



eOS<sup>®</sup>  
*family*

# Show Control

## Setup Guide

Rev B

This product is intended for professional use only.  
Read this entire document before using this product.

Copyright © 2015 Electronic Theatre Controls, Inc.  
All rights reserved. Product information and specifications subject to change.

Part Number: **4250M2410** Rev: Rev B

Released: 2015-07

ETC<sup>®</sup>, Eos<sup>®</sup>, and Unison<sup>®</sup> are either registered trademarks or trademarks of Electronic Theatre Controls, Inc. in the United States and other countries.

All other trademarks, both marked and not marked, are the property of their respective owners.

ETC intends this document, whether printed or electronic, to be provided in its entirety.

# Table of Contents

---

Introduction . . . . .	1
Using this Guide . . . . .	2
About Eos Family Show Control . . . . .	3
Show Control Setup. . . . .	4
Show Control Display . . . . .	8
Time Code . . . . .	13
Real Time Clock (RTC) . . . . .	17
Analog Inputs and Relay Outputs . . . . .	19
MIDI Show Control . . . . .	21
String Interface . . . . .	23
MIDI Raw . . . . .	26
Open Sound Control (OSC) . . . . .	27





# Introduction

---

Welcome to the Eos® Family Show Control User Manual. This manual is intended to provide basic information specific to the Eos Family show control system. You may want to explore additional resources to learn more about show control.

The following resources are recommended:

- John Huntington, Show Networks and Control Systems, (Brooklyn, NY:Zircon Designs Press, 2012)
- The ETC Support Articles: <http://www.etconnect.com/Support/>

## Help from ETC Technical Services

If you are having difficulties, your most convenient resources are the references given in this user guide. To search more widely, try the ETC Web site at [www.etconnect.com](http://www.etconnect.com). If none of these resources are sufficient, contact ETC Technical Services directly at one of the offices identified below. Emergency service is available from all ETC offices outside of normal business hours.

When calling for help, please have the following information handy:

- Console model and serial number (located on rear or right side panel)
- Dimmer manufacturer and installation type
- Other components in your system (Unison®, other consoles, etc.)

### Americas

Electronic Theatre Controls Inc.  
Technical Services Department  
3031 Pleasant View Road  
Middleton, WI 53562  
800-775-4382 (USA, toll-free)  
+1-608 831-4116  
[service@etconnect.com](mailto:service@etconnect.com)

### United Kingdom

Electronic Theatre Controls Ltd.  
Technical Services Department  
26-28 Victoria Industrial Estate  
Victoria Road,  
London W3 6UU England  
+44 (0)20 8896 1000  
[service@etceurope.com](mailto:service@etceurope.com)

### Asia

Electronic Theatre Controls Asia, Ltd.  
Technical Services Department  
Room 1801, 18/F  
Tower 1, Phase 1 Enterprise Square  
9 Sheung Yuet Road  
Kowloon Bay, Kowloon, Hong Kong  
+852 2799 1220  
[service@etcasia.com](mailto:service@etcasia.com)

### Germany

Electronic Theatre Controls GmbH  
Technical Services Department  
Ohmstrasse 3  
83607 Holzkirchen, Germany  
+49 (80 24) 47 00-0  
[techserv-hoki@etconnect.com](mailto:techserv-hoki@etconnect.com)

Please email comments about this manual to: [TechComm@etconnect.com](mailto:TechComm@etconnect.com)

# Using this Guide

---

In order to be specific about where features and commands are found, the following naming and text conventions will be used:

- Facepanel buttons are indicated in bold [brackets]. For example, **[Live]** or **[Enter]**. Optional keys are indicated in **<angle brackets>**, for example, **<Cue>** or **<Sub>**.
- Browser menus, menu items, and commands you must perform are indicated in bold text. For example: In the **File** menu, click **Open**. Or: Press **[Record] [Enter]**.
- Alphanumeric keyboard buttons are indicated in all CAPS. For example, TAB or CTRL.
- Keys which are intended to be pressed or held simultaneously are indicated with the “and” symbol. For example, **[Shift] & [+]**.
- Softkeys and clickable buttons in the Central Information Area (CIA) are indicated in bold **{braces}**. A note about **<More SK>** (more softkeys): this command is always indicated as optional, and is only indicated once in an instruction regardless of how many pages of softkeys exist. This is because there is no way to predict what softkey page you are on at any given time. Press **<More Softkeys>** until you find the required command.
- References to other parts of the manual are indicated in *italics*. When viewing this manual electronically, click on the reference to jump to that section of the manual.



**Note:** Notes are helpful hints and information that is supplemental to the main text.

---



**CAUTION:** A Caution statement indicates situations where there may be undefined or unwanted consequences of an action, potential for data loss or an equipment problem.

---



**WARNING:** *A Warning statement indicates situations where damage may occur, people may be harmed, or there are serious or dangerous consequences of an action.*

---

# About Eos Family Show Control

---

The Eos Family show control system is made up of Time Code (MIDI or SMPTE), Real Time Clock (RTC), analog inputs, relay outputs, MIDI Show Control, Serial (RS-232), Network Serial (Network UDP), and MIDI Raw (MIDI Notes).

Available Protocols:

- **SMPTE** - An analog signal indicating time that is used to sync events between multiple devices. Eos Family consoles can only receive SMPTE from a Net3 Show Control Gateway.
- **MIDI Time Code (MTC)** - A digital version of SMPTE. MTC can only be received through a local MIDI In port or through a Net3™ Show Control Gateway.
- **Real Time Clock (RTC)** - This uses the internal astronomical time clock functions built into Eos Family consoles to trigger events.
- **Analog** - These can either be 0-10v inputs that are input into the console using the I/O Gateway, or contact closures that are input into the console using either the onboard closures on Eos Ti™, Gio™, Ion®, or Element™ consoles, Eos Programming Wing, or the I/O Gateway.
- **Relays** - Normally closed and normally open contact closure outputs (SPDT) under the console's control. One relay is built in to Eos Ti, Gio, Ion, Element consoles and Eos Programming Wing. Multiple relays are available on Net3 I/O Gateways.
- **MIDI Show Control** - A subset of the MIDI protocol, which is most commonly known as MSC, these are System Exclusive (SysEx) messages that can be sent and received from Eos Family consoles to trigger actions. MIDI In/Out ports are available on Eos Ti, Gio, Ion, Element consoles, Eos Programming Wing, and through a Net3 Show Control Gateway.
- **String Interface (Serial (RS-232)/Network Serial)**- ASCII text strings that are sent or received using the RS-232 protocol or sent over the network via a UDP message.
- **MIDI Raw** - Also known as MIDI Strings, MIDI Notes, or Channelized Event Data, this is an output only from the console. MIDI Notes are most commonly used for communication between musical instruments.
- **Open Sound Control (OSC)** - Protocol using network communication (wired or wireless) that is used for communication between varying audio, video and lighting devices.

MTC, SMPTE, Analog Inputs, and RTC require the use of the show control display, which is accessed by pressing **{Show Control}** on the Display Management Home Screen. See [Show Control Display, page 9](#) for more information.

MIDI Show Control, MIDI Raw, Relays, Serial, OSC and Network Serial (Network UDP) require minimal setup from your console, and don't require the use of the show control display.

# Show Control Setup

---

In **[Setup]>[Show]>[Show Control]**, you can adjust settings for MIDI Show Control, time code (MIDI or SMPTE), analog, and serial functions.

## Setup Options

### ***{SMPTE Time Code Rx}***

This touchbutton is used to control whether your console can receive SMPTE time code. Choosing “Disabled” will disable all time code lists that have a SMPTE source. The default for this setting is “Enabled”.

### ***{MIDI Time Code Rx}***

As above, but for MIDI Time Code instead of SMPTE. The default for this setting is “Enabled”.

### ***{Resync Frames}***

This touchbutton allows you to configure how many frames need to be synced before time code starts running. Frames can be from 1-30. Default is 2 frames.

### ***{MSC Receive}***

This touchbutton toggles the setting for receiving MIDI Show Control from an external source between “Enabled” and “Disabled”. The default setting is “Disabled”.

### ***{MSC Receive Channel}***

Also known as a “Device ID” this setting configures the MIDI channel for the console to receive MIDI Show Control information. Only MSC data with the same device ID will be received. A device ID can be from 0-126, or if set to 127, Eos Ti, Gio, Ion, or Element will receive MSC data from all IDs (All Call). Eos cannot be set as an All Call receiver. For example:

- **{MSC Receive Channel} [5] [0] [Enter]**

Multiple channels can be assigned.

- **{MSC Receive Channel} [1] [+] [4] [Enter]**

### ***{ACN MIDI Rx ID(s)}***

This setting allows you to specify the MIDI Show Control data that your console will receive when transmitted over the Architecture for Control Network (ACN). When set, the console will respond to MSC data from any gateway that has its “ACN MIDI Rx ID” set to the same number. ACN MIDI Rx ID can be from 1-32. ID ranges may be used. When using the local MIDI ports on Eos Ti, Gio, Ion, or Element, this setting needs to match the MIDI Rx Group ID in the **ECU>[Settings]>[Local I/O]>Show Control Gateway>Group IDs**. For example:

- **{ACN MIDI Rx ID} [2] [5] [Enter]**
- **{ACN MIDI Rx ID} [1] [Thru] [1][0] [Enter]**

### ***{MSC Transmit}***

This setting, when enabled, allows the console to send MSC messages for actions taken on the console, such as cue actions, macros firing, and submaster bumps. The default setting is “Disabled”.

### ***{MSC Transmit Channel}***

Also know as “Device ID” this setting allows you to establish the device ID with which your console will transmit MIDI Show Control information. A device ID can be from 0-126, or if set to 127, Eos, Eos Ti, Gio, Ion, or Element will transmit MSC data to all IDs (All Call). For example:

- **{MSC Transmit Channel} [5] [0] [Enter]**



### **{ACN MIDI Tx ID}**

This setting allows you to specify the ID number of MIDI Show Control data that the console transmits over an Architecture for Control Network (ACN). When set, the console will transmit MSC data to any gateway that has its "ACN MIDI Tx ID" set to the same number. ACN MIDI Tx ID can be from 1-32. When using the local MIDI ports on Eos Ti, Gio, Ion, or Element, this setting needs to match the MIDI Tx Group ID in the **ECU>{Settings}>{Local I/O}>Show Control Gateway>Group IDs**. For example:

- **{ACN MIDI Tx ID} [2][5] [Enter]**

### **{String MIDI TX}**

When enabled, this setting will cause the console to send serial strings when certain actions happen at the console. See "[Sending from User Events](#)" on page 45.

