

Thermocouple

Created by Ladyada



Last updated on 2014-02-14 03:15:06 PM EST

Guide Contents

Guide Contents	2
Overview	3
Some Basic Stats	4
This is for a K-type thermocouple with glass overbraiding	4
Wiring a Thermocouple	5
Using a Thermocouple	6
Arduino Library	7
Adding a Display	8
F.A.Q.	10
Project Examples	13
Downloads	16
Buy a Thermocouple	17

Overview

A thermocouple is a kind of temperature sensor.

Unlike [semiconductor temperature sensors such as the TMP36 \(http://adafru.it/ckT\)](http://adafru.it/ckT) , thermocouples have no electronics inside them, they are simply made by welding together two metal wires. Because of a physical effect of two joined metals, there is a slight but measurable voltage across the wires that increases with temperature. The type of metals used affect the voltage range, cost and sensitivity, which is why we have a few different kinds of thermocouples. The main improvement of using a thermocouple over a semiconductor sensor or thermistor is that the temperature range is very much increased. For example, the TMP36 can go from -50 to 150°C, after that the chip itself can be damaged. Common thermocouples on the other hand, can go from -200°C to 1350°C (K type) and there are ones that can go above 2300°C!



Thermocouples are often used in HVAC systems, heaters and boilers, kilns, etc. There are a few different kinds but this tutorial will discuss K type, which are very common and easier to interface with.

One difficulty in using them is that the voltage to be measured is very small, with changes of about 50 μV per $^{\circ}\text{C}$ (a μV is 1/1000000 Volts). While it is possible to read these voltages using a clean power supply and nice op-amps, there are other complications such as a non-linear response (its not always 50 $\mu\text{V}/^{\circ}\text{C}$) and cold-temperature compensation (the effect measured is only a differential and there must be a reference, just as ground is a reference for voltage). For that reason, we suggest only using an interface chip that will do the heavy lifting for you, allow you to easily integrate the sensor without as much pain. In this tutorial we will use a MAX6675 K-thermocouple interface chip which doesn't even require an ADC, spitting out a nice digital data signal of the temperature.

Some Basic Stats

This is for a K-type thermocouple with glass overbraiding

- **Size:** 24 gauge, 1 meter long (you can cut it down if desired)
- **Price:** \$10 at the [adafruit store](http://adafruit.com) (<http://adafruit.com>)
- **Temperature range:** -100°C to 500°C / -150 to 900°F (After this the glass overbraiding may be damaged)
- **Output range:** -6 to +20mV
- **Precision:** +-2°C
- **Requires an amplifier such as MAX31855** (<http://adafruit.com/269>)
- **Interface:** MAX6675 (<http://adafruit.com/aLd>) (discontinued) MAX31855 (<http://adafruit.com/269>), or AD595 (<http://adafruit.com/aLe>) (analog)
- **K Thermocouple Datasheet** (<http://adafruit.com/aLf>)
- **MAX6675 Datasheet** (<http://adafruit.com/aLg>)
- **MAX31855 Datasheet** (<http://adafruit.com/aLh>)

