
Adafruit SEESAW Library Documentation

Release 1.0

Dean Miller

May 29, 2018

Contents

1	Dependencies	3
2	Usage Example	5
3	Contributing	7
4	Building locally	9
4.1	Sphinx documentation	9
5	Table of Contents	11
5.1	Simple test	11
5.2	seesaw	12
5.2.1	Implementation Notes	12
6	Indices and tables	13
	Python Module Index	15

CircuitPython module for use with the Adafruit ATSAMD09 seesaw.

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 Example

See `examples/seesaw_simpletest.py` for usage example.

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-seesaw --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/seesaw_simpletest.py

```
1 # Simple seesaw test using an LED attached to Pin 15.
2 #
3 # See the seesaw Learn Guide for wiring details:
4 # https://learn.adafruit.com/adafruit-seesaw-atsamd09-breakout?view=all#circuitpython-wiring-and-test
5 import time
6
7 from board import SCL, SDA
8 import busio
9 from adafruit_seesaw.seesaw import Seesaw
10
11 i2c_bus = busio.I2C(SCL, SDA)
12
13 ss = Seesaw(i2c_bus)
14
15 ss.pin_mode(15, ss.OUTPUT)
16
17 while True:
18     ss.digital_write(15, True) # turn the LED on (True is the voltage level)
19     time.sleep(1)             # wait for a second
20     ss.digital_write(15, False) # turn the LED off by making the voltage LOW
21     time.sleep(1)
```

5.2 seesaw

An I2C to whatever helper chip.

- Author(s): Dean Miller

5.2.1 Implementation Notes

Hardware:

- Adafruit [ATSAMD09 Breakout with seesaw](#) (Product ID: 3657)

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_seesaw.seesaw.Seesaw` (*i2c_bus*, *addr=73*, *drdy=None*)
Driver for Seesaw i2c generic conversion trip

Parameters

- **i2c_bus** (*I2C*) – Bus the SeeSaw is connected to
- **addr** (*int*) – I2C address of the SeeSaw device

sw_reset ()
Trigger a software reset of the SeeSaw chip

CHAPTER 6

Indices and tables

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

a

`adafruit_seesaw.seesaw`, 11

A

`adafruit_seesaw.seesaw` (module), 11

S

`Seesaw` (class in `adafruit_seesaw.seesaw`), 12

`sw_reset()` (`adafruit_seesaw.seesaw.Seesaw` method), 12