

---

# **AdafruitTMP006 Library Documentation**

*Release 1.0*

**Carter Nelson**

**Aug 25, 2020**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
1.1	Installing from PyPI . . . . .	3
<b>2</b>	<b>Usage Example</b>	<b>5</b>
<b>3</b>	<b>Contributing</b>	<b>7</b>
<b>4</b>	<b>Documentation</b>	<b>9</b>
<b>5</b>	<b>Table of Contents</b>	<b>11</b>
5.1	Simple test . . . . .	11
5.2	adafruit_tmp006 . . . . .	12
5.2.1	Implementation Notes . . . . .	12
<b>6</b>	<b>Indices and tables</b>	<b>13</b>
	<b>Python Module Index</b>	<b>15</b>
	<b>Index</b>	<b>17</b>



CircuitPython driver for the TMP006 contactless IR thermometer.



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](#).

## 1.1 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-tmp006
```

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

```
sudo pip3 install adafruit-circuitpython-tmp006
```

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-tmp006
```





## CHAPTER 2

---

### Usage Example

---

Ensure your device works with the simple test in the examples folder.



## CHAPTER 3

---

### Contributing

---

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



## CHAPTER 4

---

### Documentation

---

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



## 5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/tmp006\_simpletest.py

```
1 import time
2 import board
3 import busio
4 import adafruit_tmp006
5
6 # Define a function to convert celsius to fahrenheit.
7 def c_to_f(c):
8     return c * 9.0 / 5.0 + 32.0
9
10
11 # Create library object using our Bus I2C port
12 i2c = busio.I2C(board.SCL, board.SDA)
13 sensor = adafruit_tmp006.TMP006(i2c)
14
15 # Initialize communication with the sensor, using the default 16 samples per_
16     ↪ conversion.
17 # This is the best accuracy but a little slower at reacting to changes.
18 # The first sample will be meaningless
19 while True:
20     obj_temp = sensor.temperature
21     print(
22         "Object temperature: {0:0.3F}*C / {1:0.3F}*F".format(obj_temp, c_to_f(obj_
23     ↪temp))
24     )
25     time.sleep(5.0)
```

## 5.2 adafruit\_tmp006

CircuitPython driver for the TMP006 contactless IR thermometer.

- Author(s): Carter Nelson

### 5.2.1 Implementation Notes

#### Hardware:

- TMP006 Contact-less Infrared Thermopile Sensor

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's Bus Device library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_BusDevice](https://github.com/adafruit/Adafruit_CircuitPython_BusDevice)

**class** `adafruit_tmp006.TMP006` (*i2c, address=64, samplerate=2048*)

Class to represent an Adafruit TMP006 non-contact temperature measurement board.

#### **active**

True if sensor is active.

#### **read\_register** (*register*)

Read sensor Register.

#### **temperature**

Read object temperature from TMP006 sensor.



## CHAPTER 6

---

### Indices and tables

---

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



**a**

adafruit\_tmp006, 11



## A

`active` (*adafruit\_tmp006.TMP006 attribute*), 12  
`adafruit_tmp006` (*module*), 11

## R

`read_register()` (*adafruit\_tmp006.TMP006 method*), 12

## T

`temperature` (*adafruit\_tmp006.TMP006 attribute*), 12  
`TMP006` (*class in adafruit\_tmp006*), 12