

DragonBoard™ 410c Quick Start Guide for Major League Hacking

I. Out of the Box	2
1. MLH DragonBoard 410c Hack Kit Contents & Accessories	2
2. HDMI Monitor Access to DragonBoard 410c and Linux Desktop	3
a. Hardware Required	3
b. Plug in Peripherals	3
c. Power on the DragonBoard 410c	3
3. Headless Access to DragonBoard 410c	3
a. Hardware Provided	3
4. Connect to Wi-Fi using Terminal (headless or on-board terminal)	9
II. Set Up Your Linux Development Environment	13
1. Update Software and Install Dependencies	14
2. Download Examples	14
3. Connect to Bluetooth	15
III. Basic Examples	16
1. General Purpose Input/Output (GPIO)	16
2. GPIO access through Command Line	17
3. GPIO Access Through Libmraa	19
4. Example Libmraa Python Program on Your DragonBoard 410c	19
a. Download and install libmraa (if you don't already have it)	20
5. Set Up Arduino Tool Chain for DragonBoard 410c	23
IV. More Examples	25
V. Support and Troubleshooting	25

I. Out of the Box

1. MLH DragonBoard 410c Hack Kit Contents & Accessories

When you check out your DragonBoard 410c hack kit from the MLH Hardware Lab, be sure you pick up the following:

- [DragonBoard 410c](#)
Pre-loaded with Debian Linux and Examples folder
(other operating systems such as **Android** and **Windows 10 IoT Core** are also supported, and download images and reflashing instructions can be found on [96Boards.org](#))
- [Sensor Mezzanine](#)
I/O expansion board for IoT/sensor applications, with Arduino processor (Atmel ATmega328), Arduino compatible shield connectors (blue), and Grove connectors (white)
- USB to microUSB Cable
- Power Adapter (12V, 2A)



You will also want to check out some of these Grove Sensors available to you from MLH:

- [Button Sensor](#)
- [Buzzer Sensor](#)
- [Light Sensor](#)
- [Sound Sensor](#)
- [Temperature Sensor](#)
- [Touch Sensor](#)
- [Rotary Angle Sensor](#)
- [LCD RGB Backlight](#)
- [LED](#)
- [Mini Servo](#)
- [Smart Relay](#)

The easiest way to access the DragonBoard and all of its features is to directly attach a HDMI monitor, USB Keyboard, and a USB Mouse. If you do not have access to these a monitor, keyboard and mouse, please proceed to page 4 for instructions on accessing and using the board in “Headless” mode.

2. HDMI Monitor Access to DragonBoard 410c and Linux Desktop

The Debian Linux from 96Boards which is installed on your DragonBoard 410c comes with the LXDE desktop environment already installed. The DragonBoard 410c can be used as a normal Linux desktop computer if you attach a keyboard, mouse and HDMI monitor.

a. Hardware Required

- [DragonBoard 410c](#)
- HDMI Monitor
- HDMI Cable
- Power Adapter
- USB Keyboard
- USB Mouse

b. Plug in Peripherals

- Connect your mouse and keyboard to the USB ports available on the DragonBoard 410c.
- Connect your HDMI monitor cable to the full-size HDMI port on the DragonBoard 410c.

c. Power on the DragonBoard 410c

With all the peripherals connected, you can power on the DragonBoard 410c and get access to the Debian desktop environment. Jump to page 17 and set up your linux development environment. You can skip the below section that gives DragonBoard 410c access to someone without the HDMI monitor.

3. Headless Access to DragonBoard 410c

This section shows you how to access the DragonBoard 410c without the need of a monitor, i.e. "headless access." You will need a personal computer.

a. Hardware Provided

Get the following items from the MLH Hardware Lab:

- [DragonBoard 410c](#)
- [Sensor Mezzanine](#)
- USB to mircoUSB Cable

- Power Adapter

Step 1: Prepare your DragonBoard 410c

You will prepare the DragonBoard 410c to be accessed with your host machine by "serial console" using the Sensors Mezzanine and a USB to microUSB cable. The Sensors Mezzanine and USB cable will act as the interface between your DragonBoard 410c and host machine.

1. DragonBoard 410c is unplugged (power not connected)
2. Sensors Mezzanine is connected to DragonBoard 410c through the 40-pin low-speed expansion connector (black connector). Sensors Mezzanine board (male pins) will match 1-to-1 with female pins on DragonBoard 410c connector. Make sure all 40 pins are lined up properly before moving to next step.

When properly attached, the two boards will look like this:



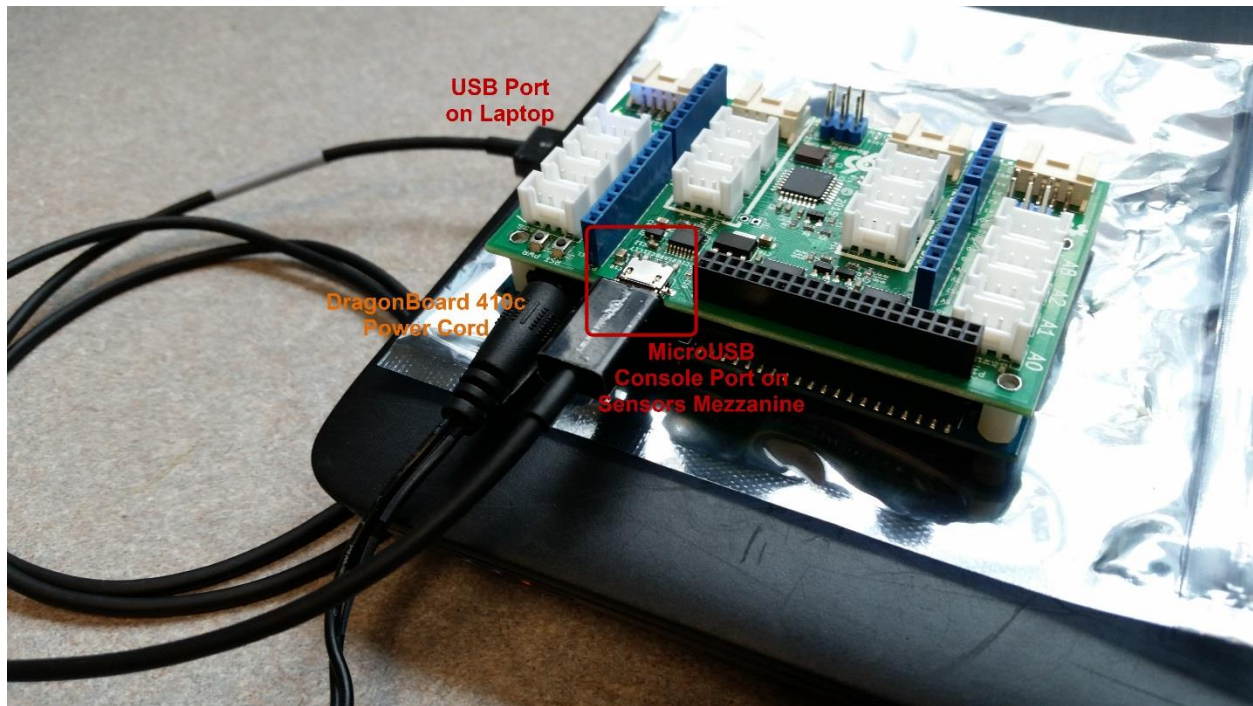
3. Connect USB cable to host machine and microUSB on mezzanine board (USB Type-A end of cable connects to host machine, microUSB end of cable connects to microUSB port on Sensors Mezzanine).

