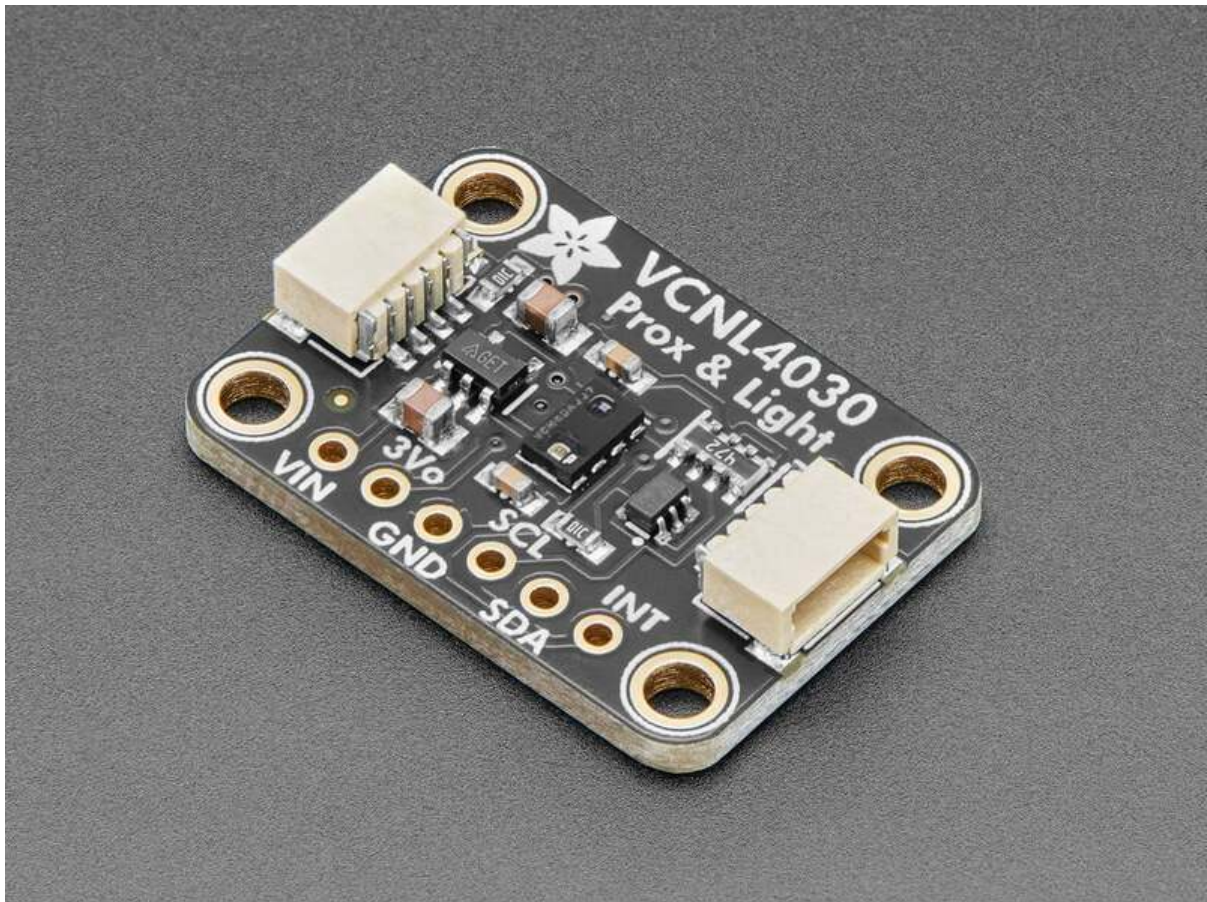




# Adafruit VCNL4030 Proximity and Lux Sensor

Created by Tim C



<https://learn.adafruit.com/adafruit-vcnl4030-proximity-and-lux-sensor>

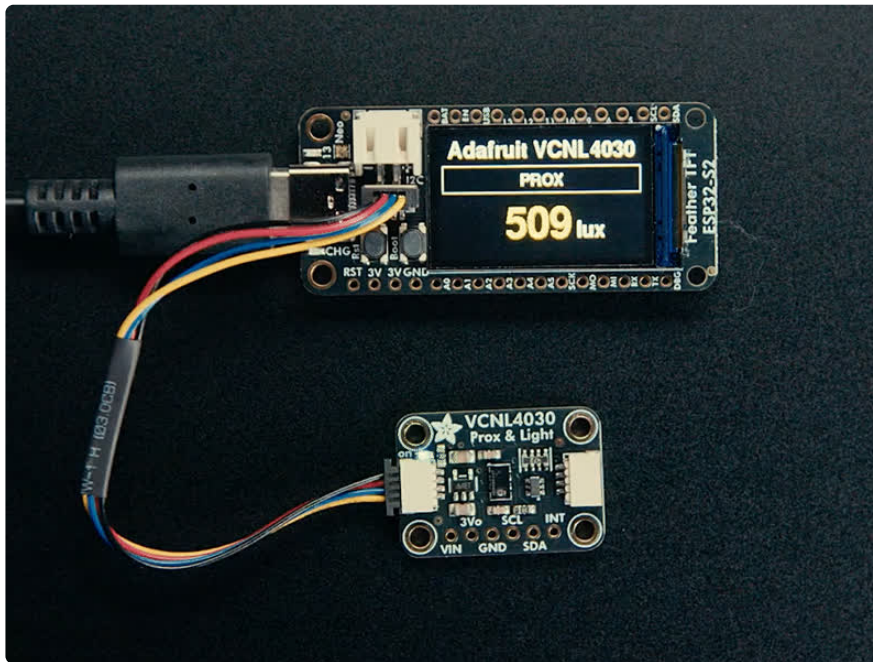
Last updated on 2026-05-08 01:42:13 PM UTC

# Table of Contents

<a href="#">Overview</a>	3
<a href="#">Pinouts</a>	6
<ul style="list-style-type: none"><li>• <a href="#">Power Pins</a></li><li>• <a href="#">I2C Data Pins</a></li><li>• <a href="#">Interrupt Pin</a></li><li>• <a href="#">LED and LED Jumper</a></li></ul>	
<a href="#">CircuitPython and Python</a>	9
<ul style="list-style-type: none"><li>• <a href="#">CircuitPython Microcontroller Wiring</a></li><li>• <a href="#">Python Computer Wiring</a></li><li>• <a href="#">Python Installation of VCNL4030 Library</a></li><li>• <a href="#">CircuitPython Usage</a></li><li>• <a href="#">Python Usage</a></li><li>• <a href="#">Example Code</a></li></ul>	
<a href="#">Python Docs</a>	14
<a href="#">Arduino</a>	14
<ul style="list-style-type: none"><li>• <a href="#">Wiring</a></li><li>• <a href="#">Library Installation</a></li><li>• <a href="#">Example Code</a></li></ul>	
<a href="#">Arduino Docs</a>	18
<a href="#">WipperSnapper</a>	18
<ul style="list-style-type: none"><li>• <a href="#">What is WipperSnapper</a></li><li>• <a href="#">Wiring</a></li><li>• <a href="#">Usage</a></li></ul>	
<a href="#">Downloads</a>	26
<ul style="list-style-type: none"><li>• <a href="#">Files</a></li><li>• <a href="#">Schematic and Fab Print</a></li></ul>	

