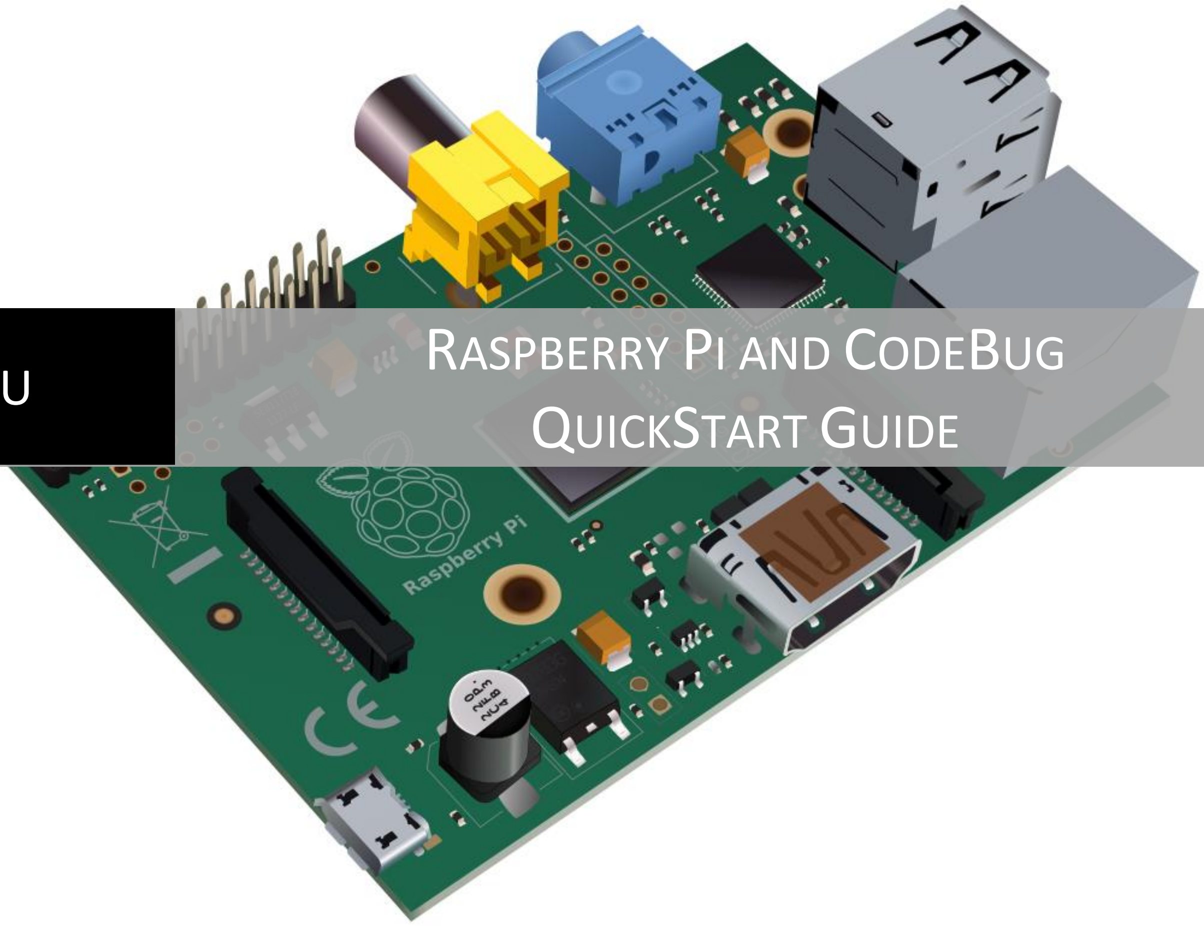
25-May-16



DCU

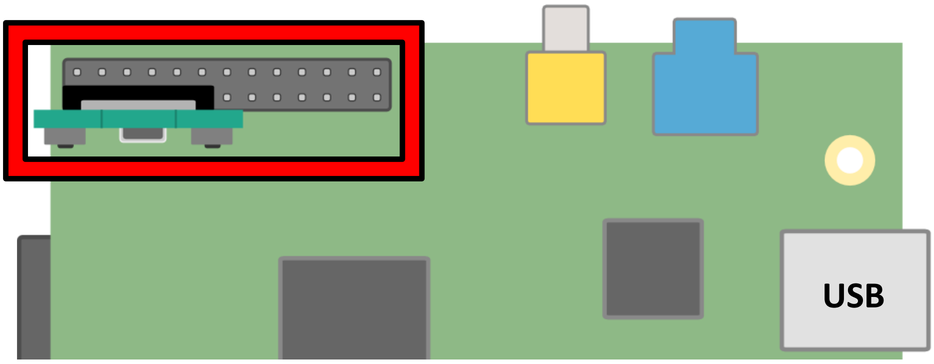
Raspberry Pi and CodeBug QuickStart Guide