NOTE: Make sure all 40 Sensors Mezzanine pins (male) are lined up with all 40 input pins (female) on DragonBoard 410c low-speed connector

NOTE: USB Type-A to microUSB cable should run from your host machine to the microUSB port on the upper (green) board. Do not connect to the micro-USB connector on the lower (blue) board at this time.

4. Connect the power cord to the barrel jack connector on the DragonBoard 410c (lower blue board), but do not plug the adapter into a power plug.

When the microUSB cable and power cord are attached correctly, it should look like this:



Step 2: For Linux or Mac

(If you have a PC, please skip to next section “Step 2: For PC”)

Check if the device is recognized. You will need to access the terminal on your host machine to make sure the Sensors Mezzanine device (and DragonBoard 410c) are detected by the host machine.

1. DragonBoard 410c is still powered off
2. Detect the DragonBoard 410c from your host machine
 - a. From your Linux or Mac OSX machine, open the terminal application and type in the following command:

```
$ ls /dev/tty*
```

If a device is detected, you will see a long list of devices. One of these devices should be recognized as the following (or similar):

Linux: `ttyUSB0` (e.g., `ttyUSBx`)

Mac OSX: `tty.usbserial-08-15` (e.g., `tty.usbserial-XX-YY-ZZ`)

Once you have confirmed your device is being recognized by the host machine, you are ready to move on.

NOTE: You may want to unplug and plug back in the USB Type-A end of the cable from your host machine while running the above command in each time. This might help you notice when the device is detected.

NOTE: Base device name (ttyUSBX and tty.usbserial-XX-YY-ZZ) should be standard when exploring your /dev with a device connected. Be sure to match the extension for your `screen` command to the device name as listed when running `$ ls /dev/tty*`.

Access the serial console of your DragonBoard 410c. You will need to execute a command on your host machine to gain access to your DragonBoard 410c through the Sensors Mezzanine microUSB interface. The following will grant you command line access to the DragonBoard 410c file system:

3. DragonBoard 410c is still powered off
4. Type in the following command:

Linux:

```
$ screen /dev/ttyUSB0 115200
```

Mac OSX:

```
$ screen /dev/tty.usbserial-08-15 115200
```

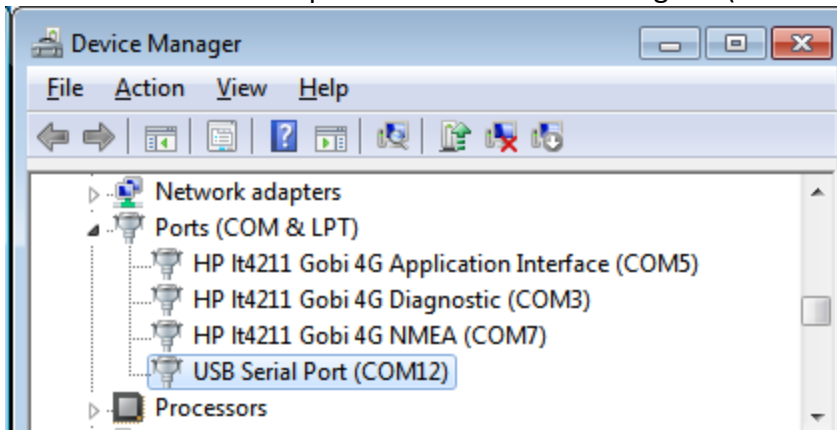
When this command is executed, your terminal will clear and you should only see black.

Step 2: For PC

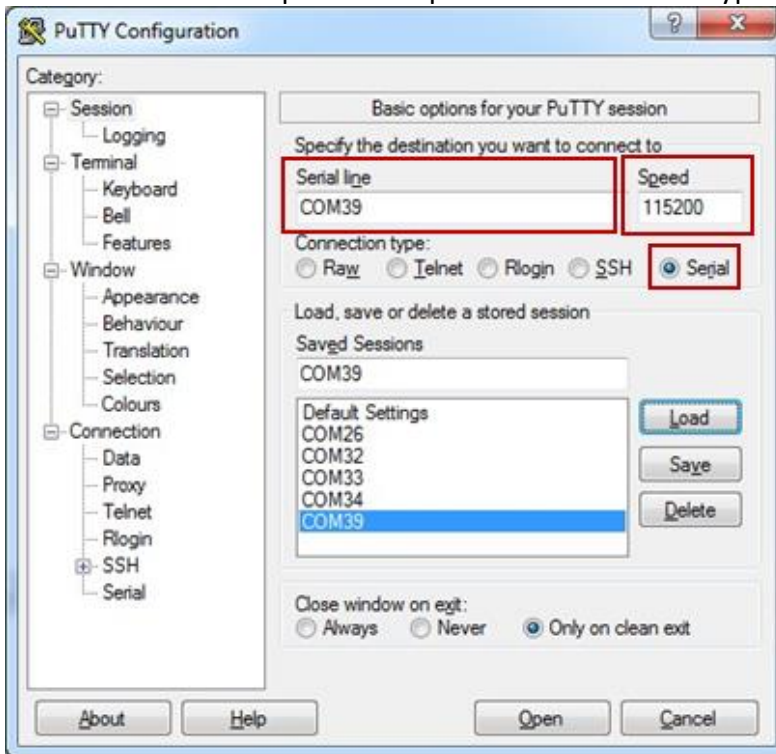
Access the serial console of your DragonBoard 410c.

