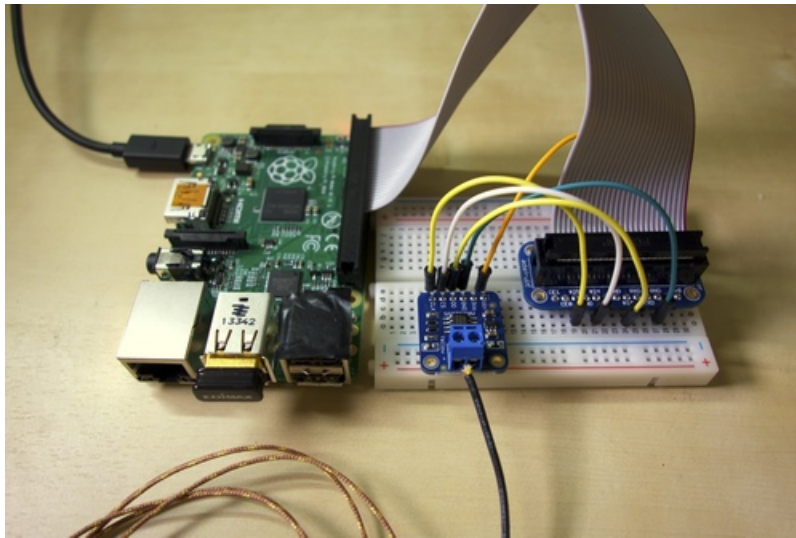




MAX31855 Thermocouple Sensor Python Library

Created by Tony DiCola

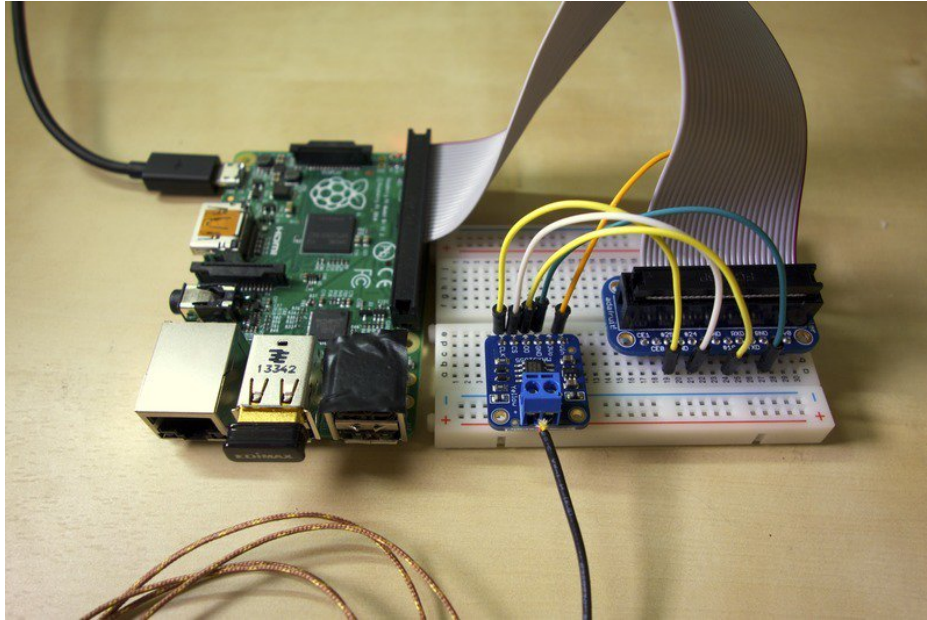


Last updated on 2019-04-02 04:50:18 PM UTC

Overview



This guide is no longer supported. For more information about using the MAX31855 with Python and CircuitPython, check out: <https://learn.adafruit.com/thermocouple>



Do you need fast and accurate temperature measurements over a wide range of values in a project? Consider using a thermocouple and the [MAX31855 thermocouple amplifier](http://adafru.it/269)! A thermocouple uses the Seebeck effect to turn differences in temperature into very small voltages which the MAX31855 can read and make available over an SPI connection. Using the [MAX31855 Python library](https://adafru.it/e02) you can add a thermocouple temperature sensor to your Raspberry Pi or BeagleBone Black project!

