REC

HDL Lab

You

➢ **Course Code :** 18EC5DLHDL
➢ **Credits :** 02

COURSE OBJECTIVES:

- To describe the functionality of combinational and sequential circuits using HDL.
- Learn how to simulate, analyze the designed program.
- To know how to debug the written code.
- To justify the design on FPGA kit through implementation.
- To illustrate interfacing concepts using FPGA kit.
- To give an in-depth exposure to the various tools in Xilinx ISE.

REC

Part A Software Programs

1	**Write a HDL code to realize all the logic gates**
2	Write a VHDL code to describe the functions of a Full Adder using three modelling styles.
3	Write a HDL program for the following combinational designs a. 2 to 4 decoder b. 8 to 3 (encoder without priority & with priority) c. 8 to 1 multiplexer d. 4 bit binary to gray converter Multiplexer, de-multiplexer, comparator
4	Develop the HDL code for the following flip-flops, a. SR Flip-Flop b. D Flip-Flop c. JK Flip-Flop

5	Write a model for 8bit ALU. ALU should use combinational logic to calculate an output based on the 3bit op-code input. ALU should decode the 3 bit op-code according to the given in example below. Opcode (2:0) OPCODE ALU OPERATION i). A + B ii). A – B iii). A Complement iv). A * B v). A AND B vi). A OR B vii). A NAND B viii). A XOR B
6	Design 4 bit Binary and BCD counters (Synchronous reset and Asynchronous reset) and “any sequence” counters

REC

: 1.Design using Verilog

- RTL Development

2.Verification using Verilog

- Test bench and Test case Development

3.Simulation and Debug

Simulation Tools: Xilinx/Modelsim/I Verilog

You

REC

HDL

- Allows designers to talk about what the hardware does without actually designing the hardware
- In other words, HDLs helps to separate **behaviour** from **implementation** of functionality
- HDLs are used for :
  - Describing the concurrent behavior of hardware,
  - Structural Modeling for cell libraries.
  - Behavior modeling for procedural and sequential description of hardware
  - Developing stimulus generator and checkers and, at large for verification.
  - Representing synthesized netlists (generated from RTL using symthesis tools)

You

Difference b/w Software Programming Language and Hardware Description Language

- **Software Programming Language** – Language which can be translated into machine instructions and then executed on a computer to obtain the desired output.

- **Hardware Description Language(HDL)** – Language with syntactic and semantic support for modelling the behaviour and structure of hardware.