### **{MIDI Cue List}**

MIDI Cue List specifies the cue list that the console will use to send MSC data. If left blank and MSC Tx is enabled, all cue lists will generate MSC events. Otherwise, only the particular list (or lists) selected will fire MSC events.

### **{Analog Inputs}**

This is a master setting for receiving analog inputs from a Net3 I/O Gateway or the local analog ports on a Eos Ti, Gio, Ion, or Element. When disabled it will turn off all analog input for all event lists. Default is "Enabled".

### **{Relay Outputs}**

This is a master setting for triggering external relays from a Net3 I/O Gateway or the local ports on a Eos Ti, Gio, Ion, or Element. When disabled it will turn off all external relays for all event lists. The default is "Enabled".

### **{OSC TX IP Address}**

This sets the destination IP address or addresses to which the console will send OSC strings.



#### **Note:**

Be careful when using a network with a DHCP server. If your external device reboots or is issued a new IP address from a DHCP server, it will no longer receive OSC strings from Eos until you change this setting to match your new IP address at your external device.

### **{OSC TX Port Address}**

This sets the UDP destination port to which the console will send OSC strings.

### **{OSC RX Port Number}**

This setting specifies the UDP port that the console will listen to for OSC receiving strings.



#### **Note:**

ETC recommends using 8000 and 8001 respectively for port numbers. Remember that when setting port numbers on your external device that they should be set to the opposite of what Eos is set.

For example, if **{OSC TX Port Number}** on Eos is set to 8000, then the RX (incoming) port on your external device needs to be set to 8000, and vice versa.

### **{String RX}**

This setting will enable receiving strings on all Serial RX formats and OSC inputs.

### **{String RX Group IDs}**

This button is for setting up which Serial Port Group ID (from I/O Gateway settings in GCE) the console will listen to. This only affects serial traffic from I/O gateways, not network UDP messages or ACN strings. Group IDs are from 1-32. Remember group IDs relate to the number set in the gateway. It can be set to listen to multiple group IDs by using **[Thru]** and **[+]**.

### {String RX Port}

This setting specifies the UDP port that the console will listen to for receiving strings.

### {String TX}

This settings will enable sending strings on all Serial TX and OSC formats.

### {String TX Group IDs}

This button is for setting up which Serial Port Group ID (from I/O Gateway settings in GCE) the console will send to. This only affects serial traffic to I/O gateways, not network UDP messages or ACN strings. Group IDs are from 1-32. Remember group IDs relate to the number set in the gateway. It can be set to send to multiple group IDs by using **[Thru]** and **[+]**.

### {String TX Port}

Setting for the UDP destination port that the console will send strings.

### {String TX IP Address}

Sets the destination IP address or addresses that the console will send strings.

## Port Setup Options

Eos Ti, Gio, Ion, Element, and Eos Programming Wing have additional ports and connectors located on the rear panel for show control. Additional setup options for these ports and connectors are in the **ECU>{Settings}>{Local I/O}** and in **ECU>{Settings}>{Network}**.



### **Note:**

The local ports on Eos Ti, Gio, Ion, Element consoles, and Eos Programming Wing will not broadcast information over the network and are only used in a local console system.

### Group IDs

Allows you to set the group number for the MIDI In and MIDI Out ports. Group numbers can be from 1-32. Default for both ports is 1.

### Contact Closure (remote trigger connector)

Allows you to set the Group and Address In values for the remote trigger input on the back of your console. The contact closure only reports on/off even though it is considered an analog input.



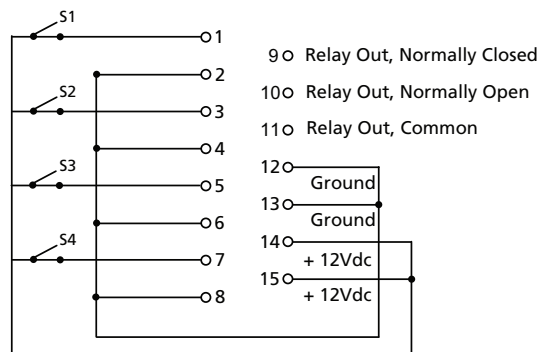
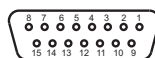
### **Note:**

The contact closures require DC voltage, which is supplied by the console on pins 14+15. See the diagram below.

### Relay Out (remote trigger connector)

Allows you to set the Group and Address In values for the remote trigger port on the back of your console.

### Remote Trigger connector pinout



## Interface Protocols

Additional settings are found in the **ECU>Settings>Network**.

Interface Protocols	Port 2	Port 1	
MultiConsole	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Standard Network    Fast File Transfer
Sensor/FDX3000 Feedback	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
RDM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	(requires Net3 gateway 5.1 or higher)
FDX2000 Feedback	<input type="checkbox"/>	<input type="checkbox"/>	Directed Broadcast
WiFi Remote	<input type="checkbox"/>	<input type="checkbox"/>	
UDP Strings & OSC	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
OSC			TCP format for OSC 1.0 (packet length headers)
			TCP format for OSC 1.0 (packet length headers)
			TCP format for OSC 1.1 (SLIP)

See [Open Sound Control \(OSC\)](#), page 27 for more information.

# Show Control Display

---

The show control display allows for the creation of event lists, which are comprised of events. Each event list uses a different type of show control (time code, analog inputs or RTC) and each event contains the information for when and what is being triggered.

The show control display can be accessed by pressing **{Show Control}** on the Display Management Home Screen.

## Settings

In addition to the show control display, your console provides additional user-definable settings for show control which can be accessed by pressing **[Displays]>{Setup}>{Show}>{Show Control}** or **[Browser]>{Setup}>{Show}>{Show Control}**. For more information, see the [Show Control Setup, page 5](#).

## Events

The upper portion of the show control display contains the specific events that are contained in each event list. An event consists of a time or address, and an action. An event plays back (or “fires”) when the clock reaches or passes the specified event time, or an input is seen on an address. Multiple events can be fired at the same time; however a single action can only be placed in each event.

Eos Family consoles support three event actions:

- Cues can be run.
- Submasters can be bumped, faded (via analog only), and turned on and off.
- Macros execute without interaction with the command line, unless the macro is set to foreground mode.



**Note:**

If another macro is fired before the first macro completes, the first macro will finish its action before the second is started.



**Note:**

If a macro is selected as the intended action, it will fire using the macro mode specified. Please see the Storing and Using Macros chapter of your console’s manual for information on macro modes.

---

The screenshot shows the 'Show Control' interface for 'ETC Demo File' at 3:36:07 PM. It features two main tables. The top table is an event list with columns for Event, Time/Address, Date, Action, and Label. The bottom table is a cue list with columns for List, Label, Type, Source, Internal, External, First Time, Last Time, FPS, and Status.

Event	Time/Address	Date	Action	Label
1	10:00:00:00		Cue 91 / 1	
2	10:00:05:03		Cue 91 / 2	
3	10:00:11:06		Cue 91 / 3	
4	10:00:23:07		Cue 91 / 4	
5	10:00:32:20		Cue 91 / 5	
6	10:00:34:20		Cue 91 / 6	
7	10:00:39:10		Cue 91 / 7	
8	10:01:11:27		Cue 91 / 8	

List	Label	Type	Source	Internal	External	First Time	Last Time	FPS	Status
1	L91	SMPTE	1	Off	On	10:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
2	L106	SMPTE	1	Off	On	01:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
3	L111	SMPTE	1	Off	On	04:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
4	L113	SMPTE	1	Off	On	05:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
5	L202	SMPTE	1	Off	On	06:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
6	L207	SMPTE	1	Off	On	02:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
7	L209	SMPTE	1	Off	On	00:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
8	L306	SMPTE	1	Off	On	03:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
9	L311	SMPTE	1	Off	On	08:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...

- Event - is the event number, used to select or create a specific event from the command line. Can also use **[Next]/[Last]** to select/navigate through the events.



**Note:**

The event number is not fixed, it is just a handle to select an event. For example, in time code lists, the events are always chronological. The event number will change to match the chronology.

- Time/Address - used to specify when the event should trigger the action. This could be either time code, time of day, or an address used for analog inputs.
- Date- used to specify when the event should trigger the action, used only for RTC. *See "Real Time Clock (RTC)" on page 37.*
- Action - is what the event is going to trigger; can be a cue, submaster, or macro.
- Label - names a specific event.

**Event List**

The lower portion of the show control display contains the event lists used in the console. Events are contained within a specific event list (like cues within a cue list). Each event list has a specific show control protocol type that it uses: MIDI, SMPTE, Analog, or RTC. Getting started with an event list is the same regardless of which show control protocol you are using.

Softkeys will be available for the various event list options.

The screenshot shows the 'Show Control' interface for 'ETC Demo File' at 3:36:07 PM. It contains two main tables.

Event	Time/Address	Date	Action	Label
1	10:00:00:00		Cue 91 / 1	
2	10:00:05:03		Cue 91 / 2	
3	10:00:11:06		Cue 91 / 3	
4	10:00:23:07		Cue 91 / 4	
5	10:00:32:20		Cue 91 / 5	
6	10:00:34:20		Cue 91 / 6	
7	10:00:39:10		Cue 91 / 7	
8	10:01:11:27		Cue 91 / 8	

List	Label	Type	Source	Internal	External	First Time	Last Time	FPS	Status
1	L91	SMPTE	1	Off	On	10:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
2	L106	SMPTE	1	Off	On	01:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
3	L111	SMPTE	1	Off	On	04:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
4	L113	SMPTE	1	Off	On	05:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
5	L202	SMPTE	1	Off	On	06:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
6	L207	SMPTE	1	Off	On	02:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
7	L209	SMPTE	1	Off	On	00:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
8	L306	SMPTE	1	Off	On	03:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...
9	L311	SMPTE	1	Off	On	08:00:00:...	23:59:59:...	30	00:00:00:00 Clock Dis...