Before you get started make sure your Raspberry Pi is running the latest [Raspbian](https://adafru.it/dpb) or [Occidentalis](https://adafru.it/dvg) operating system, and your BeagleBone Black is running the latest [official Debian operating system](https://adafru.it/dUI). It will also help to familiarize yourself with [the thermocouple guide](https://adafru.it/qhE).

Hardware



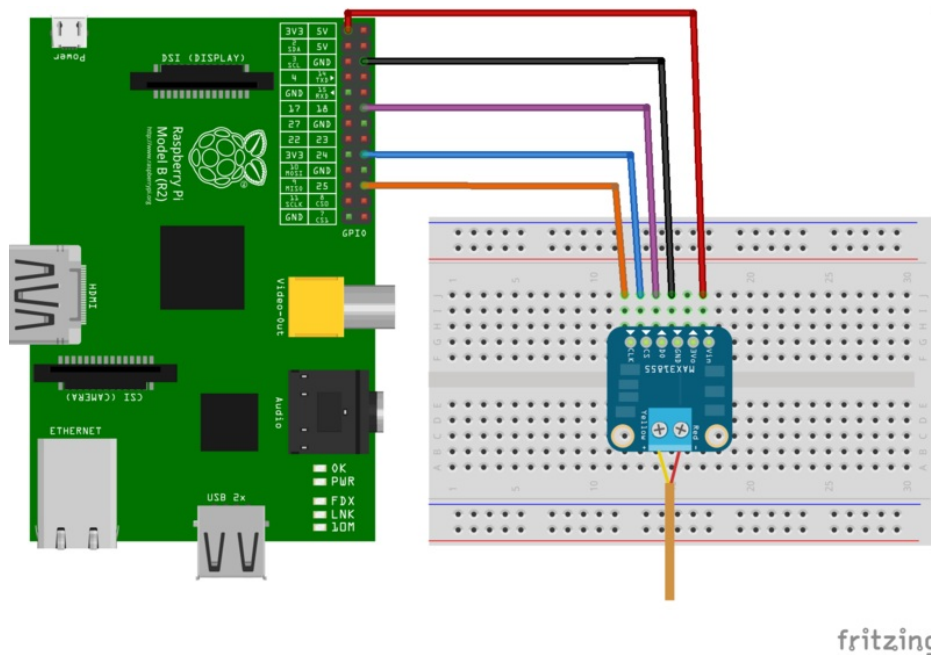
This guide is no longer supported. For more information about using the MAX31855 with Python and CircuitPython, check out: <https://learn.adafruit.com/thermocouple>

The MAX31855 board uses SPI to communicate with a Raspberry Pi or BeagleBone Black. You can use either software SPI on any 3 digital GPIO pins, or you can use hardware SPI on your board's dedicated SPI pins. If you aren't sure which option to choose, try software SPI since it's more flexible and speed isn't critical when reading the MAX31855 sensor.

