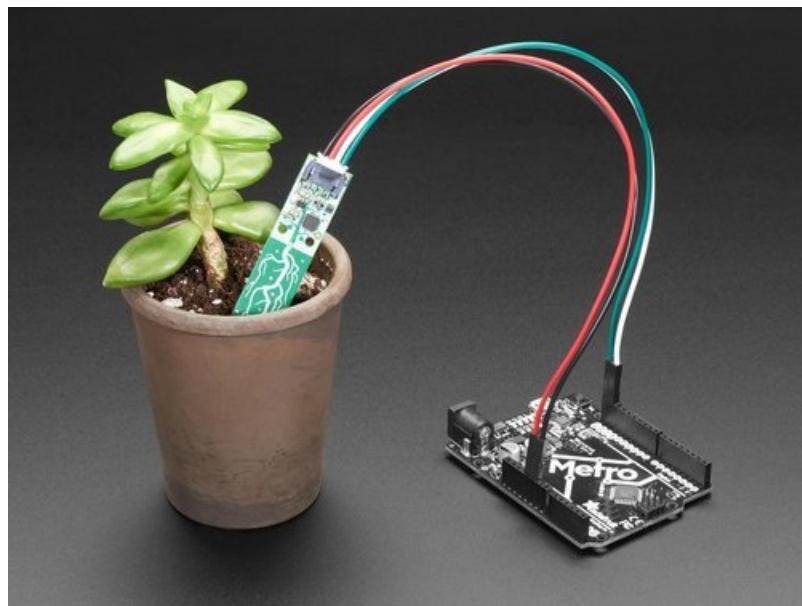




Adafruit STEMMA Soil Sensor - I2C Capacitive Moisture Sensor

Created by lady ada

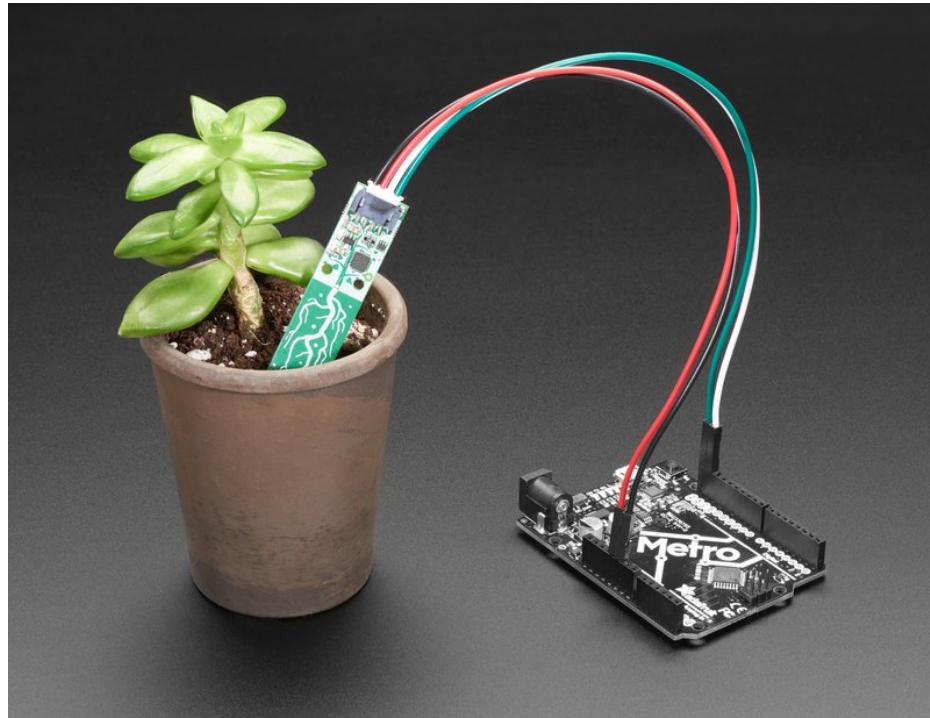


Last updated on 2018-11-22 07:41:30 PM UTC

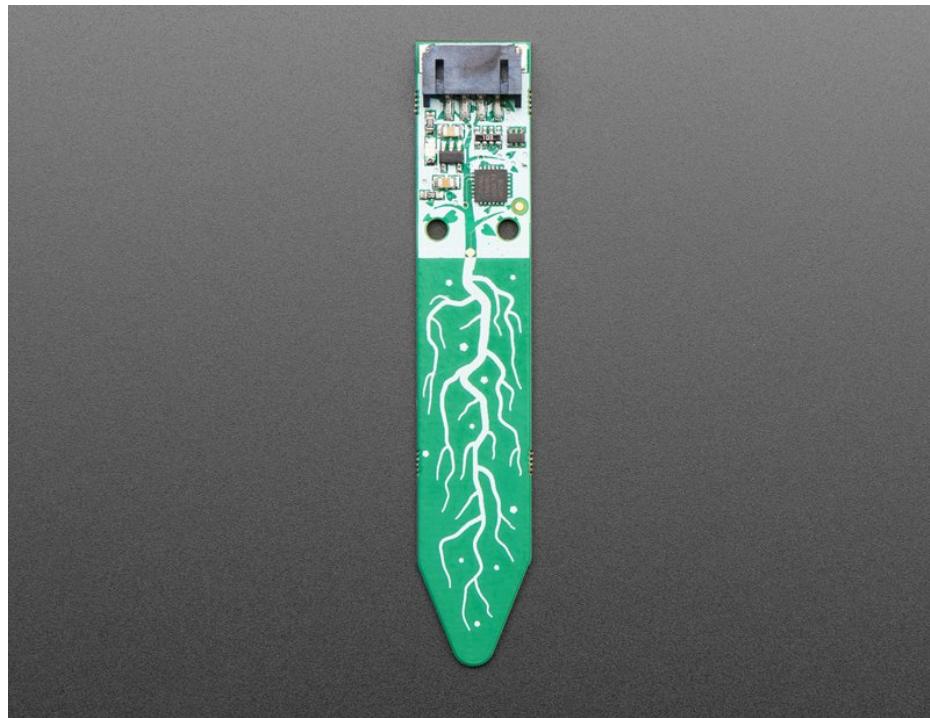
Guide Contents

Guide Contents	2
Overview	3
Pinouts	6
JST PH 4-Pin to Female Socket Cable - I2C STEMMA Cable - 200mm	6
JST PH 4-Pin to Male Header Cable - I2C STEMMA Cable - 200mm	6
Arduino Test	8
Install Adafruit_seesaw library	8
Python & CircuitPython Test	10
CircuitPython Microcontroller Wiring	10
Python Computer Wiring	10
CircuitPython Installation of seesaw Library	10
Python Installation of seesaw Library	11
CircuitPython & Python Usage	11
Downloads	13
Firmware & Files	13
Schematic & Fabrication Print	13