For Lecturers and Students | Filip Nikolic

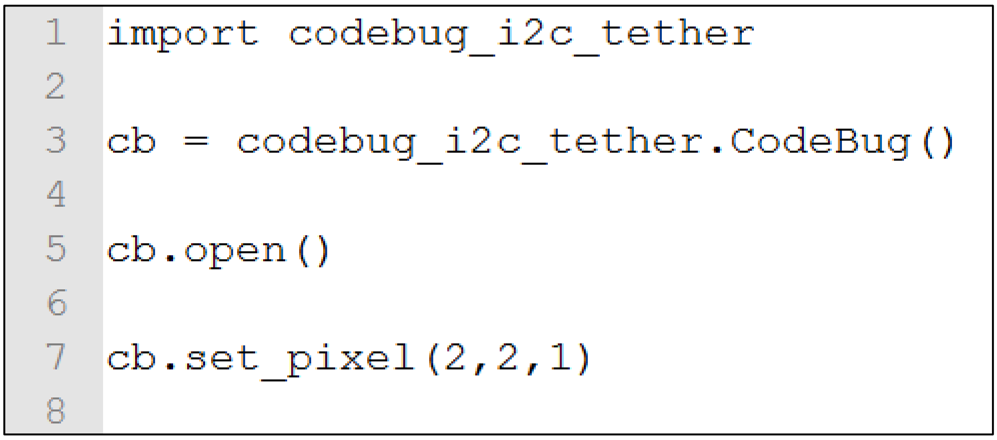
Setting up the Development Environment

Linux development environment

1. Update the OS version before attempting to get python and CodeBug packages by typing: **sudo apt-get update** followed by **sudo apt-get upgrade** in the terminal.
2. Get the CodeBug dependencies using : **sudo apt-get install python3-codebug-i2c-tether**
3. Enable I2C interface on the Pi by typing **sudo raspi-config** in the terminal, navigate through the GUI to Advanced Options -> I2C then select **yes** for both questions. You will have to reboot following this step.
4. While the Raspberry Pi is switched off, plug the CodeBug into the GPIO pins like shown in the image below and then power on the device.



5. Open the Python IDE and select the file you want to run on the CodeBug. Press F5 and the file will be compiled and uploaded, however there are some libraries which need to be included in every python file. Here is an example of a small program which will switch on an LED in the CodeBug 5X5 matrix:



```
import codebug_i2c_tether

cb = codebug_i2c_tether.CodeBug()

cb.open()

cb.set_pixel(2,2,1)

```

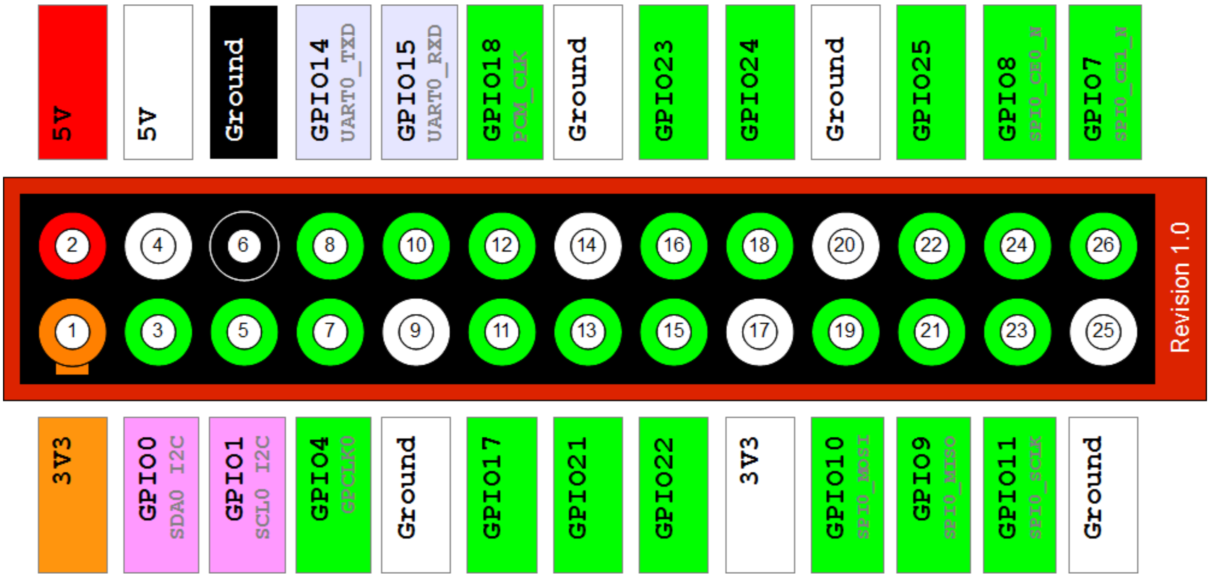
6. Finally to see the full documentation on the integration of CodeBug and Python visit this website.

***Video tutorial

Using the online IDE (Windows)

1. Plug the Micro-USB cable into a USB-port on your computer. While holding down button A on your CodeBug, plug the other end of the Micro-USB into the socket on CodeBug. The LED in the top left of CodeBug's LED display will flash when CodeBug is ready. You can now let go of button A.
2. Go to this website and use the online IDE for the codebug, which uses Scratch as it main programming interface. In addition it has an emulator, however some timing dependant programs will not execute correctly.
3. Finally, save the file and drag it onto CodeBug.

Raspberry Pi GPIO Pins:



Student Labs

1. Variables
2. Booleans, strings, floating-point and print
3. If/else flow control
4. While, for loops and input
5. Lists
6. Sorting (insertion, selection)
7. Functions
8. Assertions and binary search
9. File I/o
10. Classes

*Some very hard to find documentation