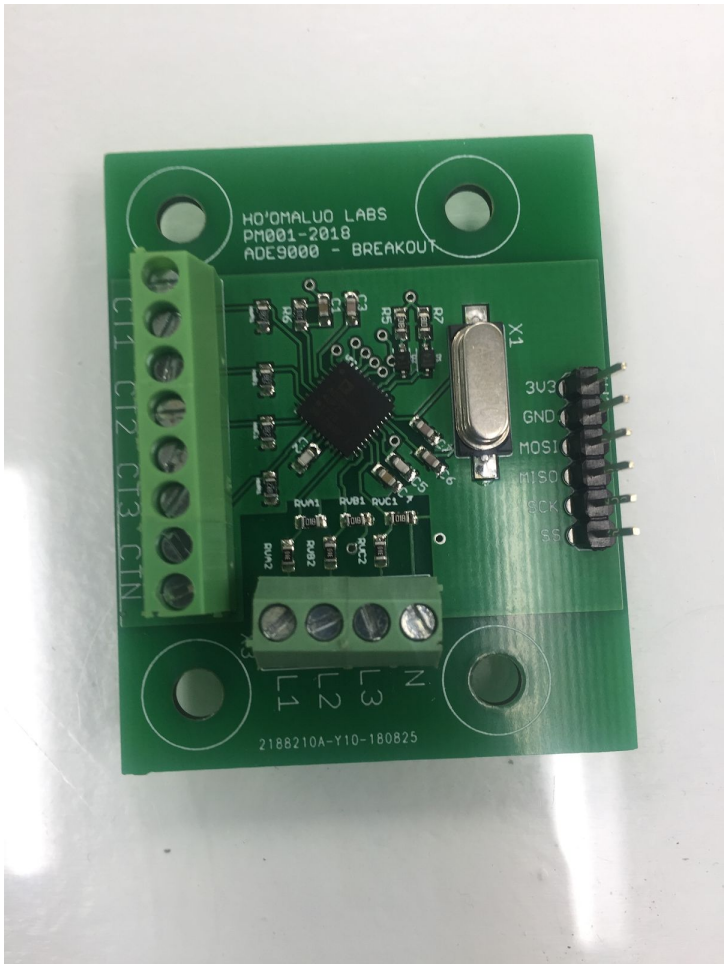
PM001-ADE9000 BREAKOUT ARDUINO (MAPLE MINI) TUTORIAL

SINGLE OR 3 PHASE POWER MONITORING IC BREAKOUT BOARD

By: Matsu



PRODUCT INFORMATION

This breakout board uses the ADE9000 power monitoring chip which uses SPI interface. This board is compatible with 32-bit microcontrollers such as the Maple Mini, ESP8266, and the

Arduino Zero or Due. This particular tutorial will use a Maple Mini as the primary microcontroller, but you should be able to adapt it to any of these other 32-bit microcontrollers using a similar approach.

DESCRIPTION OF PRODUCT FROM ANALOG DEVICES

The ADE9000 is a highly accurate, fully integrated, multiphase energy and power quality monitoring device. Superior analog performance and a digital signal processing (DSP) core enable accurate energy monitoring over a wide dynamic range. An integrated high end reference ensures low drift over temperature with a combined drift of less than ±25 ppm/°C maximum for the entire channel including a programmable gain amplifier (PGA) and an analog-to-digital converter (ADC).

The ADE9000 offers complete power monitoring capability by providing total as well as fundamental measurements on rms, active, reactive, and apparent powers and energies. Advanced features such as dip and swell monitoring, frequency, phase angle, voltage total harmonic distortion (VTHD), current total harmonic distortion (ITHD), and power factor measurements enable implementation of power quality measurements. The ½ cycle rms and 10 cycle rms/12 cycle rms, calculated according to IEC 61000-4-30 Class S, provide instantaneous rms measurements for real-time monitoring.

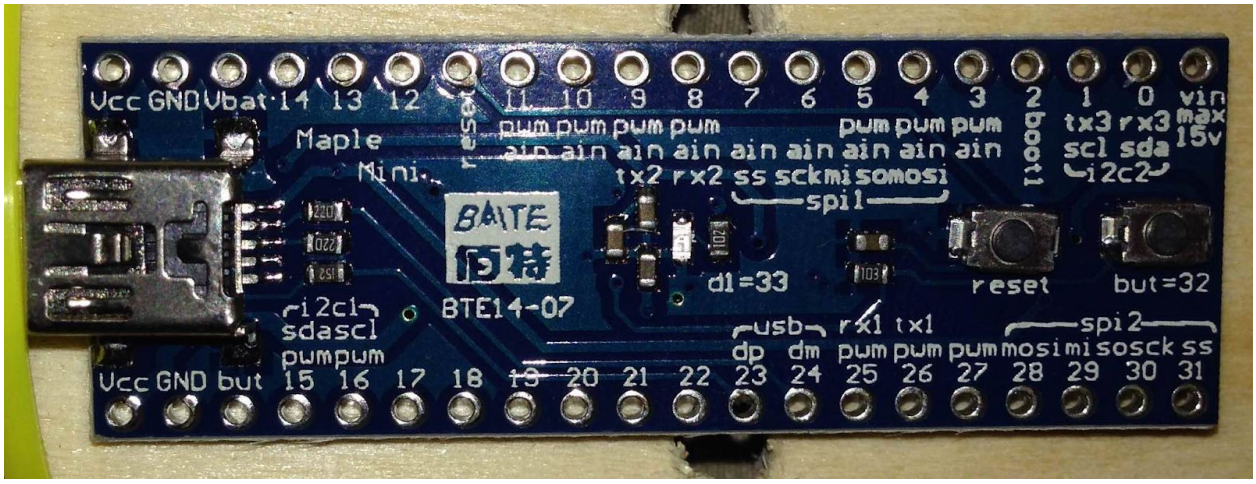
The ADE9000 offers an integrated flexible waveform buffer that stores samples at a fixed data rate of 32 kSPS or 8 kSPS, or a sampling rate that varies based on line frequency to ensure 128 points per line cycle. Resampling simplifies fast Fourier transform (FFT) calculation of at least 50 harmonics in an external processor.

The ADE9000 simplifies the implementation of energy and power quality monitoring systems by providing tight integration of acquisition and calculation engines. The integrated ADCs and DSP engine calculate various parameters and provide data through user accessible registers or indicate events through interrupt pins. With seven dedicated ADC channels, the ADE9000 can be used on a 3-phase system or up to three single-phase systems. It supports current transformers (CTs) or Rogowski coils for current measurements. A digital integrator eliminates a discrete integrator required for Rogowski coils.

The ADE9000 absorbs most complexity in calculations for a power monitoring system. With a simple host microcontroller, the ADE9000 enables the design of standalone monitoring or protection systems, or low cost nodes uploading data into the cloud.

MAPLE MINI

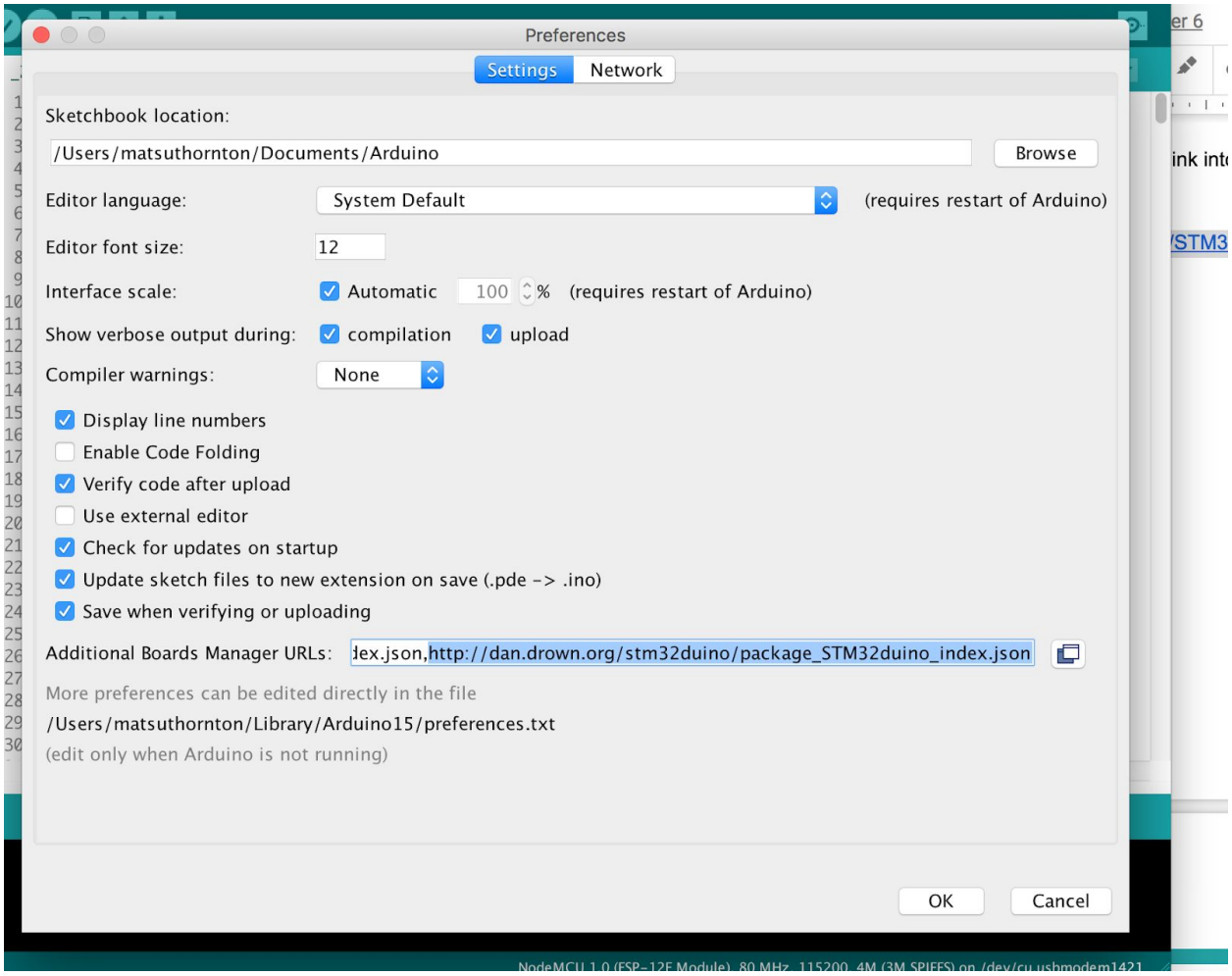
For this tutorial we will be using the Maple Mini microcontroller from Leaf Labs which is a 32-bit microcontroller. Please note that the Arduino UNO and many other boards are 8-bit microcontrollers and will not work using this tutorial, however, there are many 32-bit boards that will work with this board such as the Arduino Zero, Arduino Due, and ESP8266.



