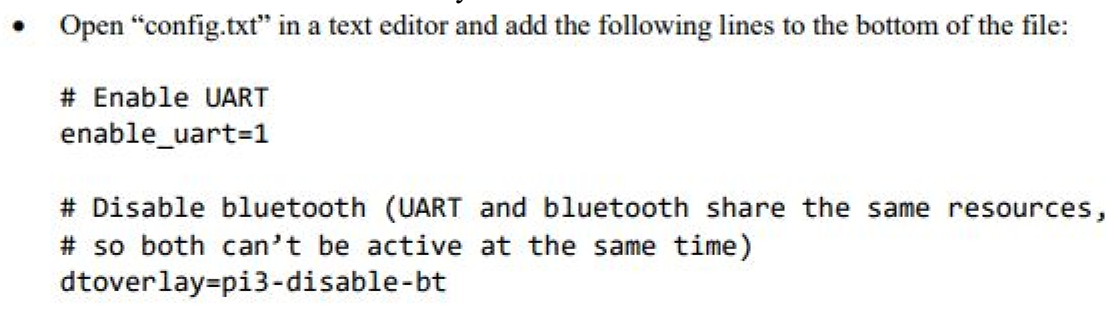
Overview

The baremetal-pi-tools project allows for programs to be loaded onto the Raspberry Pi (RPi) without removing the SD card from the board once setup. This works by compiling the program and sending it to the RPi over UART.

Usage

1. Clone or download the repository at **https://github.com/Bindernews/baremetal-pi-tools.git**

2. Load the **\release\kernal7.img** provided in the git repository onto the RPi
   a. This provides the interface for the bootloader
   b. Ensure that UART is enabled on your RPi



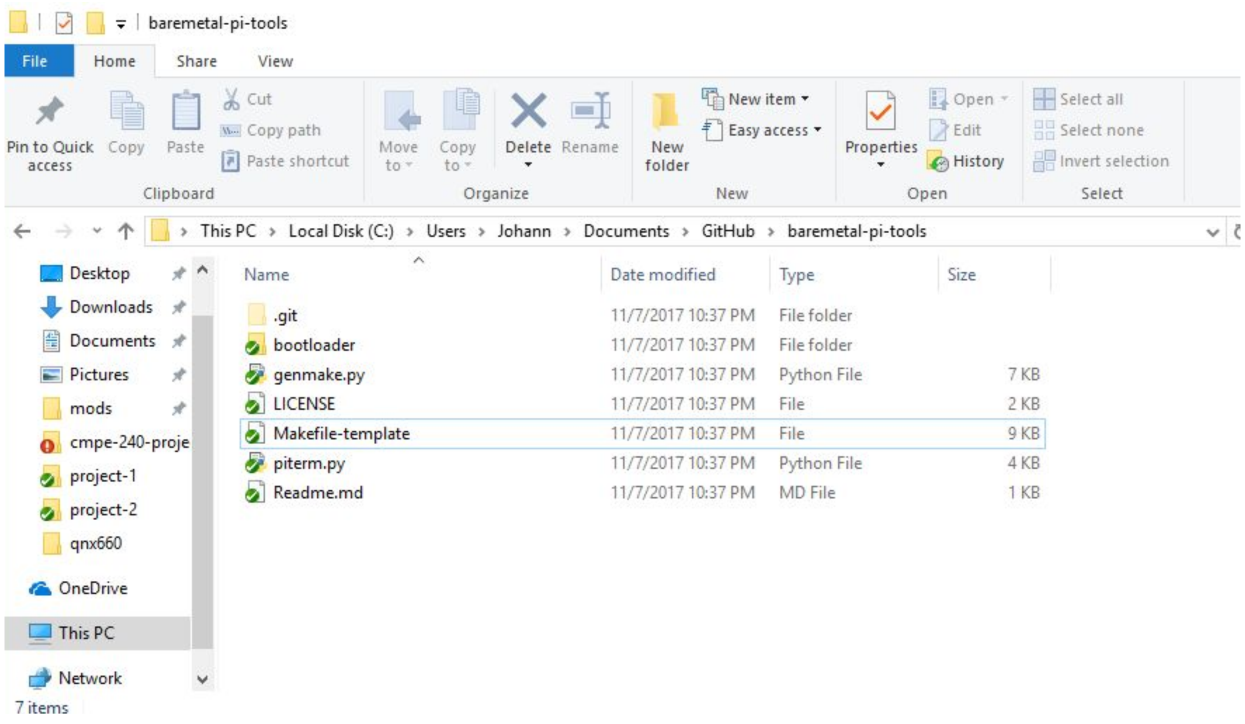
- Open "config.txt" in a text editor and add the following lines to the bottom of the file:

```
# Enable UART
enable_uart=1

# Disable bluetooth (UART and bluetooth share the same resources,
# so both can't be active at the same time)
dtoverlay=pi3-disable-bt
```