Overview

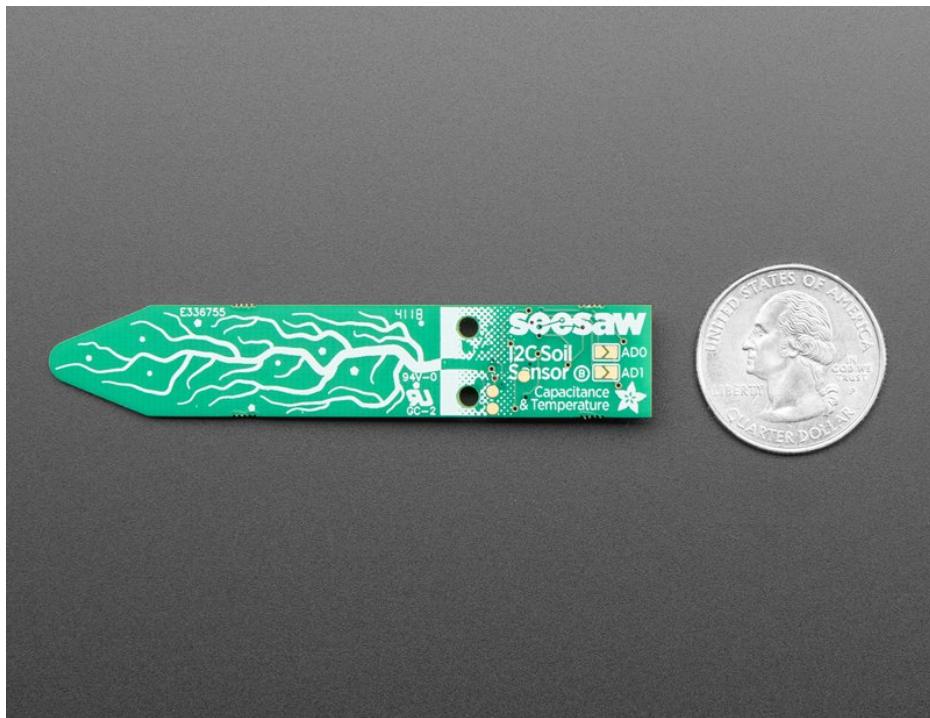


Most low cost soil sensors are *resistive* style, where there's two prongs and the sensor measures the conductivity between the two. These work OK at first, but eventually start to oxidize because of the exposed metal. Even if they're gold plated! The resistivity measurement goes up and up, so you constantly have to re-calibrate your code. Also, resistive measurements don't always work in loose soil.

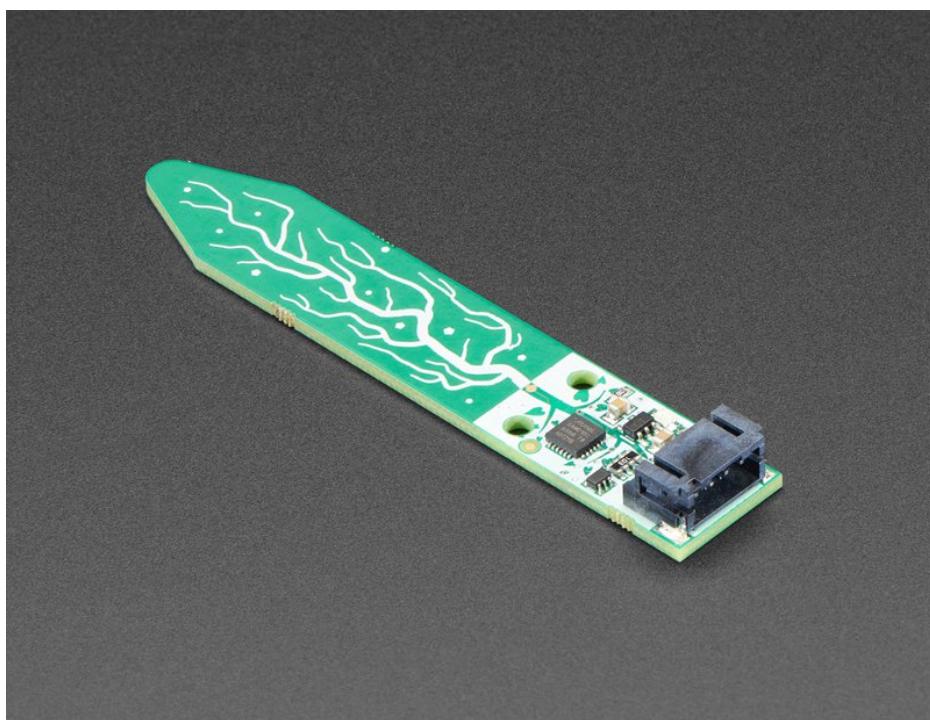


This design is superior with a **capacitive** measurement. Capacitive measurements use only one probe, don't have any

exposed metal, and don't introduce any DC currents into your plants. We use the built in capacitive touch measurement system built into the ATSAMD10 chip, which will give you a reading ranging from about 200 (very dry) to 2000 (very wet). As a bonus, we also give you the ambient temperature from the internal temperature sensor on the microcontroller, it's not high precision, maybe good to + or - 2 degrees Celsius.

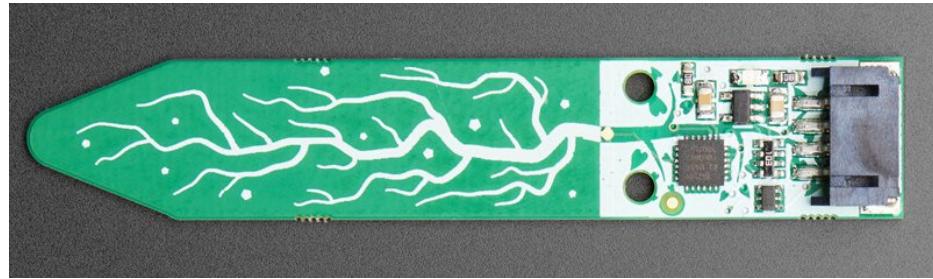


To make it so you can use the sensor with *just about any* microcontroller, we have an I2C interface. Connect a 4-pin JST-PH cable (we have a few stocked) to your microcontroller or single board computer to 3-5V power, Ground, I2C SDA and I2C SCL and then run our **Arduino** or **CircuitPython** code to read the temperature and capacitive measurement. No soldering required!



