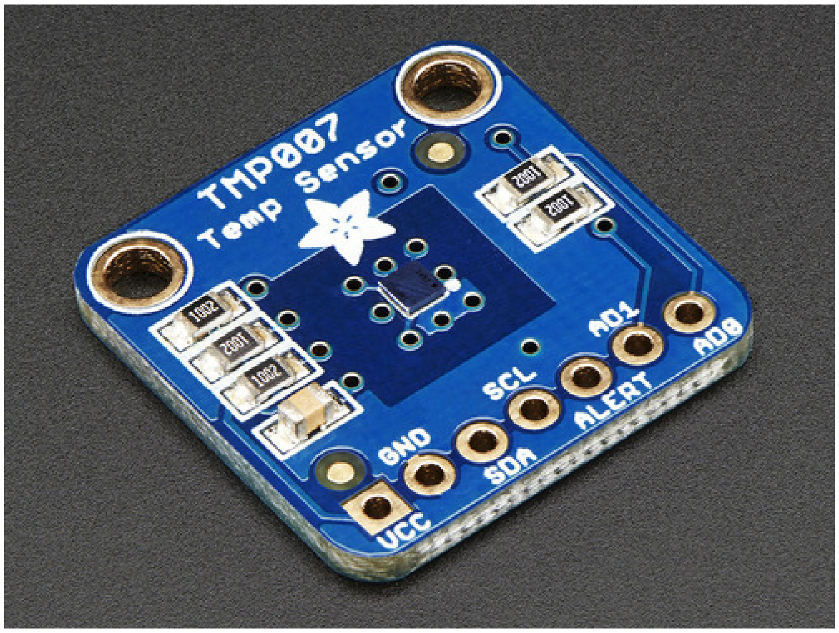
adafruit learning system

Adafruit TMP007 Sensor Breakout

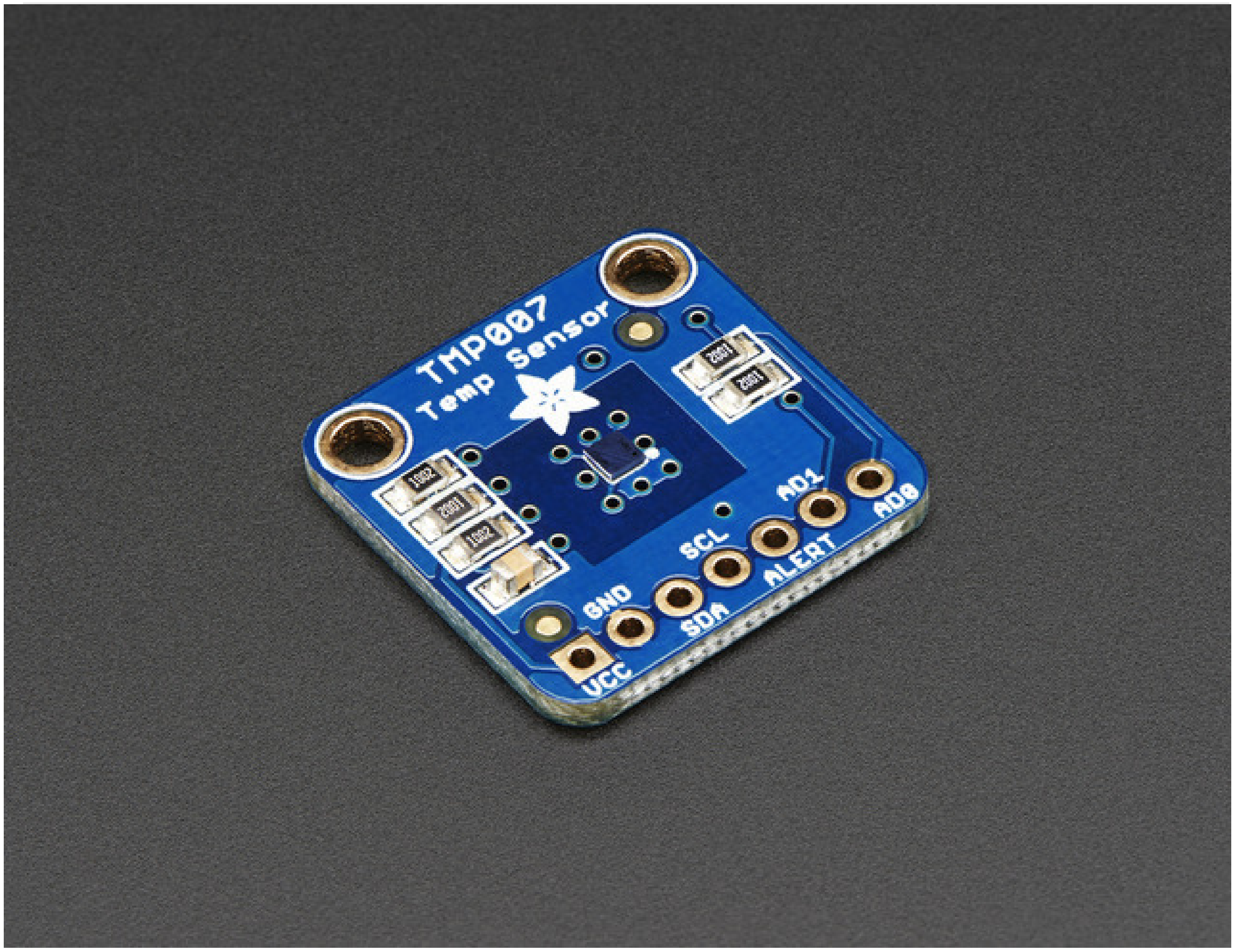
Created by lady ada



Last updated on 2014-08-05 02:15:08 PM EDT

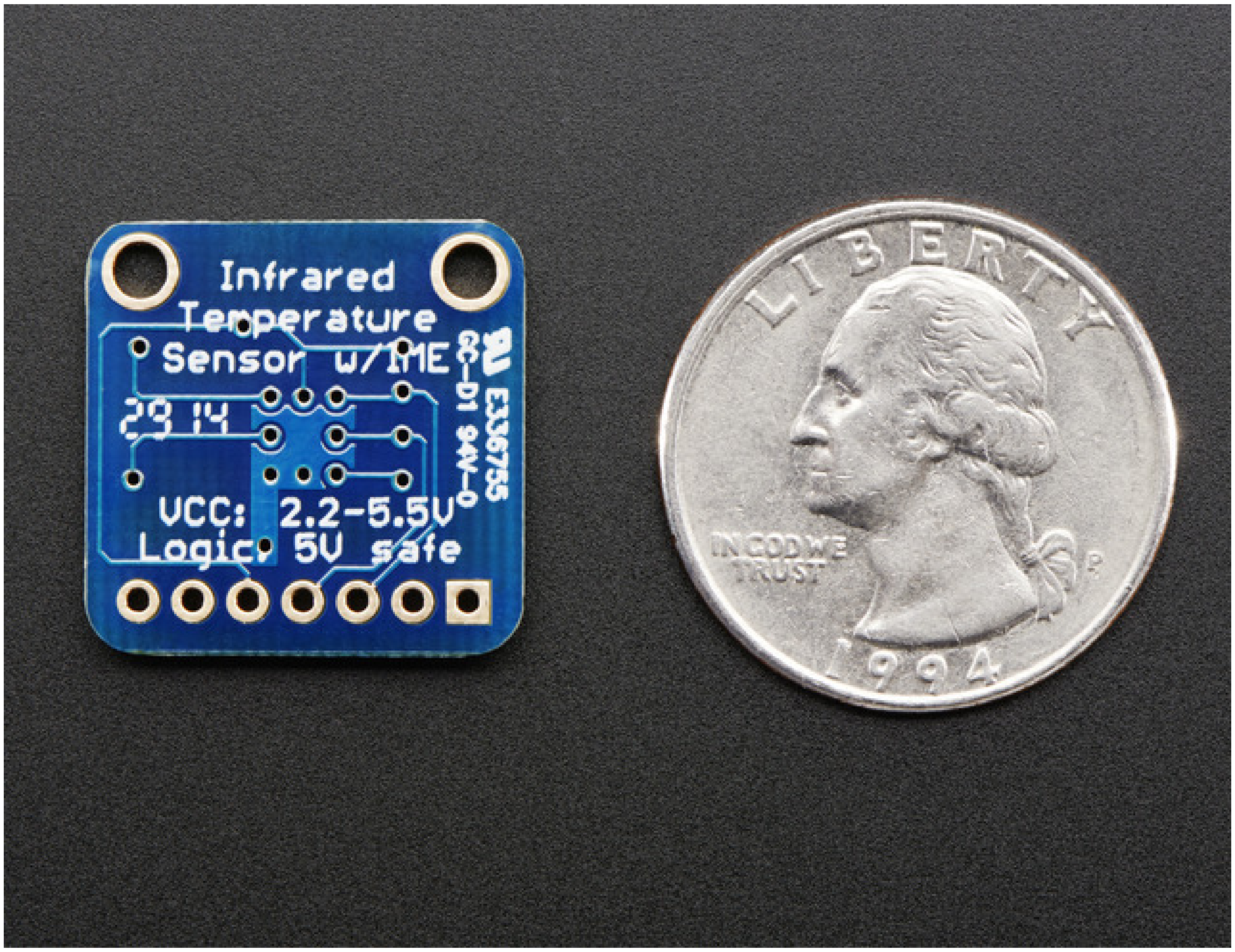
