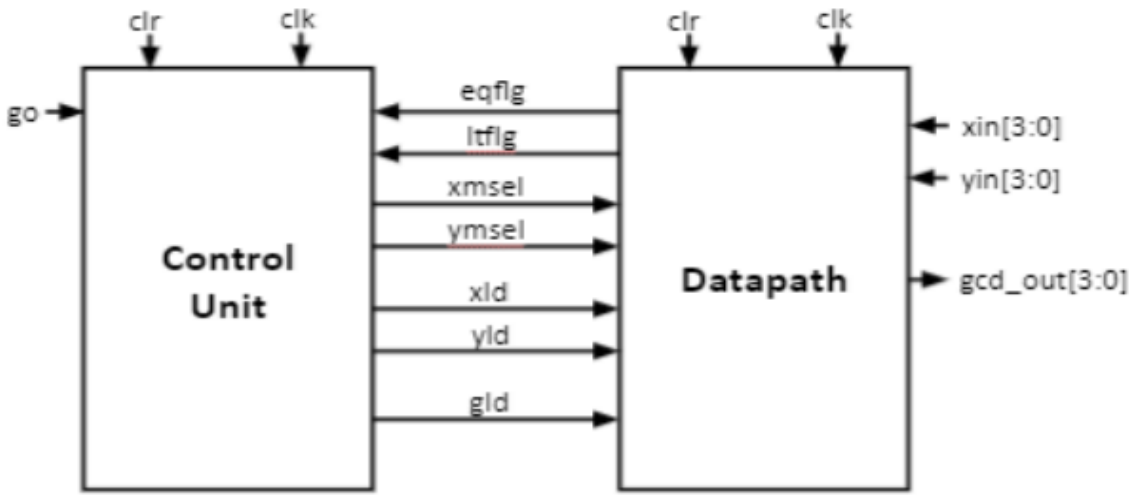
Data Path: Data Processing and operations are done here

Control Path: Gives the control logic signals and makes the Data Path do the tasks needed. The control path is the FSM.

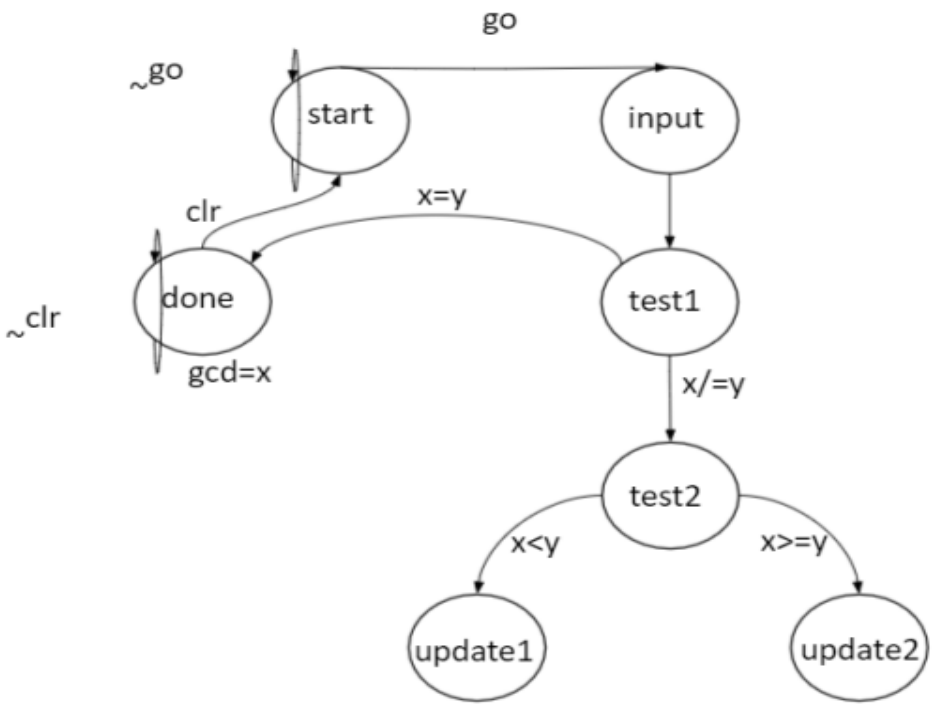
1. We have the registers for both inputs and outputs in the code. They are declared as xreg, yreg and greg. The former two are 3 bits wide, and the latter are one bit wide.
2. Connection of the register is given to the logical and arithmetic operations.
3. Following is the block diagram of the code that we are going to implement in Verilog.



