
Adafruit MAX31855 Library Documentation

Release 1.0

Radomir Dopieralski

Apr 26, 2021

Contents

1	Dependencies	3
2	Installing from PyPI	5
3	Usage Example	7
4	Contributing	9
5	Documentation	11
6	Table of Contents	13
6.1	Simple test	13
6.2	adafruit_max31855	13
6.2.1	Implementation Notes	14
7	Indices and tables	15
	Python Module Index	17
	Index	19

CircuitPython driver for the [MAX31855 Thermocouple Amplifier Breakout](#)

CHAPTER 1

Dependencies

This driver depends on:

- [Adafruit CircuitPython](#)
- [Bus Device](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).

CHAPTER 2

Installing from PyPI

On supported GNU/Linux systems like the Raspberry Pi, you can install the driver locally [from PyPI](#). To install for current user:

```
pip3 install adafruit-circuitpython-max31855
```

To install system-wide (this may be required in some cases):

```
sudo pip3 install adafruit-circuitpython-max31855
```

To install in a virtual environment in your current project:

```
mkdir project-name && cd project-name
python3 -m venv .env
source .env/bin/activate
pip3 install adafruit-circuitpython-max31855
```


CHAPTER 3

Usage Example

Of course, you must import the library to use it:

```
import adafruit_max31855
```

You also need to create an SPI interface object, and a pin object for the chip select pin. You can use any pin for the CS, but we use D5 here:

```
from digitalio import DigitalInOut
import board

spi = board.SPI()
cs = DigitalInOut(board.D5)
```

Next, just create the sensor object:

```
sensor = adafruit_max31855.MAX31855(spi, cs)
```

And you can start making measurements:

```
print(sensor.temperature)
```

The temperature is read in degrees Celsius (°C). You have to convert it to other units yourself, if you need it.

CHAPTER 4

Contributing

Contributions are welcome! Please read our [Code of Conduct](#) before contributing to help this project stay welcoming.

CHAPTER 5

Documentation

For information on building library documentation, please check out [this guide](#).

6.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/max31855_simpletest.py

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  import time
5  import board
6  import digitalio
7  import adafruit_max31855
8
9  spi = board.SPI()
10 cs = digitalio.DigitalInOut(board.D5)
11
12 max31855 = adafruit_max31855.MAX31855(spi, cs)
13 while True:
14     tempC = max31855.temperature
15     tempF = tempC * 9 / 5 + 32
16     print("Temperature: {} C {} F ".format(tempC, tempF))
17     time.sleep(2.0)
```

6.2 adafruit_max31855

This is a CircuitPython driver for the Maxim Integrated MAX31855 thermocouple amplifier module.

- Author(s): Radomir Dopieralski

6.2.1 Implementation Notes

Hardware:

- Adafruit [MAX31855 Thermocouple Amplifier Breakout](#) (Product ID: 269)

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://circuitpython.org/downloads>
- Adafruit's Bus Device library: https://github.com/adafruit/Adafruit_CircuitPython_BusDevice

class `adafruit_max31855.MAX31855` (*spi*, *cs*)
Driver for the MAX31855 thermocouple amplifier.

Parameters

- **spi** (*SPI*) – The SPI bus the MAX31856 is connected to.
- **cs** (*Pin*) – The pin used for the CS signal.

Quickstart: Importing and using the MAX31855

Here is an example of using the `MAX31855` class. First you will need to import the libraries to use the sensor

```
import board
from digitalio import DigitalInOut, Direction
import adafruit_max31855
```

Once this is done you can define your `board.SPI` object and define your sensor object

```
spi = board.SPI()
cs = digitalio.DigitalInOut(board.D5) # Chip select of the MAX31855,
↪board.
sensor = adafruit_max31855.MAX31855(spi, cs)
```

Now you have access to the `temperature` attribute

```
temperature = sensor.temperature
```

reference_temperature

Internal reference temperature in degrees Celsius.

temperature

Thermocouple temperature in degrees Celsius.

temperature_NIST

Thermocouple temperature in degrees Celsius, computed using raw voltages and NIST approximation for Type K, see: https://srdata.nist.gov/its90/download/type_k.tab

CHAPTER 7

Indices and tables

- `genindex`
- `modindex`
- `search`

a

adafruit_max31855, [13](#)

A

`adafruit_max31855` (*module*), [13](#)

M

`MAX31855` (*class in adafruit_max31855*), [14](#)

R

`reference_temperature`
(*adafruit_max31855.MAX31855 attribute*),
[14](#)

T

`temperature` (*adafruit_max31855.MAX31855 attribute*), [14](#)

`temperature_NIST` (*adafruit_max31855.MAX31855 attribute*), [14](#)