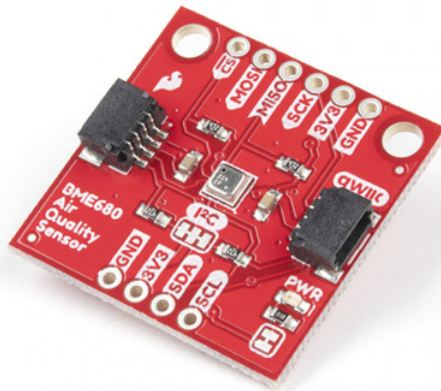




# SparkFun Environmental Sensor Breakout - BME680 (Qwiic) Hookup Guide

## Introduction

The SparkFun Environmental Sensor - BME680 (Qwiic) is a breakout for the 4-in-1 BME680 gas sensor from Bosch. The BME680 combines a gas sensor with temperature, humidity and barometric pressure sensing for a complete environmental sensor in a single package. The gas sensor on the BME680 can detect a wide variety of volatile organic compounds (or VOC for short) to monitor indoor air quality. Combine that with precise temperature, humidity and barometric pressure and the BME680 can work as a completely standalone environmental sensor all in a 1"x1" breakout! The BME680 communicates over either I<sup>2</sup>C or SPI. As you would expect from the name, the BME680's I<sup>2</sup>C pins are broken out to a Qwiic connector so integrating it into the SparkFun Qwiic System is a breeze. Simply plug it into a Qwiic-enabled microcontroller and you're well on your way to making your own weather station.



## SparkFun Environmental Sensor Breakout - BME680 (Qwiic)

○ SEN-16466

Product Showcase: SparkFun Qwiic Environmental Sensor Break...



## Required Materials

To follow along with this guide you will need a microcontroller to communicate with the BME680. Below are a few options that come Qwiic-enabled out of the box:



SparkFun Thing Plus - ESP32 WROOM

● WRL-15663



SparkFun Qwiic Pro Micro - USB-C  
(ATmega32U4)

● DEV-15795



SparkFun RedBoard Qwiic

● DEV-15123



SparkFun RedBoard Artemis

● DEV-15444

If your chosen microcontroller is not already Qwiic-enabled, you can add that functionality with one or more of the following items:



SparkFun Qwiic Cable Kit  
○ KIT-15081



SparkFun Qwiic Shield for Arduino  
● DEV-14352



SparkFun Qwiic Adapter  
● DEV-14495



SparkFun Qwiic Shield for Arduino Nano  
● DEV-16130

You will also want at least one Qwiic cable to connect your sensor to your microcontroller.



Qwiic Cable - 100mm  
● PRT-14427



Qwiic Cable - 200mm  
● PRT-14428



Qwiic Cable - 500mm



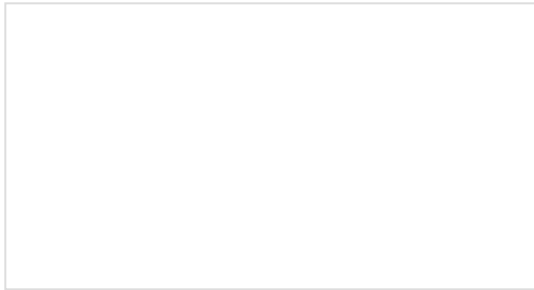
Qwiic Cable - 50mm

## Suggested Reading

If you are not familiar with the Qwiic system, we recommend reading here for an overview:

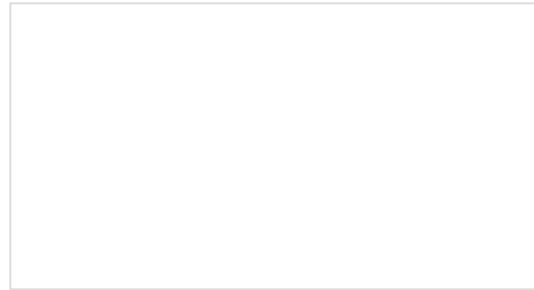


We would also recommend taking a look at the following tutorials if you aren't familiar with the concepts covered in them.



