

As per the instructions last week:

Login to the Linux CAD system

Open a Terminal Window

Open an etransport window

Select the AMI -> BEng2 -> ECE2044 -> Walkthroughs area

To add the new files:

Select the “Create Module Area” from the etransport form

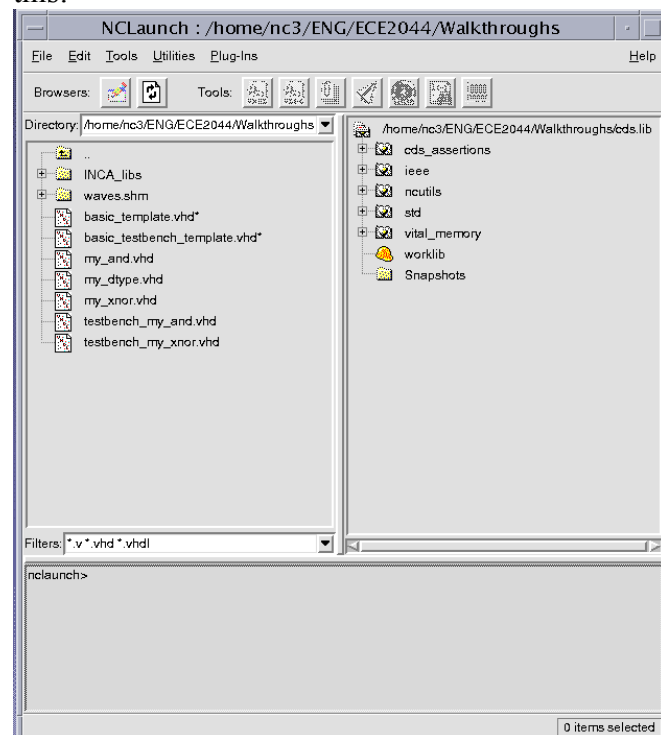
Now use the “Enter Module Area” to open up a configured terminal in the ECE2044 area.

Start the VHDL simulation tool using:

nclaunch

If you have previously configured you area you should be taken directly to the VHDL environment. If not follow the instructions from last week.

You should see the new files displayed in the tool and it should look something like this:



Exercise 1:

We are going to open and examine the my_and.vhd and testbench_my_and.vhd files.

We do this by clicking on them once to select then using the right hand side mouse button to open the context sensitive entry. We can then select “Edit” of the available options.

Resize the two windows until you are comfortable with the size.

We will now run through the structure and content in class

Setting up for Simulation

Compiling my_and.vhd

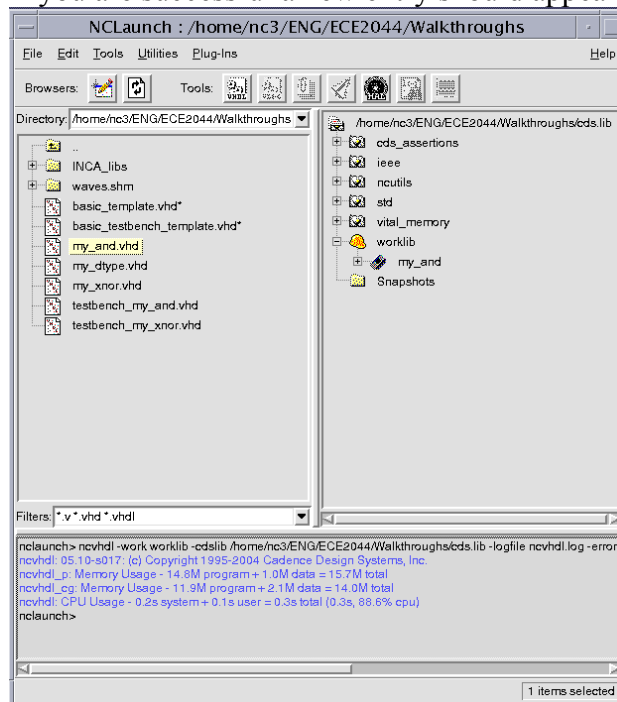
First we need to successfully compile the my_and.vhd file.

