

StageLight API Documentation

Welcome to the **StageLight API Documentation**! The StageLight API allows external control of core functions such as cue lists, **fixture control adjustments**, and tempo adjustments. This guide provides comprehensive information about the protocol, command structure, and usage examples.

Activating the API

The **StageLight API** is disabled by default. You can enable it in the **App Settings** under:

Settings > Protocols > StageLight API > Enable

General Information

Communication Protocol

- **Transport Protocol:** UDP
- **UDP Port:** 60100
- **Message Format:** The API uses a binary message structure.

General Message Structure

Each message consists of the following fields:

Field	Size	Description
Protocol Version	2 Bytes	First byte: Major version, second byte: Minor version (0100 = Version 1.0).
Module	1 Byte	Specifies the target module (e.g., cue list, fixture control, tempo).
Command	1 Byte	Specifies the action to perform in the module.
Payload	Variable	Module-specific data such as target IDs, values, or additional parameters.

Modules Overview

Modules

Module	Value	Description
CueList	01	For controlling cue lists (e.g., toggle, set levels, go to next).
FixtureControl	02	For managing fixture control adjustments (e.g., toggle, flash, set levels).
Tempo	03	For managing tempo adjustments (e.g., tap, adjust BPM).

Commands Documentation

1. CueList Module (01)

Commands

Command	Value	Description
toggle	00	Toggle activation of a cue list.
level	01	Set intensity level for a cue list.
goToNext	02	Advance to the next cue within a cue list.
pauseResume	03	Pause or resume playback of a cue list.

Payload Structure

Field	Size	Description
Index	2 Bytes	The index of the target cue list (0-65535).
Value	1 Byte	Required for level: range 0–100 (%).

Examples

- Toggle a cue list (Index 3):

```
010001000003
```

- Set cue list level to 75% (Index 5):

```
0100010100054B
```

- Go to next cue (Index 2):

```
010001020002
```

- Pause/Resume cue list (Index 7):

```
010001030007
```

2. FixtureControl Module (02)

Commands

Command	Value	Description
toggle	00	Toggle activation of a fixture.
level	01	Set intensity level for a fixture.
flash	02	Flash a fixture (on/off toggle).

Payload Structure

Field	Size	Description
Index	2 Bytes	The index of the target fixture (0-65535).
Value	1 Byte	Required for level (0-100%) or flash (>0 = ON).

Examples

- Toggle a fixture (Index 4):

```
010002000004
```

- Set fixture level to 50% (Index 10):

```
01000201000A32
```

- Flash a fixture (Index 1, on):

```
01000202000101
```

3. Tempo Module (03)

Commands

Command	Value	Description
tapTempo	00	Register a tap for tempo calculation.
adjustTempo	01	Directly set the tempo (1-255 BPM).
tapAdjust	02	Increment or decrement tempo (0 = increase, >0 = decrease).

Payload Structure

Field	Size	Description
Value	1 Byte	Required for <code>adjustTempo</code> or <code>tapAdjust</code> . For <code>tapAdjust</code> : <code>0</code> = increase, <code>>0</code> = decrease.

Examples

- Tap tempo:

```
01000300
```

- Set tempo to 120 BPM:

```
0100030178
```

- Increase tempo:

```
0100030200
```

- Decrease tempo:

```
0100030201
```

Summary of Payload Ranges

Field	Range	Notes
Level	0–100	Used for cue list and fixture intensity.
Tempo	1–255	Direct BPM values for tempo adjustment.
Adjust Value	0 (increase), >0 (decrease)	Used for tempo step adjustments.

FAQ

How do I enable the StageLight API?

Navigate to: **Settings > Advanced > StageLight API > Enable API**

Can I change the UDP port?

No, the port is fixed at 60100.

What happens if the app receives an invalid command?

Invalid commands or malformed messages are ignored, and no action is performed.

Additional Notes

- **Testing with Companion Generic UDP Module:**

- Use **Hex Byte Format** when entering payloads:

```
0100030178
```

- Do **not** use **prefixes** like `\x`.

- **Testing with Netcat:**

Use `printf` on the command line to simulate raw UDP transmission:

```
printf "\x01\x00\x03\x01\x78" | nc -u localhost 60100
```

Enjoy controlling your StageLight system!