1. Plug your DragonBoard 410c into power
2. Go to **Control Panel -> Device Manager -> Ports (COM and LPT)**
This might be different based on the version of Windows running on your PC

- Determine which COM port number has been assigned (COM12 in this example)

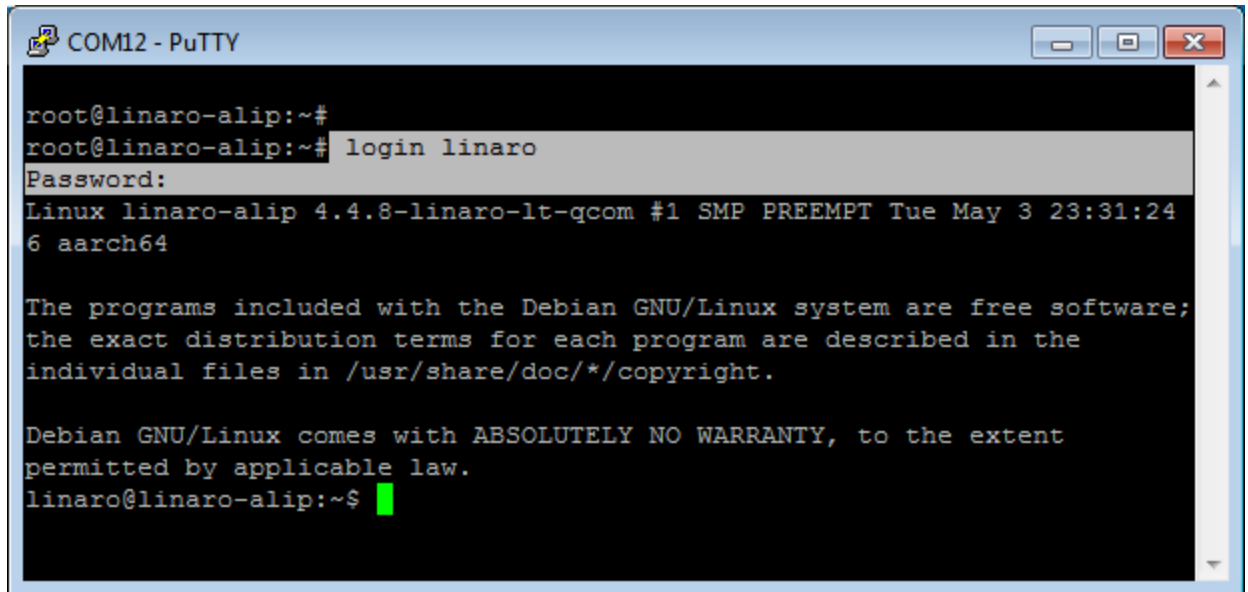


- Install PuTTY from <http://www.putty.org/>
(There are many other terminal emulator programs that will work, PuTTY is just one example)
- Start PuTTY and set up the COM port with connection type 'Serial', and Speed '115200':



- At this point, the terminal on your host machine should start to print out the boot sequence of your DragonBoard 410c. Once the DragonBoard 410c has finished booting up, you will need to press a key (ENTER), and then login as user "linaro" as follows:

```
# login linaro
Password: linaro
```



```
root@linaro-alip:~#  
root@linaro-alip:~# login linaro  
Password:  
Linux linaro-alip 4.4.8-linaro-lt-qcom #1 SMP PREEMPT Tue May 3 23:31:24  
6 aarch64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
linaro@linaro-alip:~$
```

You will then have access to your DragonBoard 410c system through your host machine terminal.

Congratulations! You are now accessing your DragonBoard 410c "Headless."

4. Connect to Wi-Fi using Terminal (headless or on-board terminal)

This section outlines two different methods of connecting to a Wi-Fi network with your DragonBoard 410c.

- Method one: Connect to Wi-Fi while accessing your DragonBoard 410c through external means (serial console or ssh "headless")
- Method two: Connect to Wi-Fi using on-board terminal when running Linaro Debian Linux on your DragonBoard 410c

Method One

In this method, you will be accessing your DragonBoard 410c through a host machine (personal computer, running either Linux or Mac OSX). (Please see the “Headless Access to DragonBoard 410c” section above to learn how to connect to your DragonBoard 410c via serial console using the Sensors Mezzanine.)

1. You are connected to the serial console of your DragonBoard 410c using the Sensors Mezzanine and a USB Type-A to microUSB cable
2. Execute the following command:

```
$ nmtui
```

This command will open an interactive GUI that will allow you to maneuver your way through the menus and choose your desired Wi-Fi network to connect to. Follow the prompt to connect to a detected Wi-Fi signal.

NOTE: The DragonBoard 410c is only capable of detecting and connecting to the 2.4GHz Wi-Fi band. If your router does not offer this frequency, the GUI will not show a useable network to connect to.

Method Two

In this method, you will be working directly on the DragonBoard 410c. By the end of this instruction you will have connected to a Wi-Fi network using only a terminal window on your local DragonBoard 410c desktop or developer OS.

1. DragonBoard 410c is running Linaro Debian OS (standard Linux OS)
2. Open Terminal application and execute the following command

```
$ nmtui
```

This command will open an interactive GUI that will allow you to maneuver your way through the menus and choose your desired Wi-Fi network to connect to. Follow the prompt to connect to a detected Wi-Fi signal.

NOTE: The DragonBoard 410c is only capable of detecting and connecting to the 2.4GHz Wi-Fi band. If your router does not offer this frequency, the GUI will not show a useable network to connect to.

Method Three

In case you don't have a HDMI monitor and have UART access to the board (e.g., [UART adapter board](#) or [Sensors Mezzanine](#)), there are a few easy ways for you to configure a wireless connection so you can then remotely access your DragonBoard 410c without any extra cables (besides the power adapter).

