

---

# **AdafruitMagTag Library Documentation**

*Release 1.0*

**Melissa LeBlanc-Williams**

**May 24, 2021**



# 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>Documentation</b>	<b>9</b>
<b>5</b>	<b>Table of Contents</b>	<b>11</b>
5.1	Simple test . . . . .	11
5.2	Other Demos . . . . .	12
5.3	adafruit_magtag.graphics . . . . .	13
5.3.1	Implementation Notes . . . . .	13
5.4	adafruit_magtag.magtag . . . . .	14
5.4.1	Implementation Notes . . . . .	14
5.5	adafruit_magtag.network . . . . .	15
5.5.1	Implementation Notes . . . . .	15
5.6	adafruit_magtag.peripherals . . . . .	16
5.6.1	Implementation Notes . . . . .	16
<b>6</b>	<b>Indices and tables</b>	<b>19</b>
	<b>Python Module Index</b>	<b>21</b>
	<b>Index</b>	<b>23</b>



Helper library for the Adafruit MagTag.



## DEPENDENCIES

This driver depends on:

- [Adafruit CircuitPython](#)
- [Adafruit PortalBase](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).





## USAGE EXAMPLE

```
# SPDX-FileCopyrightText: 2017 Scott Shawcroft, written for Adafruit Industries
#
# SPDX-License-Identifier: Unlicense
import time
import terminalio
from adafruit_magtag.magtag import MagTag

magtag = MagTag()

magtag.add_text(
    text_font=terminalio.FONT,
    text_position=(
        50,
        (magtag.graphics.display.height // 2) - 1,
    ),
    text_scale=3,
)

magtag.set_text("Hello World")

buttons = magtag.peripherals.buttons
button_colors = ((255, 0, 0), (255, 150, 0), (0, 255, 255), (180, 0, 255))
button_tones = (1047, 1318, 1568, 2093)
timestamp = time.monotonic()

while True:
    for i, b in enumerate(buttons):
        if not b.value:
            print("Button %c pressed" % chr((ord("A") + i)))
            magtag.peripherals.neopixel_disable = False
            magtag.peripherals.neopixels.fill(button_colors[i])
            magtag.peripherals.play_tone(button_tones[i], 0.25)
            break
    else:
        magtag.peripherals.neopixel_disable = True
    time.sleep(0.01)
```



## **CONTRIBUTING**

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



## **DOCUMENTATION**

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



## TABLE OF CONTENTS

### 5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/magtag\_simpletest.py

```
1 # SPDX-FileCopyrightText: 2017 Scott Shawcroft, written for Adafruit Industries
2 #
3 # SPDX-License-Identifier: Unlicense
4 import time
5 from adafruit_magtag.magtag import MagTag
6
7 magtag = MagTag()
8
9 magtag.add_text(
10     text_position=(
11         50,
12         (magtag.graphics.display.height // 2) - 1,
13     ),
14     text_scale=3,
15 )
16
17 magtag.set_text("Hello World")
18
19 button_colors = ((255, 0, 0), (255, 150, 0), (0, 255, 255), (180, 0, 255))
20 button_tones = (1047, 1318, 1568, 2093)
21
22 while True:
23     for i, b in enumerate(magtag.peripherals.buttons):
24         if not b.value:
25             print("Button %c pressed" % chr((ord("A") + i)))
26             magtag.peripherals.neopixel_disable = False
27             magtag.peripherals.neopixels.fill(button_colors[i])
28             magtag.peripherals.play_tone(button_tones[i], 0.25)
29             break
30         else:
31             magtag.peripherals.neopixel_disable = True
32             time.sleep(0.01)
```

## 5.2 Other Demos

Listing 2: examples/magtag\_bitcoin\_demo.py

```
1 # SPDX-FileCopyrightText: 2017 Scott Shawcroft, written for Adafruit Industries
2 #
3 # SPDX-License-Identifier: Unlicense
4 from adafruit_magtag.magtag import MagTag
5
6 # Set up where we'll be fetching data from
7 DATA_SOURCE = "https://api.coindesk.com/v1/bpi/currentprice.json"
8 DATA_LOCATION = ["bpi", "USD", "rate_float"]
9
10
11 def text_transform(val):
12     return "Bitcoin: ${d}" % val
13
14
15 magtag = MagTag(
16     url=DATA_SOURCE,
17     json_path=DATA_LOCATION,
18 )
19
20 magtag.network.connect()
21
22 magtag.add_text(
23     text_position=(
24         (magtag.graphics.display.width // 2) - 1,
25         (magtag.graphics.display.height // 2) - 1,
26     ),
27     text_scale=3,
28     text_transform=text_transform,
29     text_anchor_point=(0.5, 0.5),
30 )
31
32 try:
33     value = magtag.fetch()
34     print("Response is", value)
35 except (ValueError, RuntimeError) as e:
36     print("Some error occurred, retrying! -", e)
37 magtag.exit_and_deep_sleep(60)
```



## 5.3 adafruit\_magtag.graphics

Helper Library for the Adafruit MagTag.

