



OCTO MK2 – API

API requirement for the 8-universe eDMX to LED pixel controller with network chaining in a compact 4-module DIN-rail form factor.



Document Version	2.0
Last Updated	13 November 2023
Compatible firmware	OCTO MK2 Firmware Version V4.2 ENTTEC strongly advise keeping your device firmware up to date for optimal performance.

Contents

- Purpose 2
- Application Messages 2
 - 1. Current Configuration..... 2
 - 2. Reset to Factory Default..... 2
 - 3. Access to Boot..... 2
 - 4. Reboot Device 2
 - 5. Identify..... 3
 - 6. Art-Net Stats..... 3
 - 7. ESP Stats 3
 - 8. sACN Stats 3
 - 9. KiNet Stats..... 3
 - 10. Change Direct Live Mode Settings..... 4
 - 11. Standalone Mode - Enable/Disable Standalone Mode.....7
 - 12. Standalone Mode - Set Show file by Port.....7
 - 13. Standalone Mode - Preview Effect by Port.....7
 - 14. Standalone Mode - Set Effect for Each Show File..... 9
- Conclusion 9
- Appendix 1 10

Purpose

This document specifies the interface requirement for text-based application programs such as Command Prompt or any third-party applications to communicate with the ENTTEC OCTO MK2 Firmware Version V4.2.

Application Messages

This section demonstrates the API message format required for different communication purposes alongside the parameters for each setting.

The API messages are demonstrated through examples factored in as follows:

- PC Command Line Interface (CLI): utilizing CURL. **Note:** Adjust 'curl' as needed for other CLI tools.
- OCTO MK2 IP Address: [10.10.3.51](#). **Note:** Substitute this IP address with the one assigned to your device.

1. Current Configuration

This example message requests OCTO MK2 to display the current configuration.

Example Message

```
curl http://10.10.3.51/index.html?config.cgi
```

2. Reset to Factory Default

This message requests the OCTO MK2 resume settings back to factory default.

Example Message

```
curl http://10.10.3.51/index.html?set_to_defaults=1
```

The device will be reset to factory default without a return message. Request current configuration information for verification after change.

3. Access to Boot

This message requests to access the boot of the OCTO MK2.

Example Message

```
curl http://10.10.3.51/index.html?firmware.cgi
```

Once successful, the return message will be *'curl: (56) Recv failure: Connection was reset.'*

4. Reboot Device

This message requests the OCTO MK2 to reboot or to exit the boot.

Example Message

```
curl http://10.10.3.51/index.html?reboot=1
```

Once successful, the return message would be *'curl: (56) Recv failure: Connection was reset.'*

5. Identify

This message requests OCTO MK2 to send output to light up all the connected pixels for correct wiring identification without the need to provide control data.

Example Message

```
curl http://10.10.3.51/index.html?identify=1
```

6. Art-Net Stats

This API message requests OCTO MK2 to display the current Art-Net Stats.

Example Message

```
curl --http0.9 http://10.10.3.51/index.html?artnet_stats
```

7. ESP Stats

This API message requests OCTO MK2 to display the current ESP Stats.

Example Message

```
curl --http0.9 http://10.10.3.51/index.html?esp_stats
```

8. sACN Stats

This API message requests OCTO MK2 to display the current sACN Stats.

Example Message

```
curl --http0.9 http://10.10.3.51/index.html?sacn_stats
```

9. KiNet Stats

This message requests OCTO MK2 to display the current KiNet Stats.

Example Message

```
curl --http0.9 http://10.10.3.51/index.html?kinet_stats
```

10. Change Direct Live Mode Settings

This application message allows user to change settings of the OCTO MK2 within one command.

Example Message

```
curl --header "Content-Type: application/x-www-form-urlencoded" -d
"ip=192,168,0,10&netmask=255,255,255,0&dhcp_enable=1&gateway_ip=192,168,0,254&config_name=Octo&led
_strip_protocol=16,16&lighting_protocol=0&universe_count=4,4&start_universe=0,0&dmx_start_address=1,1
&pixel_fanout=1,1&pixel_order=0,0&used_pixels=0&apa102_global=31,31&tm1814_cc=63,63&custom_bit1=10
90,750&custom_bit0=300,300&custom_bitlength=1410,1350&custom_reset=280000,100000&custom_protoco
l=0,0" 10.10.3.51/settings.html
```

Space here

The message is composited with 3 elements:

- **Header line**
- **Configuration line**: where each individual setting is placed together with “&” and the changeable values are highlighted in magenta.
- **Device IP and command area line**: replace the IP address from your device.

To change settings for Direct Mode, enter the **Example Message** above with your IP address replaced and adjust the changeable values by following the **Application Message Format and Parameter Table**.

