EECE 321 Assignment #6 – Pulse Width Modulation Application

Work either individually or in groups of **two or three** to complete these exercises. Students are required to submit suitable format electronic documents through Moodle for their solutions (one per group); **paper submissions will not be graded**.

Be certain to provide references for your sources and the names of your collaborators. **Work that is claimed to be an individual effort, but is decidedly the work of others, will receive a grade of zero (0)**.

In the previous lab, you developed the core hardware to implement a PWM waveform. Now you will create a custom Qsys component using that design and add it to an existing computer and program it in software.

1. Add an Avalon wrapper file (Verilog HDL) that implements the Avalon interface and instantiates the core pwm circuit.
2. Copy the DE2-115 Media Computer from its location in C:\altera\16.0\University\_Program\Computer\_Systems\DE2-115\DE2-115\_Media\_Computer to a local directory, e.g. Documents\DE2-115\_PWM\_Computer. Remember: No spaces in path!
3. Open up Qsys and load the existing design for the overall computer.
4. Create a new Qsys component for your PWM peripheral. Name it appropriately and fill out the information.
5. Add the newly designed peripheral to the DE2-115 Media Computer and wire it to a pin on the JP5 40 pin expansion header using Qsys and Quartus. See Figure 7 in the documentation. Build your project and take note of where your *.sof* and *.sopc\_info* files are stored. You may also want to wire it to an LED such as LEDG8 for easy debugging.
6. Using NIOS Eclipse Build Tools create a new application and board support package.
7. Write an application that reads from the slider switches to output a PWM signal.
8. Using a multimeter and oscilloscope create a chart of output voltage versus switch position. Capture time domain plots of the PWM output for two different switch settings.
9. Write a report describing your results and what you learned. Include…
   1. Block diagram of your PWM
   2. Why did it take longer to configure the flash using the Active Serial mode?
   3. Describe the files required to get this to run and where on the board are they stored.
   4. Upload your source code to github.
   5. Include the charts and plots from step 6.
   6. Include a discussion of what you learned.
10. **For test repair**: Add 3 more PWM to your computer design.
11. **For test repair:** Create an external circuit to drive some motors supplied by me. You will need to see me for details.
12. **For test repair:** Using Quartus create a *.pof* file to program the configuration flash memory with the new computer so that the FPGA is configured accordingly when powered on.
    1. Open the Convert dialog by choosing *File->Convert Programming Files…*
    2. Select the appropriate EEPROM type (Consult the DE2\_115 board documentation to determine the type.)
    3. Choose a meaningful output file name, e.g. *DE2\_115\_Media\_Computer\_PWM.pof* and place the file in the same directory as your *.sof* file.
    4. Change *mode* to *Active Serial*.
    5. Click *SOF Data…Page\_0* and choose *Add File*. Use the file browser to select the *.sof* file you created in step 4.
    6. Click *Generate*.
    7. Run the *Programmer* and program the configuration flash using Active Serial Programming. Add the *.pof* file and select the *Program/Configure* check box. Change the *Run/Pgm* switch to *Pgm* on the lower left corner of the DE2\_115 board. Click Start (Click Start again if the transfer fails.) It will take much longer to configure than normal (**WHY?**) Be sure to return the switch to *Run*. Turn the board off and then back on. It should now be configured as your new computer.
13. **For test repair:** Program the flash with the program and the files above using the document *Nios II Flash Programmer User Guide* Chapters 1 & 2.