

---

# **AdafruitCAP1188 Library Documentation**

***Release 1.0***

**Carter Nelson**

**May 13, 2021**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
<b>2</b>	<b>Installing from PyPI</b>	<b>5</b>
<b>3</b>	<b>Usage Example</b>	<b>7</b>
<b>4</b>	<b>Contributing</b>	<b>9</b>
<b>5</b>	<b>Documentation</b>	<b>11</b>
<b>6</b>	<b>Table of Contents</b>	<b>13</b>
6.1	Simple test . . . . .	13
6.2	Advance test . . . . .	13
6.3	adafruit_cap1188.cap1188 . . . . .	14
6.3.1	Implementation Notes . . . . .	14
6.4	adafruit_cap1188.i2c . . . . .	16
6.4.1	Implementation Notes . . . . .	16
6.5	adafruit_cap1188.spi . . . . .	16
6.5.1	Implementation Notes . . . . .	16
<b>7</b>	<b>Indices and tables</b>	<b>17</b>
	<b>Python Module Index</b>	<b>19</b>
	<b>Index</b>	<b>21</b>



CircuitPython driver for the CAP1188 8-Key Capacitive Touch Sensor Breakout.



# 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

---

### 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-cap1188
```

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

```
sudo pip3 install adafruit-circuitpython-cap1188
```

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-cap1188
```



## CHAPTER 3

---

### Usage Example

---

See usage examples in the examples folder.



## 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/cap1188\_simpletest.py

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  import board
5  from adafruit_cap1188.i2c import CAP1188_I2C
6
7  i2c = board.I2C() # uses board.SCL and board.SDA
8  cap = CAP1188_I2C(i2c)
9
10 # SPI setup
11 # from digitalio import DigitalInOut, Direction
12 # from adafruit_cap1188.spi import CAP1188_SPI
13 # spi = board.SPI()
14 # cs = DigitalInOut(board.D5)
15 # cap = CAP1188_SPI(spi, cs)
16
17 while True:
18     for i in range(1, 9):
19         if cap[i].value:
20             print("Pin {} touched!".format(i))
```

### 6.2 Advance test

This example show the new feature included in the library allowing the possiblite the configure the averaging, cycle and sample. For reference please see Sensor Datasheet.

Listing 2: examples/cap118\_advancedtest.py

