Lab 1 – Installing Tool Chains

Overview

In this lab, you will be double checking that you have the two required tool chains for the course setup properly. A tool chain is a sequence of programs where the output of one program is the input of another and so on. For example, a source file -> compiler -> linker -> executable. For this course we will need two tool chains in place; avr-gcc and Java.

Prelab

There is no prelab this week.

Procedure

Install and Test avr-gcc

To be able to program for embedded systems, the code environment needs to be installed. The Wunderboard system contains a microcontroller from Atmel based on the company's 8bit AVR computer architecture. Many different compilers exist for this family of tools, but to be consistent the manual uses the avr-gcc tool set. The tool set works the same as the gcc tool chain used earlier in the course but instead of producing a file that runs on the computer it produces a 'hex' file for use on the Wunderboard. Trivia: This is an example of a cross compiler.

For installing the software on Windows, two options are available. One option is to download a copy of 'WinAVR' and install it. Another option is to use a tool set called 'Portable WinAVR' that is run from a USB thumb drive or the main hard drive of a computer. The labs will assume the use of Portable WinAVR. **If you would like to install to a Mac or Linux machine, refer to the Wunderboard Usage Guide on the lab webpage.**

1. The first step to prepare the tool chain is to install the software and drivers for the Wunderboard itself. To do this, download the ATMEL F.L.I.P. program from the link on the lab webpage. We recommend FLIP 3.4.2 with the JRE included. Run the installer and install it in the default location. Using a USB cable, connect the Wunderboard programming port to the PC. Press both the Reset Button and the Programming Button at the same time and release the Reset Button first, then the Programming Button.
   Windows will activate the 'Installing New Hardware' wizard. When asked, browse to the F.L.I.P. directory install the drivers for the Wunderboard microcontroller, the AT90USB647.

   NOTE: When using certain versions of Windows (Windows 64bit as an example) the default driver from Atmel may not work. If this is the case, use the alternate driver on the lab webpage. Only use this if Windows requires you to.

2. Download the Portable WinAVR zip file from the lab webpage. Once it is downloaded, it must be unzipped into the 'root' of a drive. A USB flash drive or hard drive is acceptable to use. For example, the files should be in F:\, not C:\Documents and Setting\user\PortableWinAVR. NOTE: The files must be in the root of the drive. The two '.bat' files show in the root and a directory called Portable will exist (eg. Z:\Portable). This lab will assume that you have unzipped the files to your 'Z' drive engineering directory.

3. Once the files have been unzipped, browse to the directory and run the file called 'CommandShell.bat.' A window will appear with a command line in it. This command line allows you to use both Linux and DOS commands for navigating directories.

Compile and test a program

1. Download the sample zip file for lab 1 and unzip it to a location on your hard drive. In the CommandShell window you opened before, navigate to the location of the files. If you type **ls main.c** you should see the main.c file so you know you are in the correct spot.

2. After navigating to the directory, run the **make all** command. The tool chain will run and process the file. To test the code, it needs to be downloaded to the Wunderboard.

3. To download the code to the Wunderboard, connect it to the programming port to the PC. There are two USB connectors, make sure that you use the programming connector.

4. Once connected press both the Reset and Programming Buttons at the same time. Release the Reset Button first, then the Programming Button. If you have not installed the driver for your Wunderboard before this, your computer might request a driver for the board. If this happens, refer to the instructions earlier in this lab.

5. Once the board is connected and readied for programming, run the command **make program** in the command shell. This will cause a program to download your .hex file to your board. Once the download finishes, your board should automatically run your program.

Demonstrate

Now that you have downloaded code to your Wunderboard, show your TA that it was successful. This can be done by showing them the process of downloading and that the display lights up.

Install and Test Java

Refer to the appendix on the course webpage for information about how to install and test Java on Windows, Mac, and Linux.

Demonstrate and Submit code

When your code is submitted, it will be processed both to ensure it compiles and runs correctly and to evaluate its comments. The comments will be examined through the Doxygen output not inside of the code. It is very important to realize this. In previous courses, your code comments may not have been as intensely reviewed.

Study Questions

1. Search the web and write a one sentence definition for each of the following;

    a. Tool Chain
    b. Software Testing
    c. Hardware Testing
    d. Makefile
    e. Environment Variable

2. Based on what you heard in the first week, what are your expectations for this course? What do you expect to learn about? What do you expect to do? How many hours (minus the 3 hours in lecture) do you think you will

need to spend each week to get a good grade in the course? Write a ½ page total answering all of these questions.

Lab Summary

Task	Completed?
Demonstrate avr-gcc and Wunderboard Download	2pts.
Demonstrate simple Java program	4pts.
Study Questions	4pts.