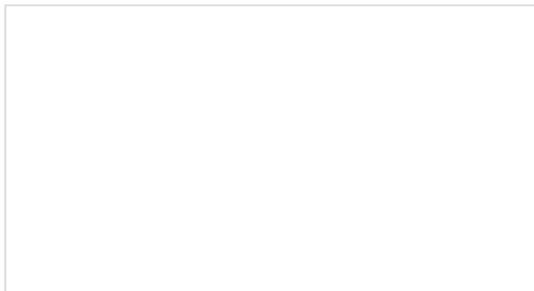
### Serial Peripheral Interface (SPI)

SPI is commonly used to connect microcontrollers to peripherals such as sensors, shift registers, and SD cards.



### I2C

An introduction to I2C, one of the main embedded communications protocols in use today.



### Serial Terminal Basics

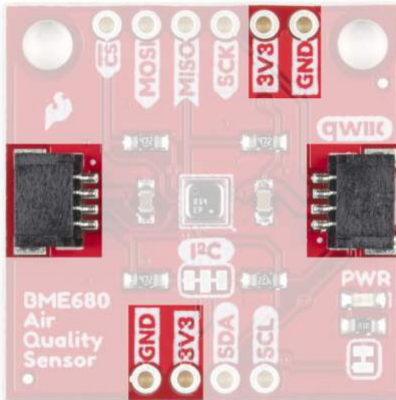
This tutorial will show you how to communicate with your serial devices using a variety of terminal emulator applications.

## Hardware Overview

The heart of this breakout, Bosch's BME680 Gas Sensor, integrates four sensors (gas, pressure, temperature and humidity) into a tiny package. The BME680 measures just 3mm x 3mm x 0.93 mm and was specifically designed for applications that depend on a small footprint and low power consumption. This makes the BME680 a great choice for remote or mobile environmental sensing applications. We will highlight some of the unique aspects of the BME680 in this section but for a full overview of the sensor package, check out the Datasheet.

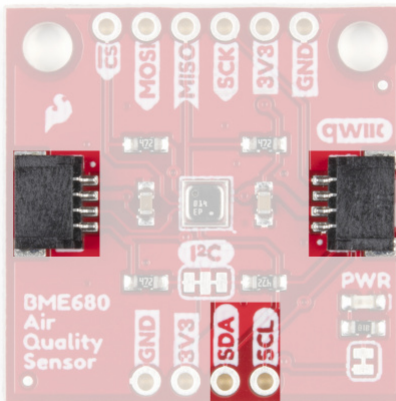
## Power

The BME680 accepts a supply voltage between **1.71 to 3.6V**. Power can be supplied to the board either through one of the Qwiic connectors or the dedicated **3.3V** and **GND** pins broken out on either side of the board.



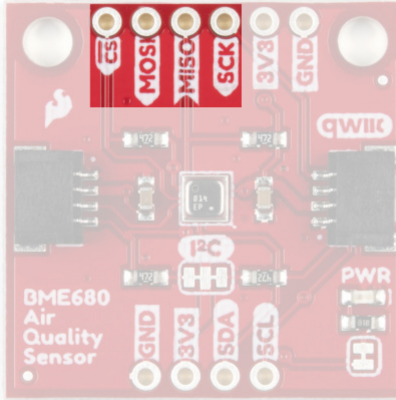
## Qwiic and I<sup>2</sup>C Interface

The SparkFun Environmental Sensor - BME680 (Qwiic) communicates over I<sup>2</sup>C by default. We have routed the BME680's I<sup>2</sup>C pins to two Qwiic connectors as well as broken them out to 0.1"-spaced the header pins highlighted below.



## Serial Peripheral Interface (SPI)

If you would prefer to communicate with your BME680 via SPI, we have broken those pins out as well to standard 0.1"-spade header pins. Communicating over SPI requires more connections than I<sup>2</sup>C but is more versatile and can be faster. It is particularly helpful if you need to use more than two BME680's in your circuit or if you have other devices using the same I<sup>2</sup>C addresses.



