ModelSim VHDL Simulator Tutorial[1]

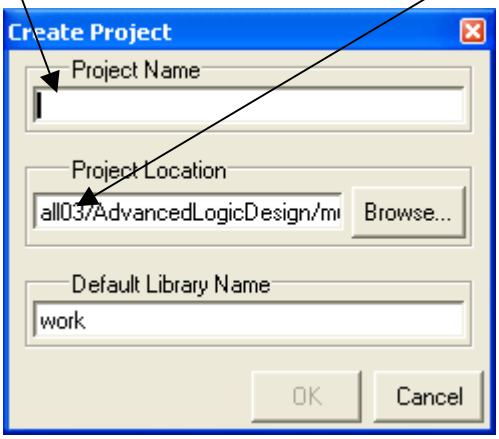
***Advanced Logic Design* 630461 – Fall 2003**

Philadelphia University

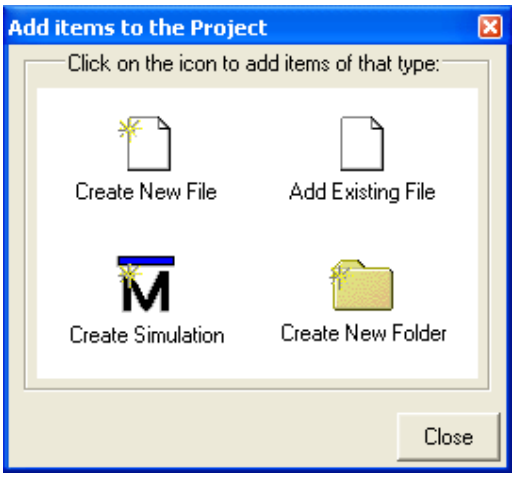
1. **When you are done creating the source files of your design, we must simulate the design** to make sure that it will function as expected. To simulate the design, we will use Mentor Graphics ***ModelSim*** Simulator. Click on the ***ModelSim*** icon on your desktop or from the program menu located in the Start button.

ModelSim XE II 5.7c.lnk

14. From the **File** menu, choose **File -> new -> project**. You will get the window below. Fill in the **Project Name** field as **Lab1.** Then fill out your working directory path in **Project Location.** (i.e. c:\msharawi\ADL\)