The changes will be conducted directly without a return message. Request current configuration information for verification after change.

Note:

Due to pdf format restriction, it is recommended to edit the code in plain text editing tool such as Notepad before copying the code to the prompt command.

Application Message Format and Parameter Table:

Settings	Description	Parameter
ip	The Static IP when DHCP is disabled.	Format: ip= <i>x,x,x,x</i> Parameter <i>x</i> = 0~255 E.g. 192.168.0.10 = <i>192,168,0,10</i>
netmask	The netmask when static IP is enabled.	Format netmask= <i>x,x,x,x</i> Parameter <i>x</i> = 0~255
dhcp_enable	Enable or disable DHCP.	Format dhcp_enable= <i>x</i> Parameter <i>x</i> = 0 or 1 0: disable 1: enable
gateway_ip	Gateway IP when static IP is enabled.	Format gateway_ip= <i>x,x,x,x</i> Parameter <i>x</i> = 0~255

config_name	The node name discoverable on the network.	<p>Format config_name=<i>x</i></p> <p>Parameter <i>x</i>= any value in alphabet and number. <i>E.g. Node Name: OCTO MK2</i> config_name=OCTO MK2</p>
led_strip_protocol	Select the output LED protocol.	<p>Format led_strip_protocol=<i>x,y</i></p> <p>Parameter <i>x</i>= 0~22, Port 1 LED protocol <i>y</i>= 0~22, Port 2 LED protocol <i>LED protocol list in Appendix 1.</i></p>
lighting_protocol	Choose input eDMX protocol.	<p>Format lighting_protocol=<i>x</i></p> <p>Parameter <i>x</i>= 0/1/2/3 0: Art-Net 1: sACN 2: ESP 3: KiNet</p>
universe_count	Set universe capacity for Port 1 and 2.	<p>Format universe_count=<i>x,y</i></p> <p>Parameter <i>x</i>= 0~8, Port 1 universe capacity <i>y</i>= 0~8, Port 2 universe capacity</p>
start_universe	Define starting universe for Port 1 and 2.	<p>Format start_universe=<i>x,y</i></p> <p>Parameter <i>x</i>= see below, Port 1 starting universe <i>y</i>= see below, Port 2 starting universe 0-32767 for Art-Net 0-255 for ESP 0-63999 for sACN 0-65535 for KiNet</p>
dmx_start_address	Nominate the starting DMX channel address for OCTO MK2 to output from Port 1 and 2.	<p>Format dmx_start_address=<i>x,y</i></p> <p>Parameter <i>x</i>= 1-512, Port 1 DMX starting address <i>y</i>= 1-512, Port 2 DMX starting address</p>
pixel_fanout	Pixel grouping which allows multiple pixels to be controlled as one 'virtual pixel'.	<p>Format pixel_fanout=<i>x,y</i></p> <p>Parameter <i>x</i>= 0-512*, Port 1 grouped pixels <i>y</i>= 0-512*, Port 2 grouped pixels</p>
pixel_order	Configure how R, G, B, W colours are mapped to pixels	<p>Format pixel_order=<i>x,y</i></p> <p>Parameter <i>x</i>= 0-35, Port 1 pixel order <i>y</i>= 0-35, Port 2 pixel order <i>Pixel order list in Appendix 1.</i></p>

* Max. value depending on other output settings such as universe, pixel order and DMX start address. If set value is higher than the max value allowed, OCTO MK2 will proceed with the max value allowed.

used_pixels	Define the number of mapped pixels.	<p>Format used_pixels=<i>x,y</i></p> <p>Parameter <i>x</i>= 0~1360*, Port 1 mapped pixels. <i>y</i>= 0~1360*, Port 2 mapped pixels.</p>
apa102_global	Maximum global brightness when APA-102 is selected.	<p>Format apa102_global=<i>x,y</i></p> <p>Parameter <i>x</i>= 0-63, Port 1 APA-102 global brightness <i>y</i>= 0-63, Port 2 APA-102 global brightness</p>
tm1814_cc	Maximum global brightness when TM-1814 is selected.	<p>Format tm1814_cc=<i>x,y</i></p> <p>Parameter <i>x</i>= 0-63, Port 1 TM-1814 global brightness <i>y</i>= 0-63, Port 2 TM-1814 global brightness</p>
custom_bit1	Bit 1 High Time (T1H) for custom LED protocol. This will only take effect when custom protocol is enabled.	<p>Format custom_bit1=<i>x,y</i></p> <p>Parameter <i>x</i>= 100-3,000#, Port 1 T1H <i>y</i>= 100-3,000#, Port 2 T1H</p>
custom_bit0	Bit 0 High Time (T0H) for custom LED protocol. This will only take effect when custom protocol is enabled.	<p>Format custom_bit0=<i>x,y</i></p> <p>Parameter <i>x</i>= 100-3,000#, Port 1 T0H <i>y</i>= 100-3,000#, Port 2 T0H</p>
custom_bitlength	Overall bit Time (TH+TL) for custom LED protocol. This will only take effect when custom protocol is enabled.	<p>Format custom_bitlength=<i>x,y</i></p> <p>Parameter <i>x</i>= 500-3,000, Port 1 overall bit time <i>y</i>= 500-3,000, Port 2 overall bit time</p>
custom_reset	Reset Time (RES) for custom LED protocol. This will only take effect when custom protocol is enabled.	<p>Format custom_reset=<i>x,y</i></p> <p>Parameter <i>x</i>= 25,000-1,000,000, Port 1 Reset time <i>y</i>= 25,000-1,000,000, Port 2 Reset time</p>
custom_protocol	Enable or disable LED custom protocol for Port 1 and 2.	<p>Format custom_protocol=<i>x,y</i></p> <p>Parameter <i>x</i>= 0 or 1, Port 1 custom protocol status <i>y</i>= 0 or 1, Port 2 custom protocol status</p> <p>0: disable 1: enable</p>