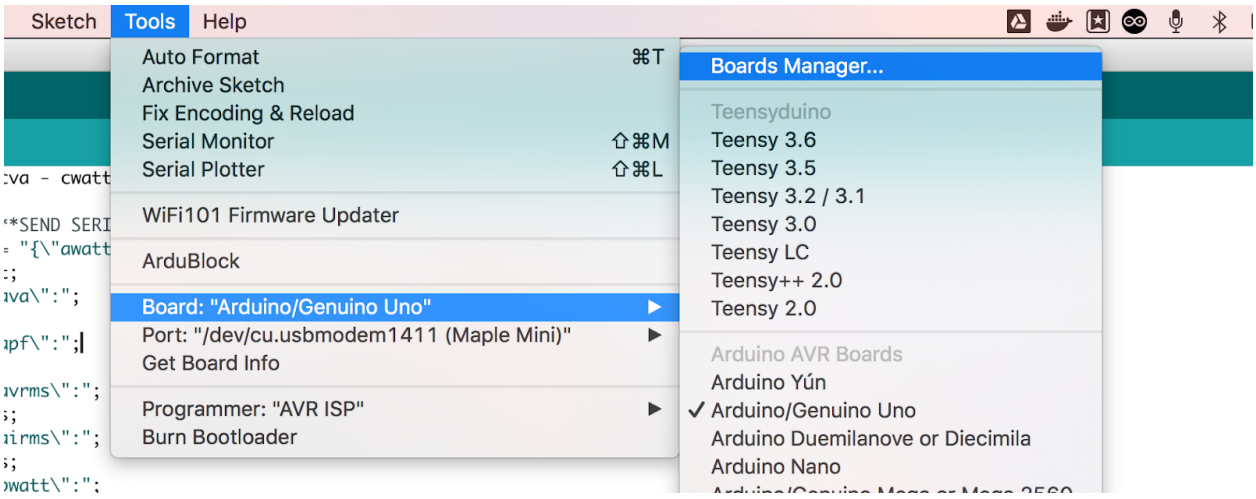
INSTALLING BOARDS

Open Arduino IDE and go to the preferences pane. Paste this link into the additional boards manager URLs:

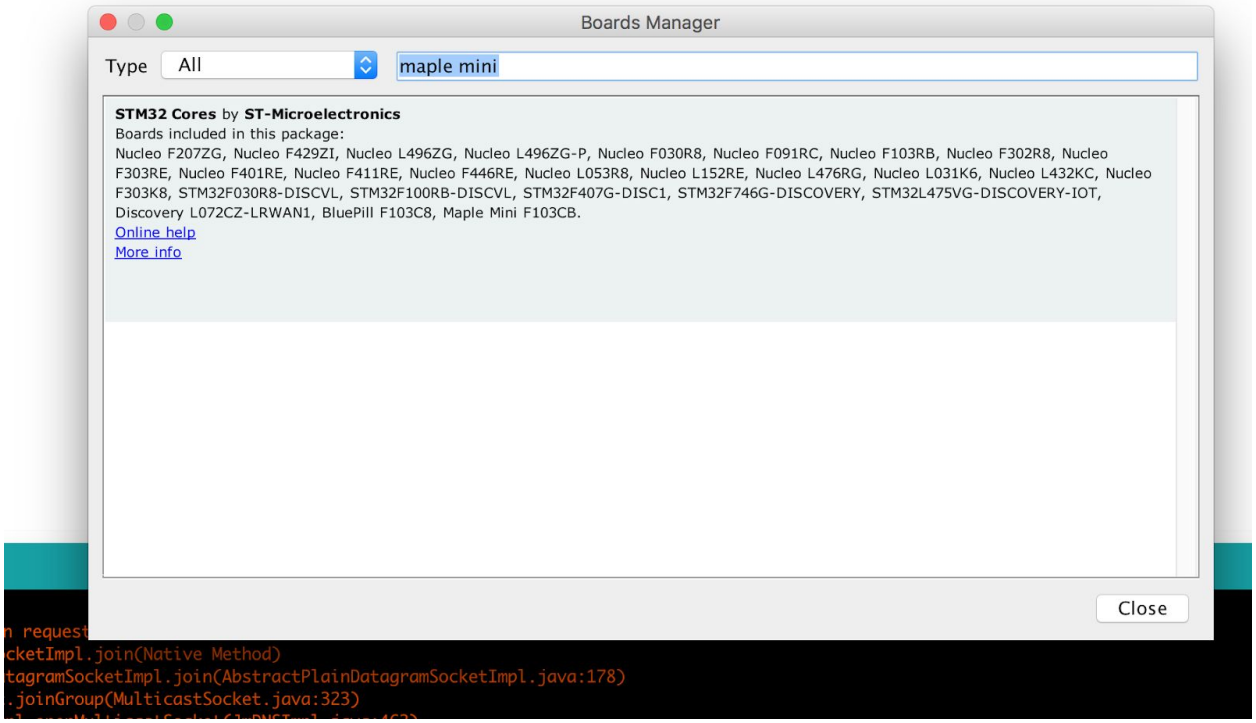
http://dan.drown.org/stm32duino/package_STM32duino_index.json



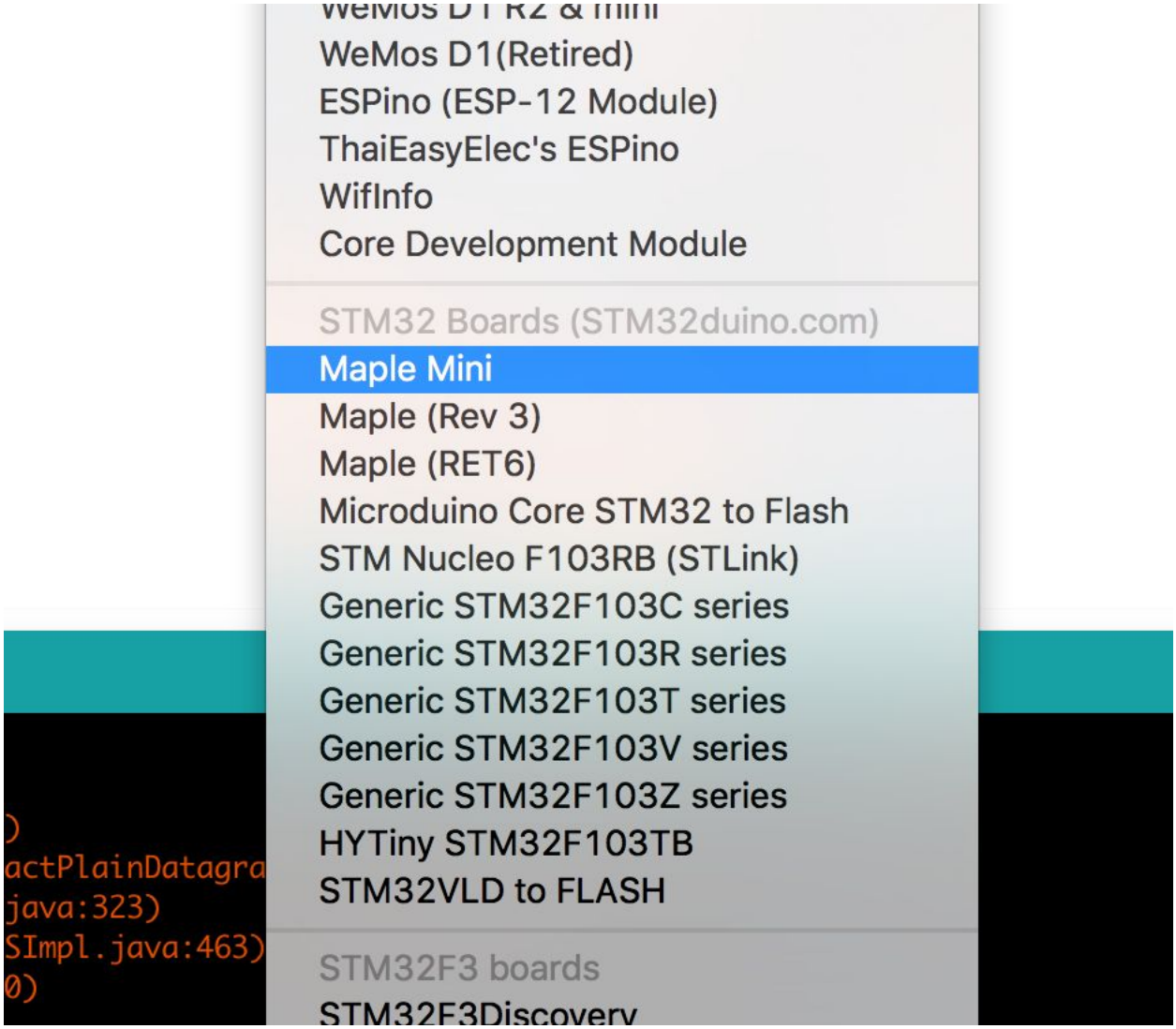
Click ok and go to the boards manager under the tools->Board menu:



Search for the maple mini board:



Install the STM32 cores and close the boards manager. You should now be able to see the maple mini in the available boards. Select the maple mini:

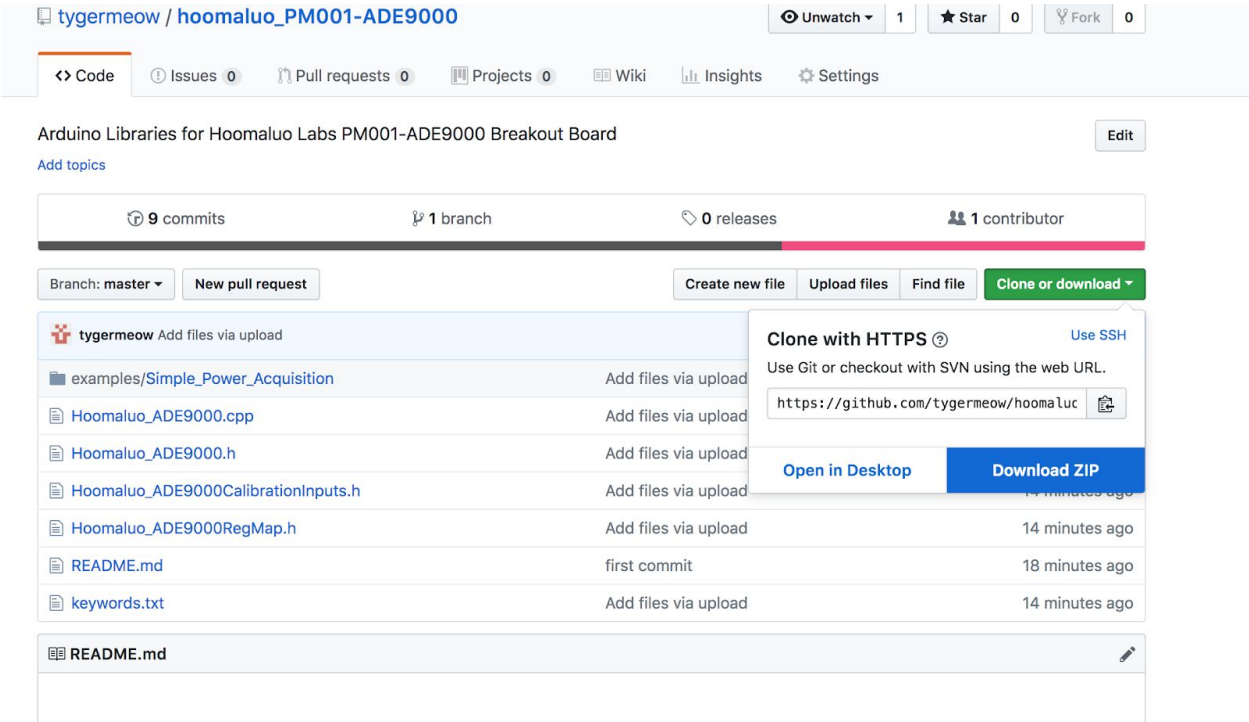


INSTALLING LIBRARIES

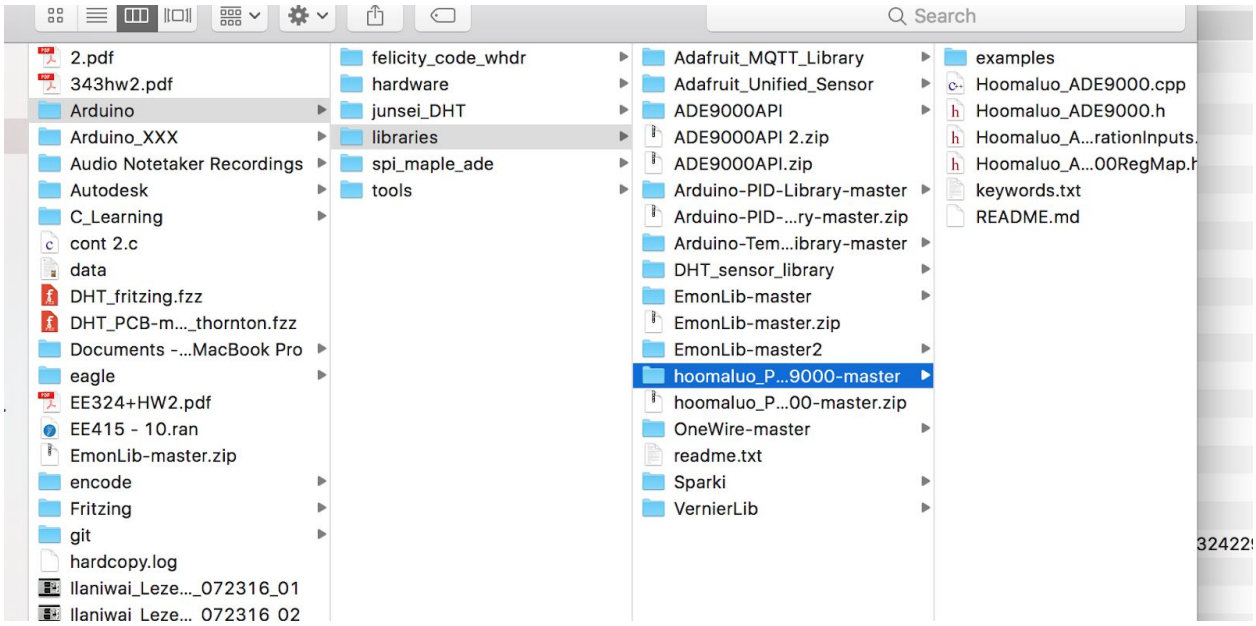
Navigate to:

https://github.com/tygermeow/hoomaluo_PM001-ADE9000

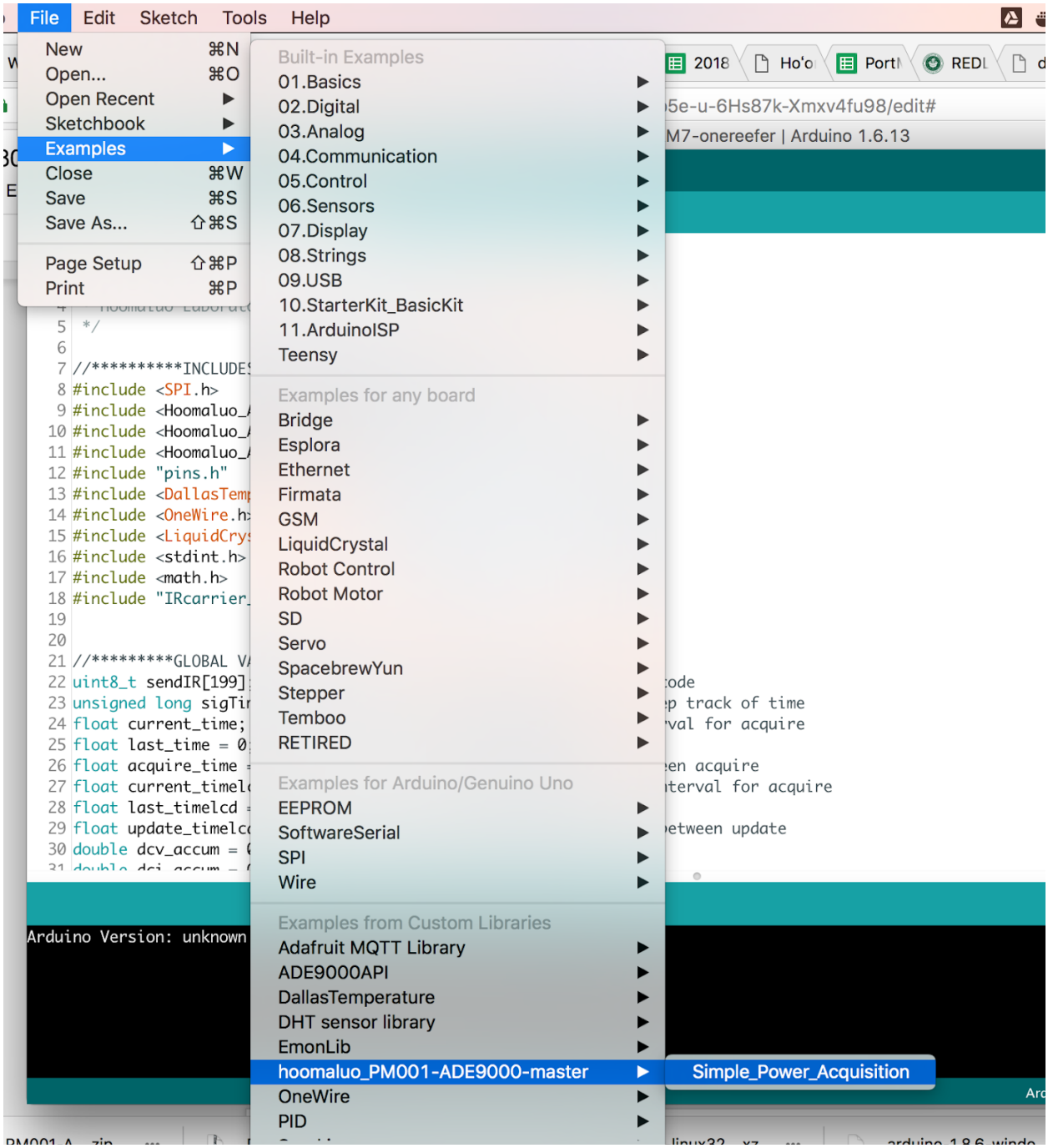
And download the zip file:



Unzip the folder into your Documents->Arduino->Libraries folder:



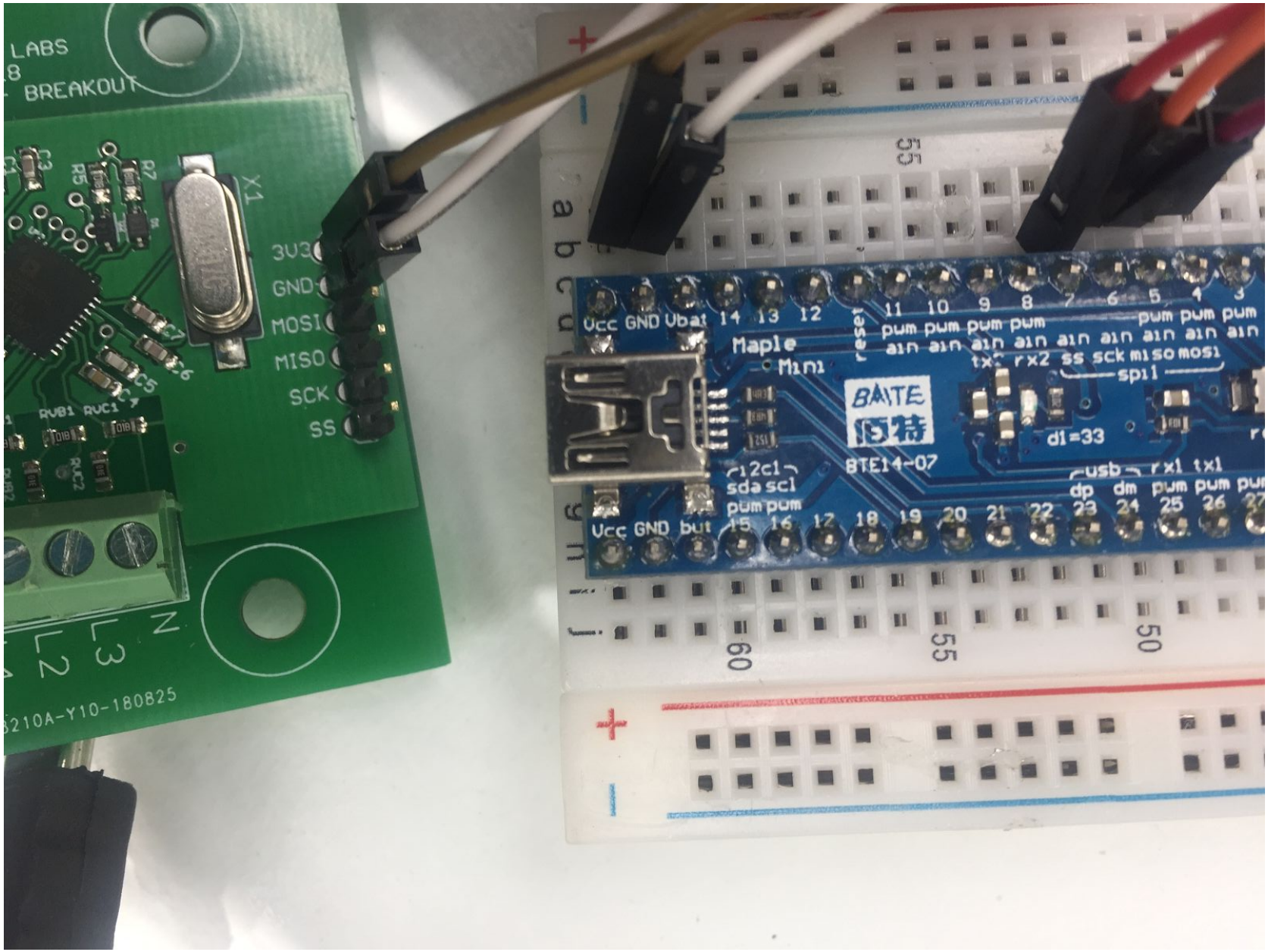
Restart Arduino IDE and open the simple power monitor example from the examples folder:

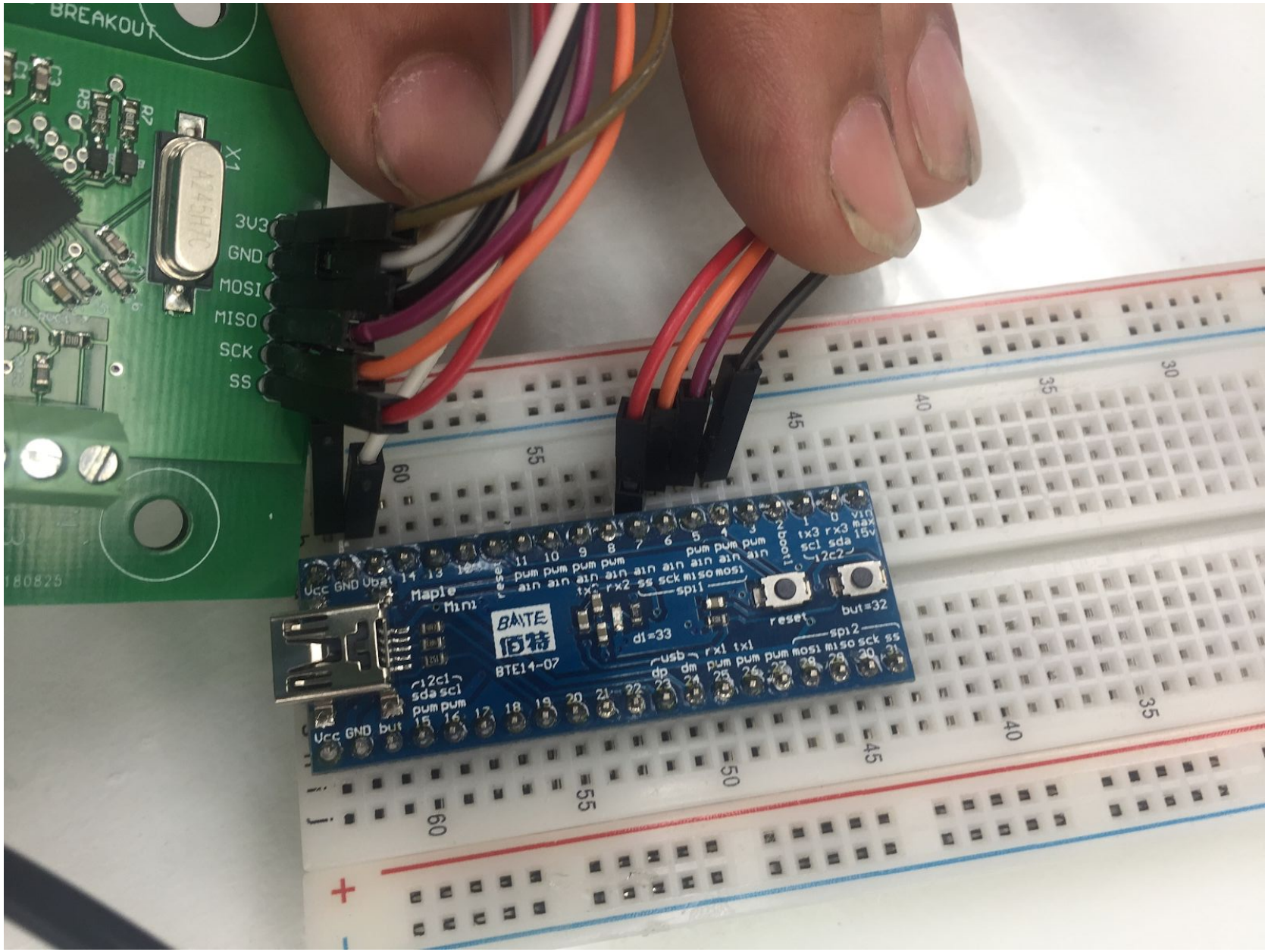


CONNECTING SPI INTERFACE

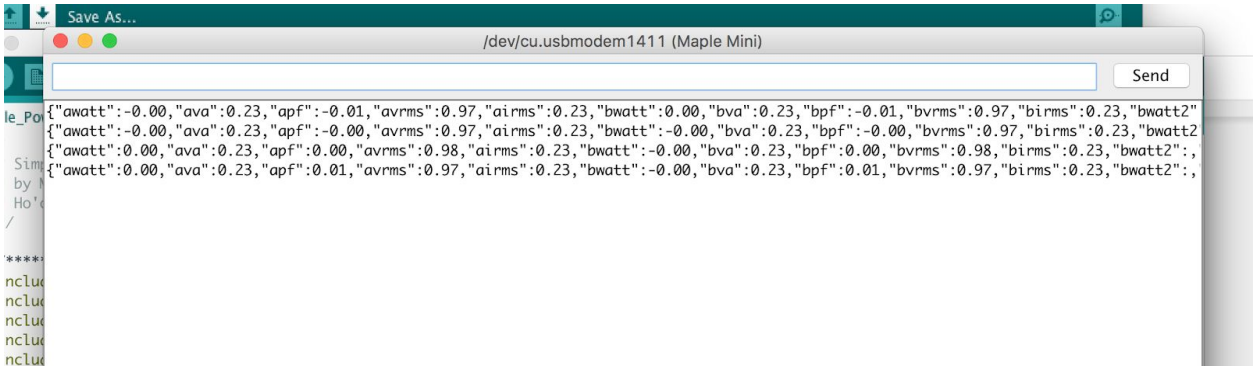
Connect the PM001 to 3.3V power, ground and the SPI1 interface of the maple Mini.

Maple Mini		PM001
Pin 7	->	SS
Pin 6	->	SCK
Pin 5	->	MISO
Pin 4	->	MOSI
VCC	->	3V3
GND	->	GND

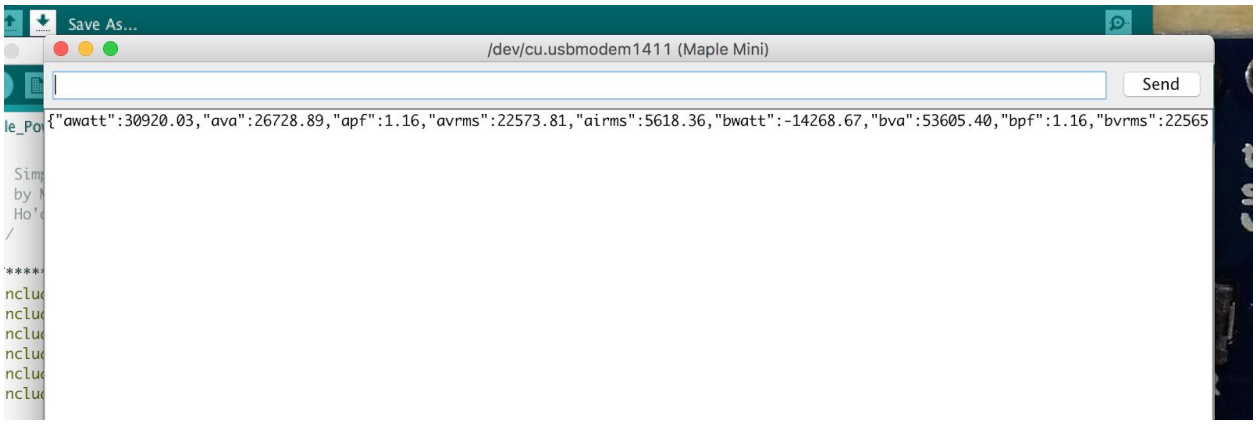

LABS
BREAKOUT
X1
3U3
GND
MOSI
MISO
SCK
SS
RVB1 RVC1
L2 L3 N
Maple
Mini
BTE14-07
Vcc GND Vbat
Vcc GND but
dl=33
sda scl
spi1
usb
rx1 tx1


BREAKOUT
3V3
GND
MOSI
MISO
SCK
SS
Maple
Mini
BTE14-07
reset
but=32
di=33
spi1
spi2
i2c1
i2c2
usb
Vcc GND
mosi miso sck ss
sda scl

Now test that the chip is working by uploading the sketch to the maple mini. After uploading, press the reset button on the maple mini twice then open the serial monitor.



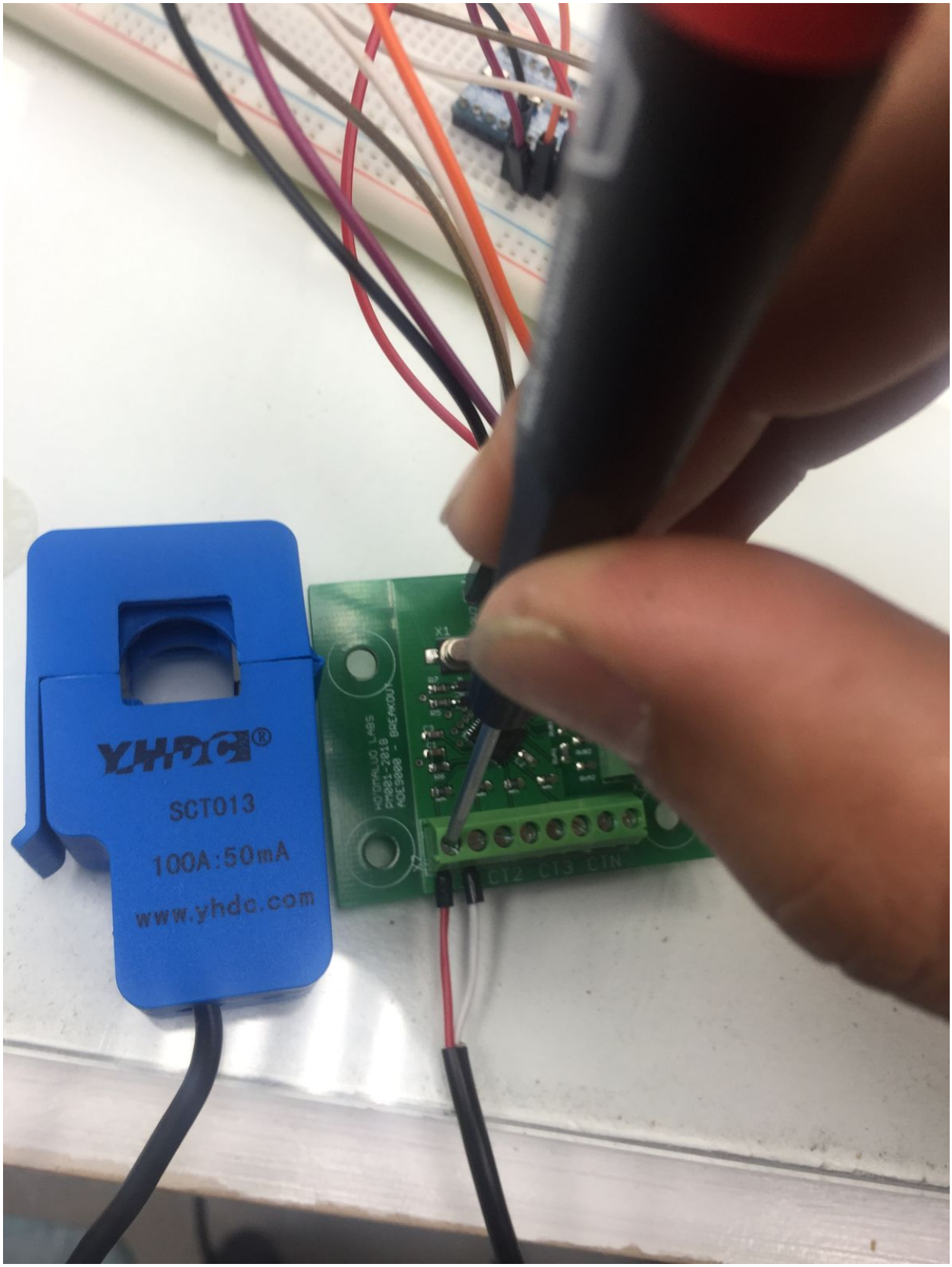
If your output looks like this:



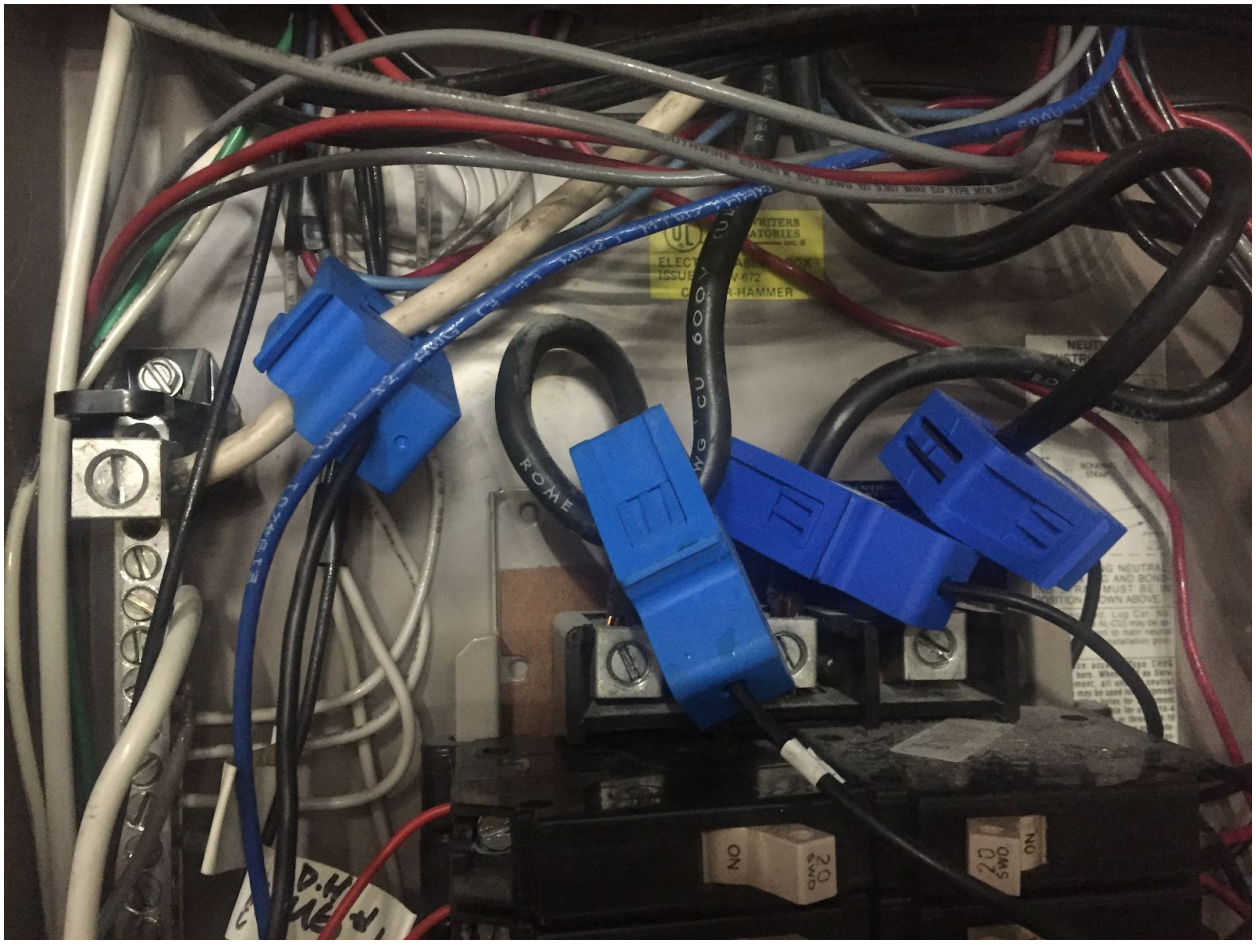
There is a problem with the SPI connection. Check that all of your connections are correct and try again.

CONNECTING CURRENT TRANSDUCERS (CT)

You can use up to 4 CTs to monitor current in and line conductors and the neutral conductor.


YHDC
SCT013
100A:50mA
www.yhdc.com
CT2 CT3 CIN

A typical situation may look something like this:



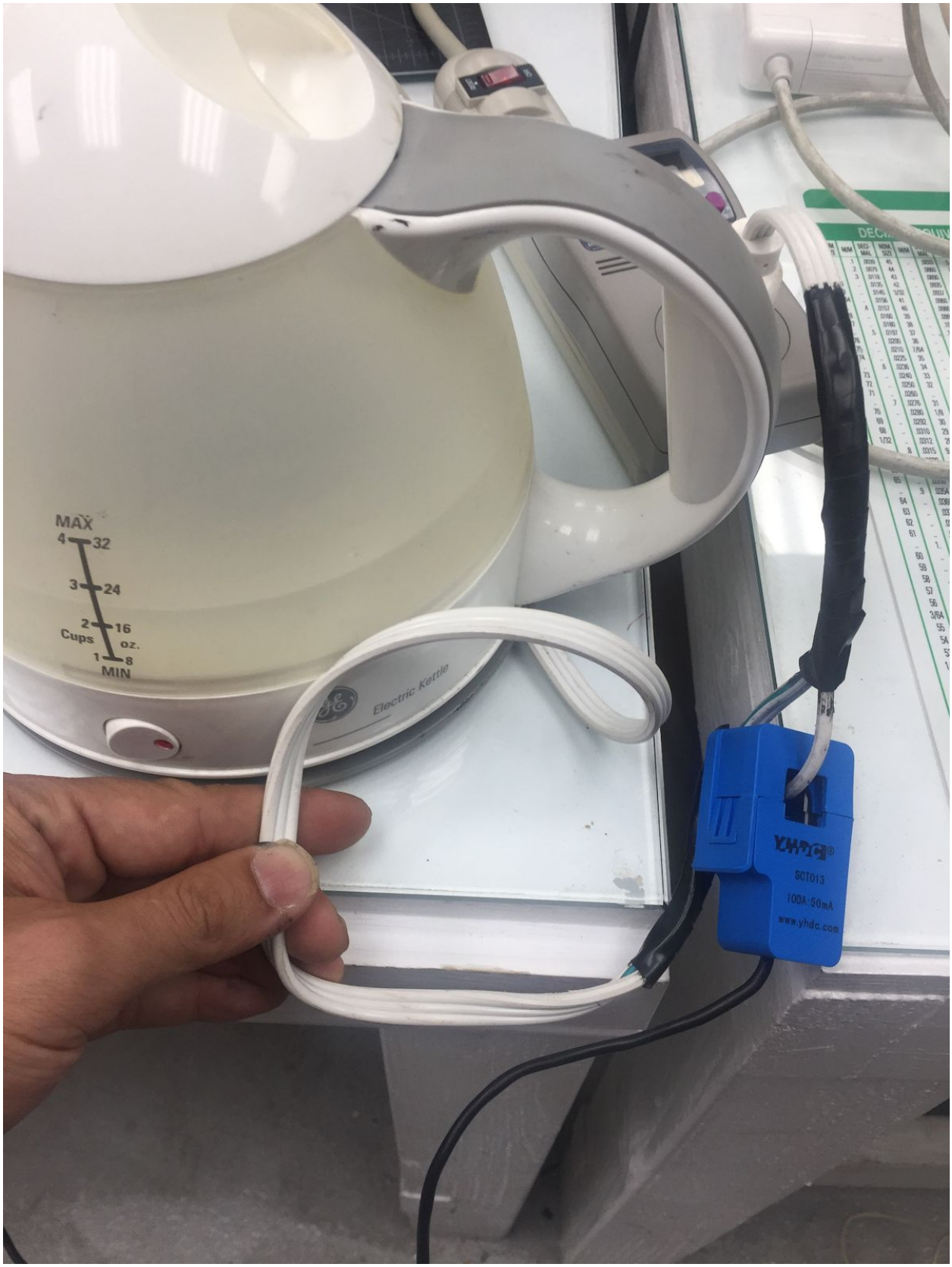
With one CT for each line in and one for the neutral line.

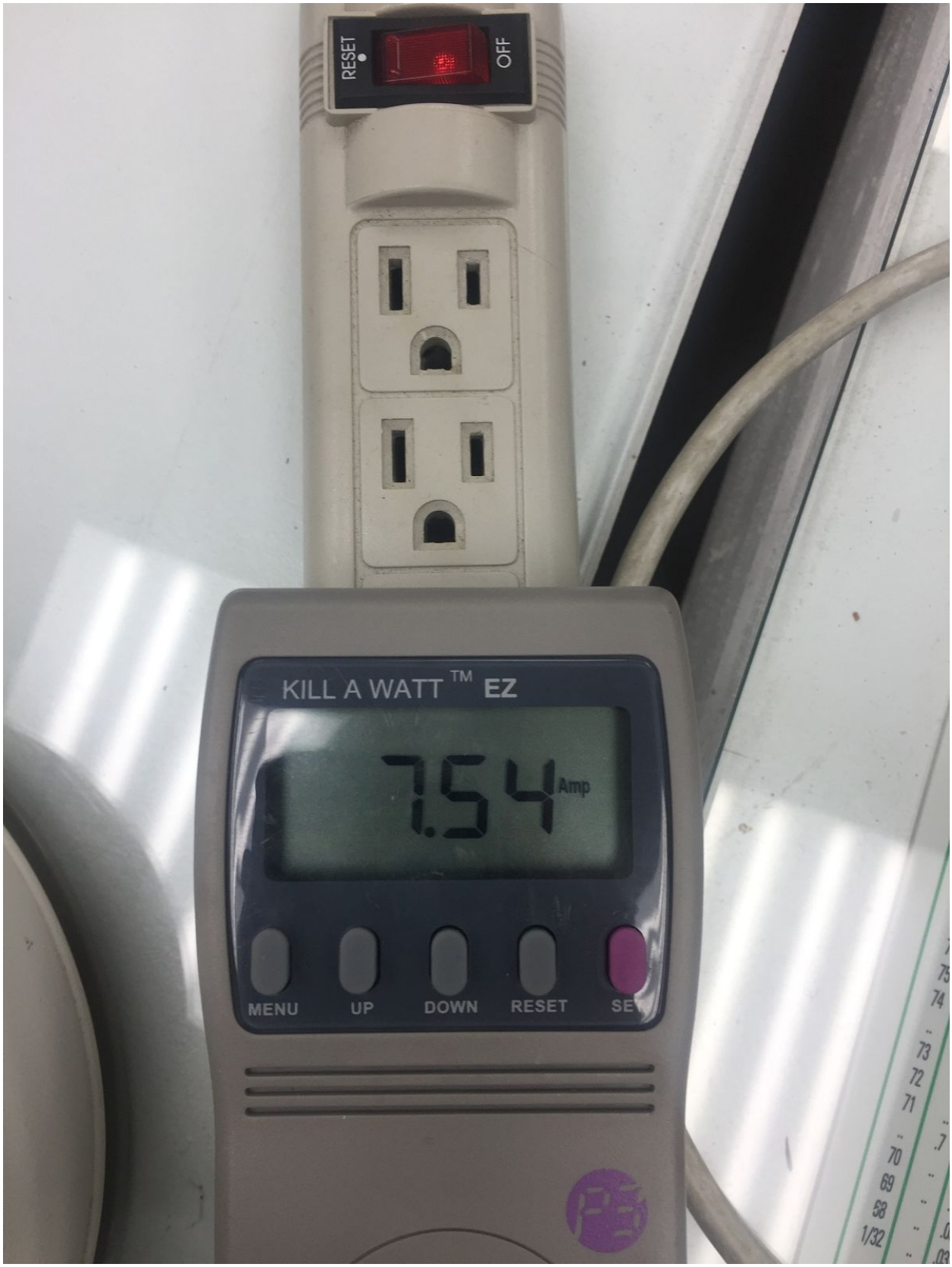
Remember that the burden resistors are 8.2 ohms so if you use your own CTs you need to make sure that the maximum voltage swing at maximum current will be under 500 mV.

