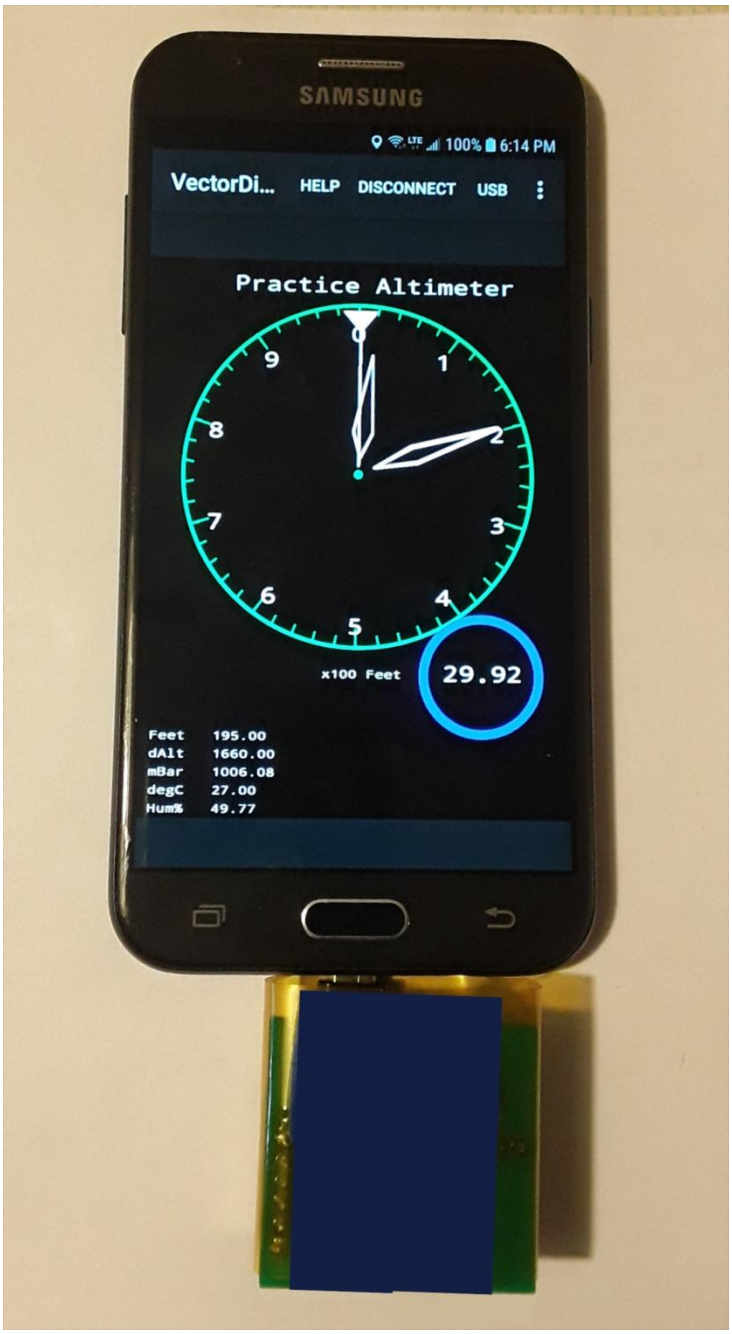
Instructions for Android Altimeter

https://github.com/kenburrell/AndroidAltimeter



Contents

1. Software Installation

If you obtained the parts for the Android Altimeter from Thorp Aviation (Las Vegas, NV) as a kit, you had the option to obtain it with the Altimeter software pre-installed to the Arduino Leonardo.

If the Leonardo has the Altimeter software pre-installed, skip to the **Assembly** instructions.

If the Leonardo did not come with the Altimeter software pre-installed, or if you purchased all of the parts of the Android Altimeter yourself, as listed in the Bill of Materials, then you will need to install the Altimeter software to the Leonardo before assembling the Android Altimeter.

Step 1:

Download all files from the GitHub project page for the Android Altimeter, and save it in your Downloads folder as a ZIP file:

https://github.com/kenburrell/AndroidAltimeter/archive/master.zip

Step 2:

Extract all the files from the ZIP archive into a new folder. The default name of the new folder will be AndroidAltimeter-master, and will be located in your Downloads folder. You will need to access the files in this folder in a later step.

Step 3:

Download ***and install*** the Arudino IDE from the following website:

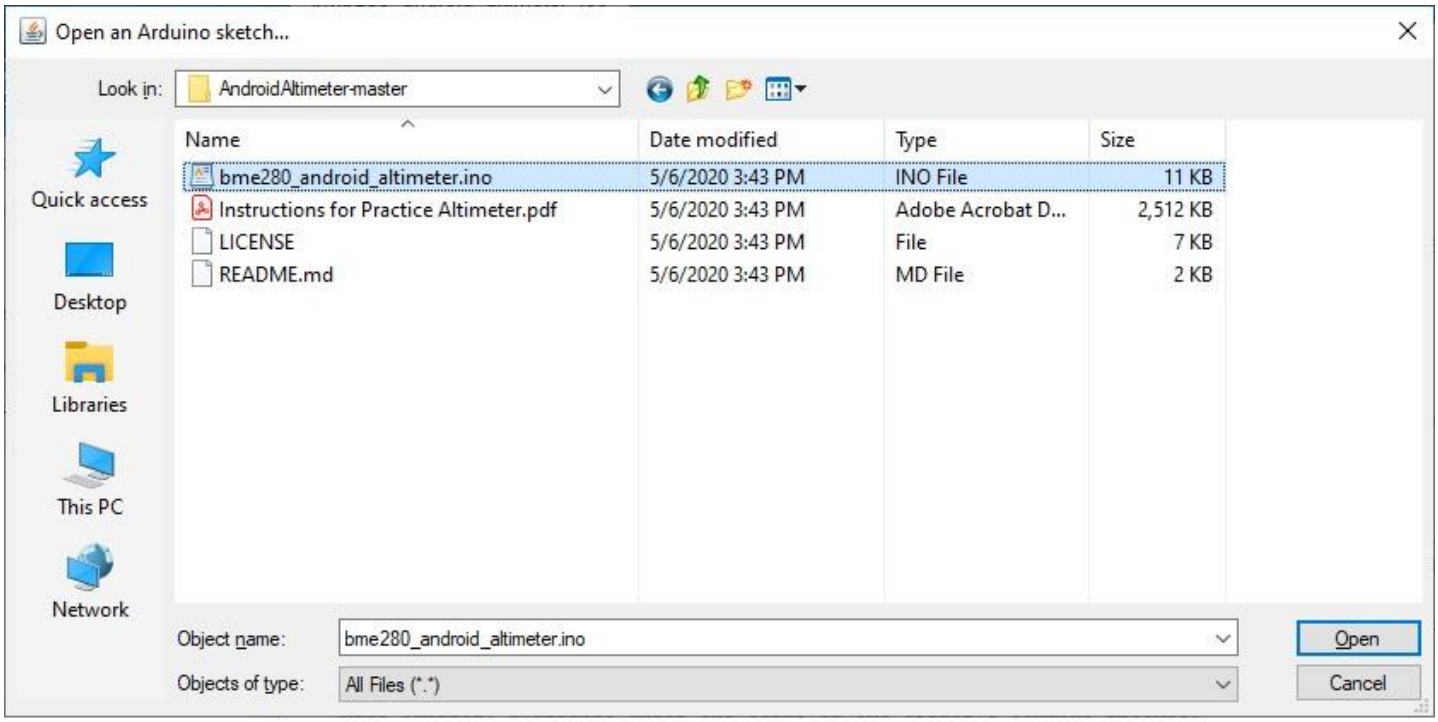
https://www.arduino.cc/en/Main/Software

Step 4:

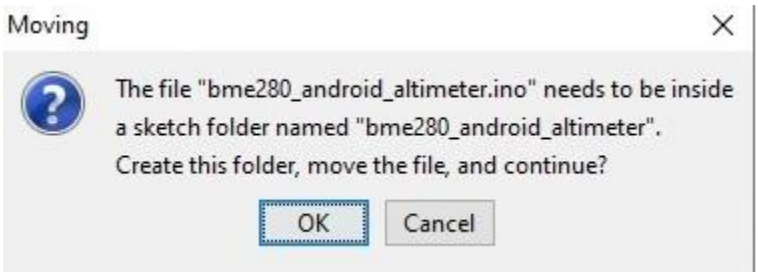
Launch the Arduino IDE from the icon that was installed on your desktop. If you installed on a Windows system, it will look like this:

Step 5:

Go to the “File” menu in the Arduino IDE, and select the “Open” option. Navigate to the folder where you extracted all of the Android Altimeter files, and select the file “bme280_android_altimeter.ino”. Click on “Open” to load that file.