- Author(s): Melissa LeBlanc-Williams

### 5.3.1 Implementation Notes

#### Hardware:

- Adafruit MagTag

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's PortalBase library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_PortalBase](https://github.com/adafruit/Adafruit_CircuitPython_PortalBase)

```
class adafruit_magtag.graphics.Graphics(*, default_bg=16777215, auto_refresh=True, rotation=270,
                                         debug=False)
```

Graphics Helper Class for the MagTag Library

#### Parameters

- **default\_bg** – The path to your default background image file or a hex color. Defaults to 0xFFFFFFFF.
- **auto\_refresh** (*bool*) – Automatically refresh the eInk after writing to displayio. Defaults to True.
- **rotation** – Default rotation is landscape (270) but can be 0, 90, 180 for portrait/rotated
- **debug** – Turn on debug print outs. Defaults to False.

```
qr_code(qr_data, *, qr_size=1, x=0, y=0, qr_color=0)
```

Display a QR code on the eInk

#### Parameters

- **qr\_data** – The data for the QR code.
- **qr\_size** (*int*) – The scale of the QR code.
- **x** – The x position of upper left corner of the QR code on the display.
- **y** – The y position of upper left corner of the QR code on the display.

```
set_background(file_or_color, position=None)
```

The background image to a bitmap file.

#### Parameters

- **file\_or\_color** – The filename of the chosen background image, or a hex color.
- **position** (*tuple*) – Optional x and y coordinates to place the background at.

## 5.4 adafruit\_magtag.magtag

Helper library for the Adafruit MagTag.

- Author(s): Melissa LeBlanc-Williams

### 5.4.1 Implementation Notes

#### Hardware:

- [Adafruit MagTag](#)

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's PortalBase library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_PortalBase](https://github.com/adafruit/Adafruit_CircuitPython_PortalBase)

```
class adafruit_magtag.magtag.MagTag(*, url=None, headers=None, json_path=None, regexp_path=None,
                                     default_bg=16777215, status_neopixel=None, json_transform=None,
                                     rotation=270, debug=False)
```

Class representing the Adafruit MagTag.

#### Parameters

- **url** – The URL of your data source. Defaults to `None`.
- **headers** – The headers for authentication, typically used by Azure API's.
- **json\_path** – The list of json traversal to get data out of. Can be list of lists for multiple data points. Defaults to `None` to not use json.
- **regexp\_path** – The list of regexp strings to get data out (use a single regexp group). Can be list of regexps for multiple data points. Defaults to `None` to not use regexp.
- **default\_bg** – The path to your default background image file or a hex color. Defaults to `0x000000`.
- **status\_neopixel** – The pin for the status NeoPixel. Use `board.NEOPIXEL` for the on-board NeoPixel. Defaults to `None`, to not use the status LED
- **json\_transform** – A function or a list of functions to call with the parsed JSON. Changes and additions are permitted for the `dict` object.
- **rotation** – Default rotation is landscape (270) but can be 0, 90, or 180 for portrait/rotated
- **debug** – Turn on debug print outs. Defaults to `False`.

#### **enter\_light\_sleep**(*sleep\_time*)

Enter light sleep and resume the program after a certain period of time.

See <https://circuitpython.readthedocs.io/en/latest/shared-bindings/alarm/index.html> for more details.

**Parameters** **sleep\_time** (*float*) – The amount of time to sleep in seconds

#### **exit\_and\_deep\_sleep**(*sleep\_time*)

Stops the current program and enters deep sleep. The program is restarted from the beginning after a certain period of time.

See <https://circuitpython.readthedocs.io/en/latest/shared-bindings/alarm/index.html> for more details.

**Parameters** **sleep\_time** (*float*) – The amount of time to sleep in seconds

**fetch**(*refresh\_url=None, timeout=10, auto\_refresh=True*)

Fetch data from the url we initialized with, perform any parsing, and display text or graphics. This function does pretty much everything Optionally update the URL

**Parameters**

- **refresh\_url** (*str*) – The overriding URL to fetch from. Defaults to None.
- **timeout** (*int*) – The timeout period in seconds.

**refresh**()

Refresh the display

**set\_text**(*val, index=0, auto\_refresh=True*)

Display text, with indexing into our list of text boxes.

**Parameters**

- **val** (*str*) – The text to be displayed
- **index** – Defaults to 0.
- **auto\_refresh** – Automatically refresh the display after setting the text. Defaults to True

## 5.5 adafruit\_magtag.network

Helper library for the Adafruit MagTag.