At the bottom, there is a status bar showing 'BLIND: Event 1 : Event' and a navigation bar with '1. Live Table' and '3. Show Control'.

- List - is the event list number.
  - <Event> [1][/][Enter] - creates event list 1.
- Label - names the event list.
- Type - sets which type of show control protocol to use; MTC, SMPTE, Analog or RTC. Only one protocol can be used at a time per list.
- Source - should match the Group ID used in the connected Net3 Show Control, I/O Gateway, or the local ports on Eos Ti, Eos RPU 3, Gio, Ion, or Element.
- Internal - enables the list to use internal timing sources. *See "Internal vs. External Time".*
- External - enables the list to use external timing sources.
- First Time - sets the first time that the console will use when using internal timing.
- Last Time - sets the last time that the console will use when using internal timing.
- FPS - sets how many frames per second (FPS) will be used; 30, 25, 24.
- Status - current status of the event list.

### Internal vs. External Time

The Eos Family show control system makes use of two timing sources, internal and external. The different protocols use internal and external clocks in different ways.

Time code (MIDI or SMPTE) normally requires timing information from an input source, like a show control gateway. This timing source is referred to as external time, since it is coming from a source external to your console.

If the external time source is not available for any reason, each time code event list within your console has an internal timing source which will assume control, if enabled. This internal timing source requires three pieces of information to determine how to generate its timing: first time, last time, and frame rate. When the internal clock reaches the last time, it will reset to the first time and continue running.



---

**Note:** External timing sources may run faster or slower than the console's internal clock.

---

Analog inputs uses external timing sources only. This is an enable or disable function for inputs.

Real Time Clock uses internal time only, but the clock can sync with an external time server. Whether using internal time from the console or using external time from a time server, it is important that all the correct information for time zone, latitude, and longitude are properly setup. Refer to the Configuration Utility appendix in your console's manual for setup information.

## Creating an Event List

Getting started with an event list is the same regardless of which show control protocol you are using.

To program an event list, you first must open the show control display.

- ***[Displays]><More SK>>[Show Control]***

Create an event list by specifying it in the command line.

- ***<Event> [1] [/] [Enter]***

Specify what the list will respond to (MIDI, SMPTE, Analog, or RTC).

- ***<Event> [1] [/] {Type} {MIDI} [Enter]***

Define the group or source.

- ***<Event> [1] [/] {Source} [8] [Enter]***

Activate the internal and/or external timing functions (see [Internal vs. External Time, page 10](#)). These functions are toggle on/off states.

- ***<Event> [1] [/] {Internal} {External} [Enter]***

When defining an event list, it is feasible to enter multiple commands in the same command line. This can speed up your programming of an event list. For example:

- ***<Event> [2] [/] {Type} {MIDI} {Source} [5] [Enter]***

Once the list is created, you will then need to create the specific events. Those steps will vary depending on what protocol is being used.





# Time Code

Eos Family consoles have time code functionality available to execute a list of events that will play back following incoming internal or external timing values.

Time code lists can receive timing data from either SMPTE or MIDI sources. Eos Family consoles accept up to 32 SMPTE sources and 32 MIDI Time Code (MTC) sources (each are numbered from 1-32). Up to 64 sources can be received simultaneously.

## Enabling Time Code

Eos Family consoles have global settings to enable or disable reception of MIDI Time Code and SMPTE. These are found in setup, [See "Show Control Setup" on page 5.](#)

If the MIDI Time Code setting is disabled, all MTC event lists are disabled as well. Each event list can be enabled/disabled individually using the internal and/or external field. Off is disabled, regardless of the global setting in the setup screen. The same is true for SMPTE.

## Timing Data

Timing data is given in the form of (hours) : (minutes) : (seconds) : (frames). Therefore a timing value of 06:25:15:24 would be 6 hours, 25 minutes, 15 seconds, 24 frames.

The number of frames per second is determined by the timing source and the event list must be set to this same number of frames when programming. Number of frames can be 24, 25, or 30.

### Color coding

The color of the time in the "Status" column of the event list display indicates the source of the timing data. The colors and indications are:

- Green - Valid external timing is being received.
- Red - The internal clock is the data source and is running.

You will also see an indicator in the upper portion of the main display if valid internal or external time is seen.

The screenshot shows the 'Show Control' interface. At the top, there is a 'Show Control' header with a red time display '00:00:38:23' circled in yellow. Below this is a table with columns: Event, Time/Address, Date, Action, and Label. The first row shows event '1' with a red time display in the Status column. Below this is a detailed table of event lists:

List	Label	Type	Source	Internal	External	First Time	Last Time	FPS	Status
1	MIDI Timecode	MIDI	1	On	Off	00:00:00:...	23:59:59:...	30	00:00:38:23
2	SMPTE Timecode	SMPTE	1	Off	On	00:00:00:...	23:59:59:...	30	00:00:00:00 Clo...
3	Followspot Input	ANALOG	1		On				Enabled
4	Work Lights	RTC		On					Enabled

## Timing and Clocks

The internal and external clocks can be enabled separately for each time code list. If you enable the external clock only, events will play back as long as valid timing data is received from an external source. When the timing data stops, the events will stop.

When you enable the internal clock only, it starts running immediately and triggers events appropriate to the time.

When both internal and external clocks are enabled, the internal clock does not start running until the first valid external time is received. After that, it will run whenever the external time source is absent.

### Loop times

First and last times define the overall loop. Times can be set anywhere from 00:00:00:00 to 23:59:59:29. When external time is being received, your console ignores these times. However, you should still set them to match whatever loop times the external clock is using. When using internal timing, it is important to set the first and last times. These set the start and stop or loop points of the sequence.

### Internal clock

You can set the internal clock at any time by pressing **[Time]** and entering a time. To reset the internal clock, press **[Time] [Enter]**.



**Note:** You must first select the Event List (<Event List> [1] [/]). Once selected, using the time function allows you to put the internal time clock at any position between the start and ending time. This is how the internal clock can be set back to 00:00:00:00.

## Programming a Time Code Event List

Open the show control display.

- **[Displays]><More SK>>[Show Control]**

Create a new event list.

- **<Event> [1] [/] [Enter]**

Define the type of input.

- **<Event> [1] [/] {Type} {MIDI} [Enter]**

Define the source of input (this is the source ID number of the Gateway 1-32).

- **<Event> [1] [/] {Source} {1} [Enter]**

Specify if internal and/or external timing is enabled.

- **<Event> [1] [/] {Internal} {External} [Enter]**

Define the first time and last time for the event list (times are given as 00:00:00:00).

- **<Event> [1] [/] {FirstTime} [2] [0] [1] [5] {LastTime} [2] [0] [4] [5] [1] [5] [Enter]**

List	Label	Type	Source	Internal	External	First Time	Last Time	FPS	Status
1		MIDI	1	On	On	00:00:20:15	00:20:45:15	25	00:00:00:00 Waiting for Input

Specify the number of frames per second (FPS) used by the timing source (24, 25, or 30).

- **<Event> [1] [/] {FrameRate} [2] [5] [Enter]**

When working in Live, to show the Time Code times in the Cue List, go to the Cue List Display.

- **[Cue] [Cue]**

Select the Cue List to be triggered.

- **[Cue] [1] [/]**

Now associate the Event List to the Selected Cue List.

- **[Cue] [1] [/] {Execute} {Timecode} {1} [Enter]**

Now in the cue display, for each cue, the execute column will display the time of the first event in the Time Code list that triggers that cue.

To change the time that a cue will be triggered by time code.

- **[Cue] [1] {Execute} {TimeCode} [5][4][5] [Enter]**

Changes the time of the first event that fires cue 1 to 5:45. If no event currently triggers cue 1, it will be created in the list.

- **[Cue] [1]{Execute} {TimeCode} [-][5] [Enter]**

Subtracts 5 from the current time, **[+]** would add 5 to the current time.

Both the Event List and the Cue List are ready for events to be added. There are three ways that events can be added into the Event List.



---

**Note:**

You will notice as events are added in the Event List that the time code will display on the Playback Status Display.

---



---

**Note:**

This is the same process for MIDI and SMPTE Time code.

---

## Adding Events from the Show Control Display

Events can be added directly from the show control display, however a new event number must be specified for each event action.

Open the show control display.

- **[Displays]><More SK>>{Show Control}**

Select the event list.

- **<Event> [1] [/] [Enter]**

Specify a new event number.

- **<Event> [1] [Enter]**

Add the action that the event should execute.

- **<Event> <1> [Cue] [1] [Enter]**



---

**Note:**

As there is no action softkey, pressing **[Cue]**, **[Sub]**, or **[Macro]** assumes you are posting to the action field.

---

Define the time code position of the event.

- **<Event> <1> [Time] [2][0][1][5] [Enter]**

It is possible to enter multiple commands in the same command line. This can speed up your programming. For example:

- **<Event> [2] [Cue] [1.5] [Time] [2][4][1][3] [Enter]**

Event timing can also be modified within a range. Let's say an additional 5 frames needs to be

added to multiple events.

- **<Event> [1] [Thru] [4] [Time] [+] [5] [Enter]**



**Note:**

You will notice as events are added to the event list, the time code will display on the Playback Status Display.

## Adding Events From Live

Events can be added directly from live using the same conventions used in the Show Control Display. The advantage to doing this from Live is there is no need to specify the event number, as the console automatically assigns the number.

From the Live Display

Select the cue to execute.

- **[Cue] [3]**

Now define the time code location.

- **[Cue] [3] {Execute} {Timecode} [3][2][1][5] [Enter]**



**Note:**

You will notice as events are added to the event list, the time code will display on the Playback Status Display.