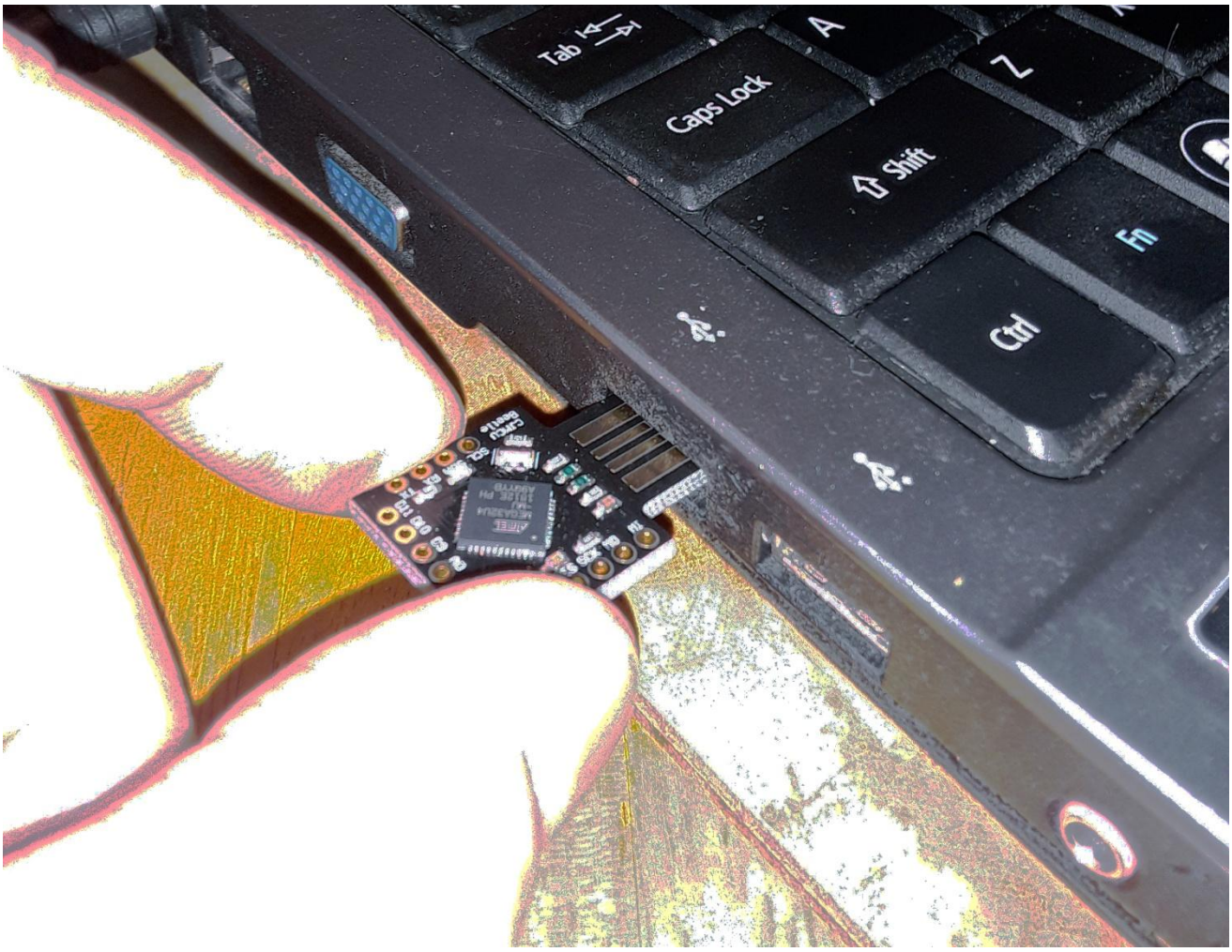


If you are prompted to create a new folder, select OK and continue.



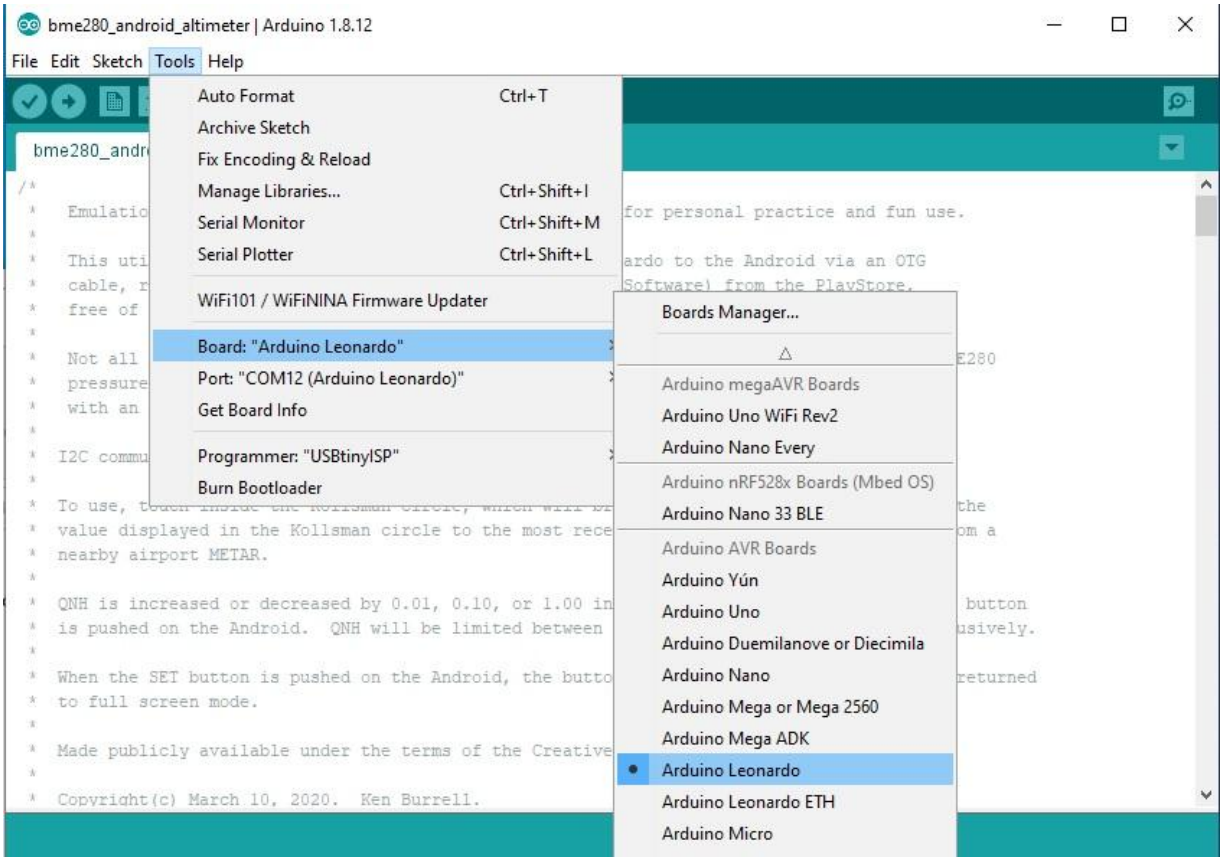
Step 6:

Insert the Arduino Leonardo board into an available standard USB port on your computer. Make sure that the gold fingers of the Leonardo are showing as in the photo below.