---

# Overview



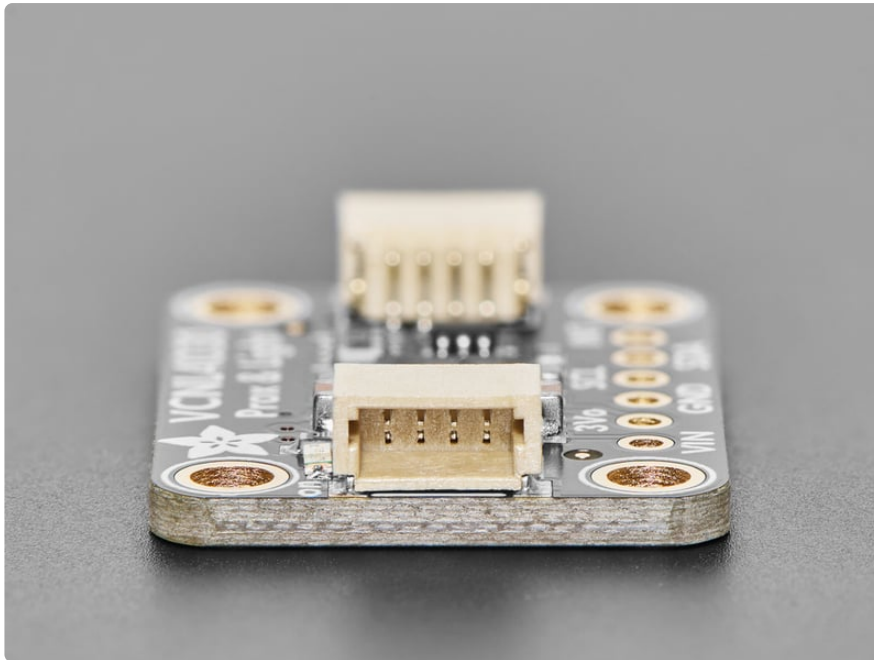
The VCNL4030 is a handy two-in-one sensor, with a proximity sensor that works from **0 to 300mm** (about 12 inches) and light sensor with range of **0.004 to 16,768 lux**.

We've all been there. That thing is close but how close? When you need to measure a small distance with reasonable accuracy, such as the rough height of particularly calm bumble bee, the VCNL4030 Proximity Sensor from Vishay can do that for you. If perchance you also needed to measure the amount of light at the same time, perhaps to let the bee to know if it's time for bed, you're in luck! The VCNL4030 can do that too (bumble bee not included, we tried putting it in the breathable anti-static bag but it started buzzing in a threatening manner).

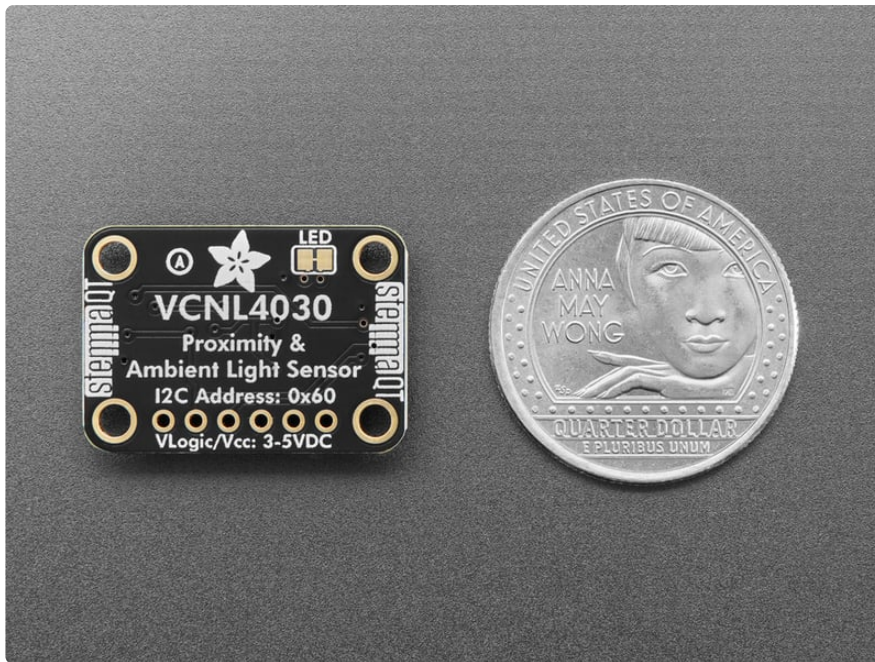


**Note** there's quite a few chips in the VCNL4xxx series, with varying ranges and prices. While the sensors look and sound similar, the firmware for these chips is not identical, so you cannot swap the VCNL4030 for VCNL4040 or others without recompiling with the respective libraries!

The VCNL4030 has a nice set of configuration knobs that let you tweak both the proximity sensor (PS) and light sensor (ALS) - for example integration time, dynamic range and sensitivity. Higher integration times mean slower conversion but you can measure dimmer light more precisely. Conversely shorter times mean fast conversion, and can measure brighter lights without oversaturating the sensor. For proximity you can also set up the duty cycle and built-in IR LED current. This lets you balance conversion speed, measurement range, precision and power usage as necessary.

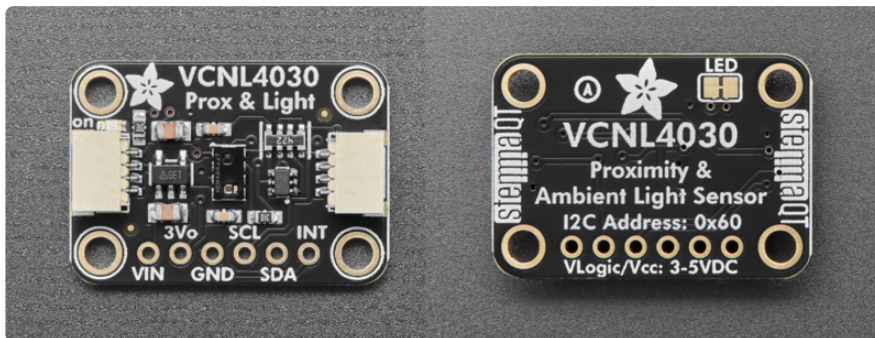


To make life easier, so you can focus on your important work, we've taken the VCNL4030 and put it onto a breakout PCB along with support circuitry to let you use this little wonder with 3.3V (Feather/Raspberry Pi) or 5V (Arduino/ Metro328) logic levels. Additionally, since it speaks I2C, you can easily connect it up with two wires (plus power and ground!). We've even included [SparkFun qwiic \(https://adafruit.it/Fpw\)](https://adafruit.it/Fpw) compatible [STEMMA QT \(https://adafruit.it/Ft4\)](https://adafruit.it/Ft4) connectors for the I2C bus so **you don't even need to solder!** [Just wire up to your favorite micro and you can use our CircuitPython/Python or Arduino drivers to easily interface with the VCNL4030 \(https://adafruit.it/1aBO\)](https://adafruit.it/1aBO) and make approximate approximations of proximity in no time! [QT Cable is not included, but we have a variety in the shop \(https://adafruit.it/17VE\)](https://adafruit.it/17VE).