Guide Contents

Overview

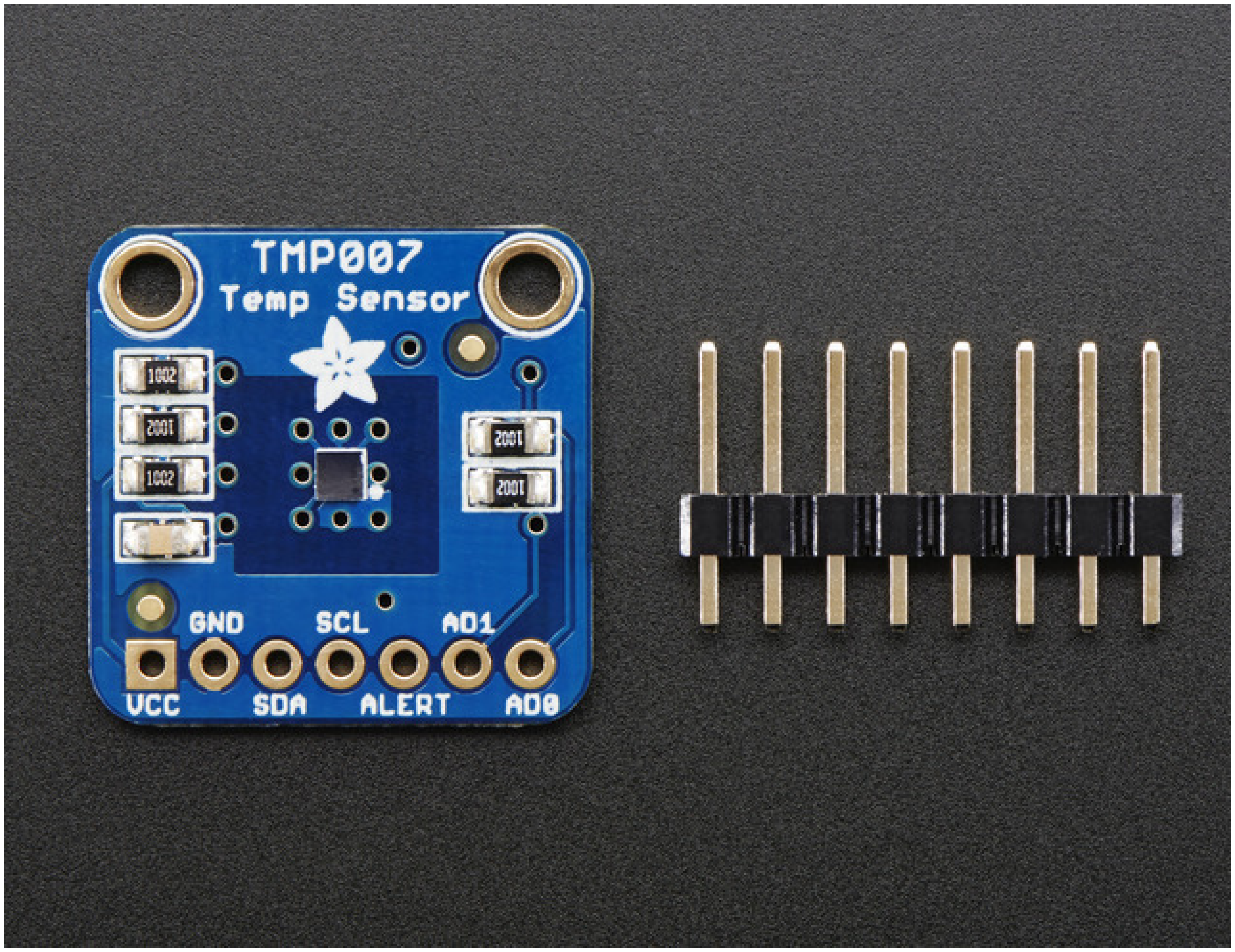


Unlike most of the other temperature sensors we have, this breakout has a really cool IR sensor from TI that can measure the temperature of an object without touching it.

