USER MANUAL

grandMA3

Version 1.7 | 2022-05-09

English

Table of contents

1. About The Manual

Founded on the legacy of the previous grandMA consoles, the grandMA3 represents a radical re-think of what's possible from a lighting control platform. The elegant new system-architecture incorporates new fixture, feature and effects handling at its very heart.

The system features ground-breaking concepts from top industry visionaries, presented within a refined user interface, and is designed to make practical tasks more intuitive.

The physical design detail of the grandMA3 range is extraordinary and focused on delivering the best possible user experience, now and into the future.

The **grandMA3 software** is being developed by

MA Lighting Technology GmbH
Dachdeckerstraße 16
97297 Waldbüttelbrunn, Germany

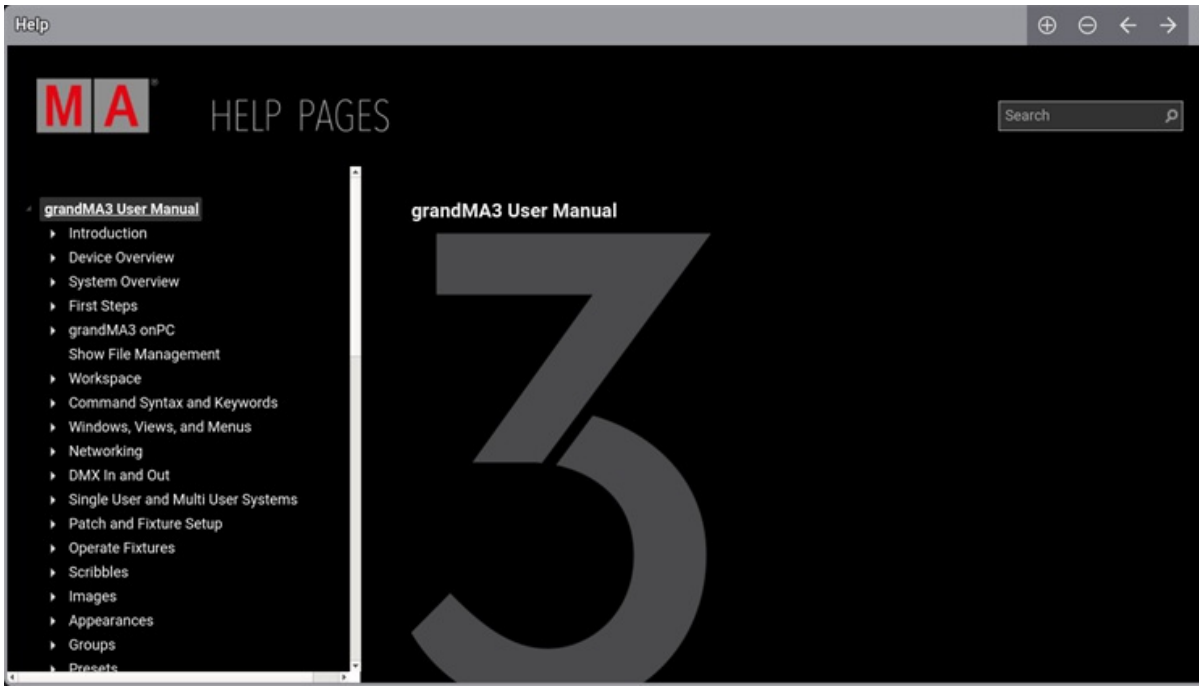
The help pages consist of a collection of topics to help the user understand the software.

These help pages are included in every software installer package and are automatically loaded into the software (exception: Nodes and Processing Units).

Access the help pages

There are different ways to access the help:

1. To access and read the help pages, press Help + Please. The help pop-up opens on the main screen:



Help pop-up

2. To open the help pop-up, tap ? in the control bar and ? again.
3. To open the help window, tap the Help view button in the view bar.

Empty screen of a new show

For more information about the vertical bars, see the topics Workspace, User Interface, **Control Bar,** and **View Bar**.

4. To open another help window, open the **Add Window**, tap the More tab, then tap Help. For more information, see the **Add Windows** topic.

Help window

5. To jump directly into a special help topic, use the context-sensitive help:

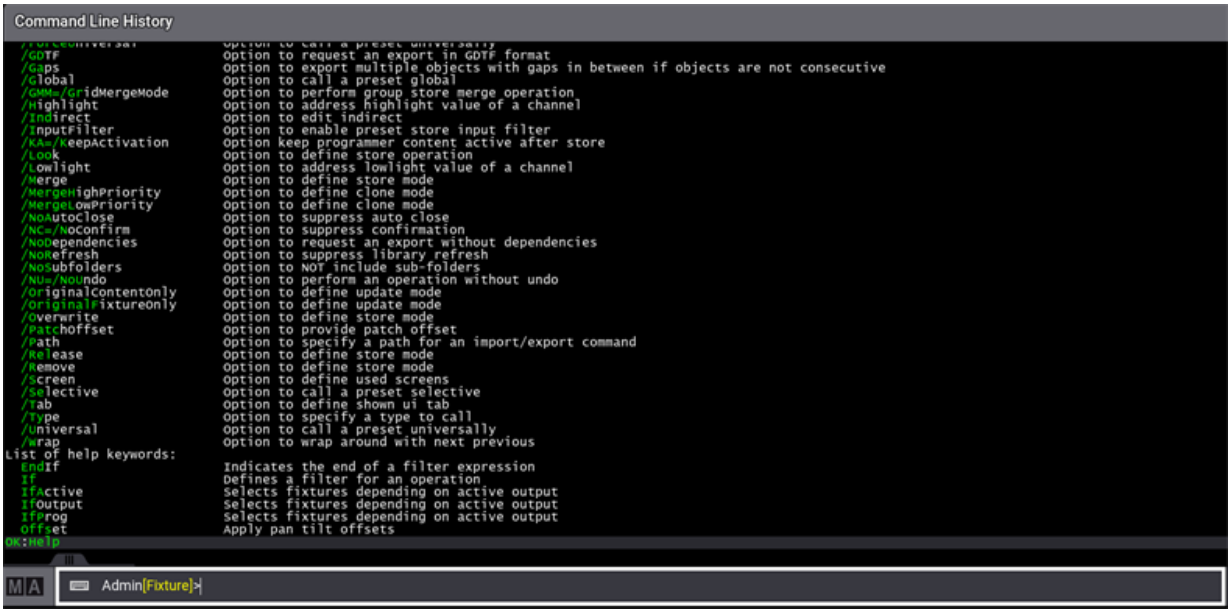
- Press and hold Help and tap the desired user interface element (UI element) to open the corresponding help topic.
- Tap ? in the control bar and tap a UI element to open the corresponding help topic.

6. To access the help pages via the command line, use one of these options:

- Press Help
- Type **Help**
- Type the shortcut **H**

- Press Please or click the command line with the mouse to confirm.

Hint:
To display all keywords in the command line history, type the command **Help** + **Enter** (keyboard) in the command line.



Keyword list in the command line history

For more information, see the **Help keyword topic**.

7. To open a keyword topic immediately via the command line, use one of these options:

- Press Help and type the desired keyword.
- Type **Help** and type the desired keyword.

The desired help pages opens.

Navigation in the Help

Buttons in the Title Bar

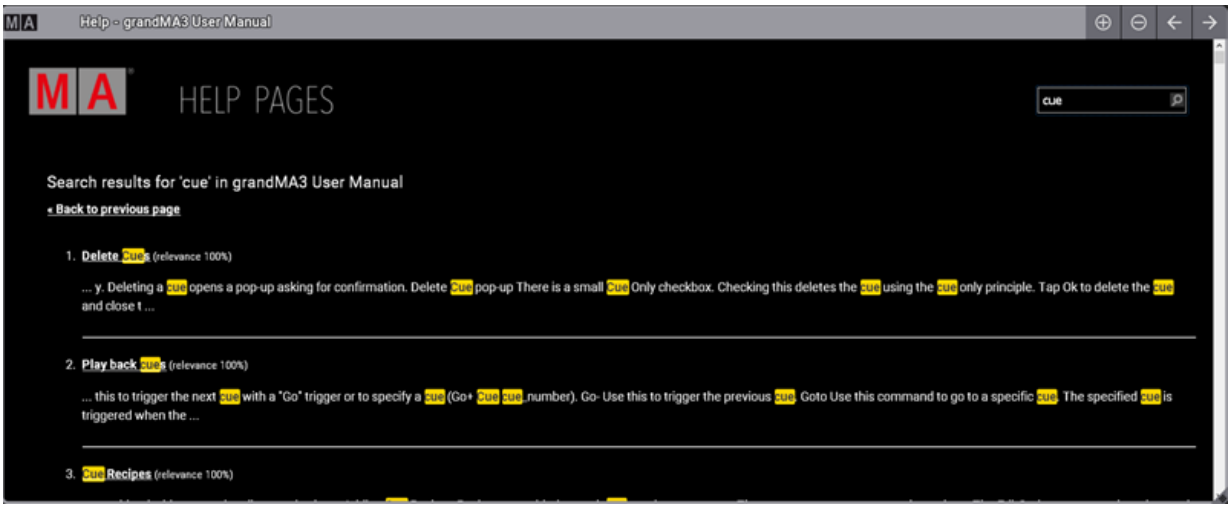
Tap the buttons with the plus or minus signs to increase or decrease the size of the content displayed within the help window.

Press the button with the left arrow to reverse through the history of previously viewed help topics. Press the button with the right arrow to advance through the history of previously viewed help topics.

Search Field

To find topics containing a specific search term, enter the term into the search field in the upper-right corner and press Please.

The **Help** window displays the search results with the search term highlighted.



Highlighted search results

To search for a specific product or search term, leave away the brand name or the series name (e.g. grandMA3).

Table of Contents

The left side of the **Help** window displays available topics in a tree structure. Tap the triangle next to any topic to reveal or hide any associated sub-topics. Tap the name of any topic to display its contents in the main area of the window.

The root topics in the tree structure are divided into the different products within the family. These main sections include:

- **grandMA3 User Manual**
- **grandMA3 Quick Manual consoles**
- **grandMA3 Quick Manual processing**
- **grandMA3 Quick Manual Nodes**

- **grandMA3 Quick Manual Nodes DIN-Rail**
- **grandMA3 Quick Manual onPC command wing XT**
- **grandMA3 Quick Manual onPC command wing**
- **grandMA3 Quick Manual onPC fader wing**
- **grandMA3 Quick Manual viz-key**
- **grandMA3 Quick Manual I/O Nodes**
- **Release Notes**

Important:
The complete grandMA3 documentation consists of the grandMA3 User Manual and the respective grandMA3 Quick Manual with technical specifications, safety instructions, and declaration of conformity.

Display of Content

Display	Definition
At	Keys found on the control surface of the hardware are displayed in a white font on a dark gray background.
Dimmer	Buttons found on the screen are displayed in a white font on a light gray background.
1. Open... 2. Select...	Step-by-step instructions are displayed in a numeric list.
• Tap	One-step instructions are displayed with the help of bullet points.
Menu - Settings - User Configuration - User Profiles	Navigation to specific menus in the console is displayed in bold.
Assign [Object] (At) [Object]	The description of the syntax is underlined and displayed in bold.

User name[Fixture]> Enter commands in the command line

OK:System response shown in the Command Line History

The following symbols display possible danger, useful hints, and information.

For information on safety, harmonized standards, and conformity see **grandMA3 Quick Manual Consoles**.

Warning:
The warning sign displays possible injury and hazard.

Restriction:
The restriction sign displays known limitations of functions.

Important:
The important sign displays essential information for the usage of the console.

Hint:
The hint sign displays tips useful for the usage of the console.

Help on the Web

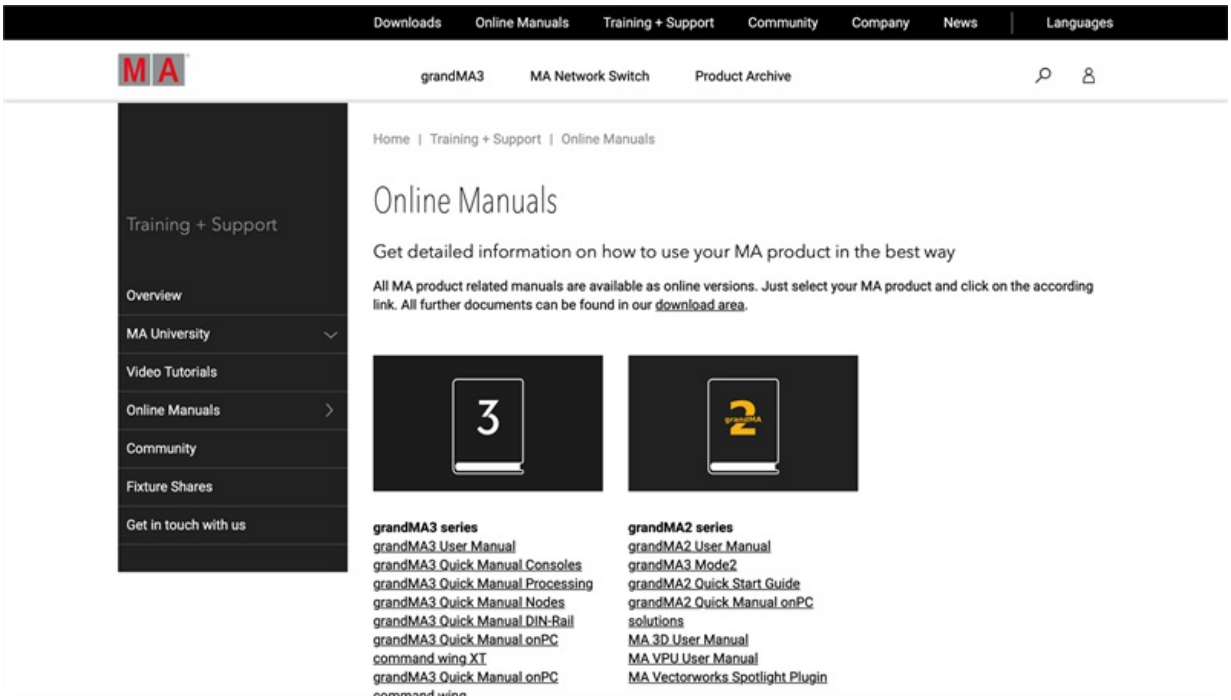
Please note that the help topics included in the software represent the status and volume of the help pages at the date of release.

The help system is constantly being updated and extended.

The most up-to-date and complete version of the manual is directly published on the following website:

https://www.malighting.com/training-support/online-manuals

Make sure that you regularly visit the online help to read the latest version of the help pages.



Online manuals

The **Online Manuals section** is divided into different **product families**:

- grandMA3 series
- grandMA2 series
- dot2 series
- MA Network Switch

PDF Files

Click the "Open topic as PDF" button in the upper-right corner of any topic viewed on the web to download the contents of that topic as a PDF file for offline viewing.

To download a single PDF of the entire manual:

1. Navigate to the **Downloads** section of the MA Lighting website: **https://www.malighting.com/downloads**
2. Click the desired product family from the list on the left side.
3. Click **Offline User Manuals + QuickMAnual**.
4. Click the desired manual to download.

Search Field

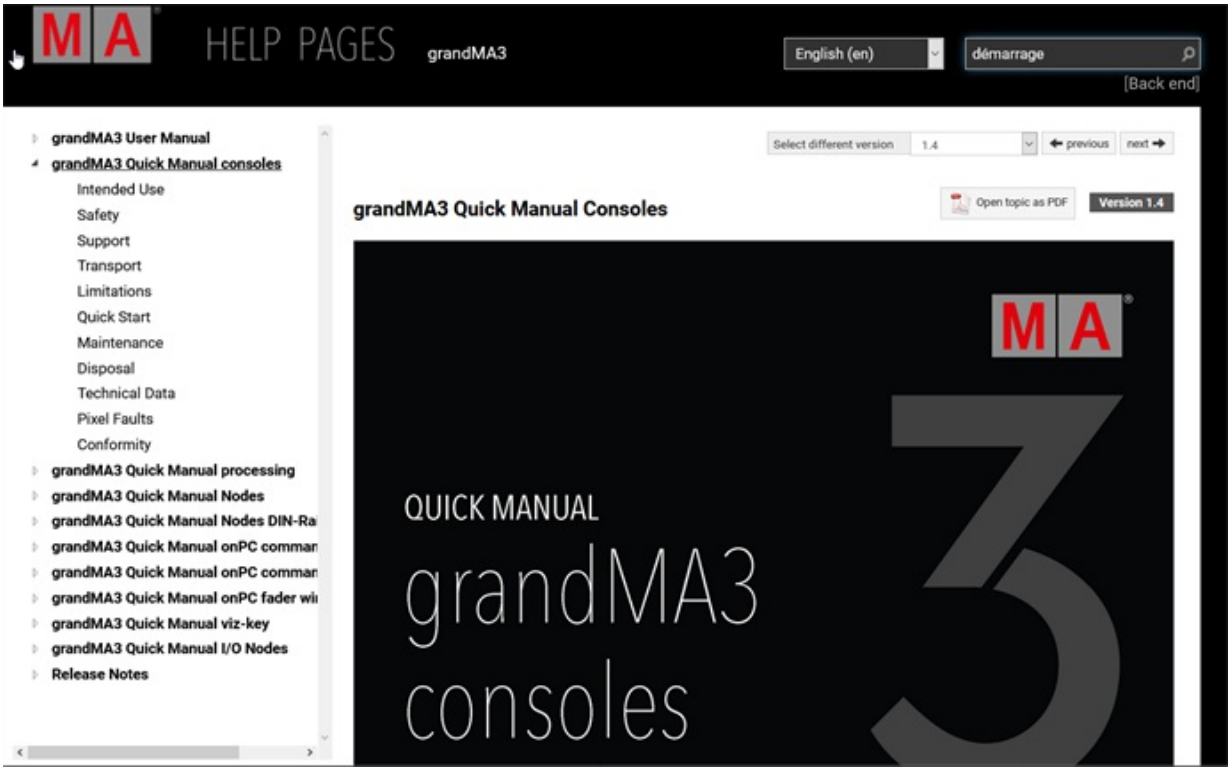
To find topics containing a specific search term, enter the term into the search field in the upper-right corner and press Enter.

The **Help** window displays the search results with the search term highlighted.

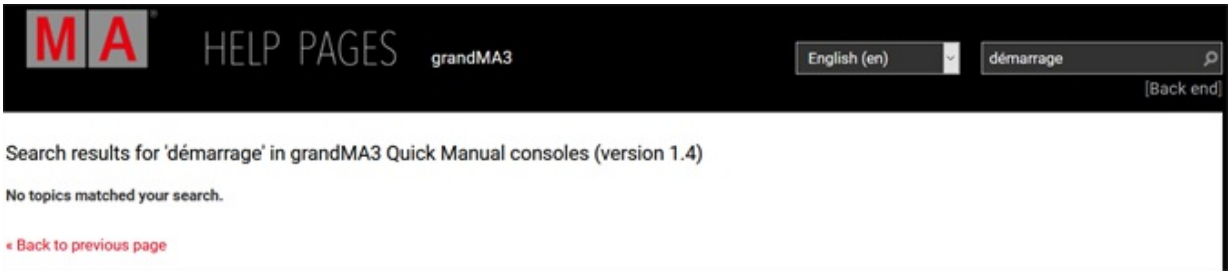
The search function is language-based. If the search term cannot be found, check your language selection.

Example:

Search for the term démarrage in the Quick Manual for the grandMA3 consoles.



The search does not find the term in the English database:



Change the language setting from English to French and search again:

The search result is now correct:

Support

Contact us for any questions about your MA product.

MA Lighting and its extensive distributor network offer unparalleled technical support. Call on our expertise for help with any problem, no matter if it is about the operation, software features, software installations, or troubleshooting.

Help Pages

The help pages are included in every software installer package and are automatically loaded into the software. To access and read the help pages, press Help + Please. The help pop-up opens on the main screen.

To jump directly into a special help topic, use the context-sensitive help:

- Press and hold Help and tap the desired user interface element (UI element) to open the corresponding help topic.
- Tap ? in the control bar and tap a UI element to open the corresponding help topic.

Online Manuals

All MA product-related manuals are available as online versions. Just select your MA product and click on the respective link. All further documents can be found in our download area. Go to **https://www.malighting.com/training-support/online-manuals**.

Video Tutorials

Get tips and tricks from our MA experts on your MA product.

MA offers a whole range of videos on how to use your MA product in the best possible way. Just select your series and click on the links to watch the videos on our YouTube channel. Go to **https://www.malighting.com/training-support/video-tutorials**.

Community

Share your knowledge and get help from other MA users.

A community can be stronger and better than the individual. Be part of the MA community! Go to **https://www.malighting.com/training-support/community**.

Fixture Shares

In our MA Fixture Share you will find a comprehensive database of different fixture types for grandMA3, grandMA2, and dot2. You can also find the grandMA2 GDTF Converter there. Simply upload the GDTF file and it will be converted into a grandMA2 Fixture Profile.

The GTDF Share contains a comprehensive library of all available GDTF files with all relevant modes and firmware versions. Like the MA Fixture Share, this database is continuously updated.

For more information see: **https://www.malighting.com/training-support/fixture-shares**

Local Support

If you have a question regarding your MA Lighting product, please first of all contact your local MA product specialists.

Central Support

If you are in need of further assistance or manufacturer support, please fill out an email request on **www.malighting.com** in Training and Support. Your request will automatically be sent to **support@malighting.com** (in English or German). This email service is monitored during MA Lighting's regular business hours in Germany from 8.30 am until 5 pm, Monday through Friday.

Emergency Hotline

In any case of a show critical emergency please contact the MA Lighting support hotline.

Phone +49 5251 68 88 65 99

Please note that this 24/7 hotline is strictly for emergency cases.

Intellectual Property

MA Lighting Technology has registered multiple patents, trademarks, design patents, and utility models for its products.
We will take legal action against the violation of this intellectual property.

For more information see: **https://www.malighting.com/intellectual-property**

2. Device Overview

The following chapters describe the different grandMA3 devices.

2.1. grandMA3 consoles

The grandMA3 consoles control all kinds of lighting genres such as conventional lights, moving lights, LEDs, video, and media via DMX signal or within a network environment.

All **grandMA3 components**, despite different hardware solutions, use the same software.

All components are fully integrable into the network environment.

This topic describes the different sections of the grandMA3 consoles.

These sections vary from model to model.

The flagship of the grandMA3 range is the **full-size** or **full-size CRV**, respectively.

The control room version (CRV) of full-size and light are models without the monitor wing but supplied with additional external monitor ports.

The grandMA3 **light** console or the **light CRV** are the work-horses of the range.

The **compact** and **compact XT** models offer the full system benefits in a compact and lightweight format.

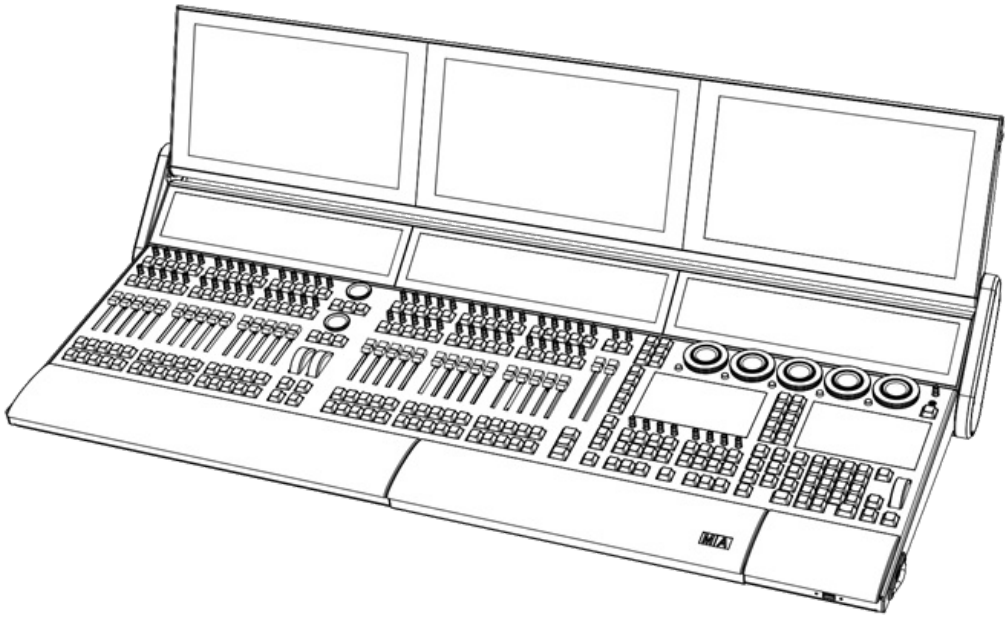
The grandMA3 consoles number their screens in a specific order, see **Screen Allocation**.

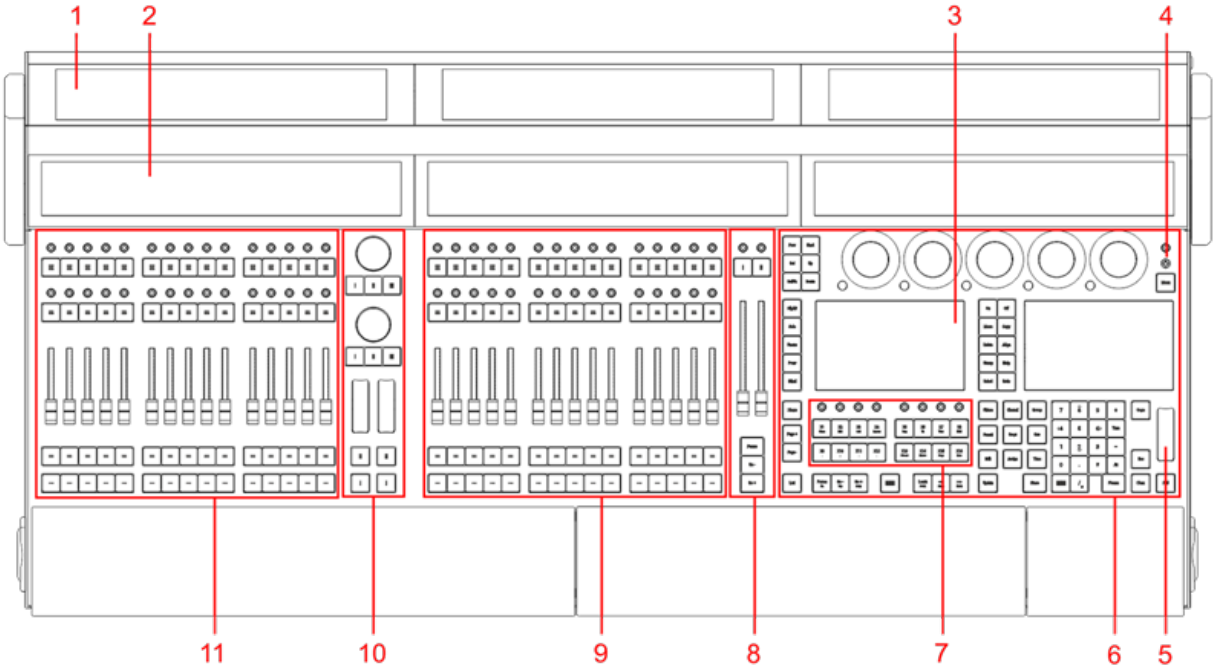
You can connect several external touch screens with a console, see **Connect External Screens**.

The display elements of the screens can be modified, see **Configuration of Displays**.

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.1.1. grandMA3 full-size

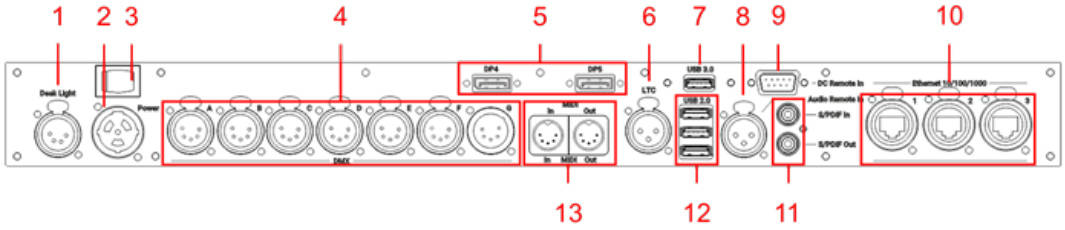




grandMA3 full-size front panel

1. **Screens 1-3**
2. **Letterbox screens 8-10**
3. **Command screens 6+7**
4. **Power key**
5. **Level wheel**

6. **Command area**
7. **Xkeys section**
8. **Master area**
9. **Right executor area with 3 executor sections**
10. **Custom area**
11. **Left executor area with 3 executor sections**



grandMA3 full-size rear panel

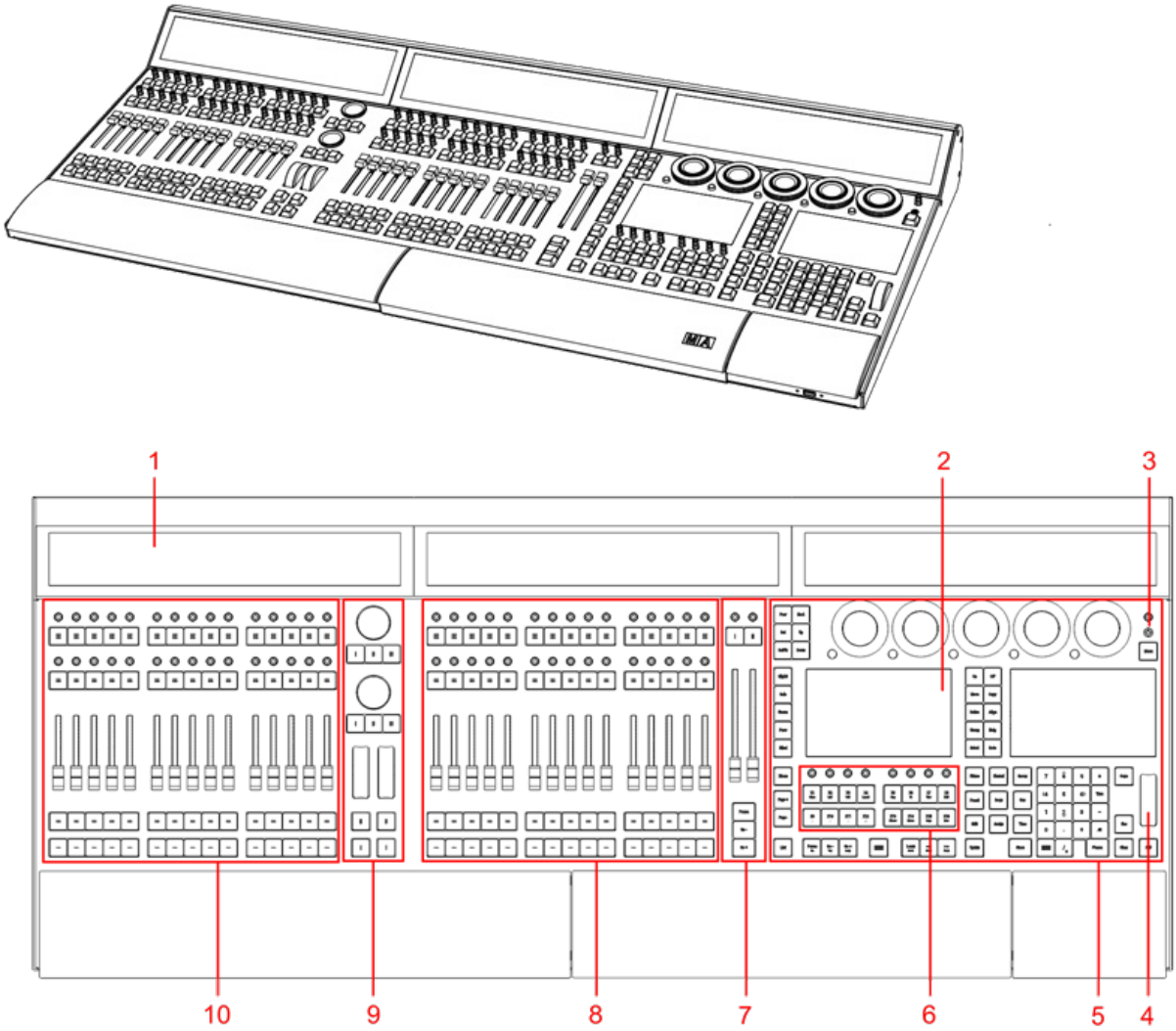
1. **Desk light**
2. **powerCON TRUE1**
3. Power switch
4. **DMX A-G**
5. **DisplayPort 4+5**
6. **LTC**
7. **USB 3.0**
8. **Audio Remote In**
9. **DC Remote Control**
10. **Ethernet 1-3**
11. **S/PDIF In+Out**
12. **USB 2.0**
13. **MIDI In+Out**

Hint:
To display the rear panel connectors on the letterbox screens, type:

User name[Fixture]> Menu "Connectorview"

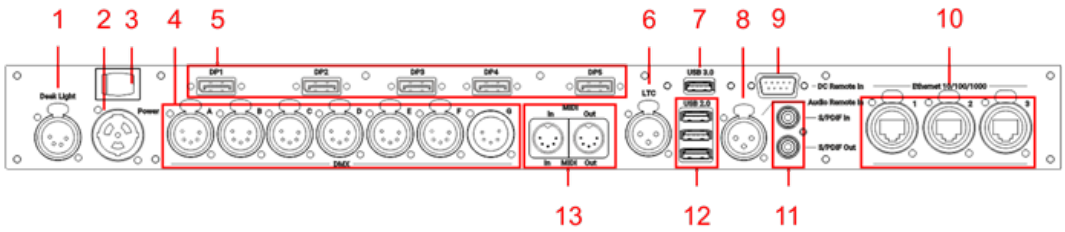
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.1.2. grandMA3 full-size CRV



grandMA3 full-size CRV front panel

1. **Letterbox screens 8-10**
2. **Command screens 6+7**
3. **Power key**
4. **Level wheel**
5. **Command area**
6. **Xkeys section**
7. **Master area**
8. **Right executor area with 3 executor sections**
9. **Custom area**
10. **Left executor area with 3 executor sections**



grandMA3 full-size CRV rear panel

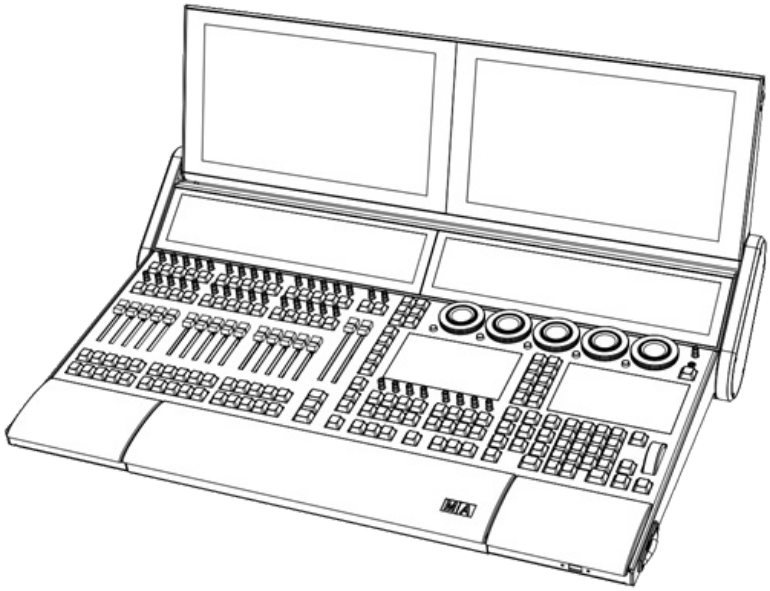
1. **Desk light**
2. **powerCON TRUE1**
3. Power switch
4. **DMX A-G**
5. **DisplayPort 1-5**
6. **LTC**
7. **USB 3.0**
8. **Audio Remote In**
9. **DC Remote Control**
10. **Ethernet 1-3**
11. **S/PDIF In+Out**
12. **USB 2.0**
13. **MIDI In+Out**

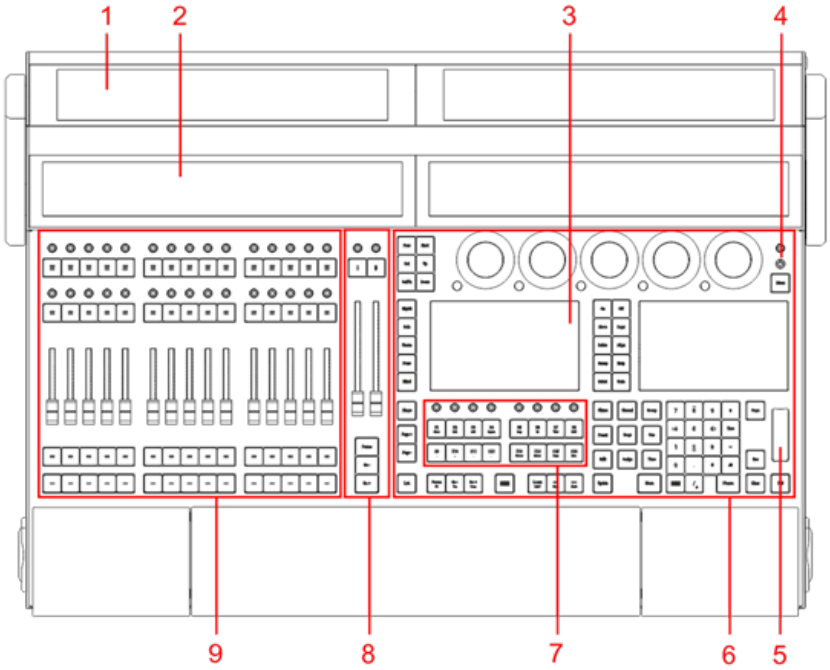
Hint:
To display the rear panel connectors on the letterbox screens, type:

User name[Fixture]> Menu "Connectorview"

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.1.3. grandMA3 light

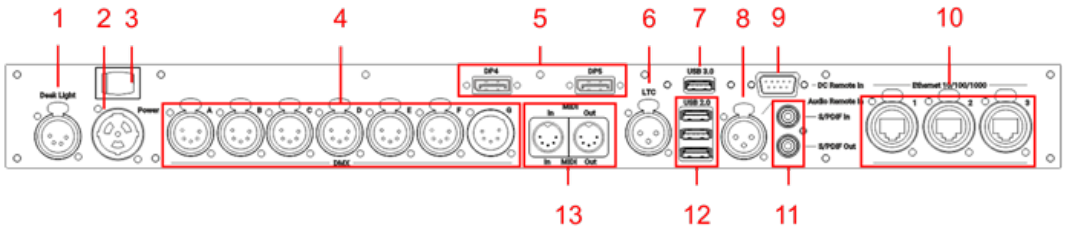




grandMA3 light front panel

1. **Screens 1+2**
2. **Letterbox screens 8+9**
3. **Command screens 6+7**
4. **Power key**

5. **Level wheel**
6. **Command area**
7. **Xkeys section**
8. **Master area**
9. **Executor area with 3 executor sections**



grandMA3 full-size rear panel

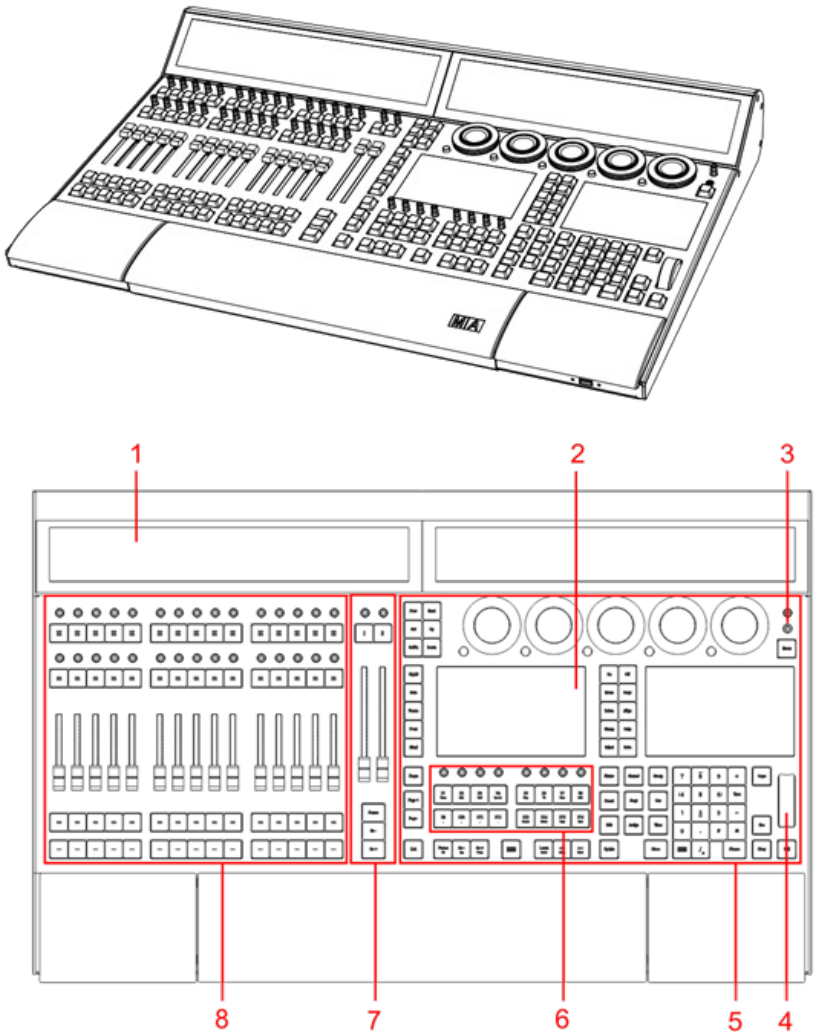
1. **Desk light**
2. **powerCON TRUE1**
3. Power switch
4. **DMX A-G**
5. **DisplayPort 4+5**
6. **LTC**
7. **USB 3.0**
8. **Audio Remote In**
9. **DC Remote Control**
10. **Ethernet 1-3**
11. **S/PDIF In+Out**
12. **USB 2.0**
13. **MIDI In+Out**

Hint:
To display the rear panel connectors on the letterbox screens, type:

User name[Fixture]> Menu "Connectorview"

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.1.4. grandMA3 light CRV

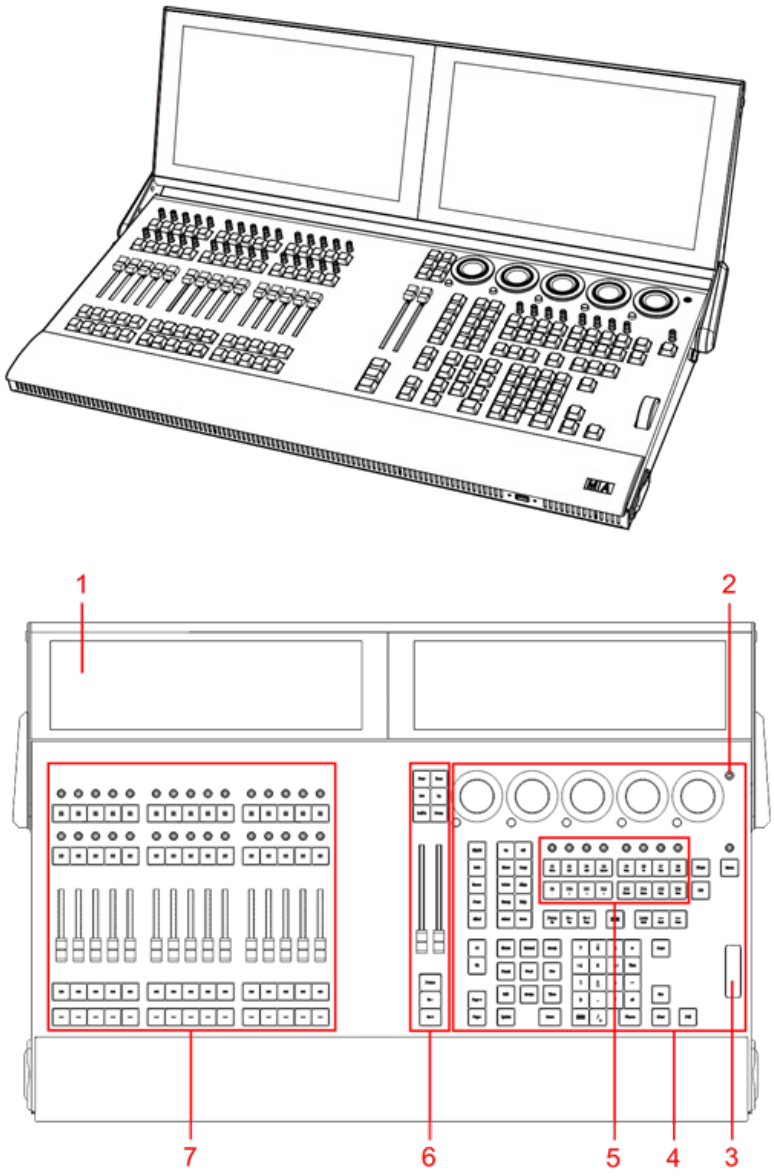


grandMA3 light CRV front panel

1. **Letterbox screens 8+9**
2. **Command screens 6+7**
3. **Power key**
4. **Level wheel**
5. **Command area**
6. **Xkeys section**
7. **Master area**
8. **Executor area with 3 executor sections**

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.1.5. grandMA3 compact XT

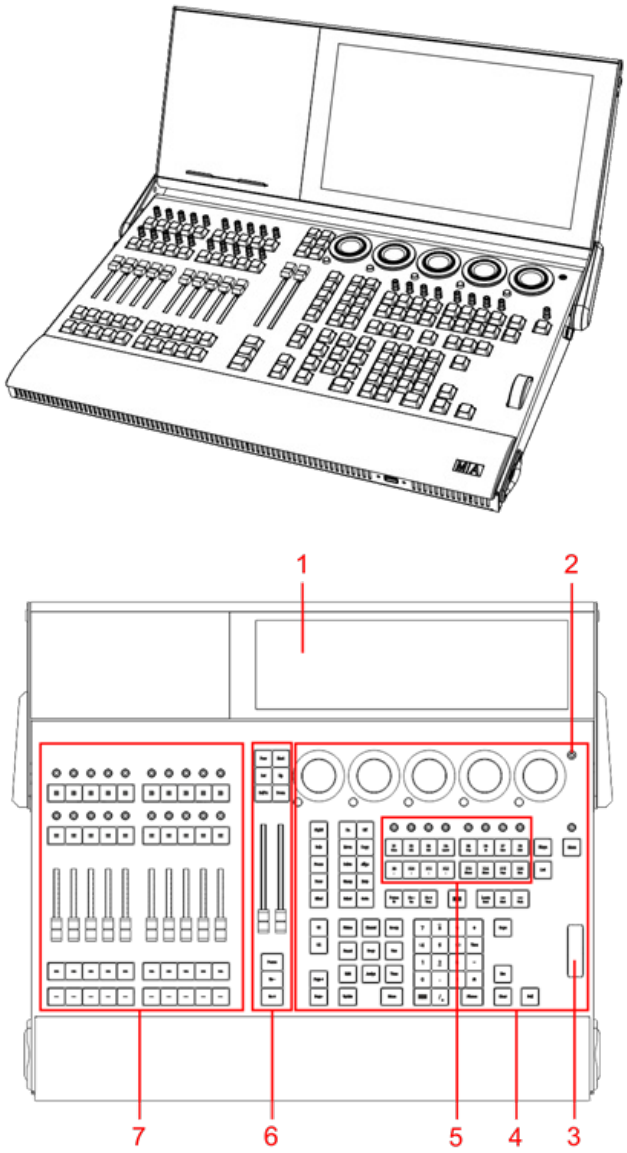


grandMA3 compact XT front panel

1. **Screens 1+2**
2. **Power key**
3. **Level wheel**
4. **Command section**
5. **Xkeys**
6. **Master section**
7. **Executor section**

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.1.6. grandMA3 compact

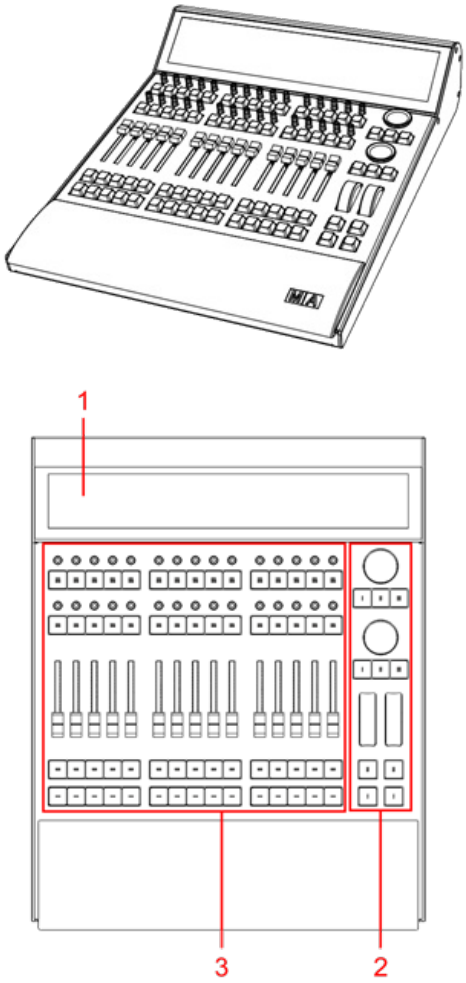


grandMA3 compact front panel

1. **Screen 1**
2. **Power key**
3. **Level wheel**
4. **Command section**
5. **Xkeys**
6. **Master section**
7. **Executor section**

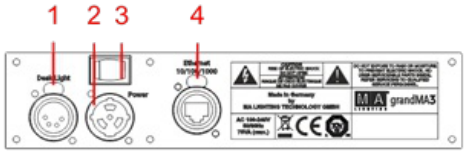
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.2. grandMA3 extension



grandMA3 extension front panel

1. **Letterbox screen 10**
2. **Custom section**
3. **Executor section**



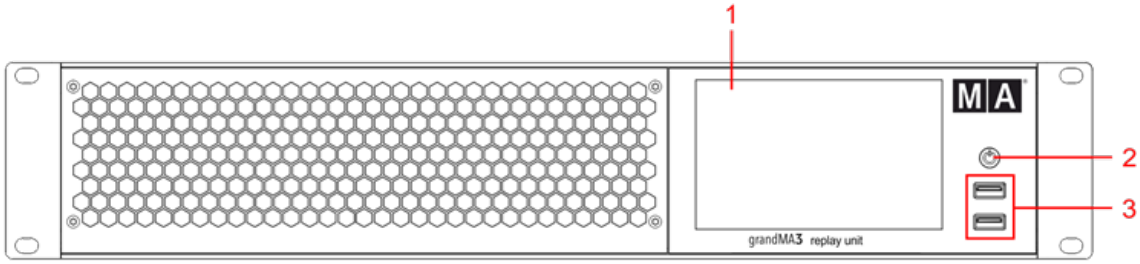
grandMA3 extension rear panel

1. **Desk light**
2. **powerCON TRUE1 connector**
3. Power switch
4. **Ethernet**

For more information, see the topics **First steps**, **Connect grandMA3 extension**.

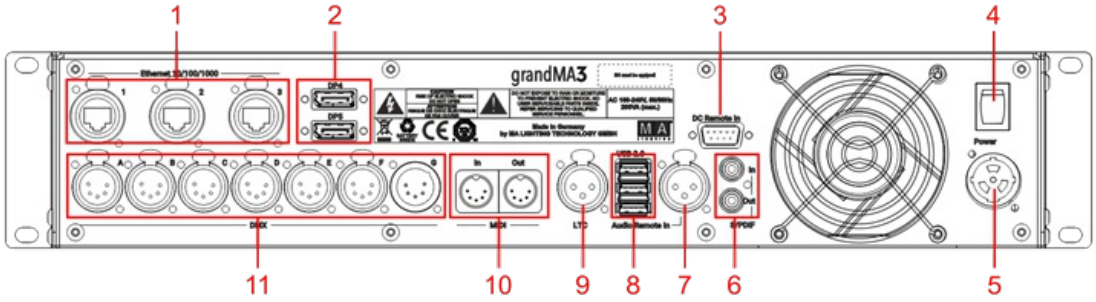
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.3. grandMA3 replay unit



grandMA3 replay unit front panel

1. Internal command multi-touch screen
2. **Power key**
3. **USB 2.0 (type A)**

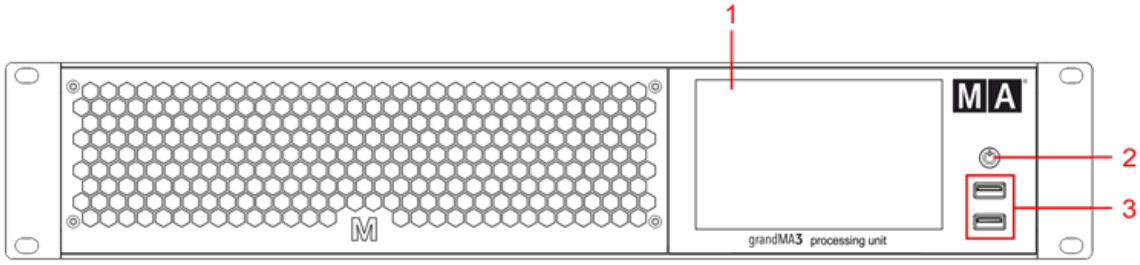


grandMA3 replay unit rear panel

1. **etherCON/RJ45**
2. **DisplayPort 1+2**
3. **DC Remote Control**
4. Power switch
5. **powerCON TRUE1**
6. **S/PDIF In+Out**
7. **Audio Remote In**
8. **USB 2.0 (type A)**
9. **LTC**
10. **MIDI In+Out**
11. **DMX-512-A OUT (5pin XLR female)**

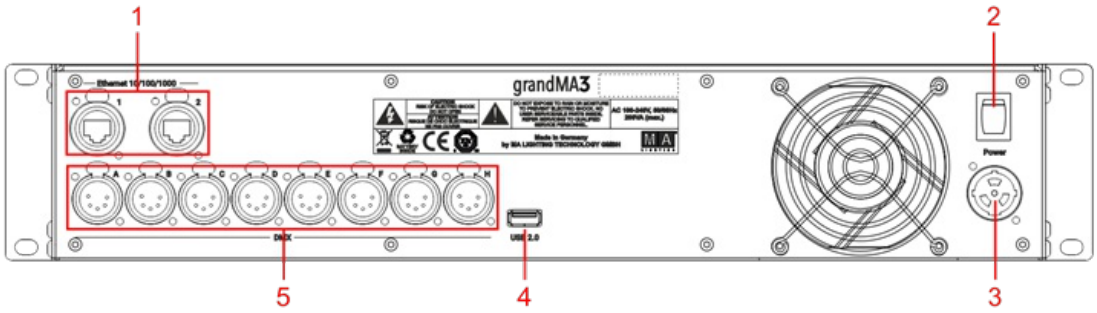
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.4. grandMA3 processing units



grandMA3 processing unit M, L, XL front panel

1. Internal command multi-touch screen
2. **Power key**
3. **USB 2.0 (type A)**



grandMA3 processing unit M, L, XL rear panel

1. **etherCON/RJ45**
2. Power switch
3. **powerCON TRUE1**
4. **USB 2.0 (type A)**
5. **DMX-512-A OUT (5pin XLR female)**

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual processing**.

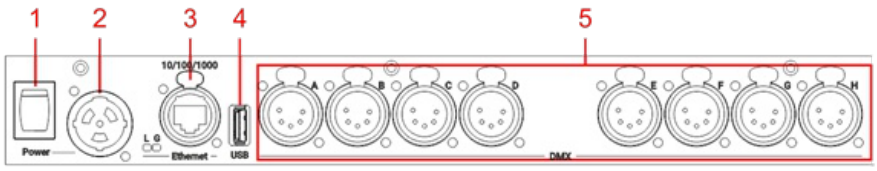
2.5. grandMA3 Nodes

grandMA3 8Port Node front panel

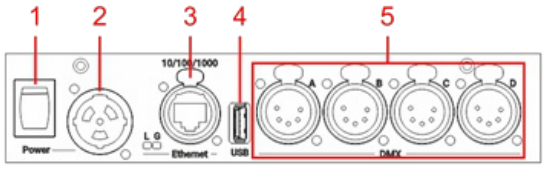
grandMA3 4Port Node front panel

grandMA3 2Port Node front panel

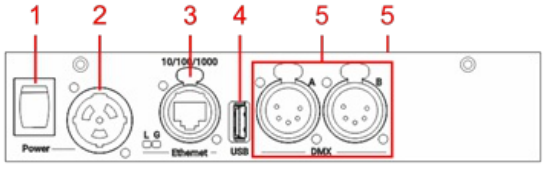
1. Display
2. Rotary knob



grandMA3 8Port Node rear panel



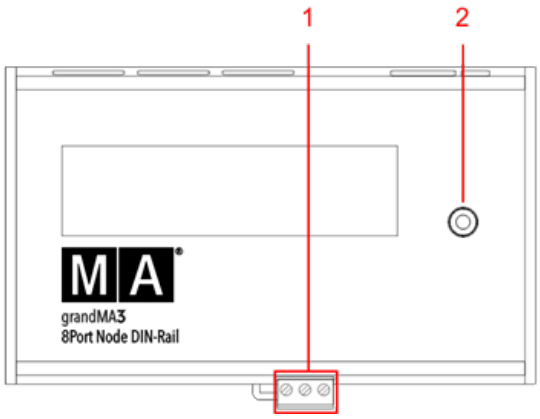
grandMA3 4Port Node rear panel



grandMA3 2Port Node rear panel

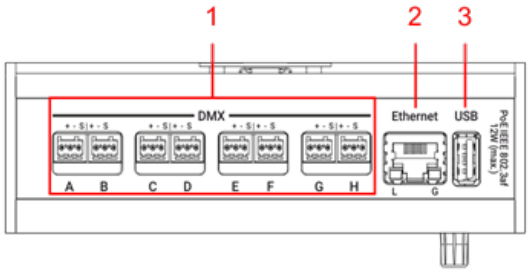
1. Power switch
2. **powerCON TRUE1**
3. **Ethernet** with **L (link) and G (gigabit) LEDs**
4. **USB port**
5. **DMX**

2.6. grandMA3 Nodes DIN-Rail



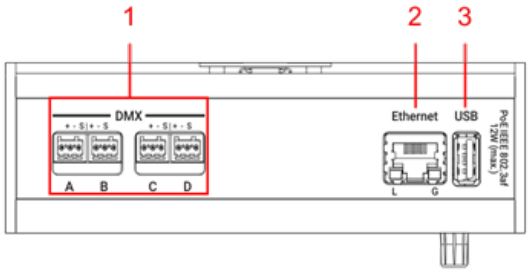
grandMA3 8Port Node DIN-Rail front panel

1. **Terminal block**
2. Rotary knob

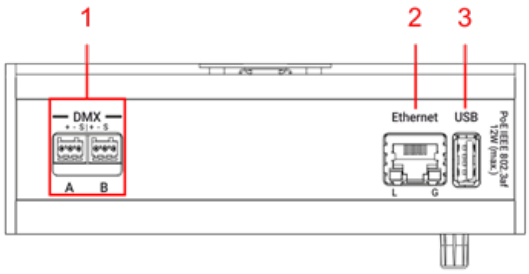


grandMA3 8Port Node DIN-Rail rear panel

1. **DMX**
2. **Ethernet** with **L (link) and G (gigabit) LEDs**
3. **USB 2.0**



grandMA3 4Port Node DIN-Rail rear panel



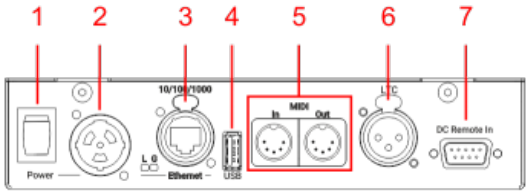
grandMA3 2Port Node DIN-Rail rear panel

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual Nodes DIN-Rail**.

2.7. grandMA3 I/O Node

grandMA3 I/O Node front panel

1. Display
2. Rotary knob

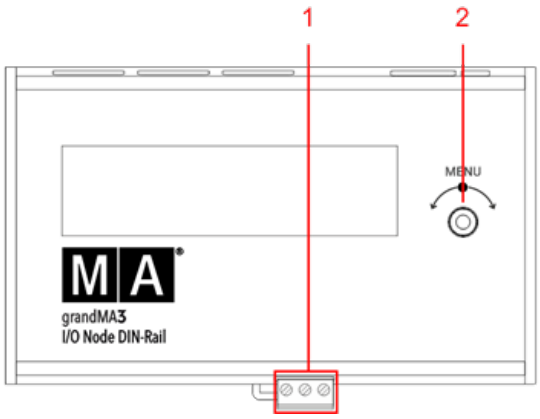


grandMA3 I/O Node rear panel

1. Power switch
2. **powerCON TRUE1**
3. **Ethernet** with **L (link) and G (gigabit) LEDs**
4. **USB 2.0**
5. **MIDI In+Out**
6. **LTC**
7. **DC Remote Control**

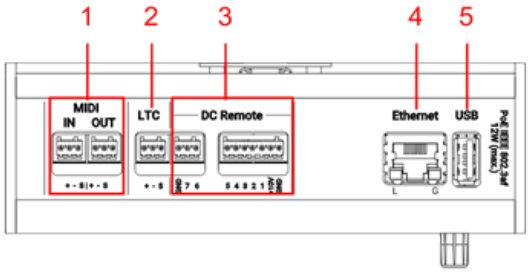
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual I/O Nodes**.

2.8. grandMA3 I/O Node DIN-Rail



grandMA3 I/O Node DIN Rail front panel

1. **Terminal block**
2. Rotary knob

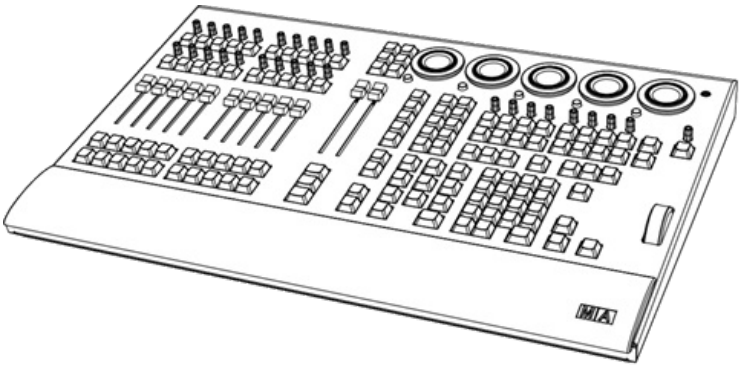


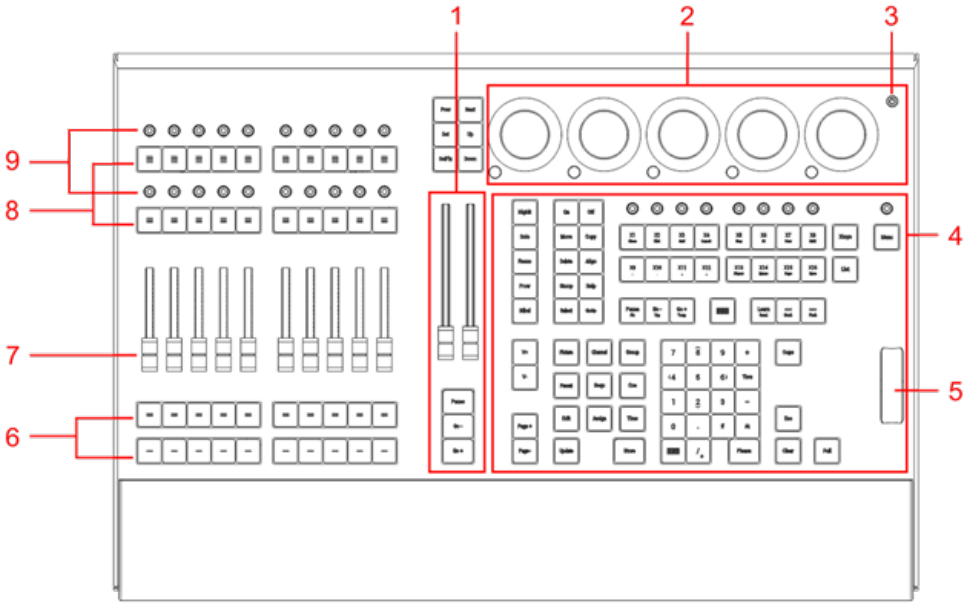
grandMA3 I/O Node DIN Rail rear panel

1. **MIDI In+Out**
2. **LTC**
3. **DC Remote Control**
4. **Ethernet** with **L (link) and G (gigabit) LEDs**
5. **USB 2.0**

For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual I/O Nodes**.

2.9. grandMA3 onPC command wing XT

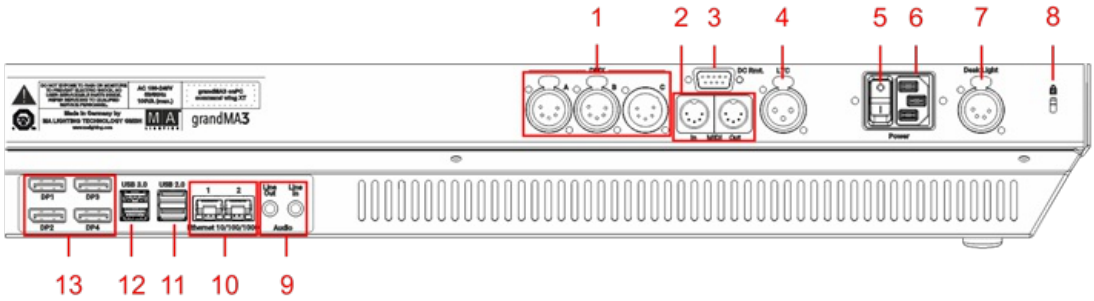




grandMA3 onPC command wing XT front panel

1. **Master area**
2. **Dual encoder section**
3. **Power key**
4. **Command area**
5. **Level wheel**
6. Executor buttons 101-190 + 201-290
7. Executor faders 201-290
8. Executor buttons 301-390 + 401-490
9. Executor knobs 301-390 + 401-490

For more information about executors, see **Executor elements**.

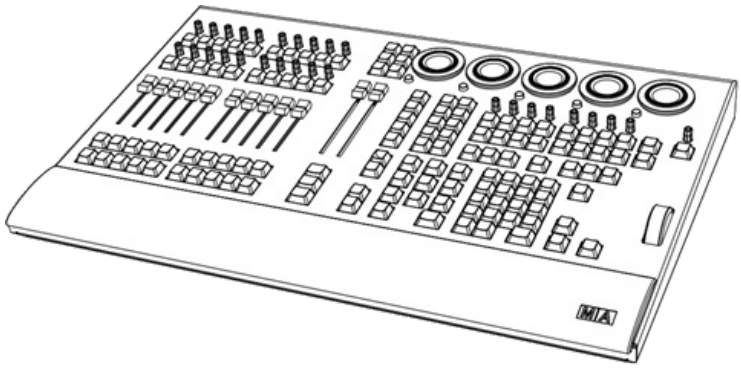


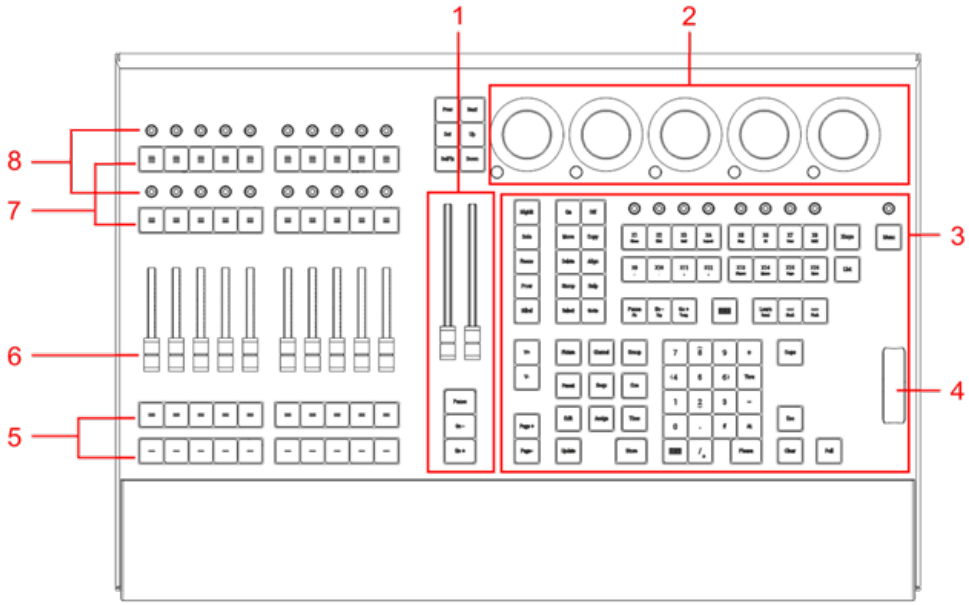
grandMA3 onPC command wing XT rear panel

1. **DMX A, B, C**
2. **MIDI In+Out**
3. **DC Remote Control**
4. **LTC**
5. Power switch
6. **IEC connector**
7. **Desk light**
8. Kensington lock
9. **Line In+Out**
10. **Ethernet 1+2**
11. **USB 2.0**
12. **USB 3.0**
13. **DisplayPort 1-4**

For technical specifications see **Technical Data** in the **grandMA3 Quick Manual onPC command wing XT**.

2.10. grandMA3 onPC command wing

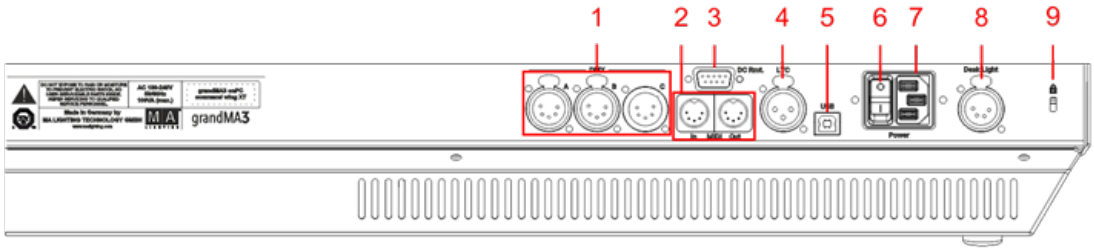




grandMA3 onPC command wing front panel

1. **Master area**
2. **Dual encoder section**
3. **Command area**
4. **Level wheel**
5. Executor buttons 101-190 + 201-290
6. Executor faders 201-290
7. Executor buttons 301-390 + 401-490
8. Executor knobs 301-390 + 401-490

For more information about executors, see **Executor elements**.

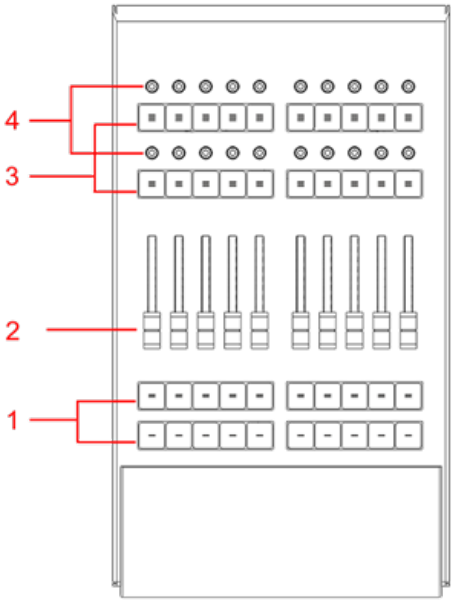


grandMA3 onPC command wing rear panel

1. **DMX A, B, C**
2. **MIDI In+Out**
3. **DC Remote Control**
4. **LTC**
5. USB
6. Power switch
7. **IEC connector**
8. **Desk light**
9. Kensington lock

For technical specifications see **Technical Data** in the **grandMA3 Quick Manual onPC command wing**.

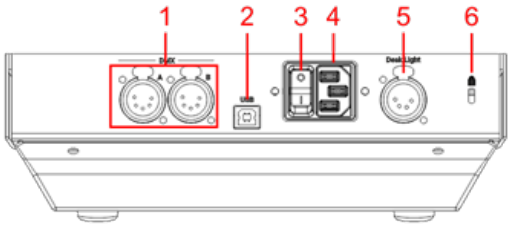
2.11. grandMA3 onPC fader wing



grandMA3 onPC fader wing front panel

1. Executor buttons 101-190 + 201-290
2. Executor faders 201-290
3. Executor buttons 301-390 + 401-490
4. Executor knobs 301-390 + 401-490

For more information about executors, see **Executor elements**.

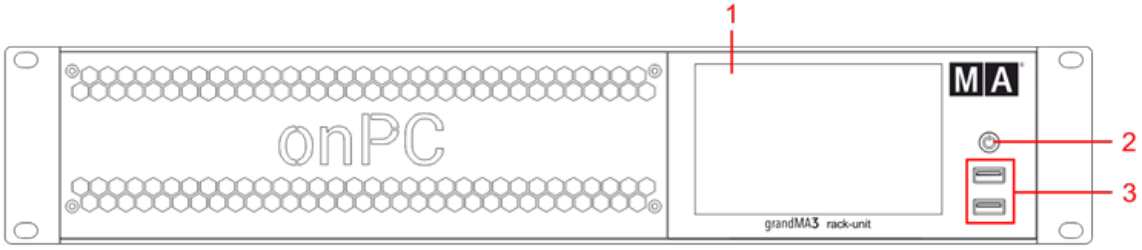


grandMA3 onPC fader wing rear panel

1. **DMX A+B**
2. USB
3. Power switch
4. **IEC connector**
5. **Desk light**
6. Kensington lock

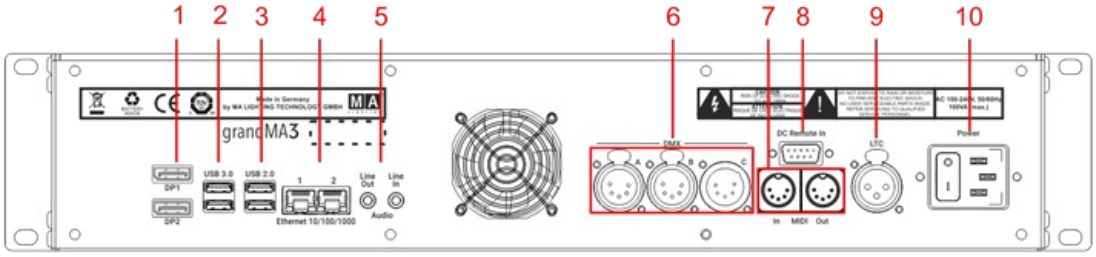
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

2.12. grandMA3 onPC rack-unit



grandMA3 onPC rack-unit front panel

1. Screen
2. **Power key**
3. **USB 2.0**



grandMA3 onPC rack-unit rear panel

1. **DisplayPort 1+2**
2. **USB 3.0**
3. **USB 2.0**
4. **Ethernet 1+2**
5. **Line In+Out**
6. **DMX A-C**
7. **MIDI In+Out**
8. **DC Remote Control**
9. **LTC**
10. Power switch + **IEC connector**

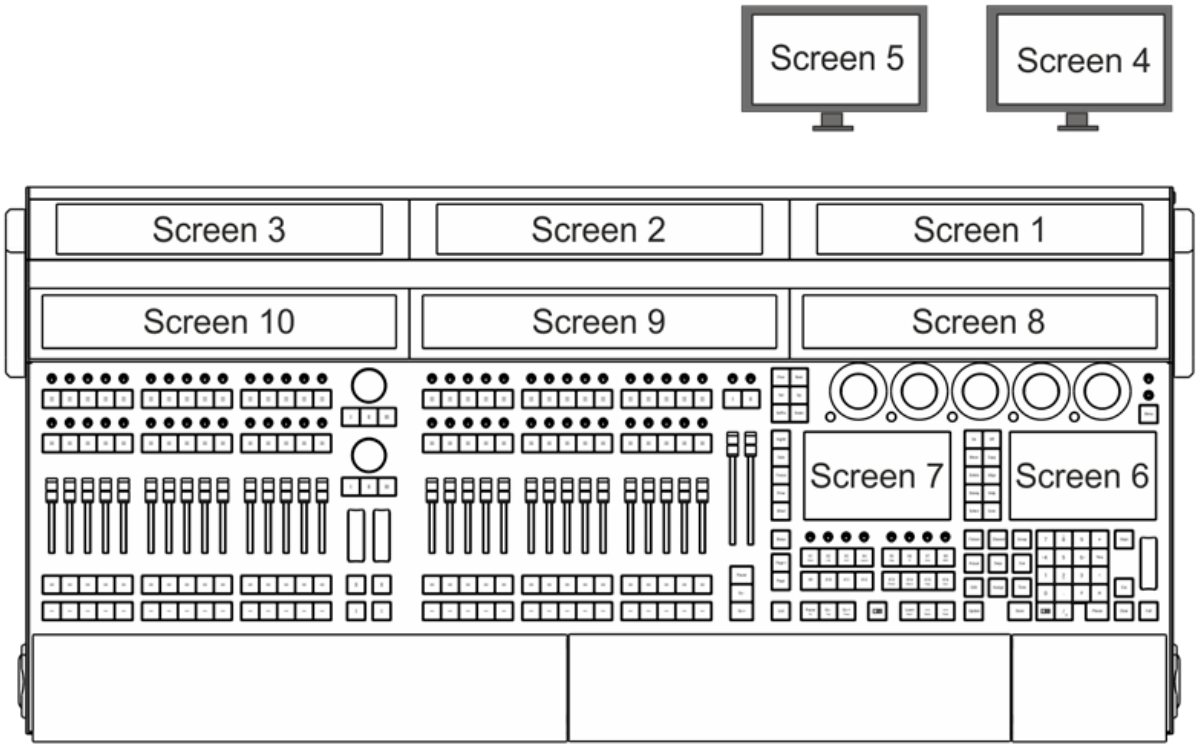
For technical specifications see **Technical Data** in the **grandMA3 Quick Manual onPC rack-unit**.

2.13. Screen Allocation

Numerical Order of Screens

grandMA3 consoles number their screens in a specific order. Different consoles within the range have different amounts of internal screens and connections for external monitors. This topic covers those physical differences as well as the numbering system applied to those screens.

grandMA3 full-size and light



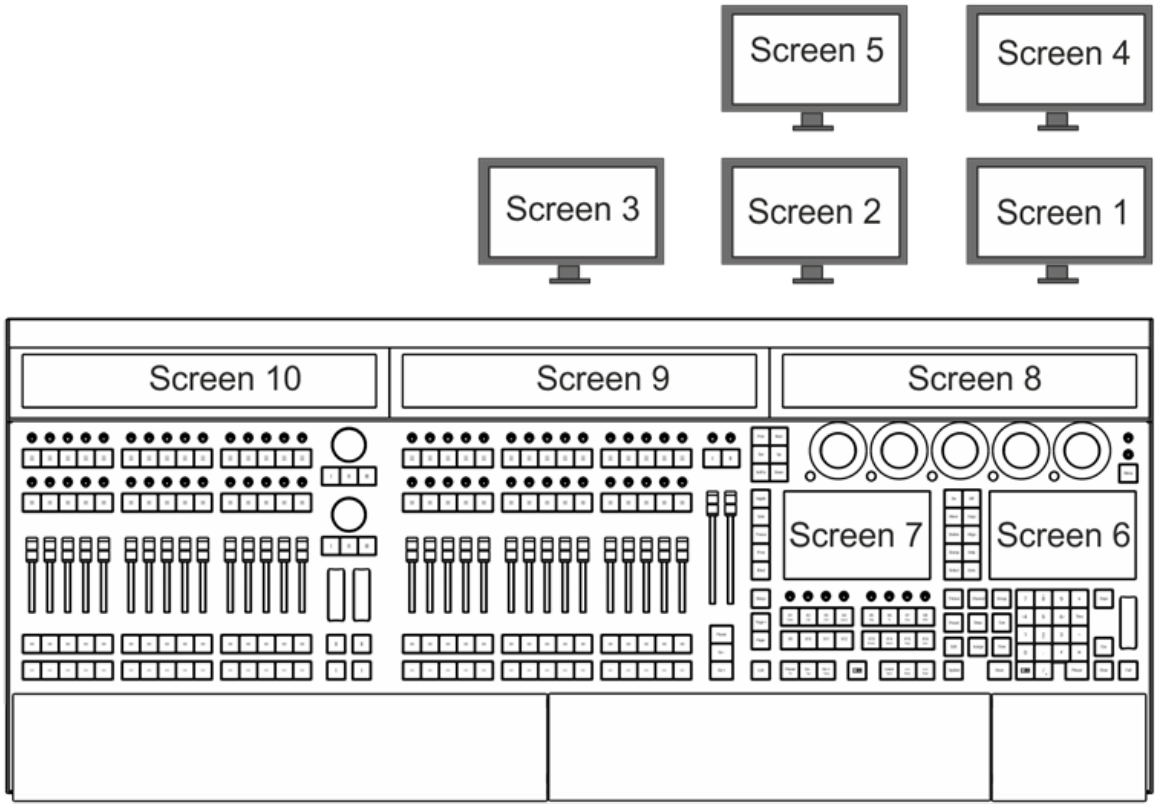
grandMA3 full-size with all screens numbered

The grandMA3 full-size includes three internal 15.6-inch screens mounted in a double-hinged monitor wing, three internal 14.9-inch letterbox screens, two internal 7-inch screens, and two connections for optional external monitors. The grandMA3 light includes two internal 15.6-inch screens mounted in a double-hinged monitor wing, two internal 14.9-inch letterbox screens, two internal 7-inch screens, and two connections for optional external monitors. The numbering of these screens is detailed in the following table:

Screen number	Size	Remark
Screen 1	15.6-inch	
Screen 2	15.6-inch	

Screen 3	15.6-inch	Only on grandMA3 full-size
Screen 4		External screen
Screen 5		External screen
Screen 6	7-inch	Right command screen
Screen 7	7-inch	Left command screen
Screen 8	14.9-inch	Letterbox encoder screen
Screen 9	14.9-inch	Right letterbox executor screen
Screen 10	14.9-inch	Left letterbox executor screen, on grandMA3 full-size or first grandMA3 extension connected to grandMA3 light
Screen 11	14.9-inch	Letterbox executor screen, on grandMA3 extension connected to grandMA3 full-size or second grandMA3 extension connected to grandMA3 light

grandMA3 full-size CRV and light CRV

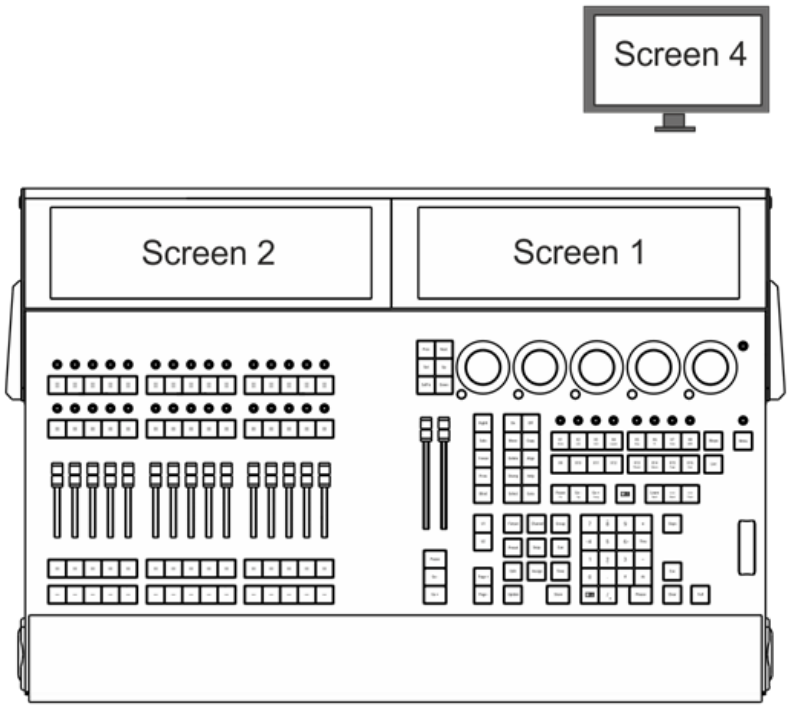


grandMA3 full-size CRV with all screens numbered

The grandMA3 full-size CRV includes three internal 14.9-inch letterbox screens, two internal 7-inch screens, and five connections for optional external monitors. The grandMA3 light CRV includes two internal 14.9-inch letterbox screens, two internal 7-inch screens, and four connections for optional external monitors. The numbering of these screens is detailed in the following table:

Screen number	Size	Remark
Screen 1		External screen
Screen 2		External screen
Screen 3		External screen (only grandMA3 full-size CRV)
Screen 4		External screen
Screen 5		External screen
Screen 6	7-inch	Right command screen
Screen 7	7-inch	Left command screen
Screen 8	14.9-inch	Letterbox encoder screen
Screen 9	14.9-inch	Right letterbox executor screen
Screen 10	14.9-inch	Left letterbox executor screen, on grandMA3 full-size CRV or grandMA3 extension connected to grandMA3 light CRV
Screen 11	14.9-inch	Letterbox executor screen, on grandMA3 extension connected to grandMA3 full-size CRV or second grandMA3 extension connected to grandMA3 light CRV

grandMA3 compact XT and compact

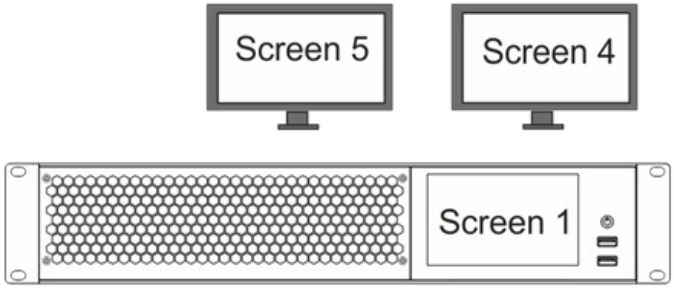


grandMA3 compact XT with all screens numbered

The grandMA3 compact XT includes two internal 15.6-inch screens mounted in a hinged monitor wing and one connection for an optional external monitor. The grandMA3 compact includes one internal 15.6-inch screen mounted in a hinged monitor wing and one connection for an optional external monitor. The numbering of these screens is detailed in the following table:

Screen number	Size	Remark
Screen 1	15.6-inch	
Screen 2	15.6-inch	Only on grandMA3 compact XT
Screen 4		External screen

grandMA3 replay unit



grandMA3 replay unit with all screens numbered

The grandMA3 replay unit includes two connections for optional external monitors. The numbering of these screens is detailed in the following table:

Screen number	Size	Remark
Screen 4		External screen
Screen 5		External screen
Screen 10	14.9-inch	Letterbox executor screen on first grandMA3 extension connected to grandMA3 replay unit
Screen 11	14.9-inch	Letterbox executor screen on second grandMA3 extension connected to grandMA3 replay unit
Screen 12	14.9-inch	Letterbox executor screen on third grandMA3 extension connected to grandMA3 replay unit

For more information on connecting external screens, see the **Connect External Screens** topic.

---

grandMA3 onPC rack-unit and grandMA3 onPC command wing XT

For onPC stations, the single windows for each display can be freely arranged. For more information, see the **Displays in grandMA3 onPC** topic.

2.14. Keyboard Shortcuts

Keyboard shortcuts allow fast operation of the console and onPC command wings via the (internal) keyboard.

Some of these user-editable shortcuts are for general use and override other commands. Others are related to specific windows and pop-ups.

The **KeyboardShortcut keyword** is used to edit and list the keyboard shortcuts. To modify the keyboard shortcuts, use the **Set keyword**.

Keyboard Shortcuts are part of the UserProfile. This makes it possible for every user/user profile to use their own shortcut definitions within a show file.

Turn the Keyboard Shortcuts on

- Tap ShCuts next to the command line or press F10 on your keyboard.

Keyboard shortcuts enabled

The keyboard shortcuts are turned on (yellow text).

Turn the Keyboard Shortcuts off

- Tap ShCuts next to the command line or press F10 on your keyboard.

Keyboard shortcuts disabled

The keyboard shortcuts are turned off (white text).

Hint:
In certain overlays and pop-ups where the shortcuts are not relevant, they are switched off and back on automatically. The shortcut symbol turns red.

Shortcut Examples

Command section window left

Command section window right

Playback controls window

The images above show the activated shortcut overlay. The overlay color can be changed in color themes.

Keyboard Shortcuts Window

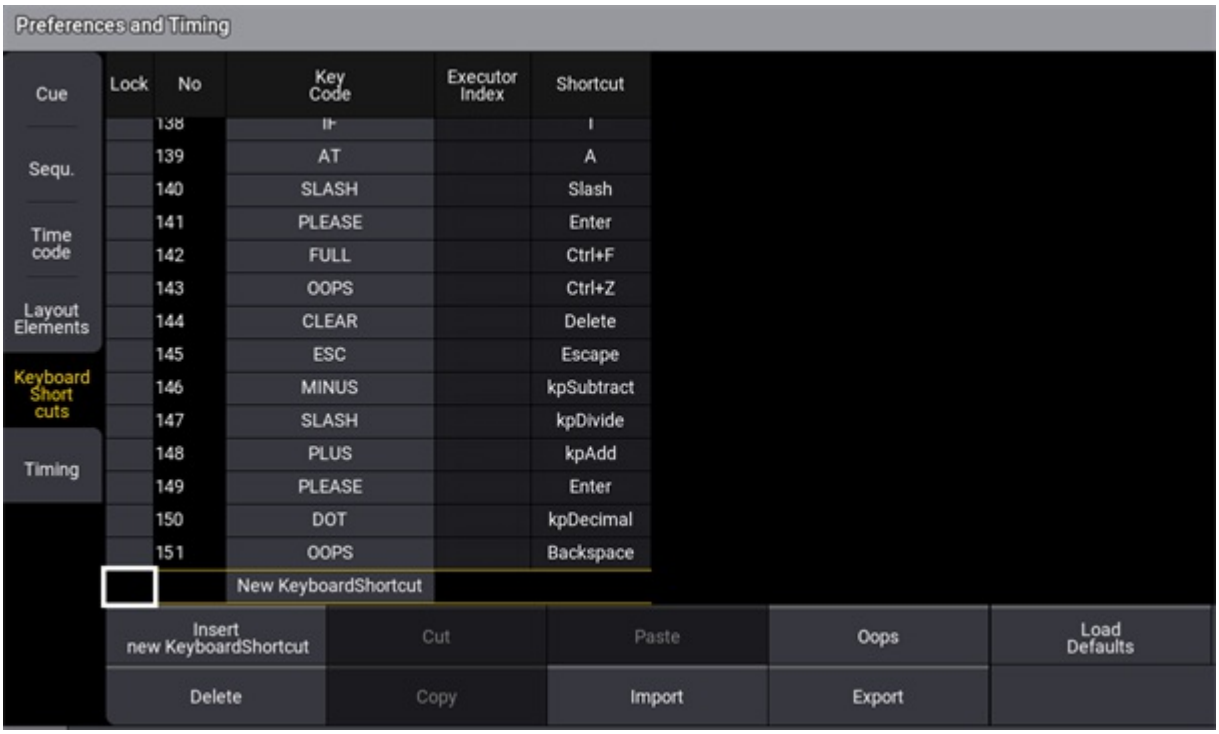
- To edit the shortcuts, open the Preferences and Timing window.
- Tap Keyboard Shortcuts.

The Keyboard Shortcuts window opens:

Edit Keyboard Shortcuts window

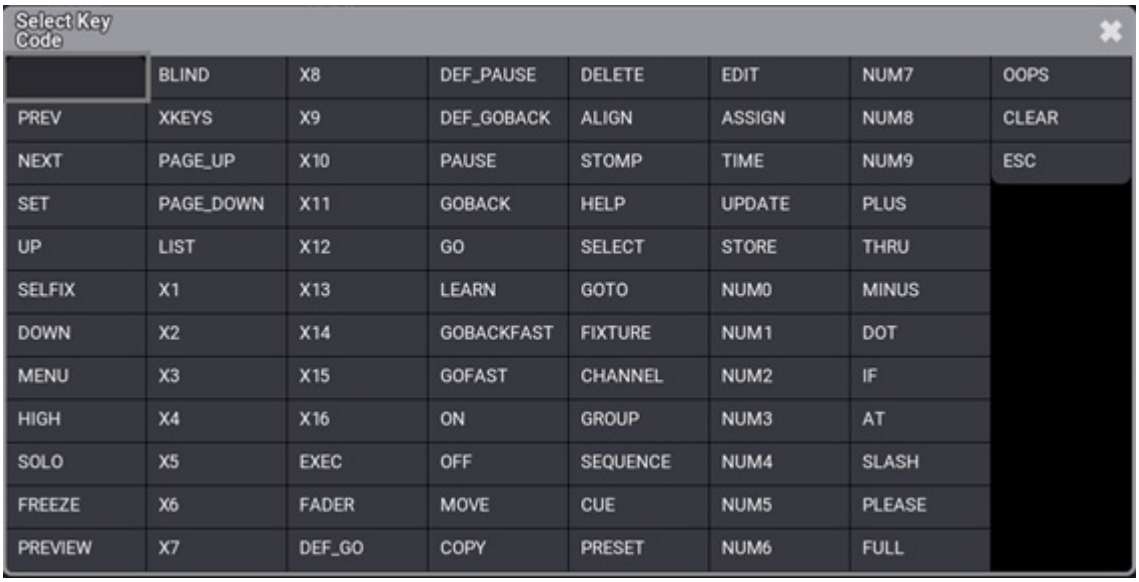
Add Keyboard Shortcuts

- To add a new keyboard shortcut, scroll down to the end of the list and tap Insert new KeyboardShortcut.



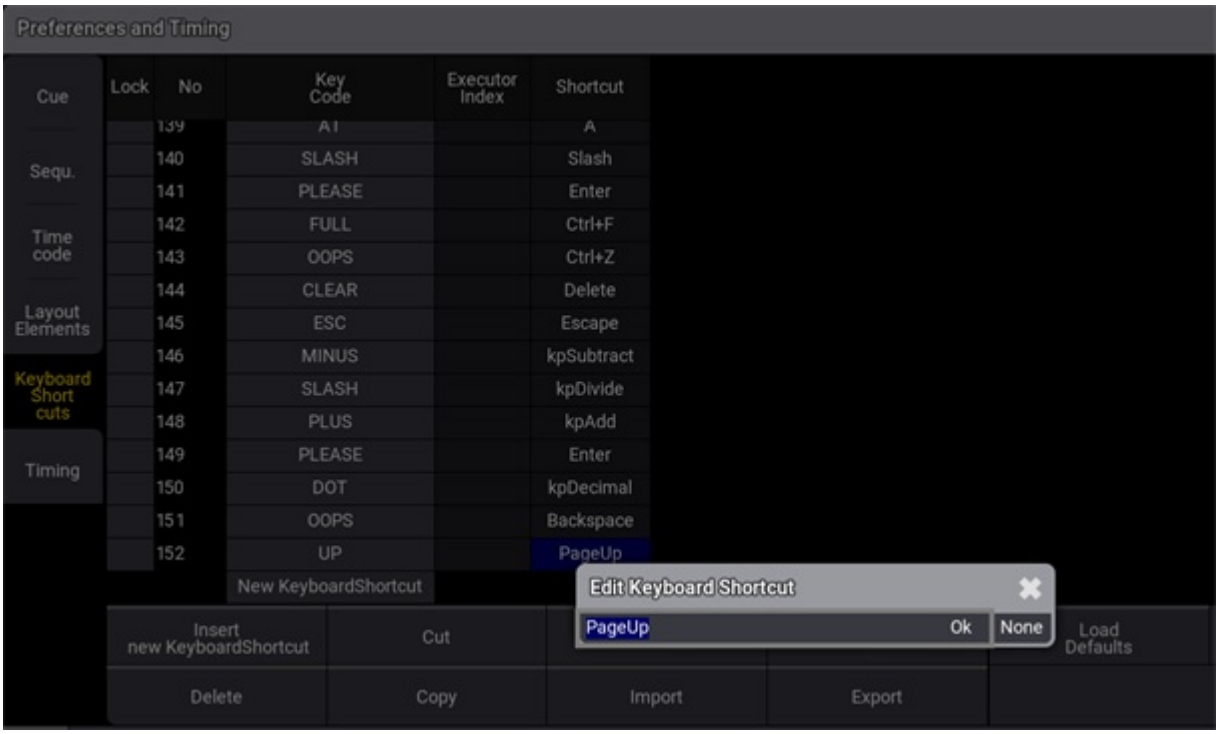
Add new keyboard shortcut

- Select the desired key code from the pop-up list.



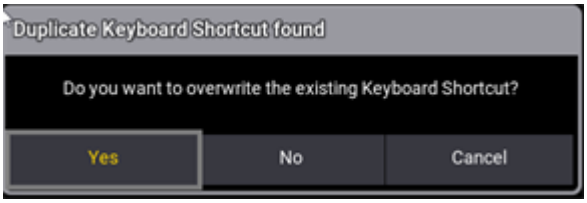
Select key code window

- To enter the desired shortcut, press the relevant key.



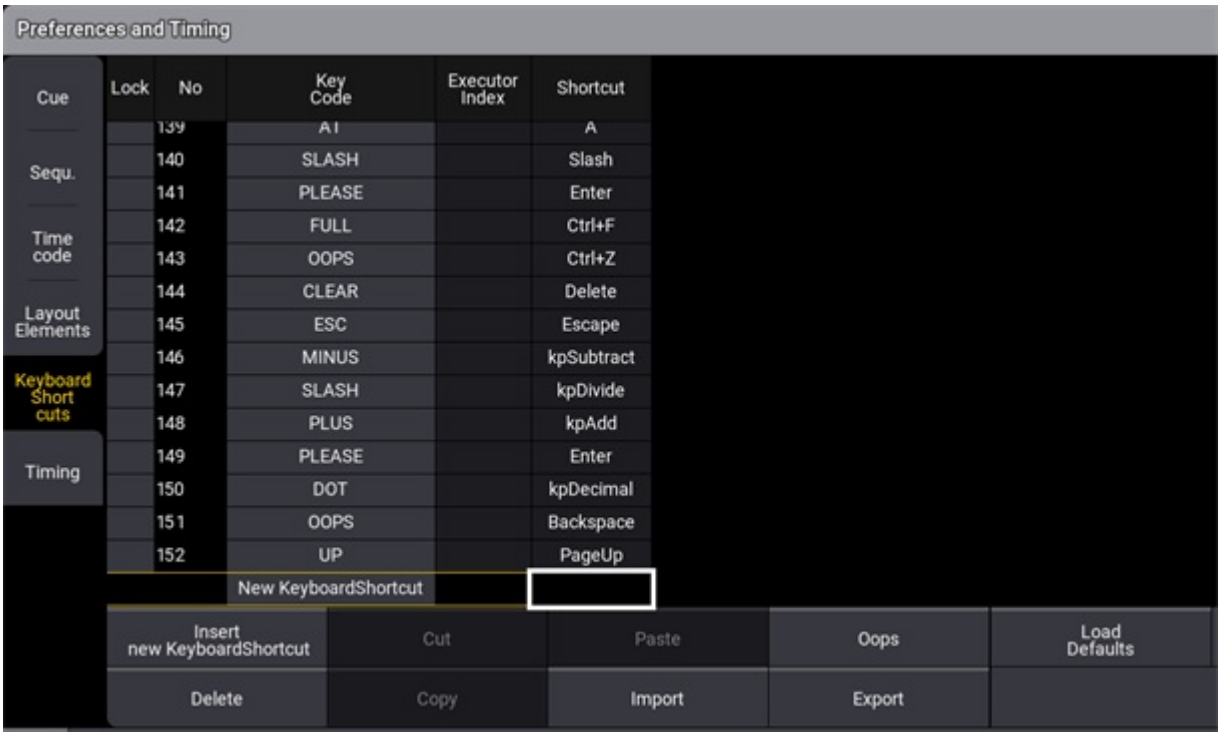
Enter keyboard shortcut window

If a keyboard shortcut already exists, a warning pop-up appears:



Warning pop-up

- Tap Yes to save the new shortcut.



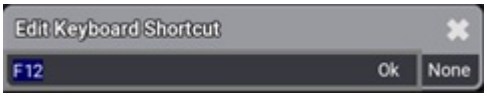
New shortcut is added.

The new shortcut is added to the list.

Edit Existing Keyboard Shortcut

- To edit an existing shortcut definition, long press the Keyboard Shortcut you want to edit.

The Edit keyboard shortcut pop-up opens



Edit keyboard shortcut pop-up

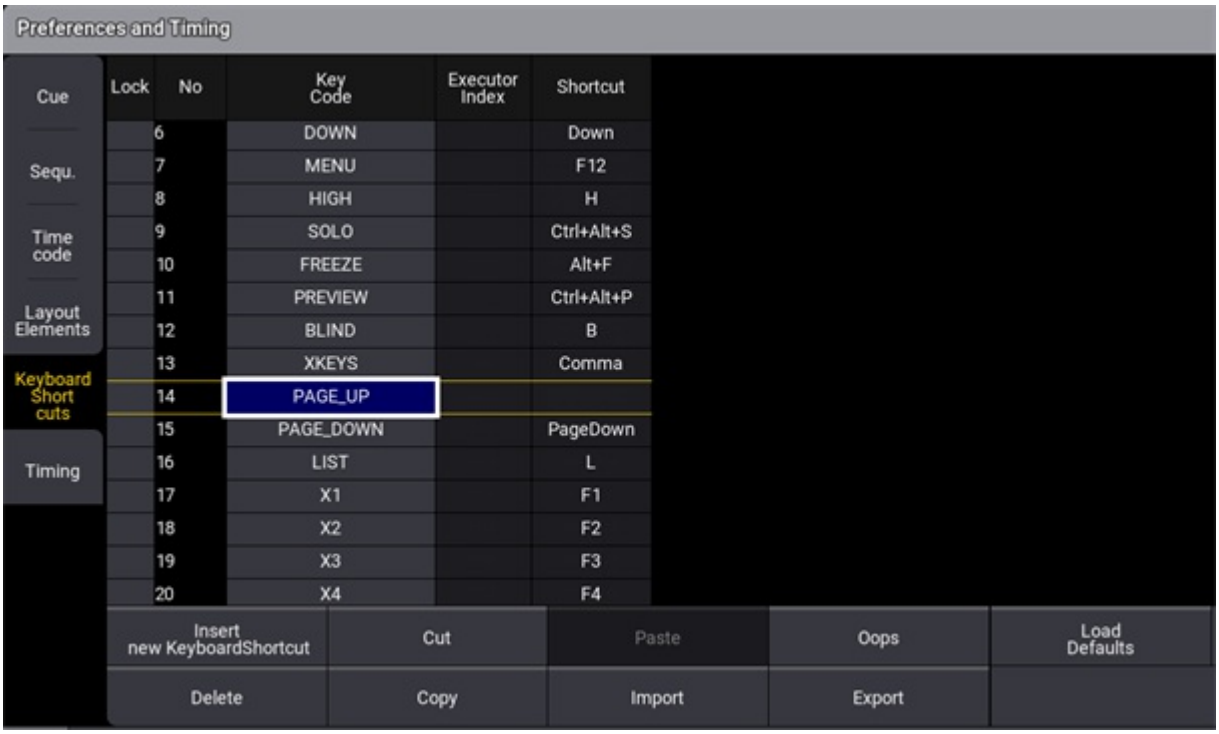
- Enter the new key.

As Enter can also be used as a shortcut, it is not possible to close the Edit keyboard shortcut pop-up with Enter.

- To confirm the shortcut tap Ok.
- To clear the shortcut, tap None.
- To cancel the edit, tap X.

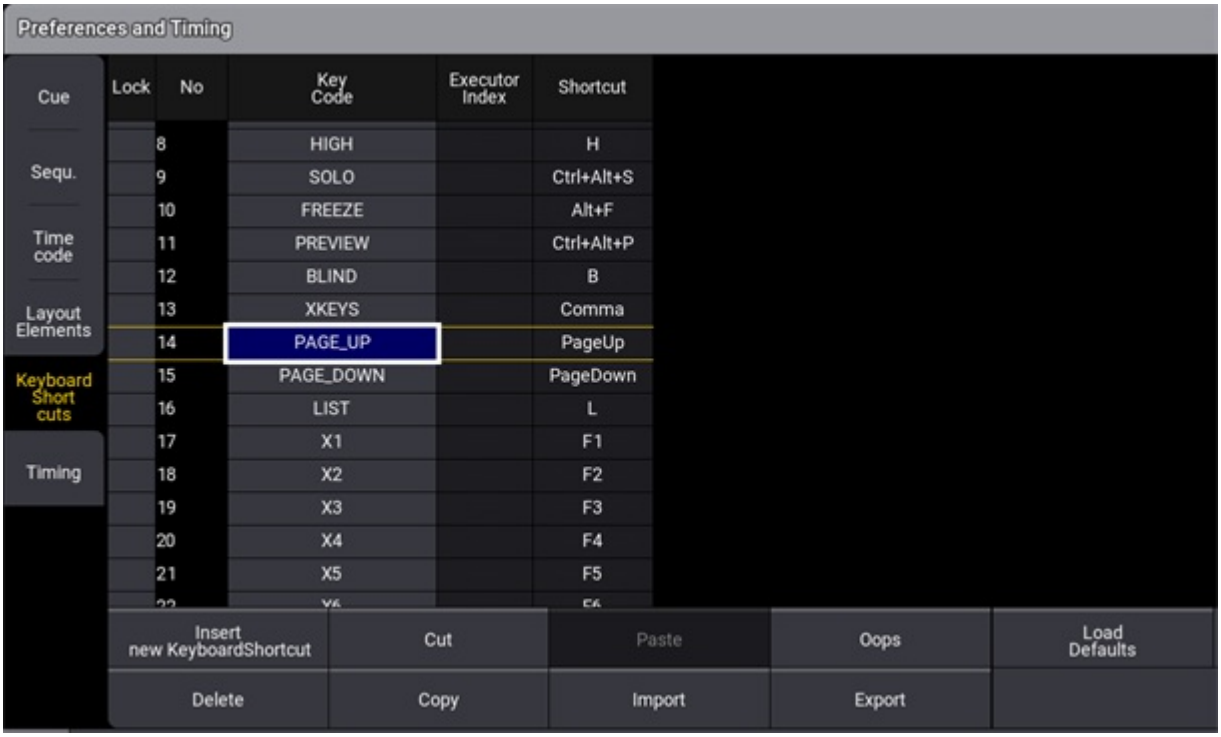
Delete Keyboard Shortcuts

- To select the keyboard shortcut to be deleted, tap the desired shortcut.



Delete keyboard shortcut window

Tap Delete.



Keyboard shortcut deleted.

The keyboard shortcut is deleted.

Reset Keyboard Shortcuts to Defaults

To reset the keyboard shortcuts to their defaults, tap Load Defaults.

The keyboard shortcuts are reset to their defaults.

Hint:
The Shift keys on the keyboard correspond with the MA keys on the console.

Import / Export Keyboard Shortcuts

To import or export the keyboard shortcuts, tap Import or Export.

By default, files will be exported to the relevant folder within the library folder structure, either on the local drive of the console or onPC station, or on a selected USB drive. For more information about this folder structure, see the **Folder Structure** topic.

For information about exporting and importing show data using command line syntax without the use of these menus, see the **Export keyword** and **Import keyword** topics.

2.15. Keys

A shortcut is a series of one or several keys that invoke the software to perform a pre-programmed action.

In grandMA3, we differentiate between the term key and button.

A key is a (hard) key on the console or on the (integrated) keyboard.

A button is an element in the User Interface.

The keys F9, F10, F11, Pause and Print Screen provide system-wide shortcuts on all operating systems always available in all contexts as long as the program is running. These shortcuts cannot be edited.

The following shortcuts are available on all OS through several input methods:

Desk lock:

- Keyboard
  - F9
  - Pause
- Key
  - MA + MA + Pause|Fix
- Button
  - Lock

Toggle keyboard shortcuts:

- Keyboard
  - F10
- Button
  - ShCuts

Print screen:

- Keyboard
  - F11
  - Print

Reload UI (If the color theme is set to unreadable values):

- Key
  - MA + MA + Clear

Hint:
The Shift keys on the keyboard correspond with the MA keys on the console.

All other shortcuts are user-editable.

This overview displays shortcuts for a quick execution of commands.

Press key	Command	Description
.	.	Sets a dot in the command line.
. .	Zero	Sets the selected fixtures dimmer value to 0.
MA .	Default	Sets the attributes to its default values.
<<<	<<<	Goes fast backward and ignores the cue timing on a given executor.
MA <<<	Black	Temporarily overrides the master level to zero on executing objects.
>>>	>>>	Goes fast forward and ignores the cue timing on a given executor.
MA >>>	Flash	Temporarily overrides the master level to full on a given executor.
-	-	Sets a minus in the command line.
- -	At -10	Decreases the intensity of the selected fixtures by 10.
+	+	Adds a plus to the command line.
+ +	At +10	Increases the intensity of the selected fixtures with +10.
/	/	Sets a slash in the command line.
MA /	*	Sets an asterisk in the command line.
At	At	Sets the values on the selected fixtures.
At At	Normal	Sets the intensity to Normal on the selected fixtures.
MA At	Integrate	Integrates preset values into another step of the phaser.
MA At At	Extract	Breaks any referenced links and applies the hard values.
Press and hold At		Opens the At menu (At filter overlay).
Store At	/Embed	With store in the command line, at adds /Embed to the command.
Assign	Assign	Assigns functions or objects to other objects.
Assign Assign	Label	Labels objects.
Assign Assign Assign	Set	Sets values to object properties.
Align		Cycles through the different align modes.

Press key	Command	Description
Press and hold Align		Sets the align mode to off.
MA Align		Cycles through the different align transitions.
Press and hold MA Align		Sets the align transition to linear.
Blind	**Blind**	Switches blind mode on and off.
Clear	**Clear**	Clears the selected fixtures.
Clear Clear	**Clear**	Clears the active values.
Clear Clear Clear	**Clear**	Clears the programmer.
Press and hold Clear	**ClearAll**	Clears the programmer + command filter. The same as 3 x Clear.
Channel	**Channel**	Selects fixtures by its Channel ID's.
Channel Channel		Cycles through the custom made ID types.
Copy	**Copy**	Creates a copy of an object.
Copy Copy	**Paste**	Pastes previously copied or cut objects.
Copy Copy Copy	**Insert**	Inserts previously copied or cut objects.
MA Copy	**Cut**	Cuts content to the clipboard.
Ctrl		Press and hold for multi-selection in sheets. Press and hold when the Mode Selector pop-up appears during the boot process for clean start.
Cue	**Cue**	Cue is an object type.
Cue Cue	**Part**	Part is an object type.
MA Cue	**Programmer**	Enters the Programmer keyword in the command line.
Down		Navigates down in the fixture/subfixture structure.
Delete	**Delete**	Use Delete to remove something from the show file.
Delete Delete	**Remove**	Loads Remove in the programmer.
Delete Delete Delete	**Release**	Loads Release in the programmer.
ESC		Closes pop-ups, returning keyboard focus to the command line. If there are no open pop-ups, ESC deletes unexecuted commands from the command line.
Edit	**Edit**	Opens an editor for objects like cues, presets, sequences, and macros.
Edit Edit	**EditSetting**	Adds EditSetting into the command line.
Freeze	**Freeze**	Toggles the Freeze function on and off.
Full	**Full**	Sets the intensity for the selected fixtures to 100%.
Fixture	**Fixture**	Selects the fixtures of a given ID.
Fixture Fixture	**Selection**	The actual programmer selection.

Press key	Command	Description
Group	**Group**	Selects a group of a given ID.
Group Group	**World**	Selects a world of a given ID.
Group Group Group	**Filter**	Selects a filter of a given ID.
Goto	**Goto**	Goes to a cue in the selected sequence.
Goto Goto	**Load**	Loads a cue from the selected sequence.
MA Goto Goto	**Loaded**	Adds Loaded to the command line.
Go+ Go+	**Unpark**	Releases a parked DMX channel or universe.
Go+ [Large]	**Go+ Executor**	Executes the next cue in the selected sequence.
MA Go+ [Large]	**Go+ Loaded**	Executes all the loaded cues at the same time.
Go- [Large]	**Go- Executor**	Executes the previous cue in the selected sequence.
Go+|Temp	**Go+**	Executes the next cue in the sequence.
MA Go+|Temp	**Temp**	Enables an executor only as long as that executor is held.
MA Go+|Temp Go+|Temp	**Toggle**	Toggles between on and off status.
Go-|Top	**Go-**	Executes the previous cue in the sequence.
MA Go-|Top	**Top**	Executes the first cue in the sequence.
Help	**Help**	Displays the help topic of a key or view.
HighIt	**Highlight**	Toggles the highlight function on and off.
MA HighIt	**Lowlight**	Toggles the lowlight function on and off.
If	**IfOutput**	Selects fixtures based on their current output.
If If	**IfActive**	Selects fixtures with active values in the programmer.
If If If	**IfProg**	Selects fixtures that contain values in the programmer.
If If If If	**If**	Selects fixtures that exist in both groups.
Learn|Rate1	**Learnrate**	Learn speed for a chaser or a phaser.
MA Learn|Rate1	**Rate1**	Sets the rate to 1:1 for the executor or speed group.
List	**List**	Lists the sub-directories in the command line.
MA List	**ListRef**	Lists references and dependencies of an object.
MA		Pressing and holding MA in combination with other keys, gives shortcuts to other functions. Same as Shift on a keyboard.
MA Please		Forces the focus to the command line.
Menu		Opens the menu pop up.
Menu Menu	**SaveShow**	Saves the show file using the current name.
Move	**Move**	Moves objects to another location.
Move Move	**Exchange**	Function keyword used to swap places.
Next	**Next**	Selects the next fixture.

Press key	Command	Description
MA Next		Selects the next step.
Oops	**Oops**	Undoes the last operation or selection.
On	**On**	Starts an executor or activates attributes.
Off	**Off**	Releases an executor or deactivates attributes.
Off Off		Opens the off menu.
MA MA Off		Closes any active RemoteHID connections.
Pause [Large]	**Pause Executor**	Toggles pause on and off for the selected sequence.
Pause|Fix	**Pause**	Pauses a crossfade or a phaser.
Pause|Fix Pause|Fix	**Park**	Enters the park keyword in the command line.
MA Pause|Fix	**Fix**	Fixes an executor for all executor pages.
MA MA Pause|Fix		Toggles the desk lock.
Page+	**Next Page**	Calls the next page.
Page-	**Previous Page**	Calls the previous page.
Press and hold Page-	**Page**	Calls page 1.
Please		Executes the syntax written in the command line.
Please Please		Activates all attributes tor the selected fixtures. Additional presses toggle this activation.
Power		Turns the console on and off.
Preset	**Preset**	Selects the preset of a given ID.
Preset Preset	**Attribute**	Selects the attribute of a given ID.
Preset Preset Preset	**Gel**	Selects the gel of a given ID.
MA Preset	**FeatureGroup**	Selects the feature group of a given ID.
MA Preset Preset	**DataPool**	Selects the data pool of a given ID.
Prev	**Previous**	Selects the previous fixture.
MA Prev		Selects the previous step.
Select	**Select**	Selects objects as defaults.
SelFix	**SelFix**	Selects all fixtures from an executor or another object.
Sequ	**Sequence**	Selects the sequence of a given ID.
Set	**Off Matricks**	Restores the selection. Toggles MATricks on and off.
Press and hold Set	**Reset Matricks**	Resets the MATricks.
MA Set	**Step Toggle**	Toggles between the first step selected and all steps selected.
Solo	**Solo**	Toggles the solo function on and off.

Press key	Command	Description
Stomp	**Stomp**	For selected parameters with a running phaser, Stomp the absolute layer to the default value and sets the relative layer to 0 in the programmer. All other steps are deleted from the programmer.
Stomp Stomp	**Capture**	Translates the combined instantaneous output of selected fixtures to the absolute layer of a single step in the programmer. All other steps will be deleted from the programmer.
Store	**Store**	Stores objects in the show file.
Press and hold Store		Opens the Store Options pop up.
Thru	**Thru**	Selects a range of items.
Time	**Time key**	Allows access to parameter and cue timing as well as relative and absolute value layers.
Up		Navigates up in the fixture/subfixture structure.
Update	**Update**	Opens the update pop up.
MA Update	**Cook**	Converts recipe data into value, timing, and/or phaser data.
X1|Clone		At the moment all Xkeys behave like executors. X1 is executor 291.
MA X1|Clone	**Clone**	Copies data from one fixture to another.
MA X1|Clone X1|Clone	**Recast**	Updates dependent objects when attributes are added to or deleted from presets.
X2|Link		At the moment all Xkeys behave like executors. X2 is executor 292.
X3|Grid		At the moment all Xkeys behave like executors. X3 is executor 293.
MA X3|Grid	**Grid**	Sets the grid cursor in the Selection Grid window.
X4|Layout		At the moment all Xkeys behave like executors. X4 is executor 294.
MA X4|Layout	**Layout**	Selects the layout of a given ID.
MA X4|Layout X4|Layout	**Appearance**	Puts some colors on buttons backgrounds and borders of a given object.
MA X4|Layout X4|Layout X4|Layout	**Scribble**	Small drawings on the background of objects.
X5|Step		At the moment all Xkeys behave like executors. X5 is executor 295.
MA X5|Step	**Step**	Selects the step of a given ID.
X6|TC		At the moment all Xkeys behave like executors. X6 is executor 296.

Press key	Command	Description
MA X6|TC	**Timecode**	Selects the timecode of a given ID.
X7|View		At the moment all Xkeys behave like executors. X7 is executor 297.
MA X7|View	**View**	Selects the view of a given ID.
MA X7|View X7|View	**ViewButton**	Selects the view button of a given ID.
MA X7|View X7|View X7|View	**ScreenContent**	Defines the size of a screen or has all the windows as content.
X8|DMX		At the moment all Xkeys behave like executors. X8 is executor 298.
MA X8|DMX	**DMXAddress**	Selects DMX addresses, using absolute numbering.
MA X8|DMX X8|DMX	**DMXUniverse**	Selects the DMX universe of a given value.
X9		At the moment all Xkeys behave like executors. X9 is executor 191.
X10		At the moment all Xkeys behave like executors. X10 is executor 192.
X11		At the moment all Xkeys behave like executors. X11 is executor 193.
X12		At the moment all Xkeys behave like executors. X12 is executor 194.
X13|Phaser		At the moment all Xkeys behave like executors. X13 is executor 195.
MA X13|Phaser	**Phaser Editor**	
X14|Macro		At the moment all Xkeys behave like executors. X14 is executor 196.
MA X14|Macro	**Macro**	Selects the macro of a given ID.
MA X14|Macro X14|Macro	**Plugin**	
X15|Page		At the moment all Xkeys behave like executors. X15 is executor 197.
MA X15|Page	**Page**	Selects the page of a given ID.
X16|Exec		At the moment all Xkeys behave like executors. X16 is executor 198.
MA X16|Exec	**Exec**	Selects the executor of a given ID.
MA X16|Exec X16|Exec	**SpecialExecutor**	Enters the SpecialExecutor keyword in the command line.
MA X16|Exec X16|Exec X16|Exec	**FaderMaster**	Enters the FaderMaster keyword in the command line.

2.15.1.. [Dot] key

Pressing . [Dot] enters a . into the command line.

Zero

Pressing . . enters the Zero keyword into the command line.

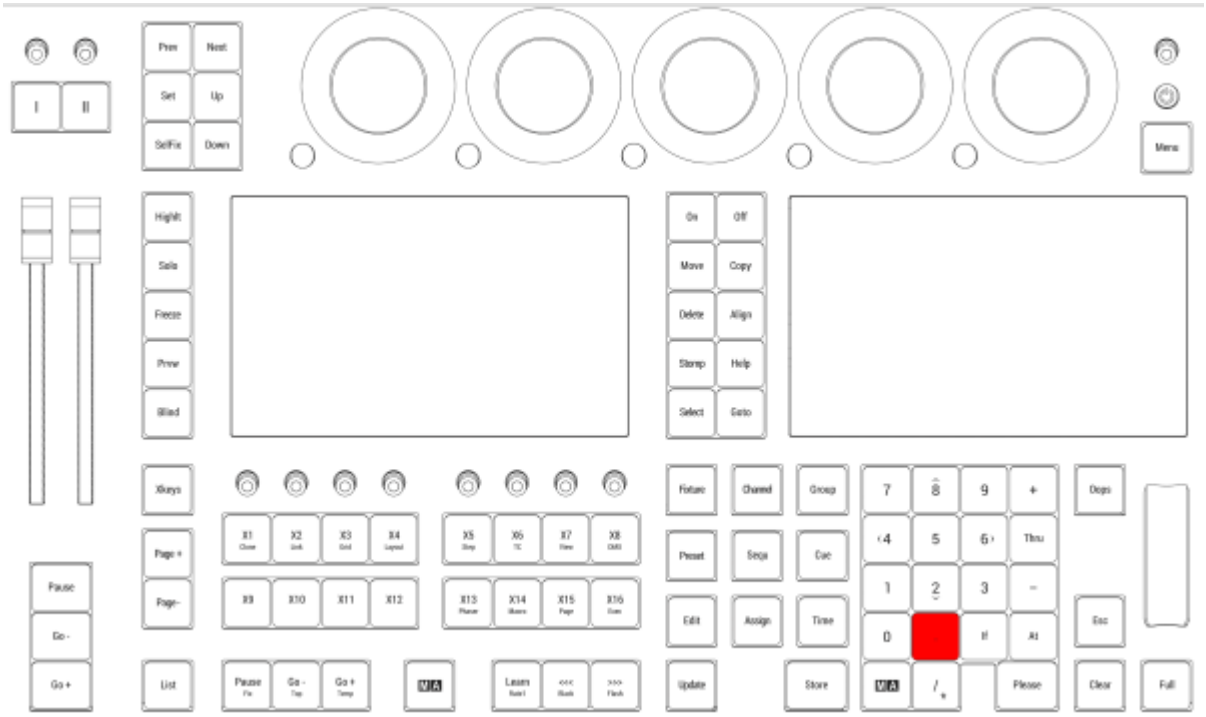
For more information about Zero, see the **Zero keyword** topic.

Default

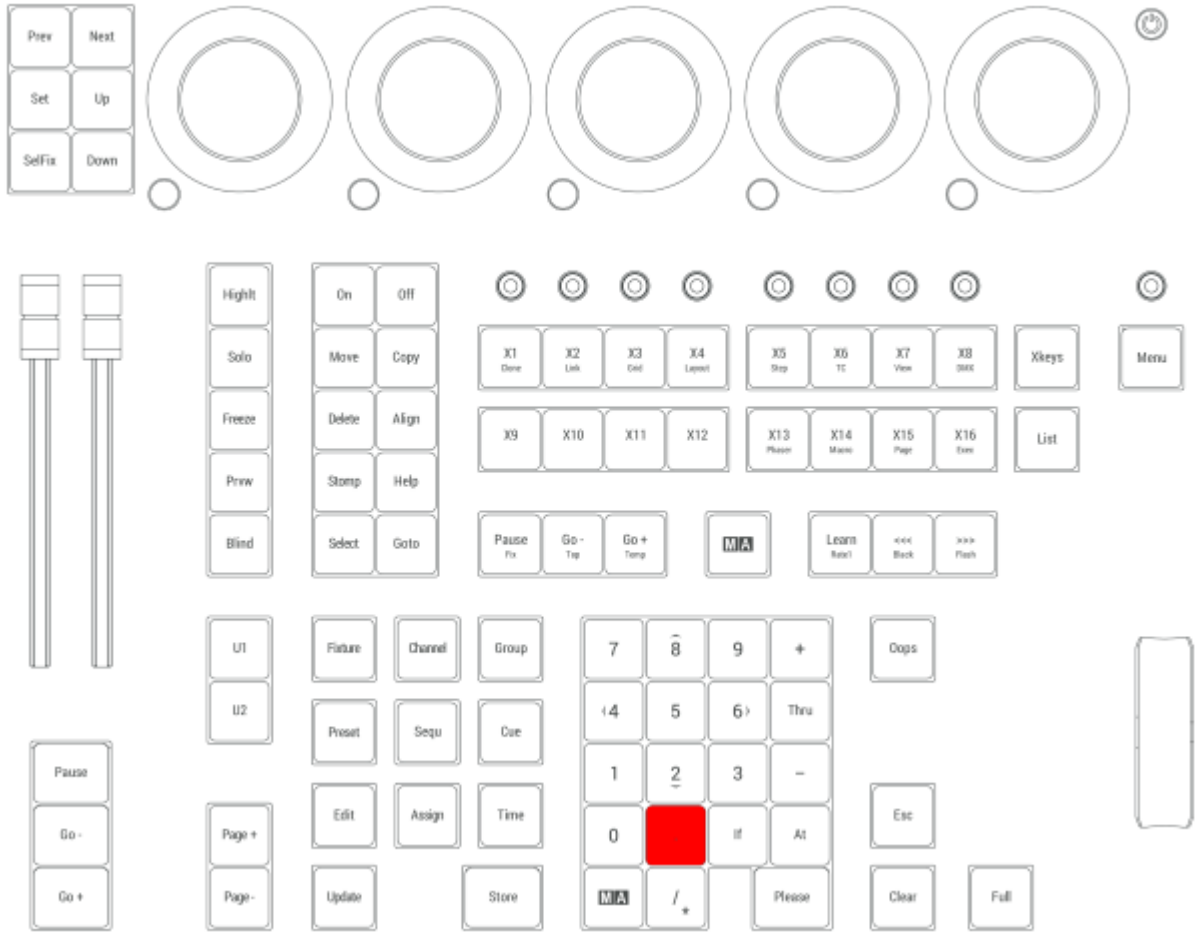
Pressing MA + . enters the Default keyword into the command line.