To show the overall status of NetworkManager:

```
root:# nmcli general status
```

STATE	CONNECTIVITY	WIFI-HW	WIFI	WWAN-HW	WWAN
disconnected	none	enabled	enabled	enabled	enabled

To show all connections:

```
root: # nmcli connection show
```

NAME	UUID	TYPE	DEVICE
------	------	------	--------

To show the device status (for the devices recognized by Network Manager):

```
root: # nmcli device status
```

DEVICE	TYPE	STATE	CONNECTION
wlan0	wifi	disconnected	--
lo	loopback	unmanaged	--

To view the list of available access points:

```
Root: # nmcli dev wifi list
```

*	SSID	MODE	CHAN	RATE	SIGNAL	BARS	SECURITY
	foonet	Infra	7	54 Mbit/s	70		WPA2
	96boards	Infra	4	54 Mbit/s	80		WPA2
	linaro-wifi	Infra	52	54 Mbit/s	7		WPA2
	debian	Infra	11	54 Mbit/s	89		WPA1 WPA2

To connect to a WIFI access point, first create the connection:

```
root: # nmcli con add con-name WiFi ifname wlan0 type wifi ssid foonet
```

```
Connection 'WiFi' (4b40221c-9af9-45ae-b5df-7d8bfe301ad5)
successfully added.
```

Then set up the password for your access point (e.g. for a WPA2 AP):

```
root: # nmcli con modify WiFi wifi-sec.key-mgmt wpa-psk
root: # nmcli con modify WiFi wifi-sec.psk myownpassword
```

Then just enable the connection:

```
root: # nmcli con up WiFi
```

Congratulations! You have connected your DragonBoard 410c to a useable Wi-Fi network using only the command line (terminal application).

Case of the Interstitial Wi-Fi

Still having difficulty connecting to Wi-Fi? It is possible that the Wi-Fi you are trying to connect to requires additional user interaction. You might need to launch a web browser on the DragonBoard to accept additional terms and conditions for using the free Wi-Fi. You can check if the Wi-Fi available to you requires this by trying to connect to it using your laptop or phone. If you are prompted to launch a browser window and accept terms of use before you can start using the Wi-Fi, you will need to do the same on the DragonBoard.

Following steps assume that you do not have a HDMI monitor connected to the DragonBoard. These will enable you to launch a web browser on the DragonBoard remotely from your host PC and accept the terms of use. You can skip these steps if you have an HDMI monitor connected to the DragonBoard, as launching the web browser and connecting to the internet over Wi-Fi should be straight forward.

A second micro-USB cable is required for this step. The basic solution is to use the micro-USB port on the DragonBoard as a virtual network connection to the host PC. Once you have a network connection to the PC you can run full X11 programs like the browser (chromium). The following steps have been tested on Windows 10, but should be OK for other operating systems.

Steps:

- 1) Connect to the serial port on the mezzanine as per the quick start instructions in this guide to “Prepare your DragonBoard” and “Access the serial console of your DragonBoard 410c.
- 2) Install WSL under Windows 10 see instructions at: https://msdn.microsoft.com/en-us/commandline/wsl/install_guide
- 3) Install XMING (X11 windowing system) see: <https://sourceforge.net/projects/xming/>

- 4) Start the Ubuntu bash shell (WSL) and edit the .bashrc on the laptop, add the following line to the bottom:
`export DISPLAY=localhost:0.0`
- 5) Determine the ip address of the Dragonboard by running the following command on the serial terminal (putty)
`ip addr`
- 6) Make sure X11Forwarding is turned on in /etc/ssh/sshd_config (config file should have the following two lines)
`X11Forwarding yes`
`X11DisplayOffset 10`
- 7) Restart the DragonBoard so the sshd changes take effect.
- 8) Start ssh under bash on the laptop and login (password linaro)
`ssh -X linaro@<ip addr>`
- 9) Test your X11 server and connection by running a simple program on the DragonBoard 410c:
`sudo apt-get install -y x11-apps`
`xeyes`
- 10) Connect the **second** micro-USB cable from your laptop to the device port on the DragonBoard (this stops operation of any USB devices on the host ports)
- 11) On the putty connection make the following changes
 - a. In the file /etc/network/interfaces add the following lines:
`auto usb0`
`iface usb0 inet static`
`address 192.168.44.2`
`netmask 255.255.255.0`
`gateway 192.168.44.3`
`network 192.168.44.0`
`broadcast 192.168.44.2`
 - b. In the file /etc/resolv.conf add the following line
`nameserver 192.168.44.2`
 - c. Run the command: `sudo modprobe g_ether`
 - d. Run the command: `sudo dhclient usb0 &`
 - e. Run the command: `sudo ifconfig usb0 192.168.44.30 netmask 255.255.255.0`
- 12) Ensure that Xming is running on the host PC and start a bash terminal (WSL)
- 13) On the bash terminal do the following
 - a. run the command: `ssh -X linaro@192.168.44.30`
 - b. log into the Dragonboard (password: linaro)
 - c. Use the nmtui program to connect to the WiFi router
 - d. Start the program `chromium&`
 - e. This will launch chrome browser on the DragonBoard that you will see on your host PC
 - f. Log into the interstitial server and accept the terms of use.
 - g. Congratulations! You should be connected to Wi-Fi

- h. Once you are logged into the network you can continue to use the bash terminal over USB

II. Set Up Your Linux Development Environment

This section will guide you through the process of setting up a useable development environment on your DragonBoard 410c. The purpose of going through this instruction set is to make sure your DragonBoard 410c possesses the appropriate dependencies needed for development.

NOTE: This step should have already been done for you with all of the latest packages and examples, but including this just in case.

Update your source.list

The following commands will ensure your DragonBoard 410c has access to all appropriate repositories when attempting to download packages using the apt-get command.

```
$ echo "deb http://deb.debian.org/debian jessie main" | sudo tee  
-a /etc/apt/sources.list  
  
$ echo "deb-src http://deb.debian.org/debian jessie main" | sudo  
tee -a /etc/apt/sources.list
```

Now that you have modified your `source.list`, you are ready to update and upgrade the package list in all your registered repositories.