The TMP007 is the latest thermopile sensor from TI, and is an update of the TMP006 (http://adafru.it/dMS). The internal math engine does all the temperature calculations so its easier to integrate - you can read the die and target temperatures directly over I2C. The TMP007 also has better transient management, so you don't get as much over/undershoot when the temperature changes a lot.

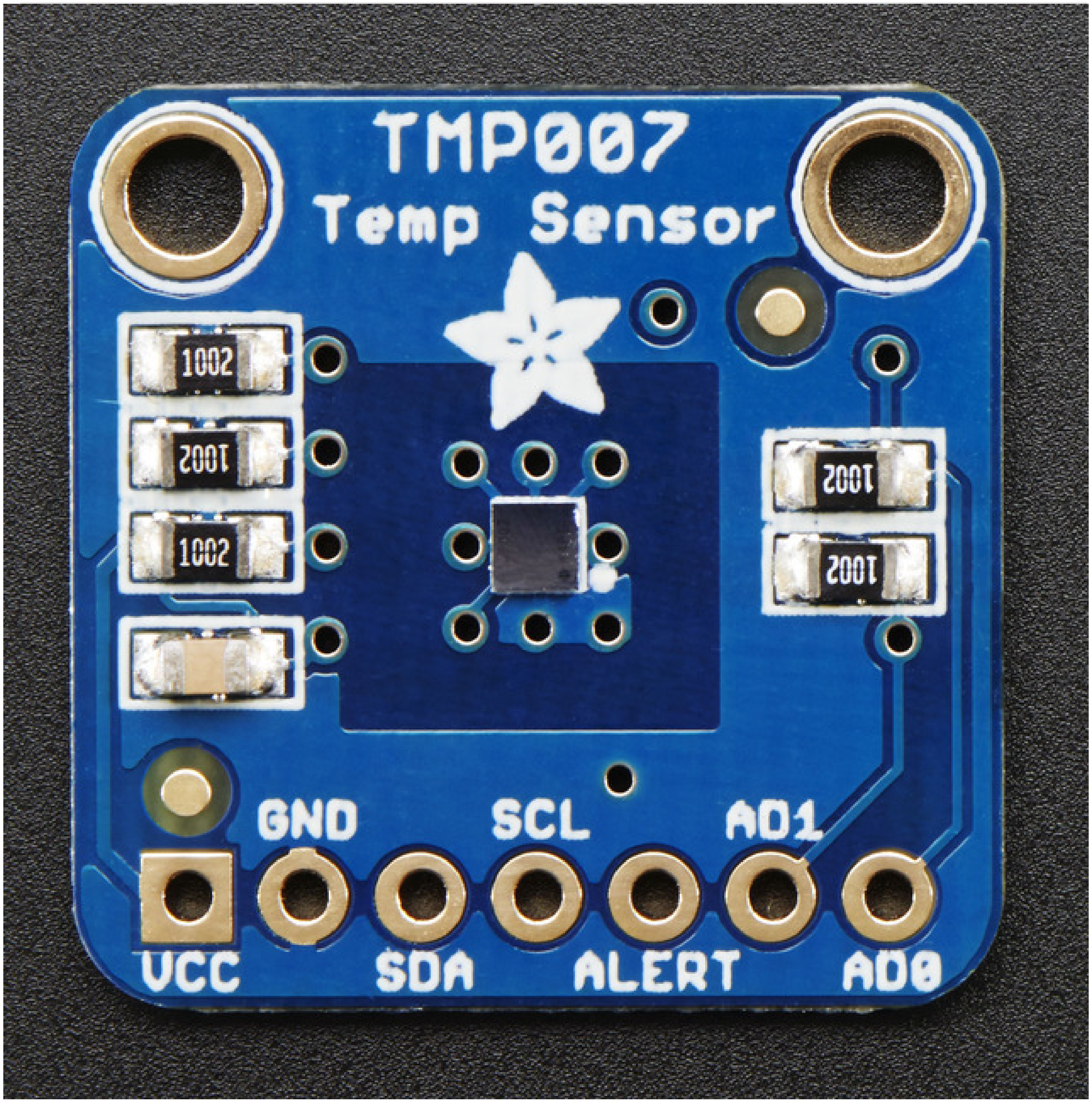


Simply point the sensor towards what you want to measure and it will detect the temperature by absorbing IR waves emitted. The embedded thermopile sensor generates a very very small voltage depending on how much IR there is, and using some math, that micro voltage can be used to calculate the temperature. It also takes the measurement over an area so it can be handy for determining the average temperature of something.



This sensor comes as a ultra-small 0.5mm pitch BGA, too hard to solder by hand. So we stuck it on an easy-to-work-with breakout board. The sensor works with 2.5V to 5V logic so it requires no logic level shifting. There are two address pins and using a funky method of connecting the pins you can have up to 8 TMP007's connected to one i2c bus. We also include a small piece of 0.1" breakaway header so you can easily solder to and use this sensor on a breadboard. Two mounting holes make it easy to attach to an enclosure.

Pinouts



The TMP007 is a very straight-forward sensor, lets go thru all the pins so you can understand what you need to connect to get started

(http://adafru.it/dMZ)Power Pins

- **VCC** - This is the positive power and logic level pin. It can be 2.2-5.5VDC, so fine for use with 3 or 5V logic. Power VCC with whatever logic level you plan to use on the i2c lines.
- **GND** - this is the ground power and logic reference pin.

(http://adafru.it/dMZ)I2C Data Pins

- **SCL** - this is the I2C clock pin. There's a 10K pull-up already on the board, so connect this directly to the i2c master clock pin on your microcontroller
- **SDA** - this is the I2C data pin. There's a 10K pull-up already on the board, so connect this directly to the i2c master data pin on your microcontroller

(http://adafru.it/dMZ)Optional Pins

These are pins you don't need to connect to unless you want to!

- **Alert** - This is the interrupt/alert pin from the TMP007. The chip has some capability to 'alert' you if the chip temperature goes above or below a set amount. This output can trigger to let you know. We don't have library support for this pin, so check the datasheet for more information.
- **AD0 AD1** - These are the address select pins. Since you can only have one device with a given address on an i2c bus, there must be a way to adjust the address if you want to put more than one TMP on a shared i2c bus. The AD0/AD1 pins set the bottom three pins of the i2c address. There are 10K pull-down resistors on the board.