Raspberry Pi

Software SPI

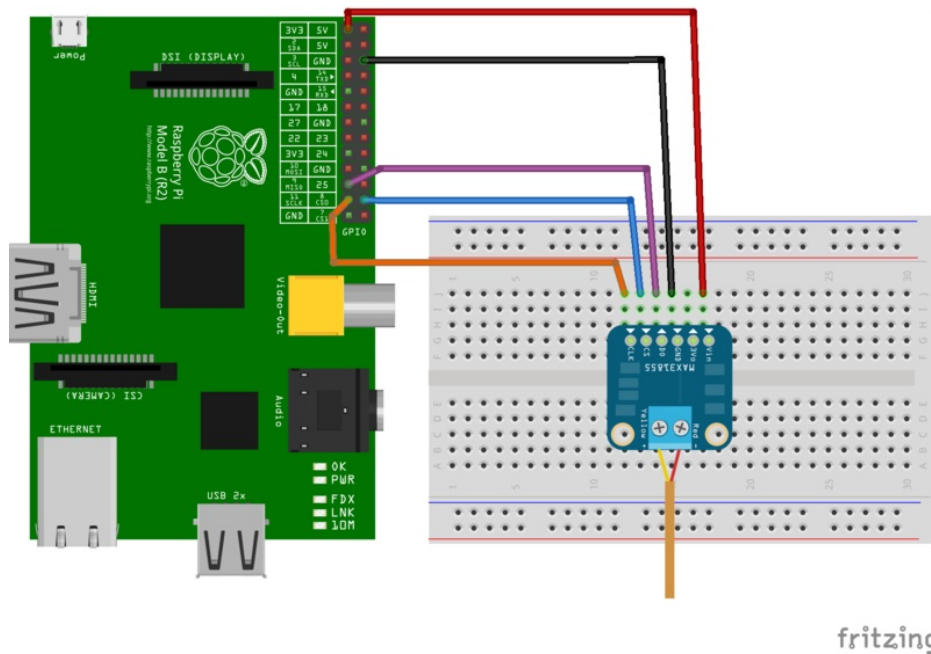
To use software SPI on a Raspberry Pi connect the MAX31855 to the Raspberry Pi as follows. Note that you can use any 3 digital IO pins for the CLK, CS, and DO pins.



- Connect Pi 3.3V to MAX31855 Vin.
- Connect Pi GND to MAX31855 GND.
- Connect Pi GPIO 18 to MAX31855 DO.
- Connect Pi GPIO 24 to MAX31855 CS.
- Connect Pi GPIO 25 to MAX31855 CLK.

Hardware SPI

To use hardware SPI on a Raspberry Pi connect to the MAX31855 as follows.



- Connect Pi 3.3V to MAX31855 Vin.
- Connect Pi GND to MAX31855 GND.
- Connect Pi MISO to MAX31855 DO.
- Connect Pi CS0 to MAX31855 CS.
- Connect Pi SCLK to MAX31855 CLK.

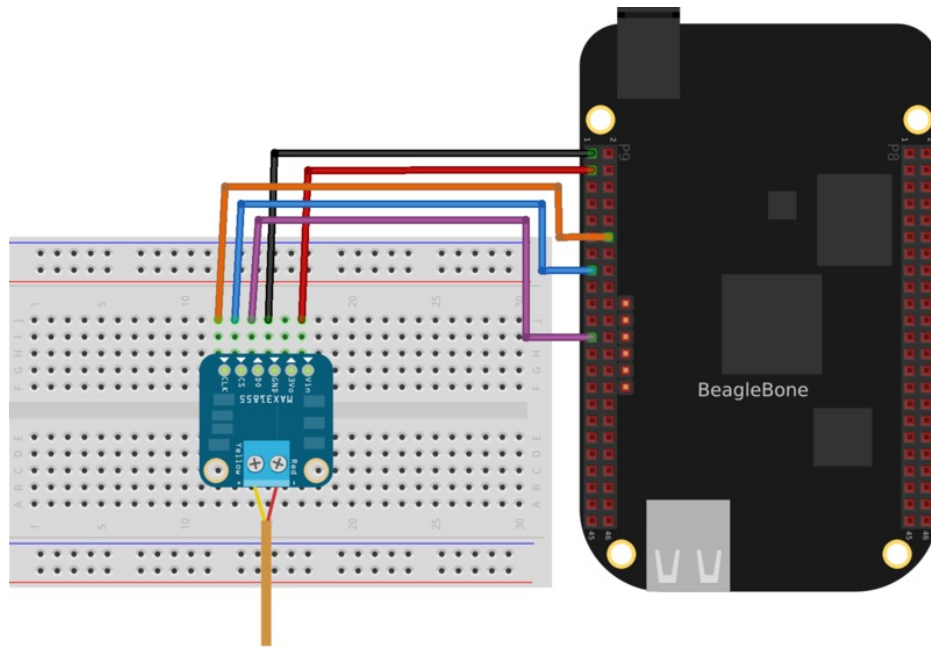
The wiring above will use the Raspberry Pi's hardware SPI bus to communicate with the MAX31855. If you haven't done so already with your Pi, make sure to [edit the blacklist.conf file \(https://adafru.it/dvD\)](https://adafru.it/dvD) to comment the line which disables SPI. Reboot your Pi and you should see the files `/dev/spidev0.0` and `/dev/spidev0.1` are now available.

