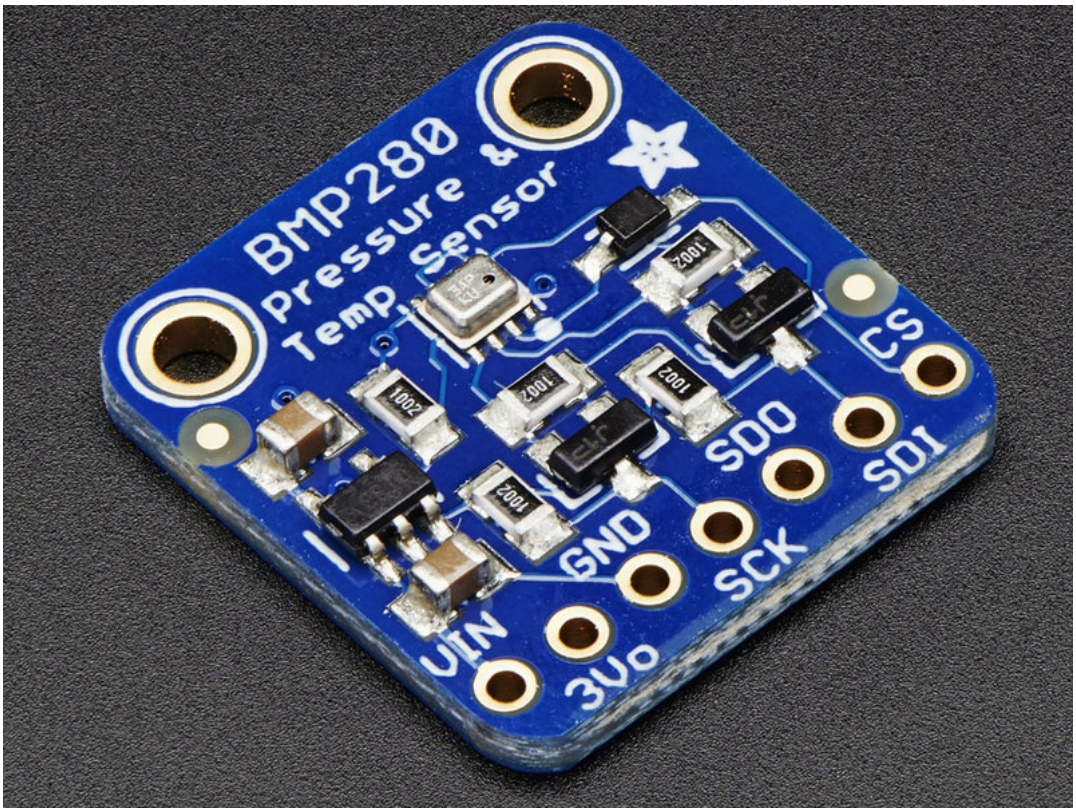
- SHOP
- BLOG
- LEARN
- FORUMS
- VIDEOS



Adafruit BMP280 Barometric Pressure + Temperature Sensor Breakout

Buld your next weather sensing system

Contributors

lady ada
Tony DiCola

Arduino Test

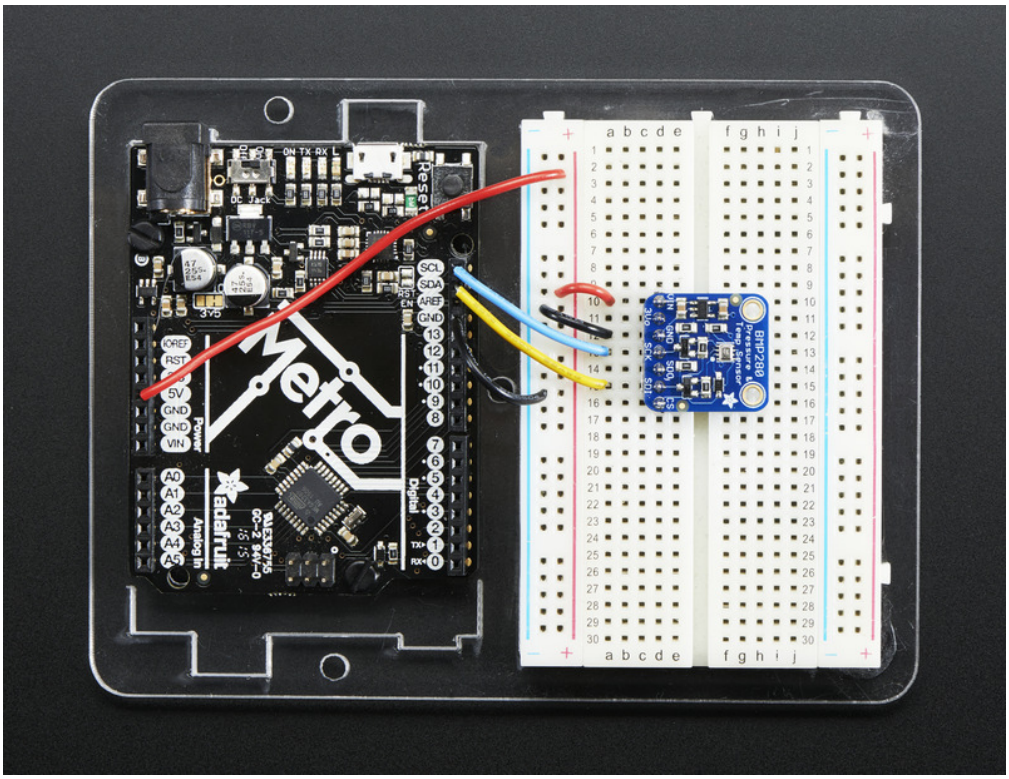
by lady ada

You can easily wire this breakout to any microcontroller, we'll be using an Arduino. For another kind of microcontroller, as long as you have 4 available pins it is possible to 'bit-bang SPI' or you can use two I2C pins, but usually those pins are fixed in hardware. Just check out the library, then port the code.

I2C Wiring

Use this wiring if you want to connect via I2C interface

- Connect **Vin** to the power supply, 3-5V is fine. Use the same voltage that the microcontroller logic is based off of. For most Arduinos, that is 5V
- Connect **GND** to common power/data ground