The default address of the TMP007 is **0x40**. By connecting the address pins as in the following table, you can generate any address between 0x40 and 0x47

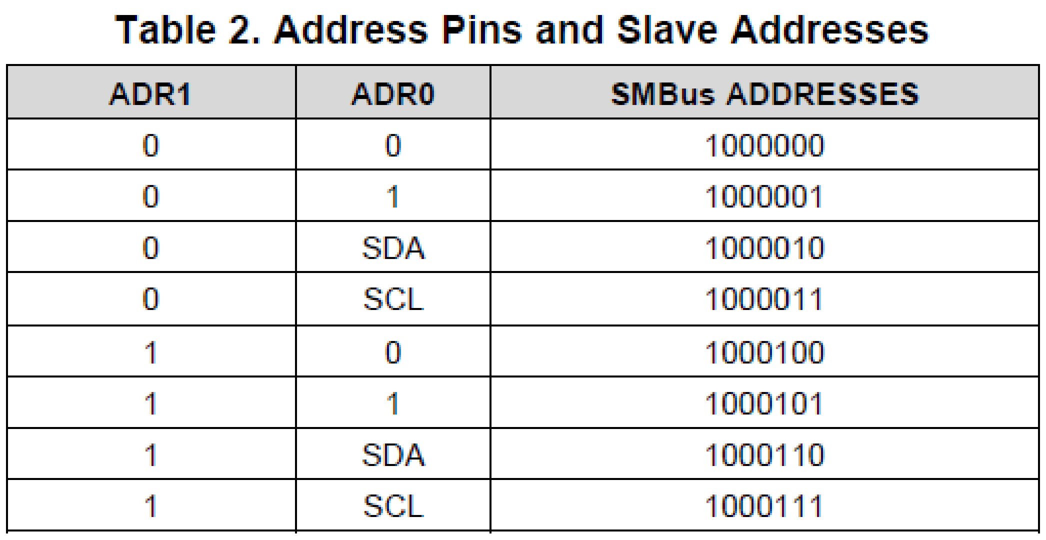
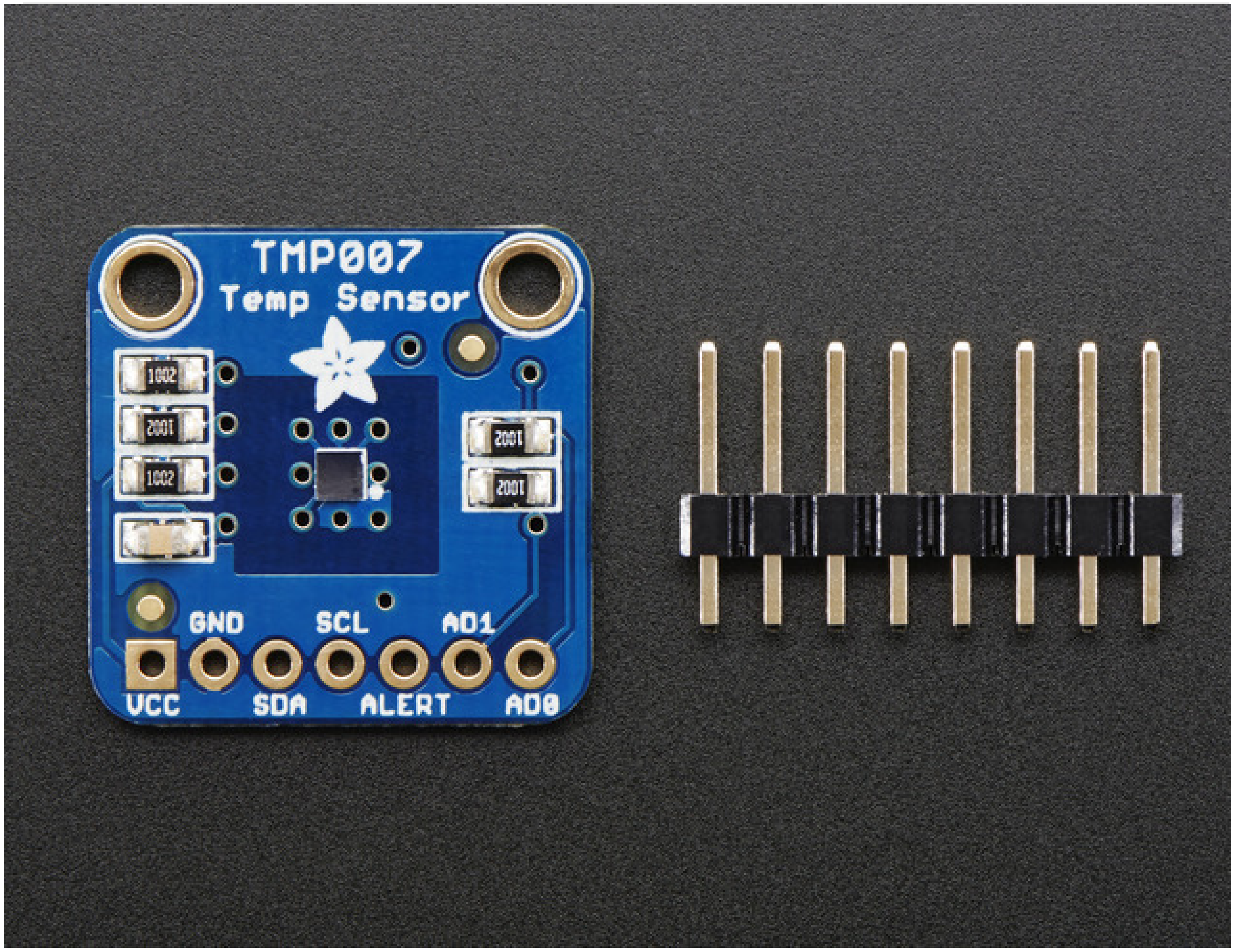
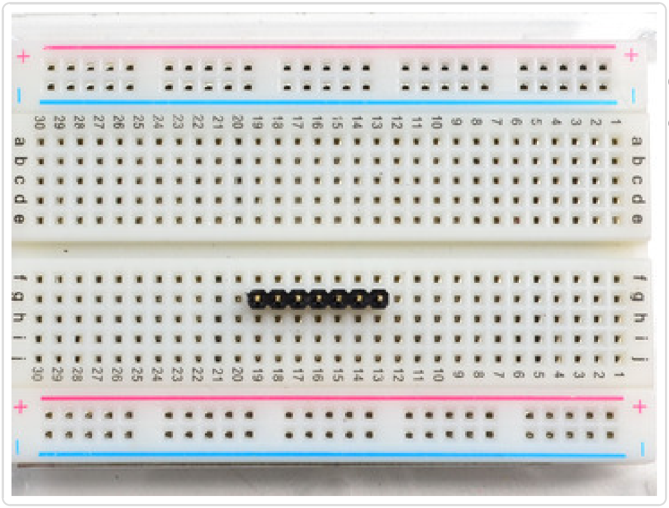


