

Cadence Verilog Tutorial Windows Vista with Cygwin X Emulation

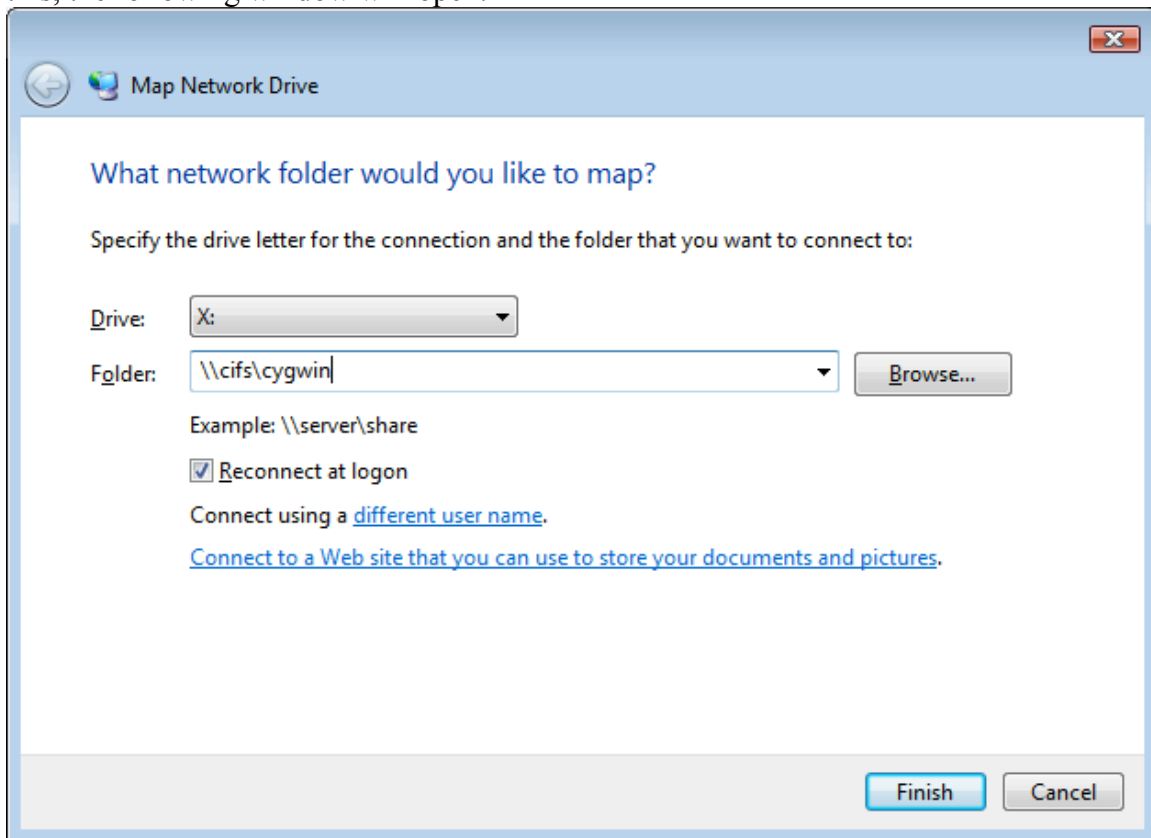
This tutorial will serve as an introduction to the use of the Cadence Verilog simulation environment and as a design tool. The Cadence design tool suite is installed on the Linux servers on our network. We will use be using the GUI interface which will allow us to view waveforms in a timing diagram. This also requires the use of X windows, meaning that you should run the program from a Linux workstation or X-terminal, or alternatively, from a PC with an X-terminal emulator installed such as Exceed or Cygwin.

The first part of this experiment will involve entering and simulating the example circuit discussed in class. This will allow you to become familiar with the Cadence Verilog environment. It will be assumed that you are working on a PC running the Windows Vista OS that has the Cygwin X-terminal emulator installed and is connected to the engr.smu.edu network.

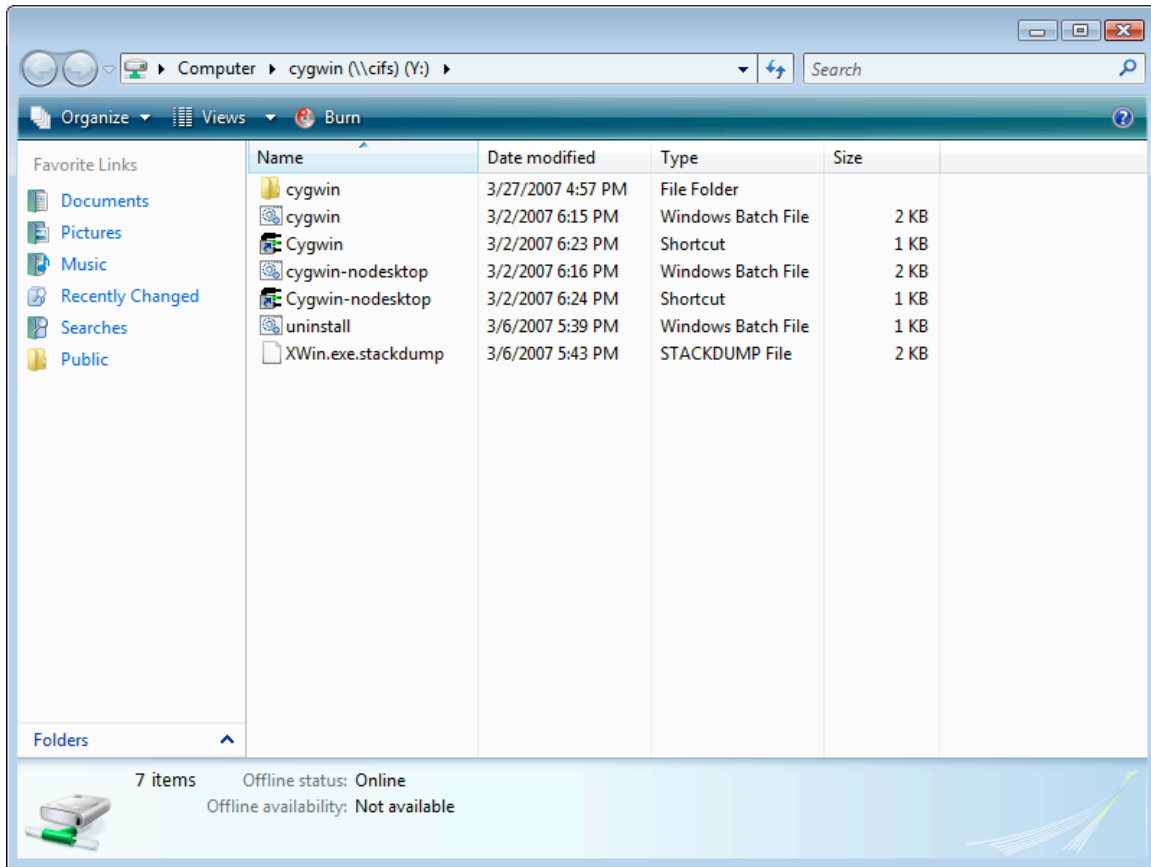
First, you will need to run a batch file on the engr.smu.edu server that configures Cygwin for your machine. To do this, map a network drive to your PC called:

[\\cifs\cygwin](#)

Right click on the “Computer” icon and select “Map Network Drive ...”. When you do this, the following window will open:



After entering [\\cifs\cygwin](#) in the “Folder” portion of the window, click on the “Finish” button at the bottom. This will cause the following window to open:



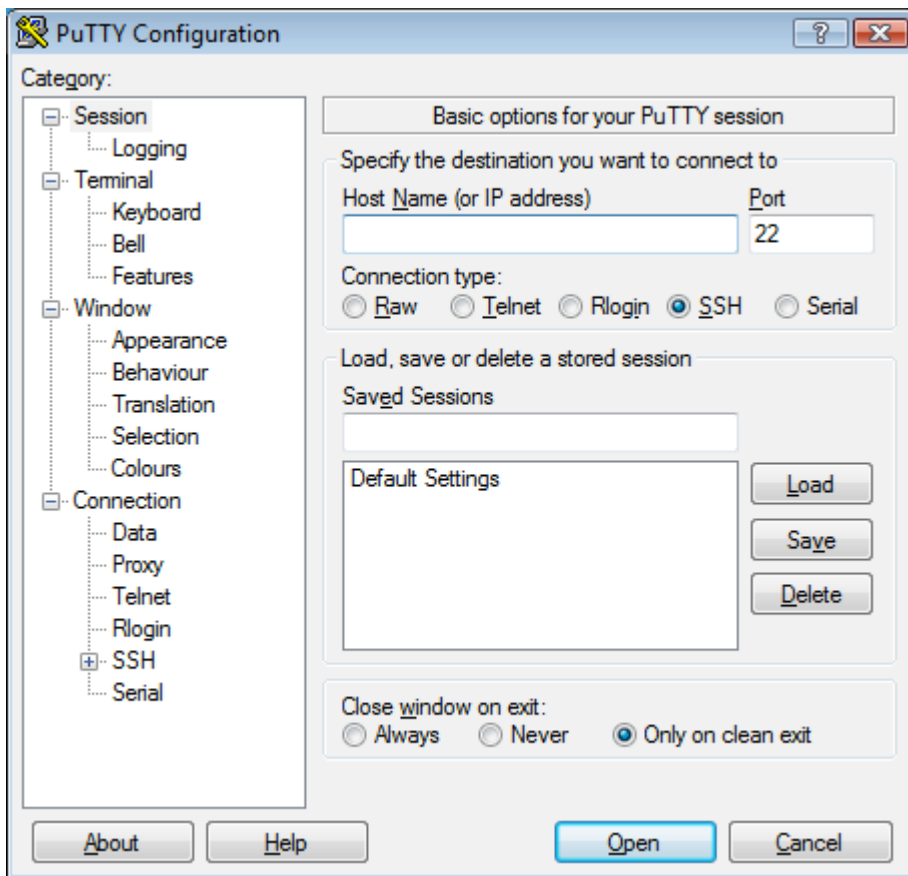
From this window, click on the cygwin-nodesktop batch file to run it. This will open a cmd window and you should see the following:

```
C:\Windows\system32\cmd.exe - \\cifs\cygwin\cygwin-nodesktop
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

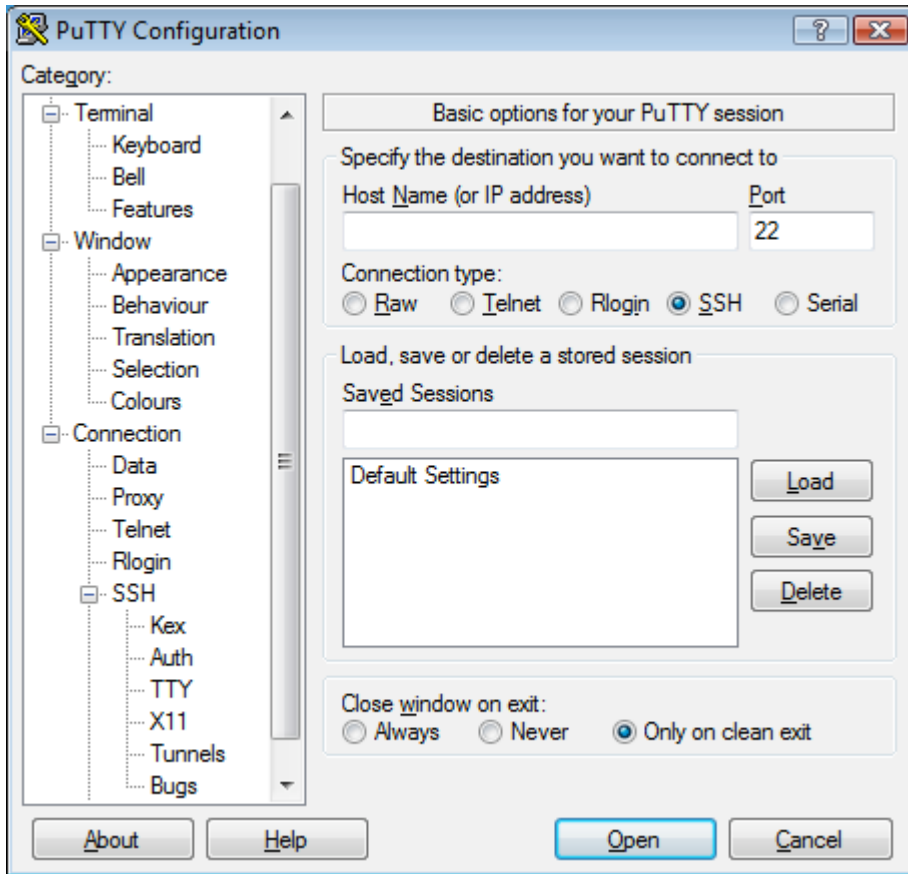
C:\Users\mitch>\\cifs\cygwin\cygwin-nodesktop
umount: remove_cygdrive_prefix: Permission denied
cygwin/bin/mkdir: cannot create directory '/tmp/mitch-S312-PC03': File exists
...
... It will take a while to start X Server
... Please be patient
...
... After you quit X Server please press anykey to start clean-up process ...
```

Next, MINIMIZE the cmd – do not exit the cmd window.