Notepad

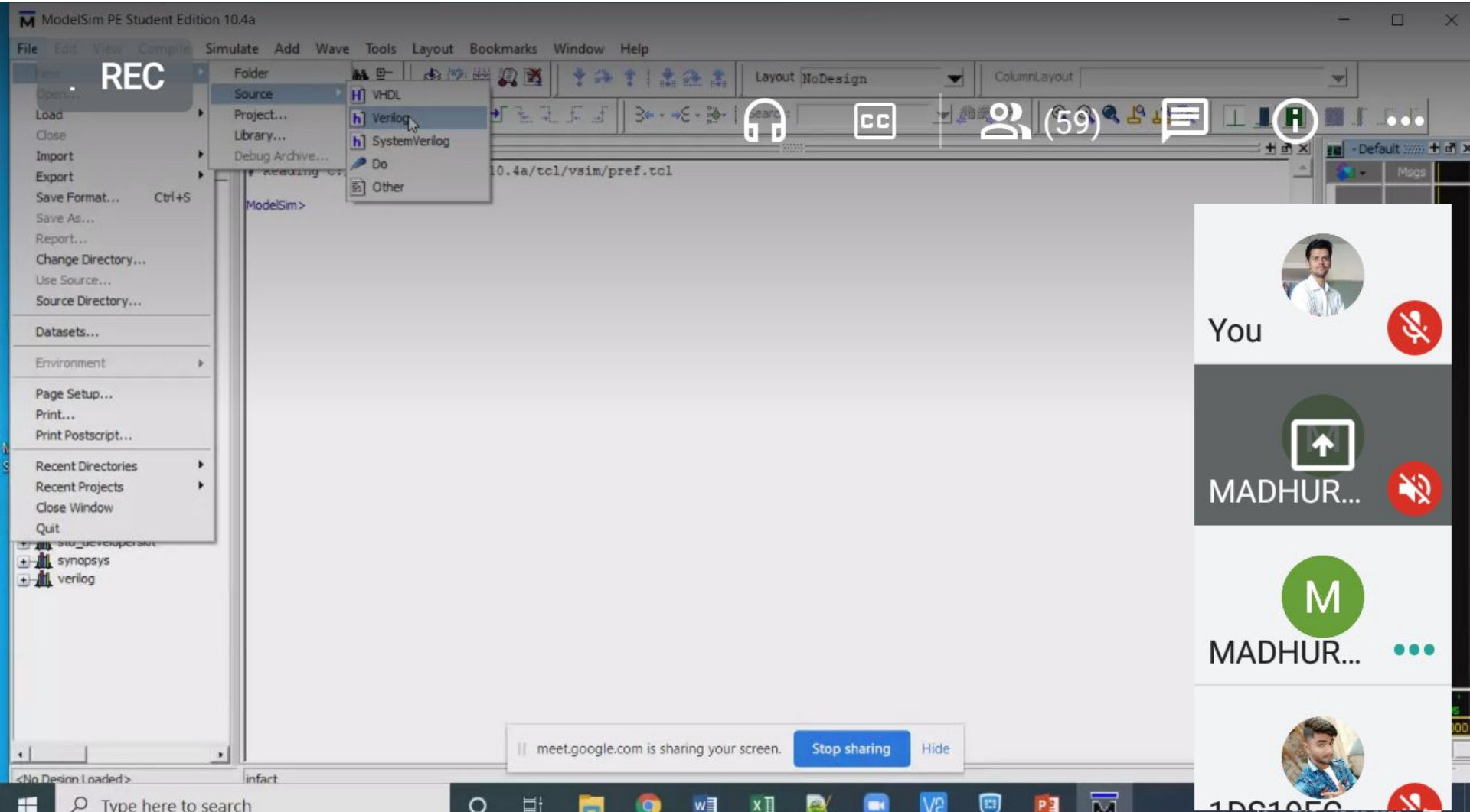
File Edit Format View Help

```
//Verilog design in data flow model
module and_gate(
    input a,b,
    output y);
   assign y = a & b;

endmodule

//TestBench

module tb_and_gate;
    reg A,B;
    wire Y;

    and_gate a1 (.a(A) ,.b(B),.y(Y));

    //Above style is connecting by name
```

You

```
    and_gate a1 (.a(A) ,.b(B),.y(Y));

    //Above style is connecting by name
    initial begin
        A =1'b0;
        B= 1'b0;
        #45 $finish;
    end

    always #6 A =~A;
    always #3 B =~B;

    always @(Y)
    $display( "time =%0t \tINPUT VALUES: \t A=%b B =%b \t output value
endmodule
```

```
) ,.b(B),.y(Y));

connecting by name



%0t \tINPUT VALUES: \t A=%b B =%b \t output value Y  =%b",$time,A,B,Y);
```


ModelSim PE Student Edition 10.4a
File Edit View Compile Simulate Add Wave Tools Layout Bookmarks Window Help
REC
Folder
Source
Project...
Library...
Debug Archive...
VHDL
Verilog
SystemVerilog
Do
Other
Load
Close
Import
Export
Save Format... Ctrl+S
Save As...
Report...
Change Directory...
Use Source...
Source Directory...
Datasets...
Environment
Page Setup...
Print...
Print Postscript...
Recent Directories
Recent Projects
Close Window
Quit
Layout NoDesign
ColumnLayout
10.4a/tcl/vsim/pref.tcl
ModelSim>
synopsys
verilog
<No Design Loaded>
infact
(59)
You
MADHUR...
MADHUR...
meet.google.com is sharing your screen.
Stop sharing
Hide
Type here to search