* Max. value depending on overall bit length set. If set value is higher than the max value allowed, OCTO MK2 will proceed with the max value allowed.

11. Standalone Mode - Enable/Disable Standalone Mode

This message requests OCTO MK2 to enable/disable standalone mode.

Example Message

```
curl --header "Content-Type: application/x-www-form-urlencoded" -d "standalone=1"
10.10.3.51/standalone.html?enable_standalone
```

Format

Standalone=*x*

Parameter

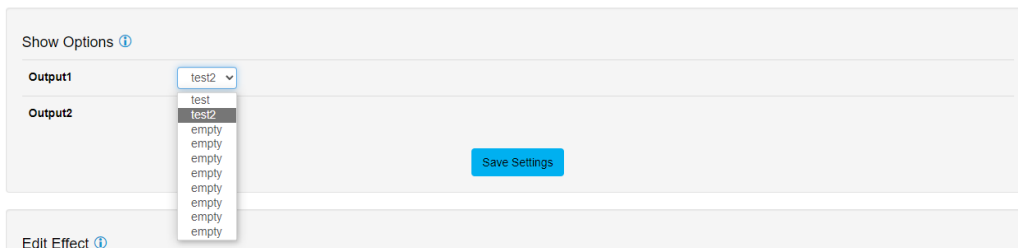
x = 0 or 1

1: Enable Standalone Mode

0: Disable Standalone Mode

12. Standalone Mode - Set Show file by Port

This message sets the show file by port in OCTO MK2 Standalone Mode.



Example Message

```
curl --header "Content-Type: application/x-www-form-urlencoded" -d "set_effects=1,4"
10.10.3.51/standalone.html?set_ports
```

Format

set_effects= *x,y*

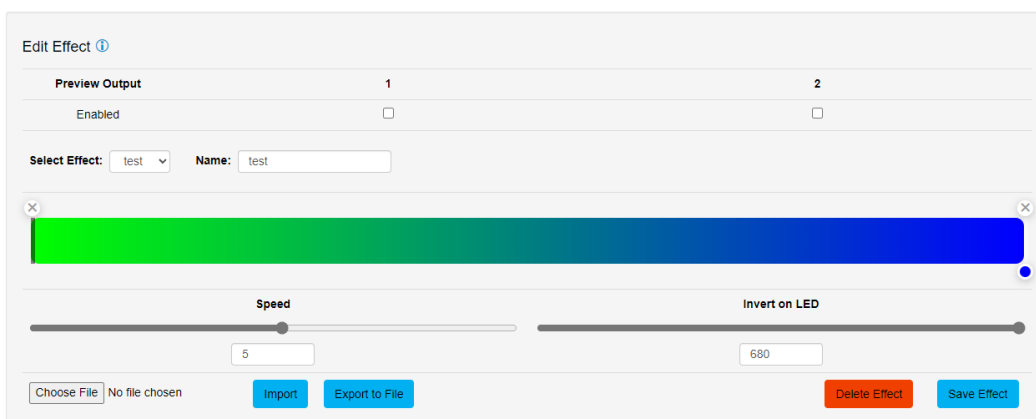
Parameter

x= 0~10, Port 1 show file number by the order in the drop-down list.

y= 0~10, Port 2 show file number by the order in the drop-down list.

13. Standalone Mode - Preview Effect by Port

This message requests the OCTO MK2 to output the data set for Standalone Effect by port enabling preview.