Update and Upgrade

The image on your board might be out of date. This is possible even when using the stock image (the operating system your board was shipped with), or a newly flashed version from the 96Boards.org website.

A few useful commands will help us make sure everything on the board is current:

- **sudo apt-get update:** Downloads package lists from online repositories and "updates" them to get information on the newest versions of packages and their dependencies.
- **sudo apt-get upgrade:** Fetches and installs newest package versions which currently exist on the system. APT must know about these new versions by way of 'apt-get update'
- **sudo apt-get dist-upgrade:** In addition to performing the function of upgrade, this option also intelligently handles changing dependencies with new versions of packages

1. Update Software and Install Dependencies

Execute the following commands when you have access to Wi-Fi or Ethernet connection:

```
$ sudo apt-get update
$ sudo apt-get upgrade -y
$ sudo apt-get dist-upgrade -u -y
$ sudo apt-get clean
$ sudo apt-get autoremove
$ sudo apt-get install -y python3-pip
$ sudo apt-get install -y python3-dev python3-tk python3-numpy
$ sudo apt-get install -y python3-scipy
$ sudo apt-get install -y python-opencv
$ sudo apt-get install -y libopencv-dev
$ sudo sed -i '$a export
PYTHONPATH=/usr/local/lib/python3.5/dist-
packages:/usr/local/lib/python2.7/site-packages\n' ~/.bashrc
```

NOTE: If at any point during this process you are prompted with a 'Y/N', select Y and press Enter.

Reboot the DragonBoard 410c:

```
$ sudo reboot now
```

Congratulations! Your DragonBoard 410c should now be running the most current development environment.

2. Download Examples

Your DragonBoard 410c should have an **Examples** folder pre-installed in your home directory containing sample code for GPIO and OpenCV. If not, you can download the Examples from:

```
git clone https://github.com/96boards/projects.git
```


3. Connect to Bluetooth

Enabling Bluetooth and Pairing

After booting the DragonBoard 410c, make sure that the Bluetooth is on and paired by clicking on the Bluetooth icon in the taskbar. The Bluetooth icon in the system tray looks like this in the:

Unpaired state:



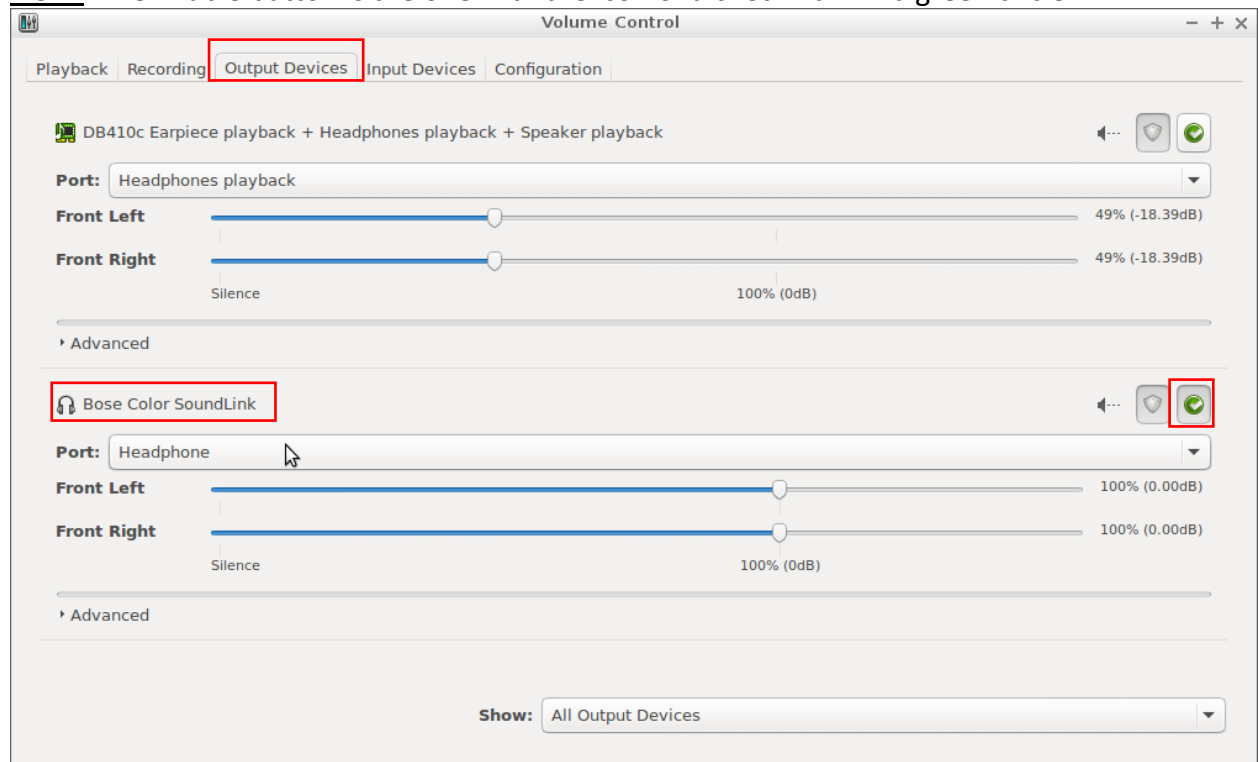
Paired state:



If you are pairing with a Bluetooth speaker, make sure your Bluetooth speaker is selected in the Sound & Video application:

1. Navigate to **[Start] -> [Sound & Video] -> [PulseAudio Volume Control]**
2. Click on the **[Output Devices]** tab
3. Click on the Enable button of the Bluetooth speaker

NOTE: The Enable button is the one with the icon of a checkmark in a green circle