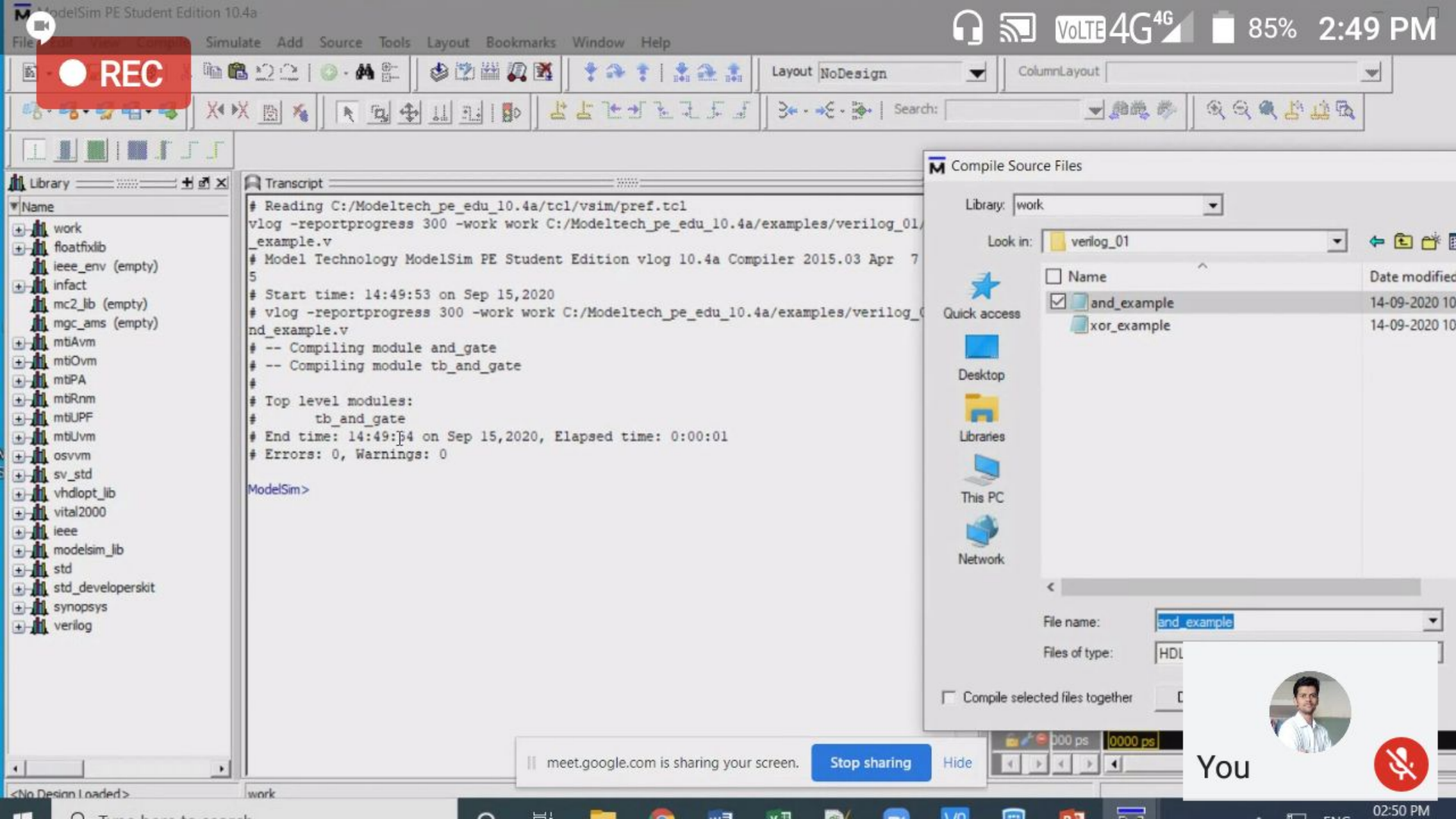
C:/Modeltech_pe_edu_10.4a/examples/verilog_01/and_example.v