Table 2. Address Pins and Slave Addresses

ADR1	ADR0	SMBus ADDRESSES
0	0	1000000
0	1	1000001
0	SDA	1000010
0	SCL	1000011
1	0	1000100
1	1	1000101
1	SDA	1000110
1	SCL	1000111

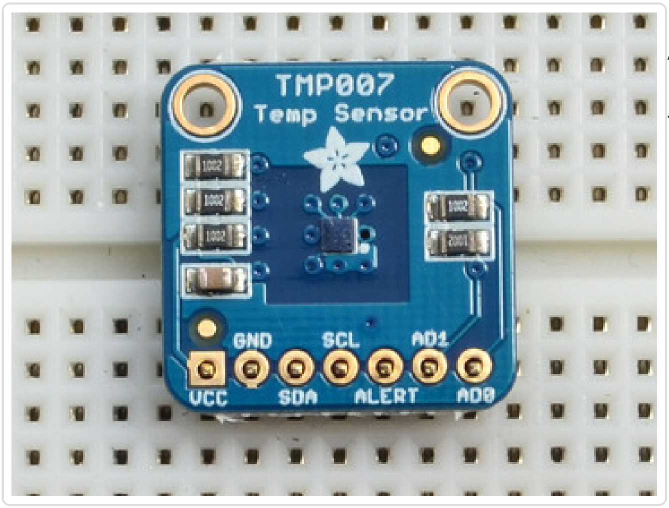
Assembly





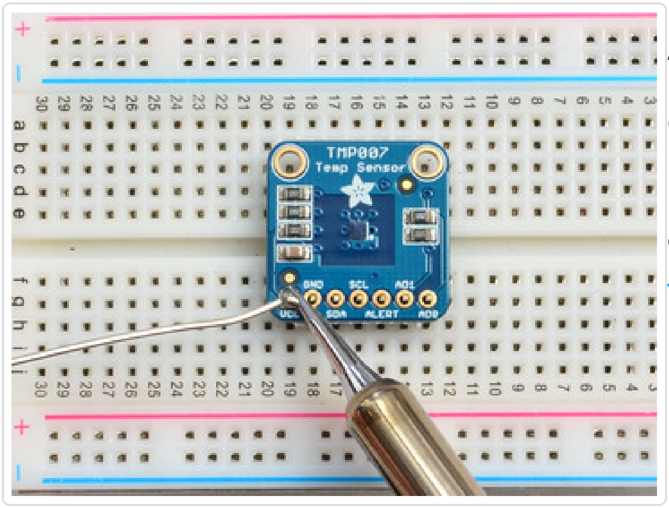
Prepare the header strip:

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:

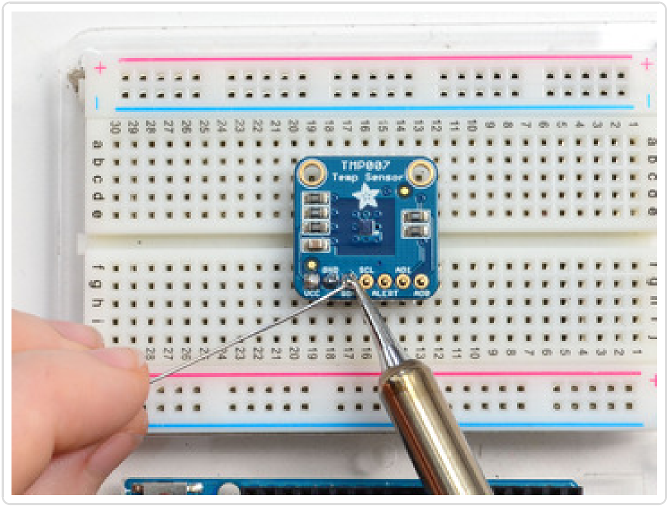
Place the breakout board over the pins so that the short pins poke through the breakout pads

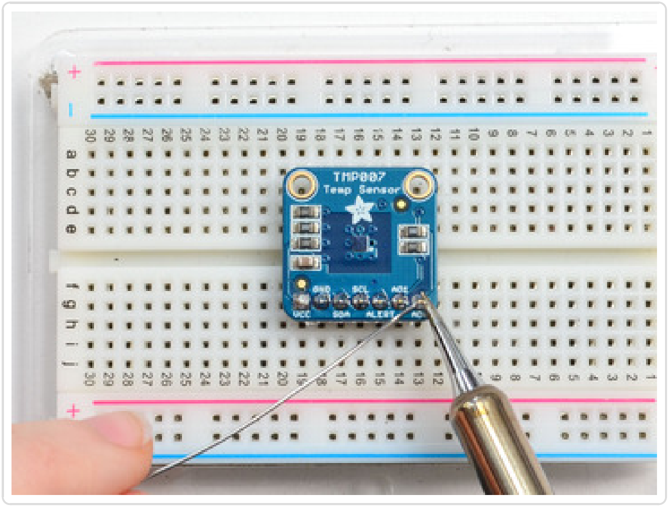


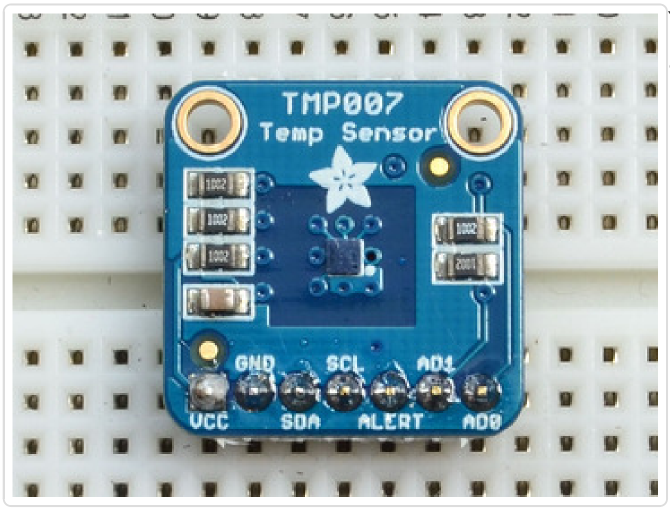
And Solder!