To do this left click twice on the my_and.vhd file in the file browser.

This will automatically compile the file.

Check the window at the base for any compiler error messages.

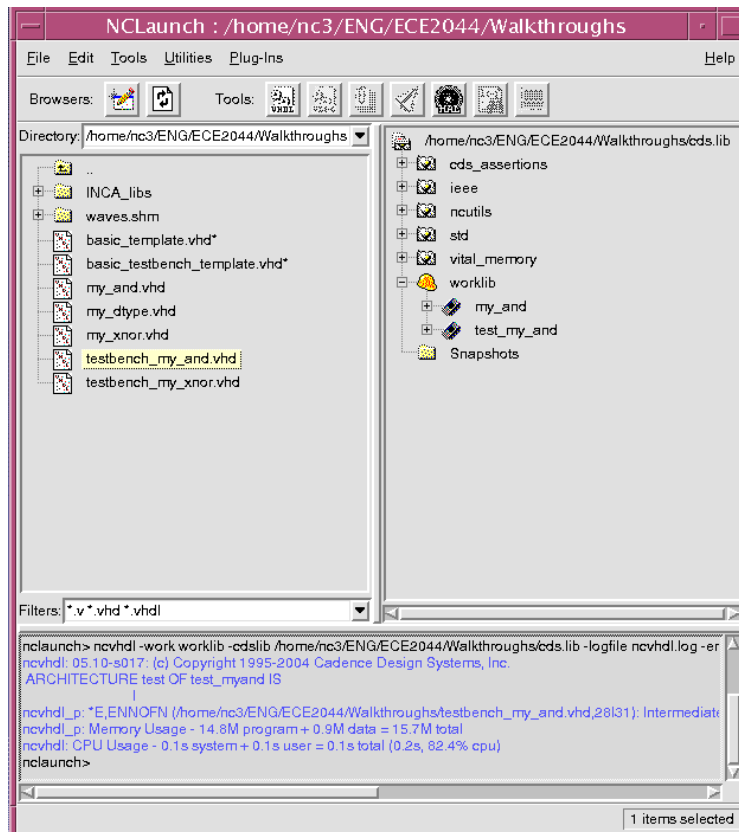
If you are successful a new entry should appear in the worklib



Compiling testbench_my_and.vhd

Now we need to compile the testbench_my_and.vhd file.

Repeat the procedure and double click on the file and again check the worklib and the base of the window for error messages.



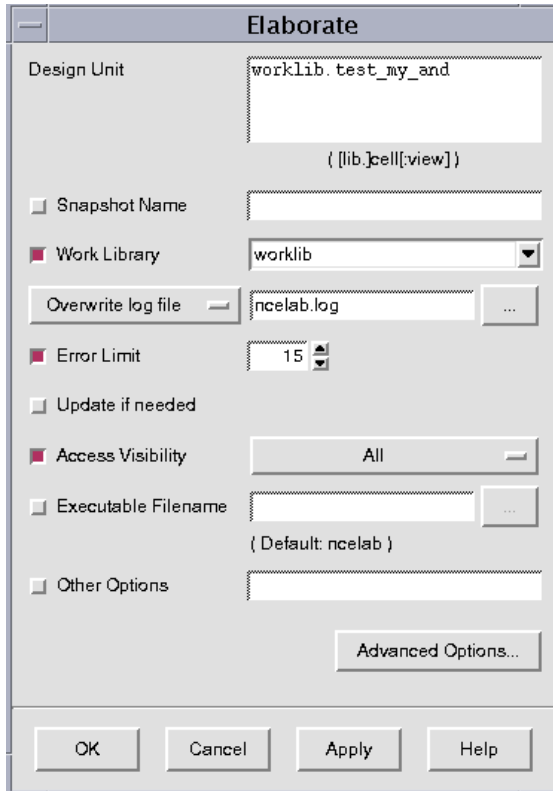
Elaborating the testbench_my_and worklib entry