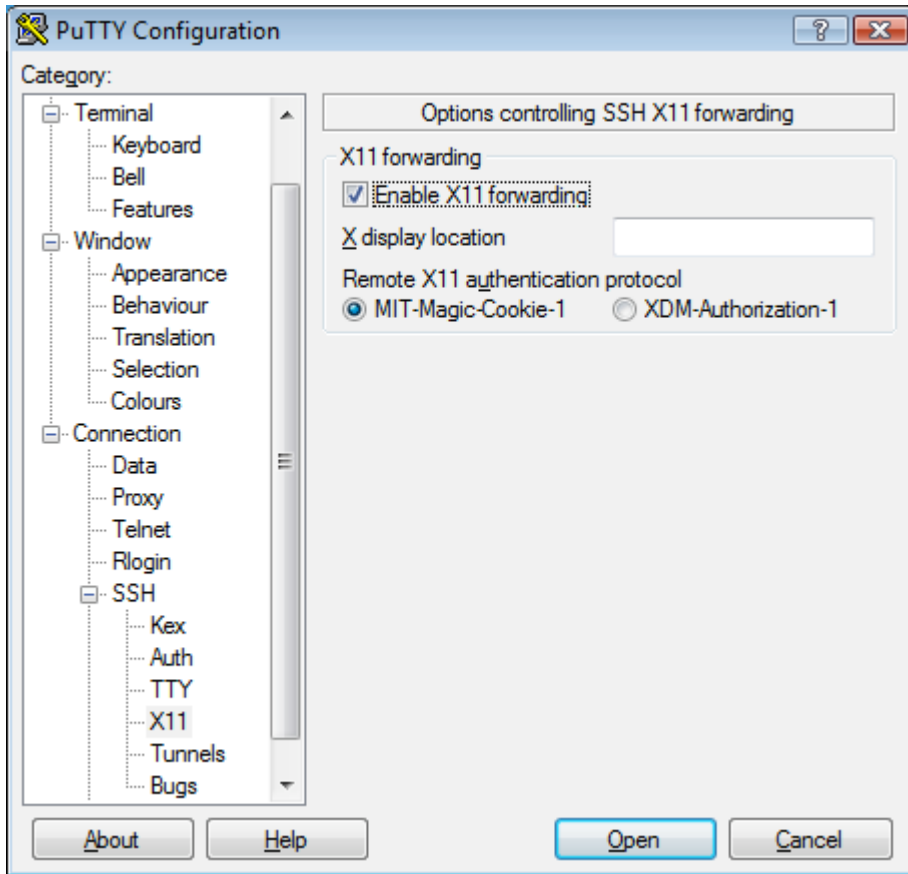
At this point, you are ready to connect to a linux compute server using the PuTTY application. From the list of installed programs on your Windows Vista main menu (All Program), select the PuTTY application and run it. This should open a window that looks like this:



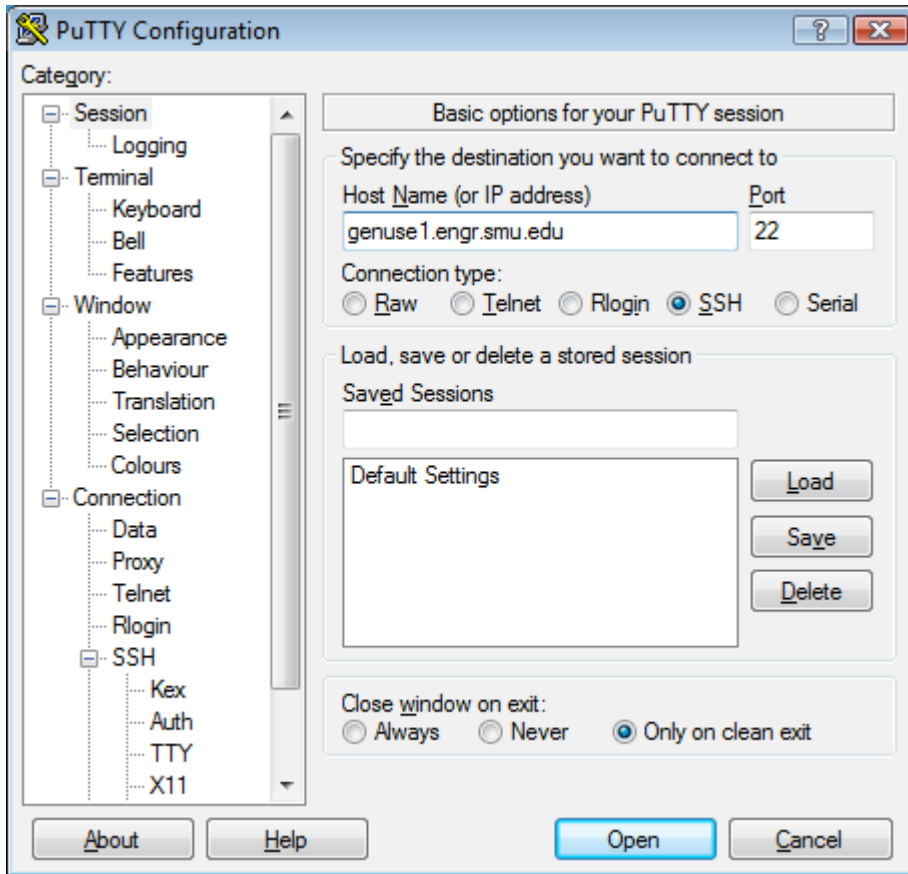
Before connecting to a linux machine, you must configure the SSH to accept X windows data. To do this expand the “SSH” option on the left side of the PuTTY menu by clicking on the “+” option. This should open the following:



Now click on the “X11” option on the menu on the left. This should change the PuTTY window and you should click on “Enable X11 Forwarding”:



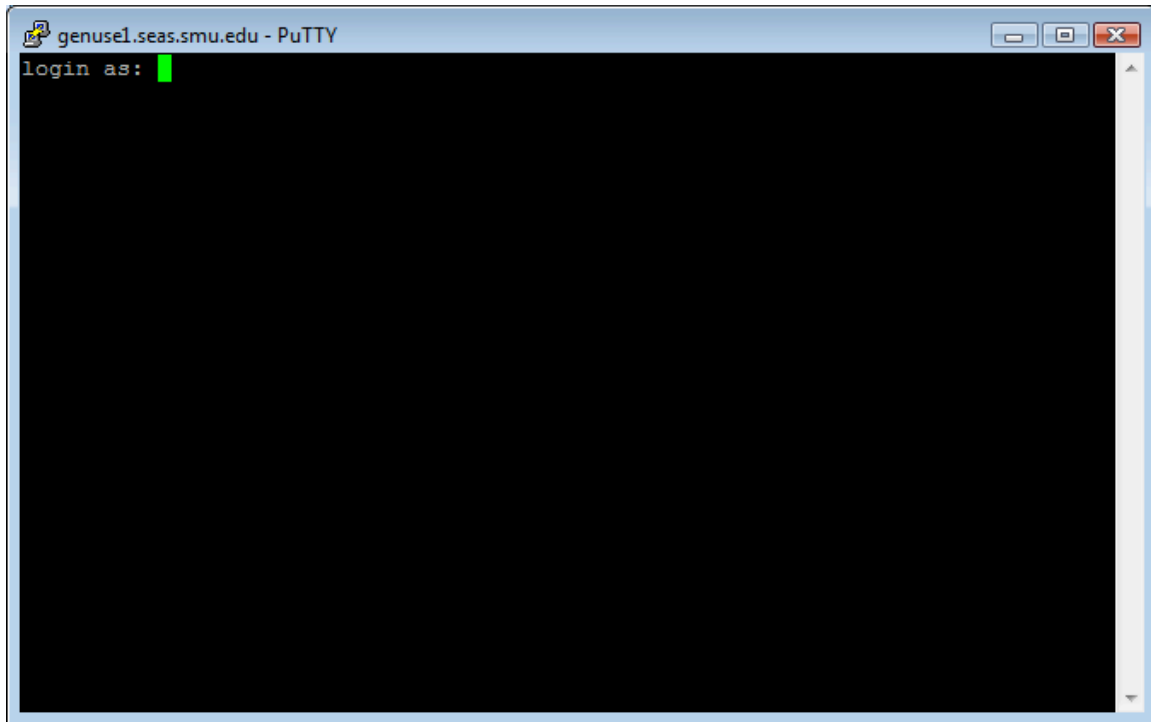
Next, click on the “Session” option on the left side of the PuTTY window and enter the hostname of a linux server:



In this example, I entered the hostname “genuse1.engr.smu.edu”; however there are many other servers you may use. A complete list can be found on the webpage:

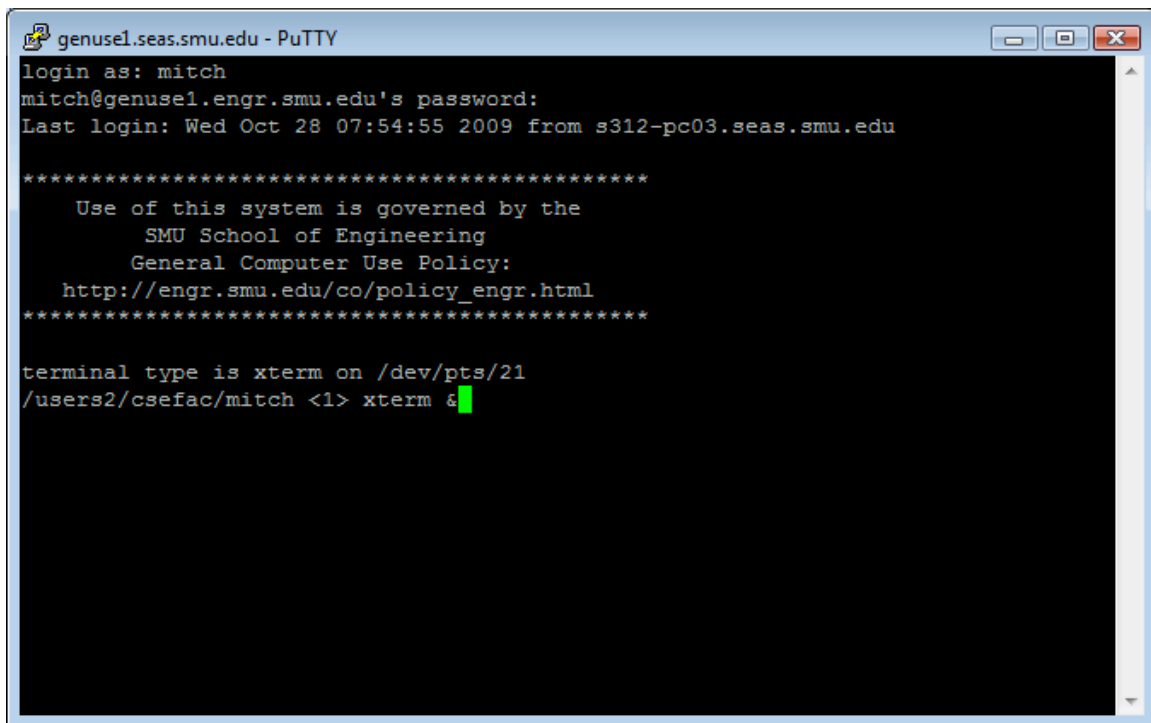
<http://lyle.smu.edu/co/unix.html>

Click on the “Open” button on the bottom of the PuTTY window and a new command window will open prompting you for your login and password for the linux server you selected:

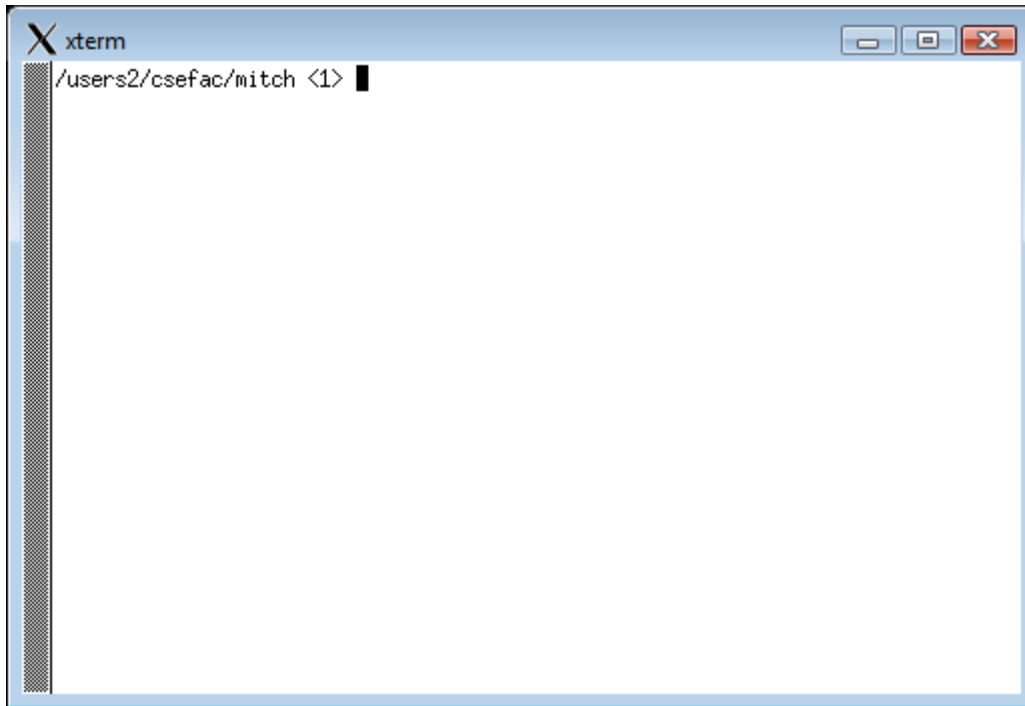


After logging in from this window, you need to start an X-terminal session, by entering the command:

xterm &



When you type this command, the X-terminal window will open as shown below. You are now emulating an X session on windows and can now run the Cadence application.



