Installing a Hosted and Embedded C Environment

In the first year you are taught how to program in both Hosted and Embedded C. To do this you need an environment to design, compile, test and run your code (or load it onto a chip). This document will guide you through the installation and setup of the recommended environment for Linux, Mac and Windows. After following the instructions you will have all the tools you need to build C programs to run on your computer or on an AVR microcontroller.


Version: November 4, 2012
©2012
Rex Bannerman and Steve Gunn
Electronics and Computer Science
University of Southampton
## Contents

1 Introduction 3  
  1.1 Software Overview ........................................... 3  

2 Windows 6  
  2.1 Hosted Environment Setup ........................................ 6  
  2.2 Embedded Environment Setup ..................................... 7  
  2.3 Adding c232hm support to AVRDUDE [Optional] ............... 8  

3 Mac 10  
  3.1 Host Environment Setup .......................................... 10  
  3.2 Embedded Environment Setup ..................................... 10  
  3.3 Adding c232hm support to AVRDUDE [Optional] ............... 11  

4 Ubuntu 13  
  4.1 Host Environment Setup .......................................... 13  
  4.2 Embedded Environment Setup ..................................... 13  
  4.3 Adding c232hm support to AVRDUDE [Optional] ............... 14
1 Introduction

This guide details the setup of a Hosted and Embedded C programming environment. The environments have been chosen to be cross platform and open source and include:

- Compilers (gcc for the Hosted and and avr-gcc for the Embedded environments)
- A GNU build system, Autotools for Mac and Ubuntu and MinGW for windows
- AVR C standard library and Simulator
- The Eclipse IDE for writing and developing code
- USB drivers to support programming the Il Matto board (c232hm and usbasp)
- AVRDUDE to program AVR microcontrollers
- Putty or Screen for serial UART communication

Though the final environment has been built with the aim of being platform independent, the install varies between platforms. Most of the programs required for the windows install are either provided by WinAVR or have a reasonably simple setup. The Ubuntu and Mac installations make heavy use of package management systems, Homebrew for Mac and the Debian package manager for Ubuntu, along with some more manual installation.

1.1 Software Overview

1.1.1 GNU Compiler Collection (GCC)

GCC is a compiler system produced by the GNU Project supporting various programming languages. GCC is a key component of the GNU tool-chain. As well as being the official compiler of the unfinished GNU operating system, GCC has been adopted as the standard compiler by most other modern Unix-like computer operating systems, including Linux, and the BSD family. Also included in this installation guide is AVR-GCC, a version of GCC built to build AVR code – it supports C and a subset of C++ (e.g. no C++ std or new operator).

1.1.2 binutils

The GNU Binary Utilities, or binutils, comprise a collection of programming tools capable of creating and managing binary programs, object files, libraries, profile data, and assembly source code. The GNU binutils are typically used in conjunction with compilers such as GCC, build tools like make, and the GNU Debugger (gdb).
1.1.3 AVR Libc

AVR Libc is a Free Software project whose goal is to provide a high quality C library for use with Atmel AVR micro-controllers. Together, avr-binutils, avr-gcc, and avr-libc form the heart of the Free Software tool-chain for the Atmel AVR microcontrollers.

1.1.4 Eclipse

Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. Being written in Java it is cross-platform and varies little between platforms. Though the standard version is aimed at Java development, C and C++ can be added via a plugin or there is a version of Eclipse called CDT that comes with the plug-in (but no Java support). In addition there is a powerful plug-in that adds AVR development, with a very good UI for working with the AVR chips (programming, fuses, etc).

1.1.5 AVRDUDE

AVRDUDE (AVR Downloader/UploadEr) runs on Linux, FreeBSD, Windows, and Mac OS X, and supports a variety of in-system programming hardware, including Atmel AVR ISP mkII, Atmel JTAG ICE, older Atmel serial-port based programmers, and various third-party and "do-it-yourself" programmers.

1.1.6 Screen

screen is a full-screen window manager that multiplexes a physical terminal between several processes; typically interactive shells. It can be used to talk to devices over UART connections and comes with all Unix systems.

1.1.7 PuTTY

PuTTY is a free implementation of Telnet and SSH for Windows and Unix platforms, along with an xterm terminal emulator. It can be used to communicate over UART connections on windows.

1.1.8 libusb

Libusb is a C library that gives applications easy access to USB devices on many different operating systems. libusb is an open source project, the code is licensed under the GNU
Lesser General Public License version 2.1 or later. There are two version you will encounter here, the standard libusb1.x.x and the libusb-compat with the latter being a wrapper for the former allowing programs that used libusb 0.1.x to use the newer version. There is a version of 0.1 for windows called libusb-win32.

