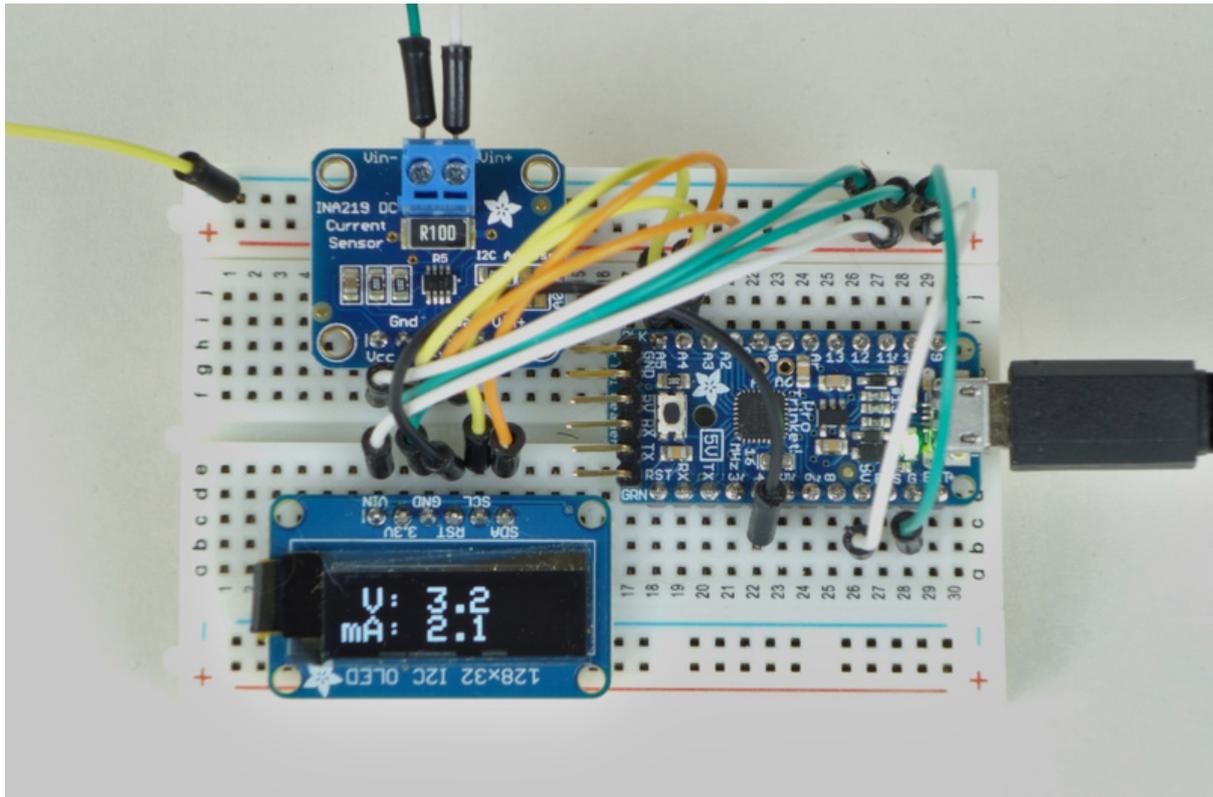




# Pro Trinket Power Meter

Created by Tony DiCola



<https://learn.adafruit.com/pro-trinket-power-meter>

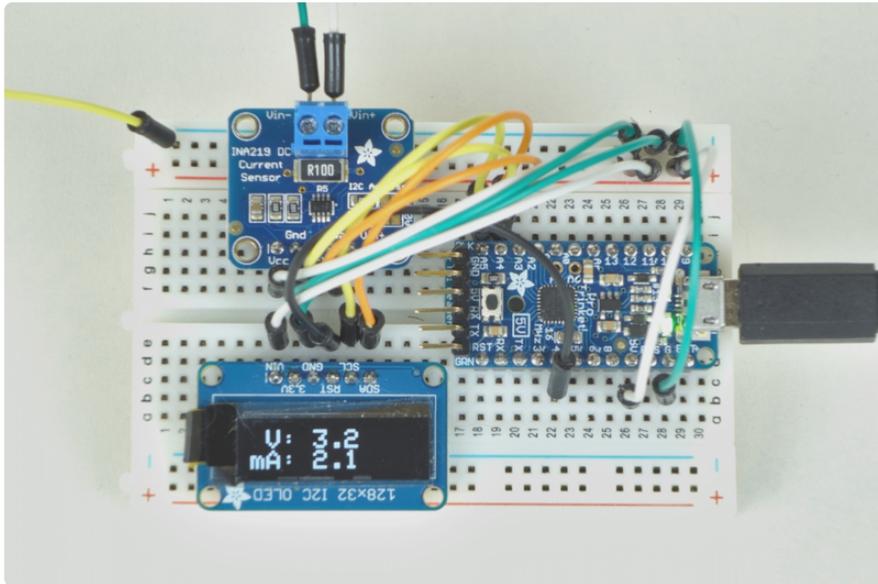
Last updated on 2024-06-03 01:44:38 PM EDT

# Table of Contents

Overview	3
Hardware	3
• Parts	
• Wiring	
Usage	5
• Software	
• Usage	

---

# Overview



This is a useful project to build a small power meter using a [Pro Trinket \(http://adafruit.it/2000\)](http://adafruit.it/2000), an [OLED display \(http://adafruit.it/931\)](http://adafruit.it/931), and the [INA219 current sensing breakout \(http://adafruit.it/904\)](http://adafruit.it/904). You can use this power meter to measure the voltage, current, power, and milliamp-hours of current that are used by a device. Why are these measurements useful? If you're estimating how long a device might run on batteries or if it's possible to run off small solar cells then you'll have to know how much power the device uses. This power meter project is perfect for connecting to a project or device to measure its power consumption and estimate how long it will run on batteries!

Before you get started you'll want to be familiar with the basics of [using an Arduino & the Arduino IDE \(https://adafruit.it/CfF\)](https://adafruit.it/CfF). You'll also need to [do a little soldering \(https://adafruit.it/dxy\)](https://adafruit.it/dxy) to connect headers to the components used in the project.