We now need to “NCElab” to elaborate the test bench file

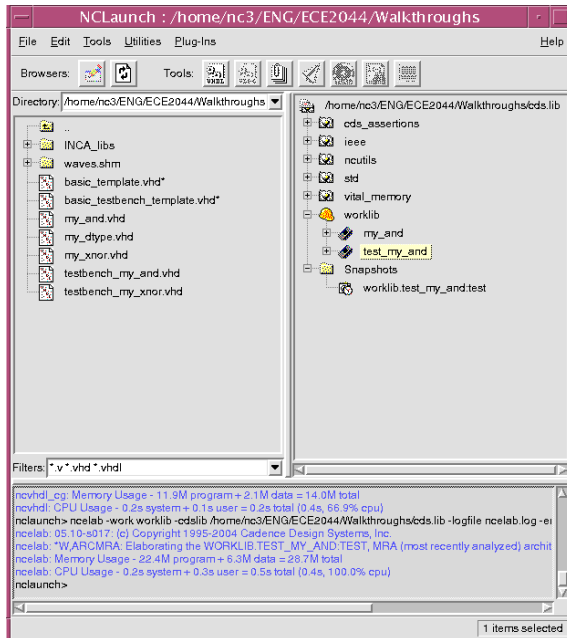
Single click on the testbench_my_and.vhd top entry in the worklib.

Use the right hand mouse button to bring up the context sensitive menu and select “Elaborate”.

This will bring up the elaboration form, “OK” this form and continue.

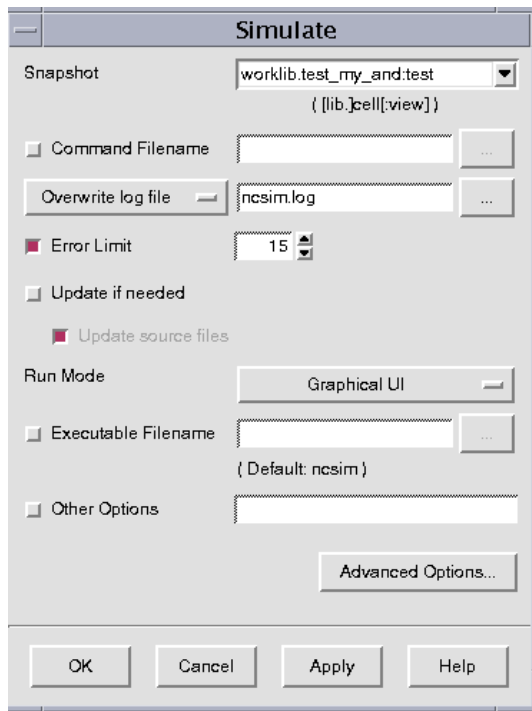


Again check the compiler window for any possible errors or problems. If this is successful then the snapshot entry below “worklib” should contain an entry for the “worklib:test_my_and:test”