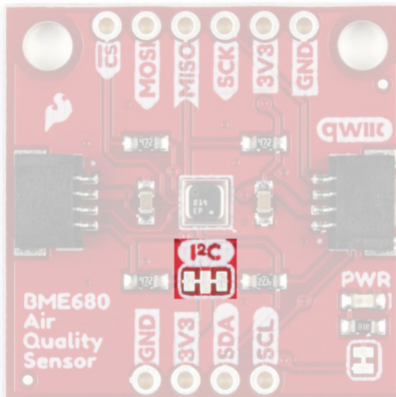
## Jumpers

The SparkFun Environmental Sensor - BME680 (Qwiic) has three solder jumpers which can be modified to alter the functionality of the sensor.

If you have never worked with solder jumpers or PCB traces before or would like a refresher, take a look at our [How to Work with Jumper Pads and PCB Traces](#) tutorial.

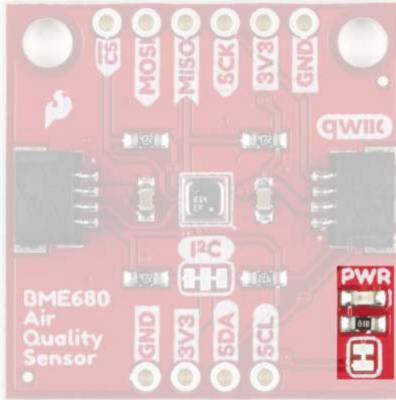
## I<sup>2</sup>C Pull-Up Jumper

The SDA/SDI and SCL/SCK pins are pulled to VDDIO (**3.3V**) through a pair of **4.7k Ohm** resistors. The jumper is normally **closed** so to disable the pull-up resistors, simply sever the traces between the three pads using a hobby knife.



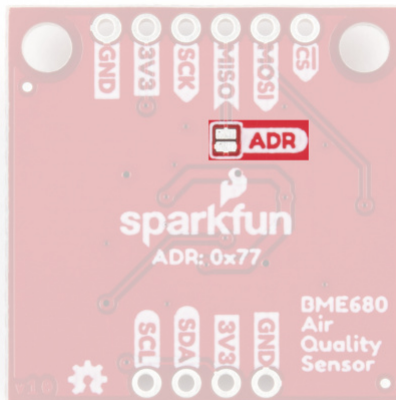
## Power LED Jumper

This jumper connects the power LED to **3.3V** via a **1K Ohm** resistor. This jumper is normally **closed** so to disable the power LED, sever the trace between the two pads. This is particularly helpful for reducing the total current draw of your breakout for low-power applications.



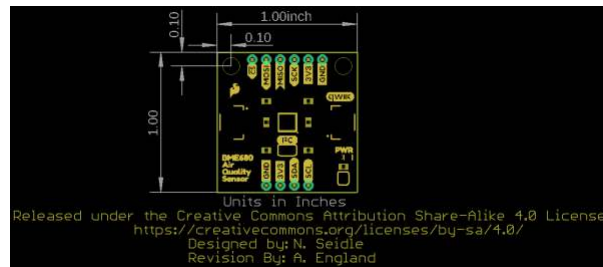
## I<sup>2</sup>C Address Jumper

This jumper sets the 7-Bit unshifted I<sup>2</sup>C address of the BME680 and is **open** by default. The default address is **0x77** and can be adjusted to **0x76** by closing this jumper.



## Board Dimensions

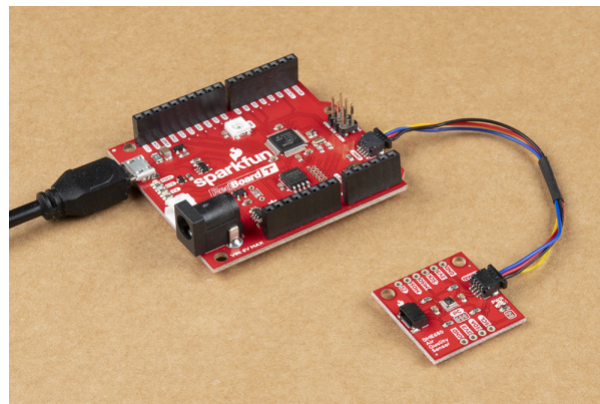
This breakout fits the Qwiic standard sizing for breakouts. It is a 1"x1" square with two mounting holes that fit a 4-40 screw.