Be sure to solder all pins for reliable electrical contact.

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





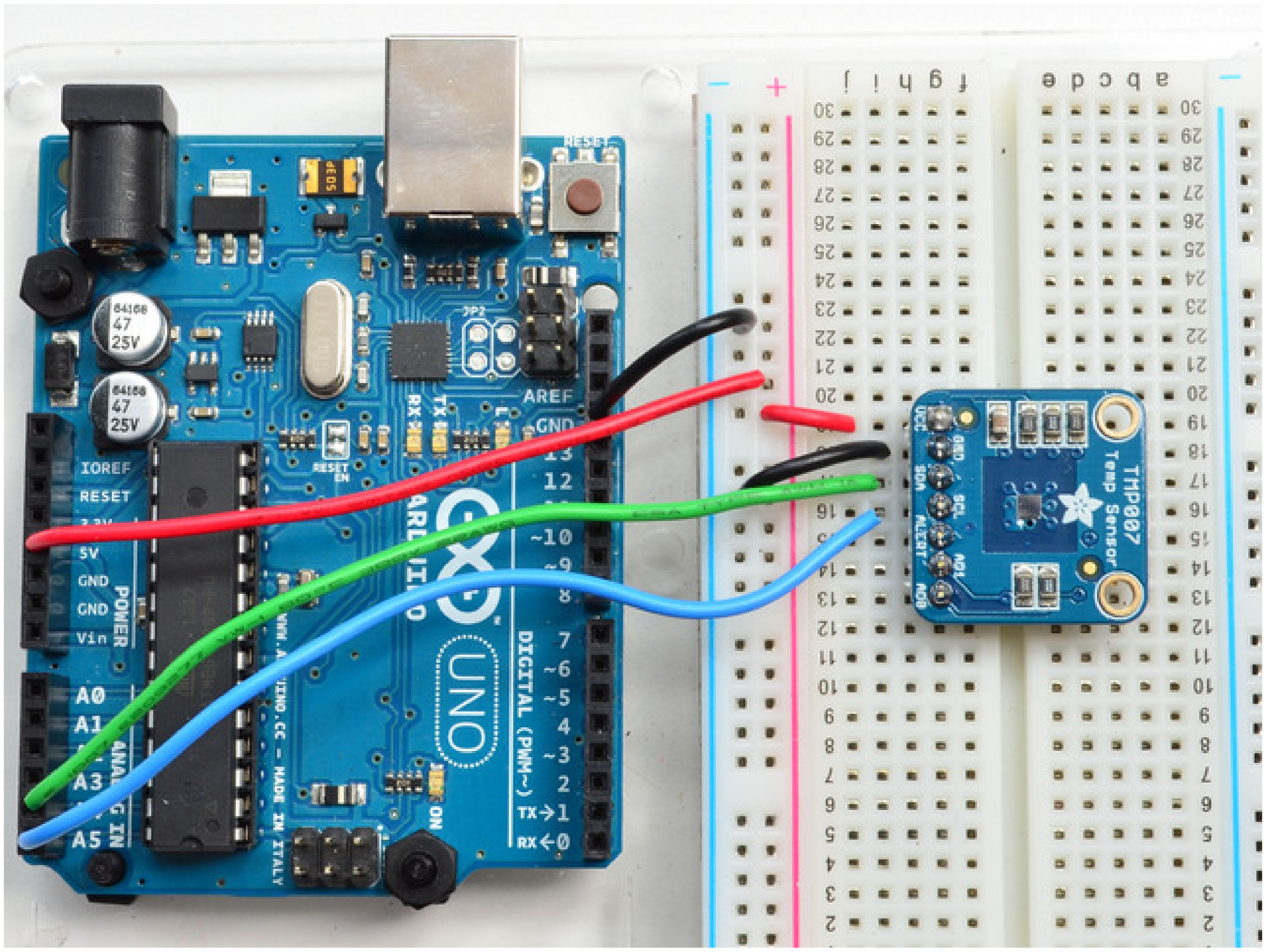


You're done! Check your solder joints visually and continue onto the next steps

Wiring & Test

Arduino Wiring

You can easily wire this sensor to any microcontroller, we'll be using an Arduino



- Connect **Vdd** 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 **SCL** 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** 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**

The TMP007 has a default I2C address of **0x40** but you can set the address to any of 8 values between 0x40 and 0x47 so you can have up to 8 of these sensors all sharing the same SCL/SDA pins.

Download Adafruit_TMP007

To begin reading sensor data, you will need to download Adafruit_TMP007 from our github repository (http://adafru.it/dN6). You can do that by visiting the github repo and manually downloading or, easier, just click this button to download the zip

Download Adafruit_TMP007 Library

http://adafru.it/dN7

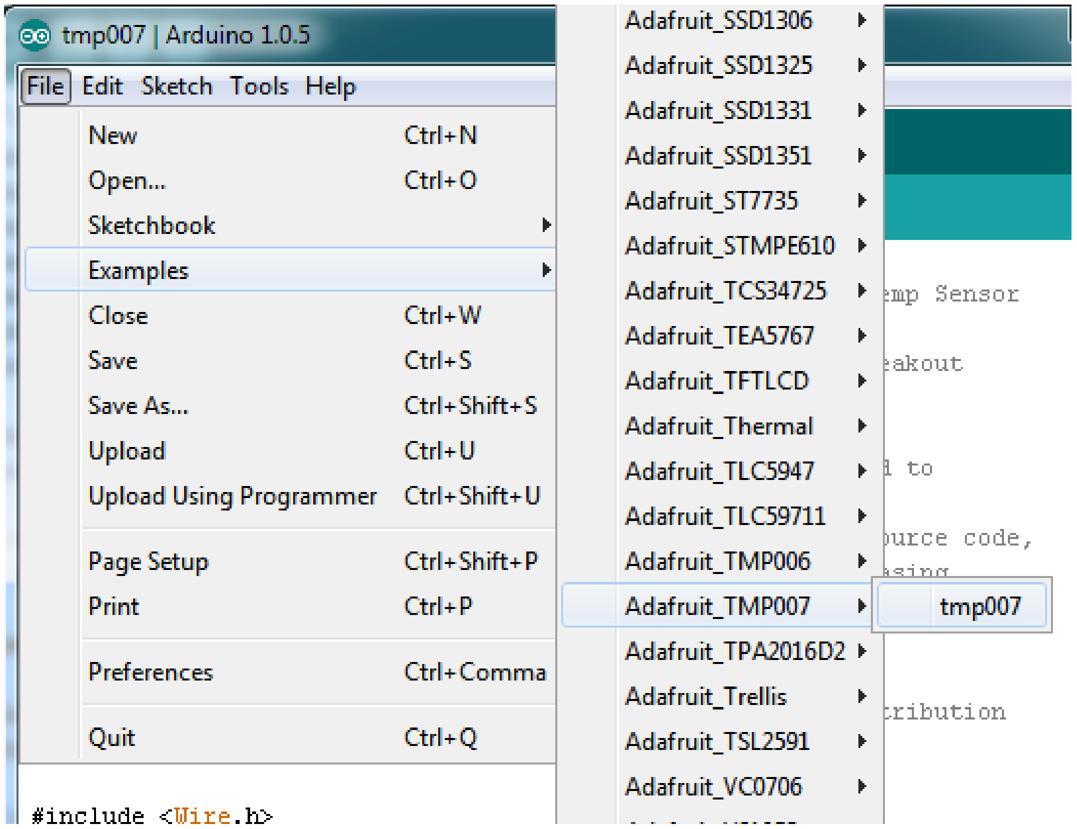
Rename the uncompressed folder **Adafruit_TMP007** and check that the **Adafruit_TMP007** folder contains **Adafruit_TMP007.cpp** and **Adafruit_TMP007.h**

