
AdafruitBMP3XX Library Documentation

Release 1.0

Carter Nelson

Jan 20, 2021

Contents

1	Dependencies	3
1.1	Installing from PyPI	3
2	Usage Example	5
3	Contributing	7
4	Documentation	9
5	Table of Contents	11
5.1	Simple test	11
5.2	adafruit_bmp3xx	12
5.2.1	Implementation Notes	12
6	Indices and tables	13
	Python Module Index	15
	Index	17

CircuitPython driver from BMP3XX Temperature and Barometric Pressure sensor.

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](#).

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-bmp3xx
```

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

```
sudo pip3 install adafruit-circuitpython-bmp3xx
```

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-bmp3xx
```


CHAPTER 2

Usage Example

See usage examples 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/bmp3xx_simpletest.py

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  import time
5  import board
6  import busio
7  import adafruit_bmp3xx
8
9  # I2C setup
10 i2c = busio.I2C(board.SCL, board.SDA)
11 bmp = adafruit_bmp3xx.BMP3XX_I2C(i2c)
12
13 # SPI setup
14 # from digitalio import DigitalInOut, Direction
15 # spi = busio.SPI(board.SCK, board.MOSI, board.MISO)
16 # cs = DigitalInOut(board.D5)
17 # bmp = adafruit_bmp3xx.BMP3XX_SPI(spi, cs)
18
19 bmp.pressure_oversampling = 8
20 bmp.temperature_oversampling = 2
21
22 while True:
23     print(
24         "Pressure: {:6.4f} Temperature: {:5.2f}".format(bmp.pressure, bmp.
↪temperature)
25     )
26     time.sleep(1)
```

5.2 adafruit_bmp3xx

CircuitPython driver from BMP3XX Temperature and Barometric Pressure sensor.

- Author(s): Carter Nelson

5.2.1 Implementation Notes

Hardware:

- [Adafruit BMP388](#)

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

class adafruit_bmp3xx.**BMP3XX**

Base class for BMP3XX sensor.

altitude

The altitude in meters based on the currently set sea level pressure.

filter_coefficient

The IIR filter coefficient.

pressure

The pressure in hPa.

pressure_oversampling

The pressure oversampling setting.

reset ()

Perform a power on reset. All user configuration settings are overwritten with their default state.

sea_level_pressure = None

Sea level pressure in hPa.

temperature

The temperature in deg C.

temperature_oversampling

The temperature oversampling setting.

class adafruit_bmp3xx.**BMP3XX_I2C** (*i2c*, *address=119*)

Driver for I2C connected BMP3XX. Default address is 0x77 but another address can be passed in as an argument

class adafruit_bmp3xx.**BMP3XX_SPI** (*spi*, *cs*)

Driver for SPI connected BMP3XX.

CHAPTER 6

Indices and tables

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

a

adafruit_bmp3xx, [11](#)

A

`adafruit_bmp3xx` (*module*), [11](#)
`altitude` (*adafruit_bmp3xx.BMP3XX attribute*), [12](#)

B

`BMP3XX` (*class in adafruit_bmp3xx*), [12](#)
`BMP3XX_I2C` (*class in adafruit_bmp3xx*), [12](#)
`BMP3XX_SPI` (*class in adafruit_bmp3xx*), [12](#)

F

`filter_coefficient` (*adafruit_bmp3xx.BMP3XX attribute*), [12](#)

P

`pressure` (*adafruit_bmp3xx.BMP3XX attribute*), [12](#)
`pressure_oversampling`
(*adafruit_bmp3xx.BMP3XX attribute*), [12](#)

R

`reset()` (*adafruit_bmp3xx.BMP3XX method*), [12](#)

S

`sea_level_pressure` (*adafruit_bmp3xx.BMP3XX attribute*), [12](#)

T

`temperature` (*adafruit_bmp3xx.BMP3XX attribute*), [12](#)
`temperature_oversampling`
(*adafruit_bmp3xx.BMP3XX attribute*), [12](#)