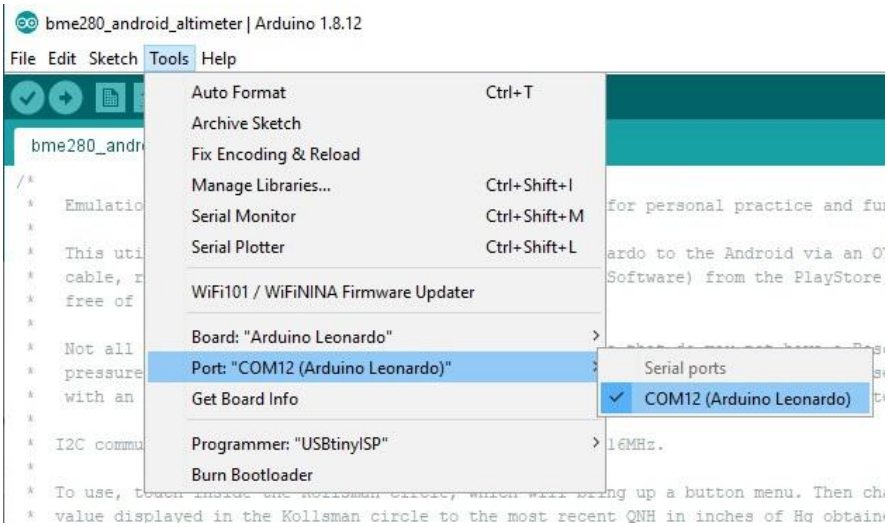
Step 7:

In the Arduino IDE go to the “Tools” menu, and select “Arduino Leonardo” from the “Board:” menu.



Step 8:

In the Arduino IDE go again to the “Tools” menu, and select the COM port from the “Serial Port” menu that is attached to your Leonardo. A check mark shows when it is selected.

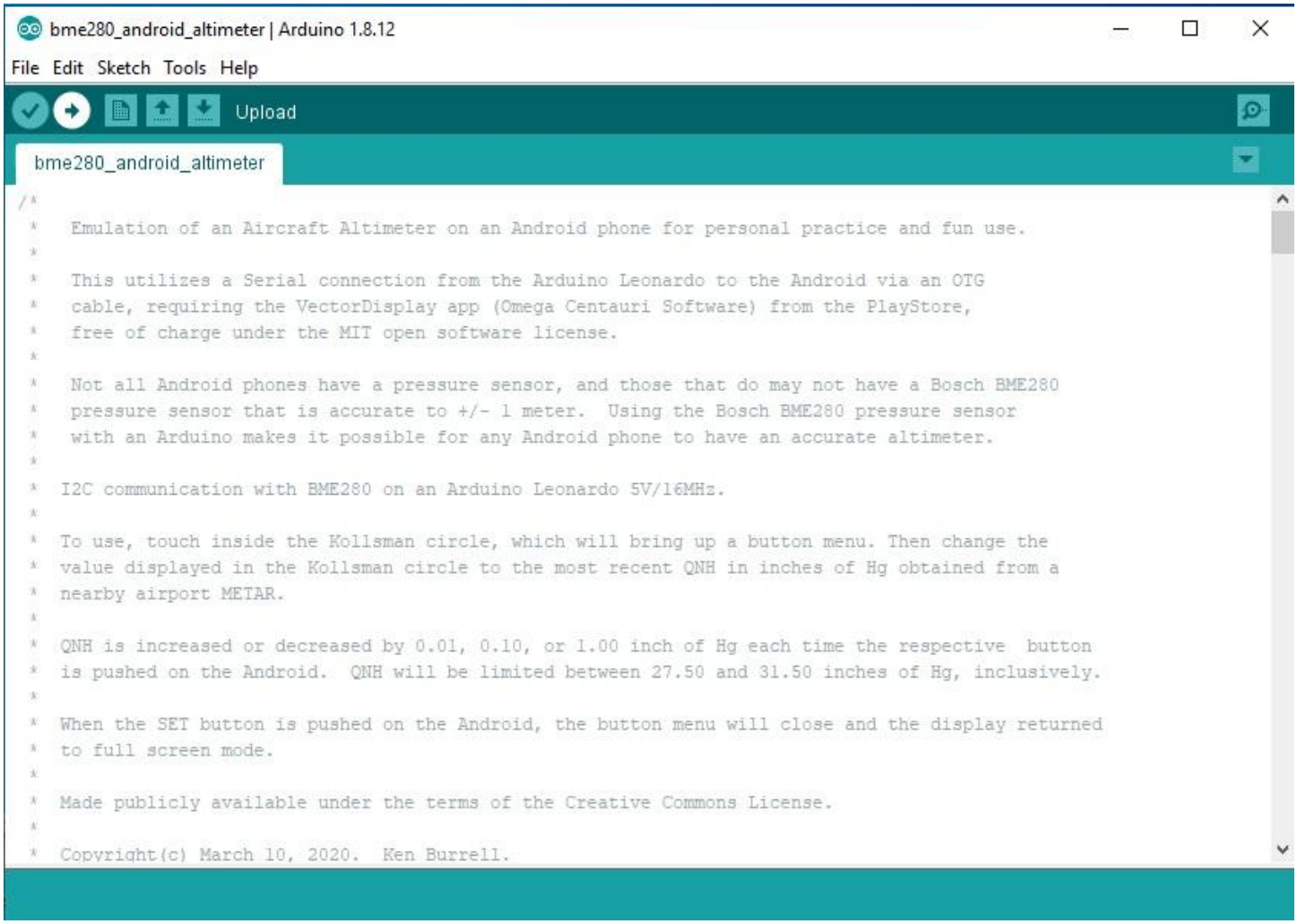


Step 9:

Compile and install the software to the Leonardo by clicking the right-arrow at the upper left of the Arduino IDE, as shown below. When selected it will be highlighted and the word “Upload” will appear in the upper status bar, as shown below.

Click on the right-arrow to begin the compilation process. While it is compiling, and before it begins uploading, the word “compiling” will show in the lower status bar.

Please be patient during the compile phase. This could take a few minutes.



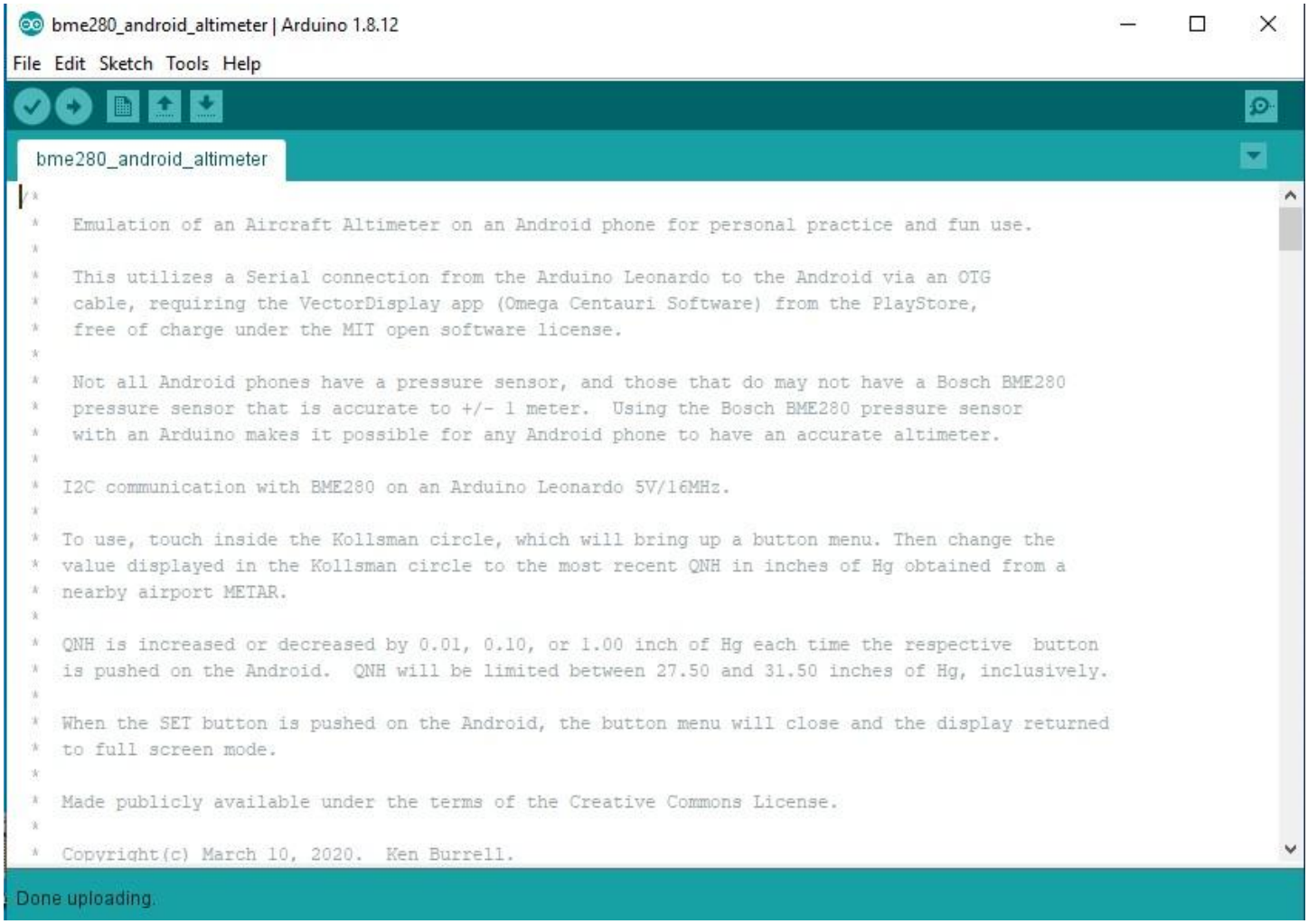
Step 10:

When the compile phase finishes, the upload to the Leonard will begin and you should see a red blinking light on the Leonardo board during the upload.

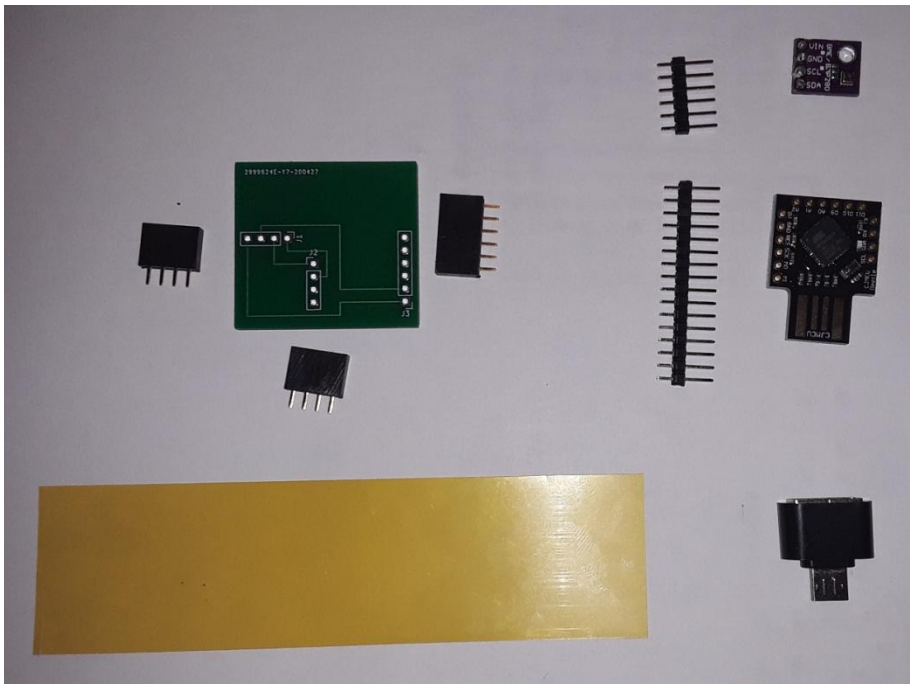
When the upload is complete, it will say “Done uploading” as shown below.

You should now remove the Leonardo from the USB port, and set it aside for the Assembly step in the next section.

You may also now close Arduino IDE.



2. Assembly



Parts required for each kit:

- 1x BME280 Sensor with 4 small mounting holes and 1 row 6-pin male header
- 1x custom PCB board with 2 sets of 4 mounting holes and 1 set of 6 mounting holes
  You can obtain this custom PCB board as part of a kit from Thorp Aviation (Las Vegas, NV) or, generate your own from the Gerber files found on my GitHub project page: https://github.com/kenburrell/AndroidAltimeter
- 1x 6-pin female header and 2x 4-pin female headers
- 1x Arduino Leonardo Board in the Beetle form factor and 1 row 16-pin male header
- 1x female USB-A to male Micro-B OTG adapter
- 1x sheet of yellow Kapton tape, 1.20” x 4.55”
- Display software for Android phone (download instructions below)
- Altimeter software for the Leonardo (installation instructions above, in Software)

Additional parts required:

- A couple inches of electrical tape to seal Kapton tape around circuit board
- A soldering iron and preferably 60/40 Sn/Pb flux core solder, 23 gauge
- A multi-meter with a continuity check, or an ohm meter
- A small cutting tool, or scissors for cutting headers

Recommended parts, but optional:

- A Helping Hand Tool, as pictured on the next page

Step 1:

Insert the 3 female headers (1x 6-pin, 2x 4-pin) into the top of the circuit board so that the female headers are on top and the short pins go through the holes, as pictured below. Note that the top of the circuit board is the side with printing. The backside is blank.

Make sure that each header is fully and squarely seated into the holes as shown below.

Step 2:

If Helping Hands were used as shown above, now remove the circuit board from the Helping Hands, being careful to keep all 3 headers fully seated in their holes.

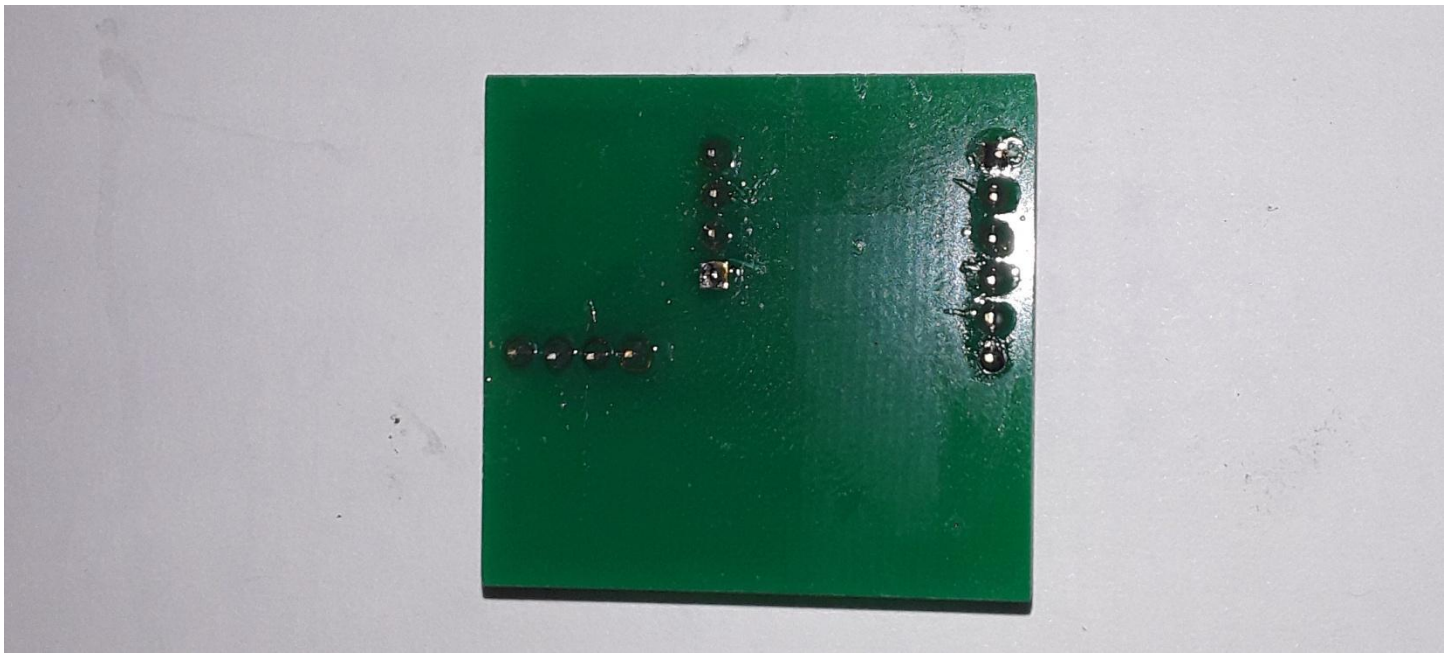
Some pieces of tape may be used temporarily to keep all 3 headers squarely seated.

