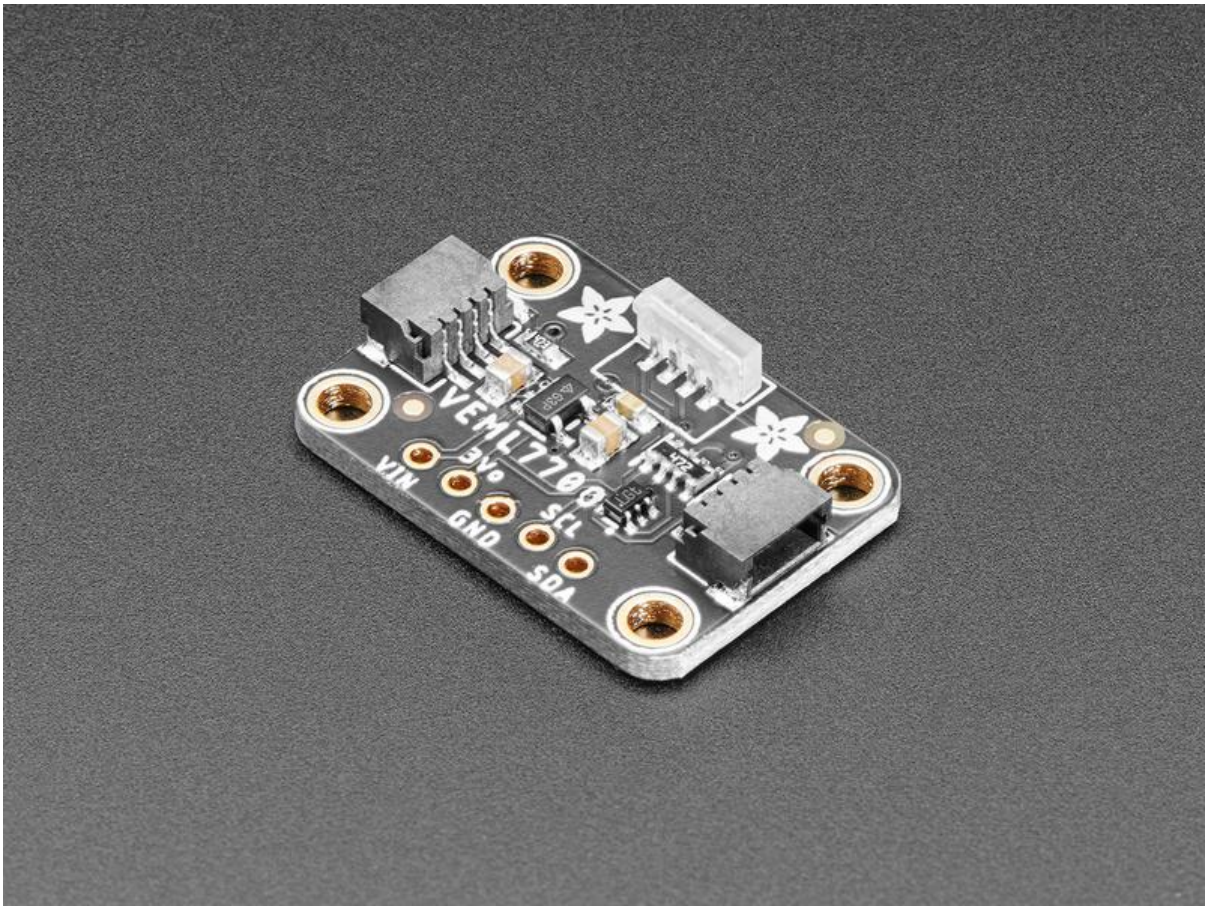
Adafruit VEML7700 Ambient Light Sensor

Created by Kattni Rembor



https://learn.adafruit.com/adafruit-veml7700

Last updated on 2022-02-16 03:54:54 PM EST

Table of Contents

Overview 3

Pinouts 5

- Power Pins 6
- I2C Logic Pins 6

Assembly 7

- Prepare the header strip: 8
- Add the breakout board: 8
- And Solder! 9

Arduino 10

- Wiring 10
- Installation 11
- Load Example 11
- Example Code 12

Arduino Docs 13

Python & CircuitPython 13

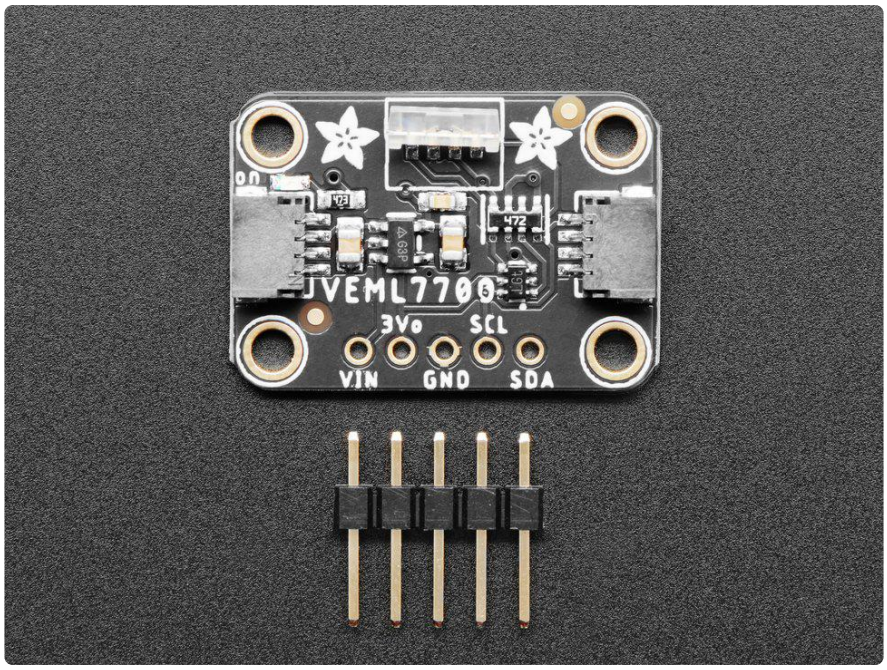
- CircuitPython Microcontroller Wiring 13
- Python Computer Wiring 14
- CircuitPython Installation of VEML7700 Library 15
- Python Installation of VEML7700 Library 16
- CircuitPython & Python Usage 16
- Full Example Code 17

