

## E-Streamer Mk2 Show Control over UDP API

<b>Version</b>	1.4
<b>Last Updated:</b>	29/10/18

### Purpose

This document specifies the UDP packet structure that allows E-Streamer Show Control over UDP. An external computer or controller sends the request message and E-Streamer acts on it.

### Setup

- Any computer or machine capable of sending UDP packets over TCP/IP stack.

### Specifications

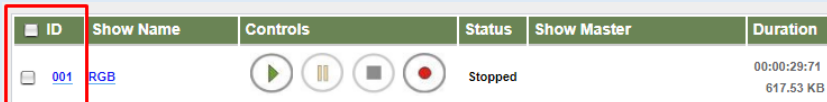
- All data packets must be sent on **UDP port 3000 (0xBB8)**.
- Packets must be sent to the **IP address of the E-Streamer Mk2 unit**.

### Response Message

- Any response will be sent back to the **originating IP address**.

### Show Control Message Structure

- The **entire message** must be sent to have the required effect.
- The same message format is sent by the E-Streamer as a response (for selected commands).
- The total size of this message is 136 bytes (always)**

Field Name	Segment No.	No. of Bytes	Description
Show ID	1	4	ShowID as shown on the Show Control page of the E-Streamer 
Command	2	4	Command to be executed on show. Refer to <a href="#">supported commands</a>
Support Data	3	128	Show data - Supporting data for command sent (e.g. show loop value) Show status - Data about status of show received as response

**Note:** Above fields in API messages must be stored in Little Endian Byte order, with the least significant byte at the lowest address of the field in the message.

## Supported Show Commands

Command		Description
Decimal value	Hex Value	
0	00	Show STOP command
12	0C	Show STOP ALL command
1	01	Show PLAY command
2	02	Show PAUSE command (or RESUME)
3	03	Show RECORD command
4	04	Show STATUS command (returns Response)
5	05	Show INIT command (Initialize show)
6	06	Show DELETE command (Delete recording)
8	08	Set Master Fader
9	09	Get Master Fader

## Support Data Segment Format

This segment contains either the Show Data (request) or Show Status (response).

### Note:

These are the **last 128 bytes in the Show Control message**

### Show Data (request)

Must be sent with all Show Control requests. Show data can be of two types as follows:

Data type	No. of Bytes	Description
Show Loop value	4	0 = Forever, 1 = once, N = n times where n = 1 to 10000
Master Fader value	4	Master fader level at which show is played (0 – 255)

The data structure of the support segment is as follows

Field Name	Segment No.	No. of Bytes	Description
Show data	1	4	Can be either show loop value or master fader value
Pad	2	124	Not used (pad with 0)

### Show Status (response)

Only sent as response for a valid **STATUS** request

### Note:

Show response only received if the show is playing or was played right before the status request

Field Name	Segment No.	No. of Bytes	Description
Status	1	4	Show status - Playing, paused, recording or stopped
Master fader value	2	4	Level at which show is played (0 – 255)
Show time	3	4	Current Show time (in milli sec)
File size	4	8	Show recording file size (in bytes)
Error	5	108	Error Text (if any) – ASCII

## Examples

1. Check ShowID 1 Status
  - ShowID = 1
  - Command = 4 (status command)
  - ShowData = null

Request message will be sent like this:

01 00 00 00 04 00 00 00 {padded with 00 for 128 bytes}

Sample Response message (Show is playing at master-fader = 255):

01 00 00 00 04 00 00 00 01 00 00 00 FF 00 00 00 {other data such as current show time}

2. Play ShowID 3, loop forever:
  - ShowID = 3
  - Command = 1 (play show)
  - ShowData.Loop = 0 (loop forever)

Message will be sent like this:

03 00 00 00 01 00 00 00 00 00 00 00 {padded with 00 for 124 bytes}

3. Play ShowID 19, loop once:
  - ShowID = 19 (hex value = 13)
  - Command = 8 (set master fader value)
  - Master Fader value = 128 (hex value = 80)

Message will be sent like this:

13 00 00 00 08 00 00 00 80 00 00 00 {padded with 00 for 124 bytes}

4. Send STOP ALL command
  - ShowID = 0 (not needed)
  - Command = 12 (hex value = 0C)
  - ShowData.Loop = 0 (not needed)

Message will be sent like this:

00 00 00 00 0C 00 00 00 00 00 00 00 {padded with 00 for 124 bytes}

First two examples can be seen in [this Wireshark capture](#), where:

The Requester (PC) is IP address 10.10.10.77 and the E-Streamer Mk2 is responding from IP address 10.10.100.2

# enttec.com

MELBOURNE AUS / LONDON UK / RALEIGH-DURHAM USA