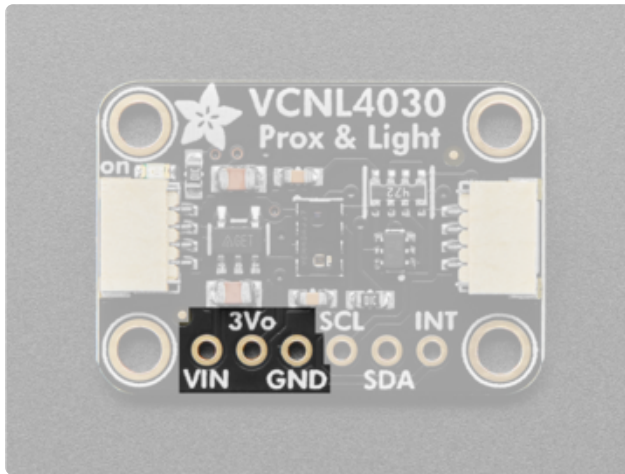
---

## Pinouts



The fixed I2C address is **0x60**.

## Power Pins



**Vin** - this is the power pin. Since the sensor chip uses 3VDC, we have included a voltage regulator on board that will take 3-5VDC and safely convert it to 3VDC. To power the board, give it the same power as the logic level of your microcontroller - e.g. for a 5V micro like Arduino, use 5V.

**3Vo** - this is the 3.3V output from the voltage regulator, you can grab up to 100mA from this if you like

**GND** - common ground for power and logic

## I2C Data Pins



**SCL** - the I2C clock pin, connect to your microcontroller's I2C clock line.

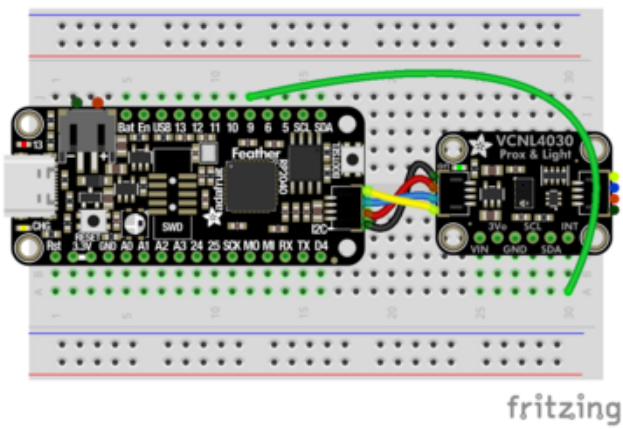
**SDA** - the I2C data pin, connect to your microcontroller's I2C data line.

**STEMMA QT** (<https://adafru.it/Ft4>) - These connectors allow you to connect to dev boards with **STEMMA QT** connectors or to other things with [various associated accessories](https://adafru.it/Ft6) (<https://adafru.it/Ft6>)

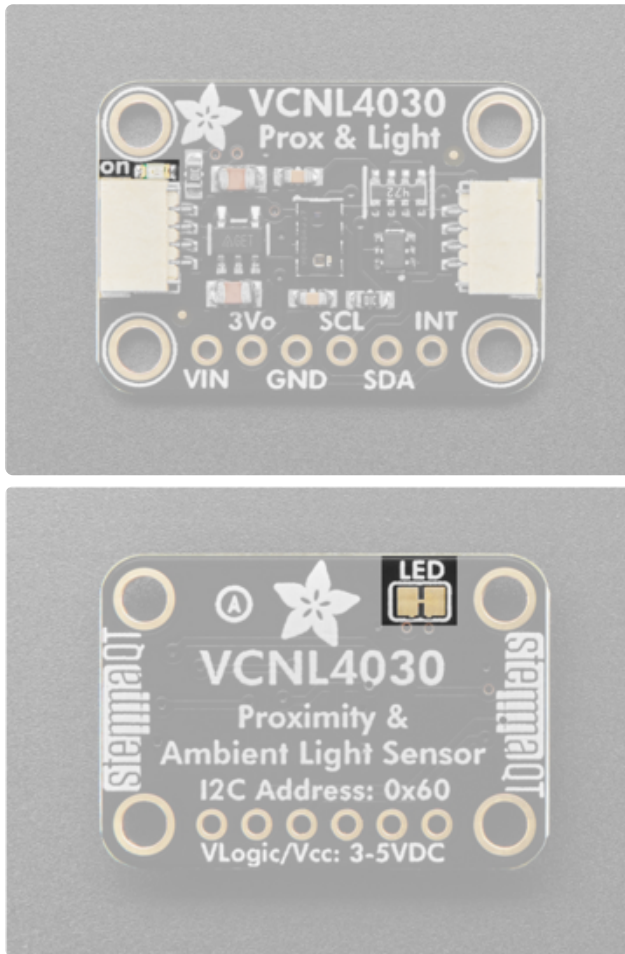
# Interrupt Pin



**INT** - the interrupt pin. It is an open-drain interrupt. To use, connect the pin to a GPIO pin on your microcontroller and configure as pull up.



## LED and LED Jumper



Above the left STEMMA QT connector there is a small green **ON** power indicator LED. When the breakout is powered this LED will turn on.

On the back of the VCNL4030 breakout there is a jumper with a small trace between two pads that can be cut to disable the **ON** power indicator LED on the board.

---

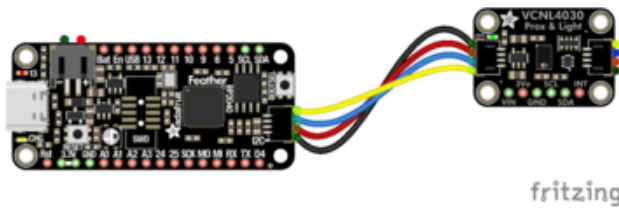
## CircuitPython and Python

It's easy to use the **VCNL4030** with Python or CircuitPython, and the [Adafruit\\_CircuitPython\\_VCNL4030](https://adafru.it/1aBP) (<https://adafru.it/1aBP>) module. This module allows you to easily write Python code to read data from the sensor.