Example Message

```
curl --header "Content-Type: application/x-www-form-urlencoded" -d
"port=0,0&speed=10&invert=0&marker_num=2&markers=255,0,0,0,0,0,255,0,100"
10.10.3.51/standalone.html?update_effect
```

Parameter Table

Settings	Description	Parameter
port	Enable or disable the preview output for each port.	<p>Format: Port= <i>x,y</i></p> <p>Parameter <i>x</i>= 0 or 1, Port 1 preview output <i>y</i>= 0 or 1, Port 2 preview output</p> <p>0: disable 1: enable</p>
speed	The effect play speed.	<p>Format speed=<i>x</i></p> <p>Parameter <i>x</i>= -125~125</p>
invert	Invert on LED.	<p>Format invert=<i>x</i></p> <p>Parameter <i>x</i>= 0~680*</p>
Marker_num	The total number of the colour marker set in the effect.	<p>Format marker_num=<i>x</i></p> <p>Parameter <i>x</i>= 0~15</p>
markers	The settings for each colour markers and its position in the effect.	<p>Format Marker_num=<i>R,G,B,W,X,R2,G2,B2,W2,X2,...etc.</i></p> <p>Parameter <i>R</i>= 0~255, R colour intensity <i>G</i>= 0~255, G colour intensity <i>B</i>= 0~255, B colour intensity <i>W</i>= 0~255, W colour intensity <i>X</i>= 0~100, the colour marker position 0 represent the start of the effect. 100represent the end of the effect.</p>

* Max. value depending on other output settings such as universe, pixel order and DMX start address. If set value is higher than the max value allowed, OCTO MK2 will proceed with the max value allowed.

14. Standalone Mode - Set Effect for Each Show File

This message sets the effect colour markers for each show file in Standalone Mode.

Example Message

```
curl --header "Content-Type: application/x-www-form-urlencoded" -d  
";test,5,2,2,0,255,0,0,0,0,255,0,100&;test2,10,4,2,255,0,0,0,0,0,255,0,100&;,0,0,2,255,0,0,0,0,0,255,0,100&;,0,0,2,2  
55,0,0,0,0,0,255,0,100&;,0,0,2,255,0,0,0,0,0,255,0,100&;,0,0,2,255,0,0,0,0,0,255,0,100&;,0,0,2,255,0,0,0,0,0,255,  
0,100&;,0,0,2,255,0,0,0,0,0,255,0,100&;,0,0,2,255,0,0,0,0,0,255,0,100&;,0,0,2,255,0,0,0,0,0,255,0,100"
```

Format

;show_1_name,speed,invert,marker_num,R,G,B,W,position, R2,G2,B2,W2,position2,...etc.

- Up to 15 RGBW colours and positions can be added for each show.
- Add & to set maximum 10 shows simultaneously.

Parameter

Refer previous table for parameters.

Conclusion

This brings us to the end of the guide. By utilizing the API messages, OCTO MK2 allows third-party hardware integration and communications from user's preferable command system. This guide provides the message format required for communication interface in examples. While there will be return messages for most of the commands, the Change Settings and Reset to Factory Default command will be executed directly. It is recommended to request current configuration information for verification and confirmation.

enttec.com

MELBOURNE AUS / LONDON UK / RALEIGH-DURHAM USA / DUBAI UAE

Due to constant innovation, information within this document is subject to change.

Appendix 1

Parameter Table

This section provides the parameter for the LED Strip Protocol and Pixel Order in [Change Settings](#).

LED Strip Protocol

1.	APA-104
2.	SK6812
3.	SPXL_16BIT
4.	SPXL_8BIT
5.	TM1804
6.	TM1812
7.	TM1814
8.	UCS1903
9.	UCS2903
10.	UCS2904
11.	UCS8903_16BIT
12.	UCS8903_8BIT
13.	UCS8904_16BIT
14.	UCS8904_8BIT
15.	GS8208B
16.	WS2811
17.	WS2812
18.	WS2812B
19.	WS2813
20.	WS2815
21.	WS2818
22.	9PDOT_16BIT
23.	9PDOT_8BIT
24.	SJ1221_16BIT
25.	SJ1221_8BIT
26.	TLC5973_16BIT
27.	TLC5973_8BIT
28.	APA-102

Pixel Order

0	RGB
1	RBG
2	GRB
3	GBR
4	BRG
5	BGR
6	RGBW
7	RGWB
8	RBGW
9	RBWG
10	RWGB
11	RWBG
12	GRBW
13	GRWB
14	GBRW
15	GBWR
16	GWRB
17	GWBR
18	BRGW
19	BRWG
20	BGRW
21	BGWR
22	BWRG
23	BWGR
24	WRGB
25	WRBG
26	WGRB
27	WGBR
28	WBRG
29	WBGR
30	WwCwA
31	WWACw
32	CwWwA
33	CwAWw
34	AWwCw
35	ACwWw