- Connect the **SCK** 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 **SDI** 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**

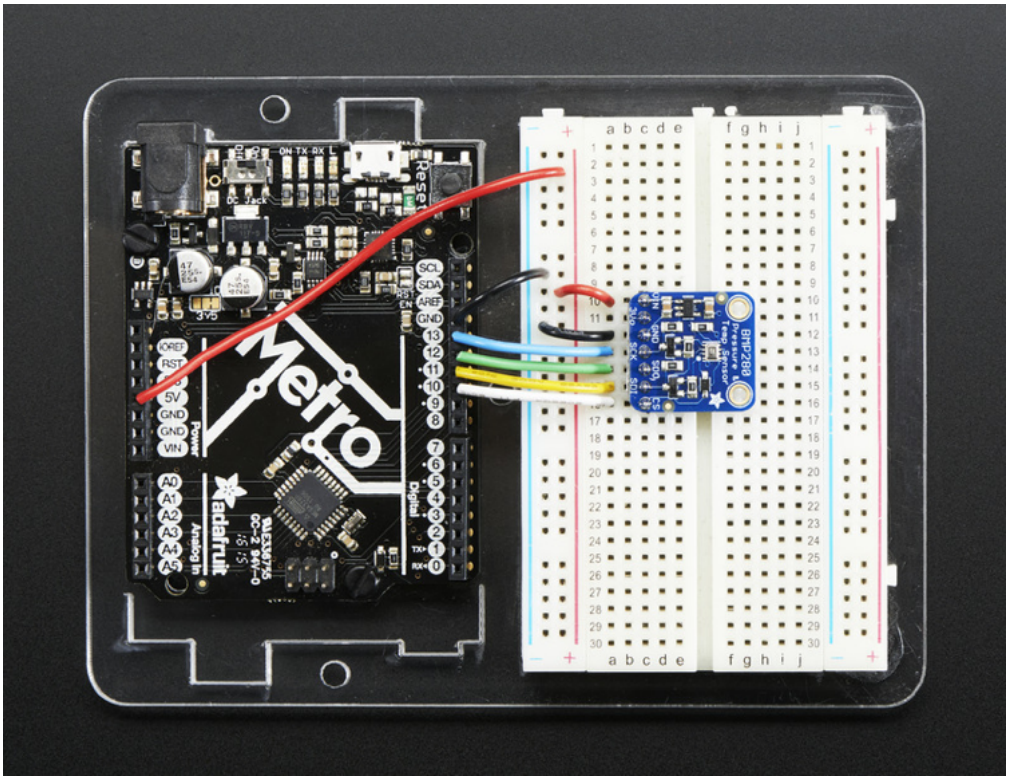


SPI Wiring

Since this is a SPI-capable sensor, we can use hardware or 'software' SPI. To make wiring identical on all Arduinos, we'll begin with 'software' SPI. The following pins should be used:

- Connect **Vin** to the power supply, 3V or 5V is fine. Use the same voltage that the microcontroller logic is based off of. For most Arduinos, that is 5V
- Connect **GND** to common power/data ground
- Connect the **SCK** pin to **Digital #13** but any pin can be used later
- Connect the **SDO** pin to **Digital #12** but any pin can be used later
- Connect the **SDI** pin to **Digital #11** but any pin can be used later
- Connect the **CS** pin **Digital #10** but any pin can be used later