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

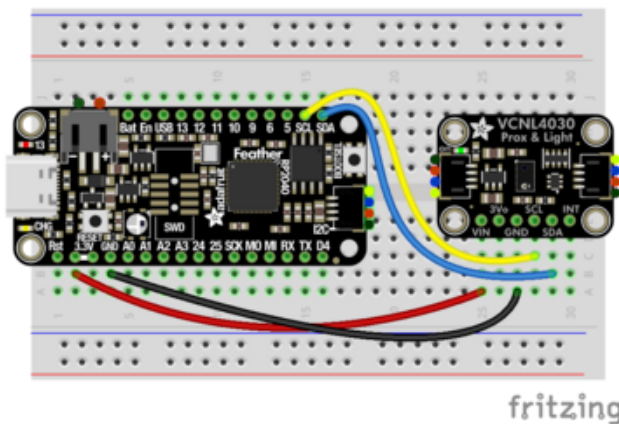
## CircuitPython Microcontroller Wiring

First wire up a VCNL4030 to your board exactly as shown below. Here's an example of wiring a Feather RP2040 to the sensor with I2C using one of the handy [STEMMA QT](https://adafru.it/Ft4) (<https://adafru.it/Ft4>) connectors:



- Board STEMMA 3V to sensor VIN (red wire)
- Board STEMMA GND to sensor GND (black wire)
- Board STEMMA SCL to sensor SCL (yellow wire)
- Board STEMMA SDA to sensor SDA (blue wire)

The following is the VCNL4030 wired to a Feather RP2040 using a solderless breadboard:

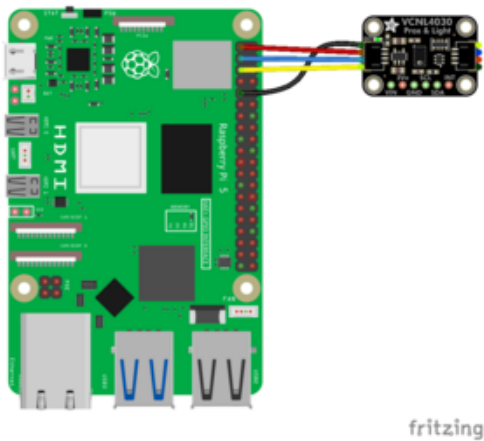


- Board 3V to sensor VIN (red wire)
- Board GND to sensor GND (black wire)
- Board SCL to sensor SCL (yellow wire)
- Board SDA to sensor SDA (blue wire)

## Python Computer Wiring

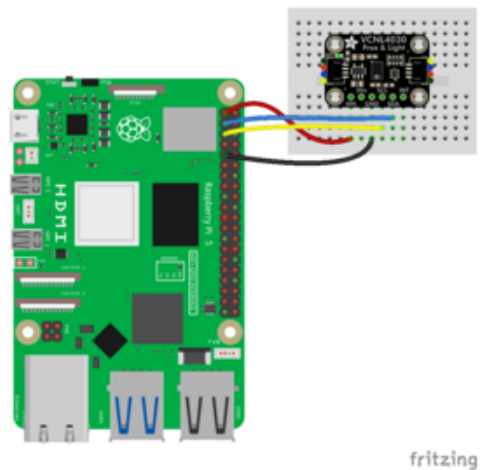
Since there are dozens of Linux computers/boards you can use, we will show wiring for Raspberry Pi. For other platforms, [please visit the guide for CircuitPython on Linux to see whether your platform is supported \(https://adafru.it/BSN\)](https://adafru.it/BSN).

Here's the Raspberry Pi wired with I2C using the STEMMA connector:



Pi 3V to sensor VIN (red wire)  
Pi GND to sensor GND (black wire)  
Pi SCL to sensor SCL (yellow wire)  
Pi SDA to sensor SDA (blue wire)

Here's the Raspberry Pi wired with I2C using a solderless breadboard:



Pi 3V to sensor VIN (red wire)  
Pi GND to sensor GND (black wire)  
Pi SCL to sensor SCL (yellow wire)  
Pi SDA to sensor SDA (blue wire)

## Python Installation of VCNL4030 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:

- `pip3 install adafruit-circuitpython-vcnl4030`

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!

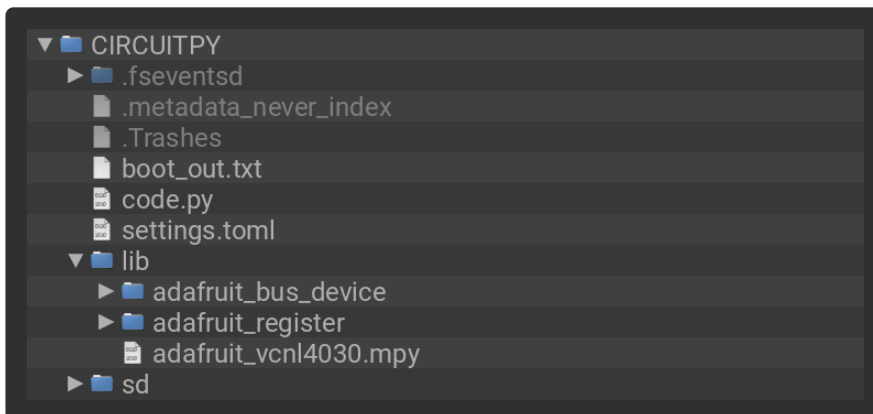
## CircuitPython Usage

To use with CircuitPython, you need to first install the **Adafruit\_CircuitPython\_VCNL4030** library, and its dependencies, into the **lib** folder on your **CIRCUITPY** drive. Then you need to update **code.py** with the example script.

Thankfully, we can do this in one go. In the example below, click the **Download Project Bundle** button below to download the necessary libraries and the **code.py** file in a zip file. Extract the contents of the zip file, and copy the **entire lib folder** and the **code.py** file to your **CIRCUITPY** drive.

Your **CIRCUITPY/lib** folder should contain the following folders and file:

- **adafruit\_bus\_device/**
- **adafruit\_register/**
- **adafruit\_vcnl4030.mpy**



## Python Usage

Once you have the library **pip3** installed on your computer, copy or download the following example to your computer, and run the following, replacing **code.py** with whatever you named the file:

```
python3 code.py
```

## Example Code

**If running CircuitPython:** Once everything is saved to the **CIRCUITPY** drive, [connect to the serial console \(https://adafru.it/Bec\)](https://adafru.it/Bec) to see the data printed out!