BeagleBone Black

Software SPI

To use software SPI on a BeagleBone Black connect to the MAX31855 as follows. Note that you can use any 3 digital GPIO pins for the CLK, CS, and DO pins.

If you aren't familiar with how GPIO pins are numbered, check out [this guide on BeagleBone Black GPIO \(https://adafru.it/dCI\)](https://adafru.it/dCI).



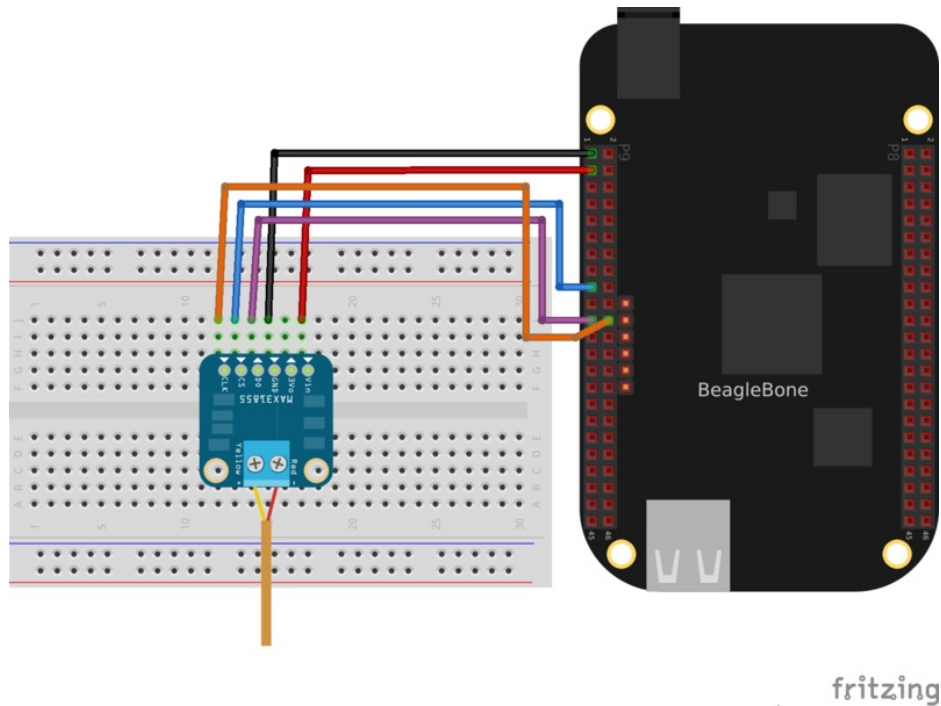
fritzing

- Connect **BeagleBone Black P9_1 DGND** to **MAX31855 GND**.
- Connect **BeagleBone Black P9_3 3.3V** to **MAX31855 Vin**.
- Connect **BeagleBone Black P9_12** to **MAX31855 CLK**.
- Connect **BeagleBone Black P9_15** to **MAX31855 CS**.
- Connect **BeagleBone Black P9_23** to **MAX31855 DO**.

Hardware SPI

To use hardware SPI with the BeagleBone Black connect the MAX31855 as follows.

If you aren't familiar with how GPIO pins are numbered, check out [this guide on BeagleBone Black GPIO \(https://adafru.it/dCI\)](https://adafru.it/dCI).