Later on, once we get it working, we can adjust the library to use hardware SPI if you desire, or change the pins to other



Download Adafruit_BMP280 library

To begin reading sensor data, you will need to download Adafruit_BMP280 from our github repository. You can do that by visiting the github repo and manually downloading or, easier, just click this button to download the zip

Download Adafruit BMP280 Library

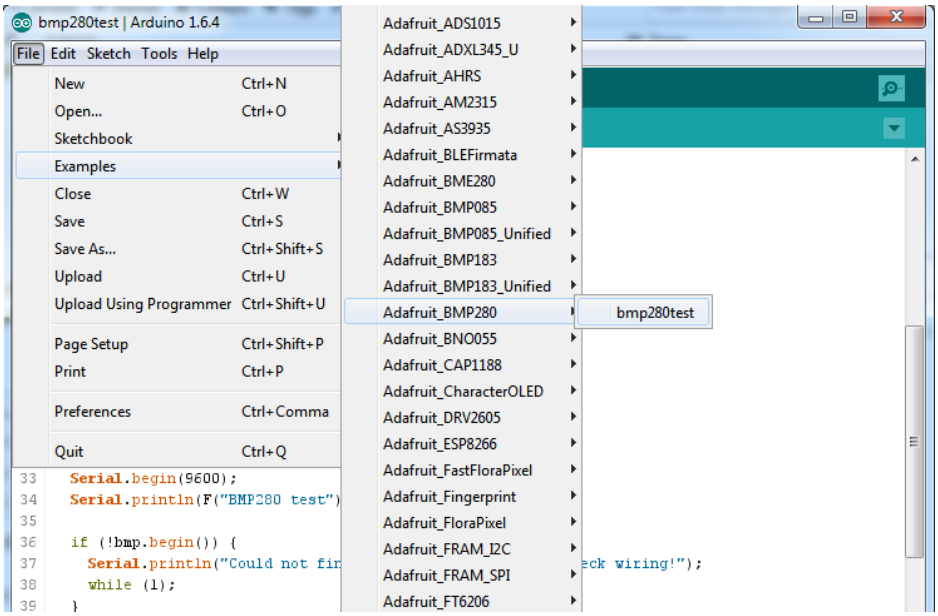
Rename the uncompressed folder **Adafruit_BMP280** and check that the **Adafruit_BMP280** folder contains **Adafruit_BMP280.cpp** and **Adafruit_BMP280.h**

Place the **Adafruit_BMP280** library folder your **arduinosketchfolder/libraries/** folder.
You may need to create the **libraries** subfolder if its your first library. Restart the IDE.

We also have a great tutorial on Arduino library installation at:
http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use

Load Demo

Open up **File->Examples->Adafruit_BMP280->bmp280test** and upload to your Arduino wired up to the sensor



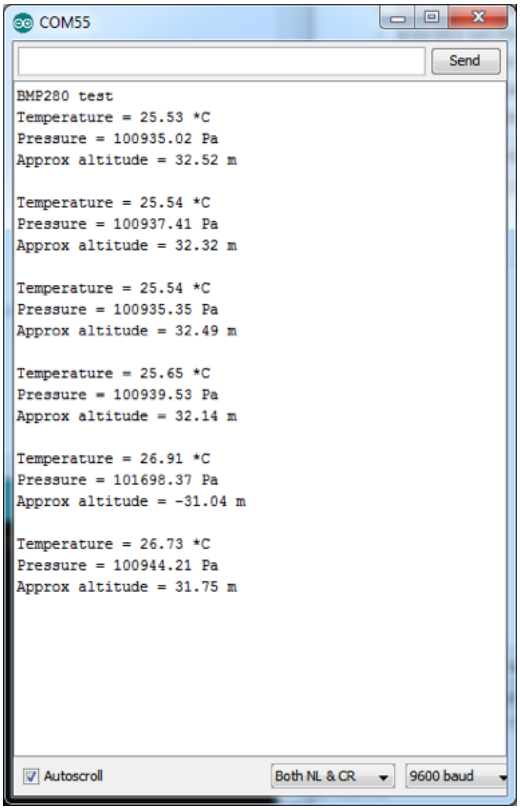
Depending on whether you are using I2C or SPI, change the pin names and comment or uncomment the following lines.

Download file
Copied!

```
#define BMP_SCK 13
#define BMP_MISO 12
```

```
3. #define BMP_MOSI 11
4. #define BMP_CS 10
5.
6. Adafruit_BMP280 bmp; // I2C
7. //Adafruit_BMP280 bmp(BMP_CS); // hardware SPI
8. //Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO,  BMP_SCK);
```

Once uploaded to your Arduino, open up the serial console at 9600 baud speed to see data being printed out



Temperature is calculated in degrees C, you can convert this to F by using the classic F = C * 9/5 + 32 equation.