For more information about Default, see the **Default keyword** topic.

The . [Dot] is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.2. <<< [GoFastBackward] | Black key

Pressing <<< enters the <<< keyword into the command line.

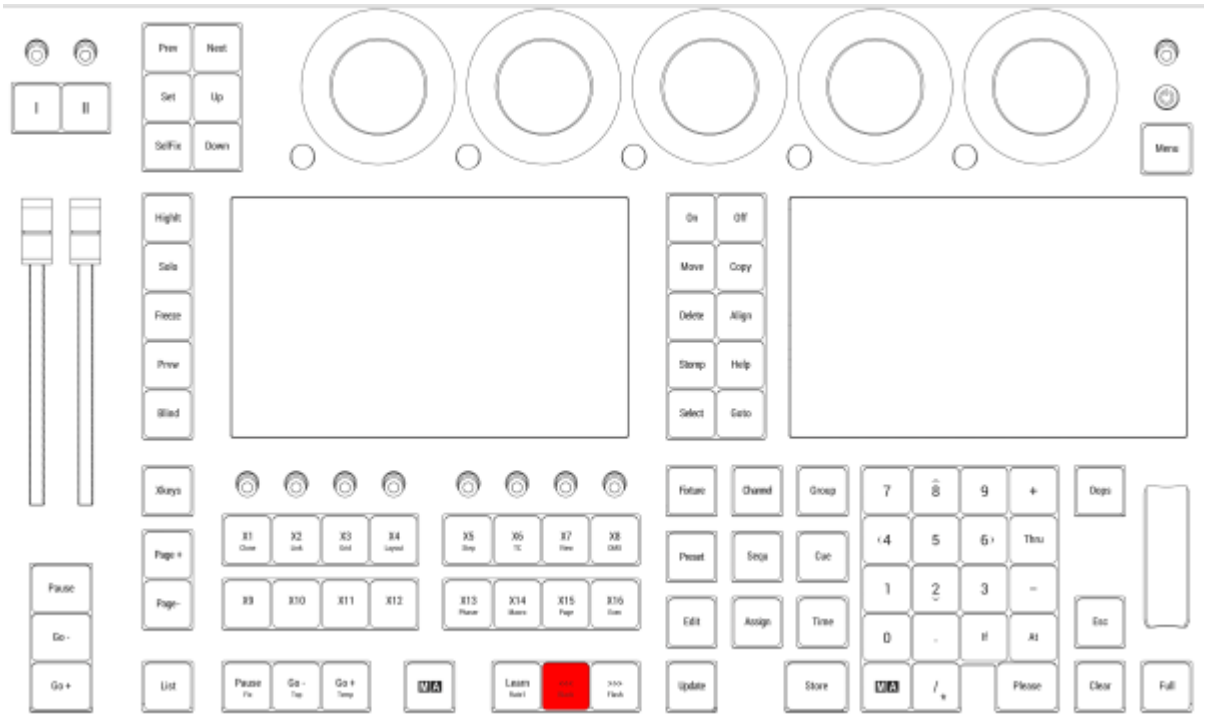
For more information about <<<, see the **GoFastBackward keyword** topic.

Black

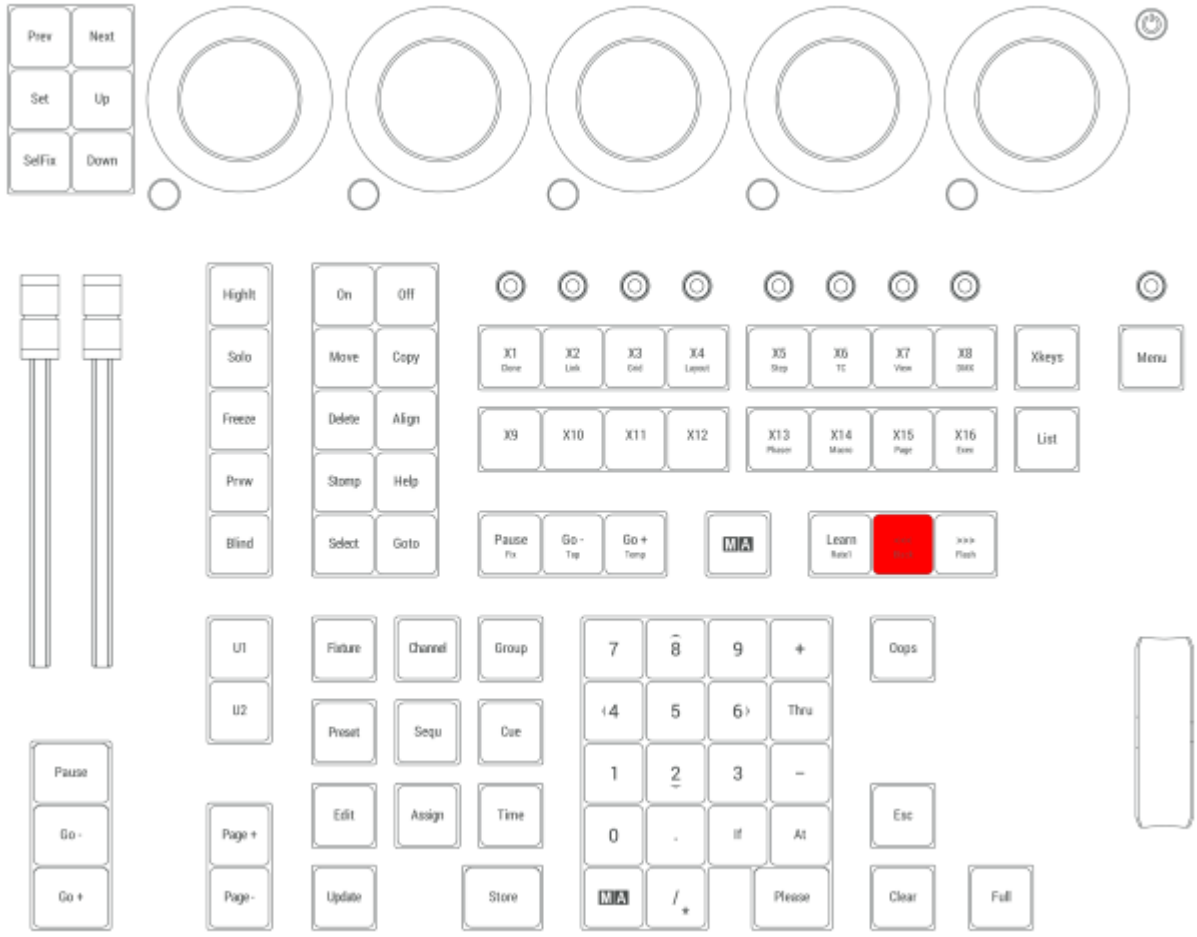
Pressing and holding MA + <<< enters the Black keyword into the command line.

For more information about Black, see the **Black keyword** topic.

<<< is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.3. >>> [GoFastForward] | Flash key

Pressing >>> enters the >>> keyword into the command line.

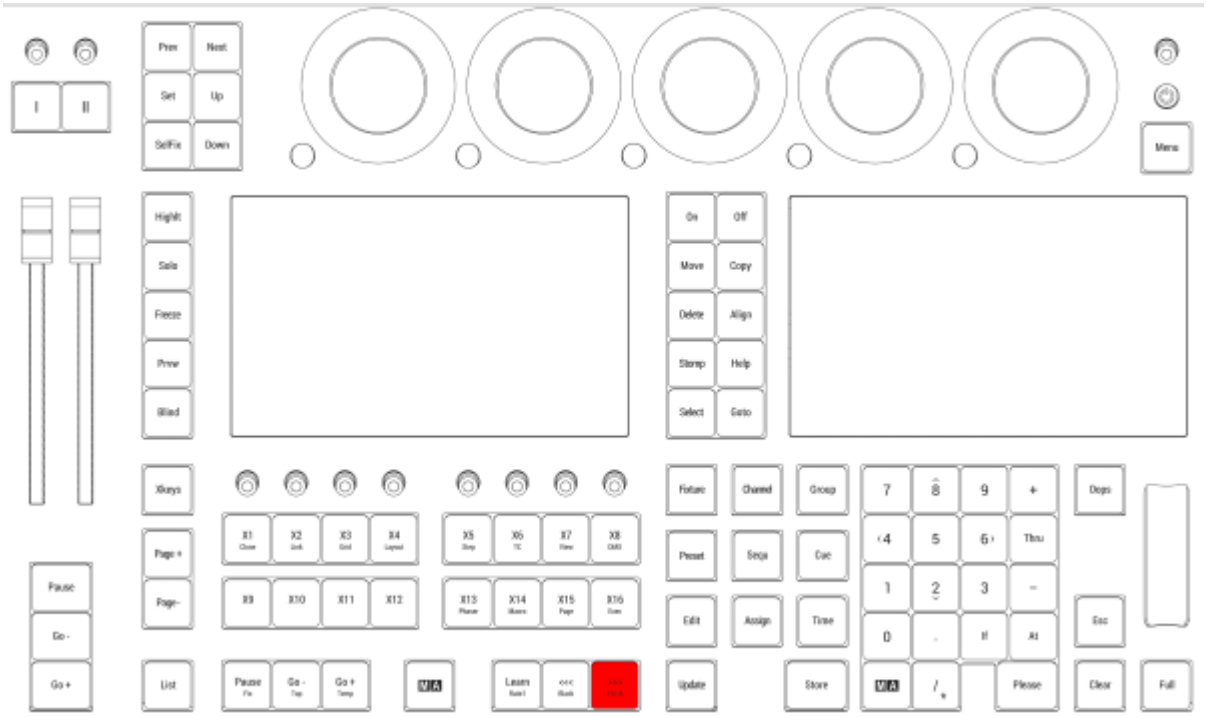
For more information about >>>, see the **GoFastForward keyword** topic.

Flash

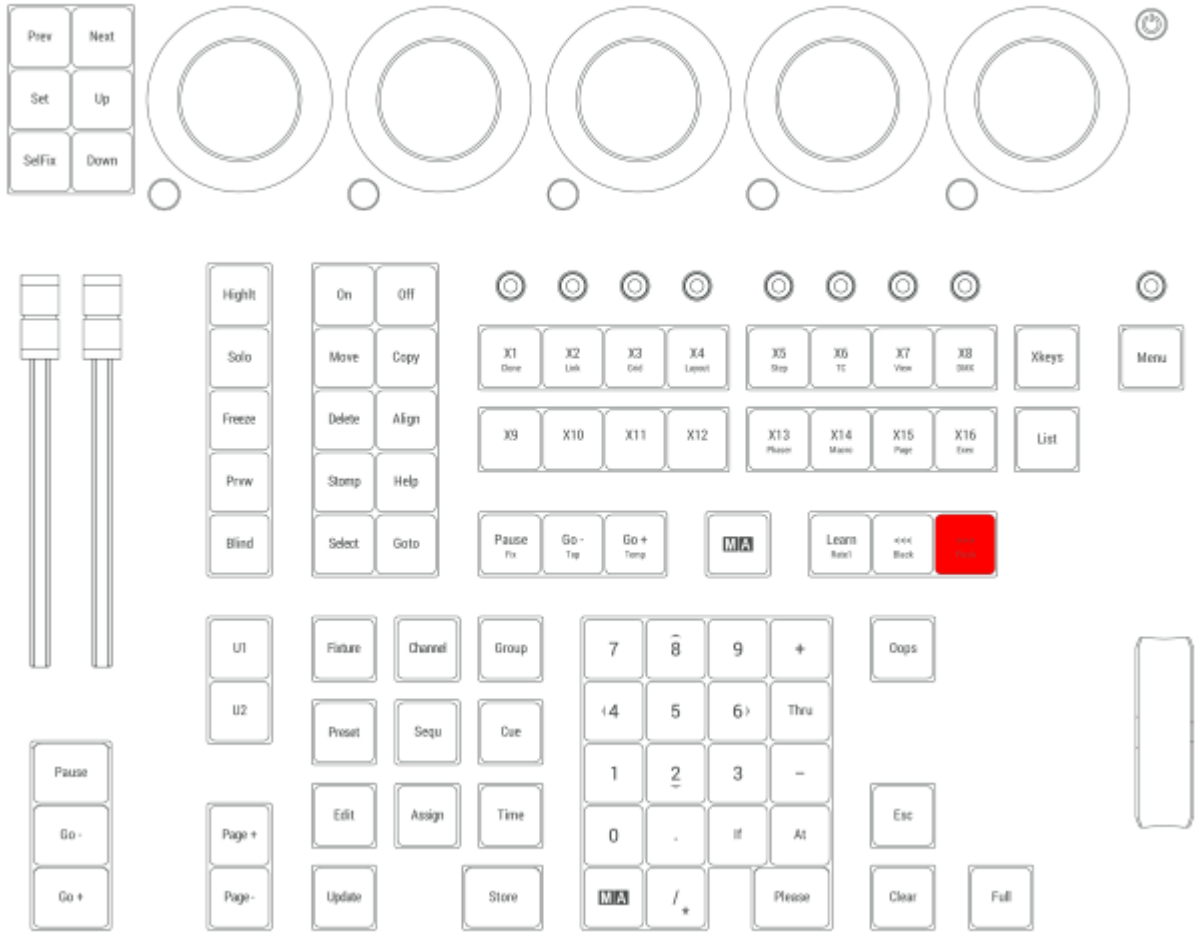
Pressing and holding MA + >>> enters the Flash keyword into the command line.

For more information about Flash, see the **Flash keyword** topic.

>>> is located in the command section.



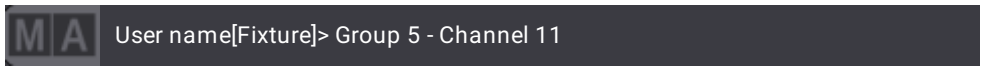
Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.4. - [Minus] key

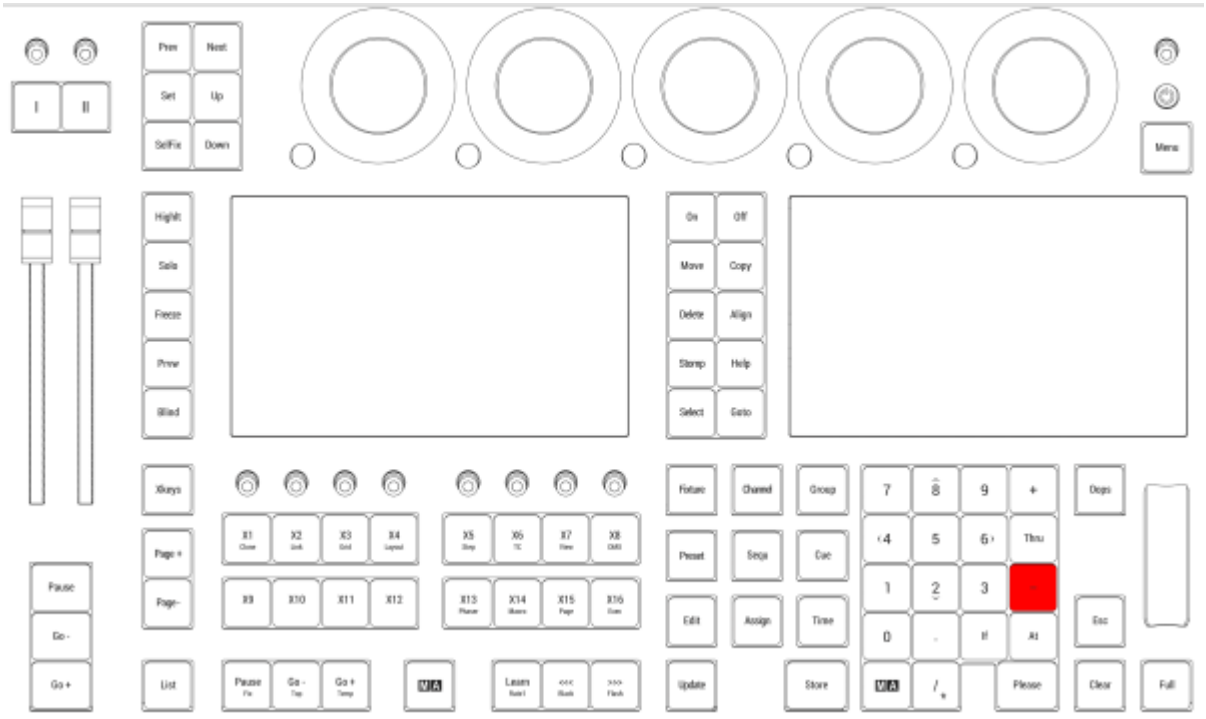
Pressing - enters a - into the command line.



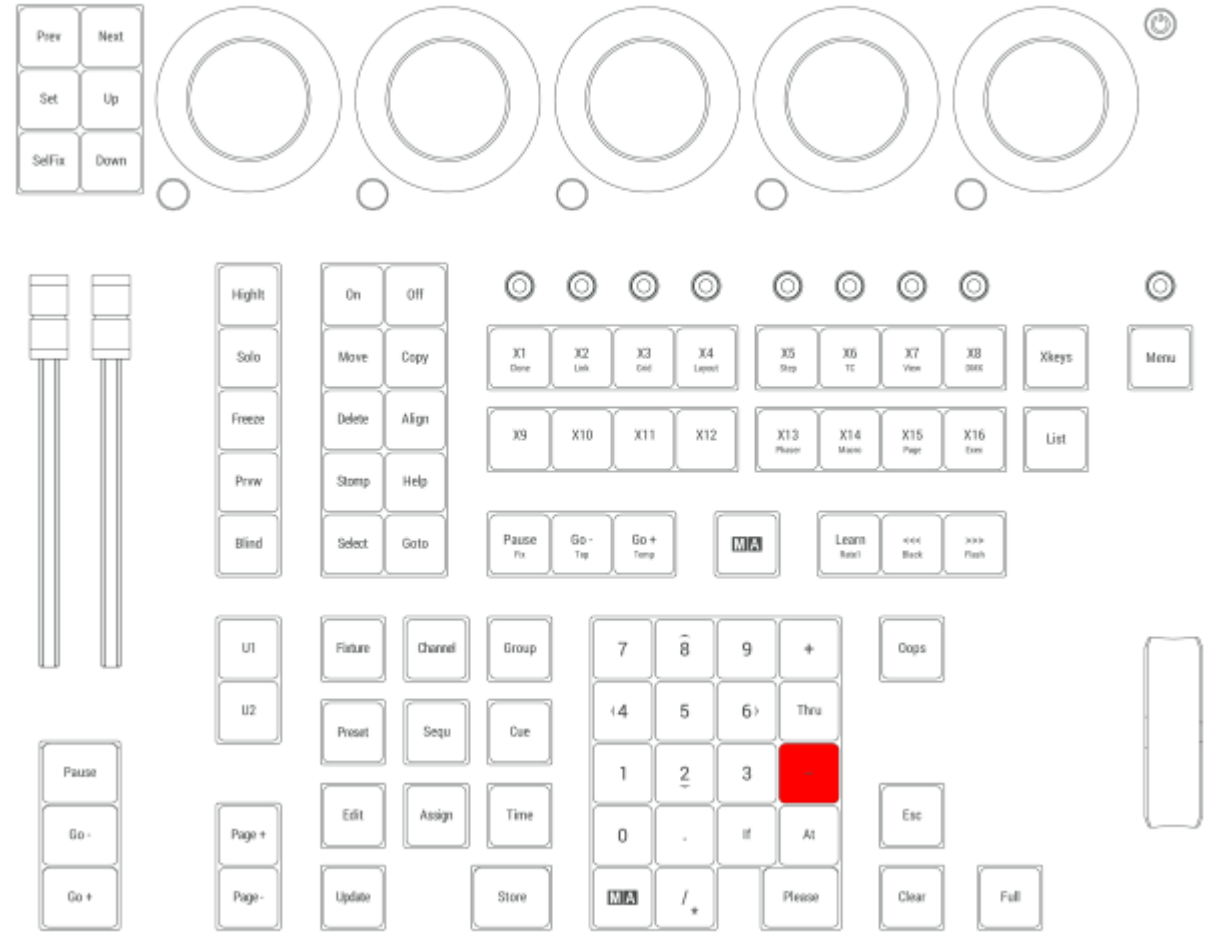
At - 10

Pressing - - decrease the dimmer level of the selected fixtures with 10%.

- is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.5. + [Plus] key

Pressing + enters the + into the command line.

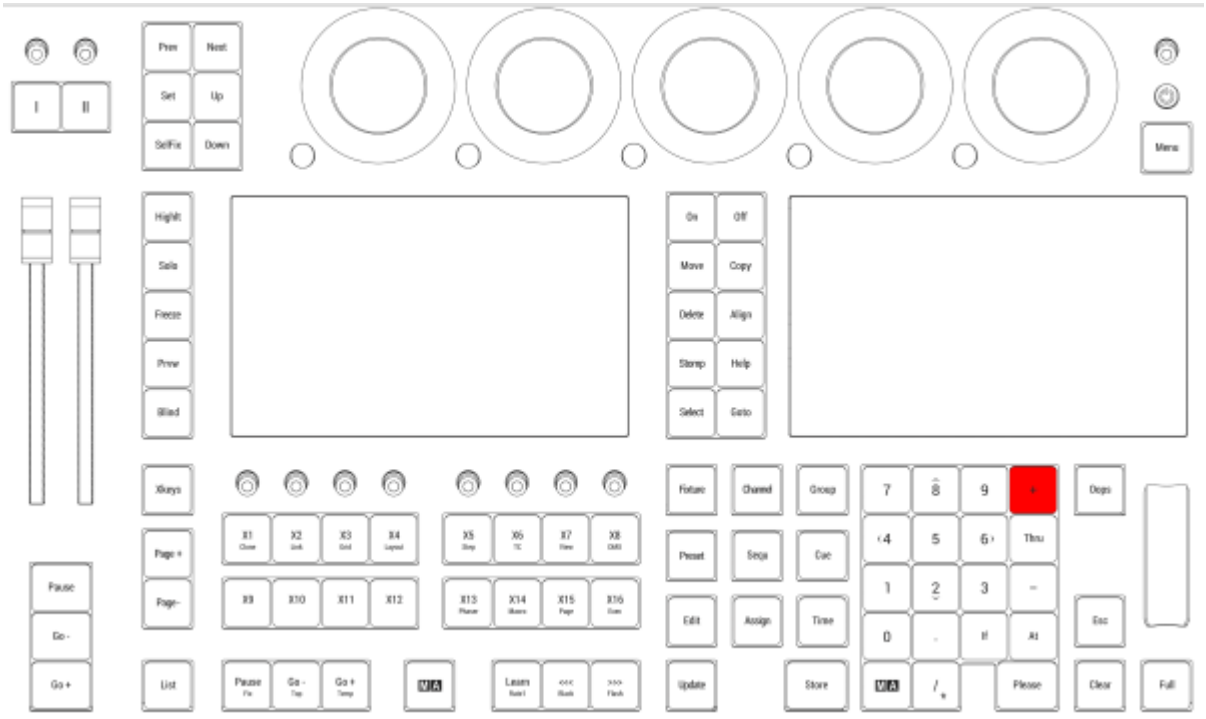
```
User name[Fixture]> Group 8 + Group 11
```

At + 10

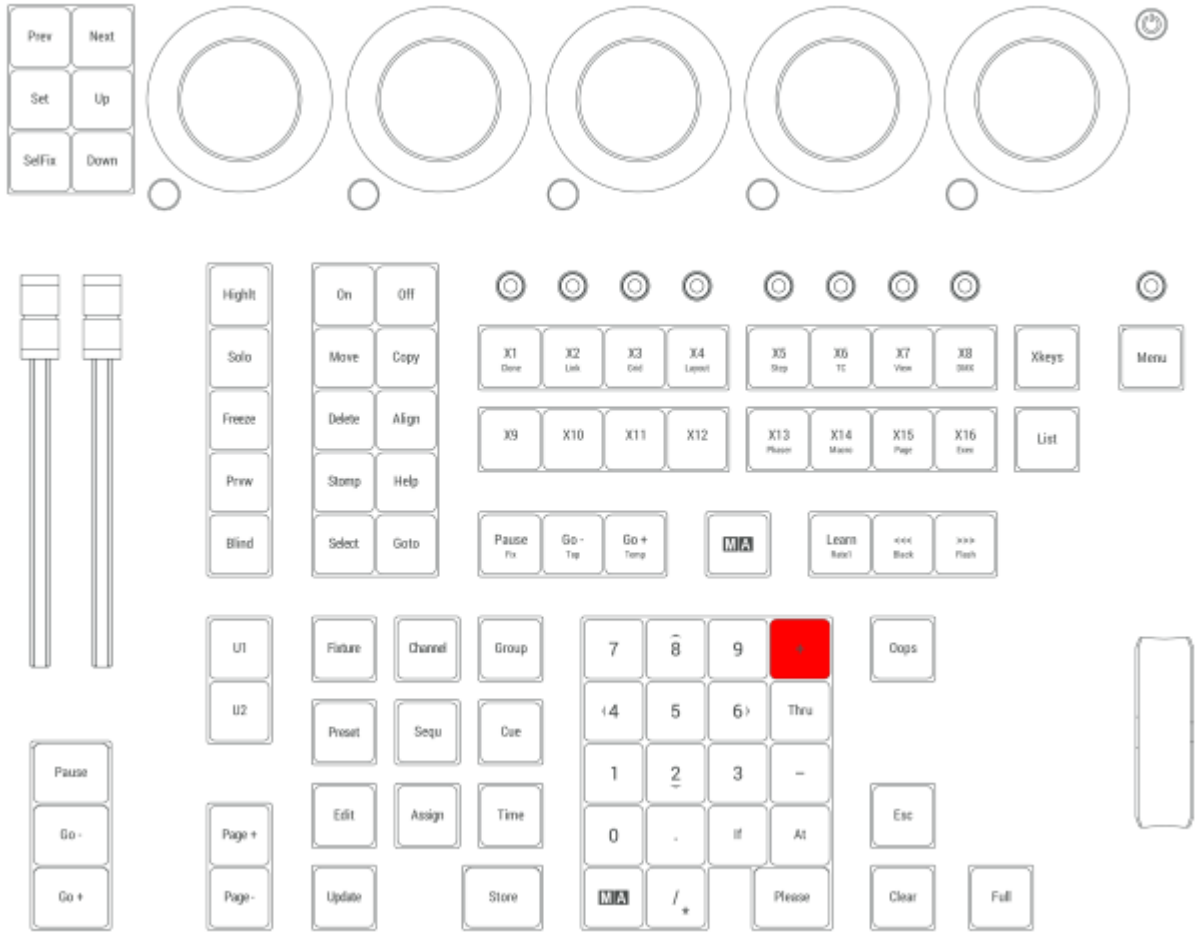
Pressing + + increase the dimmer level of the selected fixtures with 10%.

```
User name[Fixture]> At + 10
```

+ is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.6. / [Slash] | * [Asterisk/Multiply] key

Pressing / enters the / into the command line.

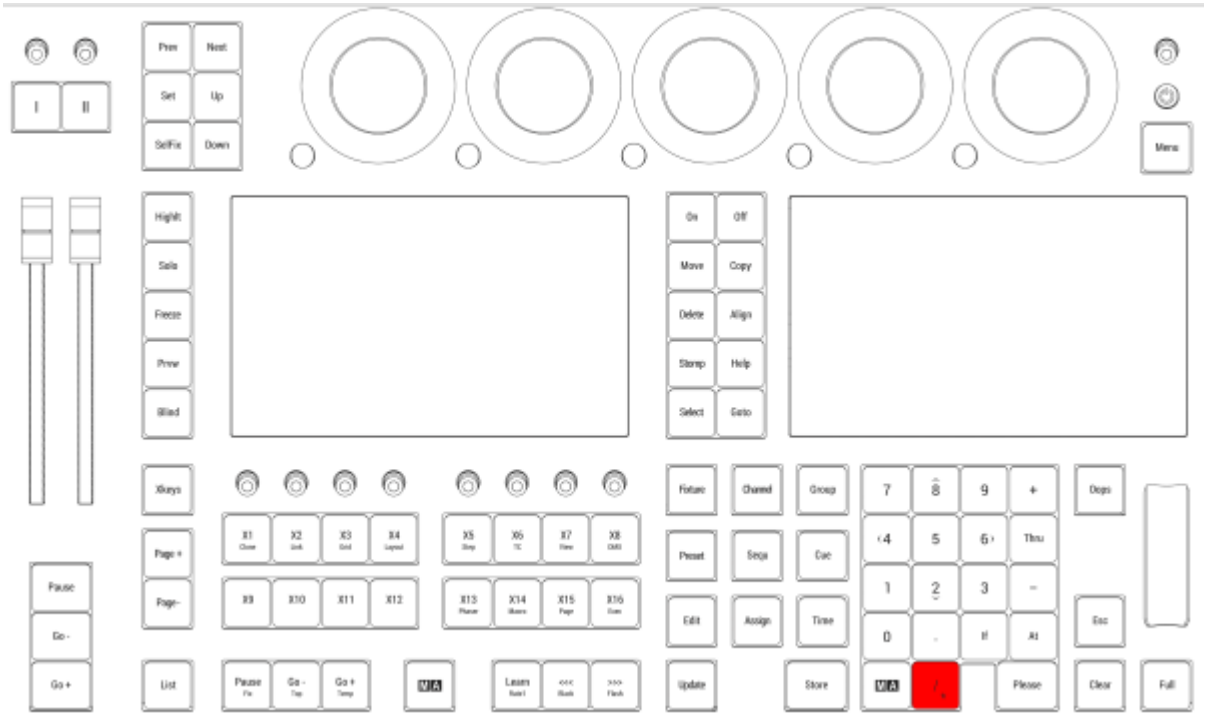
For more information about /, see the **/[Slash] keyword** topic.

Asterisk

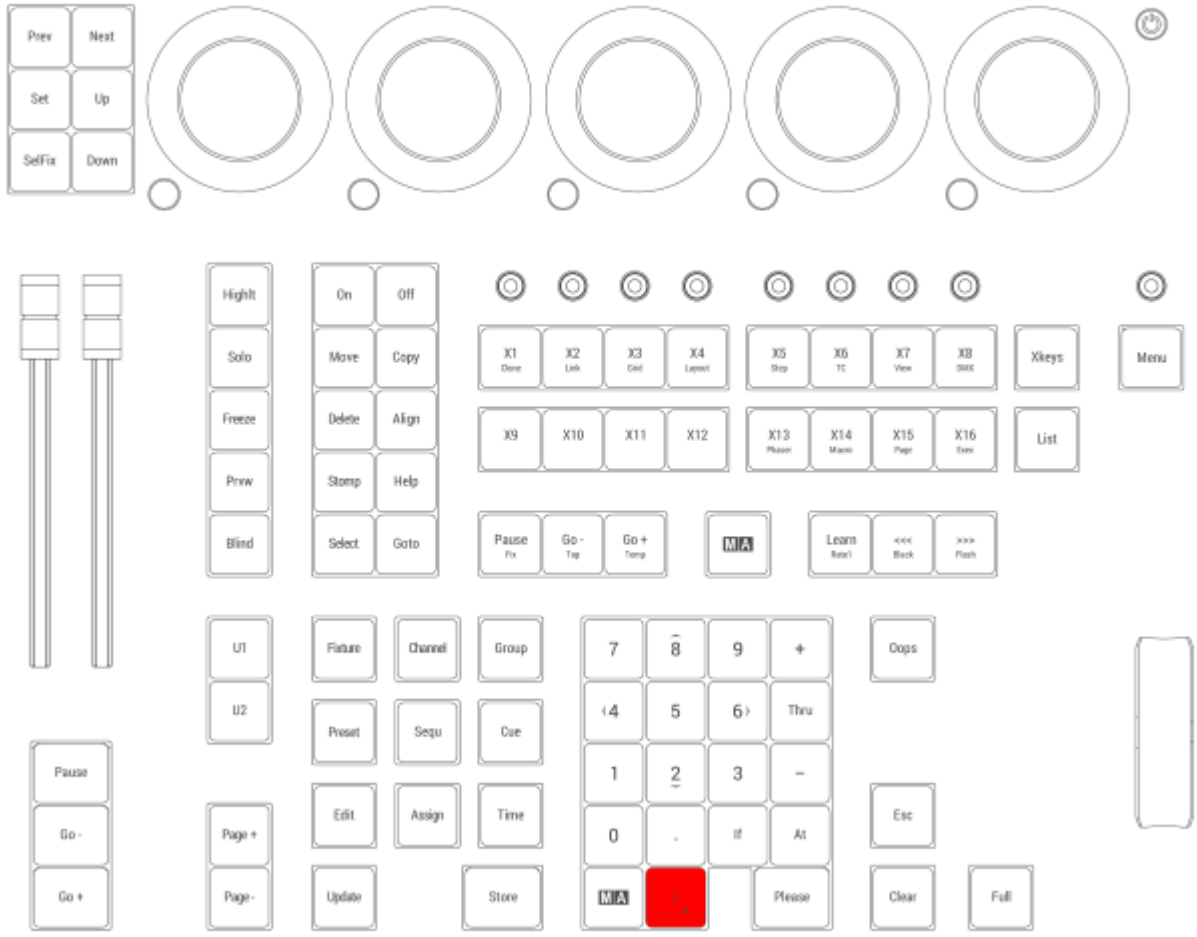
Pressing MA + / enters the * keyword into the command line.

For more information about *, see the *** [Asterisk] keyword** topic.

/ is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.7. At key

Pressing At enters the At keyword into the command line.

For more information about At, see the **At keyword** topic.

Normal

Pressing At At executes the Normal keyword in the command line.

For more information about Normal, see the **Normal keyword** topic.

Integrate

Pressing and holding MA + At enters the Integrate keyword into the command line.

For more information about Integrate, see the **Integrate keyword** topic.

Extract

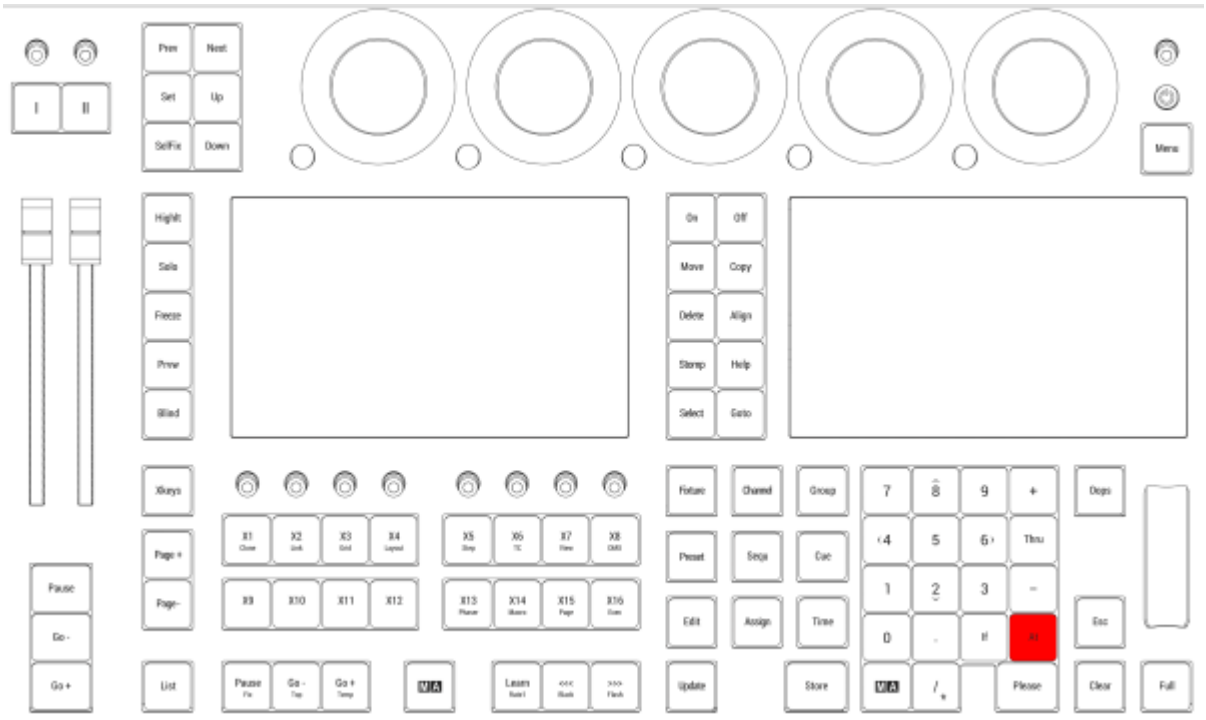
Pressing and holding MA + At + At enters the Extract keyword into the command line.

For more information about Extract, see the **Extract keyword** topic.

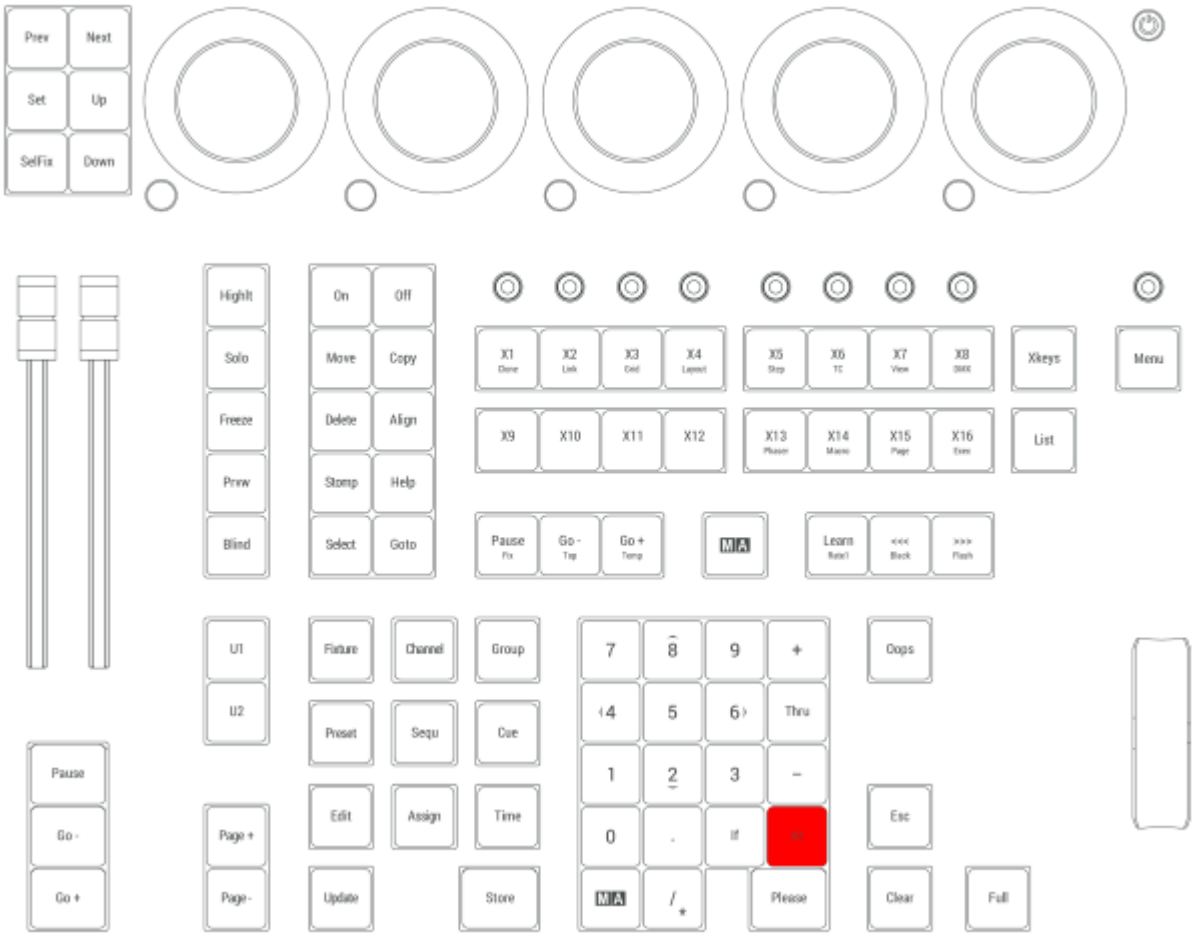
At Filter menu

Pressing and holding At opens the At Filter menu.

At is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.8. Assign key

Pressing Assign enters the Assign keyword into the command line.

For more information about Assign, see the **Assign keyword** topic.

Label

Pressing Assign Assign enters the Label keyword into the command line.

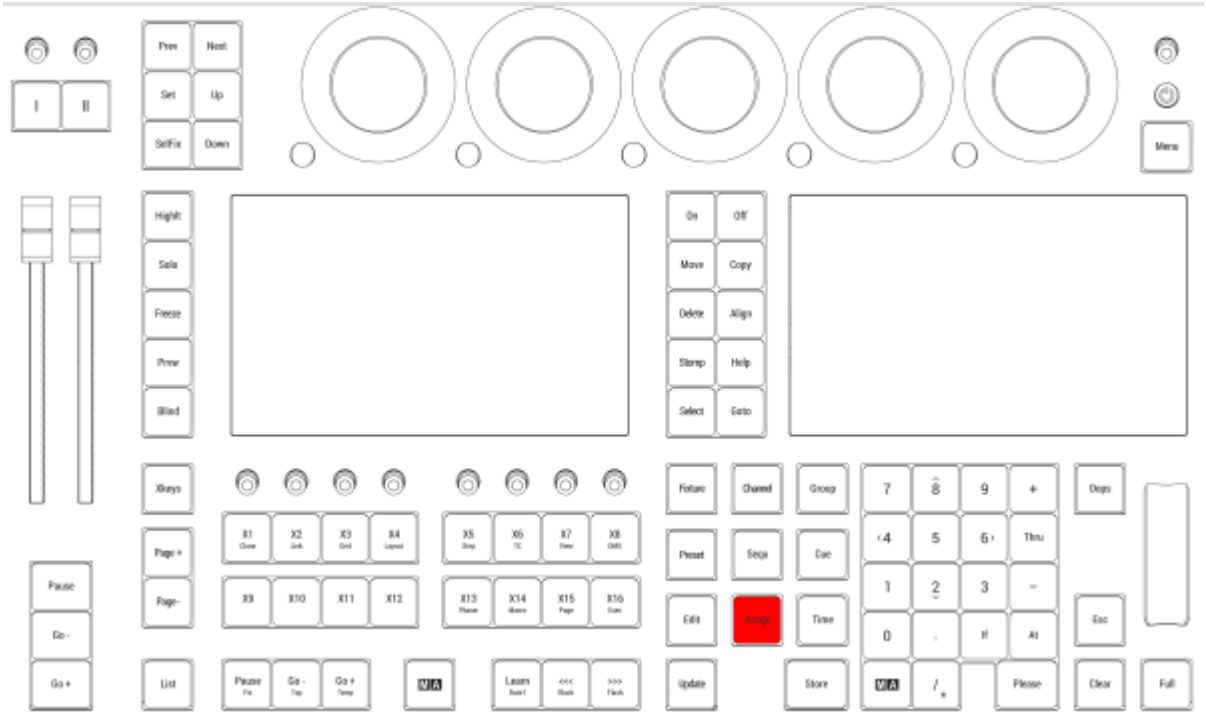
For more information about Label, see the **Label keyword** topic.

Set

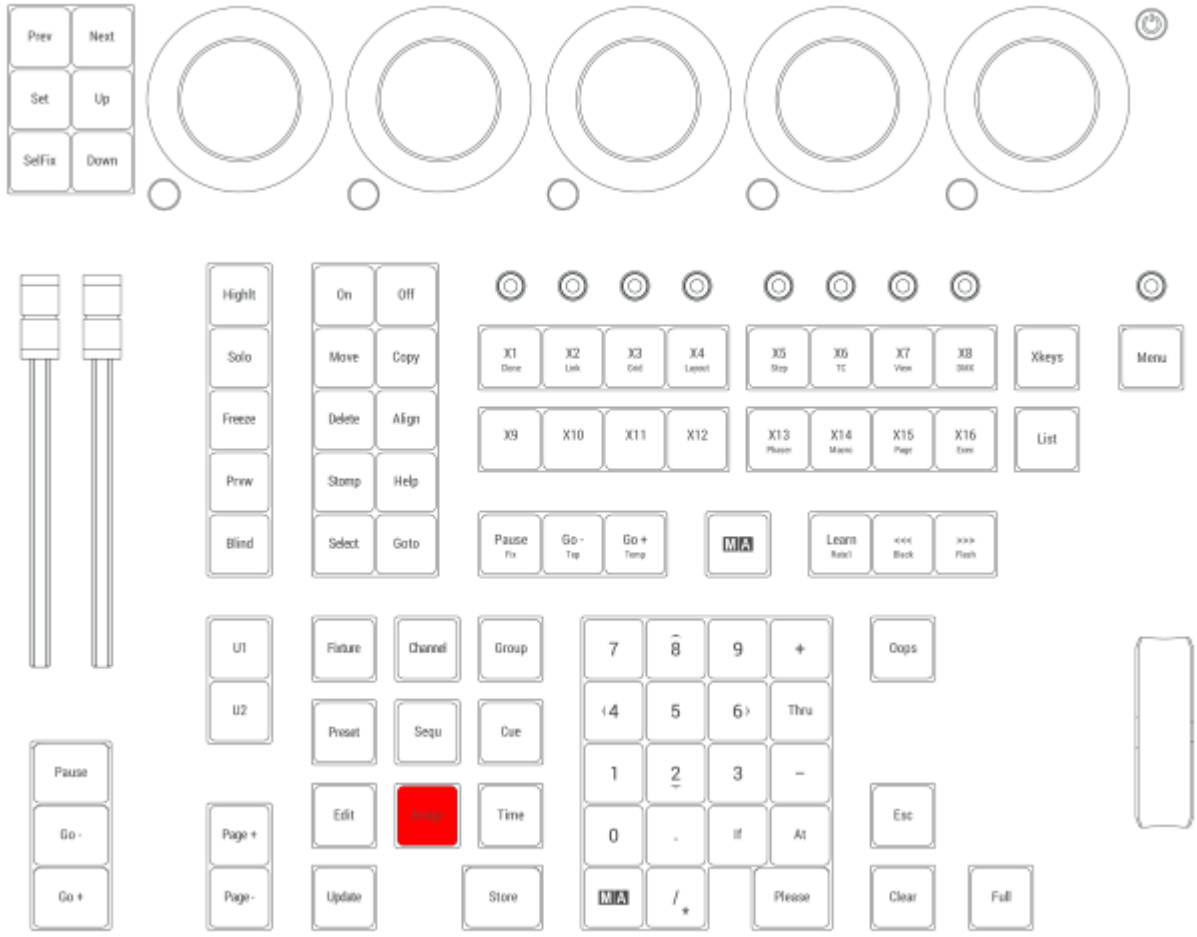
Pressing Assign Assign Assign enters the Set keyword into the command line.

For more information about Set, see the **Set keyword** topic.

Assign is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



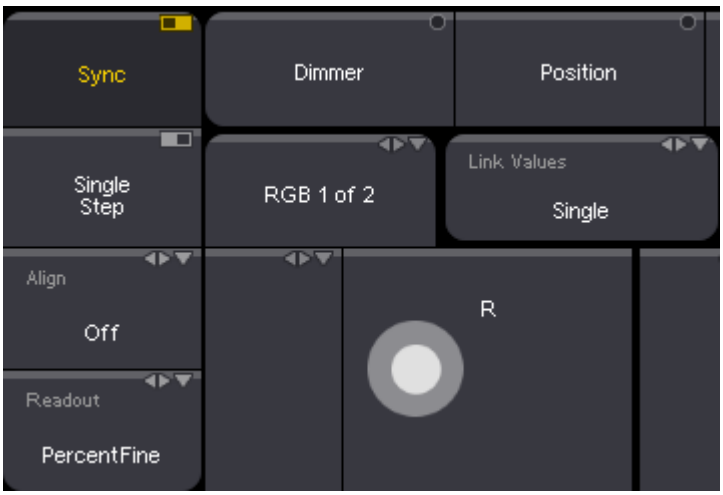
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.9. Align key

Pressing Align toggles between the 6 different Align modes. By default, the Align mode is Off.

Pressing MA + Align toggles between the four different Transition modes. By default, the Transition mode is Linear.

The Align mode is displayed in the default encoder bar.



Default encoder bar

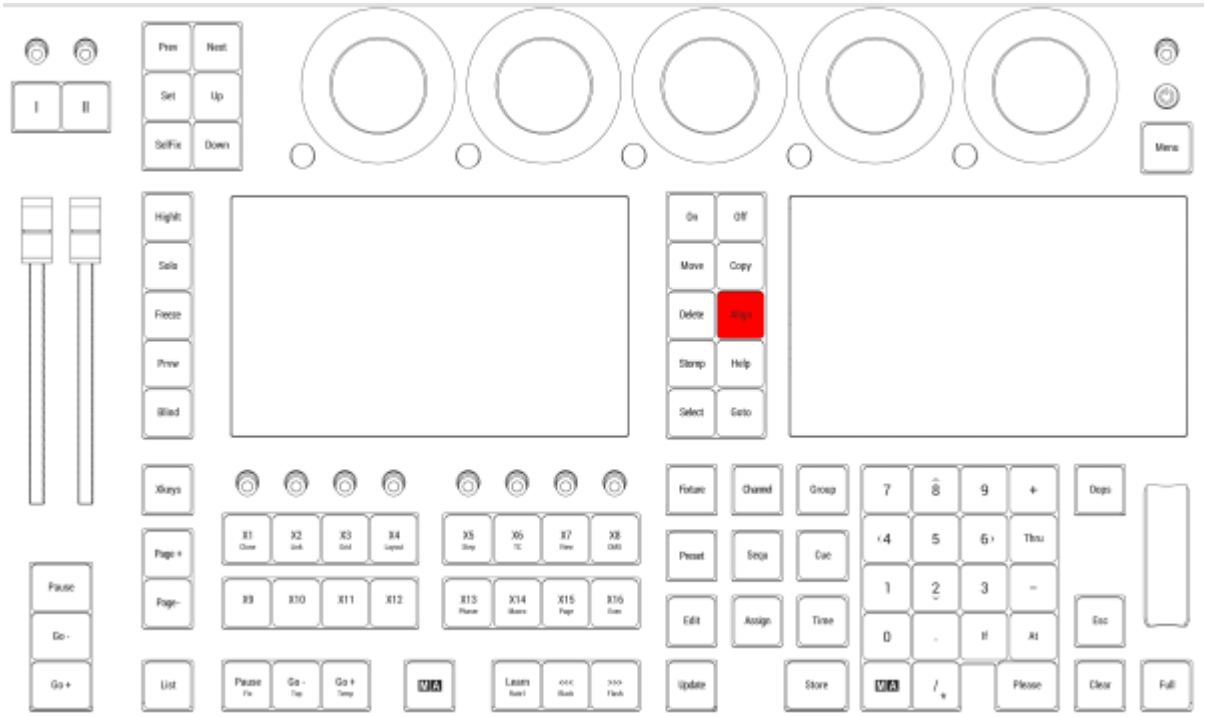
The align mode will be switched off when changing the value of a different attribute and the Transition mode becomes linear.

Pressing and holding Align resets the Align mode.

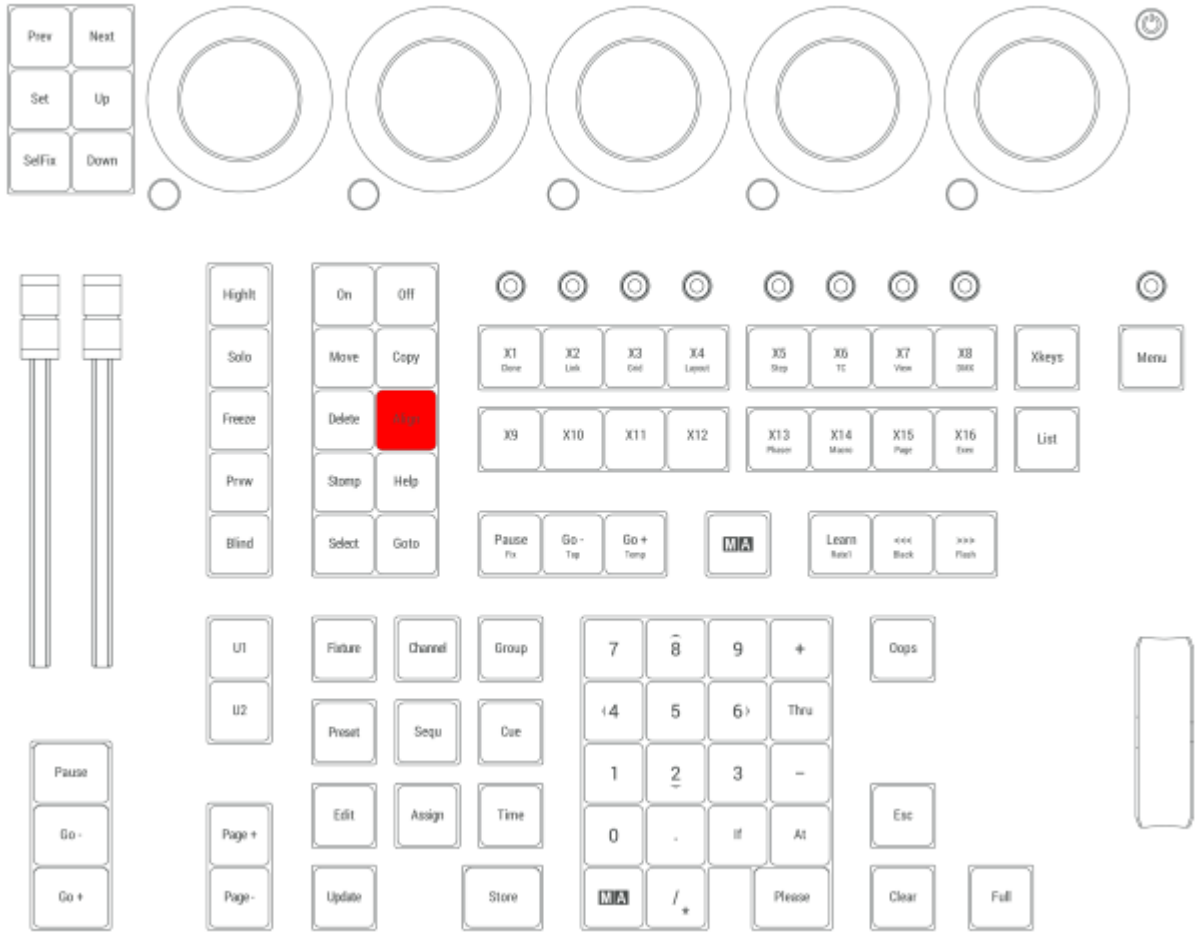
Pressing and holding MA + Align resets the Transition mode.

To read more about Align, please see the **Align topic**.

Align is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

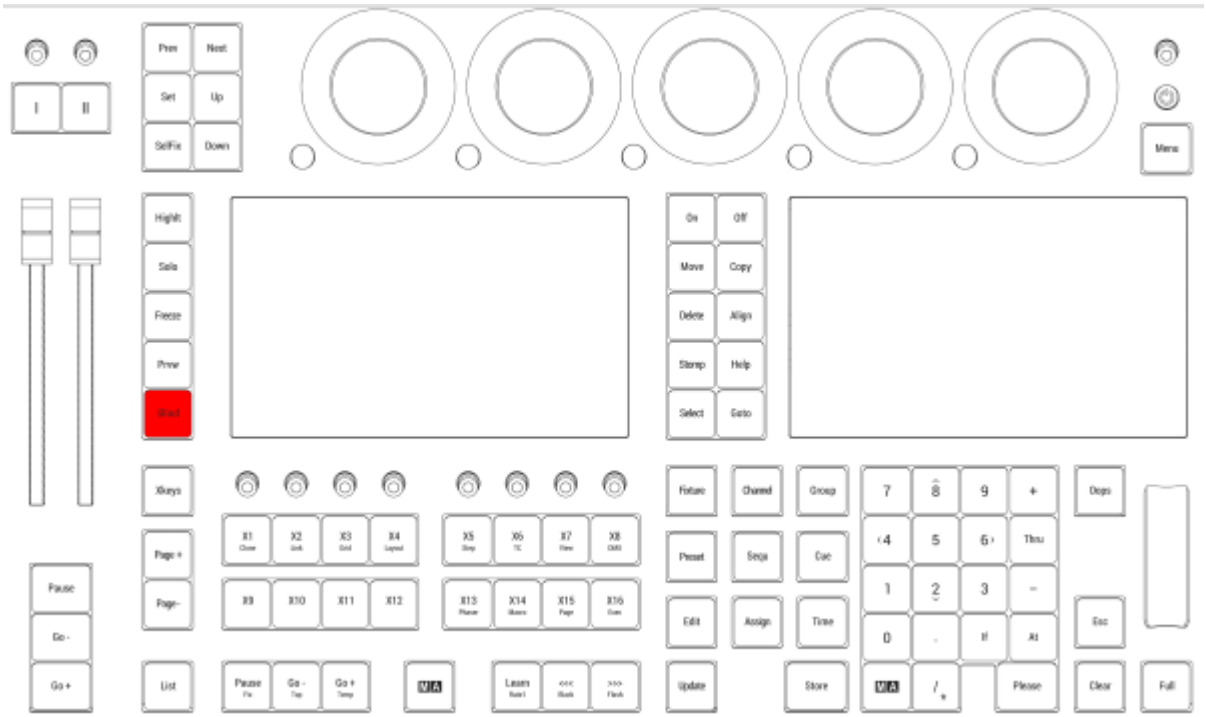
2.15.10. Blind key

Pressing Blind toggles the Blind function.

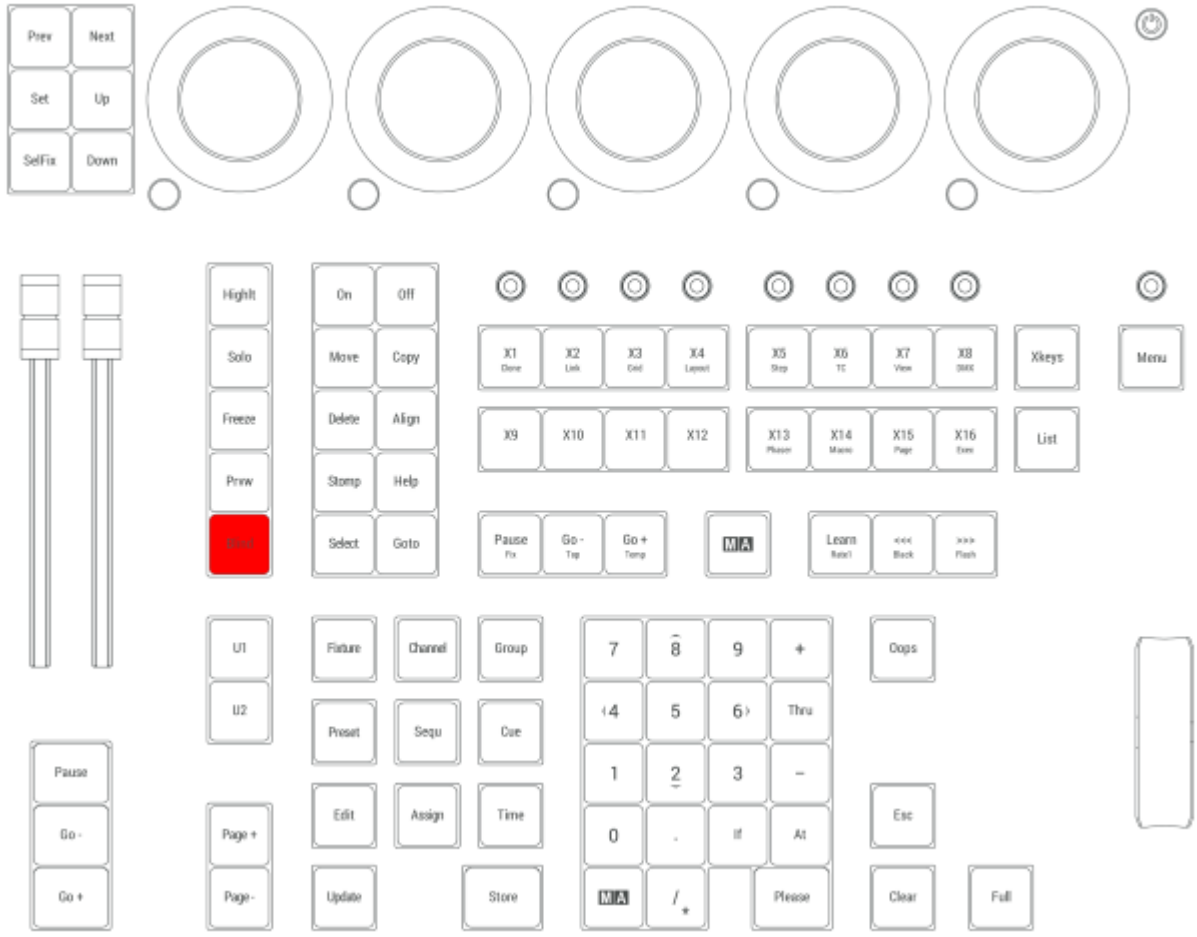
```
OK:Blind
```

For more information about Blind, see the **Blind keyword** topic.

Blind is located in the Command Section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.11. Clear key

If there is something in the programmer, Clear behaves like this:

- Pressing Clear executes the ClearSelection keyword.
- Pressing Clear Clear executes the ClearActive keyword.
- Pressing Clear Clear Clear executes the ClearAll keyword.

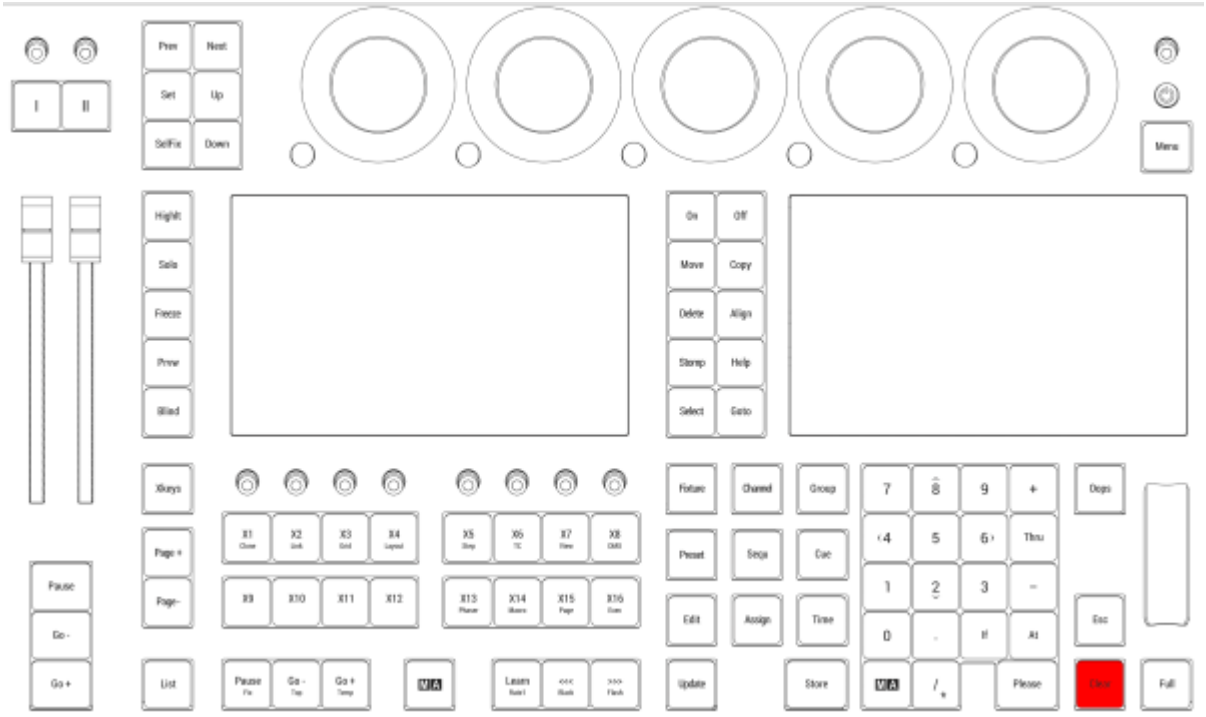
If there is nothing in the programmer, pressing Clear deletes unexecuted commands in the command line and executes the ClearActive keyword

Pressing and holding Clear executes the ClearAll keyword.

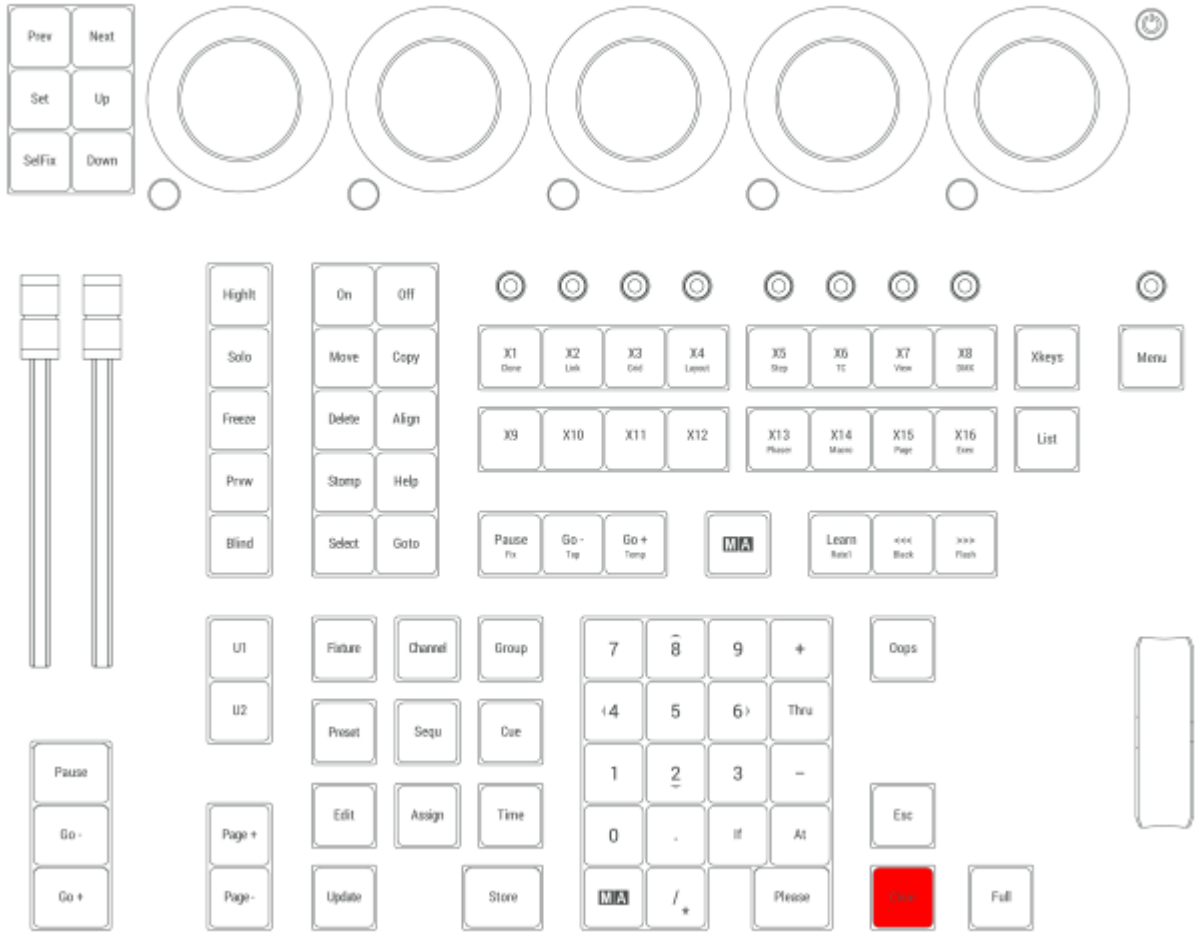
Pressing Clear, no matter how many times always gives a Clear in the Command Line Feedback

For more information about Clear, see the **Clear keyword** topic, the **ClearSelection keyword**, the **ClearActive keyword** or the **ClearAll keyword** topic.

Clear is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.12. Channel key

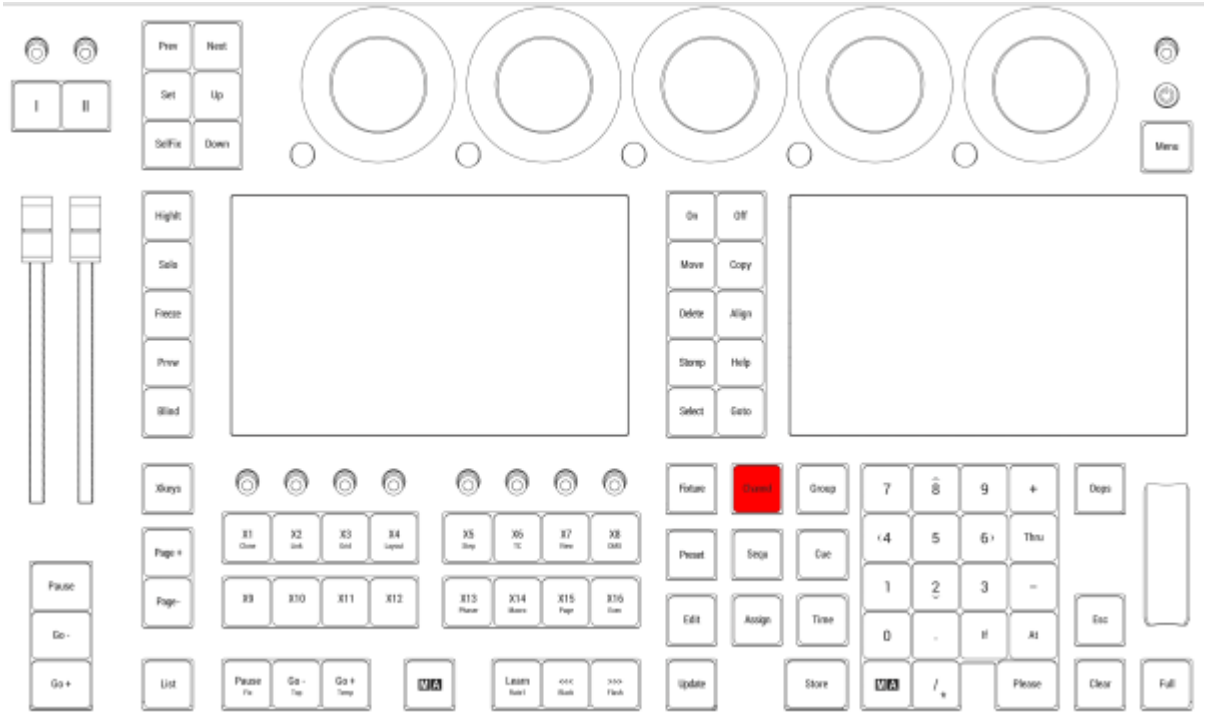
Pressing Channel enters the Channel keyword into the command line.

For more information about Channel, see the **Channel keyword** topic.

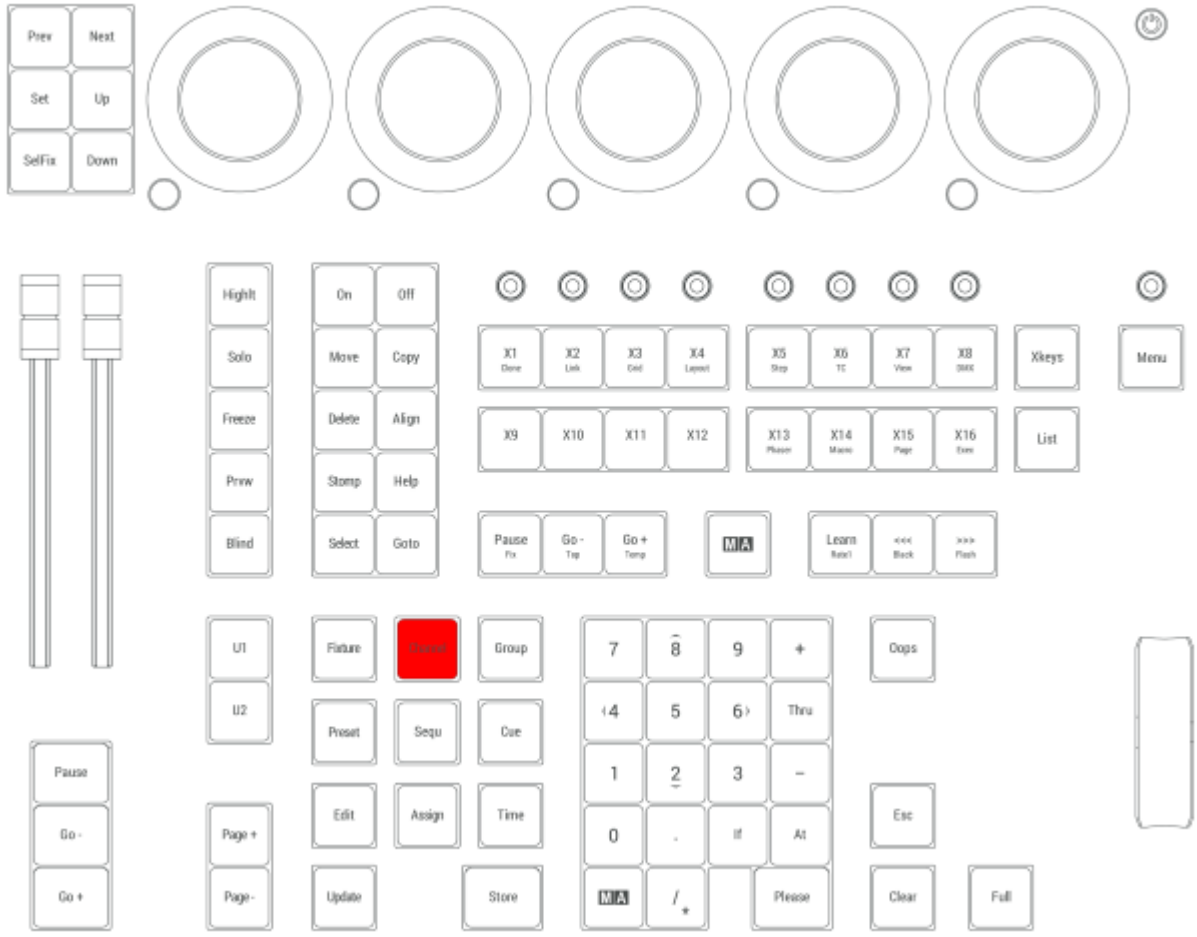
Custom ID types

Pressing Channel again, enters the next ID type. These are custom made and can be changed in the patch. For more information about ID types, see the **Add fixtures to the show** topic.

Channel is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.13. Copy key

Pressing Copy enters the copy keyword into the command line.

For more information about Copy, see the **Copy keyword** topic.

Paste

Pressing Copy Copy enters the Paste keyword into the command line.

For more information about Paste, see the **Paste keyword** topic.

Insert

Pressing Copy Copy Copy enters the Insert keyword into the command line.

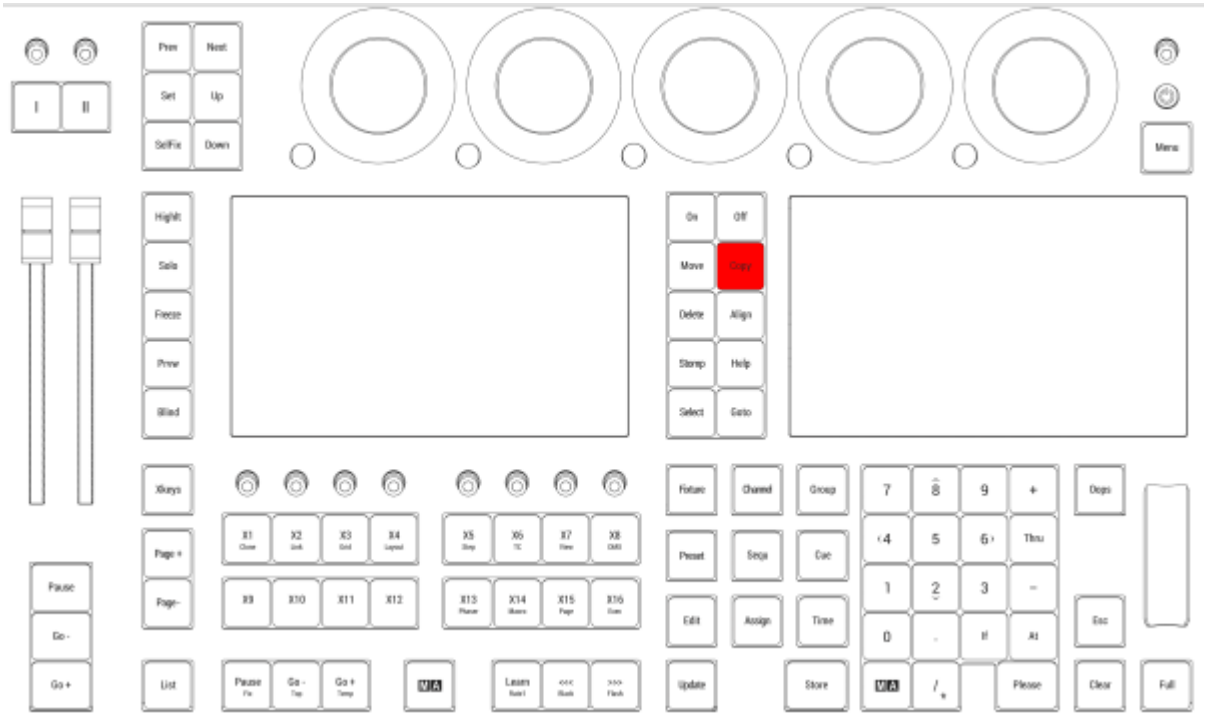
For more information about Insert, see the **Insert keyword** topic.

Cut

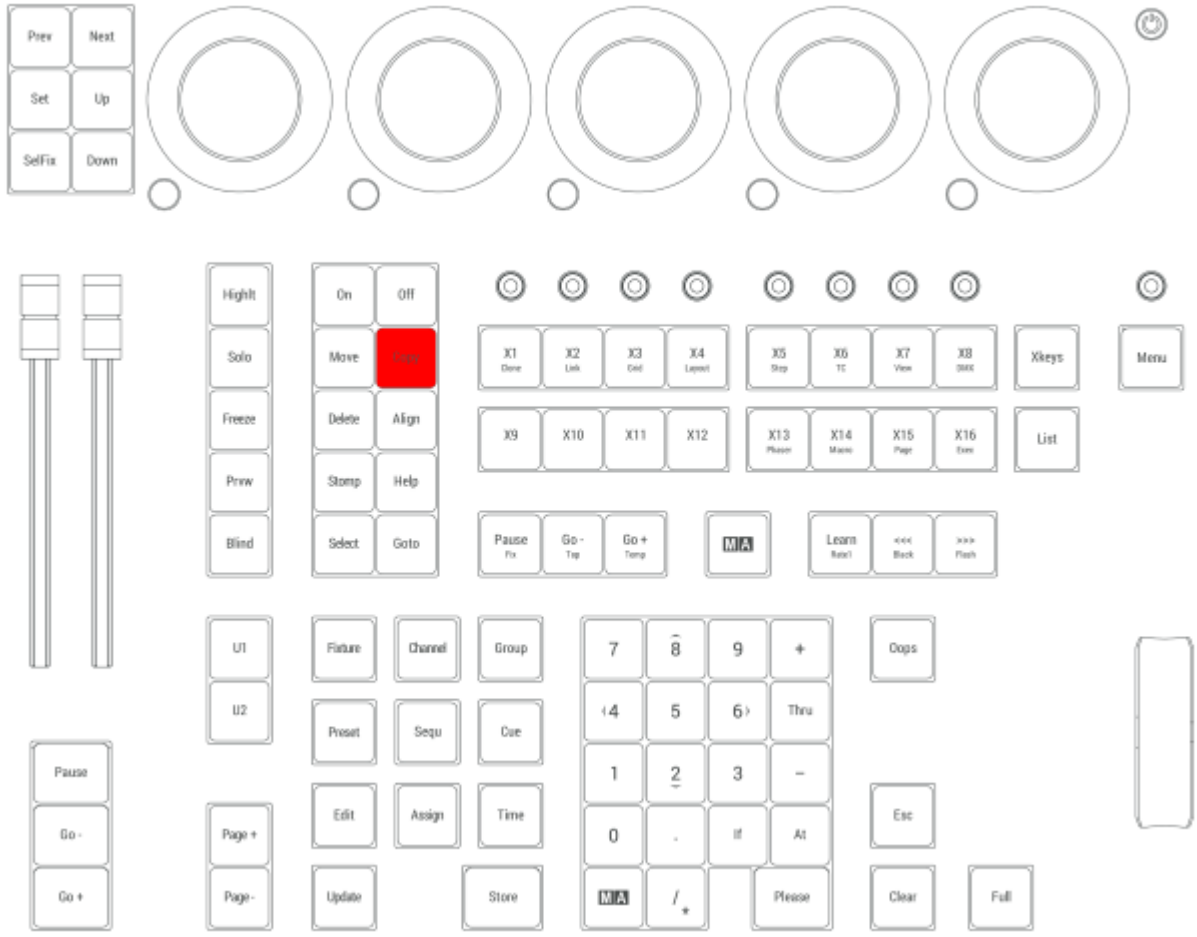
Pressing MA Copy enters the Cut keyword into the command line.

For more information about Cut, see the **Cut keyword** topic.

Copy is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.14. Ctrl

Multi-Select

To select multiple cells in a sheet, hold Ctrl and tap or click each desired cell.

Clean Start

To perform a clean start, hold Ctrl when the mode selection pop-up appears during the boot or reboot process and tap the desired mode. For more information about clean start, see the **Clean Start** topic.

Ctrl is located in two places on the bottom row of the alpha-numeric keyboard.

Location on grandMA3 full-size and grandMA3 light console

2.15.15. Cue key

Pressing Cue enters the Cue keyword into the command line.

For more information about Cue, see the **Cue keyword** topic.

Part

Pressing Cue Cue enters the Part keyword into the command line.

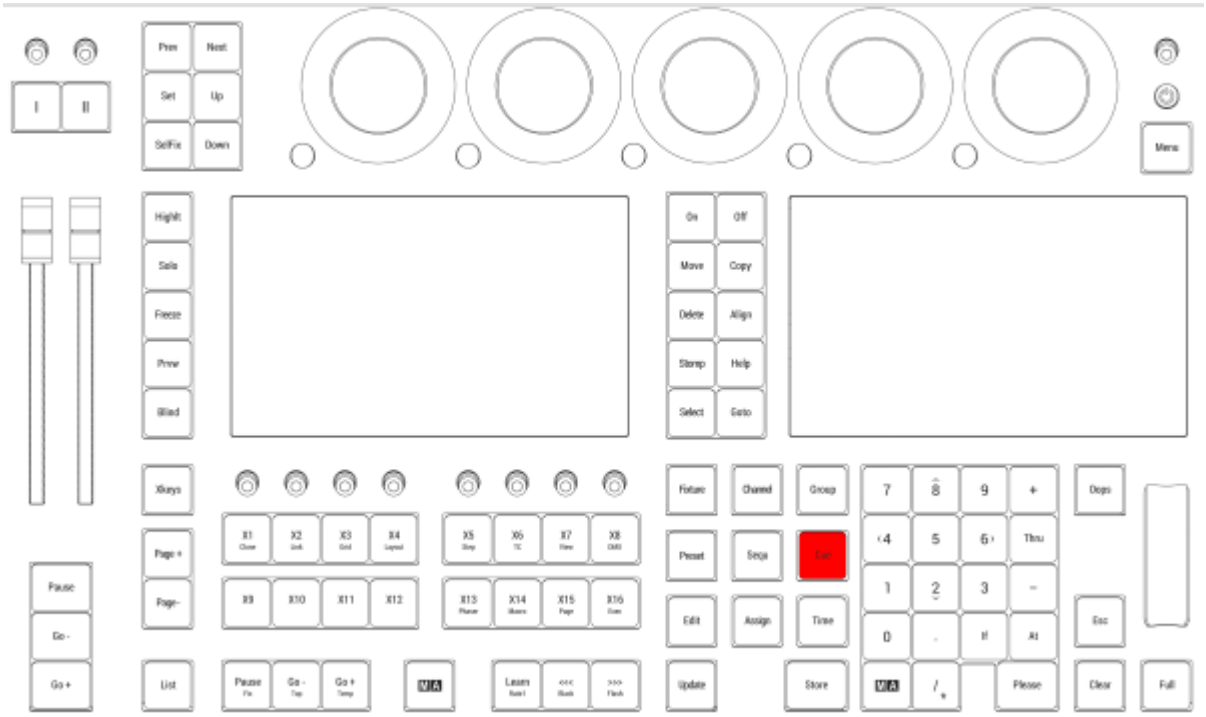
For more information about Part, see the **Part keyword** topic.

Programmer

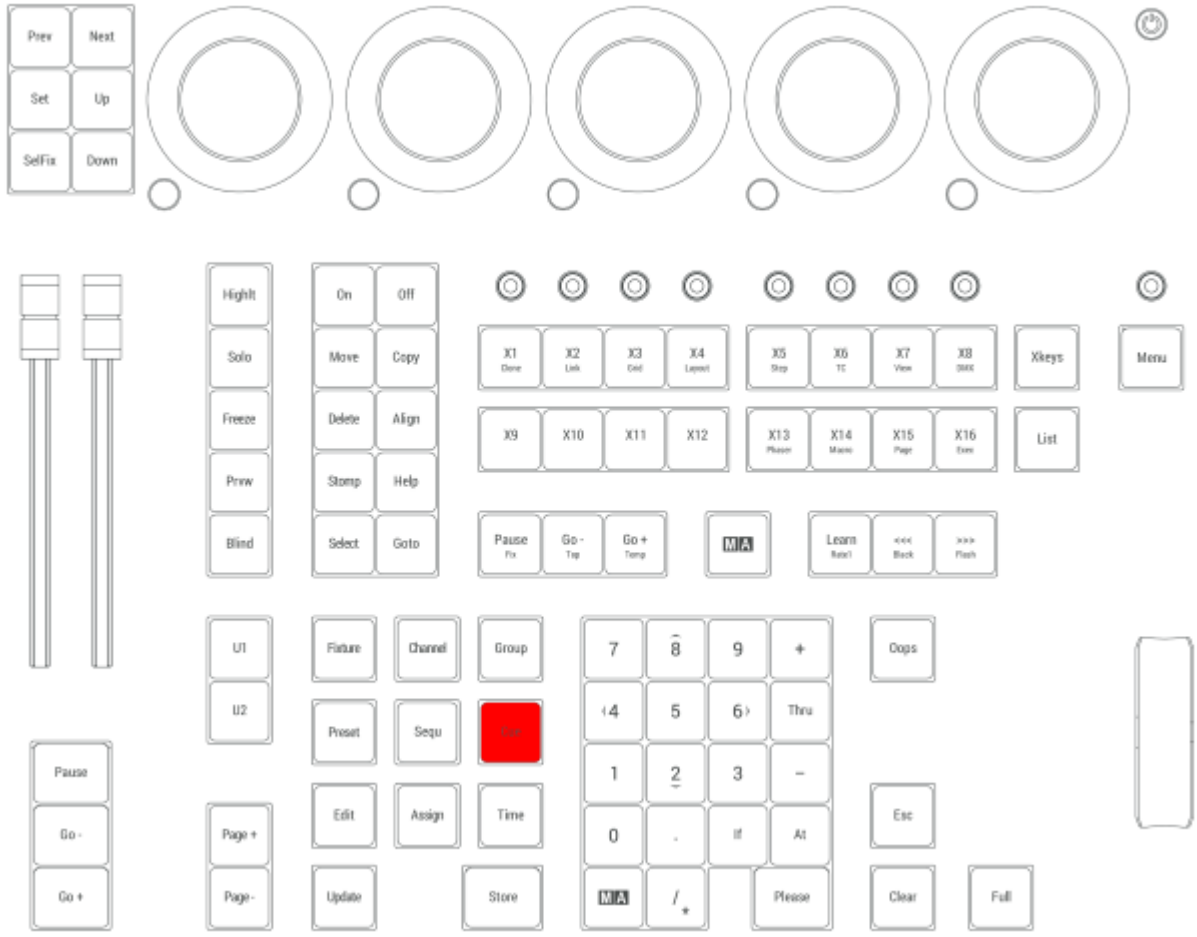
Pressing MA + Cue enters the Programmer keyword into the command line.

For more information about Programmer, see the **Programmer keyword** topic.

Cue is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

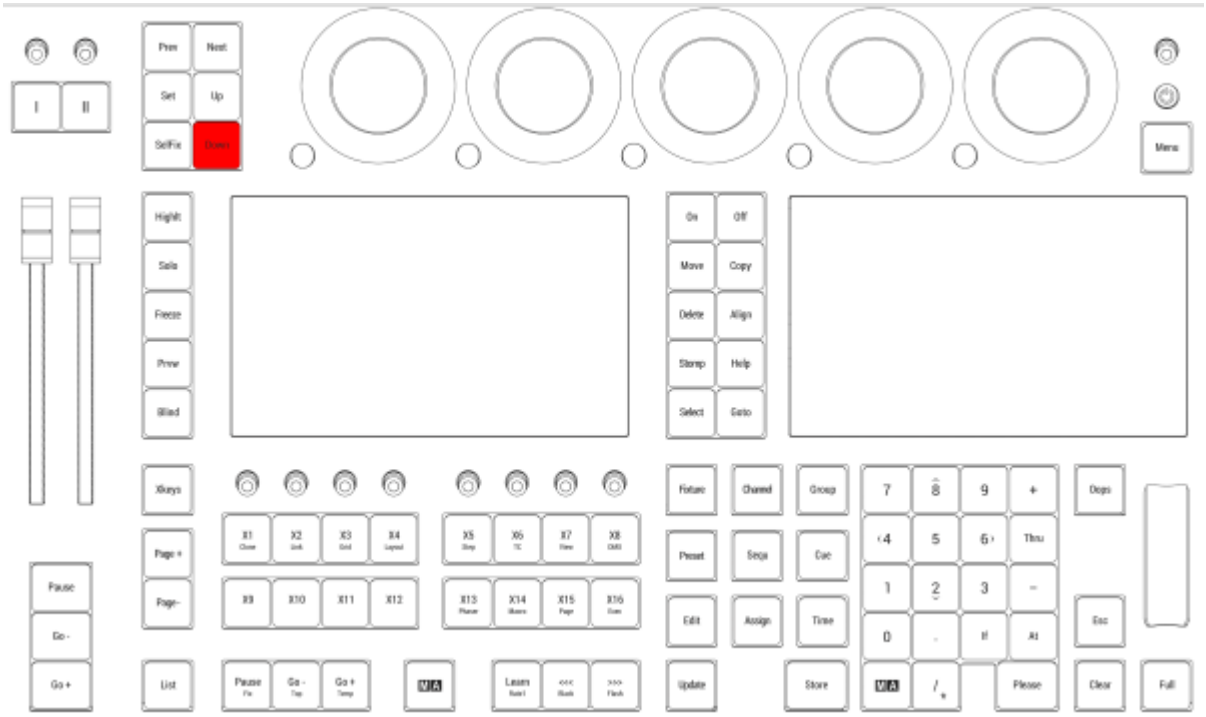


Location on grandMA3 compact consoles and grandMA3 onPC command wing

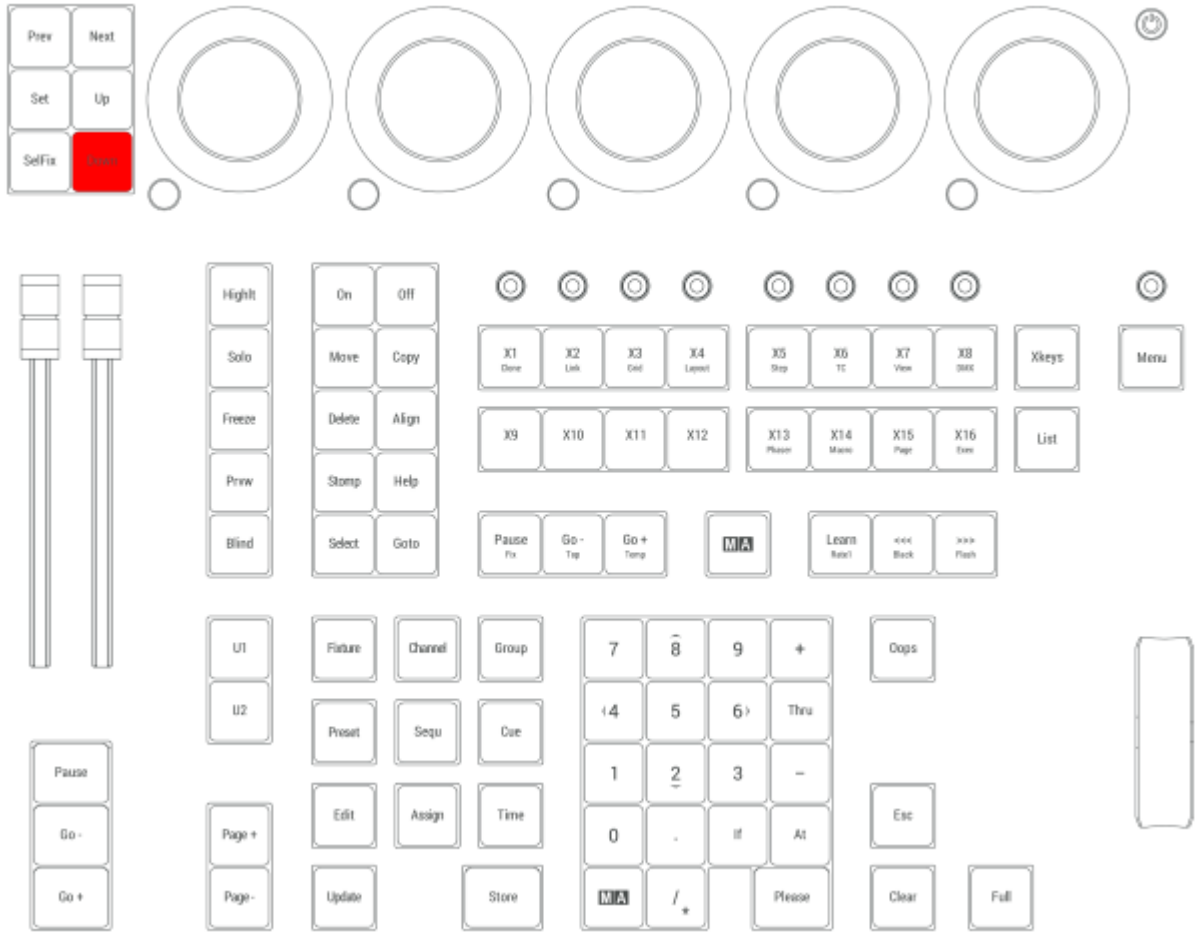
2.15.16. Down key

Pressing Down goes one level down in the fixture structure.

Down is located in the command section on the left side of the five dual encoders.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.17. Delete key

Pressing Delete enters the Delete keyword into the command line.

For more information about Delete, see the **Delete keyword** topic.

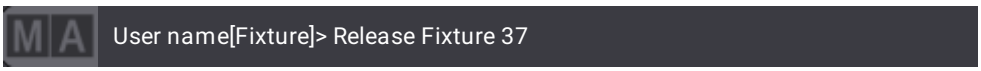
Remove

Pressing Delete Delete enters the Remove keyword into the command line.

For more information about Remove, see the **Remove keyword** topic.

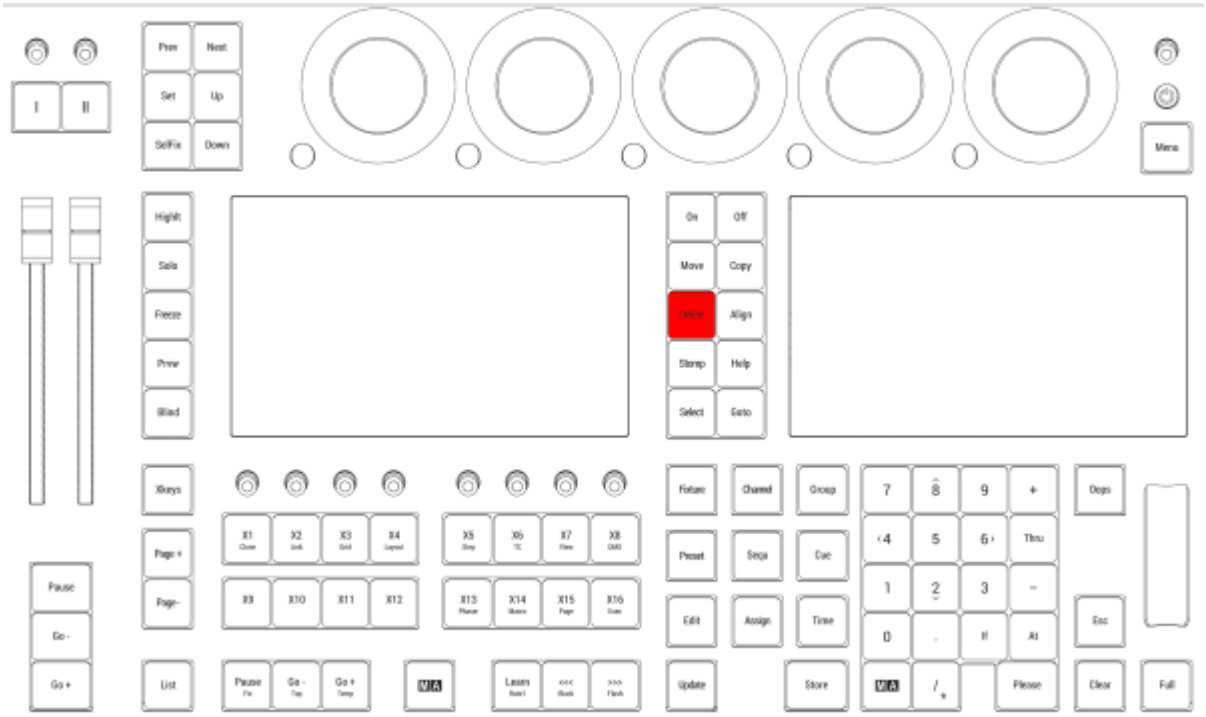
Release

Pressing Delete Delete Delete enters the Release keyword into the command line.

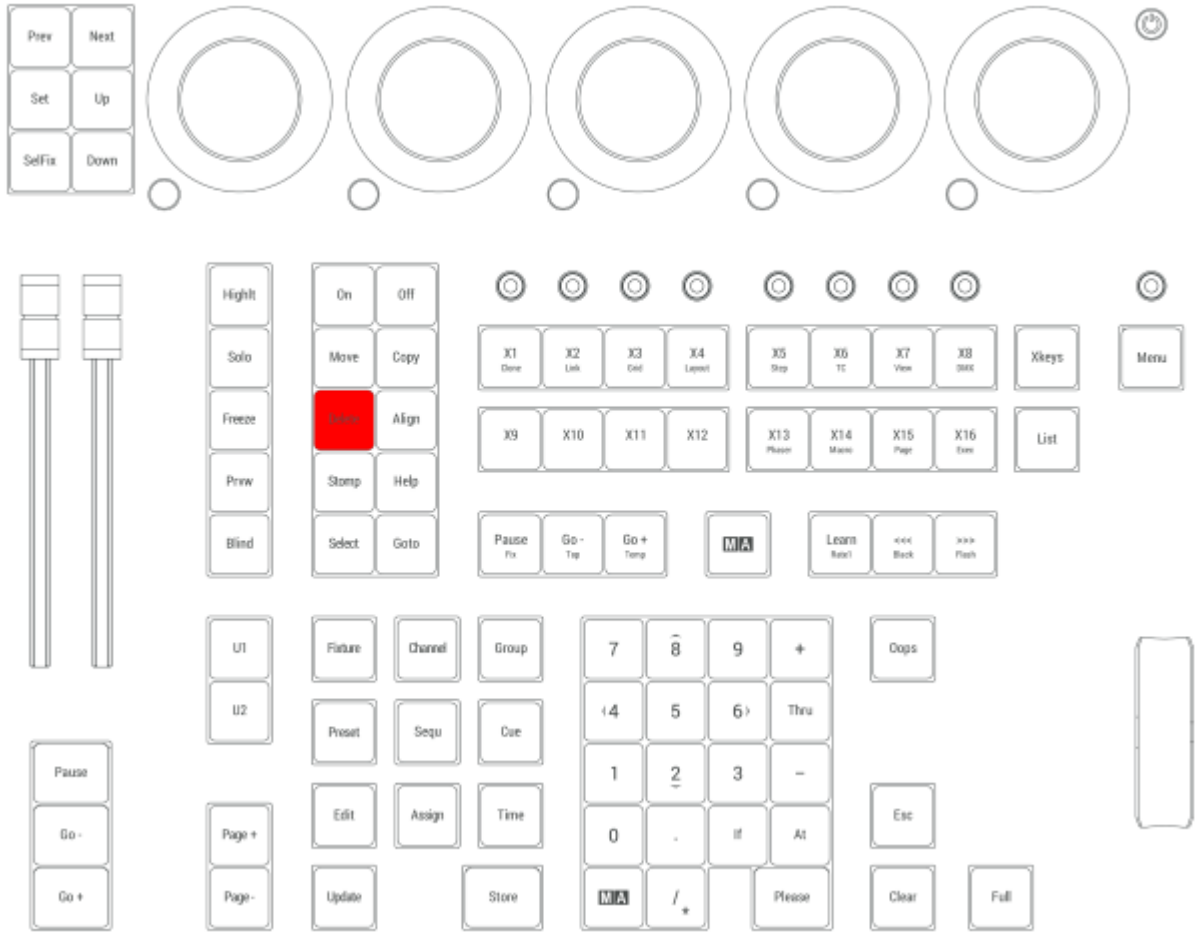


For more information about Release, see the **Release keyword** topic.

Delete is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

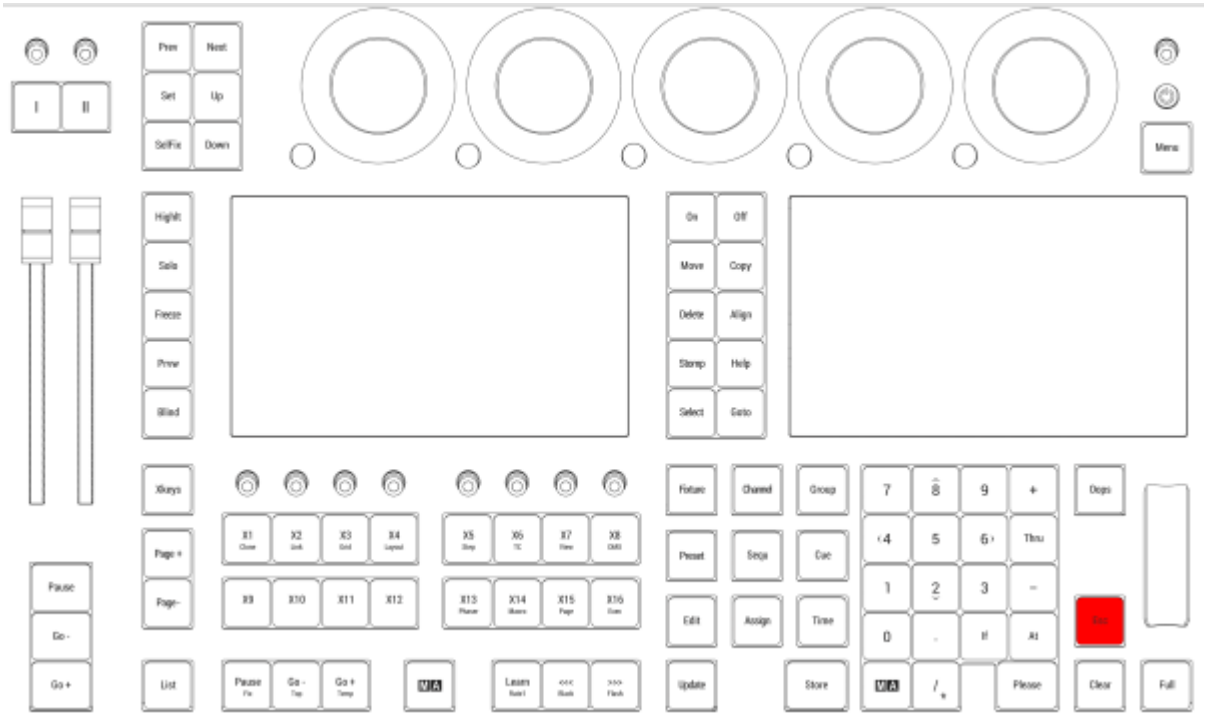


Location on grandMA3 compact consoles and grandMA3 onPC command wing

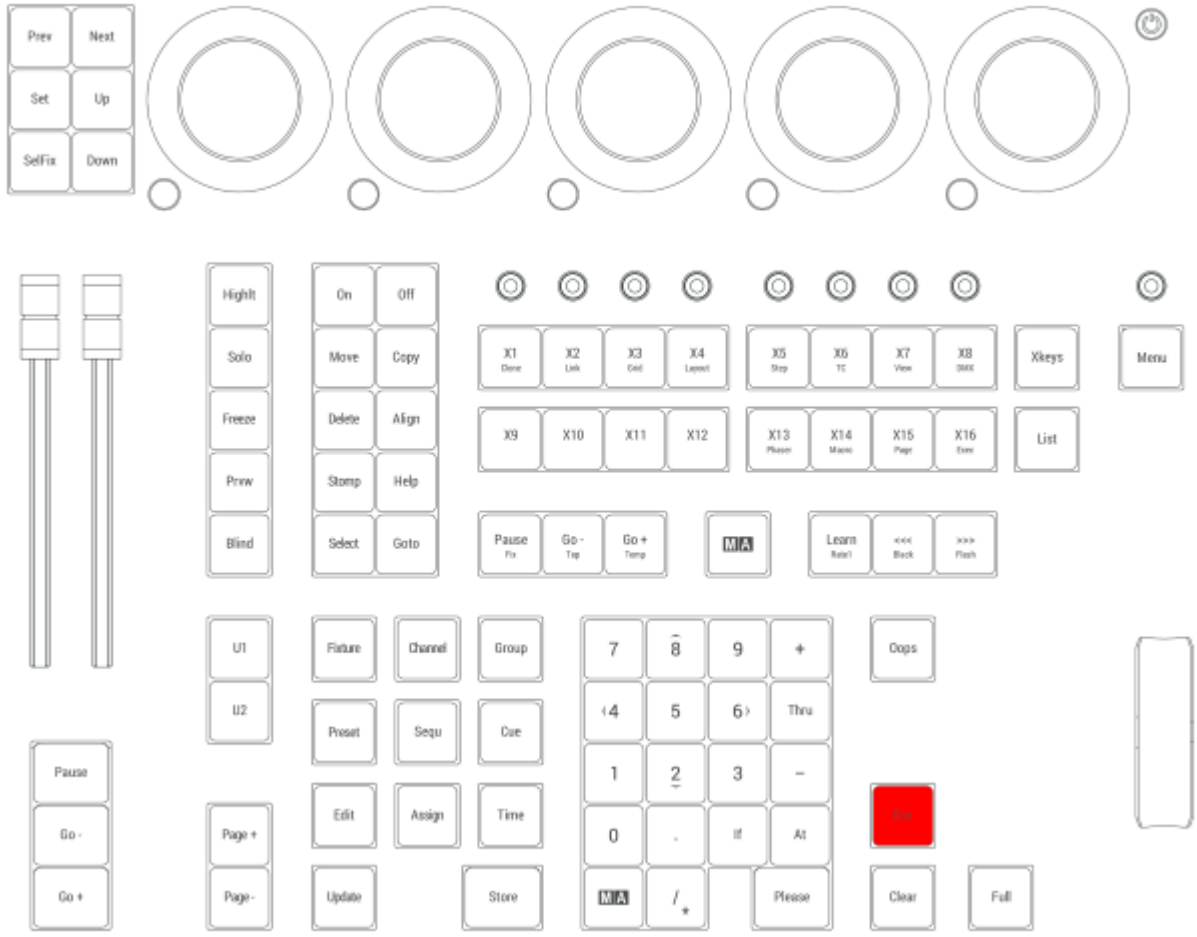
2.15.18. Esc key

Pressing Esc deletes unexecuted commands in the command line and closes pop-ups.

Esc is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.19. Edit key

Pressing Edit enters the Edit keyword into the command line.

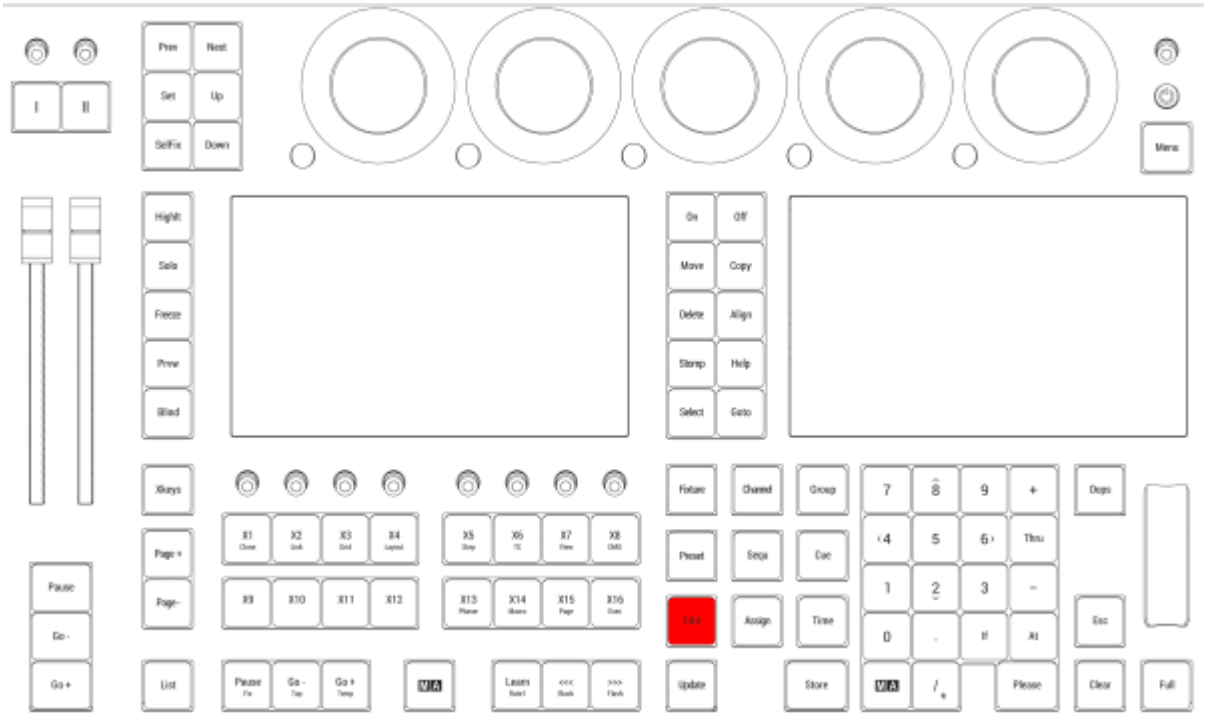
For more information about Edit, see the **Edit keyword** topic.

EditSetting

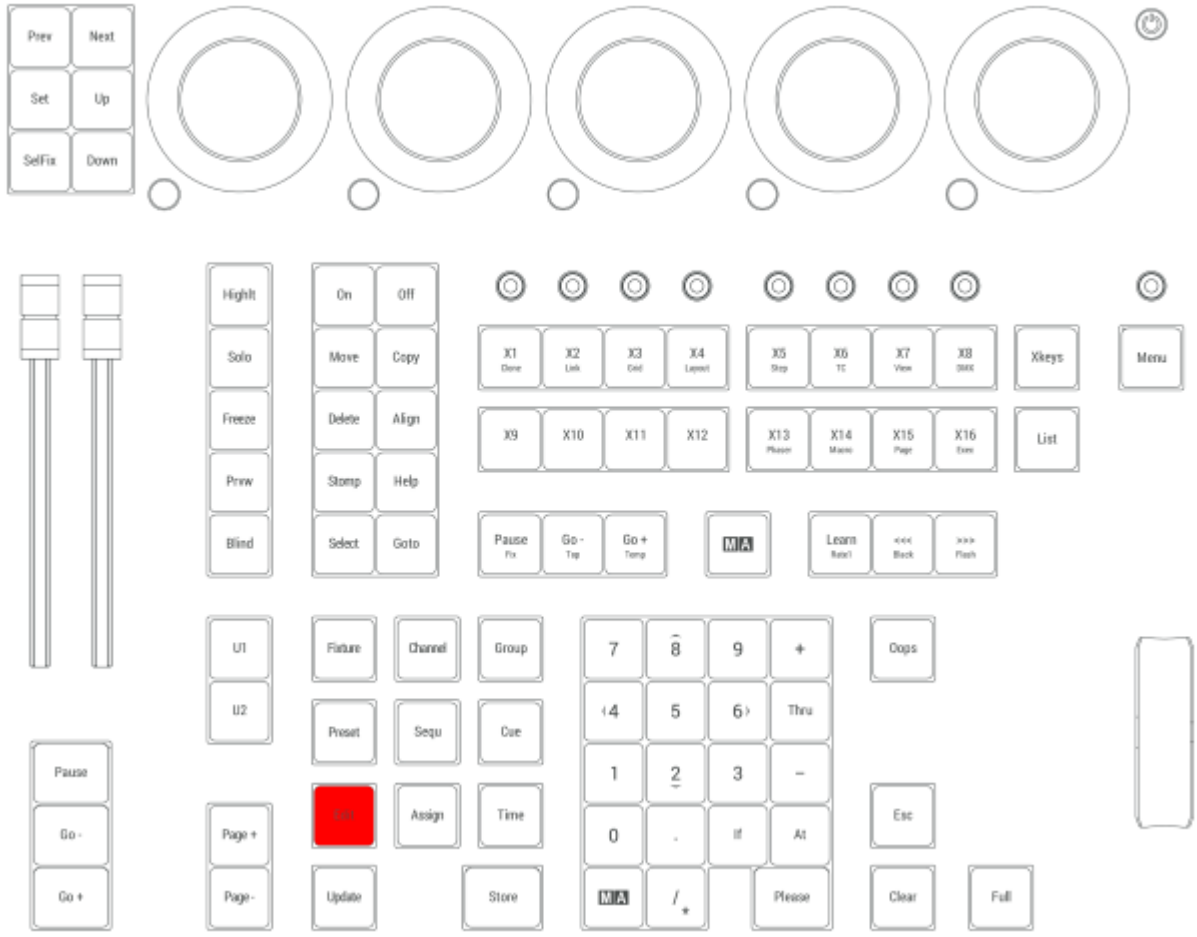
Pressing Edit Edit enters the EditSetting keyword into the command line.

For more information about EditSetting, see the **EditSetting keyword** topic.

Edit is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

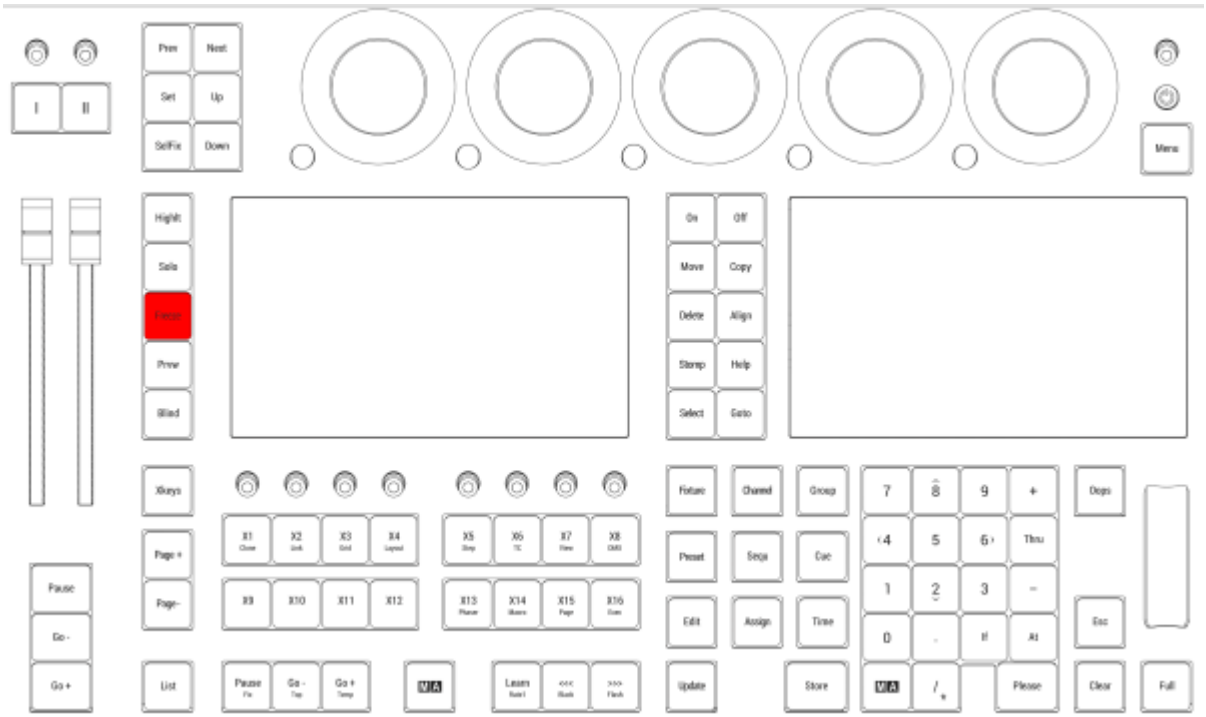
2.15.20. Freeze key

Pressing Freeze toggles the Freeze function.

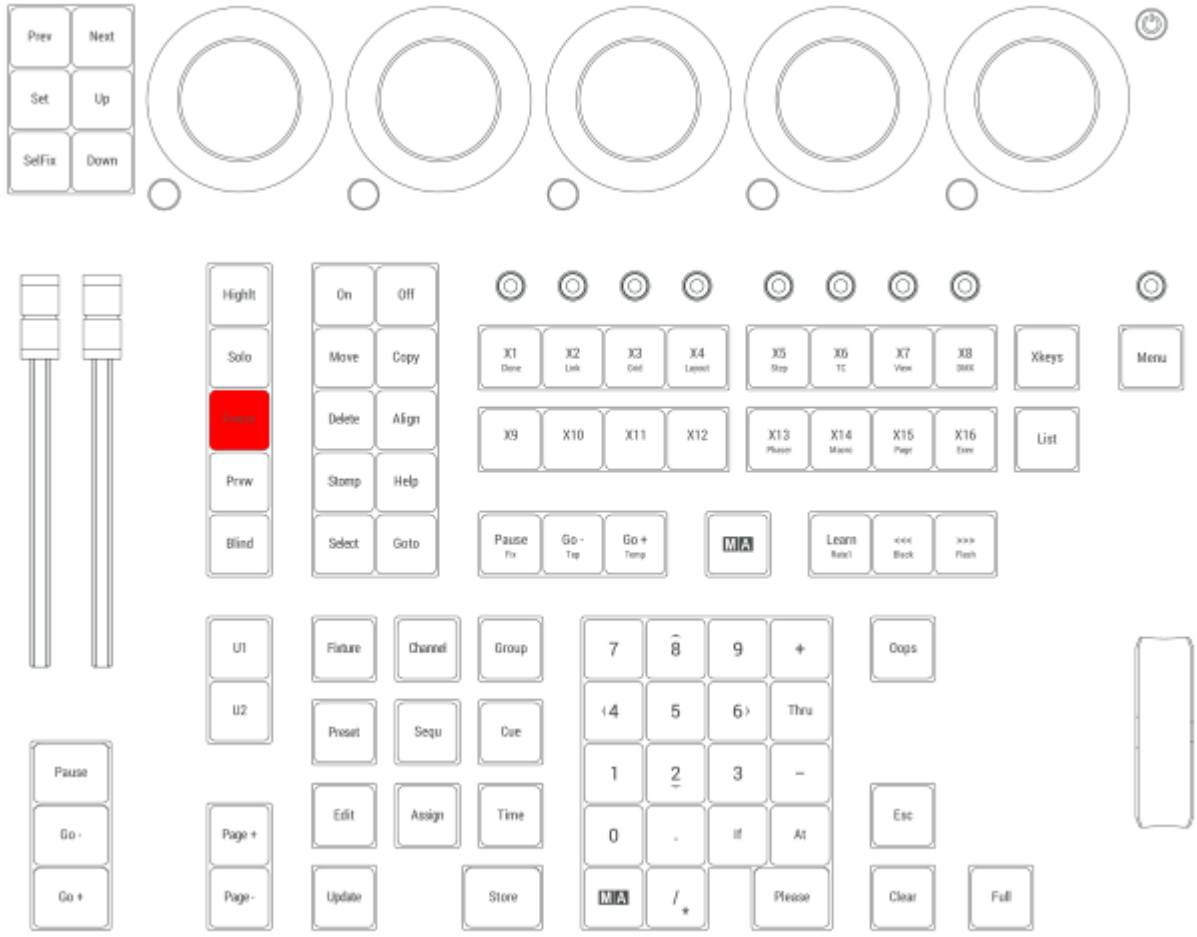
```
OK:Freeze
```

For more information about Freeze, see the **Freeze keyword** topic.

Freeze is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

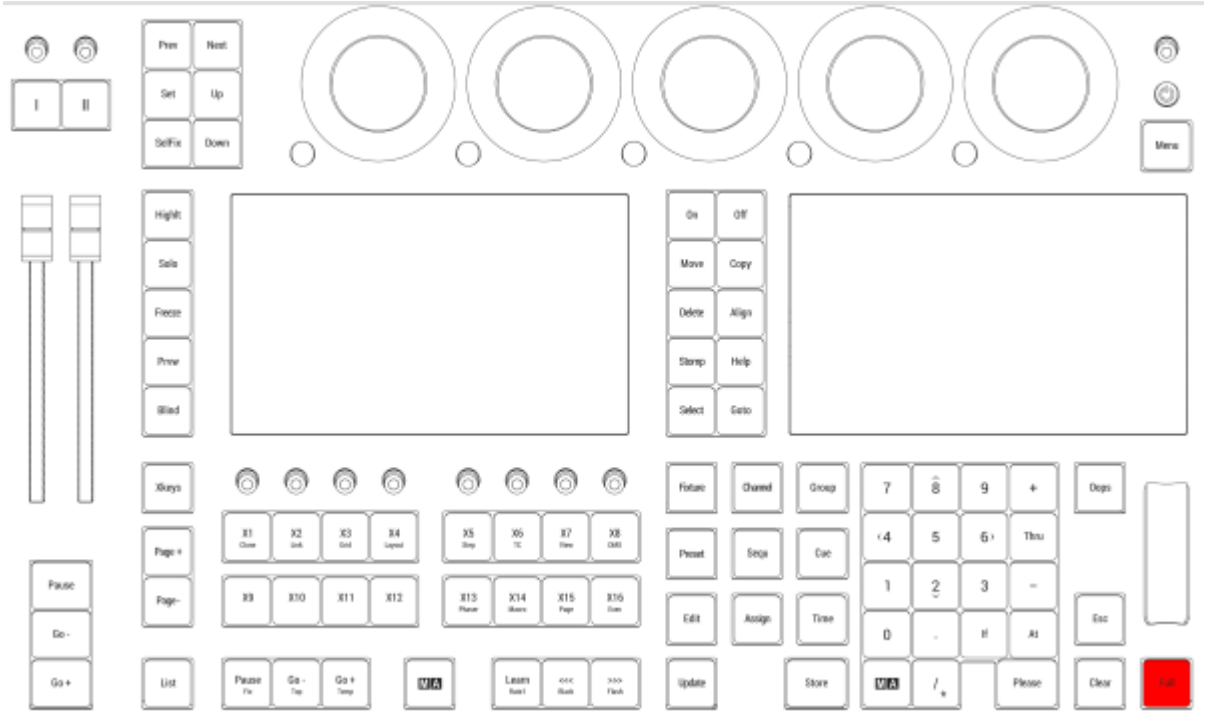
2.15.21. Full key

Pressing Full enters the Full keyword into the command line. The dimmer values for the selected fixtures sets to 100%.

