

---

# **AdafruitDS3502 Library Documentation**

*Release 1.0*

**Bryan Siepert**

**Mar 16, 2020**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
1.1	Installing from PyPI . . . . .	3
<b>2</b>	<b>Usage Example</b>	<b>5</b>
<b>3</b>	<b>Contributing</b>	<b>7</b>
<b>4</b>	<b>Documentation</b>	<b>9</b>
<b>5</b>	<b>Table of Contents</b>	<b>11</b>
5.1	Simple test . . . . .	11
5.2	adafruit_ds3502 . . . . .	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>
	<b>Index</b>	<b>17</b>



CircuitPython library for the Maxim DS3502 I2C Potentionmeter



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.

## 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-ds3502
```

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

```
sudo pip3 install adafruit-circuitpython-ds3502
```

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-ds3502
```





## CHAPTER 2

---

### Usage Example

---

```
from time import sleep
import board
import adafruit_ds3502
from analogio import AnalogIn

##### NOTE #####
# this example will not work with Blinka/raspberry Pi due to the lack of analog pins.
# Blinka and Raspberry Pi users should run the "ds3502_blinka_simpletest.py" example

i2c = board.I2C()
ds3502 = adafruit_ds3502.DS3502(i2c)
wiper_output = AnalogIn(board.A0)

while True:

    ds3502.wiper = 127
    print("Wiper set to %d"%ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f"%voltage)
    print("")
    sleep(1.0)

    ds3502.wiper = 0
    print("Wiper set to %d"%ds3502.wiper)
    voltage = wiper_output.value
    voltage *= 3.3
    voltage /= 65535
    print("Wiper voltage: %.2f"%voltage)
    print("")
    sleep(1.0)

    ds3502.wiper = 63
```

(continues on next page)

(continued from previous page)

```
print("Wiper set to %d"%ds3502.wiper)
voltage = wiper_output.value
voltage *= 3.3
voltage /= 65535
print("Wiper voltage: %.2f"%voltage)
print("")
sleep(1.0)
```

## CHAPTER 3

---

### Contributing

---

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



## CHAPTER 4

---

### Documentation

---

For information on building library documentation, please check out [this guide](#).



## 5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/ds3502\_simpletest.py

```
1 from time import sleep
2 import board
3 from analogio import AnalogIn
4 import adafruit_ds3502
5
6 ##### NOTE #####
7 # this example will not work with Blinka/raspberry Pi due to the lack of analog pins.
8 # Blinka and Raspberry Pi users should run the "ds3502_blinka_simpletest.py" example
9
10 i2c = board.I2C()
11 ds3502 = adafruit_ds3502.DS3502(i2c)
12 wiper_output = AnalogIn(board.A0)
13
14 while True:
15
16     ds3502.wiper = 127
17     print("Wiper set to %d" % ds3502.wiper)
18     voltage = wiper_output.value
19     voltage *= 3.3
20     voltage /= 65535
21     print("Wiper voltage: %.2f V" % voltage)
22     print("")
23     sleep(1.0)
24
25     ds3502.wiper = 0
26     print("Wiper set to %d" % ds3502.wiper)
27     voltage = wiper_output.value
```

(continues on next page)

(continued from previous page)

```

28 voltage *= 3.3
29 voltage /= 65535
30 print("Wiper voltage: %.2f V" % voltage)
31 print("")
32 sleep(1.0)
33
34 ds3502.wiper = 63
35 print("Wiper set to %d" % ds3502.wiper)
36 voltage = wiper_output.value
37 voltage *= 3.3
38 voltage /= 65535
39 print("Wiper voltage: %.2f V" % voltage)
40 print("")
41 sleep(1.0)

```

## 5.2 adafruit\_ds3502

CircuitPython library for the Maxim DS3502 I2C Digital Potentionmeter

- Author(s): Bryan Siepert

### 5.2.1 Implementation Notes

#### Hardware:

- Adafruit DS3502

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported 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)
- Adafruit's Register library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_Register](https://github.com/adafruit/Adafruit_CircuitPython_Register)

**class** `adafruit_ds3502.DS3502` (*i2c\_bus*, *address=40*)  
 Driver for the DS3502 I2C Digital Potentiometer.

#### Parameters

- **i2c\_bus** (*I2C*) – The I2C bus the DS3502 is connected to.
- **address** – The I2C device address for the sensor. Default is 0x40.

#### **set\_default** (*default*)

Sets the wiper's default value and current value to the given value

**Parameters** **new\_default** – The value from 0-127 to set as the wiper's default.

#### **wiper**

The value of the potentiometer's wiper.

**Parameters** **wiper\_value** – The value from 0-127 to set the wiper to.



## CHAPTER 6

---

### Indices and tables

---

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



**a**

`adafruit_ds3502`, 12



## A

`adafruit_ds3502` (*module*), 12

## D

`DS3502` (*class in adafruit\_ds3502*), 12

## S

`set_default()` (*adafruit\_ds3502.DS3502 method*),  
12

## W

`wiper` (*adafruit\_ds3502.DS3502 attribute*), 12