Now flip the board over to the backside, being careful to keep all headers in place, and place the board down on a level surface as show below.

Orient the board so that the 6-pin row is to the right and the two 4-pin rows are on the left, as shown below.

Then solder the pins to the backside of the board to make sure that all headers are held in place, as shown below.

Periodically check the front side of the board to make sure all female headers are fully inserted and perpendicular to the board.

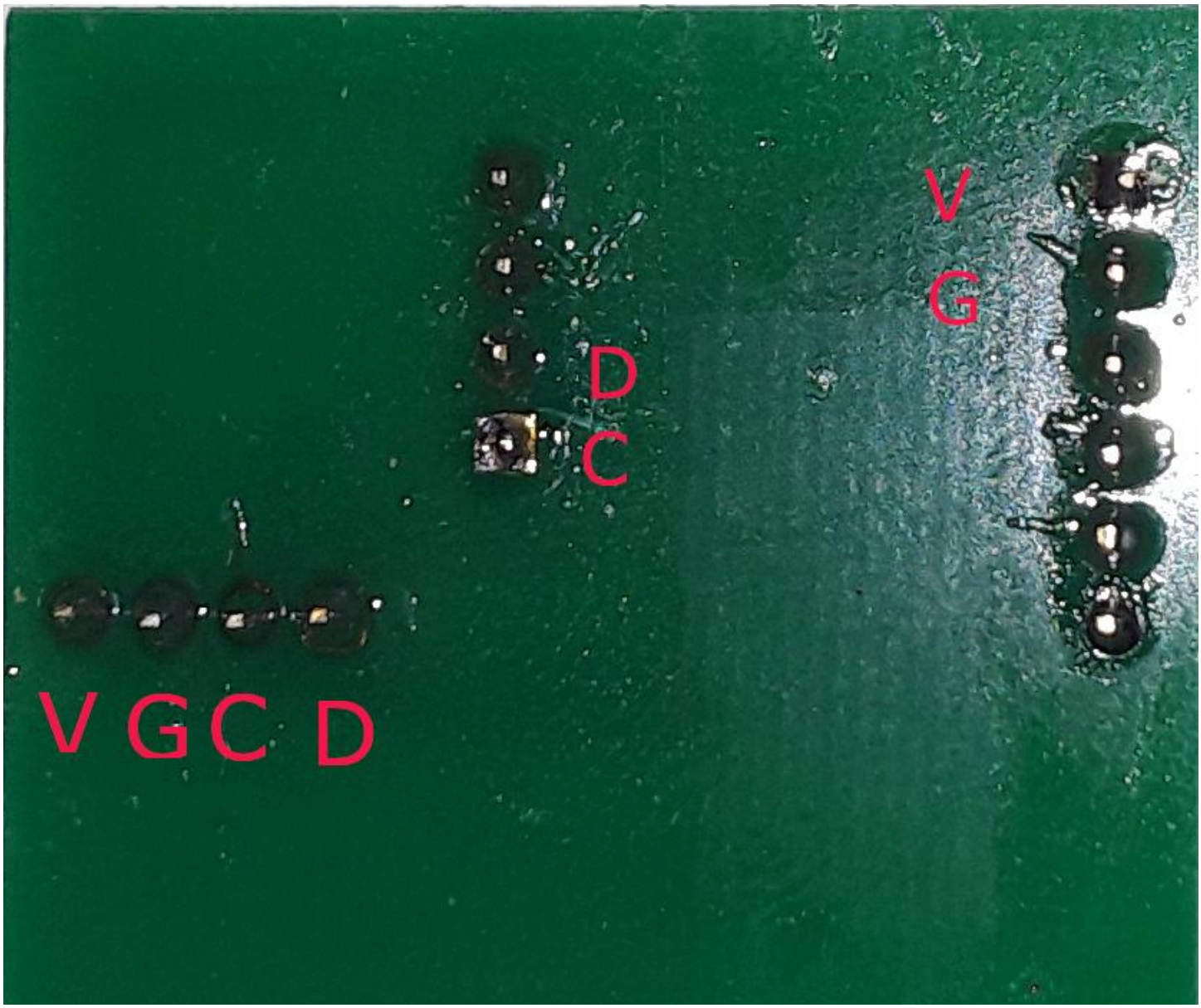


Step 3:

Use a multi-meter to check that none of the 3 headers have a pin that has been short-circuited to its neighbor.

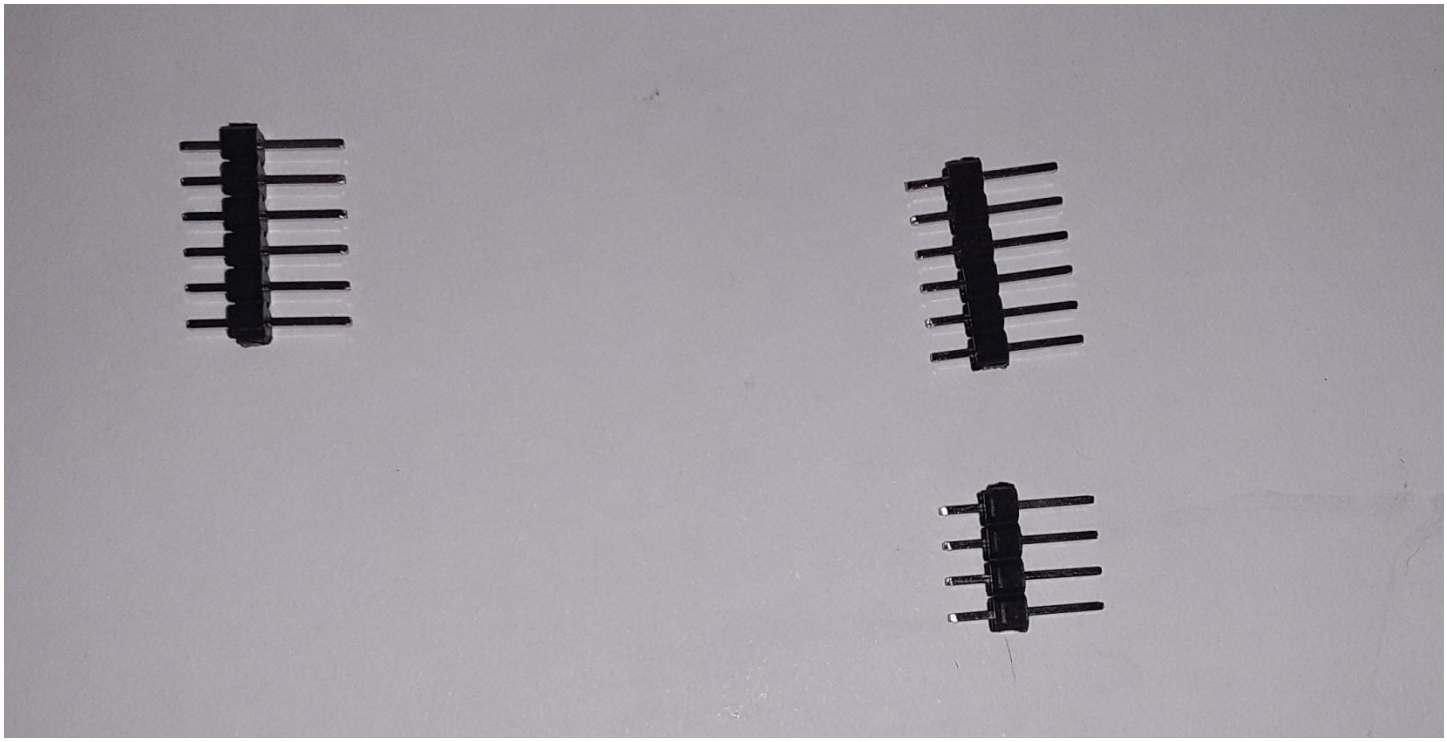
Using a multi-meter, verify that there is a connection between the following 4 pairs of pins on the back side of the board as pictured below:

- continuity between V on the lower 4-pin row and V on the 6-pin row
- continuity between G on the lower 4-pin row and G on the 6-pin row
- continuity between C on the lower 4-pin row and C on the other 4-pin row
- continuity between D on the lower 4-pin row and D on the other 4-pin row



Step 4:

Now carefully cut away from the male header of 16-pins, to create one row of 6-pins and another row of 4-pins, as shown below. You will need a small cutting tool or pair of scissors.