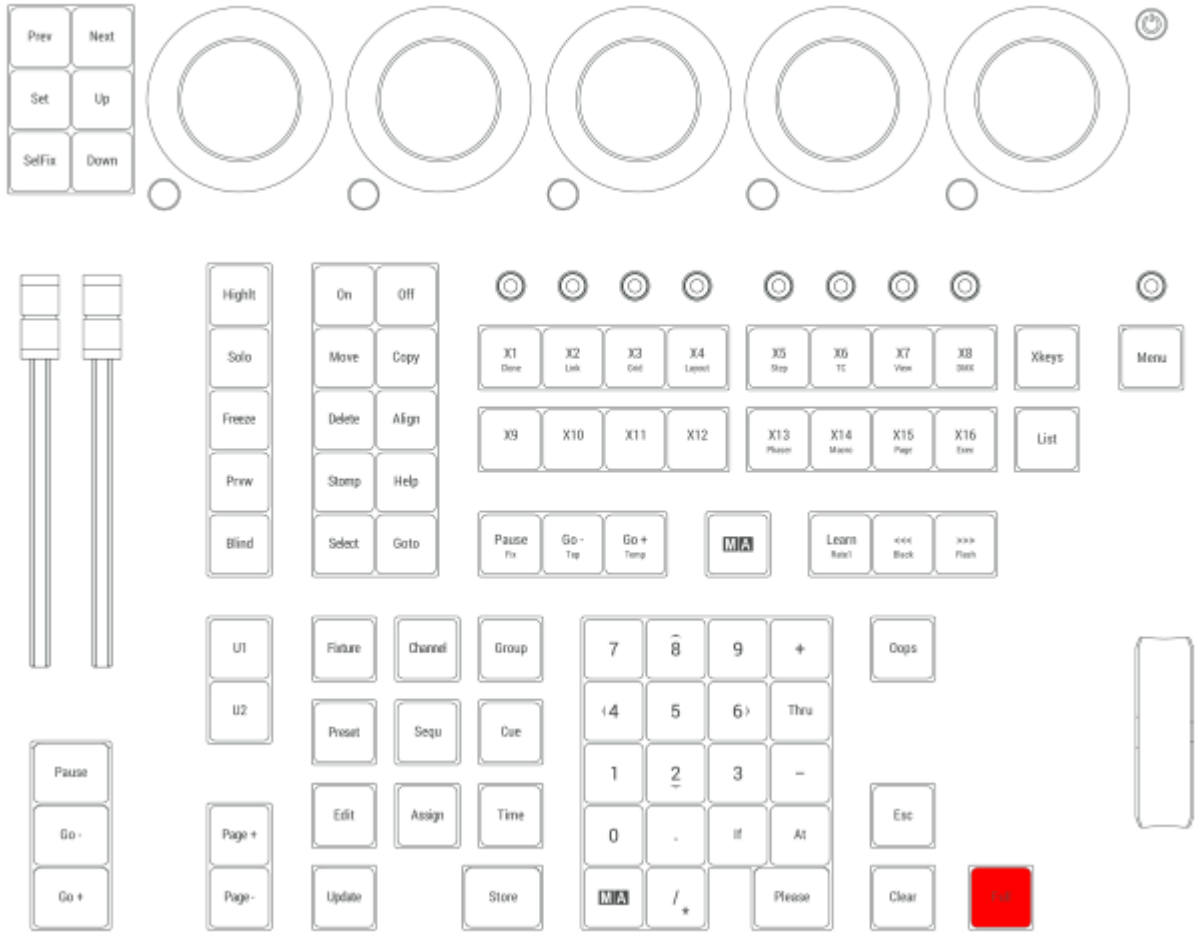
```
User name[Fixture]> Full
```

For more information about Full, see the **Full keyword** topic.

Full is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.22. Fixture key

Pressing Fixture enters the Fixture keyword into the command line.

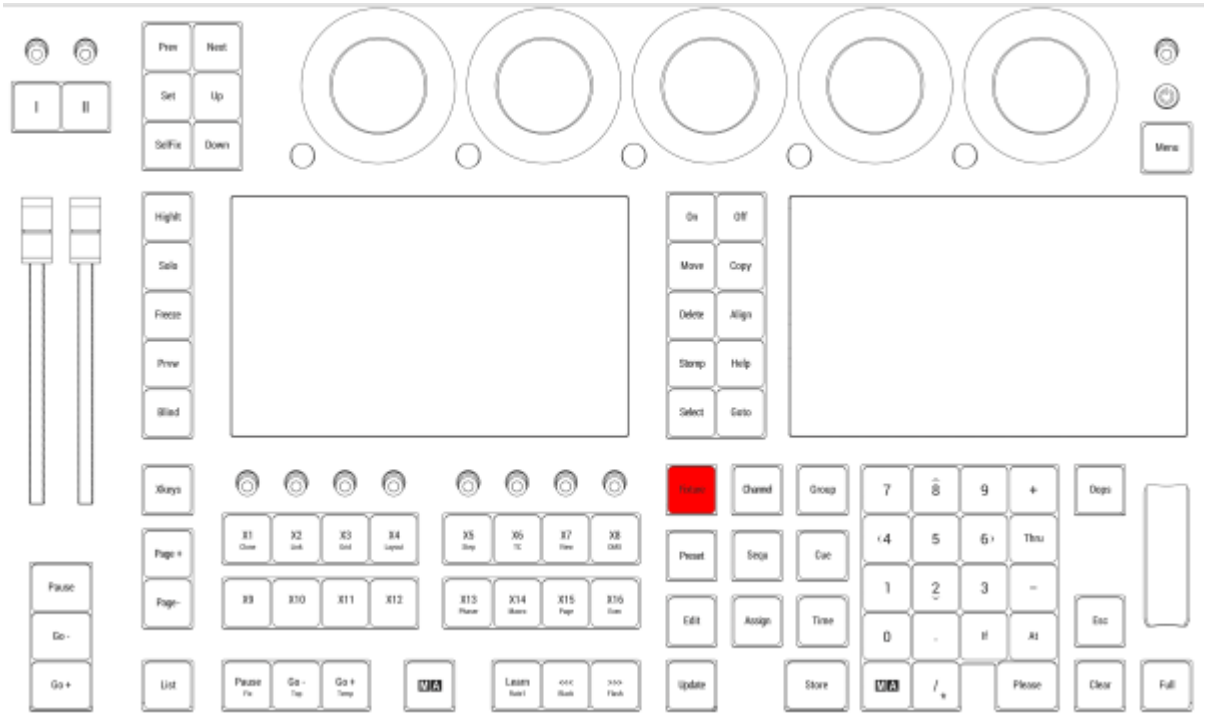
For more information about Fixture, see the **Fixture keyword** topic.

Selection

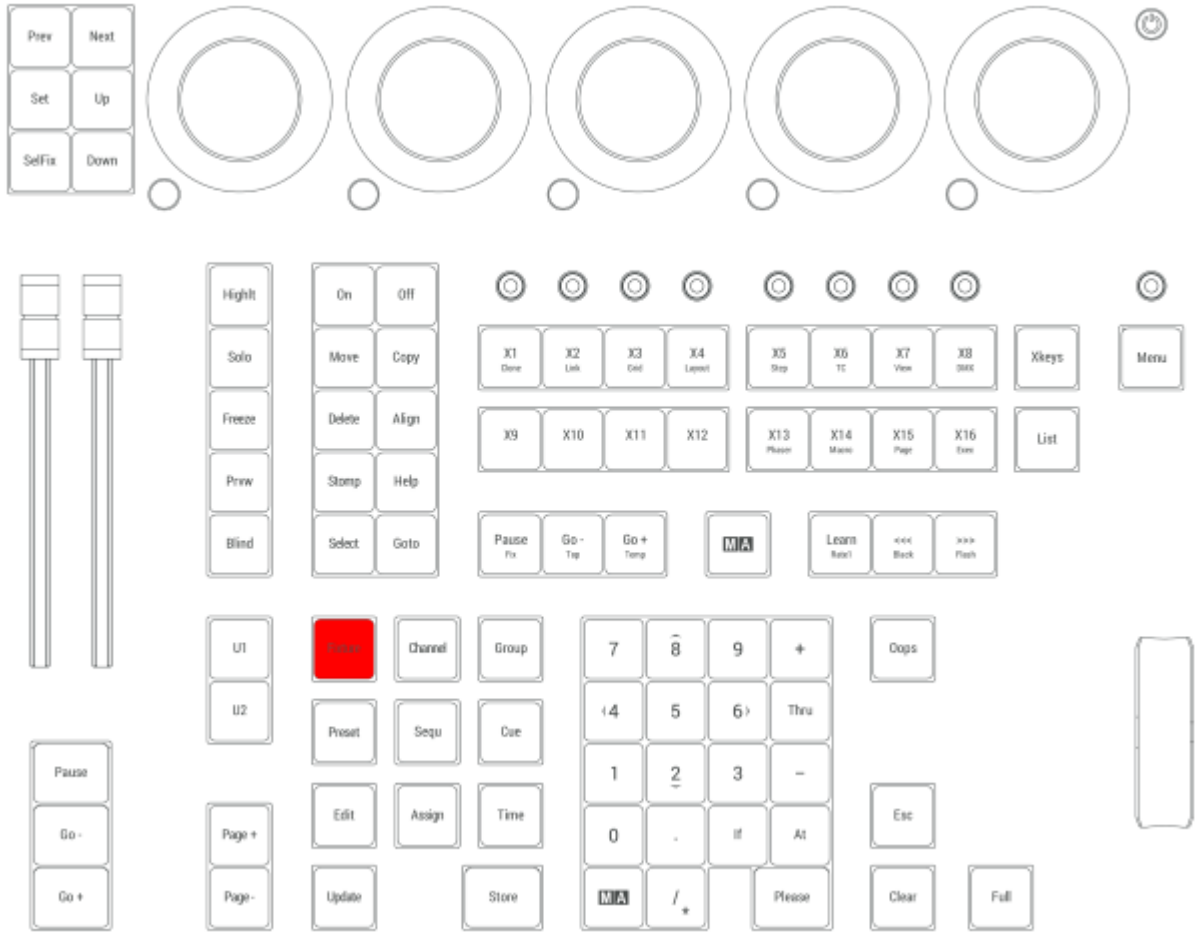
Pressing Fixture Fixture enters the Selection keyword into the command line.

For more information about Selection, see the **Selection keyword** topic.

Fixture is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.23. Group key

Pressing Group enters the Group keyword into the command line.

For more information about Group, see the **Group keyword** topic.

World

Pressing Group Group enters the World keyword into the command line.

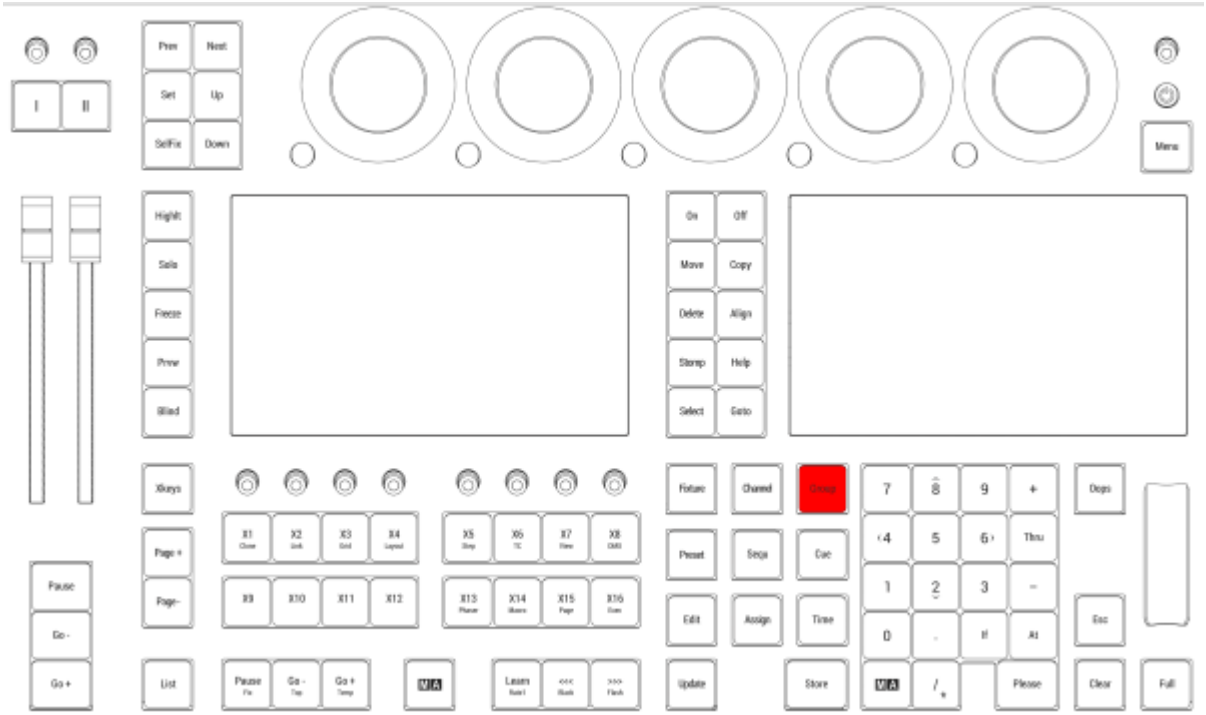
For more information about World, see the **World keyword** topic.

Filter

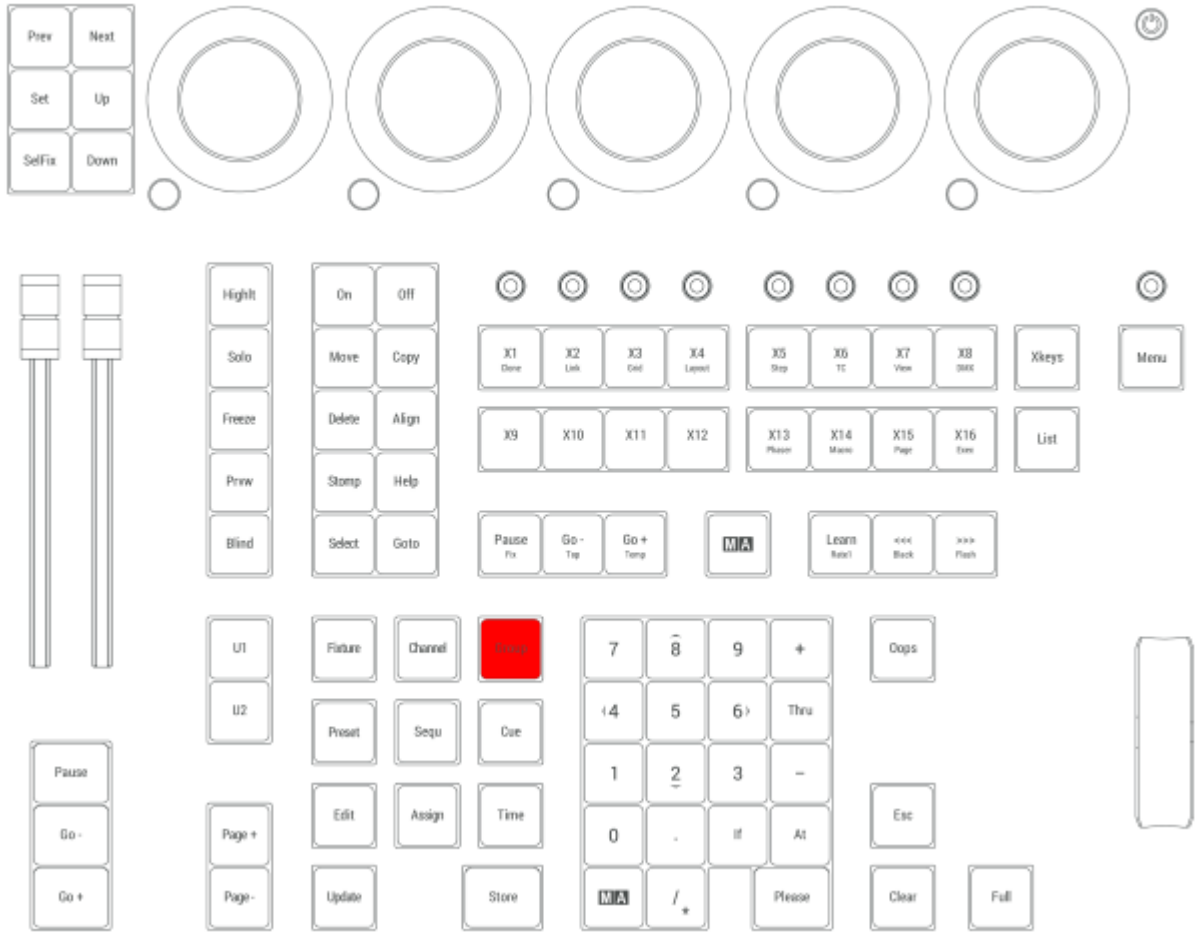
Pressing Group Group Group enters the Filter keyword into the command line.

For more information about Filter, see the **Filter keyword** topic.

Group is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.24. Goto key

Pressing Goto enters the Goto keyword into the command line.

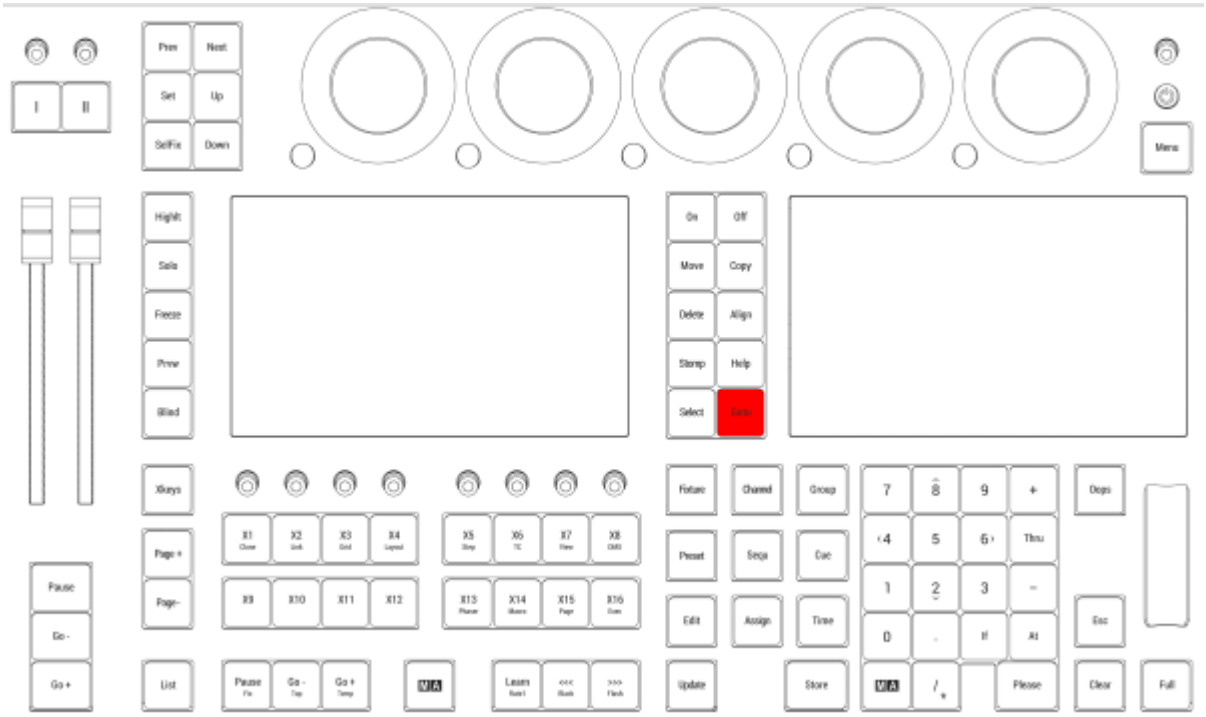
For more information about Goto, see the **Goto keyword** topic.

Load

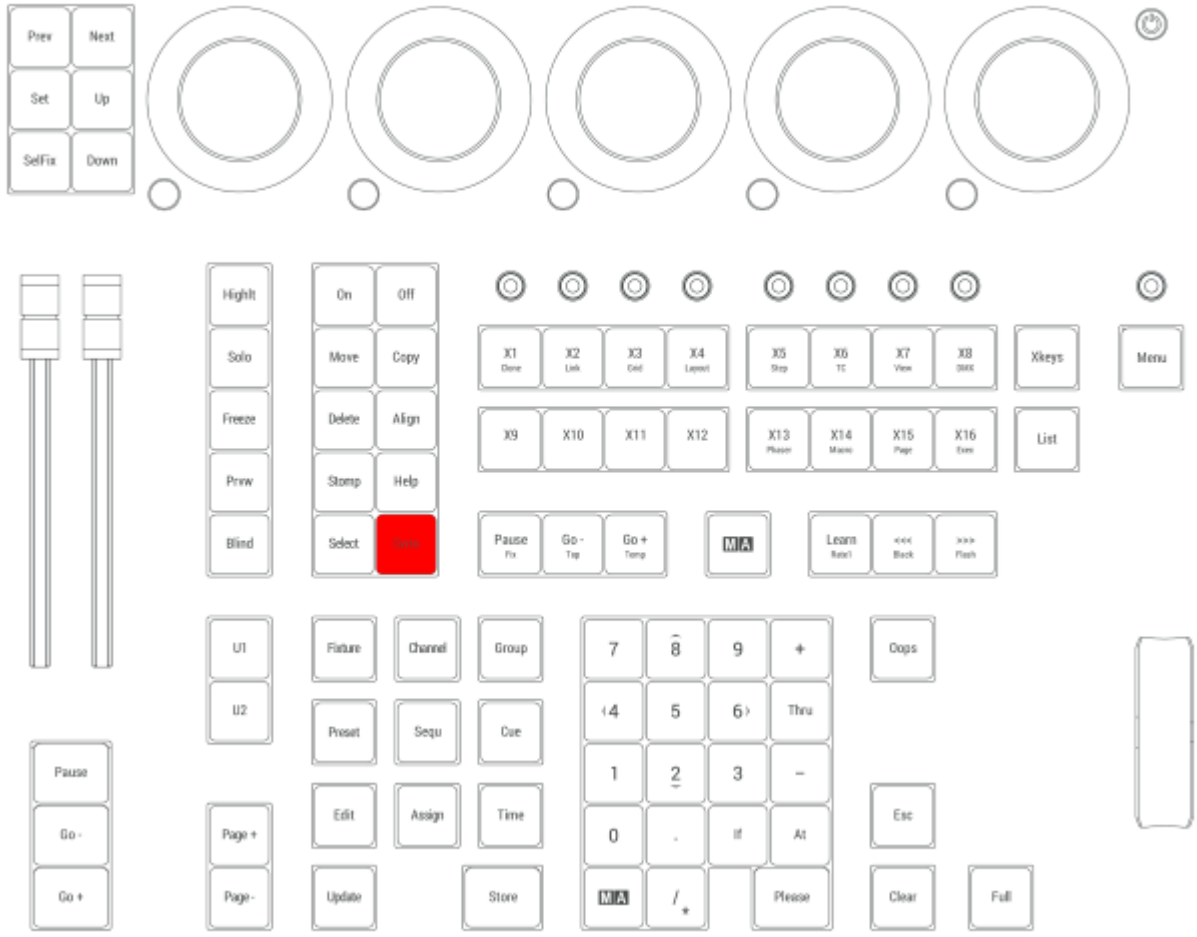
Pressing Goto Goto enters the Load keyword into the command line.

For more information about Load, see the **Load keyword** topic.

Goto is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.25. Go+ [large] key

Go+ Executor is executed immediately in the command line when pressing Go+ [large].

```
User name[Fixture]> Go+ Executor
```

For more information about Go+, see the **Go+ keyword** topic.

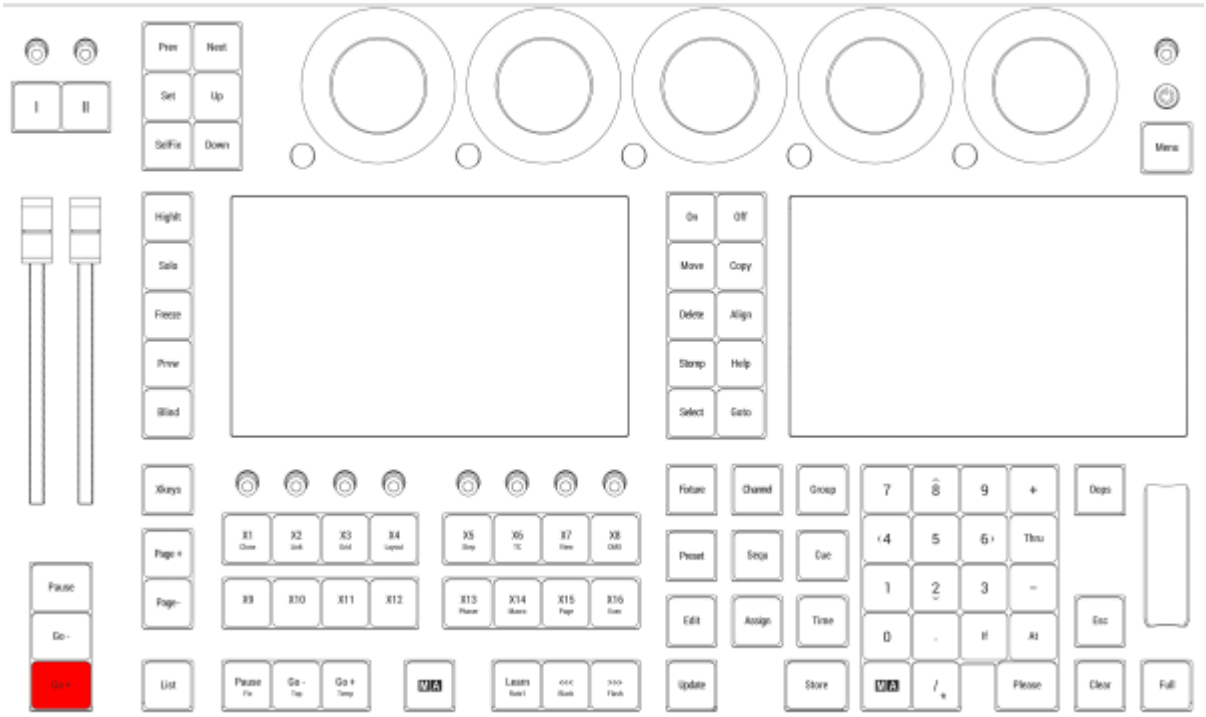
Go+ Loaded

Pressing MA + Go+ [large] executes the Go+ Loaded keyword in the command line.

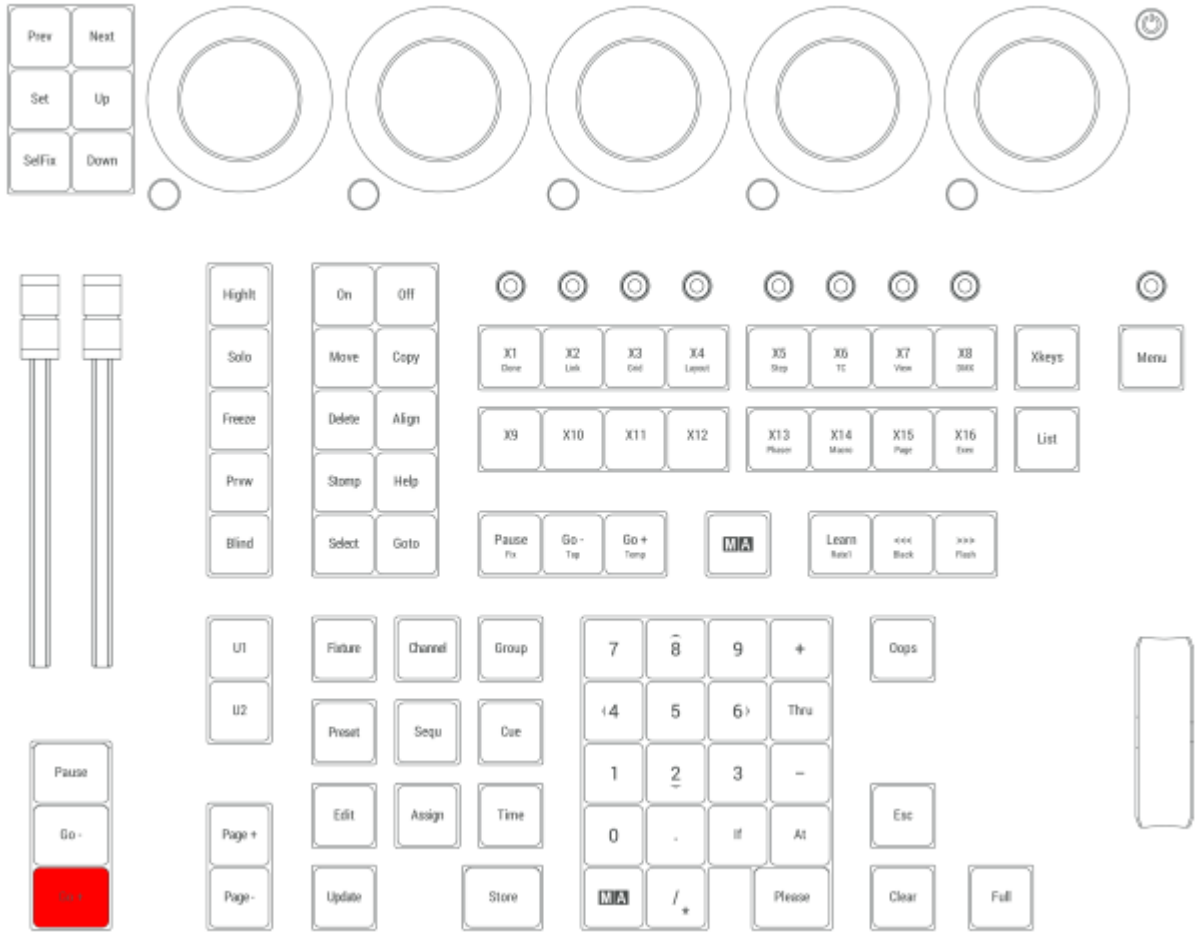
```
User name[Fixture]> Go+ Loaded
```

For more information about Go+, see the **Go+ keyword** topic.

Go+ [large] is located in the master section under the two master faders.



Location on grandMA3 full-size and grandMA3 light consoles



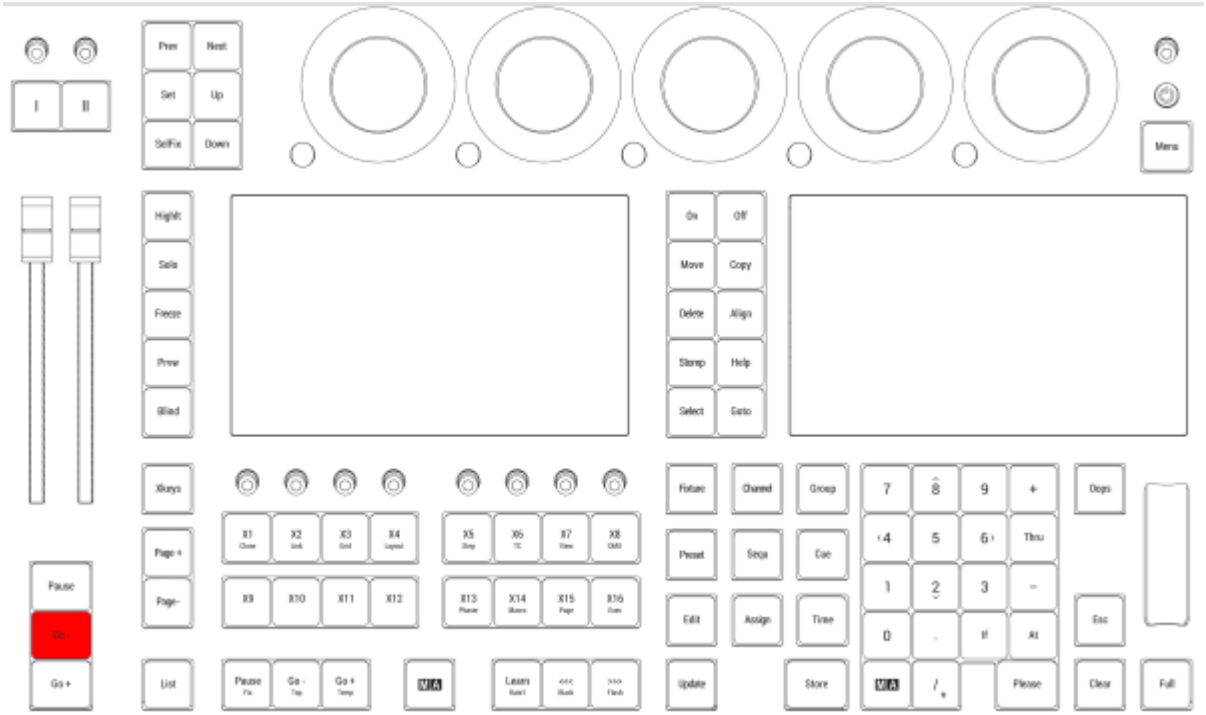
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.26. Go- [large] key

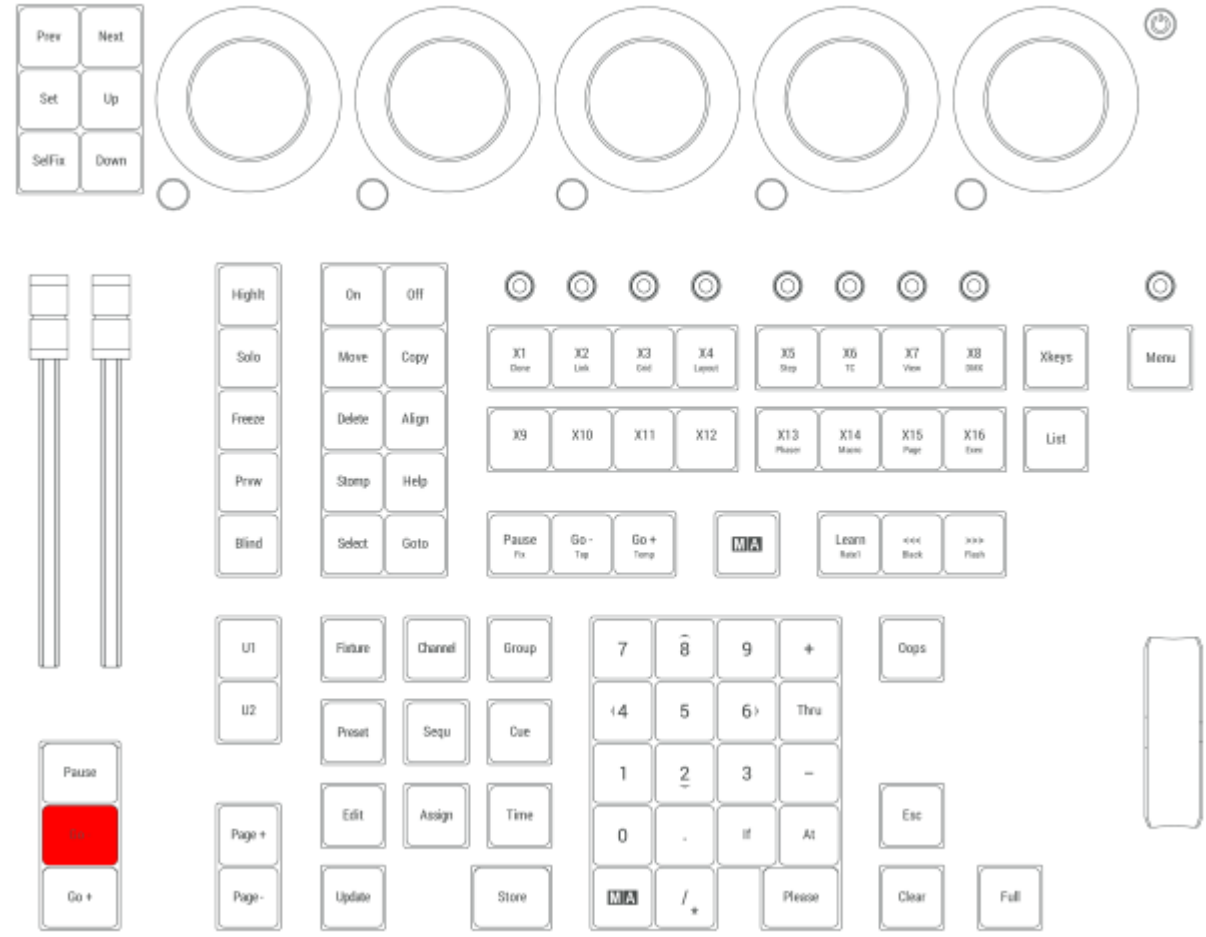
Go- Executor is executed immediately in the command line when pressing Go- [large].

For more information about Go-, see the **Go- keyword** topic.

Go- [large] is located in the master section under the two master faders.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.27. Go+ | Temp key

Pressing Go+ enters the Go+ keyword into the command line.

For more information about Go+, see the **Go+ keyword** topic.

Temp

Pressing and holding MA + Go+ enters the Temp keyword into the command line.

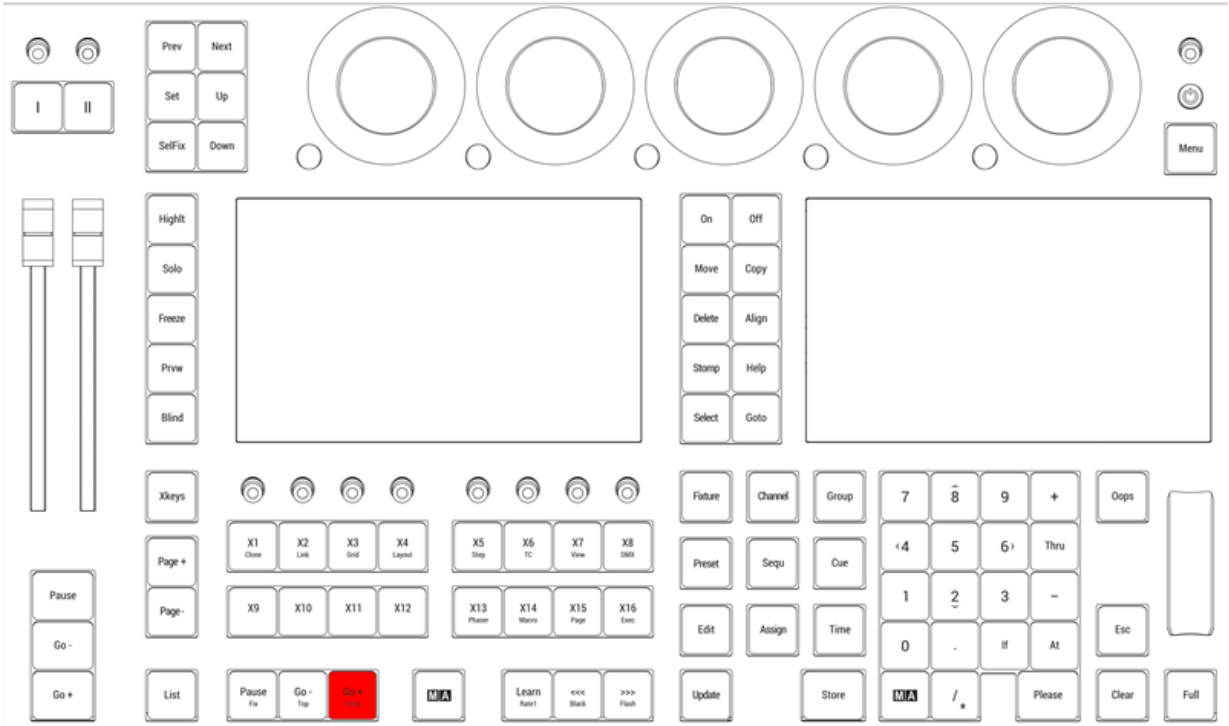
For more information about Temp, see the **Temp keyword** topic.

Toggle

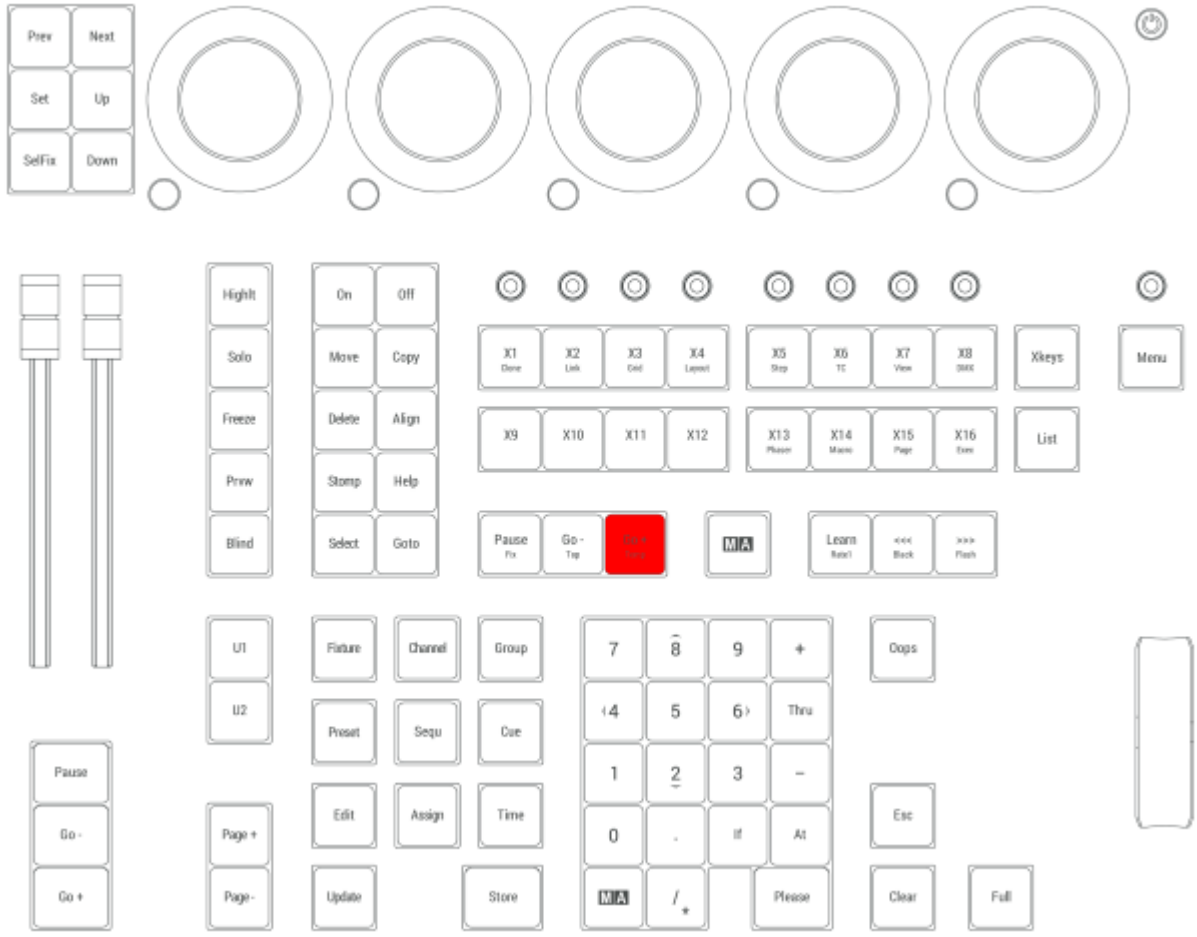
Pressing and holding MA + Go+ Go+ enters the Toggle keyword into the command line.

For more information about Toggle, see the **Toggle keyword** topic.

Go+ is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.28. Go- | Top key

Pressing Go- enters the Go- keyword into the command line.

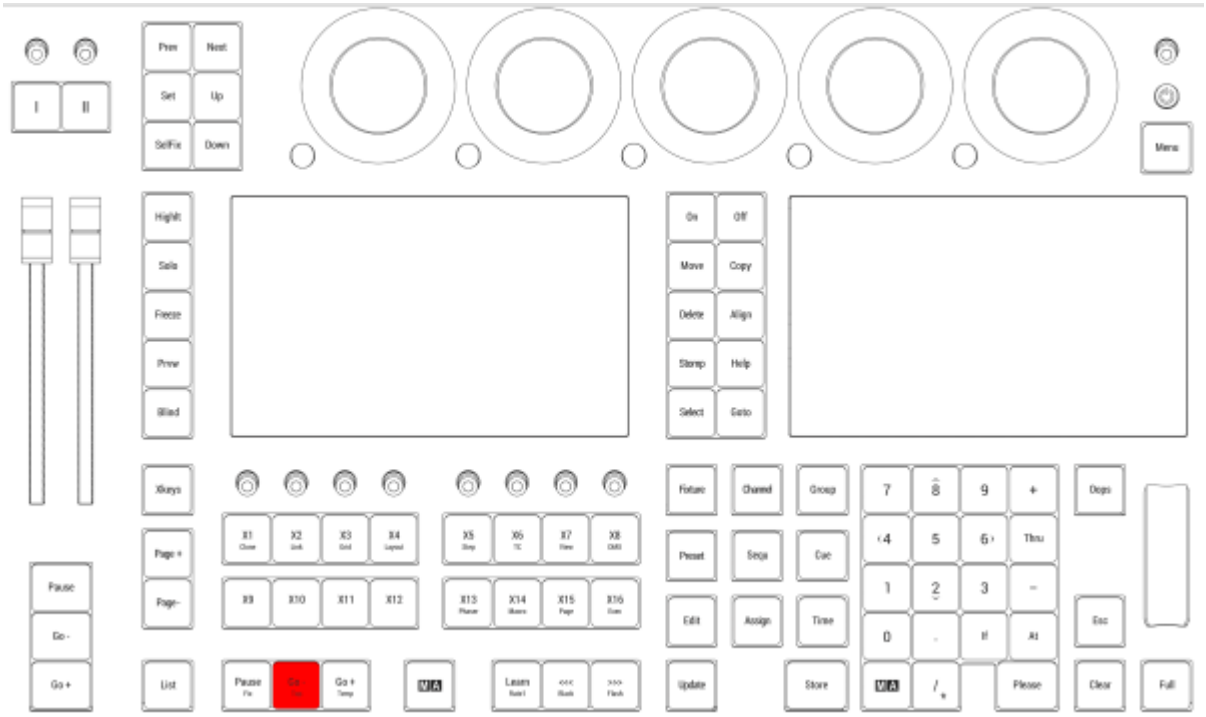
For more information about Go-, see the **Go- keyword** topic.

Top

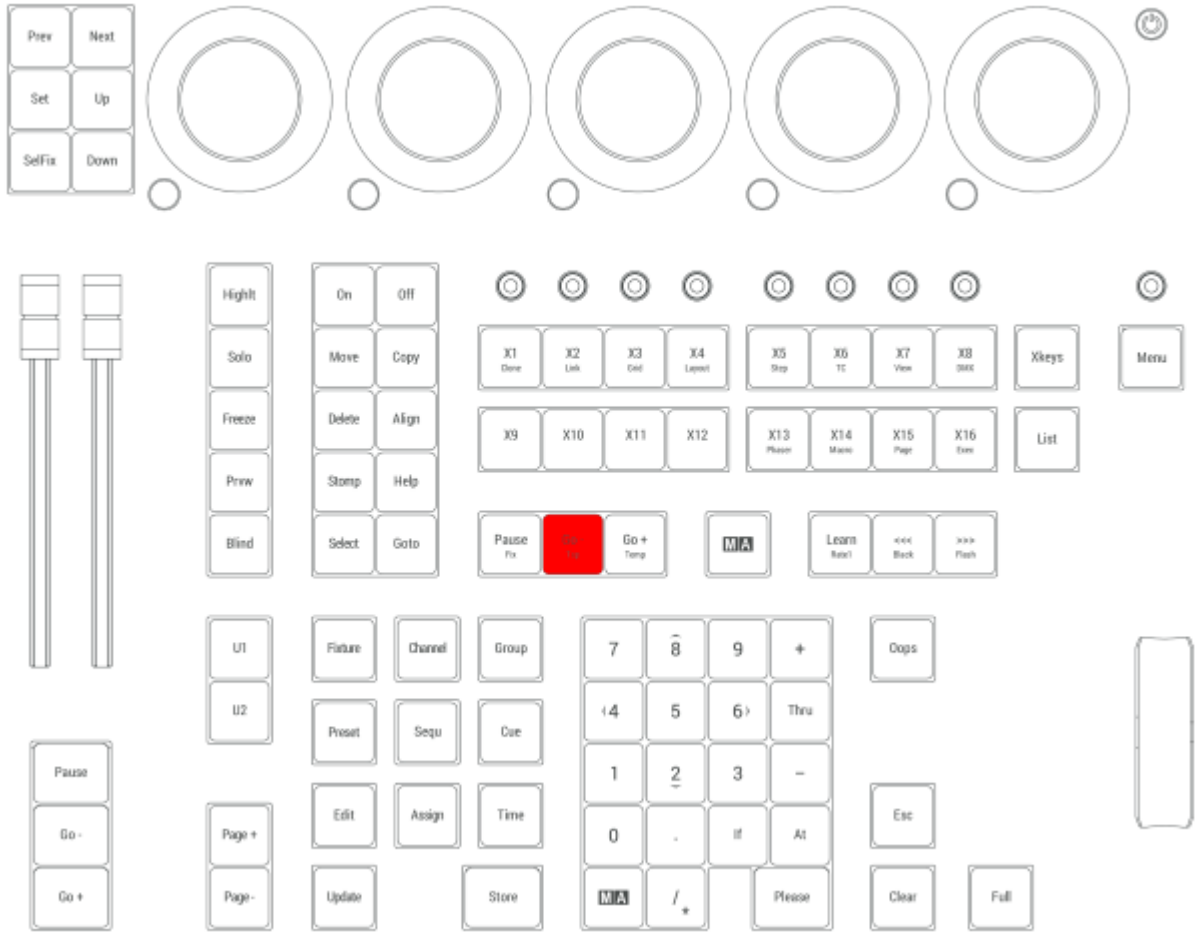
Pressing and holding MA + Go- enters the Top keyword into the command line.

For more information about Top, see the **Top keyword** topic.

Go- is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

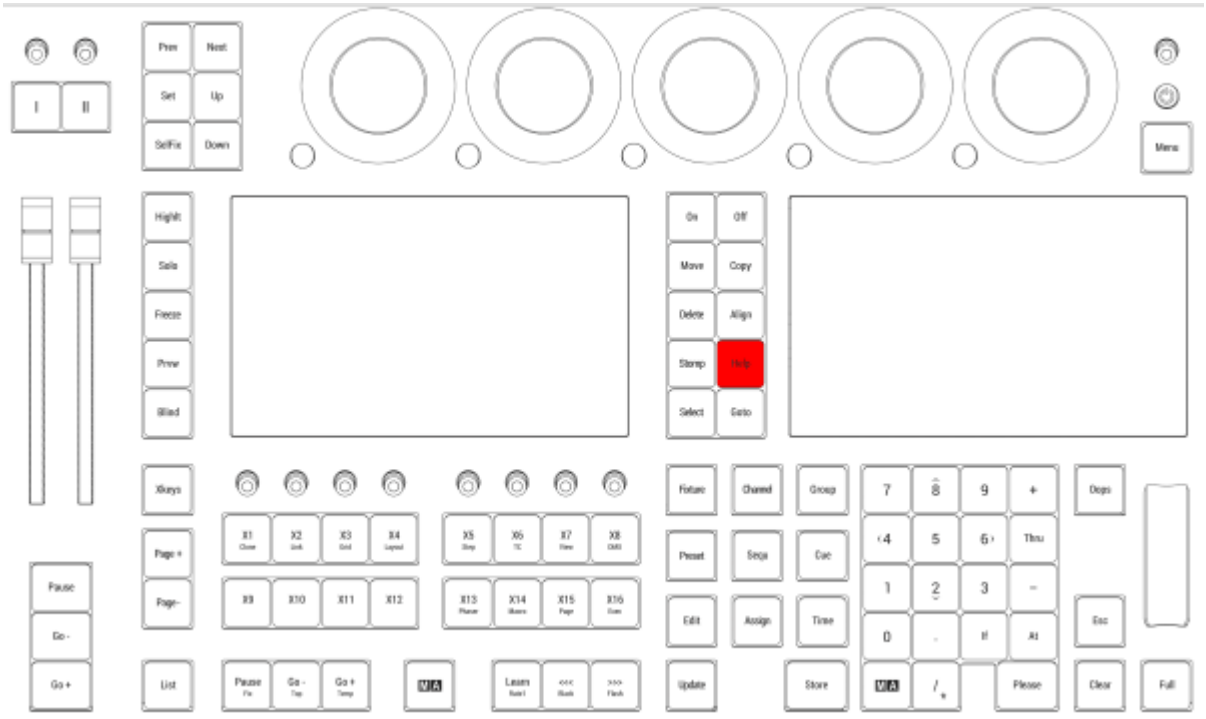
2.15.29. Help key

Pressing Help enters the Help keyword into the command line.

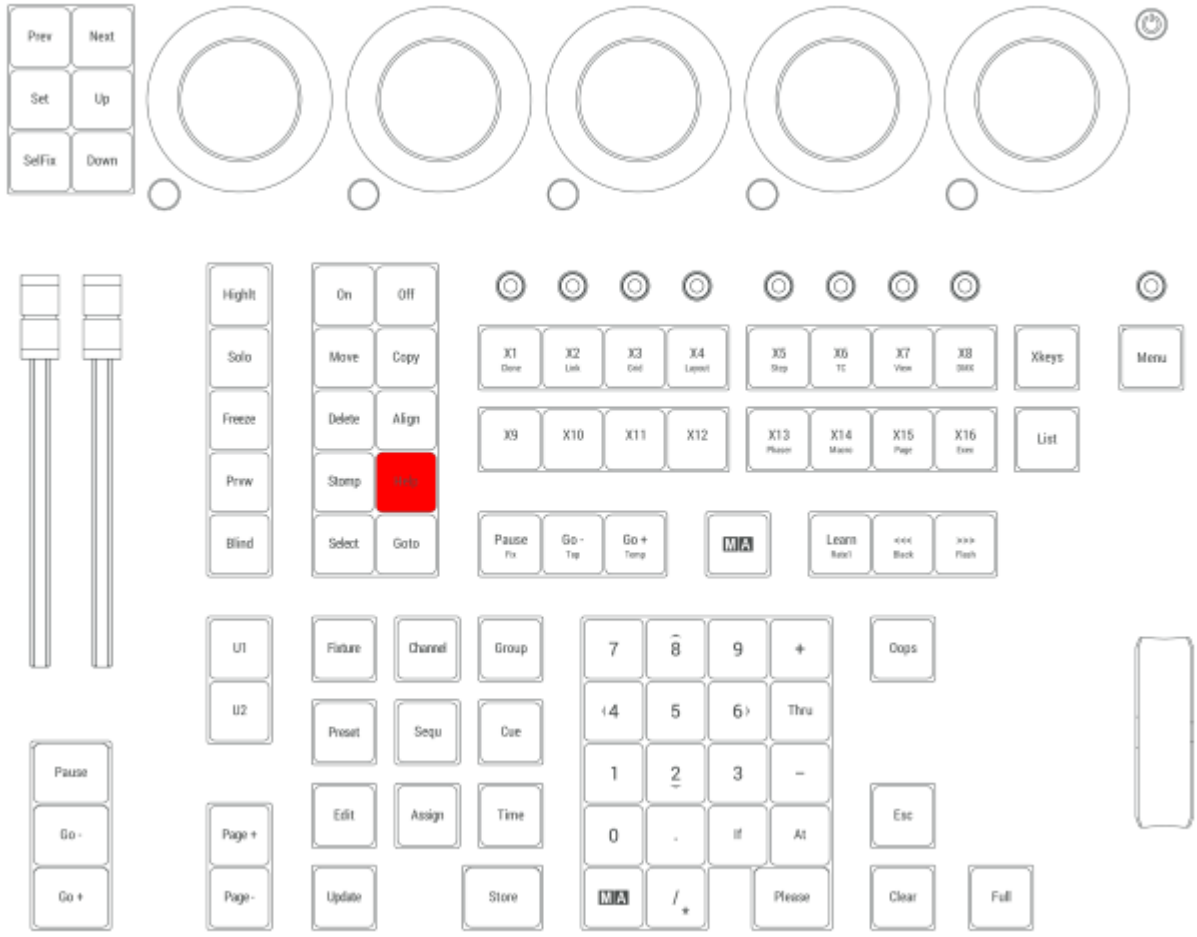
Pressing Help and an user interface element opens the corresponding help.

For more information about Help, see the **Help keyword** topic.

Help is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



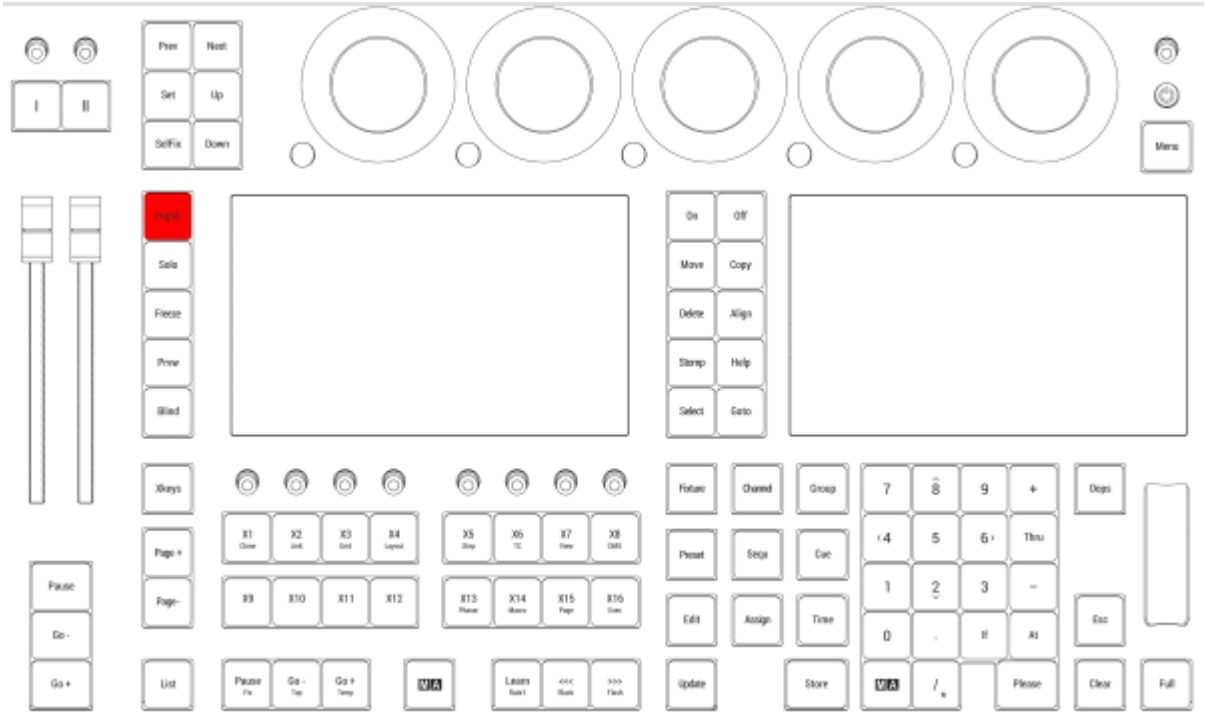
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.30. Highlt [Highlight] key

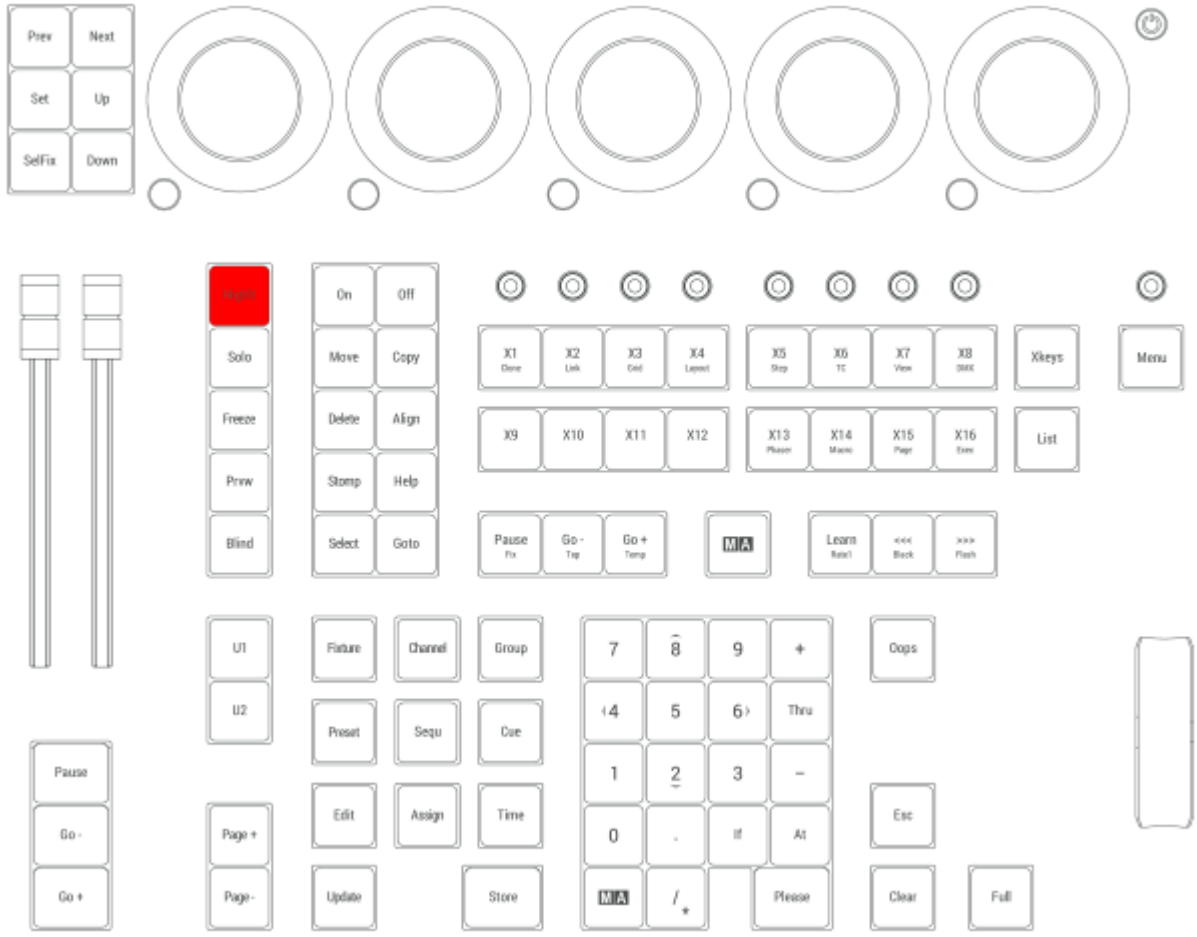
Pressing Highlt toggles the highlight function. For more information about Highlight, see the **Highlight keyword** topic.

Pressing MA + Highlt toggles the lowlight function. For more information about Lowlight, see the **Lowlight keyword** topic.

Highlt is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.31. If key

IfOutput

Pressing If enters the IfOutput keyword into the command line.

For more information about IfOutput, see the **IfOutput keyword** topic.

IfActive

Pressing If If enters the IfActive keyword into the command line.

For more information about IfActive, see the **IfActive keyword** topic.

IfProg

Pressing If If If enters the IfProg keyword into the command line.

For more information about IfProg, see the **IfProg keyword** topic.

If

Pressing If If If If enters the If keyword into the command line.

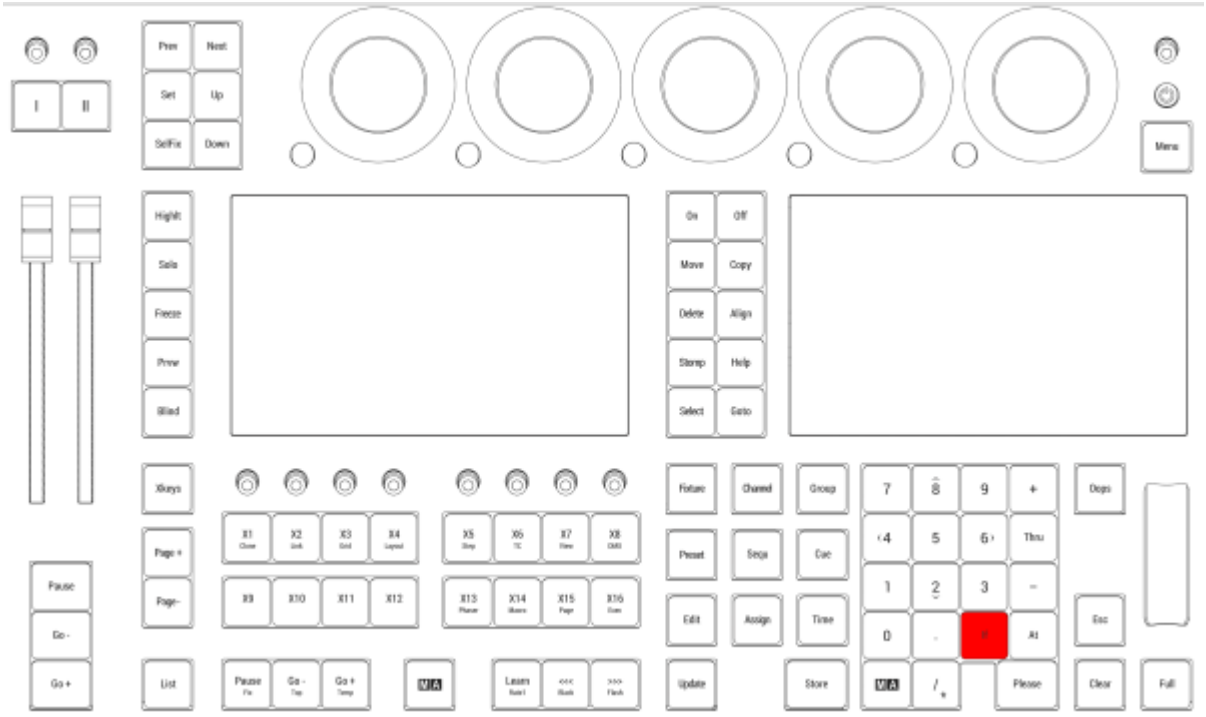
For more information about If, see the **If keyword** topic.

EndIf

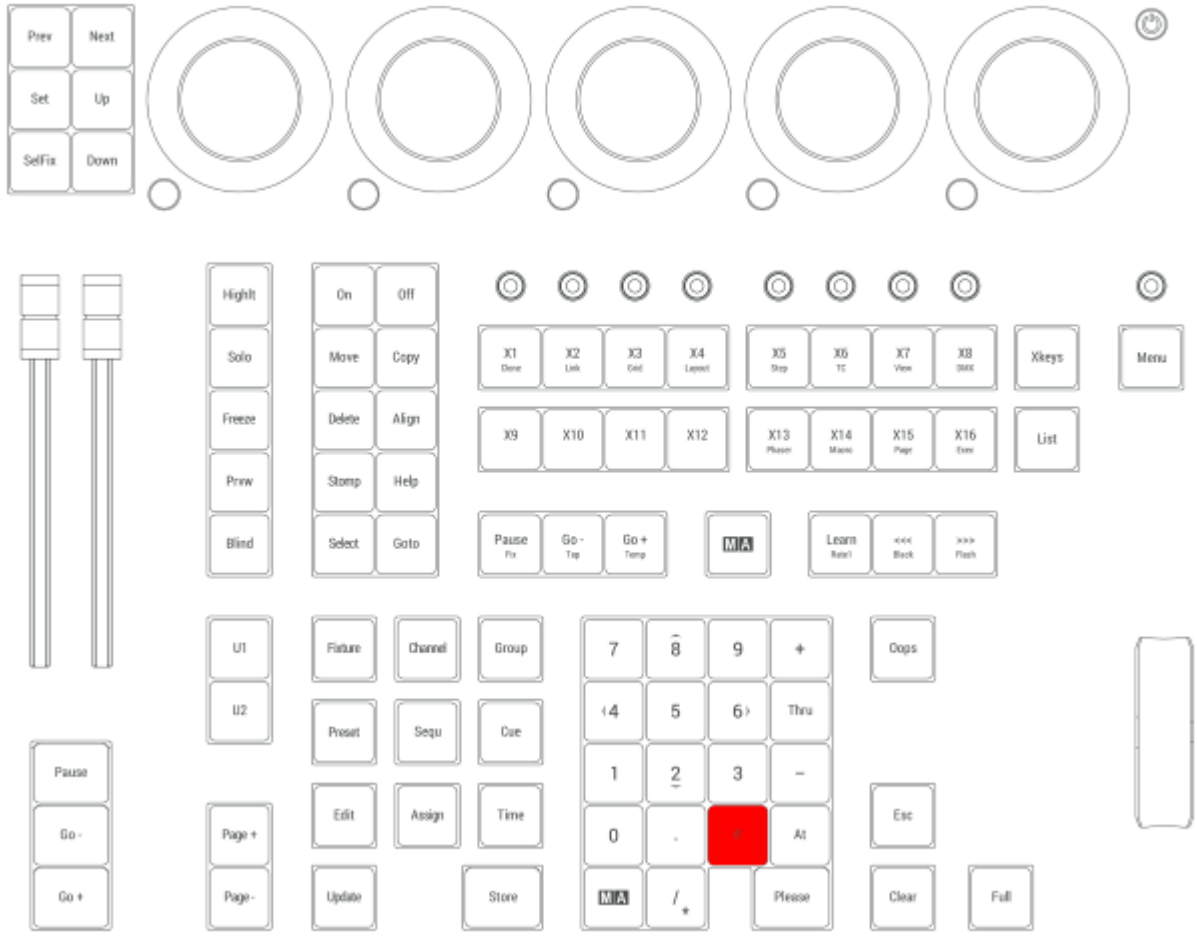
Pressing If again once the If command is already in the command line enters the EndIf keyword into the command line.

For more information about EndIf, see the **EndIf keyword** topic.

If is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.32. Learn | Rate1 key

Pressing Learn enters the LearnRate keyword into the command line.

```
User name[Fixture]> LearnRate
```

For more information about Learn, see the **LearnRate keyword** topic.

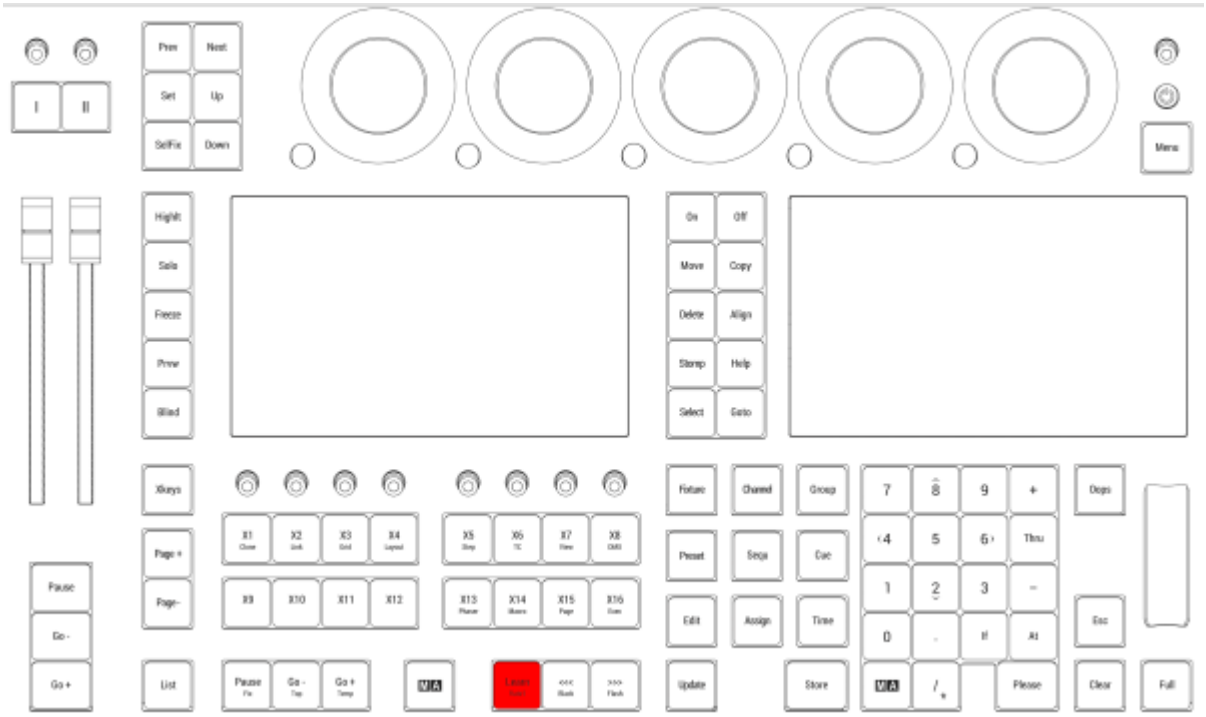
Rate1

Pressing and holding MA + Learn enters the Rate1 keyword into the command line.

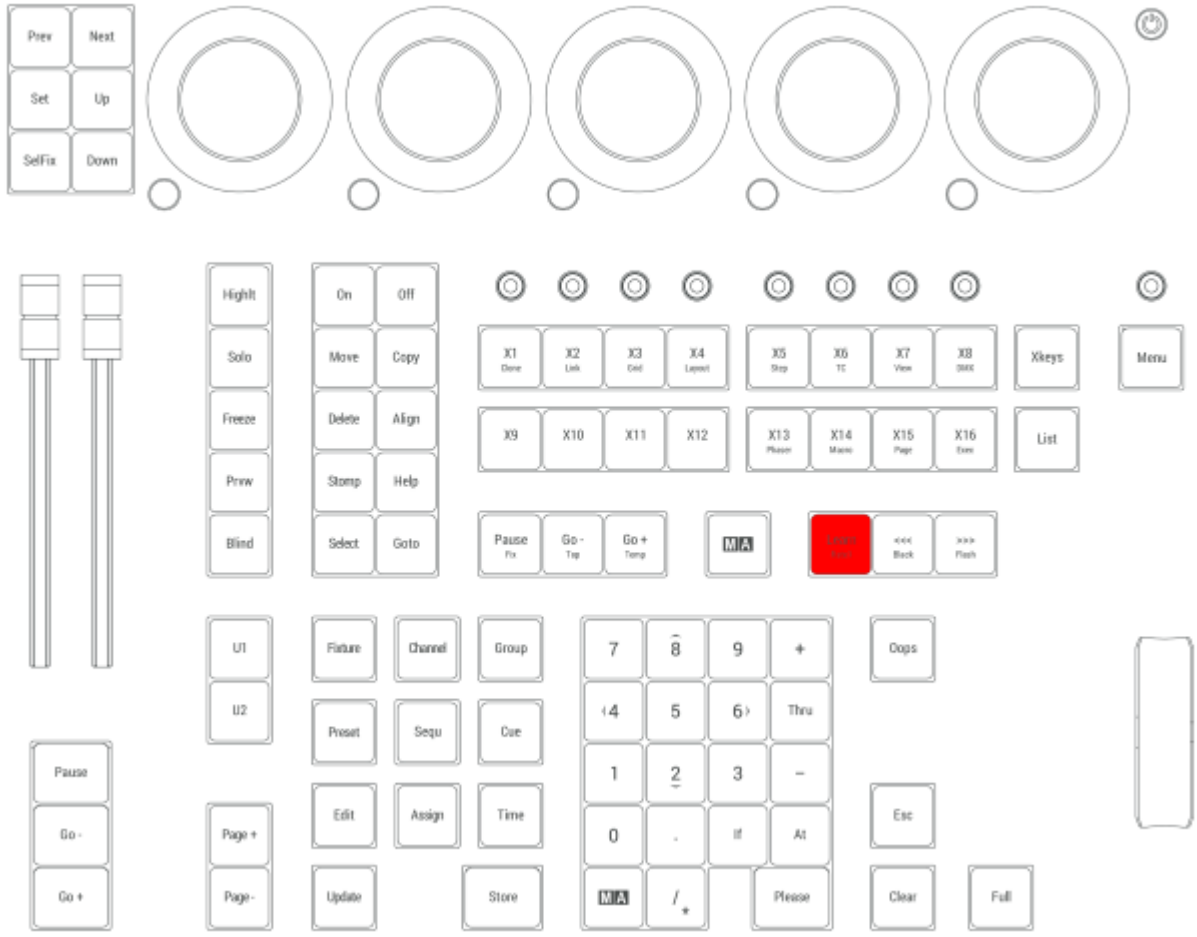
```
User name[Fixture]> Rate1
```

For more information about Rate1, see the **Rate1 keyword** topic.

Learn is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.33. List key

Pressing List enters the List keyword into the command line.

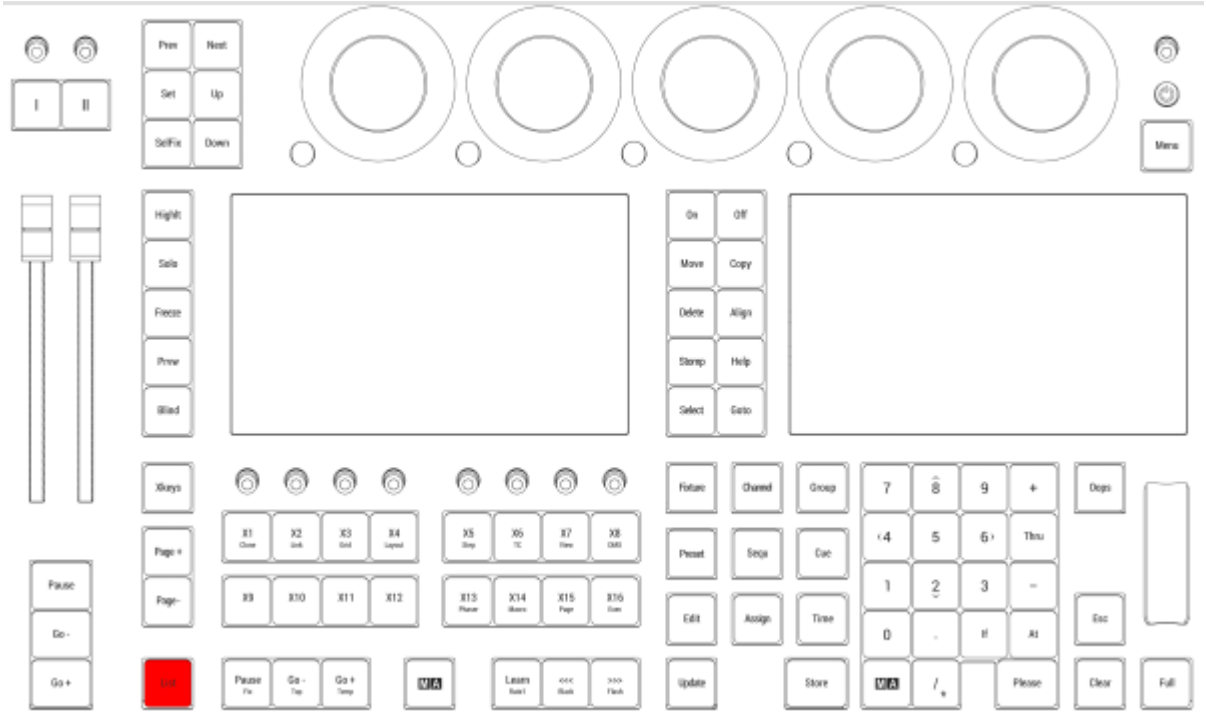
For more information about List, see the **List keyword** topic.

ListRef

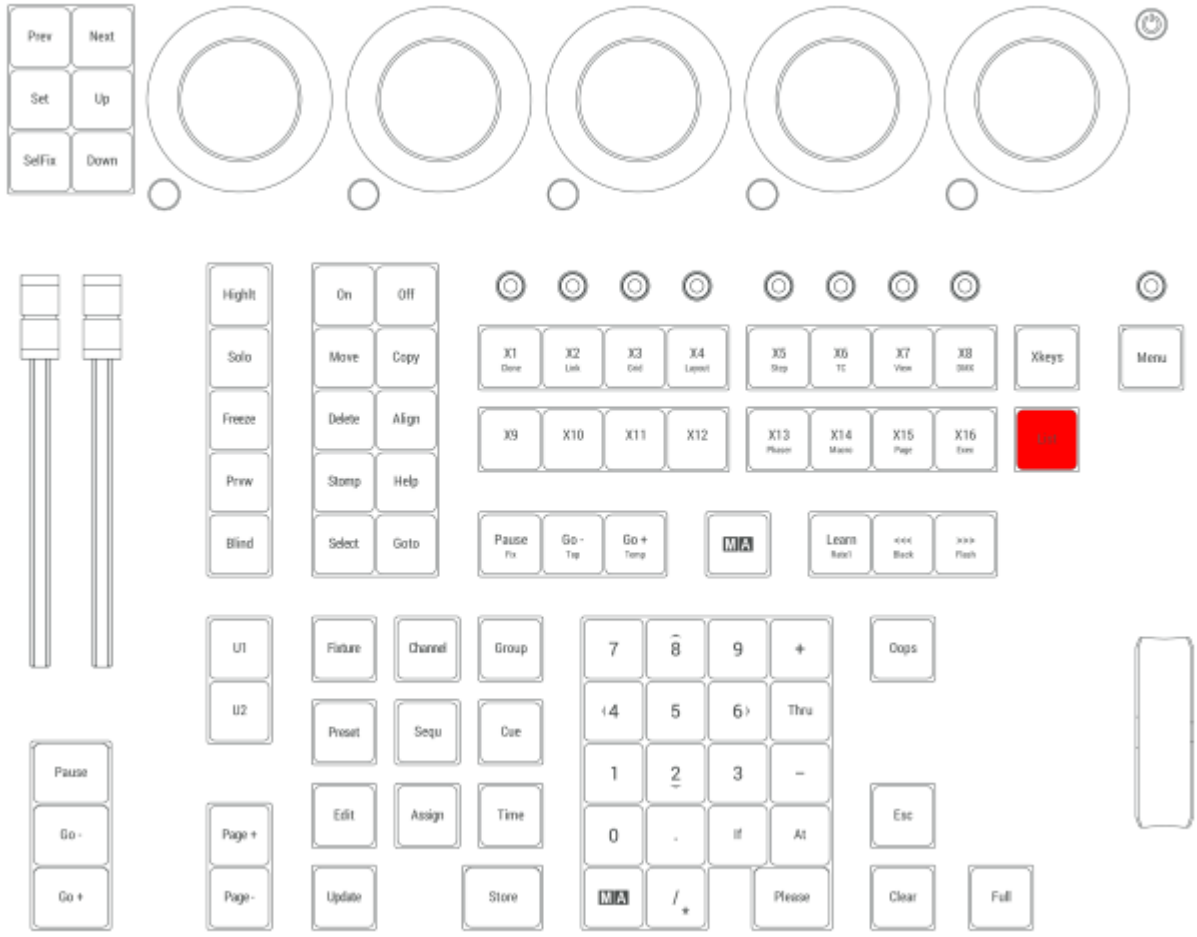
Pressing and holding MA + List enters the ListRef keyword into the command line.

For more information about List, see the **ListRef keyword** topic.

List is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

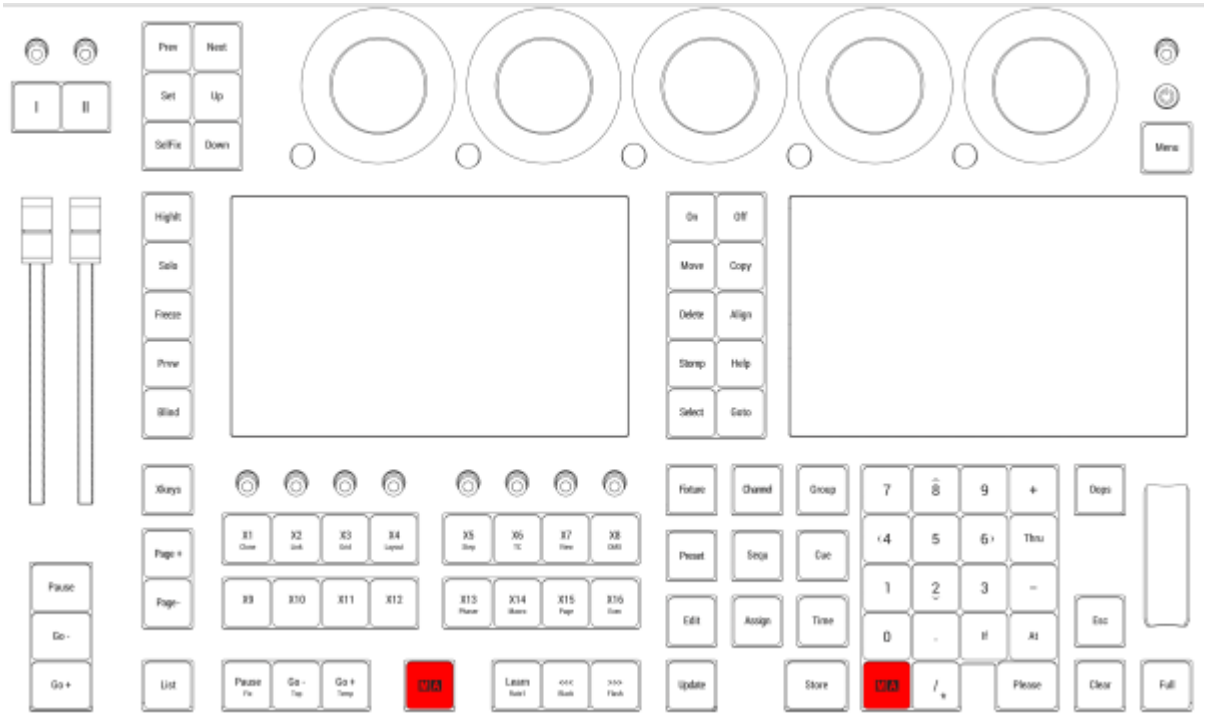
2.15.34. MA key

Pressing and holding MA in combination with other keys, gives shortcuts to other functions. On both console layouts there are two MA keys. These are identical and have the same function.

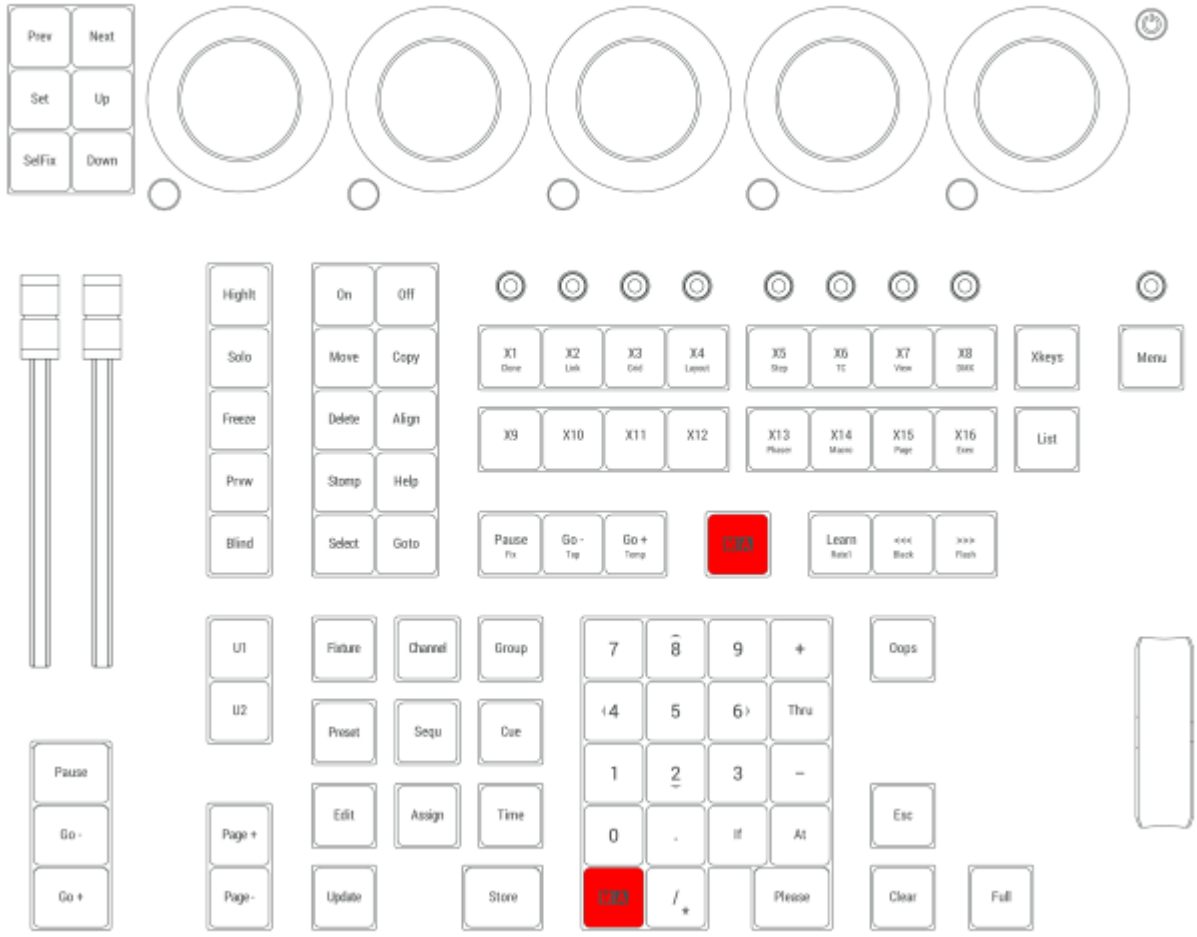
Pressing MA + Please forces the focus to the command line.

To get a complete list of functions, please see the **Keys topic**.

MA is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



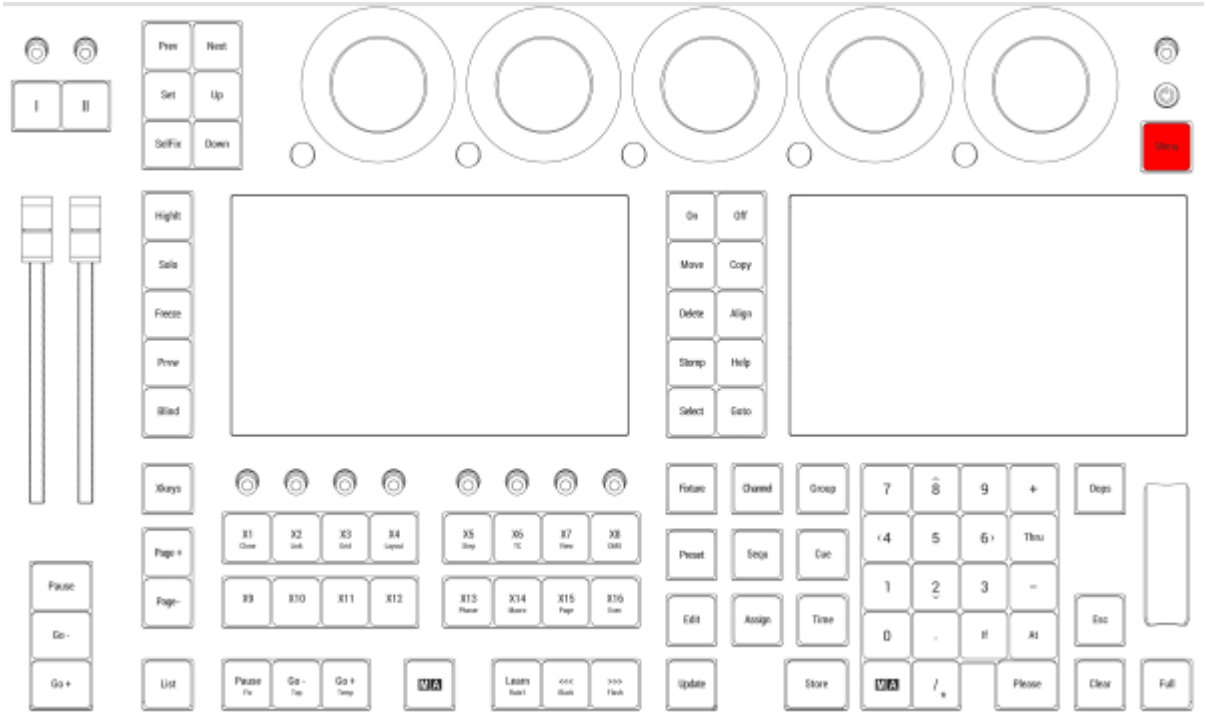
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.35. Menu key

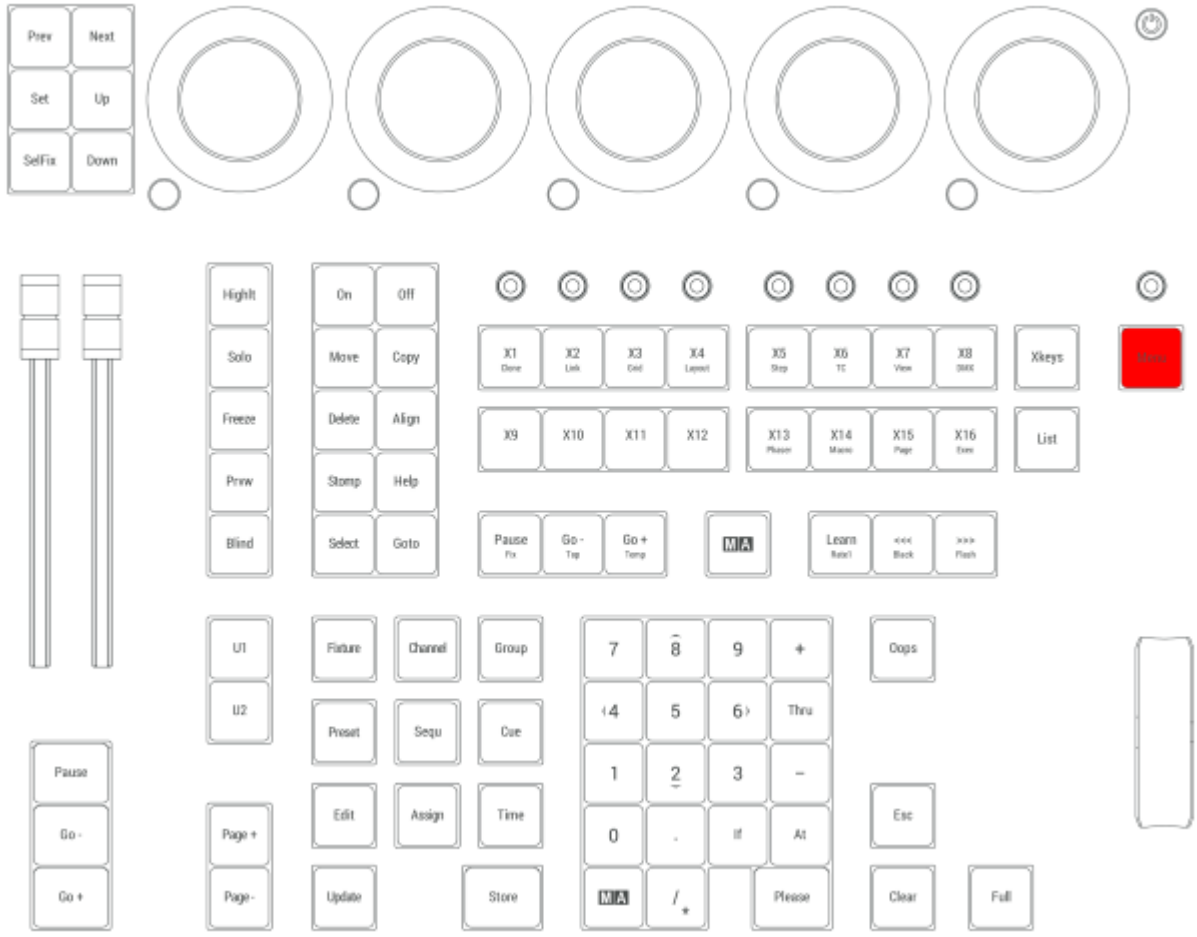
Pressing Menu opens the Menu pop-up.

For more information about Menu, see the **Menu keyword** topic.

Menu is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.36. Move key

Pressing Move enters the Move keyword into the command line.

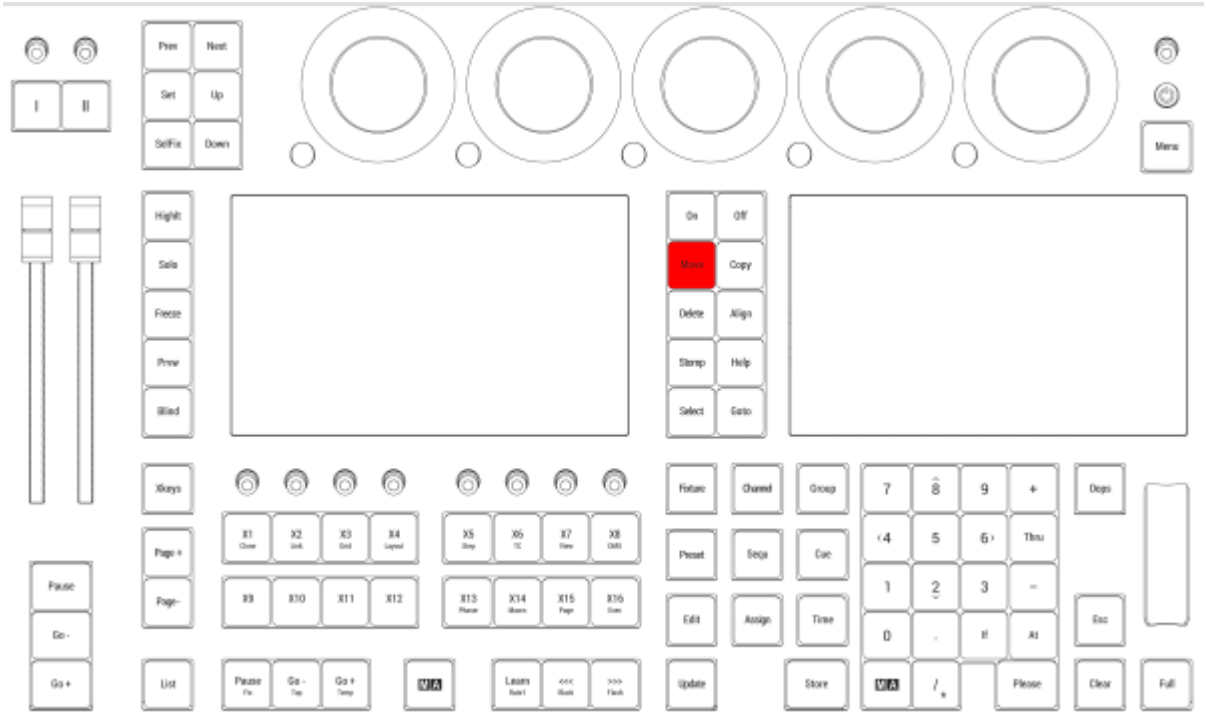
For more information about Move, see the **Move keyword** topic.

Exchange

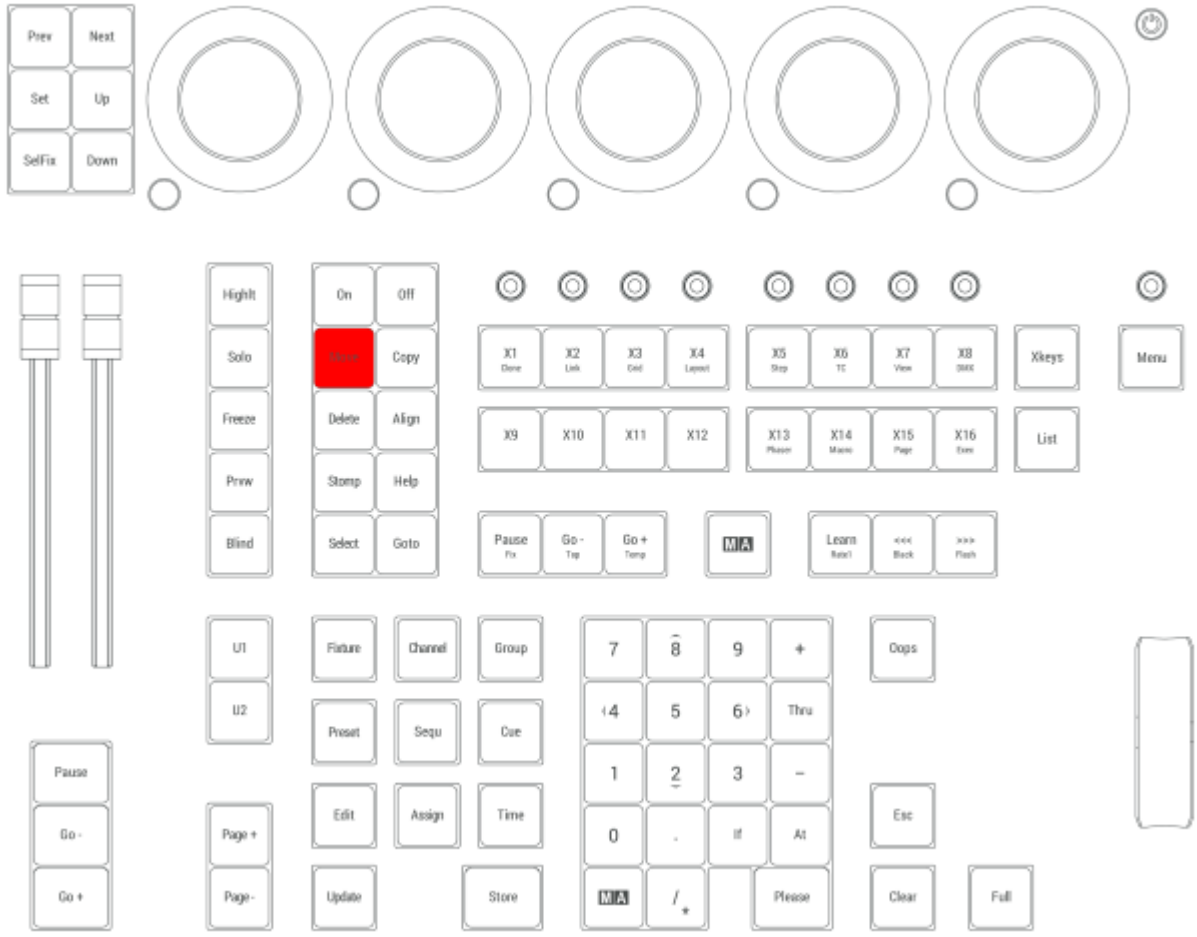
Pressing Move Move enters the Exchange keyword into the command line.

For more information about Exchange, see the **Exchange Keyword** topic.

Move is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.37. Next key

Pressing Next executes the Next keyword in the command line.

```
User name[Fixture]> Next
```

For more information about Next, see the **Next keyword** topic.

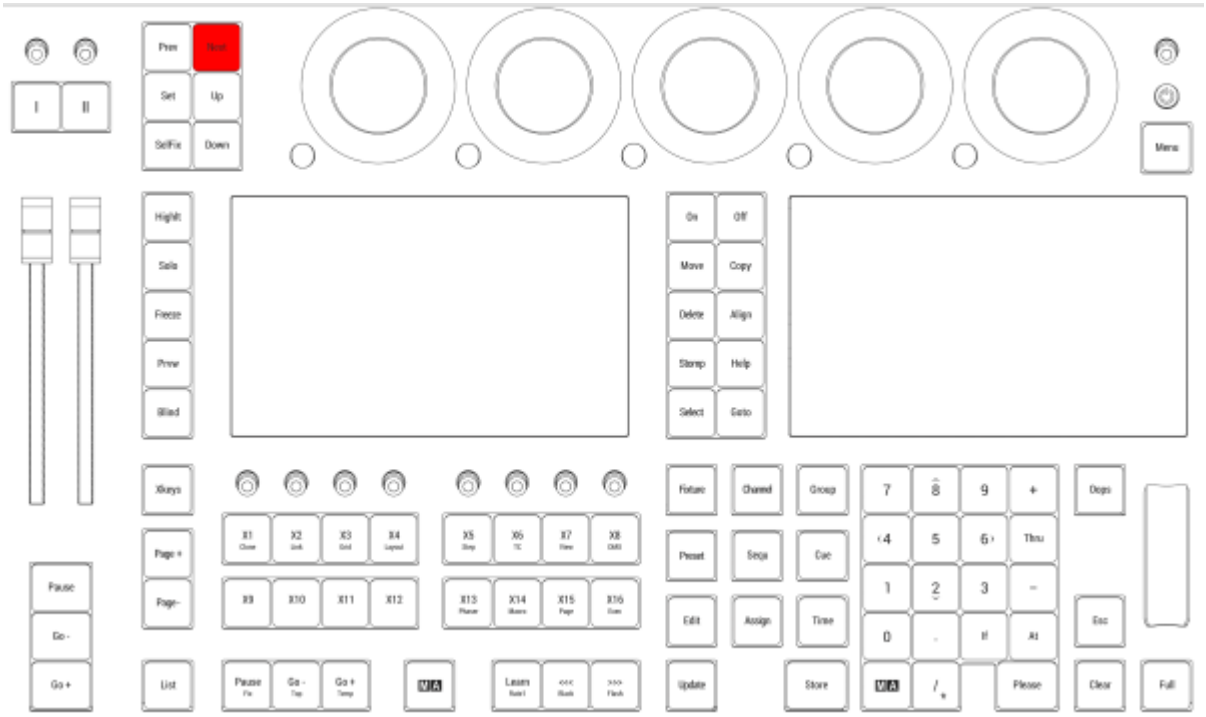
Next step

Pressing MA + Next executes the Next Step command in the command line.

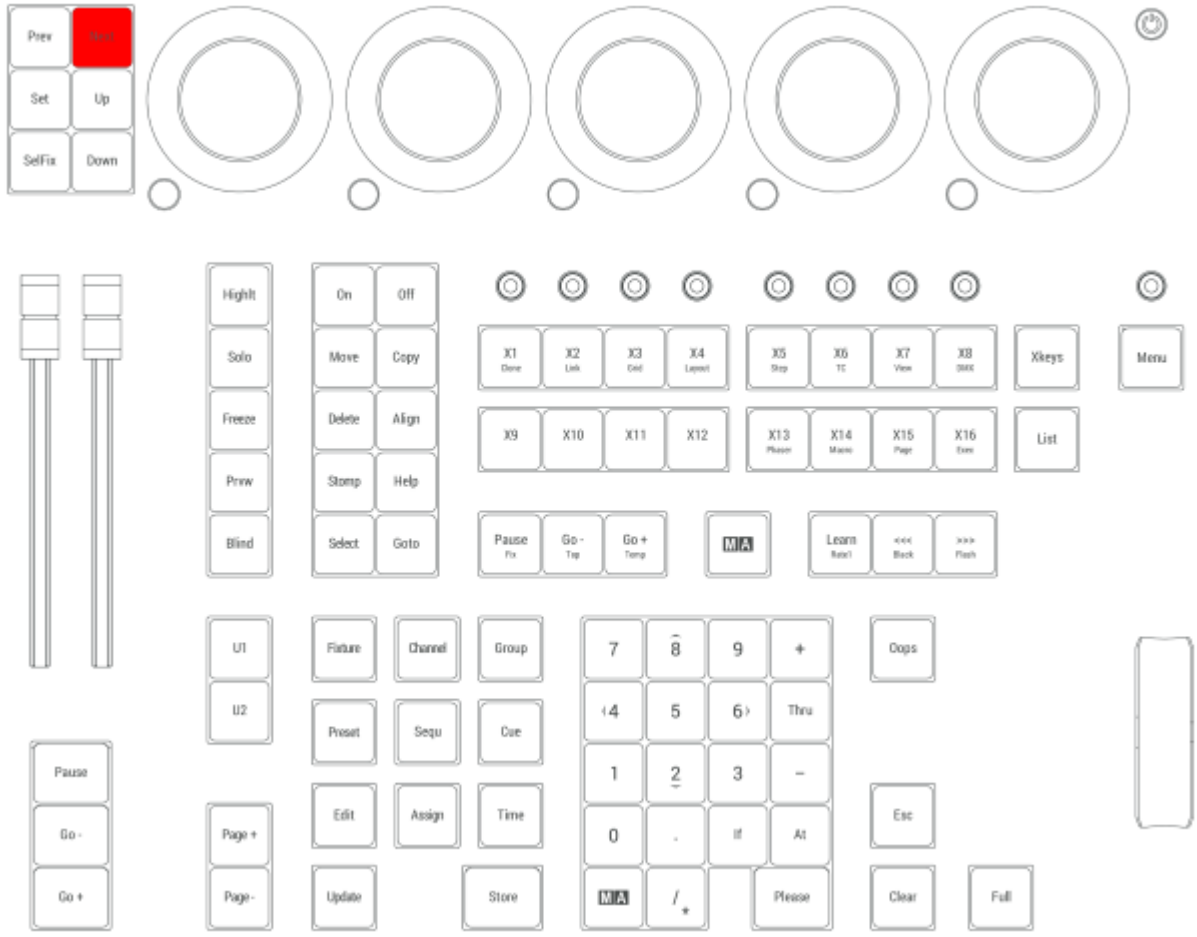
```
User name[Fixture]> Next Step
```

For more information about Next Step, see the **Next keyword** topic and **Step keyword** topic. Or see the **Phaser Topic** for more information about steps.

Next is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.38. Numeric keys | arrows

Pressing a numeric key enters the number into the command line.

Move cursor

Pressing MA + 8 move the cursor 1 row up.

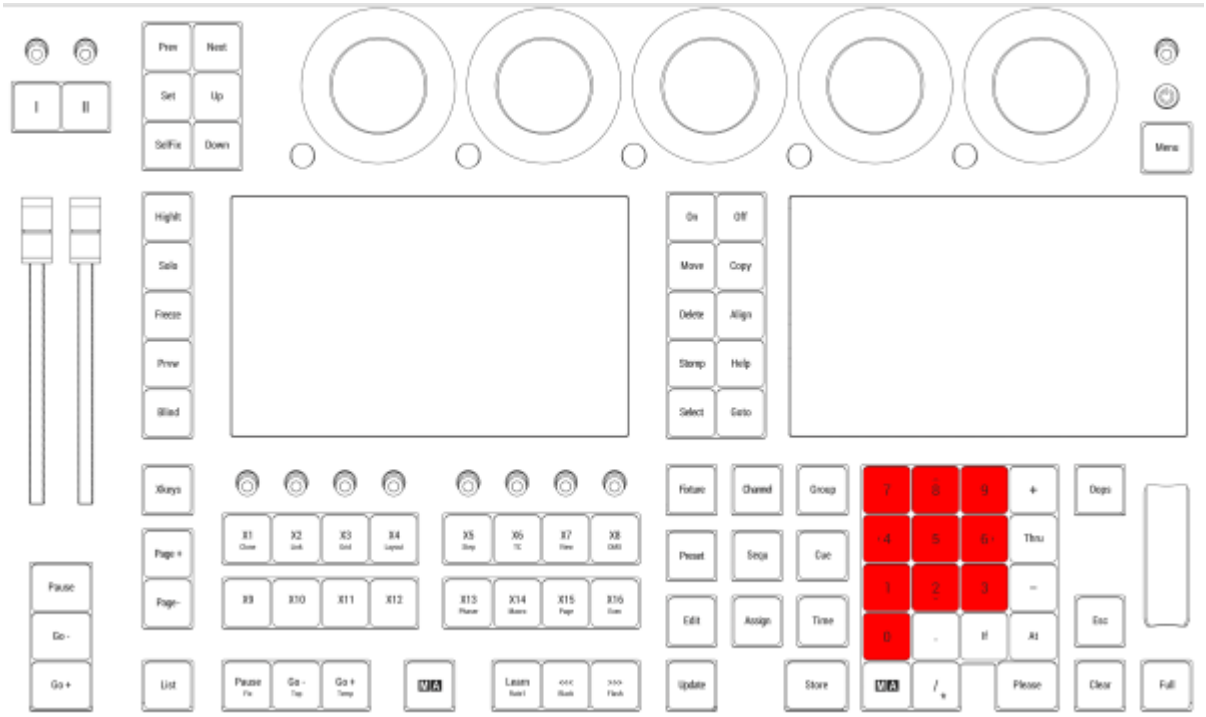
Pressing MA + 2 move the cursor 1 row down.

Pressing MA + 4 move the cursor 1 column left.

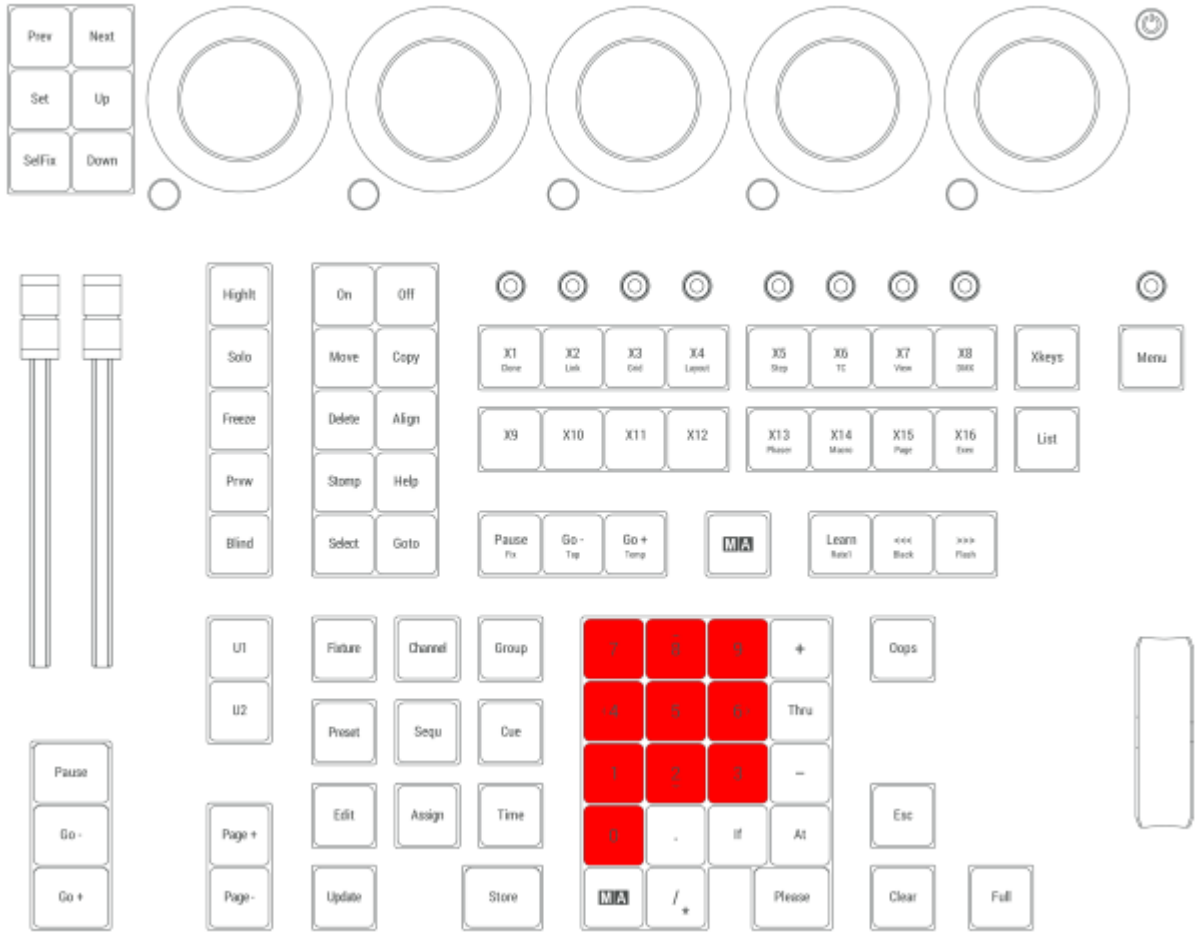
Pressing MA + 6 move the cursor 1 column right.

The arrow keys on a keyboard have the same functions.

The numeric keys is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

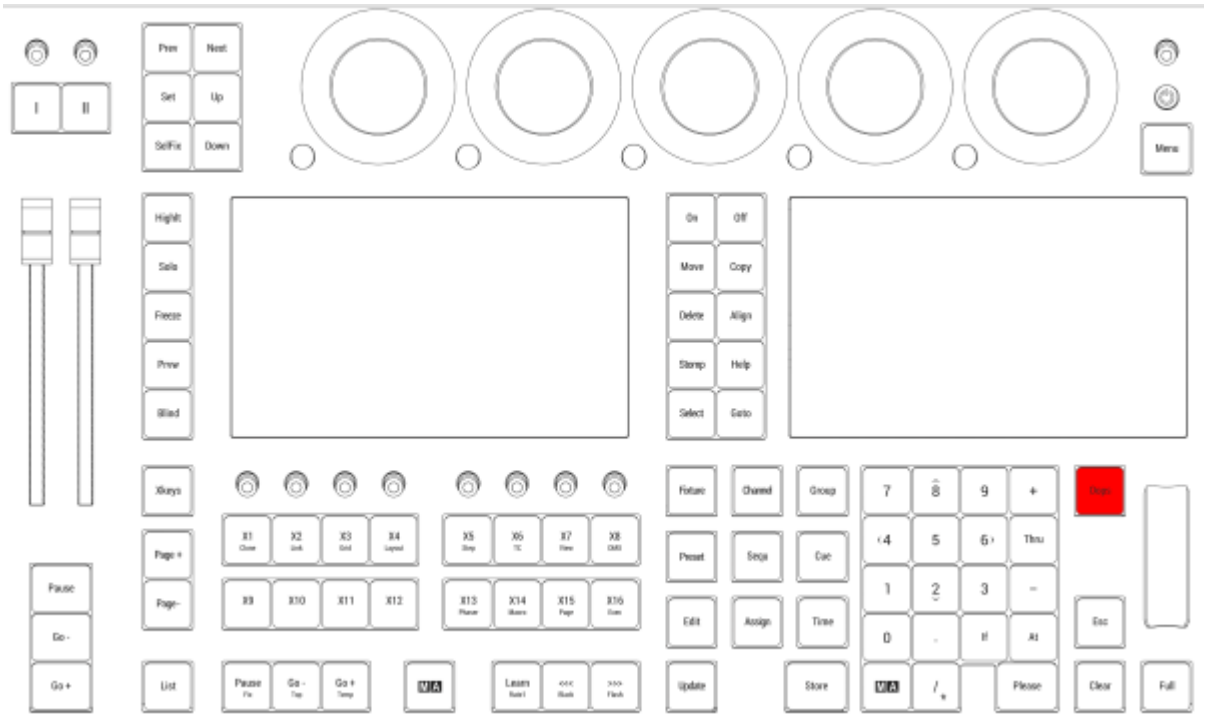
2.15.39. Oops key

Pressing Oops executes the Oops command.

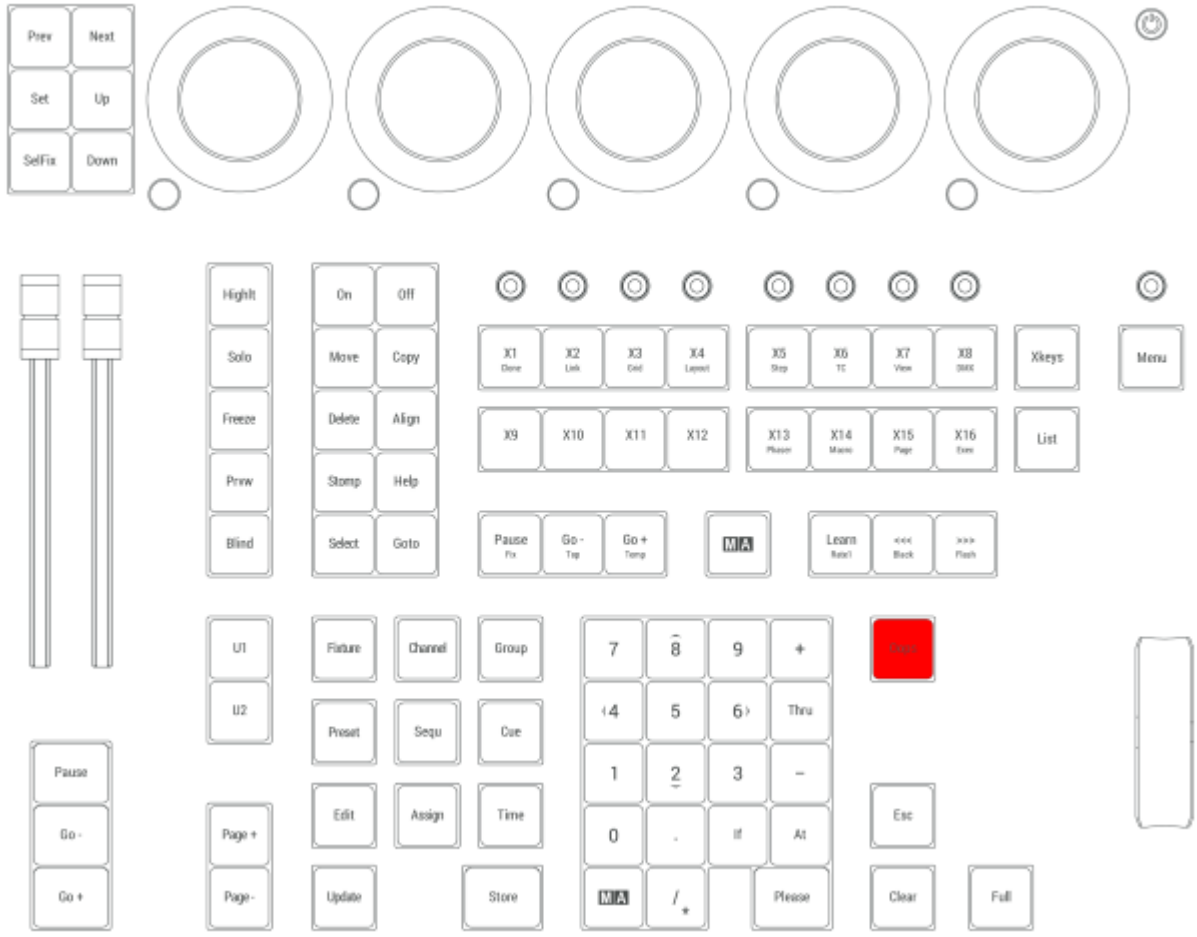
```
OK:Oops
```

For more information about Oops, see the **Oops keyword** topic.

Oops is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.40. On key

Pressing On enters the On keyword into the command line.

