

Raging Bits Magic Lights v1.0

Specs

ESP32 WROOM module based system.

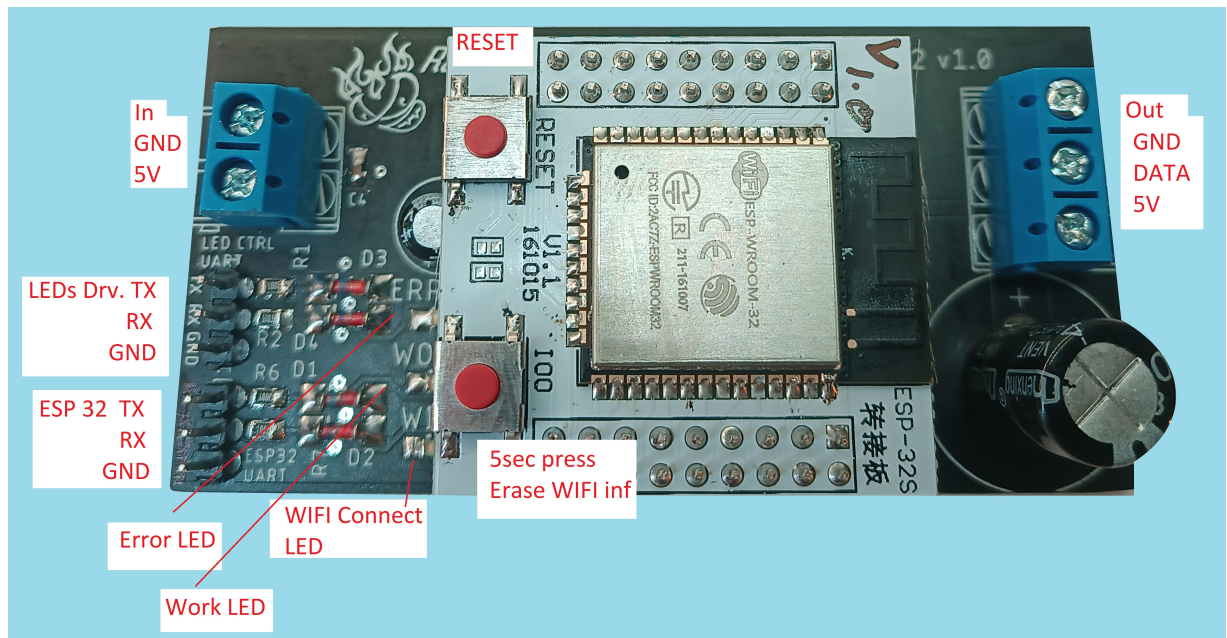
Addressable LEDs serial driver.

ESP32 UART access.

LEDs driver UART access.

5V input up to 5Amps from input to output.

Device interface



Hardware configuration

ESP32 pinout:

LEDs driver Uart RX – 16

LEDs driver Uart TX – 17

LEDs driver Reset – 5

Programming Uart RX – 3

Programming Uart TX – 1

Programming enter (also used for Wi-Fi info erase) – 0

Work LED (green) – 23

Wi-Fi connected LED (blue) – 22

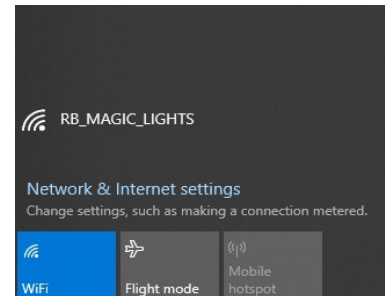
Error LED (red) – 21

Default setup work

Setting up Wi-Fi

Power the device with 5V.

Using a device capable of Wi-Fi connection and html browsing, like a mobile phone, search for a Wi-Fi SSID named "RB_MAGIC_LIGHTS".



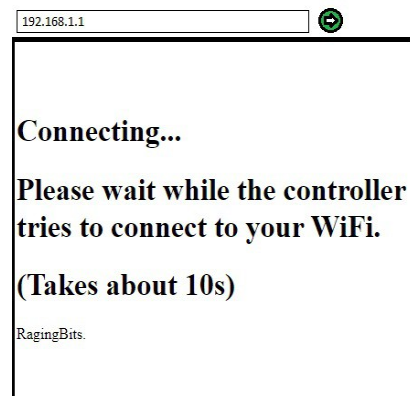
Connect to this Wi-Fi, no password needed.

Open a browser and in the address bar input the device gateway address "192.168.1.1" and press enter.