1.1.9 MinGW

MinGW (or Minimalist GNU for Windows) is a software port of the GNU Compiler Collection (gcc) and GNU Binutils to windows. It also comes with other software such as MSYS (Minimal System) which provides Unix-like shell commands such as make. It is similar to another program, Cygwin, which also includes many similar features such as a gcc compiler but aims to emulate Unix in windows rather than build native programs.

1.1.10 Homebrew

Homebrew is a package management system that is simple and easy to use for the Mac. It uses ruby and is similar to the other main package management systems on OS X, Fink and Macports. It was chosen over these however due to ease of use and installation. It requires that Command Line Tools be installed (aka autotools from apple).

1.1.11 Crosspack

This is a very convenient bundle of all the AVR related software for the Mac such as AVRDUDE and avr-gcc.
2 Windows

2.1 Hosted Environment Setup

2.1.1 MinGW

To start download the latest installer, and when this is complete run it. When presented with the options Use pre-package repositories catalogs or Download latest repository catalogs pick the later, accept the terms and keep the installation directory as default. When selecting components to install, select the C compiler, C++ compiler, MYSYS Basic System and MinGW Developer ToolKit. Click next and then install and wait for MinGW to install and download everything.

After this add the bin folders of MinGw and MSYS to the environment path variable so that you can execute these commands from any directory on your computer:

1. Go to start
2. Right click Computer
3. Select Properties
4. Click Advanced system settings
5. Then Environment Variables
6. In the System variables find the path Variable
7. Click edit and append the install locations of MinGW and MSYS; with a standard install these would be: C:\MinGW\bin; and C:\MinGW\msys\1.0\bin;
8. Click ok and then again to confirm complete

2.1.2 Eclipse

Download Eclipse-CDT (note: it is called Eclipse IDE for C/C++ Developers), extract the archive to a reasonable install location (e.g. C:\Downloads) and run the Eclipse executable within (note: you need the JAVA runtime library installed). It is advised that while running Eclipse you right click its icon in the taskbar and 'pin' it either to the start menu or task-bar itself for easy access to the IDE.
2.2 Embedded Environment Setup

2.2.1 WinAVR

Download the latest WinAVR and run the installer.

2.2.2 FTDI Drivers

Download the FTDI driver\(^1\). There are two options for installation, a setup executable which should install everything automatically or the archive file. To install via the executable just run the exe. To install via the archive file follow the instructions which are summarised below:

1. Plug the cable into the computer, if windows locates the driver automatically the installation is complete
2. In the case that it does not automatically install, open the device manager Start - right click Computer - Manage - Device Manager
3. In other devices locate the c232hm, right click it - Update driver
4. Select Browse Manually and browse to the extracted archive folder and click next
5. Windows should now install the USB serial converter (found under Universal Serial Bus controllers after installation)
6. Repeat steps 3-5 again to install the USB Serial Port (found under Ports [Comm & LPT])

2.2.3 LibUSB-Win32 Driver

Install the open-source drivers for FTDI devices by downloading libusb-win32-devel-filter-1.2.5.0.exe and installing it. Then install the Filter Driver:

1. Plug in your FTDI C232HM cable
2. Go start - LibUSB-win32 - Filter Wizard
3. Next select 'Install a Device Filter'
4. Then select 'USB Serial Converter (vid:0403 pid6014)'
5. Click install (this needs to be done), close after completion
6. Repeat for each cable that you plug into the computer

\(^1\)It does not matter if you download the VCP or the D2XX drivers as for windows these have been integrated and can function at the same time.
2.2.4 USBasp Driver

Install the USBasp driver by downloading the latest windows driver archive (at the time of writing this is usbasp-windriver.2011-05-28.zip) and extract it to C:\Downloads. Then install the USBasp Driver:

1. Insert your USBasp or Il Matto device (with bootloader) and let windows try to find the driver, which it shouldn’t

2. Now as before go to device manager and find USBasp under other devices, update the driver by locating it manually for windows, selecting the 'libusb 1.2.4.0' folder from the archive

3. When installing the driver windows may complain that it cannot verify the publisher of this driver software – ignore this, and install it anyway. It complains since newer version of windows require that driver are 'signed' for security reasons and this being free is only partially so.