Continue on to learn about the hardware used in the project, and how to wire it together.

---

## Hardware

### Parts

To build this project you'll need the following parts:

- [INA219 current sensor breakout \(http://adafruit.it/904\)](http://adafruit.it/904).

- **Pro Trinket 5V** (<http://adafru.it/2000>) or **3.3V** (<http://adafru.it/2010>).  
Unfortunately the normal Trinket doesn't have enough space to run the code for this project, so stick with the Pro Trinket or even an Adafruit Metro 328, Arduino Uno, Leonardo, or similar board.
- Appropriate USB cable for your microcontroller. For micro B you can use [this one](http://adafru.it/592) (<http://adafru.it/592>).
- **Monochrome 128x32 I2C OLED display** (<http://adafru.it/931>). You could use a larger OLED like the [128x64 display](http://adafru.it/326) (<http://adafru.it/326>) too (but only the top half will be used unless you modify the project code). An SPI or even color OLED display would work too if you modify the project code to use the appropriate display driver library.
- **Breadboard** (<http://adafru.it/64>) and **hookup wires** (<http://adafru.it/153>) to connect all the components.
- **Soldering tools** (<http://adafru.it/136>) to assemble the breakout boards by soldering on headers, etc.

Once you have all the parts make sure they are assembled by following each of their guides. Here's a handy list of guides to follow:

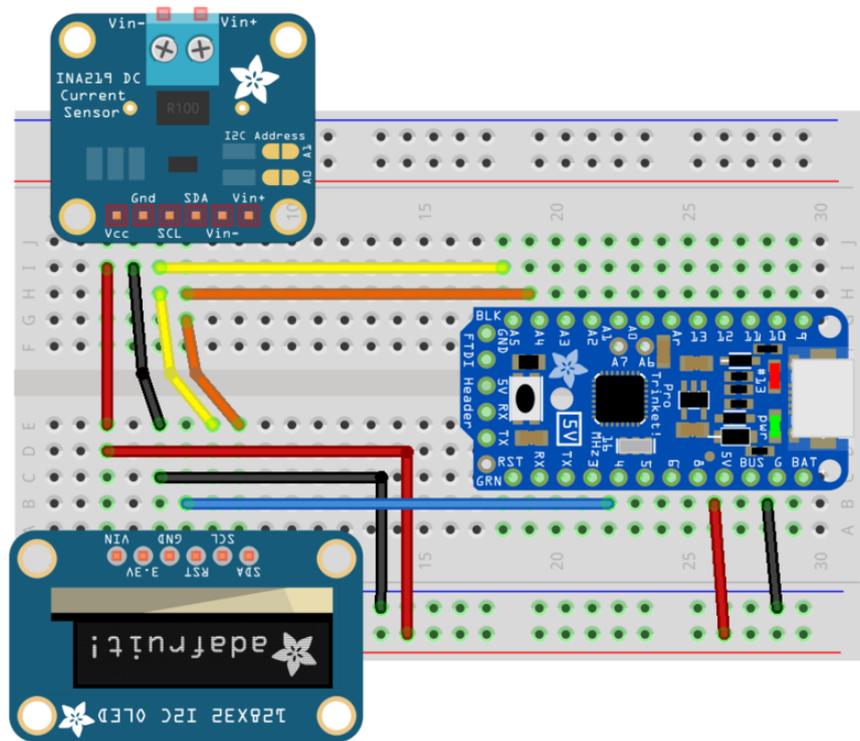
- [INA219 Guide](https://adafru.it/CfL) (<https://adafru.it/CfL>)
- [Pro Trinket Guide](https://adafru.it/iPe) (<https://adafru.it/iPe>)
- [Monochrome OLED Guide](https://adafru.it/sap) (<https://adafru.it/sap>)

## Wiring

To wire up the components you just need to connect their power, ground, and I2C pins. In addition the OLED will need a digital pin connected to its reset pin. Make the connections as follows:

- **Pro Trinket 3.3V or 5V** power to **INA219 VCC** and **OLED VIN**.
- **Pro Trinket ground (G)** to **INA219 ground (GND)** and **OLED ground (GND)**.
- **Pro Trinket I2C SCL (A5)** to **INA219 SCL** and **OLED SCL**.
- **Pro Trinket I2C SDA (A4)** to **INA219 SDA** and **OLED SDA**.
- **Pro Trinket digital 4** to **OLED RST**.

The diagram below shows the wiring connections too. Notice the OLED is rotated upside down so all the components fit on a half-size breadboard.



fritzing

Once the components are wired together continue on to learn how to load the sketch and use the tool.

---

## Usage

## Software

Once the hardware is assembled you're ready to install the Arduino sketch that drives the power meter. Make sure you've [setup the Arduino IDE to program Pro Trinket](https://adafru.it/e8N) (<https://adafru.it/e8N>) and can successfully write sketches (like the blink sketch) to the device.

Next make sure you have the following Arduino libraries installed:

- [Adafruit GFX](https://adafru.it/aJa) (<https://adafru.it/aJa>)
- [Adafruit BusIO](https://adafru.it/GxD) (<https://adafru.it/GxD>)
- [Adafruit SSD1306](https://adafru.it/aHq) (<https://adafru.it/aHq>)
- [Adafruit INA219](https://adafru.it/aRj) (<https://adafru.it/aRj>)

It's easiest to install these libraries [using the library manager \(https://adafru.it/flm\)](https://adafru.it/flm) in the latest versions of Arduino (1.6.4+), however you can also [manually install libraries too \(https://adafru.it/dNR\)](https://adafru.it/dNR).

Once the libraries are installed download the power meter Arduino sketch from its [home on GitHub \(https://adafru.it/fln\)](https://adafru.it/fln) by clicking the button below:

Download Sketch

<https://adafru.it/flo>

Open the zip file to extract the **Pro\_Trinket\_Power\_Meter** Arduino sketch and open it in the Arduino IDE.

At the top of the sketch are a few `#define` configuration values you can change:

- **OLED\_RESET** - This controls what pin is connected to the reset/RST pin on the OLED display. Keep this the default 4 if you've wired your hardware as described in this guide.
- **ROTATION** - This controls the rotation of the OLED display. By default the value is 2 which means the display is rotated 180 degrees so that it's readable when the display is upside down.
- **MODE\_SWITCH** - This control how long the meter will display a set of measurements and defaults to 5 seconds. You can increase or decrease this to change how long the meter displays measurements.