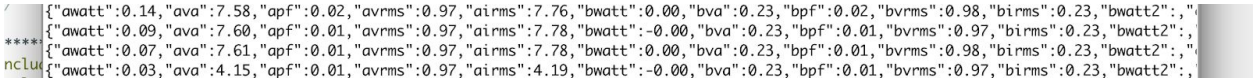
Peak Current * CT ratio * 8.2 < 0.5 Volts

CALIBRATION

You can calibrate your device by using a quality clamp meter or other current measuring device.


MAX
4
32
3
24
2
16
Cups
oz.
1
8
MIN
Electric Kettle
SCT013
100A:50mA
www.yhdc.com

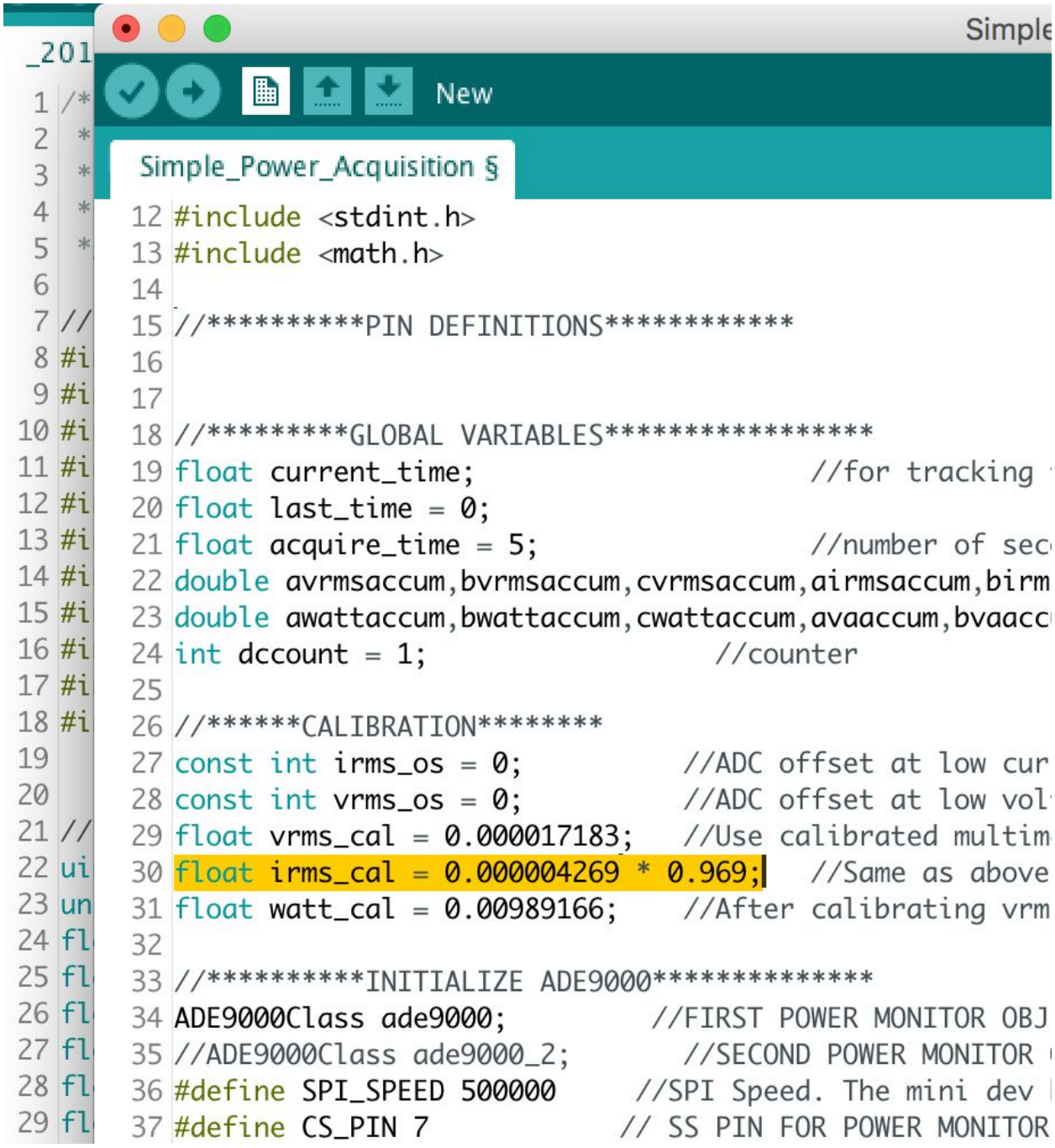

RESET
OFF
KILL A WATT ™ EZ
7.54 Amp
MENU
UP
DOWN
RESET
SET



Read the Current from both your serial monitor (airms here) and the measuring device. Notice that here we see a slight difference in readings. Perform the following calculation:

Measuring Device Reading/PM001 Reading = 7.54/7.78 = 0.969

Now find the following line in the example sketch and update by multiplying the calibration factor irms_cal by this number:



VOLTAGE MEASUREMENT

Be very careful working with dangerous voltages present in line voltage systems. Connect Fuseholders with 1 Amp fuses to L1, L2 and L3, and white neutral breakout to N.

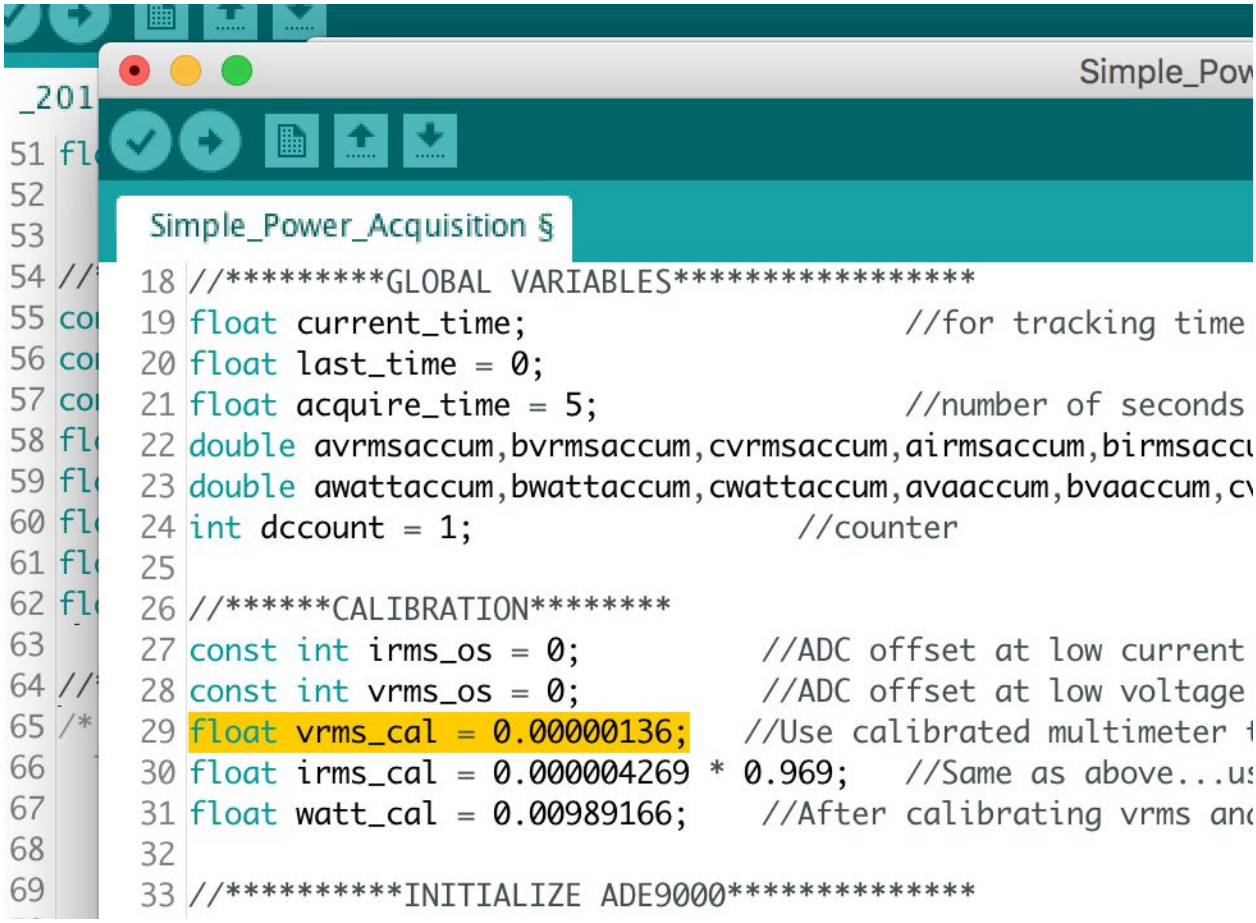

National Security
NSA.gov/Caree

Carefully connect the neutral line to the white wire and L1, L2, and L3 to respective phase legs.