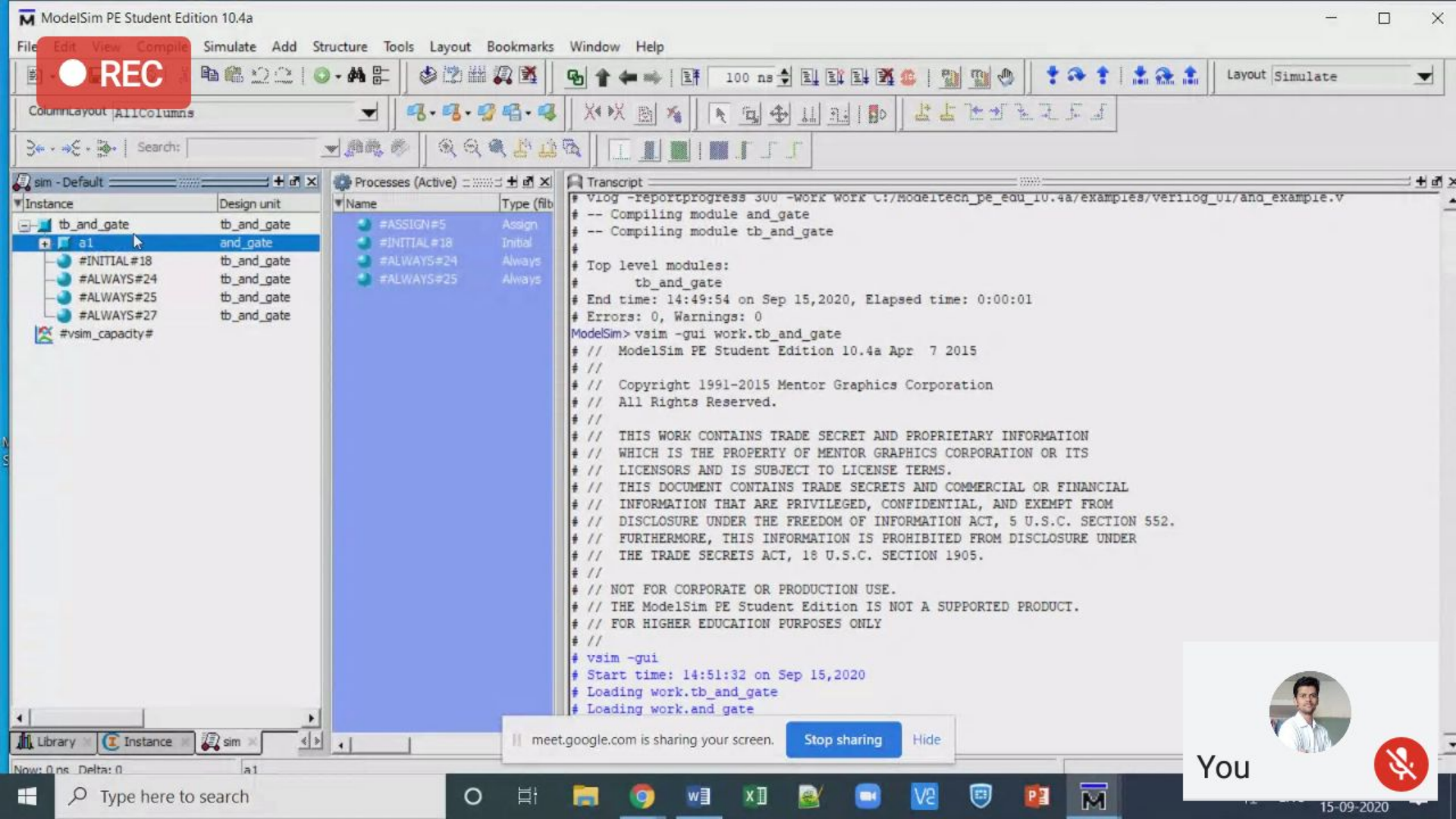
File Edit View Tools Bookmarks Window Help