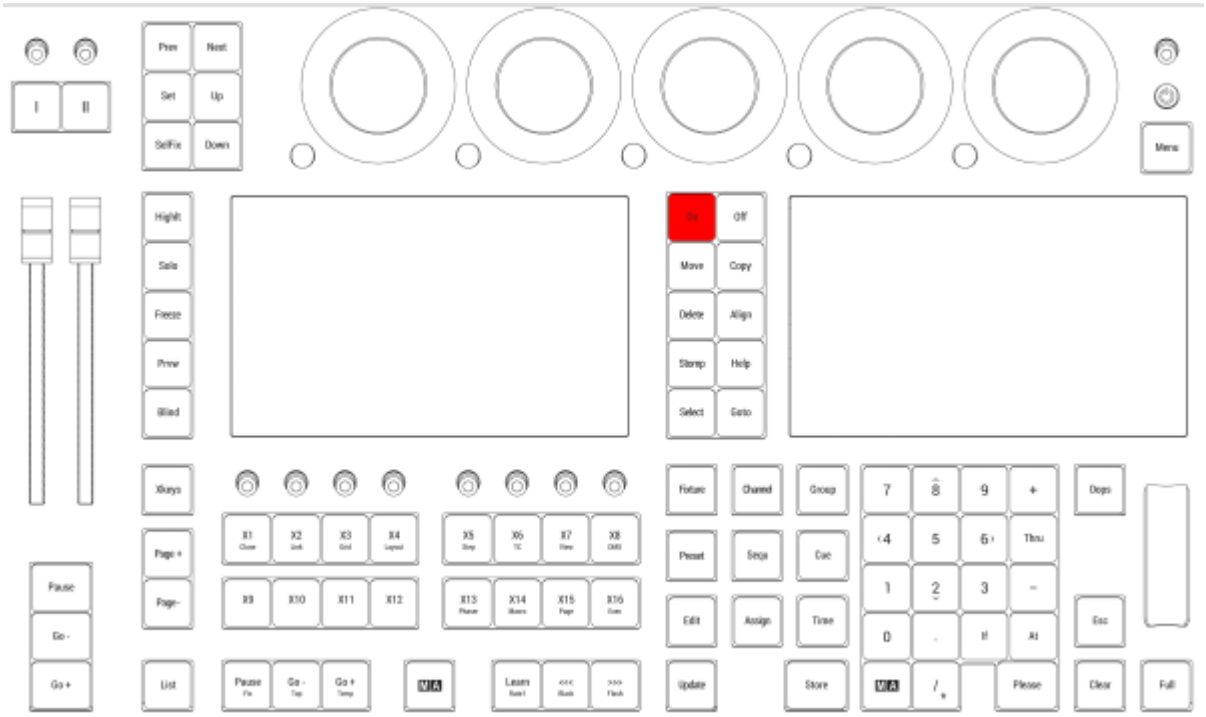
For more information about On, see the **On keyword** topic.

Call

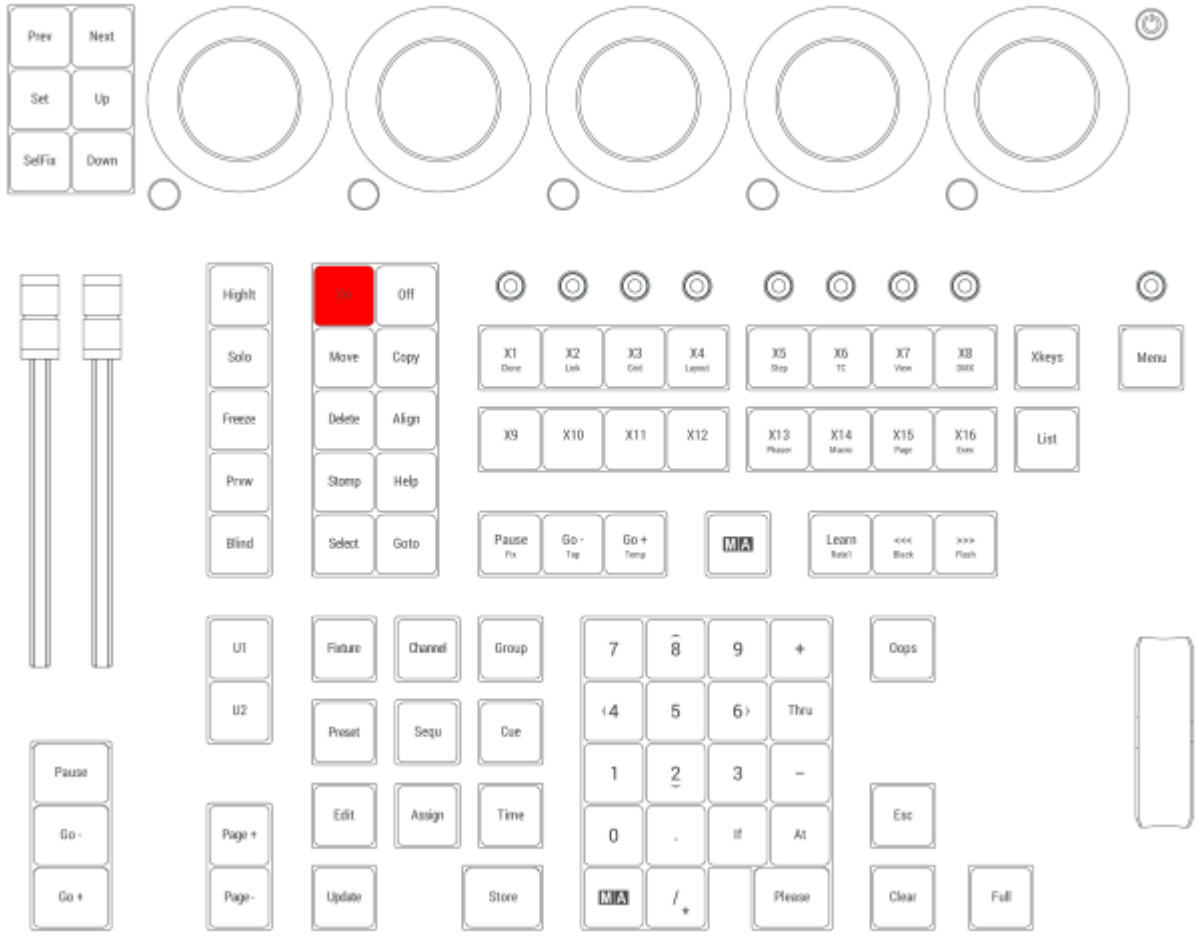
Pressing On On enters the Call keyword into the command line.

For more information about Call, see the **Call keyword** topic.

On is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.41. Off key

Pressing Off enters the Off keyword into the command line.

For more information about Off, see the **Off keyword** topic.

Off menu

Pressing Off Off opens the Off menu.

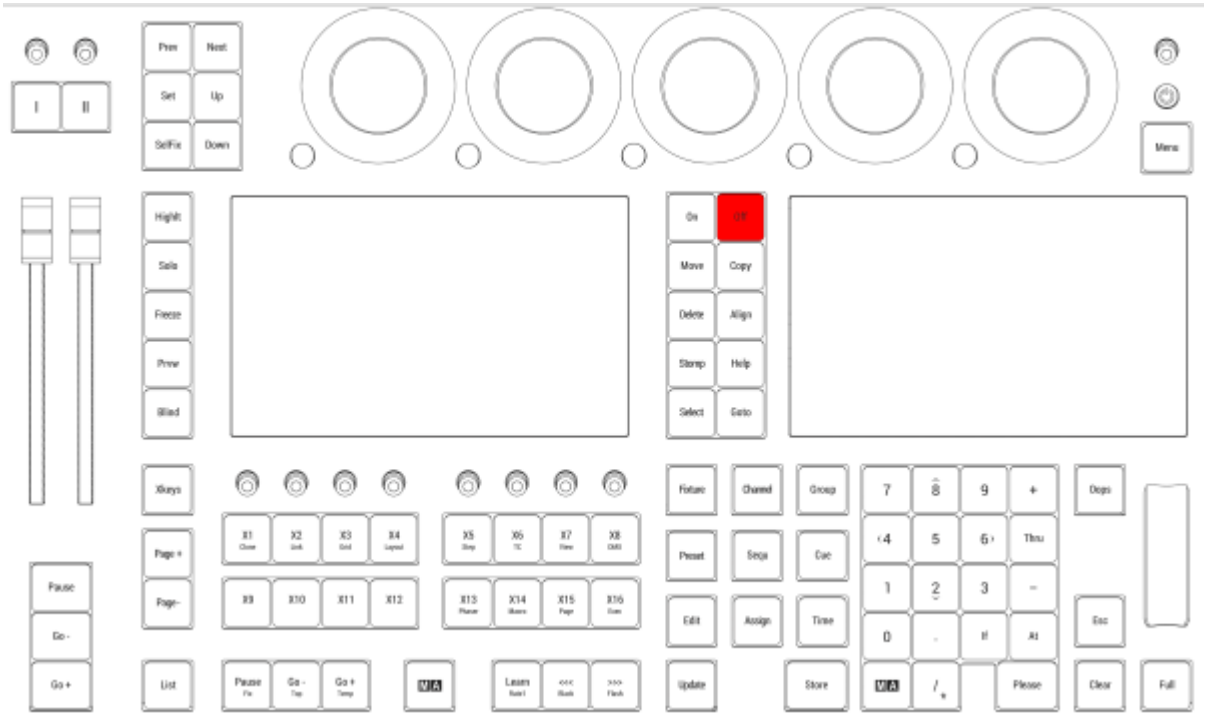
Read more about the Off menu **here**.

Close RemoteHID connection

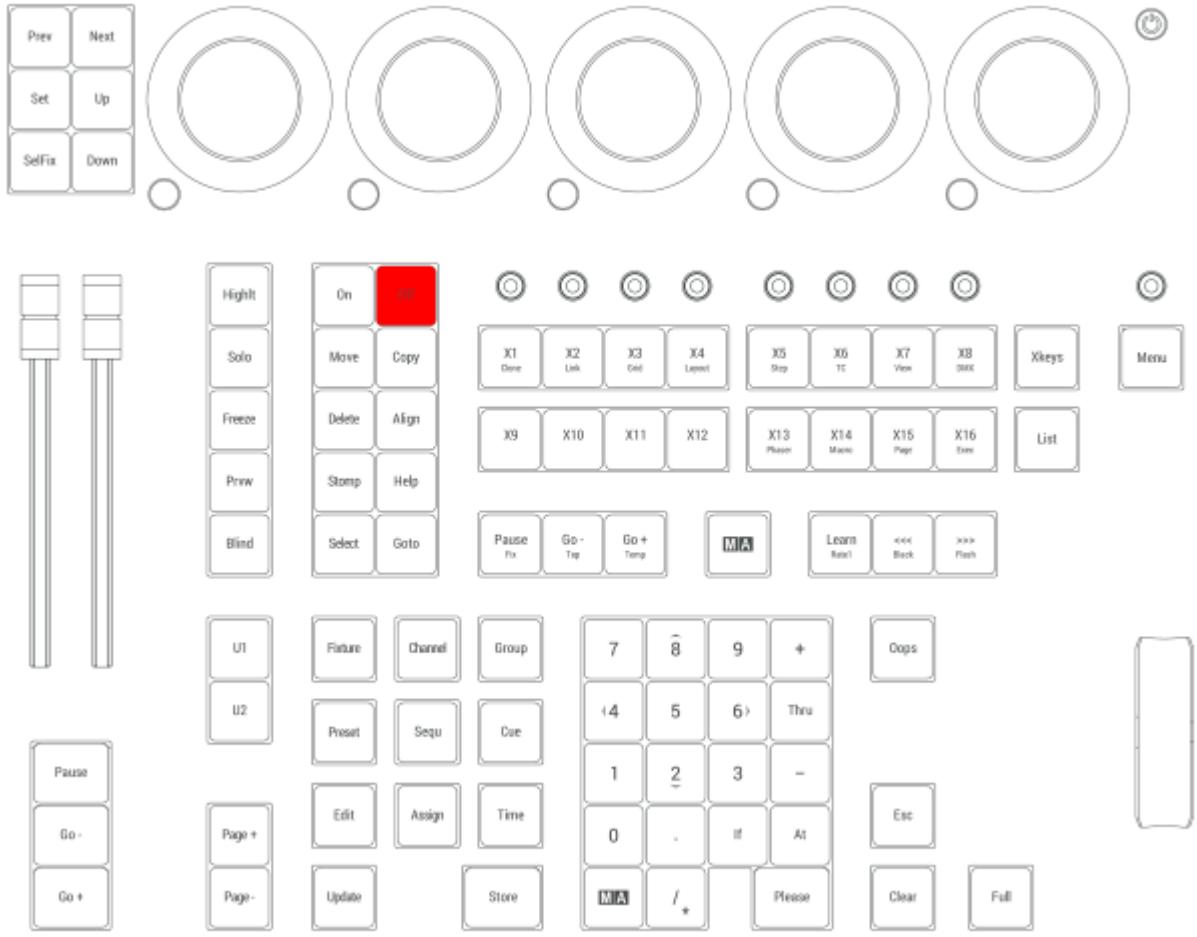
Pressing MA + MA + Off closes any active RemoteHID connections.

For more information about RemoteHID connections, see the **RemoteHID** topic.

Off is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

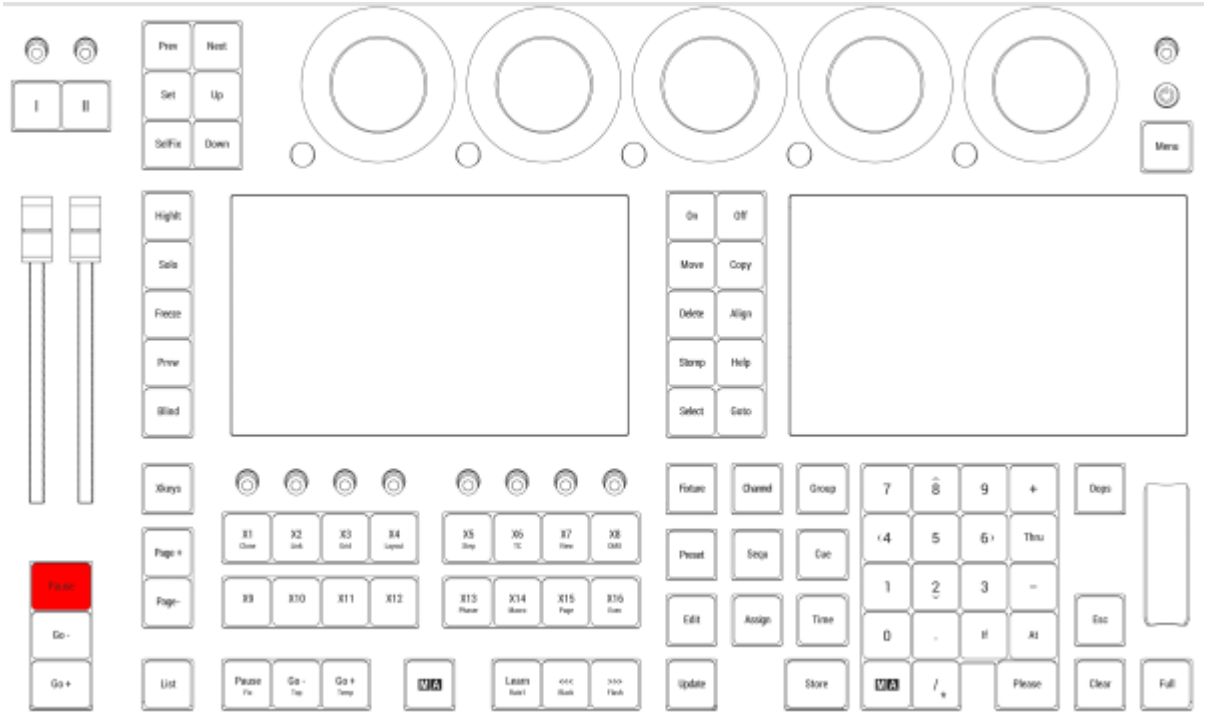
2.15.42. Pause [large] key

Pressing Pause [large] executes the Pause Executor keyword in the command line

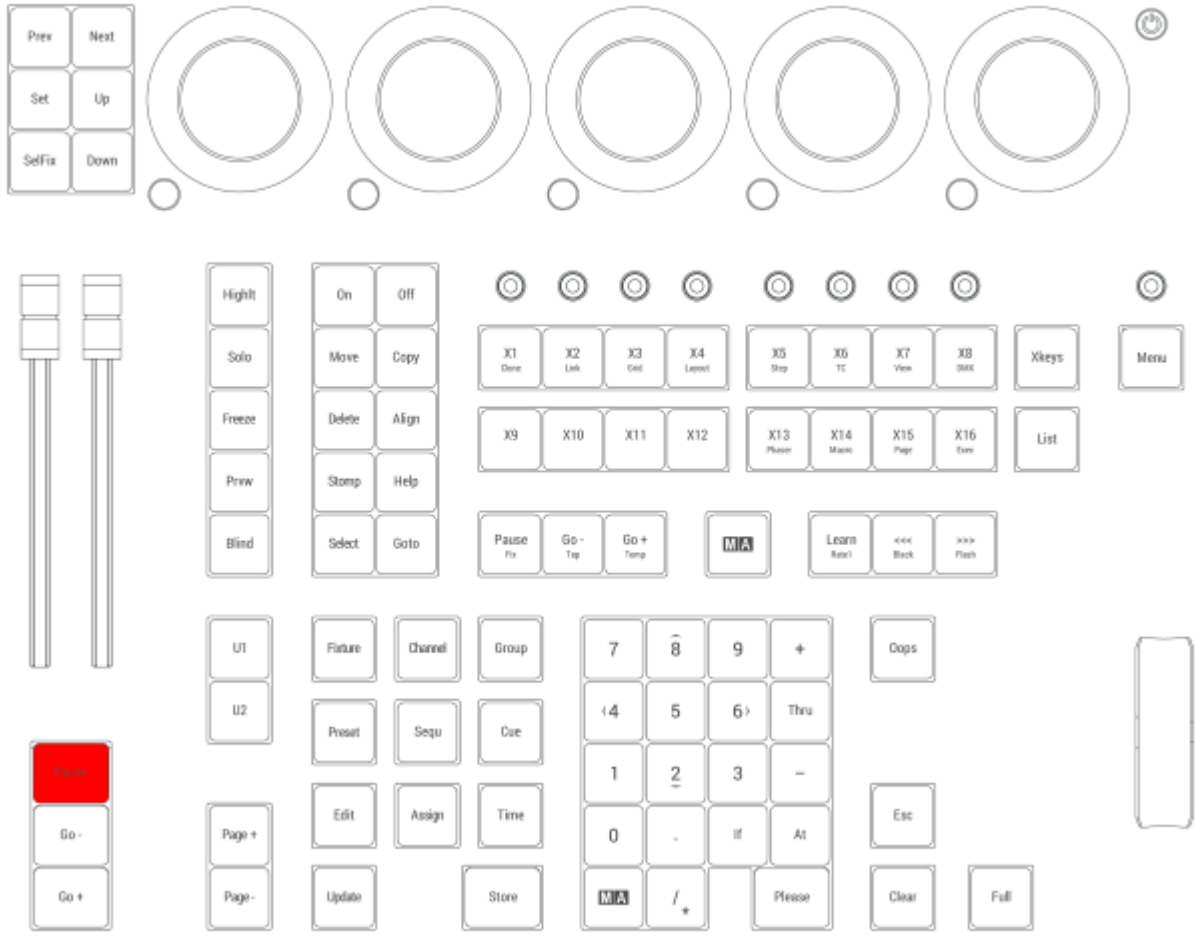
```
User name[Fixture]> Pause Executor
```

For more information about the Pause, see the **Pause keyword** topic.

Pause [large] is located in the master section under the two master faders.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.43. Pause | Fix key

Pressing Pause enters the Pause keyword into the command line.

For more information about Pause, see the **Pause keyword** topic.

Pause

Pressing Pause Pause enters the Park keyword into the command line.

For more information about Park, see the **Park keyword** topic.

Fix

Pressing and holding MA + Pause enters the Fix keyword into the command line.

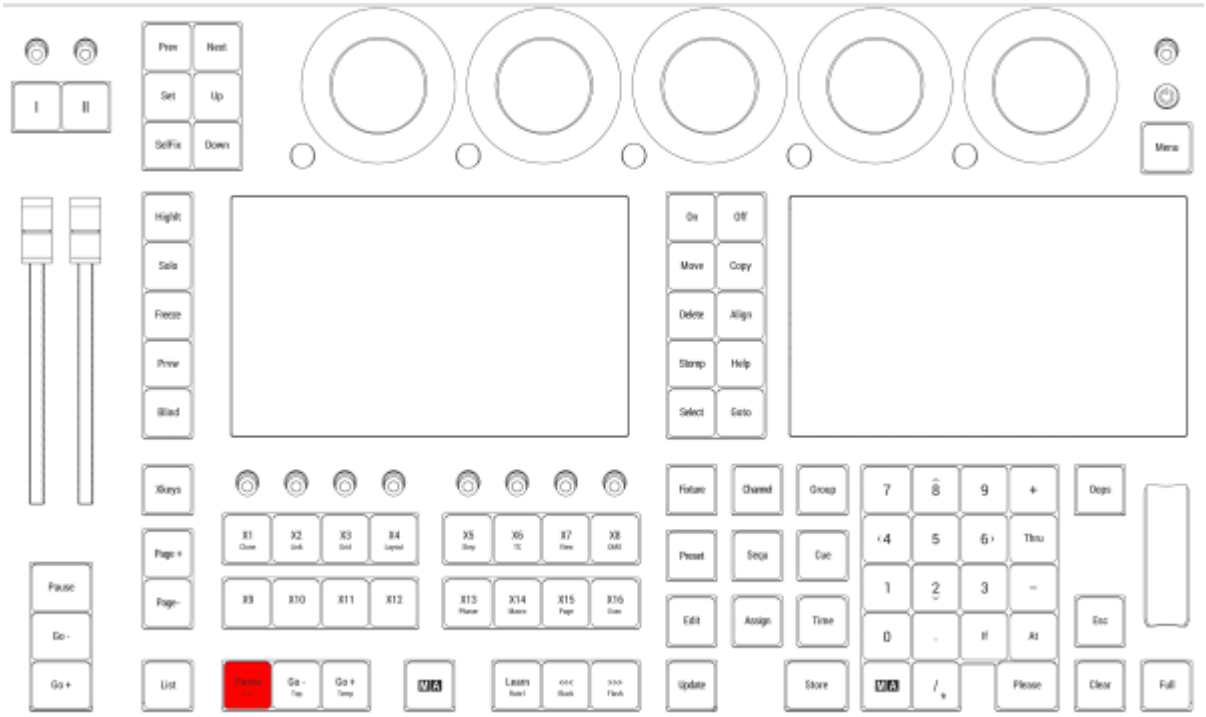
For more information about Fix, see the **Fix keyword** topic.

Desk Lock

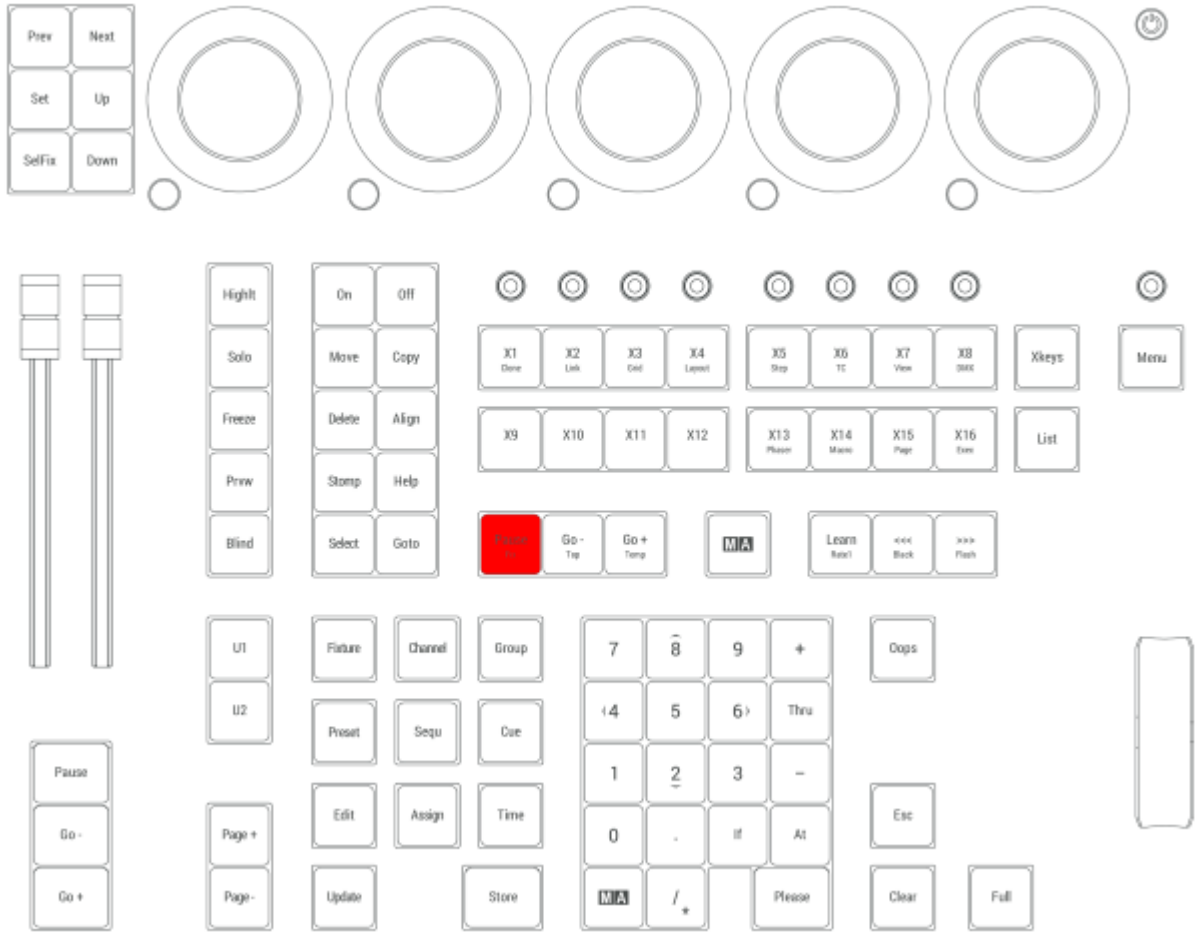
Pressing and holding MA + MA + Pause toggles the desk lock.

For more information about the desk lock, see the **Desk Lock** topic.

Pause is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



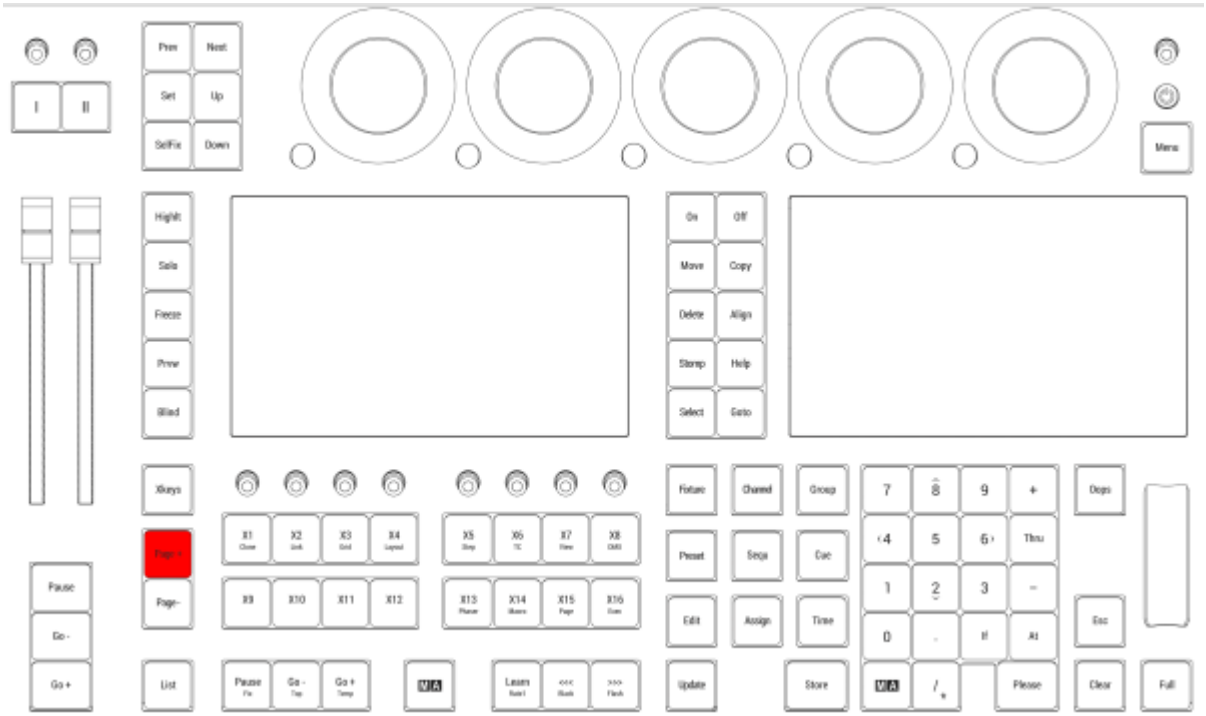
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.44. Page+ key

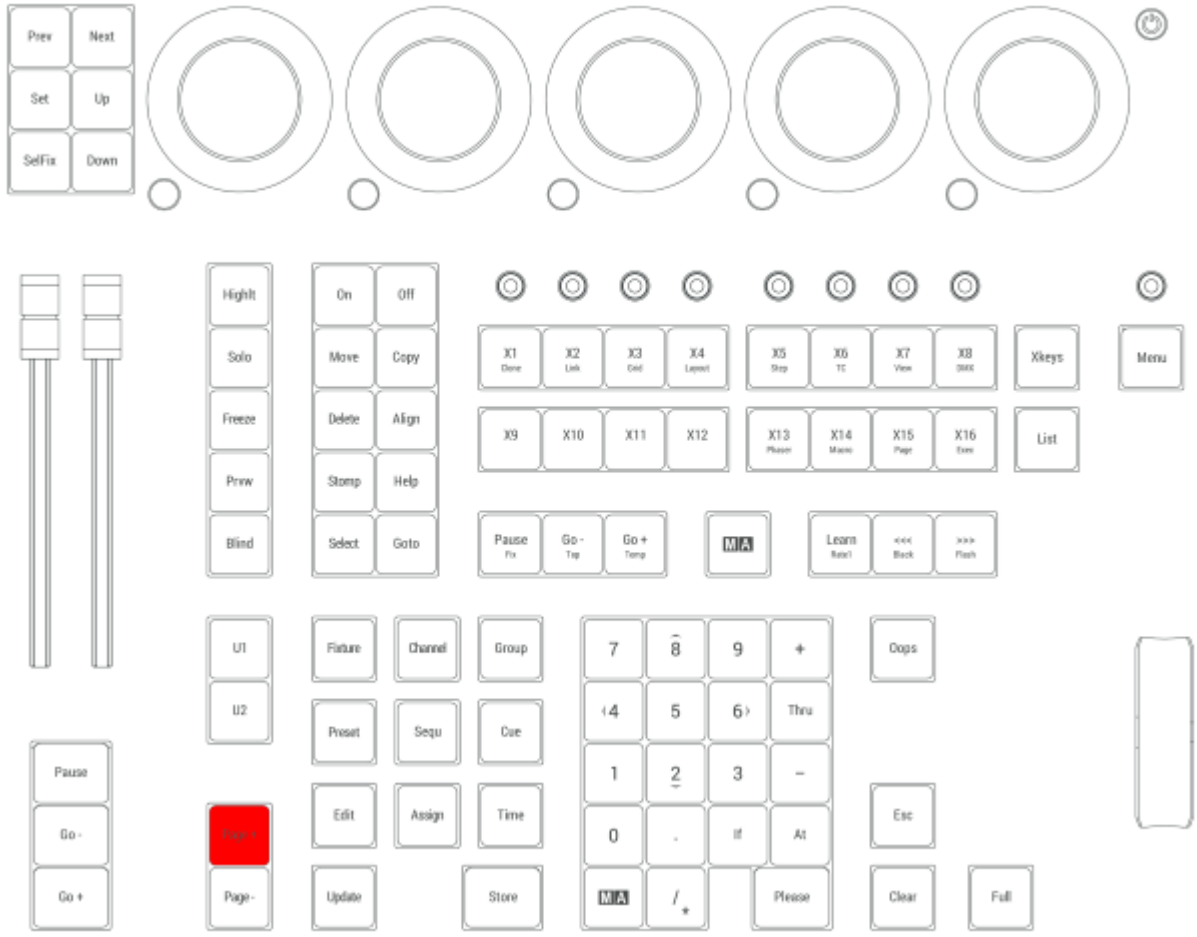
Pressing Page+ executes the command Next Page in the command line.

```
OK:Next Page
```

Page+ is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

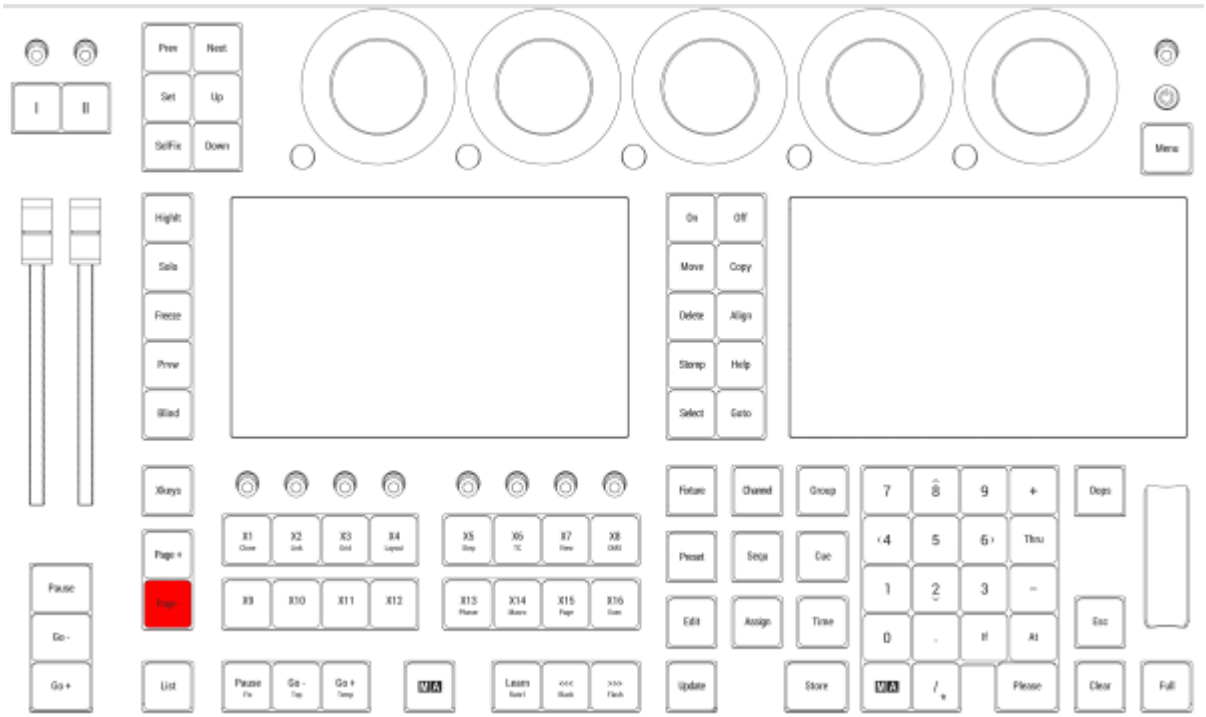
2.15.45. Page- key

Pressing Page- executes the command Previous Page in the command line.

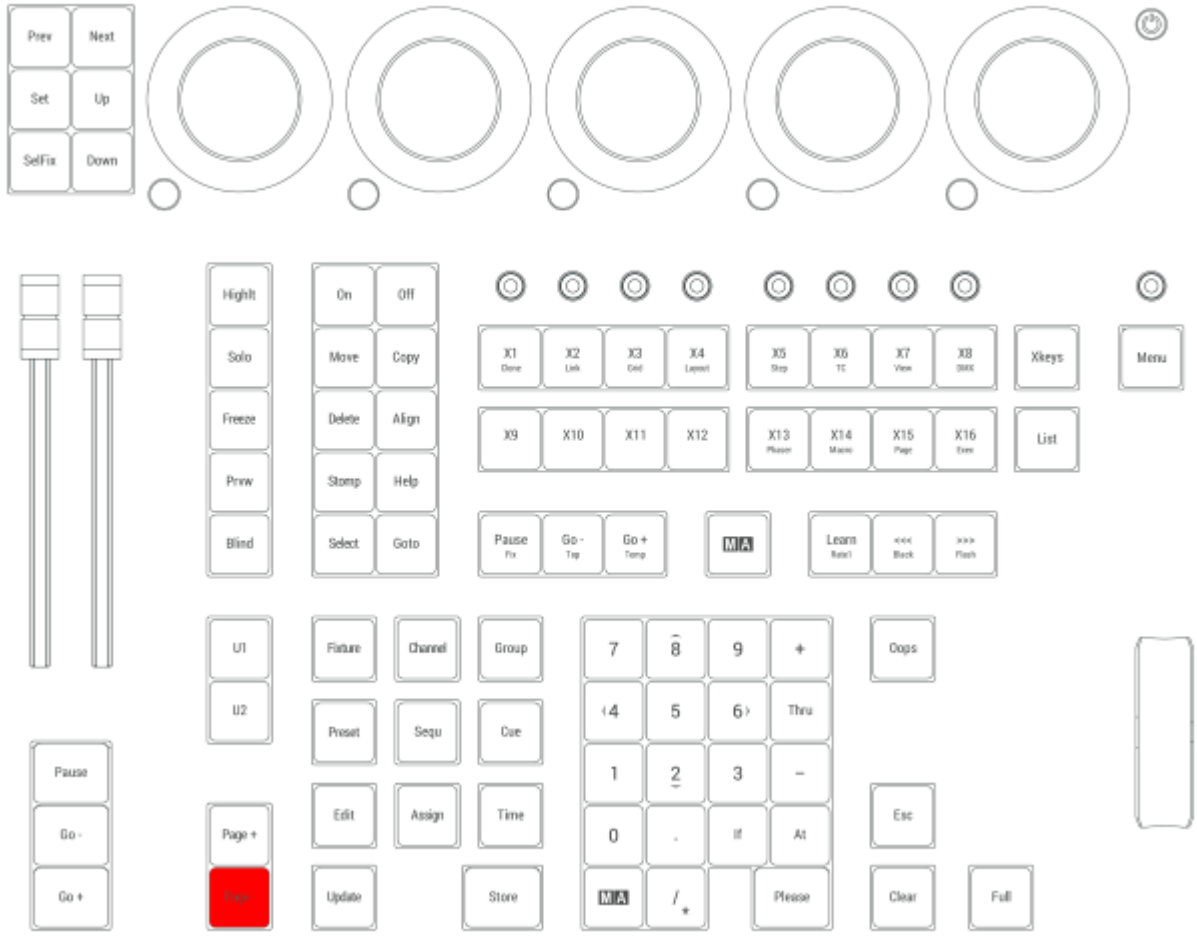
```
OK:Previous Page
```

Pressing and holding Page- for more than 1 second sets the page to page 1.

Page- is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

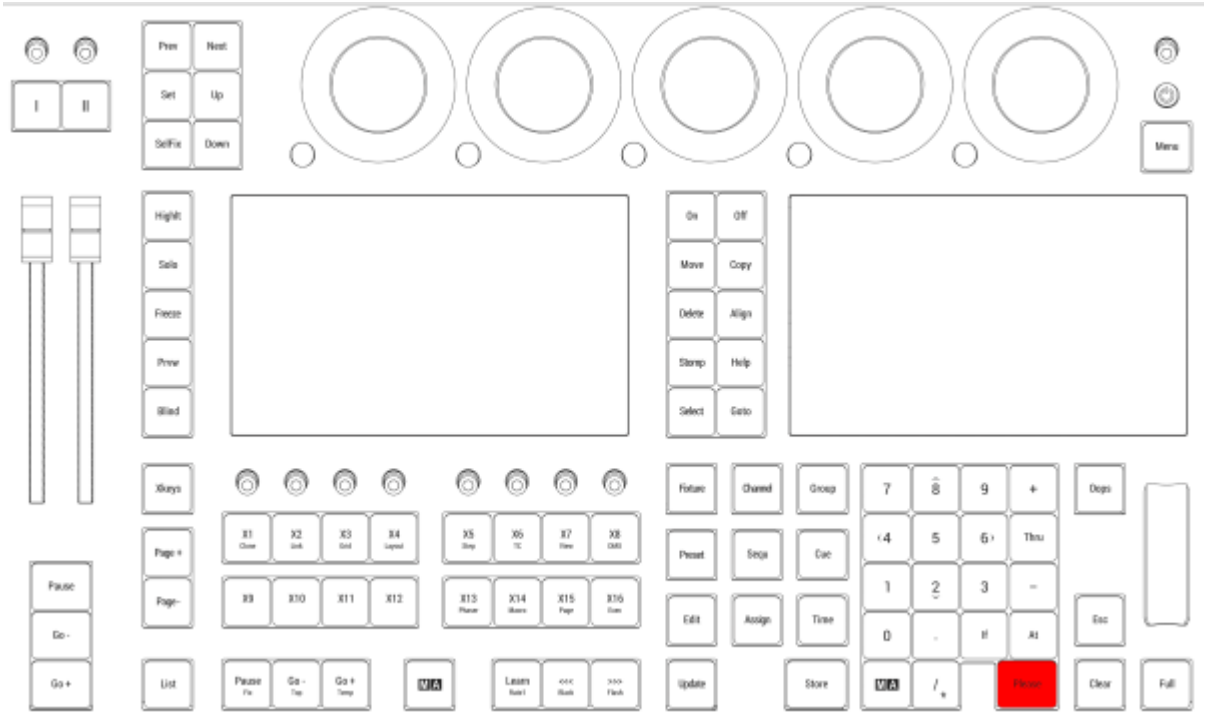
2.15.46. Please key

Pressing Please executes the command in the command line. This is the same function as pressing Enter on an external keyboard.

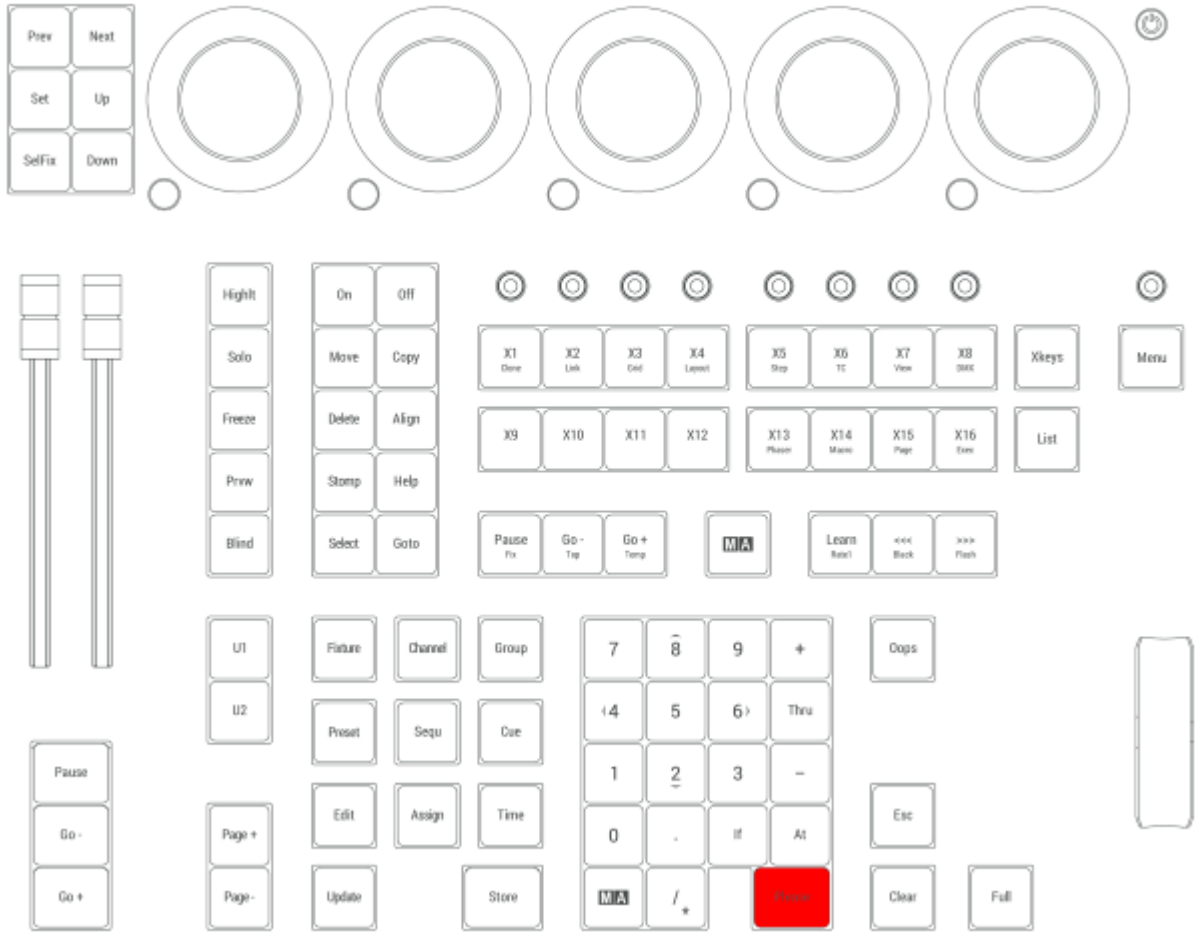
Activate all Attributes

- Pressing Please Please activates all attributes of the selected fixtures.
- Pressing Please once more deactivates all attributes of the selected fixtures.

Please is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



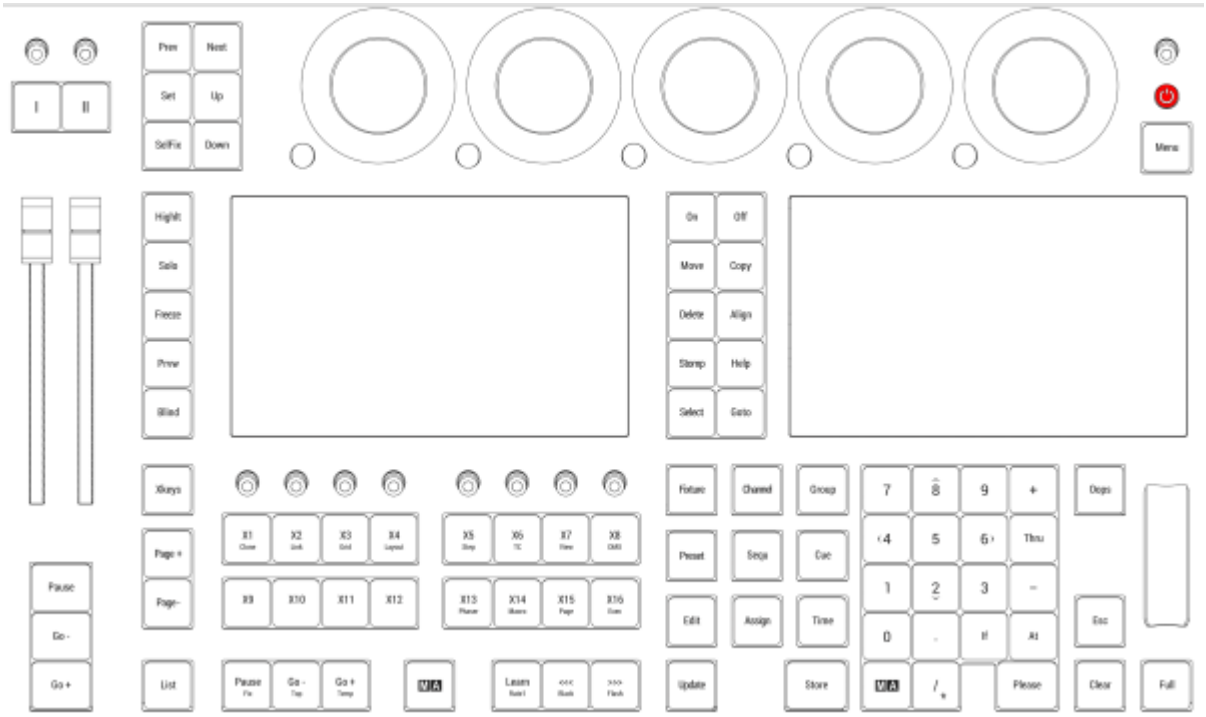
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.47. Power key

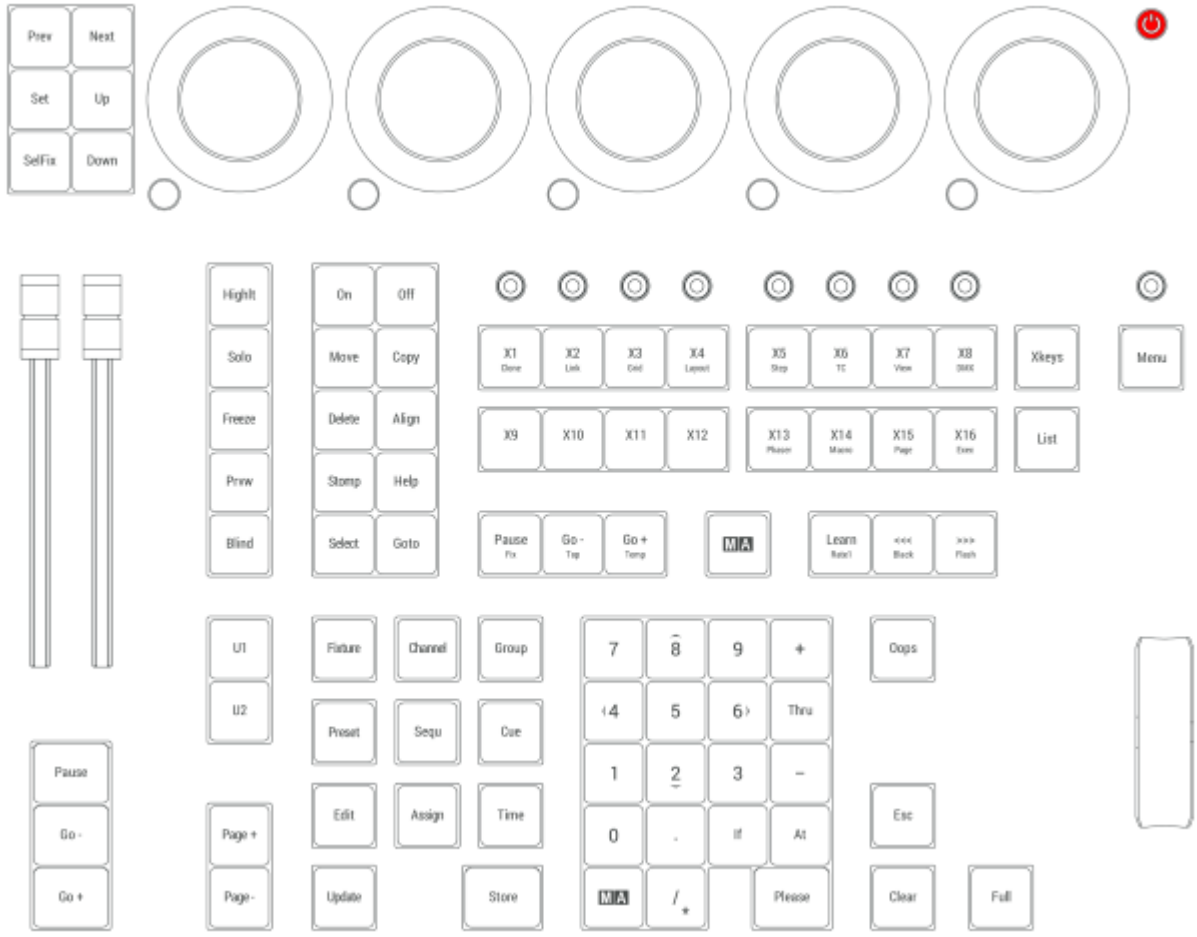
Pressing Power boots up the console or shuts it down.

Important:
To switch the console on or off, switch the power switch on the rear panel.

Power is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.48. Preset key

Pressing Preset enters the Preset keyword into the command line.

For more information about Preset, see the **Preset keyword** topic.

Attribute

Pressing Preset Preset enters the Attribute keyword into the command line.

For more information about Attribute, see the **Attribute keyword** topic.

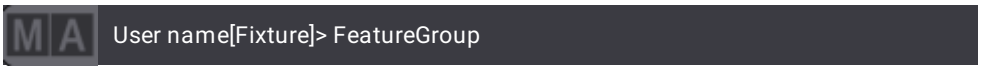
Gel

Pressing Preset Preset Preset enters the Gel keyword into the command line.

For more information about Gel, see the **Gel keyword** topic.

FeatureGroup

Pressing MA + Preset enters the FeatureGroup keyword into the command line.



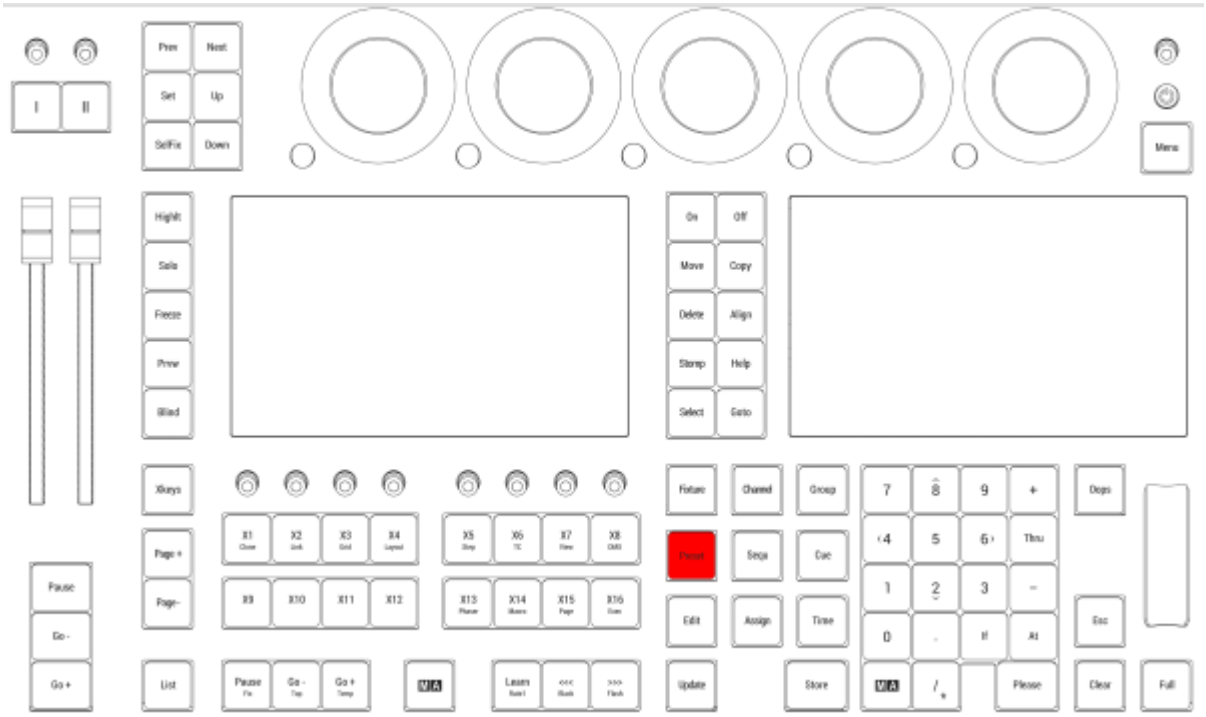
For more information about FeatureGroup, see the **FeatureGroup keyword** topic.

DataPool

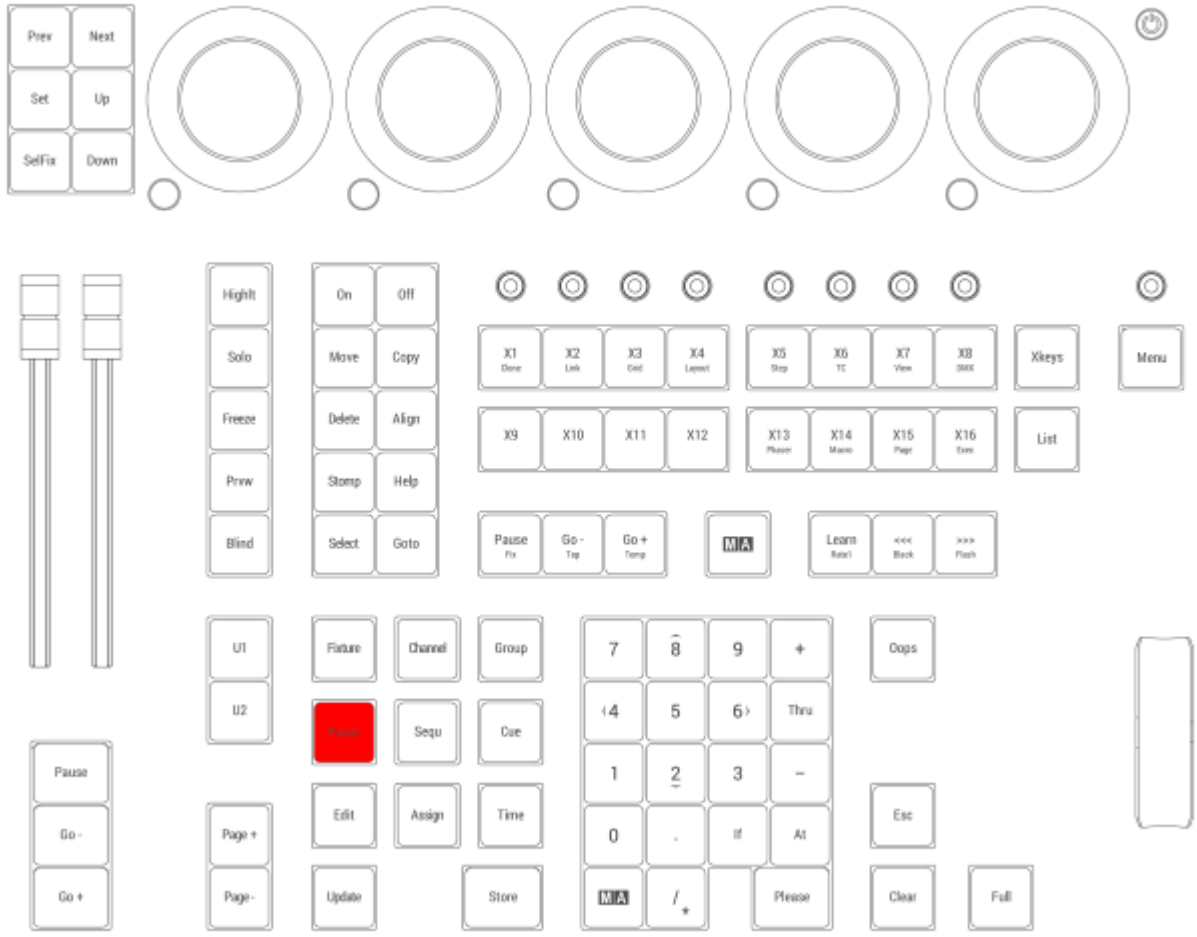
Pressing MA + Preset + Preset enters the DataPool keyword into the command line.

For more information about DataPool, see the **DataPool keyword** topic.

Preset is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



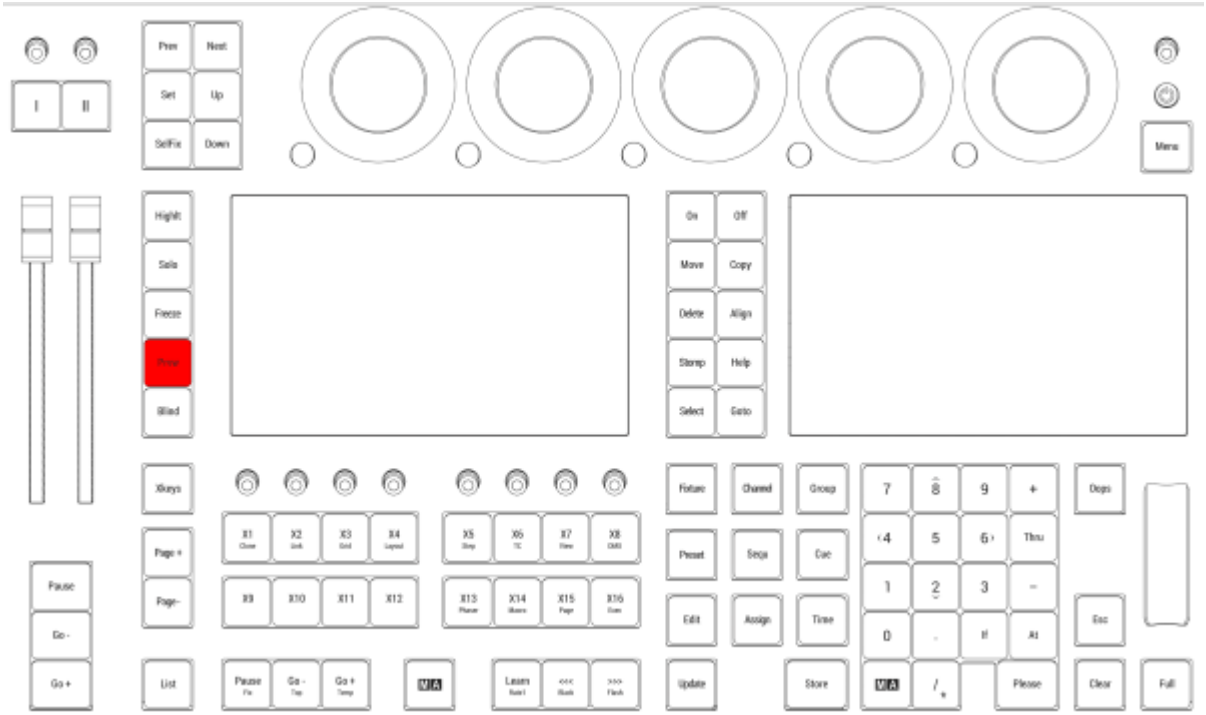
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.49. Prvw [Preview] key

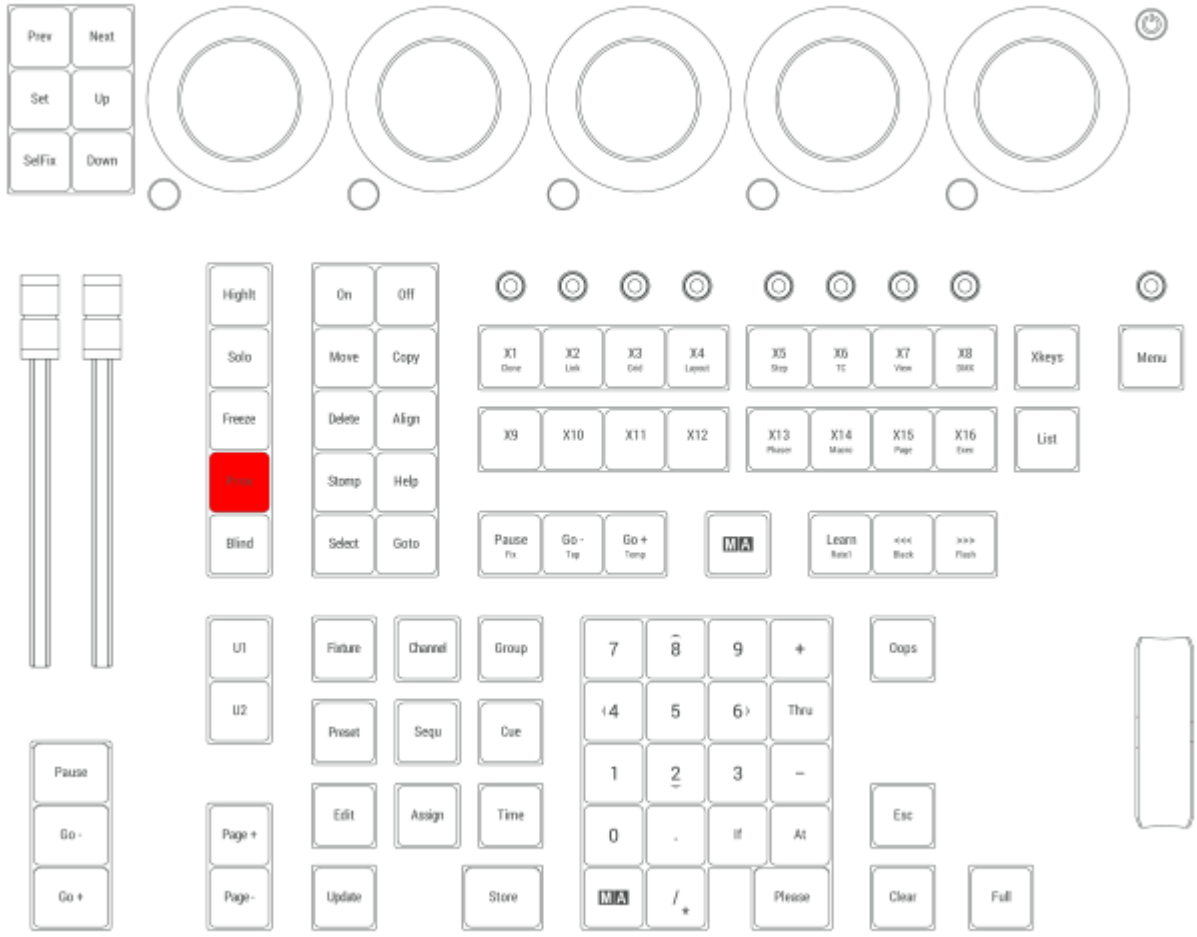
Pressing Prvw executes the the Preview keyword in the command line.

For more information about Prvw, see the **Preview keyword** topic.

Prvw is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

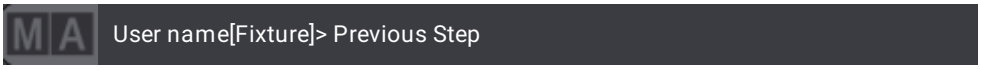
2.15.50. Prev [Previous] key

Pressing Prev executes the Previous keyword in the command line.

For more information about Previous, see the **Previous keyword** topic.

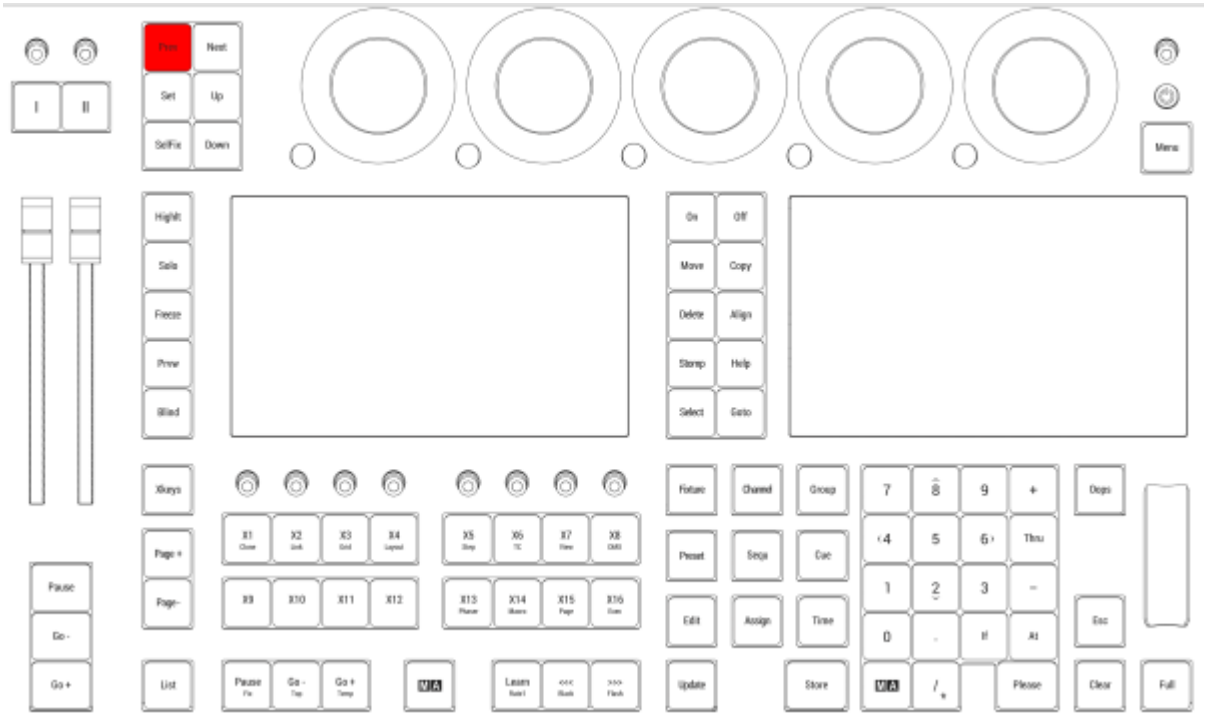
Previous step

Pressing MA + Prev executes the Previous Step command in the command line.

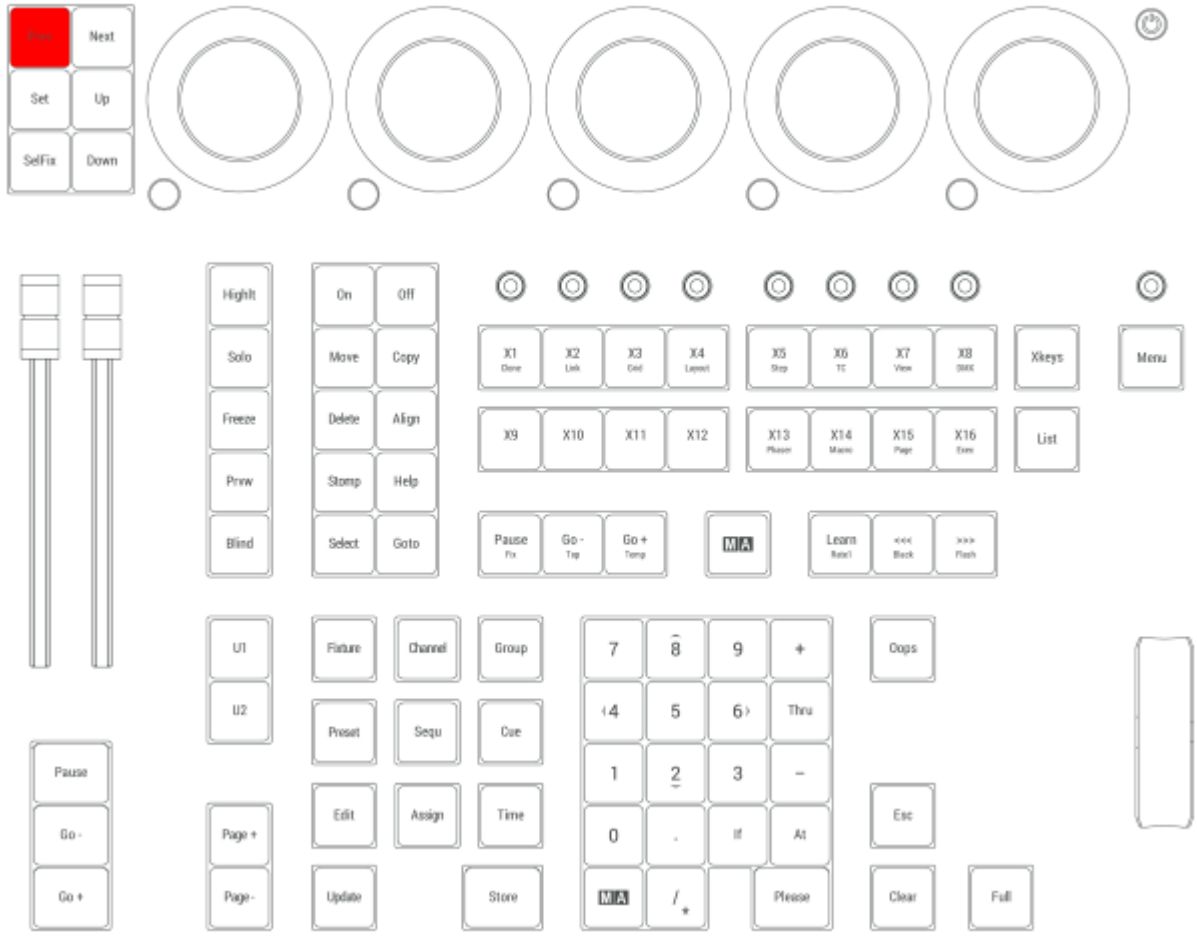


For more information about Previous Step, see the **Previous keyword** topic and **Step keyword** topic. Or see the **Phaser Topic** for more information about steps.

Prev is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



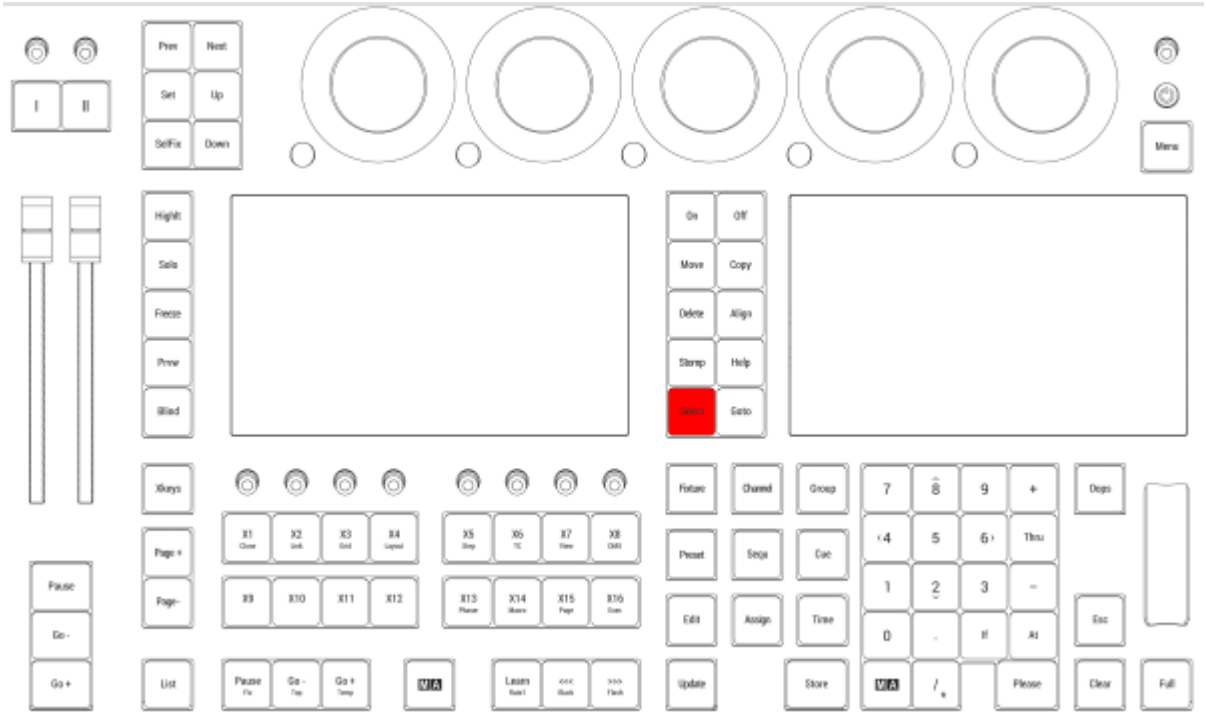
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.51. Select key

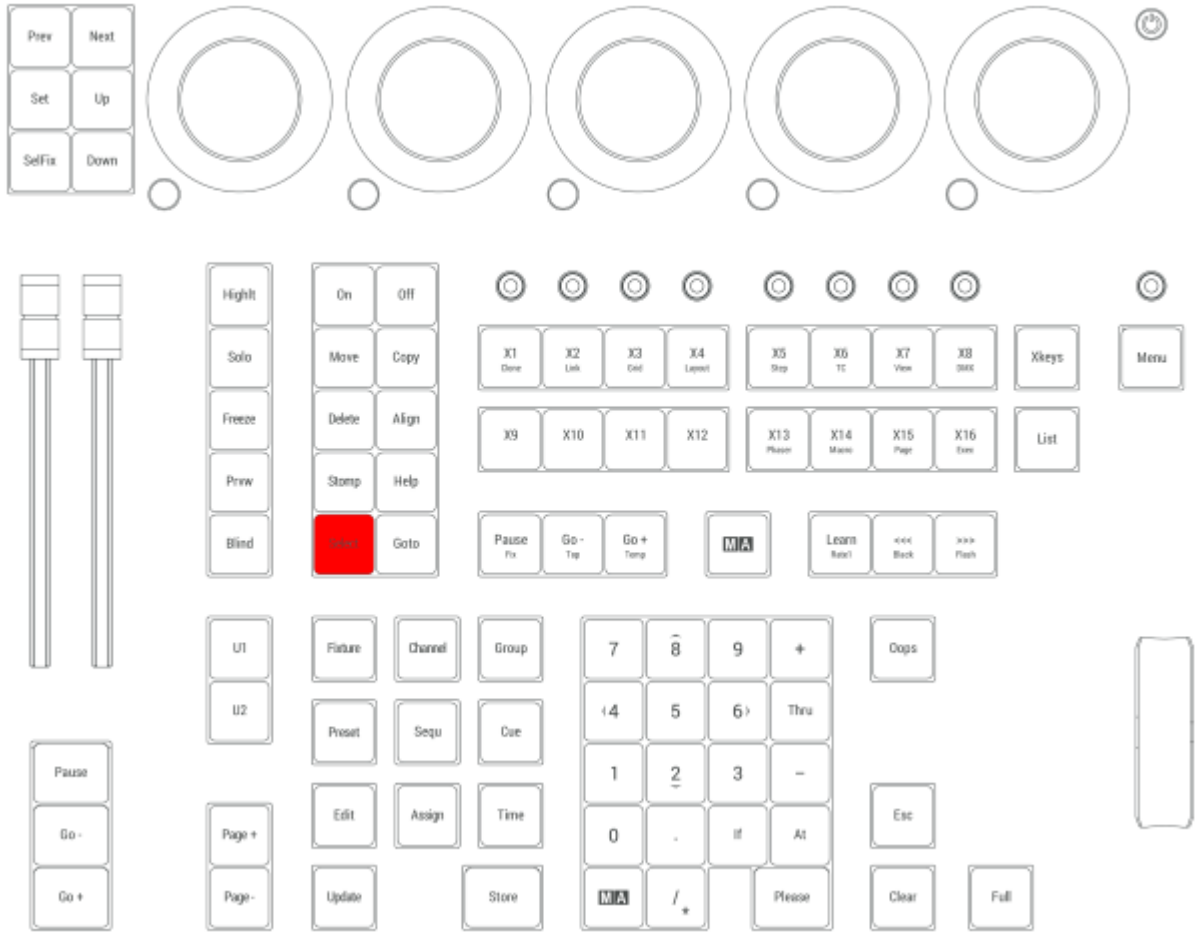
Pressing Select enters the Select keyword into the command line.

For more information about Select, see the **Select keyword** topic.

Select is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



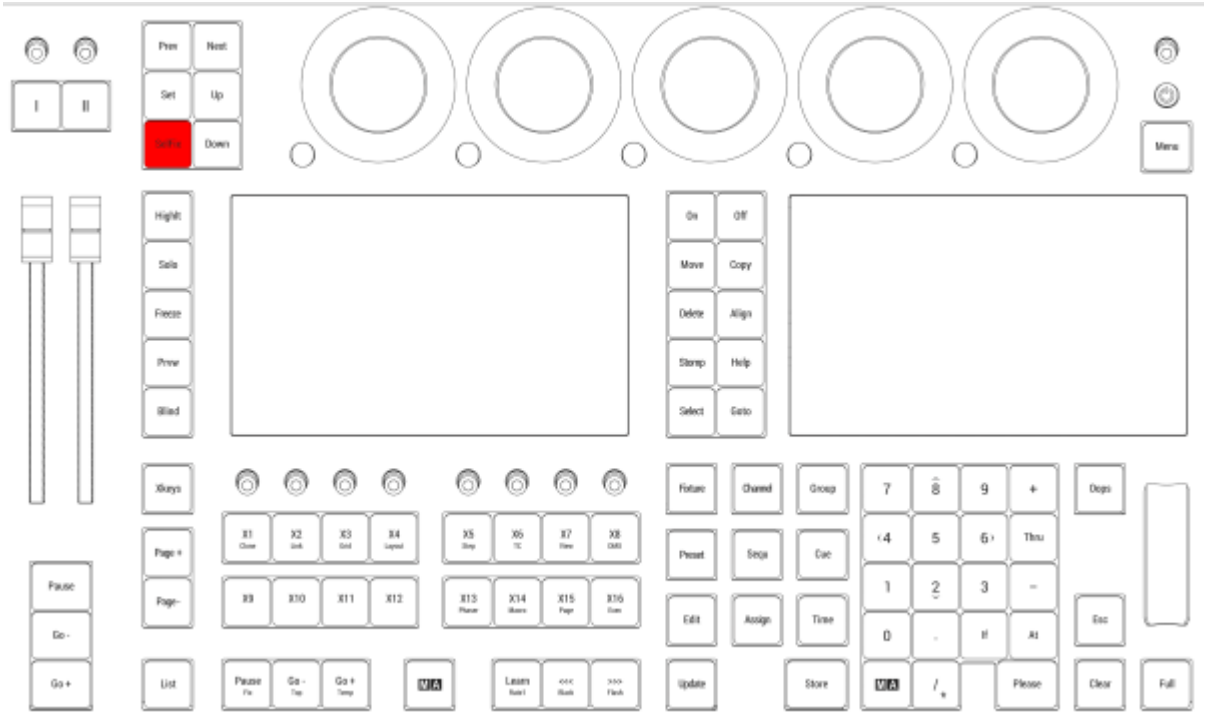
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.52. SelFix [SelectFixture] key

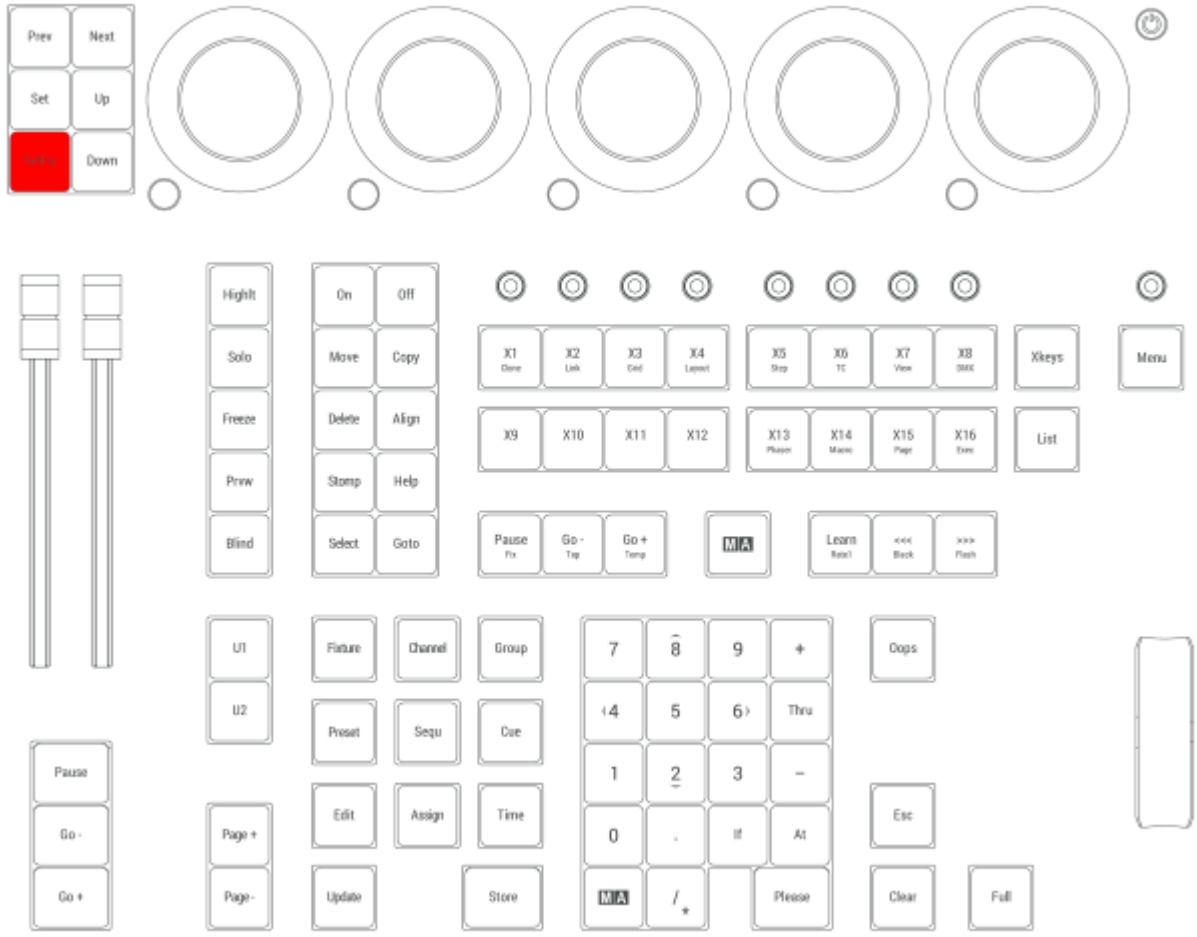
Pressing SelFix enters the SelFix keyword into the command line.

For more information about SelFix, see the **SelectFixture keyword** topic.

SelFix is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



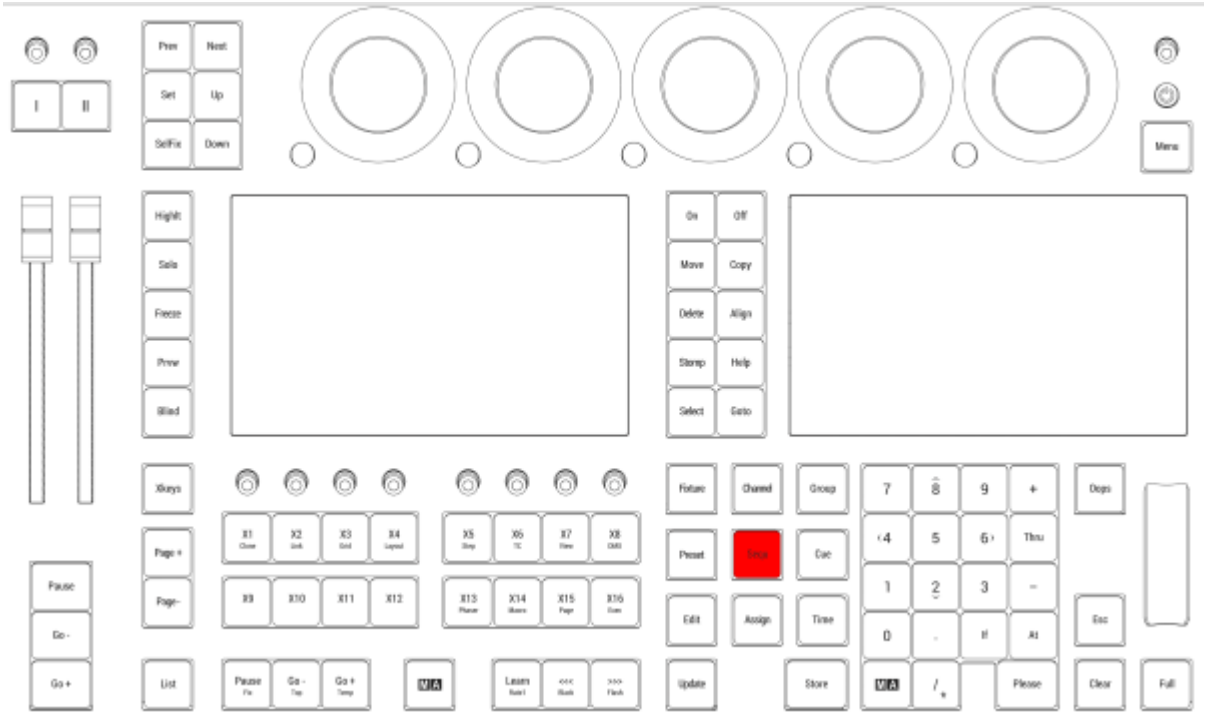
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.53. Sequ [Sequence] key

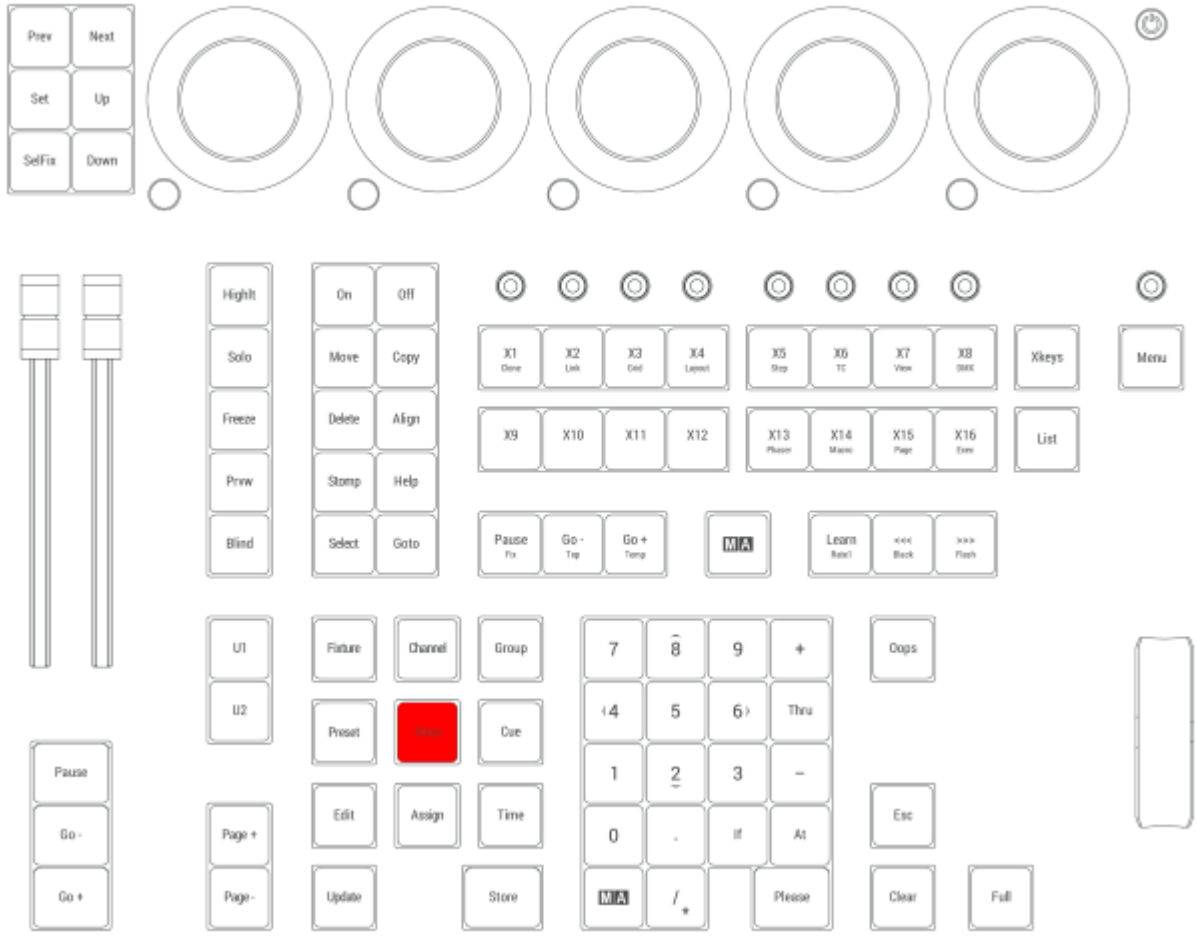
Pressing Sequ enters the Sequence keyword into the command line.

For more information about Sequence, see the **Sequence keyword** topic.

Sequ is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.54. Set key

Pressing Set executes the Toggle MAtricks command in the command line.

```
OK:Toggle MAtricks
```

For more information about MAtricks, see the **MAtricks keyword** topic.

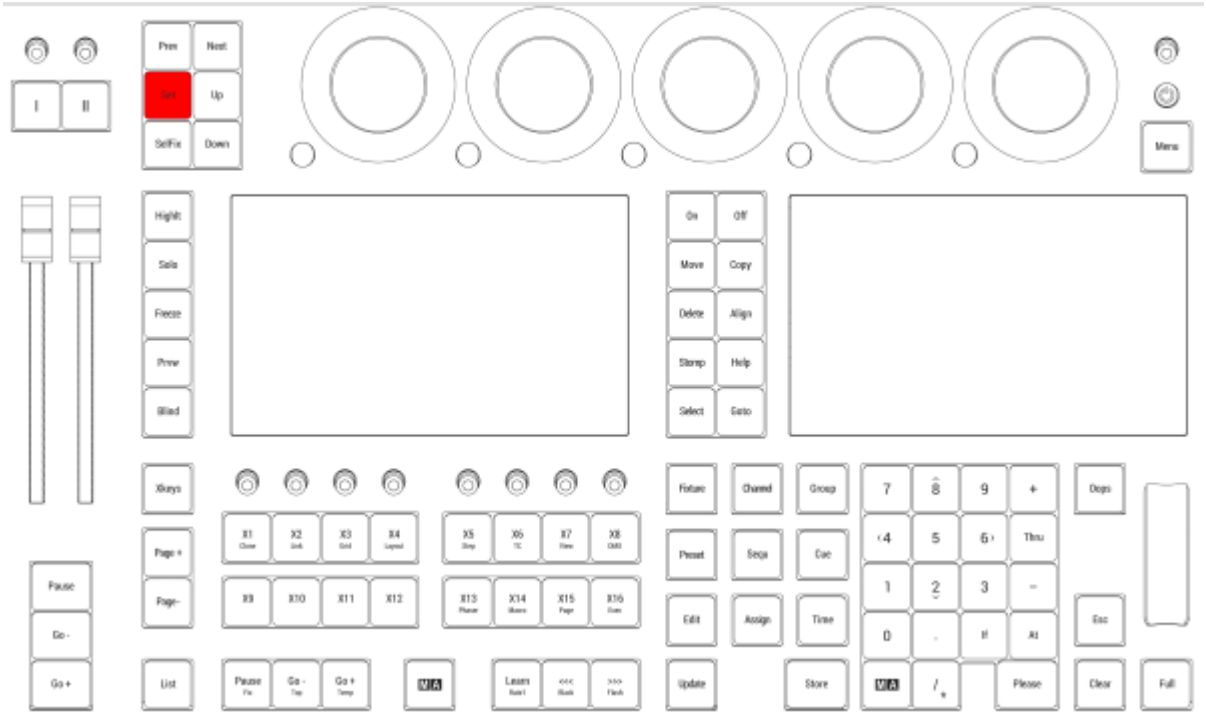
Step Toggle Executor

Pressing MA + Set executes the Step Toggle Executor command in the command line.

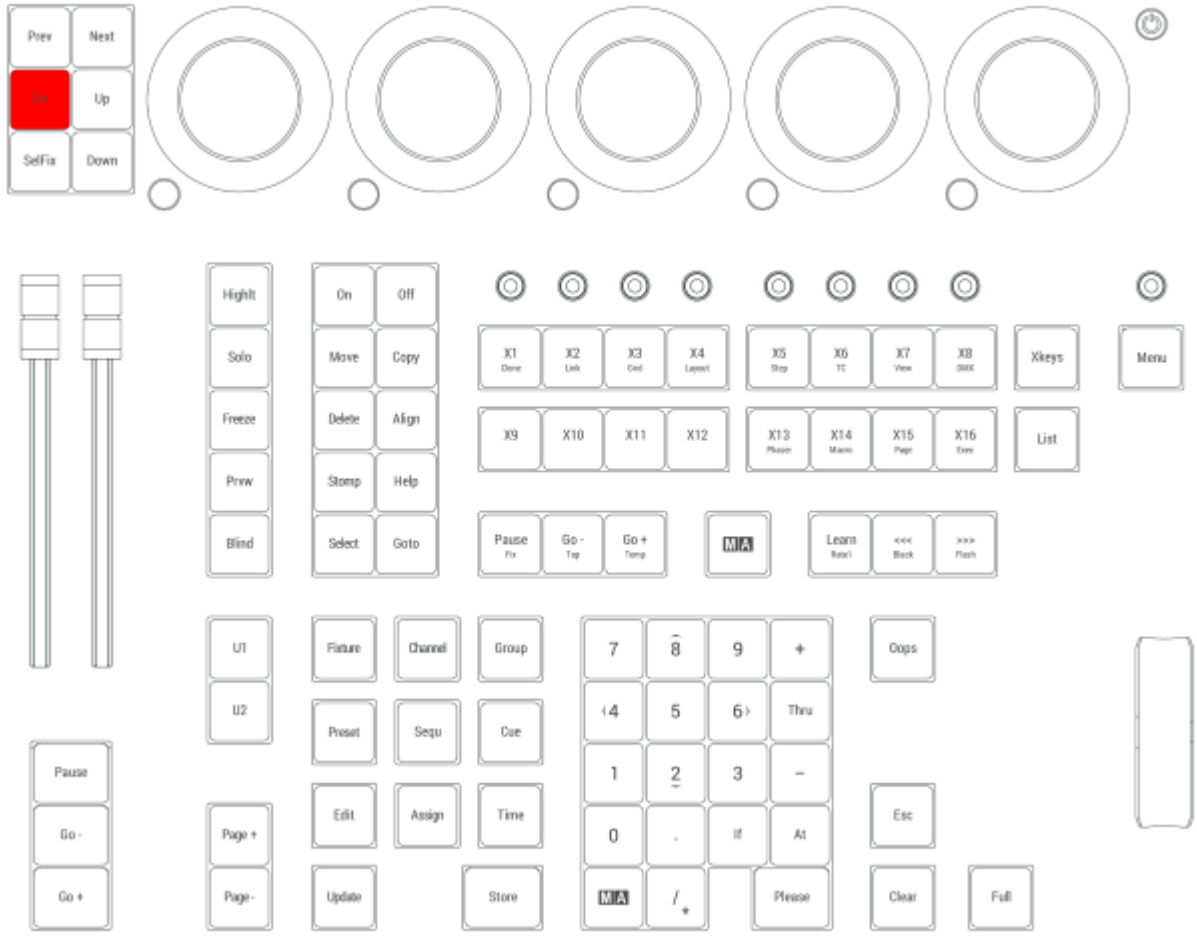
```
OK:Step Toggle Executor
```

For more information about steps, see the **Step Bar** topic.

Set is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



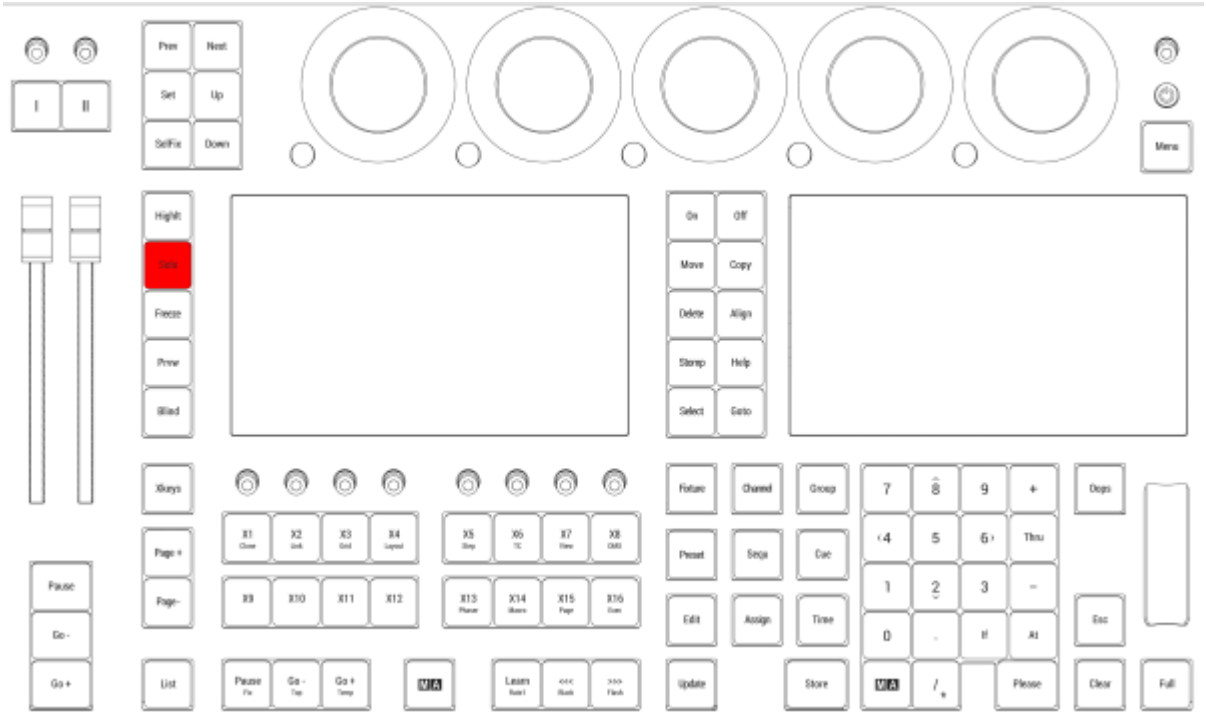
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.55. Solo key

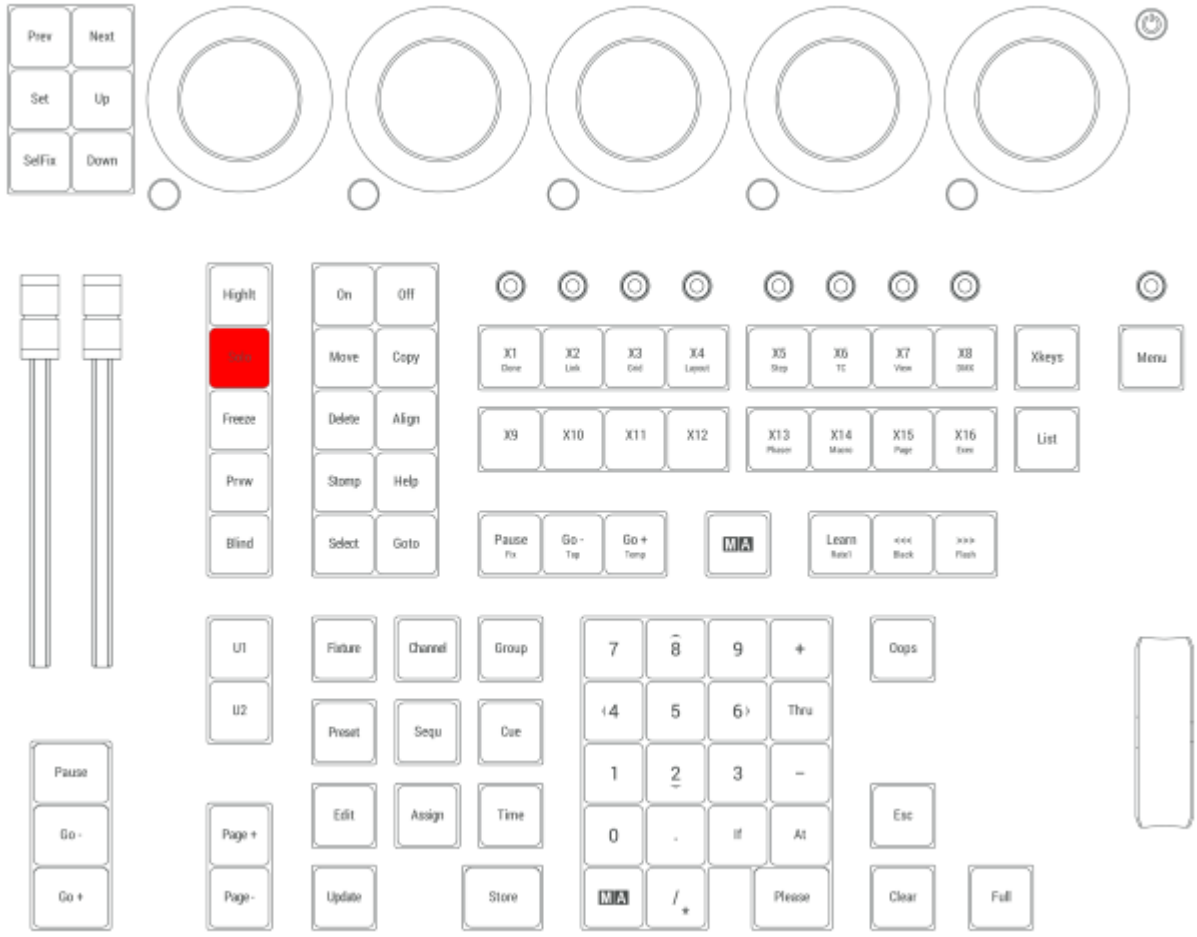
Pressing Solo executes directly solo selection and turns the Solo mode on or off.

For more information about Solo, see the **Solo keyword** topic.

Solo is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.56. Stomp key

Pressing Stomp enters Stomp into the command line.

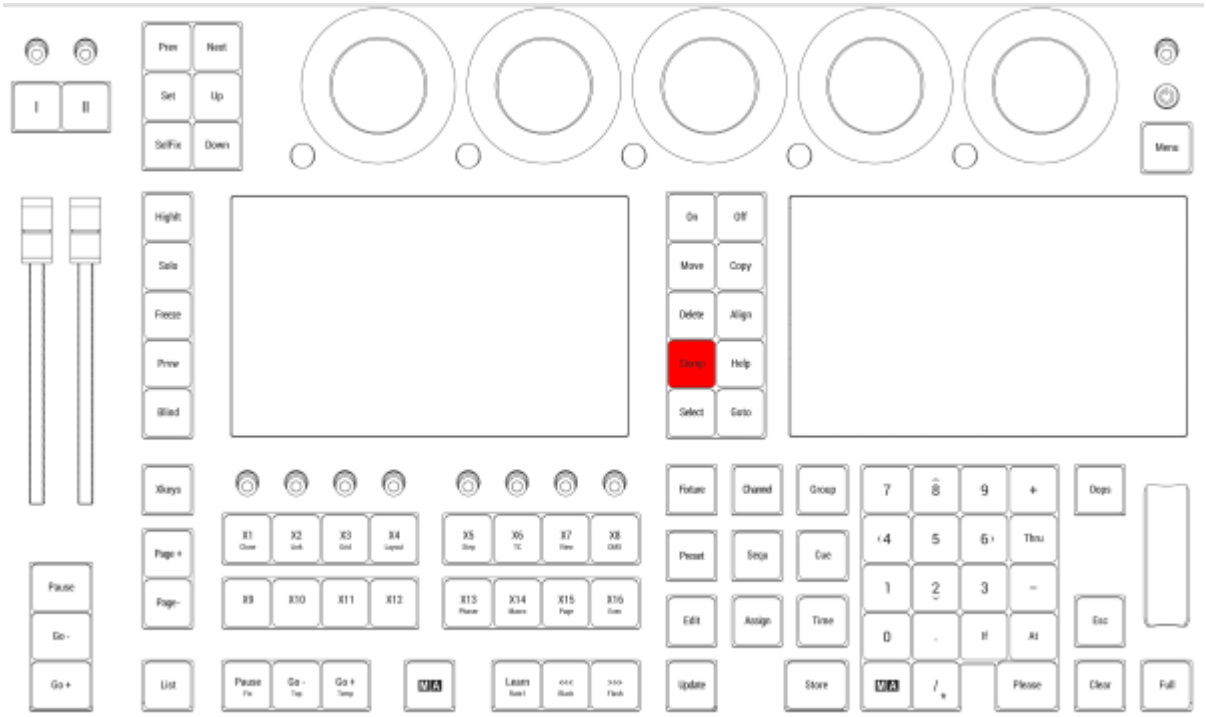
For more information on Stomp, see the **Stomp keyword** topic.

Capture

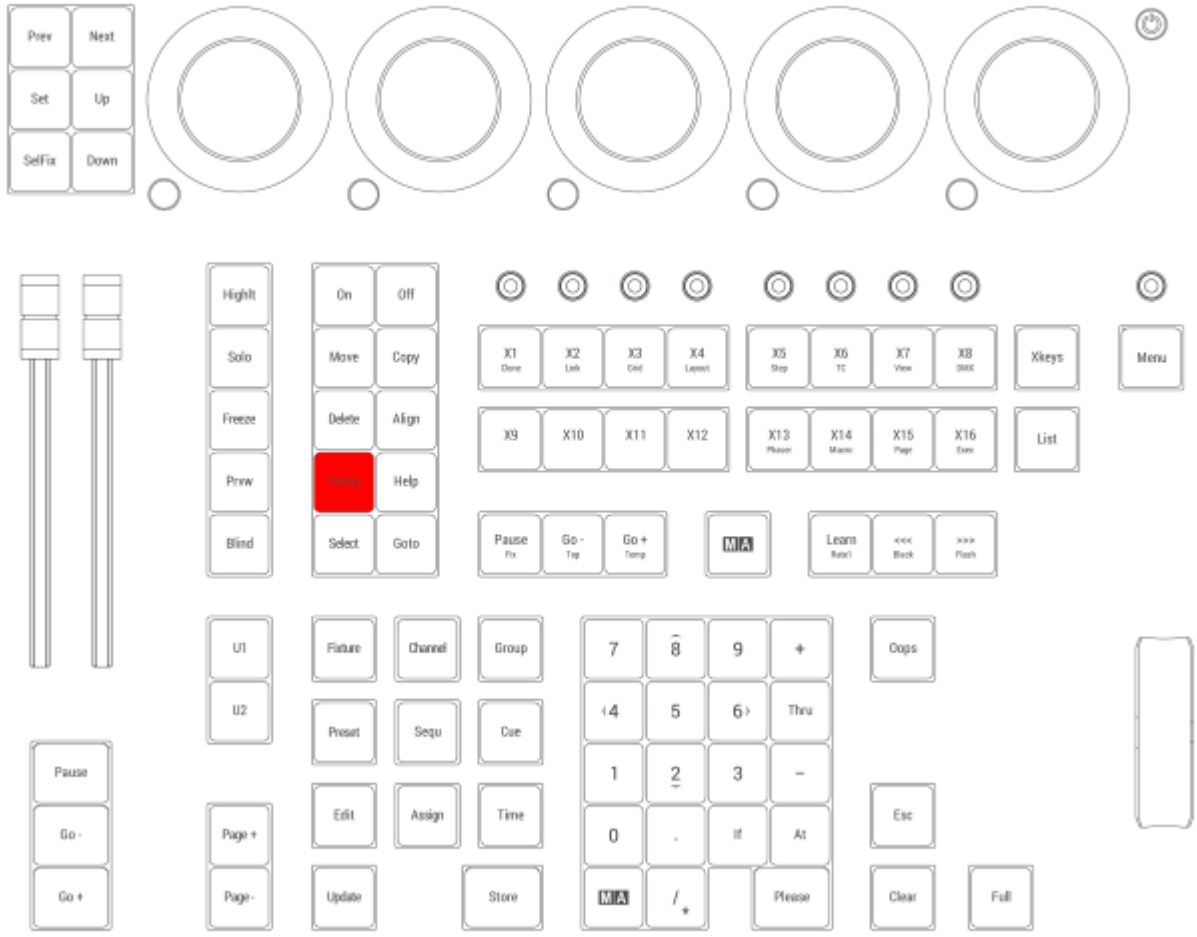
Pressing Stomp Stomp enters Capture into the command line.

For more information on Capture, see the **Capture keyword** topic.

Stomp is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



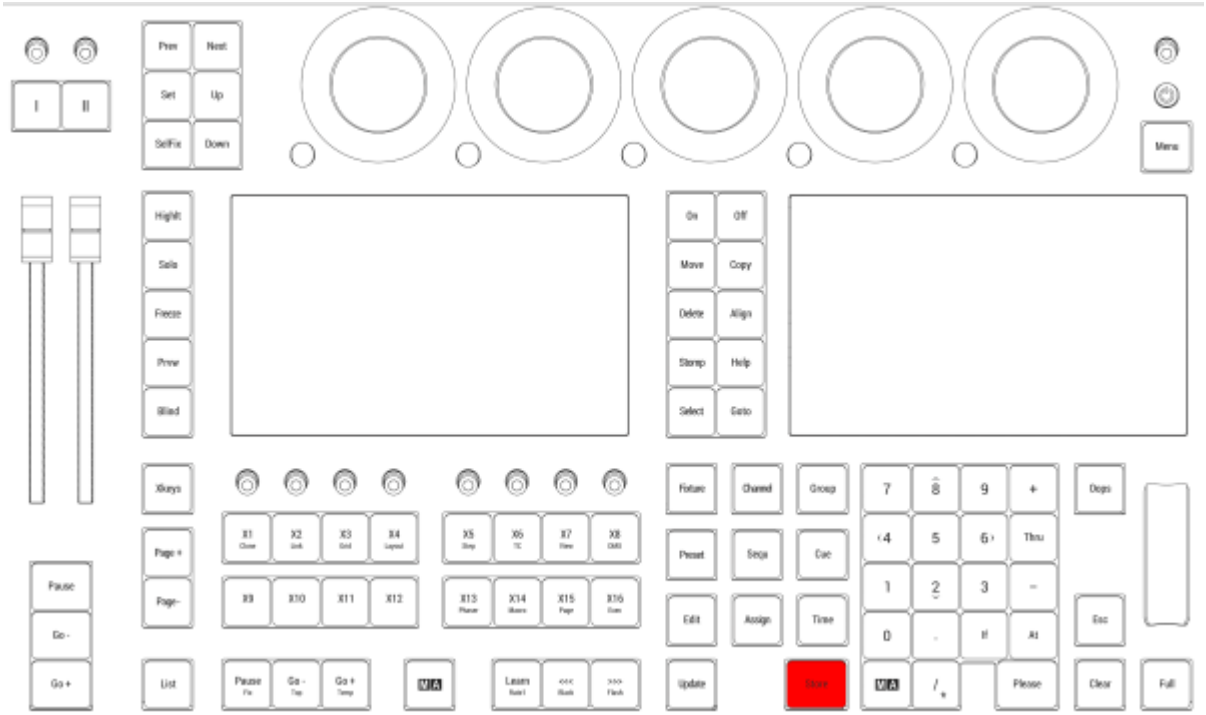
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.57. Store key

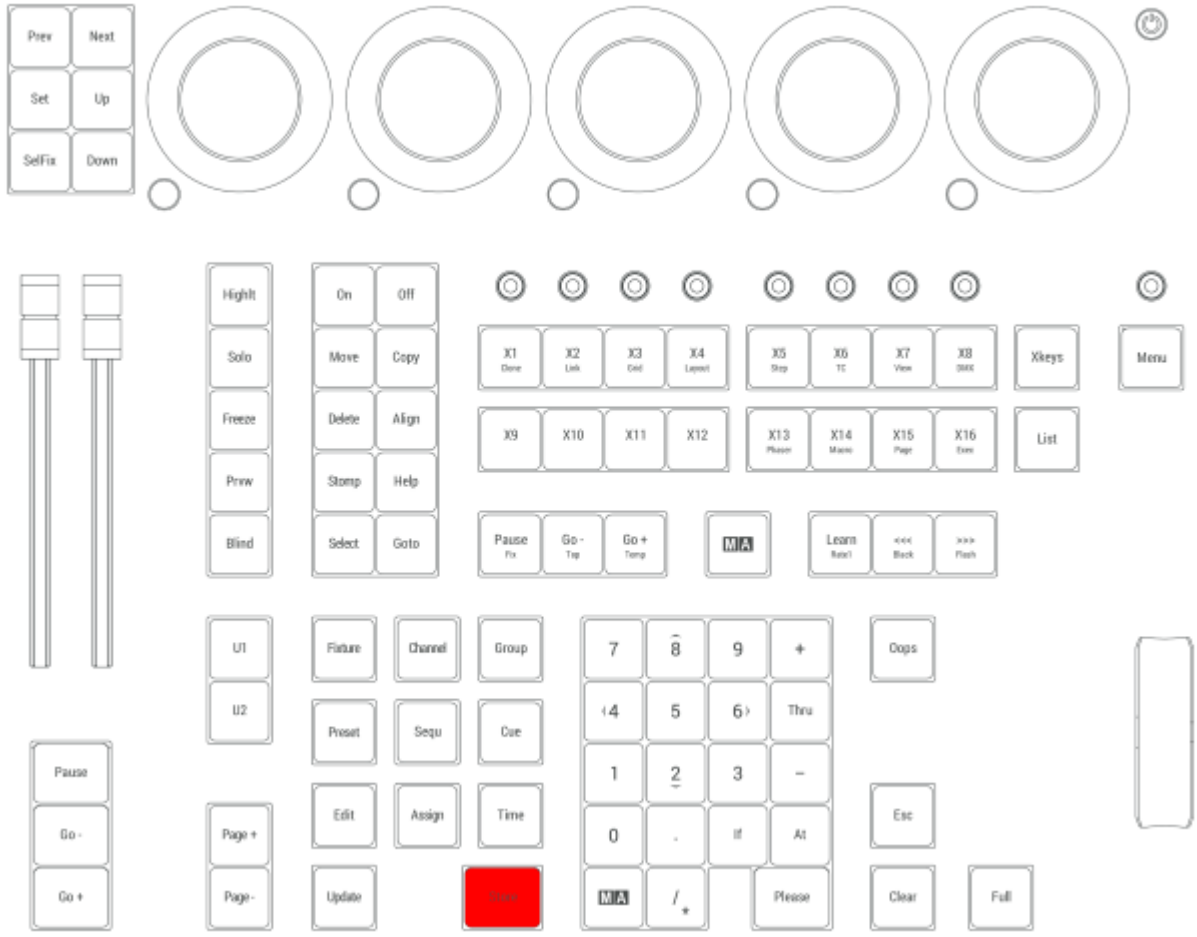
Pressing Store enters the Store keyword into the command line.

For more information about Store, see the **Store keyword** topic.

Store is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



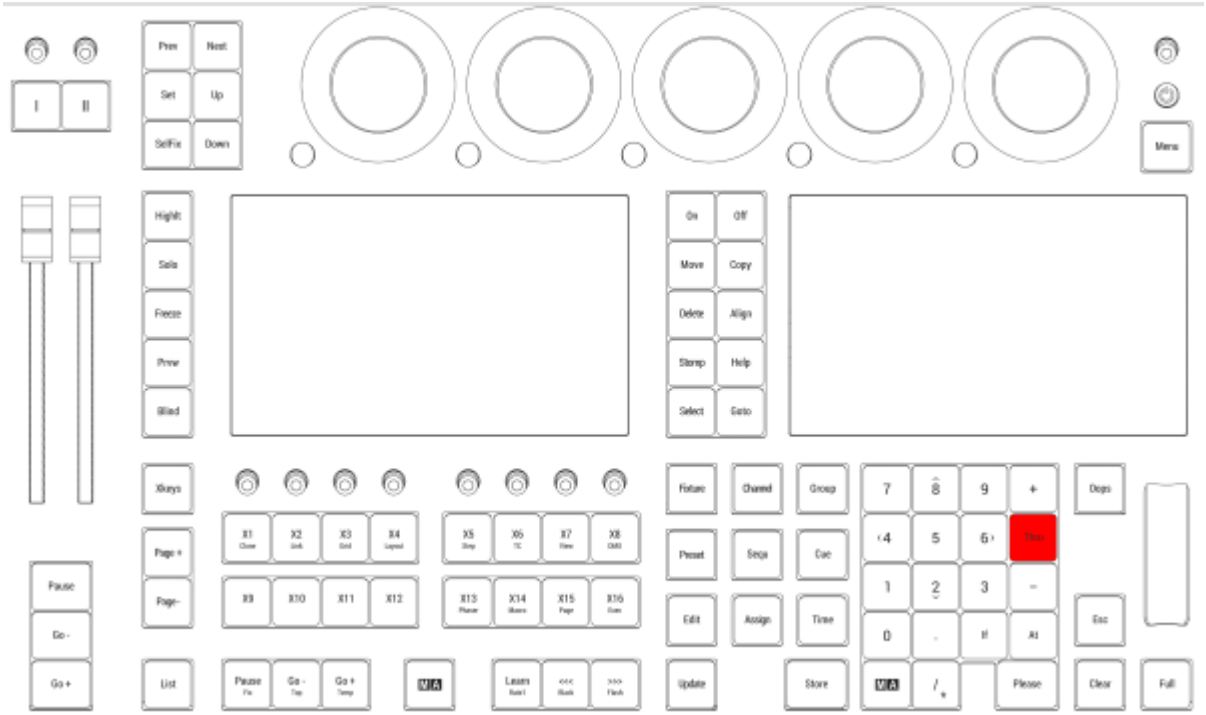
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.58. Thru key

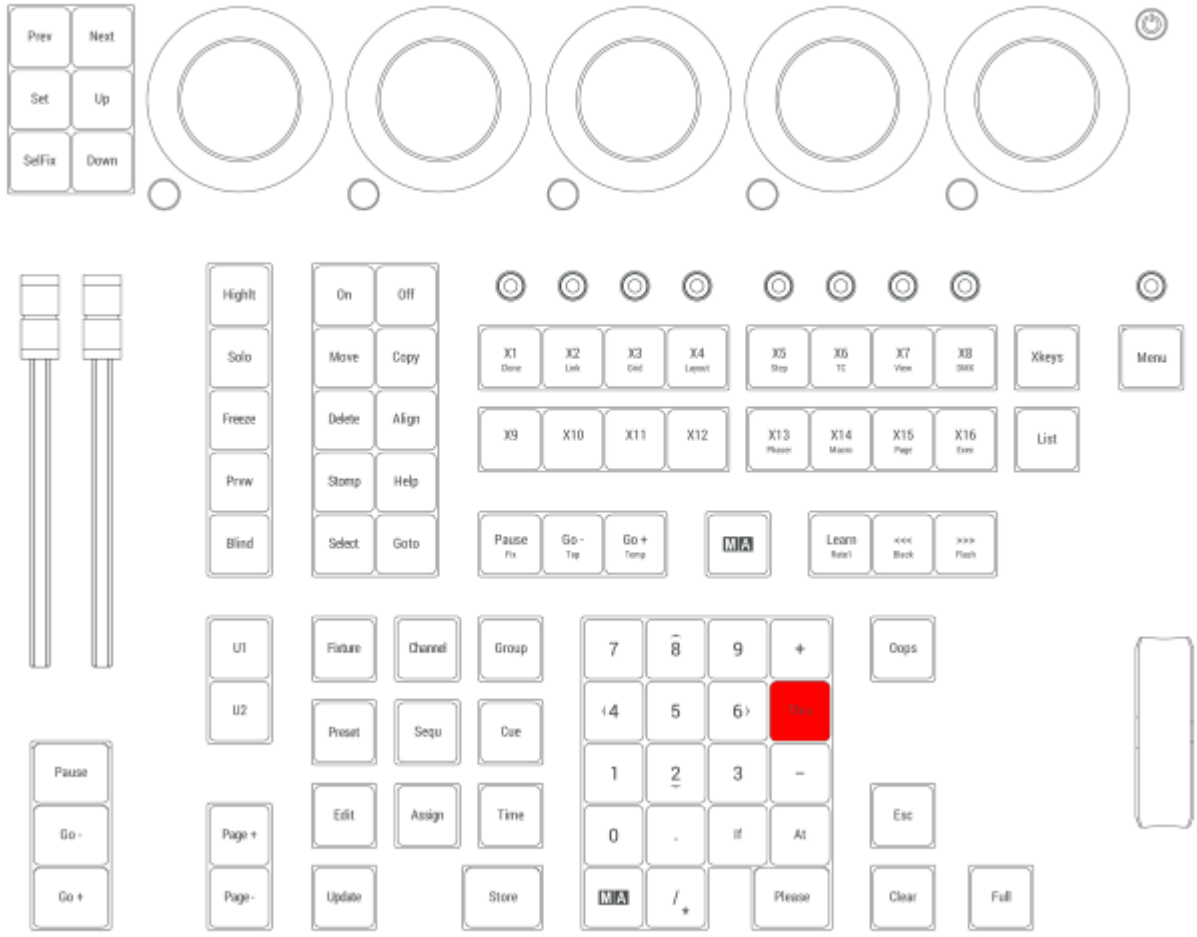
Pressing Thru enters the Thru keyword into the command line.

For more information about Thru, see the **Thru keyword** topic.

Thru is located in the numeric keys section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.59. Time key

The default functionality of the Time key depends upon the "Time Key Target" setting for the current user profile.

If the "Time Key Target" is set to "Cue" and the command line is empty, pressing Time toggles between CueFade and CueDelay in the command line.