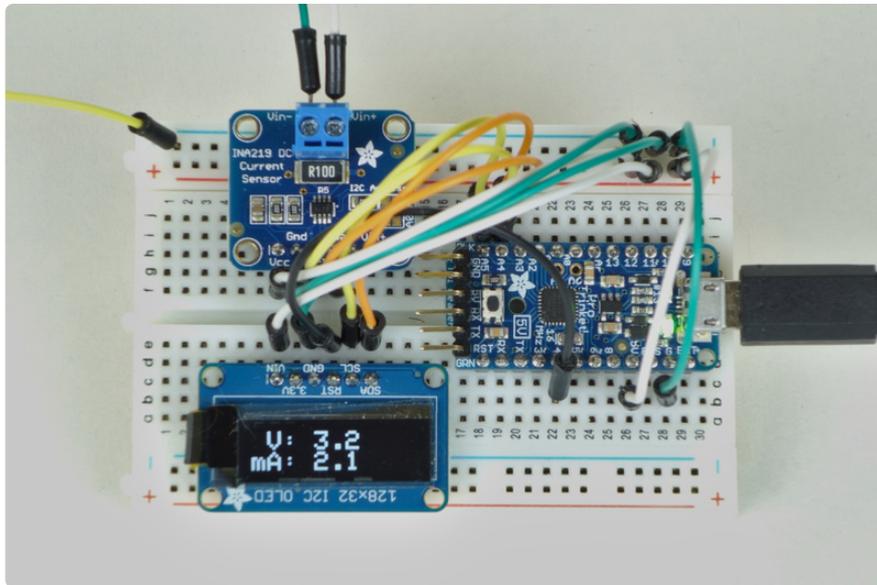
One other part of the sketch you might consider changing is the calibration of the INA219 sensor. Scroll down to the **setup** function and notice this commented note:

```
// Setup the INA219.  
ina219.begin();  
// By default the INA219 will be calibrated with a range of 32V, 2A.  
// However uncomment one of the below to change the range. A smaller  
// range can't measure as large of values but will measure with slightly  
// better precision.  
//ina219.setCalibration_32V_1A();  
//ina219.setCalibration_16V_400mA();
```

Like the comments mention the INA219 is configured by default to accept a range of 32 volts & 2 amps of power. You can uncomment one of the `setCalibration` lines to change this range, like changing to 32 volts & 1 amp or 16 volts & 400mA. When the range of values is reduced the precision of the meter increases so it's a trade-off of a wider range vs. more precision in the measurements.

Once the sketch is configured you're ready to program the Pro Trinket. Follow the same steps as [programming the Blink example sketch \(https://adafru.it/e8N\)](https://adafru.it/e8N) to program the Pro Trinket with the power meter sketch.

After the sketch is programmed you should see the OLED display power on and display readings like the photo below shows.



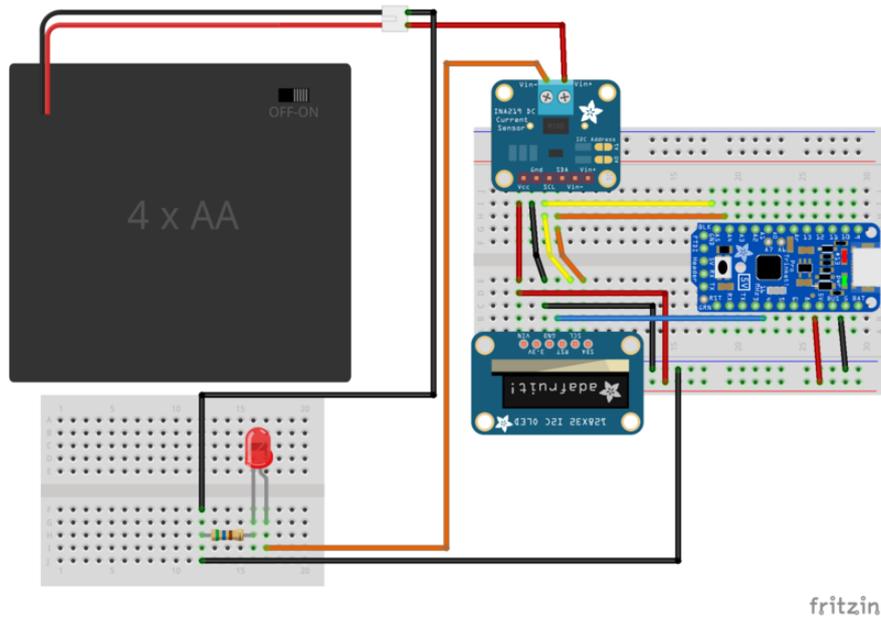
If you don't see the OLED turn on, carefully check all of the components are wired together correctly. Also try troubleshooting each component individually by loading examples from its library (for example try running an Adafruit SSD1306 library example to make sure the OLED is wired correctly and working).