4. To decide the values of xld, yld, clk, clr and gld, we will also make a control unit.

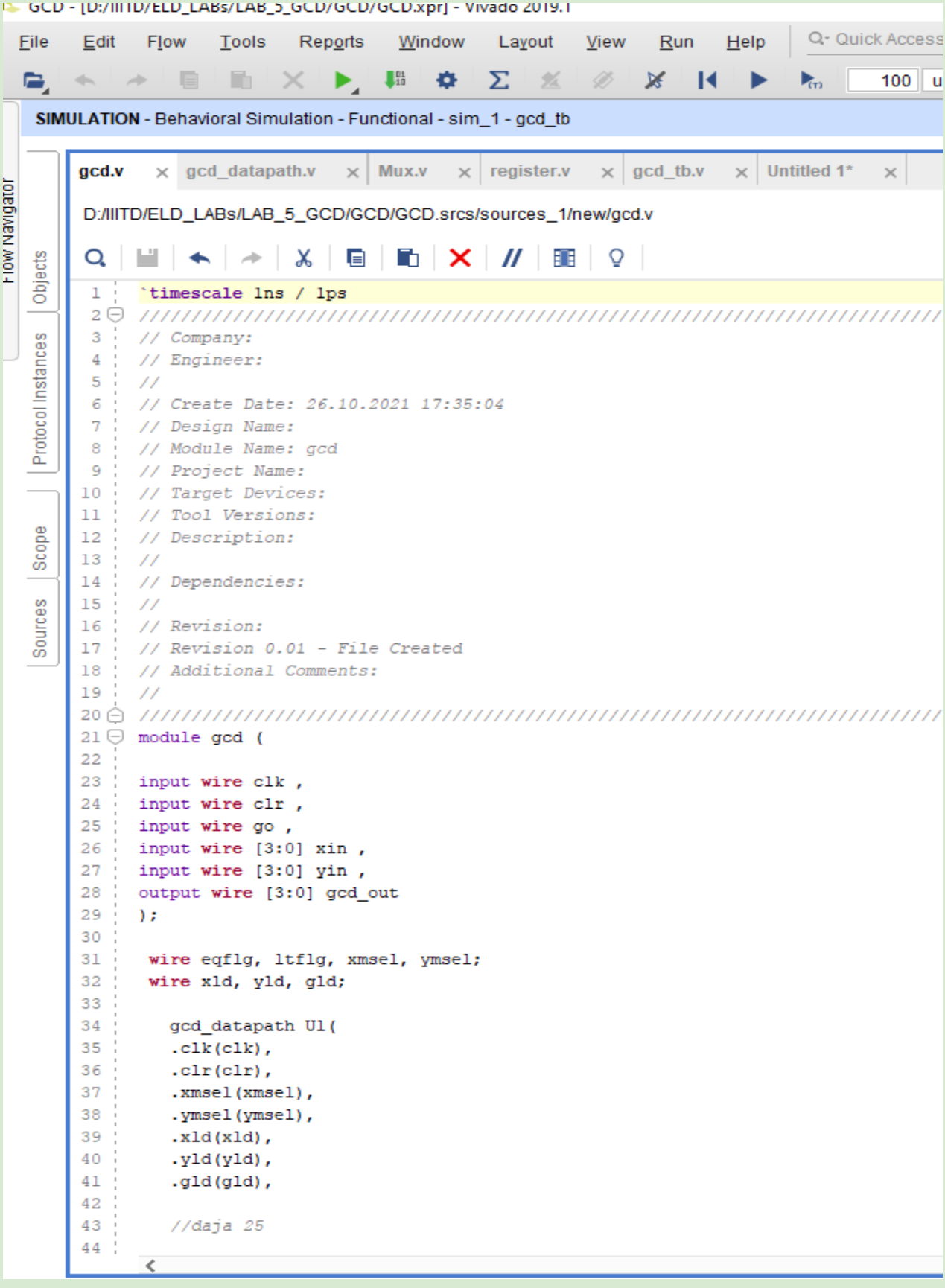


5. The above figure shows the overall control and data transfer
   a. When go is one then data xin and yin will be loaded into the data path
   b. If x=y then the controller makes the gld = 1 and x passed to the output
   c. if x<y then y = y-x will be loaded to the yreg and the other way around for the condition of x>y
6. Following is the FSM that we will be implementing for the GCD controller block:-



Observations:

programme for the functionality of GCD



GCD - [D:/IIITD/ELD_LABs/LAB_5_GCD/GCD/GCD.xpr] - Vivado 2019.1

File Edit Flow Tools Reports Window Layout View Run Help Quick Access

100 us

SIMULATION - Behavioral Simulation - Functional - sim_1 - gcd_tb

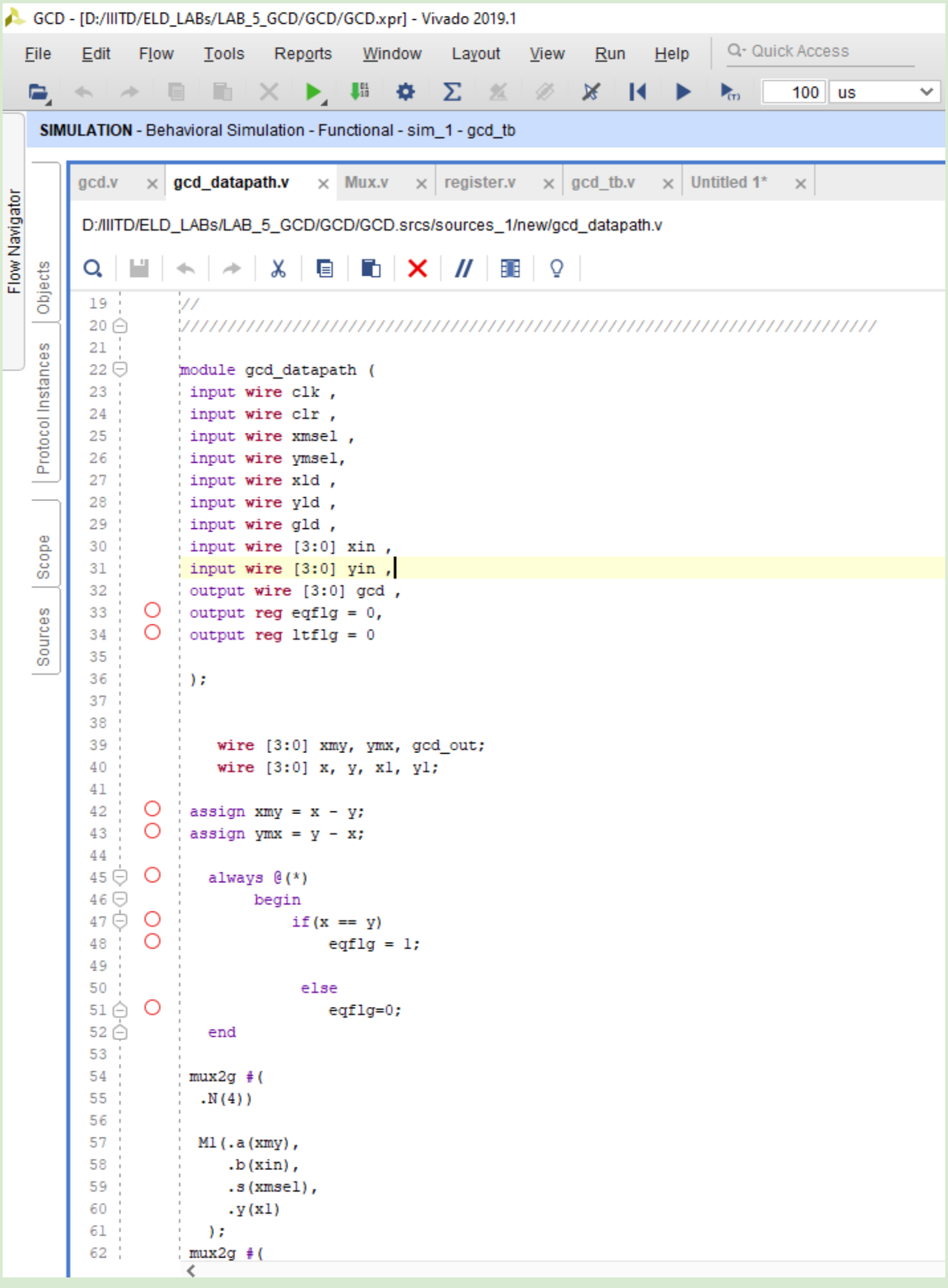
Flow Navigator | Objects | Protocol Instances | Scope | Sources