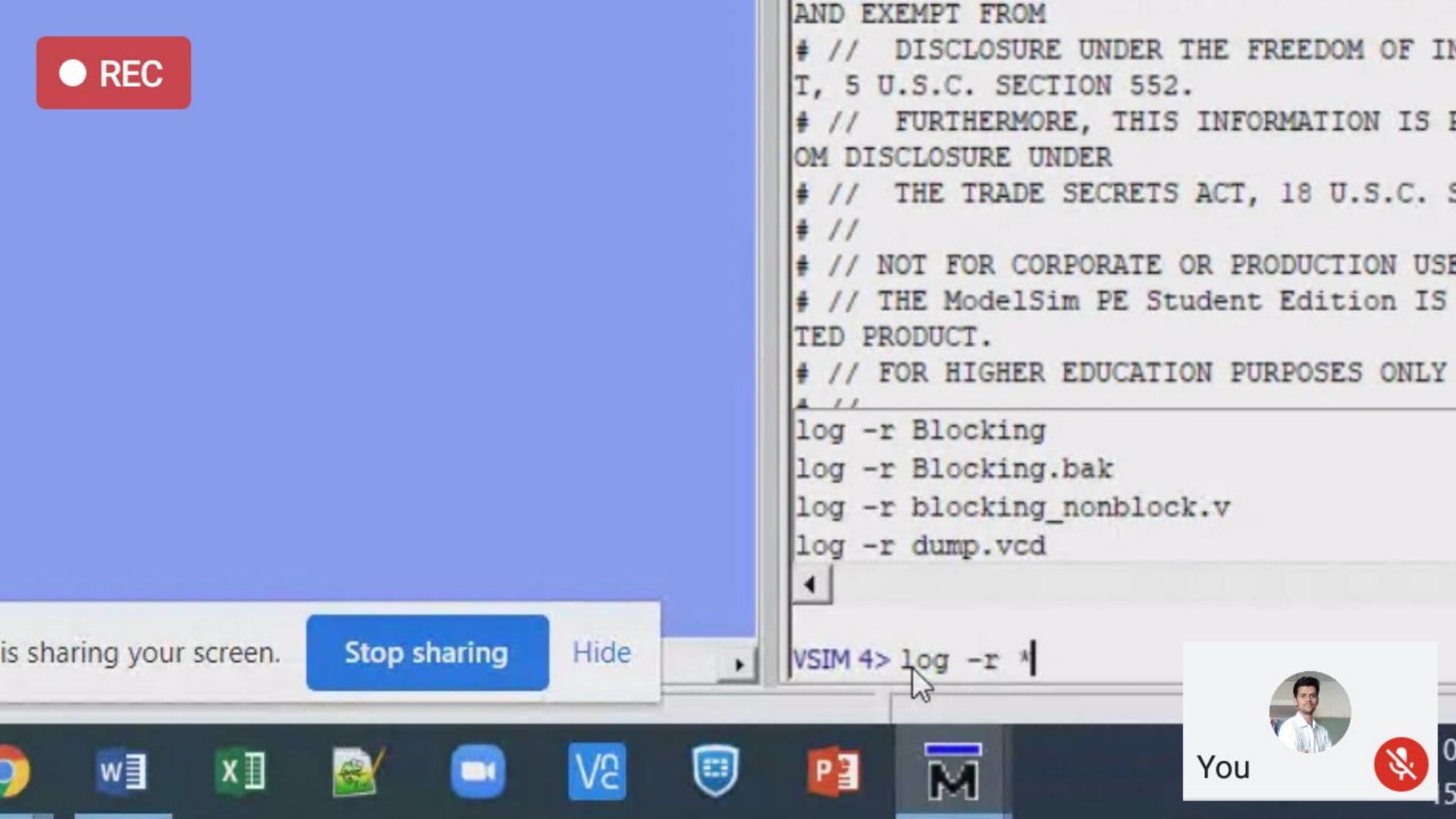
C:/Modeltech_pe_edu_10.4a/examples/verilog_01/and_example.v - Default

Ln#

```
1  //Verilog design in data flow model
2  module and_gate(
3      input a,b,
4      output y);
5     assign y = a & b;
6
7  endmodule
8
9  //TestBench
10
11 module tb_and_gate;
12     reg A,B;
13     wire Y;
14
15     and_gate a1 (.a(A) ,.b(B),.y(Y));
16
17     //Above style is connecting by name
18     initial begin
19         A =1'b0;
20         B= 1'b0;
21         #45 $finish;
22     end
23
24     always #6 A =~A;
25     always #3 B =~B;
26
27     always @(Y)
28     $display( "time =%0t \tINPUT VALUES: \t A=%b B =%b \t output value Y  =%b",$time,A,B,Y);
29 endmodule
```