## Adding Events using Learn Mode

If you specify an event list within the show control display, you can press the **[Learn]** hardkey to activate learn mode for that event list. When in learn mode, an event is created with the current list time whenever a cue is activated, a submaster bump is pressed, or a macro is run. Once in learn mode, the **[Learn]** key will remain lit to indicate that it is in learn mode. If the show control display is exited, the console remains in learn mode until deactivated.

Multiple time code lists can be in learn mode simultaneously, each with their individual times. This is accomplished by associating the event list and cue list together.

To exit learn mode, the console must be in the show control display. Once there, select the event list then press **[Learn]** again. If **[Learn]** is pressed while viewing any other screen than the show control display an empty event will be placed on all event lists that are in learn mode.

Open the show control display.

- **[Displays]><More SK>>{Show Control}**

Select the event list to enable learning on.

- **<Event> [1] [N]**

Press the **[Learn]** key.

- **<Event> [1] [N] [Learn]**

In the Status column it now says Learning, and the **[Learn]** button is illuminated..

List	Label	Type	Source	Internal	External	First Time	Last Time	FPS	Status
1		SMPTE	1	Off	Off	00:00:00:...	23:59:59:...	30	00:00:00:00 Clock Disabled Learning

## Execution While Editing

If the internal or external clock is running, events will fire, even if you are in the edit screen. As soon as an event is created, the event list is resorted, and the new event is eligible for playback.

# Real Time Clock (RTC)

---

Eos Family consoles have the ability to run Real Time Clock (RTC) events. RTC events are used to run a cue, submaster, or macro at a specific time on specific days. RTC events can run at a certain time of day, like 5:00pm on Tuesdays, a certain date, or at a time based off of astronomical events, such as sunrise and sunset.

## Enabling Real Time Clock

For RTC events to work properly, time zone, latitude, and longitude must be set up correctly. Those settings are found under the General tab in the Eos Configuration Utility (ECU).

Also, the option for internal must be set to on within the show control display. As long as the internal option is set to on, the RTC will execute. However if internal is set to off, then the events will not trigger. The status in the show control list will indicate whether it is on or off. There is no way to disable a single event from triggering within the event list, you can only remove the event number.

## Real Time Clock Events

RTC events are created using the show control display. RTC events consist of a time field, a day or date field, and an action. Time for RTC events can be specified as local time or as time relative to the astronomical events, sunrise and sunset. Local time is displayed in the 24 hour format. For example, if you want an event to run at 2:50pm, you would need to set it for 14:50.

## Programming Real Time Clock Events

Open the show control display.

- ***[Displays]><More SK>>[Show Control]***

Create a new event list.

- ***<Event> [1] [/]***

Specify the event list as RTC.

- ***<Event> [1] [/] {Type} {RTC} [Enter]***

Enable the RTC.

- ***<Event> [1] [/] {Internal} [Enter]***

Define the Time and Date for each event.



### **Note:**

Commands cannot be combined on the same command line.

- ***<Event> [1] [Time] [1] [5] [0] [0] {Days} {Mon} {Wed} {Fri} [Enter]*** - sets the time for 3:00pm.
- ***<Event> [1] {Days} {Mon} {Wed} {Fri} [Enter]*** - adds on Mondays, Wednesdays, and Fridays.



### **Note:**

Days of the week can either be entered in from their softkeys, or you can enter them in from the keypad using the conventional modifiers (+, -, thru). Monday is 1.

Define the Astronomical Time and Date.

- ***<Event> [2] [Time] {Before Sunset} [3] [0] [Enter]*** - sets the time for 30 minutes before sunset.

- **<Event> [2] {Days} [1] [+] [2] [+] [3] [+] [6] [Enter]** - adds on Monday, Tuesday, Wednesday, and Saturday.

Define the Astronomical Time and Date.

- **<Event> [3] [Time] {After Sunrise} [6] [0] [Enter]** - sets the time for an hour after sunrise.



**Note:**

The events will renumber themselves to be in the correct order of execution throughout the day. Event 3 is now listed as Event 1. The command line will change to Event 1 automatically.

- **<Event> [1] {Date} [3] [0] {Month} [1] [0] {Year} [2] [0] [1] [1] [Enter]** - sets to on October 30, 2011.

Creating the action for each event is the same as time code. Select the event and specify either cue, submaster, or macro.

- **<Event> [1] [Cue] [1][2] [Enter]**



**Note:**

As there is no action softkey, pressing **[Cue]**, **[Sub]**, or **[Macro]** assumes you are posting to the action field.

The screenshot shows the 'Show Control' interface for an RTC system. The top bar displays 'Show Control', 'RTC', and the time '3:51:14 PM'. Below this are two tables.

Event	Time/Address	Date	Action	Label
1	07:30	Mon Tue Wed Thu Fri Sat Sun	Macro 1	Lobby On
2	23:00	Mon Tue Wed Thu Fri Sat Sun	Macro 2	Lobby Off

List	Label	Type	Source	Internal	External	First Time	Last Time	FPS	Status
1	Lobby Lights	RTC		On					Enabled

A yellow arrow points to the 'On' value in the 'Internal' column of the second table, with the text: "The internal option must be set to on."

At the bottom, a status bar shows "BLIND: Event 1 : 1. Live Table 3. Show Control".

# Analog Inputs and Relay Outputs

---

## Analog Inputs

Eos Family consoles can accept analog input (0-10v or contact closure) from a Net3 I/O Gateway, or only contact closures from the local ports on Eos Ti, Gio, Ion, Element, or Eos Programming Wing. For setting up the I/O gateway, please refer to the Net3 I/O Gateway Setup Guide. For setting up your console's local ports, see [Port Setup Options, page 7](#). Analog inputs are programmed through the show control display, where each analog event in the event list consists of a port address and an action. Each event list source references a specific Group ID, which needs to match the ACN Group ID of the I/O Gateway or the Group ID of the local remote trigger port, and the events communicate to the individual addresses or inputs within the gateway.

To enable the analog event list, the internal time option must be enabled. This is an enable/disable function for the entire event list. There is also a global enable for all analog inputs found in Setup.

### Analog Input Events

Analog input events can be broken down into two categories; events triggered by a contact closure and those triggered by 0-10v dimming.

Contact closures will fire an event when input passes a certain level. Events that can be triggered this way include running a cue, controlling a submaster, and firing a macro.

0-10v dimming can be used to trigger a fader for a submaster.

### Create an Analog Event List

Open the show control display.

- ***[Displays]><More SK>>{Show Control}***

Create a new event list.

- ***<Event> [4] [/] [Enter]***

Define the type of input.

- ***<Event> [4] [/] {Type} {Analog} [Enter]***

Define the source of input.

- ***<Event> [4] [/] {Source} {1} [Enter]***

Remember this is the Source ID# of the Gateway.

Specify the event list to be active.

- ***<Event> [1] [/] {Internal} [Enter]***

Create an analog input event where an input on address 1 will fire cue 10.

- ***<Event> [4] [/] [1] {Address} [1] [Cue] [1] [0] [Enter]***

Remember, if you select the submaster option, you will have an additional option of fader, which will map the 0-10v input to the level of a specified submaster.



### **Note:**

As there is no action softkey, pressing **[Cue]**, **[Sub]**, or **[Macro]** assumes you are posting to the action field.

---

## Relay Outputs

Eos Family consoles can trigger relay outputs on Net3 I/O Gateways and on the local relay out on Eos Ti, Gio, Ion, Element consoles, and Eos Programming Wing. Relay outputs are also often called remote triggers, contact closures, or contact outputs. Relay outputs are controlled by external links in cues. To

program a relay, there are two pieces of information that are very important, the ACN Group ID and the relay address. Remember, the ACN Group ID is programmed into the gateway using GCE. You can enter relay information using the following syntax:

- **[Cue] [1] {Execute} {Relay} [1] [/] [2] {On} [Enter]** - 1 is the Group ID and the 2 is the relay output address.



# MIDI Show Control

---

## Data Sources

Eos Family consoles can receive MIDI Show Control (MSC) data from a MIDI input source. Your console can receive data from any of the available 32 sources, which have Group IDs of 1-32. While MIDI consists of many types of data, your console will respond to only MSC data. Other MIDI transmitted data is ignored by the console.

Eos Family consoles can also transmit MIDI Show Control data.

## MIDI Show Control Devices

MIDI Show Control data carries a device ID (MIDI channel) within the data packet. MSC setup, see [Show Control Setup, page 4](#), allows for the specification of two device IDs: one for reception and one for transmission. This is different than the ACN Group Tx and Rx IDs, which are the show control gateway source ID, and transmitting ID.

Only MSC data that matches the MSC Receive Channel in Setup, or an All Call ID (127), will be interpreted by your console.

All outgoing MSC data contains the specified transmit device ID, as specified in Setup, see [Show Control Setup, page 5](#).

## Eos Family Command Interpretation

When your console receives a command from a MIDI source, it will look for and support the following MIDI commands in the following ways:

- Go           Runs a cue
- Stop         Pauses a cue
- Resume      Resumes a paused cue
- Set          Controls a submaster, playback, or Grandmaster
- Fire         Runs a macro

The cue-related commands (go, stop, resume) affect the console based on the cue field also contained within the MSC data. The cue field will correspond directly to the console's cues. In the command/effect tables below, if a field is present in the MSC data, the name is indicated (such as "Cue"). If it is not present in the data, a "-" is indicated.

### Go

When accompanying a MIDI "Go" command, data for the following fields will result in the following actions:

<b>Cue Data Field</b>	<b>List Data Field</b>	<b>Action</b>
Cue	List	Runs the specified cue in the specified cue list
Cue	-	Runs the specified cue in the default cue list
-	List	Runs the next cue in the specified list
-	-	Runs the next cue in the default cue list

## Stop

When accompanying a MIDI "Stop" command, data for the following fields will result in the following actions:

<b>Cue Data Field</b>	<b>List Data Field</b>	<b>Action</b>
Cue	List	Stops the specified cue & list (if running)
Cue	-	Stops the running cue from the default cue list
-	List	Stops the current cue from the specified list
-	-	Stops all cues

## Resume

When accompanying a MIDI "Resume" command, data for the following fields will result in the following actions:

<b>Cue Data Field</b>	<b>List Data Field</b>	<b>Action</b>
Cue	List	Resumes the specified cue & list (if stopped)
Cue	-	Resumes the specified cue in the default cue list
-	List	Resumes the current cue in the specified list
-	-	Resumes all stopped cues

## Set

The "Set" command allows MSC to control potentiometers using both a numeric control number (which specifies the potentiometer) and a data field (which controls the level: 0-100). Control values are as follows

<b>Control Value</b>	<b>Data Field</b>	<b>Indication</b>
1-127	0-100	Submaster 1-127
128	0-100	Primary playback in
129	0-100	Primary playback out
510	0-100	Grandmaster

Accompanying any of these indications in the MSC data, a numeric value of 0-100 would set the specified potentiometer to that level.

