ECE 327 Synthesis Lab Tutorial

October, 2002

1 What's this about?

This is basically a beginners guide to using digital design design tools through the steps of design entry, synthesis, and hardware implementation. This will require you to become familiar with and use several different software and hardware components. Working with these tools will provide you with hands on experience with the design concepts that you learn in ECE 327.

2 Useful Terminology

- *VHDL-* A hardware description language used to represent digital designs (both structural and behavioral) in a programming language.
- *Simulation-* The process of checking a design with software tools so that you can verify it's correctness and see its behavior before actually implementing the design.
- *Synthesis-* The process of translating a VHDL representation of a design into a form that can actually be implemented with physical logic devices.
- *Download-* This is the process of transferring a physical hardware design from the PC to the FPGA device and loading it.

3 Lab Hardware

3.1 Workstations

Most of this work can now be performed on ordinary PC workstations. For this lab it will be helpful to be familiar with both Windows and Linux operating system environments (more on this later). Those of you who have taken ECE L272 with Linux based computers will find these machines to be very similar for the most part.

3.2 FPGA kit

This is the fun part. Each lab machine is equipped with a programmable FPGA board and prototyping kit. What is an FPGA? Look in your textbook or lecture notes for a technical description, but for the purposes of this tutorial we will just say that it is a reprogrammable digital device that let's you implement the digital designs you create in class. It allows you to quickly (and inexpensively) prototype a design. It can be physically tested without having to manufacture a traditional integrated circuit.

Each FPGA board is attached to a prototyping board that provides several physical interfaces to the device for experimentation. If you look at the circuit board you will see that it includes the following: 3 buttons, 8 DIP switches, 3 LED displays, 2 VGA (video) connectors, 1 ps/2 (keyboard) connector, 1 parallel port, 2 audio connectors, and many connector pins. We have the potential to create designs to control all of these things in hardware. For example, we could create an audio processor, graphic equalizer, or video controller, and actually implement them to see the results.

The FPGA board and PC are connected by a parallel port cable so that we can download new designs into the FPGA as needed.

4 Software Tools

4.1 Operating Systems

As a warning, this section may seem a little strange even to those of you who are familiar with Linux and Microsoft Windows. It will hopefully make more sense after going through the tutorial steps, so you may wish to read back over this afterwards. Basically, in order to use the tools and hardware that we have at our disposal, we must work in both Linux and Microsoft Windows NT *simultaneously*. How can we do this? We are using a software tool called VMWare to make this happen. You log into these machines just like any other Linux or Unix machine, but once you are logged in you will see that an application window has opened that appears to be starting Windows NT. Once it has finished, you will be able to work in both the Windows NT area and the Linux area at the same time. Both will be visible, both have their own desktop and software, and both have network connections. This lets us take advantage of tools that are available for both. You can think of it like this: Linux is running all the time just like it normally would, but Windows NT is being tricked into running as an application on your desktop. If this means absolutely nothing to you, then don't worry- just follow the tutorial steps and you will be fine.

4.2 Synopsys (Simulation)

This is a large software package which lets you create VHDL designs and simulate them. It is available on the Riggs 10 workstations. You can use it to anaylize files, and view waveforms of the design.

4.3 Xilinx Foundation Series (Synthesis)

This is another related software package made by Xilinx. It also allows you to enter designs and simulate them, although we prefer to use Synopsys for simulation at this time. The reason we use the Xilinx software is it supports the specific model of FPGA hardware that we wish to use. It's main role will be to perform the device specific synthesis and optimizations.

4.4 xsload and xsport (Download)

These are two software tools available under Linux that allow us to download our designs onto the actual FPGA devices. xsload copies the design and loads it. xsport allows us to send signals down the parallel port cable for testing purposes.

5 Tutorial

5.1 Logging in and Moving Around

1. The TA will supply you with a new account and password for the 327 machines. These accounts are NOT your university or engineering account! These machines are independent and have different files, passwords, etc. Your password for these machines must contain at least one letter, one number, six characters, and cannot be based on a dictionary word. Once you receive your new account, login in with your supplied username and password.

2. You should be presented with a screen that contains two windows. On the right side there is a simple black terminal window. On the left side you will see a VMWare window that may be printing messages or resizing at the moment.

3. Wait for the left side window (VMWare) to settle down and present a "Press Ctrl Alt Delete to Log On" message.

4. There are a few pointers on how to move around the desktop that you should read here even if you are familiar with Linux. The main issue is that once you click on (or in) VMWare, you sometimes cannot move your mouse out of the window unless you press CTRL+ALT+ESC. This is Important! You need to be able to switch back and forth between VMWare and Linux sometimes. Try it now to see. If you click in the middle of the VMWare window, you will be able to move your mouse around in the window, but it won't let you pull it out to click on the terminal on the right side. If this happens and you need to release the mouse cursor, just let go of the mouse and then press CTRL+ALT+ESC. You will see the cursor change color or shape slightly and you will once again be able to move it over to your other Linux applications.

5. At this time you need to log into Windows NT under VMWare. To do this, click somewhere in the VMWare window and press CTRL+ALT+DELETE like the prompt says. Log into NT with the username "student" and no password. (This should already be there so you can just click "Ok")

6. Note that if you close your VMWare session for some reason, you can always restart it by clicking on the VMWare icon on your Linux desktop.

5.2 Working in Windows NT

1. Now that you are logged into Windows NT, you will see a generic desktop setup presented there. We are going to be working within this environment for a while, so for now all of the following steps will assume that you are working within the VMWare window.