Now that we have a thorough understanding of the hardware and features on the Environmental Sensor - BME680 (Qwiic), it's time to hook it up and start taking measurements.

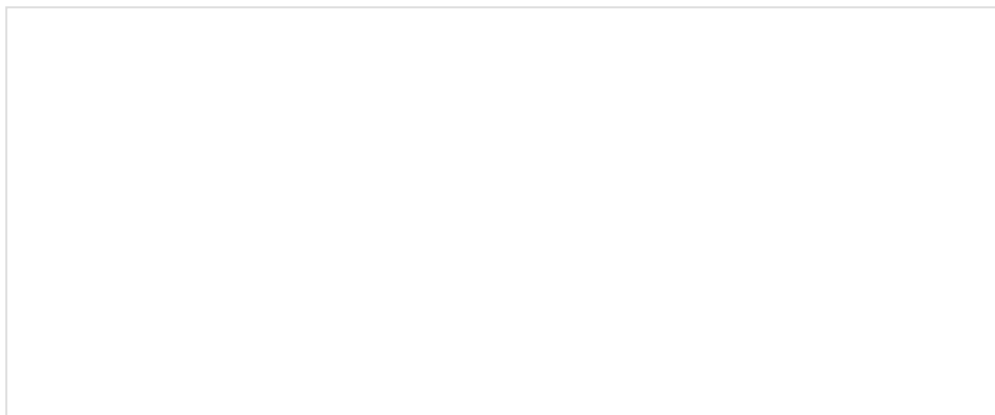
## Hardware Hookup

Using the Qwiic system, assembling the hardware is simple. All you need to do is connect your Environmental Sensor - BME680 (Qwiic) to your chosen development board with a Qwiic cable or adapter cable. Otherwise, you can use the I<sup>2</sup>C pins broken out if you do not have a Qwiic connector on your development board or if you do not want to use a Qwiic connection. If you are not using a Qwiic-enabled board, make sure your input voltage and logic are either running at **3.3V** or you are shifting the logic level from whatever logic your controller runs at to **3.3V** for the BME680.



If you would prefer to communicate with the BME680 via SPI, you will need to connect to the SPI pins broken out on this board and route them to the respective pins for SPI communication on your development board (MISO, MOSI, SCK and CS/SS). Also note that this breakout defaults to I<sup>2</sup>C mode so your code will need to toggle the CS pin **LOW** once on power up to enable SPI mode. The BME680 will remain in SPI mode until the next power cycle. The SPI examples further on in this guide do that automatically so it's only necessary to note for writing your own code.

Soldering to the pins is the best option for a secure connection but you can also create temporary connections to those pins for prototyping using something like these IC Hooks. If you are not familiar with through-hole soldering, take a look at this tutorial:





# How to Solder: Through-Hole Soldering

SEPTEMBER 19, 2013

This tutorial covers everything you need to know about through-hole soldering.

With everything connected properly, we're ready to move on to uploading a sketch and start monitoring your environment!

## BME680 Arduino Library

**Note:** This library assumes you are using the latest version of the Arduino IDE on your desktop. If this is your first time using Arduino, please review our tutorial on installing the Arduino IDE. If you have not previously installed an Arduino library, please check out our installation guide.

For the scope of tutorial, we are going to use the BME680 Arduino Library created by SV-Zanshin. You can download it with the Arduino Library Manager by searching '**BME680**' and selecting the one authored by SV-Zanshin. Alternatively, you can grab the zip of the latest release (v1.0.3 as of this writing) below or you can download the most up to date version of the library from the GitHub repository.

**BME680 ARDUINO LIBRARY (ZIP)**

Once you have the library installed you can move on to uploading the examples and gathering environmental data.