Wiring a Thermocouple

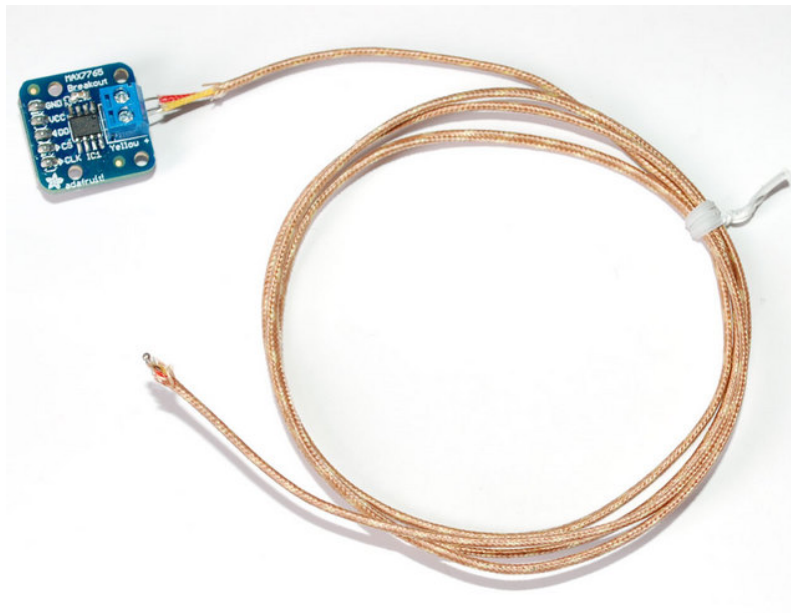
As we mentioned before, trying to actually measure the voltage across the wires will be very difficult for most people, which is why we strongly suggest using a thermocouple interface chip. The nicest one we've seen so far is the MAX6675 (and its replacement version called the MAX31855) which unfortunately is only available in SOIC package. While not too difficult to solder, [we nevertheless have in the shop a breakout board that is ready to go \(http://adafru.it/269\)](http://adafru.it/269).

First thing to determine is which wire is which. As you recall, thermocouples are made by welding together two wires, the chip reads the voltage difference between the two. One is the negative (for K-type its made of Alumel) and the other positive (ditto, Chromel). Luckily the wires are color coded, and almost all of the time you'll find the Alumel is red and the Chromel is yellow.

Connect the leads as required to your amplifier:

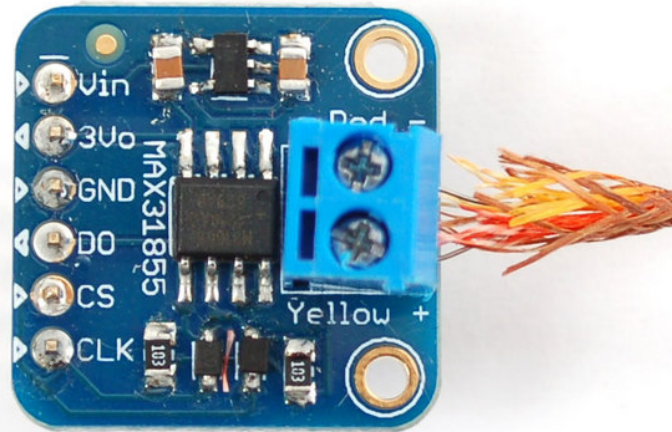
The MAX6675 and MAX31855 thermocouple amplifiers are not compatible with grounded thermocouples.

We've seen some K type thermocouples where the leads were marked incorrectly, so if you find that the thermocouple temperature goes down instead of up when heated, try swapping the red and yellow wires



Using a Thermocouple

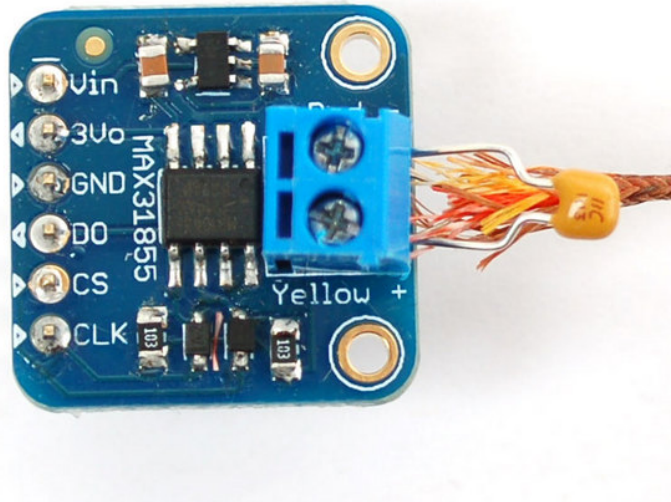
If you're using an AD595 interface chip, you can simply connect the voltage output to an analog input on your microcontroller and do some basic math to multiply the 10 mV/°C input into numerical output.