**If running Python:** The console output will appear wherever you are running Python.

```

1 # SPDX-FileCopyrightText: Copyright (c) 2026 Tim Cocks for Adafruit Industries
2 #
3 # SPDX-License-Identifier: MIT
4 """
5 Adafruit VCNL4030 Simple Test
6
7 Basic example for the Adafruit VCNL4030 library.
8 Reads proximity, ambient light, and white channel values.
9 """
10
11 import time
12
13 import board
14
15 from adafruit_vcnl4030 import VCNL4030
16
17 print("Adafruit VCNL4030 Simple Test")
18
19 sensor = VCNL4030(board.I2C())
20
21 print("VCNL4030 Found!")
22 print()
23
24 while True:
25     print(f"Proximity: {sensor.proximity} Lux: {sensor.lux:.2f} White: {sensor.white}")
26     time.sleep(1.0)

```

[https://github.com/adafruit/Adafruit\\_CircuitPython\\_VCNL4030/blob/main/examples/vcnl4030\\_simpletest.py](https://github.com/adafruit/Adafruit_CircuitPython_VCNL4030/blob/main/examples/vcnl4030_simpletest.py)

First, the sensor gets initialized over I2C. In the main loop the light and proximity data is read from the sensor and the values are printed to the serial console once per second.

```

code.py output:
Adafruit VCNL4030 Simple Test
VCNL4030 Found!

Proximity: 18 Lux: 24.35 White: 912
Proximity: 18 Lux: 24.38 White: 921
Proximity: 18 Lux: 26.88 White: 987
Proximity: 18 Lux: 73.73 White: 2292
Proximity: 17 Lux: 96.03 White: 2824
Proximity: 17 Lux: 124.16 White: 3720
Proximity: 68 Lux: 113.28 White: 3466
Proximity: 76 Lux: 111.55 White: 3410
Proximity: 82 Lux: 96.86 White: 3002
Proximity: 18 Lux: 104.22 White: 3183
Proximity: 17 Lux: 90.56 White: 2746
Proximity: 17 Lux: 92.13 White: 2811
Proximity: 17 Lux: 83.14 White: 2532
Proximity: 15 Lux: 32.96 White: 1158
Proximity: 18 Lux: 27.20 White: 987
Proximity: 17 Lux: 24.26 White: 920

```

---

# Python Docs

[Python Docs \(https://adafru.it/1aBM\)](https://adafru.it/1aBM)

---

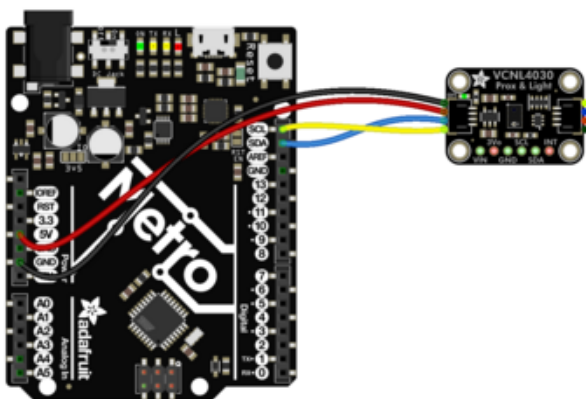
## Arduino

Using the VCNL4030 breakout with Arduino involves wiring up the breakout to your Arduino-compatible microcontroller, installing the [Adafruit\\_VCNL4030 \(https://adafru.it/1aBQ\)](https://adafru.it/1aBQ) library, and running the provided example code.

### Wiring

Wire as shown for a **5V** board like an Uno. If you are using a **3V** board, like an Adafruit Feather, wire the board's 3V pin to the sensor **VIN**.

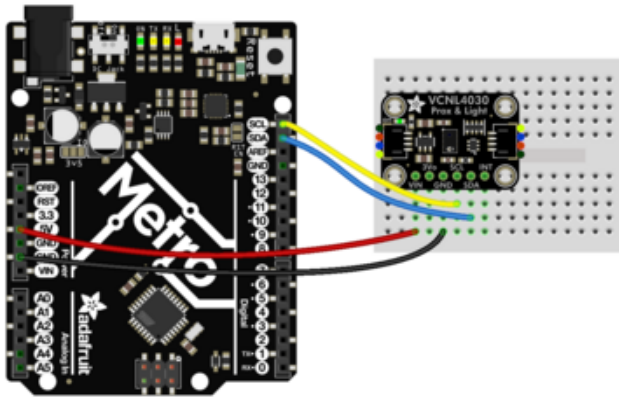
Here is an Adafruit Metro wired up to the sensor using the STEMMA QT connector:



Board 5V to breakout VIN (red wire)  
Board GND to breakout GND (black wire)  
Board SCL to breakout SCL (yellow wire)  
Board SDA to breakout SDA (blue wire)

fritzing

Here is an Adafruit Metro wired up using a solderless breadboard:

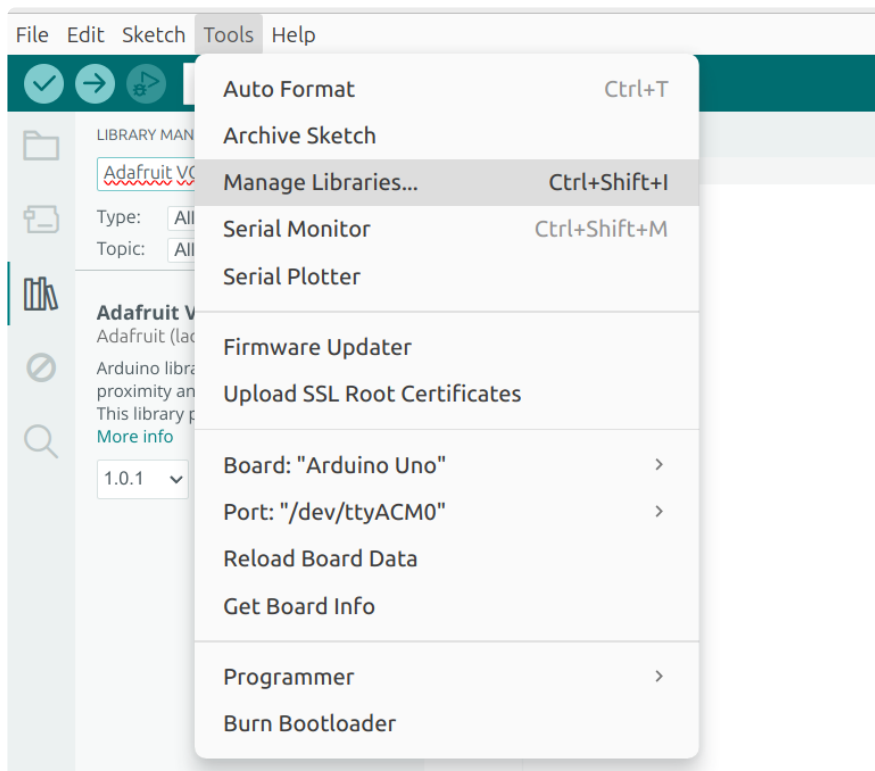


- Board 5V to breakout VIN (red wire)
- Board GND to breakout GND (black wire)
- Board SCL to breakout SCL (yellow wire)
- Board SDA to breakout SDA (blue wire)