## Usage

To use the power meter you'll need to wire the INA219 in series ahead of the power supplied to a device. Make sure to [read the INA219 guide \(https://adafru.it/BHR\)](https://adafru.it/BHR) as it describes the basic usage of the sensor and how it is wired to a device to measure its current and voltage.

**NOTE: Do not connect the power meter to devices that might exceed its range of volts & current (32 volts & 2 amps)! Remember the power meter is not a general multimeter replacement and it does not have the same safety measures and input range as a real multimeter! NEVER attempt to measure household mains powered devices!**

For example to measure the power usage of a LED connected to batteries see the diagram below:



In this diagram a small battery holder (4x AA) has its positive power output connected to the **Vin+** connection on the INA219. Then the **Vin-** connection on the INA219 is connected to the anode (longer leg) of the LED. In this way the INA219 is in series with the power to the LED and it can measure how much current is consumed by the LED.

In addition to being in series with the LED a wire connects the battery ground to the power meter ground. This allows the power meter to read the voltage from the batteries.

Finally to complete the circuit a small resistor (~560 ohm) is connected from the cathode (shorter leg) of the LED to battery ground.

Try wiring up a LED and battery as shown, then turn on the power to the batteries. Make sure the power meter's Pro Trinket has power too as it will not power itself from the measured device's power source (you can instead plug in the Pro Trinket's micro USB connector to a power supply/computer, or even power it with batteries).

You should see the meter display different power measurements. Every 5 seconds the display will change to show a new set of values, and in total there are 4 values that are displayed:

- **Volts & Amps** - This mode shows the volts (or millivolts) of the power supply at the top, and the current consumed by the device in amps (or milliamps) at the bottom. For example for an LED and 4 batteries you might see:

- **V: 6.00**

- **mA: 7.10**
- **Watts & Amp-hours** - This mode shows the power at the top in watts (or milliwatts), and the observed amp-hours (or milliamp-hours) since the power meter has been powered on. For the LED you might see values to start like:
  - **mW: 42.86**
  - **mAh: 0.0**

The amp-hour measurement is interesting because it's showing how much current has been consumed by the device since the power meter was turned on. To start the amp-hours will be around 0, but over time they will grow based on how much current a device uses.

This is a very useful measurement to understand the battery life of a device since batteries are measured in amp-hours (or more typically milliamp-hours). If you observe the milliamp-hours of current used by a device you can compare it to the milliamp-hour capacities of batteries and estimate the battery life of the device.

Check out this [guide on batteries \(https://adafru.it/CfM\)](https://adafru.it/CfM) for more great information.

Since a LED draws a constant amount of current, after an hour you should see a milliamp-hour value that's about equal to the average current usage of the LED. In this example we'd expect to see about 7.1 milliamp-hours after one hour of measurements.

Note that all the measurements are updated once a second, so devices with quickly changing power demands might draw spikes of current that the meter does not detect.

That's all there is to using the power meter! You can use this project to measure the power consumption of devices you build, and even estimate how long they might run on batteries. Some interesting ways you might extend the project include:

- Design and build a small 3D printed enclosure to hold the Pro Trinket, OLED display, INA219, and a battery to power them.
- Add buttons to allow toggling the displayed measurements.
- Switch to a larger display that can show more measurements on screen at a time.
- Add new modes to show measurements like the minimum, maximum, and average observed voltage & current values.