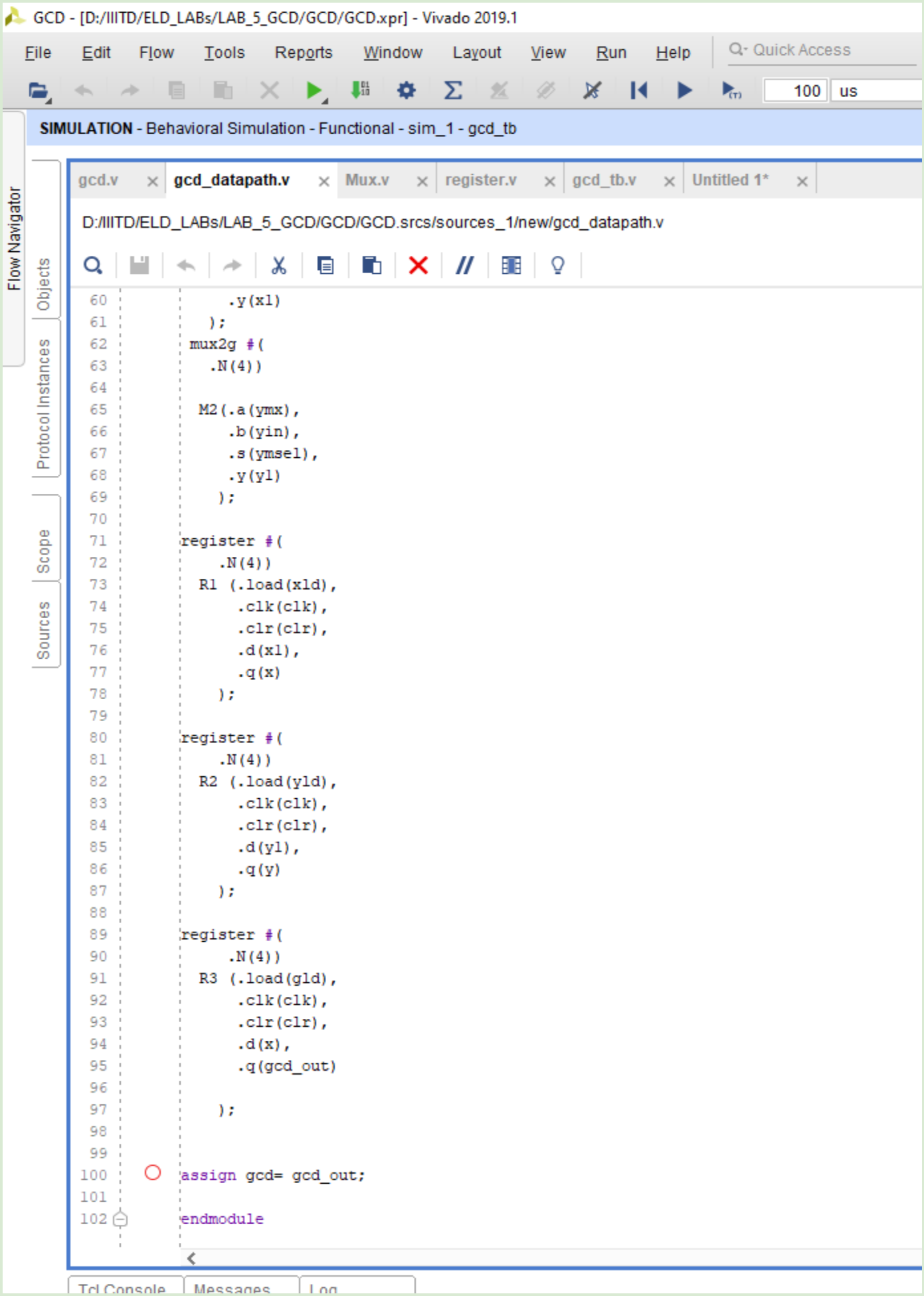
gcd.v | gcd_datapath.v | Mux.v | register.v | gcd_tb.v | Untitled 1*

D:/IIITD/ELD_LABs/LAB_5_GCD/GCD/GCD.srcs/sources_1/new/gcd.v

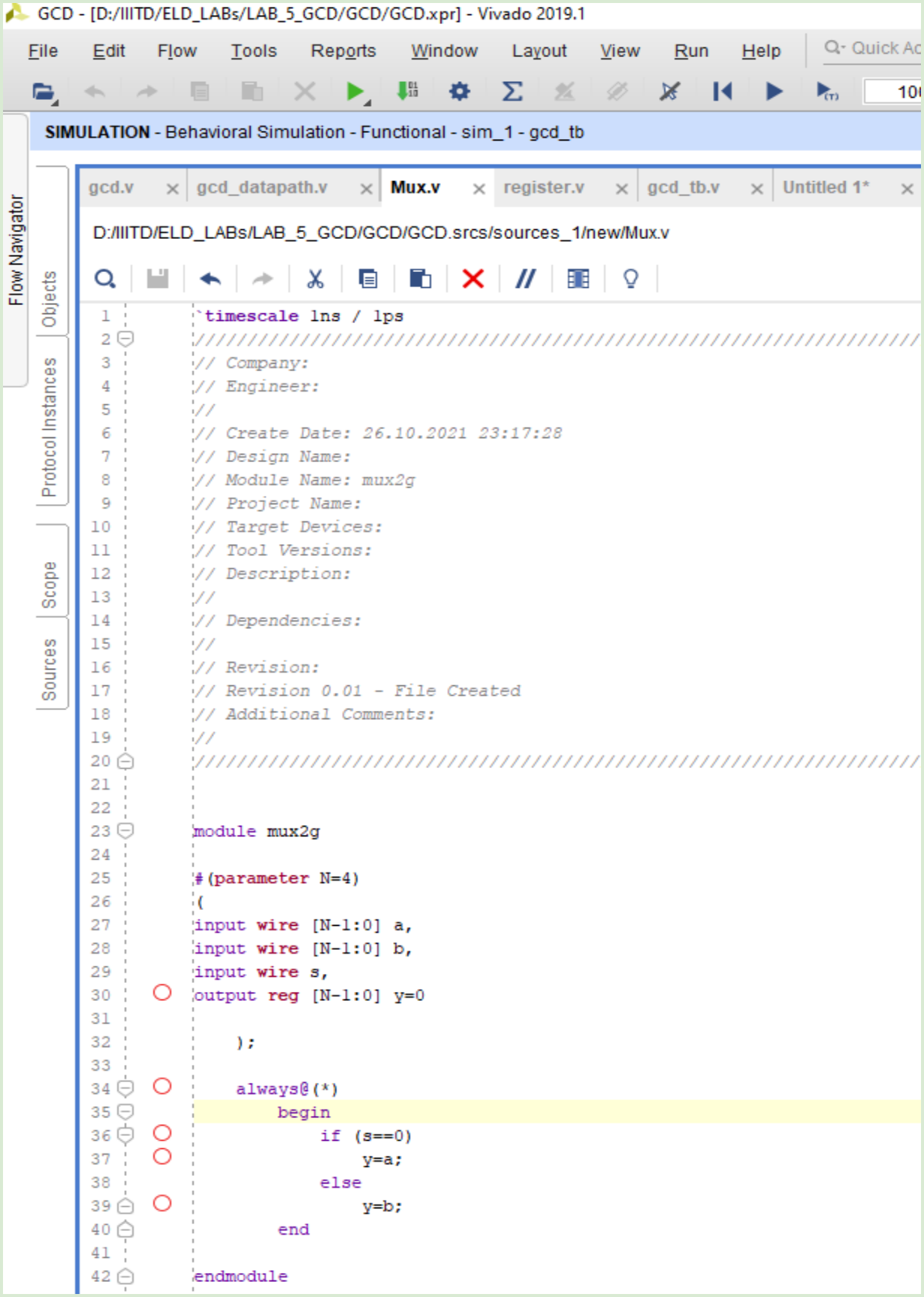
```
36      .clr(clr),
37      .xmsel(xmsel),
38      .ymsel(ymsel),
39      .xld(xld),
40      .yld(yld),
41      .gld(gld),
42
43      //daja 25
44
45      .xin(xin),
46      .yin(yin),
47      .gcd(gcd_out),
48
49      // flags to determine the weather we obtained the GCD or shoukld we keep on going.
50
51       .eqflg(eqflg),
52       .ltflg(ltflg)
53       );
54
55     gcd_control U2(
56
57       .clk(clk),
58       .clr(clr),
59       .go(go),
60
61       //flags to determine whether we obtained the GCD or should we keep going
62
63       .eqflg(eqflg),
64       .ltflg(ltflg),
65
66        // Select line for loading data into reg MUX
67       .xmsel(xmsel),
68       .ymsel(ymsel),
69
70        // load signals
71       .xld(xld),
72       .yld(yld),
73       .gld(gld)
74
75   );
76
77
78   endmodule
```