## Arduino Examples

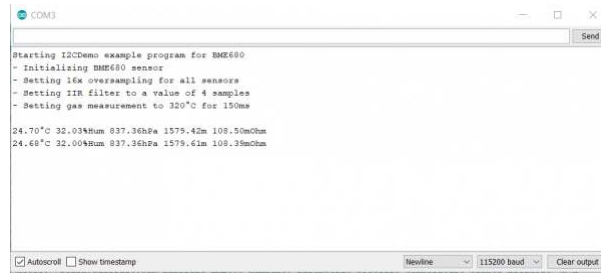
Now that the library is installed, we can move on to uploading some code. Before we discuss the individual examples, we'll cover some of the setup they perform for the BME680.

The code configures the BME680 to perform oversampling for the temperature, humidity and pressure sensors and sets an IIR (infinite impulse response) filter for these sensors. This helps smooth out environmental data from any short term outliers. Finally, the setup also configures the temperature and time settings for the hot plate on the gas sensor. If you would like to adjust any of these settings, refer to the BME680 Datasheet and the library source files for more information.

### I<sup>2</sup>C Demo

To open this example, head to **File > Examples > BME680 > I2CDemo**. Next, open the Tools menu and select your board (in our case, **Arduino Uno**) and the correct Port your board enumerated on.

Upload the code, open the Arduino Serial Monitor and set your baud rate to **115200**. You probably will see the code print out the successful initialization of the BME680 as well as the settings we discussed above and after that, you should see temperature, humidity, pressure, altitude and raw gas readings every five seconds.



```
COM3
Starting I2CDemo example program for BME680
- Initializing BME680 sensor
- Setting 16x oversampling for all sensors
- Setting IIR filter to a value of 4 samples
- Setting gas measurement to 320°C for 150ms
24.70°C 32.03%Hum 837.36hPa 1579.42m 109.50mOhm
24.69°C 32.00%Hum 837.36hPa 1579.61m 109.35mOhm
Autoscroll Show timestamp Newline 115200 baud Clear output
```

## SPI Demo

This example is nearly identical to the I<sup>2</sup>C demo but instead uses SPI mode. One thing to note is, depending on which type of microcontroller you are using, you may need to adjust this line:

```
const uint8_t SPI_CS_PIN = 53;
```

Set the CS/SS pin to the appropriate pin on your microcontroller. In our case, the Uno uses D10 for CS so the modified version of that line for an Uno or SparkFun RedBoard Qwiic would be:

```
const uint8_t SPI_CS_PIN = 10;
```

With that line adjusted, upload the code and open your serial monitor. You should see a similar stream of data as the screenshot above for the I<sup>2</sup>C example.

**Tip:** If you'd like to use this library with a development board with multiple serial ports like the RedBoard Turbo we show in the Hardware Hookup section, you'll want to add this line: `#define Serial SerialUSB` prior to your `void setup()`. This definition can be modified to any serial port on your chosen microcontroller.

## Troubleshooting

In this section we'll cover a few tips and tricks for troubleshooting common questions with the Environmental Sensor Breakout - BME680.

### Gas Readings Interpretation

The library used in this tutorial only prints out the raw resistance values from the gas sensor on the BME680. You can use these values as a rough estimate of air quality where lower resistance values equate to a higher concentration of gases measured (and vice versa). If you want to get true Indoor Air Quality (IAQ) measurements from the BME680, we recommend taking a look at Bosch's BSEC Arduino Library which includes an algorithm to convert the resistance value to an IAQ value. For more information, refer to that library as well as sections 1.2 and 4.2 in the BME680 Datasheet. Calculated IAQ measurements are beyond the scope of this tutorial.

### Incorrect Temperature Data

You may notice some deviation from the true ambient temperature in your data as residual heat from the hot plate for the gas sensor in the BME680 can cause minor fluctuations in the observed temperature. The heating phase starts after temperature, pressure and humidity measurements are complete so there should be no heating *during* those measurements but subsequent readings may be skewed. The IIR filters can help here but if needed, you can compensate for this by measuring the average deviation and subtracting that from your temperature data.

### Incorrect Altitude Data

The altitude data is collected by converting the barometric pressure. This is a great tool for approximate altitude readings but things like weather patterns can affect the accuracy of the altitude. The examples use the standard measurement for pressure at sea level (1013.25 hPa) in the calculation so you may wish to adjust that with a corrected value for a more accurate altitude data. Refer to this Wikipedia page and this section of our MPL3115 Pressure Sensor Hookup Guide for more information.