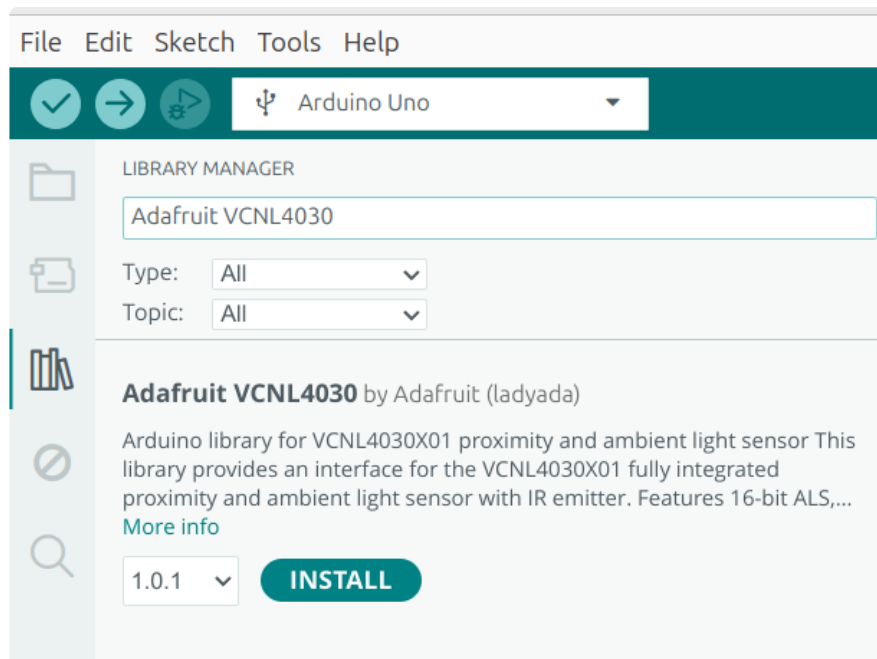
fritzing

## Library Installation

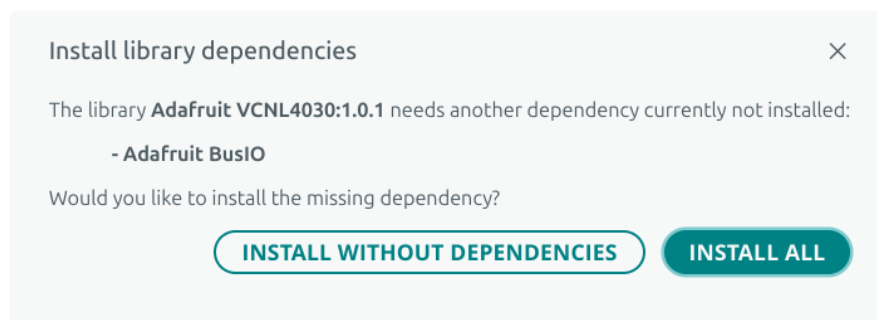
You can install the **Adafruit VCNL4030** library for Arduino using the Library Manager in the Arduino IDE.



Click the **Manage Libraries ...** menu item, search for **Adafruit VCNL4030**, and select the **Adafruit VCNL4030** library:



If asked about dependencies, click "Install all".



If the "Dependencies" window does not come up, then you already have the dependencies installed.



If the dependencies are already installed, you must make sure you update them through the Arduino Library Manager before loading the example!

## Example Code

```
1  /*!  
2   * @file simpletest.ino  
3   *  
4   * Basic test for the Adafruit VCNL4030 library.  
5   * Reads proximity, ambient light, and white channel values.  
6   *  
7   * Written by Limor Fried/Ladyada for Adafruit Industries.  
8   */  
9  
10 #include <Adafruit_VCNL4030.h>  
11  
12 Adafruit_VCNL4030 vcnl;  
13  
14 void setup() {  
15   Serial.begin(115200);  
16   while (!Serial)  
17     delay(10);  
18  
19   Serial.println(F("Adafruit VCNL4030 Simple Test"));  
20  
21   if (!vcnl.begin()) {  
22     Serial.println(F("Could not find a valid VCNL4030 sensor, check wiring!"));  
23     while (1) {  
24       delay(10);  
25     }  
26   }  
27  
28   Serial.println(F("VCNL4030 Found!"));  
29   Serial.println();  
30 }  
31  
32 void loop() {  
33   uint16_t prox = vcnl.readProximity();  
34   uint16_t als = vcnl.readALS();  
35   uint16_t white = vcnl.readWhite();  
36  
37   Serial.print(F("Proximity: "));  
38   Serial.print(prox);  
39   Serial.print(F(" Lux: "));  
40   Serial.print(vcnl.readLux());  
41   Serial.print(F(" White: "));  
42   Serial.println(white);  
43  
44   delay(100);  
45 }
```

[https://github.com/adafruit/Adafruit\\_VCNL4030/blob/main/examples/simpletest/simpletest.ino](https://github.com/adafruit/Adafruit_VCNL4030/blob/main/examples/simpletest/simpletest.ino)

Upload the sketch to your board and open up the Serial Monitor (**Tools -> Serial Monitor**) at 115200 baud. You'll see the VCNL4030 recognized over I2C. Then, the reading values of different light channels will be printed out to the Serial Monitor.

```
Output Serial Monitor x
Message (Enter to send message to 'Arduino Uno' on '/dev/ttyUSB0')
Adafruit VCNL4030 Simple Test
VCNL4030 Found!
Proximity: 66 Lux: 1237.50 White: 17092
Proximity: 68 Lux: 1266.30 White: 17475
Proximity: 65 Lux: 1279.49 White: 17660
Proximity: 64 Lux: 1300.99 White: 17660
Proximity: 66 Lux: 1300.99 White: 17881
Proximity: 63 Lux: 1296.96 White: 17781
Proximity: 64 Lux: 1043.14 White: 14160
Proximity: 65 Lux: 455.87 White: 6042
Proximity: 66 Lux: 102.27 White: 1700
Proximity: 76 Lux: 48.77 White: 838
Proximity: 120 Lux: 45.25 White: 808
Proximity: 189 Lux: 41.22 White: 734
Proximity: 273 Lux: 35.90 White: 662
Proximity: 353 Lux: 32.64 White: 602
Proximity: 413 Lux: 31.10 White: 571
Proximity: 452 Lux: 30.66 White: 566
Proximity: 405 Lux: 30.34 White: 552
Proximity: 453 Lux: 30.40 White: 564
Proximity: 343 Lux: 29.31 White: 544
Proximity: 140 Lux: 31.62 White: 592
Proximity: 65 Lux: 39.68 White: 722
Proximity: 65 Lux: 47.30 White: 832
Proximity: 65 Lux: 47.23 White: 822
Proximity: 64 Lux: 46.53 White: 819
```

# Arduino Docs

[Arduino Docs \(https://adafru.it/1aBN\)](https://adafru.it/1aBN)

# WipperSnapper

Shop Learn Blog Forums IO LIVE! AdaBox Hi, Tyeth DEMO Gundry | Account 0

adafruit Devices Feeds Dashboards Actions Power-Ups New Device

tyeth\_demo / Devices / Adafruit Feather ESP32 V2 Help Settings

New Component Auto-Config I2C Scan

VCNL4030: Light Sensor  Value: 2667.01

VCNL4030: Proximity Sensor  Raw Value: 210.00

Adafruit Feather ESP32 V2 by Adafruit  
 Online v1.0.0-beta.127 Docs Purchase Report Bugs

What is Adafruit IO? IO Status Get Help Learn Quick Guides IO Plus API Documentation News

*"Every great dream begins with a dreamer"*  
 – Harriet Tubman

## What is WipperSnapper

WipperSnapper is a firmware designed to turn any WiFi-capable board into an Internet-of-Things device without programming a single line of code. WipperSnapper connects to [Adafruit IO \(https://adafru.it/fsU\)](https://adafru.it/fsU), a web platform designed ([by Adafruit! \(https://adafru.it/Bo5\)](https://adafru.it/Bo5)) to display, respond, and interact with your project's data.