For more information about Time, see the **CueFade keyword** or **CueDelay keyword** topic.

If the "Time Key Target" is set to "Cue" and the command line contains a fixture selection, pressing Time toggles between Fade and Delay in the command line.

For more information about Time, see the **Fade keyword** or **Delay keyword** topic.

If the "Time Key Target" is set to "Fixture" and the command line is empty, pressing Time toggles between Fade and Delay in the command line.

For more information about Time, see the **Fade keyword** or **Delay keyword** topic.

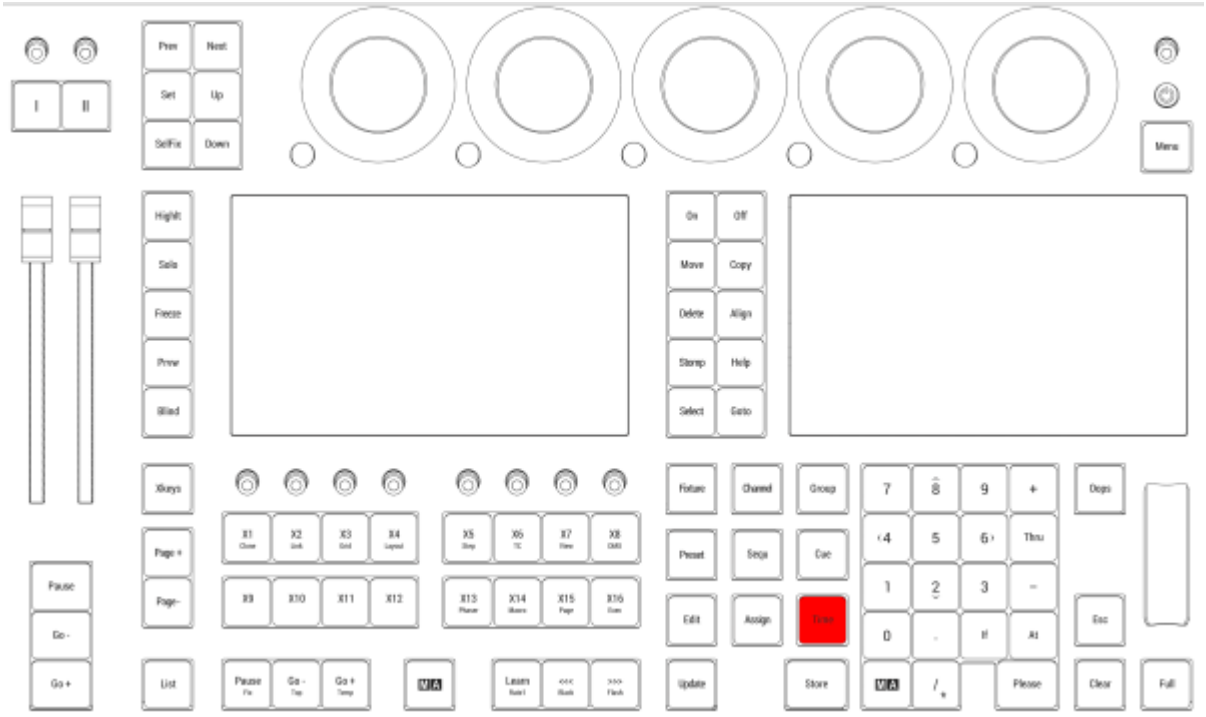
If the "Time Key Target" is set to "Fixture" and either Store or Cue is in the command line, pressing Time toggles between CueFade and CueDelay in the command line.

For more information about Time, see the **CueFade keyword** or **CueDelay keyword** topic.

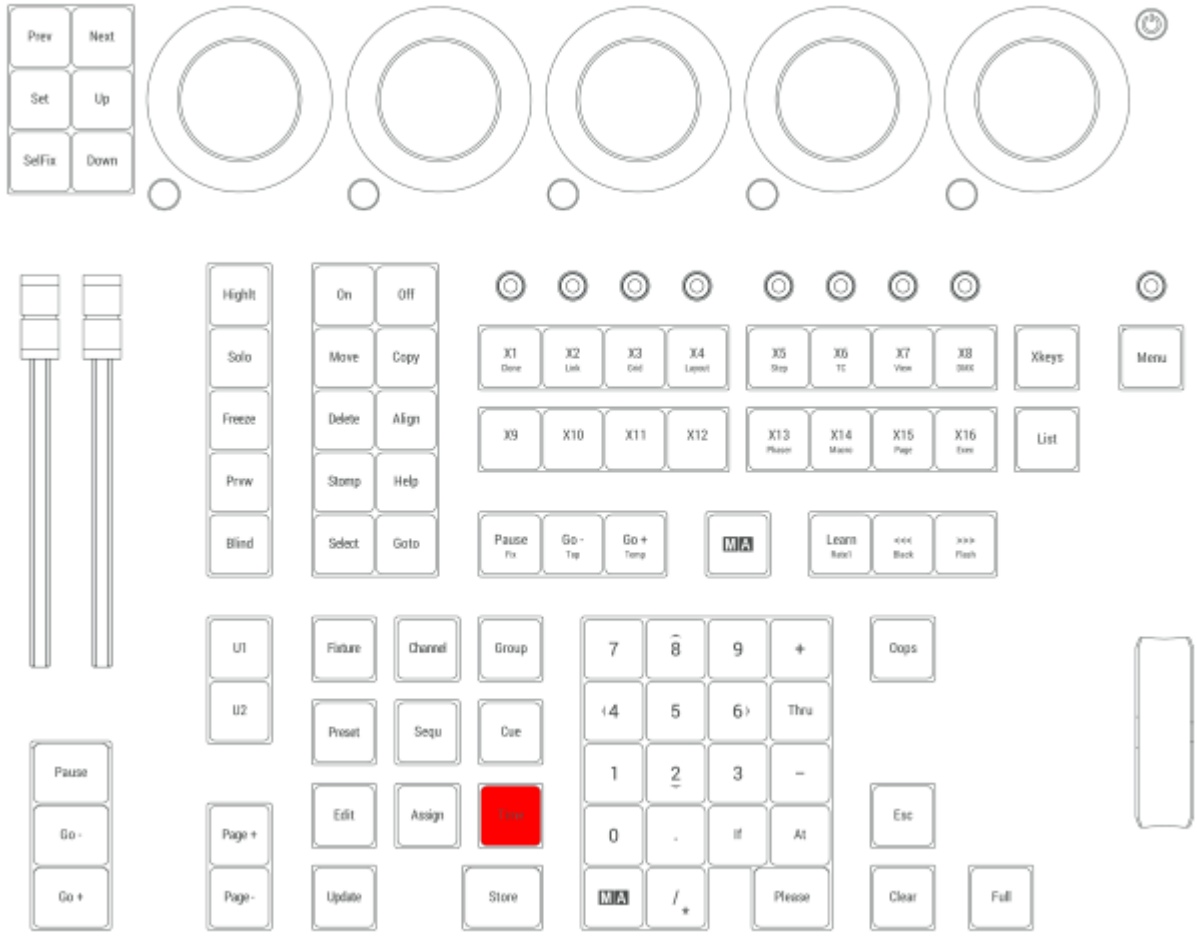
Regardless of the "Time Key Target" setting, pressing MA Time toggles between Relative, Fade, Delay, and Absolute.

For more information about Time, see the **Relative keyword**, **Fade keyword**, **Delay keyword**, or **Absolute keyword** topic.

Time is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

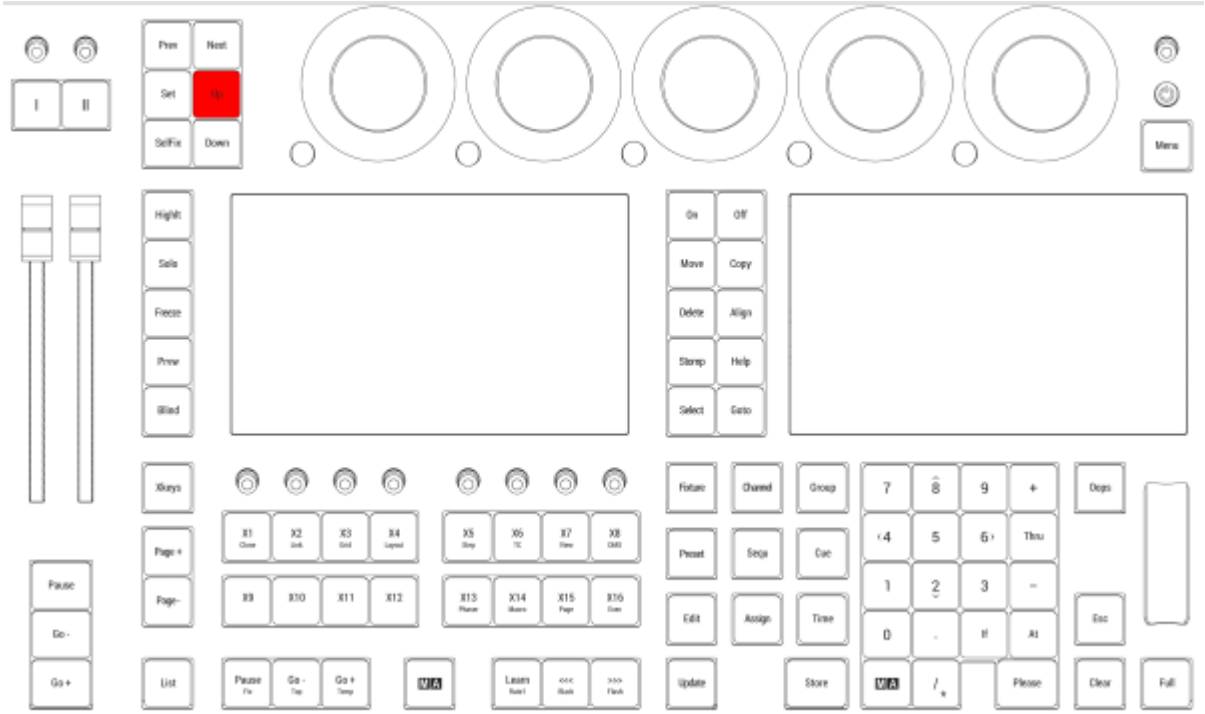


Location on grandMA3 compact consoles and grandMA3 onPC command wing

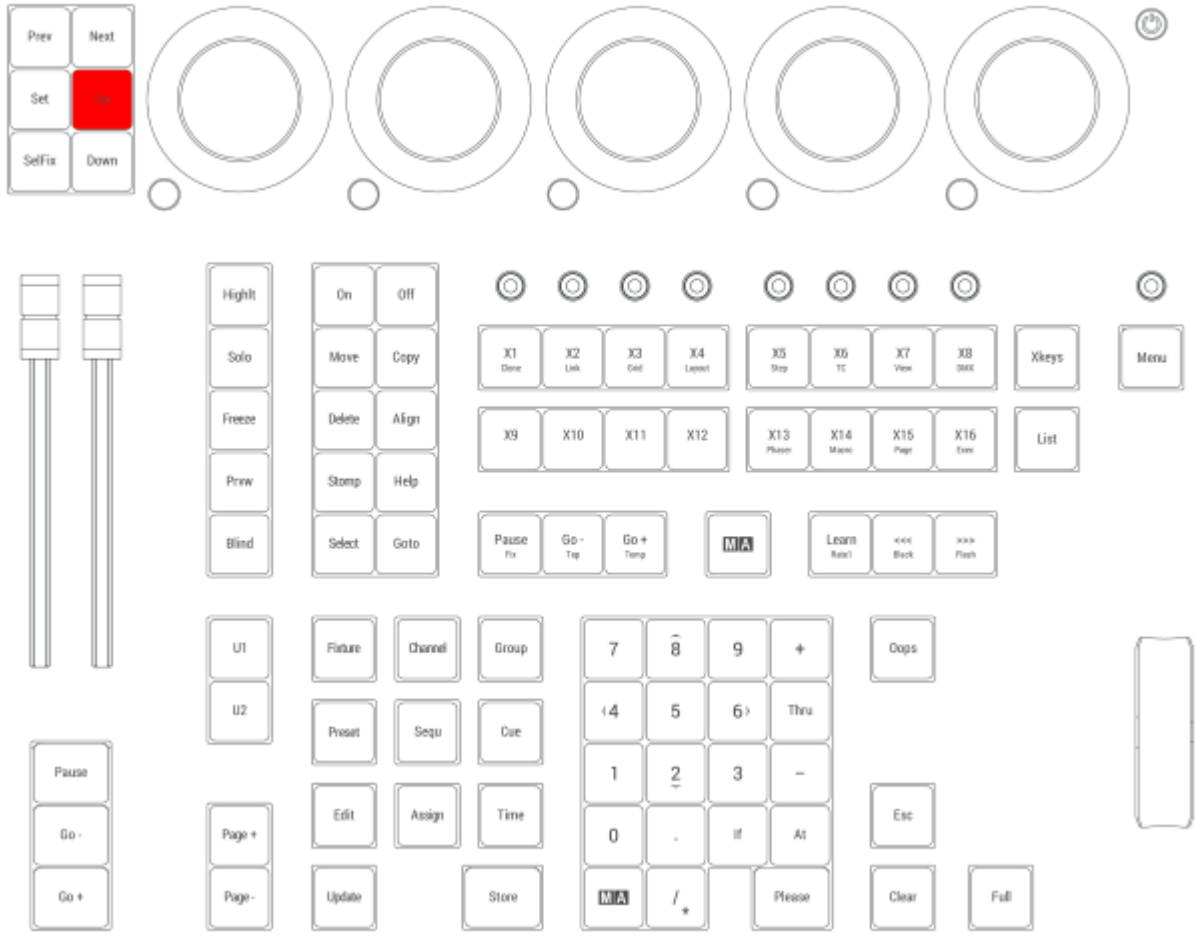
2.15.60. Up key

Pressing Up navigates up in the fixture/subfixture structure.

Up is located in the command section on the left side of the five dual encoders.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.61. Update key

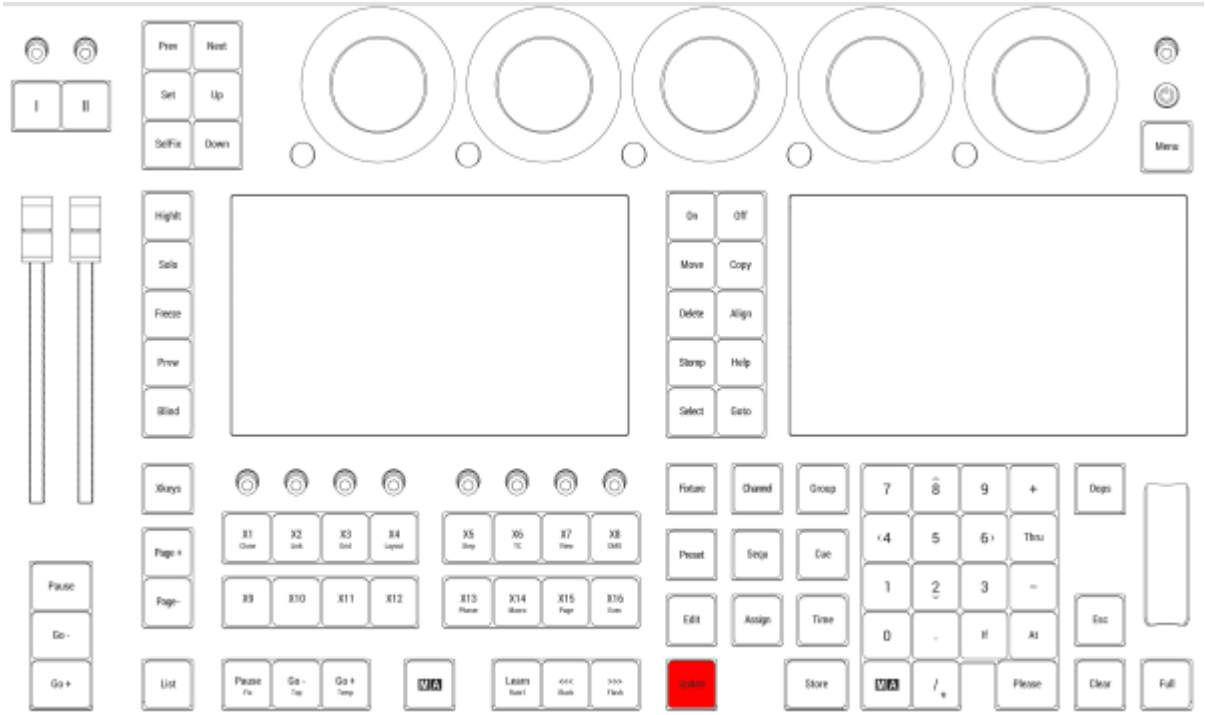
Pressing Update enters the Update keyword into the command line.

For more information about Update, see the **Update Keyword** topic.

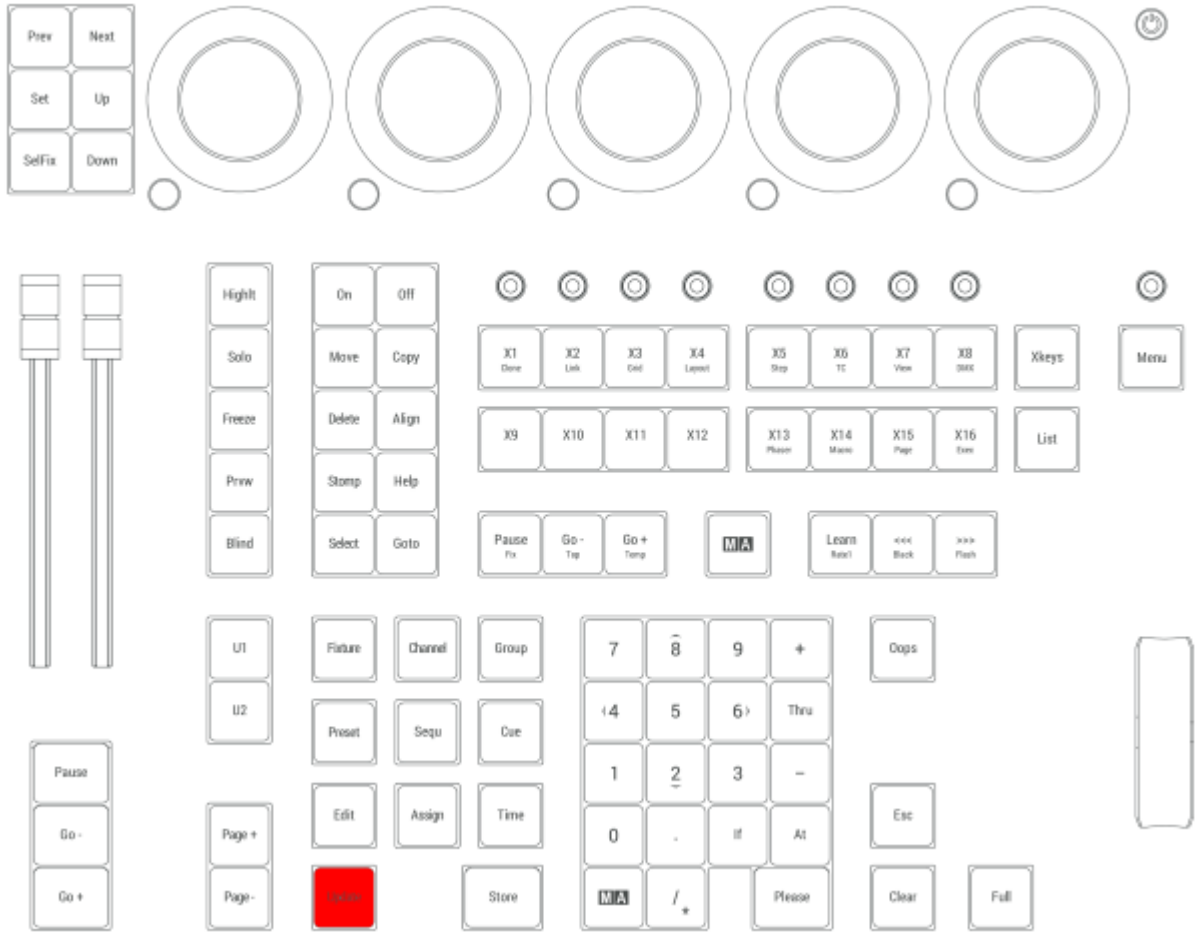
Pressing MA + Update enters the Cook keyword into the command line.

For more information about Cook, see the **Cook Keyword** topic.

Update is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

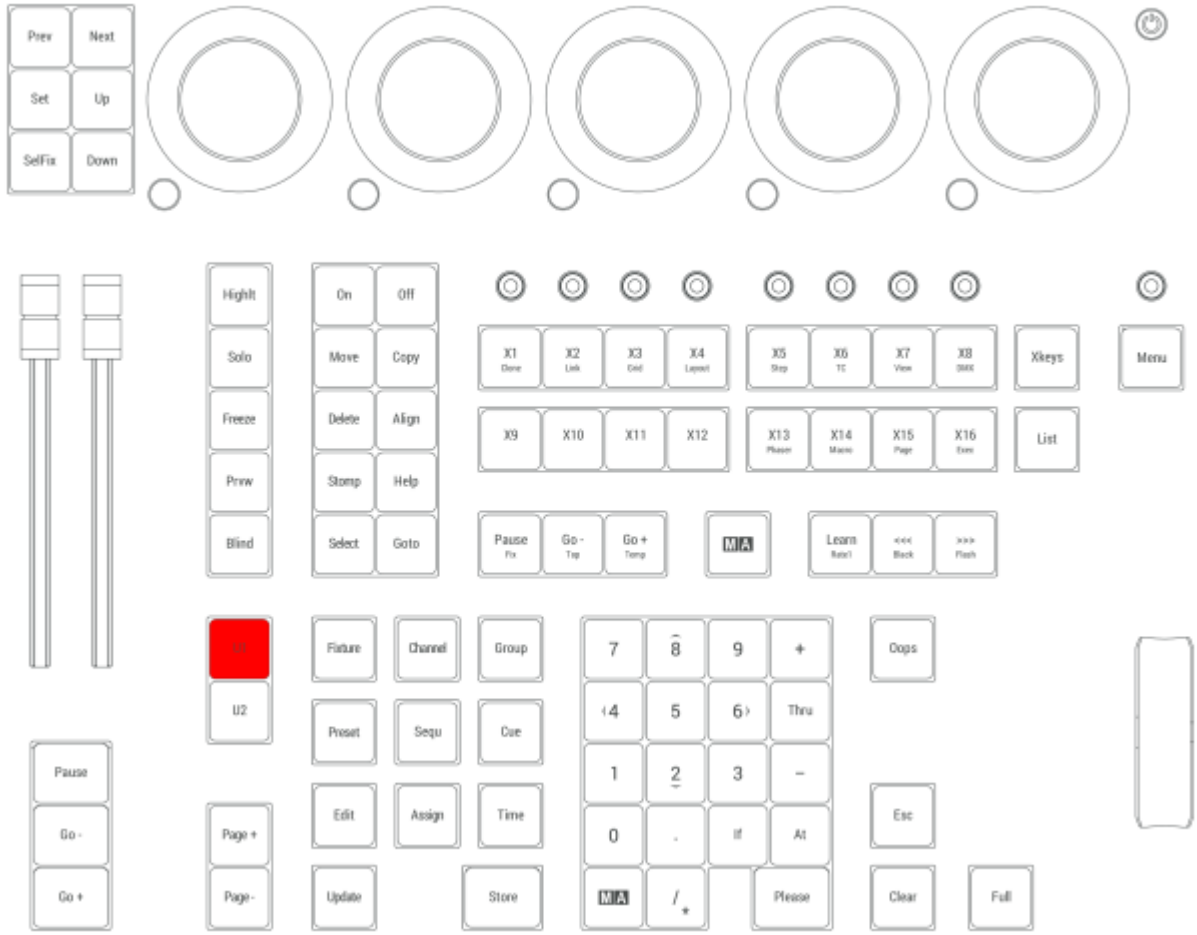
2.15.62. U1 key

Pressing U1 executes the User1 command in the command line.

This key is only present on grandMA3 compact, grandMA3 compact XT, grandMA3 onPC command wing and grandMA3 onPC command wing XT.

For more information about U1, see the **User1 keyword topic**.

U1 is located in the command section on the left side of the numeric keys.



Location on grandMA3 compact consoles and grandMA3 onPC command wings

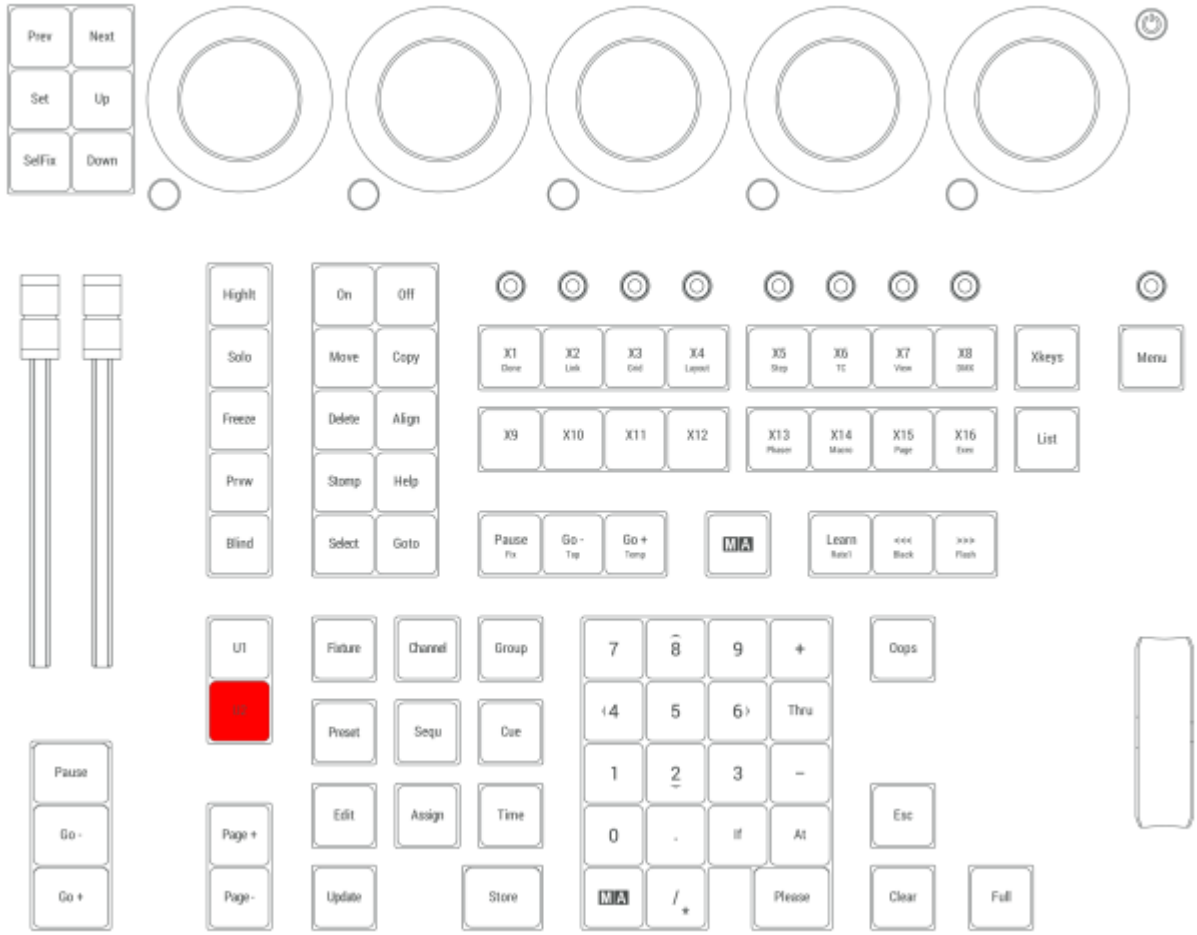
2.15.63. U2 Key

Pressing U2 executes the User2 command in the command line.

This key is only present on grandMA3 compact, grandMA3 compact XT, grandMA3 onPC command wing and grandMA3 onPC command wing XT.

For more information about U2, see the **User2 keyword topic**.

U2 is located in the command section on the left side of the numeric keys.



Location on grandMA3 compact consoles and grandMA3 onPC command wings

2.15.64. X1 | Clone key

At the moment all Xkeys behave like executors. X1 is executor 291.

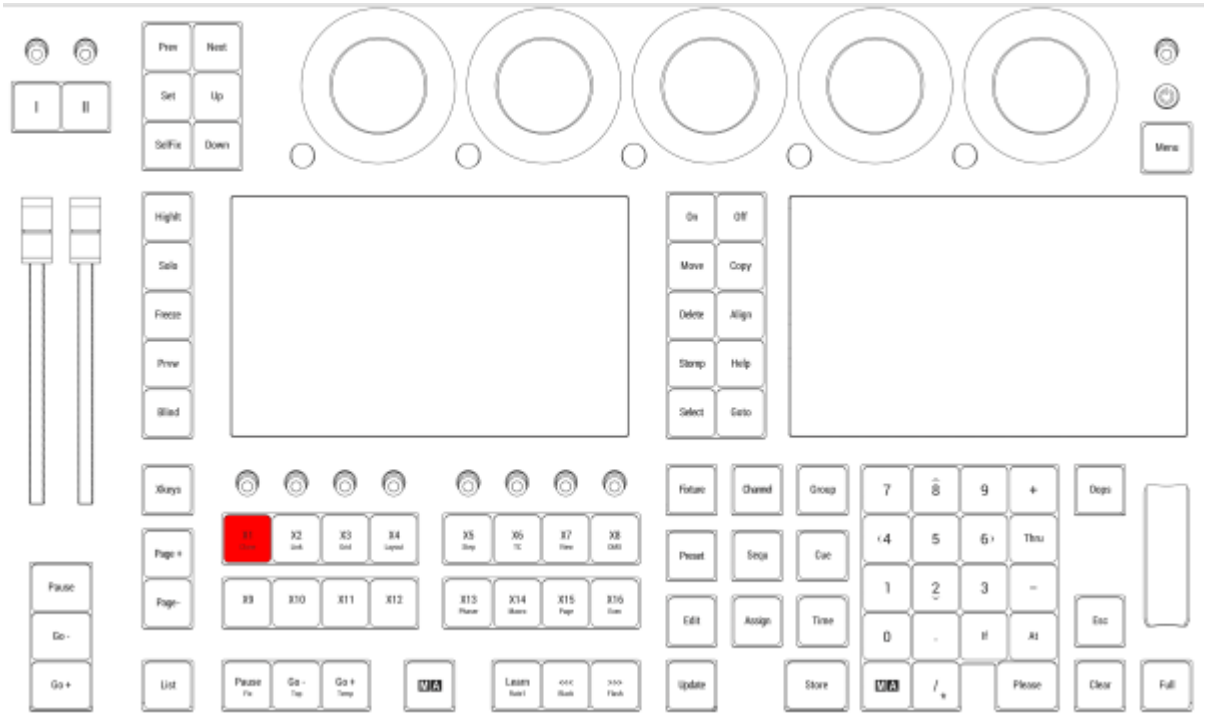
Clone

Pressing and holding MA + X1 enters the Clone keyword into the command line.

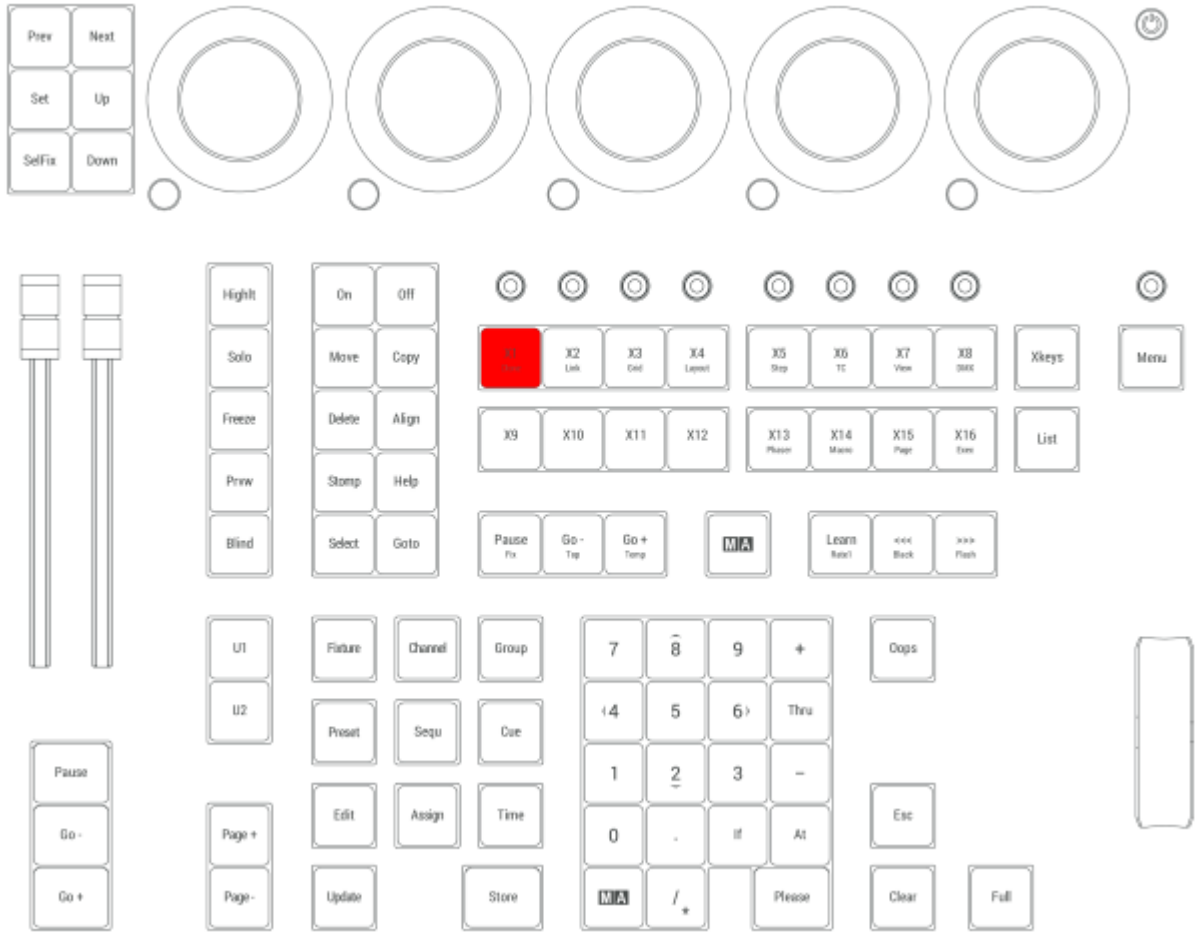
```
User name[Fixtures]> Clone
```

For more information about Clone, see the **Clone keyword** topic.

X1 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

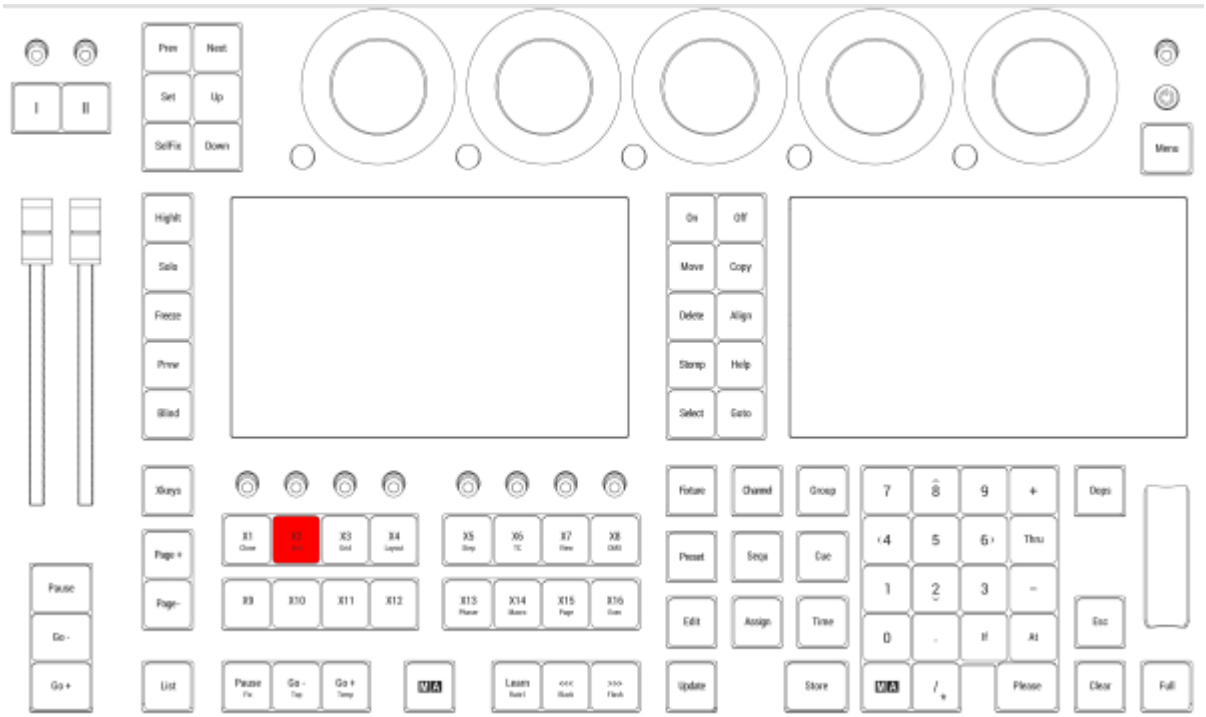


Location on grandMA3 compact consoles and grandMA3 onPC command wing

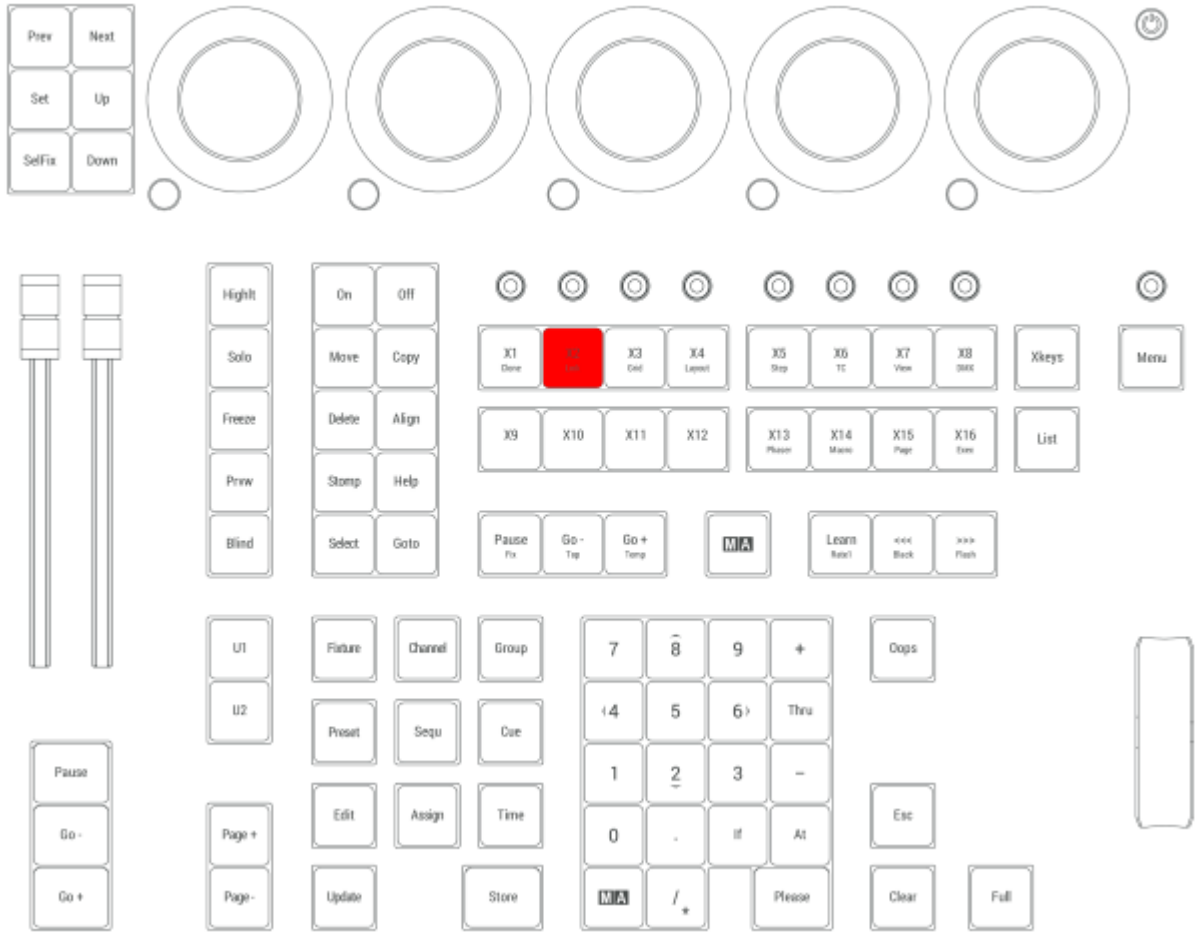
2.15.65. X2 | Link key

At the moment all Xkeys behave like executors. X2 is executor 292.

X2 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.66. X3 | Grid key

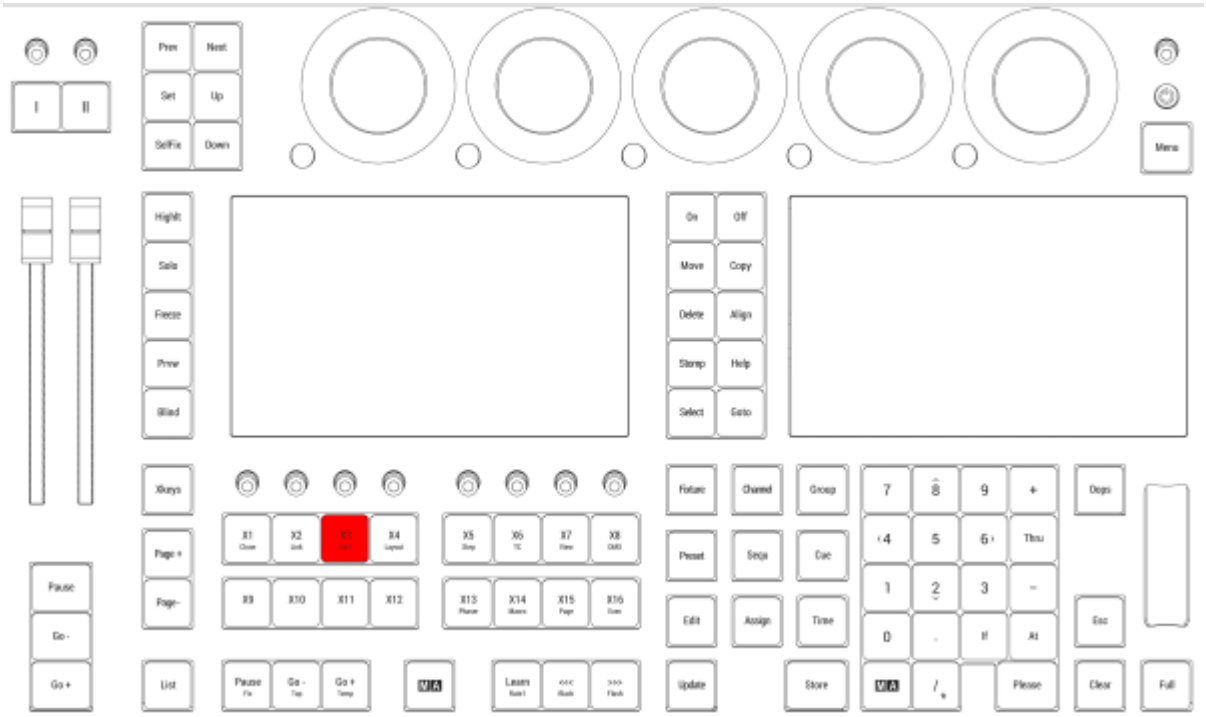
At the moment all Xkeys behave like executors. X3 is executor 293.

Grid

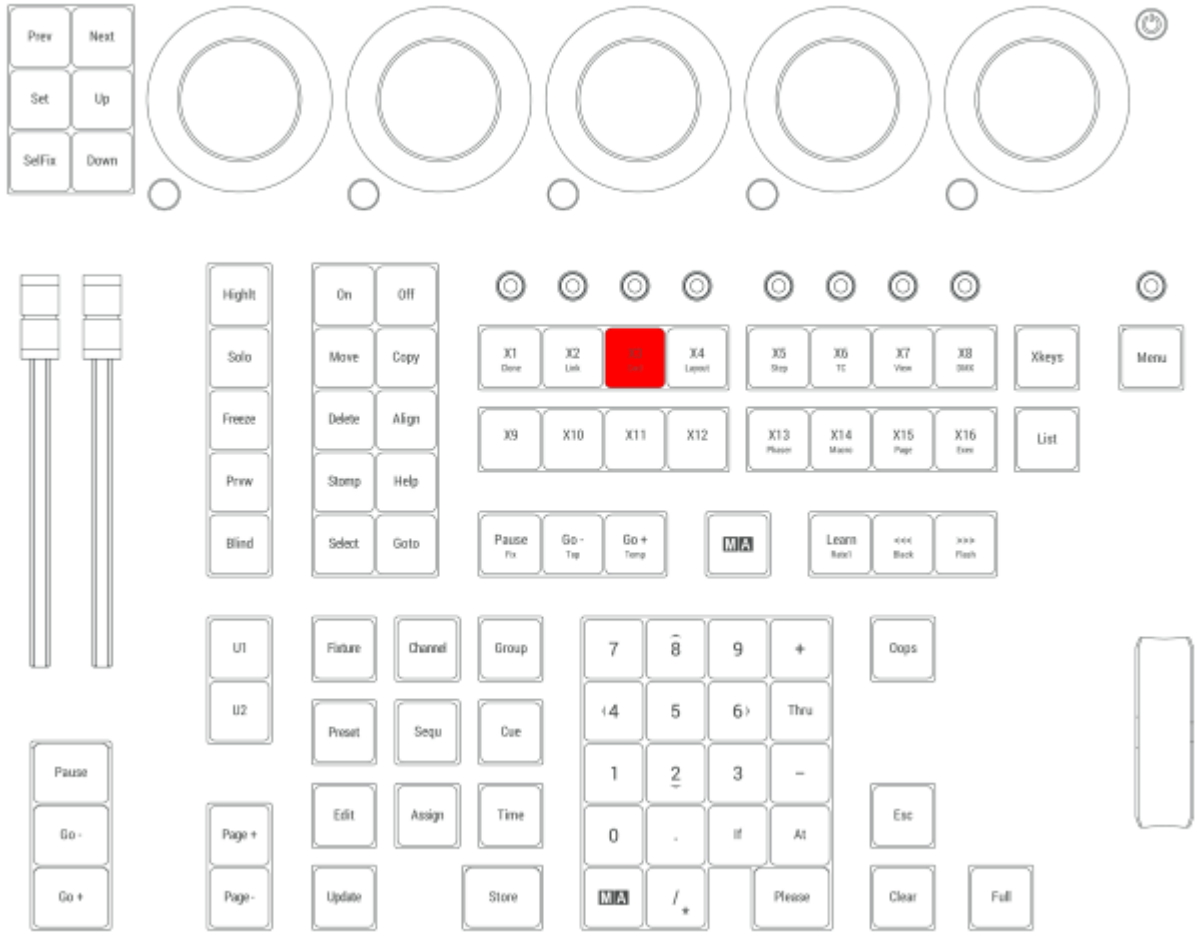
Pressing and holding MA + X3 enters the Grid keyword into the command line.

For more information about Grid, see the **Grid keyword** topic.

X3 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.67. X4 | Layout key

At the moment all Xkeys behave like executors. X4 is executor 294.

Layout

Pressing and holding MA + X4 enters the Layout keyword into the command line.

For more information about Layout, see the **Layout keyword** topic.

Appearance

Pressing and holding MA + X4 + X4 enters the Appearance keyword into the command line.

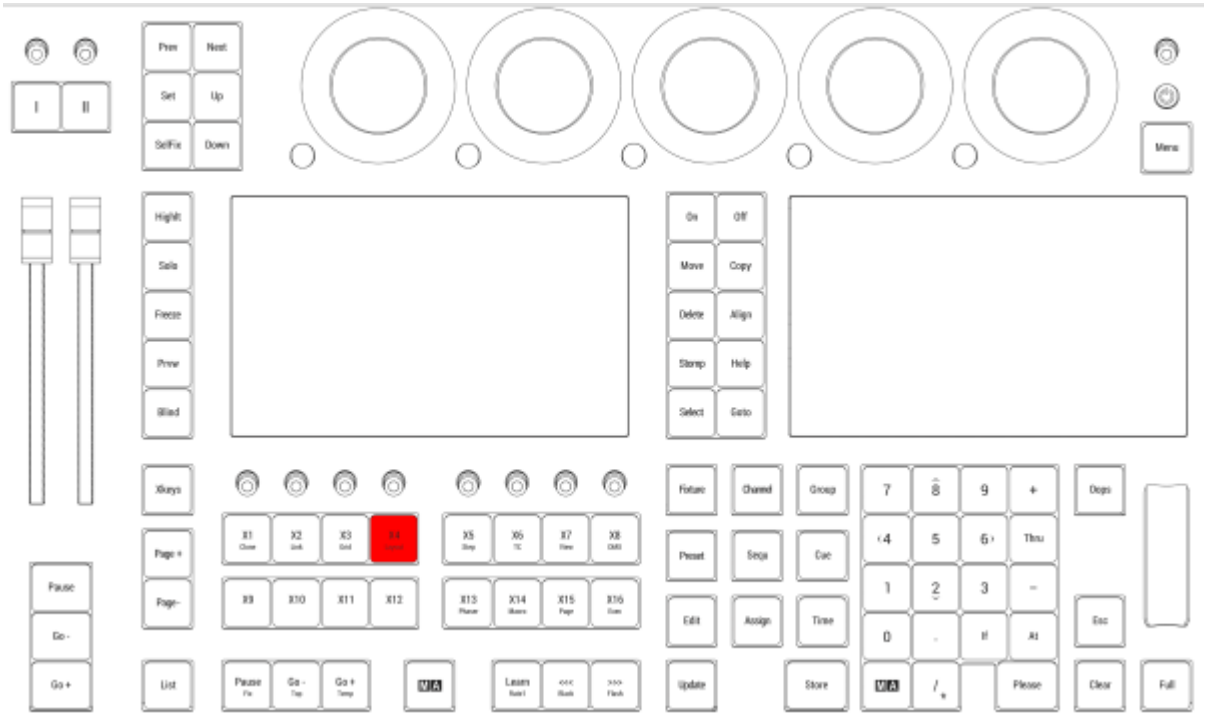
For more information about Appearance, see the **Appearance keyword** topic.

Scribble

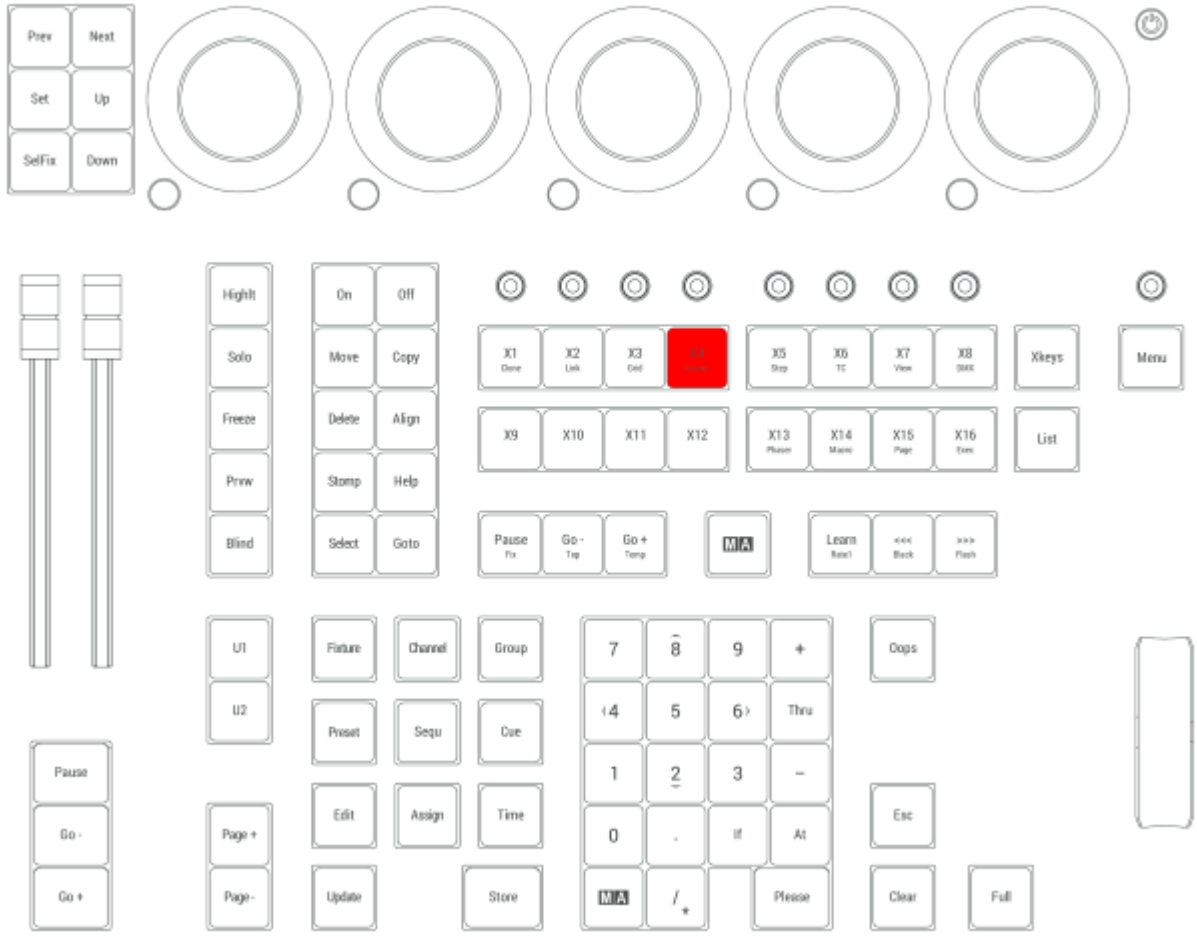
Pressing and holding MA + X4 + X4 + X4 enters the Scribble keyword into the command line.

For more information about Scribble, see the **Scribble keyword** topic.

X4 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.68. X5 | Step key

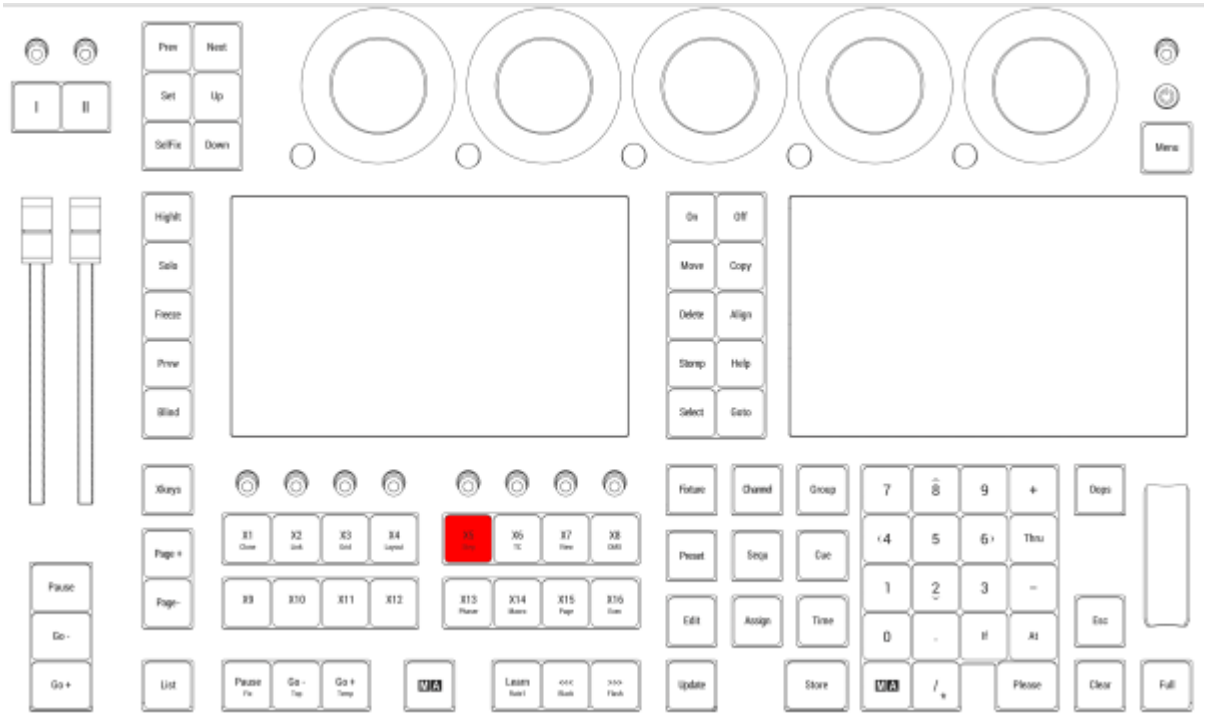
At the moment all Xkeys behave like executors. X5 is executor 295.

Step

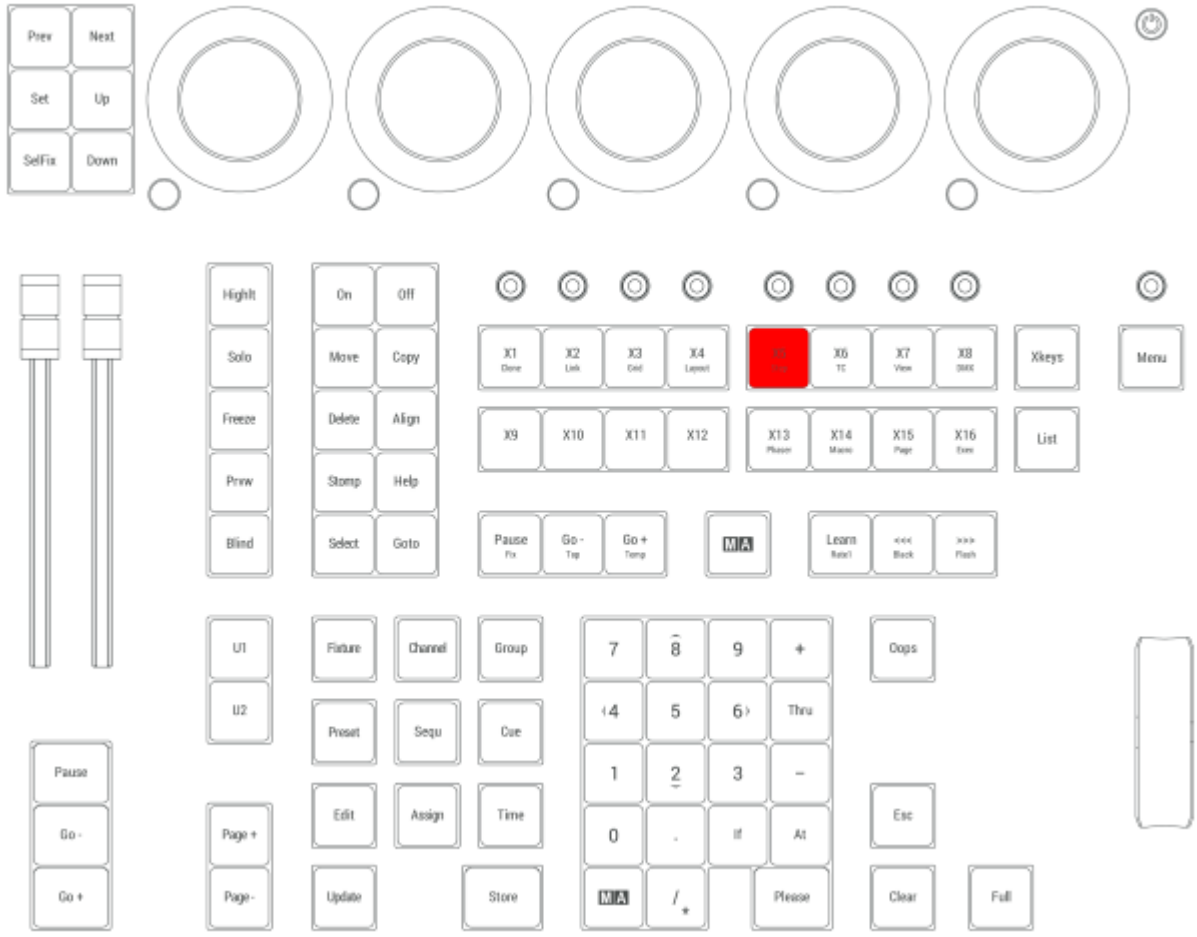
Pressing and holding MA + X5 enters the Step keyword into the command line.

For more information about Step, see the **Step keyword** topic.

X5 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.69. X6 | TC key

At the moment all Xkeys behave like executors. X6 is executor 296.

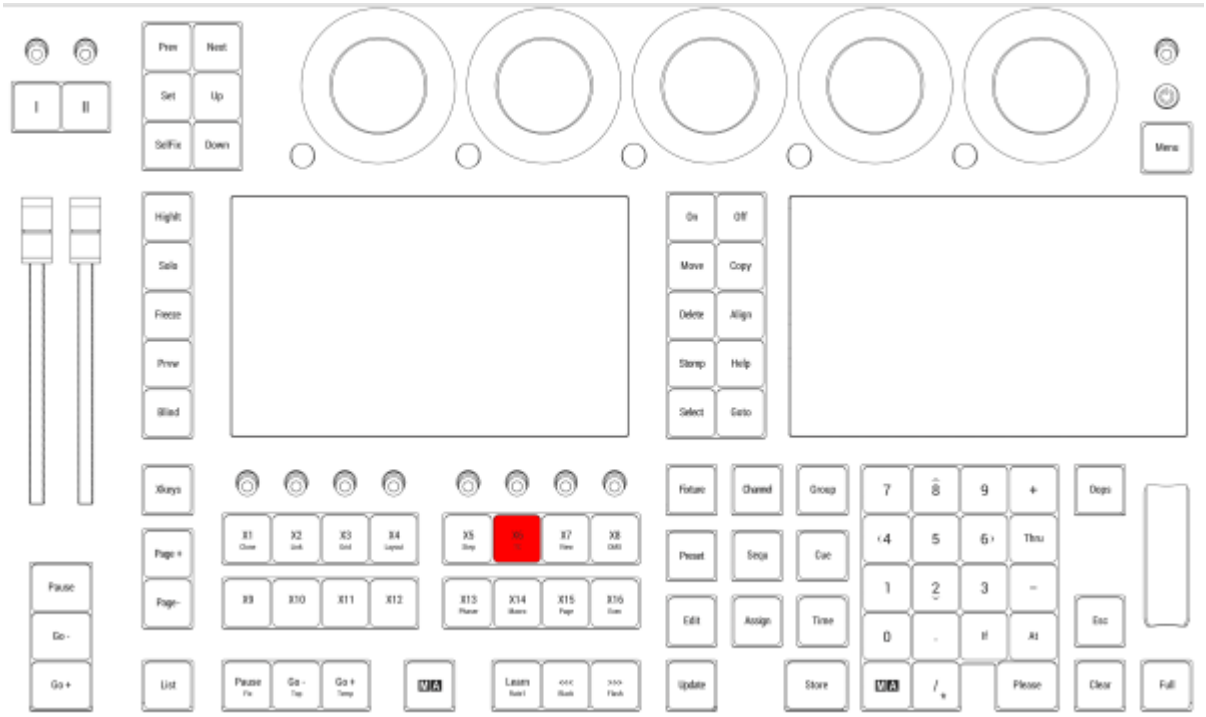
Timecode

Pressing and holding MA + X6 enters the Timecode keyword into the command line.

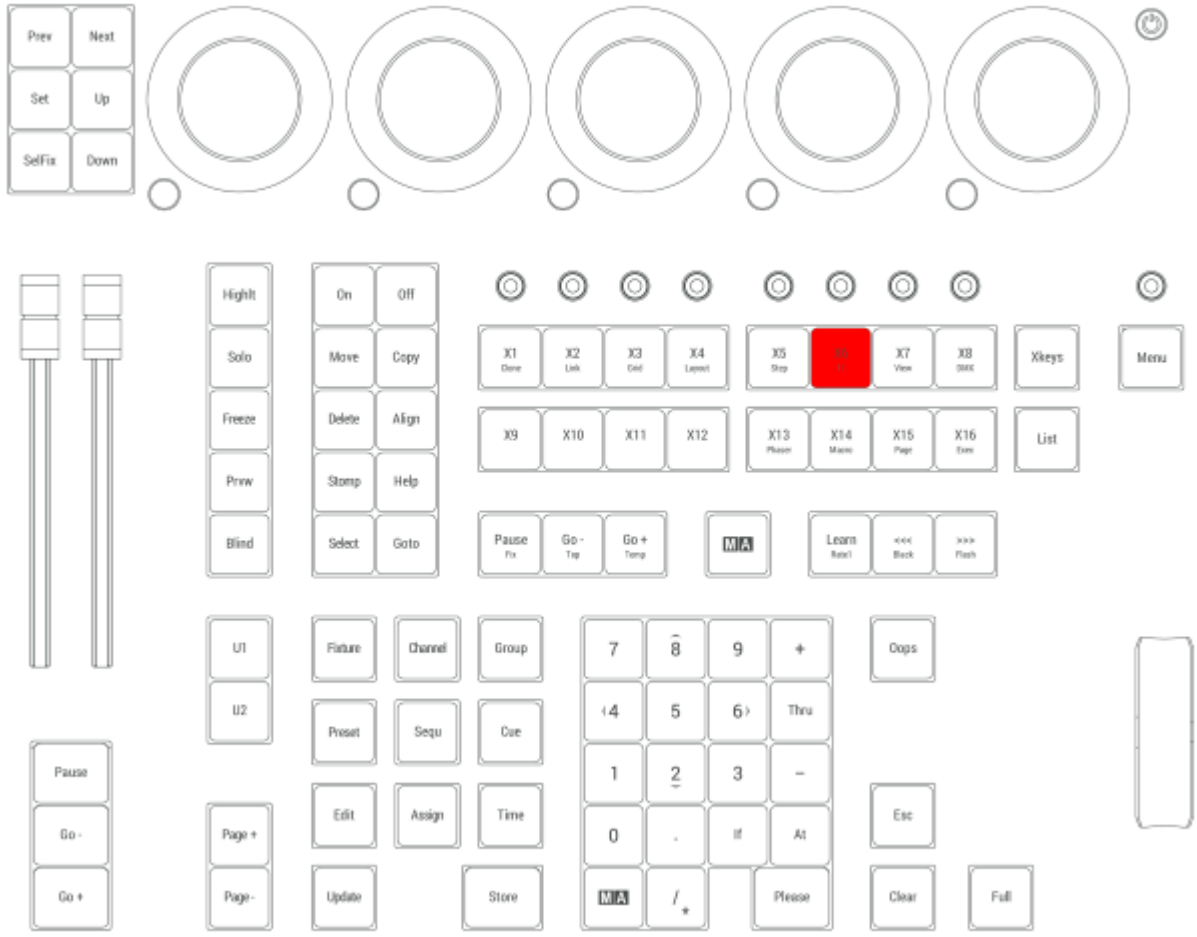
```
User name[Fixtures]> Timecode
```

For more information about Timecode, see the **Timecode keyword** topic.

X6 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.70. X7 | View key

At the moment all Xkeys behave like executors. X7 is executor 297.

View

Pressing and holding MA + X7 enters the View keyword into the command line.

For more information about View, see the **View keyword** topic.

ViewButton

Pressing and holding MA + X7 + X7 enters the ViewButton keyword into the command line.

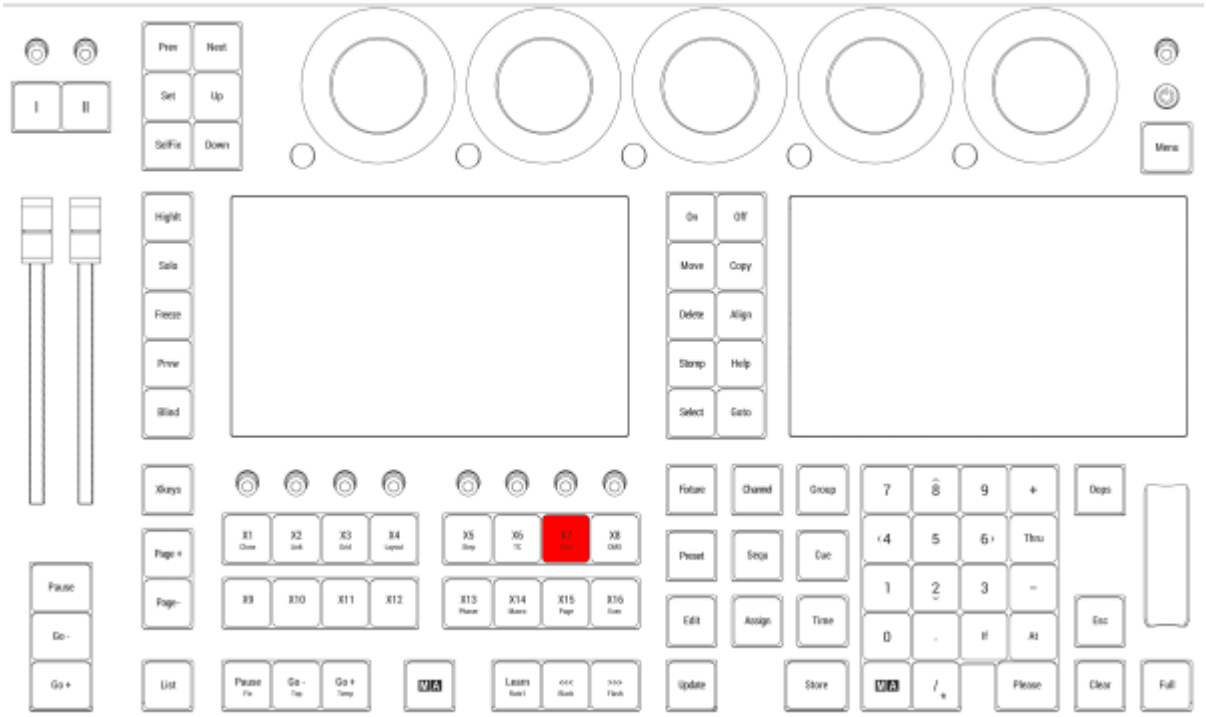
For more information about ViewButton, see the **ViewButton keyword** topic.

ScreenContent

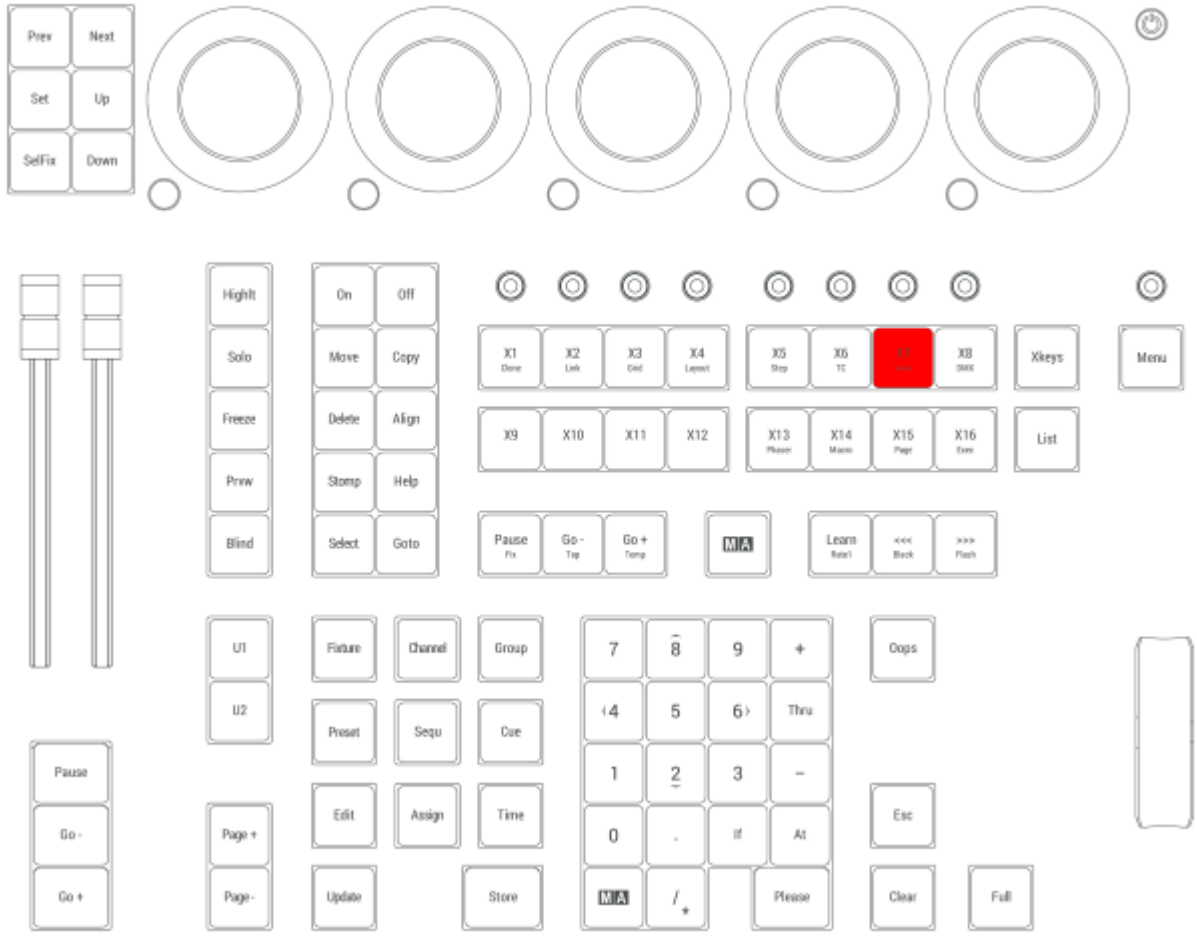
Pressing and holding MA + X7 + X7 + X7 enters the ScreenContent keyword into the command line.

For more information about ScreenContent, see the **ScreenContent keyword** topic.

X7 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.71. X8 | DMX key

At the moment all Xkeys behave like executors. X8 is executor 298.

DMXAddress

Pressing and holding MA + X8 enters the DMXAddress keyword into the command line.

```
MA  User name[Fixtures]> DMXAddress
```

For more information about DMXAddress, see the **DMXAddress keyword** topic.

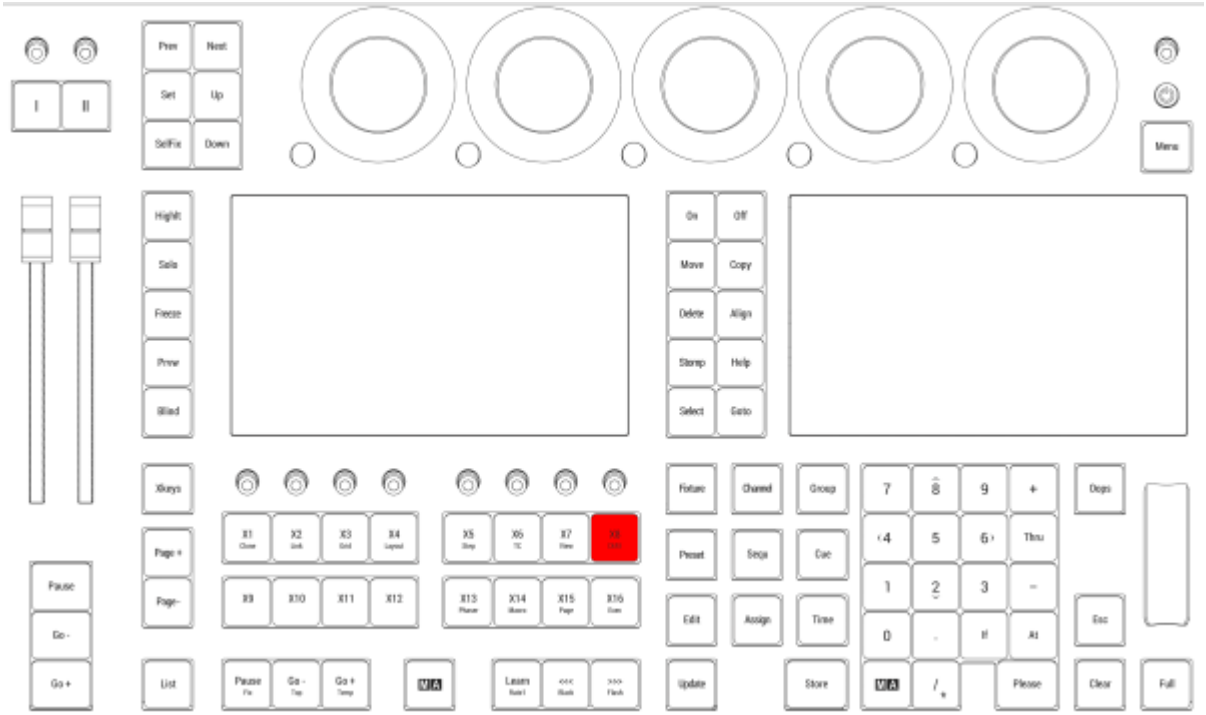
DMXUniverse

Pressing and holding MA + X8 + X8 enters the DMXUniverse keyword into the command line.

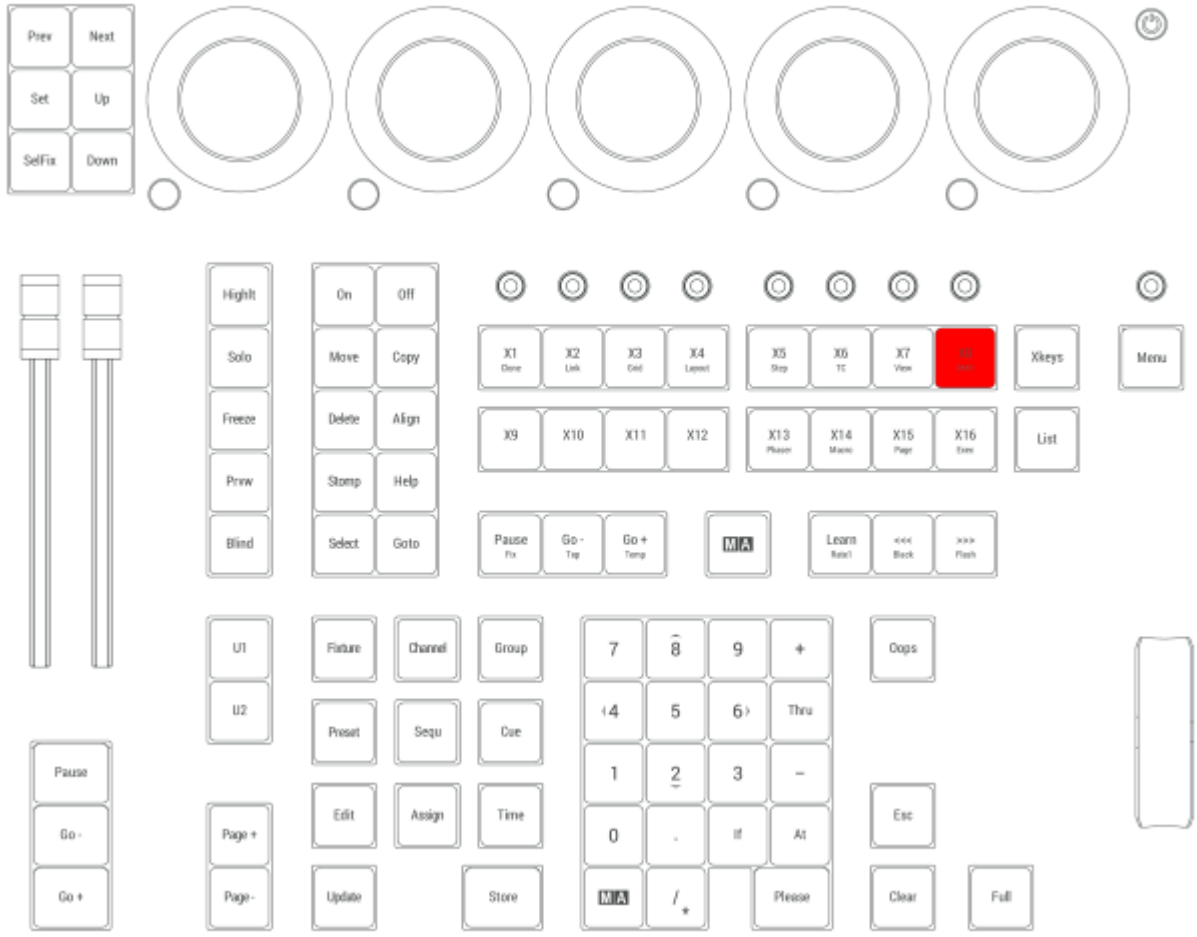
```
MA  User name[Fixtures]> DMXUniverse
```

For more information about DMXUniverse, see the **DMXUniverse keyword** topic.

X8 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

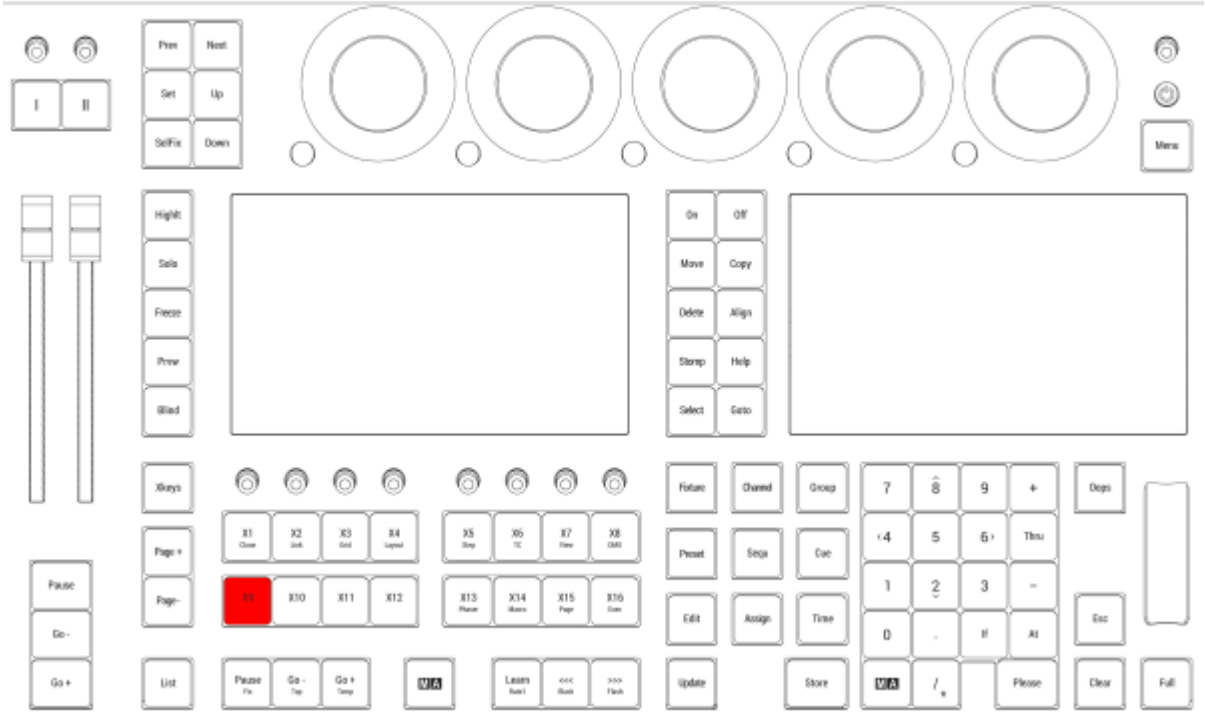


Location on grandMA3 compact consoles and grandMA3 onPC command wing

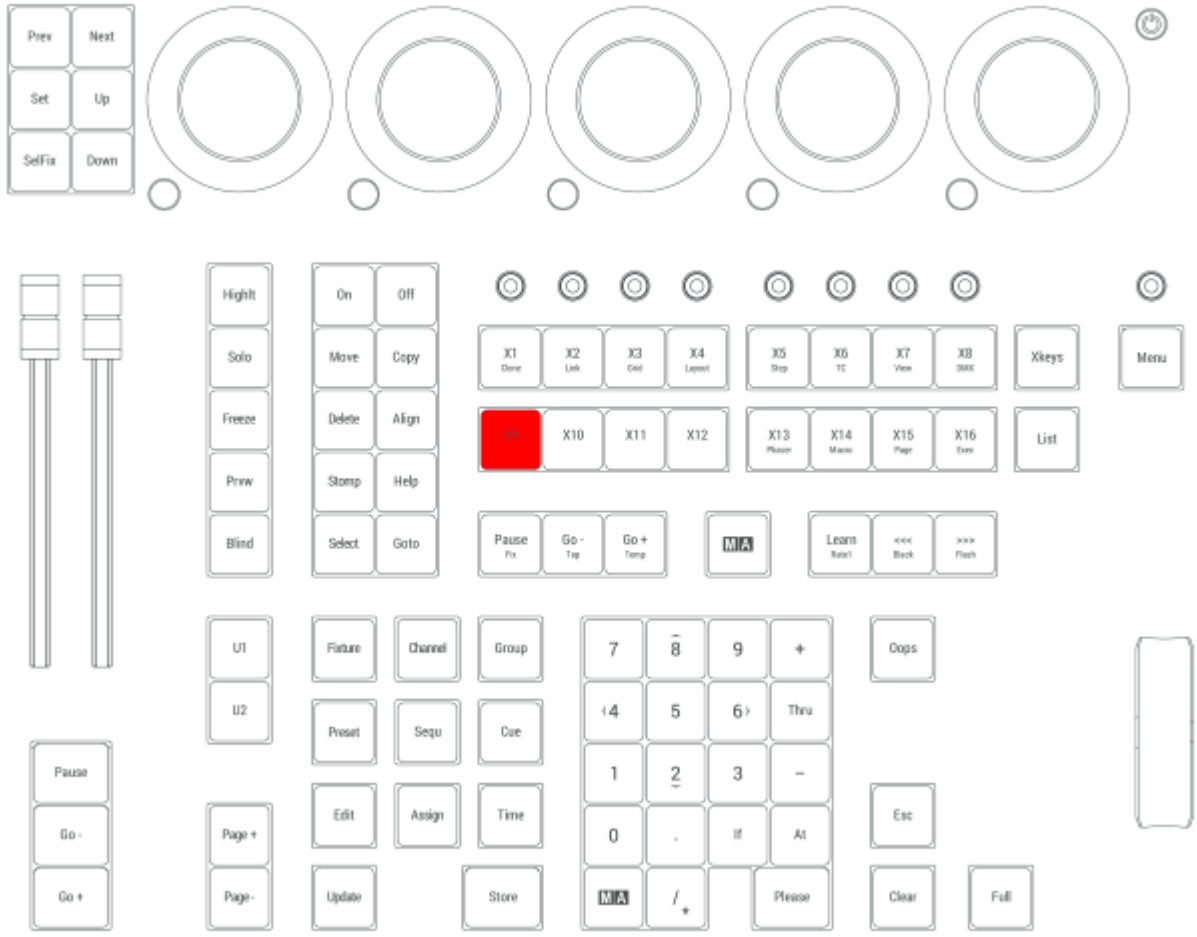
2.15.72. X9 key

At the moment all Xkeys behave like executors. X9 is executor 191.

X9 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

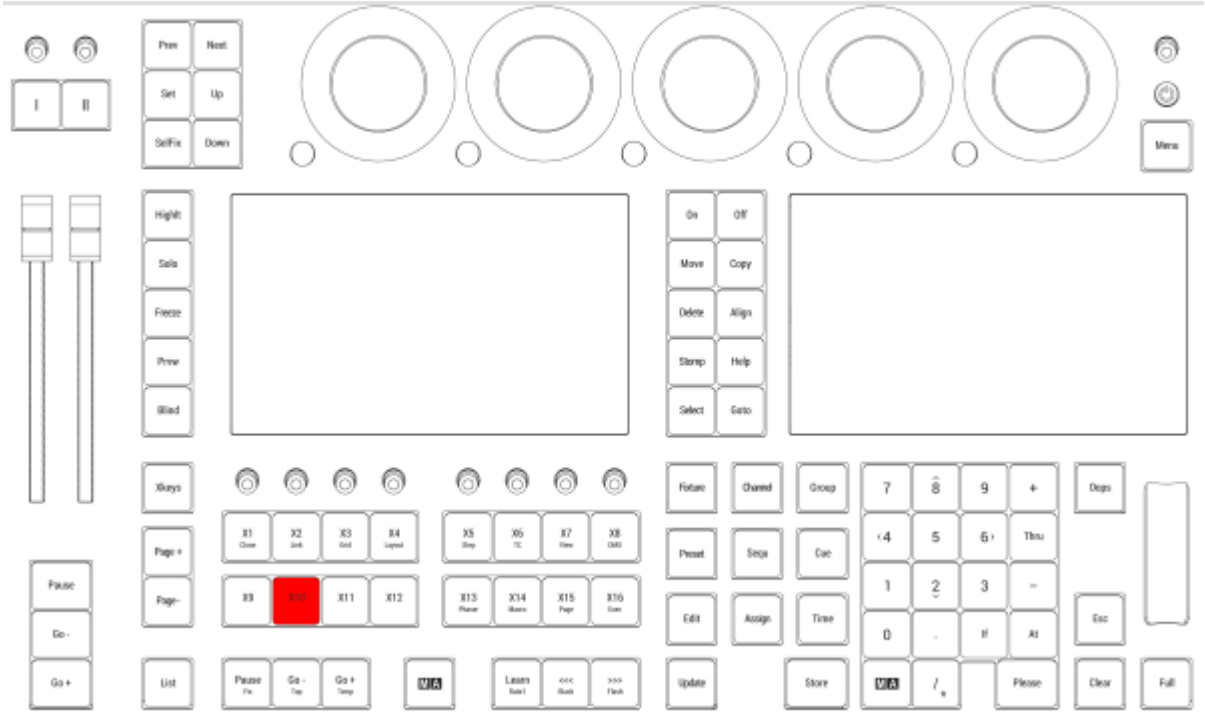


Location on grandMA3 compact consoles and grandMA3 onPC command wing

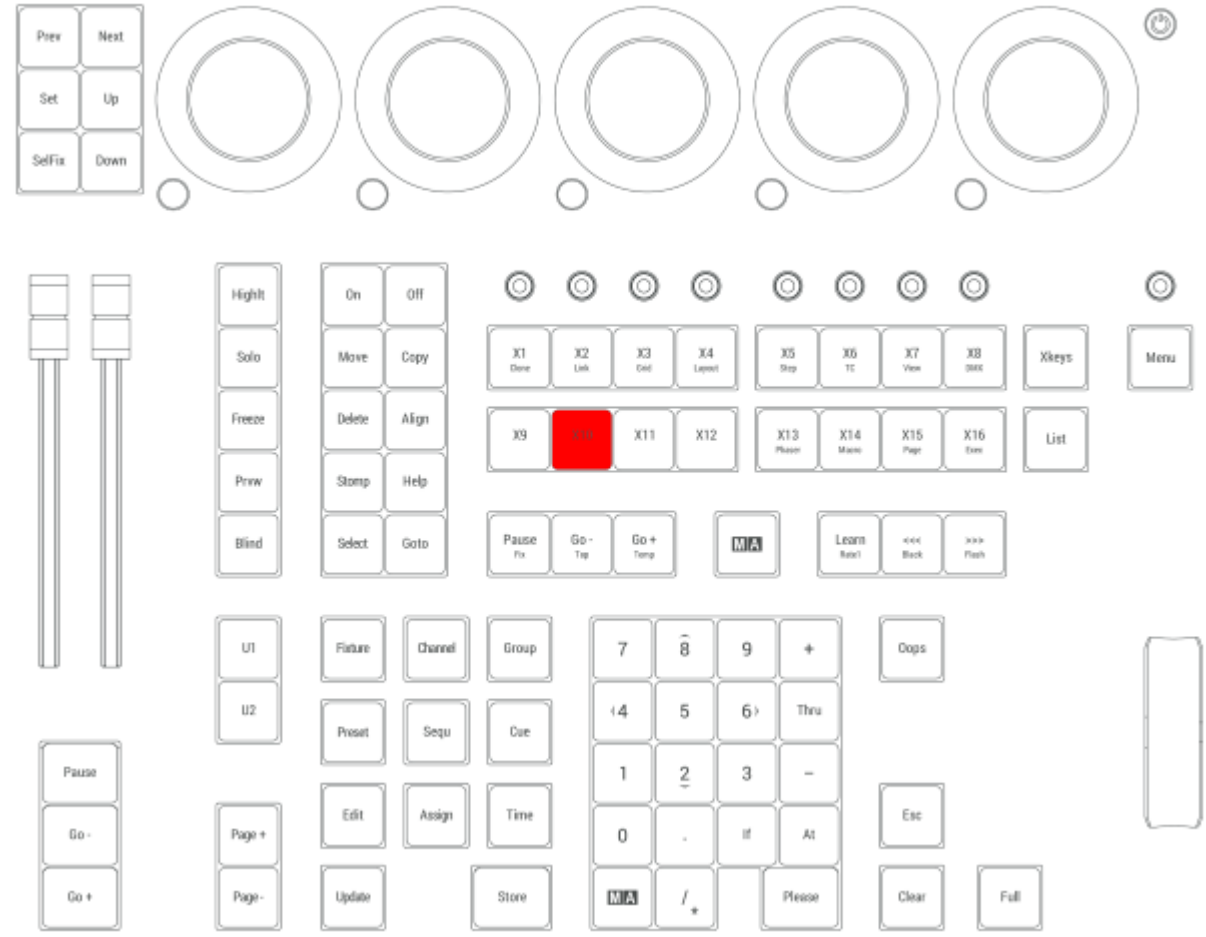
2.15.73. X10 key

At the moment all Xkeys behave like executors. X10 is executor 192.

X10 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

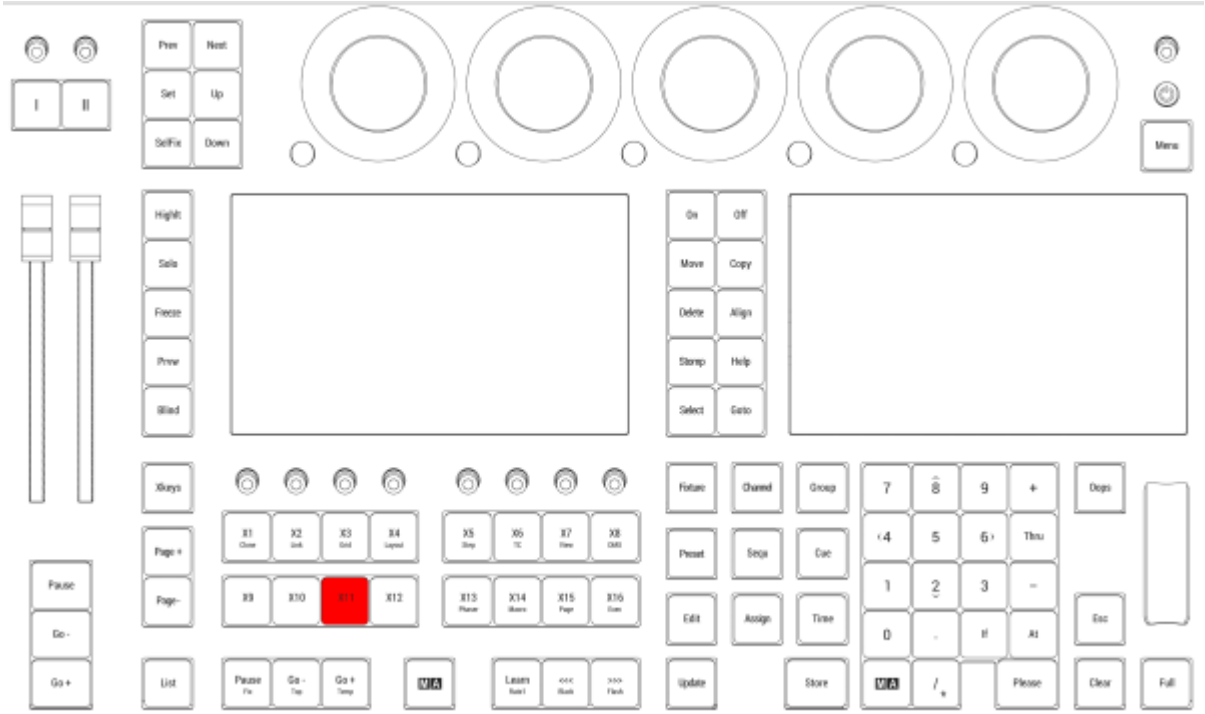


Location on grandMA3 compact consoles and grandMA3 onPC command wing

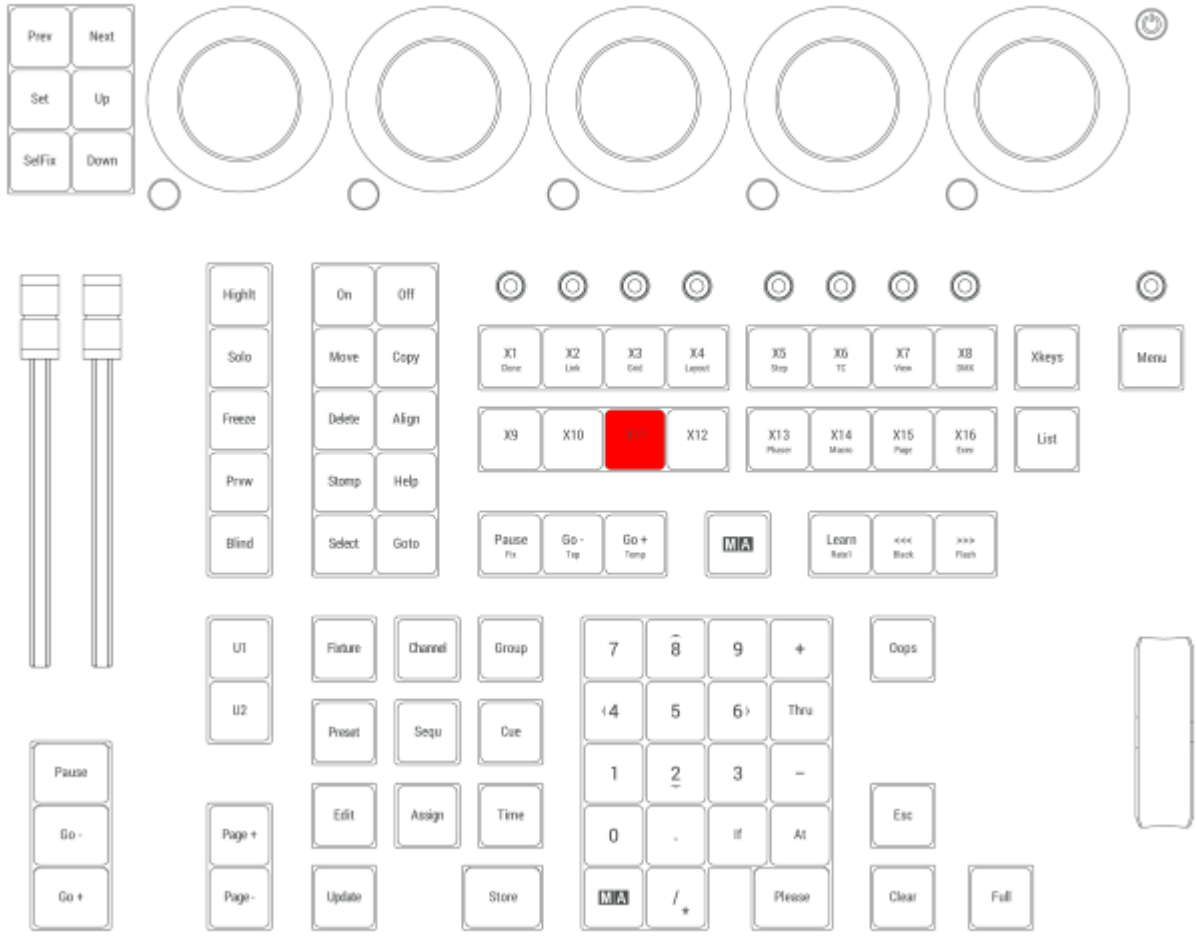
2.15.74. X11 key

At the moment all Xkeys behave like executors. X11 is executor 193.

X11 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

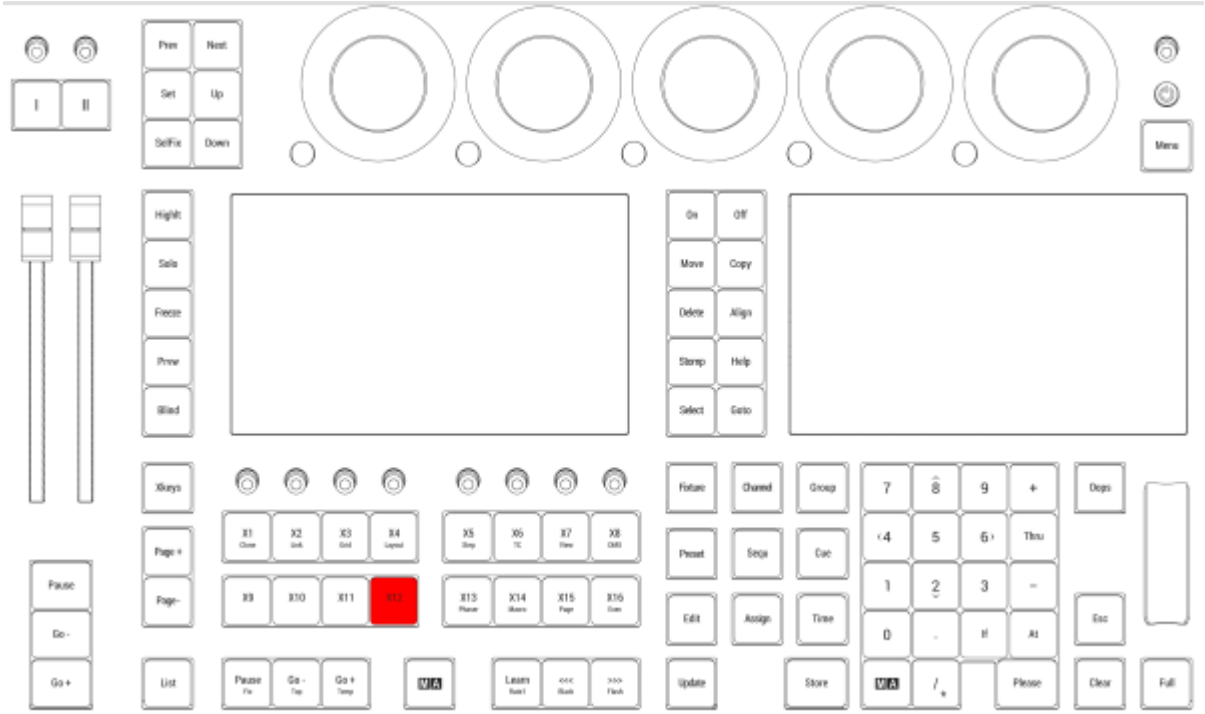


Location on grandMA3 compact consoles and grandMA3 onPC command wing

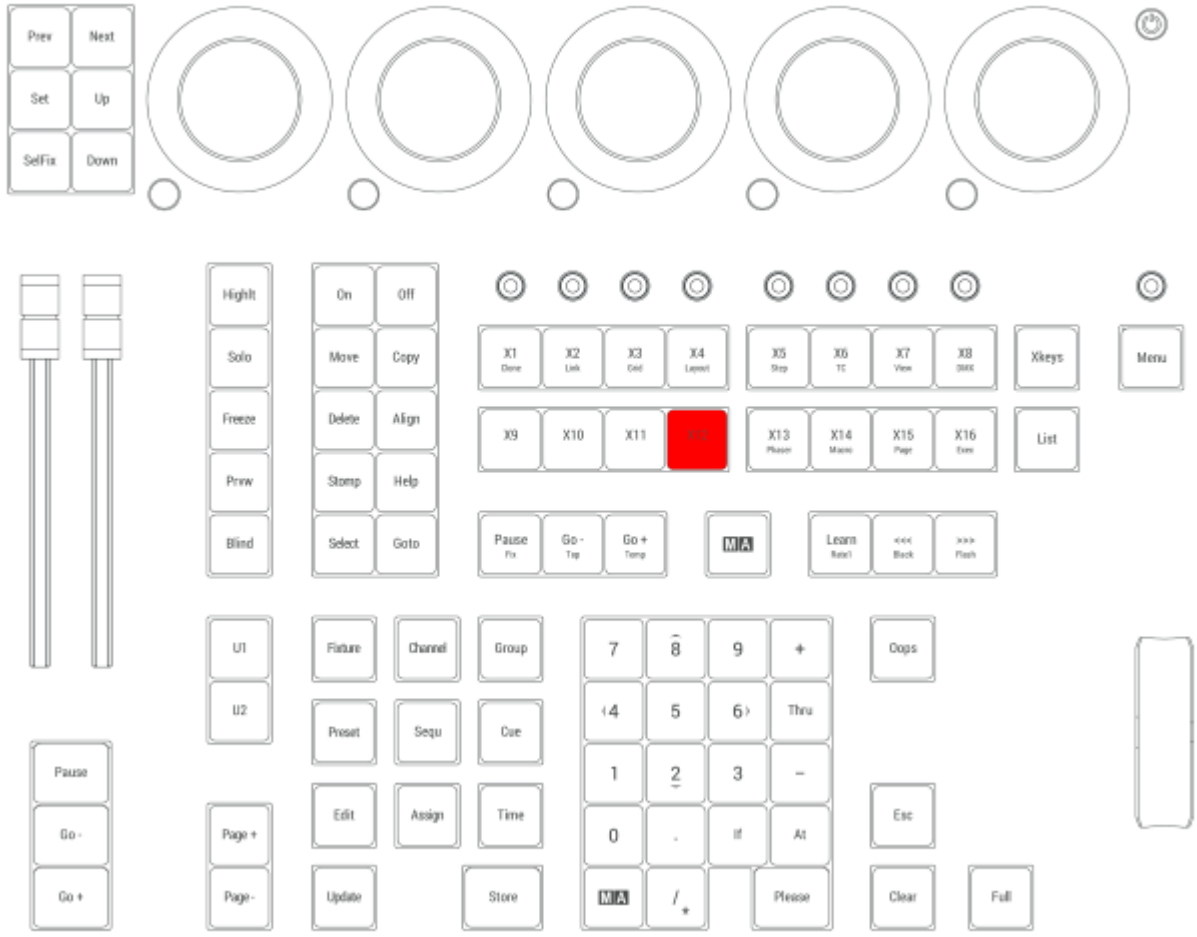
2.15.75. X12 key

At the moment all Xkeys behave like executors. X12 is executor 194.

X12 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.76. X13 | Phaser key

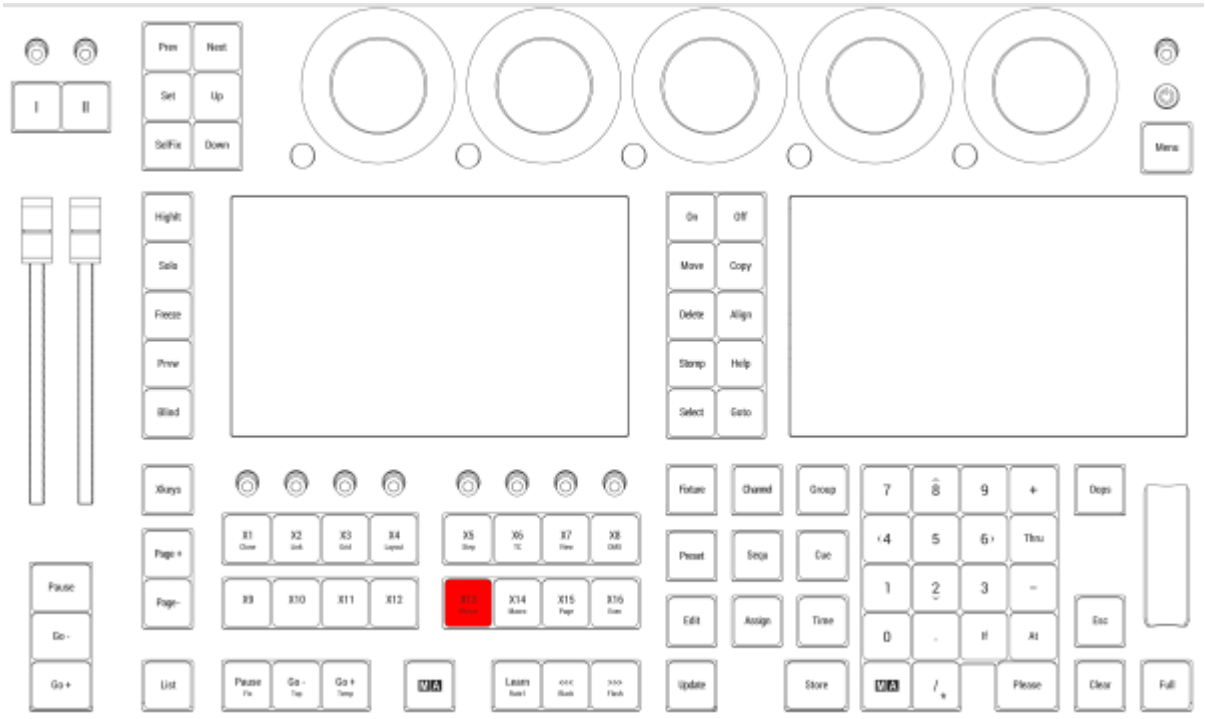
At the moment all Xkeys behave like executors. X13 is executor 195.

Phaser

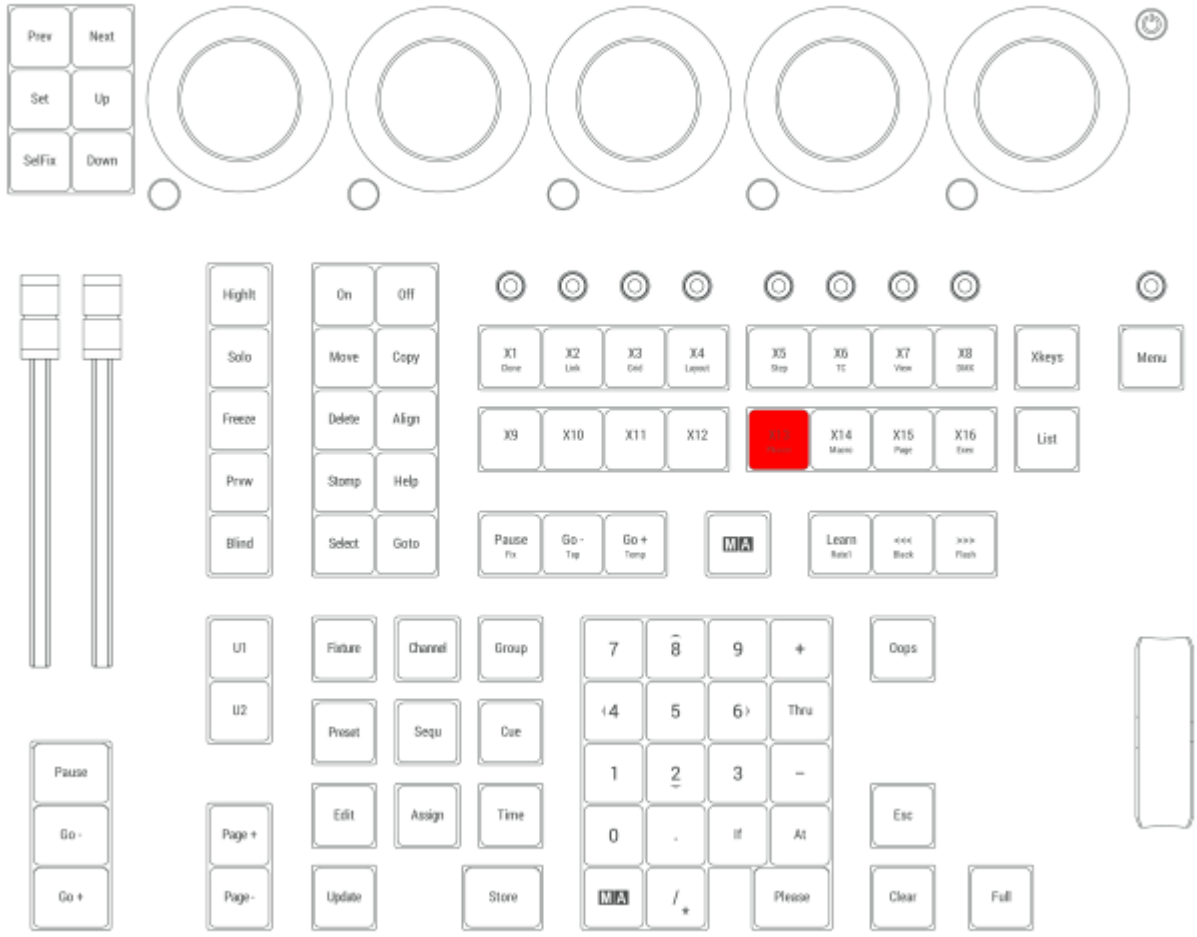
Pressing and holding MA + X13 opens the **Phaser Editor**.

For more information about Phasers, see the topics **Phasers** and **Phaser Editor**.

X13 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.77. X14 | Macro key

At the moment all Xkeys behave like executors. X14 is executor 196.

Macro

Pressing and holding MA + X14 enters the Macro keyword into the command line.

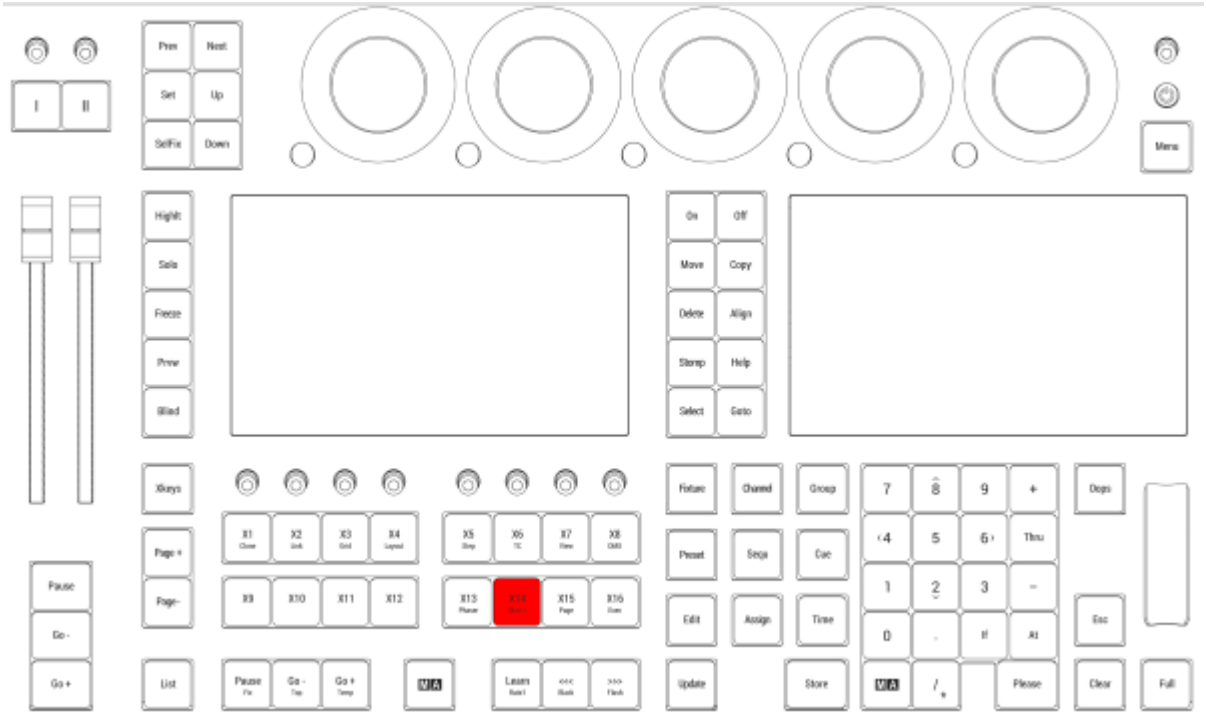
For more information about Macro, see the **Macro keyword** topic.

Plugin

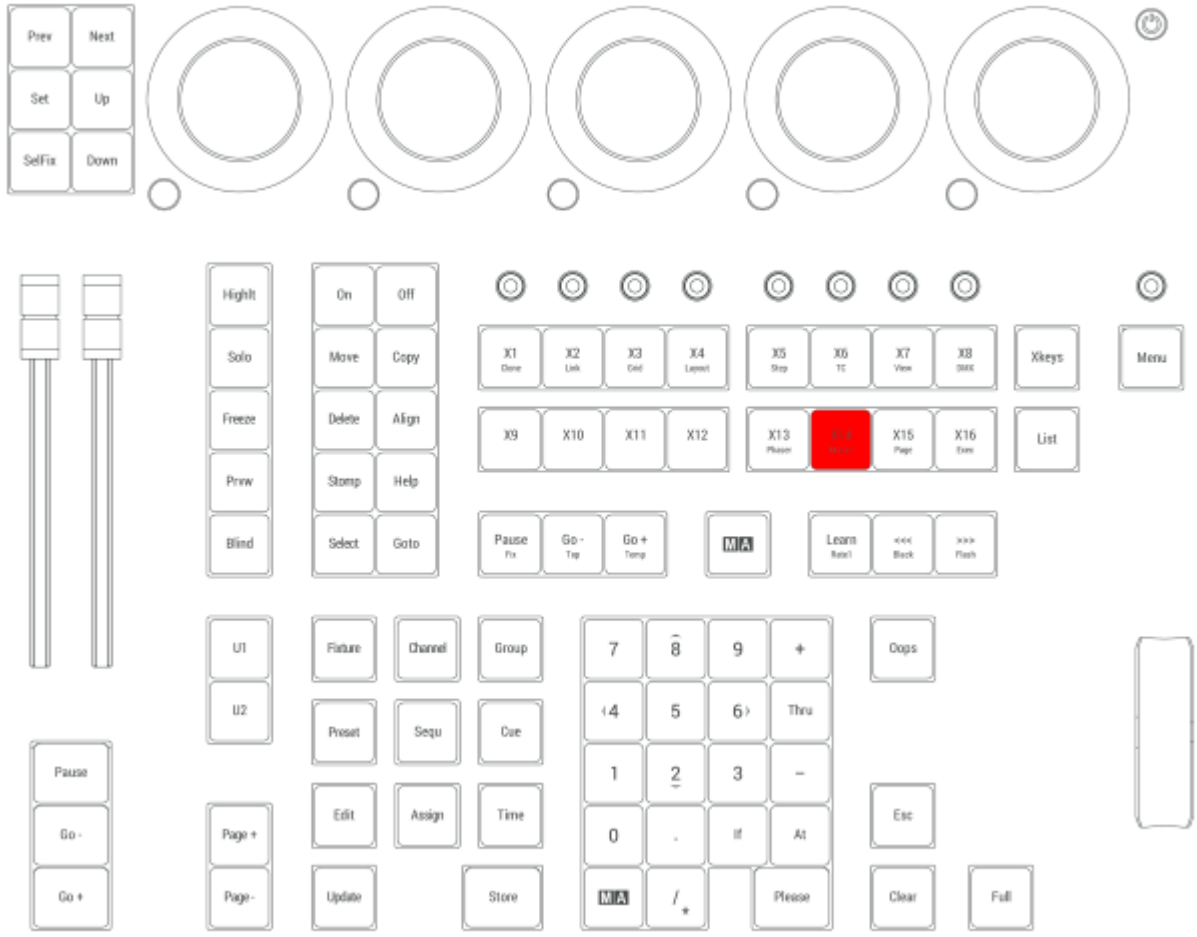
Pressing and holding MA + X14 + X14 enters the Plugin keyword into the command line.

For more information about Plugin, see the **Plugin keyword** topic.

X14 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.78. X15 | Page key

At the moment all Xkeys behave like executors. X15 is executor 197.

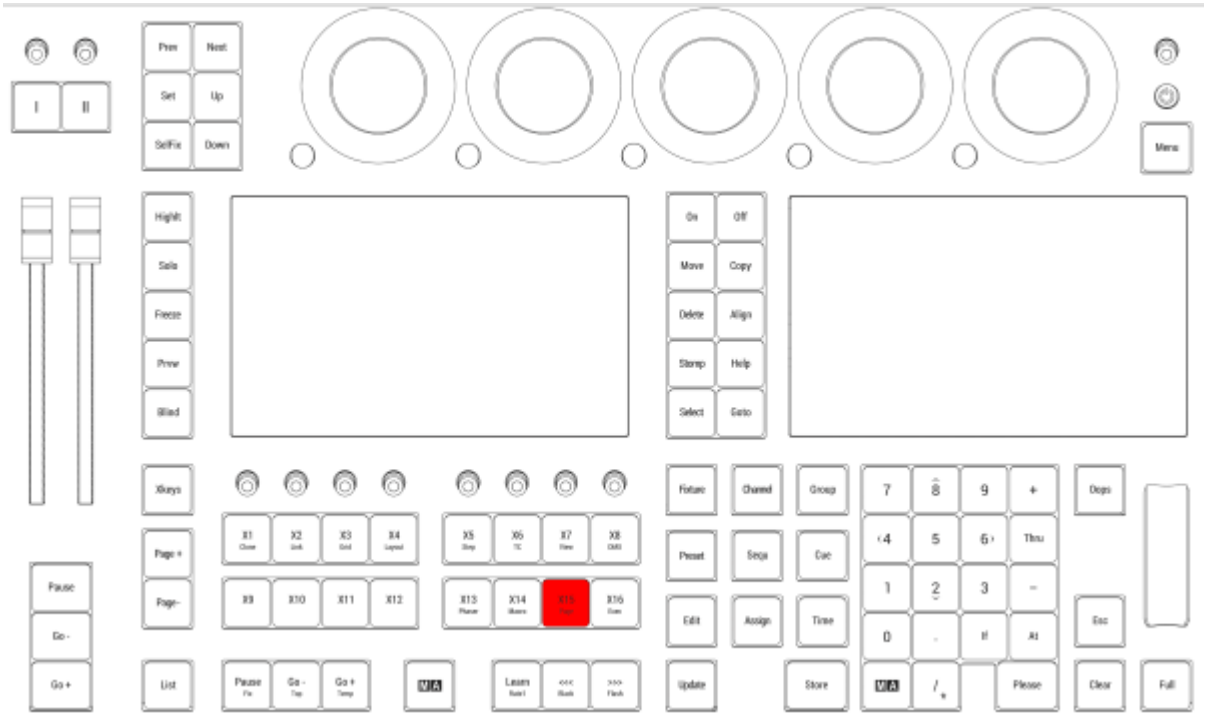
Page

Pressing and holding MA + X15 enters the Page keyword into the command line.