## Chip Select Definition

As we covered in the previous section, if you choose to communicate with the BME680 via SPI, make sure you are connecting to the correct pins on your development board (MOSI, MISO, SCK and CS) as well as modifying the Chip Select/Slave Select definition to the appropriate I/O pin used for CS on your controller. If you are not certain which pin is used for CS, refer to documentation for your particular development board.

### Need help with something not covered here?

If your product is not working as you expected or you need technical assistance or information, head on over to the SparkFun Technical Assistance page for some initial troubleshooting.

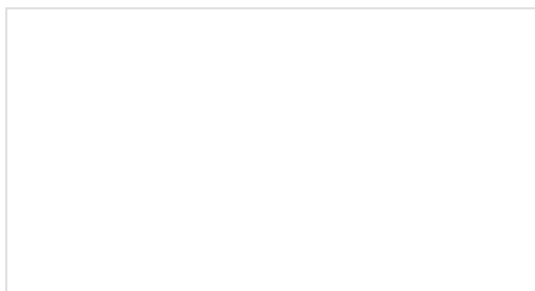
If you don't find what you need there, the SparkFun Forums are a great place to find and ask for help. If this is your first visit, you'll need to create a Forum Account to search product forums and post questions.

## Resources and Going Further

Now that your BME680 is integrated in your circuit and sending data it's time to start tracking some weather! For more information about this breakout and the BME680, take a look at the links below:

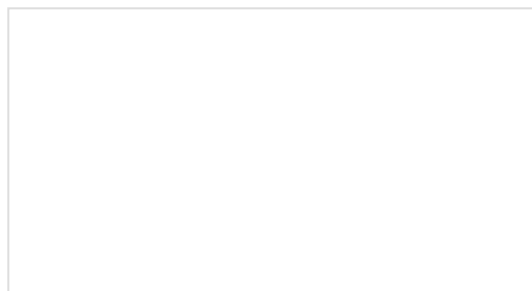
- Schematic (PDF)
- Eagle Files (ZIP)
- Board Dimensions (PNG)
- Datasheet (PDF)
- Layout Considerations (PDF)
- Hardware GitHub Repository
- BME680 Arduino Library GitHub Repository
- Bosch BSEC Arduino Library GitHub Repository

Need some inspiration for your next project? Check out some of these weather sensing related tutorials:



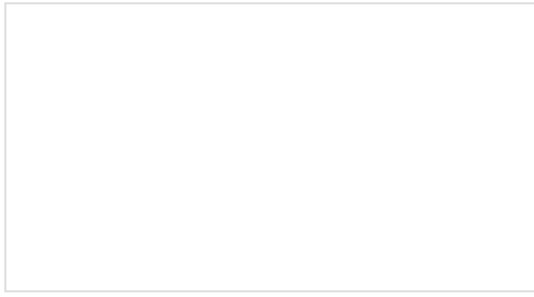
### SparkFun Inventor's Kit for Photon Experiment Guide

Dive into the world of the Internet of Things with the SparkFun Inventor's Kit for Photon.



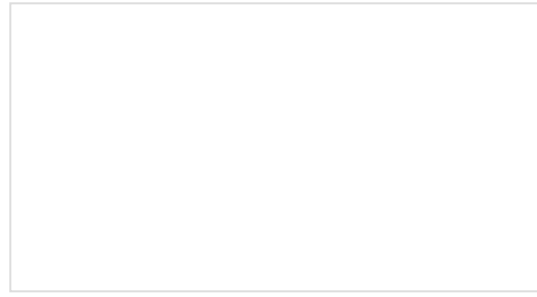
### SparkFun BME280 Breakout Hookup Guide

A guide for connecting the BEM280 sensor to a microcontroller, and for using the Sparkfun Arduino library.



## ESP32 Environment Sensor Shield Hookup Guide

SparkFun's ESP32 Environment Sensor Shield provides sensors and hookups for monitoring environmental conditions. This tutorial will show you how to connect your sensor suite to the Internet and post weather data online.



## Weather Meter Hookup Guide

How to assemble your very own weather meter!