Using Mono on the Pi for GPIO and/or I^{2}C

These instructions will assist you in creating/running C# applications on the Raspberry Pi Model B+ or Model 2 platforms that make use of the NAIT produced Pi library. It is assumed that you have already executed the steps as outlined in the document 'Raspberry Pi Initial Configuration'. The assumptions are:

- monodevelop, mono-complete, monodevelop-versioncontrol, xterm, and i2c-tools are installed on your Raspberry Pi.
- I^{2}C is enabled on your Pi

The first thing to do is obtain the library project from the GitHub repository. Ensure that you have a valid internet connection and bring your Pi up-to-date with an update and upgrade:

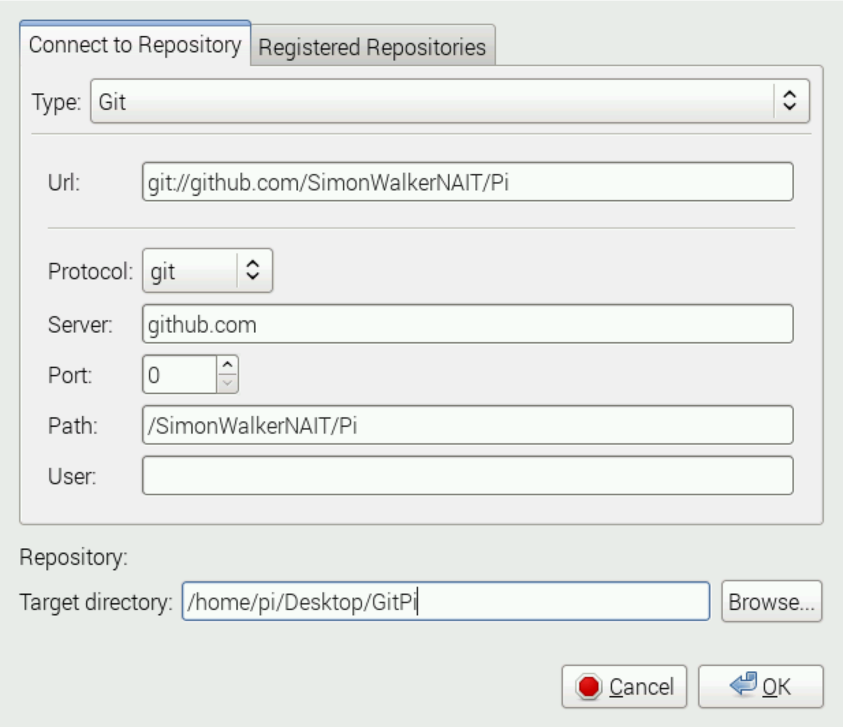
From the console:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

Next, start MonoDevelop so that you may pull down the GPIO/I^{2}C library project from GitHub.

From the menu, select 'Version Control', then 'Checkout'. You will be presented with a 'Select Repository' dialog with two tabs. On the 'Connect to Repository' tab, enter the information as shown below:



Type: Git

Url: git://github.com/SimonWalkerNAIT/Pi

Protocol: git

Server: github.com

Port: 0

Path: /SimonWalkerNAIT/Pi

User:

Repository:

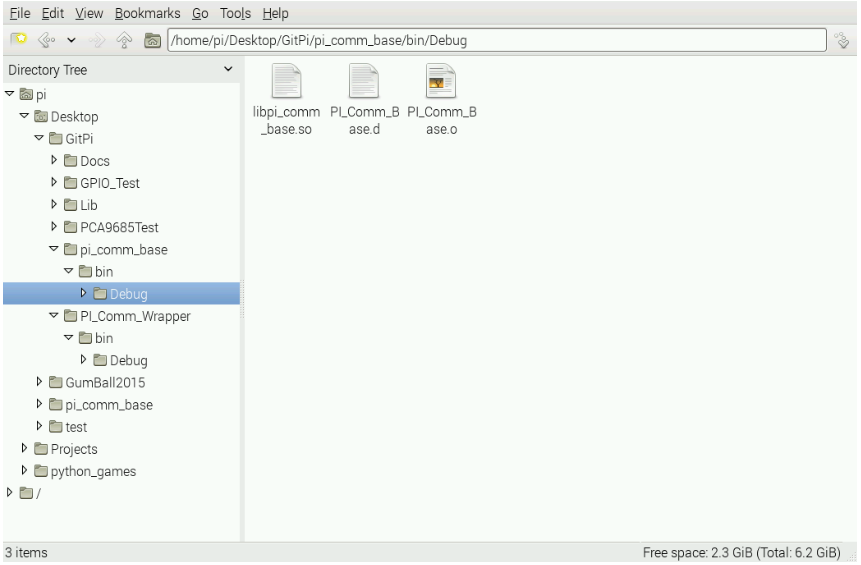
Target directory: /home/pi/Desktop/GitPi

Note the change to the 'Target directory' – ensure that you select a location that is suitable.

When you are done, hit the ‘OK’ button. The project should be pulled from GitHub to your target folder, and the project will automatically load.

You may build the project by selecting ‘Build’ then ‘Build All’.

Once built, you will need to copy the ‘libpi_comm_base.so’ and ‘Pi_Comm_Wrapper.dll’ files to the output folders (‘Debug’) of each project you wish to run. The ‘.so’ library will be in the ‘Debug’ folder of the ‘pi_comm_base’ project folder, and the ‘.dll’ file will be in the ‘Debug’ folder of the ‘Pi_Comm_Wrapper’ project folder. The ‘dll’ file should have been automatically copied to the sample projects from the build process, but the ‘so’ file will not.



/home/pi/Desktop/GitPi/pi_comm_base/bin/Debug

These two files need to be in the same location as any executables you create that use the library. Typically you will copy them to the ‘Debug’ folder of your projects. If you are running an executable as stand-alone, you will copy the files to the same folder as that executable.

The ‘GPIO_Test’ project contains a starter framework that you may use to test GPIO functionality on your Pi. You do not need sudo privileges to use the GPIO library so you may run GPIO programs directly from the IDE.

The ‘PCA9685Test’ project is specifically setup to operate the PCA9685 device through I^{2}C. It can be used to demonstrate general I^{2}C operations, but as each I^{2}C device is different, and each I^{2}C device may be addressed differently, it will only help conceptually for setups that are in any way different.

Using the I2C library requires that you have sudo privileges. In order to run these programs you can use the console at the executable location with ‘sudo mono x’ where ‘x’ is the full name of your executable.

This library is a work in progress. Additional devices will be added over time. Please feel free to use the library in any way you like.

Please send any comments or suggestions to simonw@nait.ca