Figure 1. Enable UART instructions from lab manual

3. Create makefile by using genmake.py
   a. Navigate to the baremetal-pi-tools repository clone and find **baremetal-pi-tools\genmake.py**

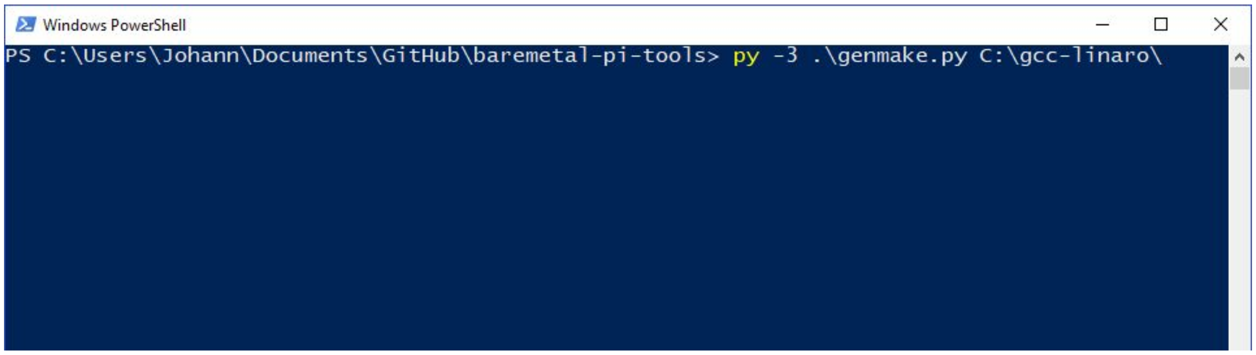


This PC > Local Disk (C:) > Users > Johann > Documents > GitHub > baremetal-pi-tools

Name	Date modified	Type	Size
.git	11/7/2017 10:37 PM	File folder	
bootloader	11/7/2017 10:37 PM	File folder	
genmake.py	11/7/2017 10:37 PM	Python File	7 KB
LICENSE	11/7/2017 10:37 PM	File	2 KB
Makefile-template	11/7/2017 10:37 PM	File	9 KB
piterm.py	11/7/2017 10:37 PM	Python File	4 KB
Readme.md	11/7/2017 10:37 PM	MD File	1 KB

Figure 2. genmake.py location in the GitHub repo

b. Run the genmake.py with the argument of the location of your linaro directory (i.e on windows:. **py -3 .\genmake.py C:\gcc-linaro**\)