Measure the voltage with a voltmeter and with the PM001. Compare and perform same calibration as before with current:

Measuring Device Reading/PM001 Reading = Calibration coefficient

Multiply by the vrms_cal line:

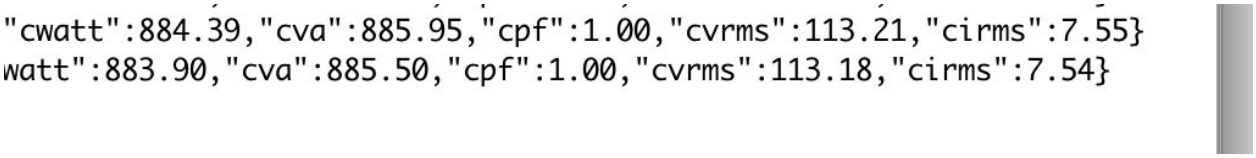


CALIBRATING POWER READINGS

Now you can connect a load to your device and take some readings. Note the values for:

VA
IRMS

VRMS



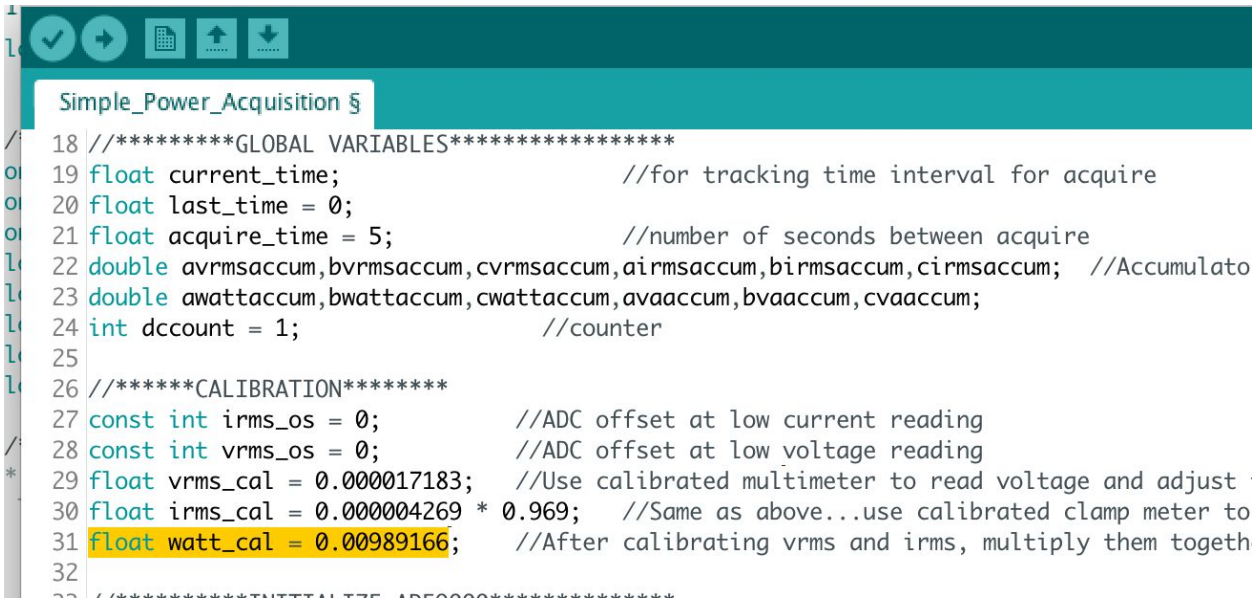
Now multiply :

Xvrms * Xirms= Xva

Where X is the respective phase you are reading. This number should be the same as Xva which is printed. If not then perform the following calculation.

Calculated VA/reading from PM001 = calibration factor

Then multiply this factor by watt_cal to calibrate the power readings.



That would be:

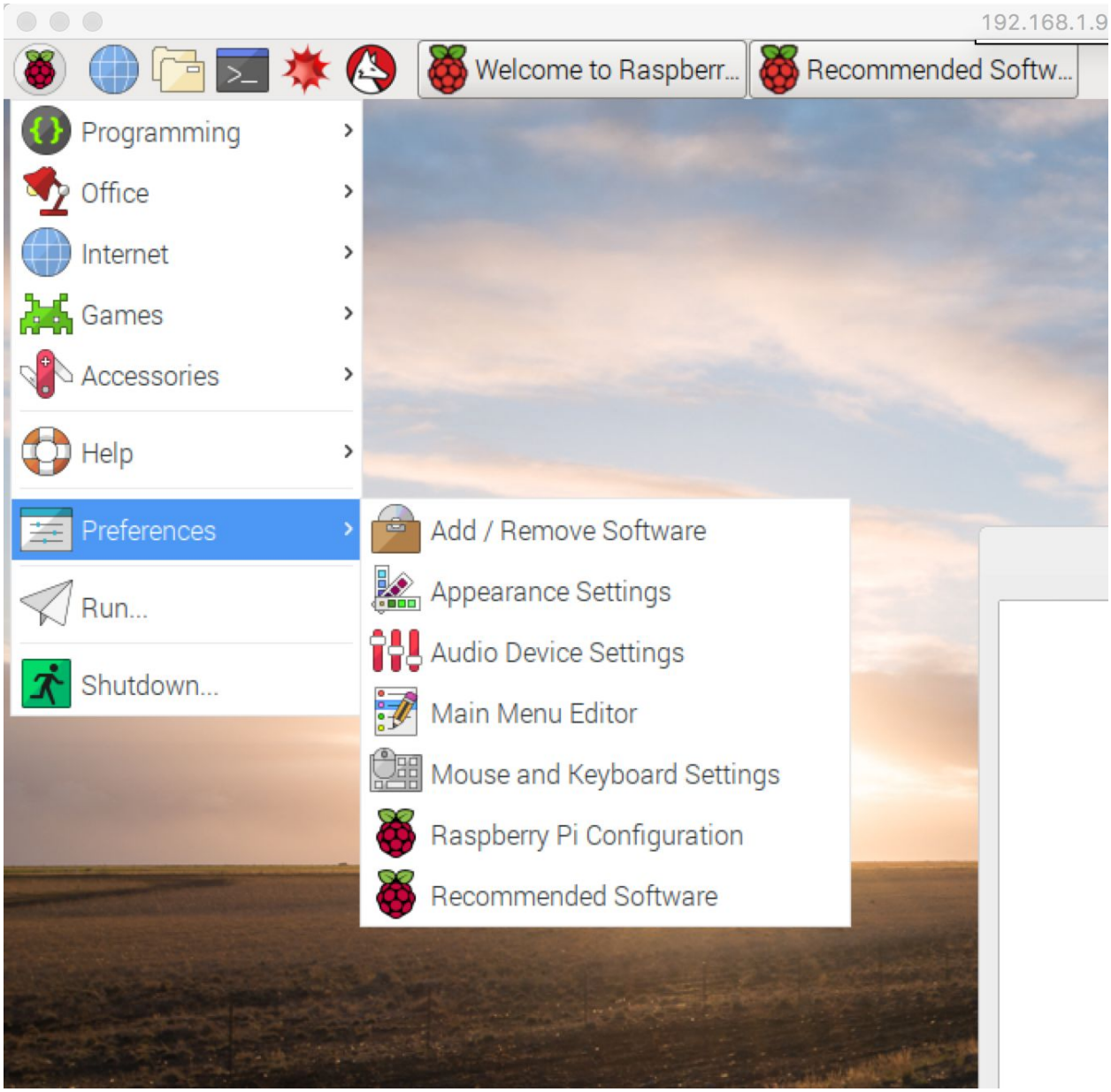
float watt_cal = 0.00989166 * calibration factor

CONNECTING TO A RASPBERRY PI THROUGH NODE RED

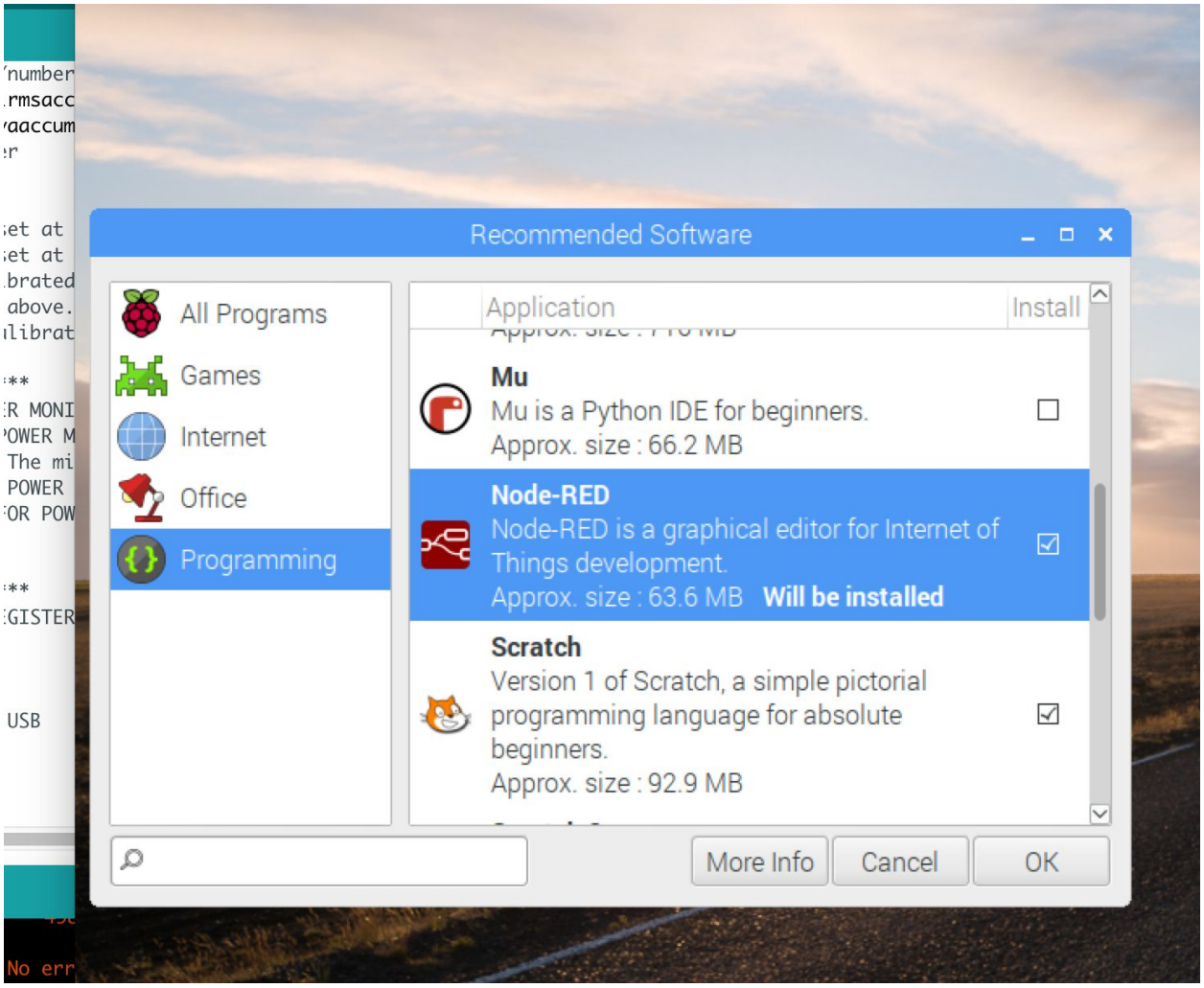
The Raspberry PI comes with a fun but powerful program called NodeRed which we love to use because it makes building programs fast and easy. It is based off of the nodeJS framework which uses javascript as its primary programming language. Some of you may have noticed that we have framed our data outputs from the Maple Mini as a JSON object. This makes it easier for NodeRed to parse the data.

