

---

# **AdafruitMMA8451 Library Documentation**

***Release 1.0***

**Tony DiCola**

**Jul 09, 2020**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
<b>2</b>	<b>Installing from PyPI</b>	<b>5</b>
<b>3</b>	<b>Usage Example</b>	<b>7</b>
<b>4</b>	<b>Contributing</b>	<b>9</b>
<b>5</b>	<b>Documentation</b>	<b>11</b>
<b>6</b>	<b>Table of Contents</b>	<b>13</b>
6.1	Simple test . . . . .	13
6.2	adafruit_mma8451 . . . . .	14
<b>7</b>	<b>Indices and tables</b>	<b>17</b>
	<b>Python Module Index</b>	<b>19</b>
	<b>Index</b>	<b>21</b>



CircuitPython module for the MMA8451 3 axis accelerometer.



# 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-mma8451
```

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

```
sudo pip3 install adafruit-circuitpython-mma8451
```

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-mma8451
```



## CHAPTER 3

---

### Usage Example

---

See `examples/simpletest.py` for a demo of the usage.



## 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/mma8451\_simpletest.py

```
1  # Simple demo of reading the MMA8451 orientation every second.
2  # Author: Tony DiCola
3  import time
4
5  import board
6  import busio
7
8  import adafruit_mma8451
9
10
11  # Initialize I2C bus.
12  i2c = busio.I2C(board.SCL, board.SDA)
13
14  # Initialize MMA8451 module.
15  sensor = adafruit_mma8451.MMA8451(i2c)
16  # Optionally change the address if it's not the default:
17  # sensor = adafruit_mma8451.MMA8451(i2c, address=0x1C)
18
19  # Optionally change the range from its default of +/-4G:
20  # sensor.range = adafruit_mma8451.RANGE_2G # +/- 2G
21  # sensor.range = adafruit_mma8451.RANGE_4G # +/- 4G (default)
22  # sensor.range = adafruit_mma8451.RANGE_8G # +/- 8G
23
24  # Optionally change the data rate from its default of 800hz:
25  # sensor.data_rate = adafruit_mma8451.DATARATE_800HZ # 800Hz (default)
26  # sensor.data_rate = adafruit_mma8451.DATARATE_400HZ # 400Hz
27  # sensor.data_rate = adafruit_mma8451.DATARATE_200HZ # 200Hz
```

(continues on next page)

(continued from previous page)

```

28 # sensor.data_rate = adafruit_mma8451.DATARATE_100HZ # 100Hz
29 # sensor.data_rate = adafruit_mma8451.DATARATE_50HZ # 50Hz
30 # sensor.data_rate = adafruit_mma8451.DATARATE_12_5HZ # 12.5Hz
31 # sensor.data_rate = adafruit_mma8451.DATARATE_6_25HZ # 6.25Hz
32 # sensor.data_rate = adafruit_mma8451.DATARATE_1_56HZ # 1.56Hz
33
34 # Main loop to print the acceleration and orientation every second.
35 while True:
36     x, y, z = sensor.acceleration
37     print(
38         "Acceleration: x={0:0.3f}m/s^2 y={1:0.3f}m/s^2 z={2:0.3f}m/s^2".format(x, y,
↪ z)
39     )
40     orientation = sensor.orientation
41     # Orientation is one of these values:
42     # - PL_PUF: Portrait, up, front
43     # - PL_PUB: Portrait, up, back
44     # - PL_PDF: Portrait, down, front
45     # - PL_PDB: Portrait, down, back
46     # - PL_LRF: Landscape, right, front
47     # - PL_LRB: Landscape, right, back
48     # - PL_LLF: Landscape, left, front
49     # - PL_LLB: Landscape, left, back
50     print("Orientation: ", end="")
51     if orientation == adafruit_mma8451.PL_PUF:
52         print("Portrait, up, front")
53     elif orientation == adafruit_mma8451.PL_PUB:
54         print("Portrait, up, back")
55     elif orientation == adafruit_mma8451.PL_PDF:
56         print("Portrait, down, front")
57     elif orientation == adafruit_mma8451.PL_PDB:
58         print("Portrait, down, back")
59     elif orientation == adafruit_mma8451.PL_LRF:
60         print("Landscape, right, front")
61     elif orientation == adafruit_mma8451.PL_LRB:
62         print("Landscape, right, back")
63     elif orientation == adafruit_mma8451.PL_LLF:
64         print("Landscape, left, front")
65     elif orientation == adafruit_mma8451.PL_LLB:
66         print("Landscape, left, back")
67     time.sleep(1.0)

```

## 6.2 adafruit\_mma8451

CircuitPython module for the MMA8451 3 axis accelerometer. See `examples/simpletest.py` for a demo of the usage.

- Author(s): Tony DiCola

**class** `adafruit_mma8451.MMA8451` (*i2c*, \*, *address*=29)

MMA8451 accelerometer. Create an instance by specifying: - *i2c*: The I2C bus connected to the sensor.

Optionally specify: - *address*: The I2C address of the sensor if not the default of 0x1D.

**acceleration**

Get the acceleration measured by the sensor. Will return a 3-tuple of X, Y, Z axis acceleration values in m/s<sup>2</sup>.

**data\_rate**

Get and set the data rate of the sensor. Must be a value of: - DATARATE\_800HZ: 800Hz (the default) - DATARATE\_400HZ: 400Hz - DATARATE\_200HZ: 200Hz - DATARATE\_100HZ: 100Hz - DATARATE\_50HZ: 50Hz - DATARATE\_12\_5HZ: 12.5Hz - DATARATE\_6\_25HZ: 6.25Hz - DATARATE\_1\_56HZ: 1.56Hz

**orientation**

Get the orientation of the MMA8451. Will return a value of: - PL\_PUF: Portrait, up, front - PL\_PUB: Portrait, up, back - PL\_PDF: Portrait, down, front - PL\_PDB: Portrait, down, back - PL\_LRF: Landscape, right, front - PL\_LRB: Landscape, right, back - PL\_LLF: Landscape, left, front - PL\_LLB: Landscape, left, back

**range**

Get and set the range of the sensor. Must be a value of: - RANGE\_8G: +/- 8g - RANGE\_4G: +/- 4g (the default) - RANGE\_2G: +/- 2g



## CHAPTER 7

---

### Indices and tables

---

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



### **a**

adafruit\_mma8451, [14](#)





## A

`acceleration` (*adafruit\_mma8451.MMA8451 attribute*), [14](#)

`adafruit_mma8451` (*module*), [14](#)

## D

`data_rate` (*adafruit\_mma8451.MMA8451 attribute*), [14](#)

## M

`MMA8451` (*class in adafruit\_mma8451*), [14](#)

## O

`orientation` (*adafruit\_mma8451.MMA8451 attribute*), [15](#)

## R

`range` (*adafruit\_mma8451.MMA8451 attribute*), [15](#)