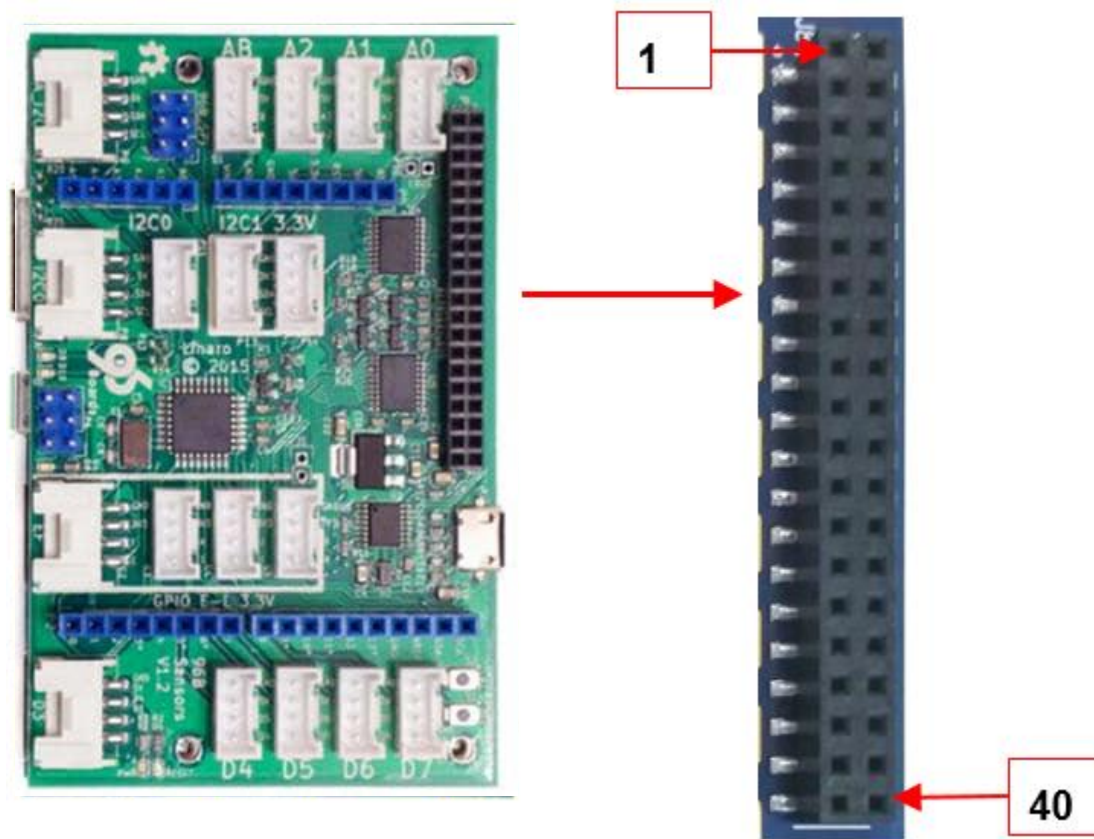
4. Close the Volume Control window

III. Basic Examples

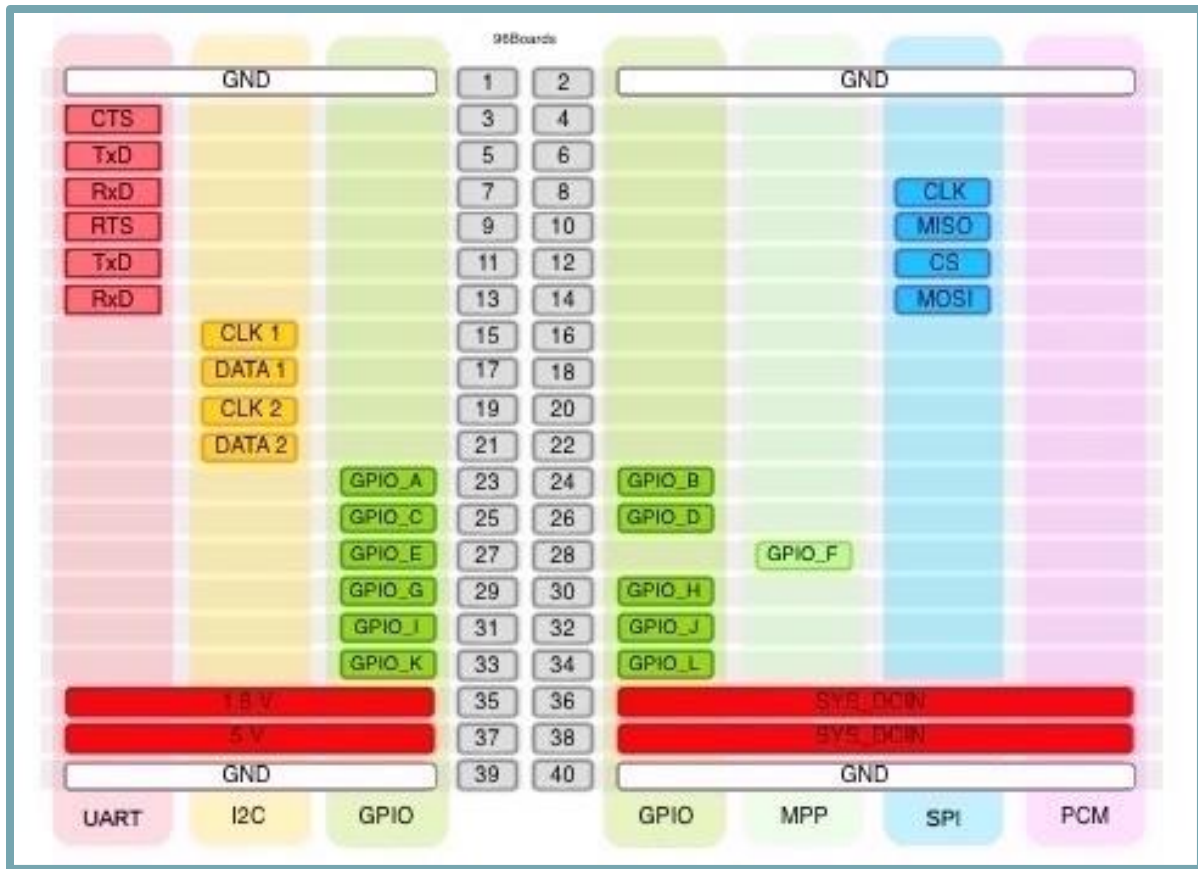
1. General Purpose Input/Output (GPIO)

GPIO provides the basic Input and Output access to the physical world for 96Boards Consumer Addition (CE) Open Hardware boards, including the DragonBoard 410c. GPIO pins can perform 'read' and 'write' operation for controlling devices connected to the DragonBoard 410c.