```
1  # SPDX-FileCopyrightText: 2021 Jose David M.
2  # SPDX-License-Identifier: MIT
3
4  # To use in the REPL >>> import cap118_advancedtest
5
6  import board
7  from adafruit_cap1188.i2c import CAP1188_I2C
8
9  i2c = board.I2C()
10 cap = CAP1188_I2C(i2c)
11
12 print(f"Sensor Initial Configuration Values: {cap.averaging, cap.sample, cap.cycle}")
13
14 averages = (1, 2, 4, 8, 16, 32, 64, 128)
15 samples = ("320us", "640us", "1.28ms", "2.56ms")
16 cycles = ("35ms", "70ms", "105ms", "140ms")
17
18 print("Setting Up Averages")
19 for i in averages:
20     cap.averaging = i
21     print(f"Average: {cap.averaging}")
22
23 print("Setting Up Samples")
24 for i in samples:
25     cap.sample = i
26     print(f"Sample: {cap.sample}")
27
28 print("Setting Up Cycles")
29 for i in cycles:
30     cap.cycle = i
31     print(f"Cycle: {cap.cycle}")
32
33 print("Done!")
```

## 6.3 adafruit\_cap1188.cap1188

CircuitPython driver for the CAP1188 8-Key Capacitive Touch Sensor Breakout.

- Author(s): Carter Nelson, Jeremiah Rose, Jose David M.

### 6.3.1 Implementation Notes

#### Hardware:

- CAP1188 - 8-Key Capacitive Touch Sensor Breakout (Product ID: 1602)

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://circuitpython.org/downloads>
- Adafruit's Bus Device library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_BusDevice](https://github.com/adafruit/Adafruit_CircuitPython_BusDevice)

**class** adafruit\_cap1188.cap1188.CAP1188  
CAP1188 driver base, must be extended for I2C/SPI interfacing.

**averaging**

Samples that are taken for all active channels during the sensor cycle. All samples are taken consecutively on the same channel before the next channel is sampled and the result is averaged over the number of samples measured before updating the measured results

if CS1, CS2, and CS3 are sampled during the sensor cycle, and the AVG[2:0] bits are set to take 4 samples per channel, then the full sensor cycle will be: CS1, CS1, CS1, CS1, CS2, CS2, CS2, CS2, CS3, CS3, CS3, CS3.

**cycle**

The programmed cycle time is only maintained if the total averaging time for all samples is less than the programmed cycle. The AVG[2:0] bits will take priority so that if more samples are required than would normally be allowed during the cycle time, the cycle time will be extended as necessary to accommodate the number of samples to be measured.

**delta\_count** (*pin*)

Return the 8 bit delta count value for the channel.

**recalibrate** ()

Perform a self recalibration on all the pins.

**recalibrate\_pins** (*mask*)

Recalibrate pins specified by bit mask.

**sample**

Determines the overall cycle time for all measured channels during normal operation. All measured channels are sampled at the beginning of the cycle time. If additional time is remaining, then the device is placed into a lower power state for the remaining duration of the cycle.

**sensitivity**

The sensitivity of touch detections. Range is 1 (least) to 128 (most).

**threshold\_values** ()

Return tuple of touch threshold values for all channels.

**thresholds**

Touch threshold value for all channels.

**touched** ()

Return 8 bit value representing touch state of all pins.

**touched\_pins**

A tuple of touched state for all pins.

**class** adafruit\_cap1188.cap1188.CAP1188\_Channel (*cap1188, pin*)

Helper class to represent a touch channel on the CAP1188. Not meant to be used directly.

**raw\_value**

The raw touch measurement.

**recalibrate** ()

Perform a self recalibration.

**threshold**

The touch threshold value.

**value**

Whether the pin is being touched or not.

## 6.4 adafruit\_cap1188.i2c

CircuitPython I2C driver for the CAP1188 8-Key Capacitive Touch Sensor Breakout.

- Author(s): Carter Nelson

### 6.4.1 Implementation Notes

#### Hardware:

- [CAP1188 - 8-Key Capacitive Touch Sensor Breakout](#) (Product ID: 1602)

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://circuitpython.org/downloads>
- Adafruit's Bus Device library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_BusDevice](https://github.com/adafruit/Adafruit_CircuitPython_BusDevice)

**class** adafruit\_cap1188.i2c.**CAP1188\_I2C** (*i2c, address=41*)  
Driver for the CAP1188 connected over I2C.

## 6.5 adafruit\_cap1188.spi

CircuitPython SPI driver for the CAP1188 8-Key Capacitive Touch Sensor Breakout.

- Author(s): Carter Nelson

### 6.5.1 Implementation Notes

#### Hardware:

- [CAP1188 - 8-Key Capacitive Touch Sensor Breakout](#) (Product ID: 1602)

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://circuitpython.org/downloads>
- Adafruit's Bus Device library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_BusDevice](https://github.com/adafruit/Adafruit_CircuitPython_BusDevice)

**class** adafruit\_cap1188.spi.**CAP1188\_SPI** (*spi, cs*)  
Driver for the CAP1188 connected over SPI.

## CHAPTER 7

---

### Indices and tables

---

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



### a

`adafruit_cap1188.cap1188`, [14](#)  
`adafruit_cap1188.i2c`, [15](#)  
`adafruit_cap1188.spi`, [16](#)





## A

`adafruit_cap1188.cap1188` (module), 14  
`adafruit_cap1188.i2c` (module), 15  
`adafruit_cap1188.spi` (module), 16  
`averaging` (`adafruit_cap1188.cap1188.CAP1188` attribute), 14

## C

`CAP1188` (class in `adafruit_cap1188.cap1188`), 14  
`CAP1188_Channel` (class in `adafruit_cap1188.cap1188`), 15  
`CAP1188_I2C` (class in `adafruit_cap1188.i2c`), 16  
`CAP1188_SPI` (class in `adafruit_cap1188.spi`), 16  
`cycle` (`adafruit_cap1188.cap1188.CAP1188` attribute), 15

## D

`delta_count()` (`adafruit_cap1188.cap1188.CAP1188` method), 15

## R

`raw_value` (`adafruit_cap1188.cap1188.CAP1188_Channel` attribute), 15  
`recalibrate()` (`adafruit_cap1188.cap1188.CAP1188` method), 15  
`recalibrate()` (`adafruit_cap1188.cap1188.CAP1188_Channel` method), 15  
`recalibrate_pins()` (`adafruit_cap1188.cap1188.CAP1188` method), 15

## S

`sample` (`adafruit_cap1188.cap1188.CAP1188` attribute), 15  
`sensitivity` (`adafruit_cap1188.cap1188.CAP1188` attribute), 15

## T

`threshold` (`adafruit_cap1188.cap1188.CAP1188_Channel` attribute), 15

`threshold_values()` (`adafruit_cap1188.cap1188.CAP1188` method), 15  
`thresholds` (`adafruit_cap1188.cap1188.CAP1188` attribute), 15  
`touched()` (`adafruit_cap1188.cap1188.CAP1188` method), 15  
`touched_pins` (`adafruit_cap1188.cap1188.CAP1188` attribute), 15

## V

`value` (`adafruit_cap1188.cap1188.CAP1188_Channel` attribute), 15