All other control values are ignored.



---

**Note:** Levels transmitted must be followed with 00 (in hex) in order to be interpreted.

---

## Fire

This command fires a macro. MSC supports macros 1 through 127 only. Therefore a numeric value of 1 - 127 followed by 00 (in hex) would accompany this command.

# String Interface

---

The serial interface allows RS-232, UDP serial, ACN, and OSC string commands to be sent to the console.

## Enabling Serial

Receiving serial commands can be enabled in Setup, see [Show Control Setup, page 5](#). Serial is disabled by default. If using RS-232 from an I/O gateway, you need to specify the ACN Serial Group ID.

## Serial Commands

All text will be queued up until either a carriage return (hex 0D), a “\r”, or a “#”, is encountered to show the end of the command. The carriage return (hex 0D), a “\r”, or a “#” will act as the [Enter] key for the command line. If the text begins with a “\$”, the text that follows until the carriage return (hex 0D), a “\r”, or a “#” will be sent to the command line for user 0 and processed as if that user had typed it. Commands can be directed to a specific user command line by adding <UX> at the beginning of the string. X= the user number.

ACN strings use the same syntax as RS-232 and UDP strings.

### For Example:

- **<U2> \$ Chan 1 Thru 10 At Full /r – sets channels 1 through 10 to full manually.**

All other text that doesn't start with a “\$” will be sent to the console's event handler. Here are a few examples:

- **Go 1** – fire playback 1
- **Cue 1 2** – run cue 2 from list 1, on the appropriate fader
- **Cue 1** – fire pending cue from list 2
- **GoCue0 1** – clear playback 1 (fires cue 0)
- **Release 1**– release playback 1
- **Off 1** – turn playback 1 off
- **Resume 1** – resume playback 1
- **Assert 1** – assert playback 1
- **Stop 1** – stop playback 1
- **Stop Cue 1 2** – stop cue 2 from list 1 if it is running
- **Stop** – stops all
- **Resume Cue 1 2** –resume cue 2 from list 1 if it is stopped
- **Resume** – resumes all
- **SubAssert 1** – assert submaster 1
- **SubUnload 1**– unload sub 1
- **SubDown 5** - presses sub 5's bump button down
- **SubUp 5** – releases sub 5's bump button
- **SubMove 1 50** - sets sub 1 to 50
- **PlaybackMove 1 50** - sets playback 1 to 50 percent
- **Grandmaster 1 100** - set Grandmaster 1 to full (there's currently only 1 Grandmaster)
- **Macro 1** - fires Macro 1



---

**Note:** Text requires either a carriage return (hex 0d), “\r”, or “#” to terminate the command line.

---

## Receiving via the I/O Gateway

For your console to receive serial data from an I/O Gateway, the Serial Port Group ID must match the String RX Group ID, and String RX must be Enabled. See “[Show Control Setup](#)” on page 5.

## Network Serial (UDP)

Eos Family consoles have the ability to send strings, which can be used to send commands to other devices, such as Paradigm®, Crestron®, and other media servers. The ability to send and receive strings can be done via RS232 ports (if equipped), ACN String EPI, and UDP messages. The same strings for Serial RS-232 can be used for Networked Serial as well.

### Configuration

The following settings for string I/O are available in **[Setup]>[Show]>[Show Control]**:

- **{String RX}** - This setting will enable receiving strings on all Serial RX formats.
- **{String TX}** - This settings will enable sending strings on all Serial TX formats.
- **{String RX Group Ids}** - This button is for setting up which serial groups the console will listen to. Group IDs are from 1-32. Remember group IDs relate to the number set in the gateway. It can be set to listen to multiple group IDs by using **[Thru]** and **[+]**.
- **{StringTX Group Ids}** - This button is for setting up which serial groups the console will send to. Group IDs are from 1-32. Remember group IDs relate to the number set in the gateway. It can be set to send to multiple group IDs by using **[Thru]** and **[+]**.
- **{String RX Port}** - Setting for the UDP port that the console will receive strings.
- **{String TX Port}** - Setting for the UDP port that the console will send strings.
- **{String TX IP Address}** - Setting the IP address the console should send string to.



---

**Note:** If you are trying to control Paradigm via UDP, ACN Strings, or RS-232, please reference the Paradigm Serial Access Protocol document for additional information.

---

### Sending Strings

There are three ways that your console can send strings: from cues, macros, or user events.

#### ***Sending from Cues***

First make sure that String TX is enabled or you will not be able to execute any commands. To send a string via a cue, it must be part of the external links. When the cue is executed, the string will be sent to all enabled string interfaces. When the **{Execute}** softkey is used, a **{String}** softkey will be displayed. When **{String}** is pressed, the alphanumeric keyboard will display and text entered will be displayed in the external links field of the playback status display.

If there is already a string linked to the selected cue, the string will be displayed for editing. When **[Cue] [n] {Execute} {String}** is on the command line, pressing **[Next]** / **[Last]** will step through all the strings used in that show file. You can then modify the displayed string to simplify the process of entering similar strings.

#### ***Sending a String to Paradigm***

Select the cue that should be executing the string.

- **[Cue] [1]**

Press the **{Execute}** softkey.

- **[Cue] [1] {Execute}**

New softkeys will appear, press the **{Strings}** softkey.

- **[Cue] [1] {Execute} {Strings}**

Define the serial string, in this case we will trigger a Macro On in Paradigm.

- **[Cue] [1] {Execute} {Strings} macro on Lights1 [Enter]**

### Sending from Macros

A **{Send String}** softkey is available in the Macro Editor display. Any text entered after the string command in the macro will be sent to all enabled string interfaces when that macro is fired.

### Sending from User Events

**{String MIDI TX}** is an option in **[Setup]>{Show}>{Show Control}**. See “*{String MIDI TX}*” on page 6. When **{String MIDI TX}** is enable, MIDI Show Control messages will be sent as serial string messages when certain actions happen at the console. Those actions are:

- A cue is fired.
- A cue is stopped.
- A cue is resumed.
- A sub is bumped up.
- A sub is bumped down.
- A macro is fired.



---

**Note:** String MIDI Tx is restricted by **{MIDI Cue List}** in Setup. See “*{MIDI Cue List}*” on page 6.

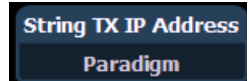
---

## Enabling ACN Strings

For the console to send or receive ACN string commands, the following settings found in **Setup>Show>Show Control** must be enabled and configured.

### Sending

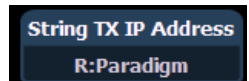
To send ACN strings, **{String TX}** must be enabled, and the ACN component name must be set in the **{String TX IP Address}** field. For example, the ACN component name could be Paradigm.



String TX IP Address  
Paradigm

### Receiving


To receive ACN strings, **{String RX}** must be enabled, and the ACN component name must be set in the **{String TX IP Address}** field with the **R:** prefix. For example, the ACN component name could be Paradigm.



String TX IP Address  
R:Paradigm

### Bidirectional Strings

To send and receive ACN strings from devices such as a Net3 Gateway, both **{String TX}** and **{String RX}** must be enabled, and the ACN component name must be set in the **{String TX IP Address}** field with the **B:** prefix. For example, the ACN component name could be Net3 4-Port Gateway.



String TX IP Address  
B:Net3 4-Port Gateway

If you have multiple devices sending or receiving strings, separate each entry in the **{String TX IP Address}** field with a comma.

# MIDI Raw

---

Also known as MIDI Strings, MIDI Notes or Channelized Event Data, this is an output only function from the console. MIDI Raw is most commonly used for communication between musical instruments. However, as certain devices do exist that use MIDI Raw, the consoles can output MIDI Raw to any device using a Show Control Gateway or local MIDI ports.

Strings are programmed either using the **{Execute}** function on a cue or using a macro.

## Programming from Cues

Once the execute button is pushed the softkeys repaint and a **{MIDI Raw}** softkey becomes available. Notes can either be input using Hexadecimal values or Decimal values depending on the connected device. The console will default to hexadecimal format, to use decimal format simply begin the MIDI string with "D".

### Sending a MIDI Raw String From a Cue

Select the cue that should be executing the string.

- **[Cue] [1]**

Press the **{Execute}** softkey.

- **[Cue] [1] {Execute}**

New softkeys will appear, press the **{MIDI Raw}** softkey.

- **[Cue] [1] {Execute} {MIDI Raw}**

Define the MIDI string, in this case we will use Hexadecimal.

- **[Cue] [1] {Execute} {MIDI Raw} 0A 00 4B 00 [Enter]**

Sending the same command as Decimal format.

Select the cue that should be executing the string.

- **[Cue] [1]**

Press the **{Execute}** softkey.

- **[Cue] [1] {Execute}**

New softkeys will appear, press the **{MIDI Raw}** softkey.

- **[Cue] [1] {Execute} {MIDI Raw}**

Define the MIDI string, in this case we will use decimal.

- **[Cue] [1] {Execute} {MIDI Raw} D 10 0 75 0 [Enter]**

## Programming from Macros

A **{MIDI Raw}** softkey is available in the macro editor display. Any text entered after the string command in the macro will be sent to all string interfaces when that macro is fired. Use the same conventions as you would for programming a string through a cue.

# Open Sound Control (OSC)

Open Sound Control (OSC) is a protocol that uses network communication (wired or wireless) to communicate between varying audio, video and lighting devices.

