
Adafruit INA219 Library Documentation

Release 1.0

Dean Miller

Dec 16, 2020

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_ina219	14
6.2.1	Implementation Notes	14
7	Indices and tables	17
	Python Module Index	19
	Index	21

CircuitPython driver for the [INA219](#) current sensor.

CHAPTER 1

Dependencies

This driver depends on:

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

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     power = ina219.power # power in watts
35
36     # INA219 measure bus voltage on the load side. So PSU voltage = bus_voltage +
    ↪ shunt_voltage
37     print("Voltage (VIN+) : {:6.3f} V".format(bus_voltage + shunt_voltage))
38     print("Voltage (VIN-) : {:6.3f} V".format(bus_voltage))
39     print("Shunt Voltage : {:8.5f} V".format(shunt_voltage))
40     print("Shunt Current : {:7.4f} A".format(current / 1000))
41     print("Power Calc. : {:8.5f} W".format(bus_voltage * (current / 1000)))
42     print("Power Register : {:6.3f} W".format(power))
43     print("")
44
45     # Check internal calculations haven't overflowed (doesn't detect ADC overflows)
46     if ina219.overflow:
47         print("Internal Math Overflow Detected!")
48         print("")
49
50     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

```
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 +/-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*), 15
`BusVoltageRange` (*class in adafruit_ina219*), 14

C

`calibration` (*adafruit_ina219.INA219 attribute*), 15
`current` (*adafruit_ina219.INA219 attribute*), 15

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