Python Docs 18

Downloads 18

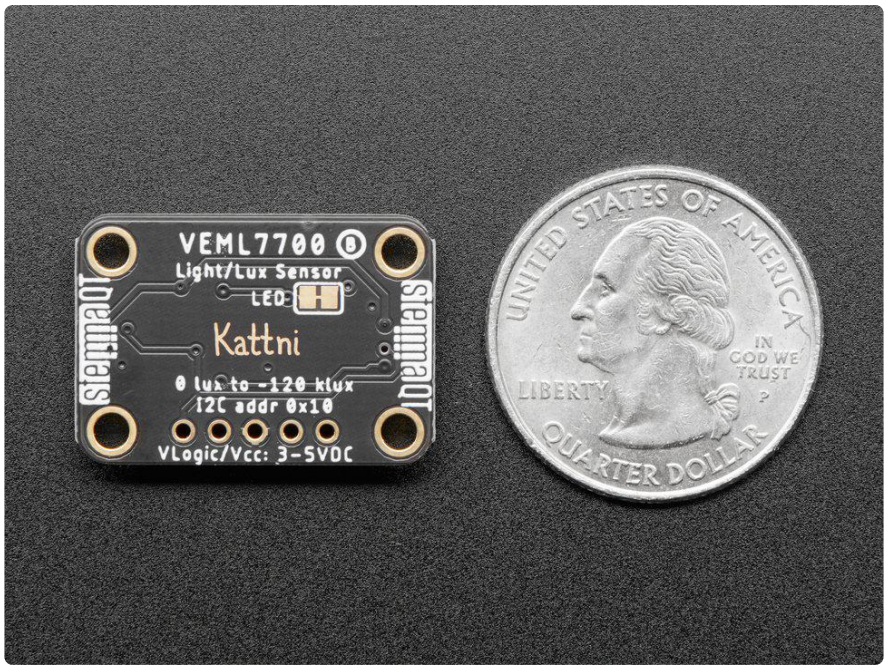
- Files 18
- Schematic and Fab Print for STEMMA QT Version 18
- Schematic and Fab Print for Original Version 19

Overview



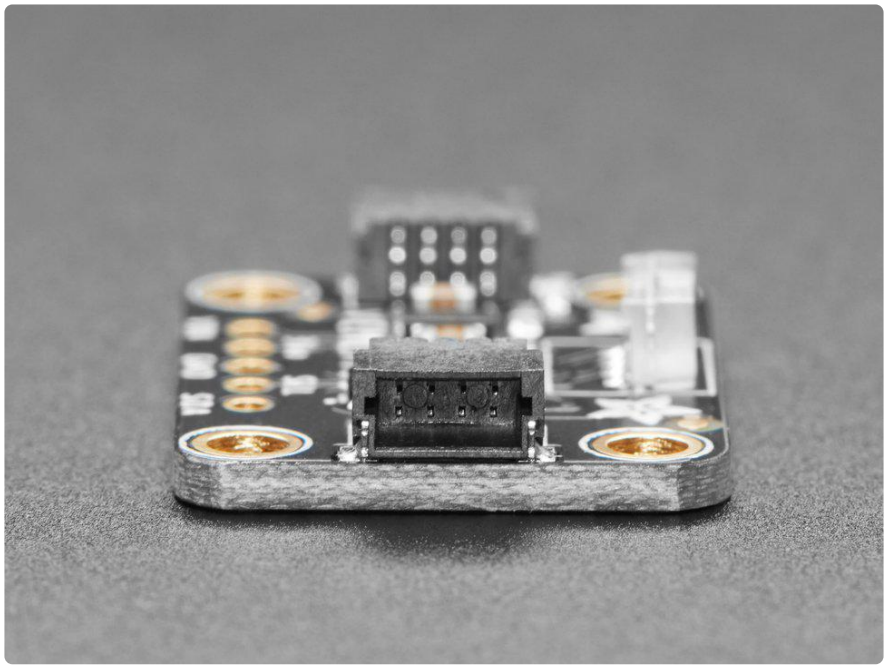
Vishay has a lot of light sensors out there, and this is a nice simple lux sensor that's easy to add to any microcontroller. Most light sensors just give you a number for brighter/darker ambient lighting. The VEML7700 makes your life easier by calculating the lux, which is an SI unit for light. You'll get more consistent readings between multiple sensors because you aren't dealing with some unitless values.

The sensor has 16-bit dynamic range for ambient light detection from 0 lux to about 120k lux with resolution down to 0.0036 lx/ct, with software-adjustable gain and integration times.



Interfacing is easy - this sensor uses plain, universal I2C. We put this sensor on a breakout board with a 3.3V regulator and logic level shifter so you can use it with 3.3V or 5V power/logic microcontrollers. We have written libraries for Arduino (https://adafru.it/EoE) (C/C++) as well as CircuitPython (Python 3) (https://adafru.it/EoF) so you can use this sensor with just about any kind of device, even a Raspberry Pi!

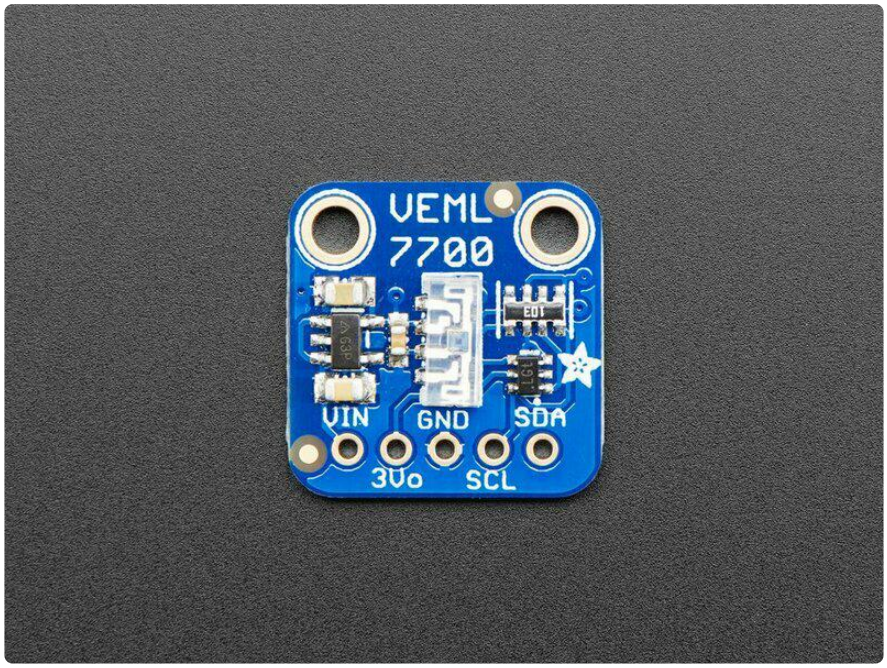
This is Kattni's first PCB design for Adafruit, it's even signed on the back!



