

---

# **AdafruitSlideshow Library Documentation**

*Release 1.0*

**Kattni Rembor**

**Oct 23, 2019**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
<b>2</b>	<b>Usage Example</b>	<b>5</b>
<b>3</b>	<b>Contributing</b>	<b>7</b>
<b>4</b>	<b>Building locally</b>	<b>9</b>
4.1	Zip release files . . . . .	9
4.2	Sphinx documentation . . . . .	9
<b>5</b>	<b>Table of Contents</b>	<b>11</b>
5.1	Simple test . . . . .	11
5.2	adafruit_slideshow . . . . .	11
5.2.1	Implementation Notes . . . . .	11
<b>6</b>	<b>Indices and tables</b>	<b>15</b>
	<b>Python Module Index</b>	<b>17</b>
	<b>Index</b>	<b>19</b>



CircuitPython helper library for displaying a slideshow of images on a display.



# CHAPTER 1

---

## Dependencies

---

This driver depends on:

- [Adafruit CircuitPython](#)

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

---

```
from adafruit_slideshow import PlayBackOrder, SlideShow
import board
import pulseio

# Create the slideshow object that plays through once alphabetically.
slideshow = SlideShow(board.DISPLAY, pulseio.PWMOut(board.TFT_BACKLIGHT), folder="/",
                      loop=False, order=PlayBackOrder.ALPHABETICAL)

while slideshow.update():
    pass
```



## CHAPTER 3

---

### Contributing

---

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



### 4.1 Zip release files

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-slideshow --
↳library_location .
```

### 4.2 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/slideshow\_simpletest.py

```
1 import board
2 import pulseio
3 from adafruit_slideshow import PlayBackOrder, SlideShow
4 #pylint: disable=no-member
5
6 # Create the slideshow object that plays through once alphabetically.
7 slideshow = SlideShow(board.DISPLAY, pulseio.PWMOut(board.TFT_BACKLIGHT), folder="/",
8                     loop=False, order=PlayBackOrder.ALPHABETICAL)
9
10 while slideshow.update():
11     pass
```

## 5.2 adafruit\_slideshow

CircuitPython helper library for displaying a slideshow of images on a display.

- Author(s): Kattni Rembor, Carter Nelson, Roy Hooper

### 5.2.1 Implementation Notes

#### Hardware:

- Adafruit Hallowing M0 Express

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>

**class** `adafruit_slideshow.PlayBackDirection`

Defines possible slideshow playback directions.

**BACKWARD** = -1

The next image is before the current image. When alphabetically sorted, this is towards A.

**FORWARD** = 1

The next image is after the current image. When alphabetically sorted, this is towards Z.

**class** `adafruit_slideshow.PlayBackOrder`

Defines possible slideshow playback orders.

**ALPHABETICAL** = 0

Orders by alphabetical sort of filenames

**RANDOM** = 1

Randomly shuffles the images

**class** `adafruit_slideshow.SlideShow`(*display*, *backlight\_pwm=None*, \*, *folder='/'*, *order=0*, *loop=True*, *dwell=3*, *fade\_effect=True*, *auto\_advance=True*, *direction=1*)

Class for displaying a slideshow of .bmp images on displays.

### Parameters

- **folder** (*str*) – Specify the folder containing the image files, in quotes. Default is the root directory, `"/"`.
- **order** (`PlayBackOrder`) – The order in which the images display. You can choose random (`RANDOM`) or alphabetical (`ALPHABETICAL`). Default is `ALPHABETICAL`.
- **loop** (*bool*) – Specify whether to loop the images or play through the list once. `True` if slideshow will continue to loop, `False` if it will play only once. Default is `True`.
- **dwell** (*int*) – The number of seconds each image displays, in seconds. Default is 3.
- **fade\_effect** (*bool*) – Specify whether to include the fade effect between images. `True` tells the code to fade the backlight up and down between image display transitions. `False` maintains max brightness on the backlight between image transitions. Default is `True`.
- **auto\_advance** (*bool*) – Specify whether to automatically advance after dwell seconds. `True` if slideshow should auto play, `False` if you want to control advancement manually. Default is `True`.
- **direction** (`PlayBackDirection`) – The playback direction.

Example code for Hallowing Express. With this example, the slideshow will play through once in alphabetical order:

```
from adafruit_slideshow import PlayBackOrder, SlideShow
import board
import pulseio

slideshow = SlideShow(board.DISPLAY, pulseio.PWMOut(board.TFT_BACKLIGHT), folder=
    ↪"/",
                        loop=False, order=PlayBackOrder.ALPHABETICAL)

while slideshow.update():
    pass
```



Example code for Hallowing Express. Sets `dwll` to 0 seconds, turns `auto_advance` off, and uses capacitive touch to advance backwards and forwards through the images and to control the brightness level of the backlight:

```

from adafruit_slideshow import PlayBackOrder, SlideShow, PlayBackDirection
import touchio
import board
import pulseio

forward_button = touchio.TouchIn(board.TOUCH4)
back_button = touchio.TouchIn(board.TOUCH1)

brightness_up = touchio.TouchIn(board.TOUCH3)
brightness_down = touchio.TouchIn(board.TOUCH2)

slideshow = SlideShow(board.DISPLAY, pulseio.PWMOut(board.TFT_BACKLIGHT), folder=
↳"/",
                        auto_advance=False, dwell=0)

while True:
    if forward_button.value:
        slideshow.direction = PlayBackDirection.FORWARD
        slideshow.advance()
    if back_button.value:
        slideshow.direction = PlayBackDirection.BACKWARD
        slideshow.advance()

    if brightness_up.value:
        slideshow.brightness += 0.001
    elif brightness_down.value:
        slideshow.brightness -= 0.001

```

**advance ()**

Displays the next image. Returns True when a new image was displayed, False otherwise.

**auto\_advance = None**

Enable auto-advance based on dwell time. Set to False to manually control.

**brightness**

Brightness of the backlight when an image is displaying. Clamps to 0 to 1.0

**current\_image\_name**

Returns the current image name.

**direction = None**

Specify the playback direction. Default is `PlayBackDirection.FORWARD`. Can also be `PlayBackDirection.BACKWARD`.

**dwell = None**

The number of seconds each image displays, in seconds.

**fade\_effect = None**

Whether to include the fade effect between images. True tells the code to fade the backlight up and down between image display transitions. False maintains max brightness on the backlight between image transitions.

**loop = None**

Specifies whether to loop through the images continuously or play through the list once. True will continue to loop, False will play only once.

**order**

The order in which the images display. You can choose random (RANDOM) or alphabetical (ALPHA).

**update ()**

Updates the slideshow to the next image.

## CHAPTER 6

---

### Indices and tables

---

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



**a**

`adafruit_slideshow`, 11



## A

adafruit\_slideshow (*module*), 11  
advance () (*adafruit\_slideshow.SlideShow method*), 13  
ALPHABETICAL (*adafruit\_slideshow.PlayBackOrder attribute*), 12  
auto\_advance (*adafruit\_slideshow.SlideShow attribute*), 13

## B

BACKWARD (*adafruit\_slideshow.PlayBackDirection attribute*), 12  
brightness (*adafruit\_slideshow.SlideShow attribute*), 13

## C

current\_image\_name  
(*adafruit\_slideshow.SlideShow attribute*), 13

## D

direction (*adafruit\_slideshow.SlideShow attribute*), 13  
dwell (*adafruit\_slideshow.SlideShow attribute*), 13

## F

fade\_effect (*adafruit\_slideshow.SlideShow attribute*), 13  
FORWARD (*adafruit\_slideshow.PlayBackDirection attribute*), 12

## L

loop (*adafruit\_slideshow.SlideShow attribute*), 13

## O

order (*adafruit\_slideshow.SlideShow attribute*), 13

## P

PlayBackDirection (*class in adafruit\_slideshow*), 12

PlayBackOrder (*class in adafruit\_slideshow*), 12

## R

RANDOM (*adafruit\_slideshow.PlayBackOrder attribute*), 12

## S

SlideShow (*class in adafruit\_slideshow*), 12

## U

update () (*adafruit\_slideshow.SlideShow method*), 14