DragonBoard 410c provides access to GPIO pins through a 40pin **Low Speed Expansion Header** available on all CE boards. The pin number assignments are shown below:



The low speed header diagram below shows the mapping of GPIOs to the DragonBoard 410c pin numbers:



The following list specifies the GPIO pins exported by the Linux kernel for the **DragonBoard 410c**: 36, 12, 13, 69, 115, 4, 24, 25, 35, 34, 28, 33

2. GPIO access through Command Line

The following instructions are applicable to all Linux based distributions and is based on sysfs interface.

To access the GPIOs through the Linux shell, open the terminal (start menu > other > LXTerminal; right click this to add a shortcut to your desktop if you desire) and give yourself super user access:

```
$ sudo su
```

Giving yourself superuser access will allow you to modify the GPIOs (if you get an access denied for a GPIO, this means you're trying to modify a GPIO the board is using for itself).

Once you have superuser access, navigate to the gpio folder with the following command:

```
# cd /sys/class/gpio
```

Before accessing any GPIO, it needs to be exported using the following command:

```
# echo 36 > export
```

The following list specifies the GPIO pins exported by Linux kernel for the DragonBoard 410c.

DragonBoard 410c: 36, 12, 13, 69, 115, 4, 24, 25, 35, 34, 28, 33

After exporting GPIO, move into that gpio directory:

```
# cd gpio36
```

The following command returns the direction of GPIO:

```
# cat direction
```

The following command sets the direction of GPIO as input

```
# echo in > direction
```

The following command returns 0 if the pin is off, 1 if the pin is on

```
# cat value
```

The following command sets the direction of GPIO as output

```
# echo out > direction
```

Once the GPIO is set as output, following command sets value of GPIO as high.

```
# echo 1 > value
```

The following command sets value of GPIO as low.

```
# echo 0 > value
```

Using a multimeter set to measuring voltage, you can probe the pin you are toggling along with one of the ground nodes (Pins 1, 2, 39, and 40), to watch the voltage switch between ~1.8V and 0V.

3. GPIO Access Through Libmraa

Setting up Libmraa on your DragonBoard 410c

The Libmraa library is an open source effort that allows you to use the GPIOs available to you on the DragonBoard 410c with different bindings such as C++ and Python.

Requirement

You must have completed setting up your Linux Development Environment (see above). This will ensure you have updated your environment repositories and will have access to libmraa.

Downloading and installing libmraa (if you don't already have it)

NOTE: If you already have libmraa, or if you are unsure, you should run this command to guarantee its availability on your DragonBoard 410c:

```
$ sudo apt-get install libmraa-dev
```

Congratulations! You have downloaded and installed libmraa to your DragonBoard 410c.

4. Example Libmraa Python Program on Your DragonBoard 410c

This section will help you execute a simple python code using libmraa.

Step 1: Hardware Setup

Hardware Required

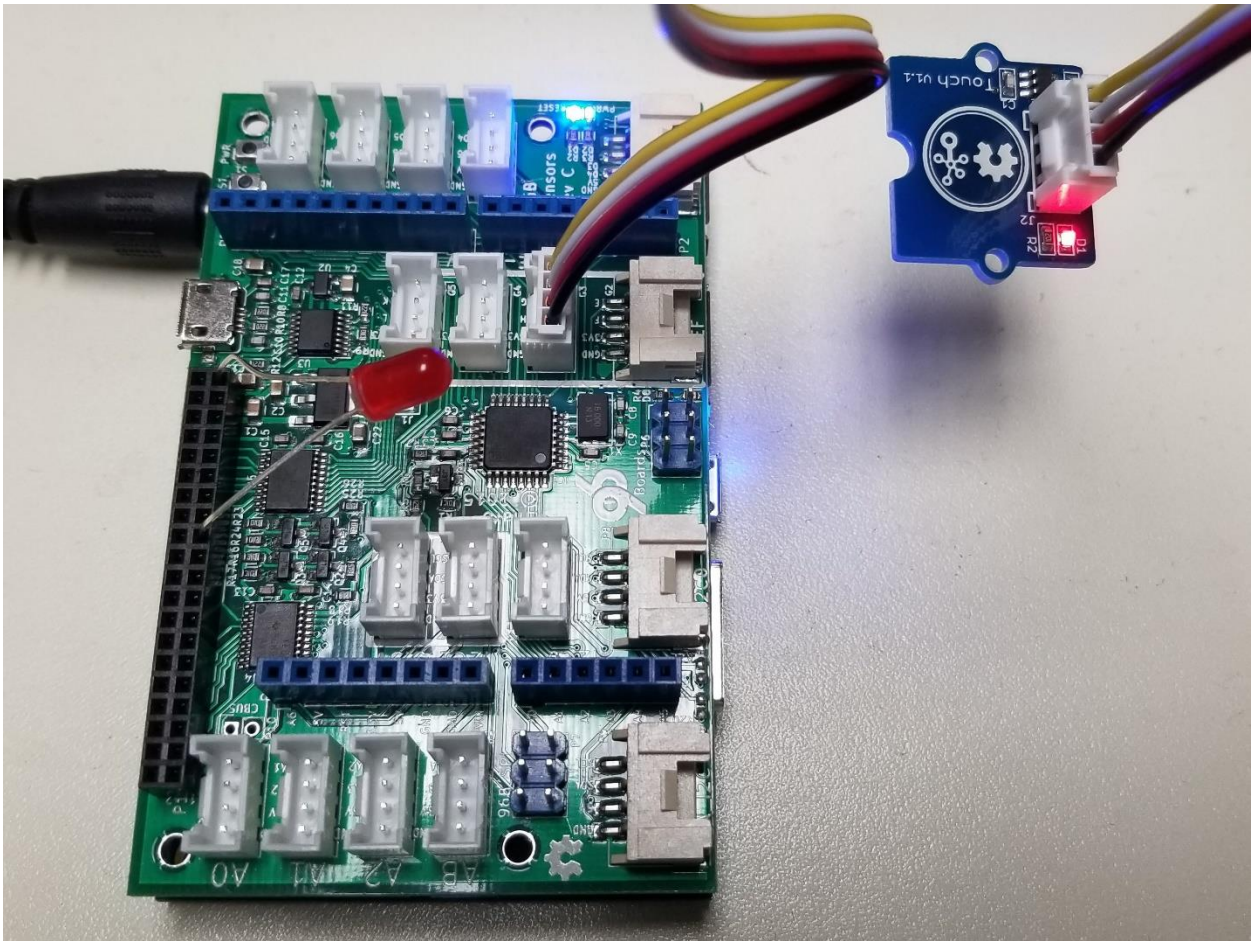
- [DragonBoard 410c](#)
- [Sensors Mezzanine](#)
- [LED](#)
- [Grove Universal 4 Pin Cable](#)
- [Grove Touch Sensor Module](#)