## Enable OSC

OSC and Serial Strings can be enabled or disabled via the Show Control settings option located within Show in Setup. To enable OSC input into the console enable *{String RX}*, page 5, likewise to enable OSC output from the console to another device enable *{String TX}*, page 6. For OSC to work over the network, the option for UDP Strings & OSC must be enabled in **ECU>Settings>Network>Interface Protocol**. Additional OSC settings are available in Setup.



**Note:** By default, the OSC user is the same as the current console user. The OSC user can be set via OSC commands. Once set the OSC user remains as specified until the user is changed.

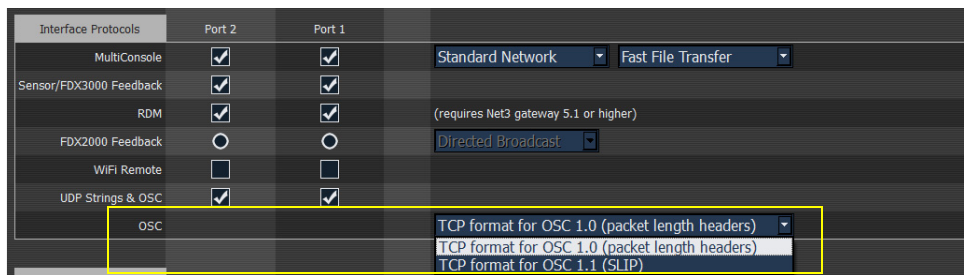
## Using OSC

OSC commands can be sent over the network or wirelessly over the network much like that of UDP. Any incoming OSC command will be sent by default to the console that is set to receive OSC. However, the OSC user can be set within the string of commands to follow any other user in the system. The basic structure of an OSC command maintaining the user number of the console communicating to should begin with `"/eos/".` If the OSC command should contain a user number different than the communicating device it should begin with `"/eos/user/<number>/"`.

OSC Method	Action
<code>/eos/chan</code>	[Chan] [X]
<code>/eos/user/&lt;number&gt;/chan</code>	[Chan] [X] (as specified user)

OSC can also be used over a TCP connection for guaranteed delivery of commands. For a remote device to send or receive OSC over TCP, the device must connect to the console on port 3032. On the console, *{String TX}*, page 6 and *{String RX}*, page 5 in Setup must be enabled.

In the **ECU>Settings>Network>Interface Protocols**, there is an option for changing the OSC TCP mode. By default, OSC 1.0 is selected.



## Supported OSC Input

All OSC commands must begin with `"/eos/..."` or `"/eos/user/<number>/..."`.



**Note:** All of the command examples given can also use the `"/eos/user/<number>/..."` variant.

## Absolute Levels

OSC Method	Arguments	Examples/Comments
/eos/at	number for the level	/eos/at=75 (useful for mapping to an OSC slider)
/eos/at/out	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/out
/eos/at/home	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/home
/eos/at/level	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/level
/eos/at/full	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/full
/eos/at/min	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/min
/eos/at/max	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/max
/eos/at/+%	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/+%
/eos/at/-%	number for the button edge 1.0=down, 0.0=up (optional)	/eos/at/-%
/eos/at/dmx	number for the DMX level	/eos/at/dmx/128 (useful for mapping to an OSC slider)
/eos/param/<parameter>	number for the level	/eos/param/pan=270 (useful for mapping to an OSC slider)
/eos/param/<parameter>/out	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/out
/eos/param/<parameter>/home	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/home
/eos/param/<parameter>/level	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/level



/eos/param/<parameter>/full	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/full
/eos/param/<parameter>/min	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/min
/eos/param/<parameter>/max	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/max
/eos/param/<parameter>/+%	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/+%
/eos/param/<parameter>/-%	number for the button edge 1.0=down, 0.0=up (optional)	/eos/param/pan/-%
/eos/param/<parameter 1>/<parameter 2>	number for all parameter levels multiple numbers for each parameter level	/eos/param/pan/tilt=45 (set pan & tilt to 45)  /eos/param/pan/tilt=45,90 (set pan to 45 & tilt to 90)

## Wheel

OSC Method	Arguments	Examples/Comments
/eos/wheel/level	number for wheel ticks for the specified wheel mode (positive or negative)	/eos/wheel/level=1.0 (increase value)  /eos/wheel/level=-1.0 (decrease value)  /eos/wheel/level=4.0 (increase value rapidly)  (defaults to Coarse mode, but wheel mode can be changed with the /eos/wheel command below)
/eos/wheel	number for wheel mode: 0=coarse, 1=fine	/eos/wheel=1.0
/eos/wheel/<parameter>	number for wheel ticks (positive or negative)	/eos/wheel/pan=1.0
/eos/wheel/<parameter 1>/<parameter 2>/...	number for wheel ticks (positive or negative)	/eos/wheel/pan/tilt=1.0
/eos/wheel/fine/<parameter>	number for fine wheel ticks (positive or negative)	/eos/wheel/fine/pan=1.0
/eos/wheel/fine/<parameter 1>/<parameter 2>/...	number for fine wheel ticks (positive or negative)	/eos/wheel/fine/pan/tilt=1.0
/eos/wheel/course/<parameter>	number for coarse wheel ticks (positive or negative)	/eos/wheel/course/pan=1.0
/eos/wheel/course/<parameter 1>/<parameter 2>/...	number for coarse wheel ticks (positive or negative)	/eos/wheel/course/pan/tilt=1.0

## Switch



**Note:**

Switch is a variant of Wheel with the added functionality of continuously repeating wheel ticks until a subsequent OSC switch command sets the wheel ticks to zero.

This may be used to continuously tick a wheel while a button is held down, for example. The expected argument range is -1.0 to 1.0, which affects the tick rate accordingly, but can be a smaller or larger range for more subtle or rapid movement.

OSC Method	Arguments	Examples/Comments
/eos/switch/level	X level wheel ticks	
/eos/switch	Set OSC wheel mode	0 = Course, 1 = Fine
/eos/switch/<parameter>	X OSC wheel mode ticks for specified parameter (ex: pan)	
/eos/switch/<parameter 1>/<parameter 2>/...	X OSC wheel mode ticks for specified parameters (ex: red, green, blue)	
/eos/switch/fine/<parameter>	X fine wheel ticks for specified parameter (ex: pan)	
/eos/switch/fine/<parameter 1>/<parameter 2>/...	X fine wheel mode ticks for specified parameters (ex: red, green, blue)	
/eos/switch/course/<parameter>	X course wheel ticks for specified parameter (ex: pan)	
/eos/switch/course/<parameter 1>/<parameter 2>/...	X course wheel mode ticks for specified parameters (ex: red, green, blue)	

## Active Parameters



**Note:**

The parameter <index> is a 1-based index referencing the list of current parameters for the selected channel(s).

Eos will send the parameter name and current value for each active parameter, so that the wheels on your OSC-enabled device may be labeled appropriately.

OSC Method	Arguments	Examples/Comments
/eos/active/wheel/<index>	number of wheel ticks for the specific wheel mode (positive or negative)	<p>/eos/active/wheel/1=1.0 (increase value)</p> <p>/eos/active/wheel/1=-1.0 (decrease value)</p> <p>/eos/active/wheel/1=4.0 (increase value rapidly)</p> <p>(defaults to Coarse mode, but wheel mode can be changed with the /eos/wheel command)</p>

<code>/eos/active/wheel/fine/&lt;index&gt;</code>	number of fine wheel ticks (positive or negative)	<code>/eos/active/wheel/fine/1=1.0</code>
<code>/eos/active/wheel/course/&lt;index&gt;</code>	number of coarse wheel ticks (positive or negative)	<code>/eos/active/wheel/course/1=1.0</code>
<code>/eos/active/switch/&lt;index&gt;</code>	number of wheel ticks for the specific switch mode (positive or negative)	<code>/eos/active/switch/1=1.0</code> (increase value) <code>/eos/active/switch/1=-1.0</code> (decrease value)  <code>/eos/active/switch/1=4.0</code> (increase value rapidly)  <code>/eos/active/switch/1=0.25</code> (increase value slowly)  (defaults to Coarse mode, but wheel mode can be changed with the <code>/eos/switch</code> command)
<code>/eos/active/switch/fine/&lt;index&gt;</code>	number of fine wheel ticks (positive or negative)	<code>/eos/active/switch/fine/1=1.0</code>
<code>/eos/active/switch/course/&lt;index&gt;</code>	number of coarse wheel ticks (positive or negative)	<code>/eos/active/switch/course/1=1.0</code>

## Direct Selects



**Note:** Eos will send the description and button labels for all OSC direct selects. See [Implicit OSC Output, page 43](#) for more information.

OSC Method	Arguments	Examples/Comments
/eos/ds/<index>/<target type>/<button count>		<p>/eos/ds/1/chan/10 (create OSC direct select bank #1 with 10 channel buttons)</p> <p>/eos/ds/2/group/25 (create OSC direct select bank #2 with 25 group buttons)</p>
/eos/ds/<index>/<target type>/flexi/<button count>		/eos/ds/1/chan/flexi/10 (create OSC direct select bank #1 with 10 channel buttons, in flexi mode)
/eos/ds/<index>/<target type>/<page number>/<button count>		<p>/eos/ds/1/chan/3/10 (create OSC direct select bank #1 with 10 channel buttons on page 3) Can also be used to quick jump to a specific page</p>
/eos/ds/<index>/<target type>/flexi/<page number>/<button count>		<p>/eos/ds/1/chan/flexi/3/10 (create OSC direct select bank #1 with 10 channel buttons on page 3, in flexi mode) Can also be used to quick jump to a specific page</p>
/eos/ds/<index>/page/<delta>	number for page delta	<p>/eos/ds/1/page/1 (page down by 1)</p> <p>/eos/ds/1/page/-1 (page up by 1)</p> <p>/eos/ds/1/page/10 (page down by 10)</p>
/eos/ds/<index>/<button index>	number for button edge: 1.0=down, 0.0=up (optional)	<p>/eos/ds/1/1=1.0 (press first button of OSC direct select bank #1)</p> <p>/eos/ds/1/1=0.0 (release first button of OSC direct select bank #1)</p>

**Note:**

<index> is a 1-based index of any number of OSC direct select banks you wish to create.