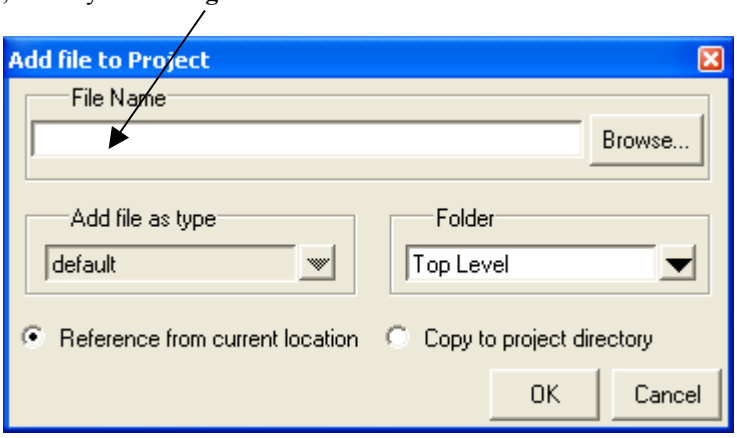


2. The following screen pops up, just HIT **CLOSE.**

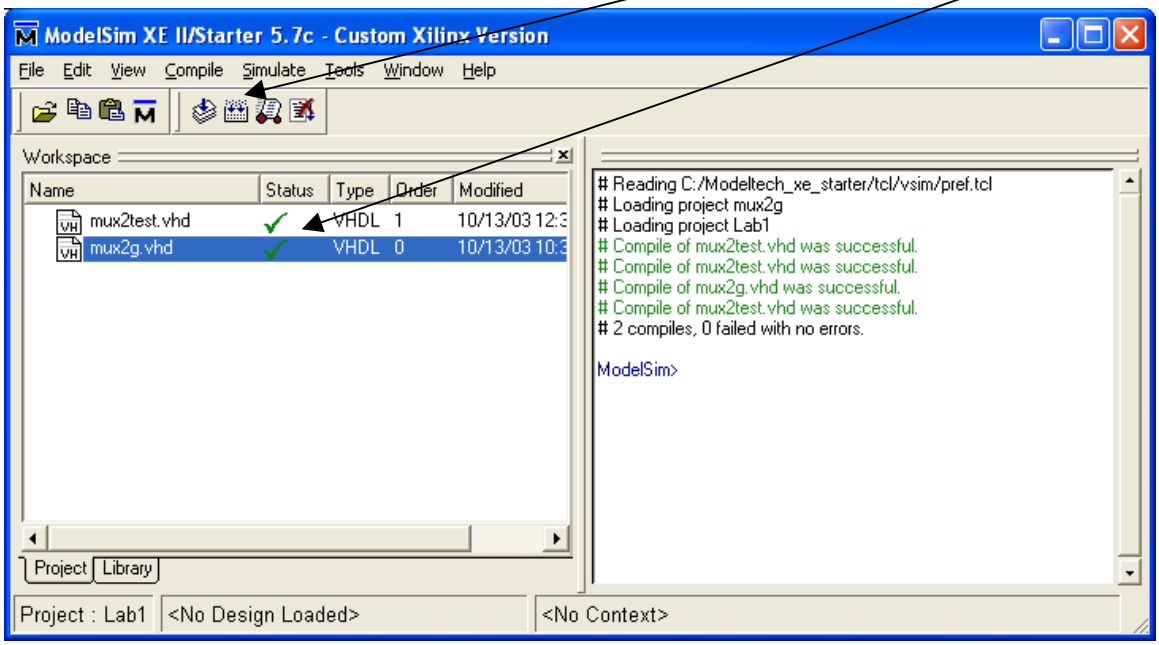


[1] **©2003**. This tutorial was created by **M. Sharawi**, *Philadelphia University*, Jordan.

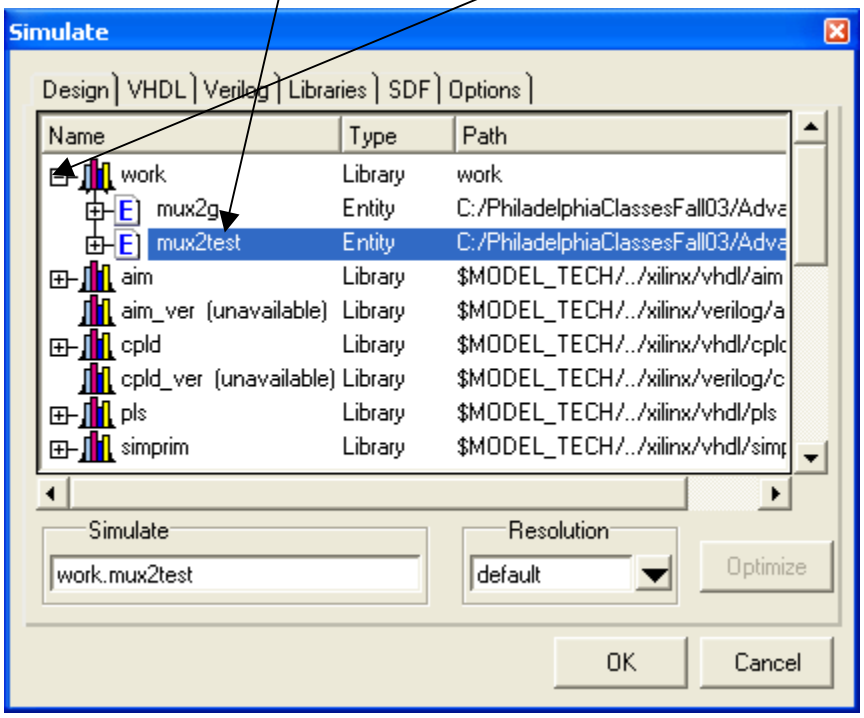
3. Now that we have created a project directory, we need to ADD our VHDL files. To add the files, we go to **File -> Add to project -> existing file.** Then we choose the files from the directory that contains it. Hit the browse button, locate your **mux2g.vhd** file.