Please note: This is just the sensor, you'll also need a JST 4-PH cable to go along for plugging it in! We have a few options such as one with [male header pins](https://adafru.it/CYU) (<https://adafru.it/CYU>), [female header pins](https://adafru.it/CVh) (<https://adafru.it/CVh>), and a [JST-JST cable](https://adafru.it/CzH) (<https://adafru.it/CzH>). You can use Seeed Grove cables and boards to connect - the Grove cable is a little different looking but will fit just fine.

Pinouts

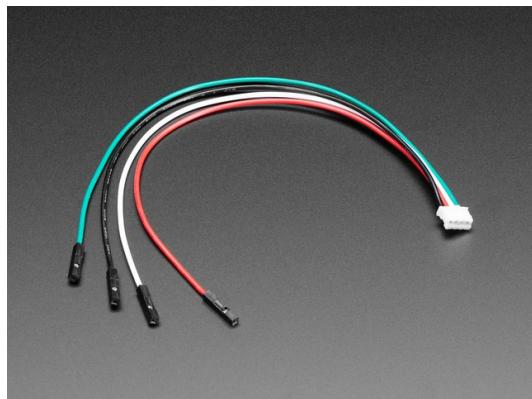


This sensor is pretty simple! Instead of headers, we have a 4-pin JST PH connector. This makes it easy to plug in a cable that goes to another board or prototyping kit.

Looking at the image above, from top to bottom, the pinout is

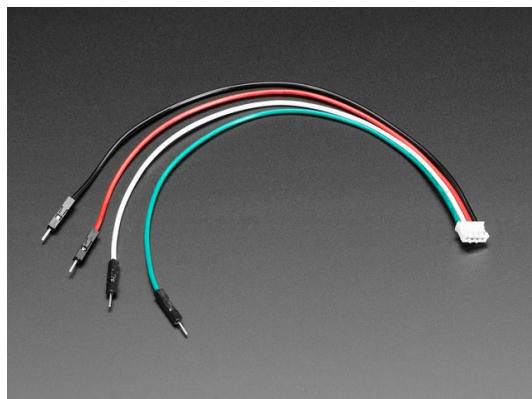
- **GND** - power and logic ground
- **VIN** - 3-5V DC (use the same power voltage as you would for I2C logic)
- **I2C SDA** - there's a 10K pullup to VIN
- **I2C SCL** - there's a 10K pullup to VIN

We suggest these cables, they're even color coded!



[JST PH 4-Pin to Female Socket Cable - I2C STEMMA Cable - 200mm](#)

\$1.50
OUT OF STOCK
[OUT OF STOCK](#)



[JST PH 4-Pin to Male Header Cable - I2C STEMMA Cable - 200mm](#)

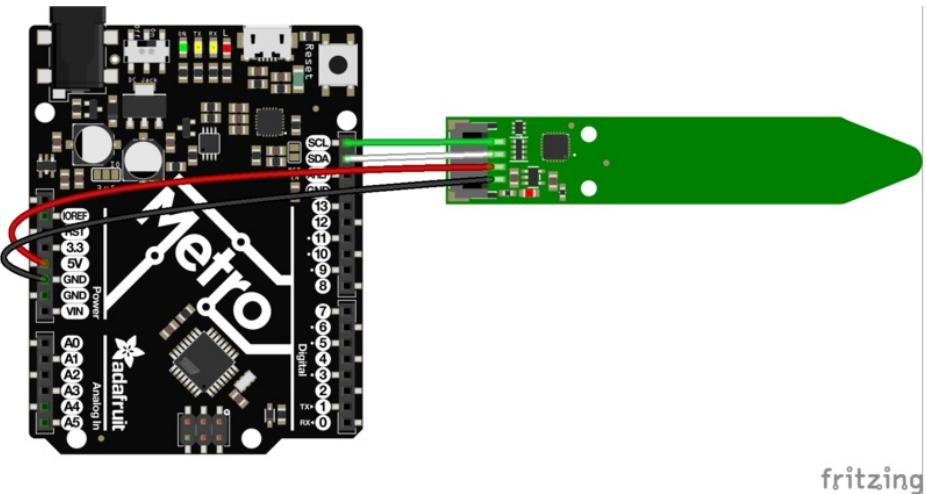
\$1.50
OUT OF STOCK
[OUT OF STOCK](#)