On the older versions of Raspbian Jessie, NodeRed comes pre-installed. If you are using the new Stretch distribution use the following procedure to install NodeRed:

Open the recommended software tab:



And install the NodeRed application:



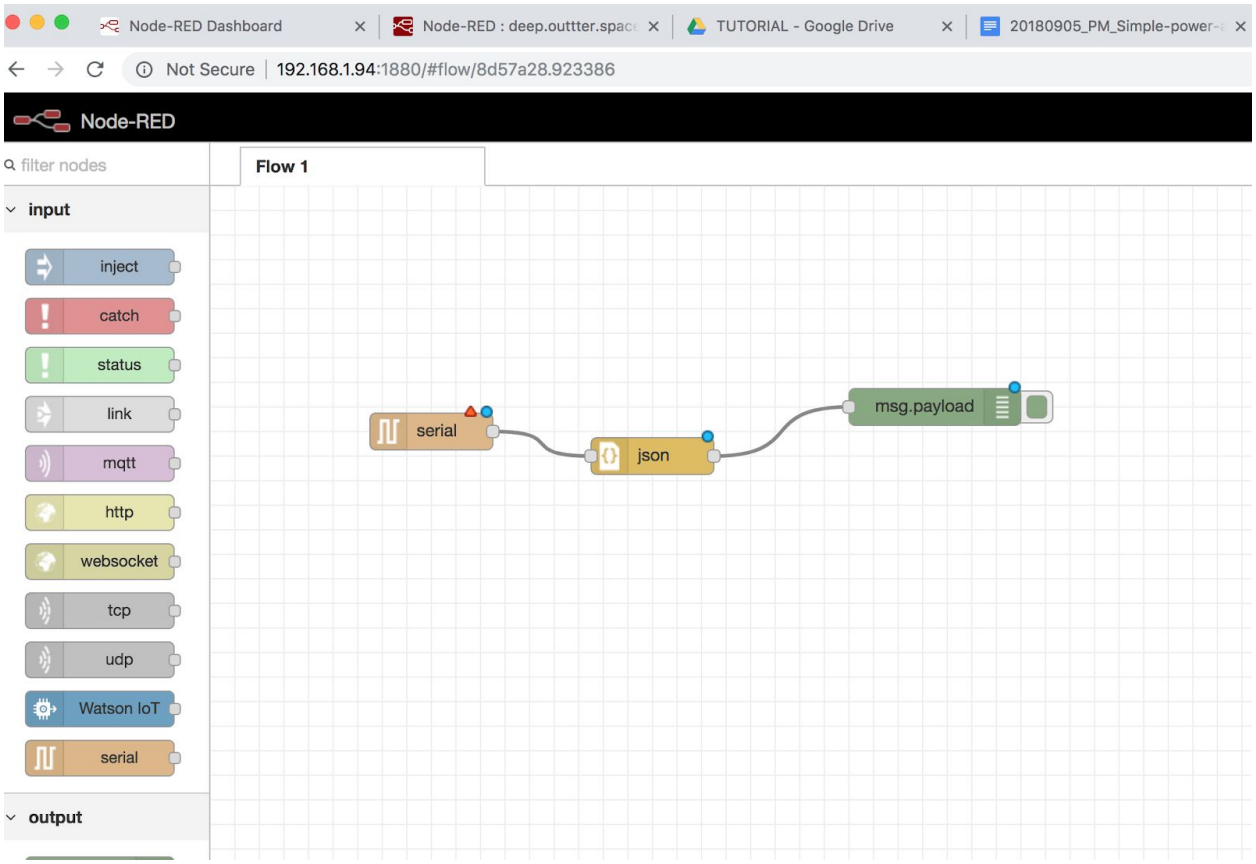
Open the terminal and type this command to have nodered autostart when you boot the PI.

sudo systemctl enable nodered.service

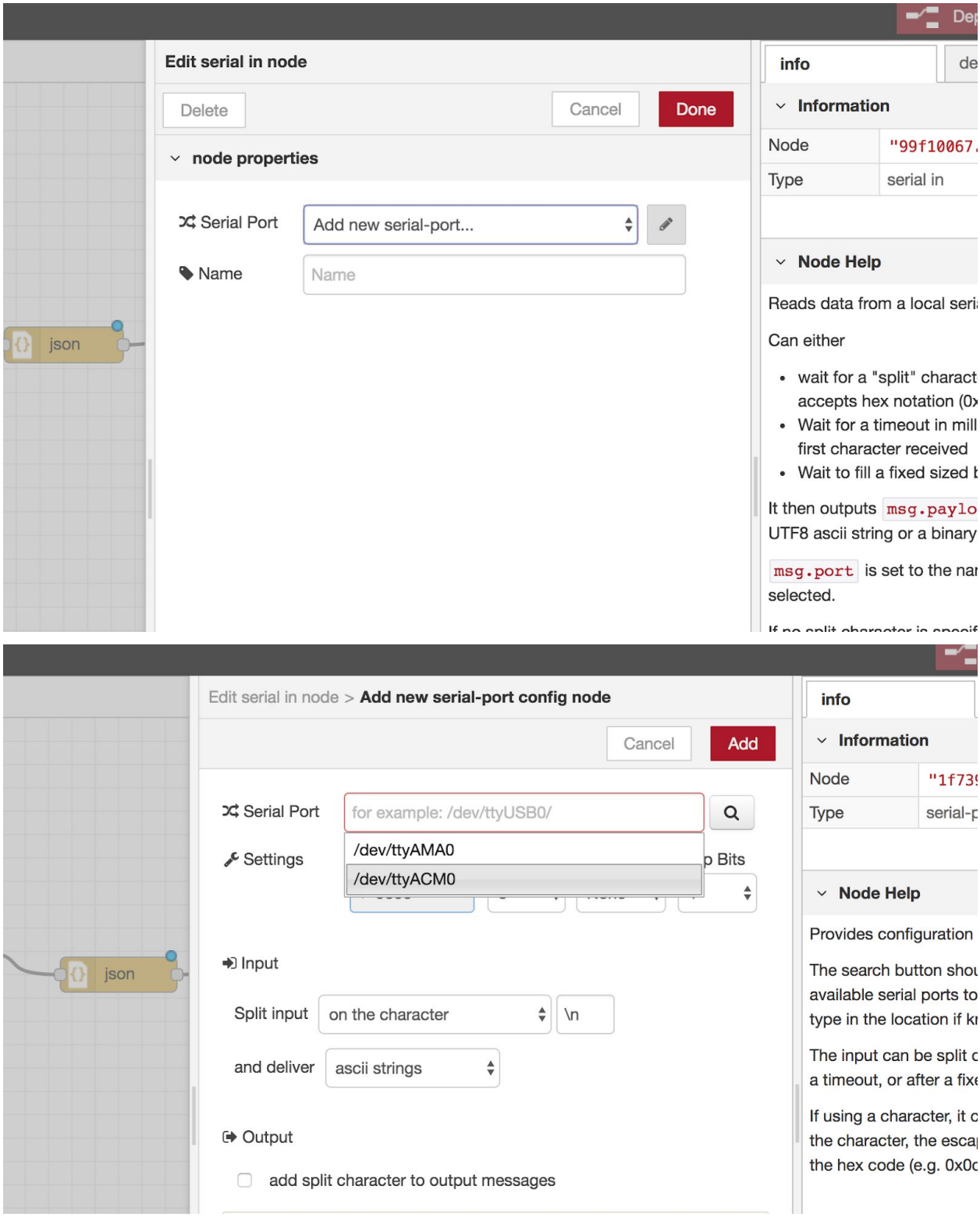
Now open your browser on the PI and type in localhost:1880 in the address bar. You should be able to pull up the NodeRed interface at this point.

Plug the USB cable from the maple mini and your power monitor into the USB port on the PI.

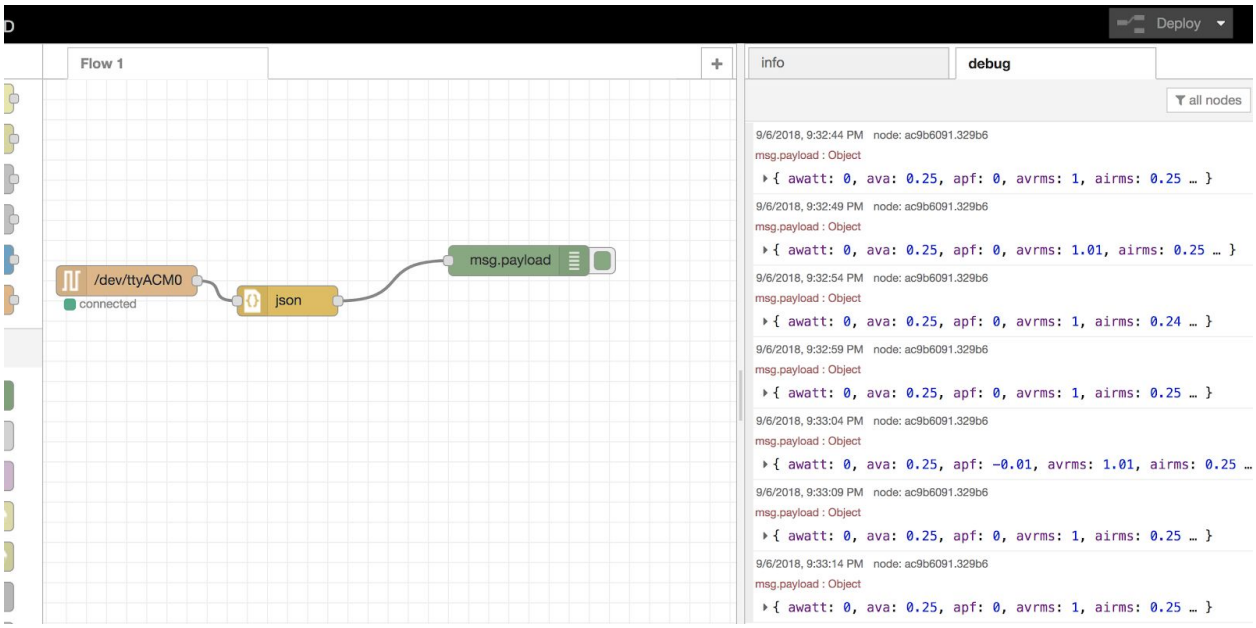
Drag the following nodes into the flow:



Connect and configure the serial port by double clicking on the node:



Deploy your flow and there you go!!! A working power monitor. From here you can decide to do what you want with your data...send it over MQTT, store it in a database, make an HTTP dashboard...you name it!


Deploy
Flow 1
/dev/ttyACM0
connected
json
msg.payload
info
debug
all nodes
9/6/2018, 9:32:44 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: 0, avrms: 1, airms: 0.25 … }
9/6/2018, 9:32:49 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: 0, avrms: 1.01, airms: 0.25 … }
9/6/2018, 9:32:54 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: 0, avrms: 1, airms: 0.24 … }
9/6/2018, 9:32:59 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: 0, avrms: 1, airms: 0.25 … }
9/6/2018, 9:33:04 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: -0.01, avrms: 1.01, airms: 0.25 …
9/6/2018, 9:33:09 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: 0, avrms: 1, airms: 0.25 … }
9/6/2018, 9:33:14 PM node: ac9b6091.329b6
msg.payload : Object
{ awatt: 0, ava: 0.25, apf: 0, avrms: 1, airms: 0.25 … }