

---

# **AdafruitWS2801 Library Documentation**

*Release 1.0*

**Damien P. George, Limor Fried  
Scott Shawcroft, Kevin J Walters**

**Nov 30, 2020**



---

## 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	adafruit_ws2801 - WS2801 LED pixel string driver . . . . .	14
<b>7</b>	<b>Indices and tables</b>	<b>15</b>
	<b>Python Module Index</b>	<b>17</b>
	<b>Index</b>	<b>19</b>



Higher level WS2801 driver that presents the LED string as a sequence. It is the same api as the [NeoPixel library](#).

Colors are stored as tuples by default. However, you can also use int hex syntax to set values similar to colors on the web. For example, `0x800000` (`#800000` on the web) is equivalent to `(0x80, 0, 0)`.



# 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-ws2801
```

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

```
sudo pip3 install adafruit-circuitpython-ws2801
```

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-ws2801
```



## CHAPTER 3

---

### Usage Example

---

This example demonstrates the library driving a strand of 25 RGB leds by a Gemma M0 using the hardware SPI capable outputs.

```
import board
import adafruit_ws2801

leds = adafruit_ws2801.WS2801(board.D2, board.D0, 25)
leds.fill((0x80, 0, 0))
```



## 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/ws2801\_simpletest.py

```
1  ### Based on example from
2  ### https://github.com/adafruit/Adafruit_CircuitPython_DotStar/tree/master/examples
3
4  import time
5  import random
6  import board
7  import adafruit_ws2801
8
9  ### Example for a Feather M4 driving 25 12mm leds
10 odata = board.D5
11 oclock = board.D6
12 numleds = 25
13 bright = 1.0
14 leds = adafruit_ws2801.WS2801(
15     oclock, odata, numleds, brightness=bright, auto_write=False
16 )
17
18 ##### HELPERS #####
19
20 # a random color 0 -> 224
21 def random_color():
22     return random.randrange(0, 7) * 32
23
24
25 ##### MAIN LOOP #####
26 n_leds = len(leds)
27 while True:
```

(continues on next page)

(continued from previous page)

```

28 # fill each led with a random color
29 for idx in range(n_leds):
30     leds[idx] = (random_color(), random_color(), random_color())
31
32 # show all leds in led string
33 leds.show()
34
35 time.sleep(0.25)

```

## 6.2 adafruit\_ws2801 - WS2801 LED pixel string driver

- Author(s): Damien P. George, Limor Fried & Scott Shawcroft, Kevin J Walters

**class** `adafruit_ws2801.WS2801` (*clock*, *data*, *n*, \*, *brightness=1.0*, *auto\_write=True*)  
 A sequence of WS2801 controlled LEDs.

### Parameters

- **clock** (*Pin*) – The pin to output dotstar clock on.
- **data** (*Pin*) – The pin to output dotstar data on.
- **n** (*int*) – The number of LEDs in the chain.
- **brightness** (*float*) – The brightness between 0.0 and (default) 1.0.
- **auto\_write** (*bool*) – True if the dotstars should immediately change when set. If False, *show* must be called explicitly.

Example for Gemma M0:

```

import adafruit_ws2801
import time
import board

darkred = 0x100000

with adafruit_ws2801.WS2801(board.D2, board.D0, 25, brightness=1.0) as pixels:
    pixels[0] = darkred
    time.sleep(2)

```

### **brightness**

Overall brightness of the pixel

### **deinit** ()

Blank out the DotStars and release the resources.

### **fill** (*color*)

Colors all pixels the given *\*color\**.

### **show** ()

Shows the new colors on the pixels themselves if they haven't already been autowritten.

The colors may or may not be showing after this function returns because it may be done asynchronously.

## CHAPTER 7

---

### Indices and tables

---

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



**a**

adafruit\_ws2801, 14



## A

adafruit\_ws2801 (*module*), 14

## B

brightness (*adafruit\_ws2801.WS2801 attribute*), 14

## D

deinit () (*adafruit\_ws2801.WS2801 method*), 14

## F

fill () (*adafruit\_ws2801.WS2801 method*), 14

## S

show () (*adafruit\_ws2801.WS2801 method*), 14

## W

WS2801 (*class in adafruit\_ws2801*), 14