Steps to run compxlib to compile Xilinx libraries in Modelsim SE 10.1 for EE101/EE201L/EE560 students as well as USC ITS

1 Finish installing Xilinx ISE 13.2 and ModelSim SE 10.1

1.1 We assume that you have successfully installed and tested the above two tools. **EE101** students should have installed ISE WebPACK 13.2 and **EE201L/EE560** students should have installed ISE 13.2 (Logic Edition). All should have installed Modelsim SE 10.1. The following procedure is common to all students.

1.2 This procedure is for compiling Xilinx libraries in ModelSim SE 10.1 and setting the Modelsim SE as the tool for simulation in the Xilinx ISE 13.2. Since Modelsim SE is used on the PCs in OHE336/RTH/SAL machines, the administrators of the PCs in OHE336/RTH/SAL user rooms may please follow these steps. Also students who installed Modelsim SE (and not PE student edition) will follow these steps.

2 compxlib

2.1 The procedure is a little murky so please follow the steps carefully.

2.2 First set Modelsim SE as the default simulation tool in ISE 13.2

2.2.1 Find the Modelsim execution path first. Right-click on the Modelsim desktop icon and select properties. In the properties dialog box, copy the executable path

![Modelsim Properties Path](image)

2.2.2 I have provided you a design "adder4bit_post_route_simulation". Place this design in C:\Xilinx_projects folder. Invoke Xilinx ISE Design Suite 13.2 and open this project.
2.2.3 In the ISE Project Navigator window, go to Edit => Preferences => ISE General => Integrated Tools". In the field, Model Tech Simulator, paste the Modelsim execution path you copied a little before, click "Apply" followed by "OK". Verify again by opening this dialogue box.
2.2.4 Also set Modelsim as the default simulator in the Design Properties dialog box.

In the implementation view, right click as shown below and select Design properties.

2.3 Now let us do the compilation of the xilinx libraries in Modelsim SE 10.1.
Open the command line window on your PC. Start => All Programs => Accessories => Command Prompt
In the command window change directory and invoke compxlib. Note: If your system is a 32-bit system, you would change directory to nt instead of nt64.

2.3.1 If the GUI in the next section does not let you select ModelSim SE simulator, then close the GUI and go back to the command prompt and type exactly as given below at the above prompt. (Note: you can type compxlib -help and it displays what switches mean what.)

```
compxlib -s mti_se -p C:\modeltech64_10.1\win64\ -l all -arch spartan3e -arch spartan6 -arch spartan6l -lib all -w -verbose
```

The above replaces the next 5 GUI screens.
2.3.2

2.3.3  I have chosen "Both VHDL and Verilog" while writing this installation note. ITS will also choose this option. EE101/EE201L students can choose just Verilog.

2.3.4  Spartan3E and the Spartan6 are enough for students
2.3.5
2.3.6

The displays in this step and next step will be slightly different for EE101/EE201 students as they selected just "Verilog" and not "Both VHDL and Verilog" in a prior step.

2.3.7

The displays in this step and next step will be slightly different for EE101/EE201 students as they selected just "Verilog" and not "Both VHDL and Verilog" in a prior step.
2.3.8 After about 15 minutes, it finishes compilation. Scroll down to see the following.

![Compilation Summary](image)

2.3.9 You can use Windows explorer to see that several files are produced under each of these two directories:

C:\Xilinx\13.2\ISE_DS\ISE\verilog\mti_se\10.1\nt64\n
C:\Xilinx\13.2\ISE_DS\ISE\vhdl\mti_se\10.1\nt64\n
3 Now verify by performing a Post-Place & Route simulation in ISE.
3.1

A selected portion of the waveform is displayed below. Notice the actual delays of the gates and interconnect causing transitional values in the waveform which confirms that the delays of the actual xilinx components were used by the simulation. This means that our compilation was successful!

4.0

Celebrate your successful completion of compiling xilinx libraries using modelsim SE.