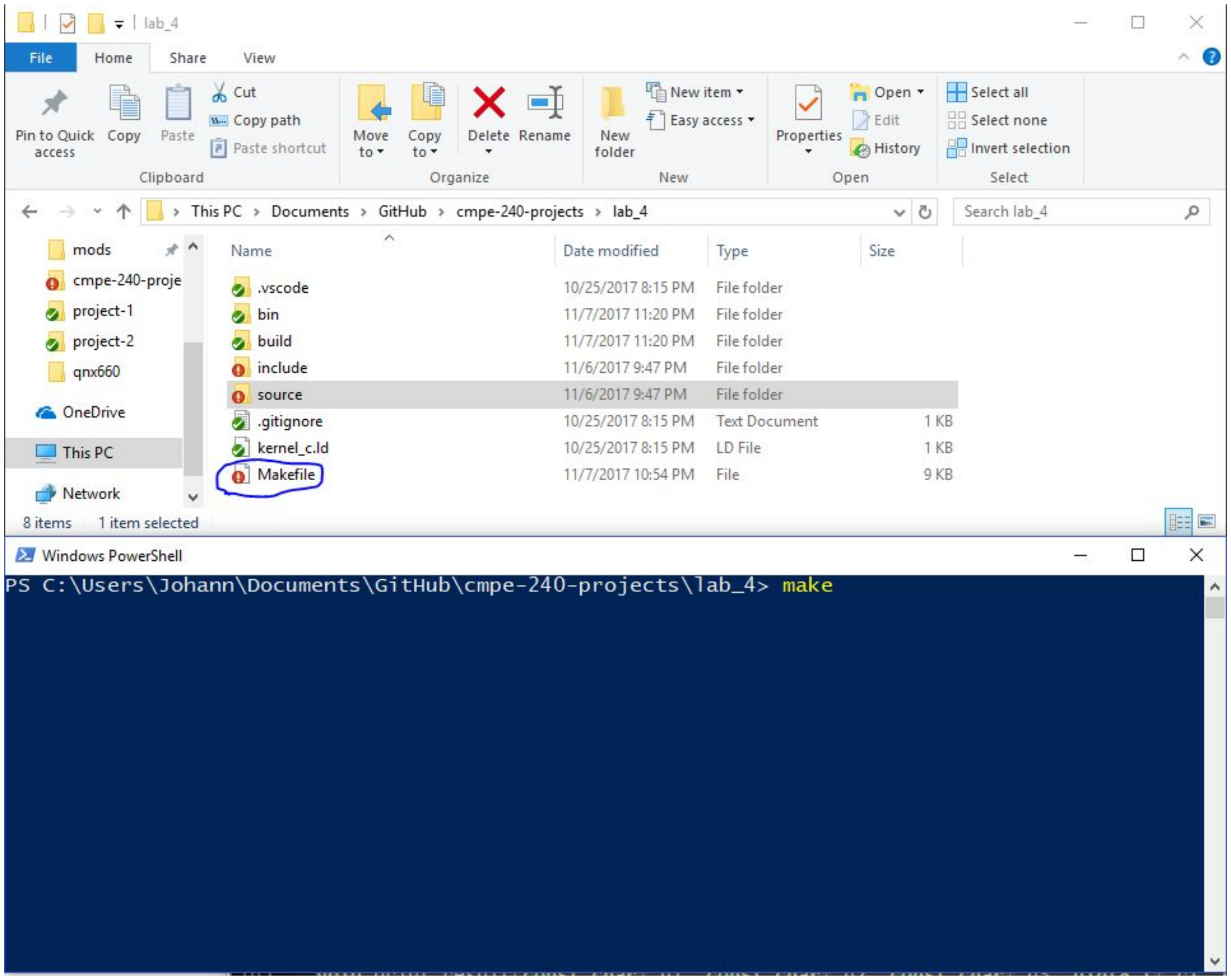


```
PS C:\Users\Johann\Documents\GitHub\baremetal-pi-tools> py -3 .\genmake.py C:\gcc-linaro\
```

Figure 3. genmake.py execution with path of linaro as parameter on windows

c. This creates a makefile which will be used when compiling labs in the **baremetal-pi-tools** directory.

4. Use generated makefile to compile lab of choice by copying into your lab directory



Name	Date modified	Type	Size
.vscode	10/25/2017 8:15 PM	File folder	
bin	11/7/2017 11:20 PM	File folder	
build	11/7/2017 11:20 PM	File folder	
include	11/6/2017 9:47 PM	File folder	
source	11/6/2017 9:47 PM	File folder	
.gitignore	10/25/2017 8:15 PM	Text Document	1 KB
kernel_c.ld	10/25/2017 8:15 PM	LD File	1 KB
Makefile	11/7/2017 10:54 PM	File	9 KB

```
PS C:\Users\Johann\Documents\GitHub\cmpe-240-projects\lab_4> make
```

Figure 4. Lab 4 directory using the newly generate makefile. Note the makefile is the same one made in the previous step.

5. Connect to the RPi via the UART bridge and then power it on
   a. Connect the UART bridge to the RPi in the manner of Figure 5. Note the **two right-most** pins are 5v pins and will fry your UART bridge if you connect to them.