A webpage will load so that the regular Wi-Fi information that the device should connect to, may be provided.



The device will then try to connect to the Wi-Fi using the information provided. During this time it will blink red/green.



Once connected, will return with the current setup, IP address given by the router and free amount of memory for the effects. At this point will light up the Wi-Fi connected blue LED.



After this, the device is ready to be used.

Install python

In order to have the basic python script working, "python 3" needs to be installed along with "pyserial".

From <https://www.python.org/downloads/> download and install python 3 current version.

After installation, add the python executable path to the system. The current setup is for windows10, so this path needs to be added to the Environment Variable PATH.

Windows 10 - Right click on Start Menu to get Power User Task Menu → Select System → Advanced System Settings → Environment variables → Choose Edit "PATH" .

Add the Python 3 folder path to the PATH variable.

After this "pyserial" needs to be installed.

Open a command prompt and set:

:>python -m pip install pyserial

After finishing, pyserial is now available, and the system is ready for the scripts.

Simple mode – Sending an example

Download both Effects and RB_Magic Lights_WinApp.zip application from https://github.com/RagingBits/RB_Magic_Lights

Extract the windows application and run RB_MagicLights.exe. (Some antivirus will run this for the first time in a sandbox. That's just fine, just wait for it to finish before moving forward.)

In the field IP, insert the IP given by the device when the Wi-Fi setup was done. *(Despite of uncommon it is normal for routers to change devices IPs eventually. If the device stops responding, this could be the case, just follow the Wi-Fi setup process, by connecting to the device hotspot "RB_MAGIC_LIGHTS" and opening a webpage to "192.168.1.1". If the device is connected to the Wi-Fi (with Blue LED on), it will simply return the information page with the router given IP.)*

Select the LEDs colour format from the drop box. For example WS2811 will have an RGB format, meanwhile WS2812 will be GRB format.

Find the LEDs pattern file (Generated using "Led Matrix Studio 0.10" – See **Generating patterns**).

Press the send button. A few seconds later the device should be prepared and working with the given pattern.

Generating patterns

Despite of plenty of ways to generate patterns, the one chosen is using "Led Matrix Studio 0.10", for RGB based LEDs like WS2811/ WS2812.

The new pattern shall have a number of columns that matches the number of LEDs used in the string.

The number of lines or/and frames will compose the sequence of strings setups.

For the Magic Lights purpose, each line, one after the other, frame after frame will always be interpreted as a single string run. Therefore a sequence of lines/frames with N columns will lead to a sequence of a string of N LEDs behaviour.

The chosen colour setup for the project will then be "RGB".

Advanced mode - Sending data

PC_trx.py is the python script used to send raw data to the device. The file to be sent **MUST** be raw binary with the following format byte:

[MSB len][LSB len][seq1 byte 0]...[seq1 byte N]...[seqM byte 0]...[seqM byte N]

[MSB LEN][LSB LEN] - 2 bytes indicating the length of 1 sequence in number of bytes

[byte 0]...[byte N] – First sequence data.

...

[byte 0]...[byte N] – M sequence data.

N is LEN.

The number of sequences is given by the total file size subtracted by 2, the number of bytes comprising the length value of a single sequence. The sequence size will be the size of all the data needed to fulfil a single string of LEDs.

For example, a string of 100 LEDS of RGB data format, will take 24bits of data per LED. This will give that each sequence will have 300bytes of information.

To send the data using PCTrx python script, open a command prompt and set:

```
> python PC_trx.py -ip <magic lights IP address> -fp <raw effects file >
```

Ex.:

```
> python PC_trx.py -ip 192.168.1.12 -fp Effects\100LED_CHASER.leds.esp32
```

Advanced mode – Converting data

AddressableLEDsConverterForESP32.py is the python script used to convert from "Led Matrix Studio" output file, to the raw data file needed by PC_trx.py.

In order to perform a conversion, open a command prompt and set:

```
> python AddressableLEDsConverterForESP32.py -f <file to be converted> -t <colour scheme of the LEDs>
```

Ex.:

```
> python AddressableLEDsConverterForESP32.py -f "Effects\50LED_RAINBOW_ROTATE.leds" -t RGB
```

Output will be a file with the same name but a new type:

50LED_RAINBOW_ROTATE.leds.esp32

References – links

https://github.com/RagingBits/RB_Magic_Lights

<https://sourceforge.net/projects/led-matrix-studio/>

<https://www.python.org/downloads/>