Hardware Connections

1. Make sure that the DragonBoard 410c is initially unplugged (powered off)
2. Connect the Sensors Mezzanine to DragonBoard 410c through the 40-pin low-speed expansion connector. Sensors Mezzanine board (male pins) will match 1-to-1 with

female pins on DragonBoard 410c connector. Make sure all 40 pins are lined up properly before moving to next step.

3. Set up your terminal interface (See “Out of Box” section for headless access method or Linux desktop method).
4. Attach the LED to GPIO 23.
5. Attach the Grove Touch Sensor Module to Grove connector G3 on the Sensors Mezzanine.



Step 2: Software Setup

a. Download and install libmraa (if you don't already have it)

NOTE: If you are using a DragonBoard 410c with [Debian Linux OS from Linaro](#), you already have libmraa.

If you get the following error,

```
ImportError: No Module named 'mraa'
```

or if you are unsure, you should run the following commands to make sure you have the correct gpio library.

```
$ git clone https://github.com/intel-iot-devkit/mraa
$ cd mraa
$ mkdir build
$ cd build
$ cmake ..
$ make
$ sudo make install
$ sudo ldconfig /usr/local/lib/
```

Step 3: Get the sample code and run it

This sample code is meant to blink an LED at a certain time interval. Follow the steps below to execute the code.

```
$ cd mraa
$ cd examples
$ cd python
$ vim blink-io8.py
```

You can go through the code to understand how it works. Since you are connecting the LED to GPIO 23, you will have to edit that in the code at line 28:

```
x = mraa.Gpio(23)
```

Now execute the code:

```
$ sudo python blink-io8.py
```

Congratulations! You have executed your first python code using libmraa.

Now that you have tried blinking the LED, tie it to an input and blink it using the touch sensor.

Create a new file named touch_blink.py, and copy the below code into the file:

```
import mraa
print (mraa.getVersion())

led = mraa.Gpio(23)
led.dir(mraa.DIR_OUT)
```

```
led.write(0)

touch = mraa.Gpio(29)
touch.dir(mraa.DIR_IN)

while True:
    touchButton = int(touch.read())
    if(touchButton == 1):
        led.write(1)
    else:
        led.write(0)
```

Now execute the code:

```
$ sudo python touch_blink.py
```

Congratulations! You have used both inputs and outputs using python code and libmraa gpio library.

To try out more projects with GPIO's clone the projects repository with:

```
$ git clone https://github.com/96boards/projects.git
```

Reference : [96Boards projects repository](https://github.com/96boards/projects)

5. Set Up Arduino Tool Chain for DragonBoard 410c

This section will teach you how to access the Arduino toolchain in DragonBoard 410c using a Sensors Mezzanine.

Hardware Required

- [DragonBoard 410c](#)
- [Sensors Mezzanine](#)
- USB to microUSB cable

Step 1: Hardware Setup

There are two ways you can use the Arduino toolchain on the DragonBoard 410c. One is through a serial console using the Sensors Mezzanine and a USB to microUSB cable, and the other is to use an HDMI monitor, keyboard and mouse with the DragonBoard 410c to use the Linux Desktop.

1. Make sure that the DragonBoard 410c is initially unplugged (powered off)
2. Connect the Sensors Mezzanine to DragonBoard 410c through the 40-pin low-speed expansion connector. Sensors Mezzanine board (male pins) will match 1-to-1 with female pins on DragonBoard 410c connector. Make sure all 40 pins are lined up properly before moving to next step.
3. Set up your terminal interface (See “Out of Box” section for headless access method or Linux desktop method).
4. Plug your DragonBoard 410c into power.

Step 2: Software setup

Serial Console Method:

Install the Arduino toolchain using this command:

```
$ sudo apt-get install arduino-mk arduino
```

Linux Desktop on DragonBoard 410c Method:

Use the “Terminal” application to get a command prompt. Install the Arduino toolchain using this command:

```
$ sudo apt-get install arduino-mk arduino
```

You can open the Arduino IDE by:

Application Menu -> Programming -> Arduino IDE

Step 3: Testing a sample code on the toolchain

In this step you will use the existing sample code in order to run a simple application. To do so, first fetch the sample code:

```
$ git clone https://github.com/96boards/Starter_Kit_for_96Boards
```

Based on the method you are using, you can access the sample code by either the serial console and use an editor like vim to edit it, or you can use the Linux Desktop to open up the Arduino IDE there.

Serial Console Method:

```
$ cd Starter_Kit_for_96Boards  
$ cd button_led
```

Add the Arduino.mk Makefile to the same directory:

```
$ ln -s /usr/share/arduino/Arduino.mk Makefile
```

Run the demo Build and execute the program

```
$ make upload reset_stty
```

Linux Desktop on DragonBoard 410c Method:

Open the sample code file in the Arduino IDE. Click on the Verify Button in order to verify and compile the code. Click on the Upload button to upload the code on the DragonBoard 410c.

Congratulations! You are now using the Arduino Toolchain!

IV. More Examples

Check out the **Examples** folder pre-installed on your DragonBoard 410c in your home directory for more sample code for GPIO and OpenCV.

V. Support and Troubleshooting

DragonBoard 410c support forum

<http://www.96boards.org/forums/forum/products/dragonboard410c/>

Real time chat with Linaro and 96Boards developers - channels #96Boards #OpenHours

<https://webchat.freenode.net>

<https://www.irccloud.com>

Documentation and software in Github

<https://github.com/96boards/documentation/tree/master/ConsumerEdition/DragonBoard-410c>