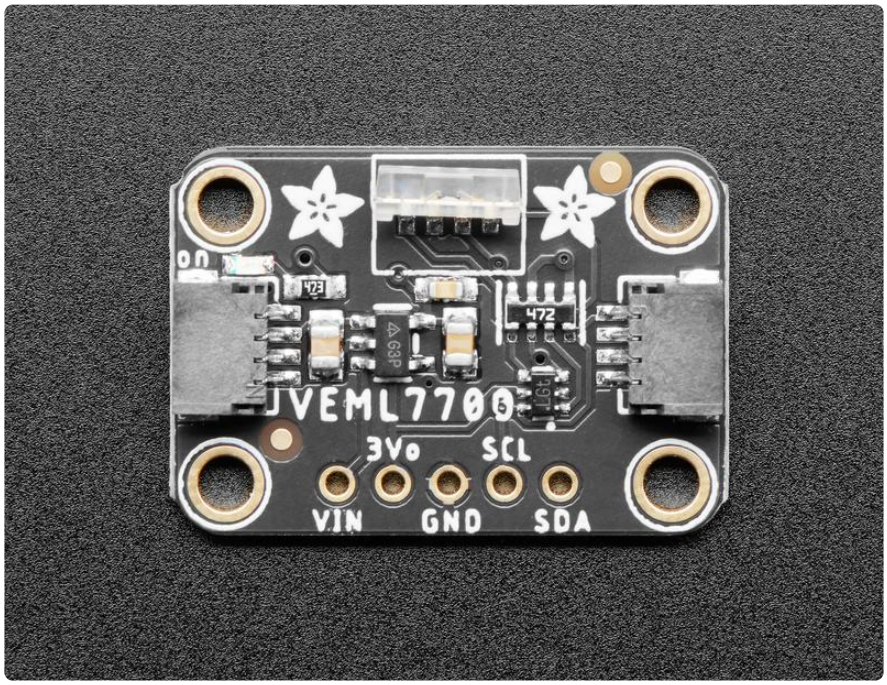
As with all Adafruit breakouts, we've done the work to make this handy light sensor super easy to use. We've put it on a breakout board with the required support circuitry and connectors to make it easy to work with. Since I2C is supported, we've

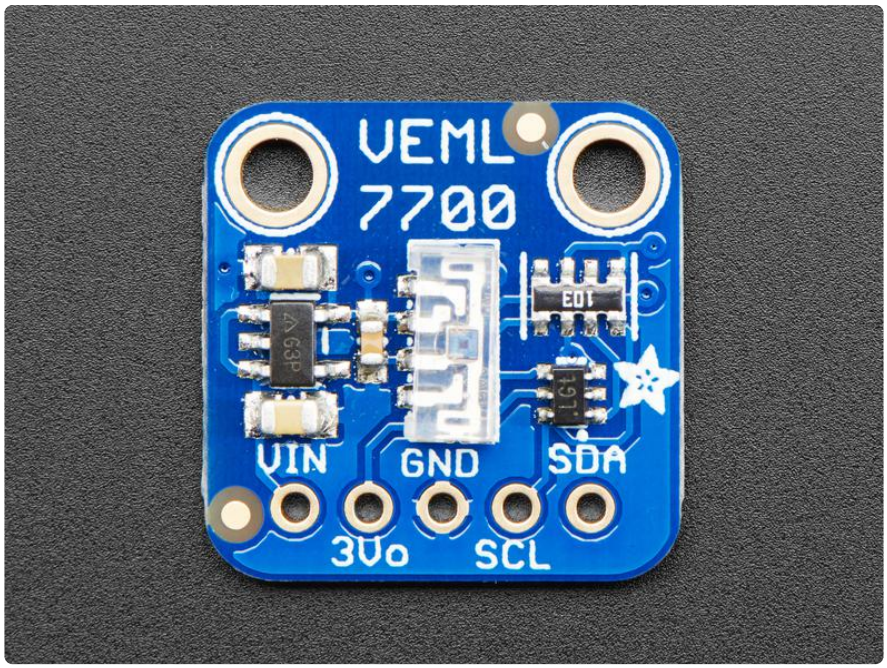
added SparkFun Qwiic (https://adafru.it/Fpw) compatible STEMMA QT (https://adafru.it/Ft4) JST SH connectors that allow you to get going without needing to solder. Just use a STEMMA QT adapter cable (https://adafru.it/FA-), plug it into your favorite microcontroller or Blinka supported SBC and you're ready to rock! QT Cable is not included, but we have a variety in the shop (https://adafru.it/JnB).

There are two versions of this board - the STEMMA QT version shown above, and the original header-only version shown below. Code works the same on both!



Pinouts



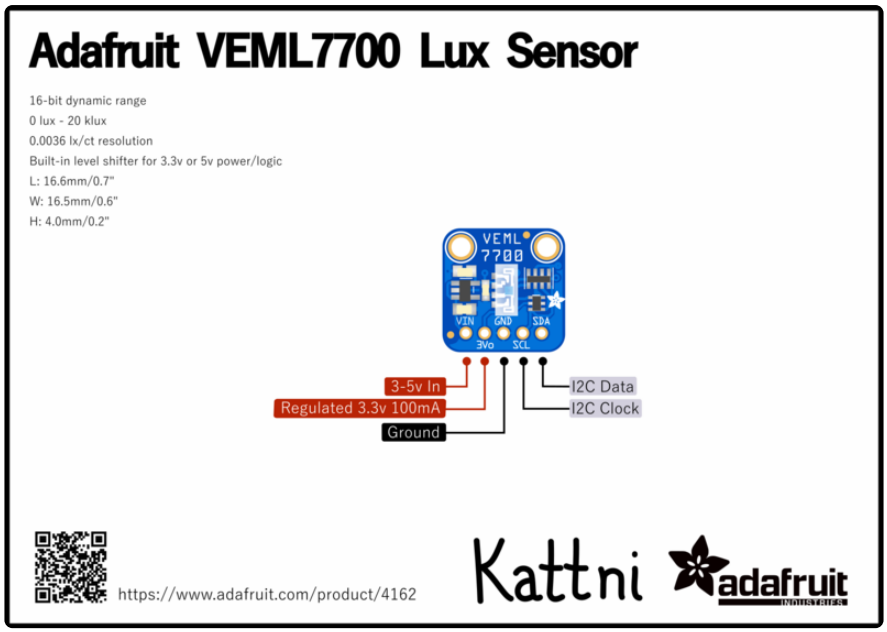


Power Pins

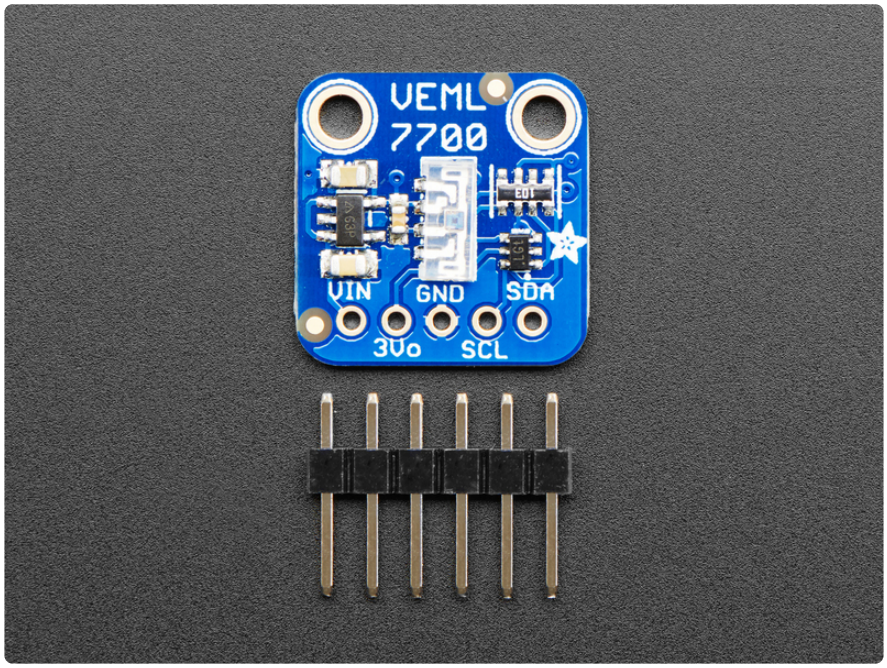
- Vin - this is the power pin. Since the sensor chip uses 3 VDC, we have included a voltage regulator on board that will take 3-5VDC and safely convert it down. To power the board, give it the same power as the logic level of your microcontroller - e.g. for a 5V micro like Arduino, use 5V
- 3Vo - this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