2.2.5 Eclipse AVR Plugin

Open Eclipse and go Help - Install New Software...\(^2\) - then click Add... - Enter AVR for Name: and http://avr-eclipse.sourceforge.net/updatesite/ for Location: - click ok - check "AVR Eclipse Plugin" - click next and install.

2.2.6 PuTTY

Download and run the installer. The installer you require is the one listed under A Windows installer for everything except PuTTYtel.

2.3 Adding c232hm support to AVRDUDE [Optional]

This section is completely optional, since if you have the bootloader on the Il Matto you can program/read flash/EEPROM/etc with just the USB cable. This is only required if you wish to change the fuses (or lock bits), or download the bootloader onto the device. If you only need this once it might be easiest to go to the level 2 lab where the software is already installed.

\(^2\) On some versions of Eclipse this may say "Software and Workspace Center"
2.3.1 AVRDUDE

Browse to the WinAVR installation location (e.g. C:\WinAVR-20100110) and enter the bin folder. Create a folder called 'old' and into it copy the files avrdude.exe and avrdude.conf from the bin folder. Then download the patched version with c232hm support which is named avrdude-5.11svn1608-Patch7729-win32.zip. Extract the the archive to a temporary directory (e.g. C:\Downloads) and copy all the contents to the bin folder of your WinAVR installation (e.g. C:\WinAVR-20100110\bin). This means that your AVRDUDE command when executed from the command prompt will use the new patched version.

2.3.2 Eclipse

The AVR plugin for Eclipse does not work well with this new patched version of AVRDUDE and therefore it is best to point Eclipse at the original version. If you wish to use the c232hm cable in programming mode use it from the command line. In Eclipse, navigate to Window->Preferences. Under AVR->Paths edit the AVRDude path (path source custom) to the folder 'old' you created in the bin folder of WinAVR, which contains the version of AVRDude that came with WinAVR.
3  Mac

3.1  Host Environment Setup

3.1.1  Command Line Tools

This guide will focus on the installation of command line tools for lion onwards as of september 2012, for other versions it is assumed you can find a guide online on how to install it. Firstly register for a free apple developer account and then login and download the command line tools .dmg for your OS version. Mount the .dmg and follow standard OS X installation.

3.1.2  Eclipse

Download Eclipse IDE for C/C++ Developers extract the archive to a reasonable install location (e.g. C:\Downloads) and run the Eclipse binary within (note: you need the JAVA runtime library installed).

3.2  Embedded Environment Setup

3.2.1  CrossPack

Download CrossPack (latest version is 20120217 at time of writing), open the .dmg and run the package installer then restart your system.

3.2.2  FTDI Drivers

Firstly download the VCP drivers, open the dmg and select the package for your version of OS-X. Run through the package installation, following the documentation.

Next download the latest D2XX drivers; follow the instructions for installation found in the download archive as they are more up to date and apply to the appropriate version.

Important: In use you may have to manually disable the serial driver in order that the D2XX drivers can be used. The serial VCP driver can be disabled and re-enabled with the following commands:

kextunload /System/Library/Extensions/FTDIUSBSerialDriver.kext
kextload /System/Library/Extensions/FTDIUSBSerialDriver.kext
3.2.3 Eclipse AVR Plugin

Open Eclipse and go Help - Software and Workspace Center. Select the Software Tab and click the add site link. Enter AVR for Name: and http://avr-eclipse.sourceforge.net/unupdatesite/ for Location: - click ok - check "AVR Eclipse Plugin" - click next and install.

3.3 Adding c232hm support to AVRDUDE [Optional]

If you have the bootloader on the the Il Matto you can program/read flash/eeprom/etc with just the USB cable. This is only required if you wish to change the fuses (or lock bits, etc), or download the bootloader onto the device. If you only need this once it might be easiest to go to the level 2 lab where the software is already installed.

3.3.1 Homebrew

Open a terminal prompt and enter the command:

```
ruby -e "$(curl -fsSkL raw.github.com/mxcl/homebrew/go)"
```

3.3.2 AVRDUDE

Use homebrew to install pkgconfig, libusb, libusb-compat, and readline with the commands

```
brew install pkgconfig
brew install libusb
brew install libusb-compat
brew install readline
```

Now download the Mac installation archive and extract the contents to a temporary directory. Now to build AVRDude with C232hm support:

1. Open a terminal in the ncurses directory of the extracted archive (type 'cd ' and drag and drop the folder onto the terminal windows and enter the command)

2. Now enter the generic build instructions for unix programs: ./configure, then make, and then sudo make install; the root password is required for sudo make install

3. Change directory to the avrdude folder inside the trunk folder. To install avrdude the process is almost identical as the previous step however this time enter ./bootstrap first to setup the ./configure command.

