CSCI130: Mobile Robotic Devices Embedded Systems
Lab1: For Mac users

Lab Description:

The purpose of this lab is to learn how to setup your Mac computer to be able to compile code for the Atmega328p microcontroller and load the binary/machine code onto the device. This lab is designed to be a tutorial instead of testing the participant's knowledge. This software is required for all subsequent assignments.

Software requirements:

In order to program the Atmega328p on the Romeo board, Mac users need to install the following software.

> Note: For most MACs, the /usr/local/bin is already on the path so before following the complete instructions, you should see if the /usr/local/bin is already on the path, and skip that section if it is. You can check the path with the following "echo $PATH". If it is, proceed directly to the Option 1 AVRMacPack and click on the download link.
>
> Note: The version of the AVR software for the Mac is called CrossPack for AVR Development: CrossPack-AVR-20131216.dmg. It will be the latest version at the top of the list.

1. http://www.ladyada.net/learn/avr/setup-mac.html

Tutorial:

Step 1. Determine the device name of your Romeo board. Before connecting the board execute "ls /dev/tty.usb*". You most likely will not see any output from this command. Repeat after connecting the board to the USB port. The new device name will appear in the output listing. Remember this and replace the XXX below with this name.

Step 2. Download helloEmbeddedWorld.zip file and extract it into a folder

Step 3. Run a Mac terminal and cd to the location of the extracted files.

Step 4. Modify the "Makefile" (eg type nano Makefile or use editor of your choice)

- change "TARGET = main" to "TARGET = helloEmbeddedWorld"
- change "AVRDUDE_PROGRAMMER = stk200" to "AVRDUDE_PROGRAMMER = arduino"
- change "AVRDUDE_PORT = lpt1" to "AVRDUDE_PORT = XXX" where XXX is the device name from step 1. For example, it should look something like AVRDUDE_PORT = /dev/tty.usbmodem411.

Step 5. Use the Mac Command Prompt to run the make file (ie type "make"). If there are no errors, several files should be generated. The .hex file is the machine code that will be uploaded. Capture the terminal screen for submission below (eg Command-shift-4 captures part of the screen and saves it to your desktop).

Step 6. To upload hex file to device type "make program"

Step 7. The helloEmbeddedWorld program causes the LED labeled “L” on the Romeo board to turn on and off at 1Hz. The factor default program is the exact same program so you will not see any changes from the first time the board is plugged in. Modify the program such that it turns on and off at 5Hz. Recompile and upload your new program and see the change.

Submission:

Submit a screenshot of a successful compilation of the code like Figure 1 below. There should not be any errors in the listing.

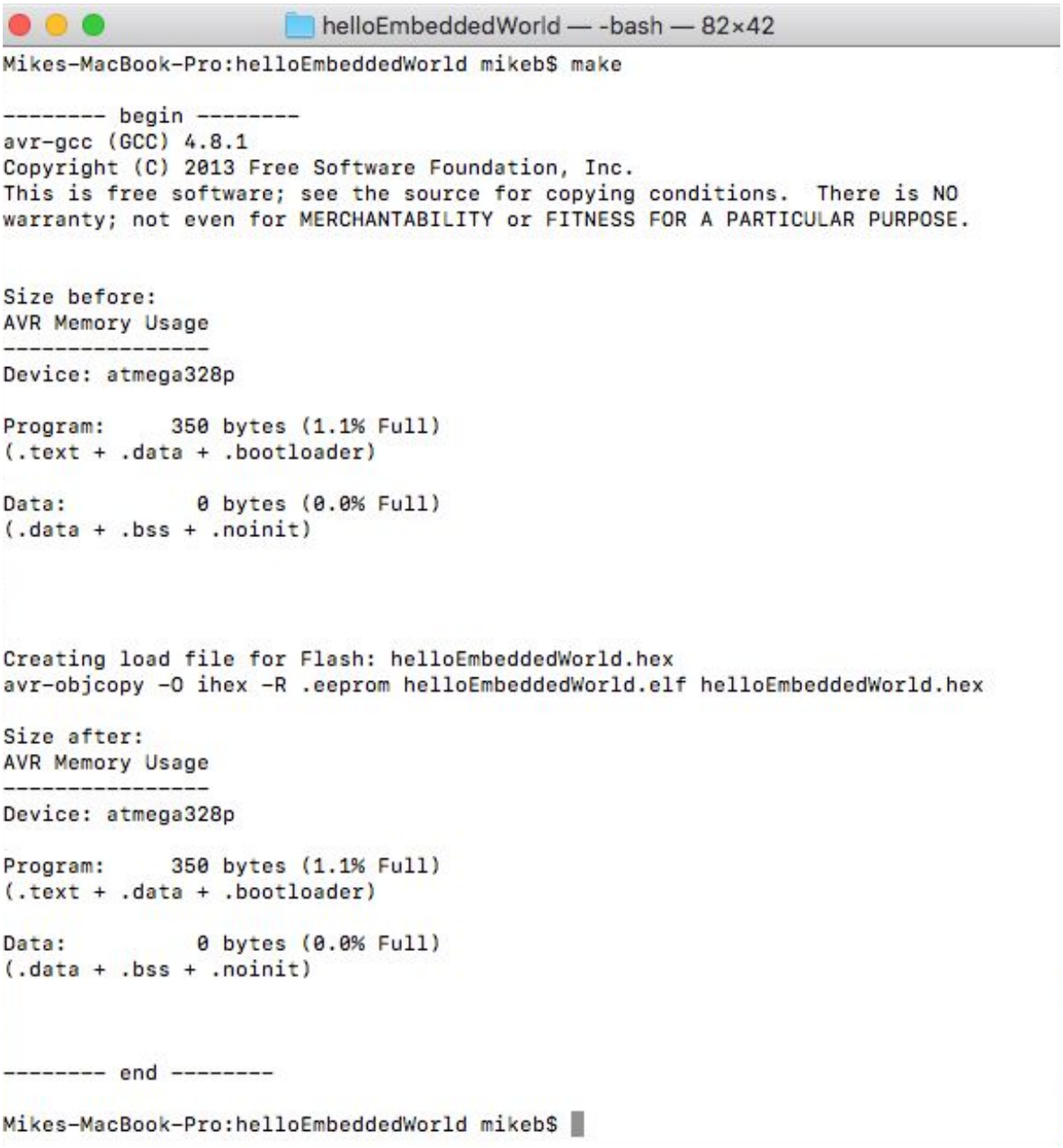

Figure 1