If you're planning to use the MAX6675/MAX31855, there's a little more work to be done. First off, **Vin** and **GND** must connect to a 3-5V supply. Then the three data pins must connect to digital IO pins:

- **CLK** (clock) is an input to the MAX6675/MAX31855 (output from microcontroller) which indicates when to present another bit of data
- **DO** (data out) is an output from the MAX6675/MAX31855 (input to the microcontroller) which carries each bit of data
- **CS** (chip select) is an input to the MAX6675/MAX31855 (output from the microcontroller) which tells the chip when its time to read the thermocouple and output more data.

In the beginning of our sketches, we define these pins. For our examples **DO** connects to digital 3, **CS** connects to digital 4, and **CLK** connects to pin 5



If you are using the MAX31855 v1.0 in a noisy environment, you may need to add a 0.01uF capacitor across the thermocouple leads.

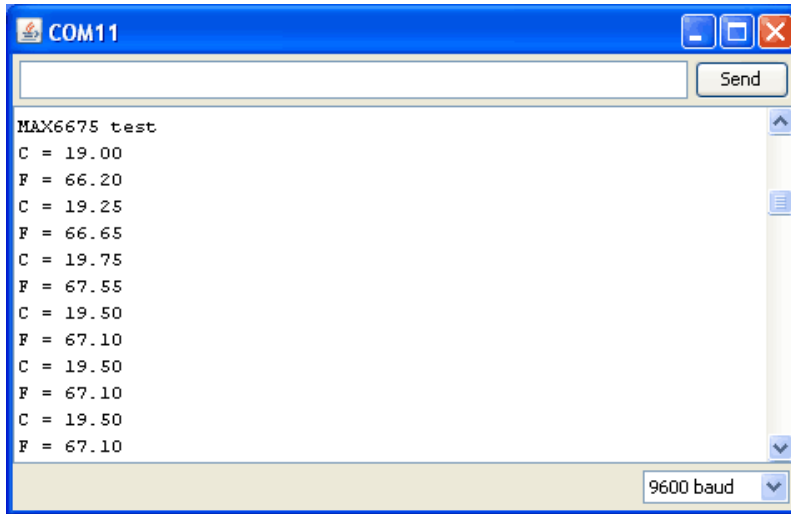
The MAX31855 does not support grounded thermocouples - if the sensor touches ground the chip will return an error

Arduino Library

If you have an older MAX6675 breakout, [download the MAX6675 Arduino library code \(http://adafru.it/aLi\)](http://adafru.it/aLi) by going to the github page and clicking Download Source. Then uncompress the folder and rename it **MAX6675** and [install it into the library folder according to our handy tutorial \(http://adafru.it/aYG\)](http://adafru.it/aYG).

If you have the newer MAX31855 breakout, [download the MAX31855 Arduino library code \(http://adafru.it/aLj\)](http://adafru.it/aLj) by going to the github page and clicking Download Source. Then uncompress the folder and rename it **Adafruit MAX31855** and [install it into the library folder according to our handy tutorial \(http://adafru.it/aYG\)](http://adafru.it/aYG).