- Connect BeagleBone Black P9_1 DGND to MAX31855 GND.
- Connect BeagleBone Black P9_3 3.3V to MAX31855 Vin.
- Connect BeagleBone Black P9_22 SPI0 CLK to MAX31855 CLK.
- Connect BeagleBone Black P9_17 SPI0 CS to MAX31855 CS.
- Connect BeagleBone Black P9_21 SPI0 MISO to MAX31855 DO.

The wiring above assumes using a hardware SPI interface on the BeagleBone Black, specifically `/dev/spidev1.0`. Before you can use this SPI interface you must [enable a device tree overlay \(https://adafru.it/dp6\)](https://adafru.it/dp6) to turn on the SPI pin functionality. The easiest way to enable this device tree overlay is to configure the BeagleBone Black to load the overlay automatically on boot.

With the BeagleBone Black connected to your computer over USB, open the USB mass storage drive named 'boot' and edit the file `uEnv.txt` on the drive. Add the following line to the file:

```
optargs=capemgr.enable_partno=BB-SPIDEV0
```

NOTE: Be careful editing the `uEnv.txt` file on Windows, as changing the line endings can cause your BeagleBone Black not to boot and require an OS reinstall! The safest option is to connect to the BeagleBone Black in a command window and [follow the steps at the end of this page to mount and edit uEnv.txt on the BeagleBone Black \(https://adafru.it/dEK\)](https://adafru.it/dEK).

Reboot your device and you should see the files `/dev/spidev1.0` and `/dev/spidev1.1` now exist.

Software



This guide is no longer supported. For more information about using the MAX31855 with Python and CircuitPython, check out: <https://learn.adafruit.com/thermocouple>

To install and use the [MAX31855 Python library \(https://adafru.it/e02\)](https://adafru.it/e02) follow the steps below.

Before you get started make sure your board is connected to the internet through an ethernet or wireless connection so you can download dependencies.

You'll also want to be familiar with connecting to a [Raspberry Pi \(https://adafru.it/jvB\)](https://adafru.it/jvB) or [BeagleBone Black \(https://adafru.it/ne9\)](https://adafru.it/ne9) terminal with SSH.

Dependencies

First install dependencies by executing in a terminal:

```
sudo apt-get update
sudo apt-get install build-essential python-dev python-pip python-smbus git
```

You can ignore warnings about dependencies which are already installed.

Raspberry Pi

On a Raspberry Pi execute the following to make sure the RPi.GPIO library is installed:

```
sudo pip install RPi.GPIO
```

BeagleBone Black

On a BeagleBone Black execute the following to make sure the Adafruit_BBIO library is installed:

```
sudo pip install Adafruit_BBIO
```

Library Install

Next download the MAX31855 Python library to your home directory and install it by executing:

```
cd ~
git clone https://github.com/adafruit/Adafruit_Python_MAX31855.git
cd Adafruit_Python_MAX31855
sudo python setup.py install
```

That's all you need to do to install the Python library!

Usage

To learn how to use the MAX31855 Python library you can run and review an example program included with the library. First navigate to the **examples** directory and open the **simpletest.py** script in a text editor. Scroll down to the

part of the code which configures the connection with the MAX31855 board:

```
# Uncomment one of the blocks of code below to configure your Pi or BBB to use
# software or hardware SPI.

# Raspberry Pi software SPI configuration.
CLK = 25
CS = 24
D0 = 18
sensor = MAX31855.MAX31855(CLK, CS, D0)

# Raspberry Pi hardware SPI configuration.
#SPI_PORT = 0
#SPI_DEVICE = 0
#sensor = MAX31855.MAX31855(spi=SPI.SpiDev(SPI_PORT, SPI_DEVICE, max_speed_hz=5000000))

# BeagleBone Black software SPI configuration.
#CLK = 'P9_12'
#CS = 'P9_15'
#D0 = 'P9_23'
#sensor = MAX31855.MAX31855(CLK, CS, D0)

# BeagleBone Black hardware SPI configuration.
#SPI_PORT = 1
#SPI_DEVICE = 0
#sensor = MAX31855.MAX31855(spi=SPI.SpiDev(SPI_PORT, SPI_DEVICE, max_speed_hz=5000000))
```

By default the configuration to use a Raspberry Pi with software SPI is uncommented, however you can comment that block and uncomment a different block to use a different connection type.