At this point, create a working subdirectory and cd into it. The next step is to create the verilog source file that you intend to simulate. You will use any text editor you desire to create this in your UNIX account (such as vi, emacs, pico, nedit, etc.). Create a file called **example.v** (Verilog source files traditionally have an extension type of .v) containing the following lines of code:

```
// Stimulus for simple circuit
module stimcrct;
reg A, B, C;
wire x, y;
circuit_with_delay cwd (A, B, C, x, y);
initial
  begin
    $stop;
    A=1'b0; B=1'b0; C=1'b0;
    #100
    A=1'b1; B=1'b1; C=1'b1;
    #100
    $stop;
  end
endmodule
// Description of circuit with delay
module circuit_with_delay (A,B,C,x,y);
```



```

    input  A,B,C;
    output x,y;
    wire e;
    and    #(30) g1(e,A,B);
    not    #(10) g2(y,C);
    or     #(20) g3(x,e,y);
endmodule

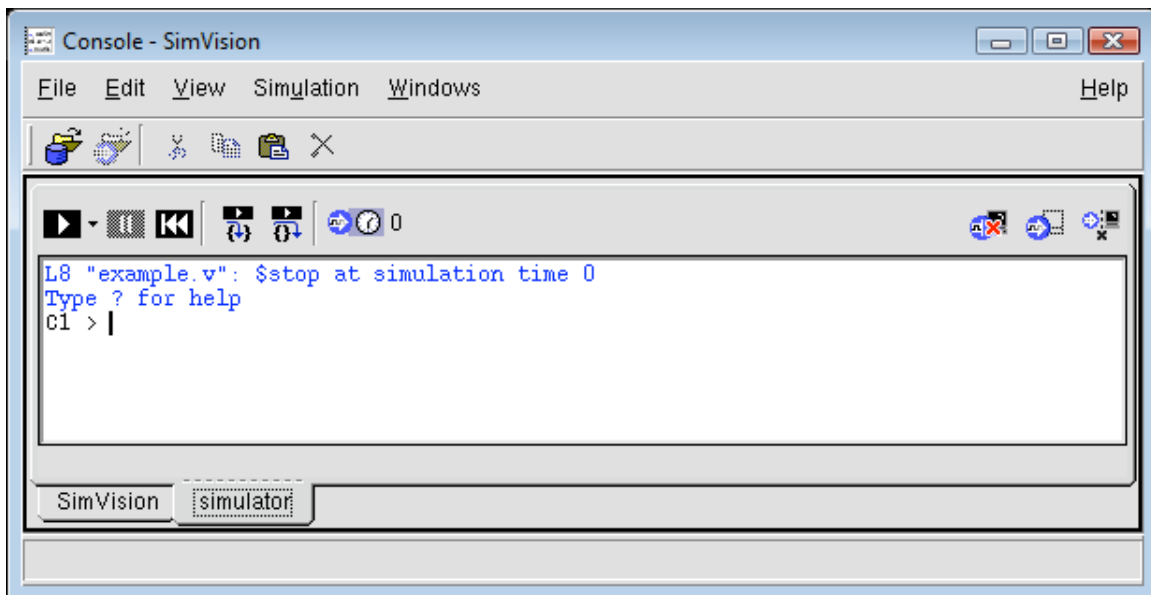
```

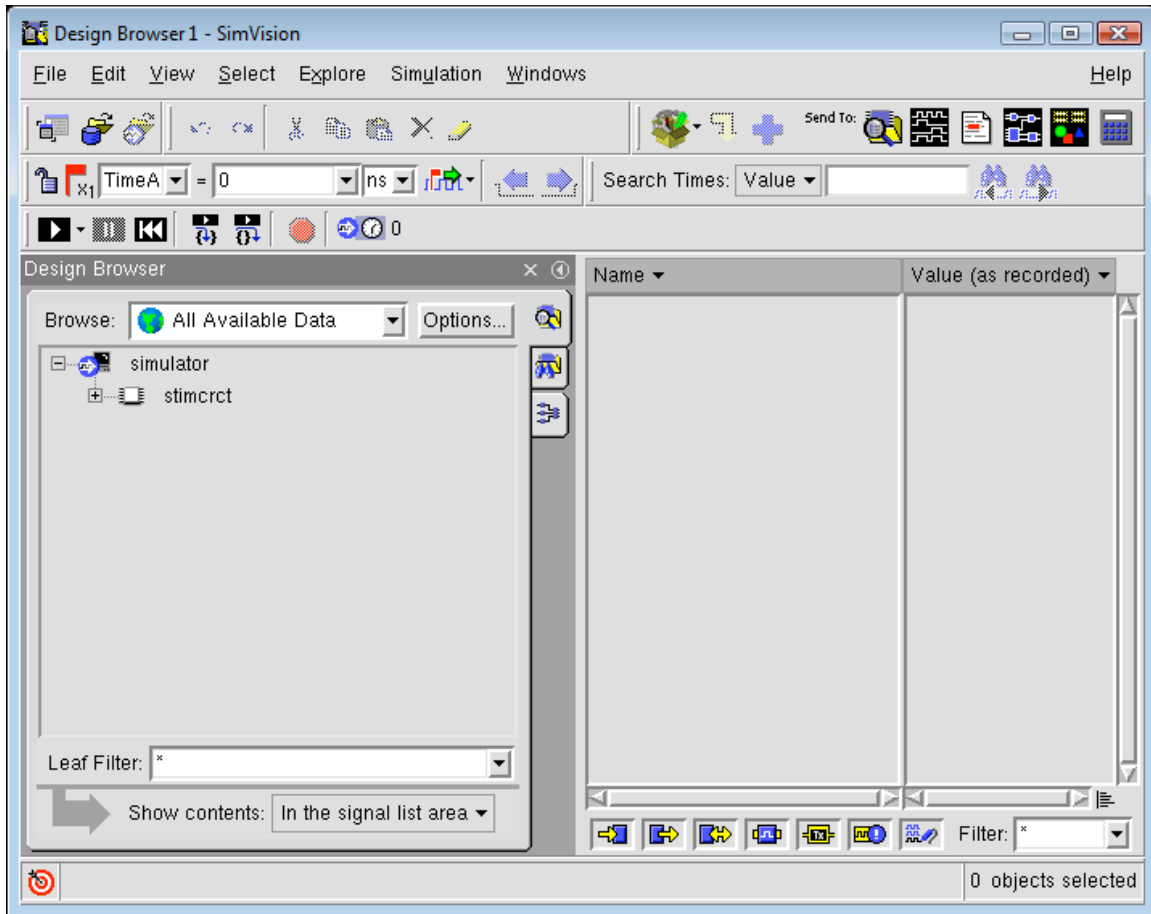
There are some important differences in this file as compared to the one we studied in the class. Instead of **\$finish** we are using **\$stop**. **\$finish** instructs the simulation tool to exit as soon as the simulation is complete. Because we are interested in looking at waveforms, we want the simulation to stop after 200 time units have elapsed but we do not want the tool to close.

