BIOE3090

Controls and Circuits Homework

30 Pts

Due: 2/15/2018 in Class

**To ensure full points, please use accepted format for completion of all problems (Given, Find, Diagram, Assumptions, Analysis, Interpretation). Please use units throughout all calculations and reference any equations used as well as any constants. Please ensure diagrams are labeled appropriately.**

Part I: C++ Primer (20 points)

The software provided on the UCDBioe Github account (see below for downloading instructions) contains all of the Arduino firmware required to control the field incubator and provides data communications between the Arduino and an optional terminal running Matlab. However, the firmware is missing failsafe modes for overheating the box and for over discharging the battery. Since it would be difficult to run the native Arduino firmware without the associated hardware, you have been provided an incomplete C++ program that you can use to develop some of the functions that would be useful in implementing these failsafes.

The code provided should run, it just will not do what it is supposed to until you fix it. This should allow you to debug the software in a functional state. I would suggest compiling and test running the code early and often rather than trying to complete all of the section then trying to debug the code.

1. (5-pts.)  
   Complete the code needed to print the average values of a given array. The code sections are outlined in the bioe3090\_HW.cpp file using the tags:

(STUDENT COMPLETE THIS SECTION >> -- HWS2----)

Your code goes here!!!!!!!!!!!!!!

(STUDENT COMPLETE THIS SECTION << -- HWS2----)

1. (5-pts)

Complete the code needed to print the values contained in the arrays in a nicely formatted way. Your output should look similar to Figure 1, where the values are aligned roughly in columns. The code sections are outlined in the bioe3090\_HW.cpp file using the tags:

(STUDENT COMPLETE THIS SECTION >> -- HWS2----)

Your code goes here!!!!!!!!!!!!!!

(STUDENT COMPLETE THIS SECTION << -- HWS2----)

1. (5-pts)

Complete the code needed to shift the buffer array and replace the last value with new data. This is the circular buffer used to average out noise in the signals. The code sections are outlined in the bioe3090\_HW.cpp file using the tags:

(STUDENT COMPLETE THIS SECTION >> -- HWS3----)

Your code goes here!!!!!!!!!!!!!!

(STUDENT COMPLETE THIS SECTION << -- HWS3----)

1. (5-pts)

Complete the code needed to check if a shutdown condition has been met. The code sections are outlined in the bioe3090\_HW.cpp file using the tags:

(STUDENT COMPLETE THIS SECTION >> -- HWS4----)

Your code goes here!!!!!!!!!!!!!!

(STUDENT COMPLETE THIS SECTION << -- HWS4----)

Your output should look similar to Figure 1.

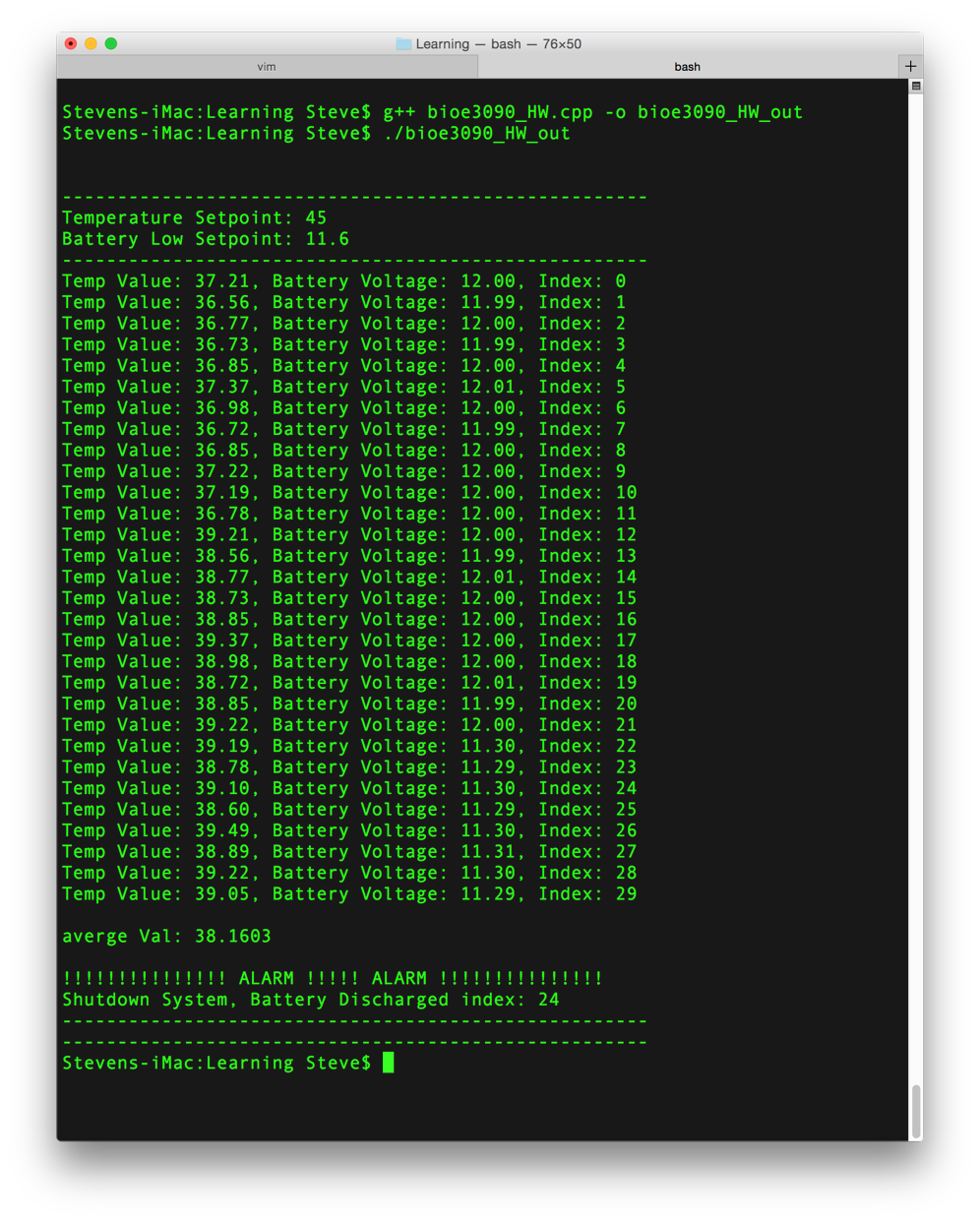


Figure : Output from completed C++ program.

You may get a warning when you compile the code before you have fixed all of the sections: **warning: control reaches end of non-void function [-Wreturn-type].** This warning simply means that you have a function defined to return some value and no value was returned. It will fix itself when you have valid return statements.

Part 2: Soldering and Components (10 points)

Watch the “Soldering Video” and the “Multimeter Tutorial Video” posted to Canvas-Modules-Technical Content and Lab Module. Answer the following questions.

1. Why do you need to keep your solder tip clean?
2. Why do we tin the soldering iron?
3. What shape indicates a good solder joint on a printer circuit board?
4. What is continuity?
5. Can you use a multimeter to measure a resistor that is wired into a circuit?
6. (5-pts) What happens if you do not remember to splice in the multimeter when checking a circuit and instead use the probes as you would use to check a voltage reading? For example, you went to check the amps of a 120V wall outlet and plugged in the probes directly to the outlet as was demonstrated in the video when AC voltage was being measured? Describe how this could be dangerous and potentially damage equipment.