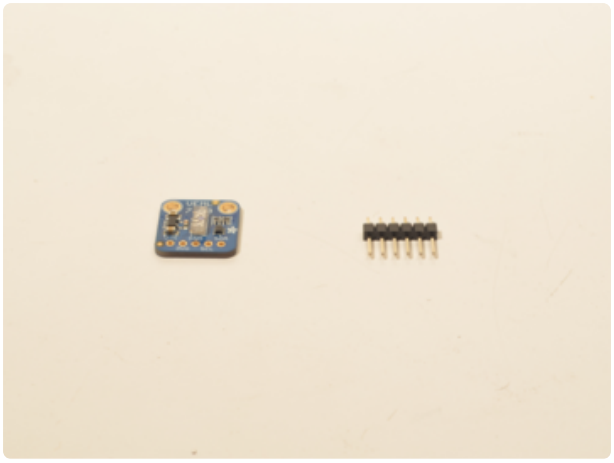
I2C Logic Pins

- SCL - this is the I2C clock pin, connect to your microcontroller's I2C clock line.
- SDA - this is 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 development boards with STEMMA QT connectors, or to other things, with various associated accessories (https://adafru.it/Ft6).


Adafruit VEML7700 Lux Sensor
16-bit dynamic range
0 lux - 20 klux
0.0036 lx/ct resolution
Built-in level shifter for 3.3v or 5v power/logic
L: 16.6mm/0.7"
W: 16.5mm/0.6"
H: 4.0mm/0.2"
VEML
7700
VIN
GND
SDA
3Vo
SCL
3-5v In
Regulated 3.3v 100mA
Ground
I2C Data
I2C Clock
https://www.adafruit.com/product/4162
Kattni
adafruit

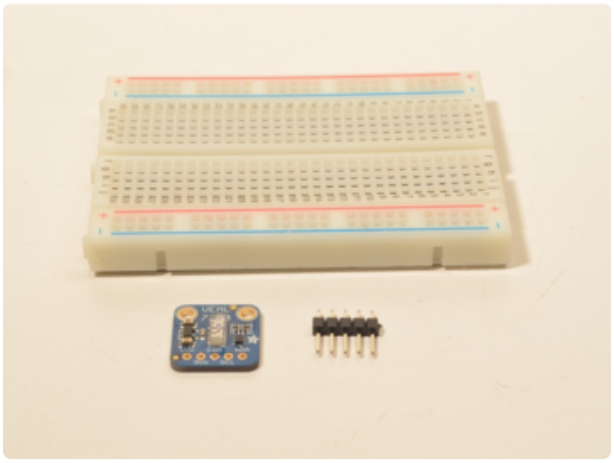
Assembly

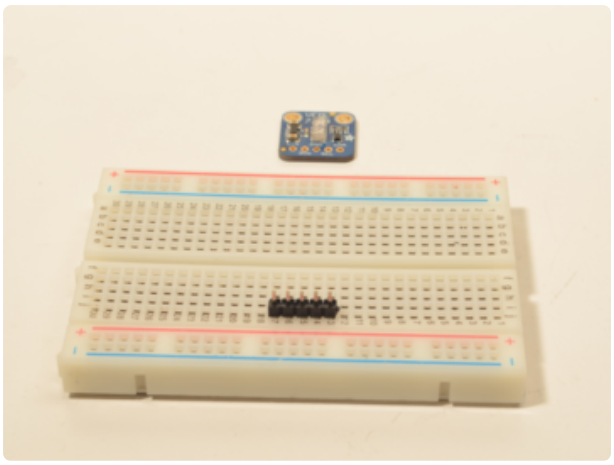

VEML
7700
VIN
GND
SDA
3Vo
SCL

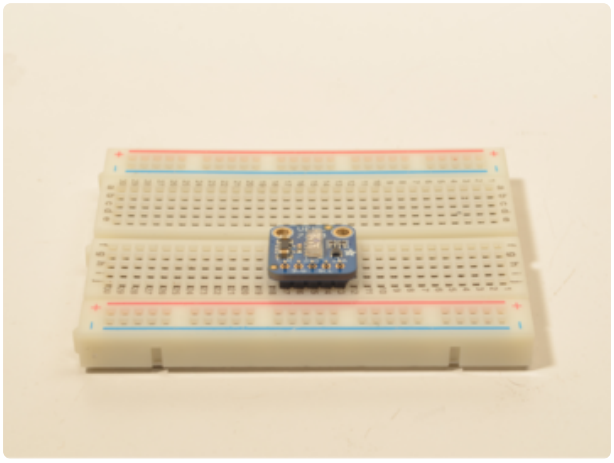


Prepare the header strip:

Cut the strip to length if necessary. It will be easier to solder if you insert it into a breadboard - long pins down