Simulating the testbench_my_and snapshot

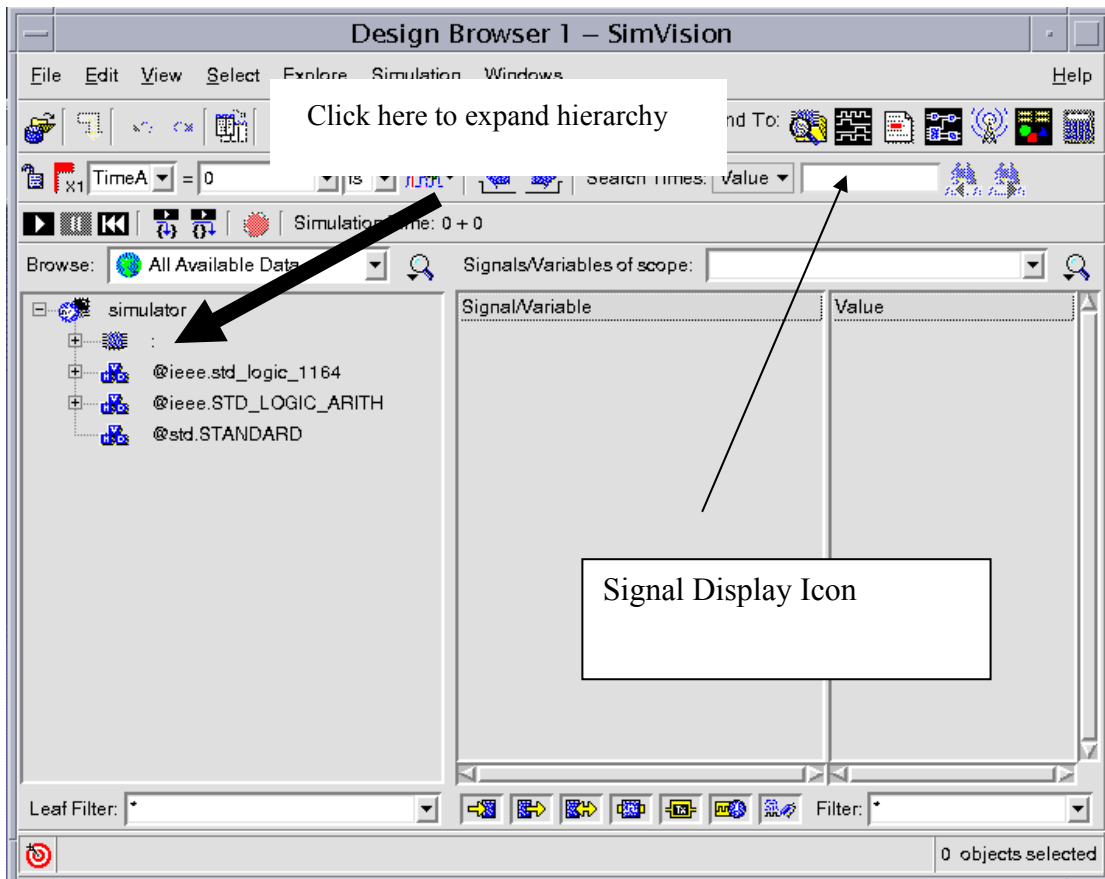
Single click on the snapshot file and use the right hand mouse button to bring up the context sensitive menu. This time select “NCSim” to open up the simulator. This will bring up “Simulate” form. “OK” this form to open up the simulation windows.