Place the **Adafruit_TMP007** 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 (http://adafru.it/aYM)

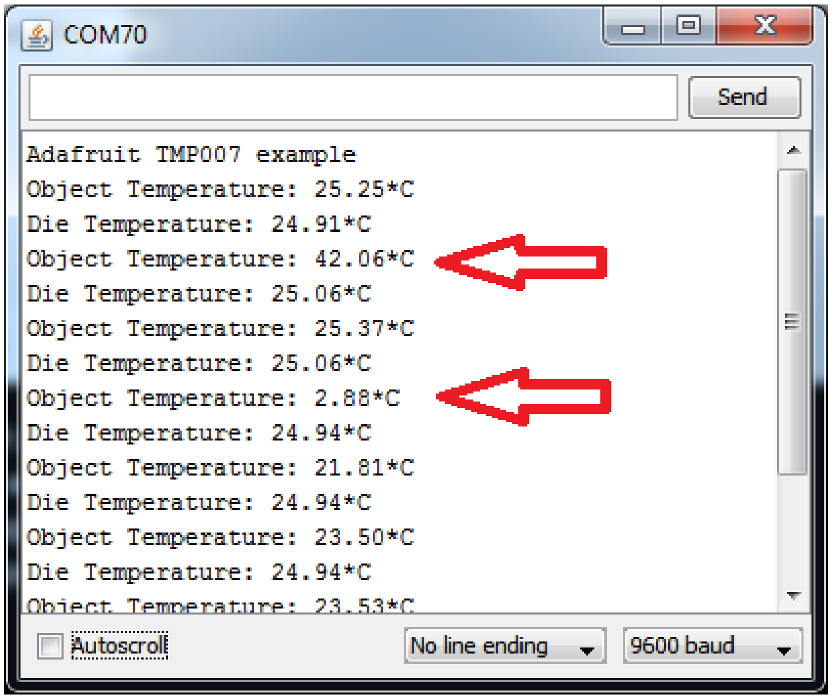
Load Demo

Open up **File->Examples->Adafruit_TMP007->tmp007** and upload to your Arduino wired up to the sensor



```
#include <Wire.h>
```

Thats it! Now open up the serial terminal window at 9600 speed to see the temperature in real time. You can try putting your hand or a cold glass of water over the sensor (not touching) to see the thermopile sensor adjust!



```
Adafruit TMP007 example
Object Temperature: 25.25*C
Die Temperature: 24.91*C
Object Temperature: 42.06*C
Die Temperature: 25.06*C
Object Temperature: 25.37*C
Die Temperature: 25.06*C
Object Temperature: 2.88*C
Die Temperature: 24.94*C
Object Temperature: 21.81*C
Die Temperature: 24.94*C
Object Temperature: 23.50*C
Die Temperature: 24.94*C
Object Temperature: 23.53*C
```

Library Reference
The TMP007 library is pretty straight forward! Start by creating the Adafruit_TMP007 object with:

```
Adafruit_TMP007 tmp007;
```

Or with an i2c address assigned

```
Adafruit_TMP007 tmp007(0x41);  // start with a diferent i2c address!
```

Then you can initialize & configure

```
tmp007.begin()
```

or, to set the samples/reading with:

```
tmp007.begin(TMP007_CFG_1SAMPLE)
```

We suggest the default, 16 samples for best accuracy. **begin()** will return true or false based on whether it found the sensor, check it using an if statment like so:

```
if (! tmp007.begin()) {
  Serial.println("No sensor found");
  while (1);
}
```

Now you can read the Die temperature (the temperature of the physical sensor itself) and Object temperature (the temperature of the stuff in front of the sensor)

```
tmp007.readObjTempC();
tmp007.readDieTempC();
```

The readings are floating point values, in degrees C.

Then wait 4 seconds between readings to get a new reading!