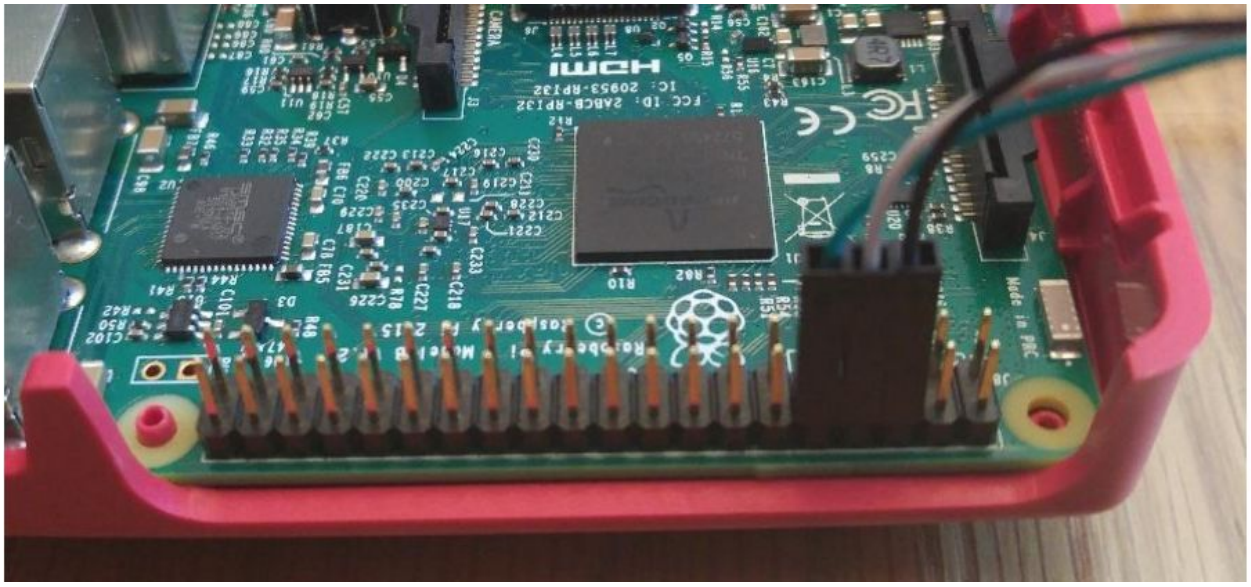
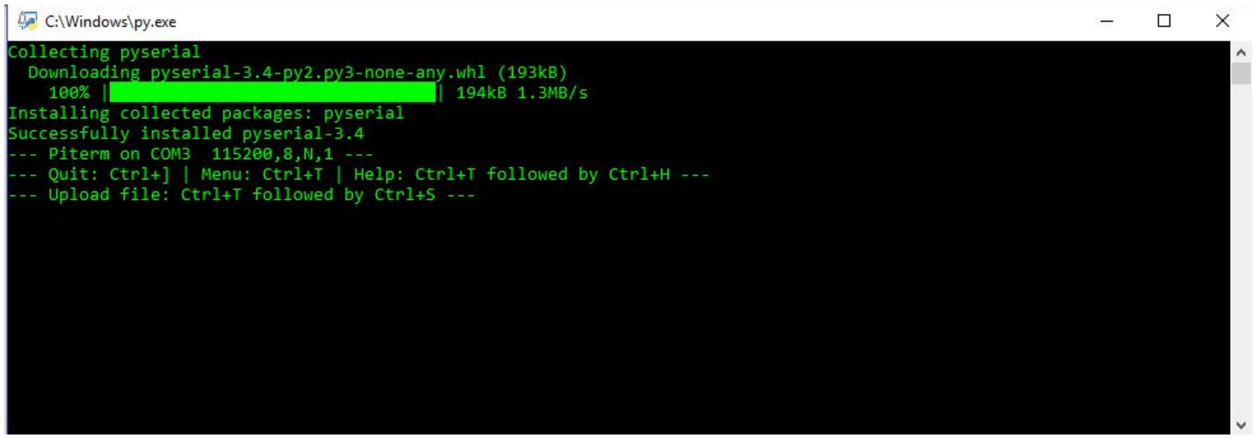