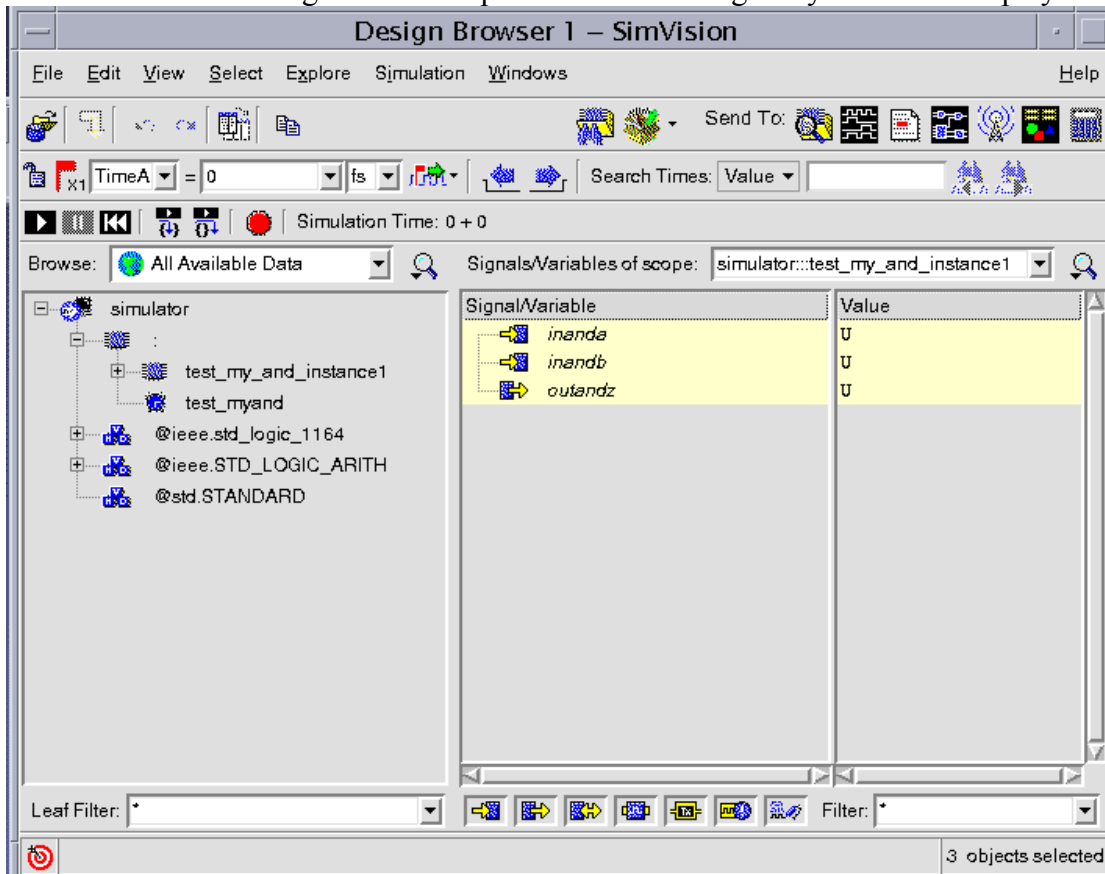
Selecting Signals and Running a Simulation

Selecting Signals

We can use the design browser window to select and display signals we are interested in. We can also use this window to move up and down the design hierarchy (when we have one) selecting signals other than at the top level.



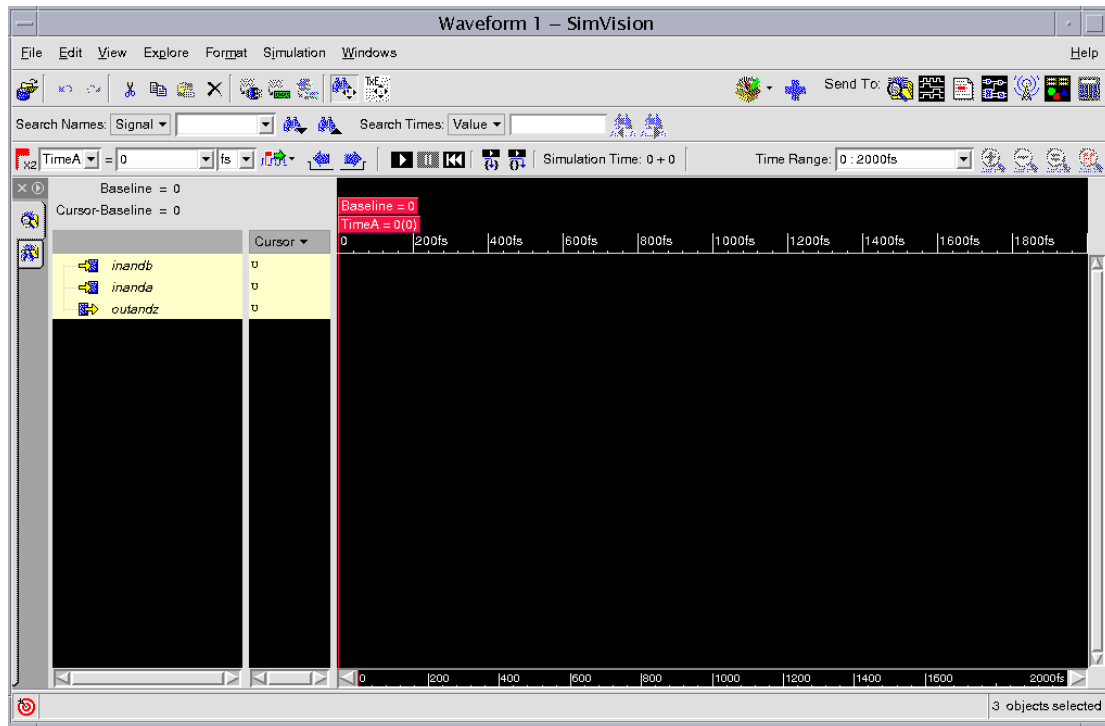
You can then use the right hand side pane to select the signals you wish to display.



