ECEN3002 - Lab 9

Fall, 2020

Due Thursday November 5, 11:59pm – 50 Points

After completing this assignment, you will submit your programming file (.sof) and Quartus archive file (.qar).

Due to the length of this lab, you have **two weeks** to complete this. We will have another lab assignment next Thursday, but that assignment will be short and sweet.

The tutorial used in this assignment is Making Platform Designer Components, and is included in Canvas along with this document.

This assignment involves creation of a custom IP block used in a Platform Designer created component. There are a lot of steps involved in this process, but the code is provided so you can concentrate on the custom component portion of the design. This tutorial is very thorough, and it is worth the time to read the tutorial carefully, as it will answer and clarify questions you likely have about how Platform Design works. Feel free to discuss with me, or email me with any questions you have.

Special Instructions:

Take a few minutes and look over the Verilog code as you create and add the files, in order to understand what is being constructed.

There are some things you have to modify, so read this part carefully and refer back to it as needed. There are both Verilog and VHDL versions of the files, you should ignore the VHDL stuff.

1) When copying the code from the document, the single quotes used in declaring Verilog data widths are (sometimes) the incorrect character. Make sure that anytime you some a 4’d0 (or something like it), you manually replace the single quote character with the correct character from your keyboard. Otherwise, the copy and paste from the tutorial document worked well.

2) This tutorial uses a NIOS processor as the master in the Platform Designer project. **Do not** use the NIOS processor, but instead use the JTAG master as you have done before. This section starts are the top of page 11, so ignore the instructions to add the NIOS processor and the On-Chip Memory block. Add the JTAG master instead, then resume the tutorial at the top of page 14.

3) At the top of page 17, the tutorial instructs you to drag and drop signals. I have found that you need to click on the signal first, release your mouse button, then click on the signal again to drag it into position. The GUI is not very intuitive in this regard.

4) When creating the instantiation template (bottom of page 25), I did not see the reset signal as a top level port. Perhaps this was an error on my part, but if you don’t see the reset signal, continue on, but you will have to remove the missing reset port in your top level file before final compilation.

5) Part 8 should be ignored, as it is specific to the NIOS processor. What you will need to do is write a few TCL commands from within System Console in order to verify correct operation of the design. Since the register is connected to HEX displays, you will see values written to those displays. Once you have verified correct operation, put the commands into a Tcl script and include the script with the rest of your souce files.