On the back there are 3 SWD programming dots (see the schematic for the exact pinout in case you want to re-program the chip)

There's also two I2C address jumpers. The default address is 0x36, but you can select between 0x36 and 0x39 inclusive by shorting **AD0** (to add 1 to the address) and/or **AD1** (to add 2 to the address)

Arduino Test

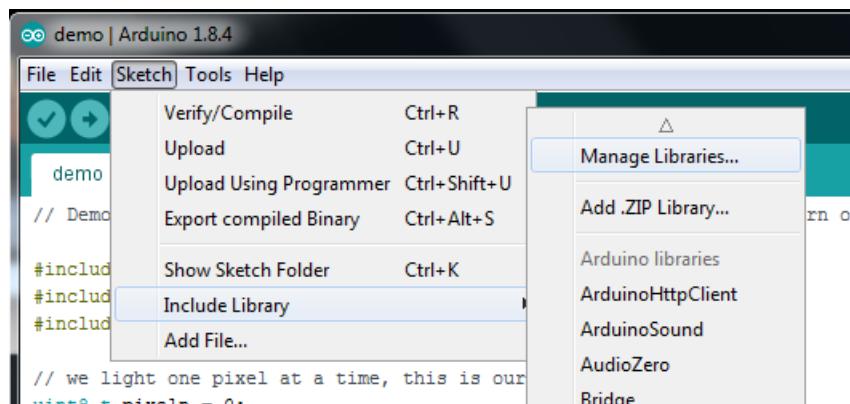


- Connect **Vin Red Wire** to the power supply, 3-5V is fine. Use the same voltage that the microcontroller logic is based off of. For most Arduino's, that is 5V. If you have 3.3V logic, use 3V
- Connect **GND Black Wire** to common power/data ground
- Connect the **SCL Green Wire** pin to the I2C clock **SCL** pin on your Arduino. On an UNO & '328 based Arduino, this is also known as **A5**, on a Mega it is also known as **digital 21** and on a Leonardo/Micro, **digital 3**
- Connect the **SDA White Wire** pin to the I2C data **SDA** pin on your Arduino. On an UNO & '328 based Arduino, this is also known as **A4**, on a Mega it is also known as **digital 20** and on a Leonardo/Micro, **digital 2**

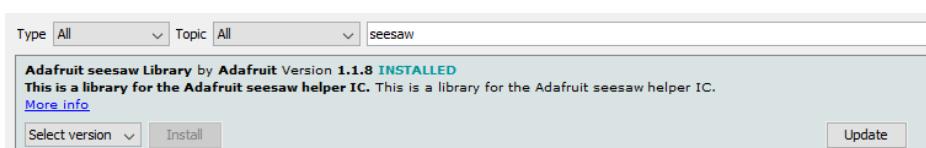
Install Adafruit_seesaw library

To begin reading sensor data, you will need to [install the Adafruit_seesaw library \(code on our github repository\)](#) (<https://adafru.it/BrV>). It is available from the Arduino library manager so we recommend using that.

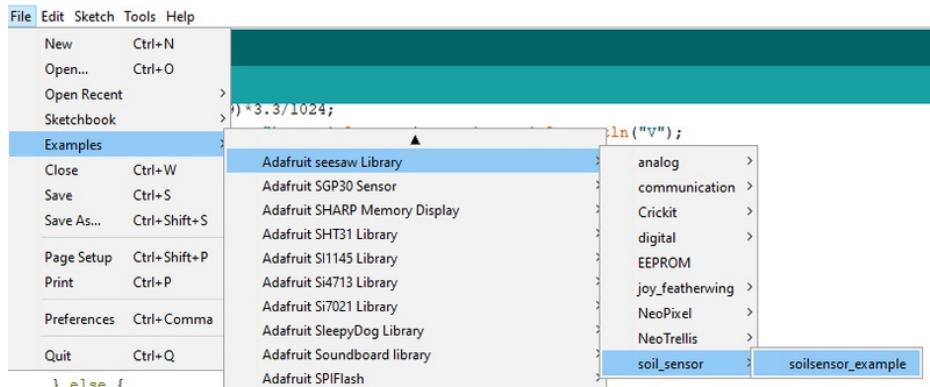
From the IDE open up the library manager...



