

---

# Adafruit INA219 Library Documentation

*Release 1.0*

**Dean Miller**

**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_ina219 . . . . .	14
6.2.1	Implementation Notes . . . . .	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 driver for the [INA219](#) current 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](#).





## 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-ina219
```

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

```
sudo pip3 install adafruit-circuitpython-ina219
```

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-ina219
```



## CHAPTER 3

---

### Usage Example

---

see [example](#)



## 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/ina219\_simpletest.py

```

1  """Sample code and test for adafruit_ina219"""
2
3  import time
4  import board
5  from adafruit_ina219 import ADCResolution, BusVoltageRange, INA219
6
7
8  i2c_bus = board.I2C()
9
10 ina219 = INA219(i2c_bus)
11
12 print("ina219 test")
13
14 # display some of the advanced field (just to test)
15 print("Config register:")
16 print("  bus_voltage_range:    0x%1X" % ina219.bus_voltage_range)
17 print("  gain:                   0x%1X" % ina219.gain)
18 print("  bus_adc_resolution:     0x%1X" % ina219.bus_adc_resolution)
19 print("  shunt_adc_resolution:   0x%1X" % ina219.shunt_adc_resolution)
20 print("  mode:                   0x%1X" % ina219.mode)
21 print("")
22
23 # optional : change configuration to use 32 samples averaging for both bus voltage_
24 ↪and shunt voltage
25 ina219.bus_adc_resolution = ADCResolution.ADCRES_12BIT_32S
26 ina219.shunt_adc_resolution = ADCResolution.ADCRES_12BIT_32S
27 # optional : change voltage range to 16V

```

(continues on next page)

(continued from previous page)

```

27 ina219.bus_voltage_range = BusVoltageRange.RANGE_16V
28
29 # measure and display loop
30 while True:
31     bus_voltage = ina219.bus_voltage # voltage on V- (load side)
32     shunt_voltage = ina219.shunt_voltage # voltage between V+ and V- across the shunt
33     current = ina219.current # current in mA
34
35     # INA219 measure bus voltage on the load side. So PSU voltage = bus_voltage +
↪shunt_voltage
36     print("PSU Voltage:    {:6.3f} V".format(bus_voltage + shunt_voltage))
37     print("Shunt Voltage:  {:9.6f} V".format(shunt_voltage))
38     print("Load Voltage:   {:6.3f} V".format(bus_voltage))
39     print("Current:       {:9.6f} A".format(current / 1000))
40     print("")
41
42     time.sleep(2)

```

## 6.2 adafruit\_ina219

CircuitPython driver for the INA219 current sensor.

- Author(s): Dean Miller

### 6.2.1 Implementation Notes

#### Hardware:

- Adafruit INA219 High Side DC Current Sensor Breakout
- Adafruit INA219 FeatherWing

#### Software and Dependencies:

- Adafruit CircuitPython firmware (2.2.0+) for the ESP8622 and M0-based 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\_ina219.**ADCResolution**  
 Constants for bus\_adc\_resolution or shunt\_adc\_resolution

**class** adafruit\_ina219.**BusVoltageRange**  
 Constants for bus\_voltage\_range

**class** adafruit\_ina219.**Gain**  
 Constants for gain

**class** adafruit\_ina219.**INA219** (i2c\_bus, addr=64)  
 Driver for the INA219 current sensor

**bus\_voltage**  
 The bus voltage (between V- and GND) in Volts

**calibration**  
 Calibration register (cached value)

**current**

The current through the shunt resistor in milliamps.

**power**

The power through the load in Watt.

**set\_calibration\_16V\_400mA()**

Configures to INA219 to be able to measure up to 16V and 400mA of current. Counter overflow occurs at 1.6A.

---

**Note:** These calculations assume a 0.1 ohm shunt resistor is present

---

**set\_calibration\_16V\_5A()**

Configures to INA219 to be able to measure up to 16V and 5000mA of current. Counter overflow occurs at 8.0A.

---

**Note:** These calculations assume a 0.02 ohm shunt resistor is present

---

**set\_calibration\_32V\_1A()**

Configures to INA219 to be able to measure up to 32V and 1A of current. Counter overflow occurs at 1.3A.

---

**Note:** These calculations assume a 0.1 ohm shunt resistor is present

---

**set\_calibration\_32V\_2A()**

Configures to INA219 to be able to measure up to 32V and 2A of current. Counter overflow occurs at 3.2A.

..note :: These calculations assume a 0.1 shunt ohm resistor is present

**shunt\_voltage**

The shunt voltage (between V+ and V-) in Volts (so +/-0.327V)

**class adafruit\_ina219.Mode**

Constants for mode



## CHAPTER 7

---

### Indices and tables

---

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



### **a**

adafruit\_ina219, [14](#)





## A

`adafruit_ina219` (*module*), [14](#)  
`ADCResolution` (*class in adafruit\_ina219*), [14](#)

## B

`bus_voltage` (*adafruit\_ina219.INA219 attribute*), [14](#)  
`BusVoltageRange` (*class in adafruit\_ina219*), [14](#)

## C

`calibration` (*adafruit\_ina219.INA219 attribute*), [14](#)  
`current` (*adafruit\_ina219.INA219 attribute*), [14](#)

## G

`Gain` (*class in adafruit\_ina219*), [14](#)

## I

`INA219` (*class in adafruit\_ina219*), [14](#)

## M

`Mode` (*class in adafruit\_ina219*), [15](#)

## P

`power` (*adafruit\_ina219.INA219 attribute*), [15](#)

## S

`set_calibration_16V_400mA()`  
    (*adafruit\_ina219.INA219 method*), [15](#)  
`set_calibration_16V_5A()`  
    (*adafruit\_ina219.INA219 method*), [15](#)  
`set_calibration_32V_1A()`  
    (*adafruit\_ina219.INA219 method*), [15](#)  
`set_calibration_32V_2A()`  
    (*adafruit\_ina219.INA219 method*), [15](#)  
`shunt_voltage` (*adafruit\_ina219.INA219 attribute*),  
    [15](#)