ModelSim PE Student Edition 10.4a
REC
85% 2:49 PM
File Edit View Compile Simulate Add Source Tools Layout Bookmarks Window Help
Layout NoDesign
ColumnLayout
Search:
Library
Name
work
floatfixlib
ieee_env (empty)
infact
mc2_lib (empty)
mgc_ams (empty)
mtiAvm
mtiOvm
mtiPA
mtiRnm
mtiUPF
mtiUvm
osvvm
sv_std
vhdlopt_lib
vital2000
ieee
modelsim_lib
std
std_developerskit
synopsys
verilog
Transcript
Reading C:/Modeltech_pe_edu_10.4a/tcl/vsim/pref.tcl
vlog -reportprogress 300 -work work C:/Modeltech_pe_edu_10.4a/examples/verilog_01/
_example.v
Model Technology ModelSim PE Student Edition vlog 10.4a Compiler 2015.03 Apr 7
5
Start time: 14:49:53 on Sep 15,2020
vlog -reportprogress 300 -work work C:/Modeltech_pe_edu_10.4a/examples/verilog_
nd_example.v
-- Compiling module and_gate
-- Compiling module tb_and_gate
#
Top level modules:
tb_and_gate
End time: 14:49:54 on Sep 15,2020, Elapsed time: 0:00:01
Errors: 0, Warnings: 0
ModelSim>
Compile Source Files
Library: work
Look in: verilog_01
Quick access
Desktop
Libraries
This PC
Network
Name
Date modified
and_example
14-09-2020 10
xor_example
14-09-2020 10
File name: and_example
Files of type: HDL
Compile selected files together
meet.google.com is sharing your screen.
Stop sharing
Hide
You
<No Design Loaded>
work
02:50 PM


ModelSim PE Student Edition 10.4a
File Edit View Compile Simulate Add Structure Tools Layout Bookmarks Window Help
REC
sim - Default
Instance
Design unit
tb_and_gate
a1
#INITIAL#18
#ALWAYS#24
#ALWAYS#25
#ALWAYS#27
#vsim_capacity#
Processes (Active)
Name
Type
#ASSIGN#5
#INITIAL#18
#ALWAYS#24
#ALWAYS#25
Transcript
vlog -reportprogress 300 -work work C:/Modeltech_pe_edu_10.4a/examples/verilog_01/and_example.v
-- Compiling module and_gate
-- Compiling module tb_and_gate
Top level modules:
tb_and_gate
End time: 14:49:54 on Sep 15,2020, Elapsed time: 0:00:01
Errors: 0, Warnings: 0
ModelSim> vsim -gui work.tb_and_gate
// ModelSim PE Student Edition 10.4a Apr 7 2015
// Copyright 1991-2015 Mentor Graphics Corporation
// All Rights Reserved.
// THIS WORK CONTAINS TRADE SECRET AND PROPRIETARY INFORMATION
// WHICH IS THE PROPERTY OF MENTOR GRAPHICS CORPORATION OR ITS
// LICENSORS AND IS SUBJECT TO LICENSE TERMS.
// THIS DOCUMENT CONTAINS TRADE SECRETS AND COMMERCIAL OR FINANCIAL
// INFORMATION THAT ARE PRIVILEGED, CONFIDENTIAL, AND EXEMPT FROM
// DISCLOSURE UNDER THE FREEDOM OF INFORMATION ACT, 5 U.S.C. SECTION 552.
// FURTHERMORE, THIS INFORMATION IS PROHIBITED FROM DISCLOSURE UNDER
// THE TRADE SECRETS ACT, 18 U.S.C. SECTION 1905.
// NOT FOR CORPORATE OR PRODUCTION USE.
// THE ModelSim PE Student Edition IS NOT A SUPPORTED PRODUCT.
// FOR HIGHER EDUCATION PURPOSES ONLY
vsim -gui
Start time: 14:51:32 on Sep 15,2020
Loading work.tb_and_gate
Loading work.and_gate
meet.google.com is sharing your screen.
Stop sharing
Hide
You
Library
Instance
sim
Type here to search
15-09-2020


REC
AND EXEMPT FROM
// DISCLOSURE UNDER THE FREEDOM OF IN
T, 5 U.S.C. SECTION 552.
// FURTHERMORE, THIS INFORMATION IS E
OM DISCLOSURE UNDER
// THE TRADE SECRETS ACT, 18 U.S.C. S
//
// NOT FOR CORPORATE OR PRODUCTION USE
// THE ModelSim PE Student Edition IS
TED PRODUCT.
// FOR HIGHER EDUCATION PURPOSES ONLY
log -r Blocking
log -r Blocking.bak
log -r blocking_nonblock.v
log -r dump.vcd
is sharing your screen.
Stop sharing
Hide
VSIM 4> log -r *
You