program for GCD_datapath



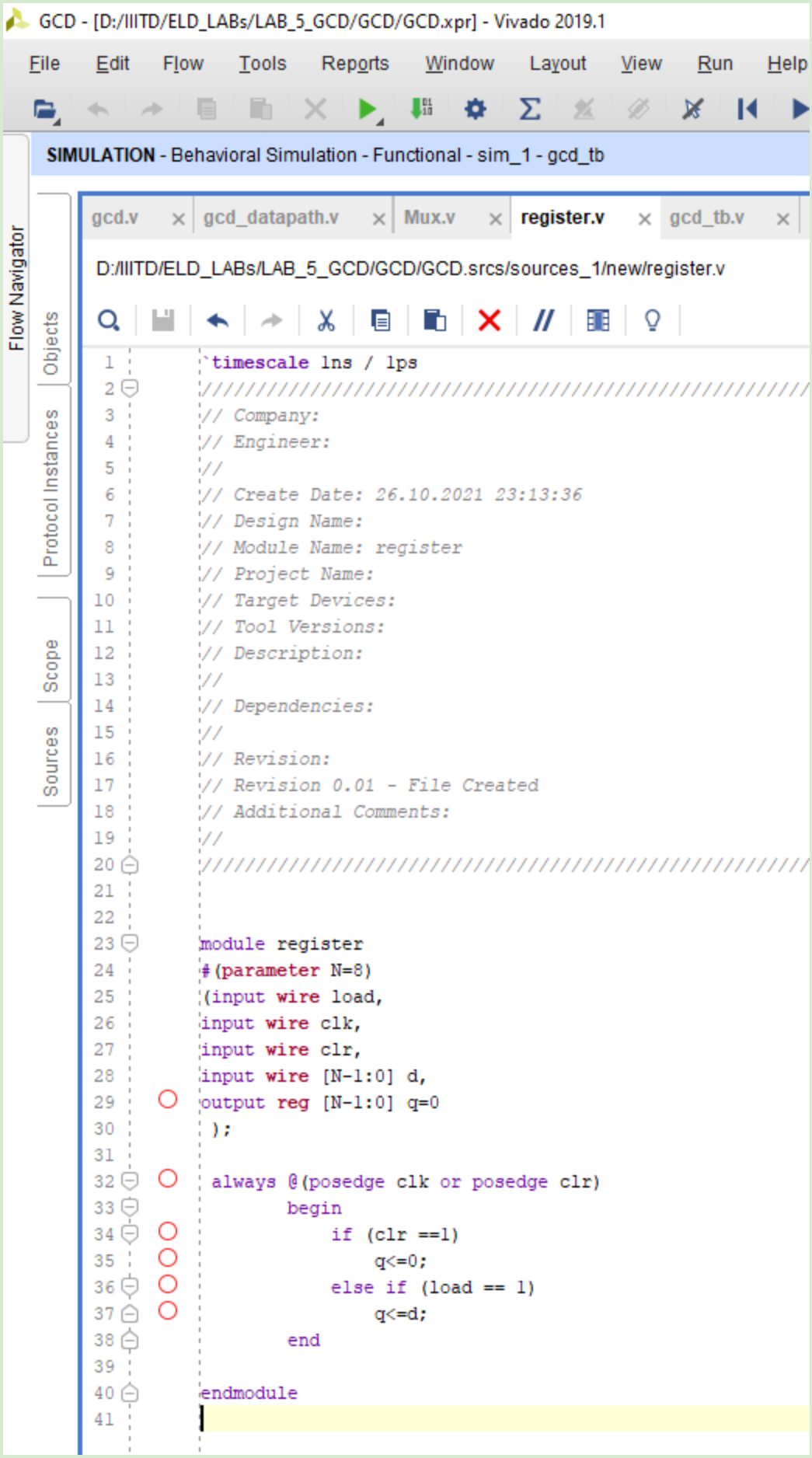

GCD - [D:/IIITD/ELD_LABs/LAB_5_GCD/GCD/GCD.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Run Help
Quick Access
100 us
SIMULATION - Behavioral Simulation - Functional - sim_1 - gcd_tb
Flow Navigator
Objects
Protocol Instances
Scope
Sources
gcd.v
gcd_datapath.v
Mux.v
register.v
gcd_tb.v
Untitled 1*
D:/IIITD/ELD_LABs/LAB_5_GCD/GCD/GCD.srcs/sources_1/new/gcd_datapath.v
.y(x1)
);
mux2g #(
.N(4))
M2(.a(ymx),
.b(yin),
.s(ymsel),
.y(y1)
);
register #(
.N(4))
R1 (.load(xld),
.clk(clk),
.clr(clr),
.d(x1),
.q(x)
);
register #(
.N(4))
R2 (.load(yld),
.clk(clk),
.clr(clr),
.d(y1),
.q(y)
);
register #(
.N(4))
R3 (.load(gld),
.clk(clk),
.clr(clr),
.d(x),
.q(gcd_out)
);
assign gcd= gcd_out;
endmodule
Tcl Console
Messages
Log