Step 5:

Now orient the green PCB (printed circuit board) board so that the 6-pin female header is on the left.

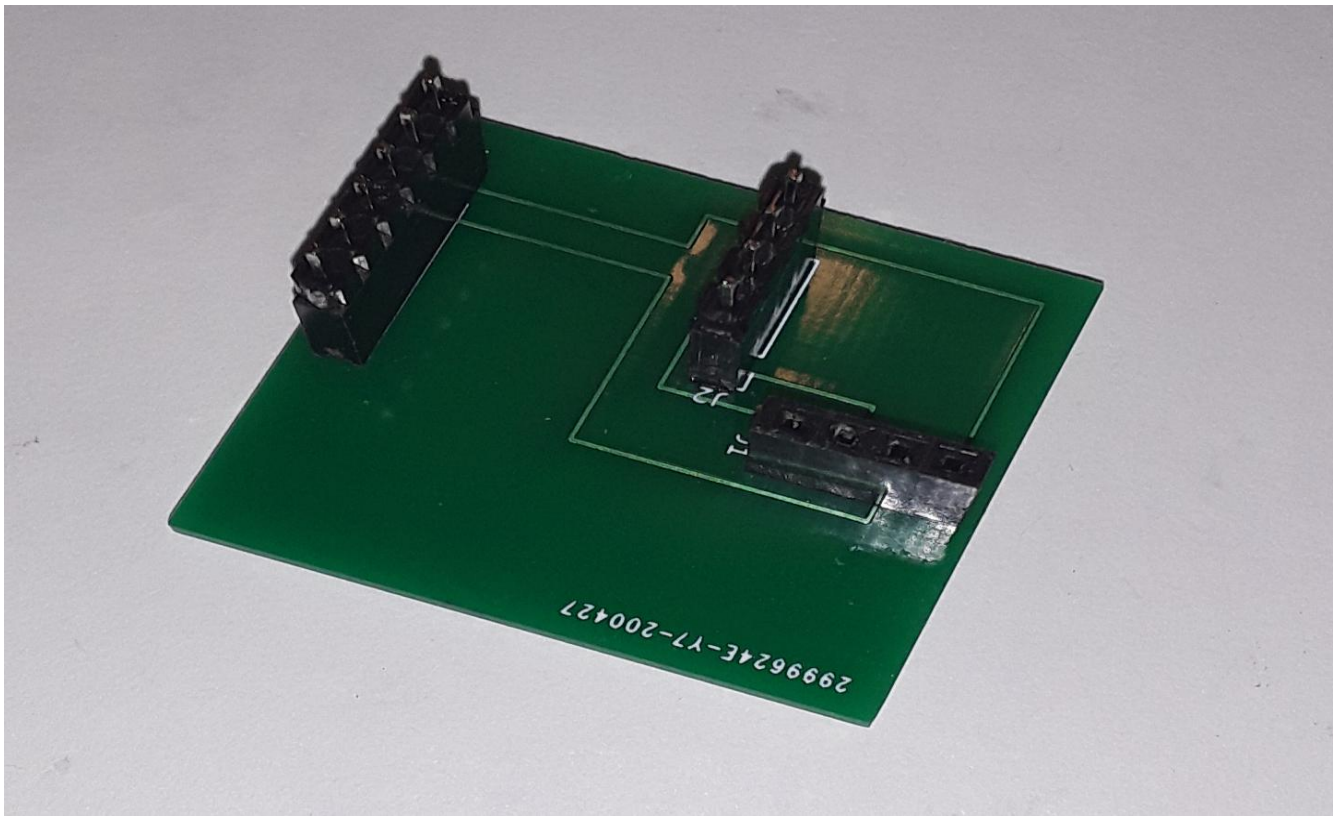
Insert the 6-pin male header that you created in **Step4** into the 6-pin female header on the PCB.

Insert the 4-pin male header that you created in **Step4** into the 4-pin female header in the center of the board (it runs parallel to the 6-pin header).

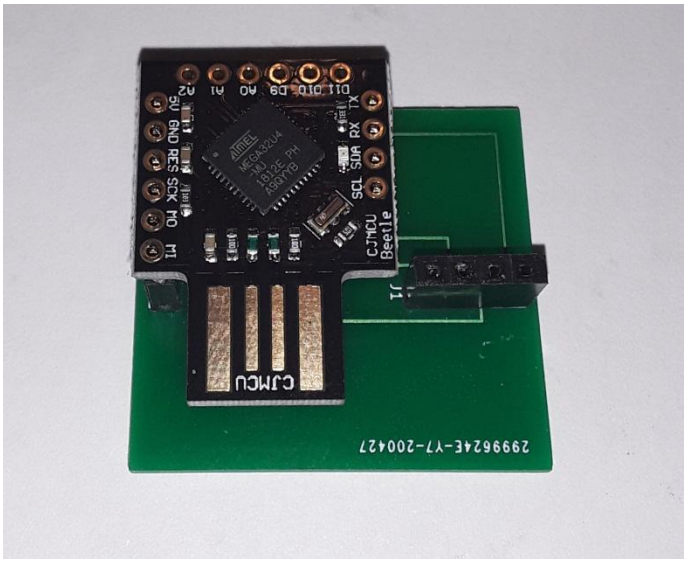
Make certain that ***the long pins*** are inserted into the female headers. Do not insert the short pins into the female headers.

Make certain all the pins are inserted into the female headers, and that none are "hanging out" due to having offset the placement of the male pins.

Everything should be as pictured below.



Now place the Leonardo board on top of the male header pins, so that they poke through the holes in the Leonardo, as shown below.

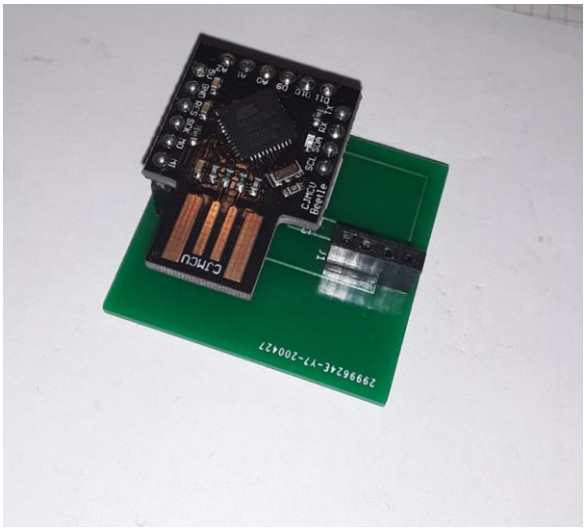


Solder the 6-pin header and 4-pin header, on the topside of the Leonardo board, while checking periodically to make sure that the header pins are fully seated in the Leonardo holes.

Step 6:

Cut off a 4-pin male header from the 6-pin header that accompanied the BME280 sensor, and insert the 4-pin male header into the remaining 4-pin female header of the PCB as shown below.

Make certain that the long pins go into the female header and that the short pins are on the top.