Simply load the WipperSnapper firmware onto your board, add credentials, and plug it into power. Your board will automatically register itself with your Adafruit IO account.

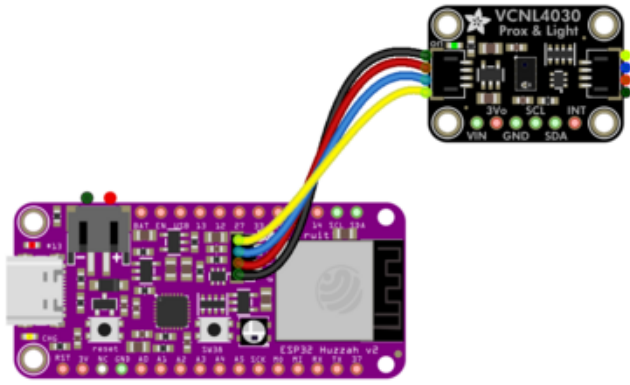
From there, you can add components to your board such as buttons, switches, potentiometers, sensors, and more! Components are dynamically added to hardware, so you can immediately start interacting, logging, and streaming the data your projects produce without writing code.

If you've never used WipperSnapper, click below to read through the quick start guide before continuing.

<https://adafru.it/Vfd>

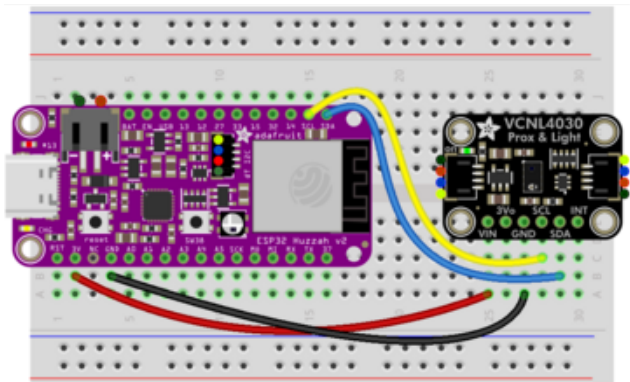
## Wiring

Wire up the VCNL4030 exactly as follows. Here it's shown connected via our convenient StemmaQT cable, or alternatively wired up on a prototyping breadboard:



fritzing

- Board 3V to sensor VIN (red wire)
- Board GND to sensor GND (black wire)
- Board SCL to sensor SCL (yellow wire)
- Board SDA to sensor SDA (blue wire)

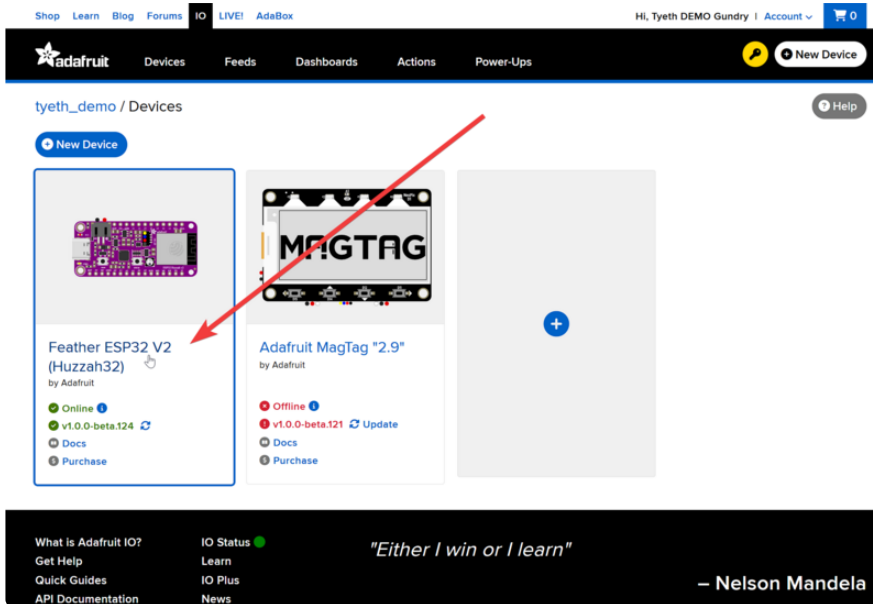


fritzing

## Usage

Connect your board to Adafruit IO Wippersnapper and [navigate to the WipperSnapper board list \(https://adafru.it/TAu\)](https://adafru.it/TAu).

On this page, select the WipperSnapper board you're using to be brought to the board's interface page.



If you do not see your board listed here - you need [to connect your board to Adafruit IO](https://adafru.it/Vfd) (<https://adafru.it/Vfd>) first.

## Feather ESP32 V2 (Huzzah32)

by Adafruit

- ✓ Online ⓘ
- ✓ v1.0.0-beta.124 ↻
- 📖 Docs
- 💰 Purchase

## Adafruit MagTag "2.9"

by Adafruit

- ✓ Online ⓘ
- ! v1.0.0-beta.122 ↻ Update
- 📖 Docs
- 💰 Purchase

On the device page, quickly **check** that you're running the latest version of the WipperSnapper firmware.

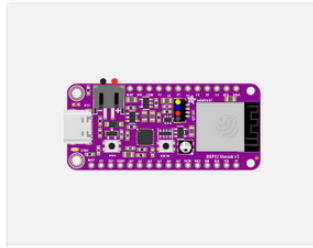
The device tile on the left indicates the version number of the firmware running on the connected board.

If the firmware version is green with a checkmark - continue with this guide. If the firmware version is red with an exclamation mark "!" - [update to the latest WipperSnapper firmware](https://adafru.it/Vfd) (<https://adafru.it/Vfd>) on your board before continuing.

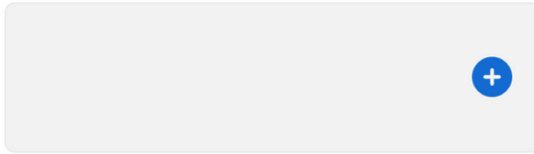
Next, make sure the sensor is plugged into your board and click the **I2C Scan** button.

brubell / Devices / Adafruit Feather ESP32 V2

New Component I2C Scan Device Settings



Adafruit Feather ESP32...  
Adafruit Feather ESP32 V2 by Adafruit



You should see the VCNL4030's default I2C address of `0x60` pop-up in the I2C scan list.

### I2C Scan Complete



	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00								--	--	--	--	--	--	--	--	--
10	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
20	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
30	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
40	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
50	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
60	60	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
70	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Close Scan Again

## I don't see the sensor's I2C address listed!

First, double-check the connection and/or wiring between the sensor and the board.

Then, reset the board and let it re-connect to Adafruit IO WipperSnapper.