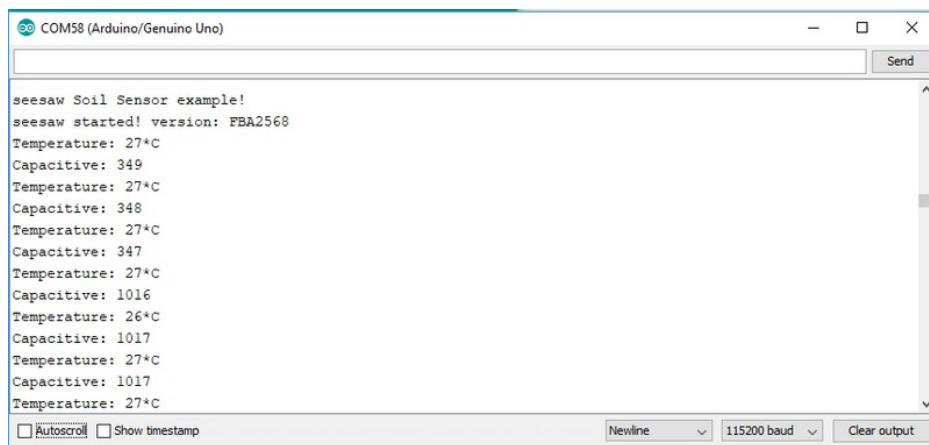
Search for **Adafruit seesaw**, and click **Install**



Now from within the IDE, open up the Adafruit seesaw Soil Sensor Example sketch:



Open up the serial console at 115200 baud to see the temperature and capacitive reading



Try touching the body of the sensor to make the 'moisture' measurement go up. In soil, you'll see this range from about 300 to 500, but make sure to test it by placing in a pot of soil and looking at the serial console to get an idea of what the range is. Note that it does depend on how packed/loose the soil is!

The temperature reading is only on the chip itself. It's not terribly precise but it is within a few degrees. Good for getting a basic sense of how the temperature is moving around the house

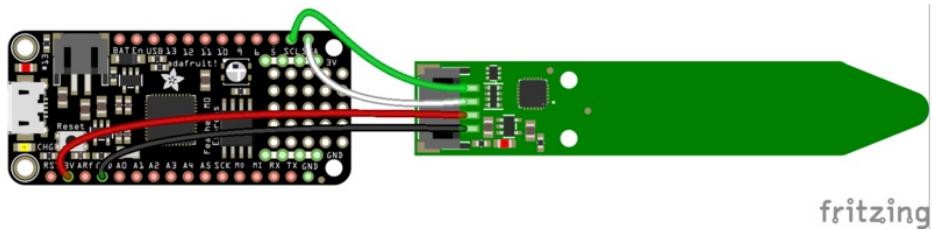
Python & CircuitPython Test

It's easy to use the seesaw soil sensor with Python or CircuitPython and the [Adafruit CircuitPython seesaw](https://adafru.it/BrW) (<https://adafru.it/BrW>) module. This module allows you to easily write Python code that reads the humidity and temperature, pressure from the sensor.

You can use this sensor with any CircuitPython microcontroller board or with a computer that has GPIO and Python thanks to [Adafruit_Blinka](https://adafru.it/Blinka), our CircuitPython-for-Python compatibility library (<https://adafru.it/BSN>).

CircuitPython Microcontroller Wiring

First wire up the soil sensor to your board exactly as shown on the previous pages for Arduino. Here's an example of wiring a Feather M0 to the sensor with I2C:

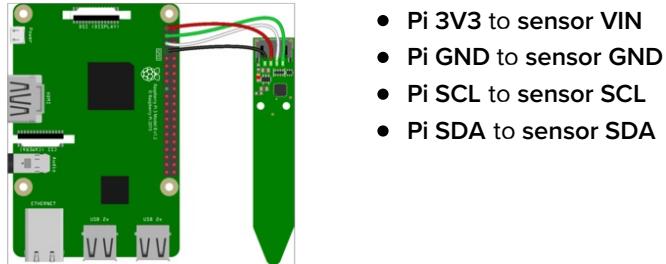


- Board 3V to sensor VIN
- Board GND to sensor GND
- Board SCL to sensor SCL
- Board SDA to sensor SDA

Python Computer Wiring

Since there's *dozens* of Linux computers/boards you can use we will show wiring for Raspberry Pi. For other platforms, [please visit the guide for CircuitPython on Linux to see whether your platform is supported](https://adafru.it/BSN) (<https://adafru.it/BSN>).