If you've changed the pins which are connected to the hardware make sure to update the pin values in the code here!

Once the right block of configuration for your hardware is uncommented, save the file and exit the text editor. Then run the example by executing:

```
sudo python simpletest.py
```

If everything goes well you should see the thermocouple and internal temperature displayed every second:

```
Press Ctrl-C to quit.
Thermocouple Temperature: 22.000°C / 71.600°F
  Internal Temperature: 23.312°C / 73.963°F
Thermocouple Temperature: 22.250°C / 72.050°F
  Internal Temperature: 23.375°C / 74.075°F
Thermocouple Temperature: 27.500°C / 81.500°F
  Internal Temperature: 23.312°C / 73.963°F
...
```



If you get the reported temperature going up when the thermocouple is cooled, and vice versa - try swapping the thermocouple red/yellow wires!

If you see an error make sure you're running the program as root with the sudo command, and that the dependencies & library were installed successfully.

To understand the usage, open **simpletest.py** in a text editor and follow along with the description of the code below.

```
import Adafruit_MAX31855.MAX31855 as MAX31855
```

First the MAX31855 module is imported with a Python import statement.

```
# Uncomment one of the blocks of code below to configure your Pi or BBB to use
# software or hardware SPI.

# Raspberry Pi software SPI configuration.
CLK = 25
CS = 24
DO = 18
sensor = MAX31855.MAX31855(CLK, CS, DO)

# Raspberry Pi hardware SPI configuration.
#SPI_PORT = 0
#SPI_DEVICE = 0
#sensor = MAX31855.MAX31855(spi=SPI.SpiDev(SPI_PORT, SPI_DEVICE, max_speed_hz=5000000))

# BeagleBone Black software SPI configuration.
#CLK = 'P9_12'
#CS = 'P9_15'
#DO = 'P9_23'
#sensor = MAX31855.MAX31855(CLK, CS, DO)

# BeagleBone Black hardware SPI configuration.
#SPI_PORT = 1
#SPI_DEVICE = 0
#sensor = MAX31855.MAX31855(spi=SPI.SpiDev(SPI_PORT, SPI_DEVICE, max_speed_hz=5000000))
```

Next you see the hardware configuration which was described earlier. It's important to note that you can configure using software SPI by passing explicit CLK, DS, and DO digital GPIO pins, or you can configure hardware SPI by passing a SpiDev hardware SPI object to the MAX31855 class initializer.

```
# Loop printing measurements every second.
print 'Press Ctrl-C to quit.'
while True:
    temp = sensor.readTempC()
    internal = sensor.readInternalC()
    print 'Thermocouple Temperature: {0:0.3F}*C / {1:0.3F}*F'.format(temp, c_to_f(temp))
    print '    Internal Temperature: {0:0.3F}*C / {1:0.3F}*F'.format(internal, c_to_f(internal))
    time.sleep(1.0)
```

Finally the example enters a loop where it reads temperature measurements and prints them out every second. The important thing to see are the two temperature measurement functions, **readTempC()** and **readInternalC()**.

The **readTempC()** function will read the temperature at the thermocouple probe. If you aren't familiar with how thermocouples determine temperature, make sure to [read the thermocouple guide \(https://adafru.it/qhE\)](https://adafru.it/qhE).

The `readInternalC()` function will read the temperature internal to the MAX31855 chip.

That's all there is to use the MAX31855 Python library! If you run into issues or wish to contribute, feel free to followup on [the library's Github page \(https://adafru.it/e02\)](https://adafru.it/e02).