2. You may have already read the warning notice in the upper right hand corner of the screen. Take it seriously! DO NOT save any files on the C:\ drive or D:\ drive. Anything you put there will be ERASED when you logout. NOTE: This is not a suitable excuse for not turning in a project! If you lost something by putting it on the C: drive then you will get an appropriate grade for not following directions. In a related matter, don't bother customizing your desktop, changing backgrounds, etc. All of these settings are reset when you log out also.

3. Now let's find out how to get to the files you can change. The first command that you type in the MSDOS prompt every time that you log into NT should be:

   map_drives [username]

   where [username] is the name(NO brackets) you used to log into Linux a few minutes ago. It will ask you for your password twice. Enter your ECE 327 lab password both times. This will give you access to both an f: drive (which is your personal storage space) and a g: drive (which is a common storage area for assignments, etc.). If it doesn't work, type it again. It does not hurt to run map_drives more than once if you typed the wrong password. You can check to see if this worked by double clicking on the "my computer" icon in the upper left area of the desktop. You should see several drives: a, b, c, d, e, homes(f), and common327(g). If homes or common327 is not there try to run map_drives again.

4. You should store any work that you do on the f: drive (also refered to as "homes"). This is just a pointer to your Linux account storage area. Anything that you store in Windows NT in the f: drive will also be visible in your home directory under Linux.

5. This document references several tools that need to be run from the command line in NT. Whenever you need to open another command line window just double click on the "Command Line" icon on the desktop.

6. The first thing you need to do is copy the tutorial files into your user space so that you can play with them(in later labs we will will use a different method. I will explain that in a minute). At the NT command line, type the following commands:

   ```
   mkdir f:\tutorial
   xcopy g:\tutorial f:\tutorial /s /e /h
   f:
   cd f:\tutorial
   dir
   ```

7. After typing the last command you should see 4 files: str_led.vhd, str_led.ucf, testbench.vhd, and sig.all. This an extremely simple VHDL design that maps one of the buttons on the prototype board to one of the LED's on the board.

5.3 Retreiving Files in Later Labs

1. In later labs you will need to pull your relevant *.vhd file off of the engineering computers if you do not have the information with you on disk. You can skip this section for now, but keep the information handy for next time.

2. There are two ways to retrieve the information. Both must be done in the LINUX terminal. NT WILL NOT RECOGNIZE THE PROGRAMS. First, in the LINUX terminal create a directory for your information by typing:

   mkdir [directory name]
   cd [directory name]

   where [directory name] is the project directory name you create(NO brackets).

3. (Option 1)Using pftp. To start pftp enter:

   pftp ash.ces.clemson.edu

   At this point you will be asked for a password. This is your ENGINEERING PASSWORD, not your ECE 327 one. Once logged in you use the commands "get [filename]" to retrieve files, and "put [filename]" to place your files back on the engineering computers. You can "cd" into your project directory and "get" your files. Once you have your information "exit".

4. (Option 2)Using scp. Scp is a copy command much like cp.

   scp [dir/filename of source] [dir/filename of destination]

   where [dir/filename] is the directory and filename of the information. The destination should be just a "." if you are in your project directory. The source should look like:

   ash.ces.clemson.edu:/tutorial/tutorial.vhd

   At this point you will be asked for a password. This is your ENGINEERING PASSWORD, not your ECE 327 one.

5. When you retrieve your files, you should not need your testbench. Just retrieve your source code .vhd file.

5.4 Running Xilinx Foundation

1. Launch the Xilinx Project Manager by double clicking on the project manager icon on the desktop.

2. Choose "Create a New Project".

3. Enter the name as "tutorial", the directory as "f:\tutorial", and select the "HDL" option. Then click "Ok".

4. You now have a fresh project area to work with in the Foundation tools. The project manager has 3 main window sections. The bottom one logs messages and errors from the current operation. The left side shows files active within your project. The right side provides control over the design process.

5. At this point we want to bring in the VHDL files that we simulated earlier with Synopsys. Choose "Add Source Files" from the "Project" menu. This will present you with a file browser dialog. Move back one directory to find str_led.vhd (f:\tutorial\str_led.vhd) and select it. You should now see str_led.vhd present in the left side window of your project manager.

6. Now we can synthesize the design. Click on the Synthesis button on the right panel. In the dialog box, choose XC4000XL as the family, 4010XLPC84 as the device, and xl-3 as the speed. Be very careful with this. If the wrong item is chosen the implementation will not work. Click "Run".

7. Xilinx breaks synthesis up into two steps. The next one is called implementation. Click on "Implementation" on the right panel now to continue.

8. Under "control files" choose "set". Then under "Constraints File" choose "Custom". Next browse and go up one directory. Select "str_led.ucf" as your constraints file. Choose "Ok" to get back to the implementation menu.

9. Click "Run" to run the implementation. You should see a program pop up that shows the steps that the software is going through to implement the design for our particular piece of hardware.

10. If everything has completed successfully, then your design is complete. Go on to the next step to download it to the device.

5.5 Downloading the design

1. In order to download the design you must switch back to Linux. Either drag your mouse out of the VMWare window (remember the CTRL+ALT+ESC keys if needed) and click on a terminal in Linux, or close VMWare entirely by choosing the "Power Off" button at the top. Normally you would leave VMWare running in case you needed to back up to a previous step to fix anything, but it is not necessary for the tutorial.

2. In a Linux terminal type the following commands:

```
cd ~/tutorial/tutorial
xsload tutorial.bit
```

3. You will see a progress indicator go across the prompt to indicate how long it will take to download the design. Wait until it is completed.

4. To prove that the design was downloaded correctly, press the button on the prototype board labeled "SPARE" (button is the leftmost of three). One of your seven segment LED panels should light up when this button is pressed.

5. Congradulations! You have completed all of the steps involved in pushing a design through the synthesis tools. Show the TA that your prototype board works before you leave.