<target type> may be one of the following:

- Chan
- Group
- Macro
- Sub
- Preset
- IP (intensity palette)
- CP (color palette)
- BP (beam palette)
- MS (Magic Sheet)
- Curve
- Snap (snapshot)
- FX (Effects)
- Pixmap

## Fader Banks

These fader banks are not the same as fader pages.

**Note:**

<index> is a 1-based index of any number of discrete sets of OSC fader banks you wish to create.

Eos will send the description and fader labels for all OSC fader banks. See [Explicit OSC Output, page 42](#) for more information.

OSC Method	Arguments	Examples/Comments
/eos/fader/<index>/config/<fader count>		/eos/fader/1/config/10 (create OSC fader bank #1 with 10 faders)
/eos/fader/<index>/config/<page number>/<fader count>		/eos/fader/1/config/2/10 (create OSC fader bank #1 with 10 faders on page 2) Can also be used to quick jump to a specific page
/eos/fader/<index>/page/<delta>		/eos/fader/1/page/1 (page down by 1)  /eos/fader/1/page/-1 (page up by 1) /eos/fader/1/page/10 (page down by 10)
/eos/fader/<index>/<fader index>	floating point number to set sub percent to	/eos/fader/1/2=0.75 (set the second fader in OSC fader bank #1 to 75%)
/eos/fader/<index>/<fader index>/load		/eos/fader/1/2/load
/eos/fader/<index>/<fader index>/unload		/eos/fader/1/2/unload

/eos/fader/<index>/<fader index>/stop		/eos/fader/1/2/stop
/eos/fader/<index>/<fader index>/fire		/eos/fader/1/2/fire
/eos/fader/<index>/<fader index>/out		/eos/fader/1/2/out
/eos/fader/<index>/<fader index>/home		/eos/fader/1/2/home
/eos/fader/<index>/<fader index>/min		/eos/fader/1/2/min
/eos/fader/<index>/<fader index>/max		/eos/fader/1/2/max
/eos/fader/<index>/<fader index>/full		/eos/fader/1/2/full
/eos/fader/<index>/<fader index>/level		/eos/fader/1/2/level
/eos/fader/<index>/<fader index>/+%		/eos/fader/1/2/+%
/eos/fader/<index>/<fader index>/-%		/eos/fader/1/2/-%

Key



**Note:**

For a list of supported key names - in the magic sheet editor select "Console Button" magic sheet item, and see the list of **[Hardkeys]** options.

OSC Method	Arguments	Examples/Comments
/eos/key/<name>	number for button edge: 1.0=down, 0.0=up (optional)	/eos/key/select active=1.0 (press <b>[Select Active]</b> button)  /eos/key/select active=0.0 (release <b>[Select Active]</b> button)  /eos/key/go 0 (press & release <b>[Go]</b> button)



**Note:**

For the slash key, use a backslash /eos/key\

## Address

OSC Method	Arguments	Examples/Comments
/eos/addr	number for address to select	/eos/addr=513
/eos/addr/<address>	number for level to set address to	/eos/addr/513=100 (useful for mapping to an OSC slider)
/eos/addr/<address>/dmx	number for DMX level to set address to	/eos/addr/513/dmx=255 (useful for mapping to an OSC slider)

## Magic Sheet

OSC Method	Arguments	Examples/Comments
/eos/ms	number of the magic sheet to open	/eos/ms=1
/eos/<ms number>	number of the magic sheet view to open (optional)	/eos/ms/1=2



## Chan

OSC Method	Arguments	Examples/Comments
/eos/chan	number for channel to select	/eos/chan=1
/eos/chan/<number>	number for channel level	/eos/chan/1=75 (useful for mapping to an OSC slider)
/eos/chan/<number>/out	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/out
/eos/chan/<number>/home	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/home
/eos/chan/<number>/level	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/level
/eos/chan/<number>/full	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/full
/eos/chan/<number>/min	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/min
/eos/chan/<number>/max	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/max
/eos/chan/<number>/+%	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/+%
/eos/chan/<number>/-%	number for button edge: 1.0=down, 0.0=up (optional)	/eos/chan/1/-%
/eos/chan/<number>/dmx	number for channel DMX level	/eos/chan/1/dmx=255
/eos/chan/<number>/param/ <parameter>	number for parameter level	/eos/chan/1/param/pan=90 (useful for mapping to an OSC slider)
/eos/chan/<number>/param/ <parameter 1>/<parameter 2>/...	number for all parameter levels  multiple numbers for each parameter level	/eos/chan/1/param/pan/ tilt=90 (set channel 1 pan & tilt to 90)  /eos/chan/1/param/pan/ tilt=45,90 (set channel 1 pan to 45 & tilt to 90)
/eos/chan/<number>/param/ <parameter>/dmx	number for parameter DMX level	/eos/chan/1/param/pan/ dmx=255

/eos/chan/<number>/param/ <parameter 1>/<parameter 2>/.../ dmx	number(s) for parameter DMX level(s)	/eos/chan/1/param/pan/tilt/ dmx=255
--	---	--

## Group



**Note:** Same syntax and behavior as Chan.

OSC Method	Arguments	Examples/Comments
/eos/group		
/eos/group/<number>		
/eos/group/<number>/out		
/eos/group/<number>/home		
/eos/group/<number>/level		
/eos/group/<number>/full		
/eos/group/<number>/min		
/eos/group/<number>/max		
/eos/group/<number>/+%		
/eos/group/<number>/-%		
/eos/group/<number>/dmx		
/eos/group/<number>/param/ <parameter>		
/eos/group/<number>/param/ <parameter 1>/<parameter 2>/...		
/eos/group/<number>/param/ <parameter>dmx		
/eos/group/<number>/param/ <parameter 1>/<parameter 2>/.../ dmx		

## Macro

OSC Method	Arguments	Examples/Comments
/eos/macro	number for macro to select	/eos/macro=1
/eos/macro/fire	number for macro to run	/eos/macro/fire=1
/eos/macro/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/macro/1/fire=1.0

## Sub

OSC Method	Arguments	Examples/Comments
/eos/sub	number for sub to select	/eos/sub=1
/eos/sub/<number>	floating point number to set sub percent to	/eos/sub/1=0.75 (useful for mapping to an OSC slider)
/eos/sub/<number>/out	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/1/out
/eos/sub/<number>/home	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/home
/eos/sub/<number>/level	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/1/level
/eos/sub/<number>/full	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/1/full
/eos/sub/<number>/min	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/1/min
/eos/sub/<number>/max	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/max
/eos/sub/fire	number for sub to bump	/eos/sub/fire=1
/eos/sub/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/sub/1/fire=1.0 (bump sub 1 on)  /eos/sub/1/fire=0.0 (bump sub 1 off)

## Preset

OSC Method	Arguments	Examples/Comments
/eos/preset	number of preset to select	/eos/preset=1
/eos/preset/fire	number of preset to recall	/eos/preset/fire=1
/eos/preset/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/preset/1/fire

## Intensity Palette

OSC Method	Arguments	Examples/Comments
/eos/ip	number of intensity palette to select	/eos/ip=1
/eos/ip/fire	number of intensity palette to recall	/eos/ip/fire=1
/eos/ip/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/ip/1/fire

## Focus Palette

OSC Method	Arguments	Examples/Comments
/eos/fp	number of focus palette to select	/eos/fp=1
/eos/fp/fire	number of focus palette to recall	/eos/fp/fire=1
/eos/fp/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/fp/1/fire

## Color Palette

OSC Method	Arguments	Examples/Comments
/eos/cp	number of color palette to select	/eos/cp=1
/eos/cp/fire	number of color palette to recall	/eos/cp/fire=1
/eos/cp/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/cp/1/fire

## Beam Palette

OSC Method	Arguments	Examples/Comments
/eos/bp	number of beam palette to select	/eos/bp=1
/eos/bp/fire	number of beam palette to recall	/eos/bp/fire=1
/eos/bp/<number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/bp/1/fire

## Cue

OSC Method	Arguments	Examples/Comments
/eos/cue	number of cue to select	/eos/cue=1.5
/eos/cue/<list number>	number of cue to select (in the specified cue list)	/eos/cue/1=1.5
/eos/cue/<list number>/<cue number>	number of cue part to select (in the specified cue list and cue number)	/eos/cue/1/1.5=2
/eos/cue/fire	number of cue to run	/eos/cue/fire=1
/eos/cue/<cue number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/cue/1.5/fire
/eos/cue/<list number>/<cue number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/cue/1/1.5/fire
/eos/cue/<list number>/<cue number>/<part number>/fire	number for button edge: 1.0=down, 0.0=up (optional)	/eos/cue/1/1.5/2/fire

## Other Targets

OSC Method	Arguments	Examples/Comments
/eos/curve	number of curve to select	/eos/curve=1
/eos/fx	number of effect to select	/eos/fx=1
/eos/snap	number of snapshot to recall	/eos/snap=1
/eos/pixmap	number of pixel map to select	/eos/pixmap=1

## User

OSC Method	Arguments	Examples/Comments
/eos/user	number for OSC user ID	/eos/user=1 (set OSC user ID to 1)  /eos/user=-1 (set OSC user to match console)  /eos/user=0 (set OSC user as background user)

## Command Line

OSC Method	Arguments	Examples/Comments
/eos/cmd	string with command line text	/eos/cmd="Chan 1 At 75" (unterminated command)  /eos/cmd="Chan 1 At 75#" (terminated command)  /eos/cmd="Chan 1 At 75 Enter" (terminated command)
/eos/cmd	in-line command line arguments	/eos/cmd="Chan 1 At %1#", 75 (results in command line "Chan 1 At 75#"  /eos/cmd="Chan %1 At %2#", 1, 75
/eos/cmd/<text>/<text>/<text>/...	in-line command line arguments (optional)	/eos/cmd/Chan/1/At/75 /eos/cmd/Chan/%1/At/ %2#=1, 75
/eos/newcmd	Same behavior as /eos/cmd, but it resets the command line first	
/eos/event	Same behavior as /eos/cmd, but treated as console event	
/eos/newevent	Same behavior as /eos/cmd, but it resets the command line first	

## Other

OSC Method	Arguments	Examples/Comments
/eos/reset		/eos/reset  clears any active switches resets all persistent OSC settings (like OSC user ID & wheel modes) send ALL implicit OSC output commands

## Explicit OSC Output

OSC outputs exactly like Serial String outputs, but the string must start with an OSC address (ex: "/device/fader").