Hint:

If you hold the shift key down you can select multiple signals at the same time.

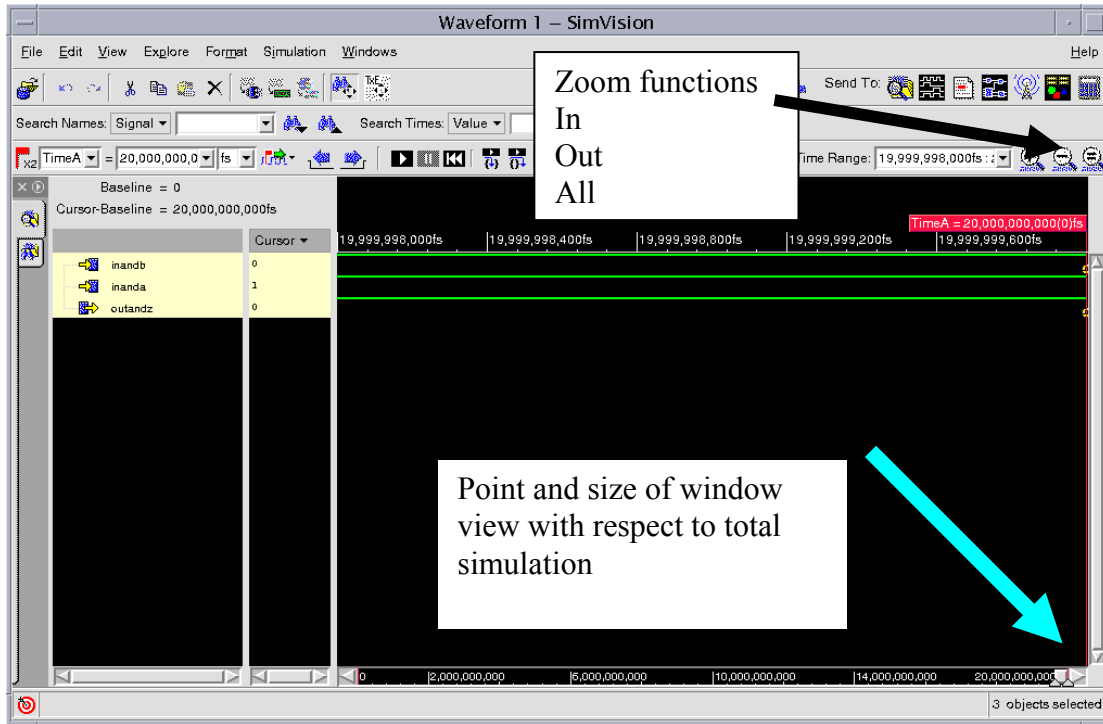
Once you have selected the signals use the Signal Display Icon to transfer the selection to the waveform viewing window, if this is not open it will be opened automatically.