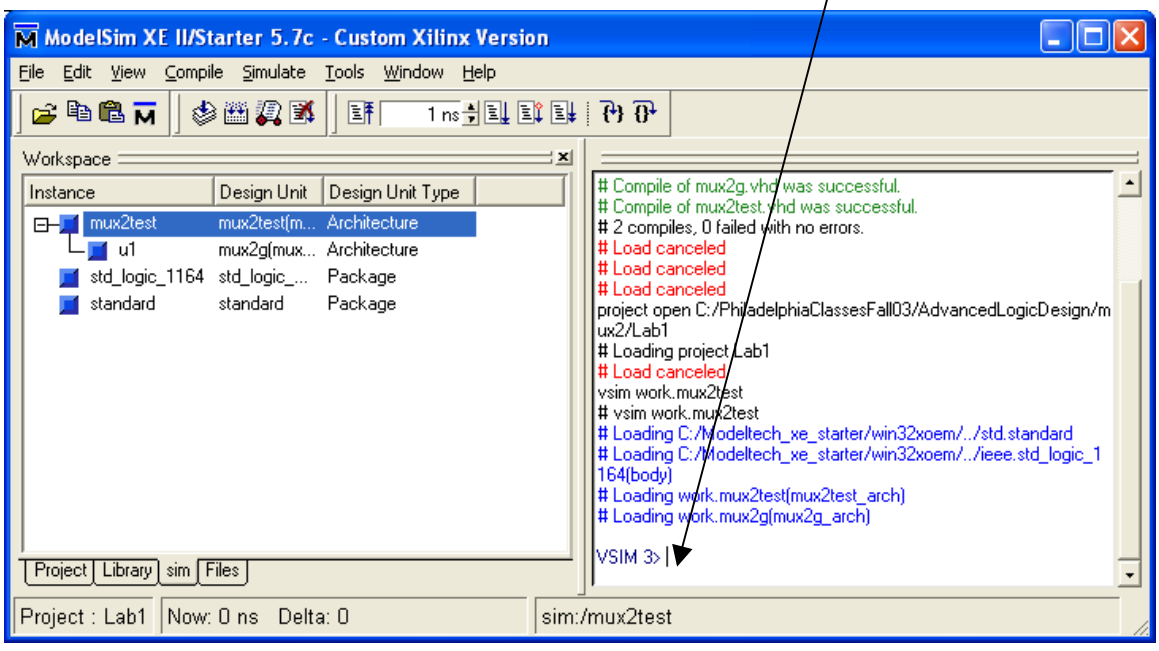
4. Once you ADD the **mux2g.vhd** and **mux2gtest.vhd** files to your project, HIT the **COMPILE ALL** button on the window. You should have both files with the check marks as indicated. If you have an **x** instead, go back and revise your code for an error.



5. After completing the compilation process, and ending up with no errors. Go to the **Simulate** menu and choose **simulate -> simulate… ,** then expand the **work folder** , you should see the two files you have compiled. Choose **mux2gtest.vhd** file, and HIT **OK.**



6. The following screen pops up. Note that the prompt was changed to **VSIM 3>**, this indicates that the simulator is ready for your commands and testing.

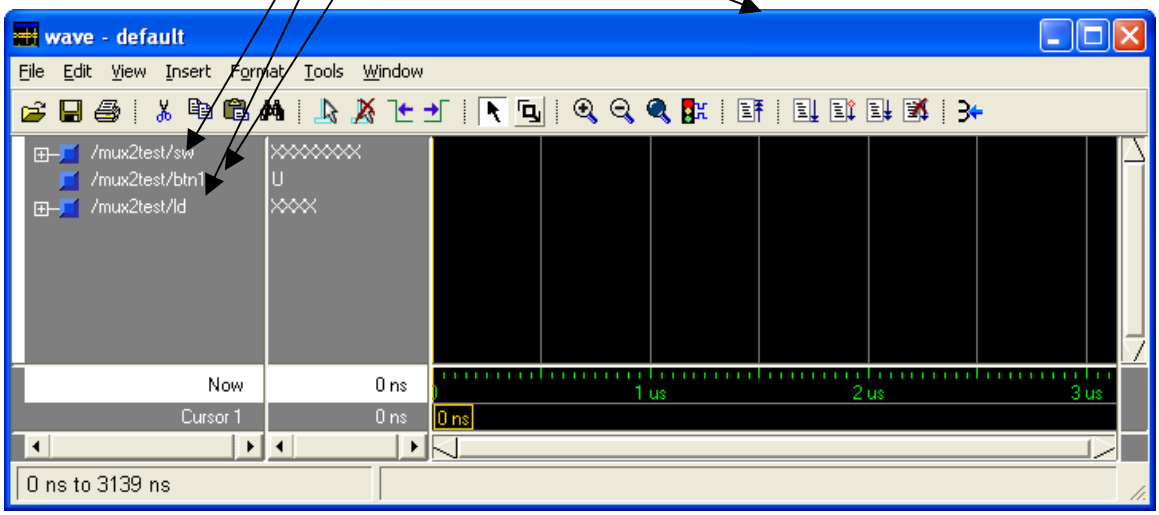


7. From you high level test file **mux2gtest.vhd**, recall that the inputs to the file and the outputs where:

SW (8 switches, the inputs ‘A’ and ‘B’)
BTN1
LD (4 LEDs , the output ‘Y’)

To run the simulation, we have to **ADD** the waveforms for every single signal so that we can view it on the *waveform viewer*. We add the signals by using the commands (*Hit ENTER after each command*). The *waveform viewer* pops up.