Pressure is returned in the SI units of **Pascals**. 100 Pascals = 1 hPa = 1 millibar. Often times barometric pressure is reported in millibar or inches-mercury. For future reference 1 pascal =0.000295333727 inches of mercury, or 1 inch Hg = 3386.39 Pascal. So if you take the pascal value of say 100734 and divide by 3389.39 you'll get 29.72 inches-Hg.

You can also calculate Altitude. **However, you can only really do a good accurate job of calculating altitude if you know the hPa pressure at sea level for your location and day!** The sensor is quite precise but if you do not have the data updated for the current day then it can be difficult to get more accurate than 10 meters.

Library Reference

You can start out by creating a BMP280 object with either software SPI (where all four pins can be any I/O) using

Download file
Copy Code

```
1. Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO,  BMP_SCK);
```

Or you can use hardware SPI. With hardware SPI you *must* use the hardware SPI pins for your Arduino - and each arduino type has different pins! Check the SPI reference to see what pins to use.
In this case, you can use any CS pin, but the other three pins are fixed

Download file
Copy Code

```
1. Adafruit_BMP280 bmp(BMP_CS); // hardware SPI
```

or I2C using the default I2C bus, no pins are assigned

Download file
Copy Code

```
1. Adafruit_BMP280 bmp; // I2C
```

Once started, you can initialize the sensor with

Download file
Copy Code

```
1.   if (!bmp.begin()) {
2.     Serial.println("Could not find a valid BMP280 sensor, check wiring!");
3.     while (1);
4.   }
```

begin() will return True if the sensor was found, and False if not. If you get a False value back, check your wiring!

Reading temperature and pressure is easy, just call:

Download file
Copy Code

```
bmp.readTemperature()
bmp.readPressure()
```

Temperature is always a floating point, in Centigrade. Pressure is a 32 bit integer with the pressure in Pascals. You may need to convert to a different value to match it with your weather report.

It's also possible to turn the BMP280 into an altimeter. If you know the pressure at sea level, the library can calculate the current barometric pressure into altitude

ASSEMBLY CIRCUITPYTHON TEST
Last updated on 2017-11-12 at 12.37.39 PM Published on 2015-07-31 at 03.34.19 PM

Adafruit BMP280 I2C or SPI Barometric Pressure & Altitude Sensor
$9.95 ADD TO CART
ADD ALL TO CART

RELATED GUIDES

POPULAR

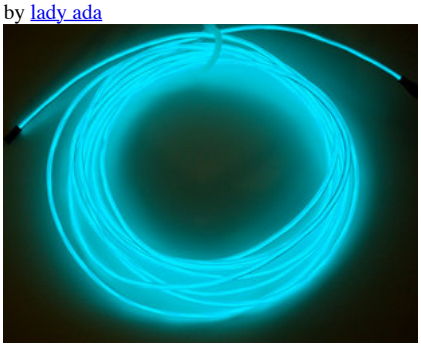
TTL Serial Camera

Snap, Snap!
by lady ada

This guide is for our new TTL serial camera module with NTSC video output. These modules are a nice addition to a microcontroller project when you want to take a photo or control a video stream. The modules have a few features built in, such as the ability to change the brightness/saturation/hue of images, auto-contrast and auto-brightness adjustment, and motion detection.

EL Wire

Working with Electroluminescent Wire
by lady ada



EL Wire, also known as Electroluminescent wire, is a stiff wire core coated with phosphor and then covered with a protective PVC sheath. When an AC signal is applied to it, it glows a cool neon color. Find out how to solder, power, and work with EL Wire in your next project.

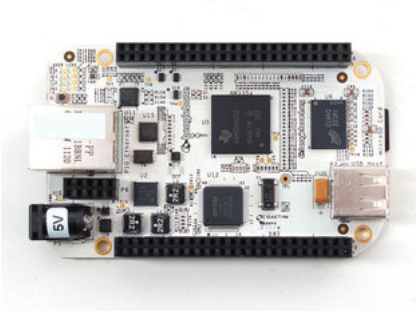
Hacking the Kinect

Reverse engineering the Microsoft Kinect
by lady ada

Here's a step by step guide on how you can reverse engineer a Microsoft Kinect for the Xbox 360.

BeagleBone

Tutorials for the TI embedded Linux board
by lady ada



New from the fine people who have brought us the Beagle Board, we now have a smaller, lighter, but powerful single board linux computer, Beagle Bone! We like this move to a more compact and integrated SBC. For example, there is onboard Ethernet and USB host, as well as a USB client interface (a FTDI chip for shell access). It even comes preloaded with Angstrom Linux on the 4 GB microSD card! Here are some tips and tricks to get your BeagleBone up and running.
×

OUT OF STOCK NOTIFICATION

YOUR NAME
YOUR EMAIL
NOTIFY ME