Here's the Raspberry Pi wired with I2C:



CircuitPython Installation of seesaw Library

You'll need to install the [Adafruit CircuitPython seesaw](https://adafru.it/BrW) (<https://adafru.it/BrW>) library on your CircuitPython board.

First make sure you are running the [latest version of Adafruit CircuitPython](https://adafru.it/tBa) (<https://adafru.it/tBa>) for your board.

Next you'll need to install the necessary libraries to use the hardware--carefully follow the steps to find and install these libraries from [Adafruit's CircuitPython library bundle](https://adafru.it/zdx) (<https://adafru.it/zdx>). For example the Circuit Playground Express guide has a [great page on how to install the library bundle](https://adafru.it/Bf2) (<https://adafru.it/Bf2>) for both express and non-express

boards.

Remember for non-express boards like the Trinket M0, Gemma M0, and Feather/Metro M0 basic you'll need to manually install the necessary libraries from the bundle:

- `adafruit_seesaw.mpy`
- `adafruit_bus_device`

You can also download the `adafruit_seesaw.mpy` from [its releases page on Github \(https://adafru.it/D5O\)](https://adafru.it/D5O).

Before continuing make sure your board's lib folder or root filesystem has the `adafruit_seesaw.mpy`, and `adafruit_bus_device` files and folders copied over.

Next [connect to the board's serial REPL \(https://adafru.it/pMf\)](https://adafru.it/pMf) so you are at the CircuitPython >>> prompt.

Python Installation of seesaw Library

You'll need to install the Adafruit_Blinka library that provides the CircuitPython support in Python. This may also require enabling I2C on your platform and verifying you are running Python 3. [Since each platform is a little different, and Linux changes often, please visit the CircuitPython on Linux guide to get your computer ready \(https://adafru.it/BSN\)](https://adafru.it/BSN)!

Once that's done, from your command line run the following command:

- `sudo pip3 install adafruit-circuitpython-seesaw`

If your default Python is version 3 you may need to run 'pip' instead. Just make sure you aren't trying to use CircuitPython on Python 2.x, it isn't supported!

CircuitPython & Python Usage

This example sketch shows connecting to the sensor and reading temperature and moisture/capacitive measurement

```
import time

from board import SCL, SDA
import busio

from adafruit_seesaw.seesaw import Seesaw

i2c_bus = busio.I2C(SCL, SDA)

ss = Seesaw(i2c_bus, addr=0x36)

while True:
    # read moisture level through capacitive touch pad
    touch = ss.moisture_read()

    # read temperature from the temperature sensor
    temp = ss.get_temp()

    print("temp: " + str(temp) + " moisture: " + str(touch))
    time.sleep(1)
```

```
Adafruit CircuitPython REPL
main.py output:
temp: 23.6042  moisture: 343
temp: 23.2944  moisture: 348
temp: 23.2944  moisture: 347
temp: 23.5009  moisture: 1015
temp: 23.9204  moisture: 1015
temp: 23.7075  moisture: 335
temp: 24.1269  moisture: 332
--
```

Try touching the body of the sensor to make the 'moisture' measurement go up. In soil, you'll see this range from about 300 to 500, but make sure to test it by placing in a pot of soil and looking at the serial console to get an idea of what the range is. Note that it does depend on how packed/loose the soil is!

The temperature reading is only on the chip itself. It's not terribly precise but it is within a few degrees. Good for getting a basic sense of how the temperature is moving around the house

Downloads

Firmware & Files

- Adafruit seesaw firmware (<https://adafru.it/D1->)
- CircuitPython seesaw library (<https://adafru.it/BrW>)
- Arduino seesaw library (<https://adafru.it/BrV>)
- CAD files in GitHub (<https://adafru.it/D20>)

Schematic & Fabrication Print