VSIM 3> add wave SW
VSIM 4> add wave BTN1
VSIM 5> add wave LD



8. Now we come to the part where we want to apply some test values and test the output based on the supplied input. We will use the command line commands for forcing some values into the input signals (**SW** and **BTN1**), and see the corresponding output on the **LD** signal. To force a value into a signal we use the command:

VSIM 5> force SW 16#FA 0, 16#01 50, 16#AD 100, 16#89 150 –repeat 200

Forces the value indicated — Signal name — HEX value — TIME — OTHER values at other times — Repeat values every 200 ns.

HIT ENTER after entering the above line. Then add the values for other input signals.

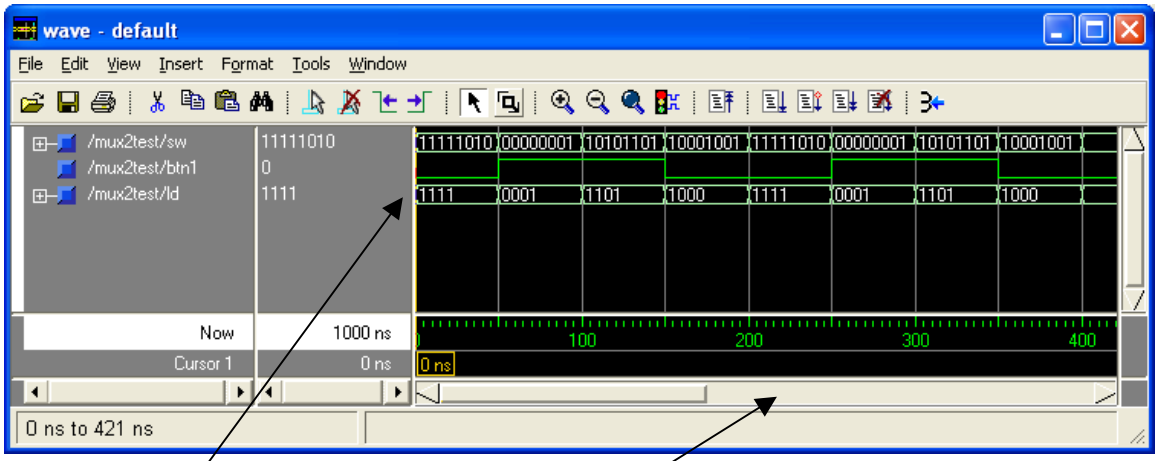
VSIM 6> force BTN1 0 0, 1 50, 1 100, 0 150 –repeat 200

The default input is in Binary for single BIT inputs.

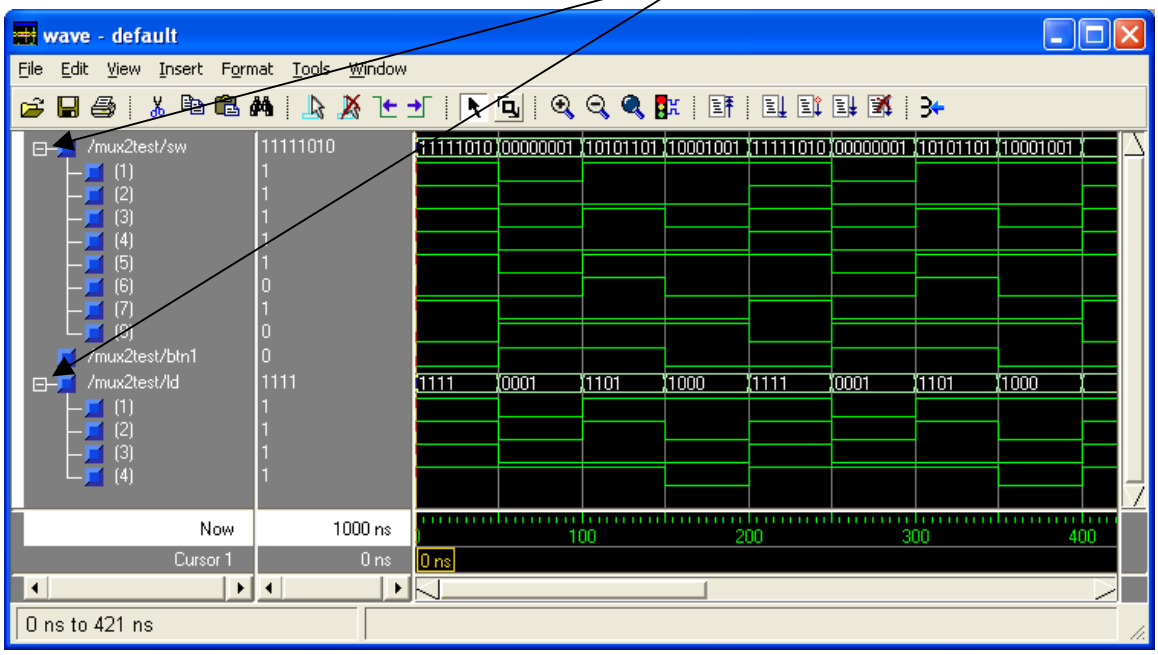
9. Once you are DONE entering the TEST values into your input signals via the FORCE command, you can run you simulation and view the output, and verify the function of operation by running the command:

VSIM 9> run 1000

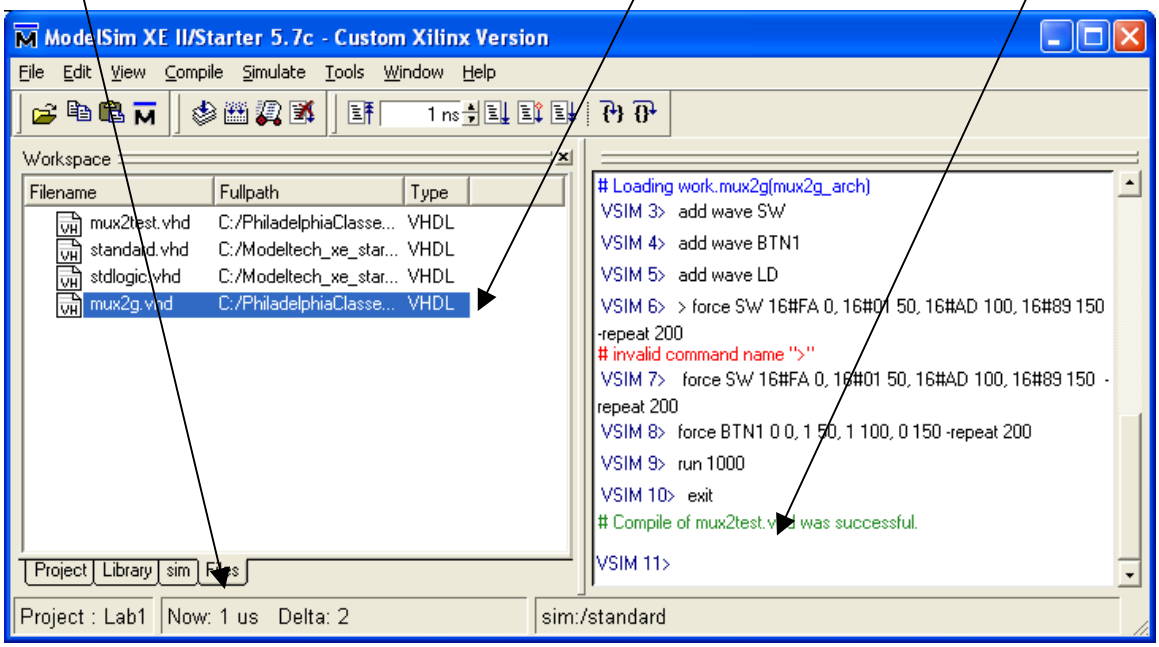
Duration of simulation in **ns**. (sometimes its in pico seconds, based on the default time set)



You can use the scroll bar to view the output at other times not shown on screen.
The output of your MUX2G circuit is shown here, verify?
You can expand the in/output signals by clicking the (+) next to each in the waveform viewer.



10. If the output was not as expected, you may edit the VHDL files directly in ***ModelSim*** by clicking on the **file** tab, and choosing double clicking on the file to be edited. Edit the file, and save it. Then recompile it.



If you want to unload the current project to edit the files, you can quit the simulator by

VSIM> quit –sim

11. This concludes the simulation process. You should verify that the output is the expected one before claiming that your design is functional. If you have any further questions, please ask your instructor.

THE END.