
Adafruitlfx Library Documentation

Release 1.0

Brent Rubell

Jan 14, 2020

Contents

1	Dependencies	3
1.1	Installing from PyPI	3
2	Usage Example	5
3	Contributing	7
4	Documentation	9
5	Table of Contents	11
5.1	Simple test	11
5.2	API	12
5.2.1	adafruit_lifx	12
5.2.1.1	Implementation Notes	12
6	Indices and tables	15
	Python Module Index	17
	Index	19

Control [LIFX devices](#) over the internet using CircuitPython.

This driver depends on:

- [Adafruit CircuitPython](#)

You'll also need a library to communicate with an ESP32 as a coprocessor using a `WiFiManager` object. This library supports connecting an ESP32 using either SPI or UART.

- SPI: [Adafruit CircuitPython ESP32SPI](#)
- UART: [Adafruit CircuitPython ESP_ATcontrol](#)

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-lifx
```

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

```
sudo pip3 install adafruit-circuitpython-lifx
```

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-lifx
```


CHAPTER 2

Usage Example

Initialize the LIFX API Client with a WiFiManager object and a LIFX Personal Access token:

```
lifx = adafruit_lifx.LIFX(wifi, lifx_token)
```

Set a LIFX selector label to identify the LIFX device to communicate with.

```
lifx_light = 'label:Lamp'
```

List all connected LIFX devices:

```
lights = lifx.list_lights()
```

Toggle the state of a LIFX device:

```
lifx.toggle_light(lifx_light)
```

Set the brightness of a LIFX device to 50%:

```
lifx.set_brightness(lifx_light, 0.5)
```

Set the color of a LIFX device to blue and the brightness to 100%:

```
lifx.set_color(lifx_light, 'on', 'blue', brightness=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/lifx_simpletest.py

```
1 import board
2 import busio
3 from digitalio import DigitalInOut
4 from adafruit_esp32spi import adafruit_esp32spi
5 from adafruit_esp32spi import adafruit_esp32spi_wifimanager
6 import neopixel
7
8 import adafruit_lifx
9
10 # Get wifi details and more from a secrets.py file
11 try:
12     from secrets import secrets
13 except ImportError:
14     print("WiFi and API secrets are kept in secrets.py, please add them there!")
15     raise
16
17 # ESP32 SPI
18 esp32_cs = DigitalInOut(board.ESP_CS)
19 esp32_ready = DigitalInOut(board.ESP_BUSY)
20 esp32_reset = DigitalInOut(board.ESP_RESET)
21 spi = busio.SPI(board.SCK, board.MOSI, board.MISO)
22 esp = adafruit_esp32spi.ESP_SPIcontrol(spi, esp32_cs, esp32_ready, esp32_reset)
23 status_light = neopixel.NeoPixel(board.NEOPIXEL, 1, brightness=0.2)
24 wifi = adafruit_esp32spi_wifimanager.ESP8266WiFiManager(esp, secrets, status_light)
25
26 # Add your LIFX Personal Access token to secrets.py
27 # (to obtain a token, visit: https://cloud.lifx.com/settings)
```

(continues on next page)

(continued from previous page)

```

28 lifx_token = secrets['lifx_token']
29
30 # Set this to your LIFX light separator label
31 # https://api.developer.lifx.com/docs/selectors
32 lifx_light = 'label:Lamp'
33
34 # Initialize the LIFX API Client
35 lifx = adafruit_lifx.LIFX(wifi, lifx_token)
36
37 # List all lights
38 lights = lifx.list_lights()
39
40 # Turn on the light
41 print('Turning on light...')
42 lifx.toggle_light(lifx_light)
43
44 # Set the light's brightness to 50%
45 light_brightness = 0.5
46 lifx.set_brightness(lifx_light, light_brightness)
47
48 # Cycle the light using the colors of the Python logo
49 colors = ['yellow', 'blue', 'white']
50 for color in colors:
51     print('Setting light to: ', color)
52     lifx.set_color(lifx_light, power='on', color=color, brightness=light_brightness)
53
54 # Turn off the light
55 print('Turning off light...')
56 lifx.toggle_light(lifx_light)

```

5.2 API

5.2.1 adafruit_lifx

A CircuitPython/Python library for communicating with the LIFX HTTP Remote API.

- Author(s): Brent Rubell for Adafruit Industries

5.2.1.1 Implementation Notes

Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- **Adafruit ESP32SPI or ESP_ATcontrol library:** https://github.com/adafruit/Adafruit_CircuitPython_ESP32SPI https://github.com/adafruit/Adafruit_CircuitPython_ESP_ATcontrol

class `adafruit_lifx.LIFX` (*wifi_manager*, *lifx_token*)

HTTP Interface for interacting with the LIFX API

effects_off (*selector*, *power_off=False*)

Turns off any running effects on the selected device. :param dict selector: Selector to control which lights are requested. :param bool power_off: If true, the devices will also be turned off.

list_lights ()

Enumerates all the lights associated with the LIFX Cloud Account

move_effect (*selector, move_direction, period, power_on*)

Performs a linear move effect on a light, or lights. :param str move_direction: Move direction, forward or backward. :param double period: Time in second per effect cycle. :param bool power_on: Turn on a light before performing the move.

set_brightness (*selector, brightness*)

Sets the state of the lights within the selector. :param dict selector: Selector to control which lights are requested. :param double brightness: Brightness level of the light, from 0.0 to 1.0.

set_color (*selector, **kwargs*)

Sets the state of the light's color within the selector. Valid arguments: <https://api.developer.lifx.com/docs/set-state>

toggle_light (*selector, all_lights=False, duration=0*)

Toggles current state of LIFX light(s). :param dict selector: Selector to control which lights are requested. :param bool all: Toggle all lights at once. Defaults to false. :param double duration: Time (in seconds) to spend performing a toggle. Defaults to 0.

CHAPTER 6

Indices and tables

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

a

`adafruit_lifx`, 12

A

`adafruit_lifx` (*module*), 12

E

`effects_off()` (*adafruit_lifx.LIFX method*), 12

L

`LIFX` (*class in adafruit_lifx*), 12

`list_lights()` (*adafruit_lifx.LIFX method*), 12

M

`move_effect()` (*adafruit_lifx.LIFX method*), 13

S

`set_brightness()` (*adafruit_lifx.LIFX method*), 13

`set_color()` (*adafruit_lifx.LIFX method*), 13

T

`toggle_light()` (*adafruit_lifx.LIFX method*), 13