Adafruit SI7021 Library Documentation

Release 1.0

Radomir Dopieralski

Aug 05, 2019

Contents

CircuitPython module for use with the SI7021 Temperature and Humidity 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

Usage Notes

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

```
import adafruit_si7021
```

This driver takes an instantiated and active I2C object (from the `busio` or the `bitbangio` library) as an argument to its constructor. The way to create an I2C object depends on the board you are using. For boards with labeled SCL and SDA pins, you can:

```
from busio import I2C
from board import SCL, SDA

i2c = I2C(SCL, SDA)
```

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

```
sensor = adafruit_si7021.SI7021(i2c)
```

And then you can start measuring the temperature and humidity:

```
print(sensor.temperature)
print(sensor.relative_humidity)
```

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-si7021 --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.

CHAPTER 5

Table of Contents

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/si7021_simpletest.py

```python
import time
import board
import busio
import adafruit_si7021

# Create library object using our Bus I2C port
i2c = busio.I2C(board.SCL, board.SDA)
sensor = adafruit_si7021.SI7021(i2c)


while True:
    print("\nTemperature: %0.1f C" % sensor.temperature)
    print("Humidity: %0.1f %%" % sensor.relative_humidity)
    time.sleep(2)
```

5.2 adafruit_si7021

This is a CircuitPython driver for the SI7021 temperature and humidity sensor.

- Author(s): Radomir Dopieralski

5.2.1 Implementation Notes

Hardware:

- Adafruit Si7021 Temperature & Humidity Sensor Breakout Board (Product ID: 3251)

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

class `adafruit_si7021.`**`SI7021`**(*i2c_bus*, *address=64*)

A driver for the SI7021 temperature and humidity sensor.

Parameters

- **`i2c_bus`** – The `busio.I2C` object to use. This is the only required parameter.
- **`address`** (*`int`*) – (optional) The I2C address of the device.

`relative_humidity`

The measured relative humidity in percent.

`start_measurement`(*what*)

Starts a measurement.

Starts a measurement of either `HUMIDITY` or `TEMPERATURE` depending on the `what` argument. Returns immediately, and the result of the measurement can be retrieved with the `temperature` and `relative_humidity` properties. This way it will take much less time.

This can be useful if you want to start the measurement, but don't want the call to block until the measurement is ready – for instance, when you are doing other things at the same time.

`temperature`

The measured temperature in degrees Celcius.

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

Index