- Author(s): Melissa LeBlanc-Williams

### 5.5.1 Implementation Notes

**Hardware:**

- [Adafruit MagTag](#)

**Software and Dependencies:**

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's PortalBase library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_PortalBase](https://github.com/adafruit/Adafruit_CircuitPython_PortalBase)

**class** `adafruit_magtag.network.Network`(\**, status\_neopixel=None, extract\_values=True, debug=False*)

Class representing the Adafruit MagTag.

**Parameters**

- **status\_neopixel** – The pin for the status NeoPixel. Use `board.NEOPIXEL` for the on-board NeoPixel. Defaults to `None`, not the status LED
- **extract\_values** (*bool*) – If true, single-length fetched values are automatically extracted from lists and tuples. Defaults to `True`.
- **debug** – Turn on debug print outs. Defaults to `False`.

**property enabled**

Get or Set whether the WiFi is enabled

## 5.6 adafruit\_magtag.peripherals

Helper Library for the Adafruit MagTag.

- Author(s): Melissa LeBlanc-Williams

### 5.6.1 Implementation Notes

#### Hardware:

- Adafruit MagTag

#### Software and Dependencies:

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>
- Adafruit's PortalBase library: [https://github.com/adafruit/Adafruit\\_CircuitPython\\_PortalBase](https://github.com/adafruit/Adafruit_CircuitPython_PortalBase)

**class** `adafruit_magtag.peripherals.Peripherals`

Peripherals Helper Class for the MagTag Library

**property** `any_button_pressed`

Return whether any button is pressed

**property** `battery`

Return the voltage of the battery

**property** `button_a_pressed`

Return whether Button A is pressed

**property** `button_b_pressed`

Return whether Button B is pressed

**property** `button_c_pressed`

Return whether Button C is pressed

**property** `button_d_pressed`

Return whether Button D is pressed

**deinit**()

Call deinit on all resources to free them

**property** `light`

Return the value of the light sensor. The `neopixel_disable` property must be false to get a value.

```
import time
from adafruit_magtag.magtag import MagTag

magtag = MagTag()

while True:
    print(magtag.peripherals.light)
    time.sleep(0.01)
```

**property** `neopixel_disable`

Enable or disable the neopixels for power savings

**play\_tone**(*frequency*, *duration*)

Automatically Enable/Disable the speaker and play a tone at the specified frequency for the specified duration. It will attempt to play the sound up to 3 times in the case of an error.

**property speaker\_disable**

Enable or disable the speaker for power savings



## INDICES AND TABLES

- [genindex](#)
- [modindex](#)
- [search](#)





## PYTHON MODULE INDEX

### a

adafruit\_magtag.graphics, 12  
adafruit\_magtag.magtag, 13  
adafruit\_magtag.network, 15  
adafruit\_magtag.peripherals, 15



## INDEX

### A

adafruit\_magtag.graphics  
  module, 12

adafruit\_magtag.magtag  
  module, 13

adafruit\_magtag.network  
  module, 15

adafruit\_magtag.peripherals  
  module, 15

any\_button\_pressed (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16

### B

battery (*adafruit\_magtag.peripherals.Peripherals* *prop-*  
  *erty*), 16

button\_a\_pressed (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16

button\_b\_pressed (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16

button\_c\_pressed (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16

button\_d\_pressed (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16

### D

deinit() (*adafruit\_magtag.peripherals.Peripherals*  
  *method*), 16

### E

enabled (*adafruit\_magtag.network.Network* *property*),  
  15

enter\_light\_sleep()  
  (*adafruit\_magtag.magtag.MagTag* *method*), 14

exit\_and\_deep\_sleep()  
  (*adafruit\_magtag.magtag.MagTag* *method*), 14

### F

fetch() (*adafruit\_magtag.magtag.MagTag* *method*), 14

### G

Graphics (*class in adafruit\_magtag.graphics*), 13

### L

light (*adafruit\_magtag.peripherals.Peripherals* *prop-*  
  *erty*), 16

### M

MagTag (*class in adafruit\_magtag.magtag*), 14

module

- adafruit\_magtag.graphics, 12
- adafruit\_magtag.magtag, 13
- adafruit\_magtag.network, 15
- adafruit\_magtag.peripherals, 15

### N

neopixel\_disable (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16

Network (*class in adafruit\_magtag.network*), 15

### P

Peripherals (*class in adafruit\_magtag.peripherals*), 16

play\_tone() (*adafruit\_magtag.peripherals.Peripherals*  
  *method*), 16

### Q

qrcode() (*adafruit\_magtag.graphics.Graphics* *method*),  
  13

### R

refresh() (*adafruit\_magtag.magtag.MagTag* *method*),  
  15

### S

set\_background() (*adafruit\_magtag.graphics.Graphics*  
  *method*), 13

set\_text() (*adafruit\_magtag.magtag.MagTag*  
  *method*), 15

speaker\_disable (*adafruit\_magtag.peripherals.Peripherals*  
  *property*), 16