program for MUX

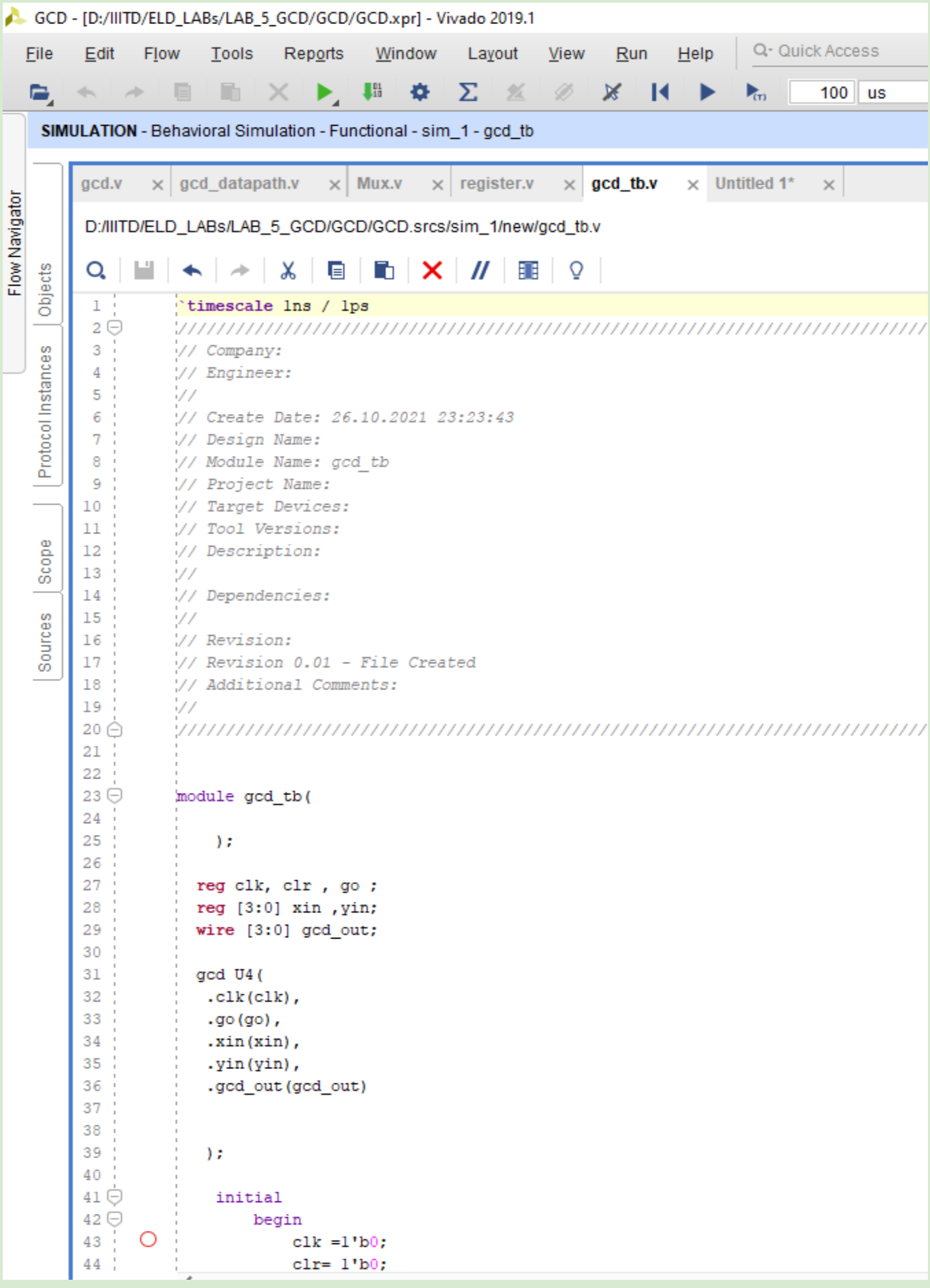


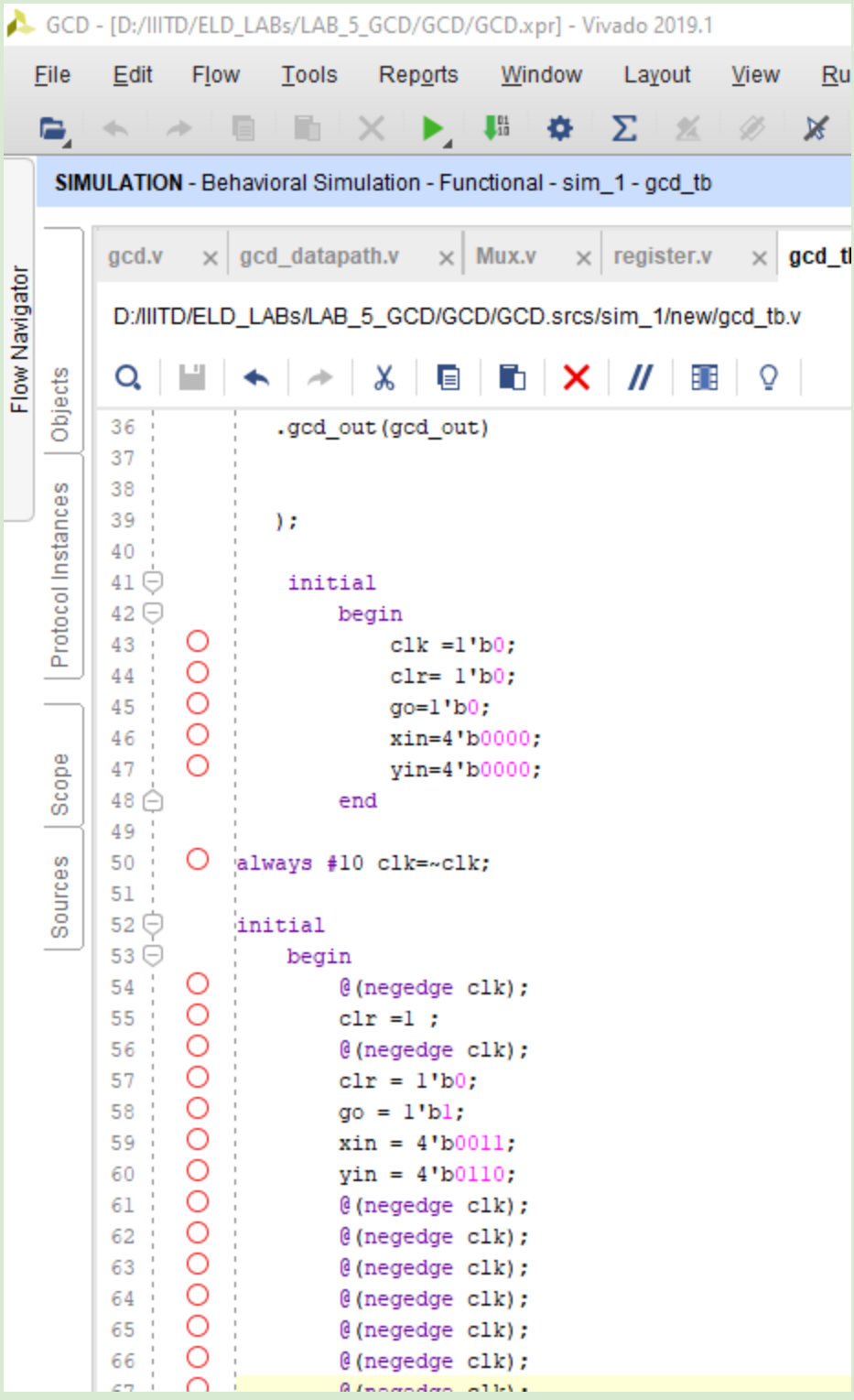

```
1  `timescale 1ns / 1ps
2  //////////////////////////////////////////////////////////////////////////////////
3  // Company:
4  // Engineer:
5  //
6  // Create Date: 26.10.2021 23:17:28
7  // Design Name:
8  // Module Name: mux2g
9  // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 //////////////////////////////////////////////////////////////////////////////////
21
22
23 module mux2g
24
25 #(parameter N=4)
26 (
27 input wire [N-1:0] a,
28 input wire [N-1:0] b,
29 input wire s,
30 output reg [N-1:0] y=0
31
32     );
33
34     always@(*)
35         begin
36             if (s==0)
37                 y=a;
38             else
39                 y=b;
40         end
41
42 endmodule
```