With the sensor detected in an I2C scan, you're ready to add the sensor to your board.

Click the **New Component** button or the **+** button to bring up the component picker.

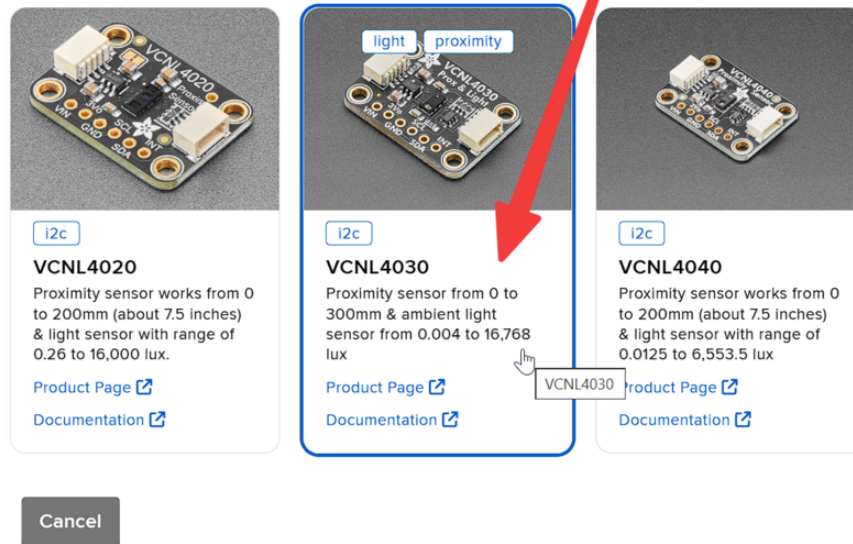


Adafruit IO supports a large amount of components. To quickly find your sensor, type **VCNL4030** into the search bar, then select the **VCNL4030** component.

Which component would you like to set up?

✕ VCNL40

Displaying 3 matching Components.



The screenshot shows a selection interface for components. At the top, a search bar contains 'VCNL40'. Below it, three component cards are displayed:

- VCNL4020**: Proximity sensor works from 0 to 200mm (about 7.5 inches) & light sensor with range of 0.26 to 16,000 lux. Includes 'i2c' tag, 'Product Page', and 'Documentation' links.
- VCNL4030**: Proximity sensor from 0 to 300mm & ambient light sensor from 0.004 to 16,768 lux. Includes 'i2c' tag, 'Product Page', and 'Documentation' links. This card is highlighted with a blue border and a red arrow points to it. A tooltip labeled 'VCNL4030' is visible over the 'Product Page' link.
- VCNL4040**: Proximity sensor works from 0 to 200mm (about 7.5 inches) & light sensor with range of 0.0125 to 6,553.5 lux. Includes 'i2c' tag, 'Product Page', and 'Documentation' links.

A 'Cancel' button is located at the bottom left of the component selection area.

On the component configuration page, the VCNL4030's sensor address should be listed along with the sensor's settings.

The **Send Every** option is specific to each sensor's measurements. This option will tell the Feather how often it should read from the VCNL4030 sensor and send the data to Adafruit IO. Measurements can range from every second to every 24 hours.

For this example, set the **Send Every** interval to every 30 seconds.

## Create VCNL4030 Component



Select I2C Address  
0x60

Enable VCNL4030: Light Sensor?

Name:  
VCNL4030: Light Sensor

Send Data:  
Every 30 seconds

Enable VCNL4030: Proximity Sensor?

Name:  
VCNL4030: Proximity Sensor

Send Data:  
Every 30 seconds



[← Back to Component Type](#)

Create Component

Your device interface should now show the sensor components you created. After the interval you configured elapses, WipperSnapper will automatically read values from the sensor(s) and send them to Adafruit IO.

The screenshot shows the Adafruit IO dashboard for a device named 'tyeth\_demo'. The device is an 'Adafruit Feather ESP32 V2'. Two sensor components are visible:

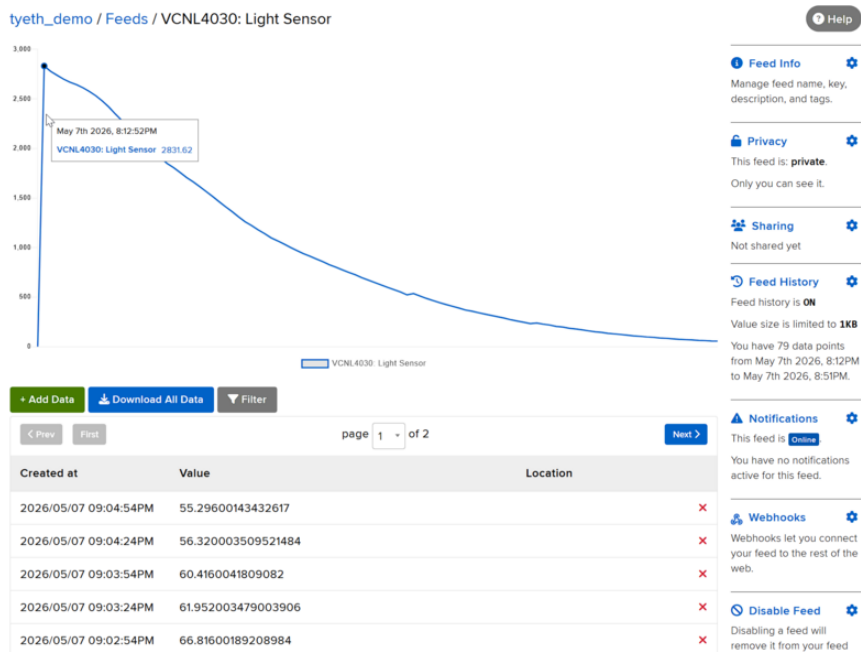
- VCNL4030: Light Sensor** (id: vcnl4030:light) with a current value of 1846.53.
- VCNL4030: Proximity Sensor** (id: vcnl4030:proximity) with a current raw value of 286.00.

The dashboard also includes navigation links for 'New Component', 'Auto-Config', and 'I2C Scan', as well as a 'Report Bugs' link. At the bottom, there is a footer with 'What is Adafruit IO?', 'IO Status', and a quote: "Tomorrow belongs to those who can hear it coming" - David Bowie.

To view the data that has been logged from the sensor, click on the graph next to the sensor name.



Here you can see the feed history and edit things about the feed such as the name, privacy, webhooks associated with the feed and more. If you want to learn more about how feeds work, [check out this page \(https://adafru.it/10aZ\)](https://adafru.it/10aZ).



## Downloads

### Files

- [VCNL4030 Datasheet \(https://adafru.it/1aBR\)](https://adafru.it/1aBR)
- [EagleCAD PCB files on GitHub \(https://adafru.it/1aBS\)](https://adafru.it/1aBS)
- [Fritzing object in the Adafruit Fritzing Library \(https://adafru.it/1aBT\)](https://adafru.it/1aBT)

# Schematic and Fab Print