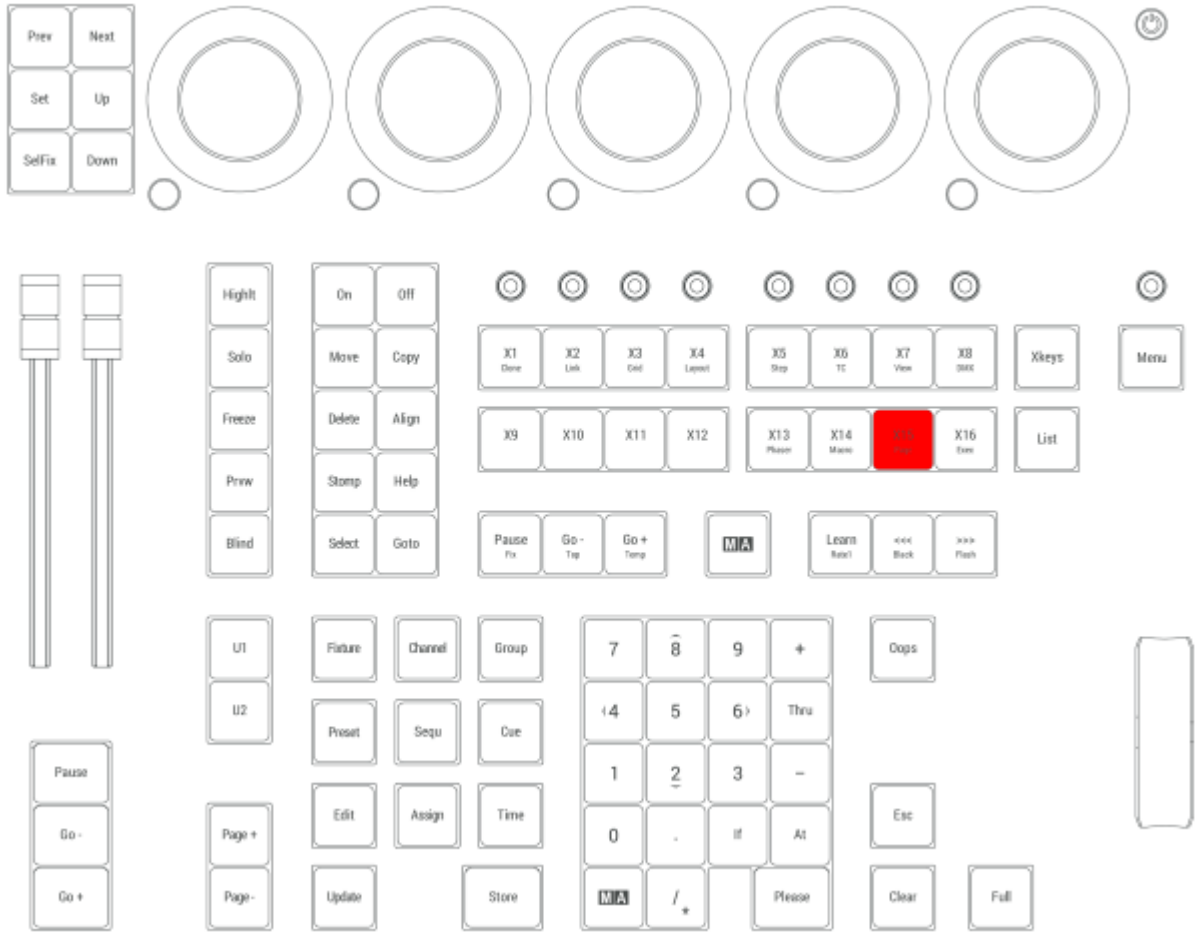
```
User name[Fixtures]> Page
```

For more information about Page, see the **Page keyword** topic.

X15 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.15.79. X16 | Exec key

At the moment all Xkeys behave like executors. X16 is executor 198.

Executor

Pressing and holding MA + X16 enters the Executor keyword into the command line.

For more information about Executor, see the **Executor keyword** topic.

SpecialExecutor

Pressing and holding MA + X16 + X16 enters the SpecialExecutor keyword into the command line.

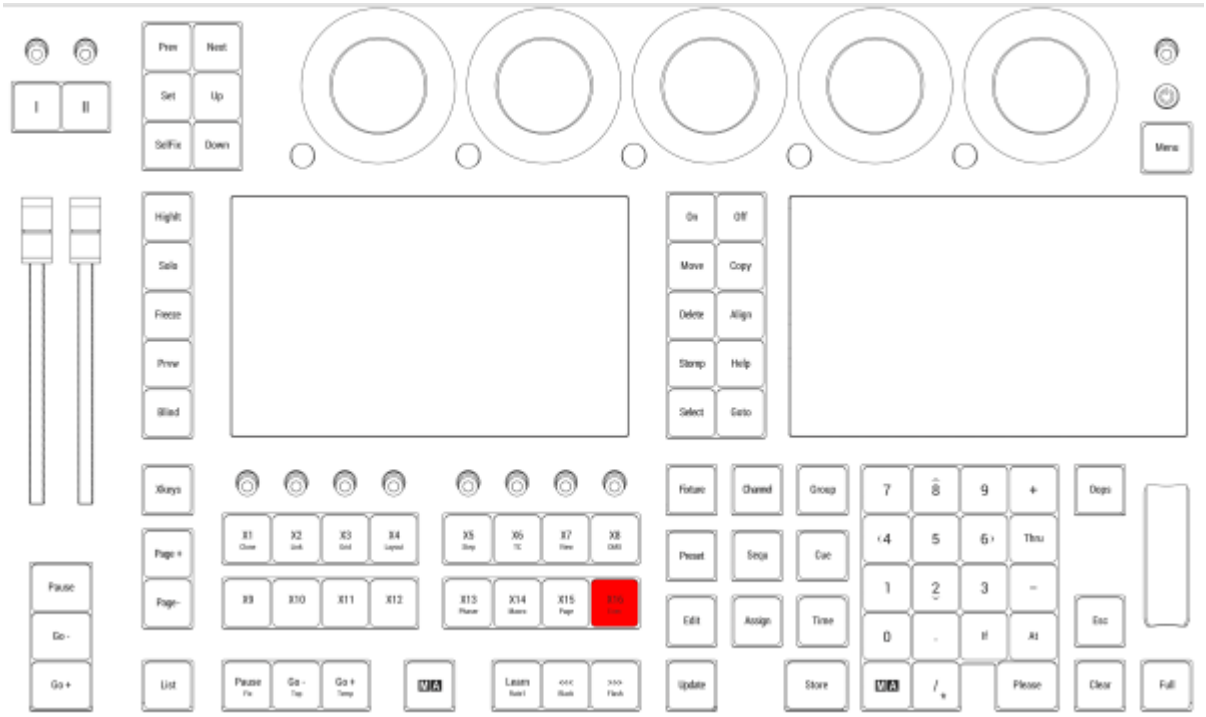
For more information about SpecialExecutor, see the **SpecialExecutor keyword** topic.

FaderMaster

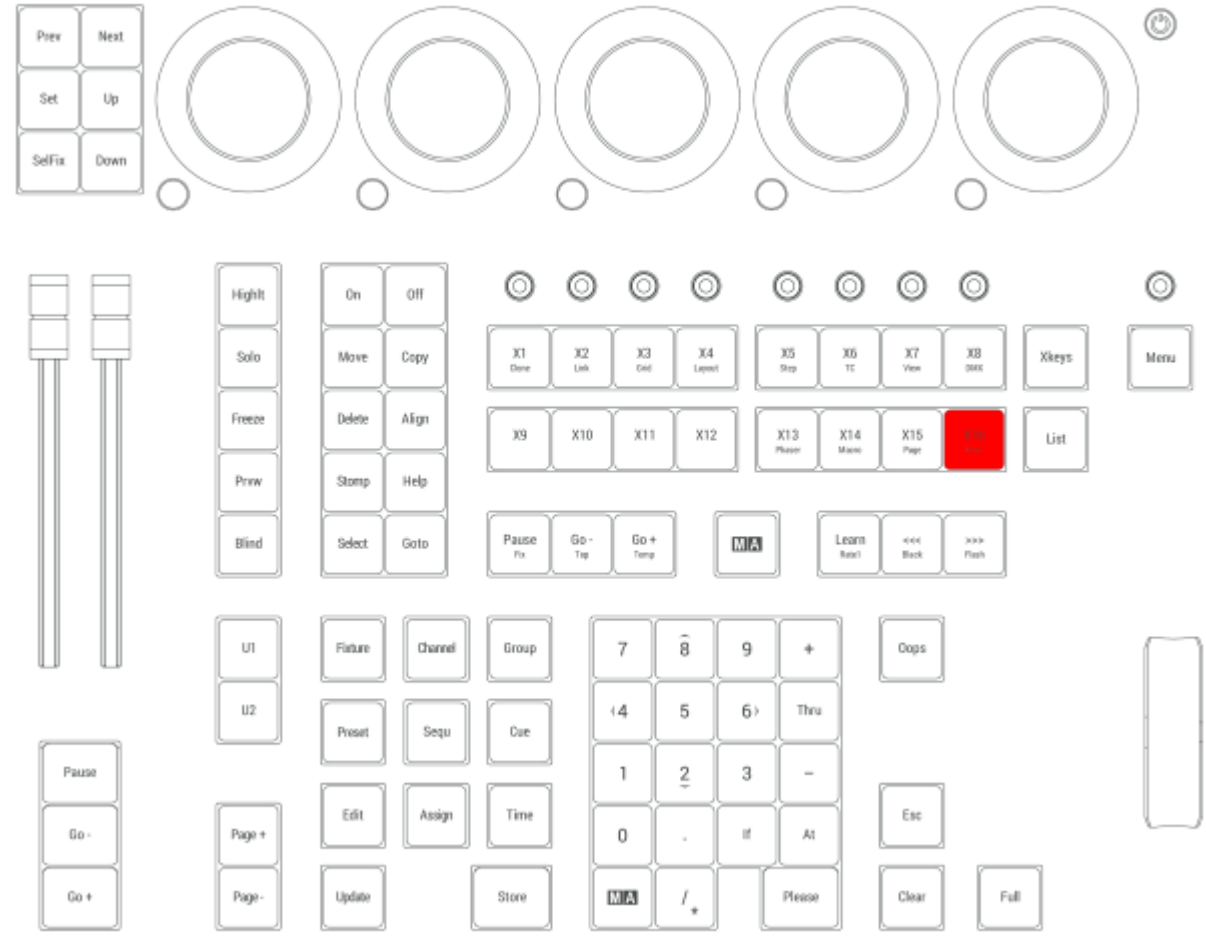
Pressing and holding MA + X16 + X16 + X16 enters the FaderMaster keyword into the command line.

For more information about FaderMaster, see the **FaderMaster keyword** topic.

X16 is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

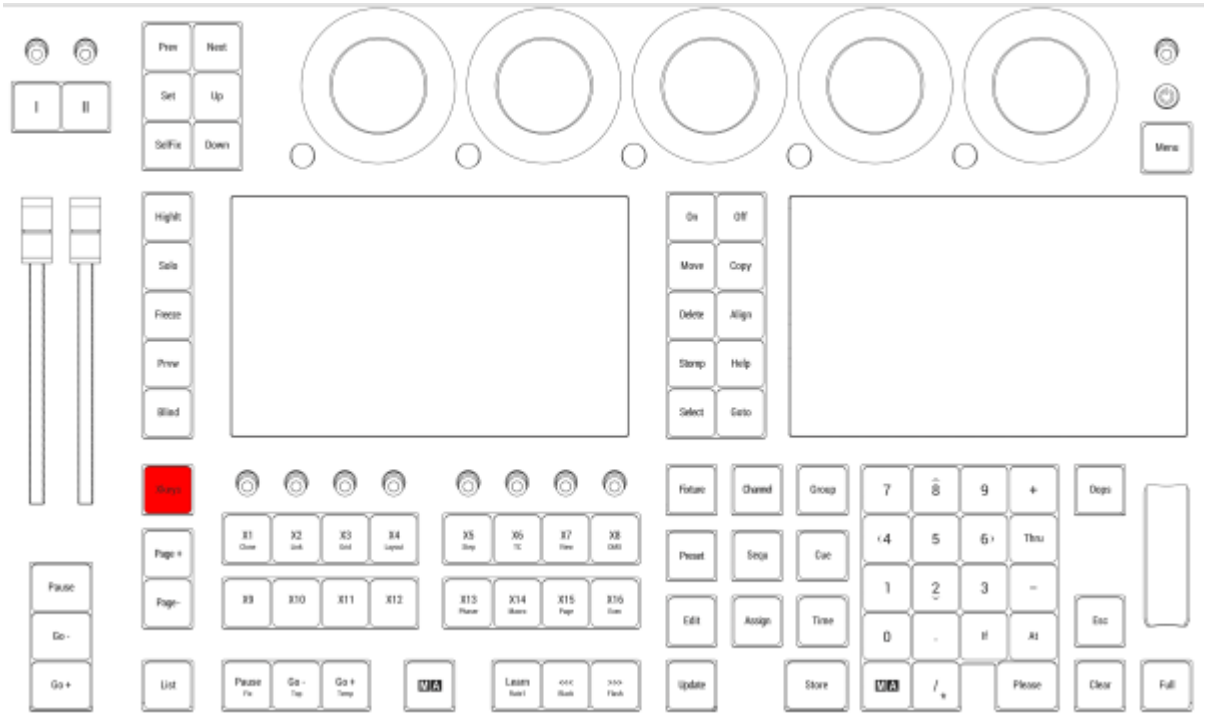


Location on grandMA3 compact consoles and grandMA3 onPC command wing

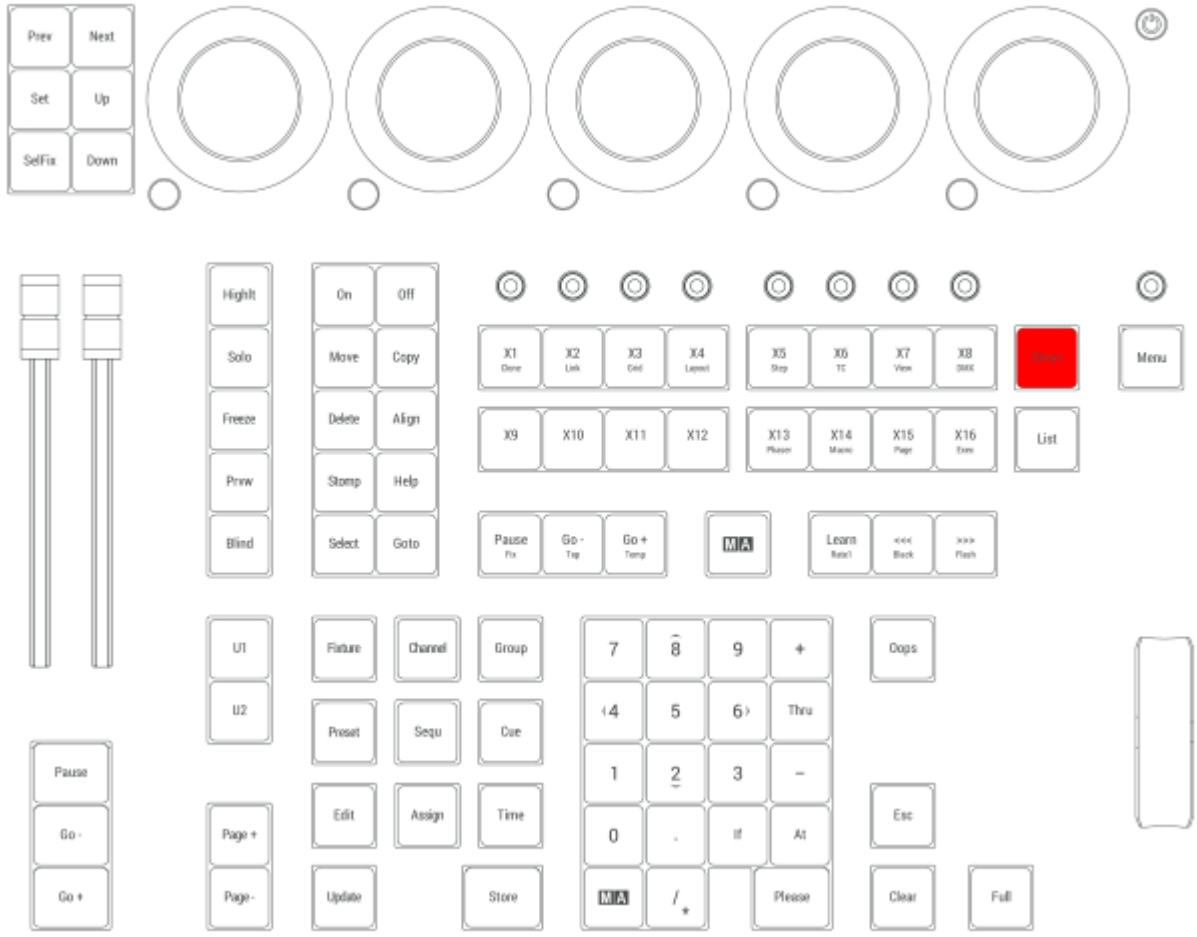
2.15.80. Xkeys key

Pressing Xkeys executes the Xkeys keyword into the command line.

Xkeys is located in the command section.



Location on grandMA3 full-size and grandMA3 light consoles

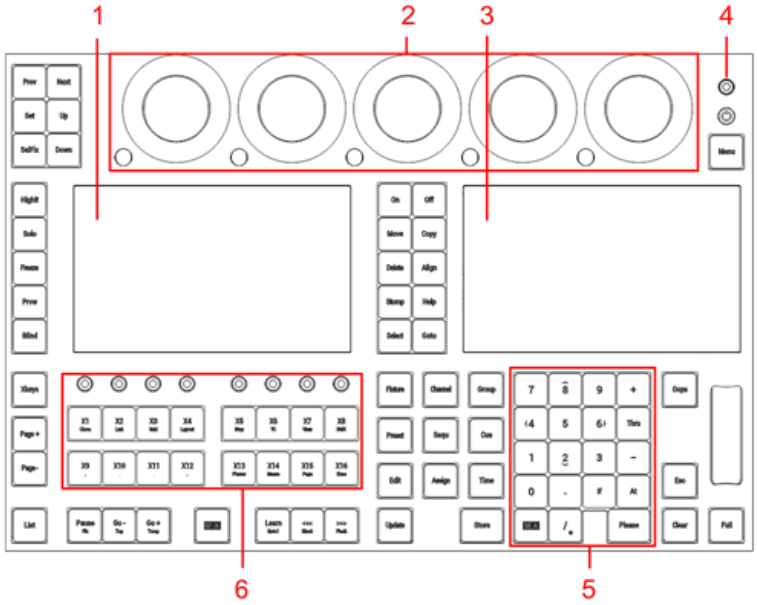


Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.16. Control Elements

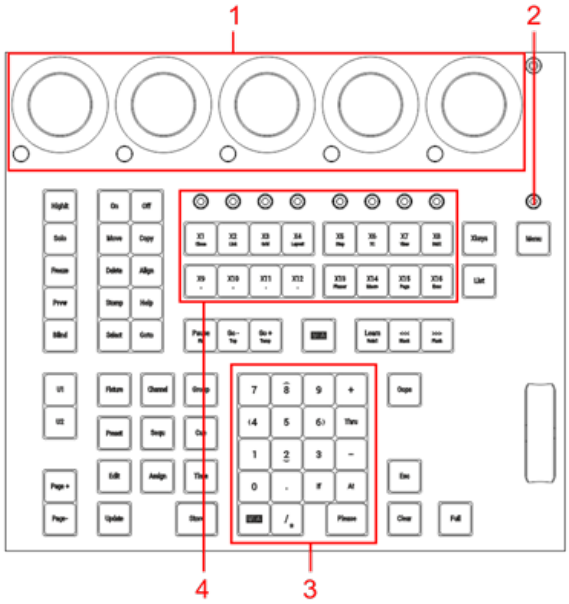
The following chapters describe the different control elements of the grandMA3 devices.

2.16.1. Command area



Command area of grandMA3 full-size (CRV) and grandMA3 light (CRV)

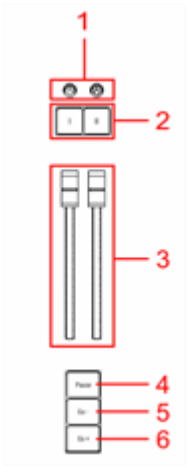
1. Left command screen
2. **Dual encoder section**
3. Right command screen
4. **Grand master**
5. Number pad
6. **Xkeys section**



Command area of grandMA3 compact (XT) and grandMA3 onPC command wing (XT)

1. **Dual encoder section**
2. **Grand master**
3. Number pad
4. **Xkeys section**

2.16.2. Master area



Master area

1. Master knobs 1+2
2. Master keys 1+2
3. Master faders 1+2
4. Pause key
5. Go- key
6. Go+ key

Go+, Go- and Pause

The Go+ [Large] key, the Go- [Large] key, and the Pause [Large] key are located in the lower part of the master area. These keys execute the Go+, Go- or Pause command for the selected sequence.

For more information about Go+, Go- and Pause, please see the **Go+[Large] key**, **Go-[Large] key**, or the **Pause[Large] key** topic.

Master Fader 1 and Master Fader 2

The two faders in the master area are master fader 1 and master fader 2.

By default, master fader 1 is set to the master function for the selected sequence.

By default, master fader 2 is set to the crossfade function for the selected sequence.

To change the function of the faders, please read below.

Master Key 1 and Master Key 2, Master Knob 1 and Master Knob 2

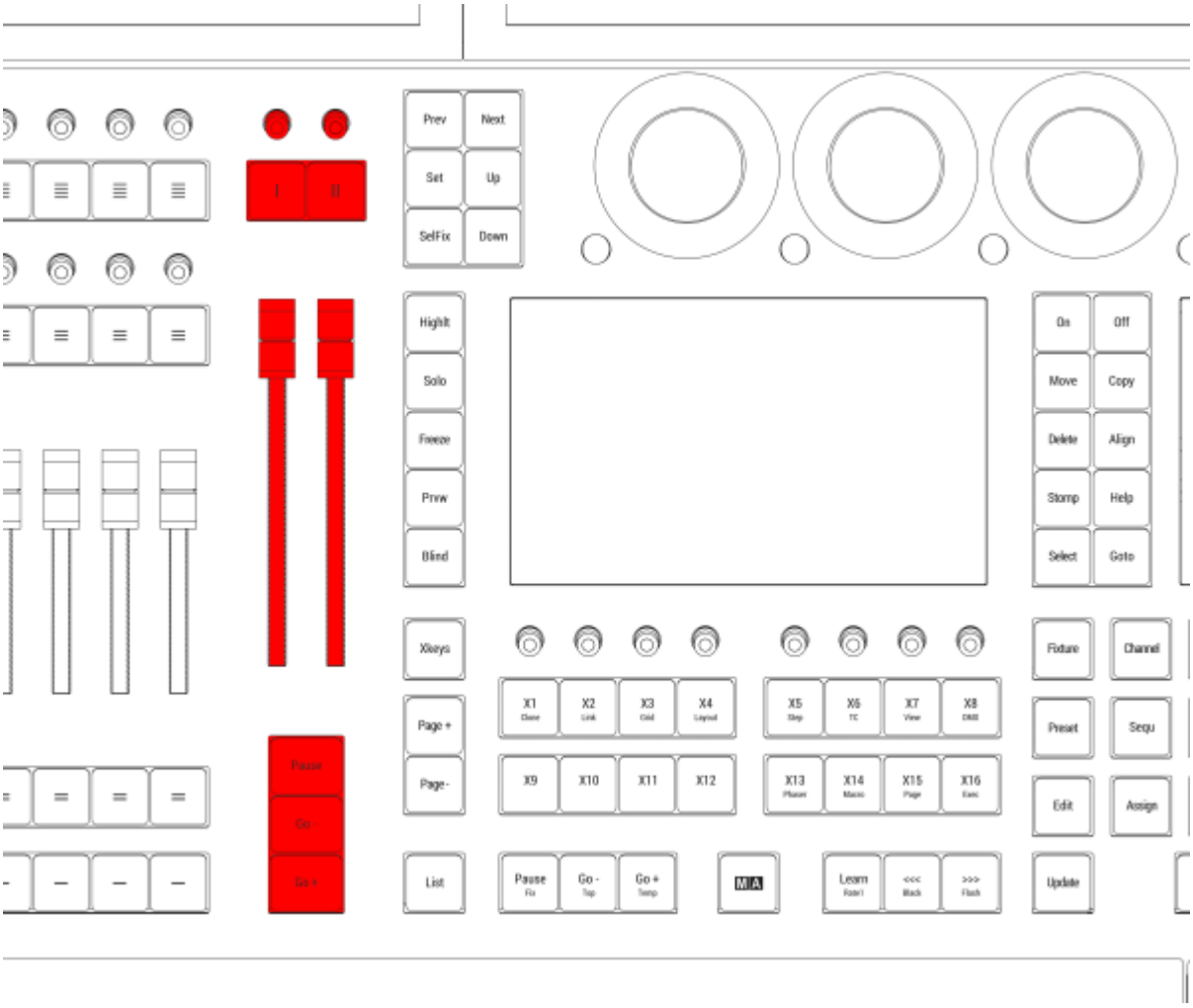
Restriction:
Master keys and master knobs are only available on grandMA3 full-size (CRV) and grandMA3 light (CRV) consoles.

The master keys 1+2 and the master knobs 1+2 are located in the upper part of the master area.

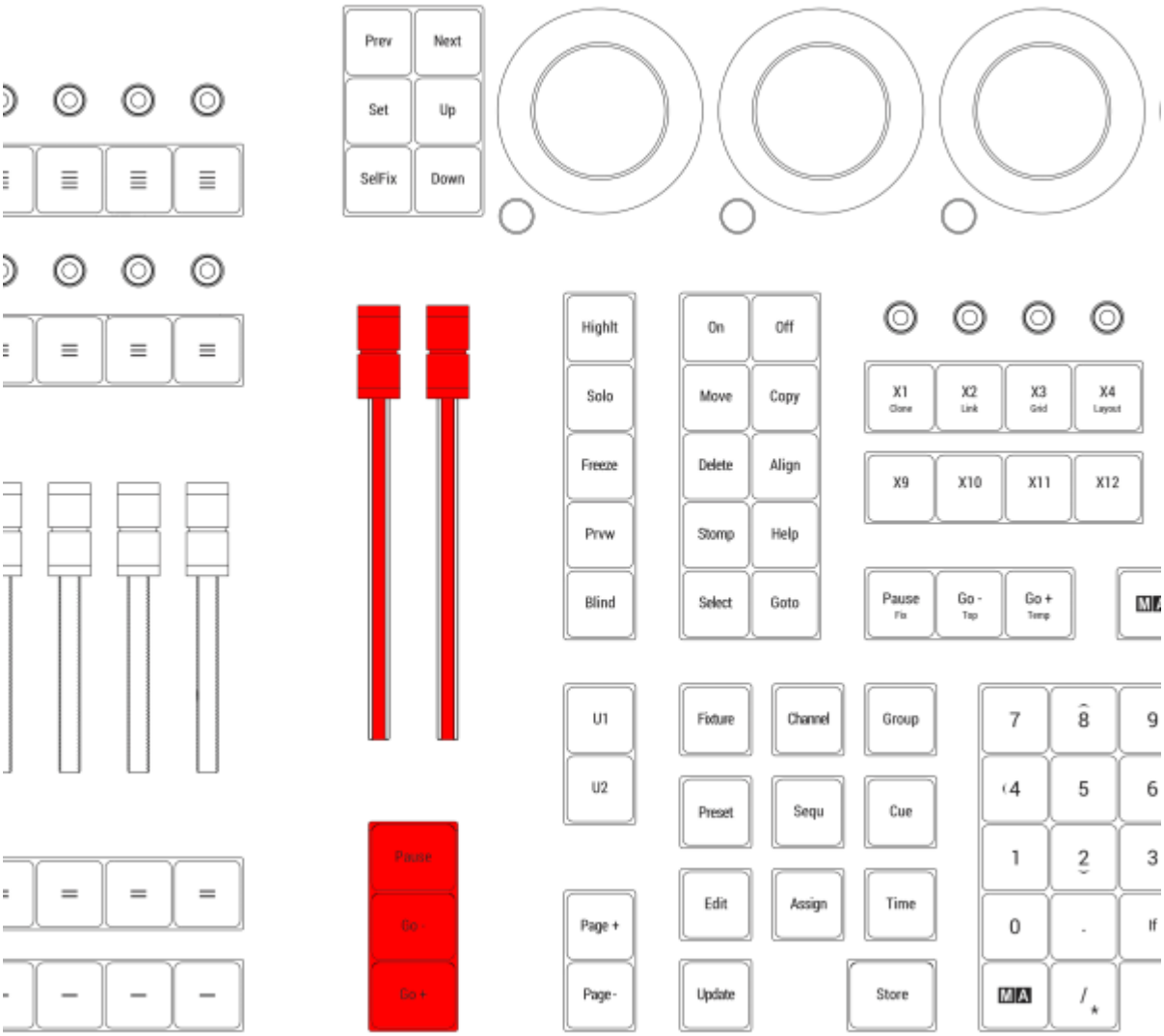
By default, these are set to toggle Highlight and Solo on and off and set the intensity of these.

To change the function of these buttons and knobs, please read the topic **Special Executors**.

The master area is located between the **command area** and the **executor area**.



Location on grandMA3 full-size (CRV) and grandMA3 light (CRV) consoles

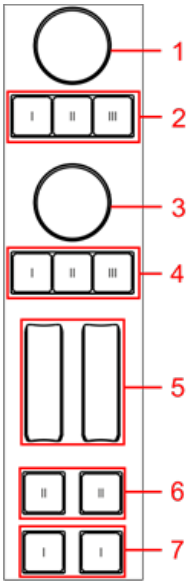


Location on grandMA3 compact consoles and grandMA3 onPC command wings

2.16.3. Custom area

Restriction:
The custom area is not available on some grandMA3 consoles.

Custom Area



Custom area

1. Upper encoder
2. Upper encoder keys 1-3
3. Lower encoder
4. Lower encoder keys 1-3
5. Wheels 1+2
6. Wheels 1+2 upper keys
7. Wheels 1+2 lower keys

To change the function of these control elements, please read the topic **Special Executors**.

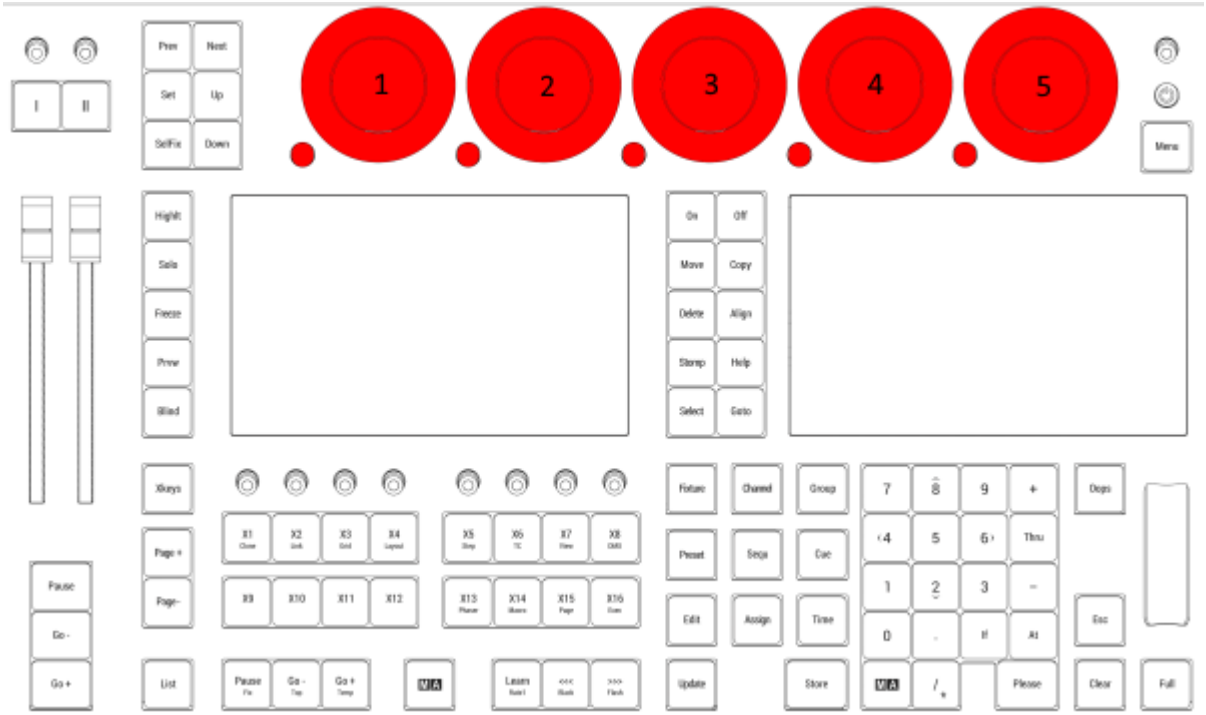
2.16.4. Dual encoders

The five dual encoders are used to adjust the different attributes of the fixtures. There are four different actions on the encoders:

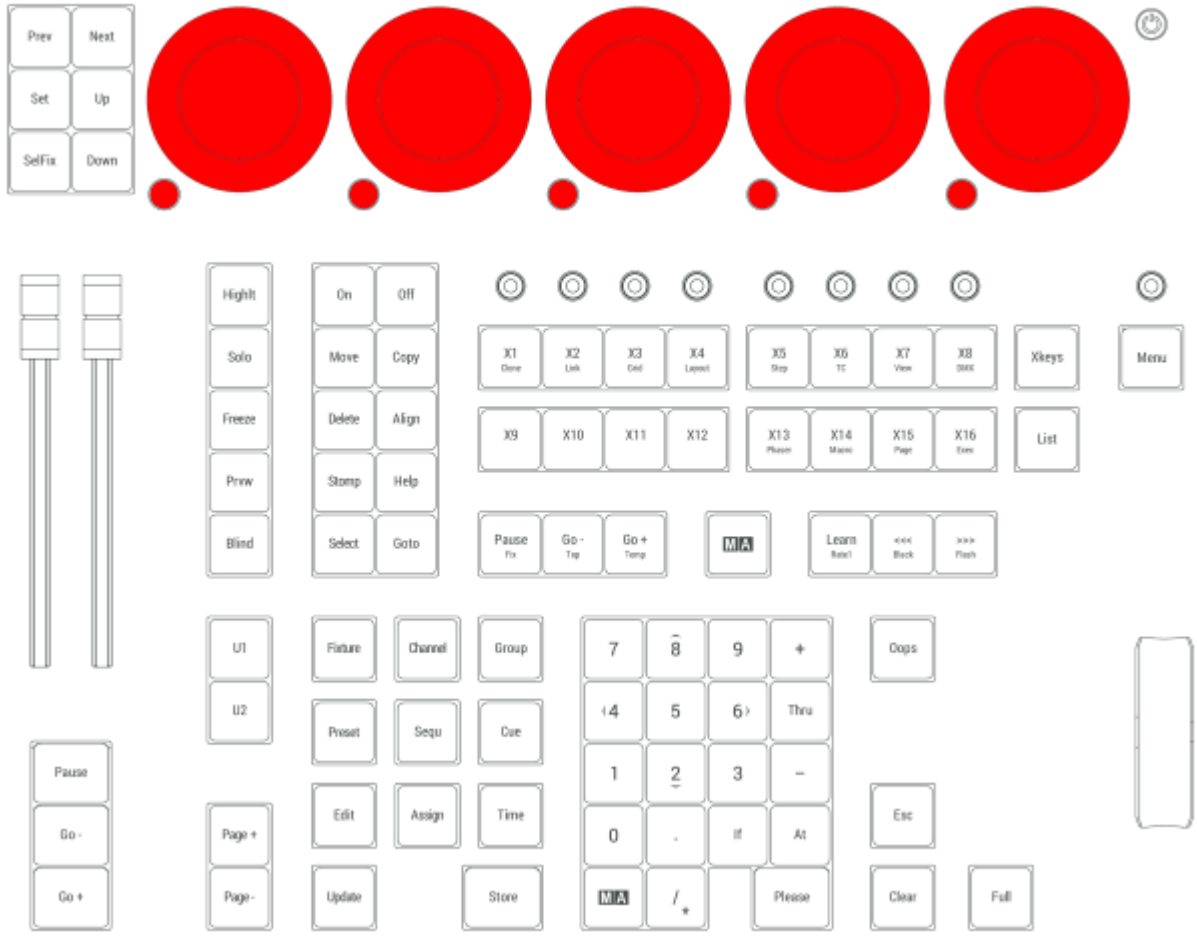
- Turn the inner ring to adjust the attribute "Coarse".
- Turn the outer ring to adjust the attribute "Fine".
- Press the inner ring to open a calculator to type in the value.
- Press and turn the inner ring to adjust the attribute "Coarse" with a higher speed.
- Press the dual encoder key to open a calculator to type in the value. The dual encoder key replaces the press function of the outer ring.

Hint:
If Dimmer is the selected attribute, the inner ring adjusts with 1 percent per click and the outer ring adjusts with 0.01 percent per click.

These encoders are placed above the command section.



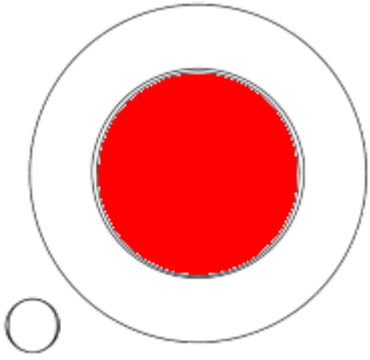
Location on grandMA3 full-size and grandMA3 light consoles



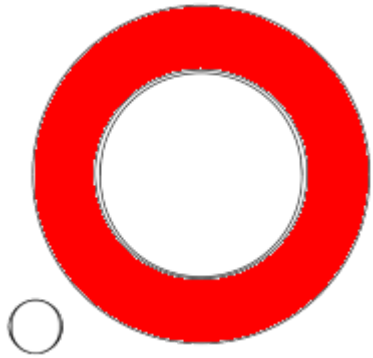
Location on grandMA3 compact consoles and grandMA3 onPC command wing and command wing XT

Each of the dual encoders is numbered 1 to 5 from left to right. There are three elements in each encoder:

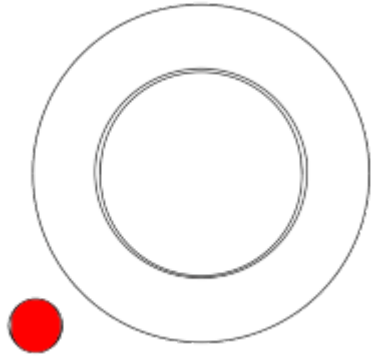
The inner ring



The outer ring



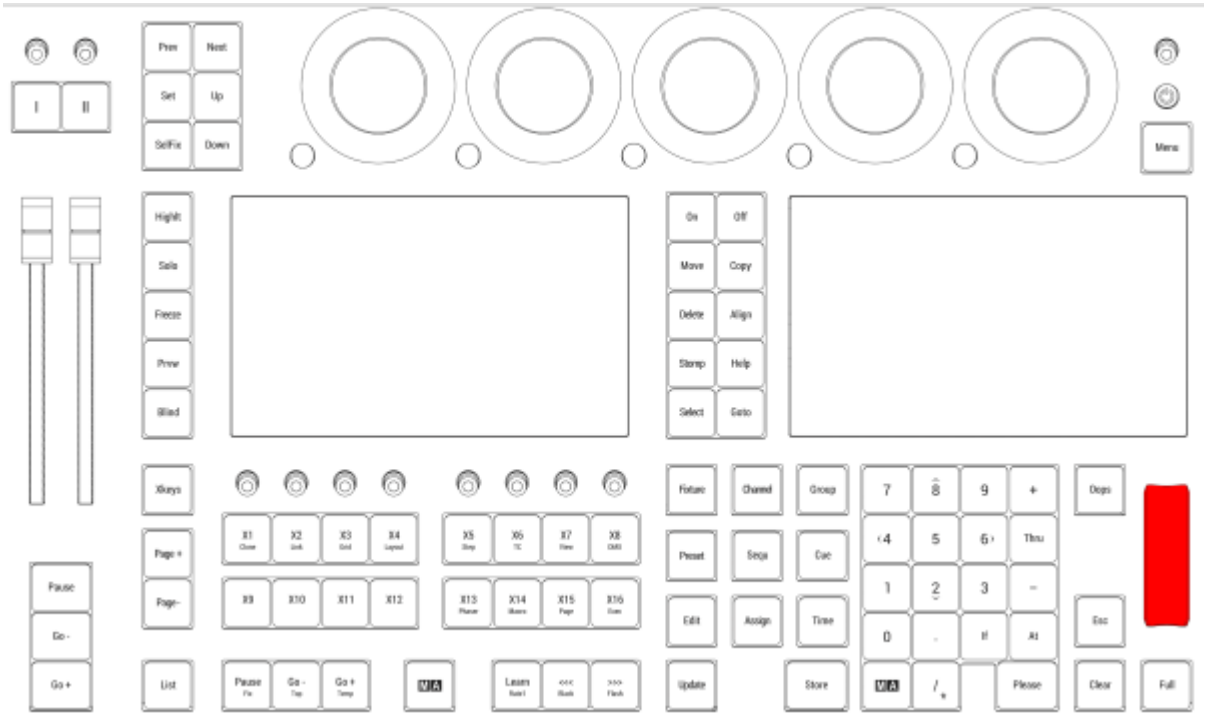
The dual encoder key



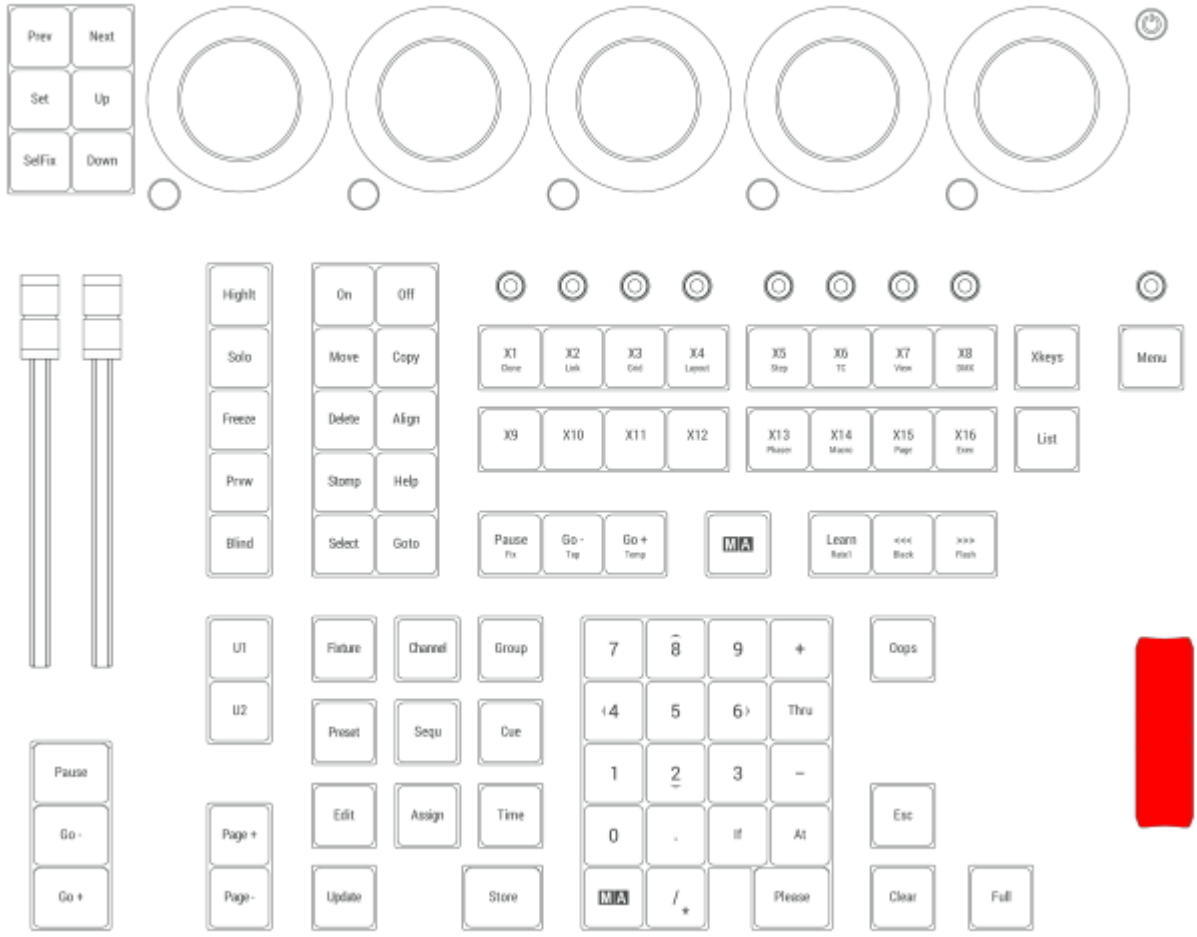
2.16.5. Level wheel

The level wheel is used to adjust the intensity of the selected fixtures.

The level wheel is placed on the right side of the numeric keys.



Location on grandMA3 full-size and grandMA3 light consoles



Location on grandMA3 compact consoles and grandMA3 onPC command wing/command wing XT

2.16.6. Grand Master

The Grand Master is used to limit the output of the intensity of all the fixtures patched in the show.

By default, turning the knob adjusts intensity. This can be changed in the Special Executor Configuration menu. To access the Special Executor Configuration menu press Assign and the Grand Master knob. This menu can also be opened using the SpecialExecutor keyword in the command line. For more information about SpecialExecutor, see the **SpecialExecutor keyword** topic.

To read more about Executor Configuration, see the **Executor Configuration** topic.

The Grand Master is also displayed on the right side of the encoder bar on screen 1:

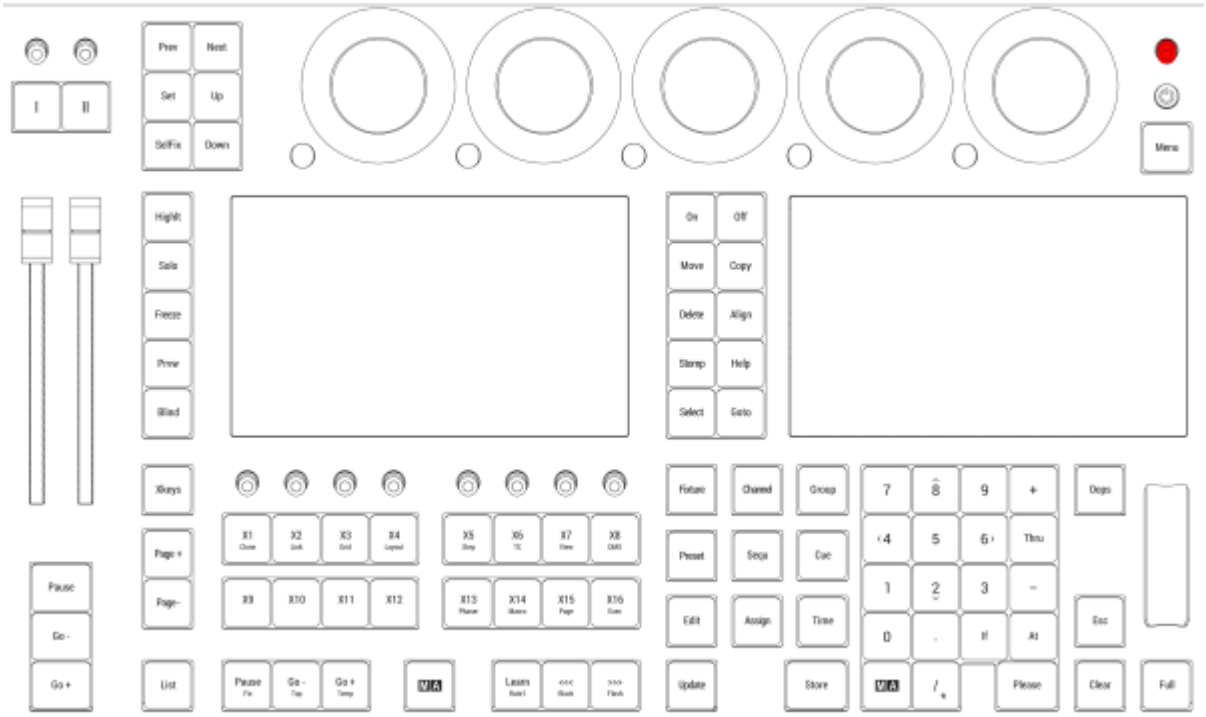
Grand Master in the encoder bar

The Grand Master level can be set by the fader and the Grand Master function can be enabled or disabled by tapping Grand Master. A green marker indicates that the grand master is enabled.

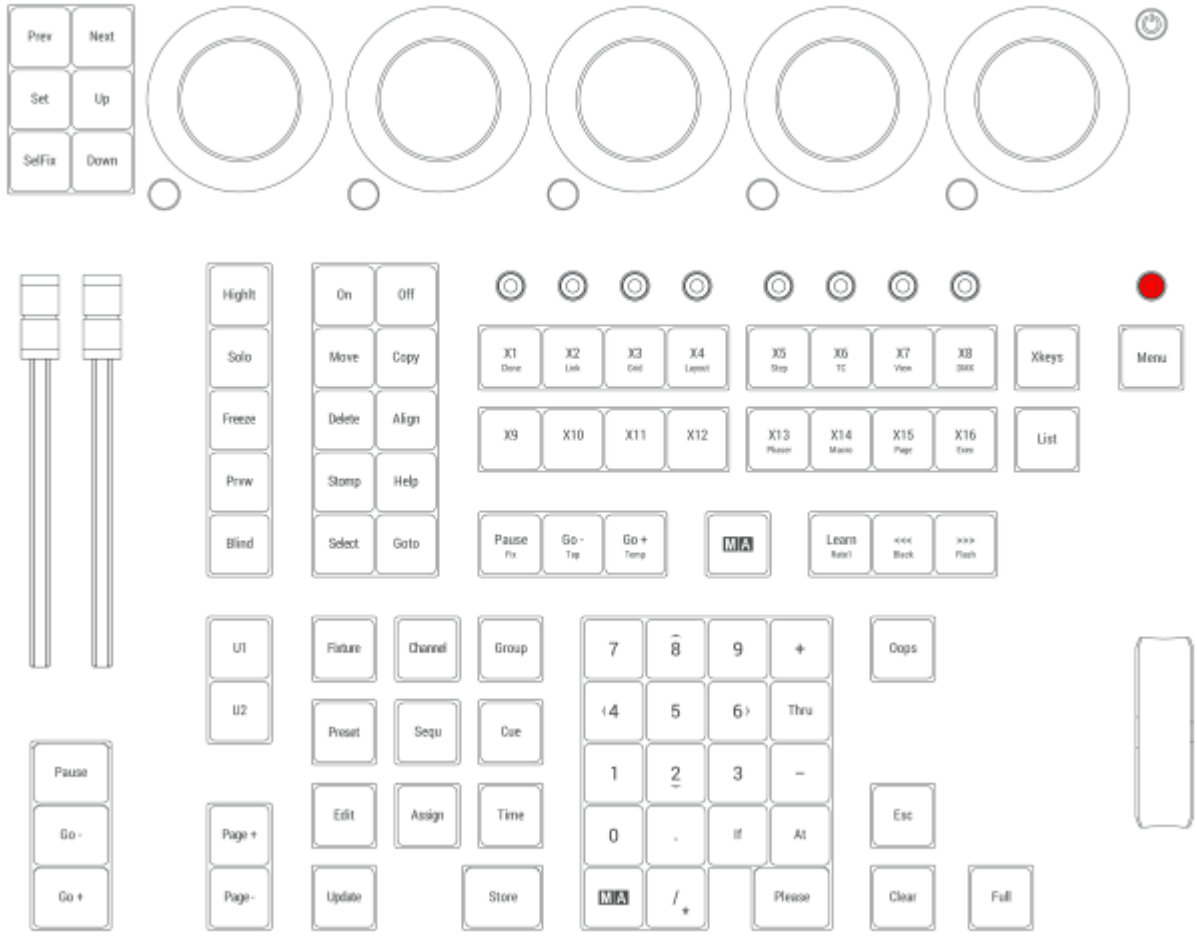
The Grand Master can also be assigned to any executor. See the **Assign object to an Executor** topic.

Hint:
The Output layer and the DMX layer in both the **Fixture sheet** and the **Layout view** as well as the **DMX sheet** and the **3D window** all display values as adjusted by the Grand Master.

The Grand master is located on the right side of the console.



Location on grandMA3 full-size and grandMA3 light consoles



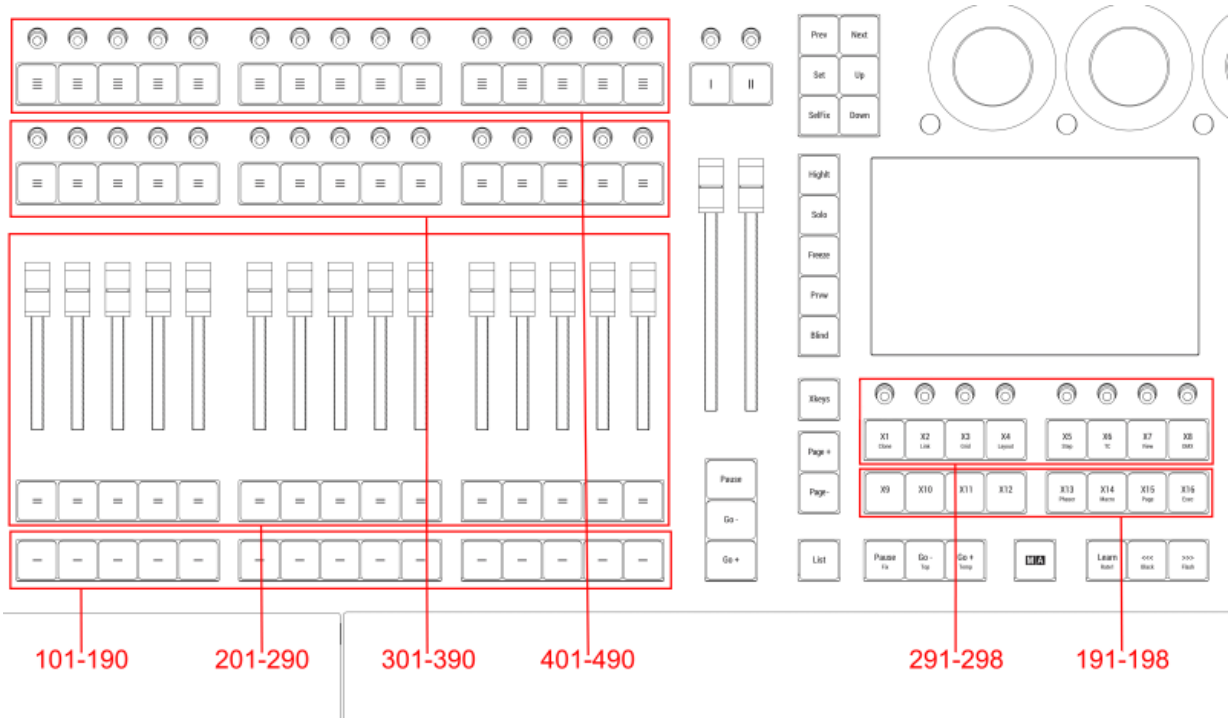
Location on grandMA3 compact consoles and grandMA3 onPC command wing

2.16.7. Executor elements

On the onPC and on the consoles, the executors are also in the Playback Control overlay and the Playback window. To access the Playback Control menu press F5 on the keyboard or the Playback Controls icon.

The number of physical executors depends on the console type.

The executors are handles to control objects in the show file. The executors are located on the left part of the console.



Executors on large consoles

Executor 101 thru 190

The executors 101 thru 190 are located as the lower buttons in the executor section. On the hardware keys, these have labeled one horizontal line on top of them. In the Playback Control overlay or the Playback window, they are named by their number.

Executor 191 thru 198

The executors 191 thru 198 are located as the lower row of Xkeys (X9 to X16). Please see the **Keys topic** for more information.

Executor 201 thru 290

The executors 201 thru 290 are located as the second lower buttons and the faders in the executor section. One button and the above fader is one executor. On the hardware keys, these have labeled two horizontal lines on top of them. In the Playback Control overlay or the Playback window, they are named by their number.

Executor 291 thru 298

The executors 291 thru 298 are located as the upper row of Xkeys (X1 to X8). Please see the **Keys topic** for more information.

Executor 301 thru 390

The executors 301 thru 390 are located as the second upper buttons and knobs in the executor section. One button and one knob is one executor. On the hardware keys, these have labeled three horizontal line on top of them. In the Playback Control overlay or the Playback window, they are named by their number.

Executor 401 thru 490

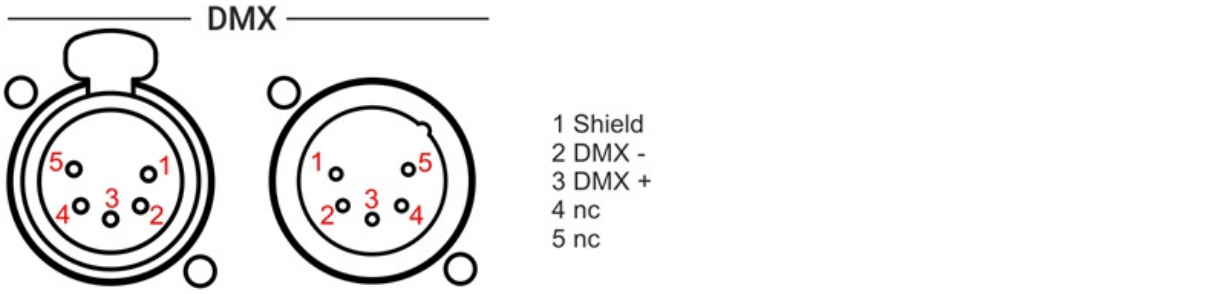
The executor 401 thru 490 are located as the upper buttons and knobs in the executor section. One button and one knob is one executor. On the hardware keys, these have labeled four horizontal line on top of them. In the Playback Control overlay or the Playback window, they are named by their number.

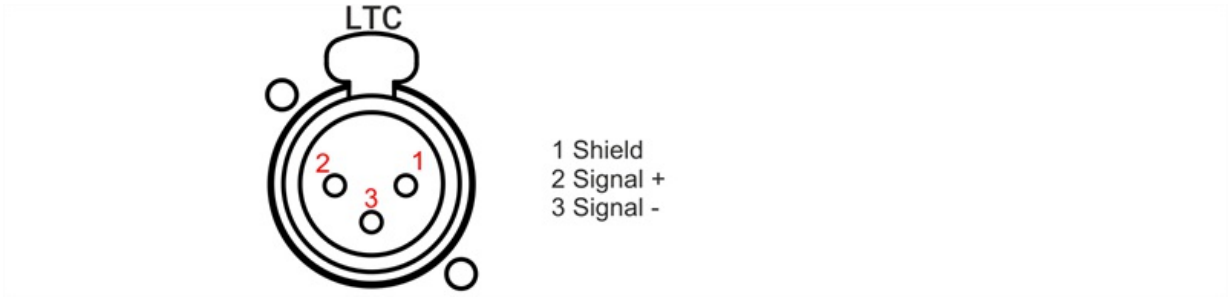
For more informations about executors, see the **Executors topic**

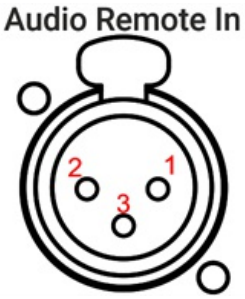
2.17. Connector Pin Assignment

This page gives an overview of all connectors and their respective pinouts.

XLR connectors

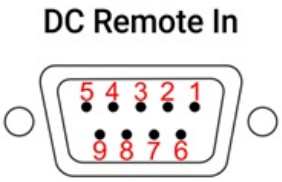






1 Shield
2 Signal +
3 Signal -

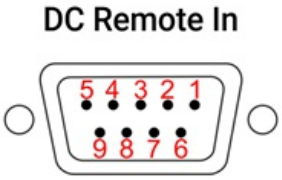
D-Sub connectors



1 Remote 1
2 Remote 2
3 Remote 3
4 Remote 4
5 +10V DC, max. 100mA
6 Remote 5
7 Remote 6
8 Remote 7
9 GND

Thread UNC 4-40

GPI for grandMA3 consoles and I/O Node

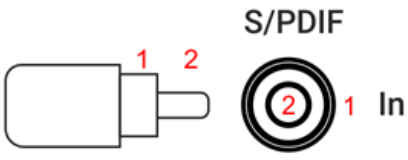


1 Remote 1
2 Remote 2
3 Remote 3
4 Remote 4
5 +5V DC, max. 100mA
6 Remote 5
7 Remote 6
8 Remote 7
9 GND

Thread UNC 4-40

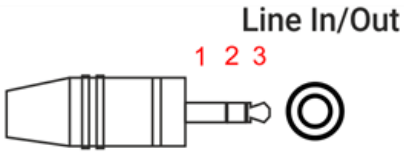
GPI for grandMA3 onPC command wing and command wing XT

RCA connectors



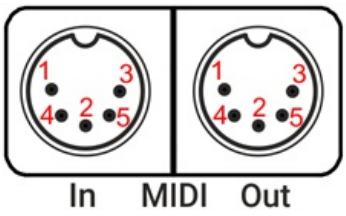
1 Shield
2 Signal

Jack sockets 1/4"



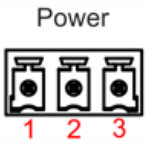
1 Shield
2 Signal R
3 Signal L

DIN connectors



1 nc
2 Shield
3 nc
4 Signal+ (Voltage Reference Line)
5 Signal- (Data Line)

Phoenix terminals



1 PE
2 N
3 L

Power connector: MC 1.5/ 3-ST1-5.08

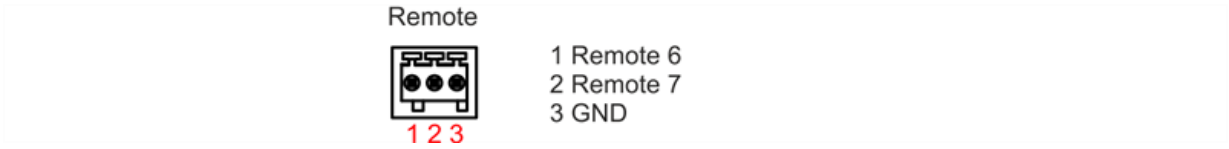
1 Shield
2 Signal -
3 Signal +

LTC connector: FK-MC 0.5/ 3-ST-2.5

1 Shield
2 Signal -
3 Signal +

DMX connector: FK-MC 0.5/ 3-ST-2.5

MIDI connector: FK-MC 0.5/ 3-ST-2.5



Remote connector: FK-MC 0.5/ 3-ST-2.5

Remote connector: FK-MC 0.5/ 7-ST-2.5

Hint:
To display the rear panel connectors on the letterbox screens, type:

Menu "Connectorview"

3. System Overview

This section takes a look at the different possibilities with a standalone console but also the options when expanding the system.

3.1. Standalone Device

If a console is not connected to anything, then it is a **Standalone** system. It is also a standalone system using a grandMA3 onPC with a grandMA3 onPC command wing connected.

Any grandMA3 device that can create and run a **session** is called **Station**.

Stations with network disabled are in **Standalone** mode.

A station is in **IdleMaster** mode when the network is On and it is ready to be in a session with other stations, but currently, it is alone. So a console that is connected to a network, but not in a session with other stations is considered as a master that is ready to connect to other stations.

The current status can be seen in the Network menu. The title bar has an area that displays the status.

If the station is in a session and it needs to be set to Standalone mode, then turn off the network. This can be done using the **GUI** or the command line (**LeaveSession keyword**).

In the **Standalone** or **IdleMaster** mode, the station is limited to only controlling the number of **parameters** the console/onPC unlocks.

You can only use the DMX ports on the console/wings.

If DMX is output via an Ethernet connection, then the console needs to be in a session. Even though not connected to other stations, it still needs to run an active session as IdleMaster.

3.2. Locally Networked Devices

When more stations are connected with each other in a **session**, then it is a networked system. Local networked systems are sessions running in the same network. This might be a big network spanning a big area, but if it is a closed system without contact with the rest of the world (for instance through the internet), then it is considered a local networked system.

The smallest networked system is two stations running in a session together where one is the master of the session and the other is connected running as a backup. A single station that outputs DMX locally is considered a **standalone system**.

Most systems have one console with one backup console or onPC and some networked devices to output the DMX.

If the devices that translate Ethernet data to DMX are MA nodes, then the DMX output is in sync, no matter where the DMX is output. So a blackout cue on LED fixtures is in sync when some LEDs got signal from the console and some LEDs from a node somewhere in the system.

Besides the stations, there are a lot of different devices that can be added to the system. Read more about the different devices in the **Device Overview topics**.

In all networked systems, there is one station that is the session master - this is the **GlobalMaster**. This is the one that is in charge of the communication and the one that executes commands triggered by remote inputs or from a command in a sequence.

Other stations in the system are in a **Connected** status. They are connected to the master station.

Traditionally we often recommend creating local networks with good quality switches, cables, and equipment. Most systems are local systems without any connection to the outside world.

3.3. World Server

Each station can be connected to the internet and connect to a World Server.

If an internet connection is detected, then the station automatically tries to connect to a world server.

The connection status of the world server can be seen in the **Command Line**. It is indicated by a globe icon. If the globe is green then the station is connected to the server.

Learn more about networking in the **Networking topic**.

The address of the official world server, provided by MA Lighting is: worldserver.malighting.de

The server address can be changed in the Network Menu. Learn more about this in the **Session topic**.

The server offers two functions: Fixture type files and Crash Log upload.

Fixture Type Files

Fixture type files from the GDTF-share and grandMA3 Share are provided as a direct import to the show file. Learn more about manually importing GDTF in the **Import GDTF topic** and learn more about patching fixtures in the **Add Fixtures to the Show topic**.

Crash Logs

A station creates a crash log if it crashes. These log files are automatically sent to the server when there is a connection. This means that the file is sent the next time the station is connected. This is also true if the station returns from a job and it is connected to the world server when it returns to the workshop.

If Tech Support is needed, please make sure the station has been online and provide the Tech Support date and time the crash happened and the serial number of the station (grandMA3 onPC stations also have a generated serial number). The serial number can be found using the **Version Keyword**.

Important:
If these functions are not desired, then change the server address in the **Network menu** to "0".

3.4. Parameters

Most people are used to think in DMX channels when considering how many fixtures can be controlled by a system.

MA Lighting cares more about **Parameters**.

What are Parameters

Parameters are also called **Attributes** in the software. A dimmer function is a parameter and a pan function is also a parameter.

The software is calculating the different attributes using a higher precision than what is output via DMX. So the software is calculating the parameter or attribute once and it is then scaled to the amount of DMX channels a fixture uses - typically one or two per attribute.

Why counting in Parameters and not DMX

It is to your advantage. In the MA world, you do not pay extra for using fixtures that are running 16 or 24 bit instead of the 8 bit used by one DMX channel.

There might be a need to add more networked nodes or devices to output all the parameters that are allowed to be controlled.

Example:

A simple moving head with a dimmer might use 5 DMX channels. The channels can be defined like this:

Definition:	DMX Channel:
Dimmer	1
Pan	2
Pan Fine	3
Tilt	4
Tilt Fine	5

The pan and tilt is one attribute each. Even though both pan and tilt each use two DMX channels it is only counted as one each in the parameter count. This means that the fixture only costs 3 parameters.

Definition:	DMX Channel:	Parameter cost:
Dimmer	1	1
Pan	2	2
Pan Fine	3	free

Tilt	4	3
Tilt Fine	5	free

This can be a big advantage when there are many fixtures with many 16-bit (or fine) channels.

What about Preprogramming and Parameters?

The show can be preprogrammed and visualized in the **3D window** without any parameter unlocking hardware. The lights are still visualized.

If third-party visualizers are used, then grandMA3 hardware is needed to give access to the parameters.

3.4.1. Calculate Parameters

Calculating the needed amount of parameters can be a big task depending on the size of the setup.

It is generally true that most modern fixtures use fewer parameters than DMX channels.

The best way to see how many parameters a show needs is to patch all the fixtures and then open the **Details** in the **System Info Window**.

Here the amount of needed (used) parameters can be seen, and the amount of currently available parameters.

Read the **Expand the amount of parameters topic** to learn how to get more.

Parameter Count

Some devices provide parameters:

- **grandMA3 full-size and grandMA3 full-size CRV:**
  20 480 parameters.
- **grandMA3 light and grandMA3 light CRV:**
  16 384 parameters.
- **grandMA3 extension:**
  none parameters.
- **grandMA3 replay unit:**
  8 192 parameters.
- **grandMA3 compact XT:**
  8 192 parameters.
- **grandMA3 compact:**
  8 192 parameters.
- **grandMA3 processing unit XL:**
  16 384 parameters.
- **grandMA3 processing unit L:**
  8 192 parameters.
- **grandMA3 processing unit M:**
  4 096 parameters.
- **grandMA3 8Port Node**:
  none parameters.
- **grandMA3 4Port Node:**
  none parameters.
- **grandMA3 2Port Node:**
  none parameters.
- **grandMA3 8Port Node DIN-Rail**:
  none parameters.
- **grandMA3 4Port Node DIN-Rail**:
  none parameters.

- **grandMA3 2Port Node DIN-Rail**:
  none parameters.
- **grandMA3 onPC rack-unit**:
  4 096 parameters.
- **grandMA3 onPC command wing XT:**
  4 096 parameters.
- **grandMA3 onPC command wing:**
  4 096 parameters.
- **grandMA3 onPC 8Port Node 4k:**
  4 096 parameters (when connected to grandMA2 onPC this node provides only 2 048 parameters).
- **grandMA3 onPC 4Port Node 4k:**
  4 096 parameters (when connected to grandMA2 onPC this node provides only 2 048 parameters).
- **grandMA3 onPC 2Port Node 2k:**
  4 096 parameters (when connected to grandMA2 onPC this node provides only 1 024 parameters).
- **grandMA3 onPC 8Port Node DIN-Rail 4k**:
  4 096 parameters (when connected to grandMA2 onPC this node provides only 2 048 parameters).
- **grandMA3 onPC 4Port Node DIN-Rail 4k**:
  4 096 parameters (when connected to grandMA2 onPC this node provides only 2 048 parameters).
- **grandMA3 onPC 2Port Node DIN-Rail 2k**:
  4 096 parameters (when connected to grandMA2 onPC this node provides only 1 024 parameters).
- **grandMA3 onPC Software**:
  none parameters.

3.4.2. Expand the Amount of Parameters

The grandMA3 processing units are the **only units** that expand the parameter count when using grandMA3 consoles.

Every grandMA3 processing unit added to the network also adds a number of parameters depending on the model!

A maximum of 15 processing units can be added to give parameters.

There is a **maximum limit of 250 000 parameters** in a grandMA3 session.

The grandMA3 processing units help with parameter calculations.

Examples:

1 grandMA3 full-size (20 480) + 1 grandMA3 processing unit XL (16 384) = 36 864 parameters

1 grandMA3 full-size (20 480) + 15 grandMA3 processing unit XL (16 384) = 250 000 parameters (the calculation is 266 240, but the limit is 250 000)

1 grandMA3 light (16 384) + 1 grandMA3 processing unit M (4 096) = 20 480 parameters

1 grandMA3 light (16 384) + 15 grandMA3 processing unit XL (16 384) = 250 000 parameters (the calculation is 262 144, but the limit is 250 000)

1 grandMA3 full-size (20 480) + 1 grandMA3 light (16 384) = 20 480 parameters (consoles cannot expand the parameter count so the parameters from the console with the highest number are unlocked)

1 grandMA3 light (16 384) + 15 grandMA3 onPC 2Port Node 2k (4 096) = 16 384 parameters (nodes cannot expand the parameter count with consoles, so the parameters from the console is used)

1 grandMA3 light (16 384) + 1 onPC computer with a grandMA3 onPC command wing (4 096) = 16 384 parameters (onPC command wings cannot expand the parameter count with consoles, so the parameters from the console are used)

Using a grandMA3 onPC with grandMA3 onPC command wing and grandMA3 onPC xPort Nodes

When a grandMA3 onPC is used, some grandMA3 **onPC** hardware is needed to unlock parameters.

Important:
The maximum number of parameters that are allowed in a grandMA3 onPC system is 4 096.

Any amount of onPC hardware can be used with a computer or the grandMA3 onPC rack-unit. Every piece of hardware will add its parameters until the limit of 4 096 parameters is reached.

grandMA3 xPort Nodes need to be onPC versions to unlock parameters.

These are the only two rules for parameters with grandMA3 onPC.

Remember more units can be added to get more DMX ports even after the parameter limit is reached.

Examples:

grandMA3 onPC rack-unit (4 096) + grandMA3 onPC 2Port Node 2k (4 096) = 4 096 parameters (the limit)

grandMA3 onPC rack-unit (4 096) + grandMA3 onPC 4Port Node 4k (4 096) = 4 096 parameters (the limit)

grandMA3 onPC + grandMA3 onPC command wing (4 096) = 4 096 parameters (the limit)

grandMA3 onPC rack-unit (4 096) + grandMA3 onPC command wing (4 096) = 4 096 parameters (the limit)

grandMA3 onPC command wing XT (4 096) = 4 096 parameters (the limit)

grandMA3 onPC + grandMA3 onPC command wing (4 096) + grandMA3 onPC 2Port Node 2k (4 096) = 4 096 parameters (the limit)

grandMA3 onPC + grandMA3 onPC command wing (4 096) + grandMA3 onPC fader wing (4 096) = 4 096 parameters (the limit)

grandMA3 onPC + grandMA3 onPC 2Port Node DIN-Rail 2k (4 096) = 4 096 parameters

grandMA3 onPC + grandMA3 onPC 2Port Node 2k (4 096) + grandMA3 8Port Node (none) = 4 096 parameters (the 8Port Node is not an onPC node)

grandMA3 onPC + grandMA3 onPC command wing (4 096) + grandMA3 onPC 8Port node 4k (4 096) = 4 096 parameters (the limit is reached and cannot be exceeded)

4. First Steps

This chapter describes how to prepare the grandMA3 consoles and the grandMA3 onPC to use it.

4.1. Unpack the device

- Unpack the device. Remove all packing material, strips and protection films.
- Keep the packing material for transport.

Information:
The device was tested for proper function at the factory.

4.2. Check scope of delivery

The list below shows the scope of delivery. If anything is missing, contact your local distributor.

grandMA3 full-size

- 2 x LED desk light grandMA3
- 1 x Dust cover grandMA3 full-size
- 2 x Magnetic plate for easy labeling of 15 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 full-size CRV

- 2 x LED desk light grandMA3
- 1 x Dust cover grandMA3 full-size CRV
- 2 x Magnetic plate for easy labeling of 15 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 light

- 2 x LED desk light grandMA3
- 1 x Dust cover grandMA3 light
- 1 x Magnetic plate for easy labeling of 15 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 light CRV

- 2 x LED desk light grandMA3
- 1 x Dust cover grandMA3 light CRV
- 1 x Magnetic plate for easy labeling of 15 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 compact XT

- 1 x LED desk light grandMA3
- 1 x Dust cover grandMA3 compact XT
- 1 x Magnetic plate for easy labeling of 15 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 compact

- 1 x LED desk light grandMA3
- 1 x Dust cover grandMA3 compact
- 1 x Magnetic plate for easy labeling of 10 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 extension

- 1 x LED desk light grandMA3
- 1 x Dust cover
- 1x Magnetic plate for easy labeling of 15 faders
- 1 x grandMA3 Quick Manual consoles

grandMA3 replay unit

- 1x grandMA3 Quick Manual consoles

grandMA3 processing units

- 1x grandMA3 Quick Manual processing

grandMA3 xPort Nodes

- 1x grandMA3 Quick Manual Nodes

grandMA3 Nodes DIN-Rail

- 1x grandMA3 Quick Manual Nodes DIN-Rail

grandMA3 I/O Nodes

- 1x grandMA3 Quick Manual I/O Nodes

grandMA3 I/O Nodes DIN-Rail

- 1x grandMA3 Quick Manual I/O Nodes

grandMA3 onPC command wing

- 1 x Dust cover
- 1 x Magnetic plate for easy labeling of 10 faders
- 1 x USB cable
- 1 x grandMA3 Quick Manual onPC command wing

grandMA3 onPC command wing XT

- 1 x Dust cover
- 1 x Magnetic plate for easy labeling of 10 faders
- 1 x grandMA3 Quick Manual onPC command wing XT

grandMA3 onPC fader wing

- 1x Dust cover
- 1x USB cable
- 1x Magnetic plate for easy labeling of 10 faders
- 1x grandMA3 Quick Manual onPC fader wing

grandMA3 rack-unit

- 1x grandMA3 Quick Manual onPC rack-unit

grandMA3 viz-key

- 1x USB cable (C/C)
- 1x USB cable (C/A)
- 1x grandMA3 Quick Manual viz-key

4.3. Position the device

Follow the instructions below:

- Do not place the device in humid rooms.
- Place the device onto a stable, flat and dry area.
- Do not cover the ventilation holes.
- Do not place the device directly into the sun.
- Keep a minimum distance of 15 cm (5.91 inches) between multi-touch screen surface and radio intercom systems. If you go below the minimum distance, this may lead to unexpected behaviors such as unwanted multi-touch actions or mouse movements.

4.4. Connect power

Devices with powerCON connector

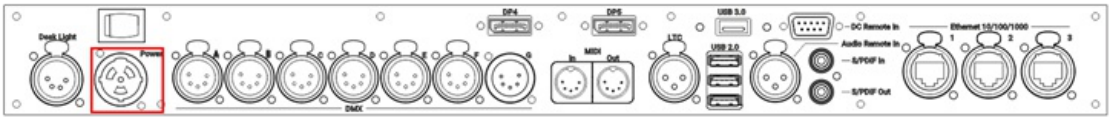
Warning:
If only the powerCON TRUE1 connector is provided, you need to attach a suitable cable to the connector.
To attach a suitable cable to the powerCON connector, contact your local dealer.

1. Insert the powerCON TRUE1 connector in the powerCON plug and twist it to lock clockwise.
   An audible click is heard.
2. Connect the other end of the cable to a suitable power source.

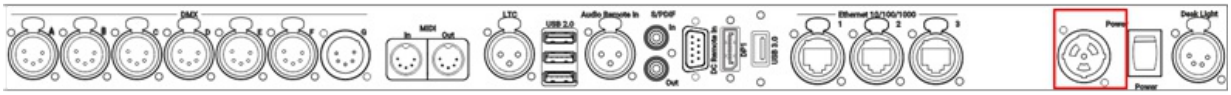
Devices with IEC connector

1. Insert the IEC connector into the corresponding jack.
2. Connect the power plug.

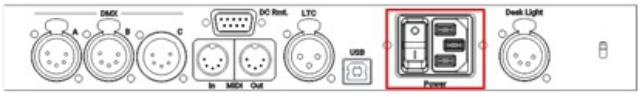
The device is connected to power.



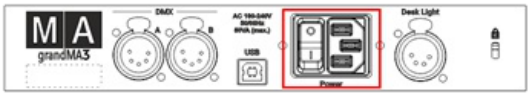
grandMA3 full-size (CRV) and light (CRV) rear panel – power



grandMA3 compact (XT) rear panel – power

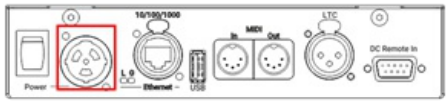


grandMA3 onPC command wing (XT) rear panel – power



grandMA3 fader wing rear panel – power

grandMA3 extension rear panel – power



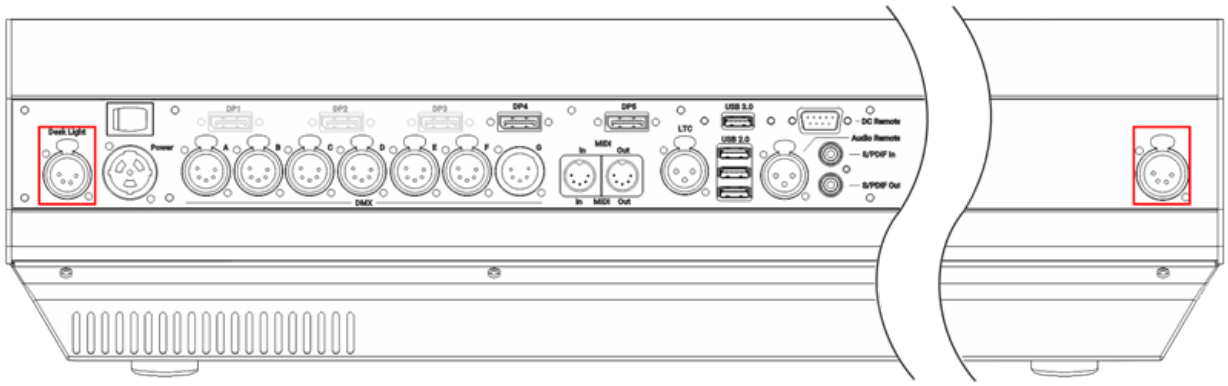
grandMA3 I/O Node rear panel – power

4.5. Connect desk light

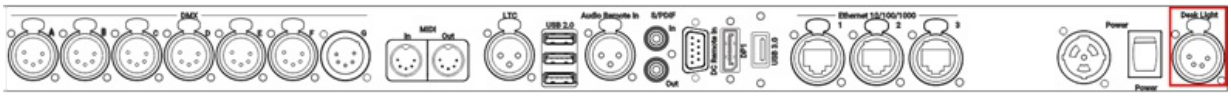
The grandMA3 comes with one or two desk lights, depending on the model. For more information see, **scope of delivery**.

- Connect the 4 pin XLR connector to the desk light connectors on the rear panel.

The desk lights are connected.

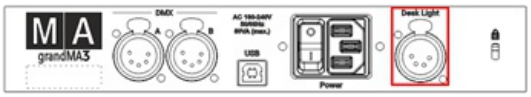


grandMA3 full-size (CRV) and light (CRV) rear panel – desk light



grandMA3 compact (XT) rear panel – desk light

grandMA3 onPC command wing (XT) rear panel – desk light



grandMA3 fader wing rear panel – desk light

grandMA3 extension rear panel – desk light

To view the pinout of the XLR connector, refer to the topic **Connector Pin Assignment**.

4.6. Connect External Screens

You can connect up to five external touch screens with a grandMA3 full-size CRV (= Control Room Version). For more information about the amount of display port connectors of each model see **Quick Manual Consoles - Technical Data**.

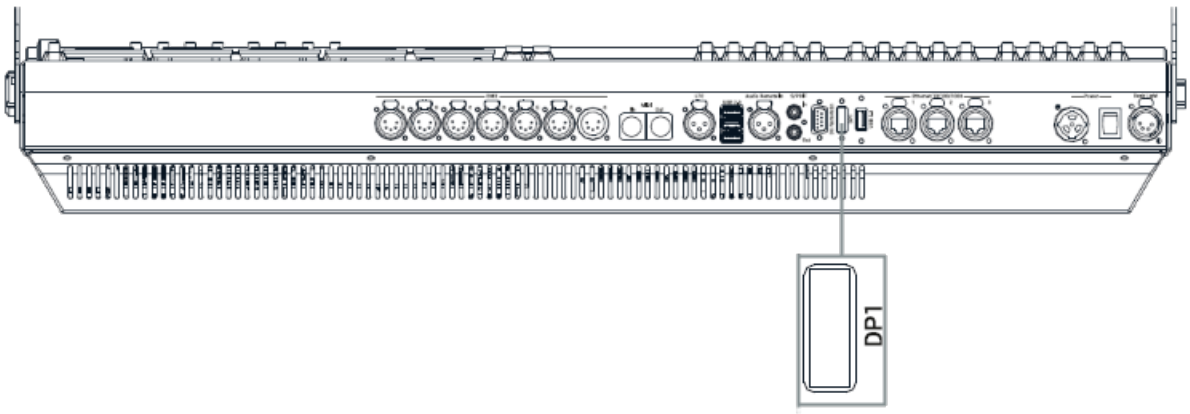
Important:
Display port connector
The grandMA3 series supports only native display port connectors on external screens.

Important:
In grandMA3 CRV versions use at least one Full HD display and a touch display or a mouse.

Requirement:

- The console is turned off
- DisplayPort

Hint:
The grandMA3 compact and compact XT models only have one DisplayPort called DP1.



Example of a DisplayPort connector on a grandMA3 compact XT

Place the screens in the following order:

grandMA3 full-size screen order

Screens 5 and 4 are always external screens. Screens 3, 2, and 1 can be internal screens and can be external screens. The screen order is important for the mouse behavior from screen to screen.

Recommendations for external monitors:

- Native DisplayPort connection
- Resolution of 1920 x 1080 (full HD)
- Microsoft Windows® 10 multi-touch compliance
- Separated USB connection for touch functionality

Connect external touch screens:

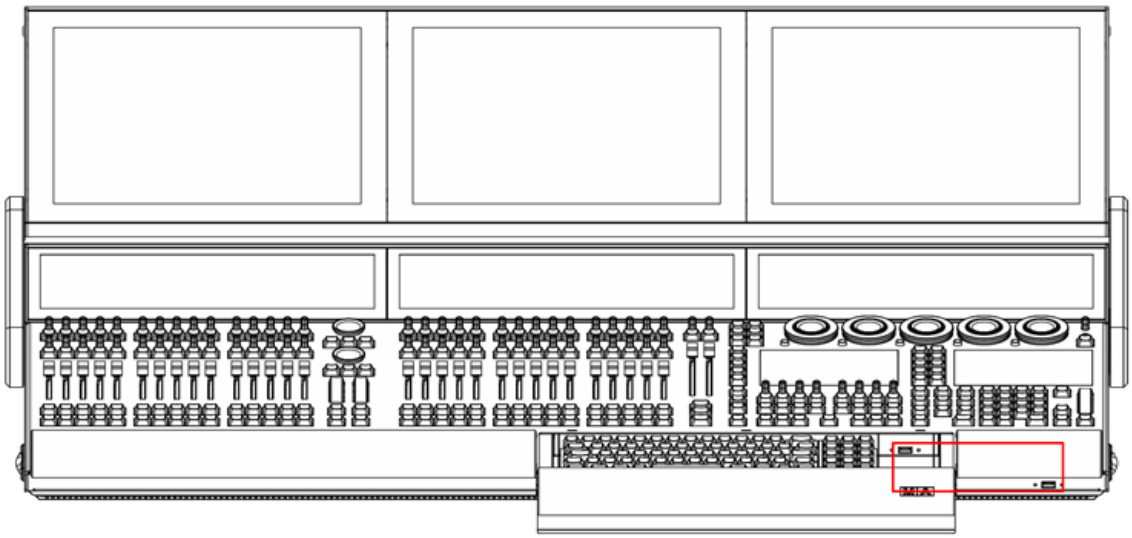
1. Place the first external screen right beside or above screen 1 and the second external screen left beside or above screen 3.
   The screen number of the external screens will be numbered consecutively.
2. Connect the display port cable of screen 4 with the DP connector DP4.
3. Connect the USB cable with the corresponding USB port.
4. Connect the display port cable of screen 5 with the DP connector DP5.
5. Connect the USB cable with the corresponding USB port.
6. To boot up the console, press Power.
7. To check the USB setting, tap Settings, USB configuration.
8. To adjust the touch configuration, tap Settings, Touch configuration.

4.7. Connect USB devices

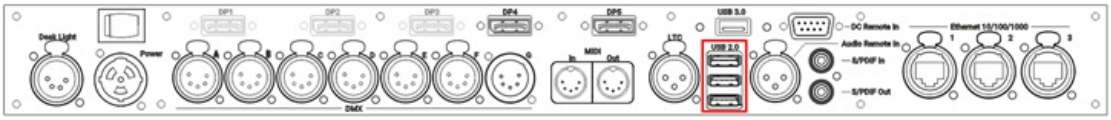
You can connect an external USB mouse, an external USB keyboard, or an external touch screen to the grandMA3 device using the USB ports.

Depending on the type of the grandMA3 device there are up to 6 USB connectors:

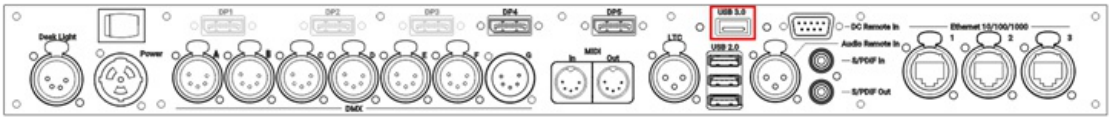
- 1x USB 3.0 at the front on the right of the console
- 1x USB 3.0 inside the keyboard drawer
- 3x USB 2.0 and 1x USB 3.0 on the rear panel



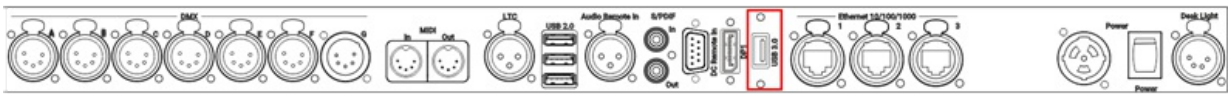
grandMA3 full-size USB 3.0 (type A) inside keyboard drawer and on the right



grandMA3 full-size (CRV) rear panel USB 2.0 (type A)

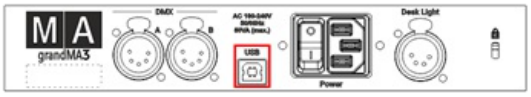


grandMA3 full-size rear panel USB 3.0 (type A)



grandMA3 compact (XT) rear panel USB 3.0 (type A)

To connect a grandMA3 onPC device with the computer or laptop, use the USB connector (type B):



grandMA3 onPC fader wing rear panel USB 2.0 (type B)

grandMA3 onPC command wing rear panel USB 2.0 (type B)

4.8. Connect DMX

You can connect DMX devices to the grandMA3 devices.

Depending on the type of the grandMA3 device there are up to six DMX Out ports (e.g. A-F) and one DMX In port (e.g. G).

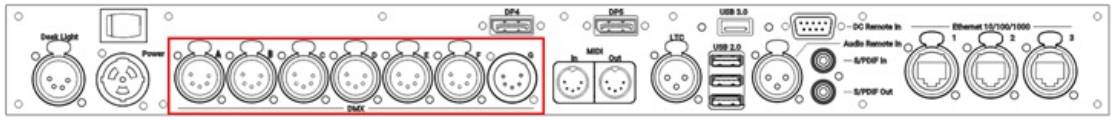
No matter which physical direction the connector has, it can be configured in the software and is therefore capable for both directions: in and out.

To adjust the DMX Remote settings, read the **In and Out topic**.

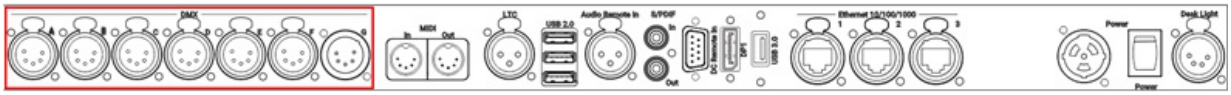
To configure the DMX ports (Off, Out, RDM, In), read the **DMX Port Configuration topic**.

- Connect the 5pin XLR DMX cable to a DMX connector on the console and a DMX device.

The DMX device is connected to the **XLR connector**.

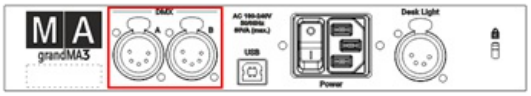


grandMA3 full-size (CRV) and light (CRV) rear panel – DMX

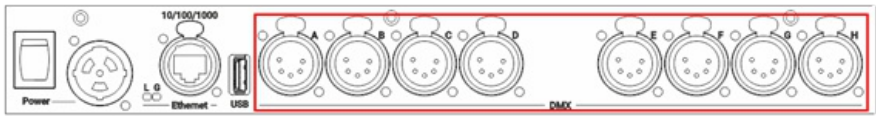


grandMA3 compact (XT) rear panel – DMX

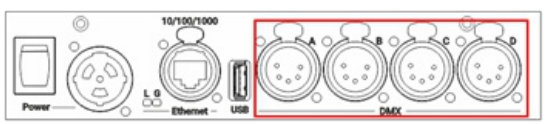
grandMA3 onPC command wing (XT) rear panel – DMX



grandMA3 onPC fader wing rear panel – DMX

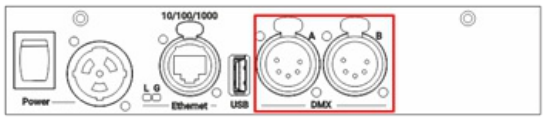


grandMA3 8Port Node rear panel – DMX



grandMA3 8Port Node rear panel – DMX

grandMA3 4Port Node rear panel – DMX



grandMA3 2Port Node rear panel – DMX

To view the pinout of the XLR connector, refer to the topic **Connector Pin Assignment**.

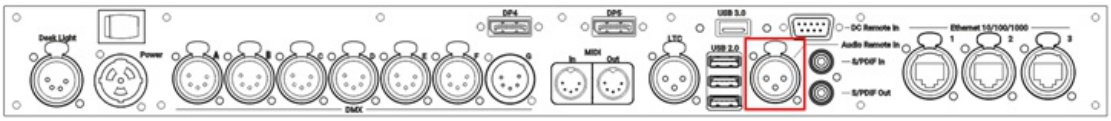
4.9. Connect Audio In

To use a sound trigger or BPM, connect a sound source to the console.

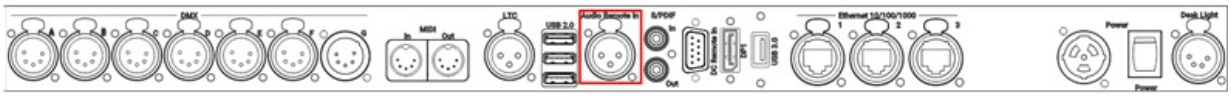
Information:
The signal strength should be a minimum of 50 mV.

- Connect the 3pin XLR sound source cable to the **Audio Remote In connector** on the rear panel.

The sound source is connected to the **Audio Remote In connector**.



grandMA3 full-size rear panel – Audio Remote In



grandMA3 compact (XT) rear panel – Audio Remote In

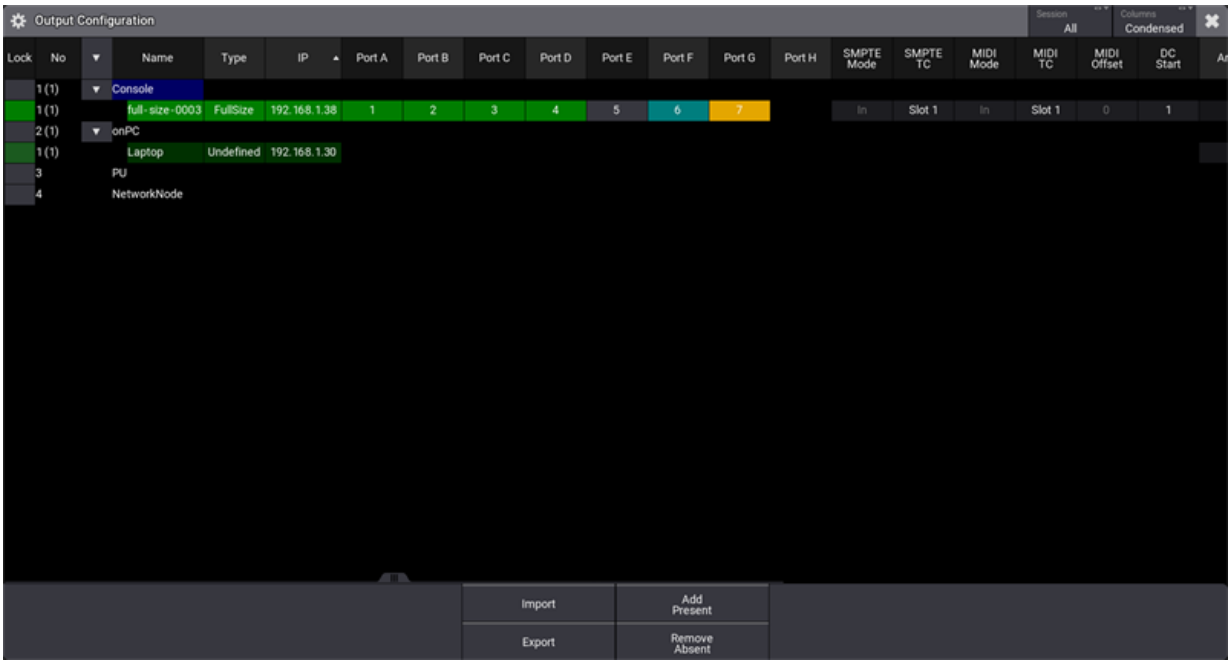
To view the pinout of the DIN connector, refer to the topic **Connector Pin Assignment**.

4.10. Connect MIDI

The MIDI connectors can be used for MIDI input or output, e.g. to send the MIDI timecode signal to the sound engineer.

To adjust the MIDI Remotes settings, read the **Remote In and Out topic**.

You can configure the MIDI port settings per grandMA3 device in the **Output Configuration menu**:



Output configuration menu with a console and an onPC

- Access the menu by pressing Menu and then tap Output Configuration.

-OR-

- Use the command line to open the menu:

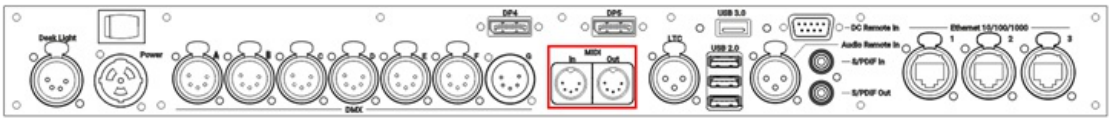
```
User name[Fixture]> Menu "OutputConfig"
```

- To adjust the MIDI direction, tap and hold or right-click MIDI Mode.

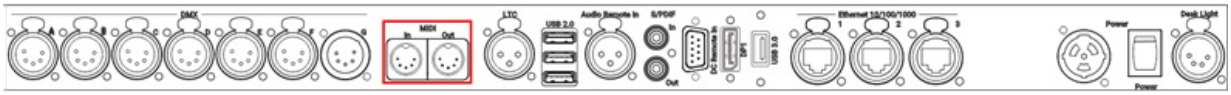
- Select In, Out, or Through.
  The MIDI direction is set.

To learn more about MIDI TC settings, read the **What are Timecode Slots** topic.

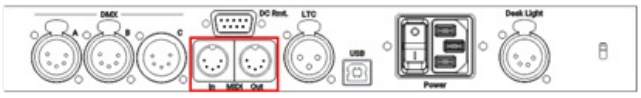
- Connect the MIDI source to the **DIN connector** on the rear panel of the console.
  The MIDI source is connected to the **DIN connector**.



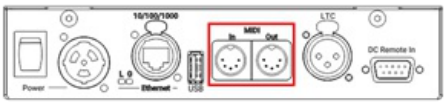
grandMA3 full-size (CRV) and light (CRV) rear panel – MIDI



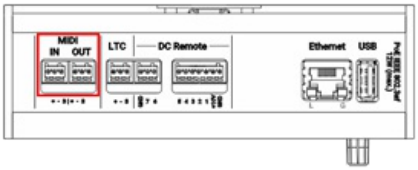
grandMA3 compact (XT) rear panel – MIDI



grandMA3 onPC command wing (XT) rear panel – MIDI



grandMA3 I/O Node rear panel – MIDI



grandMA3 I/O Node DIN-Rail rear panel – MIDI

To view the pinout of the DIN connector, refer to the topic **Connector Pin Assignment**.

4.11. Connect LTC

The LTC connector can be used for timecode input or for timecode output, e.g. to send the timecode signal to the sound engineer.

You can configure the direction of the LTC port in the **output configuration**.

To synchronize the console with an SMPTE timecode source, connect an SMPTE source to the LTC port.

Information:
The supported time formats are:
- 24 fps (frames per second)
- 25 fps
- 30 fps

If you send 29.97 or 30 drop frame, it will be interpreted as 30 fps.

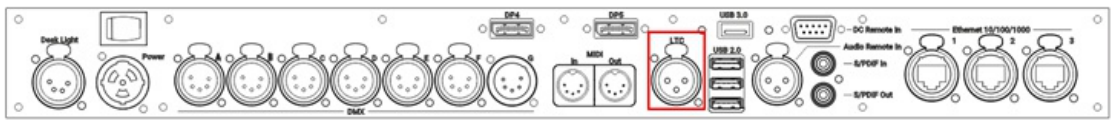
Sound and timecode signal levels

Min. level	Max. level	Recom. level
-11 dBu	+15 dBu	0 dBu
0.2 Veff	4.4 Veff	0.8 Veff

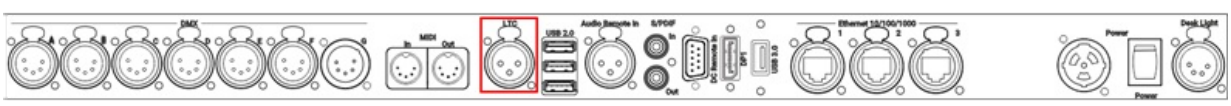
Information:
The signal strength should be a minimum of 200 millivolts.
Pin 1: Ground
Pin 2: - (minus)
Pin 3: + (plus)

- Connect the SMPTE source to the LTC connector on the rear panel of the console.

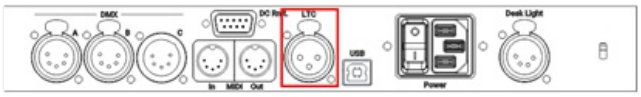
The SMPTE source is connected to the LTC connector.



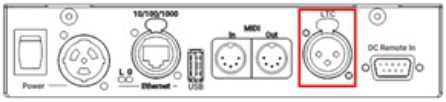
grandMA3 full-size (CRV) and light (CRV) rear panel – LTC



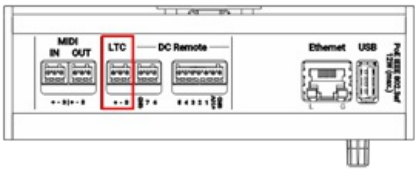
grandMA3 compact (XT) rear panel – LTC



grandMA3 onPC command wing (XT) rear panel – LTC



grandMA3 I/O Node rear panel – LTC

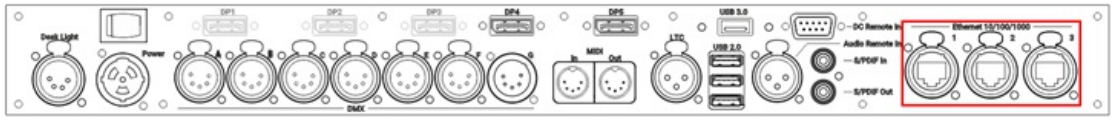


grandMA3 I/O Node DIN-Rail rear panel – LTC

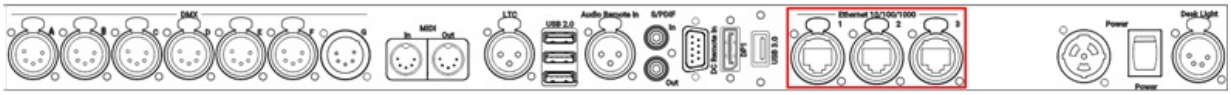
To view the pinout of the DIN connector, refer to the topic **Connector Pin Assignment**.

4.12. Connect Ethernet

Depending on the type of grandMA3 device, there are up to three Ethernet ports on the rear panel.

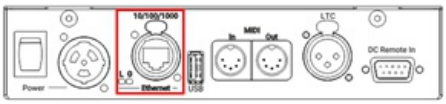


grandMA3 full-size (CRV) and light (CRV) rear panel – Ethernet

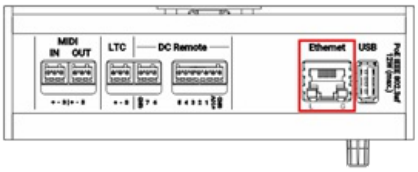


grandMA3 compact (XT) rear panel – Ethernet

grandMA3 extension rear panel – Ethernet



grandMA3 I/O Node rear panel – Ethernet



grandMA3 I/O Node DIN-Rail rear panel – Ethernet

Hint:
The naming of the Ethernet ports is individual, for example CON1, CON2, CON3 in the console.
In the onPC software, the Ethernet ports are named by the operating system, for example Ethernet, Ethernet2 etc.

Information:
All Ethernet ports can use these DMX protocols:
- Art-Net
- sACN

Important:
To avoid damage, the Ethernet ports must not be connected with Power over Ethernet (PoE) except for the grandMA3 devices qualified for PoE (e.g. xPort Nodes PoE, onPC xPort Nodes PoE, or I/O Node PoE).

Ethernet Ports 1-3

Requirement: Use STP (shielded twisted pair) cable with an RJ45 connector, at least CAT-5e.

To connect one of the three Ethernet ports:

1. Connect one end of an Ethernet cable to the **Ethernet 1, 2, or 3**.
2. Connect the other end to a suitable switch.
3. Connect other grandMA3 equipment to the switch and the switch to the power supply.

Ethernet is connected to a port.

Ethernet LEDs

The L (link) LED flashes when the Ethernet connection is active.

The G (gigabit) LED flashes when a gigabit connection is available.

4.13. Connect DC Remote In

To use the DC Remote In with grandMA3 consoles, onPC command wings, and I/O Nodes connect a contact closure switch, for example, a light barrier or a push button.

For further information see the topics **Remote keyword**, **Remote In and Out**, and **Output configuration**.

Hint:
You can use up to 64 input channels within a session.

Hint:
It is possible to analogously move the Master Fader using a grandma3 console or I/O Node connected with a potentiometer (0 to +10V DC).
The onPC products can switch on and off, but do not fade.

- Generate a switch or connect an external source that sends up to +10V DC to pin 1 for the console to react to analog input number 1.
- The recommended resistance is 5 kohms to 10 kohms.
- To use the DC Remote, feed a voltage signal (max. +10V DC into the corresponding input pin. For more information see the pinout image below.

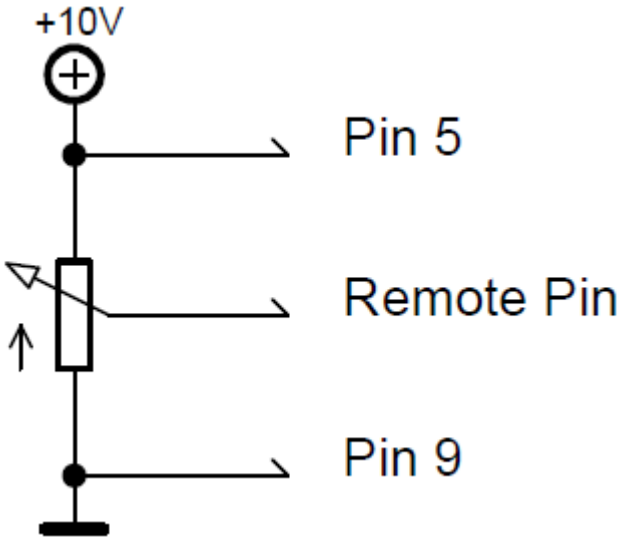
To connect a switch:

1. Take +10V DC voltage in pin 5 for the grandMA3 console or I/O Node
   -or-
   +5V DC voltage for grandMA3 onPC command wing and command wing XT
2. Take an external voltage source (+10V DC in grandMA3 consoles or I/O Nodes and +5V DC in grandMA3 onPC command wing and command wing XT), connect its ground to the common ground pin of the device.

Connect the +10V DC voltage source to one input pin 1-4 or 6-8 with a potential-free contact (switch, buzzer, motion detector, or any other switching device) in between.

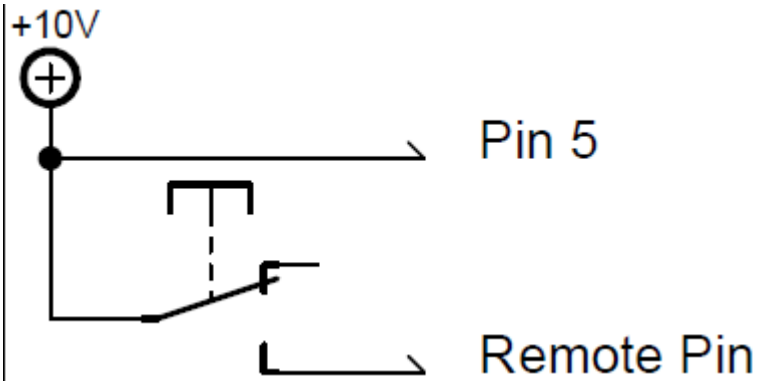
Circuit examples:

- Potentiometer +10V DC in grandMA3 consoles or I/O Nodes



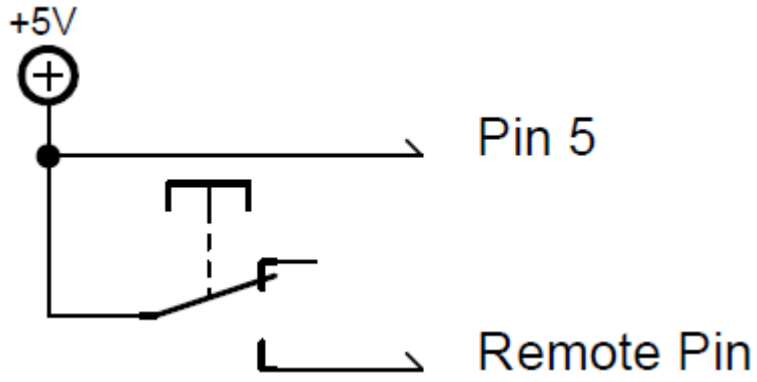
Potentiometer +10V DC

- Switch +10 DC in grandMA3 consoles or I/O Nodes



Switch +10V DC

- Switch +5V DC in grandMA3 onPC command wing and command wing XT

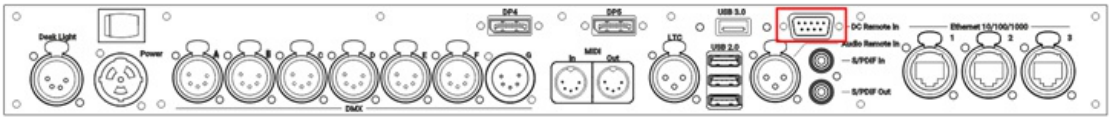


Switch +5V DC

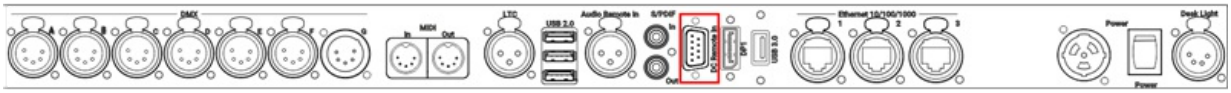
Important:
Pin layout in grandMA3 consoles or I/O Nodes:
The grandMA3 consoles or I/O Nodes have a 9-pin D-sub, enabling 7 remote inputs:
Pin 1-4 = input channels 1, 2, 3, 4
Pin 5 = +10V DC grandMA3 consoles or I/O Nodes/+5V DC grandMA3 onPC command wing and command wing XT
Pin 6-8 = input channels 5, 6, 7
Pin 9 = common ground

- Connect a D-sub plug to the DC Remote In connector on the rear panel.

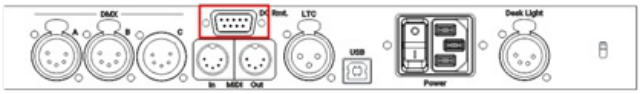
DC Remote In control is connected.



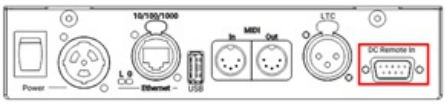
grandMA3 full-size (CRV) and light (CRV) rear panel – DC Remote In



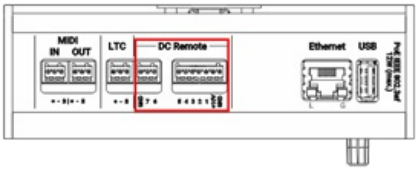
grandMA3 compact (XT) rear panel – DC Remote In



grandMA3 onPC command wing rear panel – DC Remote In



grandMA3 I/O Node rear panel – DC Remote In



grandMA3 I/O Node DIN-Rail rear panel – DC Remote In

To view the Pinout of the Sub-D connector, refer to the topic **Connector Pin Assignment**.

4.14. Connect grandMA3 extension

The grandMA3 extension allows to extend the amount of physical executor handles for a grandMA3 full-size, grandMA3 light or RPU.

You can connect up to 3 grandMA3 extensions to a grandMA3 console depending on its type:

Device	Number of grandMA3 extensions
grandMA3 full-size / full-size CRV	1
grandMA3 light / light CRV	2
grandMA3 RPU	3

Important:
It is only possible to connect extensions to grandMA3 full-size (CRV), grandMA3 light (CRV) or grandMA3 RPUs.

A grandMA3 extension has to be connected to the same network or network interface that is used for the MA-Net communication.

The software version of the extension must be the same as the console.

Hint:
You can see the software version of the extension in the lower-left corner of the extension's display or in the network menu of the connected console.

Furthermore, an extension will connect to a specific console but not to a session.

To establish a connection with an extension, the IP address of the extension has to be in the same IP range as the console. The IP address of an extension can only be changed remotely from a console or onPC station. To do so, go to Menu - Network Menu and select the entry of the extension. Press and hold the IP cell. Use the pop-up to change the IP settings of the extension remotely.

To establish a connection between a console and an extension, select the extension in the network menu on the console you want to connect, then tap Invite Station. The connection will be initiated.

It is not possible to invite the extension to a third console when it is already connected to a different console. To disconnect the extension from a console, cancel the connection on the console the extension is connected to at the moment. To do so, select the console in the network menu and tap Dismiss Station. After that, it can be connected to a different console.

The column Remote IP in the network menu displays to which console the extensions are connected at the

moment.

When an extension is successfully connected with a console, it will display the first wing of executors by default. To display a different executor wing, go to Menu - Settings - Extension Configuration. Within the Extension Configuration menu, it is only possible to change the WingID for the connected extension.
There are 2 column modes: Condensed (default) and Full. The condensed mode only displays information about the connection state, IP, and WingID. The full mode also displays the columns Connected Count and Device Type.

- Connection State: Can be Yes or No. This cell cannot be edited by the user and displays if the extension is connected with the console or not.
- IP: Displays the IP address of the extension. This cell also cannot be edited by the user.
- Wing ID: Displays the ID of the executor wing the extension controls. This property can be changed by the user.
- Connected Count: The higher the number the more often the extension tried to connect to the console. In a faulty network environment, the number can increase fast due to reconnection. This value cannot be edited by the user.
- Device Type: This column displays the device type of the extension. Typically it is grandMA3 Fader Module Encoder (MFE). This cell also cannot be edited.

The desk light of an extension is controlled together with the desk lights of the console.

The custom section of an extension can mirror the **custom section of the console** or can be independent of the console, depending on the user settings.

To learn more about user settings, read the **User settings topic**.

4.15. Connect grandMA3 fader wing

The grandMA3 onPC fader wing expands the onPC system with additional playback capabilities.

Important:
It is only possible to connect fader wings to grandMA3 onPC systems.
A maximum of two fader wings and one command wing can be connected.

Important:
Independent of the devices connected to the grandMA3 onPC software, the number of parameters can only reach a maximum of 4 096.

*For more information see: **Expand the Amount of Parameters topic**.*

Procedure:

1. Connect the grandMA3 onPC fader wing with your command wing XT, PC, or laptop using a USB cable.
2. Start the grandMA3 command wing XT, PC or laptop.
3. Switch on the grandMA3 onPC fader wing.

To connect a grandMA3 onPC fader wing in combination with a grandMA3 onPC command wing to expand the number of executors, the wing configuration has to be changed:

1. To check the USB setting, tap Settings, USB configuration.

USB configuration auto adjust

1. To adjust the wing ID, tap WingID, and change the ID.

USB configuration set to Wing2

4.16. Turn on the Device for the First Time

Hint:
The safety instructions and technical specifications can be found in the Quick Manual of the respective product, e.g. consoles:
For information on safety instructions see **grandMA3 Quick Manual Consoles – Safety**.
For technical specifications, see **Technical Data** in the **grandMA3 Quick Manual consoles**.

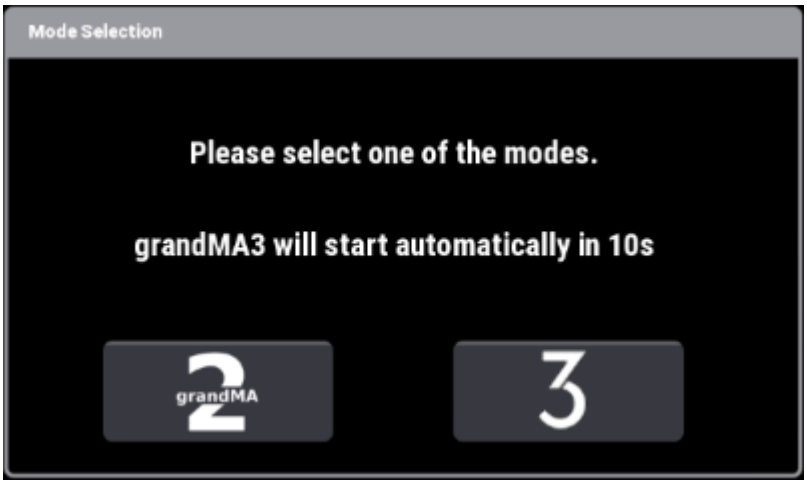
Turn on the grandMA3 Device

1. Unpack the grandMA3 device.
2. Remove the packaging and cushioning material.
3. Place the device indoors on a stable surface.
4. Connect external monitors using the native DisplayPort connectors (optional).
5. Connect an external mouse or keyboard using the USB ports (optional).
6. Insert the power connector into the corresponding jack.
7. Connect the power plug with the mains.
8. Turn on the power switch on the rear panel.
9. Press the power key on the front panel.

The device starts booting.

Select Mode

During start-up, the **Mode Selection** dialog appears:

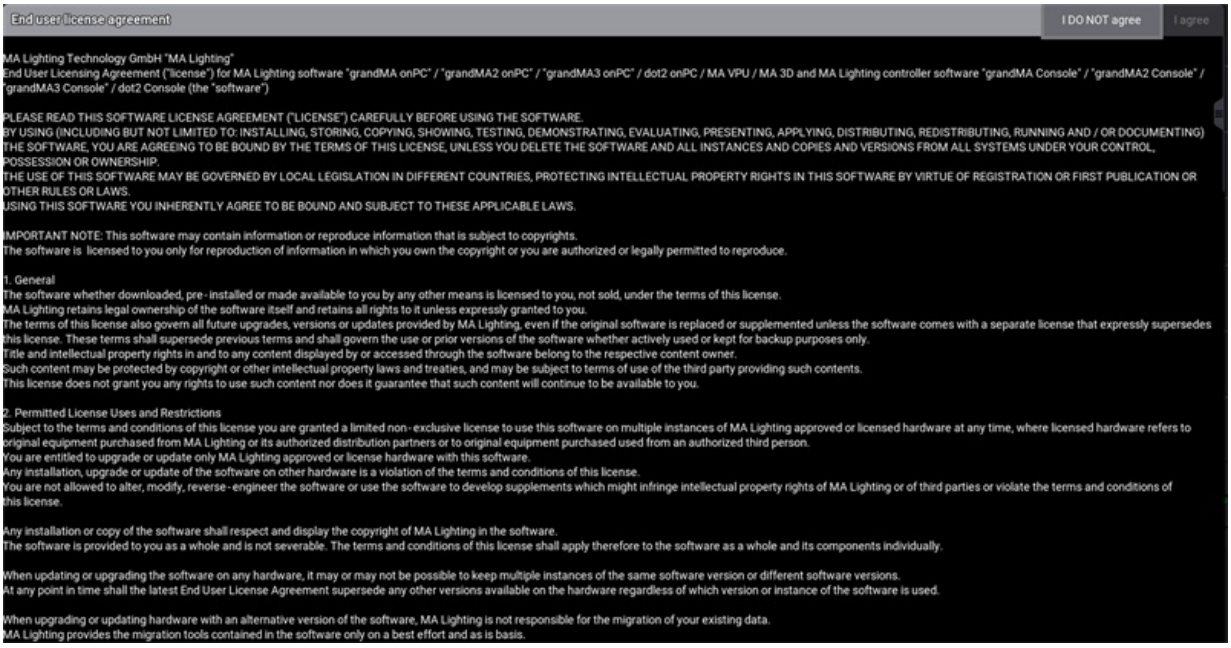


- Tap to select a mode.

For more information, read the **Mode2 topic** in the section grandMA3 Mode2 of the **grandMA2 User Manual**.

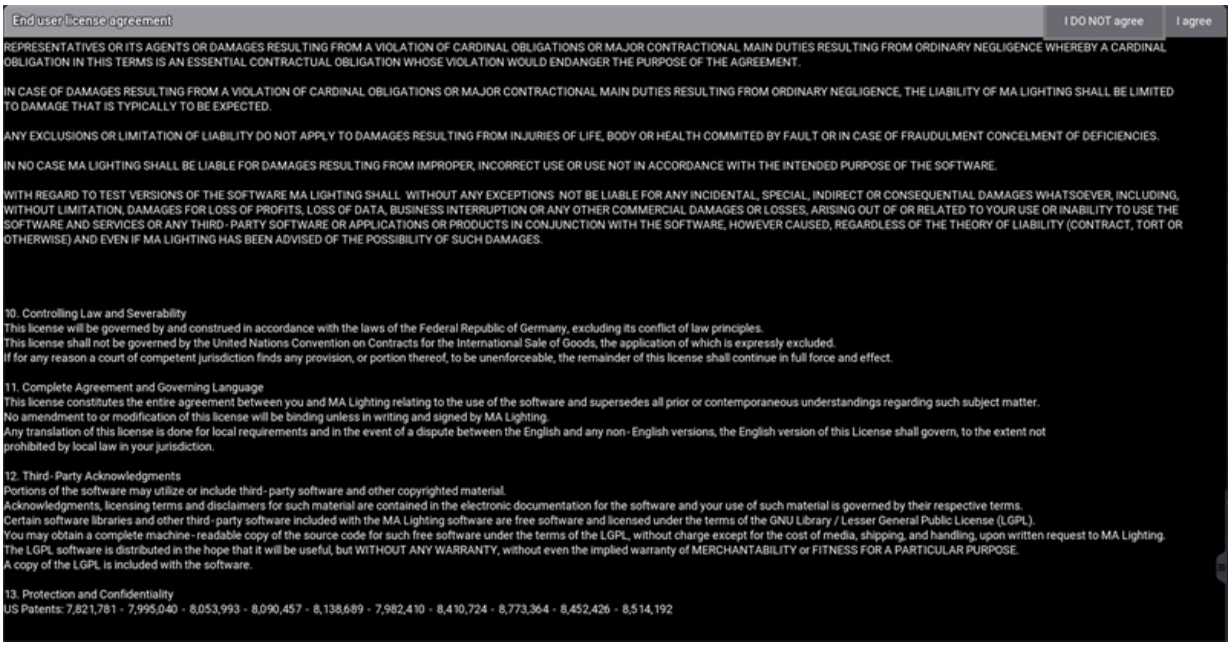
End-User License Agreement (EULA)

After selecting grandMA3 mode, the console starts to boot and the pop-up **End-User License Agreement (EULA)** opens.



End-user License Agreement (EULA)

- Scroll down to read the complete EULA. The button I agree in the upper right corner of the pop-up turns white.



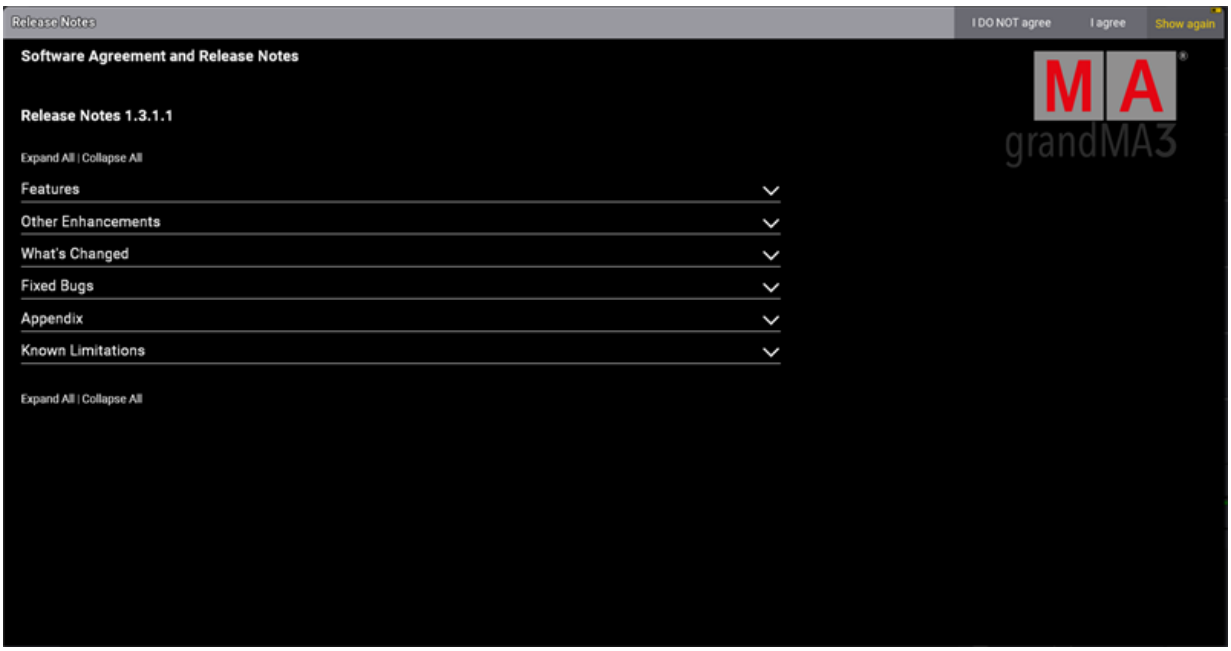
End-user License Agreement (EULA) I agree button

- To confirm the EULA, tap I agree.