program for Register



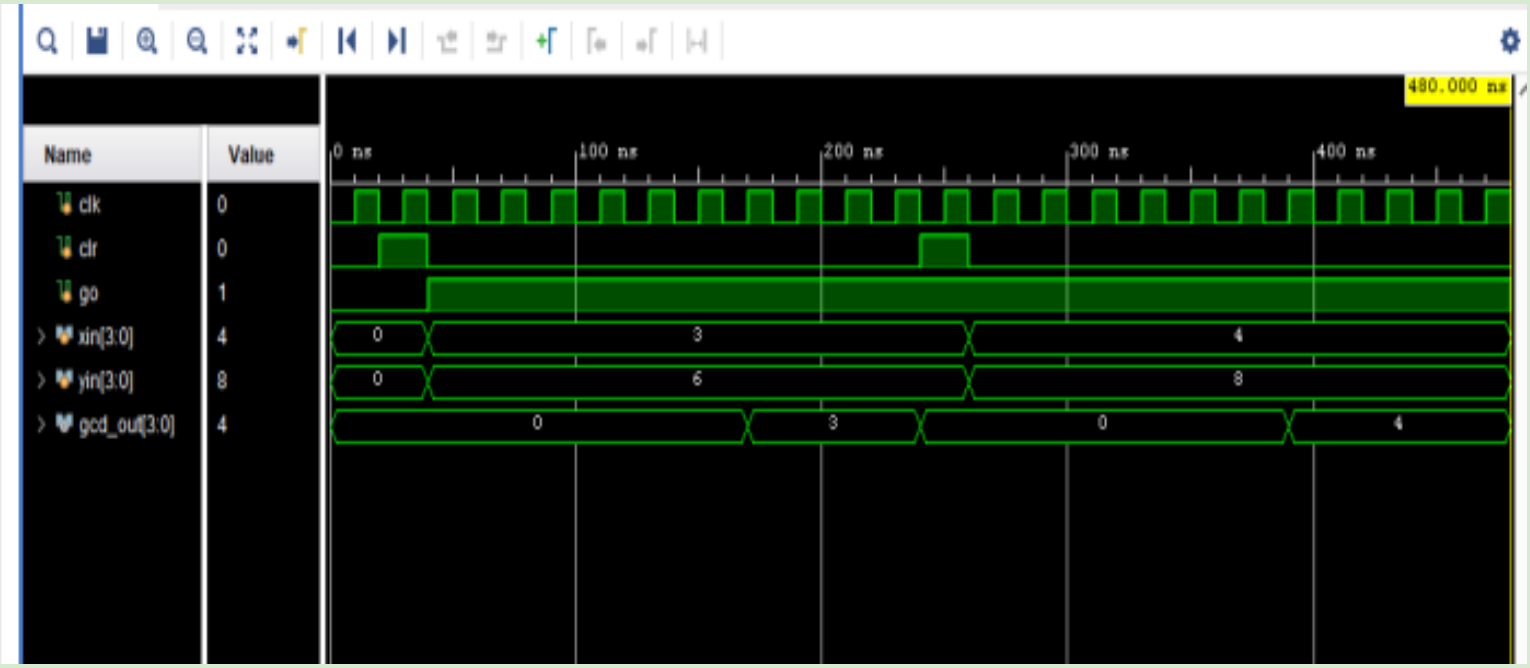
Testbench for GCD




GCD - [D:/IIITD/ELD_LABs/LAB_5_GCD/GCD/GCD.xpr] - Vivado 2019.1
File Edit Flow Tools Reports Window Layout View Ru
SIMULATION - Behavioral Simulation - Functional - sim_1 - gcd_tb
Flow Navigator
Objects
Protocol Instances
Scope
Sources
gcd.v
gcd_datapath.v
Mux.v
register.v
gcd_t
D:/IIITD/ELD_LABs/LAB_5_GCD/GCD/GCD.srcs/sim_1/new/gcd_tb.v
36 .gcd_out(gcd_out)
37
38
39);
40
41 initial
42 begin
43 clk =1'b0;
44 clr= 1'b0;
45 go=1'b0;
46 xin=4'b0000;
47 yin=4'b0000;
48 end
49
50 always #10 clk=~clk;
51
52 initial
53 begin
54 @(negedge clk);
55 clr =1 ;
56 @(negedge clk);
57 clr = 1'b0;
58 go = 1'b1;
59 xin = 4'b0011;
60 yin = 4'b0110;
61 @(negedge clk);
62 @(negedge clk);
63 @(negedge clk);
64 @(negedge clk);
65 @(negedge clk);
66 @(negedge clk);

@negedge is for delay purpose

Results for automated testbench:



Conclusion:

GCD_ module has been implemented successfully on vivado.