

Appendix A

Setup Eclipse and Toolchain

In this course Eclipse is used as the IDE, as it's cross-platform and highly customizable. It is recommended to use a clean install of Eclipse for C/C++. Besides Eclipse you'll need to install some Eclipse plugins and drivers for STLink, the device that connects to your Nucleo board. The guide is written with Windows and GNU/Linux in mind, but it should work on OSX as well. For 64-bit windows the packages are available on <http://www.scintilla.utwente.nl/docs/cursus/MicrocontrollerCourse2017/Software>

A.1 Windows

A.1.1 Eclipse

1. Go to <http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers/neon1> and download **Eclipse IDE for C/C++ Developers**. Make sure you install the correct version (x64 or x86).
2. Unpack and install.
3. If the installer prompts you that you need a newer Java Runtime Environment (JRE), download the newest JRE.
4. In Eclipse, go to Help → Install New Software
5. click on Add..., fill in Name: *GNU ARM Eclipse Plug-ins*. Location: <http://gnuarmeclipse.sourceforge.net/updates> and press OK.
In case you get an *Unable to get repository* error, please check <http://gnuarmeclipse.github.io/blog/2017/01/29/plugins-install-issue/>
6. select all but the Freescale Project Templates, click Next and install.

A.1.2 Toolchain

1. Get the toolchain from <https://launchpad.net/gcc-arm-embedded>. (gcc-arm-none-eabi***-win32.exe)¹
2. Install the toolchain, but in the final window disable "Add path to the environment variable".
3. Download the latest build tools (gnuarmeclipse-build-tools-win32-2.*-*.setup.exe) from <https://github.com/gnuarmeclipse/windows-build-tools/releases>
4. Run the installer, remember the path of the Build Tools.
5. In Eclipse, go to Window → Preferences. C/C++ → Build → Global Tools Paths
6. Locate the installed Build tools, enter the path in Build Tools folder.
7. Select *GNU Tools for ARM Embedded Processors*, locate the toolchain and enter the path in Toolchain folder, and click Apply.

¹On 13/2/2017: https://launchpad.net/gcc-arm-embedded/5.0/5-2016-q3-update/+download/gcc-arm-none-eabi-5_4-2016q3-20160926-win32.exe

A.1.3 STLink v2.1 driver

1. Download the STLink v2.1 driver from <http://www.scintilla.utwente.nl/docs/cursus/MicrocontrollerCourse2019/Software/stsw-link009.zip>
2. Extract files and run `dpinst_amd64.exe` for a 64-bit system, or `dpinst_x86.exe` for a 32-bit system.

A.1.4 Debugger - OpenOCD

This is optional, but a debugger might help you a lot. We're using OpenOCD as it's available for all platforms, and easily integrates with the Eclipse plugins we installed.

1. Download the latest stable (0.10.*) version of OpenOCD for your architecture from <https://github.com/gnuarmclipse/openocd/releases/download/gae-0.10.0-20170124/gnuarmclipse-openocd-win64-0.10.0-201701241841-setup.exe>.
2. Follow the installation procedure.
3. In Eclipse go to Window → Preferences → Run/Debug → String Substitutions
4. Fill in the path to the bin directory of OpenOCD in the Value field of `openocd_path`, then click OK.
5. The next steps only apply after making a project, see section ??.

A.1.5 Packs - Device Support

1. In Eclipse, open the Packs perspective.
2. Click on the Refresh button. It will now load all available packs from Keil.
3. Select the device menu, locate the STM32F4 series and install the package.

A.2 Linux

This guide assumes a working Java runtime environment. It was tested using OpenJDK 1.8, and should work equally well with a recent version of Oracle JRE.

A.2.1 Toolchain and Eclipse

This guide works for Eclipse NEON 1 and Eclipse Mars. The newer NEON 2 is **not** supported.

1. Get the toolchain from <https://launchpad.net/gcc-arm-embedded> (`gcc-arm-none-eabi***-linux.tar.bz2`)
2. Extract to a directory of your liking.
3. Install Eclipse Luna for C/C++ development, if you haven't already.²
4. open Eclipse, set a workspace and click on Help → Install New software.
5. click on Add..., fill in Name: GNU ARM Eclipse Plug-ins. Location: <http://gnuarmclipse.sourceforge.net/updates> and press OK
6. select all but the Freescale Project Templates, click Next and install.
7. In case you get an *Unable to get repository* error, please check <http://gnuarmclipse.github.io/blog/2017/01/29/plugins-install-issue/>
8. In Eclipse, go to Window → Preferences. C/C++ → Build → Global Tools Paths
9. Select GNU Tools for ARM Embedded Processors, locate the toolchain and enter the path in Toolchain folder.

²Either via your package manager, or by downloading from <http://www.eclipse.org/downloads/>

STLink flasher for Linux

The STLink Utility provided by STMicroelectronics is Windows only, but an open source alternative is available on GitHub.

1. Go to <http://github.com/texane/stlink>, download the Zip file
2. Extract the zip file to a preferred location.
3. In a terminal cd to the directory the files are in, and build stlink using:

```
./autogen.sh
./configure
make
sudo make install
```
4. get the path of st-flash using:

```
whereis st-flash
```
5. In Eclipse, click Run → External Tools → External Tools Configurations
6. Click on Program, and then on New. Name the new configuration *st-linkv2 flash*, and paste the path to st-flash in the Location field.
Working directory: `${project_loc}/Release`
Arguments: write `${project_name}.bin 0x8000000`
7. Click Apply and close.

A.2.2 Debugger - OpenOCD

This is optional, but a debugger might help you a lot. We're using OpenOCD as it's available for all platforms, and easily integrates with the Eclipse plugins we installed.

1. Download the latest stable (0.10.*) version of OpenOCD for your architecture from <http://sourceforge.net/projects/gnuarmeclipse/files/OpenOCD/GNULinux/>.
2. Extract the package to a directory.
3. In Eclipse go to Window → Preferences → Run/Debug → String Substitutions
4. Fill in the path to the bin directory of OpenOCD in the Value field of *openocd_path*, then click OK.
5. The next steps only apply after making a project, see section ??.

A.2.3 Packs - Device Support

1. In Eclipse, open the Packs perspective.
2. Click on the Refresh button. It will now load all available packs from Keil.
3. Get a cup of coffee.
4. Select the device menu, locate the STM32F4 series and install the package.
5. Drink the coffee.