ECE 2534 FALL 2012 LAB #4 FINAL VALIDATION DUE: 10 PM ON 12/10/12

THIS LAB IS TO BE DONE INDIVIDUALLY. CODE UPLOAD DUE: 11 PM ON 12/10/12

Read the entire lab before beginning.

"The Cerebot Oscilloscope"

The goal of this lab is to create an oscilloscope by combining the on-board analog to digital converter (ADC), the timer peripheral, and the OLED output device. The lab write-up below will outline how the system is to behave, but will not specify how you are to implement that behavior. This is meant to mimic the way specifications are normally given to engineers. You should use the insights you have obtained from the previous labs to structure your code in a readable and maintainable way. The structure of your code will be a major portion of the grade.

Oscilloscope Specification

1. Upon first running the code, the message "0.2 msec/div" will be displayed on the top line of the OLED. Rotating the rotary knob will allow the user to change the time per division among the following values: 0.2, 0.5, 1, 2 and 5 msec per division. The message should update accordingly. Rotating the knob clockwise should increase the value and counterclockwise should decrease the value. The values should be saturated so that it is not possible to go above the maximum or below the minimum value.
2. Pressing button 1 should lock in the time per division value to the value currently displayed and then display the message "1.5V trigger". Rotating the knob should change the trigger value in increments of 0.1 V with a minimum value of 0.2V and a maximum value of 2.8V.
3. Pressing button 1 again should lock in the trigger value currently displayed and begin the oscilloscope display on the OLED. The observed voltage should be displayed as a function of time starting with the first sample to cross the trigger level with a positive slope. The lowest voltage to be displayed is 0V and the highest is 3.0V, with a scale of 0.1V/pixel. The grid on which the waveform is displayed should be laid out as follows:
    a. Solid horizontal lines at 0V and 3.0V
    b. Dashed horizontal line at 1.5V (2 dots on, 2 dots off, …)
    c. Solid vertical lines at the 0^{th} and 127^{th} x-coordinate, running between the lowest and highest voltages.
    d. Dashed vertical lines at the 40^{th}, 80^{th}, and 120^{th} x-coordinates. This means that there need to be 40 time samples per division.

    Note that there is no need to write the samples to the display as they are received. You may collect all 128 samples, write a full screen of data at

one time, and then go back to collecting data. The refresh times are fast enough that your eye will not notice the delay.

4. Pressing button 1 again should bring you to step 1 above. Steps 1 to 3 repeat with continued button presses. The only difference is that the initial messages in steps 1 and 2 begin with the previously saved values of the time per division and the trigger value.

Testing

To test your code, use the function generator on your ANDY board. Set the generator to sine mode and adjust the amplitude and offset so that the sine wave voltage lies in the interval of 0V to 3V. This can be done using your USB oscilloscope. Connect the ground of the ANDY board to the ground of the CEREBOT board. Then **connect the generator output to your analog input via a 10K ohm resistor** to protect the input from excessive current due to any out of range voltages.

Notes

If you are using ADC interrupts, you must read all ADC buffer locations written since the last interrupt before you attempt to clear the ADC flag. This is not clearly stated in the documentation for our hardware, but appears to be the case.

Handing in your Lab 4 validation sheet

You should return the validation sheet to your instructor no later than the start of class on Tuesday (Wednesday) December 11th (12th).

Submitting your lab to Scholar

Create a ZIP archive of all source code used in Lab 4 (i.e. only .c and .h files), and submit it to Scholar before 11:00 pm on Monday, December 10th. The ZIP archive should be named

`<Last name>_<First name>_Lab_4.zip`

In Windows, this is accomplished by selecting your files, right clicking on one of them, and choosing `Send to` and then `Compressed (zipped) folder.`

Be sure to comment your code appropriately.

Honor Code

For laboratory assignments, students are not allowed to share or discuss specific algorithms or other detailed solutions. In particular, source code may not be

shared: obtaining code from *any* source other than your instructor, the course notes, or the PIC32 peripheral libraries is an honor code violation.