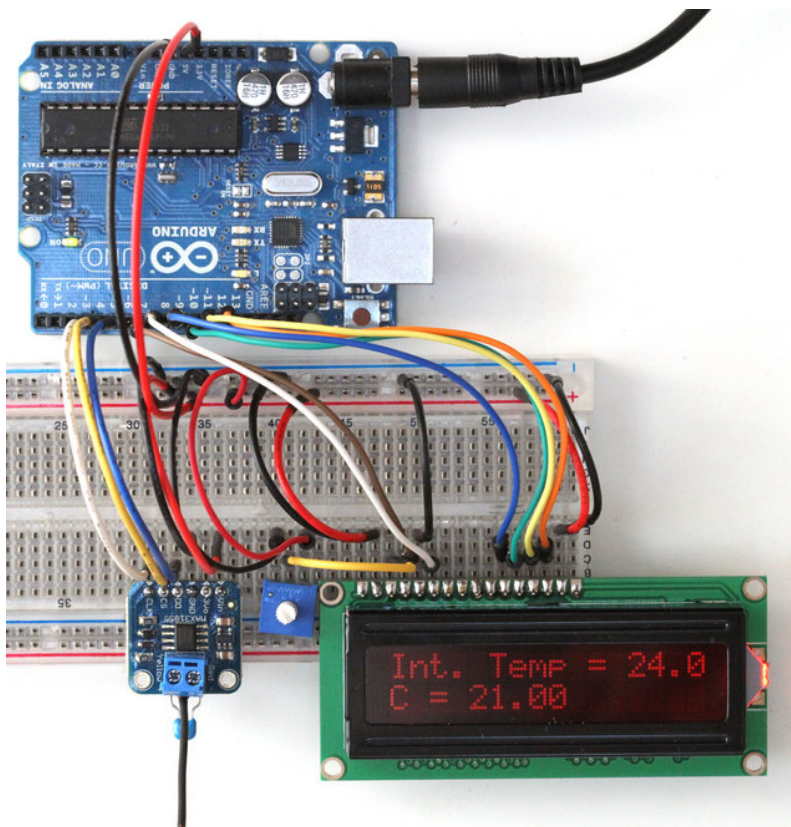
Restart the Arduino IDE and open up the **File->Examples->MAX6675/Adafruit MAX31855->serialthermocouple** sketch and upload it to your Arduino. Once uploaded, open up the serial port monitor to display the current temperatures in both Celsius and Fahrenheit



As you can see, its pretty simple to use the library, simply tell the sensor object what the clock, chip select and data pins are, then call **readCelsius()** or **readFahrenheit()** to get a floating point result.

Adding a Display

A common request is to have the temperature output onto a 'classic' character LCD such as the ones in this tutorial (<http://adafruit.it/c8r>).



For this wiring, we connected **CLK** to digital **3**, **CS** to digital **4** and **DO** to digital **5**. Once you get it working, you can change the pin connections in the sketch

We have an example sketch for this as well. [First get the LCD working by following our tutorial \(http://adafru.it/c8r\)](http://adafru.it/c8r). Now load up the new sketch **File->Examples->MAX31855>lcdthermocouple** and plug in the thermocouple module as we did in the serial thermocouple test, you'll see the internal temperature and the thermocouple temperature displayed in Celsius

F.A.Q.

The temperature of my thermocouple seems to be backwards! If I heat up the probe the reported temperature goes down

