Technical Objective:

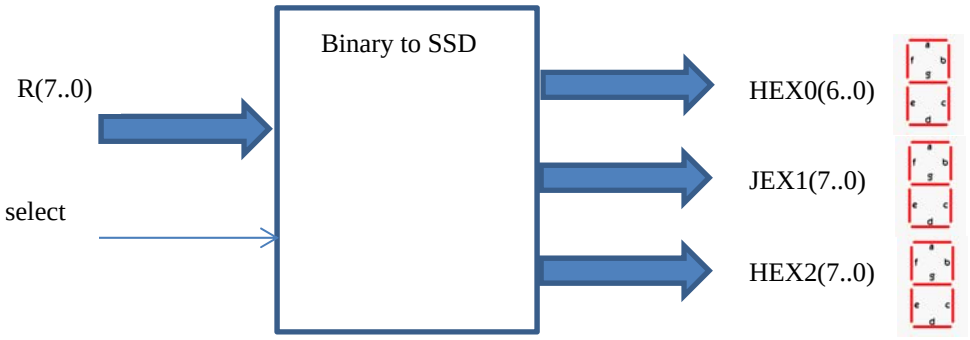
A common display device is the seven-segment display. It is made up of seven separate LEDs in a single package. By turning the individual LEDs on and off, letters and numbers can be represented. As a numeric display is much easier to read than a binary one, the addition of three seven-segment displays on the output put of the ALU in lab 5 will make it much more useful. This lab adds an output display to the ALU of lab 5. New concepts introduced in this lab include constants and seven-segment displays

Pre-Laboratory: (20%)

1. Review the lecture notes from 9/27/13.
2. Create a VHDL module (entity and architecture) that takes in an 8-bit vector and a selection bit and outputs the constants to drive 3 seven-segment displays as described below:
   a. If the selection input is 1, the displays should show the two hexadecimal digits that are equivalent to the 8-bit input. The third display should be blank.
   Ex: 1101 0001 = D1

   b. If the selection switch is 0, the displays should show the signed decimal value of the 8 bit input
   Ex: 1111 0001 = -15

3. The ports of the module are as illustrated below:



4. Submit your VHDL code to the dropbox prior to your lab section

Procedure: (60%)

1. Download the Display_tb.VHD file from MyCourses. Open the file and edit such that the component in the testbench matches exactly with your entity. If your port names are different than the ones on the component in the testbench, you will have to edit the port map as well.
2. Using the Modelsim tutorial from MyCourses, simulate binary_to_SSD module using the testbench provided.
3. Verify the operation of your module by inspecting the output waveform. Remember that the outputs you will be looking at are the constants needed to illuminate the correct segments on the display. They will not make sense in decimal radix.
4. Print and annotate comments onto the Modelsim waveform printouts.
5. Obtain a signoff when you are sure your simulation is correct.
6. Integrate your code into the ALU design from Lab 5. You will have to add the HEX outputs.
    a. The logic operations should result in a hexadecimal display (choice 1 from prelab).
    b. The arithmetic operations should result in a signed decimal display.
7. In Quartus, assign pins for the DE2 board.
8. Recompile the project with the pin assignments. Program the board.
9. Verify that your board is working and get a signoff.

Documentation:

This will be a formal lab report that covers labs 5 and 6. Your report must include the following:

- Abstract – objective, how the objective was met, summary of result
- Detailed discussion. This includes all pertinent design work
    - Derivation of constants
    - Vhdl code
    - Modelsim waveforms
    - Other data collected
- Discussion and conclusion section. Are there practical applications for what you did? How can this lab be expanded?

Name Clif Chapman

ALU (VHDL)

Signoff

1. **Simulation results** 10/9/13
2. **Working Board** 10/9/13