

Useful Xilinx Tips

General

- **SAVE CONSTANTLY.** Unfortunately Xilinx is a little temperamental, and has been known to crash frequently. The best way to avoid losing large amounts of work is to save frequently
- Do not open project files from your ENIAC drives (running Xilinx over the servers is really slow). Copy and paste your project from the ENIAC drive to the local hard drive (in C:\user). Work on this local copy, and when finished, then copy it back to your ENIAC drive (or a flash drive). **Please remove all local data after finishing because anyone can access it. The local drives are also reformatted from time to time, and we are not responsible for any work lost because you forgot to back it up.**
- Sometimes Xilinx can be annoying about adding sources from previous projects into new projects. The easiest way to do this is usually to copy the files over into the newly created directories, and then adding these files, rather than adding sources from older directories. This is also a good way of rebuilding a project in case Xilinx starts acting up.
- **Never start any source names with a number**, especially VHDL files. Xilinx does not allow this for some reason, and sometimes gives undecipherable errors because of this.
- When Xilinx acts up, sometimes closing the program (after saving everything) and then opening it again solves the problem. This is one of those solutions which has no good explanation why, but it still works.
- In Ketterer lab, when plugging in the FPGA to the computer, be sure to use the left USB drive on the front of the computer. The ones on the right require administrator privileges to install the hardware.

Schematic

- When tapping buses, remember to name your wires appropriately to tap the right bits. If this is done incorrectly, Xilinx will give you errors.
- To increase the size of the schematic (if you are running out of space), double click on the background of the schematic, and a drop down menu will give you larger size options.
- For VHDL and other code sources, you must create a schematic symbol before you can add the symbol into any schematic. Sometimes it takes a little while to update, so closing the program and reopening can often solve the problem.
- Make sure your connections are made properly. The way to tell usually is by checking if you can see any red boxes in your schematic. These indicate a loose wire.

- Merging nets can sometimes help reduce cluttering (by reducing the number of visual interconnections shown), however this can make your schematic a little complicated. Make sure you understand how it works before using this feature.

VHDL

- When assigning values, multiple bit values are designated in double quotes (e.g. "0101"), but single bit values are designated in single quotes (e.g. '1').
- VHDL has a lot of functionality, and we only scratch the surface in this semester. We recommend sticking to the basics for this course, but more details can be found in the [VHDL Primer](#).

FPGA Debugging

- The LEDs function on active high signals (lights up when sent a 1), but the seven segment displays function on active low signals (lights up when sent a 0). If your alphanumerical displays are inverted, this may explain why.
- Avoid using the buttons in the middle of the board (the UP, DOWN, LEFT, RIGHT ones). These buttons are not debounced, and thus may not function like the buttons on the daughter board.
- Remember that you may need one-pulse circuits to ensure that a single button press over multiple clock cycles is not understood as multiple button presses.

Last Modified: December 5, 2009

If you have any good suggestions in this list, feel free to contact your TA so that we can add to the list. This list is by no means exhaustive, and if none of these suggestions work, be sure ask your TA for help.