Hint:
The EULA will only be displayed the first time of booting the console and every first boot after updating the software.

Release Notes

The Release Notes of the current version appear on the screen.



Release Notes

- To confirm the Release Notes, tap I agree.
- If you do not want the software to show the Release Notes again after booting, tap Show again.

You can now use your grandMA3 device.

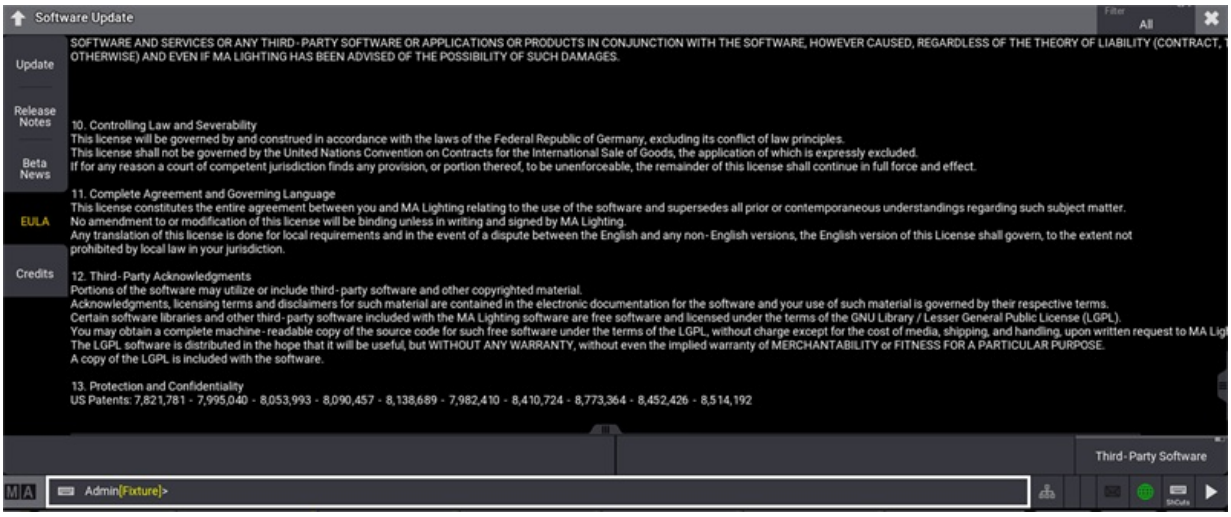
Third-party Software

The software comes with standard codecs to play videos and display images.

When you install third-party software, make sure that you have the required licenses.

Third-party software can be activated by accepting the additional third-party software agreement.

- To accept the additional third-party software agreement, **open the Software Update menu**.



- Tap EULA, then tap Third-Party Software.

- Tap I AGREE to accept the additional third-party software agreement

5. grandMA3 onPC

MA Lighting offers the perfect solutions for getting started with. Simply download the grandMA3 onPC software and connect to any of the grandMA3 onPC products to unlock parameters.

The amount of DMX outputs varies with each grandMA3 onPC product but all of them unlock parameters for the full grandMA3 software functionality. There is no compromise in feature set and the range of products allow you to make the right choice for your budget or individual needs.

There are three choices for onPC solutions: the grandMA3 onPC command wing, which offers the ultimate grandMA3 mobile solution, the grandMA3 onPC command wing XT, featuring a built-in MA mother board, and the grandMA3 onPC xPort Nodes, which offer the most cost-effective DMX output solution for software only show control.

Every grandMA3 onPC xPort Node is capable of handling MA-Net3, MA-Net2, sACN and Art-Net data and is fully RDM compliant. A 3.9" color display on the front face allows for simple configuration and provides a quick overview of the status of each Node.

All grandMA3 onPC xPort Nodes are equipped with a powerful processor and a 1,000Mbit/s Ethernet connection to ensure the most reliable and stable Ethernet to DMX and vice versa conversion.

The grandMA3 onPC xPort Nodes are high quality solutions that can handle busy Ethernet connections without ditching DMX or RDM messages.

5.1. System Requirements

If you would like to run grandMA3 on your PC, here is what it takes.

Type	Minimum	Recommended
Operating system	Windows® 10 64bit version 1803 or later Windows® 11 64bit version 21H2 or later Apple® macOS® High Sierra 10.13 or later Admin rights are required for all operating systems.	
Processor	6th Generation Intel® Core CPU™ (processor-based platform or later) with 4 or more cores and SSE 4.2 or comparable AMD® CPU. Apple® M1® SoC CPU with 8 or more cores.	
RAM	8 GB	16 GB
Hard disk	32 GB of available space.	type SSD
Graphics card	Any graphic card with hardware acceleration, OpenGL 4.1 support, and 1024 MB VRAM.	4 096 MB VRAM
Resolution	1 920 x 1 080 or higher.	
Network card	1000BASE-T	

Additional requirements:

- To use the grandMA3 onPC in session with grandMA3 consoles, a minimum of 16 GB of RAM is required.
- To use the online help and download the latest version of grandMA3 onPC, make sure that you have internet access.
- To save the software on a USB flash drive, use a USB 2.0 or 3.0 port.

Hint:
In general, we recommend more cores and more cache memory size instead of a higher processor clock speed.

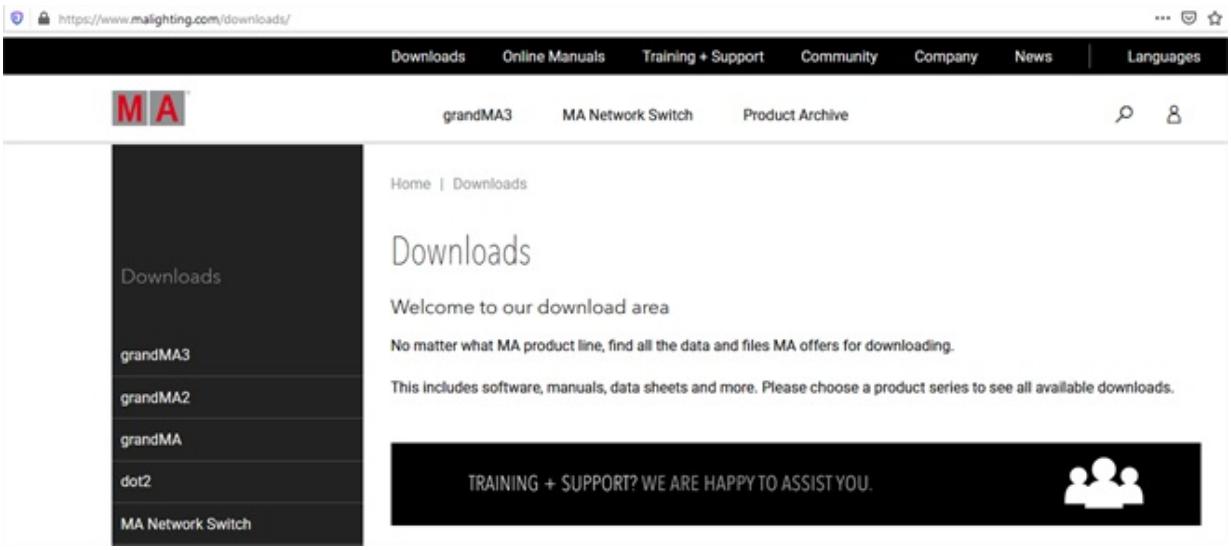
5.2. Windows installation

To run the grandMA3 onPC software, copy and install the program files on your PC.

The installation is possible in every root directory or in the **standard directory** "C:\Program Files\MALightingTechnology".

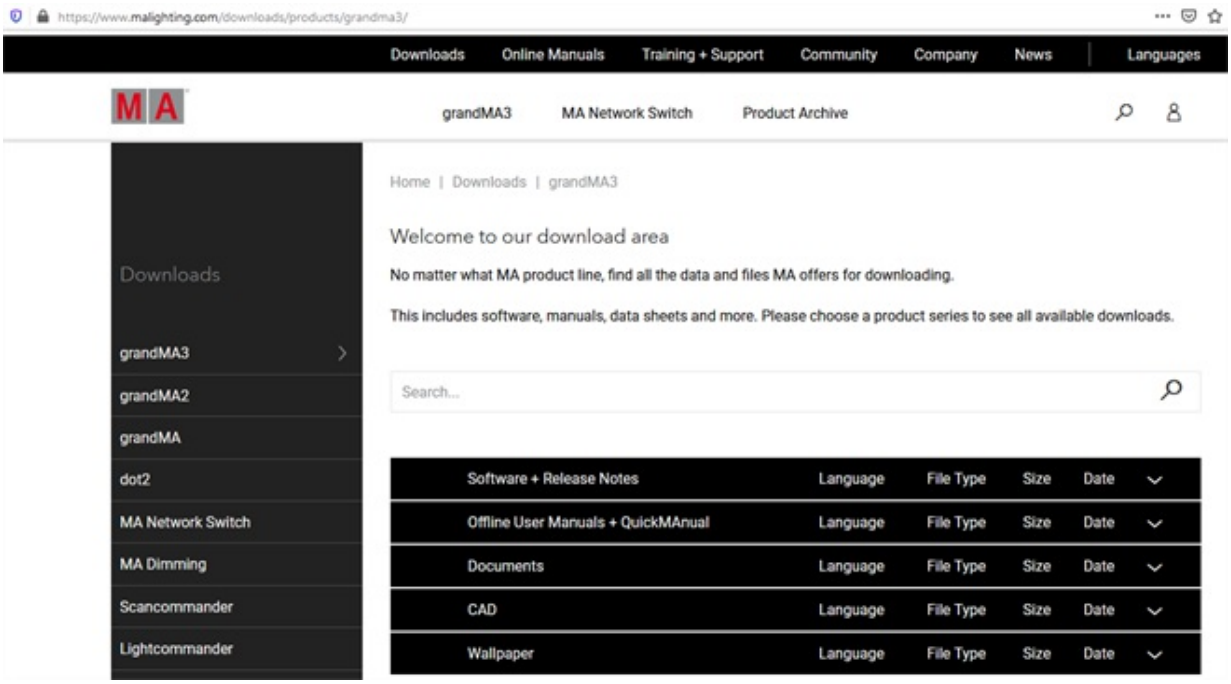
To download the grandMA3 software:

1. Go to **www.malighting.com**, click **Downloads.**



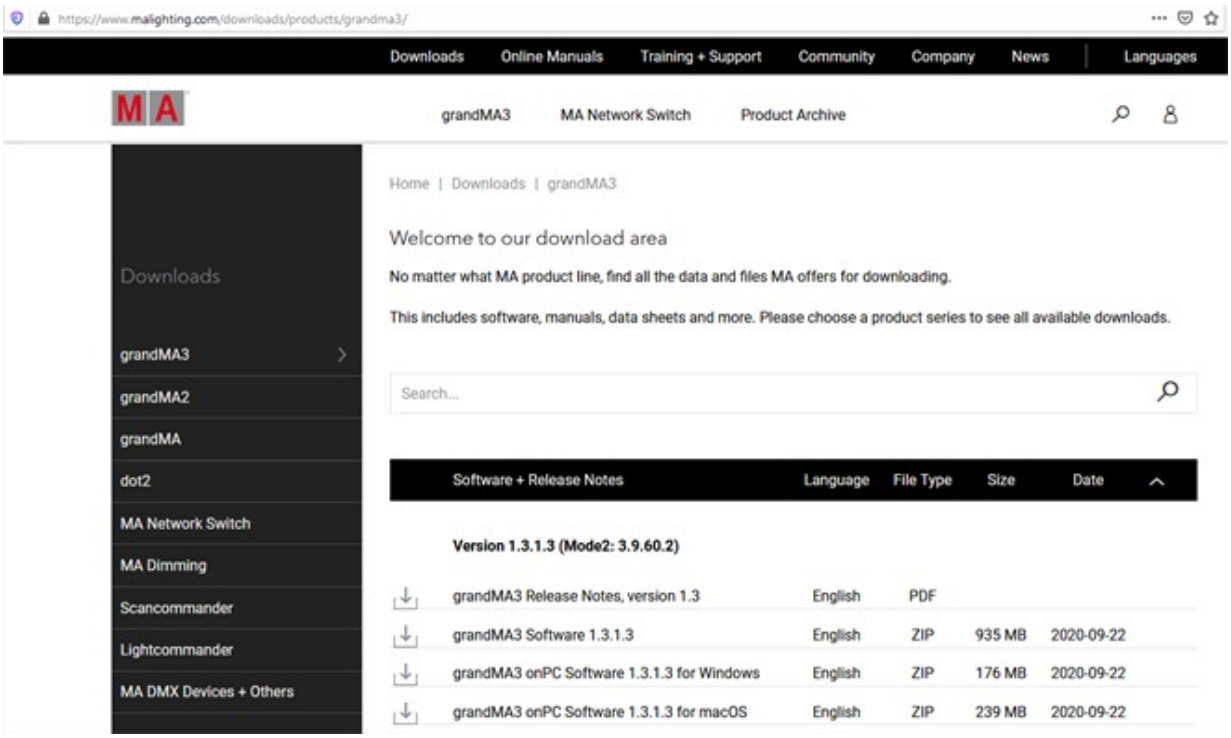
Website malighting.com Downloads

2. Click **grandMA3** in the bar on the left.



grandMA3 section

3. Click **Software + Release Notes** to find the latest software version.

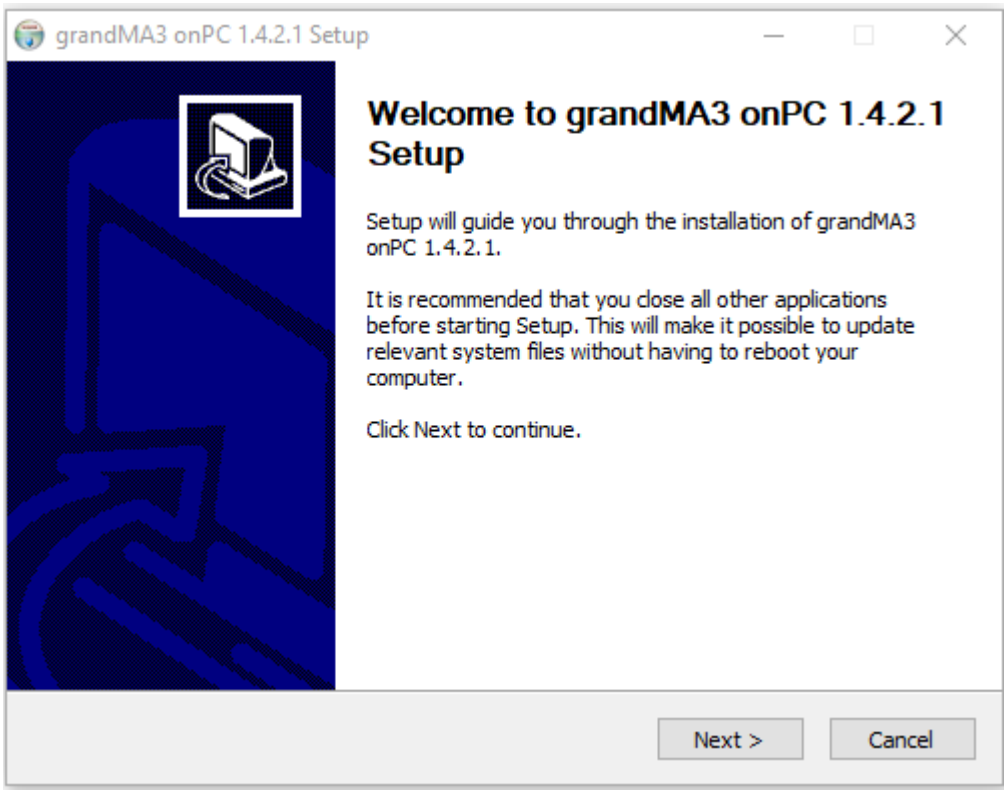


grandMA3 Software section

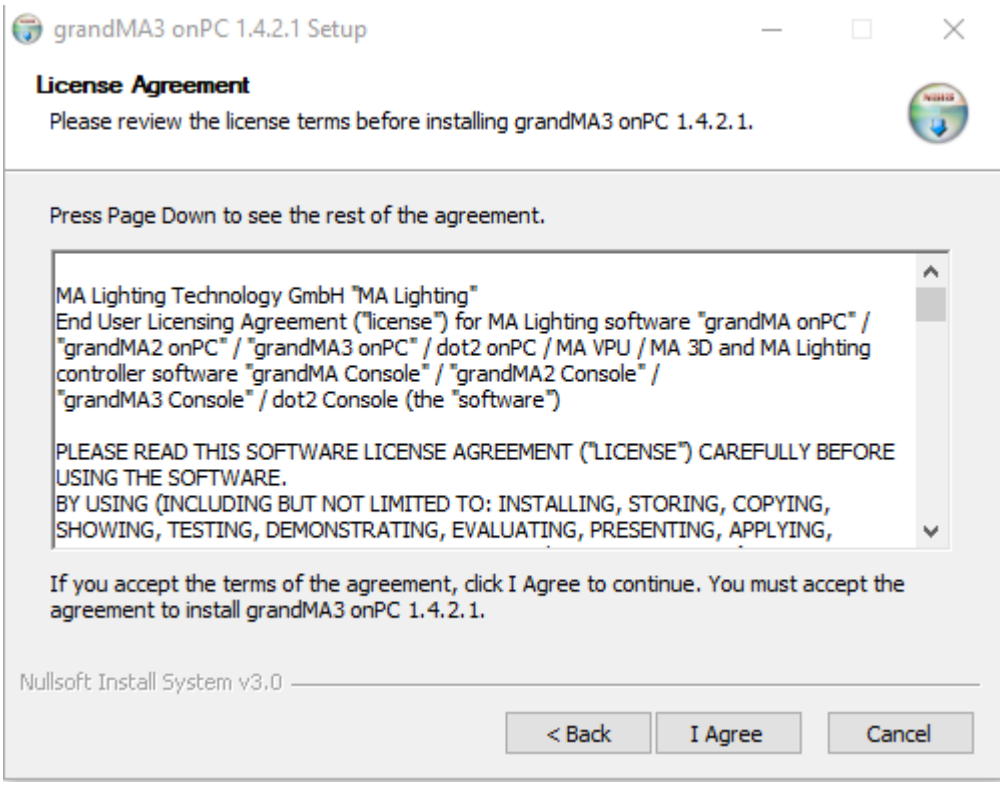
4. Click grandMA3 onPC Software x.x.x.x for Windows to download the desired installation package.

The download process starts.

- To open the zip file, double-click it.
- Go to the subfolder "ma".
- Double-click the installation program grandMA3_onPC-vx.x.x.x.exe.
- Confirm with Execute.
  The installation program opens:

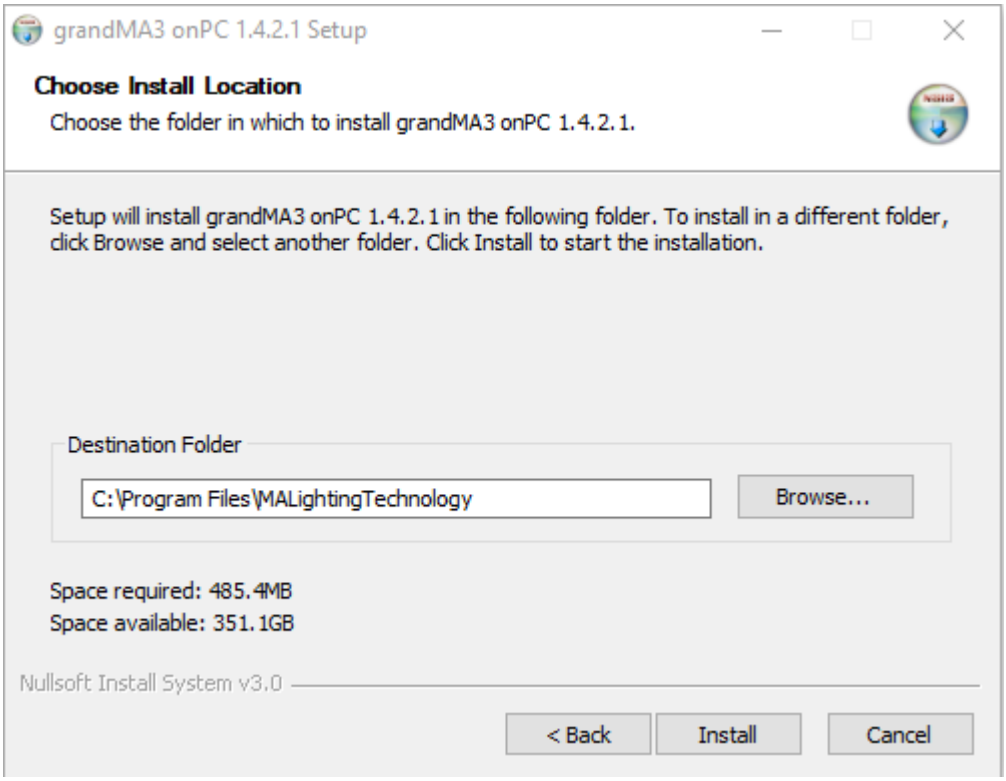


- Click Next. The End User License Agreement (EULA) appears:



- Click I Agree to accept the agreement.

The Install Location window appears:



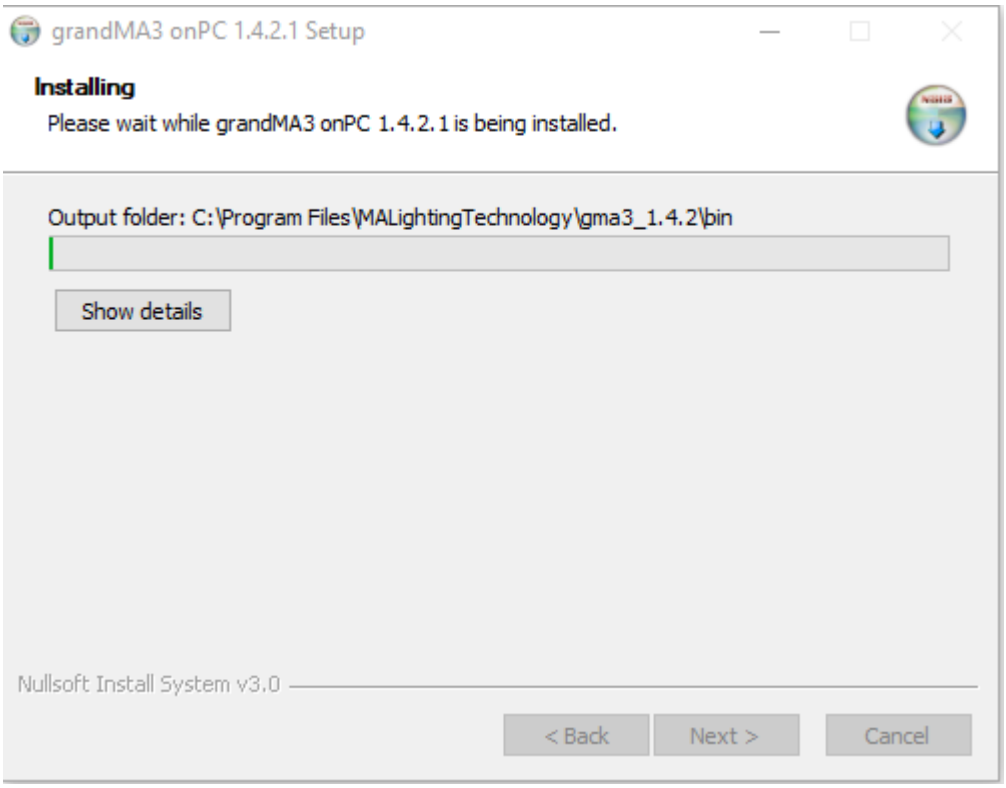
Watch out for the suggested directory and change it if you would like to do so.

The standard directory for the resources and user files is "C:\Program Files\MALightingTechnology".

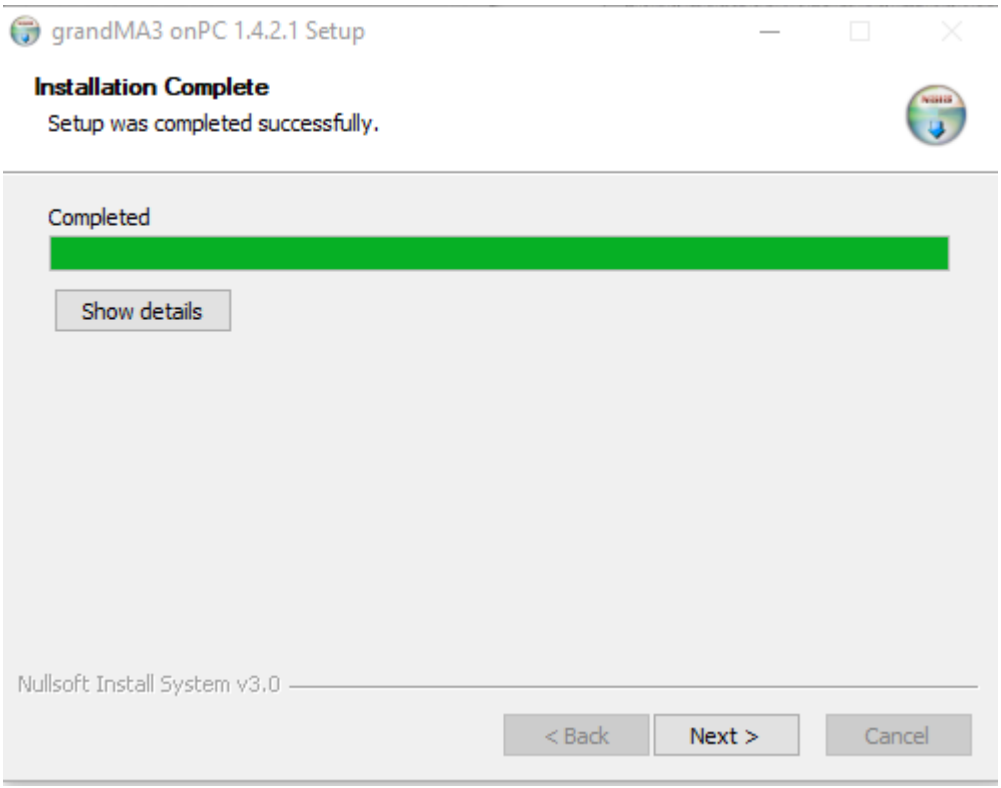
To learn more about the folder structure, read the **Folder Structure topic** in **File Management**.

To confirm the installation directory, click Install.

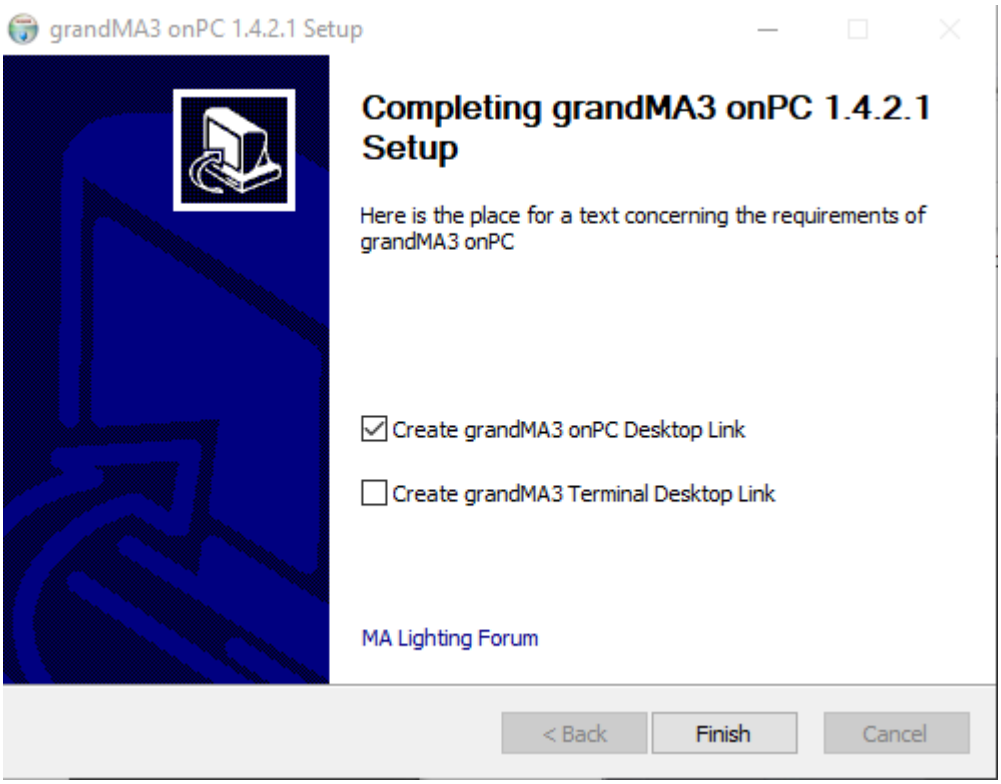
The program files are copied into the selected directory.



Click Next to finish the installation.

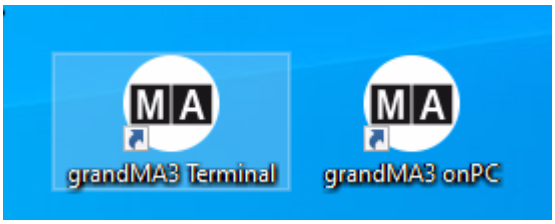


At the end of the installation process, you can choose to set a link to the terminal app or not.



It is disabled by default.

Click the checkbox if you want to create a desktop link to the terminal app.



grandMA3 onPC for Windows

As soon as the installation is finished, the MA icon appears on the desktop.

If several versions of the software have been installed, several subfolders are created in the main folder "C:\ProgramData\MALightingTechnology".

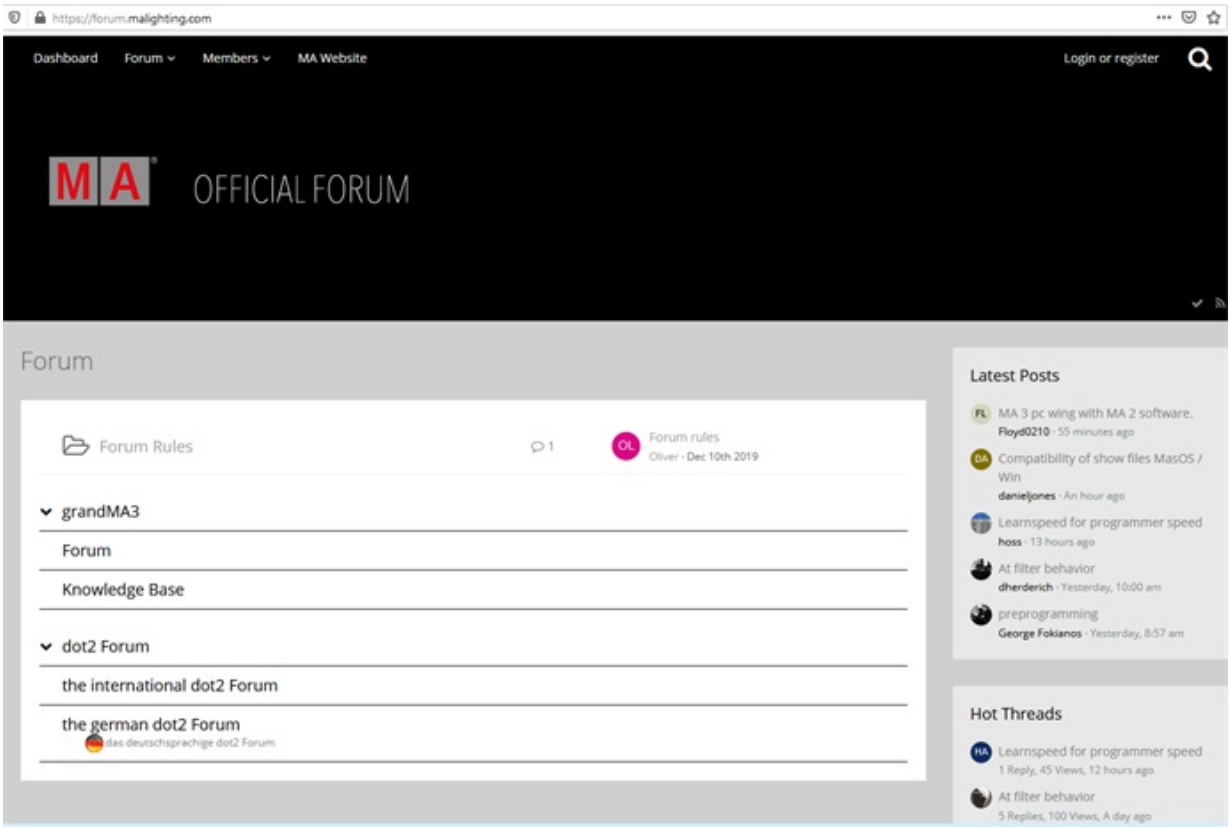
The MA icon on the desktop will always link to the latest version:

This is the main app.

MA Lighting Forum

- To visit the MA Lighting Forum on the internet, click **MA Lighting Forum**.

The MA Lighting Forum opens in the browser:



5.3. Optimize Windows

The grandMA3 onPC software is available for Windows™® and **macOS™®**.

Please refer to the **system requirements topic** to learn more about the computer specifications.

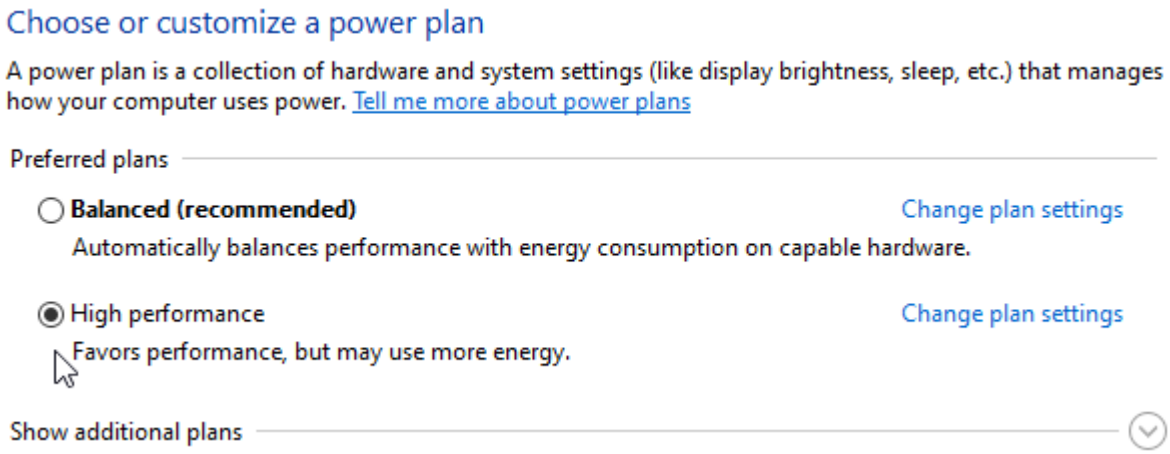
For more information about the grandMA3 onPC app, the terminal app, and the different version icons, read the **Installation of grandMA3 onPC topic**.

To run the grandMA3 onPC software even more efficiently, we recommend adjusting the following settings on your computer.

Turn on High-Performance Power Plan

To turn on the high-performance power plan:

- Click the **Start** button, and then click **Control Panel**. In the search box, type power options, and then click **Power Options**. Under Preferred plans, click **High performance**.



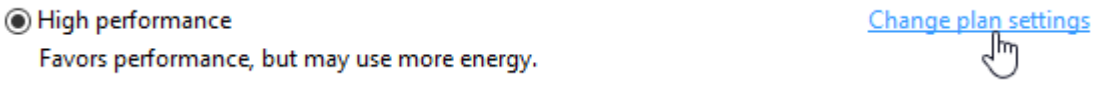
The high-performance power plan is turned on.

Optimize High-Performance Power Options

Turn off everything that can interrupt the grandMA3 onPC.

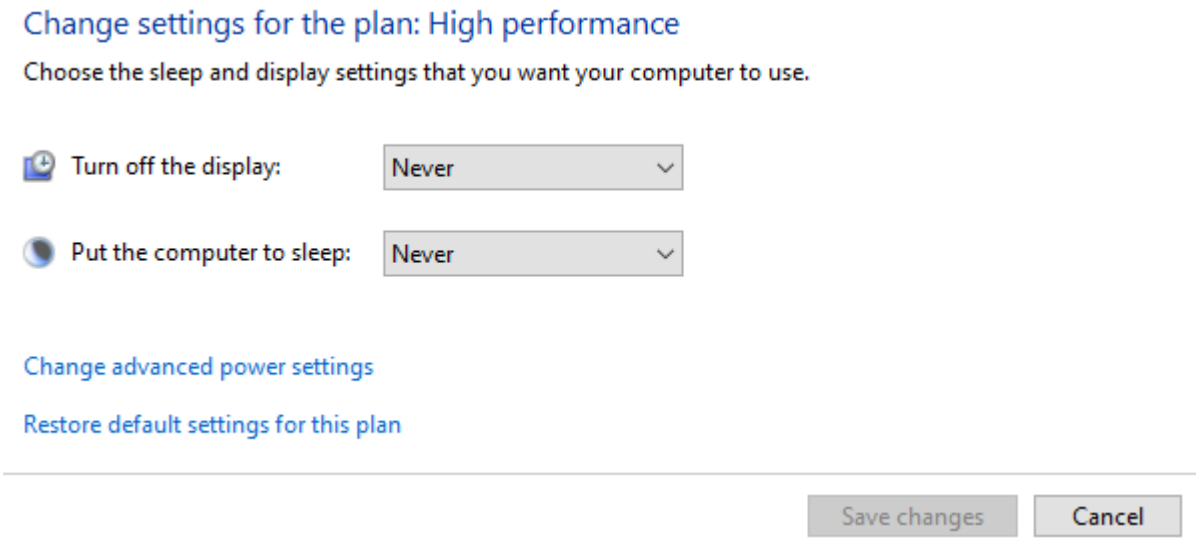
To optimize the power options:

1. Click **Change plan settings**.

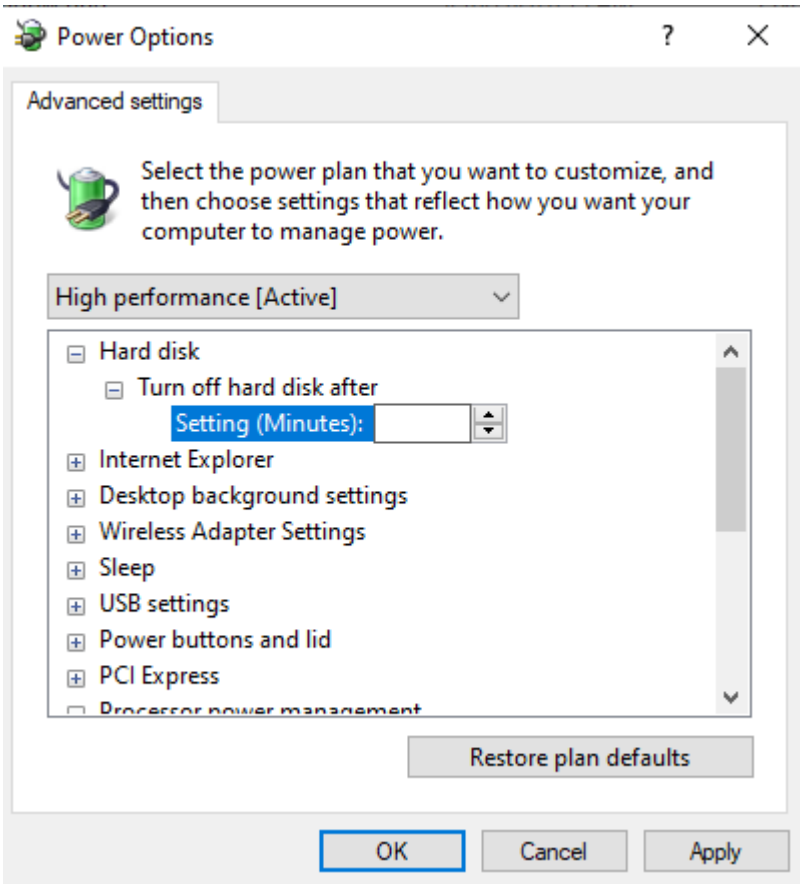


2. For Turn off the display, choose **Never**.

3. For Put the computer to sleep, choose **Never**.



4. Click **Save changes**.
5. Click **Change advance power settings**.



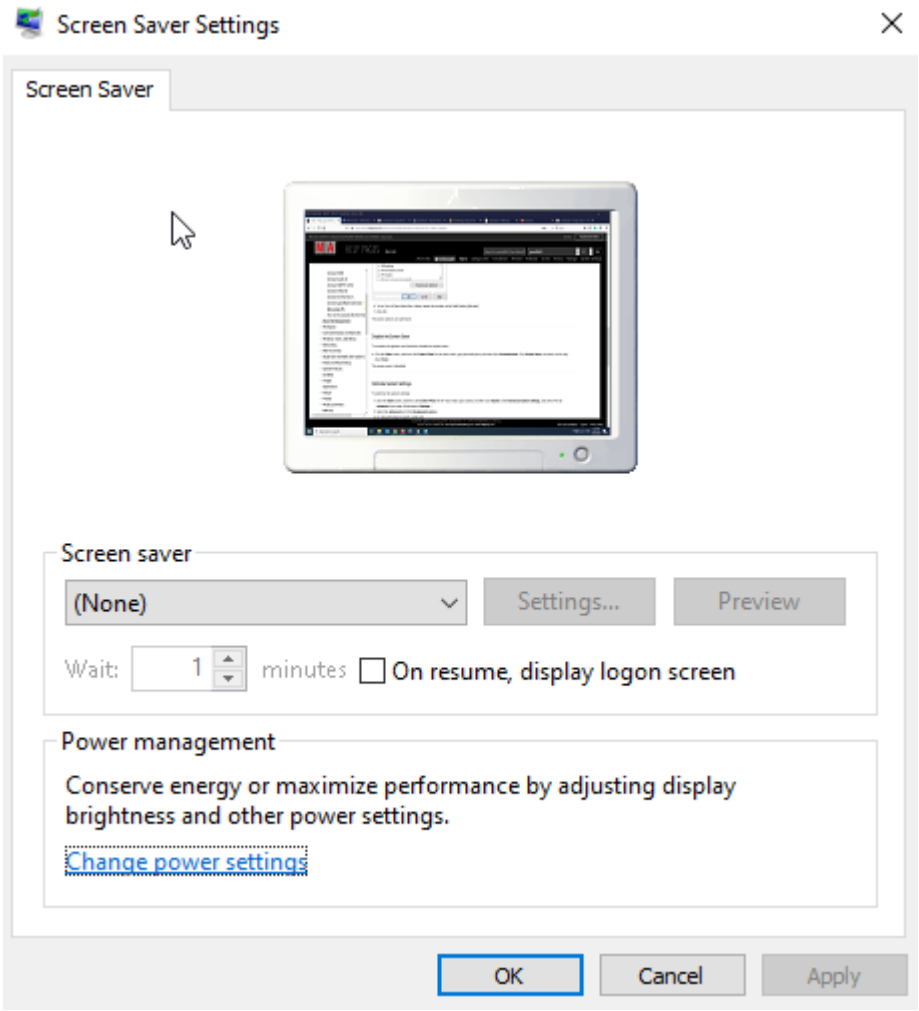
6. To set Turn off hard disk after to Never, delete the number in the field Setting (Minutes).
7. Click OK.

The power options are optimized.

Disable the Screen Saver

To conserve the graphic card resources, disable the screen saver.

- Click the **Start** button, and then click **Control Panel**. In the search box, type **Screen Saver**. Click **Screen Saver**, and select in the drop-down **None**.

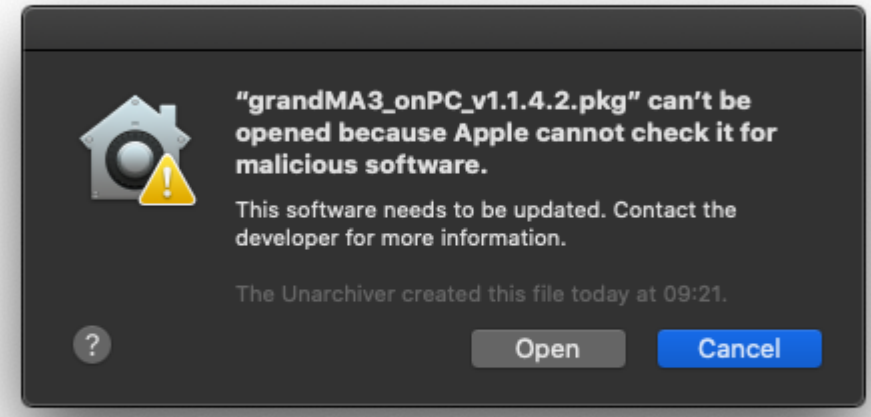


The screen saver is disabled.

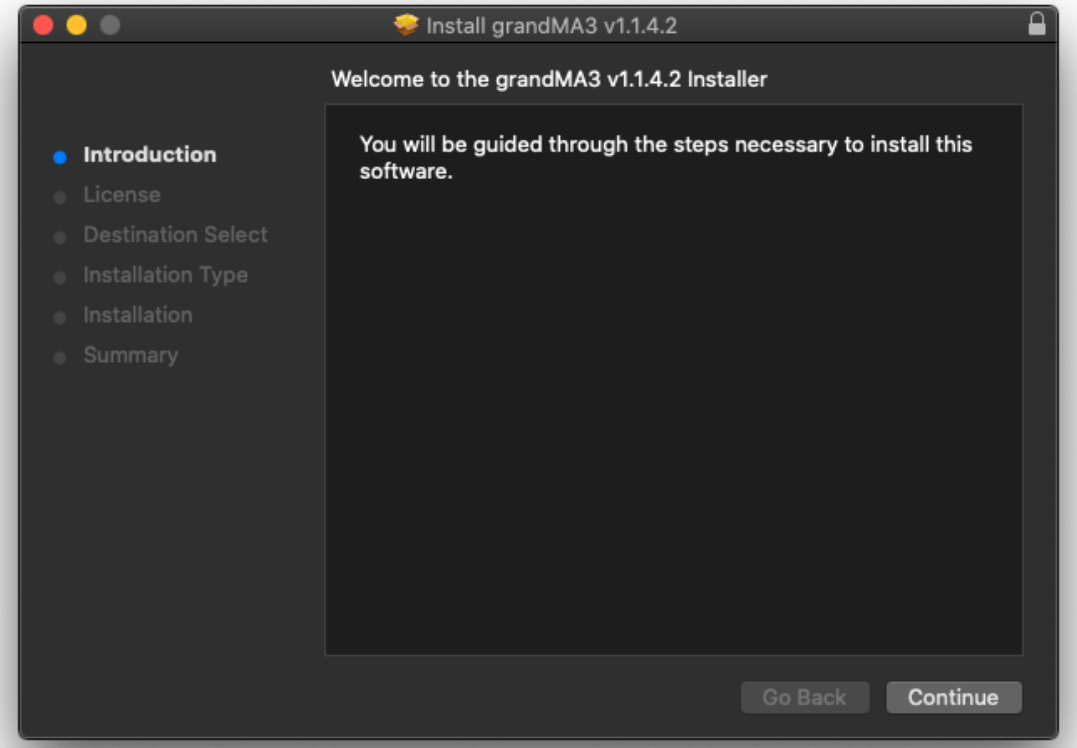
5.4. macOS installation

To run the grandMA3 onPC software, copy and install the program files on your computer.

1. To download the installation file, go to **www.malighting.com**.
2. Click grandMA3 onPC Software x.x.x.x for macOS to download the installer.
3. To open the zip file, double-click it.
4. Go to the subfolder "ma".
5. Right-click the installation program grandMA3_onPC_x.x.x.x.pkg.
6. Click Open when this warning message pops up:



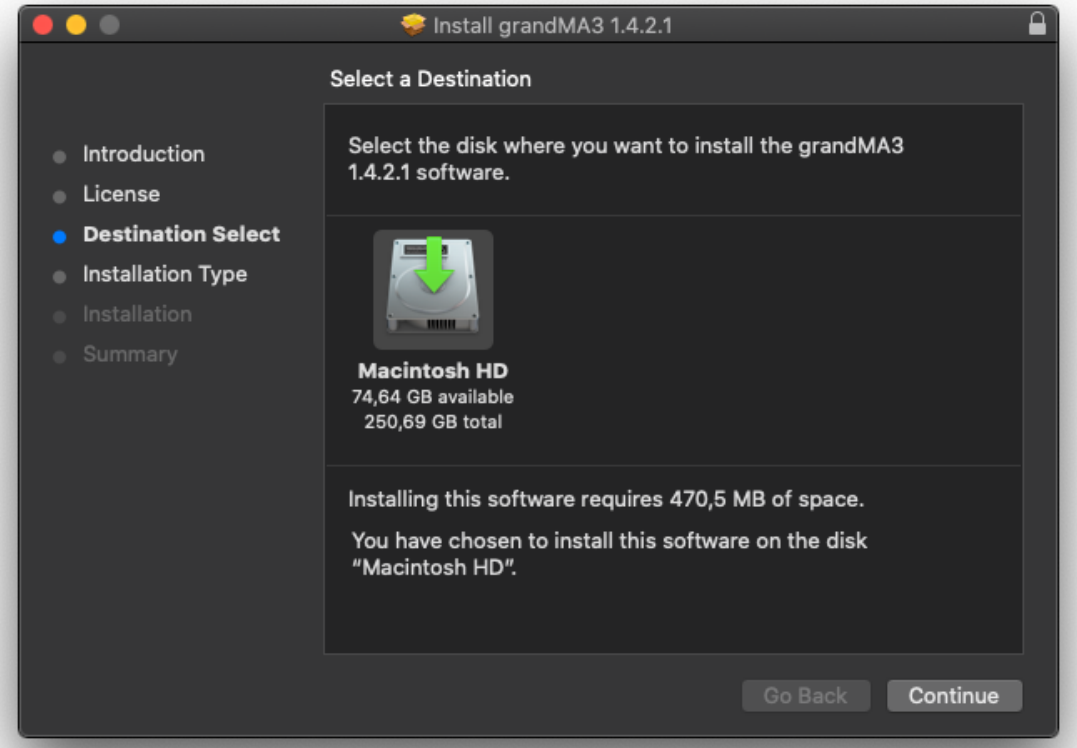
7. Click Continue when the installation program opens:



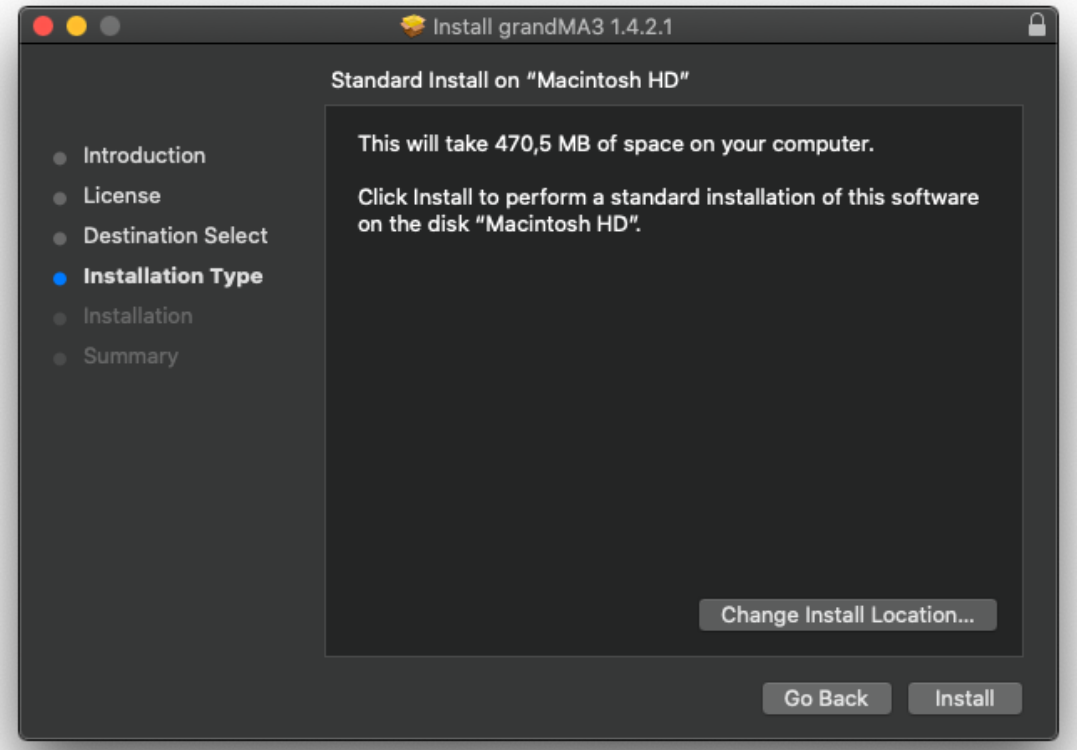
8. The software license agreement opens. You need to Click on Continue and then on Agree.

The software is installed in the standard directory /Applications/. The resources are located in /Users/.

- To change the destination disk, select the desired disk and click Continue:



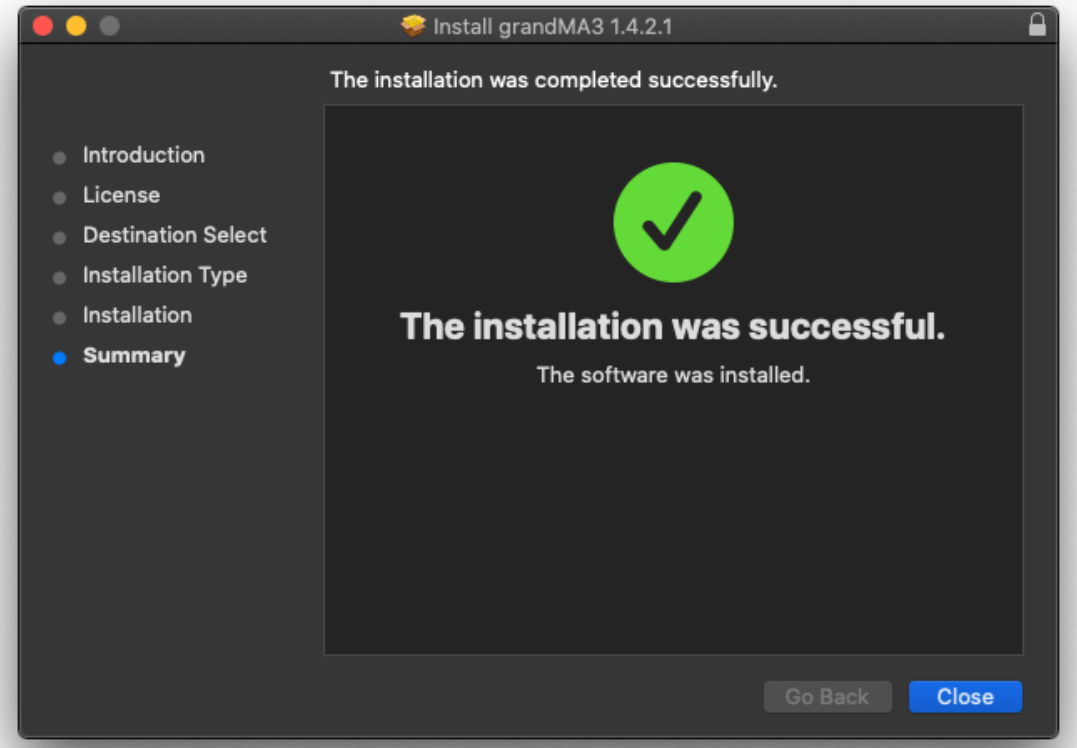
- Click Install to install the application or Change Install Location to change the destination:



- To install the software, click Install.

During the installation, it is possible that you are asked for the administrator password.

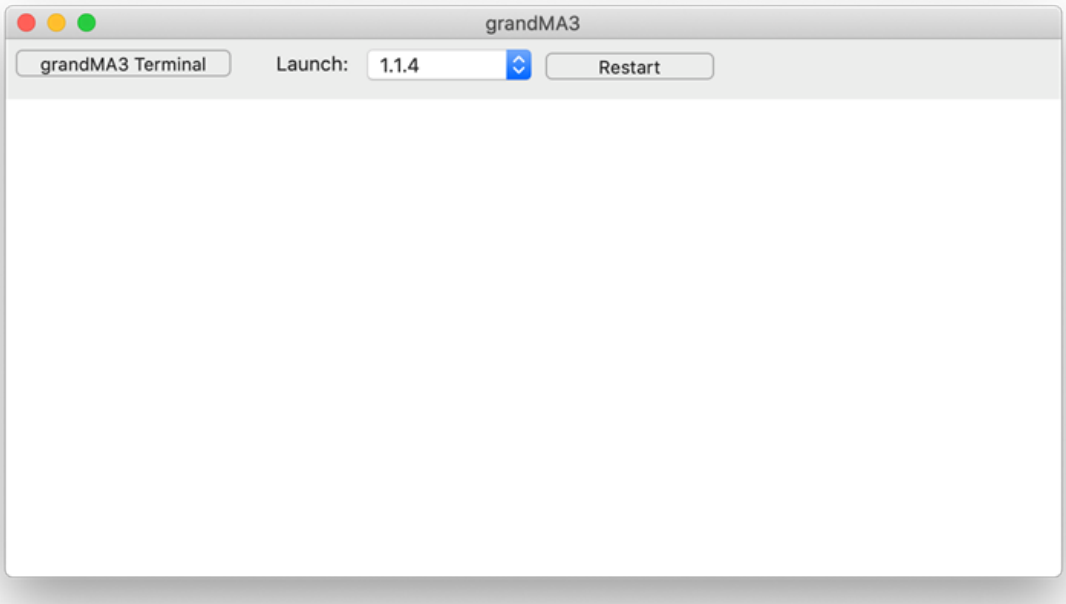
- Click Close after the installation is complete:



Go to Applications in the finder and double-click grandMA3.

The grandMA3 launcher starts.

The launcher will automatically start the last installed version.



To choose another version, select the desired version via the launch dropdown.

Click Restart in order to start the selected version.

Confirm the shutdown in the running application.

5.5. Optimize macOS

The grandMA3 onPC software is available for macOS™® and **Windows™®**.

Please refer to the **system requirements topic** to learn more about the computer specifications.

For more information about the grandMA3 onPC software and the terminal app, read the **Installation of grandMA3 onPC topic**.

To run the grandMA3 onPC software even more efficiently, we recommend adjusting the following settings on your computer.

Optimize Power Options

Click Energy Saver in System Preferences.

Settings for Battery

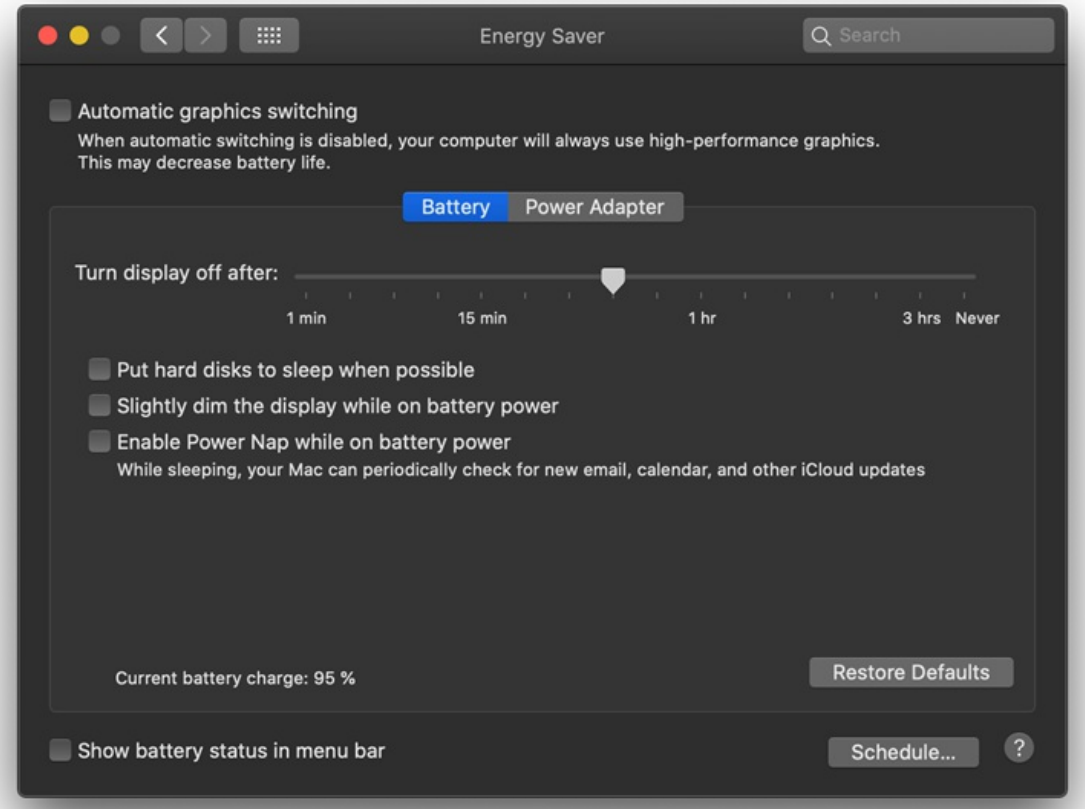
Automatic graphics switching: Off

Turn display off after: Any time.

Put hard disks to sleep when possible: Off

Slightly dim the display while on battery power: Off

Enable power nap while on battery power: Off



Hint:
When working with grandMA3 onPC on macOS by using the battery only, high-performance settings would also be required.
We recommend to always connect the computer with the power adapter.

Settings for Power Adapter

Automatic graphics switching: Off

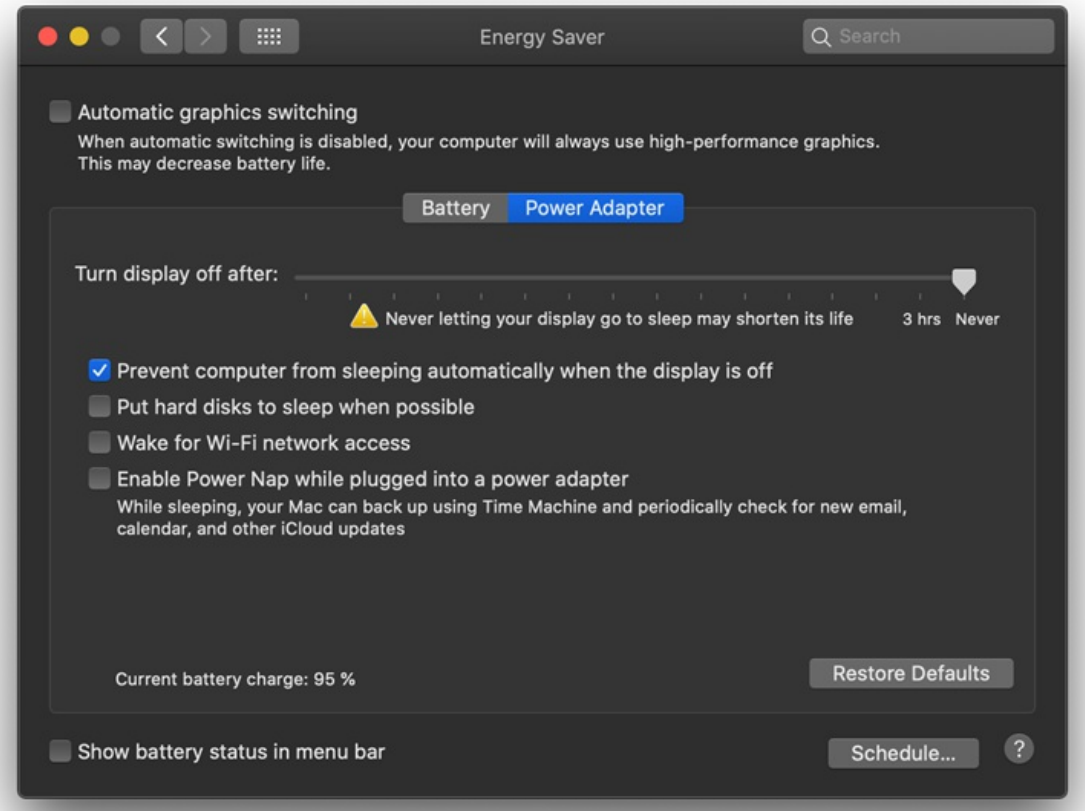
Turn display off after: Never.

Prevent computer from sleeping automatically when the display is off: On

Put hard disks to sleep when possible: Off

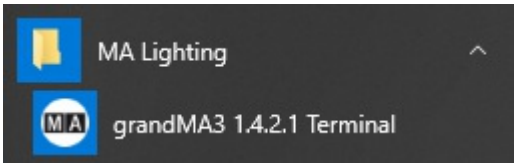
Wake for Wi-Fi network access: Off

Enable power nap while plugged into a power adapter: Off

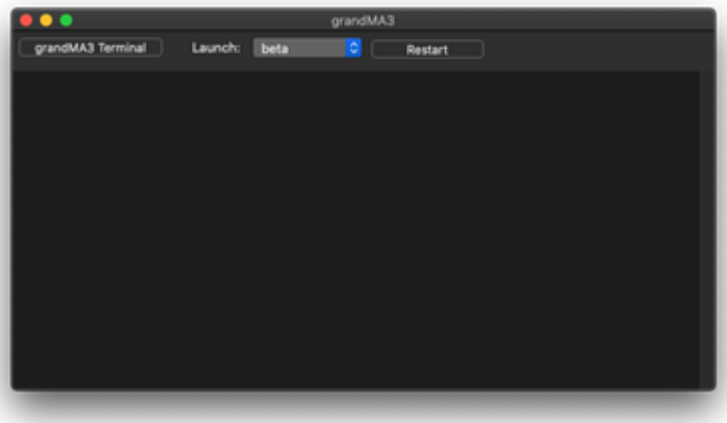


5.6. grandMA3 onPC Terminal App

In Windows®, the grandMA3 Terminal app can be found in the start menu in the folder MA Lighting:



In macOS®, click grandMA3 Terminal to start the grandMA3 Terminal:



The terminal app can be used to connect to the same station or any other grandMA3 station inside your network with the command line or the system monitor of the main app using the Terminal.

To monitor the software of a connected network station, start the terminal app.

To connect to the system monitor of a station in the network, enter: sysmon [IP Address].

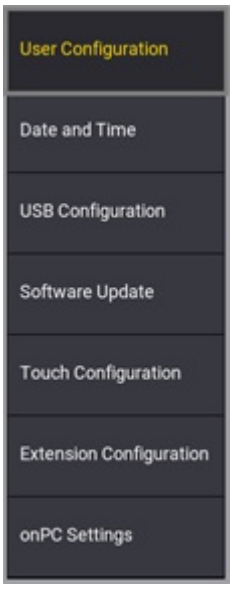
5.7. onPC Settings

The onPC settings menu enables different settings for onPC, for example set the Audio In Device or MIDI control.

To learn more about MIDI, read the topics **Connect MIDI** in **First Steps**, and **MIDI Remotes** in **Remote In and Out**.

- To open the **Menu pop-up**, click on the control bar.
- Click Settings.

The Settings pop-up opens:



Settings pop-up

- Click onPC Settings.

The onPC Settings menu opens:

onPC Settings menu

MIDI

- To enable or disable MIDI from onPC Command Wing, click MIDI via OnPC command Wing.
  If disabled, the device configured in MIDI In Device and MIDI Out Device is used.
- To select a MIDI In device, click MIDI In Device.
- To select a MIDI Out device, click MIDI Out Device.
- To adjust the MIDI Offset, click MIDI Offset.
- To adjust the MIDI Timecode Slot, click MIDI Timecode Slot.

Sound

- To select the Audio In device, click Audio In Device.

6. Show File Handling

The Show File

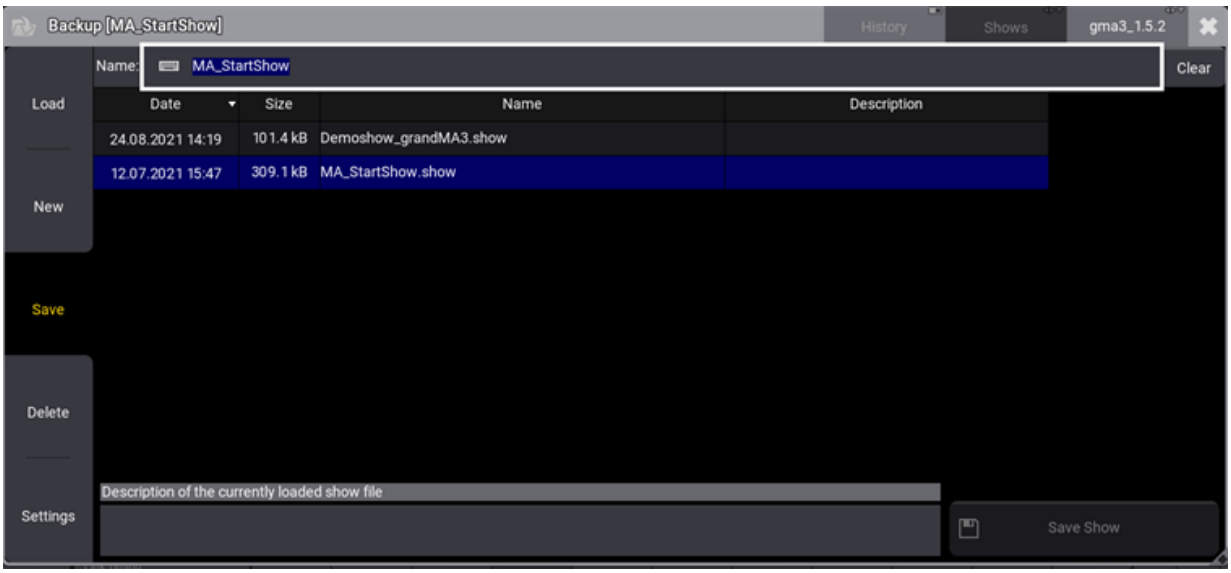
The show file contains all the information that is related to the show. That means the patch, fixture profiles, cues, timings, the 3D information, etc. The show file also contains the users and user profiles.

Important:
Remember to save the old show before loading or renaming a new one.

A show file software version can only be moved forward. A show saved on a USB stick that was programmed in a previous version can be loaded in a newer version. If the show is then re-saved on the stick in this new version, it cannot be taken back to the old software. If this is important, then it is needed to load the show and save it with a new name. Then the old version is not saved as the new version and it can still be accessed by an older version.

Hint:
To learn more about the folder structure of shows, demo shows, and backups read the **File Management** topic.

All the management of the show files is handled in the Backup menu. To access the Backup menu, press Menu, and tap Backup. This opens the Backup menu which could look like this:



Backup menu

The Backup menu can also be called by the command line with the **Menu command**:

```
User name[fixture]> Menu "Backup"
```

Important:
The size of a show file is limited to a maximum of 10 GB when it is loaded.

Important:
The overall size of the media pools is limited to a maximum of 200 MB.

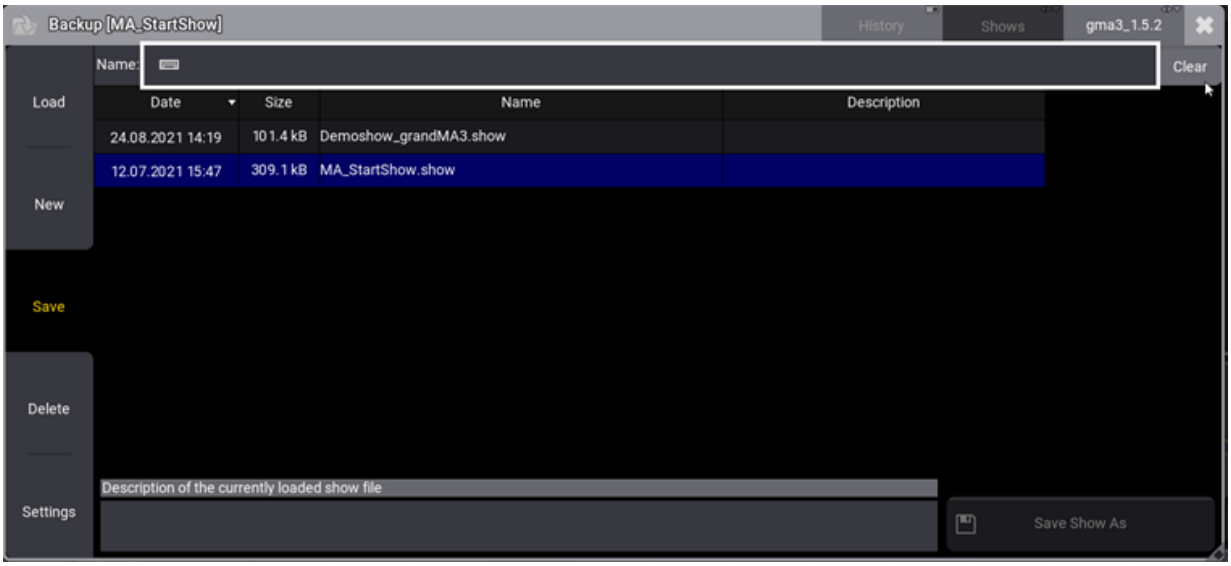
Important:
The show file name is limited to a maximum of 31 characters.

Important:
The show file name must not contain the following characters: \ " $ * ? ^ | / : < > `

Save a Show File

The Backup menu starts with the Save dialog. The show file is saved with the current name and with the timestamp of the current session time. The resulting files on the hard drive of the system will be also stamped using the current session time.

To clear the Name field, tap Clear in the upper right corner:



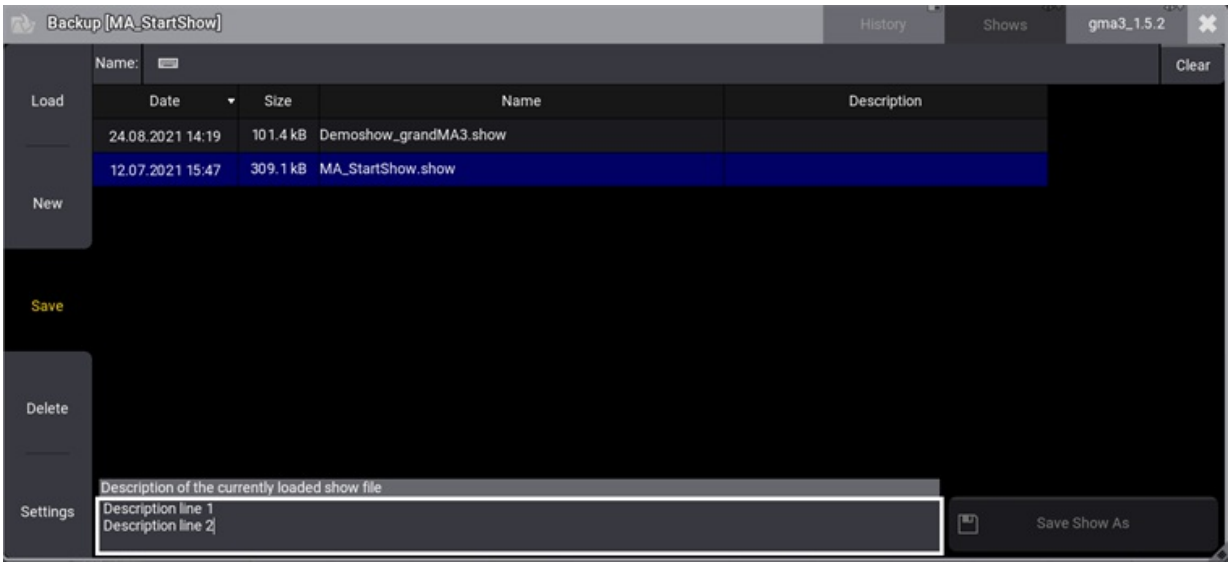
Clear the name field

To save a show file with a new name, type the name in the Name field.

To save a show by the command line, use the **SaveShow keyword**.

Add a Description

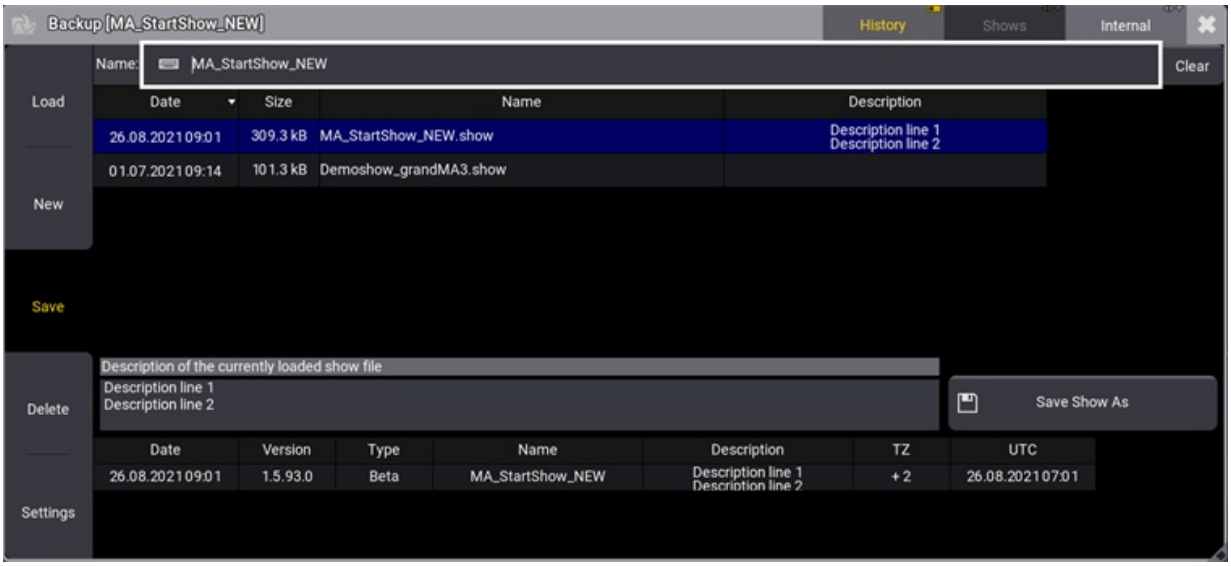
At the bottom of the backup menu, an area allows you to enter and read the description of the currently loaded show file. Modifying or entering a description will only be applied to the show file when saving it afterward.
Pressing Enter while editing the description immediately saves and closes the backup menu. Shift + Enter can be used to add another line.



Show file name with description lines

Show File History

By enabling History in the title bar of the backup menu, the history of the selected show file will be displayed at the bottom of the backup menu.
The history gives useful information regarding the dates and time, used software version, the type of the software, the name, time zone, and UTC time when saving the show file.
A new entry within the history will be entered when the show file will be saved again and one of these criteria has changed: Version, type, name, time zone, or description. The newest entry will be on top of the history list.



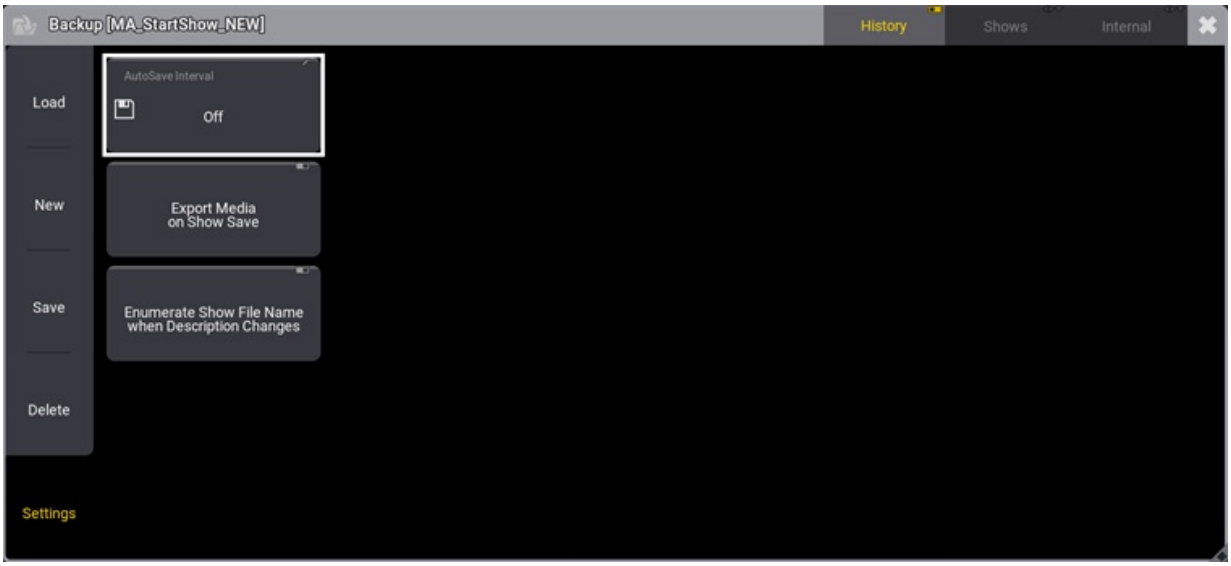
Show file name with description and time stamp

Add Automatic Numbering

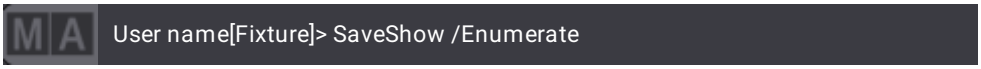
It is possible to add 3-digit numbers to the show file name.

In order to separate the show file name from the number, the user can add a hyphen (-) or an underscore (_) or any other character to separate the name from the numbers at the end of the show file name.

The option Enumerate Show File Name when Description Changes in the Settings of the backup menu provides an automatic increase of a 3-digit number, added to the end of the show file name if the description is changed when saving the show.



If the original show file name is too long to add the number, the command line will return an error. In this case, the former show file name will be used again.
Enumerating a show file can also be called with the /Enumerate option. To do so, type:



Select the Drive

The show files are stored on the internal drive or a USB stick. The internal drive is Drive 1 and the USB stick is Drive 2. If more than one USB stick is connected the order of the connection selects the drive number. That means that the latest connected drive has the highest number. When a USB stick is disconnected, the drive numbers can change.

In the upper right corner of the Backup menu, it is possible to select the drive you are working on. Tap Internal and it toggles between the different drives connected or shows saved in older software versions. Tap and swipe right on the drive and a little pop-up opens with the different opportunities.

The drive can also be selected by the command line.

Hint:
Saving a show on a USB stick automatically saves the show on the internal drive as well.

New Show

- To create a new empty show file, tap New.
  A new show is created with an automized file name like New Show-date-time.
- To rename the show file, open the Backup menu again.
- Enter a file name for the new show in the Name field. When the name is correct, tap Save.

To create a new show by the command line, use the **LoadShow keyword**.

Hint:
It is possible to use local letters in the filename, like æ, õ, and ä.

Load a Show File

Open the Backup menu and tap the show file needed. Tap Load and the selected show file appears.

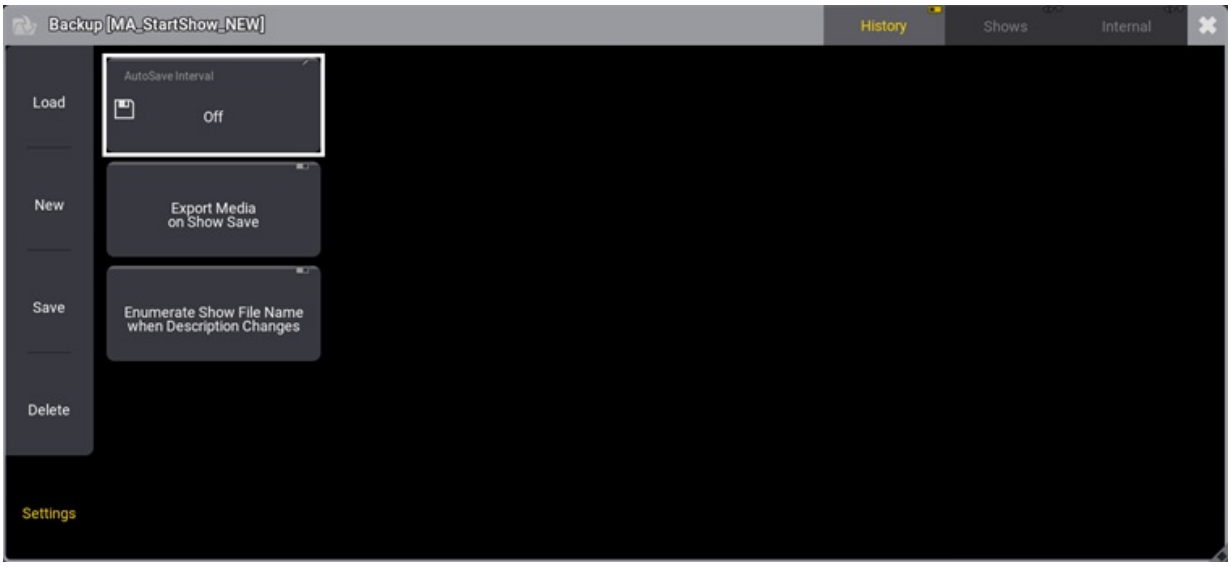
To load a show by the command line, use the **LoadShow keyword**.

Quick Save

It is possible to make a quick save of the show file. Press Menu and tap Quick Save. Pressing 2 x Menu also makes a Quick Backup.

Auto Save

In order to save the show automatically in regular intervals, activate the Auto Save function. Tap Settings and AutoSave Interval.



Select the desired Auto Save interval.

Save Show As

To save a show with another name, open the Backup Menu and write the new name of the show file in the Name field. Tap Save and the show file is saved by its new name.

Delete a Show File

Open the Backup Menu and tap the show file to delete. Tap Delete and the show file is deleted.

Backup Files and Demo Shows

Open the Backup Menu and tap Shows. This toggles between shows, backups, and demo shows. Here it is possible to load a previously saved backup or a demo show. It is not possible to save or create a new show file in the backup or demo show folder. It is also impossible to delete demo shows.

Load a Show File During Startup

It is possible to specify a show file or a plugin that shall be loaded upon starting the onPC application.

To do so, add the SHOW or RUNPLUGIN parameter to the shortcut that launches the onPC application:

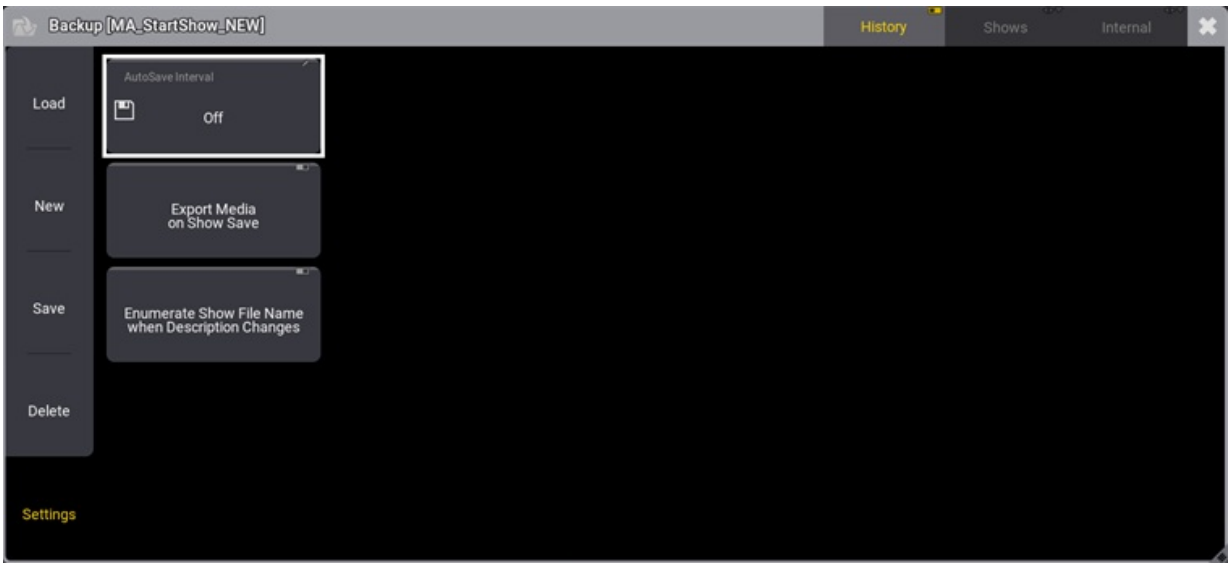
- "C:\Program Files\MALightingTechnology\gma3_xx\bin\app_system.exe" HOSTTYPE=onPC SHOW="startshow.show"
- "C:\Program Files\MALightingTechnology\gma3_xx\bin\app_system.exe" HOSTTYPE=onPC RUNPLUGIN="startplugin.xml"-x
  - x represents the number of the LUA component inside the plugin, that shall be started, e.g., 1
  - It is not recommend to specify a path for the plugin although the plugin may be put into a subfolder.

In addition, the optional parameters NOLOAD and CLEANSTART are also available when starting onPC.

- NOLOAD will not load the last show file but start with an empty show. Other device related configurations are kept.
- CLEANSTART will reset device related configuration back to default and come up with an empty show file.

Export Media on Show Save

In order to save media files (e.g. images, videos, etc.) to the **gma3_library folder**, activate the Export Media on Show Save function. Tap Settings and Export Media on Show Save.



7. Workspace

In general, **Workspace** deals with the visual elements on screens and input modes such as:

- Windows
- Views
- Coloring
- Executor bar
- Encoder bar
- Gestures
- Keyboard shortcuts

It represents the usage of the working area in the grandMA3 console.

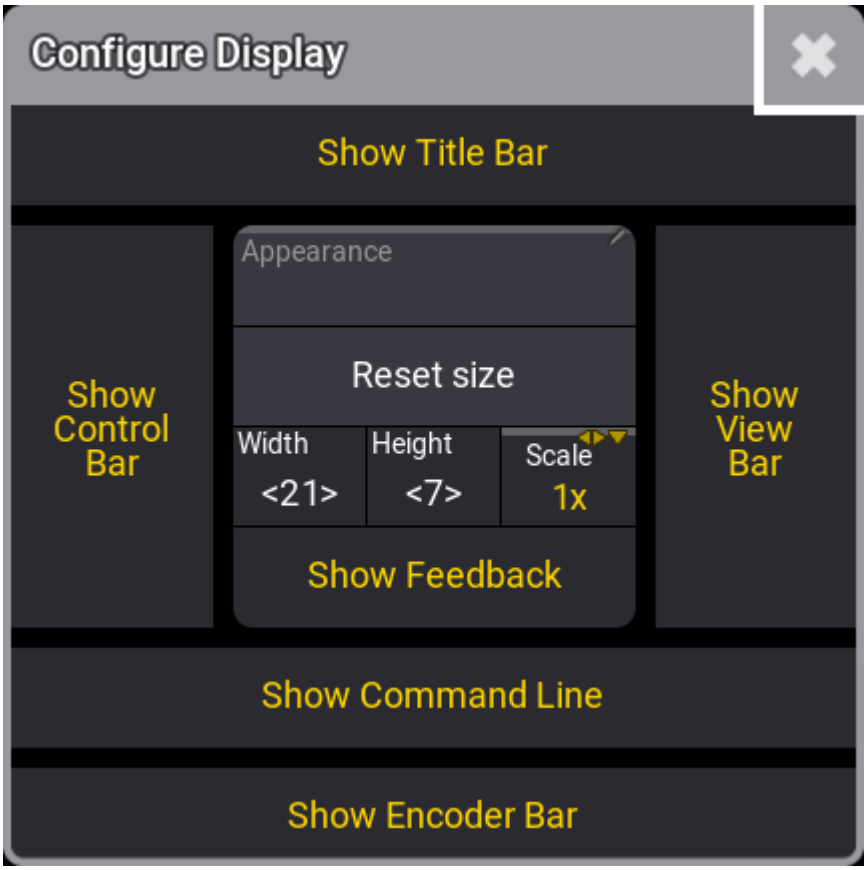
7.1. User Interface

This section covers the control elements that help you operate and control the grandMA3 in an efficient and effective way.

7.1.1. Configuration of Displays

The displays are populated with a combination of a user-defined area and a collection of additional display elements. The additional elements offer quick access to many helpful features, and their availability differs from screen to screen.

Pressing Menu then tapping Configure Display on any screen (excluding screen 8, screen 9, screen 10, screen 11, and screen 12) opens the Configure Display dialog for that screen. This dialog can also be opened by tapping any user-defined area on the desired screen and tapping Configure in the upper-right corner of the Add Window dialog.



Tap areas to activate or deactivate them

This pop-up is used to configure or change the appearance of the display. It has different areas:

- **Show Title Bar**
  Activate this button to see the window frame in the grandMA3 onPC. This is not an option in the MA hardware.
- **Show Control Bar**
  Activate this to see the control bar on the left side of the display.

- **Appearance**
  Tap this to open the Select Appearance pop-up and select one of the existing appearances for this display. The appearance is added as a background for the screen area. Read more in the **Appearance section**.
- **Reset size**
  Tap this to reset the size of the user-defined area. The size can be changed using the **Width** and **Height**. Allowing each user to set the size of the user-defined area in the display. A user-defined area bigger than the display will appear within a brown frame. A user-defined area that is smaller than the screen will gray out the unused area.
- **Scale**
  The scale can be used to change the scale of the interface. This option is only available on onPC and on external screens. It is very useful with high DPI screens or screens that have a lower resolution than Full HD.
- **Show Feedback**
  Tap to show or hide a small feedback overlay at the bottom of the display. This feedback appears briefly when the show is saved.
- **Show View Bar**
  This option hides or displays the column of view buttons on the right side of the display.
- **Show Command Line**
  Activate this to see the command line input bar at the bottom of the display.
- **Show Encoder Bar or Show Playback Bar**
  This hides or shows the encoder bar or the playback bar at the bottom of the display.

The in the title bar closes the pop-up.

These additional elements can be hidden to increase the amount of available user-defined area or shown in different combinations on most screens, whether internal screens or external monitors.

Screen 1 with additional display elements hidden, allowing more user-defined area.

Screen 1 with all additional display elements shown.

Command line with additional feedback overlay.

Screen 1, Screen 2, and Screen 3

Screen 1, screen 2, and screen 3 are the main screen workspaces on the console. The displays on these screens offer the most available user-defined area on any of the internal screens. They can be configured with a selection of additional display elements.

By default, screen 1 includes the command line and view bar. It can also be configured to display the control bar and encoder bar.

By default, screen 2 and screen 3 include the view bar. They can also be configured to display the command line, control bar, and playback bar.

Screen 4 and Screen 5

Screen 4 and screen 5 are on the optional external monitors. The displays on these screens offer a large amount of user-defined area. They can be configured with a selection of additional display elements. They can be configured to display the view bar, command line, control bar, and playback bar.

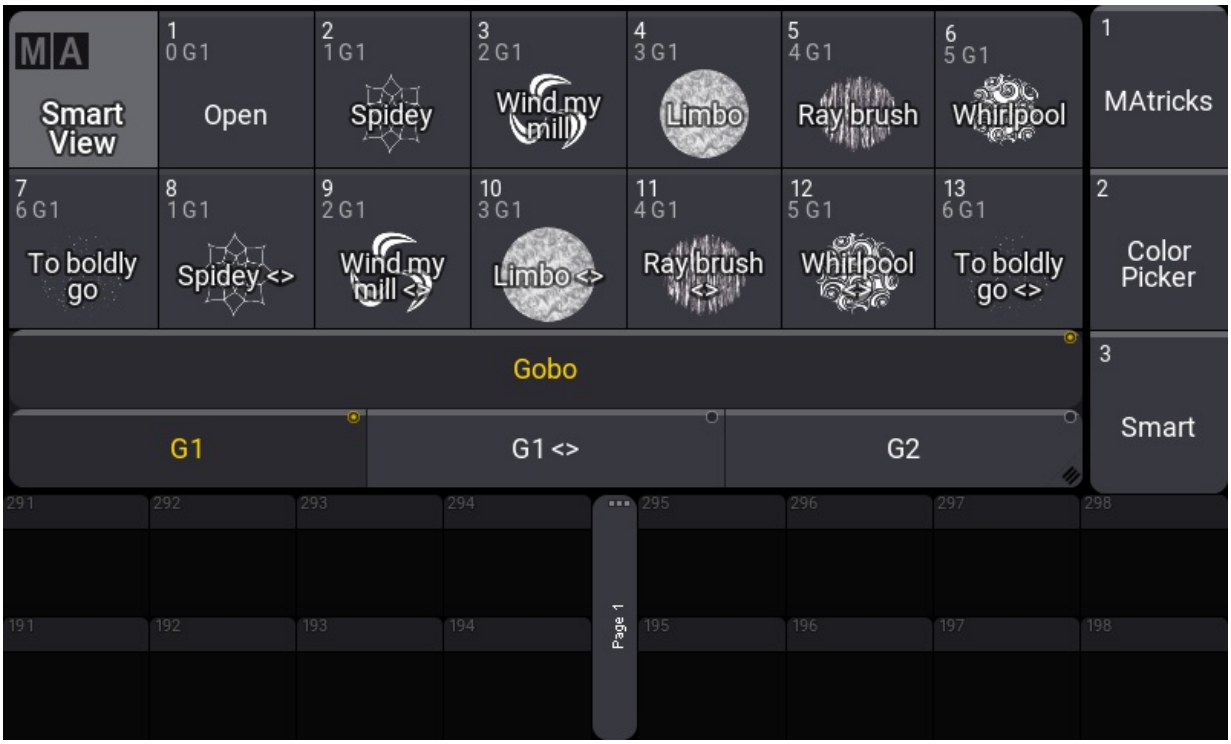
The scale option is available on screen 4 and screen 5 to take advantage of monitors of different resolutions.

Screen 6 and Screen 7

Screen 6 and screen 7 are the right command screen and left command screen, respectively. The displays on these screens offer a small amount of user-defined area. They can be configured with a selection of additional display elements.

Screen 6 can be configured to display the view bar, command line, and control bar.

Screen 7 can be configured to display the view bar, command line, control bar, and playback bar. The playback bar available on screen 7 shows the status of objects assigned to the Xkeys.



Screen 7 with view bar and playback bar visible, showing the assignments of the Xkeys.

Screen 8, Screen 9, Screen 10, Screen 11, and Screen 12

Screen 8, Screen 9, Screen 10, Screen 11, and Screen 12 are the letterbox screens, which are available on the grandMA3 full-size, light, and extension. The displays on these screens offer no user-defined area. The configurations of these displays cannot be changed. These screens are dedicated to specific display information and user input.

Screen 8 displays the encoder bar and grand master.

Screen 9 displays the playback bar and master section.

Screen 10, Screen 11, and Screen 12 display the playback bar and custom section encoders and wheels.

7.1.2. Desk lock

To lock the desk, use one of these options:

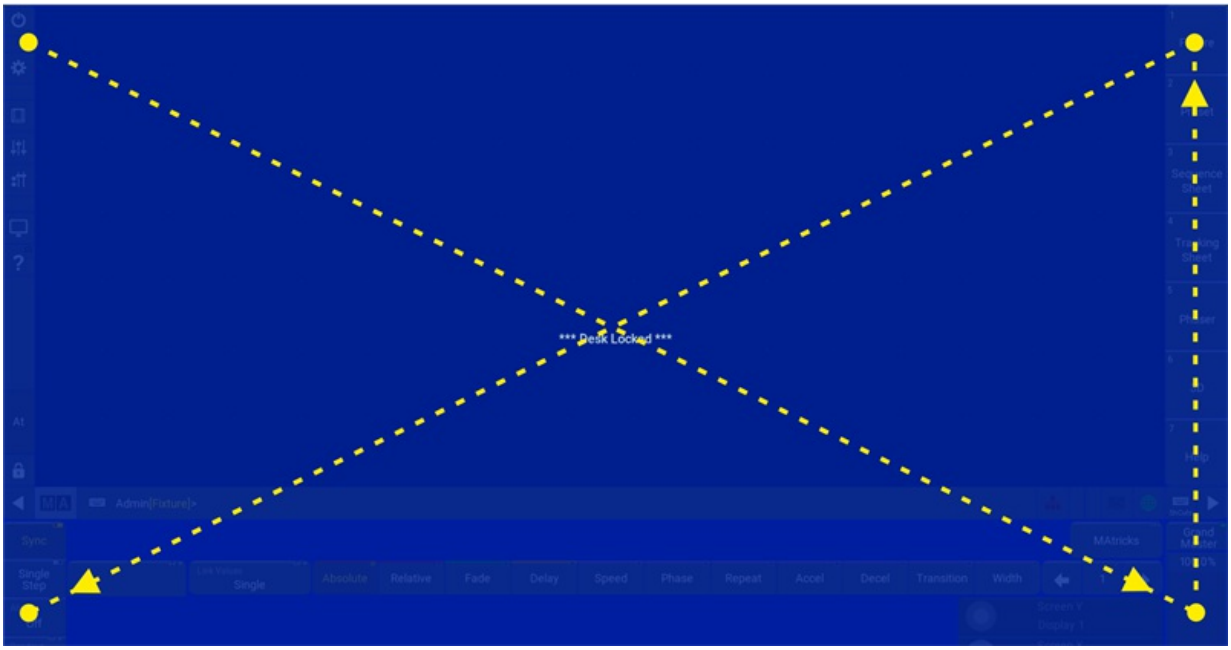
- Press MA + MA + Pause.
- Press Pause on the keyboard.
- Press F9 on the keyboard.
- Tap the icon at the top of the **control bar** to open the **shutdown menu**. Then tap the icon.

Desk Locked

The desk is locked

To unlock the desk, use one of these options:

- Press MA + MA + Pause.
- Press Pause on the keyboard.
- Press F9 on the keyboard.
- Tap the upper left corner, lower right corner, upper right corner, and lower left corner of any screen.



Touch pattern to unlock the desk.

The desk is unlocked.

7.1.3. User-Defined Area

The user-defined area is located on screens 1 through 7. It appears as a grid of small dots.

Empty user-defined area with a grid of small dots

This is the area where windows can be added and arranged to be stored and recalled as views. For information on how to manage windows in the user-defined area and views, see **Windows, Views, and Menus**.

The Configure Display menu allows adjustment of the grid size within the user-defined area. It is possible to configure the grid of the user-defined area to a size larger or smaller than the available display space. For more information on the Configure Display menu, see **Configuration of Displays**.

Grid of the user-defined area configured larger than the available display space

If the grid is larger than the available display space, a dark orange outline appears around the edge of the user-

defined area. Horizontal and vertical scroll bars appear, allowing access to the full space of the grid. The 3 finger scroll gesture also scrolls the user-defined area within the visible portion of the display. For more information on gestures, see the **Gestures** topic.

Grid of the user-defined area configured smaller than the available display space

If the grid is smaller than the available display space, a partially transparent gray overlay appears over the excess area. Any previously stored views that utilize this area will still recall as stored. The overlay does allow interaction with the contents of windows underneath, but windows cannot be resized or created outside of the user-defined area.

7.1.4. Command Line

The main purpose of the command line is to display active command syntax as it is entered and to display any errors generated by failed syntax. The command line also includes several helpful status indicators and quick-access buttons to frequently used features.

By default, the command line is located at the bottom of screen 1. It can also appear at the bottom of screens 2, 3, 4, 5, 6, and 7.

Command line as it appears on screen 7

Show or Hide the Control Bar

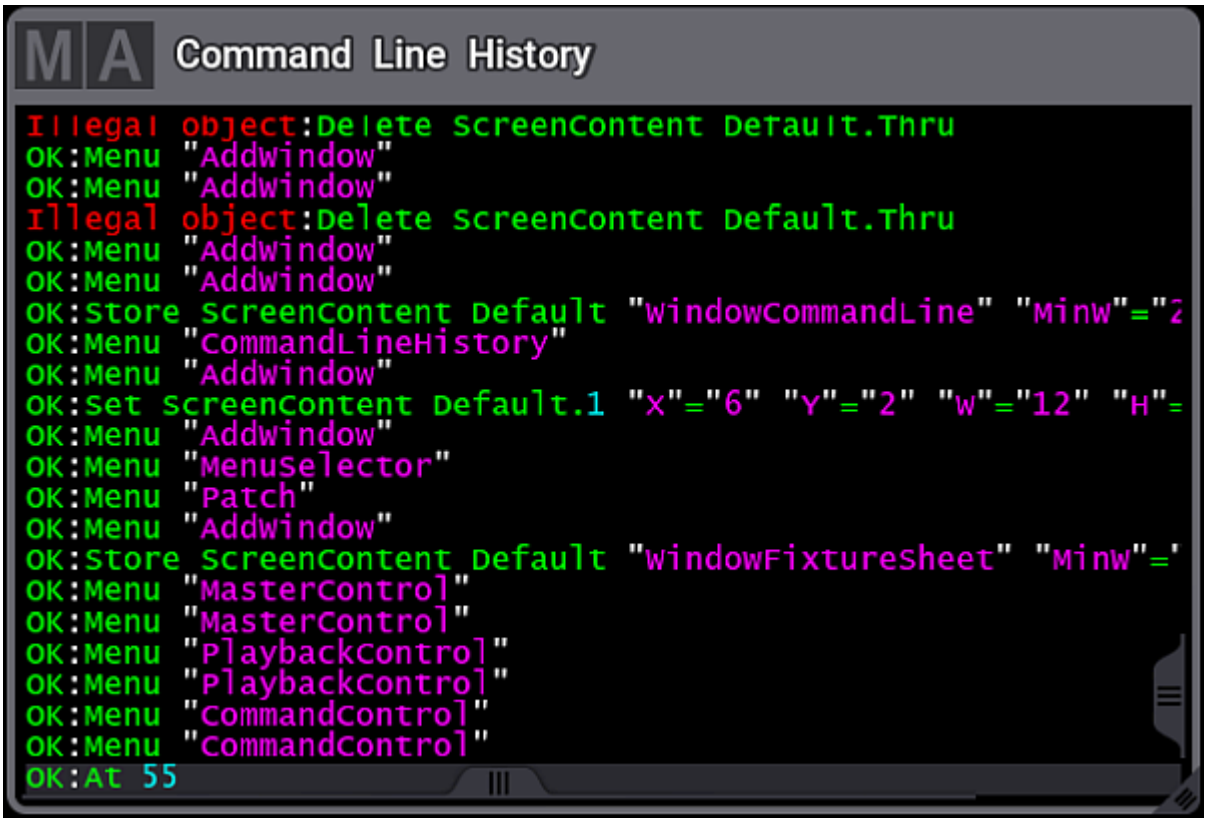
Tap the left arrow at the left edge of the command line to show or hide the control bar along the left edge of the screen.

Open the Command Line History

To open a temporary window showing the Command Line History:

- Tap MA on the left side of the command line

The Command Line History opens.



Command Line History

To close the window:

- Tap MA once again

-or-

- Tap in the upper right corner of the window

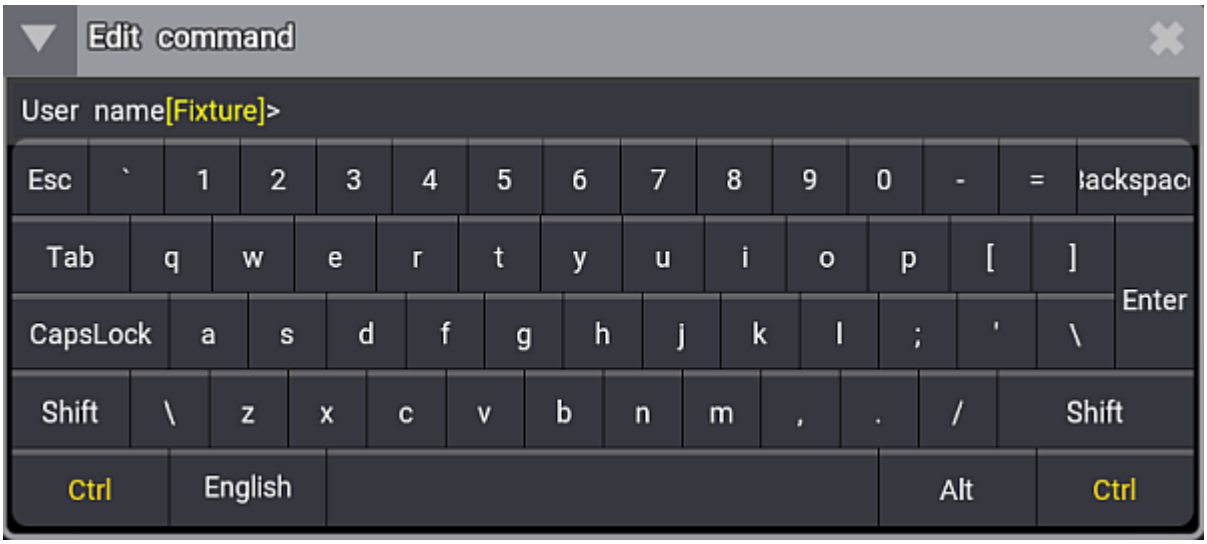
The Add Window menu also includes a storable version of the Command Line History window. Tap Common + Command Line. For more information, see **Add Windows**.

Open the Virtual Keyboard

To open a temporary command line window with a virtual keyboard:

- Tap on the left of the command line

The window opens with a virtual keyboard.



Open virtual keyboard

To close the virtual keyboard:

- Tap  once again

-or-

- Tap  in the upper right corner of the window

---

Information in the Command Prompt

The command prompt is the point where entered syntax appears. The prompt ends with the '>' symbol, but it contains two pieces of helpful information before the symbol.

- **User Name**
  The user name of the current user appears first in the command prompt.
- **Default Keyword**
  The default keyword appears in square brackets after the user name. The console uses the displayed default keyword for any numerical syntax entered without a specified object keyword.

Important:
It is possible to navigate through the directory structure of the console using the command line. In this case, the current location in the directory replaces the default keyword displayed in the command line. When returning to the root directory, the default keyword replaces the directory path display.

Change the Default Keyword

Possible default keywords of the command line can include:

- **Fixture**
- **Channel**
- **Global**
- **Houselights**
- **NonDim**
- **Media**
- **Fog**
- **Effect**
- **Pyro**

To change the default keyword:

1. Enter the desired keyword into the command line. For example, press the Channel key.
2. Press Please.

Set the Cursor in the Command Line

When entering command line syntax using a keyboard, it is necessary to bring the keyboard focus to the command line.

To set the cursor in the command line:

- Tap the command line

- or-

- Press Esc one or more times. Once all open menus and pop-ups are closed, the keyboard focus returns to the command line.

- or-

- Press and hold MA and subsequently press Please

Set cursor in the command line

The cursor is set and starts blinking after the prompt and a white box surrounds the command line.

For more information, see **Command Syntax and Keywords**.

Power Status

On full-size, light, full-size CRV, and light CRV consoles, which are equipped with an internal battery backup, there is an indicator of power status on the right side of the command line. The battery percentage is always displayed at the bottom of the indicator. Status indications include the following states:

- 

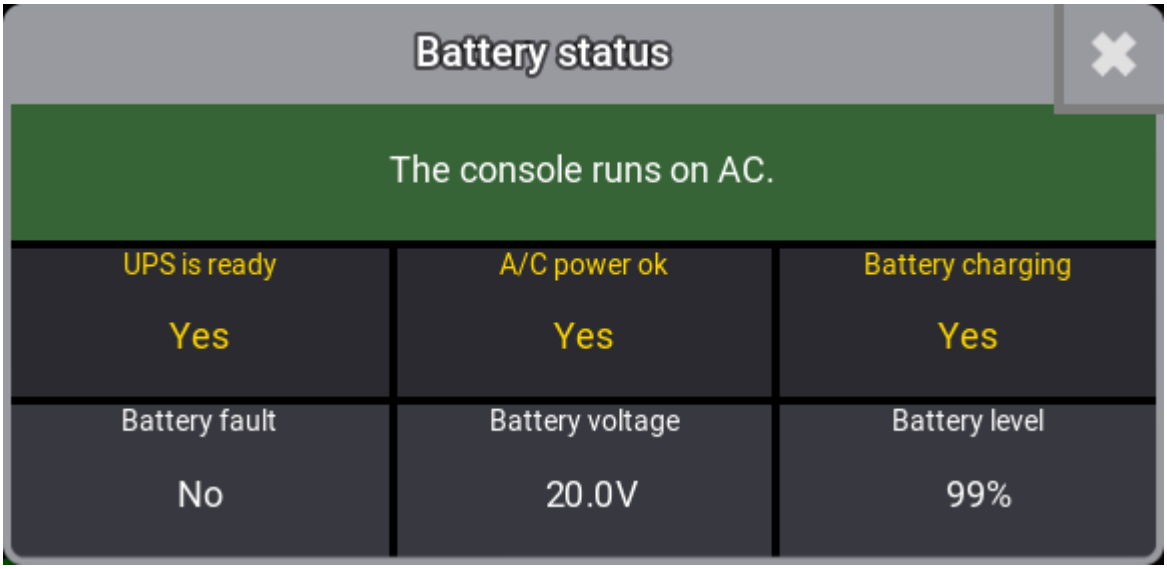
Power is connected and the battery is full.
- 

Power is connected and the battery is charging.
- 

Power is disconnected and the battery is discharging.

Battery Status Pop-up

Tap the power status indicator to open the Battery status pop-up.



Battery status pop-up

The Battery status pop-up displays multiple relevant metrics about the battery as well as A/C power.

Session Status and Quick Access to the Network Menu

An indicator of network session status appears on the right side of the command line. Tapping the indicator toggles the network menu open or closed. The color of the network icon indicates one of the following session states:

- **Red:** Standalone
- **Black:** Startup
- **Gray:** IdleMaster
- **Light Blue:** GlobalMaster
- **Green:** Connected

For more information about networking, see the **Networking** topic.

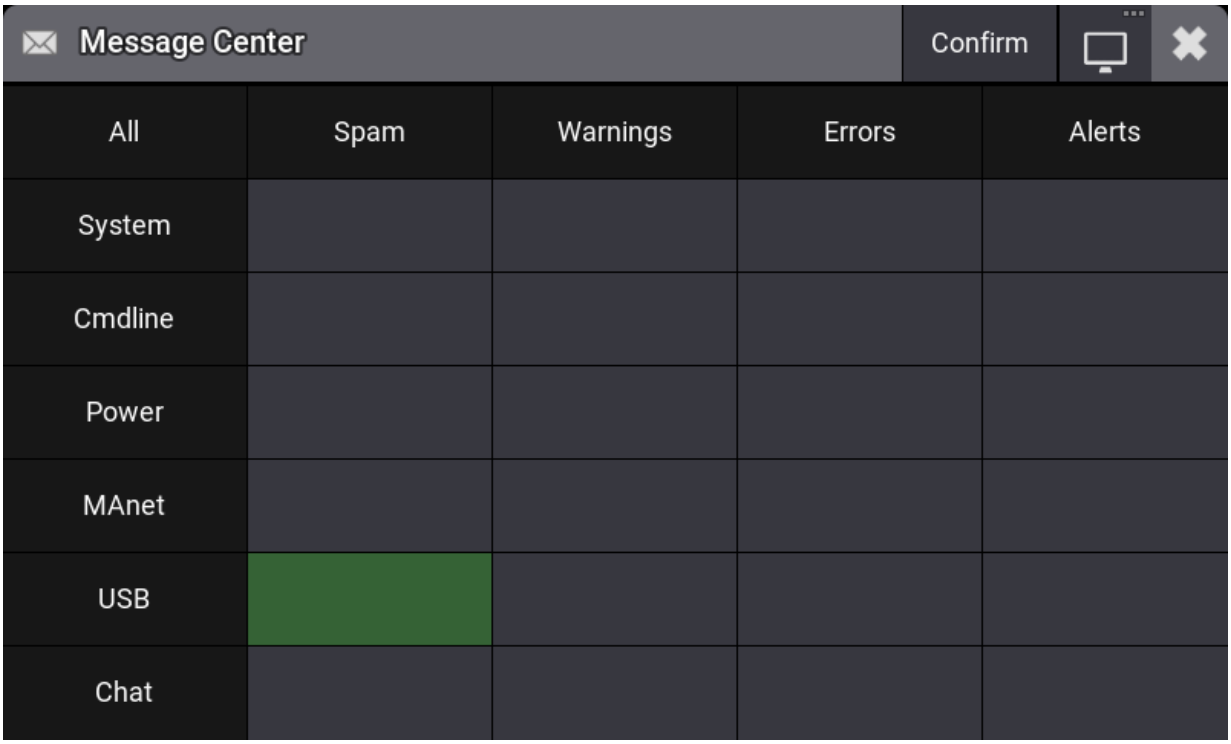
Phaser and Phaser Values Workload

Two status bars between the network and message center icons in the command line show the current processing workload dedicated to phaser calculation. A green indication in the left bar increases with the number of parameters with actively running phasers. A green indication in the right bar increases with the total number of phaser values, which are currently running. Each phaser step for each parameter in a phaser counts as one value.

For more information on phasers, see the **Phasers** topic.

Open the Message Center

The envelope icon on the right side of the command line changes color to indicate new messages in the message center. Tap the icon to open a temporary version of the Message Center window.



Open message center

1. To confirm all messages, tap Confirm.
2. To open the message center on another screen, tap .;
3. To close the message center, tap the message center icon in the command line once again

-or-

- Tap in the upper right corner of the window

World Server Indicator

The globe icon on the right side of the command line changes color from black to green to indicate an active connection to the world server.

For more information on the world server, see the **World Server** topic.

Enable or Disable Keyboard Shortcuts

Tapping the keyboard icon, with the text "ShCuts" beneath it, on the right side of the command line toggles the keyboard shortcuts on or off.

- A white color indicates that keyboard shortcuts are disabled.
- A yellow color indicates that keyboard shortcuts are enabled.
- A red color indicates that keyboard shortcuts are enabled but temporarily blocked by a menu requiring normal keyboard input.

For more information on keyboard shortcuts, see the **Keyboard Shortcuts** topic.

Show or Hide the View Bar

Tap the right arrow at the right edge of the command line to show or hide the bar of view buttons along the right edge of the screen.

7.1.5. Control Bar

The control bar allows fast access to the shutdown command, essential menus, additional displays (in grandMA3 onPC), online help, and at menu.

On console displays, the control bar is hidden on all screens by default. Within the onPC application, the control bar is shown along the left edge of screens 1 through 5 by default. In consoles and the onPC application, the control bar can appear along the left edge of screens 1 through 7.

To show or hide the control bar, tap Show Control Bar in the configure display menu for the desired screen or press the left arrow at the left side of the command line.

Control bar

- Tap the icon at the top of the control bar to open the **shutdown menu**.
- Tap the icon in the control bar to open the main menu. For more information, see the **Menus** topic.

- Tap the  icon in the control bar to open the command section overlay. For more information, see the **Command Section** topic.
- Tap the  icon in the control bar to open the master controls overlay. For more information, see the **Master Controls** topic.
- Tap the  icon in the control bar to open the playback controls overlay. For more information, see the **Playback Controls** topic.
- Tap the  icon in the control bar to open the custom master section overlay. For more information, see the **Special Executors** topic.
- Tap the  icon in the control bar for access to all of the available displays within the grandMA3 onPC application. This option is only available in grandMA3 onPC. For more information, see the **Displays in grandMA3 onPC** topic.
- Tap the  icon in the control bar to enter the Help keyword into the command line. For more information, see the **Help keyword** topic.
- Tap the  icon in the control bar to toggle the main encoder bar to the command wing bar in the grandMA3 onPC application. This option is only available in grandMA3 onPC. For more information, see the **Command Wing Bar** topic.
- Tap the At button in the control bar to open the at overlay. For more information, see below.

At Overlay

The at overlay allows quick access to various programmer functions and special values.

At overlay

- Tap At Filter to open the at filter overlay. For more information, see the **At Filter** topic.
- Tap Cut Programmer, Copy Programmer, and Paste Programmer to cut, copy, and paste the contents of the programmer. For more information, see the **What is the Programmer** topic.
- Tap Delete Steps to delete the current phaser step or steps from the programmer. For more information, see the **Phasers** topic.
- Tap Clear to access the same three clear keywords available by pressing the Clear key. For more information, see the **Clear Key** topic.
- Tap Full, Zero, Default, Normal, On, Off, Release, or Remove to apply the desired special attribute value to the selection. On and off apply to the selection. Full, zero, default, release, and remove apply to the dimmer attribute of the selection. For more information, see the Specials section in the **Calculator** topic.

7.1.6. View Bar

The view bar displays a number of view buttons. The view bar is visible along the right edge of screens 1 through 7 by default. To show or hide the view bar on a screen, tap Show View Bar in the configure display menu for the desired screen, or press the right arrow at the right side of the command line.

View bar displaying seven view buttons

View buttons allow storage and recall of user-defined views. View buttons also allow assignment and recall of macros and plugins. It is possible to:

- Store views to view buttons.
- Assign views, macros, and plugins to view buttons.
- Call views, macros, and plugins on the view buttons.
- Move and copy views, macros, and plugins to different view buttons.
- Delete views, macros, and plugins from view buttons.

Important:
Deleting an object from a view button only deletes the assignment of the view button referencing the object. The previously assigned object still exists in the appropriate view, macro, or plugin pool.

For more information on storing and calling views, see the **Windows and Views – Store and Recall Views** topic.

7.1.7. Tables in General

Compared to sheets that serve as an overview, tables is where you can actively enter, change, and modify values.

Tables have parents and children in form of new "object lines".

- A parent is the main object
- A child is the sub object of the parent
- A child can have several children

To disable or enable children in tables, tap 'New object' line.

Simplify operation and edit in tables:

- **Move/Select**

-To move the focus in a table, tap to select an object and use the arrow buttons of the integrated or virtual keyboard.

- To move the selection from one column to another, tap the heading of the corresponding column.

-Tap the heading twice to select the entire column.

- **Inertial scrolling**

Use two fingers to tap and quickly brush the table in either direction.

Depending on the momentum, the scroll slows down after a while until it comes to a halt.

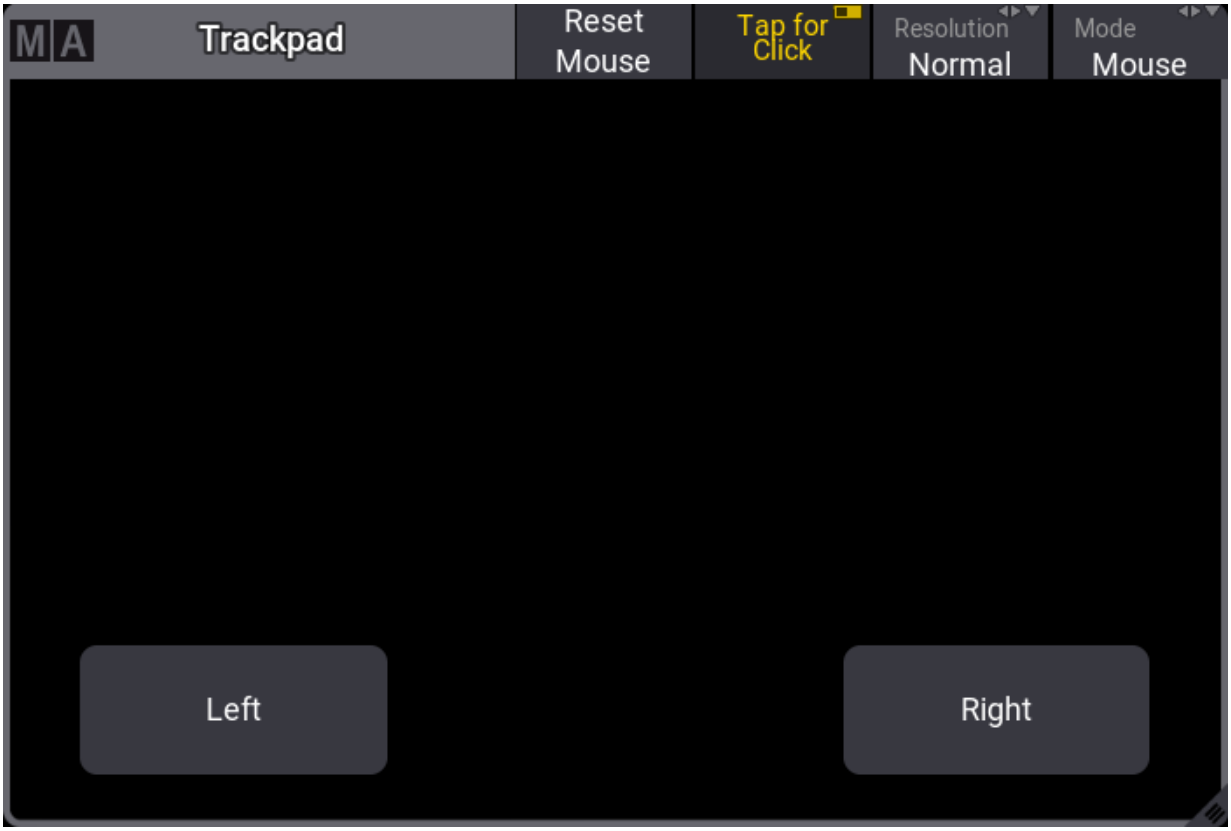
7.1.8. Trackpad Window

The trackpad window allows precise control of a mouse cursor as well as an alternate method for controlling the position of selected fixtures.