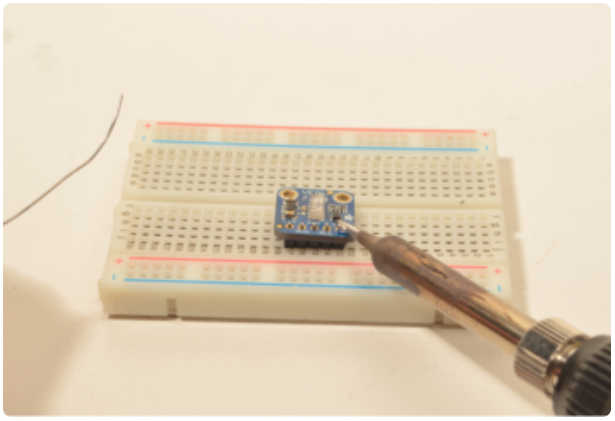


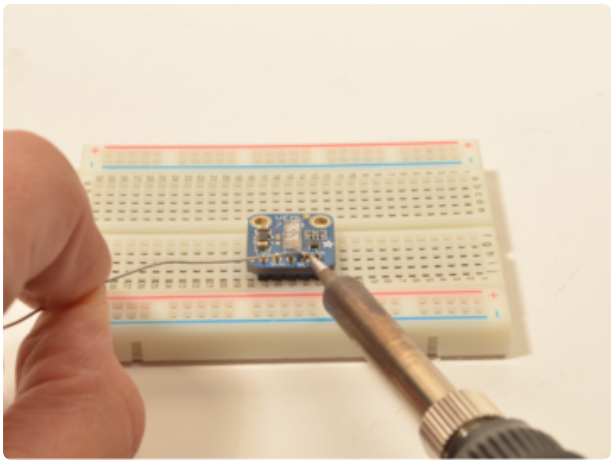


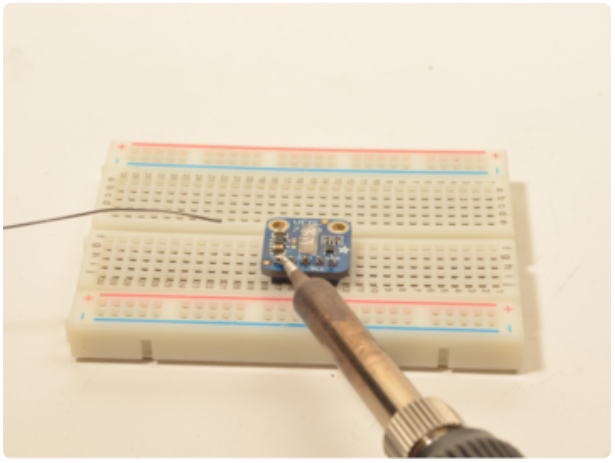


Add the breakout board:

Place the breakout board over the pins so that the short pins poke through the breakout pads



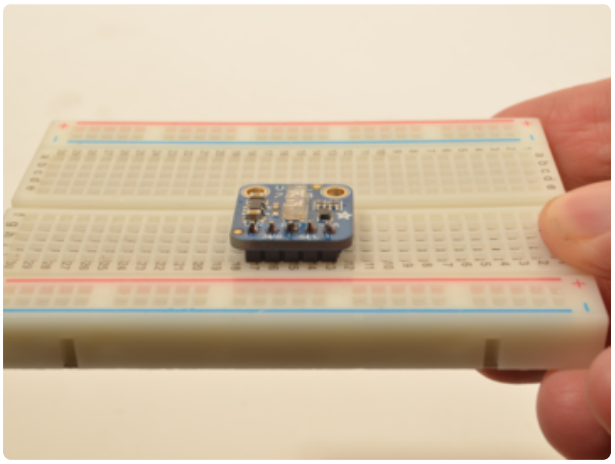




And Solder!

Be sure to solder all 5 pins for reliable electrical contact.

(For tips on soldering, be sure to check out our Guide to Excellent Soldering (https://adafru.it/aTk)).

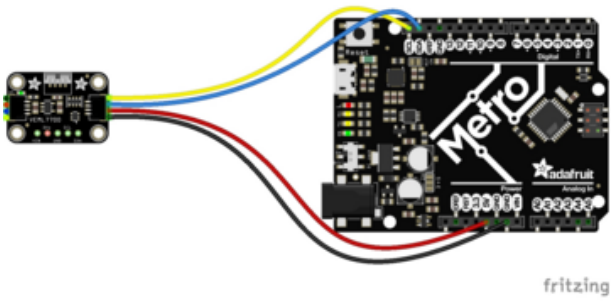


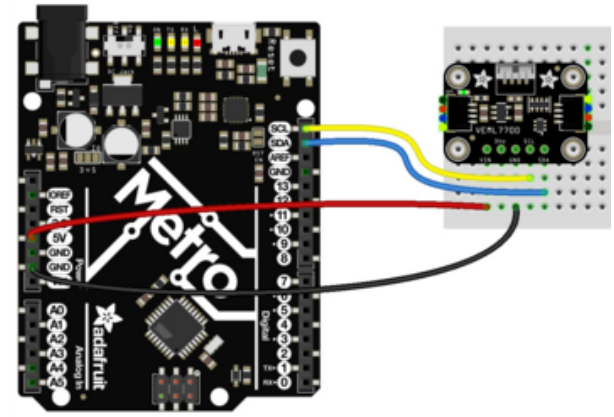
You're done! Check your solder joints visually and continue onto the next steps

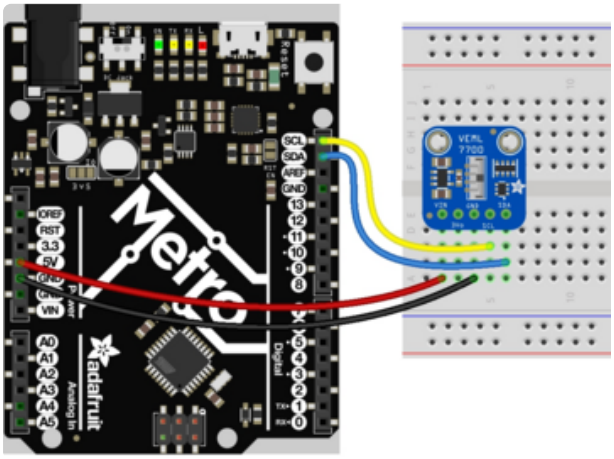
Arduino

Wiring

Connecting the VEML7700 to your Feather or Arduino is easy:





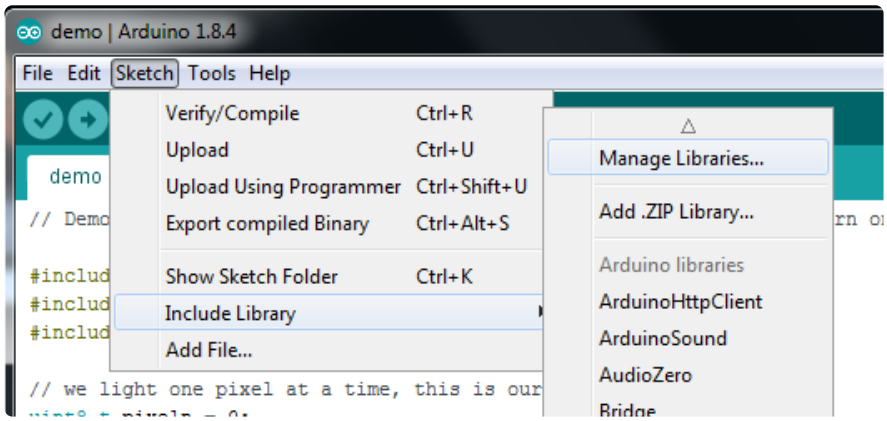


- If you are running a Feather (3.3V), connect Feather 3V to board VIN (red wire on STEMMA QT version)
- If you are running a 5V Arduino (Uno, etc.), connect Arduino 5V to board VIN (red wire on STEMMA QT version)
- Connect Feather or Arduino GND to board GND (black wire on STEMMA QT version)
- Connect Feather or Arduino SCL to board SCL (yellow wire on STEMMA QT version)
- Connect Feather or Arduino SDA to board SDA (blue wire on STEMMA QT version)

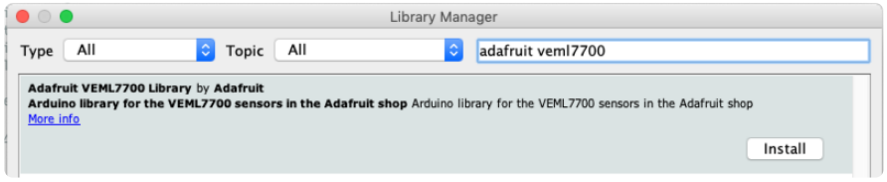
The final results should resemble the illustration above, showing an Adafruit Metro development board.