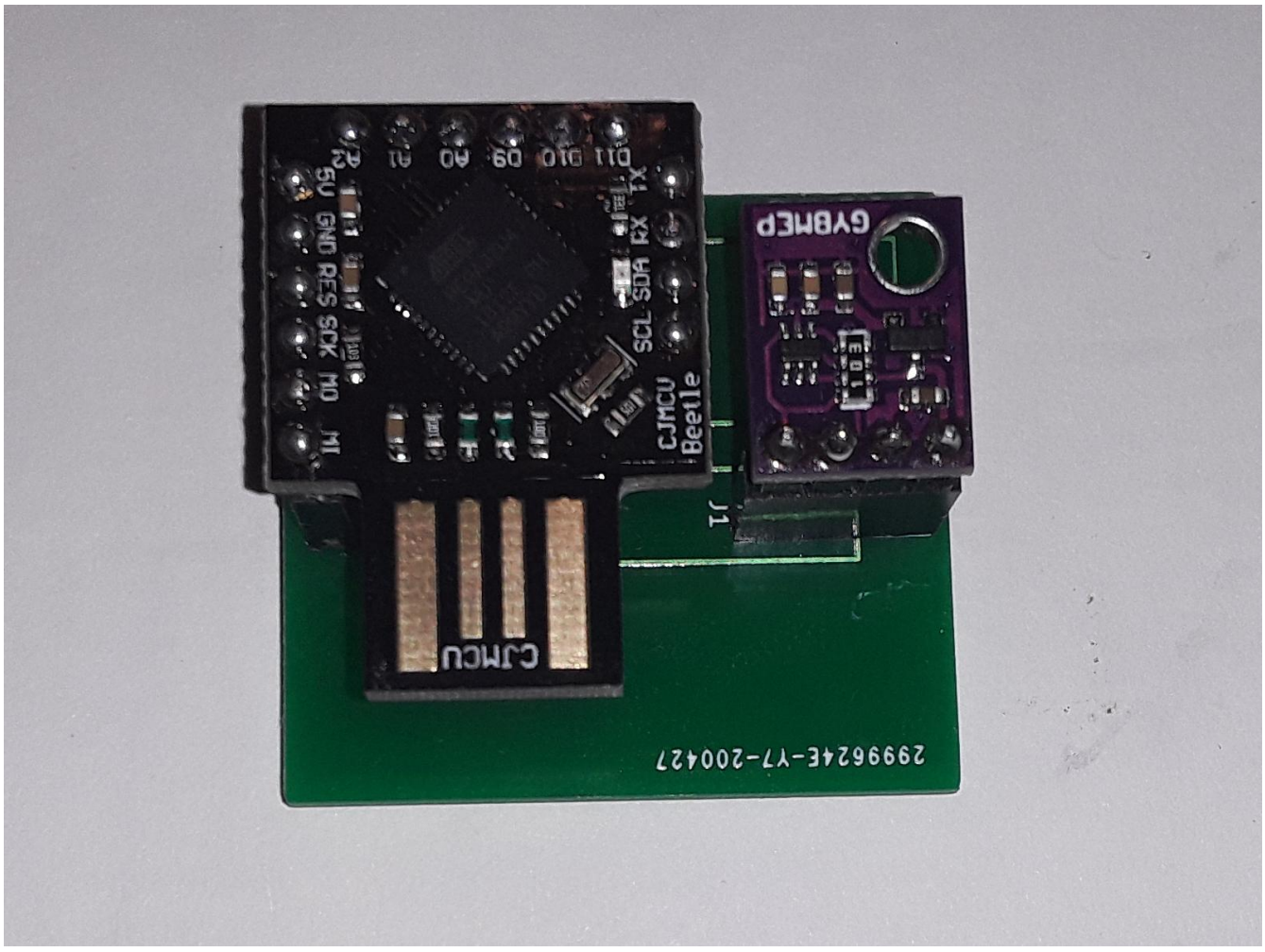


Place the BME280 face down onto the 4 male pins sticking up out of the female header, so that the BME280 hole that is labeled **VIN** is on the underside of the BME280 board, and is on top of the right-most pin. Make sure all 4 pins poke through the 4 holes of the BME280 board.

Note that the large hole in the BME280 sensor board will be above the 4-pin header and to the right, near the edge of the board, when it is oriented correctly, as show below.

Now carefully solder the 4-pins. ***<u>Exercise great care</u>***, as spilling solder anywhere other than on the pins inside the 4 holes will damage the sensor and make it unusable. Keep the BME280 board level while soldering.

<u>Failure to properly orient the board will destroy the sensor</u>, so make sure it is oriented properly as shown below before soldering the 4 pins holding the sensor.

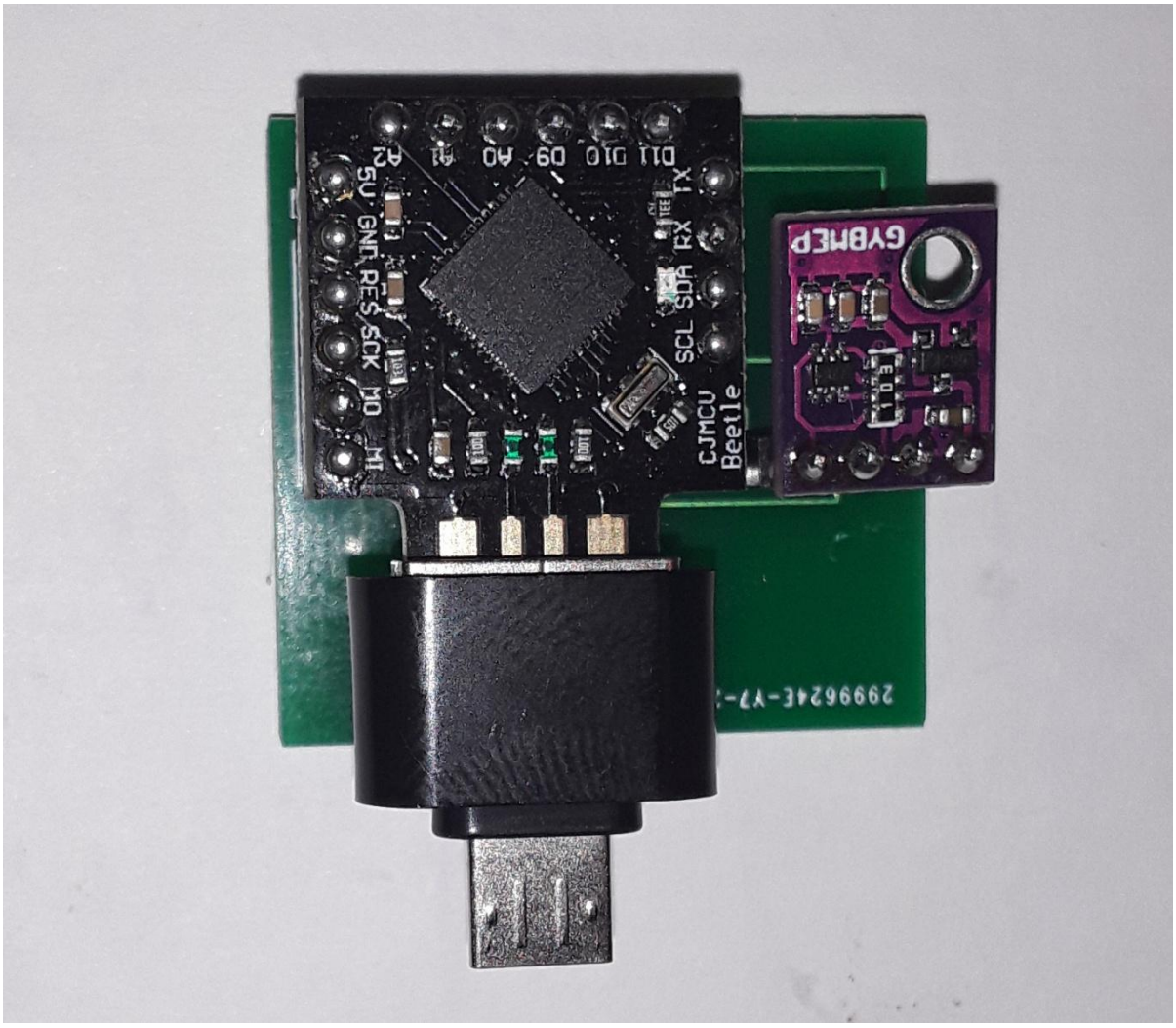


When soldering is complete, the board shold appear as shown above.

Step 7:

Now it is time to attach the OTG adapter, as pictured below. The correct orientation of the OTG adapter is such that the wide end of the micro-B is on the top. That will make certain that the gold “fingers” of the USB-A jack of the Leonardo make contact with the internal wires of the USB-A to Micro-B OTG adapter.

Note that an ordinary USB adapter between USB-A and Micro-B will not work, because OTG functionality is require for the Android phone to control the software running on the Leonardo.



Step 8:

Now wrap the PCB with the provided Kapton tape, and secure the edges of the tape where they meet on the underside of the circuit board.