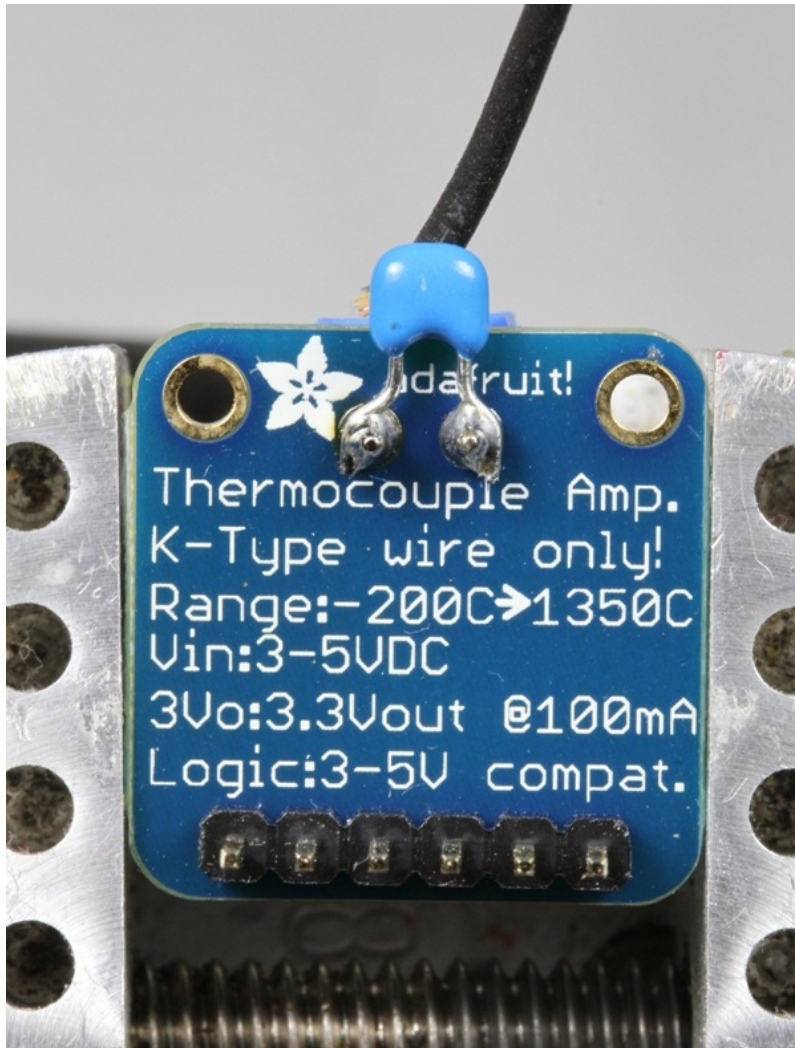
This is likely caused by the thermocouple wires being labeled incorrectly. Try swapping the two thermocouple leads, even if yellow and red wires are in the right slots - we've seen some thermocouple where the wire colors are wrong.

<div>This is likely caused by the thermocouple wires being labeled incorrectly. Try swapping the two thermocouple leads, even if yellow and red wires are in the right slots - we've seen some thermocouple where the wire colors are wrong.
 </div>

My MAX31855 output is really unstable and noisy - If I touch or move the probe the temperature output goes crazy

The MAX31855 is surprisingly sensitive, we've found a good way to fix this is to place a 0.01uF to 0.1uF capacitor across the thermocouple leads (that is, place the capacitor into the blue terminal block, or solder to the bottom as shown below).

The MAX31855 is surprisingly sensitive, we've found a good way to fix this is to place a 0.01uF to 0.1uF capacitor across the thermocouple leads (that is, place the capacitor into the blue terminal block, or solder to the bottom as shown below).



My thermocouple output has an offset / I have multiple thermocouples and they dont 'agree' on a temperature

K thermocouples are not precision temperature measurement devices! There will be offsets & differences between thermocouples. Most thermocouple thermometers have the offset corrected in software which is what we suggest. For precision temperature measurement, we suggest a 1% Thermistor.

K thermocouples are not precision temperature measurement devices! There will be offsets & differences between thermocouples. Most thermocouple thermometers have the offset corrected in software which is what we suggest. For precision temperature measurement, we suggest a 1% Thermistor.

How can I connect multiple thermocouples?

You can connect as many MAX31855's as you have pins. Simply share the CLK and DO pins of all the breakouts and have a unique CS pin for each one

Then you can create new thermocouples using the following style:

```
Adafruit_MAX31855 thermocouple1(thermoCLK, thermoCS1, thermoDO);
Adafruit_MAX31855 thermocouple2(thermoCLK, thermoCS2, thermoDO);
Adafruit_MAX31855 thermocouple3(thermoCLK, thermoCS3, thermoDO);
```

You can also try having same CS and CLK pins but all different DO pins

```
Adafruit_MAX31855 thermocouple1(thermoCLK, thermoCS, thermoDO1);
```

```
Adafruit_MAX31855 thermocouple2(thermoCLK, thermoCS, thermoDO2);  
Adafruit_MAX31855 thermocouple3(thermoCLK, thermoCS, thermoDO3);
```

You can connect as many MAX31855's as you have pins. Simply share the CLK and DO pins of all the breakouts and have a unique CS pin for each one. Then you can create new thermocouples using the following style:

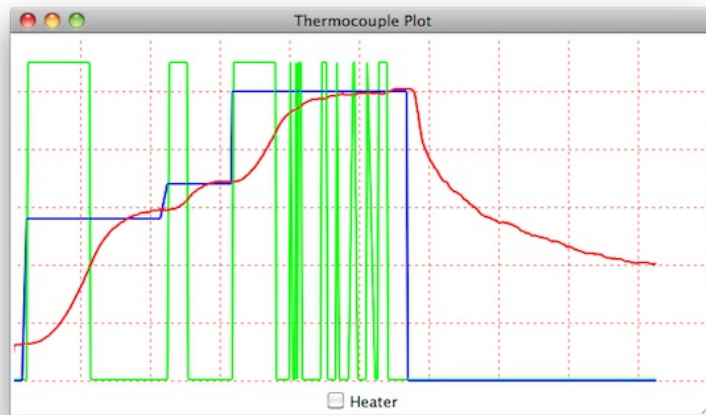
```
Adafruit_MAX31855 thermocouple1(thermoCLK, thermoCS1,  
thermoDO);  
Adafruit_MAX31855 thermocouple2(thermoCLK, thermoCS2, thermoDO);  
Adafruit_MAX31855 thermocouple3(thermoCLK, thermoCS3,  
thermoDO);
```

You can also try having same CS and CLK pins but all different DO pins

```
Adafruit_MAX31855 thermocouple1(thermoCLK, thermoCS,  
thermoDO1);  
Adafruit_MAX31855 thermocouple2(thermoCLK, thermoCS, thermoDO2);  
Adafruit_MAX31855 thermocouple3(thermoCLK, thermoCS,  
thermoDO3);
```

Project Examples

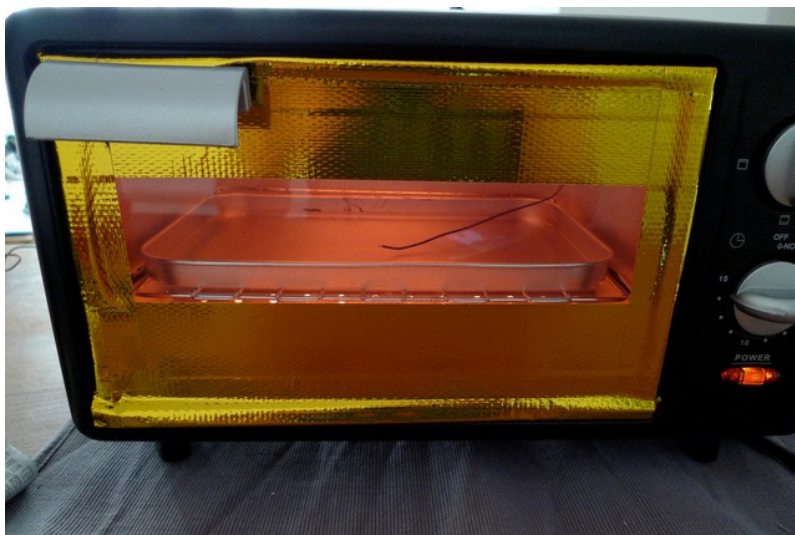
Need ideas? Check out these projects!



Jeelabs has a detailed walkthrough for a reflow controller (uses an AD595-type chip) (<http://adafru.it/aLI>)



Terran's Pi controlled coffee roaster (<http://adafru.it/aLm>)



RocketNumberNine's reflow toaster project (<http://adafru.it/aLn>)

Downloads

MAX6675 Schematic and layout files can be found at GitHub - click [Download Source](#) to get the zip! (<http://adafru.it/aLo>)

MAX31855 Schematic and layout files can be found at GitHub - click [Download Source](#) to get the zip! (<http://adafru.it/aLp>)

Buy a Thermocouple

Buy a Thermocouple (<http://adafru.it/270>)