Installation

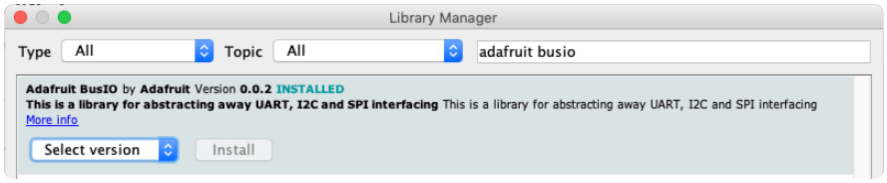
You can install the Adafruit VEML7700 Library for Arduino using the Library Manager in the Arduino IDE:



Click the Manage Libraries ... menu item, search for Adafruit VEML7700, and select the Adafruit VEML7700 library:



Then follow the same process for the Adafruit BusIO library.



Load Example

Open up File -> Examples -> Adafruit VEML7700 -> veml7700_test and upload to your Arduino wired up to the sensor.

Upload the sketch to your board and open up the Serial Monitor (Tools->Serial Monitor). You should see the the values for Lux, white light, and raw ambient light levels.

Example Code

The following example code is part of the standard library, but illustrates how you can retrieve sensor data from the VEML7700 for the Lux, white light and raw ambient light values:

```
#include "Adafruit_VEML7700.h"

Adafruit_VEML7700 veml = Adafruit_VEML7700();

void setup() {
  while (!Serial) { delay(10); }
  Serial.begin(115200);
  Serial.println("Adafruit VEML7700 Test");

  if (!veml.begin()) {
    Serial.println("Sensor not found");
    while (1);
  }
  Serial.println("Sensor found");

  veml.setGain(VEML7700_GAIN_1);
  veml.setIntegrationTime(VEML7700_IT_800MS);

  Serial.print(F("Gain: "));
  switch (veml.getGain()) {
    case VEML7700_GAIN_1: Serial.println("1"); break;
    case VEML7700_GAIN_2: Serial.println("2"); break;
    case VEML7700_GAIN_1_4: Serial.println("1/4"); break;
    case VEML7700_GAIN_1_8: Serial.println("1/8"); break;
  }

  Serial.print(F("Integration Time (ms): "));
  switch (veml.getIntegrationTime()) {
    case VEML7700_IT_25MS: Serial.println("25"); break;
    case VEML7700_IT_50MS: Serial.println("50"); break;
    case VEML7700_IT_100MS: Serial.println("100"); break;
    case VEML7700_IT_200MS: Serial.println("200"); break;
    case VEML7700_IT_400MS: Serial.println("400"); break;
    case VEML7700_IT_800MS: Serial.println("800"); break;
  }

  //veml.powerSaveEnable(true);
  //veml.setPowerSaveMode(VEML7700_POWERSAVE_MODE4);

  veml.setLowThreshold(10000);
  veml.setHighThreshold(20000);
  veml.interruptEnable(true);
}

void loop() {
  Serial.print("Lux: "); Serial.println(veml.readLux());
  Serial.print("White: "); Serial.println(veml.readWhite());
  Serial.print("Raw ALS: "); Serial.println(veml.readALS());

  uint16_t irq = veml.interruptStatus();
  if (irq & VEML7700_INTERRUPT_LOW) {
    Serial.println("** Low threshold");
  }
  if (irq & VEML7700_INTERRUPT_HIGH) {
    Serial.println("** High threshold");
  }
```

```
  delay(500);
}
```

You should get something resembling the following output when you open the Serial Monitor at 115200 baud:

Arduino Docs

Arduino Docs (https://adafru.it/Erg)

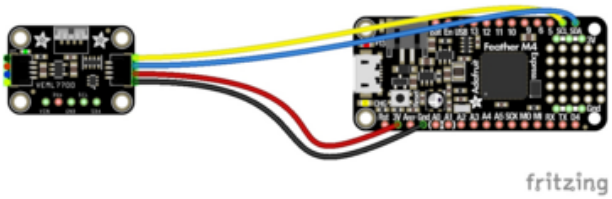
Python & CircuitPython

It's easy to use the VEML7700 sensor with CircuitPython and the Adafruit CircuitPython VEML7700 (https://adafru.it/EoF) module. This module allows you to easily write Python code that reads the ambient light levels, including Lux, from the sensor.

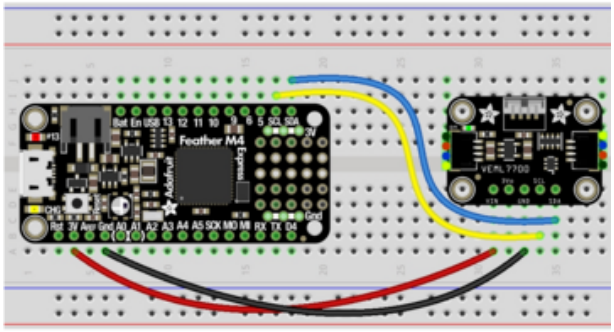
You can use this sensor with any CircuitPython microcontroller board or with a computer that has GPIO and Python thanks to Adafruit_Blinka, our CircuitPython-for-Python compatibility library (https://adafru.it/BSN).

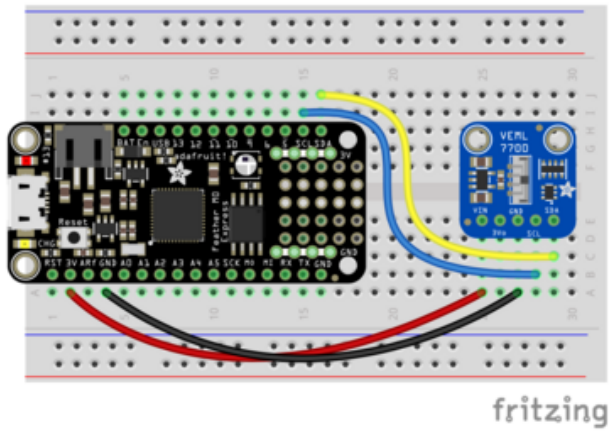
CircuitPython Microcontroller Wiring

First wire up a VEML7700 to your board exactly as follows. Here is an example of the VEML7700 wired to a Feather using I2C:



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



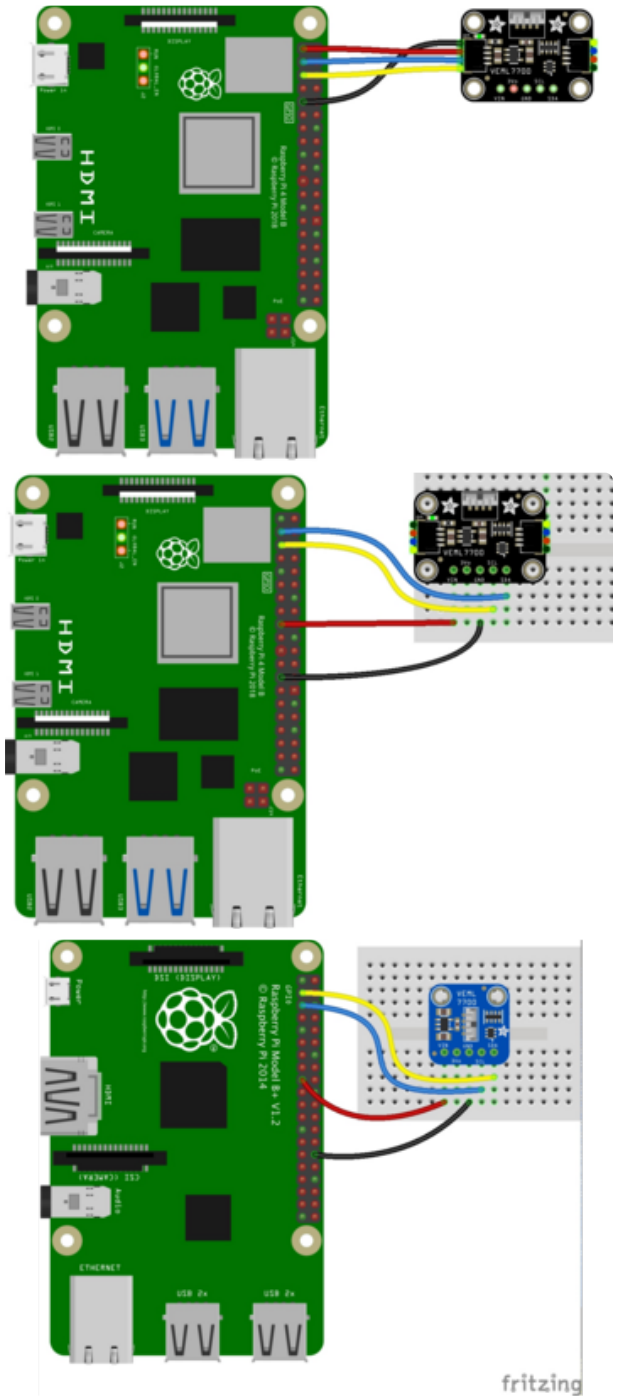


Note: This breakout includes pullup resistors on the I2C lines, no external pullups are required.

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).

Here's the Raspberry Pi wired with I2C:



- Pi 3V3 to sensor VIN (red wire on STEMMA QT version)
- Pi GND to sensor GND (black wire on STEMMA QT version)
- Pi SCL to sensor SCL (yellow wire on STEMMA QT version)
- Pi SDA to sensor SDA (blue wire on STEMMA QT version)

CircuitPython Installation of VEML7700 Library

You'll need to install the Adafruit CircuitPython VEML7700 (https://adafru.it/EoF) library on your CircuitPython board.

First make sure you are running the latest version of Adafruit CircuitPython (https://adafru.it/Amd) 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/uap). Our CircuitPython starter guide has a great page on how to install the library bundle (https://adafru.it/ABU).

For non-express boards like the Trinket M0 or Gemma M0, you'll need to manually install the necessary libraries from the bundle:

- adafruit_veml7700.mpy
- adafruit_bus_device
- adafruit_register

Before continuing make sure your board's lib folder or root filesystem has the adafruit_veml7700.mpy, adafruit_bus_device, and adafruit_register files and folders copied over.

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

Python Installation of VEML7700 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)!

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

- `sudo pip3 install adafruit-circuitpython-veml7700`

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

To demonstrate the usage of the sensor we'll initialize it and read the ambient light levels from the board's Python REPL.

Run the following code to import the necessary modules and initialize the I2C connection with the sensor:

```
import time
import board
import busio
import adafruit_veml7700

i2c = busio.I2C(board.SCL, board.SDA)
veml7700 = adafruit_veml7700.VEML7700(i2c)
```

Now you're ready to read values from the sensor using these properties:

- light - The ambient light data.
- lux - The light levels in Lux.

For example to print ambient light levels and lux values:

```
print("Ambient light:", veml7700.light)
print("Lux:", veml7700.lux)
```

```
>>> print("Ambient light:", veml7700.light)
Ambient light: 7107
>>> print("Lux:", veml7700.lux)
Lux: 1641.6
```

For more details, check out the library documentation (https://adafru.it/EoQ).

That's all there is to using the VEML7700 sensor with CircuitPython!

Full Example Code

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

import time
import board
import adafruit_veml7700

i2c = board.I2C()  # uses board.SCL and board.SDA
veml7700 = adafruit_veml7700.VEML7700(i2c)