We are now ready to simulate the Verilog file. From the xterm command line, enter the following command:

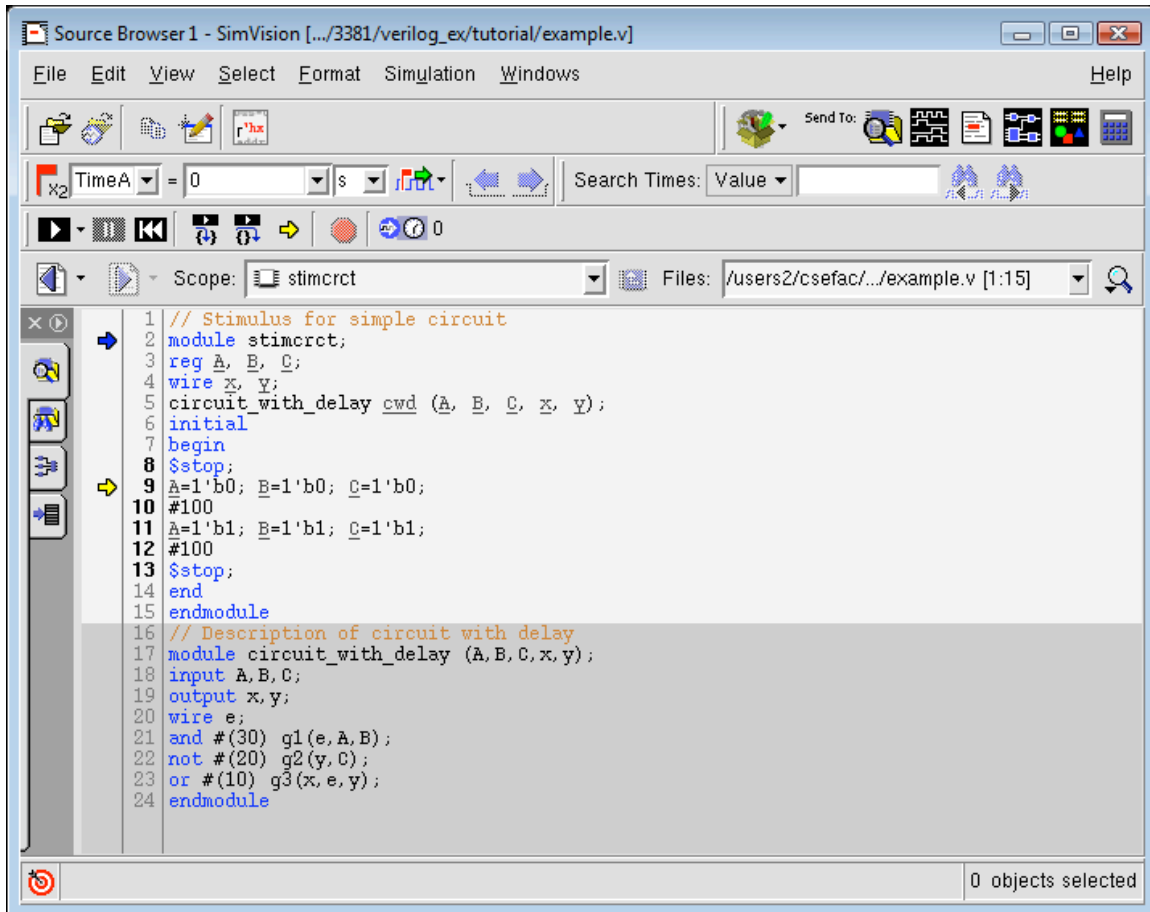
```
verilog +gui example.v &
```

The following two windows will open and the verilog file will be simulated (if no errors are found). If errors are found, you should note the message and exit the simulator and fix your source file.



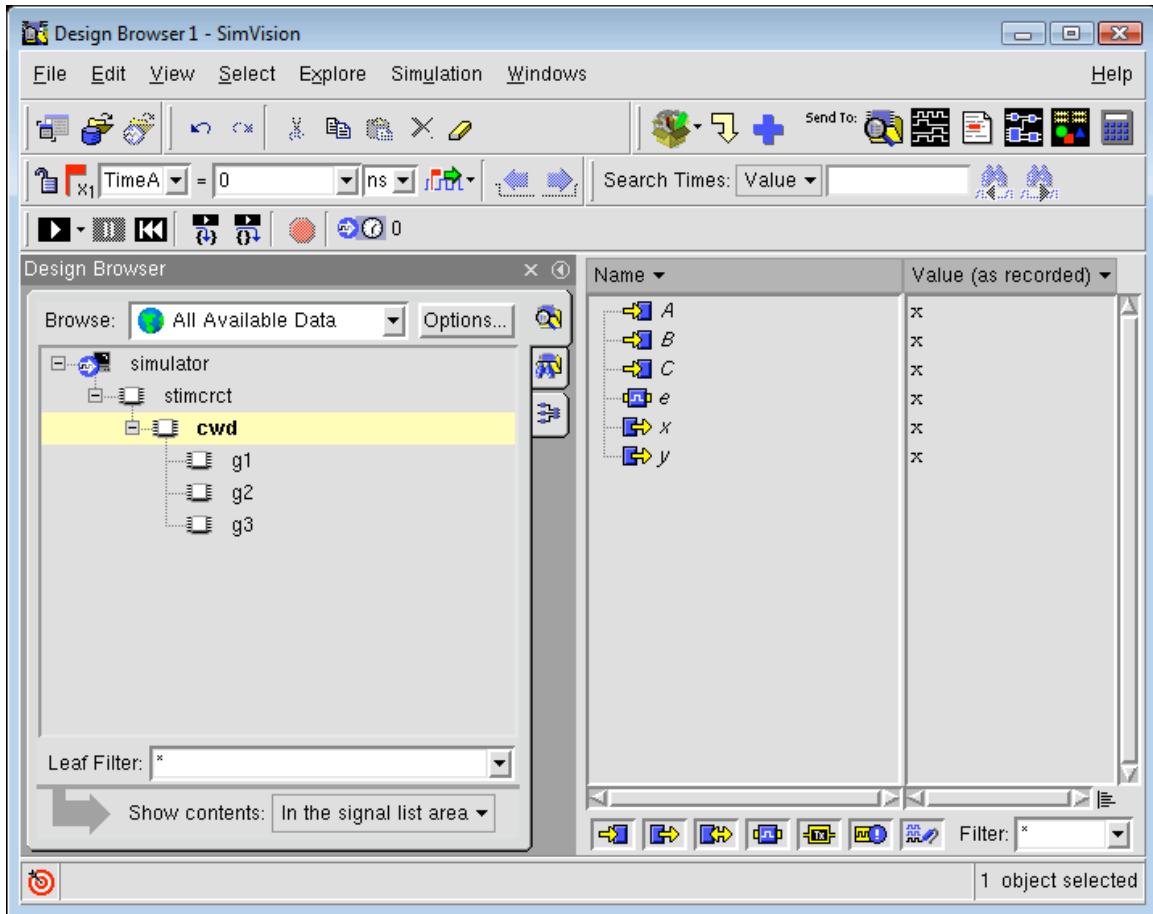


From the SimVision Console Window, you may choose the “Open Source File” option under the “File” menu to view your Verilog code and modify it if you wish.