Optionally, you may add arguments by adding "=" to the string, followed by a comma-delimited list of arguments.

Numeric arguments with a decimal are treated as 32-bit floating point numbers. Numeric arguments without a decimal are treated as 32-bit integer numbers. Non-numeric arguments are treated as

strings.

Examples:

- `"/device/command"`
- `"/device/command=1"`
- `"/device/command=1.5"`
- `"/device/command=1.5,3.0"`
- `"/device/command=1.5,3.0,text"`

## Implicit OSC Output

When UDP transmit is enabled, certain OSC commands are sent out as appropriate:

Command Lines:

- `"/eos/out/user/<number>/cmd"`, <string argument with current command line text for the current console user>
- `"/eos/out/cmd"`, <string argument with current command line text>

OSC Settings:

- `"/eos/out/user"`, <integer argument with current OSC user ID>
- `"/eos/out/wheel"`, <float argument with current OSC wheel mode: 0.0=Coarse, 1.0=Fine>
- `"/eos/out/switch"`, <float argument with current OSC switch mode: 0.0=Coarse, 1.0=Fine>
- `"/eos/out/show/name"`, <string argument with show title>

Active Channels and Parameters:

- `"/eos/out/active/chan"`, <string argument with active channels and current value from the 1st channel>
- `"/eos/out/active/wheel/<number>`, <string argument with parameter name and current value from the 1st channel>



---

**Note:**

This allows you to create an ML-Controls style interface via OSC. For example, on the OSC-enabled device, setup 10 wheels (`/eos/active/wheel/<1-10>`) with matching labels.

If using in conjunction with `"/eos/active/switch/<number>`, you should still use `"/eos/out/active/wheel/<number>` to display feedback for that switch

---

Active Cue:



---

**Note:**

Updated once per second.

---

- `"/eos/out/active/cue/<cue list number>/<cue number>`, <float argument with percent complete (0.0-1.0)>
- `"/eos/out/active/cue"`, <float argument with percent complete (0.0-1.0)>
- `"/eos/out/active/cue/text"`, <string argument with descriptive text about the active cue, ex: `"1/2.3 Label 0:05 75%"`>
- `"/eos/out/pending/cue/<cue list number>/<cue number>`

- `"/eos/out/pending/cue/text"`, <string argument with descriptive text about the pending cue, ex: "1/2.4 Label 0:30">

#### OSC Direct Select Banks:

- `"/eos/out/ds/<index>"`, <string argument with descriptive text for the OSC direct select at <index>: target name, page number, and mode>>
- `"/eos/out/ds/<index>/<button index>"`, <string argument with button label for OSC direct select at <index> for button <button index>>

#### OSC Fader Banks:

- `"/eos/out/fader/<index>"`, <string argument with descriptive text for the OSC fader bank at <index>>
- `"/eos/out/fader/<index>/<fader index>/name"`, <string argument with fader label for OSC fader bank at <index> for fader <fader index>>
- `"/eos/fader/<index>/<fader index>"`, <floating point number for fader percent: 0.0-1.0>



#### **Note:**

Eos will delay sending fader levels for faders that have been moved via OSC commands for 3 seconds. If you move a fader on an OSC remote control, Eos will send the actual fader level 3 seconds later.

#### OSC Show Control Events:

- `"/eos/out/event/cue/<cue list number>/<cue number>/fire"`
- `"/eos/out/event/cue/<cue list number>/<cue number>/stop"`
- `"/eos/out/event/sub/<sub number>"`, <integer argument, 0=Bump Off, 1=Bump On>
- `"/eos/out/event/macro/<macro number>"`
- `"/eos/out/event/relay/<relay number>/<group number>"`, <integer argument, 0=On, 1=Off>
- `"/eos/out/event"` (used for time code learn)

#### Miscellaneous Console Events:

- `"/eos/out/event/show/saved"`, <string argument with file path>
- `"/eos/out/event/show/loaded"`, <string argument with file path>
- `"/eos/out/event/show/cleared"`
- `"/eos/out/event/state"`, <integer argument, 0=Blind, 1=Live>



#### **Note:**

When Eos receives the command `"/eos/ping"` it will reply with `"/eos/out/ping"`. You may optionally add any number of arguments and Eos will reply with the same arguments. This may be useful for testing latency.



# Index

## Symbols

{ACN MIDI Rx ID(s)} . . . . .	4
{ACN MIDI Tx ID} . . . . .	5
{Analog Inputs} . . . . .	5
{MIDI Cue List} . . . . .	5
{MIDI Time Code Rx} . . . . .	4
{MSC Receive Channel} . . . . .	4
{MSC Receive} . . . . .	4
{MSC Transmit Channel} . . . . .	4
{MSC Transmit} . . . . .	4
{OSC RX Port Address} . . . . .	5
{OSC TX IP Address} . . . . .	5
{OSC TX Port Address} . . . . .	5
{Relay Outputs} . . . . .	5
{Resync Frames} . . . . .	4
{SMPTE Time Code Rx} . . . . .	4
{String MIDI TX} . . . . .	5
{String RX Group IDs} . . . . .	5
{String RX Port} . . . . .	6
{String RX} . . . . .	5
{String TX Group IDs} . . . . .	6
{String TX IP Address} . . . . .	6
{String TX Port} . . . . .	6
{String TX} . . . . .	6

## A

ACN String . . . . .	25
analog	
{Analog Inputs} . . . . .	5
definition . . . . .	3
analog input	
events . . . . .	19
analog inputs . . . . .	19

## E

ETC locations . . . . .	1
event . . . . .	8
creating . . . . .	11
external time . . . . .	10
internal time . . . . .	10
list . . . . .	9
events	
adding . . . . .	15

adding from live . . . . .	15, 16
adding using learn mode . . . . .	16

## L

learn (key)	
using in show control . . . . .	16
learn mode	
adding events . . . . .	16
live	
adding events . . . . .	15, 16

## M

MIDI raw . . . . .	26
definition . . . . .	3
programming from cues . . . . .	26
programming from macros . . . . .	26
MIDI show control . . . . .	21
{ACN MIDI Rx ID(s)} . . . . .	4
{ACN MIDI Tx ID} . . . . .	5
{MIDI Cue List} . . . . .	5
{MSC Receive Channel} . . . . .	4
{MSC Receive} . . . . .	4
{MSC Transmit Channel} . . . . .	4
{MSC Transmit} . . . . .	4
command . . . . .	21
data sources . . . . .	21
definition . . . . .	3
devices . . . . .	21
MIDI time code	
{MIDI Time Code Rx} . . . . .	4
definition . . . . .	3
enabling . . . . .	13

## N

network serial (UDP) . . . . .	24
--------------------------------	----

## O

open sound control . . . . .	27
{OSC RX Port Address} . . . . .	5
{OSC TX IP Address} . . . . .	5
{OSC TX Port Address} . . . . .	5
{String RX} . . . . .	5
definition . . . . .	3



enable . . . . .	27	{Resync Frames} . . . . .	4
output . . . . .	42	clocks . . . . .	14
supported input . . . . .	27	color coding . . . . .	13
using . . . . .	27	enabling . . . . .	13
<b>R</b>		execution while editing . . . . .	16
real time clock (RTC) . . . . .	17	MIDI . . . . .	13
definition . . . . .	3	programming . . . . .	14
enabling . . . . .	17	SMPTE . . . . .	13
events . . . . .	17	timing . . . . .	14
programming . . . . .	17	timing data . . . . .	13
relay outputs . . . . .	19	<b>U</b>	
relays		UDP . . . . .	24
{Relay Outputs} . . . . .	5	<b>W</b>	
definition . . . . .	3	web site . . . . .	1
<b>S</b>			
serial, see string interface			
setup . . . . .	4		
show control			
analog inputs . . . . .	19		
relay outputs . . . . .	19		
using time code . . . . .	??–16		
show control display			
adding events . . . . .	15		
SMPTE			
{SMPTE Time Code Rx} . . . . .	4		
definition . . . . .	3		
enabling . . . . .	13		
string			
{String MIDI TX} . . . . .	5		
{String RX Group IDs} . . . . .	5		
{String RX Port} . . . . .	6		
{String RX} . . . . .	5		
{String TX Group IDs} . . . . .	6		
{String TX IP Address} . . . . .	6		
{String TX Port} . . . . .	6		
string interface . . . . .	23		
commands . . . . .	23		
definition . . . . .	3		
enabling . . . . .	23		
receiving via the I/O Gateway . . . . .	24		
<b>T</b>			
technical services . . . . .	1		
time code			



**Corporate Headquarters** ■ Middleton, WI, USA ■ Tel +608 831 4116 ■ Service: (Americas) [service@etconnect.com](mailto:service@etconnect.com)  
**London, UK** ■ Tel +44 (0)20 8896 1000 ■ Service: (UK) [service@etceurope.com](mailto:service@etceurope.com)  
**Rome, IT** ■ Tel +39 (06) 32 111 683 ■ Service: (UK) [service@etceurope.com](mailto:service@etceurope.com)  
**Holzkirchen, DE** ■ Tel +49 (80 24) 47 00-0 ■ Service: (DE) [techserv-hoki@etconnect.com](mailto:techserv-hoki@etconnect.com)  
**Hong Kong** ■ Tel +852 2799 1220 ■ Service: (Asia) [service@etcasia.com](mailto:service@etcasia.com)  
**Web** ■ [www.etconnect.com](http://www.etconnect.com) ■ Copyright © 2015 ETC. All Rights Reserved.  
**4250M2410** ■ Rev B ■ Released 2015-07 ■ Product information and specifications subject to change.