while True:
    print("Ambient light:", veml7700.light)
    time.sleep(0.1)
```

Python Docs

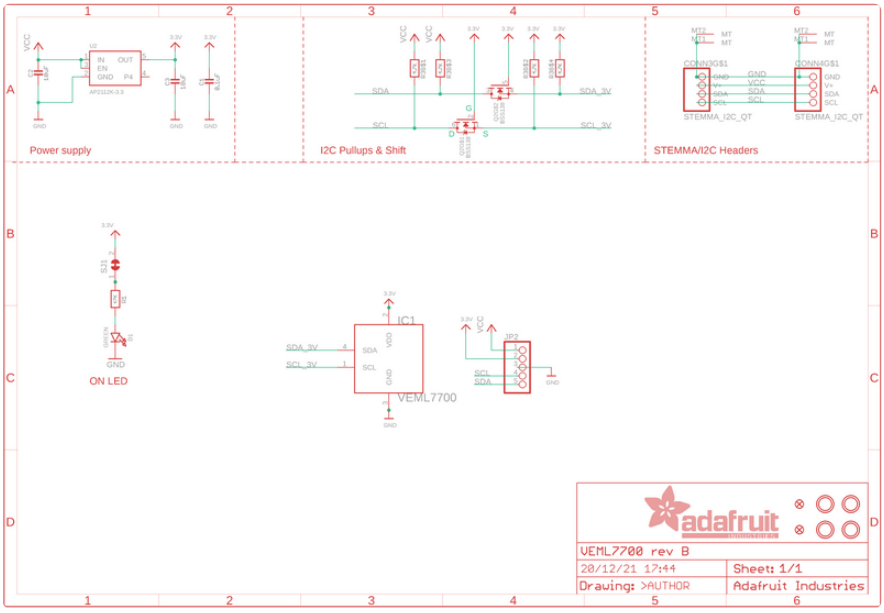
Python Docs (https://adafru.it/EoG)

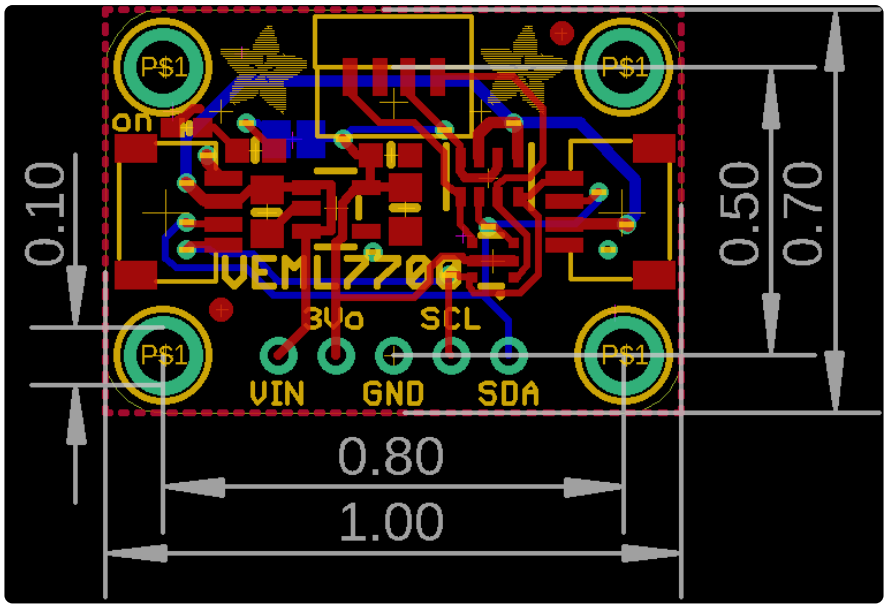
Downloads

Files

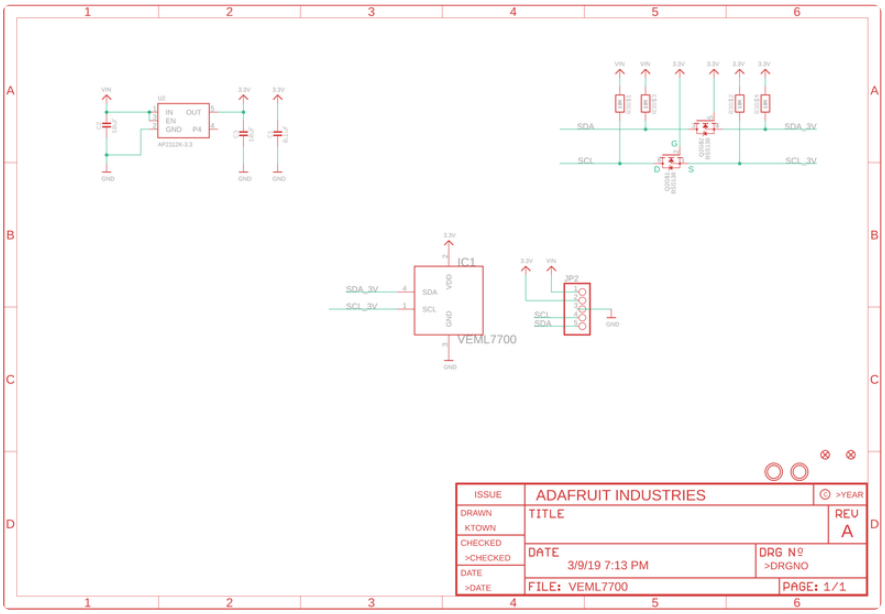
- VEML7700 Datasheet (https://adafru.it/EoR)
- EagleCAD files on GitHub (https://adafru.it/EoS)
- Fritzing object for STEMMA QT version in Adafruit Fritzing Library (https://adafru.it/Ytb)
- Fritzing object for original version from Adafruit Fritzing Library (https://adafru.it/EoT)

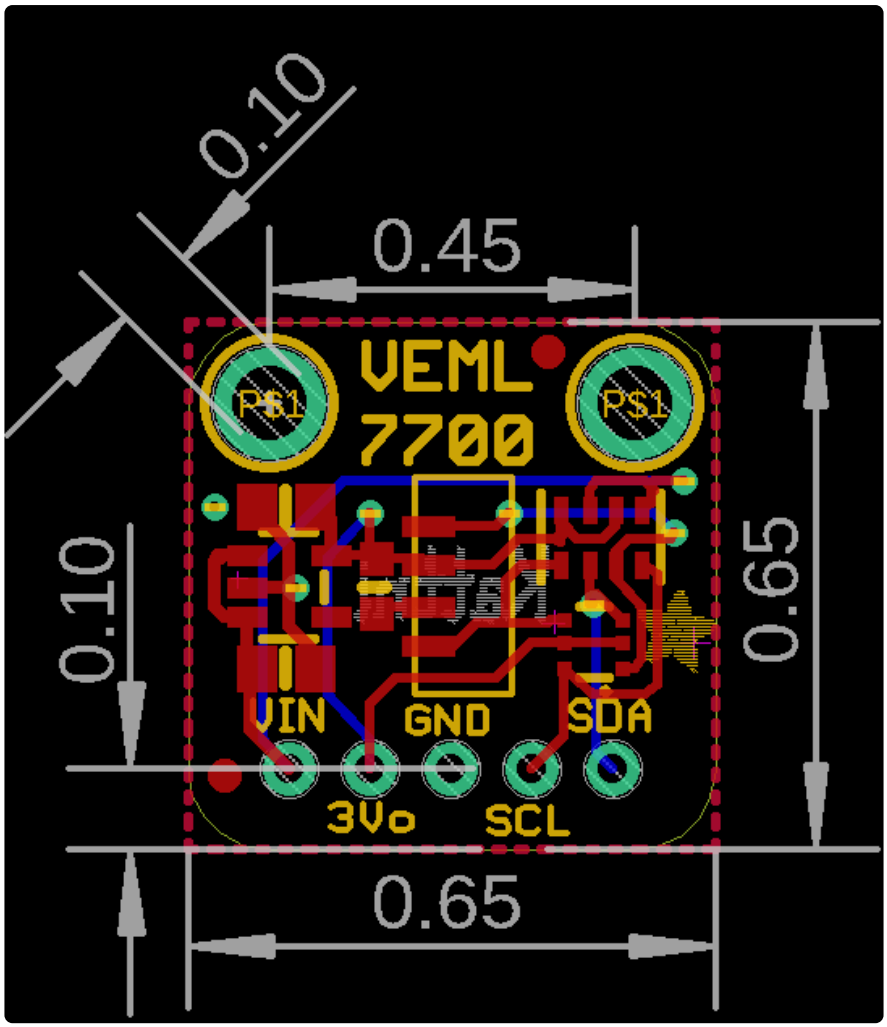
Schematic and Fab Print for STEMMA QT Version





Schematic and Fab Print for Original Version




0.10
0.45
VEML
7700
P$1
P$1
0.10
0.65
VIN
GND
SDA
3Vo
SCL
0.65