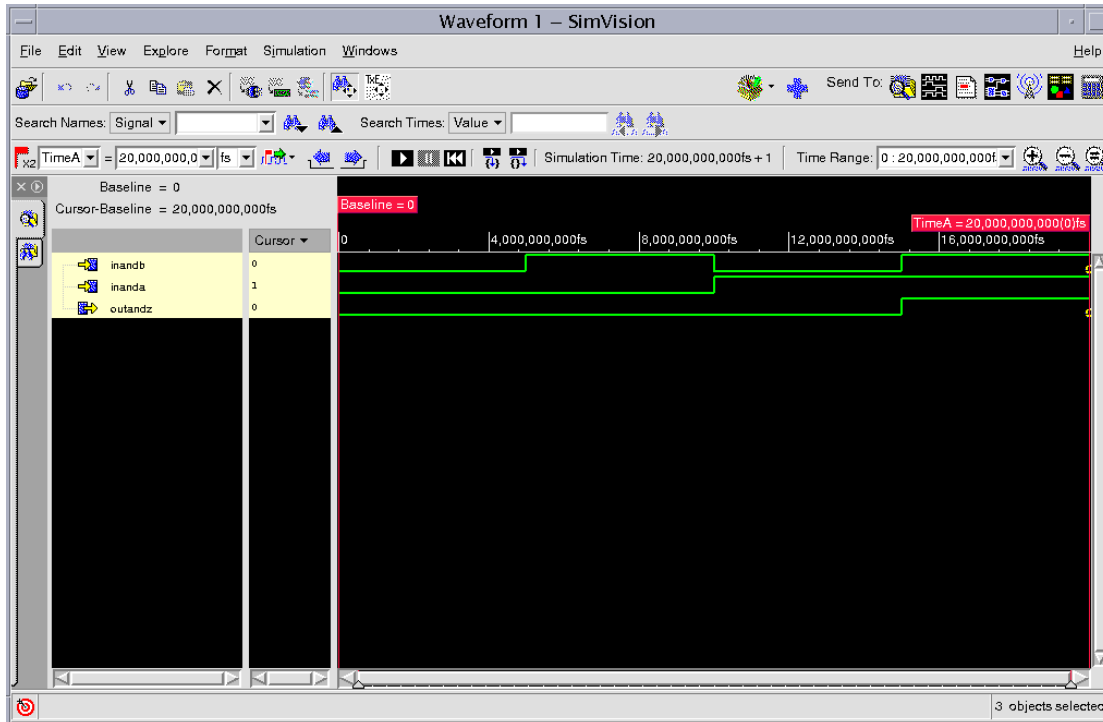
Notice that at this point there is no simulation result and the signals are “u” for undefined. We now need to run the simulation to examine the results.

Now you can run a simulation for a set time, for example using the command “run 50 us” in the console window. However in this case we added a “wait” to stop simulation in the test bench so we can just use the “Simulation->Run” option from the menu.

The result should be:



Still not very helpful. However we are not seeing the whole of the simulation. We need to view the whole simulation and we can use the Zoom -> All icon to do this.



Now we can see the inputs and the outputs from the simulation and we can check to see that the results are correct.

Summary

We should have managed to compile, elaborate and simulate the my_and.vhd and the testbench_my_and.vhd files.

You are expected to experiment and look at the options and how things work. I would strongly recommend that you make notes to refresh your memories when you use the commands again.

Additional Work to be undertaken

You can if you wish change the testbench_my_and “use” entries to select the other architectures and simulate those should you wish.

There is now additional exercise for the student to do:

1. Use the xnor templates to create an xnor system and test it to ensure correct function.
2. Create an SR Latch using the my_and gates.
 - a. You will need to add inverts using “not” as an SR latch requires nand gates!
3. Use the dtype templates to create a dtype and test it to ensure correct operation
 - a. Create one using “clock”, “indata” and “outdata”
 - b. Now extend this to include “inresetn” and “insetn”
 - c. Make sure the test bench covers all the operations

Hints

1. If the system does not understand “xor” then make sure you turn on the VHDL 93 options on the VHDL compile.
2. You can generate a clock on a test bench in the following way

```
Clock_process: Process ()  
Begin
```

```
Clock = 0,1 after 10 us;
```

```
End process;
```

**In this case you will need to use the “run NN us” command. Where “NN” is a value chosen by you!. As the clock will run for ever.
For those of you who forget the “Simulation -> Stop” option will help!**

You are not expected to finish these for “homework”, however using a VHDL manual to clarify and potential issues and trying to write your VHDL would be recommended as an exercise for the next week.