

---

# Adafruitsgp30 Library Documentation

*Release 1.0*

**Ladyada**

**Apr 12, 2019**



---

## Contents

---

|          |                                      |           |
|----------|--------------------------------------|-----------|
| <b>1</b> | <b>Installation and Dependencies</b> | <b>3</b>  |
| 1.1      | Installing from PyPI . . . . .       | 3         |
| <b>2</b> | <b>Usage Notes</b>                   | <b>5</b>  |
| 2.1      | Reading from the Sensor . . . . .    | 5         |
| <b>3</b> | <b>Contributing</b>                  | <b>7</b>  |
| <b>4</b> | <b>Building locally</b>              | <b>9</b>  |
| 4.1      | Sphinx documentation . . . . .       | 9         |
| <b>5</b> | <b>Table of Contents</b>             | <b>11</b> |
| 5.1      | Simple test . . . . .                | 11        |
| 5.2      | adafruit_sgp30 . . . . .             | 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> |



A CircuitPython driver for the Sensirion SGP30 gas sensor with eCO2 and TVOC output. This sensor uses I2C!



---

## Installation and 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-sgp30
```

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

```
sudo pip3 install adafruit-circuitpython-sgp30
```

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-sgp30
```





## CHAPTER 2

---

### Usage Notes

---

See [the guide](#) for wiring and installation instructions.

First, import the library:

```
import busio
import adafruit_sgp30
```

Next, initialize the I2C bus object:

```
import board
i2c_bus = busio.I2C(board.SCL, board.SCL, frequency=100000)
```

Since we have the I2C bus object, we can now use it to instantiate the SGP30 object:

```
sgp30 = adafruit_sgp30.Adafruit_SGP30(i2c_bus)
```

### 2.1 Reading from the Sensor

To read from the sensor:

```
eCO2, TVOC = sgp30.iaq_measure()
print("eCO2 = %d ppm \t TVOC = %d ppb" % (eCO2, TVOC))
```



## CHAPTER 3

---

### Contributing

---

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



## CHAPTER 4

---

### Building locally

---

To build this library locally you'll need to install the `circuitpython-build-tools` package.

```
python3 -m venv .env
source .env/bin/activate
pip install circuitpython-build-tools
```

Once installed, make sure you are in the virtual environment:

```
source .env/bin/activate
```

Then run the build:

```
circuitpython-build-bundles --filename_prefix adafruit-circuitpython-sgp30 --library_
↪location .
```

### 4.1 Sphinx documentation

Sphinx is used to build the documentation based on rST files and comments in the code. First, install dependencies (feel free to reuse the virtual environment from above):

```
python3 -m venv .env
source .env/bin/activate
pip install Sphinx sphinx-rtd-theme
```

Now, once you have the virtual environment activated:

```
cd docs
sphinx-build -E -W -b html . _build/html
```

This will output the documentation to `docs/_build/html`. Open the `index.html` in your browser to view them. It will also (due to `-W`) error out on any warning like Travis will. This is a good way to locally verify it will pass.



## 5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/sgp30\_simpletest.py

```
1  """ Example for using the SGP30 with CircuitPython and the Adafruit library """
2
3  import time
4  import board
5  import busio
6  import adafruit_sgp30
7
8  i2c = busio.I2C(board.SCL, board.SDA, frequency=100000)
9
10 # Create library object on our I2C port
11 sgp30 = adafruit_sgp30.Adafruit_SGP30(i2c)
12
13 print("SGP30 serial #", [hex(i) for i in sgp30.serial])
14
15 sgp30.iaq_init()
16 sgp30.set_iaq_baseline(0x8973, 0x8aae)
17
18 elapsed_sec = 0
19
20 while True:
21     print("eCO2 = %d ppm \t TVOC = %d ppb" % (sgp30.eCO2, sgp30.TVOC))
22     time.sleep(1)
23     elapsed_sec += 1
24     if elapsed_sec > 10:
25         elapsed_sec = 0
26         print("**** Baseline values: eCO2 = 0x%x, TVOC = 0x%x"
27               % (sgp30.baseline_eCO2, sgp30.baseline_TVOC))
```

## 5.2 adafruit\_sgp30

I2C driver for SGP30 Sensirion VoC sensor

- Author(s): ladyada

### 5.2.1 Implementation Notes

#### Hardware:

- Adafruit [SGP30 Air Quality Sensor Breakout - VOC and eCO2](#) (Product ID: 3709)

#### Software and Dependencies:

- Adafruit CircuitPython firmware 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\_sgp30.**Adafruit\_SGP30** (*i2c*, *address*=88)

A driver for the SGP30 gas sensor.

#### **TVOC**

Total Volatile Organic Compound in parts per billion.

#### **baseline\_TVOC**

Total Volatile Organic Compound baseline value

#### **baseline\_eCO2**

Carbon Dioxide Equivalent baseline value

#### **eCO2**

Carbon Dioxide Equivalent in parts per million

#### **get\_iaq\_baseline** ()

Retrieve the IAQ algorithm baseline for eCO2 and TVOC

#### **iaq\_init** ()

Initialize the IAQ algorithm

#### **iaq\_measure** ()

Measure the eCO2 and TVOC

#### **set\_iaq\_baseline** (*eCO2*, *TVOC*)

Set the previously recorded IAQ algorithm baseline for eCO2 and TVOC

#### **set\_iaq\_humidity** (*gramsPM3*)

Set the humidity in g/m3 for eCO2 and TVOC compensation algorithm



## CHAPTER 6

---

### Indices and tables

---

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



### **a**

adafruit\_sgp30, [11](#)



## A

Adafruit\_SGP30 (*class in adafruit\_sgp30*), [12](#)  
adafruit\_sgp30 (*module*), [11](#)

## B

baseline\_eCO2 (*adafruit\_sgp30.Adafruit\_SGP30 attribute*), [12](#)  
baseline\_TVOC (*adafruit\_sgp30.Adafruit\_SGP30 attribute*), [12](#)

## E

eCO2 (*adafruit\_sgp30.Adafruit\_SGP30 attribute*), [12](#)

## G

get\_iaq\_baseline()  
(*adafruit\_sgp30.Adafruit\_SGP30 method*),  
[12](#)

## I

iaq\_init() (*adafruit\_sgp30.Adafruit\_SGP30 method*), [12](#)  
iaq\_measure() (*adafruit\_sgp30.Adafruit\_SGP30 method*), [12](#)

## S

set\_iaq\_baseline()  
(*adafruit\_sgp30.Adafruit\_SGP30 method*),  
[12](#)  
set\_iaq\_humidity()  
(*adafruit\_sgp30.Adafruit\_SGP30 method*),  
[12](#)

## T

TVOC (*adafruit\_sgp30.Adafruit\_SGP30 attribute*), [12](#)