
MagicBlue Documentation

Release 0.6.0

Benjamin Piouffle

Jan 21, 2018

Contents

1	License	3
2	Contribute	5
3	Basic usage	7
4	MagicBlue API reference	9
4.1	MagicBlue	9
4.2	Effect	11
	Python Module Index	13

This is the reference for MagicBlue's API. To get information about how to install it or to use magicblueshell, please refer to this link: <https://github.com/Betree/magicblue/blob/master/README.md>

CHAPTER 1

License

The project is licensed under the MIT license.

CHAPTER 2

Contribute

- Issue Tracker: <https://github.com/Betree/magicblue/issues>
- Source Code: <https://github.com/Betree/magicblue>

CHAPTER 3

Basic usage

```
from magicblue import MagicBlue

bulb_mac_address = 'XX:XX:XX:XX:XX:XX'
bulb = MagicBlue(bulb_mac_address, 9) # Replace 9 by whatever your version is,
↪ (default: 7)
bulb.connect()
bulb.set_color([255, 0, 0])           # Set red
bulb.set_random_color()              # Set random
bulb.turn_off()                      # Turn off the light
bulb.turn_on()                       # Set white light
```


4.1 MagicBlue

class `magicbluelib.MagicBlue` (*mac_address*, *version=7*, *addr_type=None*)

Class to interface with Magic Blue light

connect (*bluetooth_adapter_nr=0*)

Connect to device

Parameters **bluetooth_adapter_nr** – bluetooth adapter name as shown by “hciconfig” command. Default : 0 for (hci0)

Returns True if connection succeed, False otherwise

disconnect ()

Disconnect from device

get_date_time ()

Retrieve date/time from bulb

get_device_info ()

Retrieve device info

get_device_name ()

Returns Device name

get_time_schedule ()

Request the time schedule

is_connected ()

Returns True if connected

set_color (*rgb_color*)

Change bulb's color

Parameters **rgb_color** – color as a list of 3 values between 0 and 255

set_date_time (*datetime_value*)

Set date/time in bulb

Parameters **datetime_value** – datetime to set

set_effect (*effect, effect_speed*)

Set an effect, with effect_speed as speed

Parameters

- **effect** – An effect (see [Effect](#))
- **effect_speed** – integer (range: 1..20) where each unit represents around 200ms

set_random_color ()

Change bulb's color with a random color

set_time_schedule (*timer_items*)

Set the time schedule

Parameters **timer_items** – list with TimerItem, max of 6, dict with items:

- used, boolean
- turn, 'on'/'off'
- date_time, datetime.datetime
- time, datetime.time
- repeat, set with MagicBlue.Weekday
- effect, MagicBlue.Effect
- effect_speed, 1..20
- r, 0..255
- g, 0..255
- b, 0..255

date_time and time+repeat are exclusive

set_warm_light (*intensity=1.0*)

Equivalent of what they call the “Warm light” property in the app that is a strong white / yellow color, stronger than any value you may get by setting rgb color.

Parameters **intensity** – the intensity between 0.0 and 1.0

test_connection ()

Test if the connection is still alive

Returns True if connected

turn_off ()

Turn off the light

turn_on (*brightness=None*)

Set white color on the light

Parameters **brightness** – a float value between 0.0 and 1.0 defining the brightness

4.2 Effect

class magicbluelib.Effect

An enum of all the possible effects the bulb can accept

```
blue_gradual_change = 40
blue_strobe_flash = 51
cyan_gradual_change = 42
cyan_strobe_flash = 53
green_blue_cross_fade = 47
green_gradual_change = 39
green_strobe_flash = 50
purple_gradual_change = 43
purple_strobe_flash = 54
red_blue_cross_fade = 46
red_gradual_change = 38
red_green_cross_fade = 45
red_strobe_flash = 49
seven_color_cross_fade = 37
seven_color_jumping_change = 56
seven_color_strobe_flash = 48
white_gradual_change = 44
white_strobe_flash = 55
yellow_gradual_change = 41
yellow_strobe_flash = 52
```


m

magicbluelib, 9

B

blue_gradual_change (magicbluelib.Effect attribute), 11
blue_strobe_flash (magicbluelib.Effect attribute), 11

C

connect() (magicbluelib.MagicBlue method), 9
cyan_gradual_change (magicbluelib.Effect attribute), 11
cyan_strobe_flash (magicbluelib.Effect attribute), 11

D

disconnect() (magicbluelib.MagicBlue method), 9

E

Effect (class in magicbluelib), 11

G

get_date_time() (magicbluelib.MagicBlue method), 9
get_device_info() (magicbluelib.MagicBlue method), 9
get_device_name() (magicbluelib.MagicBlue method), 9
get_time_schedule() (magicbluelib.MagicBlue method), 9
green_blue_cross_fade (magicbluelib.Effect attribute), 11
green_gradual_change (magicbluelib.Effect attribute), 11
green_strobe_flash (magicbluelib.Effect attribute), 11

I

is_connected() (magicbluelib.MagicBlue method), 9

M

MagicBlue (class in magicbluelib), 9
magicbluelib (module), 9

P

purple_gradual_change (magicbluelib.Effect attribute), 11
purple_strobe_flash (magicbluelib.Effect attribute), 11

R

red_blue_cross_fade (magicbluelib.Effect attribute), 11

red_gradual_change (magicbluelib.Effect attribute), 11
red_green_cross_fade (magicbluelib.Effect attribute), 11
red_strobe_flash (magicbluelib.Effect attribute), 11

S

set_color() (magicbluelib.MagicBlue method), 9
set_date_time() (magicbluelib.MagicBlue method), 9
set_effect() (magicbluelib.MagicBlue method), 10
set_random_color() (magicbluelib.MagicBlue method), 10
set_time_schedule() (magicbluelib.MagicBlue method), 10
set_warm_light() (magicbluelib.MagicBlue method), 10
seven_color_cross_fade (magicbluelib.Effect attribute), 11
seven_color_jumping_change (magicbluelib.Effect attribute), 11
seven_color_strobe_flash (magicbluelib.Effect attribute), 11

T

test_connection() (magicbluelib.MagicBlue method), 10
turn_off() (magicbluelib.MagicBlue method), 10
turn_on() (magicbluelib.MagicBlue method), 10

W

white_gradual_change (magicbluelib.Effect attribute), 11
white_strobe_flash (magicbluelib.Effect attribute), 11

Y

yellow_gradual_change (magicbluelib.Effect attribute), 11
yellow_strobe_flash (magicbluelib.Effect attribute), 11