---

3See http://support.apple.com/kb/ht1528 for how to configure the root user in Mac OS X.
Note: If permission is denied for ./configure or ./bootstrap use chmod +x ./configure or chmod +x ./bootstrap. If sudo make install fails due to lack of permission for the command install-sh use chmod 755 ./install-sh. sudo may be required to run these commands.

3.3.3 Eclipse

The AVR plugin for Eclipse does not work well with this new patched version of AVRDUDE and therefore it is best to point Eclipse at the original version. If you wish to use the c232hm cable in programming mode use it from the command line. In Eclipse, navigate to Eclipse->Preferences. Under AVR->Paths edit the AVRDude path (path source custom) to the bin folder (the one that contains the avrdude program) found at /usr/local/CrossPack-AVR. This gets Eclipse to use the older version of AVRDude. Note if you having trouble finding the bin with avrdude in it folder in Finder use the Go To Folder dialog (under Go) to open /usr/local/CrossPack-AVR in finder.
4 Ubuntu

Note that `apt-get` is heavily used in this install. Make sure that your version is updated with `sudo apt-get update` terminal command. This guide should be applicable to other distributions of Linux, although with a different package manager on non-debian distros. All commands in typewriter font should be entered from a terminal window.

4.1 Host Environment Setup

4.1.1 GNU Build Tools

```
sudo apt-get install build-essential automake autoconf gdb
```

4.1.2 Eclipse

```
sudo apt-get install eclipse eclipse-cdt
```

4.2 Embedded Environment Setup

4.2.1 AVR Build Tools

```
sudo apt-get install gcc-avr gdb-avr avr-libc
```

4.2.2 AVRDUDE

```
sudo apt-get install avrdude
```

4.2.3 FTDI Drivers

Download the FTDI drivers for Linux and extract them. Navigate to the build directory and change directory to your system architecture.

```
sudo -s
cp lib* /usr/local/lib
chmod 0755 /usr/local/lib/libftd2xx.so.1.1.12
ln -sf /usr/local/lib/libftd2xx.so.1.1.12 /usr/local/lib/libftd2xx.so
exit
```

Note: VCP drivers are already installed by default.
4.2.4 Eclipse AVR Plugin

Open Eclipse and go Help - Install New Software... - then click Add... - Enter AVR for Name: and http://avr-eclipse.sourceforge.net/updatesite/ for Location: - click ok - check "AVR Eclipse Plugin" - click next and install.

4.3 Adding c232hm support to AVRDUDE [Optional]

If you have the bootloader on the the Il Matto you can program/read flash/eeprom/etc with just the USB cable. This is only required if you wish to change the fuses (or lock bits, etc), or download the bootloader onto the device. If you only need this once it might be easiest to go to the level 2 lab where the software is already installed.

4.3.1 AVRDUDE

To begin you need to move the existing AVRDUDE files to a different directory to avoid them being over-written. Execute which avrdude to find the current versions of avrdude and avrdude.conf. Create a directory called avrdude.dist in the same directory as avrdude and move avrdude and avrdude.conf into this directory.

Now you can build and install the patched version.

```bash
sudo apt-get install patch libreadline-dev libncurses-dev libusb-dev
sudo apt-get install libftdi-dev bison flex avrdude
```

and then to set up all avrdude dependencies:

```bash
sudo apt-get build-dep avrdude avrdude-doc
```

Download the Linux installation archive, extract it and change directory to the folder you just extracted. Now to build AVRDude with C232hm support:

1. Open a terminal in the libusb directory on the extracted archive (type cd and drag and drop the folder onto the terminal windows and enter the command)
2. Now enter the generic build instructions for unix programs: ./configure, then make, and then sudo make install; the root password is required for sudo make install
3. Repeat for libusb-compat and libftdi
4. Now change directory to the avrdude folder inside the trunk folder. To install avrdude the process is almost identical as the previous step however this time the enter ./bootstrap first to setup the ./configure command
4.3.2 Eclipse

The AVR plugin for Eclipse does not work well with this new patched version of AVRDUDE and therefore it is best to point Eclipse at the original version. If you wish to use the c232hm cable in programming mode use it from the command line. In Eclipse, navigate to Eclipse->Preferences. Under AVR->Paths edit the AVRDude path (path source custom) to point to the avrdude.dist directory. This gets Eclipse to use the older version of AVRDude.