Adafruit's CCS811 Library Documentation

Release 1.0

Dean Miller, Scott Shawcroft

Aug 23, 2019

Contents

CircuitPython driver for the CCS811 air quality sensor.

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.

Usage Notes

See the guide for wiring and installation instructions.

Of course, you must import the library to use it:

```python
import busio
import adafruit_ccs811
```

Next, initialize the I2C bus object.

```python
from board import *
i2c_bus = busio.I2C(SCL, SDA)
```

Once you have created the I2C interface object, you can use it to instantiate the CCS811 object

```python
ccs =  adafruit_ccs811.CCS811(i2c_bus)
```

2.1 Reading Sensor

To read the gas sensor simply read the attributes:

```python
print("CO2: ", ccs.eco2, " TVOC:", ccs.tvoc)
```

Contributing

Contributions are welcome! Please read our Code of Conduct before contributing to help this project stay welcoming.

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-ccs811 --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.

Table of Contents

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/ccs811_simpletest.py

```python
import time
import board
import busio
import adafruit_ccs811

i2c = busio.I2C(board.SCL, board.SDA)
ccs811 = adafruit_ccs811.CCS811(i2c)

# Wait for the sensor to be ready
while not ccs811.data_ready:
    pass

while True:
    print("CO2: {} PPM, TVOC: {} PPB"
            .format(ccs811.eco2, ccs811.tvoc))
    time.sleep(0.5)
```

5.2 `CCS811` - Adafruit CCS811 Air Quality Sensor Breakout - VOC and eCO2

This library supports the use of the CCS811 air quality sensor in CircuitPython.

Author(s): Dean Miller for Adafruit Industries

Notes:

#. Datasheet

class adafruit_ccs811.**CCS811**(*i2c_bus*, *address=90*)
     CCS811 gas sensor driver.

          Parameters

                   • **i2c** (*I2C*) – The I2C bus.

                   • **addr** (*int*) – The I2C address of the CCS811.

     data_ready
          True when new data has been read.

     eco2
          Equivalent Carbon Dioxide in parts per million. Clipped to 400 to 8192ppm.

     error
          True when an error has occured.

     error_code
          Error code

     reset()
          Initiate a software reset.

     set_environmental_data(*humidity*, *temperature*)
          Set the temperature and humidity used when computing eCO2 and TVOC values.

          Parameters

                   • **humidity** (*int*) – The current relative humidity in percent.

                   • **temperature** (*float*) – The current temperature in Celsius.

     set_interrupt_thresholds(*low_med*, *med_high*, *hysteresis*)
          Set the thresholds used for triggering the interrupt based on eCO2. The interrupt is triggered when the
          value crossed a boundary value by the minimum hysteresis value.

          Parameters

                   • **low_med** (*int*) – Boundary between low and medium ranges

                   • **med_high** (*int*) – Boundary between medium and high ranges

                   • **hysteresis** (*int*) – Minimum difference between reads

     temp_offset = 0.0
          Temperature offset.

     temperature
          Deprecated since version 1.1.5: Hardware support removed by vendor

          Temperature based on optional thermistor in Celsius.

     tvoc
          Total Volatile Organic Compound in parts per billion.

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

a

A

C

D

E

R

S

T