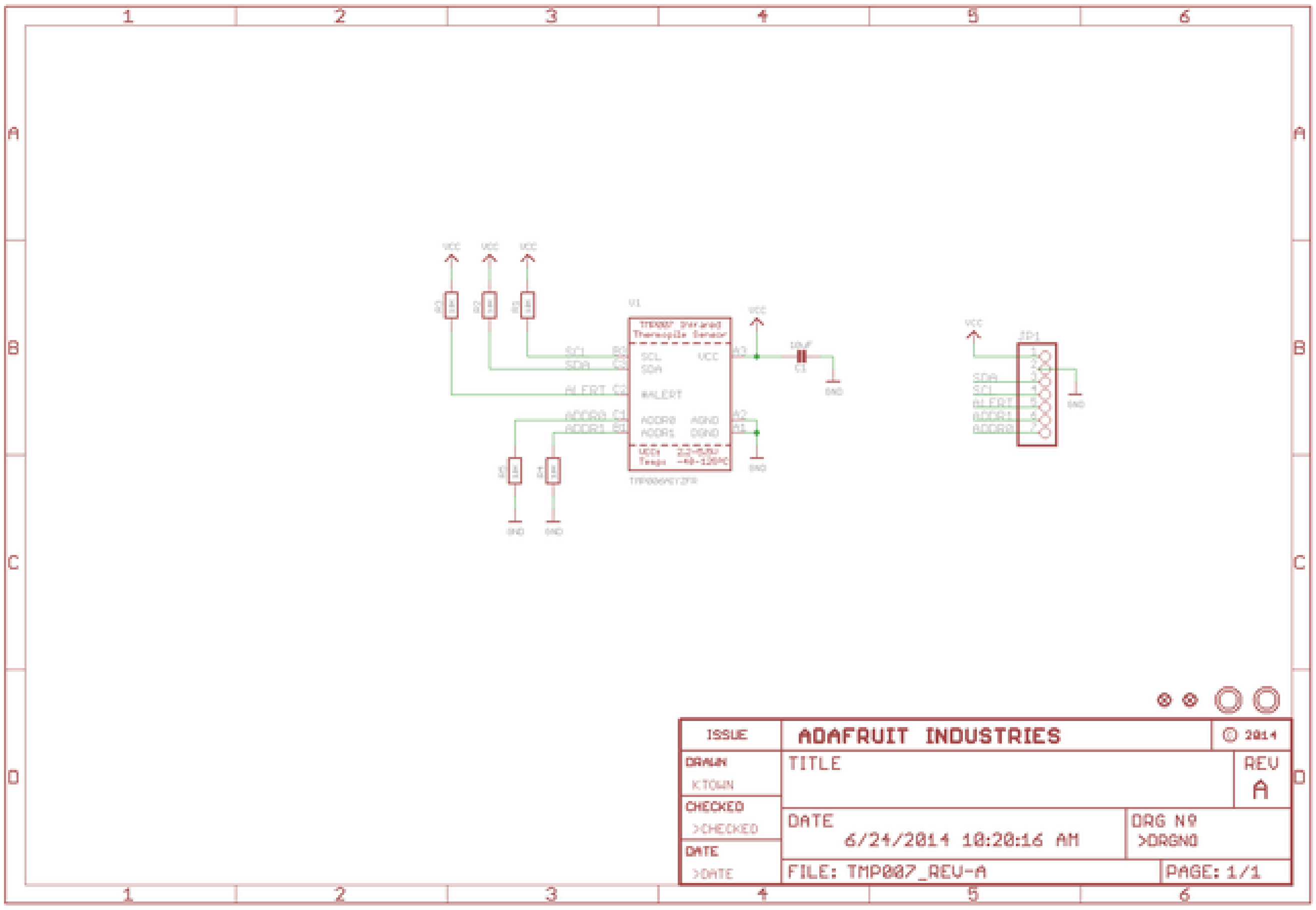
```
delay(4000); // 4 seconds per reading for 16 samples per reading
```

Downloads

Datasheet

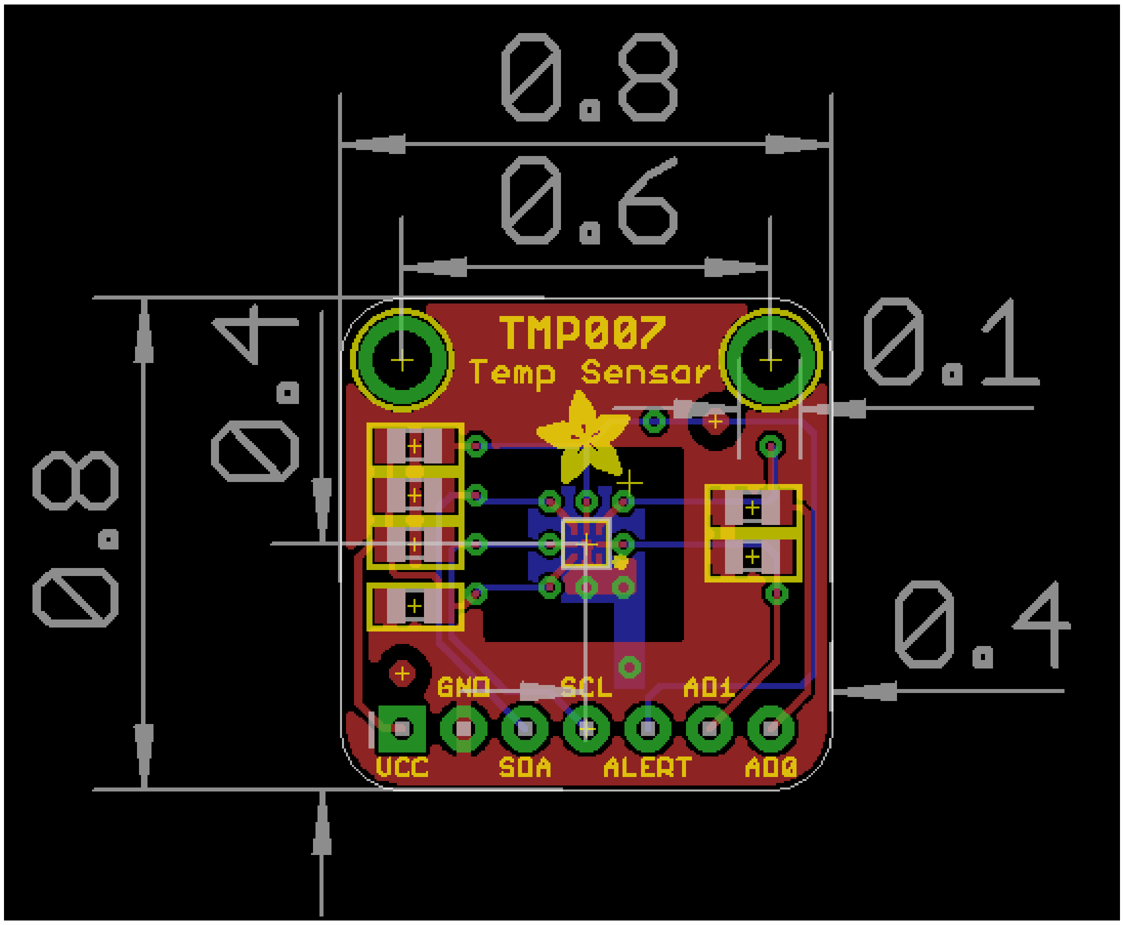
- TMP007 datasheet (http://adafru.it/dNg)
- TMP007 product page at TI (http://adafru.it/dNh)

Schematic



PCB Fabrication Print

Dimensions in Inches


0.8
0.6
0.8
0.4
0.1
0.4
TMP007
Temp Sensor
GND
SCL
AD1
VCC
SDA
ALERT
AD0