The assembly of the Android Altimeter is now complete, and should look like this:

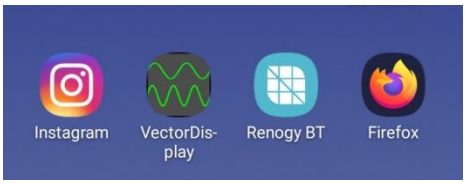
3. Operation

Before connecting the Android Altimeter to the phone, the display software must be installed from the Play Store.

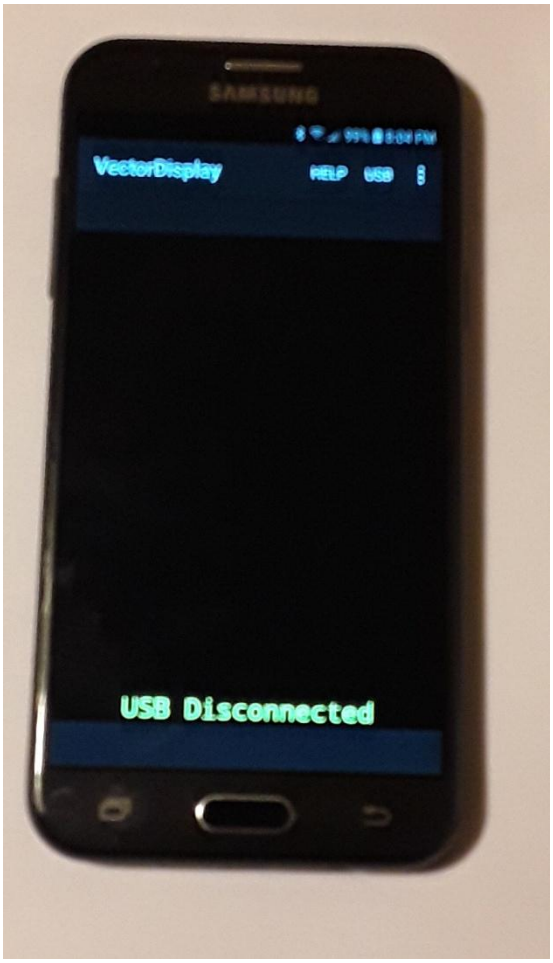
Go to the Play Store on the Android phone. Under Apps, search for “VectorDisplay”.

If more than one App appears, select the one made by “Omega Centauri Software”.

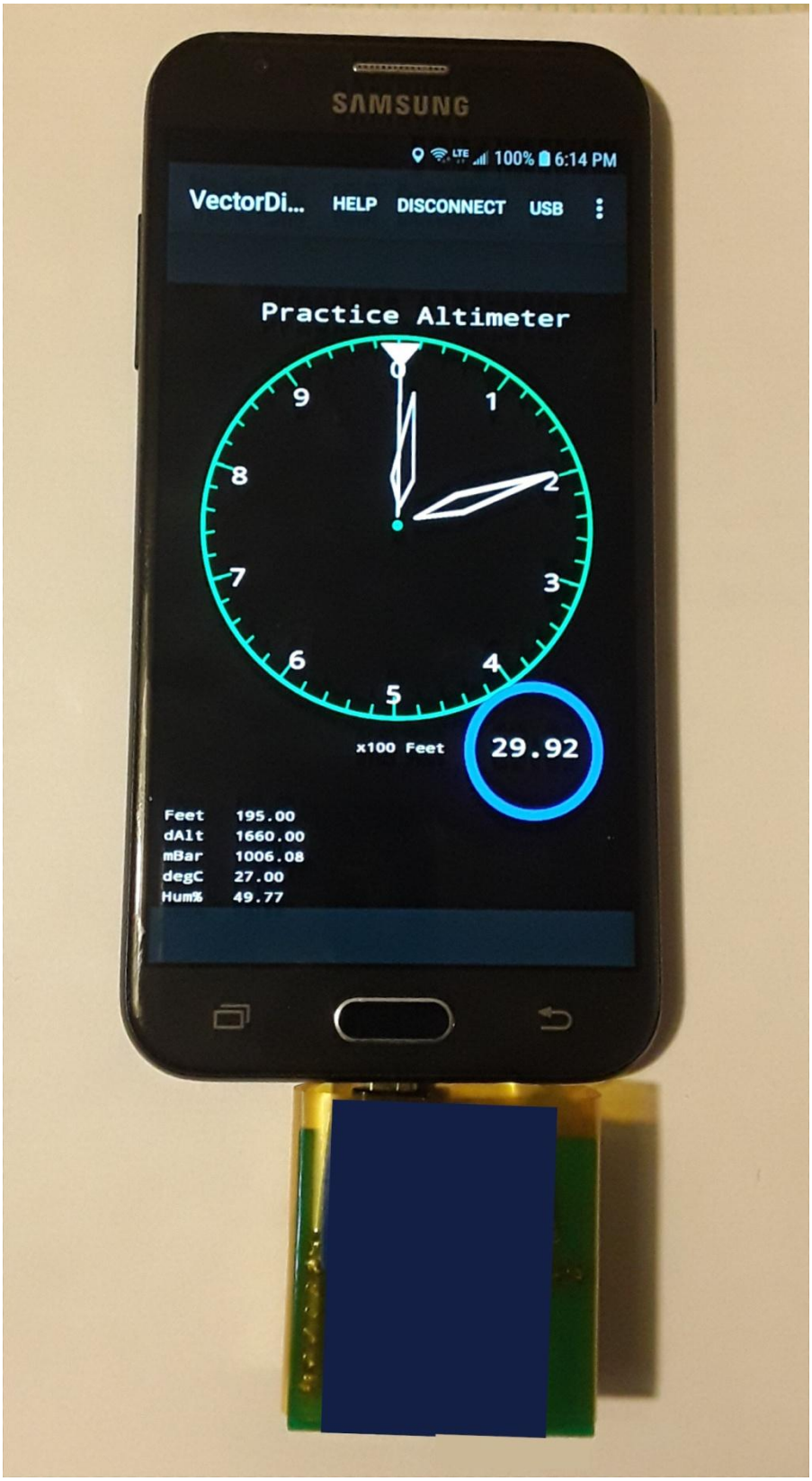
Install VectorDisplay to your Android phone. It should appear as an icon like this:



Startup the VectorDisplay App, and it will say “USB Disconnected”:



Plug the Android Altimeter into the phone USB jack, and it will prompt you to allow the connection. Select “OK”, and the following screen will appear. Note that the board in this case is upside down, due to the orientation of the phone’s micro-USB jack.



Also displayed in the lower left hand corner are digital altitude, density altitude, temperature in Centigrade, and percent humidity. Note that the default setting is for an equivalent sea level pressure of 29.92 inches of Hg. That can be changed by entering the menu.

Touch the inner circle of the display, and it will bring up the menu as shown below.

By obtaining a local METAR or AWOS, the Barometer setting can be obtained and entered via the menu keys, which can increment or decrement by 0.01, 0.10, and 1.00 inches of Hg. Pressing the SET key will update the display with the corrected altitude for the given Barometer setting, and return to the full screen display.

Note that the Barometer setting is restricted to 27.50 to 31.50 inches of Hg, inclusively, as in most aviation altimeters.

4. Bill of Materials

for 10 Kits

- BME280 Sensor, Amazon #**B07KR24P6P,** 12 units $ 53.97
- Arduino Leonardo Beetle, Amazon #**B011AYAOHS,** 10 units $ 93.00
- USB-A / Micro-B OTG adapter, Amazon #**B07CL6GBTT**, 10 $ 5.99
- JLCPCB custom PCB, 10 units $ 22.05
- Kapton tape, Fry's #**8220976**, 15 units $ 3.99
- PCB Female Pin Headers, Amazon #**B07Q1XBGFB**, 10 units $ 9.99

+Tax $ 12.94
==========

Total $201.84

Average bare cost per kit: 201.84 / 10 = $20.18 .

The break even cost should be about $22 for each kit, including postage and packaging, if 10 kits are made from the above bulk purchase.

There will be a few leftover parts after 10 kits, but not enough to build a whole kit, unless an even larger purchase is made.

It is possible to reduce the cost even more by purchasing cheap Leonardo Beetles from China in groups of 10, but that would reduce the average cost by only about $2 due to the shipping costs from China, whereas the Amazon shipping is included in the above. Plus, shipping from China normally takes 6 weeks, and Amazon can deliver within a few days.