Figure 5. UART bridge connection on RPi

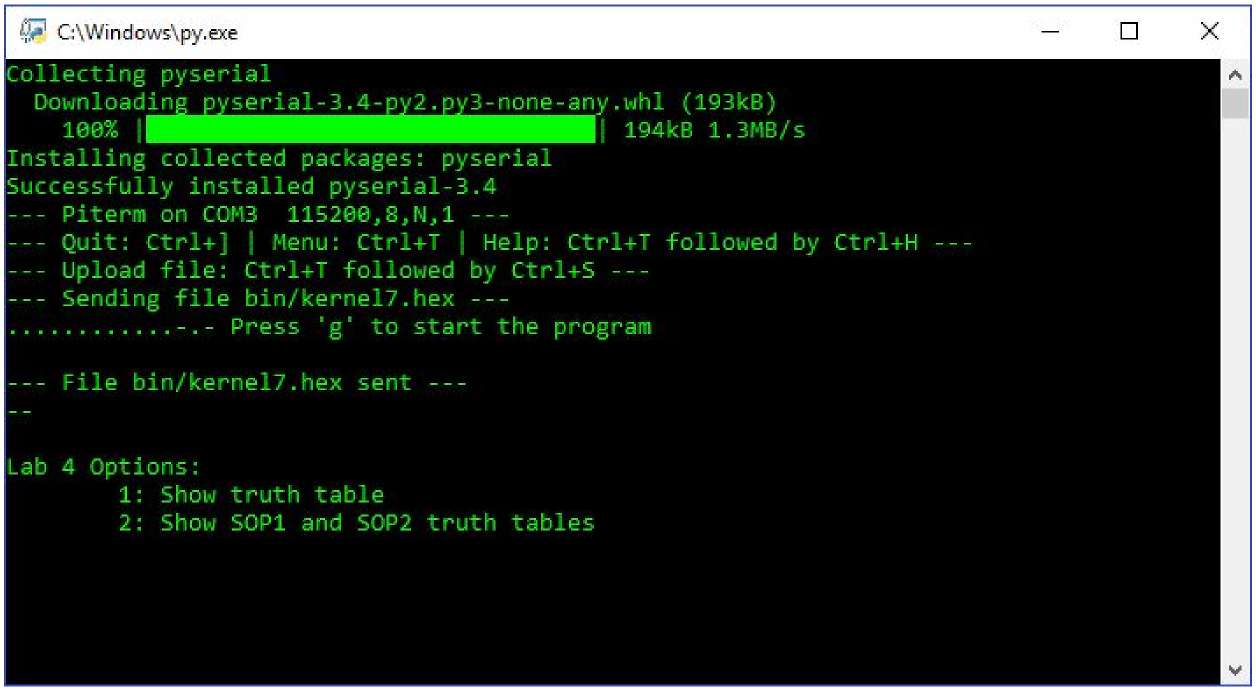
6. Actually moving your lab to the RPi
   a. Copy the **\baremetal-pi-tools\piterm.py** file into the working directory of your lab. This should be the same level as the makefile.
   b. Run **piterm.py** with your desired com port as a parameter. (i.e. **python3 piterm.py com3** or **pi -3 .\piterm.py COM3**)



```
Collecting pyserial
  Downloading pyserial-3.4-py2.py3-none-any.whl (193kB)
    100% |████████████████████████████████| 194kB 1.3MB/s
Installing collected packages: pyserial
Successfully installed pyserial-3.4
--- Piterm on COM3  115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
--- Upload file: Ctrl+T followed by Ctrl+S ---
```

Figure 6. Connecting to the RPi with **piterm.py**

   c. Use the Ctrl+T and Ctrl+S commands to load up your lab. **NOTE** if your lab is not using a baud rate or speed of 115200, you will have to reconnect to the RPi with a different serial port communications client with the appropriate settings.



```
Collecting pyserial
  Downloading pyserial-3.4-py2.py3-none-any.whl (193kB)
    100% |████████████████████████████████| 194kB 1.3MB/s
Installing collected packages: pyserial
Successfully installed pyserial-3.4
--- Piterm on COM3  115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
--- Upload file: Ctrl+T followed by Ctrl+S ---
--- Sending file bin/kernel7.hex ---
............-.- Press 'g' to start the program

--- File bin/kernel7.hex sent ---
--

Lab 4 Options:
        1: Show truth table
        2: Show SOP1 and SOP2 truth tables
```

Figure 7. Installation of lab 4 onto the RPi. Note how the lab is now accessible from this terminal window to test and run.

d. Update the program on the RPi, unplug the RPi and rerun **piterm.py** as shown in figure 6 after making the desired changes.