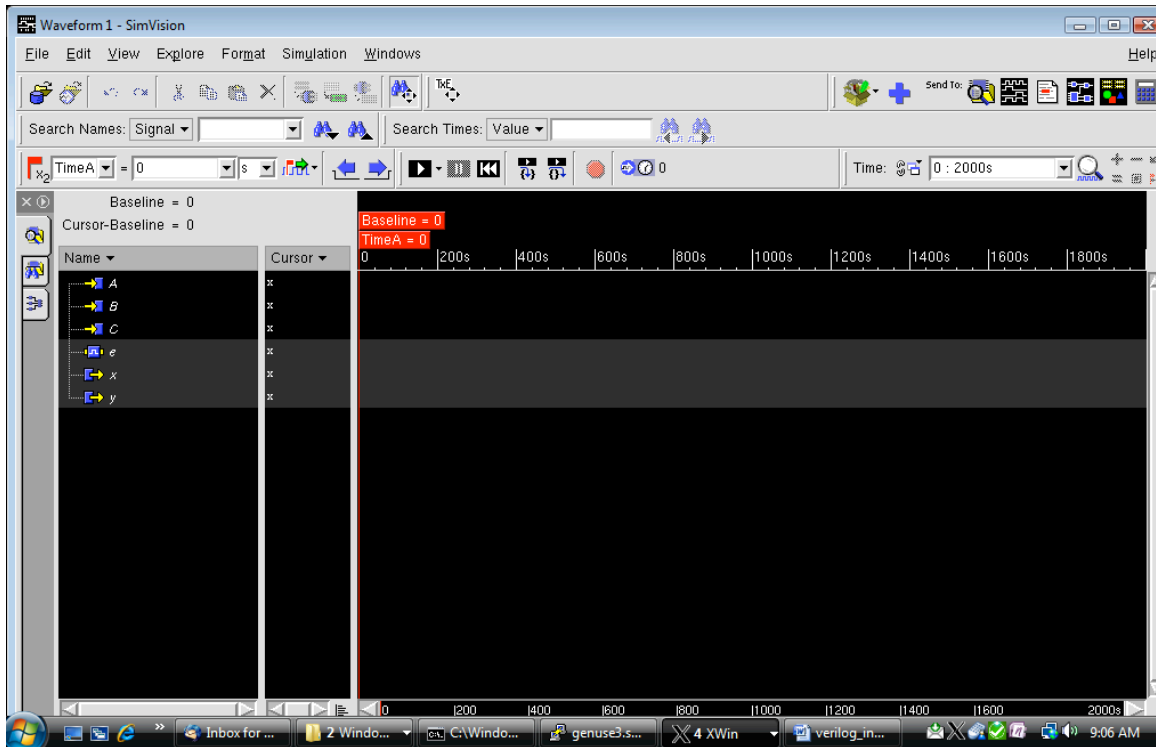


There are several things to note on Design Browser window. An important button is the one in the upper right-hand corner labeled “Help”. Clicking on this button will open the Cadence help utility and display a list of topics on the use of the Verilog tools. Notice that the SimVision Console window indicates that the simulation has stopped at time 0. This is because we used a \$stop that was scheduled to occur at time 0 in the source file. This was inserted in the course file on purpose so that we can define the signals that we wish to display on the waveform viewer before the entire module is simulated.

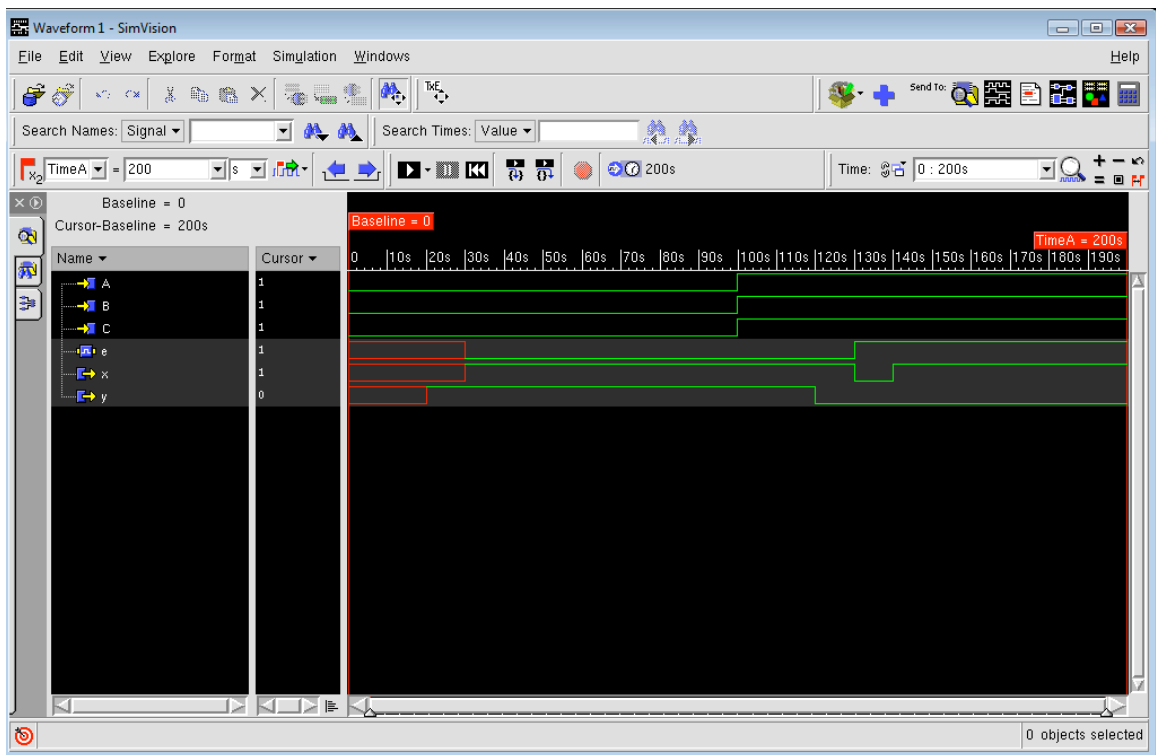
Next, we want to view the waveforms. To do this, we will use the SimVision Design Browser window and we will click on the “+” next to stimcrct in the left-hand portion of the window. This will cause cwd to be displayed underneath stimcrct. Click on the “+” next to cwd and you will see “g1”, “g2”, and “g3” displayed. Finally click on “cwd”, this will cause all signals to be displayed in the right-hand side of the window as shown below:



Next, the waveform viewer must be opened. Click on the small icon at the top of the Design Browser Window with square waves on it. You should see a window that looks like the following:

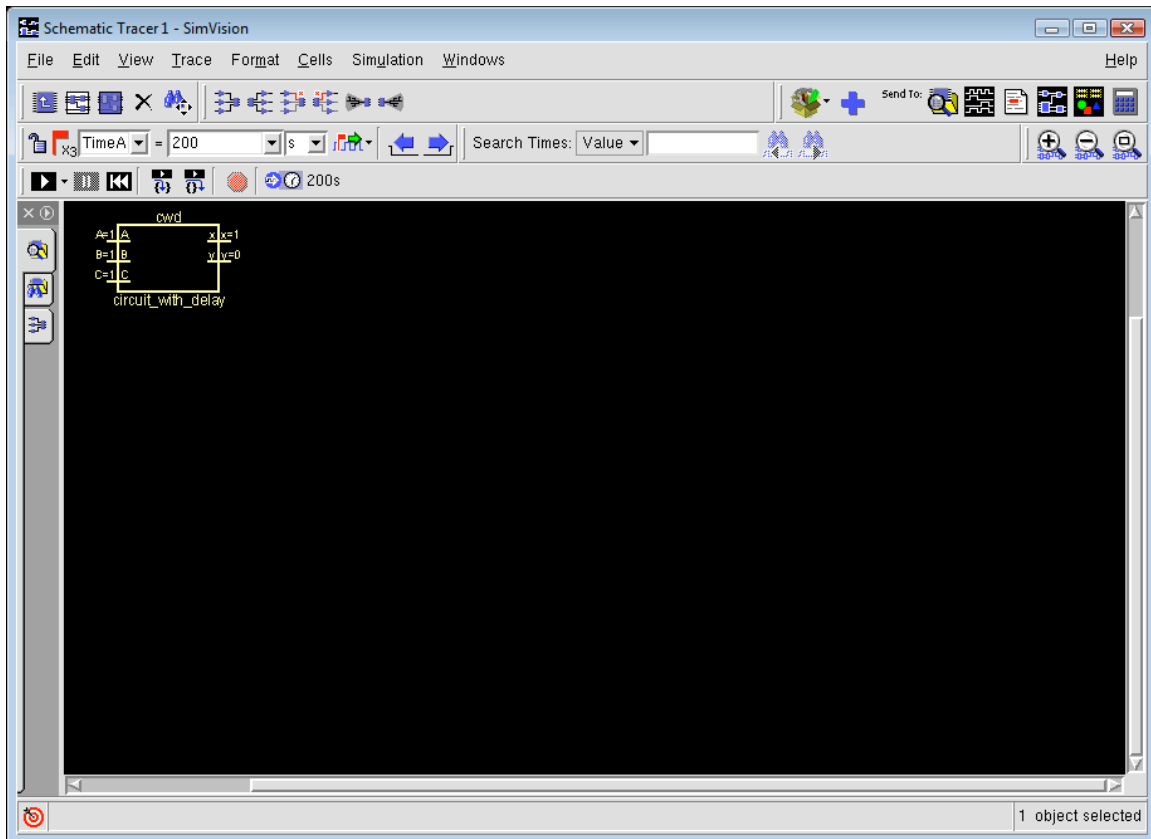


Now, we can finish the simulation. To do so, click on the small triangular “run” button just above the red squares in the above figure. This will tell the simulator to resume and waveforms will be recorded on the waveform window. It should look like the following:

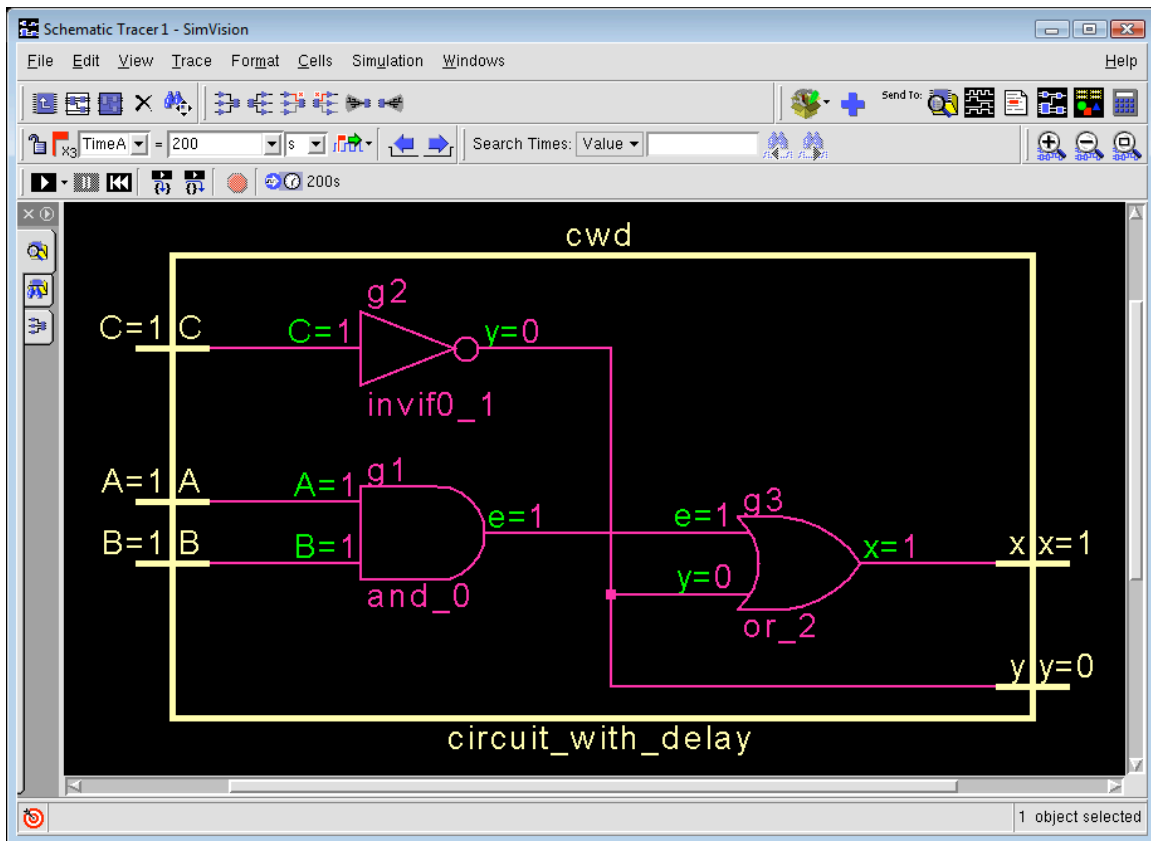


Notice that **x** and **y** are red lines during the first of the simulation. This is because they are set to the value **x** because the simulator cannot determine a logic level until the gate delays have been accounted for.

Another interesting feature is the “Schematic Tracer” feature. To invoke this, click on the icon on the waveform viewer window that has the small gates (two icons to the right of the waveform icon). You should see a window that looks like:



This window shows a schematic symbol representing the cwd instantiation. In order to view the internal content of the cwd instantiation, click on the second icon from the left at the top of the menu that looks like small interconnected blocks. You should see a window that looks like the following:



To exit the verilog tool, use the pulldown menu on any of these windows labeled “File” and choose “Exit SimVision”.