The trackpad window can appear within the user-defined area of screens 1 through 7. Views can store and recall the appearance and settings of the trackpad window. For more information on adding windows to the user-defined area, see the **Add windows** topic.

Tap the Mode button in the upper-right corner of the trackpad window to toggle between mouse mode and pan/tilt mode, or tap and swipe to see a menu of available modes.

Mouse Mode

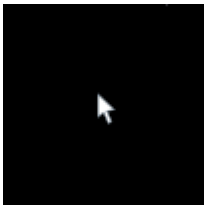


Trackpad window in mouse mode

Trackpad Mouse Cursor

While in mouse mode, tapping inside the main area of the trackpad window and swiping in any direction presents a dedicated cursor and moves that cursor in the direction of the swipe.

The main cursor in the user interface of the console appears as a black arrow with a white outline. This cursor is not affected by the trackpad window. A separate cursor appears when using the trackpad in mouse mode. The trackpad mouse cursor appears as a white arrow with a light gray outline.



Trackpad mouse cursor

The trackpad cursor initially appears slightly larger than normal. After three seconds, it reverts to its normal size. After five seconds of idle time, the trackpad cursor disappears.

The trackpad mouse cursor can move freely across the user interface of the console within screens 1 through 5. Although the trackpad window can appear on screen 6 and screen 7, the trackpad cursor cannot appear on these screens.

Trackpad Window Buttons in Mouse Mode

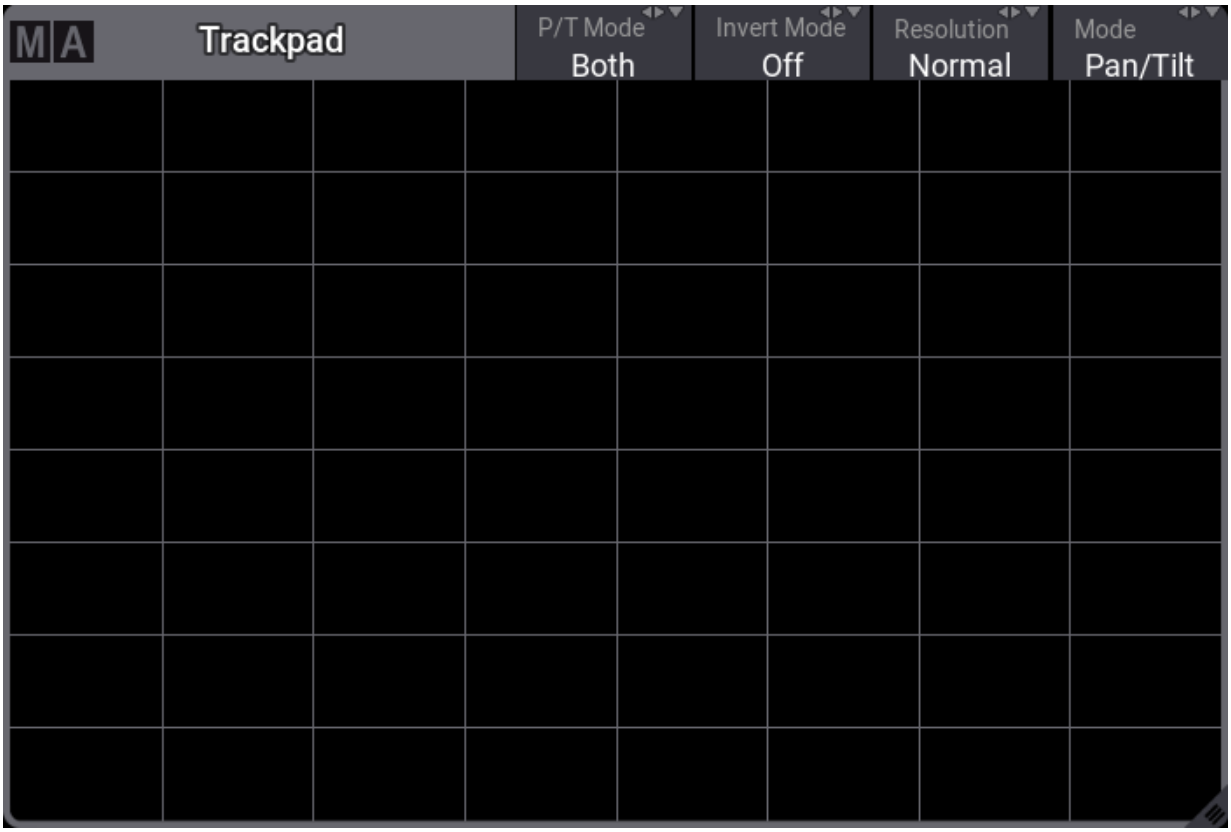
The main area of the trackpad window in mouse mode includes two buttons:

- Left : Tap for a left mouse click. This selects the item under the point of the trackpad cursor. Tap and hold while swiping in the main area of the trackpad window with a second finger to mimic a 1 finger swipe gesture.
- Right : Tap for a right mouse click. This generates the same result as a 2 finger edit gesture.

The title bar of the trackpad in mouse mode includes tools, which control the behavior of the trackpad. These tools include:

- Reset Mouse : Tap to respawn the trackpad mouse cursor in the center of screen 1.
- Tap for Click : When enabled, tapping anywhere in the main area of the trackpad window, except for the Right button, executes a left mouse click. When disabled, only a tap on the Left button executes a left mouse click.
- Resolution : Tap to cycle through the available resolution options for the trackpad mouse, or tap and swipe to see a menu of available resolution options. This resolution setting is separate from the resolution setting in pan/tilt mode.

Pan/Tilt Mode



Trackpad window in pan/tilt mode

While in pan/tilt mode, horizontal swipes within the main area of the trackpad window adjust the pan attribute of any selected fixtures. Vertical swipes adjust the tilt attribute. Diagonal and curved swipes adjust both pan and tilt accordingly. The main area of the trackpad window in pan/tilt mode displays a simple grid as a visual guide.

The title bar of the trackpad in pan/tilt mode includes tools, which control the behavior of the trackpad. These tools include:

- P/T Mode : Tap to cycle through the available pan/tilt modes, or tap and swipe to see a menu of available pan/tilt mode options. These options include:
  - Pan Only : This mode only allows adjustment of the pan attribute, ignoring any vertical movement.
  - Tilt Only : This mode only allows adjustment of the tilt attribute, ignoring any horizontal movement.
  - Both : This mode allows simultaneous adjustment of both the pan and tilt attributes.
- Invert Mode : Tap to cycle through the available invert modes, or tap and swipe to see a menu of available invert mode options. These options include:
  - Off : Both pan and tilt are adjusted normally.
  - Pan Invert : Pan inputs are interpreted in reverse of the normal direction. Tilt inputs remain normal.
  - Tilt Invert : Tilt inputs are interpreted in reverse of the normal direction. Pan inputs remain normal.
  - Both : Both pan and tilt inputs are interpreted in reverse of the normal direction.

- Resolution: Tap to cycle through the available resolution options for the trackpad control of pan and tilt, or tap and swipe to see a menu of available resolution options. This resolution setting is separate from the resolution setting in mouse mode.

Hint:
The **TrackpadMode**, **Mouse Resolution**, **P/T Resolution**, **P/T Invert Mode**, and **P/T Mode** settings are also available in the **Trackpad Window Settings** pop-up. To open this pop-up, tap MA in the upper-left corner of the **Trackpad** window.

7.2. Gestures

Use gestures on the touchscreens of the grandMA3 console to quickly navigate and adjust settings.

- **Tap**

Briefly tap the surface with a fingertip.

- **Tap & Hold**

Tap the surface and hold it for at least one second.

- **Zoom**

Tap the surface using two fingers and move them apart or toward each other.

- **Resize**

1. Tap the title bar or the title field using any finger of one hand and hold it.
2. Briefly tap anywhere on the surface using a second finger.
3. Release both fingers.
4. The window is resized.

- **1 Finger Swipe**

Tap and brush in any direction.

- **2 Finger Scroll**

To scroll within a window, tap the surface with two fingertips and brush the surface in the desired direction.

> **Hint:**
> When using a mouse, press and hold the right and the left mouse buttons, and move the mouse accordingly.

- **3 or More Finger Scroll**

Requirement:

- Set the height and/or width of the user-defined area larger than the available screen space. For more information see the **Configuration of Displays** topic.

To scroll the user-defined area within the available screen space, tap the surface with three or more fingertips and brush the surface in the desired direction.

> **Hint:**
> When using a mouse, press and hold the option key (on macOS) or Alt key (on Windows), then press and hold the right and the left mouse buttons, and move the mouse accordingly.

- **Drag & Drop**

Use this gesture to move and resize windows within the user-defined area as well objects and elements within certain graphical editors (such as the 3D, Layout, Phaser Editor, or Timecode window).

1. Tap and hold with one finger.
2. Move the finger to the desired position.
3. Release the finger.

- **2 Finger Edit**

In sheets and other similar grids:
The **Precise Edit** user profile setting defines the behavior of the **2 Finger Edit** gesture when it is used within sheets and other similar grids (such as the macro editor or patch menu).

Precise Edit disabled:

1. Tap the desired cell.
2. Tap with two fingers anywhere within the same window of the desired cell.

Precise Edit enabled:

1. Tap and hold one finger in the desired cell.
2. Tap anywhere on the same screen with a second finger.
3. Release both fingers.

In pools:
The **Precise Edit** user preference setting does not change the behavior of the **2 Finger Edit** gesture when it is used to edit objects within pools. To edit a pool object, follow the steps above under "**Precise Edit** enabled."

For more information about user profile settings, such as **Precise Edit**, see the **User Settings** topic.

- **Virtual Encoder**

The encoder icons in the Encoder Toolbar include a gesture for turning a virtual encoder to adjust values. For more information, see the **Encoder Toolbar** topic.

7.3. Command Area

This menu displays the Command Area of a console.

As only consoles have a command area, this menu is useful whenever you use an onPC.

To open Command Controls, tap  in the control bar on the left of the screens 1 to 7 or F3 on a keyboard.

Command area

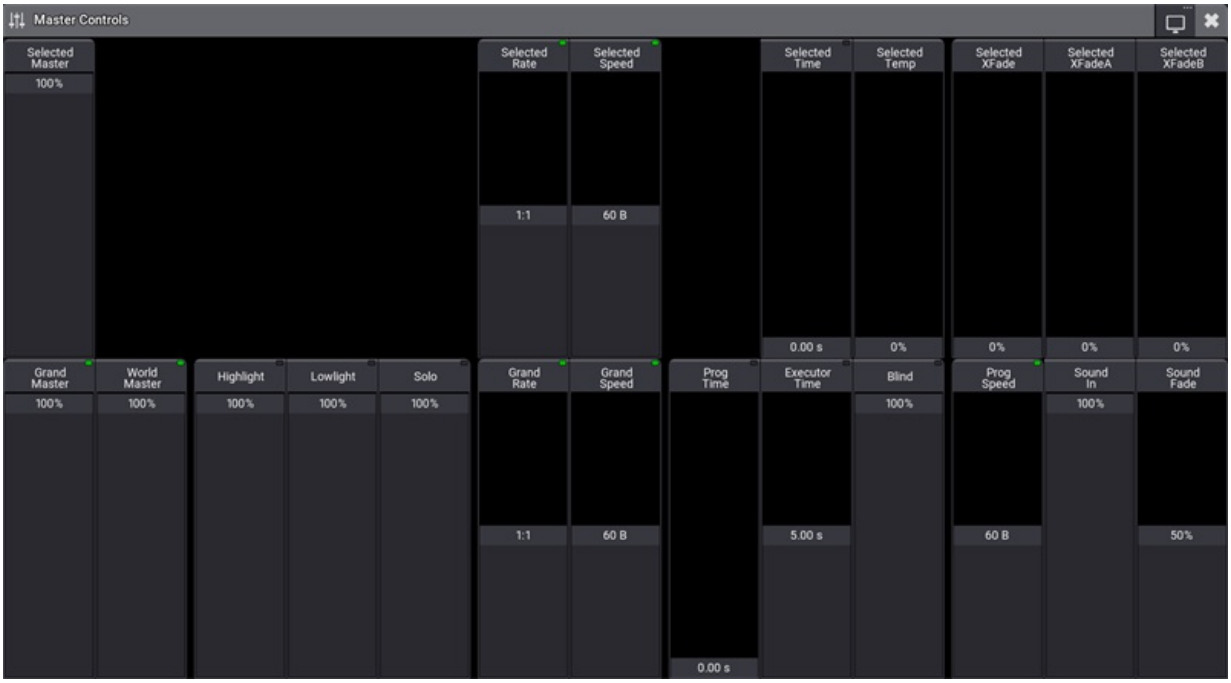
Apart from the command area this menu also displays:

1. **Xkeys**
2. **Grand master**
3. **Level wheel**

7.4. Master Controls

The menu Master Controls displays all special masters.

To open Master Controls, tap in the control bar on the left of screens 1 to 7 or press F4 on a keyboard.



Master Controls

1. Upper section is used in the selected sequence.
2. Lower section is used for global settings.

Enable or adjust value of the on-screen faders depending on their type:

- On-screen faders with dots in the right corner of the title bar:
  -To enable this fader, tap the title bar.
  The dot turns green.
  -To adjust values, tap and slide upward or downward.

- On-screen faders without dots:
  To adjust values, tap and slide upward or downward.

Hint:
You can also enable or disable the on-screen faders **Highlight, Solo, and Blind** using HighIt, Solo, and Blind on the console.

For more information on masters see **Special Masters**.

7.5. Playback Controls

The menu Playback Controls graphically displays physical faders of a console and/or wings.

To open Playback Controls, tap in the control bar on the left of screens 1 to 7 or F5 on a keyboard.

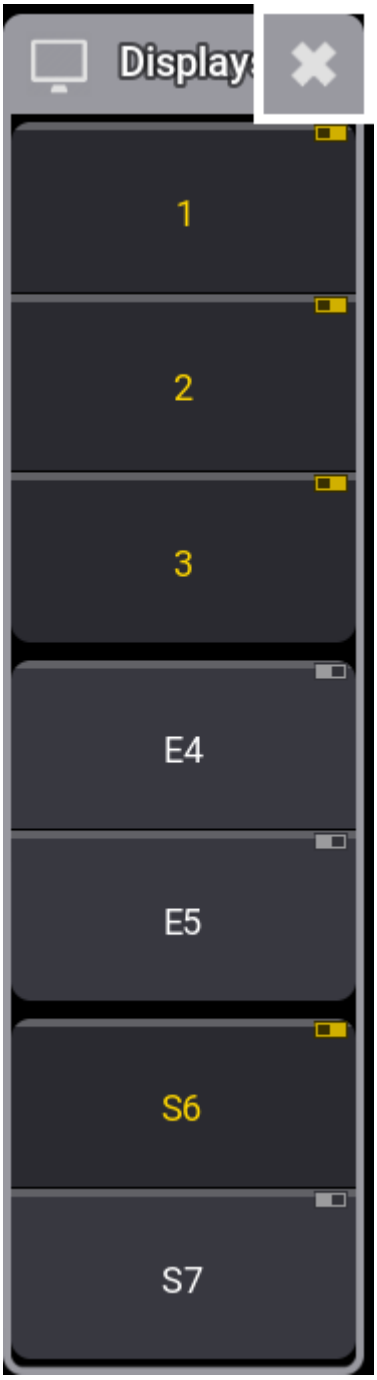
For more information on the functionality of executors see **Executors**.

Empty Playback Controls

7.6. Displays in grandMA3 onPC

The Displays pop-up allows toggling showing or hiding all 7 screens on the grandMA3 onPC.

To open the Displays pop-up on the grandMA3 onPC, tap in the **control bar**.



Displays

Displays can be toggle On or Off by tapping the relevant buttons.

The number and the indicator turn yellow on the displays that are shown.

The pop-up can be closed by tapping the icon again in the control bar or by tapping the X in the upper right corner of the pop-up.

For information on the location of menus see **Change Menu Locations**.

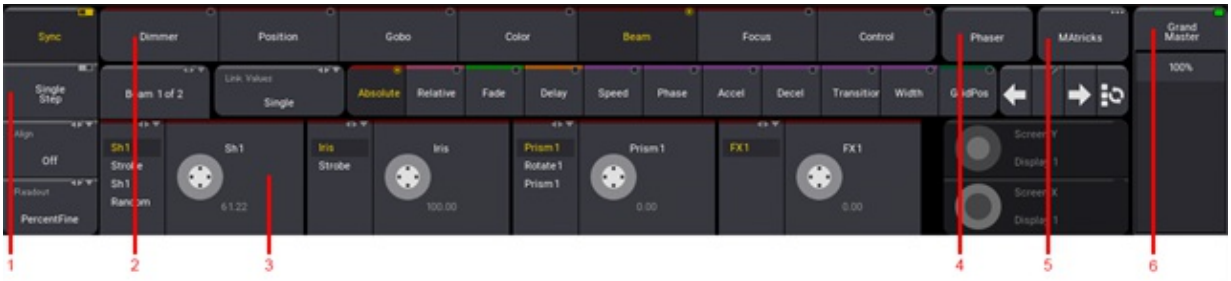
7.7. Encoder Bar

The encoder bar displays the attributes or functions linked to the encoders. The encoder bar also includes shortcuts for quick access to a selection of user profile settings and programming tools.

On the grandMA3 full-size, the grandMA3 full-size CRV, the grandMA3 light, and the grandMA3 light CRV; the encoder bar appears on screen 8. By default, the encoder bar appears at the bottom of screen 1 on all other grandMA3 consoles and the grandMA3 onPC software.

To show or hide the encoder bar on screen 1 on any grandMA3 console or the grandMA onPC software, tap Show Encoder Bar in the configure display menu on screen 1. For more information on the configure display menu, see the **Configuration of Displays** topic.

The encoder bar is also available as a window under the **More** and **All** tabs in the **Add Window** pop-up. For more information on adding windows, see the **Add Windows** topic.



Encoder bar

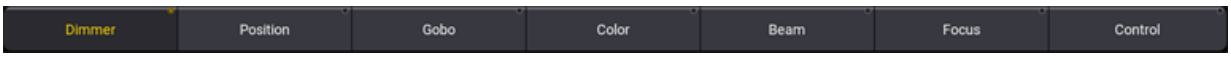
The encoder bar includes the following main sections:

1. User profile settings and align mode.
   - Tap Sync to disable or enable sync. For more information on sync, see the **User Settings** topic.
   - Tap Single Step to enable or disable single step. For more information on the single step setting, see the **User Settings** topic. For more information on using steps in general, see the **Phasers** topic.
   - Tap Align to cycle through align modes or tap and swipe to open a pop-up menu containing all of the available align modes. For more information on align, see the **Align** topic.
   - Tap Readout to cycle through attribute value readout options or tap and swipe to open a pop-up menu containing all of the available readout options. For more information on readout, see the **User Settings** topic.
2. The feature group control bar appears along the top of the encoder bar. For more information, continue to the following sub-topics.
3. The encoder toolbar occupies most of the area in the encoder bar. For more information, continue to the following sub-topics.
4. Tap Phaser to open a temporary version of the Phaser Editor window. For more information on the Phaser Editor window, see the **Phaser Editor** topic.

5. Tap MAtricks to open a temporary version of the MAtricks window. The background, top bar, and three dots in the upper-right corner of this button will change color to indicate that MAtricks is active. For more information on MAtricks, see the **MAtricks and Shuffle** topic.
6. The grand master appears at the right edge of the encoder bar. For more information on the grand master, see the **Grand Master** topic.

7.7.1. Feature Group Control Bar

The feature group control bar appears along the top of the encoder bar. Each feature group used in the show file appears in the form of a radio button.



Feature group control bar

Adding fixtures, which use additional feature groups, to the patch automatically adds the necessary feature groups to the feature group control bar.

Hint:
When the programmer is clear, the bars across the top of the feature group buttons are grey. As the selection changes, the bars across the top of any feature groups available in the current selection change color. The color coordinates with the color of the current layer in the layer toolbar.

Select a Feature Group

To select a feature group in the control bar, tap the desired feature group. The radio button is enabled, and the encoder toolbar adjusts to display the attributes of the selected feature group.

Active Programmer Values

To activate the attributes of a feature group in the programmer, tap twice on the desired feature group. The information activates and a colored marker appears on the feature group button. To deactivate, tap the feature group once again.

Important:
The color of the marker or markers on the feature group button corresponds to the active layer or layers of information.
These markers will appear as long as there is active values, no matter how the values were activated.

Hint:
To activate information in a specific layer, tap the desired layer in the layer control bar in the encoder toolbar before tapping the feature group.

Feature Group as the Object of a Command

Many function keywords (including On, Off, Park, Unpark, Remove, Release, Stomp, and Default) can execute their commands using feature groups as their destination objects.

Example

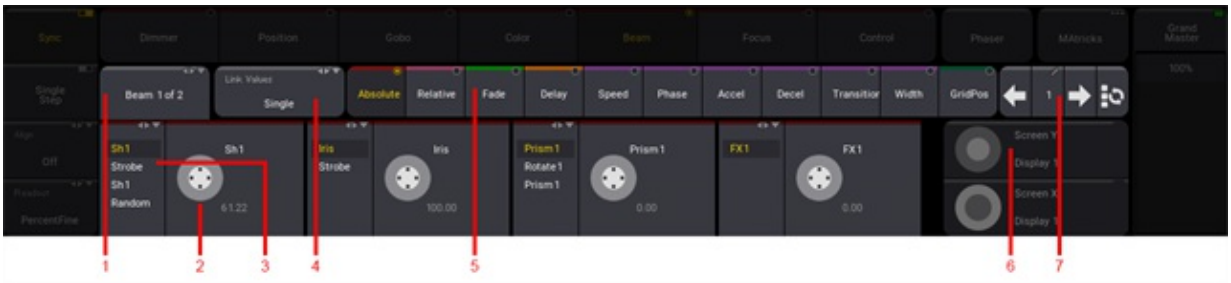
To remove all dimmer values within the current selection from the programmer:

- Press Off and tap Dimmer.

The dimmer values of the current selection are removed from the programmer.

7.7.2. Encoder Toolbar

The encoder toolbar represents the largest portion of the encoder bar. The encoder toolbar is context-sensitive and displays different sets of controls as different editors become active. The default set of controls in the encoder toolbar provides access to and display of attribute encoder information.



Encoder toolbar

Attribute controls available in the encoder toolbar include:

1. Feature button. This button displays the name of the current feature. It also indicates the total number of features available in the current feature group.
   Tap to cycle through features or tap and swipe to open a pop-up menu with available features.
   For more information about features and feature groups, see the **Feature Group** topic.
2. Attribute encoder display. These buttons display the names of the attributes currently linked to the encoders, and show the values of those attributes.
   The center of the encoder icon includes an additional image, which changes depending upon the current resolution of the encoder. For more information on encoder resolution, see the **Encoder Resolution** topic.
   To open the calculator, tap the attribute name field or the attribute value field. For more information about attributes and subattributes, see the **Attribute Definitions** topic.
   To change the value using a gesture in the the user interface; tap and hold or click and hold on the desired encoder icon; if desired, drag the pointer to another area of the screen (this moves the center of the virtual encoder to this new point); drag to draw circles around the center of the virtual encoder in the direction of the desired change (larger circles result in finer control resolution).
3. Channel function of the attribute. These buttons display the current channel function of the attribute displayed directly to the right. They also display additional channel functions when available.
   Tap to cycle through subattributes or tap and swipe to open a pop-up menu with available subattributes.
4. Link button. In cases where the same value, timing, phaser, or resolution adjustments should apply to multiple attributes, the link button offers a few options for defining multiple, simultaneous attribute destinations. For more information about the link button, see the section below.
5. Layer toolbar. The default layer is absolute. Tap any layer button in the toolbar to access the desired layer on the attribute encoders. For more information about the layer toolbar, see the section below.
6. Screen encoder. If the screen encoder is enabled in the user profile settings of the current user, this area displays the basic functionality of the screen encoder. If the screen encoder is disabled in the current user profile, the functionality of the fifth dual encoder will be similar to that of the other four, and the display in this area will also be similar to the displays above the other four encoders. For more information about the screen encoder, see the **User Settings** topic.

7. Step bar. The left and right arrows allow navigation between phaser steps in the programmer. The step number button displays the current phaser step in the programmer. Tap the step number button to open a calculator to choose a specific step. Tap the button to the right of the right-arrow to select all phaser steps. For more information about phaser steps, see the **Phasers** topic.

Link Button

The link button maintains different link settings for the different types of layers found in the layer toolbar as well as encoder resolution. The text in the link button will update automatically to show the link setting of the current layer type. The available link settings are:

- Single. Adjustments made on one encoder apply only to the attribute on that encoder.
- Feature. Adjustments made to one encoder within a feature apply simultaneously to all attributes in the feature.
- At filter. Adjustments made to one encoder apply simultaneously to all attributes currently enabled in the at filter.

Link Values

Value layers include:

- Absolute
- Relative

When a value layer is selected, tap Link Values to cycle through the link options or tap and swipe to open a pop-up menu with all of the available link options. The available link options for value layers include:

- Single
- Feature

Link Timing

Timing layers include:

- Fade
- Delay

When a timing layer is selected, tap Link Timing to cycle through the link options or tap and swipe to open a pop-up menu with all of the available link options. The available link options for timing layers include:

- Single
- Feature
- At filter

Link Phasers

Phaser layers include:

- Speed
- Phase
- Accel
- Decel
- Transition
- Width

When a phaser layer is selected, tap Link Phaser to cycle through the link options or tap and swipe to open a pop-up menu with all of the available link options. The available link options for phaser layers include:

- Single
- Feature
- At filter

Link GridPos

The only available link type when the GridPos layer is selected is the at filter. The link button changes to an At Filter button. Tap At Filter top open a temporary version of the at filter window. For more information about the at filter, see the **At Filter** topic.

Link Resolution

While the MA key is pressed and held, tap Link Resolution to cycle through the link options or tap and swipe to open a pop-up menu with all of the available link options. The available link options for encoder resolution include:

- Single
- Feature

Layer Toolbar

Presets and cues can store and recall multiple layers of data for each attribute. The layer toolbar provides access to all available layers. Layers are color-coded. Markers and text backgrounds using matching colors in the attribute encoder displays, feature group buttons, fixture sheet, and sequence sheet denote active or stored data for the corresponding layer. For more information about colors, see the **Colors** topic.

Tap a button in the layer toolbar to access data for the desired layer. The attribute encoders and any sheet with a layer selection set to auto will display the desired layer. For more information about value and timing layers, see the **What is the Programmer** topic. For more information about phaser layers, see the **Phasers** topic. For more information about using GridPos to create MAgic presets, see the **Create New Presets** topic.

Hint:
When the programmer is clear, the bars across the top of the attribute encoder displays and the channel function displays are grey. As the selection changes, the bars across the top of any attributes and channel functions available in the current selection change color. The color coordinates with the color of the current layer in the layer toolbar.
The lights under the dual-encoders also follow the same behavior.

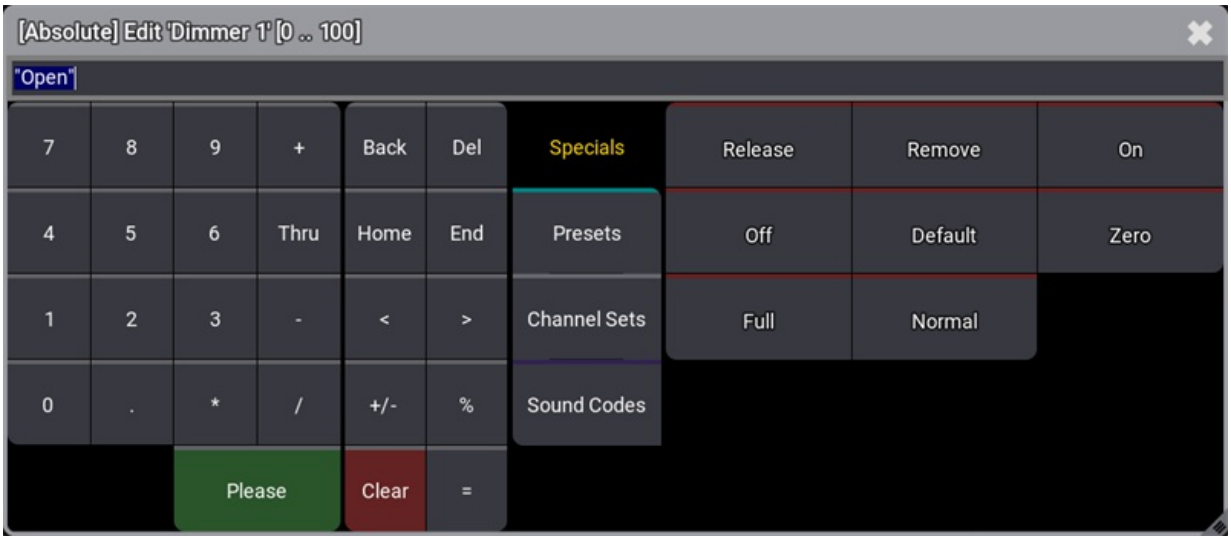
7.8. Calculator

When editing a field that accepts numeric values, the calculator appears. This includes pressing one of the dual encoders while it is displaying an editable value.

The calculator is a dynamic window, and its appearance changes depending upon the edited item and the type of value to be entered.

Example

- Edit a dimmer attribute



The calculator editing a dimmer attribute

The title bar displays several useful pieces of information about what the calculator is currently editing:

- **Layer:** If applicable, the first set of brackets contains the value layer.
- **Attribute** or **parameter:** The name of the attribute or parameter appears after the word "Edit."
- **Value Range:** The last set of brackets contains the allowed range of input values.

Input Field

The input field appears below the title bar. When the calculator opens, the input field displays the current value of the edited object. The text of this value is completely selected so that any immediate entry will replace the existing value. Alternatively, the text can be deselected and retained as part of the entry.

Standard Buttons

The following buttons appear in all calculators.

Number Pad

The calculator displays a number pad on the left. These buttons share the same functions in the calculator as the number keypad in the command section of the console.

Function Buttons

Function buttons appear to the right of the number pad. These buttons include:

- Back : Tap to delete characters to the left of the cursor.
- Del : Tap to delete characters to the right of the cursor.
- Home : Tap to set the cursor at the beginning of the input field.
- End : Tap to set the cursor at the end of the input field.
- < : Tap to move the cursor to the left.
- > : Tap to move the cursor to the right.
- +/- : Tap to insert a negative or a positive value. For more information see **- [Minus] Key** or the **+ [Plus] Key**.
- % : Tap to enter the percent sign.
- = : Tap to enter the equals sign.
- Clear : Tap to delete the entire entry.
- Please : Tap to confirm and apply the value. The calculator closes.

Hint:
Entries using a standard keyboard, whether integrated under the console's armrest or connected through USB, also populate the input field of the calculator.
With one exception, keyboard shortcuts are disabled when entering text into a calculator. The exception is that T will enter the Thru keyword.

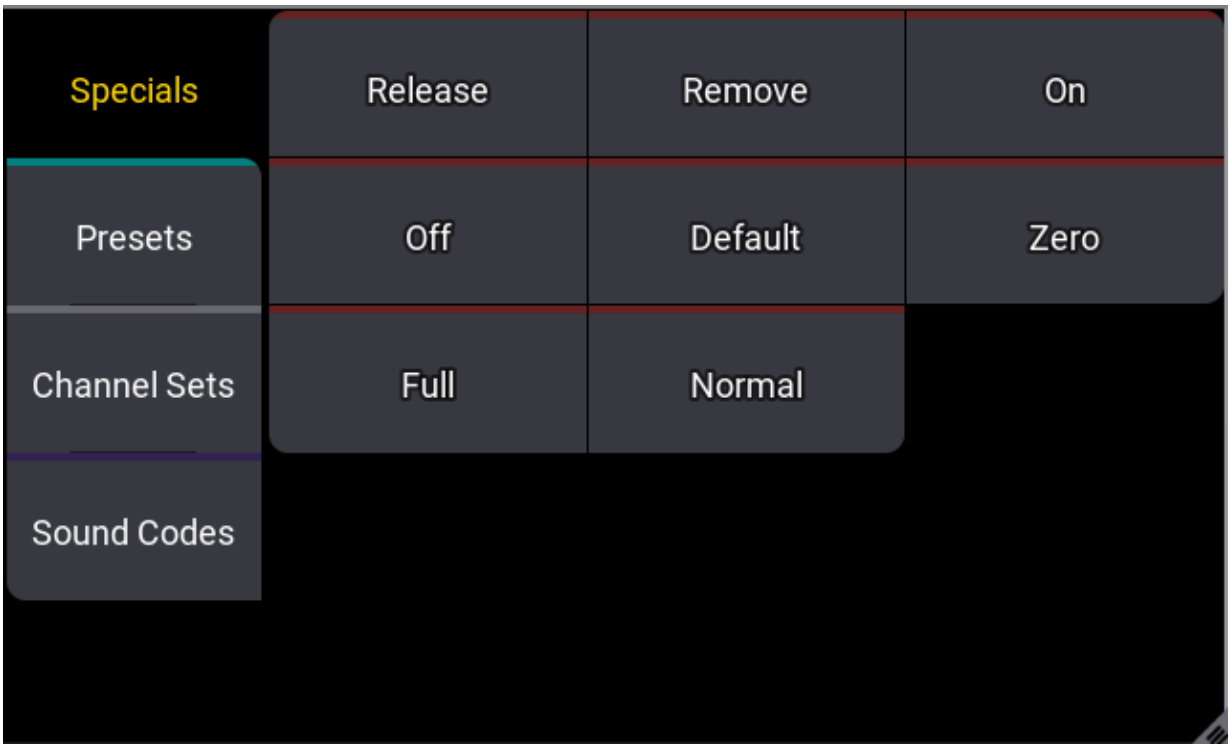
Hint:
When the input readout is set to one of the hex options, an additional row of buttons appears below the number pad, allowing entries of hexadecimal values A, B, C, D, E, and F.

Additional Input Options

Four tabs of additional input options appear to the right of the function buttons.

Specials

To open the special buttons tab, tap Specials .



Specials in the calculator

Use these buttons to enter special commands for attribute values:

- Release : Tap to enter a release value. For more information see the **Release Keyword**.
- Remove : Tap to enter a remove value. For more information see the **Remove Keyword**.
- On : Tap to activate values in the programmer without changing the value.
- Off : Tap to clear the values in the programmer.
- Default : Tap to enter the default value defined for the fixture type. For more information see the **Default Keyword**.
- Zero : Tap to set the values to 0. For more information see the **Zero Keyword**.
- Full : Tap to set values to 100%. For more information see the **Full Keyword**.
- Normal : Tap to set the normal value stored in the user profile. For more information see the **Normal Keyword**.

The at overlay in the control bar also offers quick access to limited versions of these special values. For more information, see the **Control Bar** topic.

Presets

To open the presets tab, tap Presets .

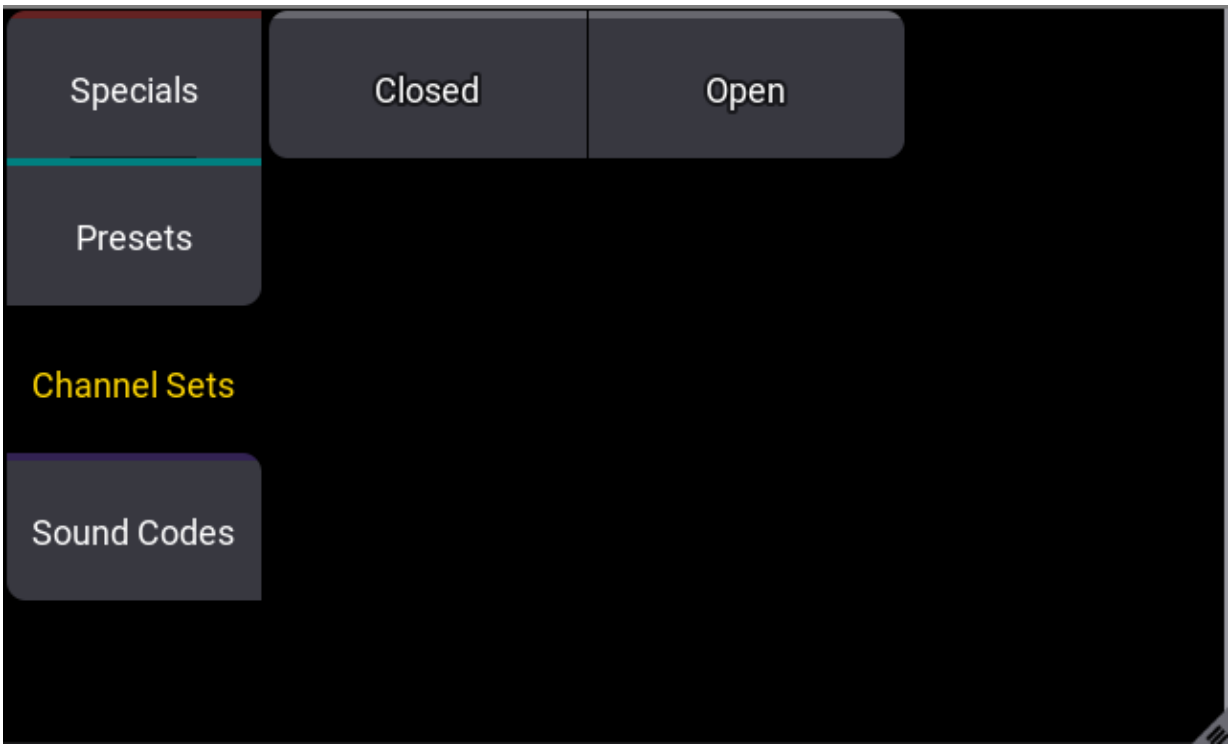
Presets in the calculator

Use these buttons to call presets for attribute values. This tab displays any preset from any preset pool in the currently selected data pool, which contains data that can be applied to the attribute currently edited by the calculator. Each button displays the preset number, using the format: [Feature Group].[ID], as well as the preset label. For more information on presets, see the **Presets** topic.

Important:
Only the buttons within the presets tab recall presets in the calculator. All other value sources in the calculator produce unlinked values, which may be more difficult to reproduce or update.

Channel Sets

To open the channel sets tab, tap Channel Sets.



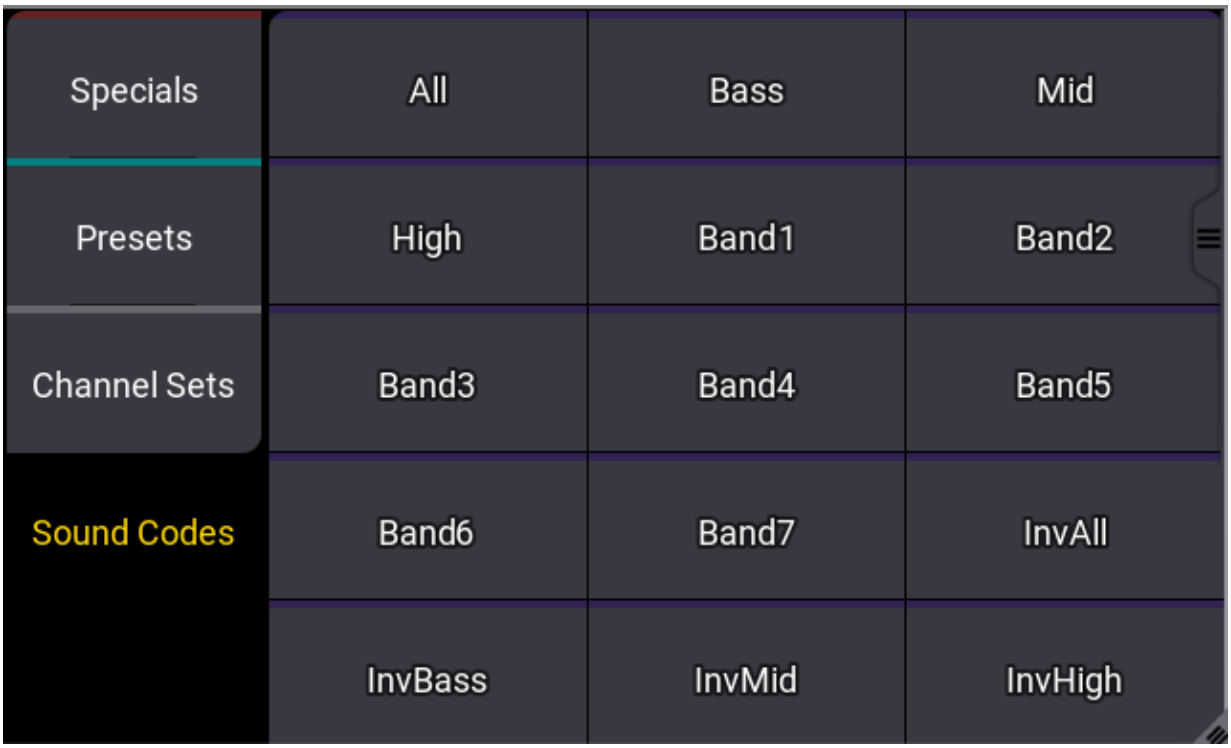
Channel Sets in the calculator

Use these buttons to enter attribute values based on channel sets defined for the fixture type. The channel sets displayed in this tab are the same as those displayed in the smart window. For more information on the smart window, see the **Smart View** topic.

Hint:
When editing attributes with multiple channel functions, a button for each channel function appears between the title bar and the input field of the calculator. Selecting a channel function shows only the relevant channel sets within the channel sets tab.

Sound Codes

To open the sound codes tab, tap Sound Codes.



Sound codes in the calculator

Use these buttons to link the attribute values to an incoming audio signal. Values can reference the total volume of the audio input or they can reference smaller frequency bands within the signal. The available options include:

- All: Use for the total volume of the incoming signal.
- Bass, Mid, and High: Use to reference one of three broad frequency bands.
- Band1 through Band7: Use to reference one of seven narrower frequency bands.
- Inv…: Use to reference the inverse value of any of the above options.

Hint:
Use the SoundIn and SoundFade masters to adjust how the console reacts to incoming audio signals.

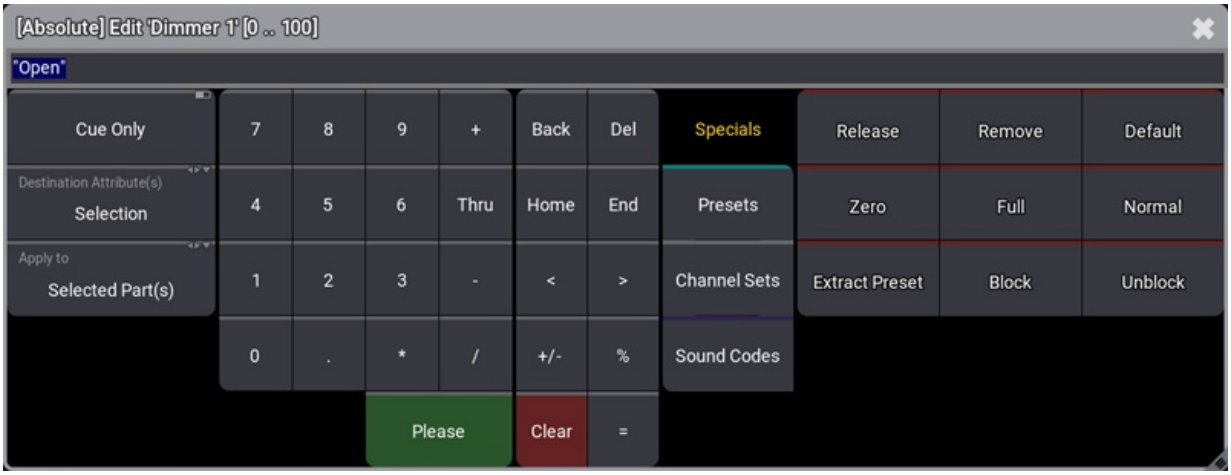
For more information about using sound input, see the **Sound Window** topic.

Track Sheet Calculator

When editing an attribute within the sequence sheet with the track sheet option enabled, the calculator appears with additional options to the left of the number pad. To edit attributes in the track sheet, drag the cursor across the desired attributes and up or down over the desired cues or cue parts. Then, use a 2-finger tap to edit the highlighted cells.

Example

- Edit a dimmer attribute in the track sheet



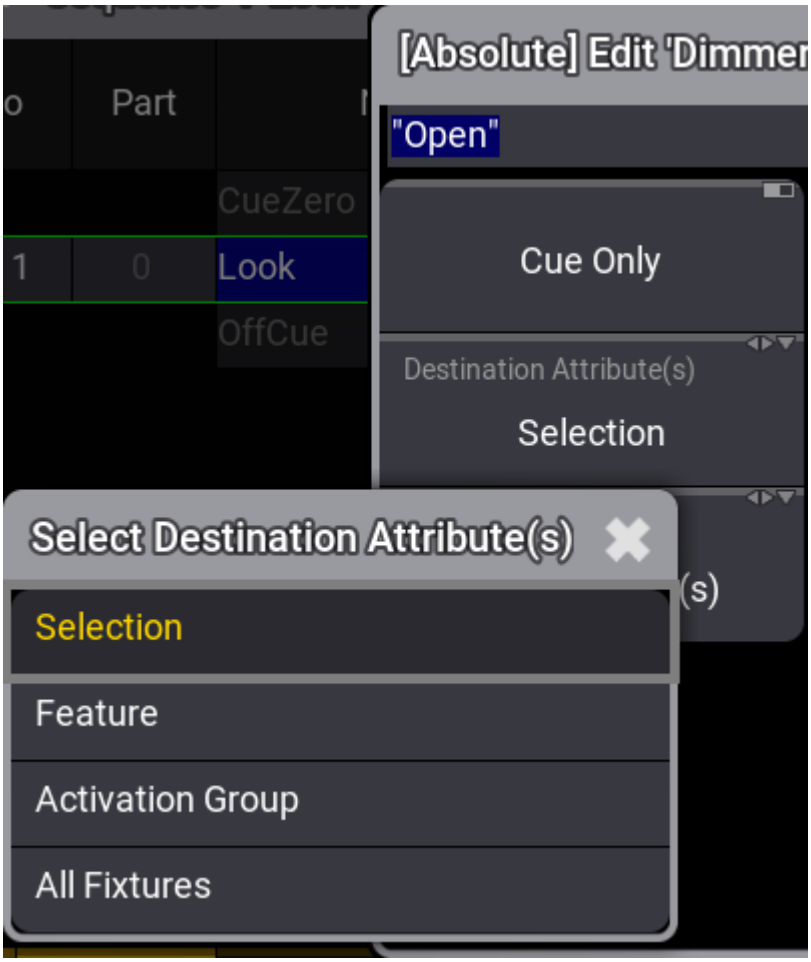
Calculator editing a dimmer in the track sheet

Cue Only

The cue only toggle appears at the top of the additional options to the left of the number pad. Toggle Cue Only off to allow changes to track forward. Toggle Cue Only on to prevent changes from tracking beyond the range of cues defined by the edit options. For more information on cue only, see the **Store Cues** topic.

Destination Attribute(s)

Tap Destination Attribute(s) to cycle through the available attribute destination options, or tap and swipe to see a menu of all available attribute destination options.



Attribute destination options

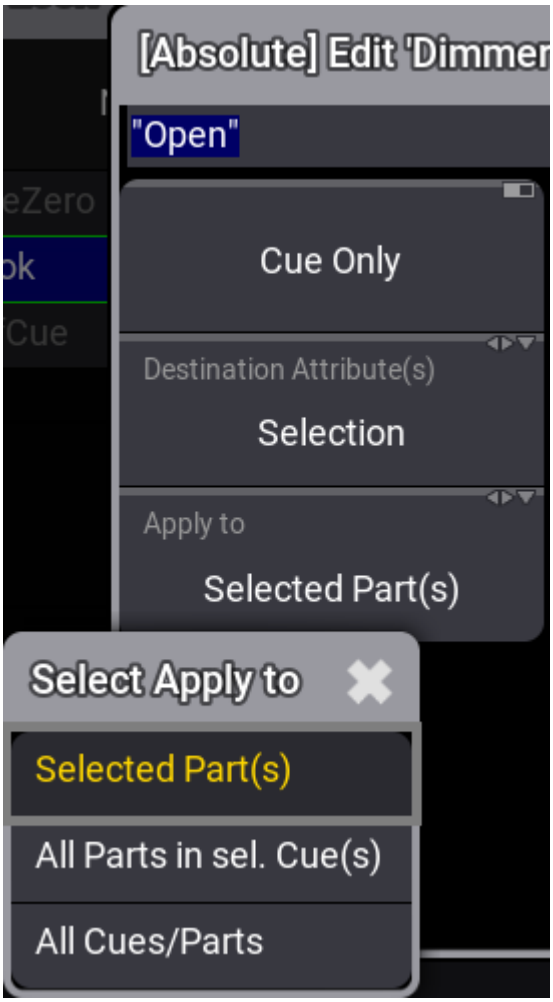
The select destination attributes menu includes the following options:

- Selection: Edits will only apply to the selected attribute cells.
- Feature: Edits will apply to all attributes within the feature group of the selected attribute cells, staying within the selected fixtures.
- Activation Group: Edits will apply to all attributes within the activation group of the selected attribute cells, staying within the selected fixtures.
- All Fixtures: Edits will only apply to all attributes of all fixtures within the cue part.

Hint:
Think of this option as expanding the effect of the edit horizontally within the track sheet.

Apply to

Tap Apply to to cycle through the available cue and part options, or tap and swipe to see a menu of all available cue and part options.



Apply to options

The apply to menu includes the following options:

- Selected Part(s) : Edits will only apply to the selected cue part cells.
- All Parts in sel. Cue(s) : Edits will apply to all parts within the selected cues.
- All Cues/Parts : Edits will apply to all parts of all cues within the sequence.

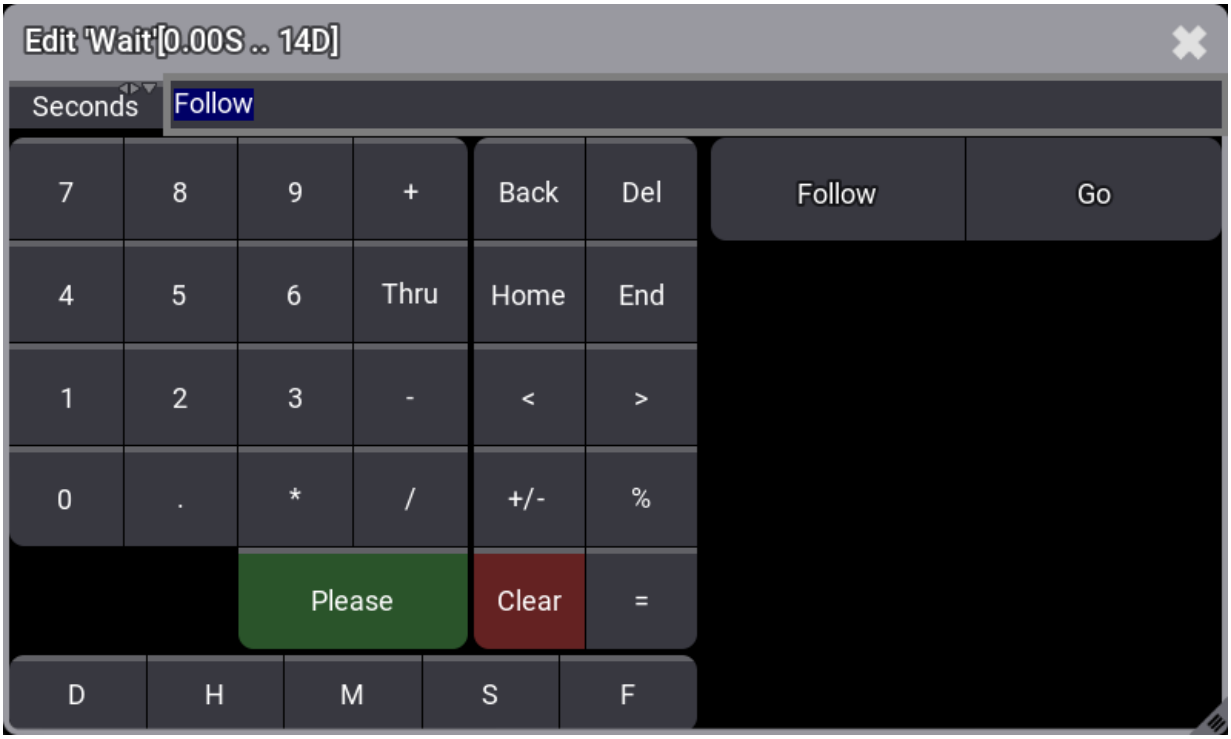
Hint:
Think of this option as expanding the effect of the edit vertically within the track sheet.

Time Calculator

When editing times, the calculator appears with different readout options and additional buttons.

Example

- Edit the wait time for a line in a macro



Calculator editing the wait time for a line in a macro

Additional buttons appear below the number pad. These buttons allow for quick entry of time units:

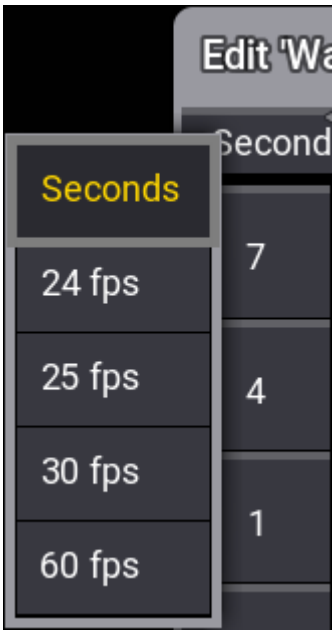
- D : Day
- H : Hour
- M : Minute
- S : Second
- F : Frame

Additional Input Options

Additional input options appear to the right of the function buttons. When editing a cell that can also contain triggers, this area displays trigger options.

Time Readouts

The readout selection appears to the left of the input bar. Tap the readout to cycle through the available readout options, or tap and swipe to see a menu of all available readout options.

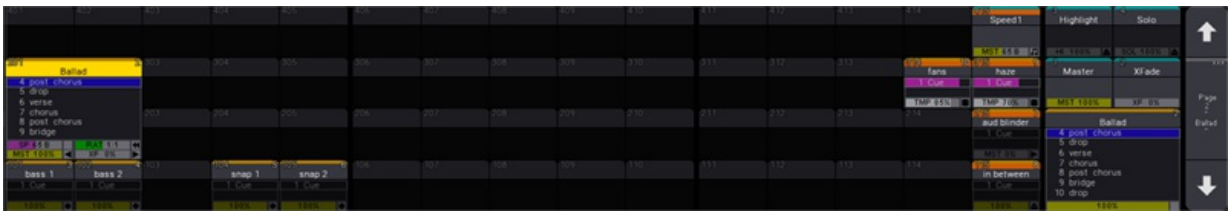


Time readout options

7.9. Playback Bar

The playback bar displays the objects currently assigned to the corresponding executors. The bar includes the playback status of each object as well as the available controls assigned to the executor keys, faders, and knobs.

The playback bar appears permanently on some screens and optionally on other screens. Depending upon where an instance of the playback bar appears, the bar may contain additional information about the assignment and status of the **Master Area** or **Custom Area** as well as page navigation controls.

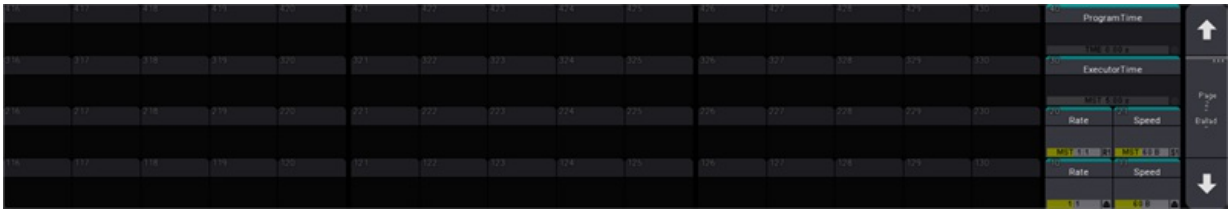


Playback bar with master area and page navigation.

Tap the display of any executor in the playback bar to open the **Assign Menu** for that executor. For more information about the **Assign Menu**, see the **Assign Object to an Executor** topic.

Tap and hold the display of any object assigned to an executor in the playback bar to edit the object. When editing a sequence, a temporary version of the **Sequence Sheet** appears. For more information about the **Sequence Sheet**, see the **Look at Cues and Sequences** topic.

The playback bar with the master area and page navigation always appear on screen 9. To show or hide this version of the playback bar on screen 2 on any grandMA3 console or the grandMA3 onPC software, tap Show Playback Bar in the **Configure Display** pop-up on screen 2.



Playback bar with custom area and page navigation.

The playback bar with the custom area and page navigation always appear on screens 10, 11, and 12. To show or hide this version of the playback bar on screens 3, 4, or 5 on any grandMA3 console or the grandMA3 onPC software, tap Show Playback Bar in the **Configure Display** pop-up on the desired screen.

Playback bar with Xkeys.

The playback bar can also appear on screen 7, showing the assignment and status of the Xkeys. To show or hide the playback bar on screen 7 on any grandMA3 console or the grandMA3 onPC software, tap Show Playback Bar in the **Configure Display** pop-up on screen 7.

A window showing the assignment and status of the Xkeys is also available under the **Common** and **All** tabs in the **Add Window** pop-up. Likewise, a version of the playback bar is also available as a window under the **Common** and **All** tabs in the **Add Window** pop-up. The **Playback** window does not include the custom area, master area, or the same page navigation controls as the **Playback Bar**. For more information on adding windows, see the **Add Windows** topic.

Both the **Playback** window and the **Xkeys** window additionally include on-screen copies of the relevant faders, knobs, and keys. To show or hide these additional controls, tap Executors in the **Window Settings** pop-up. To show or hide the labels, tap Labels in the **Window Settings** pop-up. To display a specific page in the current instance of the **Playback** or **Xkeys** window, tap the Page button in the **Window Settings** pop-up and tap the desired page in the **Select Page** pop-up. Tap <Link Selected> at the top of the **Select Page** pop-up to always display the current executor page.

In the **Playback Window Settings** pop-up, tap the Row400, Row300, Row200, or Row100 button to show or hide labels and executors for the specified executor row. Tap the WingID button to open a pop-up to choose which set of executor columns to display. Each **WingID** represents a different set of 15 columns of executors.

For more information about the **Configure Display** pop-up, see the **Configuration of Displays** topic.

For more information about screen allocation, see the **Screen Allocation** topic.

For more information about executors, page navigation, and the playback window, see the **Executors** topic.

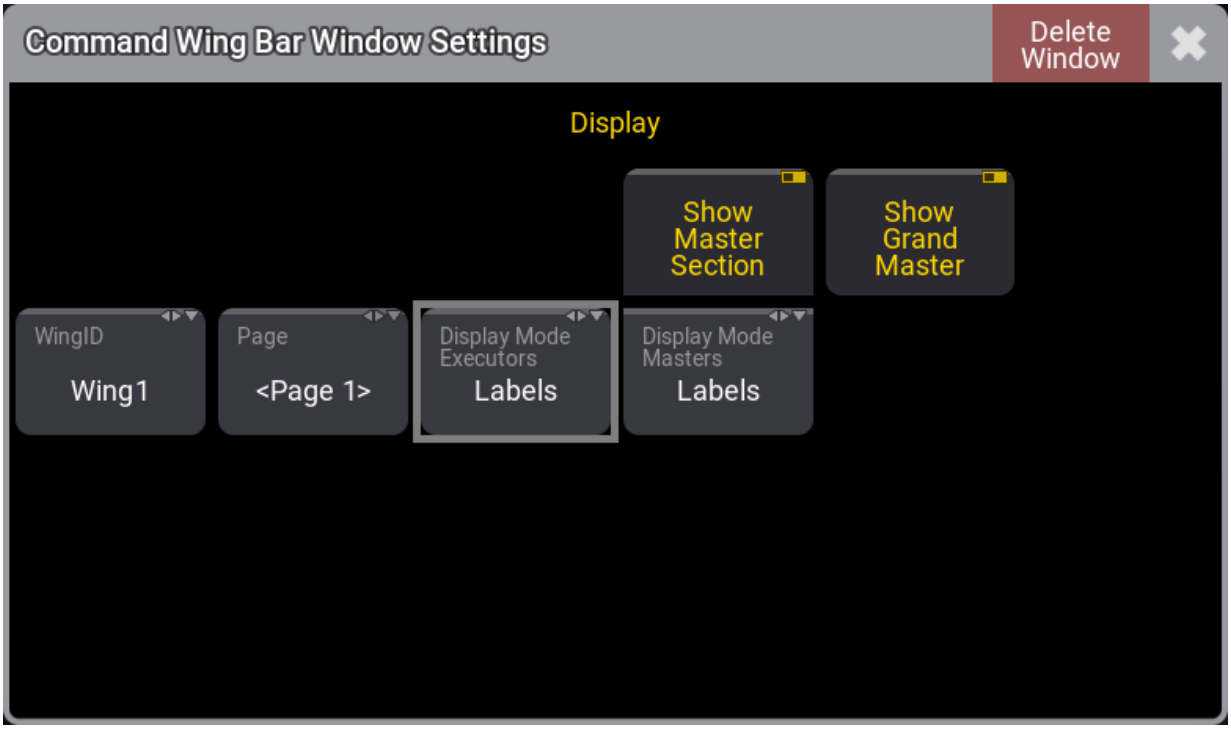
7.10. Command Wing Bar

The **Command Wing Bar** combines a small version of the **Playback Bar**, including two sections of executors and the **Master Area**, with a compressed version of the **Encoder Bar**. This combined bar is helpful for optimizing screen space and minimizing the views necessary to move between programming and playback, especially when using the grandMA3 onPC software. The executors displayed on the playback side of the bar coincide with the executor sections available on the grandMA3 onPC command wing and command wing XT.

In the grandMA3 onPC software, tap the icon in the **Control Bar** to toggle the **Encoder Bar** to the **Command Wing Bar**. **The Command Wing Bar** is also available as a window under the More and All tabs in the **Add Window** pop-up. For more information on adding windows, see the **Add Window** topic.

Command Wing Bar

Tap MA in the upper-left corner of the **Command Wing Bar Window** to access the **Command Wing Bar Window Settings** pop-up.



Command Wing Bar Window Settings pop-up

This pop-up includes the following settings:

- Display Mode Executors: Tap to toggle the display of the executors in the window between Labels and Hardware Buttons.

- Display Mode Masters : Tap to toggle the display of the special masters in the window between Labels and Hardware Buttons .

The **Window Settings** topic describes additional window settings.

7.11. Colors

grandMA3 uses a number of colors to provide information about states of values and their origin. There are two editable color themes included in the software, and it is possible to create, import, and export additional color themes. The following topics give an overview of the colors defined in the default color theme, as well as an introduction to the color theme editor.

Important:
Use caution when editing the color theme, as it is possible to make information on the screens difficult or impossible to read; for example, selecting the same color for both text and the background behind the text.

Hint:
MA + MA + Clear activates the default color theme and resets all values of that theme to the manufacturer defaults.

7.11.1. System Colors

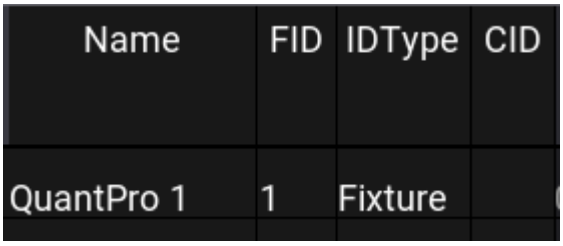
Important:
The colors described here are included in the default color theme.
For more information about alternate color themes, see the **Color theme** topic.

Hint:
MA + MA + Clear activates the default color theme and resets all values of that theme to the manufacturer defaults.

Sheet Colors for Fixture Names and IDs

The fixture names and numbers in sheets have three different color combinations.

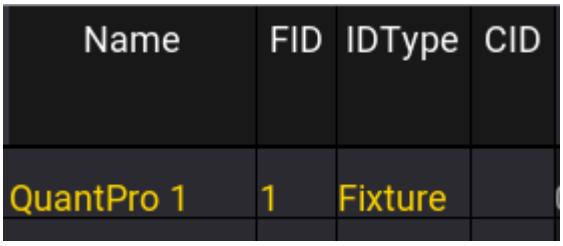
White:



White names and numbers with a dark gray background

White fixture names and IDs on a dark gray background indicate fixtures that are not selected. It is the default color combination.

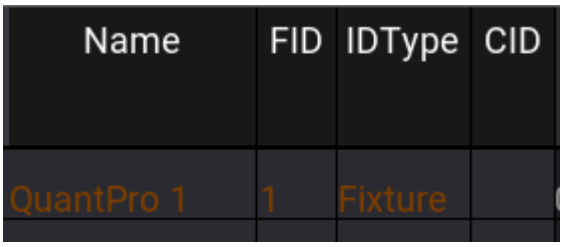
Yellow:



Yellow names and numbers on a medium gray background

Yellow fixture names and IDs on a medium gray background indicate fixtures that are selected.

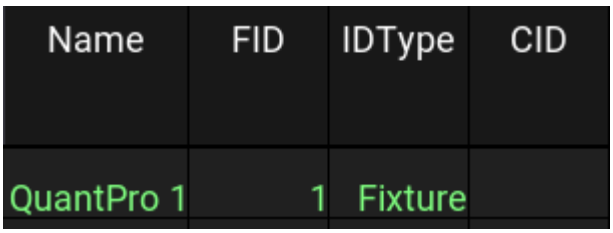
Dim Orange:



Orange names and numbers on a medium gray background

Orange fixture names and IDs on a medium gray background indicate fixtures that are in the current selection, but not the current sub-selection. For example, multiple fixtures are selected and Next has been pressed.

Green:



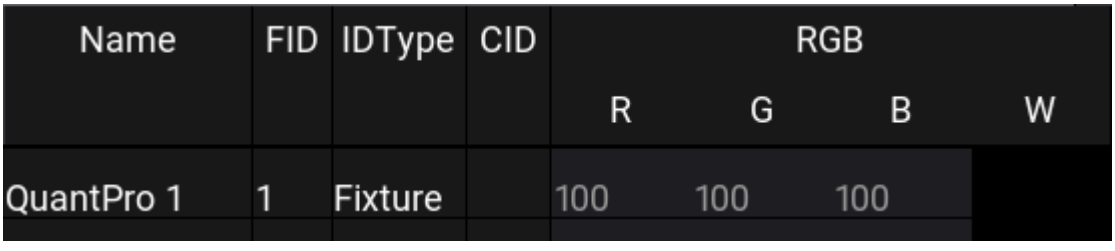
Green names and numbers on a medium gray background

Green fixture names and IDs on a medium gray background indicate fixtures with inverted encoder control of at least one attribute, based on selection and MAtricks settings. For more information about the MAtricks, see the **MAtricks and Shuffle** topic.

Sheet Colors for Absolute Attribute Values

The cells showing attribute values in the fixture sheet and sequence sheet, with the track sheet mode enabled, have several combinations of text and background colors.

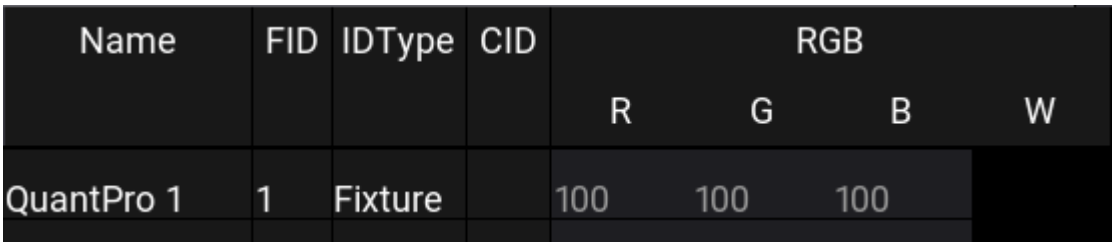
Black:



Black background for on the W attribute

There is no object to be edited. In this case, this fixture does not include a white color mixing attribute.

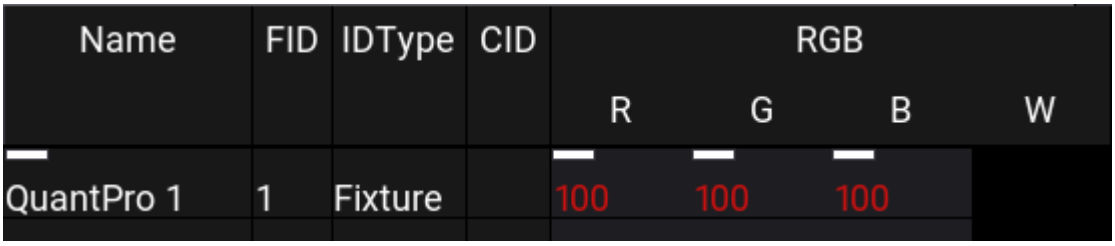
Light gray text with a gray background:



Light gray text with a gray background on R, G, and B attributes

The values of these attributes are at their default levels.

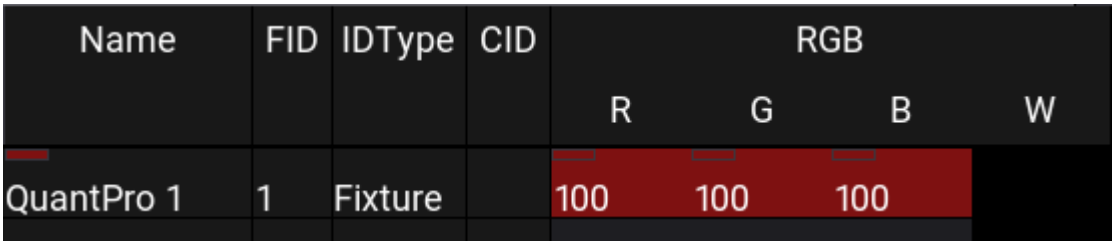
Red text with gray background:



Red text with gray background on R, G, and B attributes

These attributes have values in the programmer, but they are not active. By default, they will not be included in any store or update actions.

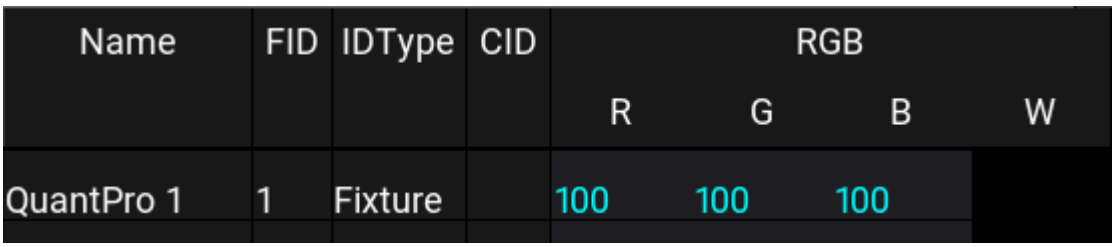
White text with a red background:



White text with a red background on R, G, and B attributes

These attributes have active values in the programmer. By default, they will be included in any store or update actions.

Cyan text with a gray background:



Cyan text with a gray background on R, G, and B attributes

These attributes have new values in the current cue outputting from the selected sequence. A dimmer value displayed with cyan text indicates a new, higher value in the current cue outputting from the selected sequence.

Green text with a gray background:

Name	FID	IDType	CID	Dimmer	PanTilt		
				Dim	P	T	R
QuantPro 1	1	Fixture		50	50	50	

Green text with a gray background on R, G, and B attributes

A dimmer value displayed with green text indicates a new, lower value in the current cue outputting from the selected sequence. This color is only used for descending dimmer values.

Magenta text with a gray background:

Name	FID	IDType	CID	RGB			
				R	G	B	W
QuantPro 1	1	Fixture		100	100	100	

Magenta text with a gray background on R, G, and B attributes

These attributes have tracked values in the current cue outputting from the selected sequence.

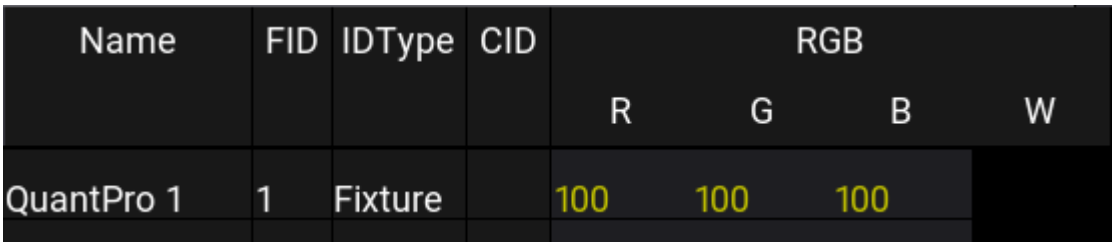
White text with a gray background:

Name	FID	IDType	CID	RGB			
				R	G	B	W
QuantPro 1	1	Fixture		100	100	100	

White text with a gray background on R, G, and B attributes

These attributes have blocked values in the current cue outputting from the selected sequence.

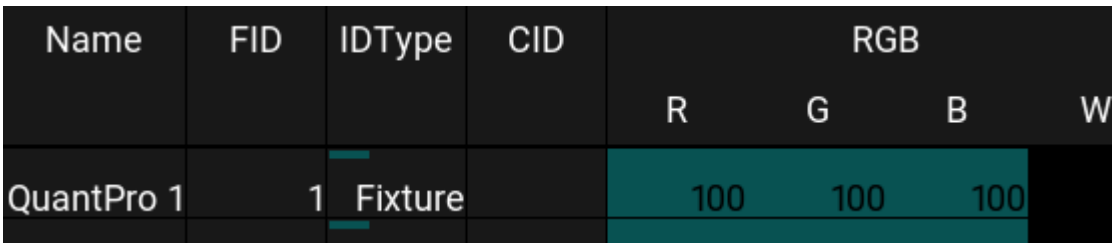
Yellow text with a gray background:



Yellow text with a gray background on R, G, and B attributes

These attributes have new or blocked values in the current cue outputting from a sequence that is not selected. This color combination does not appear in the sequence sheet.

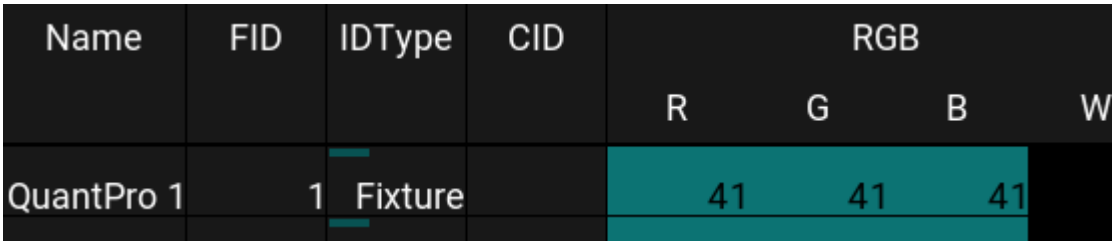
Black text with a deep sea green background:



Black text with a deep sea green background on R, G, and B attributes

These attributes have completed a move in black. For more information on move in black, see the **Move in black** topic.

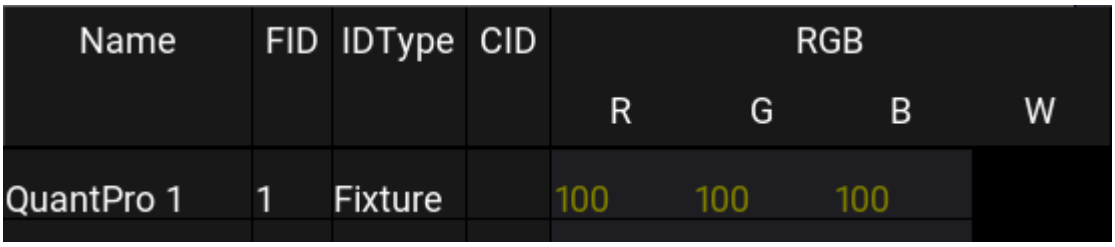
Black text with a sea green background:



Black text with a sea green background on R, G, and B attributes

These attributes are currently fading through a move in black command. This color combination does not appear in the sequence sheet. For more information on move in black, see the **Move in black** topic.

Dim yellow text with a gray background:



Dim yellow text with a gray background on R, G, and B attributes

These attributes have tracked values in the current cue outputting from a sequence that is not selected. This color combination does not appear in the sequence sheet.

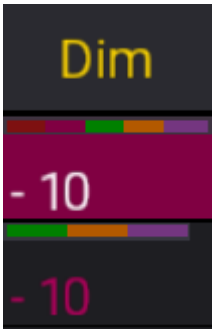
Sheet Colors for Additional Attribute Layers

Along with the absolute value, other layers of values may influence the output of an attribute. The color of the values from these layers matches the colors in the layer control bar in the encoder toolbar. For more information on the encoder toolbar, see the **Encoder Toolbar** topic.

The combinations of text and background colors for the additional layers all use the following pattern:

- White text on a background of the layer's color shows a value in the programmer.
- The text of the layer's color on a gray background shows a value from the current cue in the selected sequence.
- Yellow text on a gray background shows a value from a cue outputting from a sequence that is not selected.

Mauve background or text:



Mauve background and text.

There is a relative value on the dimmers of these two fixtures. The top value with a mauve background comes from the programmer. The bottom value with mauve text comes from the current cue in a selected sequence.

Green background or text:

Green background and text.

There is an individual fade time on the dimmers of these two fixtures. The top value with a green background comes from the programmer. The bottom value with green text comes from the current cue in a selected sequence.

Orange background or text:

Orange background and text.

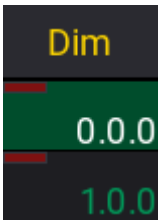
There is an individual delay time on the dimmers of these two fixtures. The top value with an orange background comes from the programmer. The bottom value with orange text comes from the current cue in a selected sequence.

Purple background or text:

Purple background and text.

There is a phaser value on the dimmers of these two fixtures. All layers of phaser values share the same purple color. The top value with a purple background comes from the programmer. The bottom value with purple text comes from the current cue in a selected sequence.

Shamrock background or text:



Shamrock background and text.

There is a GridPos value on the dimmers of these two fixtures. Both values come from the programmer. The top value with a shamrock background shows an active GridPos value. The bottom value with shamrock text shows a deactivated GridPos value.

7.11.2. Markers

Important:
The colors described here are included in the default color theme.
For more information about alternate color themes, see the **Color theme** topic.

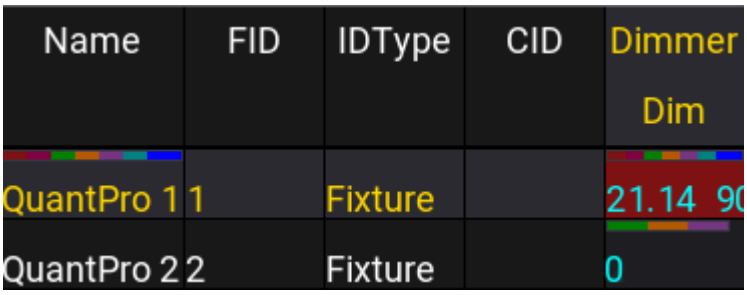
Hint:
MA + MA + Clear activates the default color theme and resets all values of that theme to the manufacturer defaults.

Markers are colored indicators in the form of a bar or a thin line. Markers appear in:

- Fixture Sheet
- Sequence Sheet with the Track Sheet option enabled
- Feature group control bar
- Attribute encoder display

The name column in the fixture sheet, the feature group control bar, and the attribute encoder display all show color-coded markers indicating values, individual timing, and phasers in the programmer. Attribute columns in the fixture sheet show markers indicating values, individual timing, and phasers either in the programmer or currently outputting from a sequence. Attribute columns in the sequence sheet show markers indicating values, individual timing, and phasers stored within the displayed cues.

Marker colors:

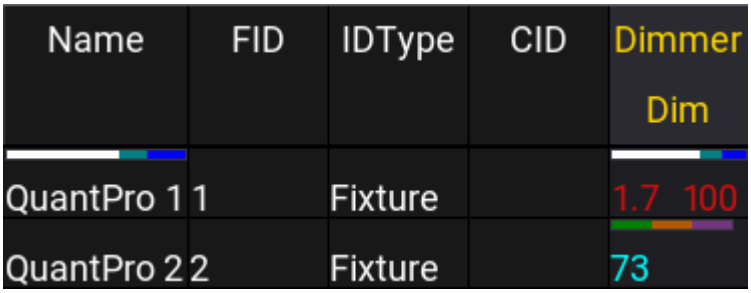


Fixture sheet with markers on one fixture name and both dimmer attributes

Markers above the name and dimmer value of fixture 1 indicate active information in the programmer. Markers only above the dimmer value of fixture 2, but not above the name, indicate information output from a sequence.

- Red markers indicate active absolute programmer values.
- Mauve markers indicate active relative programmer values.
- Green markers indicate individual fade times.
- Orange markers indicate individual delay times.
- Pink markers indicate phaser values.

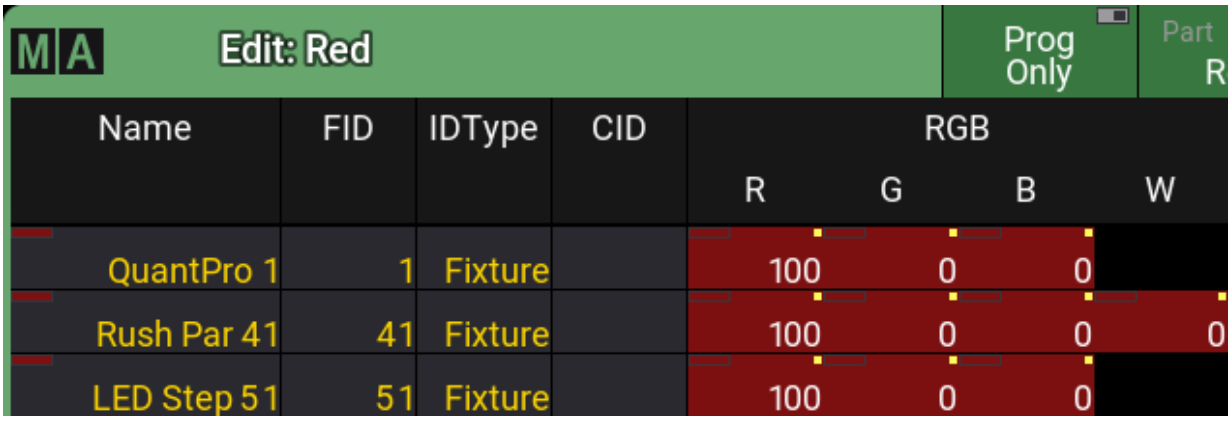
- Cyan markers indicate preset values.
- Blue markers indicate parked attributes.



Fixture sheet with white markers

White markers indicate information in the programmer, which will not be stored when storing active values.

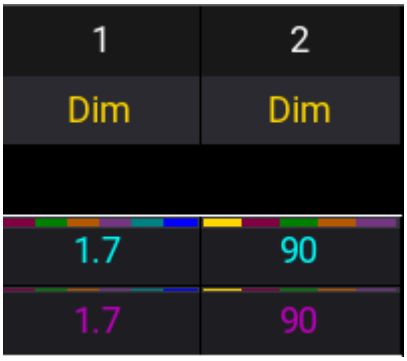
Small, yellow markers when editing a preset:



Fixture sheet while editing a global preset.

Small, yellow markers in the fixture sheet while editing a global preset indicate an attribute holding the global value for that attribute for all fixtures of the same type within the preset. For more information on global presets, see the **Presets** topic.

Markers in the Sequence Sheet with the Track Sheet option enabled:



Sequence sheet attribute columns with markers

Markers in the sequence sheet include an additional color and two different heights.

- Yellow markers indicate values with no preset reference.
- Markers in the form of a bar indicate a new value stored in the cue.
- Markers in the form of a thin line indicate a tracked value.

7.11.3. Color Theme

Important:
Use caution when editing the color theme, as it is possible to make information on the screens difficult or impossible to read; for example, selecting the same color for both text and the background behind the text.

Hint:
MA + MA + Clear activates the default color theme and resets all values of that theme to the manufacturer defaults.

Color themes allow for customization of a wide variety of colors within the graphical user interface of the console. Alternate color themes can be activated quickly, changing the look of the interface on demand. Color themes can be exported to and imported from external sources. The console initially includes two color themes. The "default" color theme, which provides optimal readability when using the console under most lighting conditions. The "defaultDAYLIGHT" color theme creates a bolder interface for use in brighter conditions, such as daylight.

Activating Alternate Color Themes

Color themes can be imported using the menu.

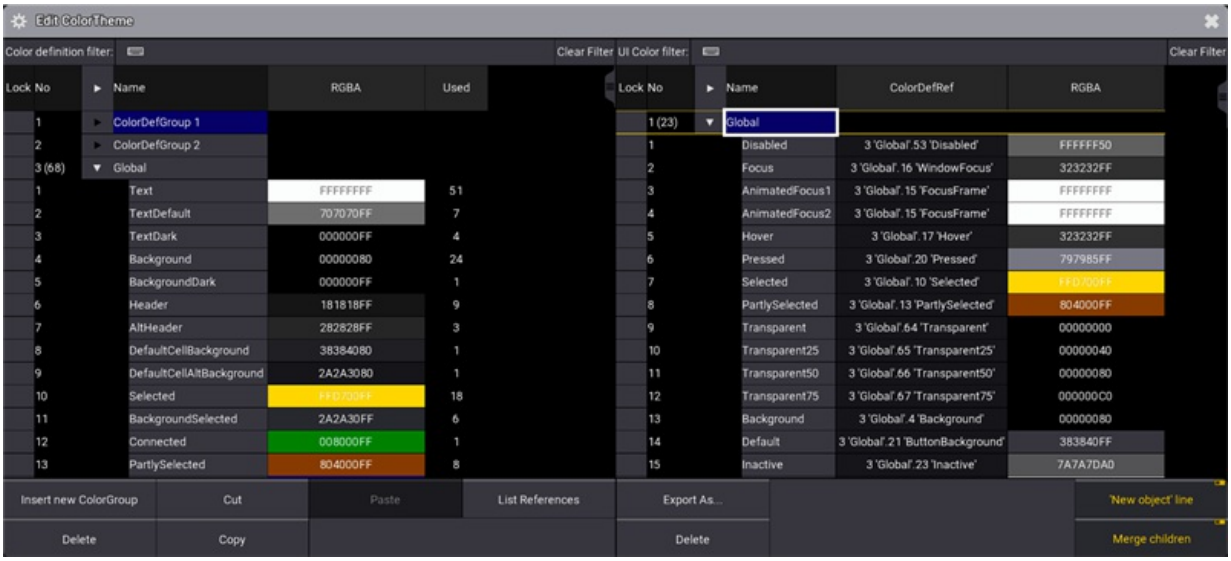
1. Open the menu.
2. Tap Desk Lights Color Theme in the top-right corner.
3. Tap Active Color Theme. This button will also display the name of the current color theme.
4. Tap the desired color theme from the list. If the color theme is on a USB drive, tap Internal in the top-right to select the drive.

Color themes can also be imported using the command line. For example, activate the "defaultDAYLIGHT" color theme using the following command:

```
User name[Fixture]>Import ColorThemeLibrary "defaultDAYLIGHT.xml" At ColorTheme /NoConfirm
```

Editing the Current Color Theme

In the main menu, tap Desk Lights Color Theme then Edit in the bottom-right corner to open the color theme editor.



Color theme editor

The left side of the color theme editor includes a list of color definitions. These function like color presets for user interface elements. Any changes to color definitions will be reflected in all of the UI elements referencing those definitions. This list shows the name of the color definition, the hex value of the color (including red, green, blue, and alpha values), and the number of UI elements referencing each color definition.

The right side of the color theme editor includes a list of user interface elements. This list shows the name of the UI color, a reference to a color definition, and the actual color defined by that reference. Changing the color definition reference to reference another color definition changes the color used for that element without altering any other elements. Deleting the reference for a UI element allows for a direct color definition for that element. This is analogous to storing a cue with hard values instead of referencing a preset.

Tap Export As... at the bottom of the menu to export the current color theme, either to the internal drive or to an external USB drive.

8. Function of the Command Line

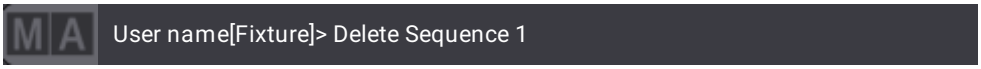
The command line is an essential way of communication between the console and its operator.

Using keywords, special characters, and numerical identifiers is how the operator tells the console to perform a task.

Important:
Depending on the use case, some commands are executed by entering **Please** in the command line.

Press Please in the command section.

Example



Or use the keys on the console as follows:

Delete + Sequ + 1 + Please

Keywords

The keywords that are used mostly have their own corresponding keys.

For more information on the multiple functions of the keys see the **Keys topic**.

Important:
All keywords can be typed in the command line using the keyboard.

To view commands ordered to the console, tap MA on the left of the command line.

The window Command Line History opens.

For more information on the usage of the command line in detail see **Command Line**.

8.1. General Syntax Rules

The command line syntax is used to create valid commands.

Object keywords are used to allocate objects in the show file.

Help keywords are used to create a relation between functions and objects.

Playback keywords provide control over playback functionalities.

Fader keywords are related to anything with faders.

For more information see **Command Line History**.

General Rules

The general rules are:

Important:
The basic syntax is as follows: **[Function] [Object]**

- All objects have a default function which is used if no function is given.
- Most functions have a default object or an object type which is used if no object is given.
- Objects are arranged in a hierarchical tree structure.
- If an object does not support the function applied, the function is passed on to a child or parent object.

Terminology

- [Square brackets]:
  Description of non-literal content.
- (Parentheses):
  Description of optional content.
- "Quotation marks"/'Quotation marks':
  Quotation marks are used to enter a definite name or content. If the line ends after the word in quotation marks, the quotation marks at the end may be omitted. If the name or content is not a keyword and does not contain special characters, the quotation marks can be left out altogether.

Important:
If an option or any other part of the keyword command requires two types of quotation marks - single (') and double quotation marks (") - make sure to always use an equal pair. For more information see the **RemoteCommand keyword**.

- Capitalization:
  In general, capitalization is important. Only when using keyword commands, the console does not distinguish between upper case and lower case. In such topics, capitalization is used to improve readability only.

Use the Command Line

It is possible to abbreviate all the commands using the shortcuts of the corresponding keywords.

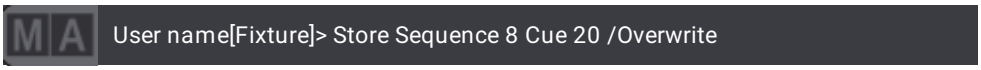
Each keyword has its own shortcut. Every keyword topic also names the respective shortcut.

For more information see **All Keywords**.

Example

- Store cue 20 in sequence 8 using the overwrite function.

Written-out version:



Abbreviated version:

Super short version (not used in the manual):

- To copy cue 2 to cue 6 of the selected sequence, type:

Used in the manual:

8.2. General Keywords

The following subtopics cover each of the available grandMA3 keywords. They are arranged in alphabetical order.

Each topic describes various means of entering the keyword into the command line, defines the keyword, shows its use with proper syntax, and includes usage examples.

8.2.1. ; [Semicolon] Character

To get the ; semicolon in the command line, press ;.

Description

The semicolon separates multiple commands.

Example

- To turn off sequence 5 and delete group 3, enter in the command line:

8.2.2. / [Slash] Character

To enter the slash / in the command line, press /.

Description

The slash character has two functions:

- It is used to set options during a command. For example, see **Store command**.
- It is used as a division sign in the calculator.

> **Important:**
> To set an option with the slash, make sure that the option is written right after the slash (without blank space).

Syntax

Keyword [object] "name" (/option)

Keyword [object] "name" (/option "option value")

Example

- To store screen 2 on view button 2.7, type:

```
User name[Fixture]> Store viewbutton 2.7 /screen "2"
```

8.2.3. . [Dot] Character

To enter the . (dot) in the command line, press .

Description

The . (dot) is a separator used for fractional digits and hierarchic IDs of objects.

Important:
Make sure that any text before and behind the . (dot) is written right after the . (dot) (without blank space).

Syntax

Keyword [object] [value] . [value fraction]

Keyword [object] [ID] . [child ID]

Example

- To add fixture 31.2 to preset 4.2, type:

```
User name[Fixture]> Fixture 31.2 At Preset 4.2
```

- To set the pan attribute to 50.5, type:

```
User name[Fixture]> Attribute "Pan" At 50.5
```

8.2.4. .. [DotDot] Keyword

To enter the .. [DotDot] keyword in the command line, type **..**

Description

The .. [DotDot] keyword is an object keyword used in connection with the ChangeDestination keyword.

For more information on changing the destination see the **ChangeDestination keyword**.

Syntax

ChangeDestination ..

The space between ChangeDestination and the two dots is important. If there is no space then the software cannot understand the command.

Example

- If the command line shows that the current command destination is inside the structure (in this example the current destination is Sequence folder), then the .. keyword can be used to move one level up in the structure until the destination is at the root.
- To enter a higher level in the destination of the command line, type:

8.2.5. = [Equal] Keyword

To enter the = [Equal] keyword in the command line, type **=**.

Description

The = [Equal] keyword is used when assigning a value to an objects setting.

Many objects have different settings and these settings can be set by using the setting name followed by the equal sign and then the desired value.

It is used in conjunction with the **Set keyword**.

Hint:
The equal sign might be omitted from the command.
If there is the possibility that a command can be misinterpreted, then it is a good idea to use it.

Syntax

Set [Object] [Setting] = [Value]

Example

- To set the Enabled option in macro line 1 of macro 3 to No, type:

```
User name[Fixture]> Set Macro 3.1 "Enabled" = "No"
```

8.2.6. <<< [GoFastBackward] Keyword

To enter the <<< GoFastBackward keyword in the command line, use one of the options:

- Press <<<
- Type **<<<**
- Type the shortcut **<**

Description

The <<< GoFastBackward keyword is a function keyword used to jump to the previous cue in a sequence, without cue timing.

Syntax

<<< [Sequence List]

<<< [Executor List]

<<< [Page List]

Example

- To jump to the previous cue in sequence 3, type:

- To jump to the previous cue in the sequence assigned to executor 103 (on the current page), type:

- To jump to the previous cue on all sequences assigned to executors on page 4, type:

For more information see the **<<<** [GoFastBackward] key.

8.2.7. >>> [GoFastForward] Keyword

To enter the >>> [GoFastForward] keyword in the command line, use one of the options:

- Press >>>
- Type **>>>**
- Type the shortcut **>**

Description

The >>> [GoFastForward] keyword is a function keyword used to jump to the next cue in a sequence, without cue timing.

Syntax

>>> [Sequence List]

>>> [Executor List]

>>> [Page List]

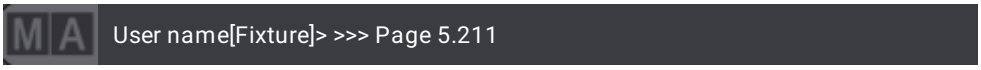
Example

- To jump to the next cue in sequence 3, type:

- To jump to the next cue in the sequence assigned on executor 103, type:

- To jump to the next cue on all sequences assigned to executors on page 4, type:

- To jump to the next cue on the sequences assigned to executor 211 on page 5, type:



For more information see the **>>>** [GoFastForward] key.

8.2.8. - [Minus] Keyword

To enter the - [Minus] keyword in the command line, use one of the options:

- Press -
- Type **-** [Minus]

Description

The - [Minus] keyword is used to remove objects from a list or to indicate negative values.

If the - [Minus] keyword is used to indicate values, it indicates absolute or relative values:

- A space between the - [Minus] and the value is automatically added. The space makes the value relative
- To obtain an absolute value, remove the space between the - [Minus]

Syntax

[At] - [Number]

[Object] [Number] - [Object] [Number]

[Object] -

Examples

- To reduce the percentage readout of pan by 10 percent, type:

```
User name[Fixture]> Attribute "Pan" At - 10
```

- To select the entire group 5 without selecting fixture 2, type:

```
User name[Fixture]> Group 5 - Fixture 2
```

- To reduce 10 % from the current dimmer value in the selected fixtures, type:

```
User name[Fixture]> At - 10
```

- To remove fixtures 5, 6, and 7 in the current selection of fixtures, type:

8.2.9. * [Asterisk] Character

To enter the Asterisk * in the command line, use one of the options:

- Press and hold Shift + 8 on the internal keyboard.
- Press and hold MA + / on the numeric keypad.

Description

The Asterisk * is a wildcard used to substitute any other character or characters in a name.

Syntax

[Object]

Examples

- To select the fixtures of all groups beginning with "Mac" in the group pool object name, type:

```
User name[Fixture]> Group "Mac*"
```

- To select all fixtures with a name beginning with "backt" and ending with "blue", type:

```
User name[Fixture]> Fixture "backt*blue"
```

8.2.10. % [Percent] Keyword

To enter the % [Percent] keyword in the command line, type %.

Description

The % [Percent] keyword is a helping keyword that can be used as a scaling operator.

Syntax

[At] % [Number]

Attribute "Attribute" [At]% [Number]

Examples

Requirement: The dimmer needs to be set to 70 % before.

- To reduce the dimmer value from 70 % to 35 % in a selected fixture, type:

- To set the tilt to 90 % from the current value in a selected fixture, type:

8.2.11. + [Plus] Keyword

To enter the + [Plus] keyword in the command line, use one of the options:

- Press +
- Type **+**

Description

The + [Plus] keyword is a helping keyword that has various functions.

It is used to combine multiple objects in a list or to indicate relative values. When used as a relative indicator, without a value, the value 1 is used.

If used as a starting keyword, the + [Plus] keyword creates a selection list which is added to the current selection.

Syntax

[At] + [Number]

[Object] [Number] + [Object] [Number]

Examples

- To delete cue 1 and 2 on the selected executor, type:

```
User name[Fixture]> Delete Cue 1 + 2
```

- To add 5 % to the current dimmer value, type:

```
User name[Fixture]> At + 5
```

- To add fixtures 5, 6 and 7 to the current selection, type:

```
User name[Fixture]> Fixture + 5 Thru 7
```

8.2.12. Absolute Keyword

To enter the Absolute keyword in the command line, use one of the options:

- Press MA + Time + Time + Time + Time
- Type **Absolute**
- Type the shortcut **Ab**

Description

The Absolute keyword is used to set the absolute layer.

Syntax

Absolute

[Attribute "Name"] [At] Absolute [Value]

Examples

- To set the layer to absolute, type:

- To set the color red to 10 in the programmer, type:

8.2.13. Acceleration Keyword

To enter the Acceleration keyword in the command line, use one of the options:

- Type **Acceleration**
- Type the shortcut **Acc**

Description

The Acceleration keyword is used to set the acceleration curve of a step in the phaser.

For more information see **Phasers**.

Syntax

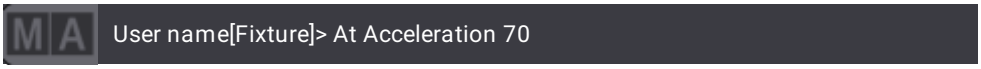
Acceleration

[At] Acceleration [Value]

Examples

- To set the layer to accelerate, type:

- To set the acceleration curve of the dimmer to 70, type:



8.2.14. Action Keyword

To enter the Action keyword in the command line, use one of the options:

- Type **Action**
- Type the shortcut **Actio**

Description

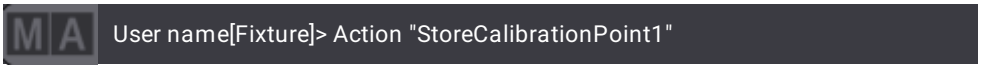
The Action keyword is used to call functions that do not have a designated keyword.

Syntax

Action [Funtion]

Examples

- To store the pan/tilt position to calibration point 1 of the currently selected fixtures, type:



- To call the pan/tilt position of the currently selected fixture of calibration point 2 into the programmer, type:

- To solve the stage calibration, type:

8.2.15. ActivationGroup Keyword

To enter the Action keyword in the command line, use one of the options:

- Type **ActivationGroup**
- Type the shortcut **AG**

Description

The ActivationGroup keyword represents attributes that are activated together.

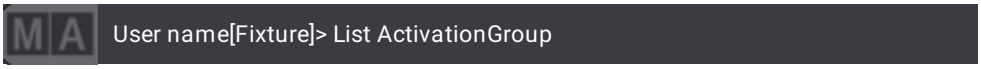
For more information see **Activation Group**.

Syntax

[Function] ActivationGroup

Example

- To list all activation groups, type:



8.2.16. Agenda keyword

To enter the Agenda keyword in the command line, use one of the options:

- Type **Agenda**
- Type the shortcut **Age**

Description

The Agenda keyword represents the single agenda events.

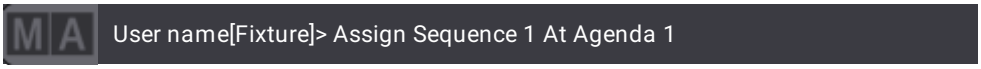
For more information see **Agenda**.

Syntax

[Function] Agenda [ID/"Name"]

Examples

- To assign sequence 1 to the agenda event 1, type:



- To give the first agenda event the name "Sunset", type:

- To change the mode of the second agenda event to "Dawn", type:

8.2.17. Align Keyword

To enter the Align keyword in the command line, use one of the options:

- Press Align
- Type **Align**
- Type the shortcut **Al**

Description

The Align keyword is a function keyword used to toggle through the align modes of the attribute encoders.

- To toggle through the modes, repeatedly press Align.

Align Modes	Index Number
Off	0
/	1
<	2
>	3
><	4
<>	5

To read more about the align modes, see **operate fixtures - align**.

Syntax

Align [Mode]

Align [Index Number]

Examples

- To set the encoders to the first align mode, type:

- To turn off any align mode, type:

8.2.18. AlignTransition Keyword

To enter the AlignTransition keyword in the command line, use one of the options:

- Type **AlignTransition**
- Type the shortcut **AlignT**

Description

The AlignTransition keyword is used to toggle between the align modes

- Linear
- Sinus
- Slow
- Fast

Syntax

AlignTransition

AlignTransition [Form Name]

Examples

- To set the AlignTransition to Sinus, type:

```
User name[Fixture]> AlignTransition "Sinus"
```

- To toggle the AlignTransition to the previous form, type:

```
User name[Fixture]> AlignTransition Previous
```

- To toggle the AlignTransition to the next form, type:

```
User name[Fixture]> AlignTransition Next
```

8.2.19. Appearance Keyword

To enter the Appearance keyword in the command line, use one of the options:

- Press MA + X4 + X4
- Type **Appearance**
- Type the shortcut **Ap**

Description

The Appearance keyword is used to assign an appearance to an object.

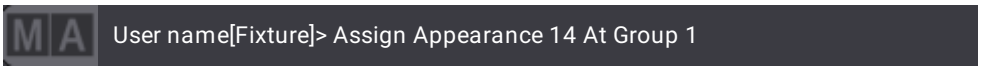
For more information see **Appearance**.

Syntax

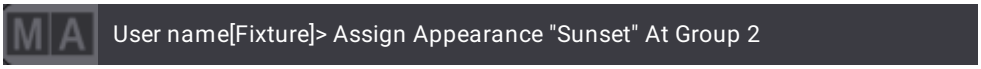
[Assign] Appearance [Number] [At] [Object] [Number]

Examples

- To assign appearance 14 to group 1, type:



- To assign appearance "Sunset" to group 2, type:



8.2.20. Assign Keyword

To enter the Assign keyword in the command line, use one of the options:

- Press Assign
- Type **Assign**
- Type the shortcut **As**

Description

The Assign keyword is a function used to assign an object to another object.

To set parameters for objects, use the **Set keyword**.

For more information see:

- **Assign macros to keys and buttons**
- **Assign input filter**
- **Use appearances**
- **Store and recall views**

Syntax

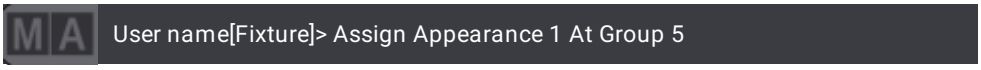
Assign [Object] At [Object]

Examples

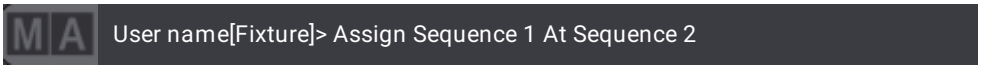
- To assign macro 2 at executor page 1 executor 402, type:

- To assign view 2 at display 2 and view button 3, type:

- To assign appearance 1 at group 5, type:



- To assign a sequence to an empty sequence in the pool get a playback independent but content linked copy, type:



8.2.21. At Keyword

To enter the At keyword in the command line, use one of the options:

- Press At
- Type **At**
- Type the shortcut **A**

Description

The At keyword is a function keyword and a helping keyword at once.

- As a function keyword it is used to apply values.
- As a helping keyword it is used along with other function keywords to indicate destination.

Hint:
At applies values live in the programmer. For information on how to apply values throughout the show file see the **Clone keyword**.

At is "the exception that proves the rule". At is one of the few functional keywords which accept objects before the function.

As a starting keyword, At is a function that applies values in the programmer to the current selection.

If value type Fade or Delay is used, the value list will be applied as individual fade/delay times.

Following an object list, At is a function that applies values to the object list. If the object list does not support the At function, the object list is resolved into a selection list and At applies values in the programmer.

Following an object list that follows a function, At is a helping keyword for the starting function.

Syntax

At [Value List]

At [Value Type] [Value List]

At [Object List]

[Object List] At [Value List]

[Object List] At [Value Type] [Value List]

[Object List] At [Object List]

[Destination Selection List] At [Source Selection List] (If [Scope Object List]) (Option)

Helping keyword

[Function] [Object List] At [Object List]

Example

- To set the MasterFader of the sequence 1 to 30 %, type:

- To set the dimmer attributes of the current selection to 75%, type:

- To set the fixture selection to the values of cue 3 from the selected sequence, type:

- To set the pan attributes of the selected fixtures to 20%, type:

- To set an individual delay time of 2 seconds to attribute 2, type:

- To copy group 4 to group 10, type:

- To set a speed to 60 using the speed readout specified in the user profile (e.g., BPM), type:

8.2.22. Attribute Keyword

To enter the Attribute keyword in the command line, use one of the options:

- Press Preset Preset
- Type **Attribute**
- Type the shortcut **Att**

Description

The Attribute keyword is an object keyword used to set attributes of a fixture.

The default function of attributes is Call. If you call an attribute, you can use the encoders to modify the values. Calling attributes also selects the attributes in the fixture sheet.

Important:
The number of an attribute may vary if new fixtures and attributes are added to the show file. We recommend you use the unique library name of attributes.

- Attributes are organized by subattributes.
- Subattributes are organized by features.
- Features are organized by feature groups.

Attribute "Name"

Attribute [Number]

Examples

- To view the list of attributes along with their corresponding names and numbers in the command line history, type:

- To set the attribute "pan" to 120 degrees for the selected fixtures, type:

- Knock out the first attribute (Dim) in the current selection, type:

8.2.23. AutoCreate Keyword

To enter the AutoCreate keyword in the command line, use one of the options:

- Type **AutoCreate**
- Type the shortcut **Au** or **Ac**

Description

The AutoCreate keyword creates objects depending on predefined source objects. It can, for example, create groups in the fixture types of the patched fixtures.

Hint:
If your selection is defined as source list, the active MAtricks settings will be taken into cosideration whenever executing AutoCreate.

Syntax

AutoCreate [Source List] At [Destination List] (/Option)

Options

- /All
- /Merge
- /Overwrite
- /Single

Examples

- To create single fixture groups starting with group pool object 1 using all selected fixtures, type:

```
User name[Fixture]> AutoCreate Selection At Group 1
```

Requirement:

1. Insert at least ten fixture types in the show file and patch at least two fixtures of every fixture type.

- To create a group in the group pool object 42 containing all fixtures of fixture type 10, type:

```
User name[Fixture]> AutoCreate FixtureType 10 At Group 42
```

- To create single fixture groups starting with group pool object 42 that contain all patched fixtures of the fixture type 9, type:

- To create single fixture groups that contain all patched fixtures of fixture types 9 and 10 starting with group pool object 101, type:

Requirement:

1. Select eight fixtures.
2. Set MAtricks XGroup to 2.
3. Set MAtricks X to 0.

- To create group 21 in the group pool containing all main fixtures of the selection (odd fixtures of the current selection), type:

Requirement:

1. Create layers and classes withtin the patch.
2. Set fixtures to these layers and classes.

Hint:
The demoshow already uses these settings.

- To create single fixture groups starting with group pool object 201 using all patched fixtures that are set to the layer "Backtruss" within the patch, type:

- To create a group in the group pool object 301 using all patched fixtures that are set to class "Spots" within the patch, type:

- To create single fixture groups starting with group pool object 401 using all fixtures of fixture type 9 that are also set to fixture class "Spots", type:

8.2.24. Black Keyword

To enter the Black keyword in the command line, use one of the options:

- Press MA + <<<
- Type **Black**
- Type the shortcut **Bla**

Description

The Black keyword is used to temporarily override the master level to zero on executing objects.

Syntax

Black [Executor List]

Black On [Executor List]

Black Off [Executor List]

Examples

- To override the master level of executor 201 to zero, type:

- To return the master level of executor 201 to the master fader, type:

8.2.25. Blind Keyword

To enter the Blind keyword in the command line, use one of the options:

- Press Blind
- Type **Blind**
- Type the shortcut **Bl**

Description

The Blind keyword is used to suppress the output of the live programmer. It is possible to program without putting the content live. Disabling Blind activates the programmer and the changes made during the mode.

Use the **DMX sheet** to see that the output is set to 0.

Hint:
Blind is a toggle function meaning that entering Blind without using a helping keyword enables or disables Blind.

Syntax

Blind

Blind On

Blind Off

Example

- To enable Blind and to disable the output of the programmer in reverse, type:

8.2.26. Block Keyword

To enter the Block keyword in the command line, use one of the options:

- Type **Block**
- Type the shortcut **Blo**

Description

Block is a function used to add data and prevent them from tracking. Tracking values are converted to stored values.

If the object list does not contain any references to any cues, the Block function is applied to the selected sequence.

If syntax does not contain any selection list filter, all fixtures will be used.

If syntax does not contain any attribute list filter, all attributes will be used.

Syntax

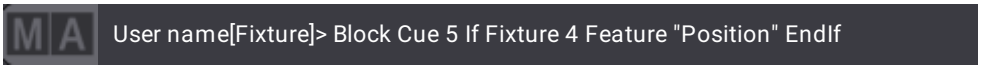
Block [Object-list] If [Selection List] [Attribute List] EndIf

Examples

- To block all parameters in cue 2 of the selected sequence, type:

To unblock parameters the **keyword Unblock** can be used in the same way as the keyword **Block**.

- To block pan and tilt of Fixture 4 in Cue 5 of the selected sequence, type:



8.2.27. BPM Keyword

To enter the BPM keyword in the command line, use one of the options:

- Type **BPM**
- Type the shortcut **Bp**

Description

The BPM keyword is used to set the speed of a fixture selection using the unit BPM.

Syntax

[At] [Speed] BPM [Value]

Example

- To set the speed layer to 5 bpm, type:

8.2.28. Call Keyword

To enter the Call keyword in the command line, use one of the options:

- Press On On
- Type **Call**
- Type the shortcut **Cal**

Description

The Call keyword is used to apply an object or its content.

If the Call keyword is used on objects with parameters, these parameters are loaded into the programmer without selecting the fixtures.

Syntax

Call [object list]

Call [object list] (/option)

Examples

- To call view button 2.1, type:

- To call macro 2, type:

8.2.29. Camera Keyword

To enter this keyword into the console, use one of the options:

- Type **Camera**
- Type the shortcut **Cam**

Description

The Camera keyword selects a camera in the camera pool.

Syntax

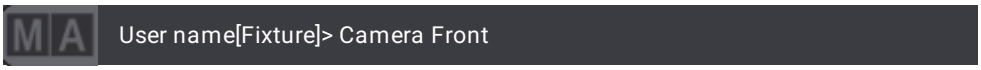
Camera [ID]

Camera [Name]

Examples

- To display the auto camera of the 3D, type:

- To display the front camera of the 3D, type:



8.2.30. Capture Keyword

To enter the Capture keyword in the command line, use one of the following options:

- Press Stomp Stomp
- Type **Capture**
- Type the shortcut **Cap**

Description

The Capture keyword is a command keyword used to activate the current output values of specified parameters.

Capturing parameters during a running phaser or cue fade activates the actual values at the moment the command is executed. The result is a single step of static values resembling a freeze-frame of the output.

Important:
The capture command produces only numeric values, losing any existing preset references.

Important:
The capture command translates the output of each parameter into the appropriate value on the absolute layer while activating a 0 value on the relative layer. For any parameter where the output was previously composed of a combination of absolute and relative values, the output will appear the same while relying only on the absolute layer with no relative offset.

Syntax

Capture [Object]

Examples

- To capture all attributes of the selected fixtures, type:

```
User name[Fixture]> Capture
```

- To capture only the position attributes of the selected fixtures, type:

```
User name[Fixture]> Capture FeatureGroup "Position"
```

- To create the same result using an alternate method:

1. Select the fixtures using any desired method (for example, groups, selection tools in the layout or 3D windows, or the **SelFix** command).
2. Press Stomp Stomp to enter the **Capture** keyword in the command line.
3. Press the Position feature group button in the encoder bar.

8.2.31. Certificate Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

Only use this keyword if instructed by a support technician from MA Lighting or from a local distributor.

8.2.32. ChangeDestination Keyword

To enter the CD keyword in the command line, use one of the options:

- Type **ChangeDestination**
- Type the shortcut **CD**
- Type the shortcut **CH**

Description

The ChangeDestination keyword is a function keyword used to change the current destination of the command line.

Syntax

ChangeDestination [Element Index]

ChangeDestination "Element Name"

ChangeDestination [Object Type] [Object ID]

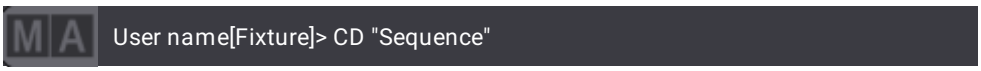
ChangeDestination ..

Examples

- To enter the first element of the current destination, type:

Result:

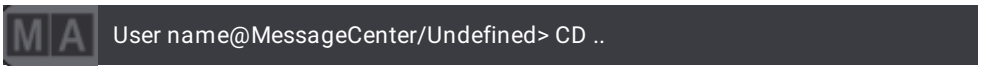
- To enter the element of the current destination called "Sequence", type:



- To leave the destination "Sequence", type:

Result:

- To go one level back in the tree structure, type:



Result:

8.2.33. Channel Keyword

To enter the Channel keyword in the command line, use one of the options:

- Press Channel
- Type **Channel**
- Type **F** + **1**

Description

The Channel keyword is an object keyword used to select fixtures of the ID type Channel.

For more information see **Assign an ID to the fixtures**.

Use the Channel keyword as a default keyword for the command line when programming with channels. For more information see **Workspace - Command Line - Change the Default Keyword**.

Syntax

Channel [Channel ID]

Example

- To select the channel with channel ID 10, type:

8.2.34. ChannelSet Keyword

To enter the ChannelSet keyword in the command line, use one of the options:

- Type **ChannelSet**
- Type the shortcut **Chann**

Description

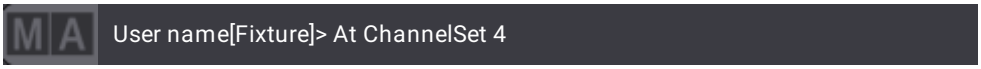
The ChannelSet keyword is used to take channel sets, such as gobos, actively into the programmer.

Syntax

[At] ChannelSet [ID]

Example

- To activate the values of channel set 4 in the programmer, type:



8.2.35. Chat Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.36. Cleanup Keyword

To enter the Cleanup keyword in the command line, type **Clean**.

Description

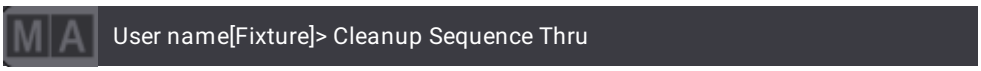
The Cleanup keyword is a command keyword used to delete unused (contains no reference) objects in the show file. For example, sequences that are not assigned to an executor.

Syntax

Cleanup

Example

- To delete all unassigned sequences, type:



- To delete all unused color presets in preset pool 4, type:

- To delete all images without reference, type:

8.2.37. Clear Keyword

To enter the Clear keyword in the command line, use one of the options:

- Press Clear
- Type **Clear**
- Type the shortcut **CL**

Description

The Clear keyword is a function keyword used to clear the selection, active values, or the programmer.

Depending on the status of the programmer Clear successively:

1. clears selection (deselects all fixtures)
2. clears active values (deactivates all values)
3. clears all (empties the entire programmer)

For information on the key see the **Clear Key**.

For more information on each function of the Clear keyword see **ClearSelection**, **ClearActive**, **ClearAll**.

The Clear button in the AT Filter window executes the Clear command. For more information see **At overlay**.

Syntax

Clear

Example

- To clear the selection, active values or the programmer depending on the status/content of the programmer, type:

8.2.38. ClearActive Keyword

To enter the ClearActive keyword in the command line, use one of the options:

- Type **ClearActive**
- Type the shortcut **Ca**

Description

The ClearActive keyword is used to deactivate all values in the programmer.

For information on the key see the **Clear Key**.

For more information on the additional functions of the Clear keyword see **Clear**, **ClearAll**, **ClearSelection**.

The Clear button in the AT Filter window executes the Clear command. For more information see **At overlay**.

Syntax

ClearActive

Example

- To deactivate any active values in the programmer, type:

8.2.39. ClearAll Keyword

To enter the ClearAll keyword in the command line, use one of the options:

- Type **ClearAll**
- Type the shortcut **ClearA**

Description

The ClearAll function clears the selection and discards all values in the programmer.

For information on the key see the **Clear Key**.

For more information on the additional functions of the Clear keyword see **Clear**, **ClearActive**, **ClearSelection**.

The Clear button in the AT Filter window executes the Clear command. For more information see **At overlay**.

Syntax

ClearAll

Example

- To clear the programmer, type:

8.2.40. ClearSelection Keyword

To enter the ClearSelection keyword in the command line, use one of these options:

- Type **ClearSelection**
- Type the shortcut **Cs**

Description

The ClearSelection keyword is a function keyword used to deselect the selected fixtures.

For information on the key see the **Clear Key**.

For more information on the additional functions of the Clear keyword see **Clear**, **ClearActive**, **ClearAll**.

The Clear button in the AT Filter window executes the Clear command. For more information see **At overlay**.

Syntax

ClearSelection

Example

- To deselect selected fixtures, type:

8.2.41. Clone Keyword

To enter the Clone keyword in the command line, use one of the options:

- Press MA + X1
- Type **Clone**
- Type **Clo**

Description

Hint:
Clone replicates data from one fixture or selection to another throughtout the show file. For more information on applying values live in the programmer only see the **At keyword**.

Important:
If no object type is specified, using the keword **If**, Clone only adds data within the programmer. Adding the keyword **If** changes the scope of cloning from the programmer to other objects.

Syntax

Clone [Source Selection List] At [Destination Selection List] (If [Scope Object List]) (Option)

Options

- **/MergeLowPriority** or **/MergeL**:
  Adds data from the source to the destination only where the destination currently contains no data. This option preserves existing data at the destination and is the least destructive form of cloning.
- **/MergeHighPriority** or **/MergeH**:
  Adds all available data from the source to the destination. Data from the source overwrites data at the destination.
- **/Overwrite** or **/O**:
  Replaces all data at the destination with data from the source. This option deletes information at the destination where there is no corresponding data at the source.

Examples

- To clone the data from fixture 1 to fixture 2 in the programmer, type:

```
User name[Fixture]> Clone Fixture 1 At Fixture 2
```

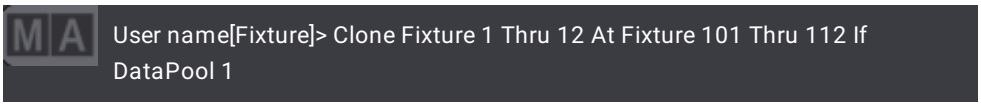
The Clone pop-up appears, offering the choices of **Low Priority**, **High Priority, Overwrite**, and **Cancel**.

- To clone data from fixture 1 to fixture 2 in the programmer as a low priority merge, type:

- To create programmer selective data for fixture 2 using the option **Overwrite**, type:

- To clone data from fixtures 1 and 2 to fixtures in group 10 within sequence 1 thru 10 only, type:

- To clone all objects within data pool 1 from fixtures 1 thru 12 to fixtures 101 thru 112, type:



For more information on data pools, see the **Data Pools** topic.

Hint:
When cloning preset and cue data, all presets referenced by the destination objects are automatically cloned in addition.

Hint:
The syntax:
Clone [Source Selection List] At [Destination Selection List] (If [Scope Object List]) (Option)
… produces the same results as:
[Destination Selection List] At [Source Selection List] (If [Scope Object List]) (Option)
All options described above work the same for both syntax structures. When cloning without the **Clone** keyword, keep in mind that the source and destination change places within the syntax.
For more information on the **At** keyword, see the **At keyword** topic.

8.2.42. CmdDelay Keyword

To enter the CmdDelay keyword in the command line, use one of the options:

- Type **CmdDelay**
- Type the shortcut **Cmdd**

Description

The CmdDelay sets the command line delay time of an object.

Syntax

CmdDelay [Value List]

Example

- To create cue 3 and set the cue's cmd delay to 4 seconds, type:

8.2.43. Collect Keyword

To enter the Collect keyword in the command line, use one of the options:

- Press Select Select
- Type **Collect**
- Type the shortcut **Coll**

Description

The Collect keyword is used to add objects of the same type to a collection. For more information on collection see **Collection Keyword**.

Hint:
It is possible to collect empty pool objects during import or when storing.

Syntax

Collect [Object List]

Examples

- To add the second preset of the color preset pool to the collection, type:

```
User name[Fixture]> Collect Preset 4.2
```

- To add a different color preset to the collection, type:

```
User name[Fixture]> Collect Preset 4.10
```

- To reset a collection, type:

```
User name[Fixture]> Collect
```

8.2.44. Collection Keyword

To enter the Collection keyword in the command line, use one of the options:

- Press Fixture + Fixture + Fixture
- Type **Collection**
- Type the shortcut **Collecti**

Description

The Collection keyword is an object keyword used to represent the current collection of objects.

Important:
A collection stays always at the same spot it was collected at.

Hint:
Pool objects that are part of a collection have an orange frame around them. They also display the order of the collection which is located on top of a pool object.

Syntax

[Function] Collection (At) (Object List)

Examples

Requirement:

Create a collection of color presets

- To move the collection to the twentieth color preset, type:

8.2.45. ColorTheme keyword

To enter the ColorTheme keyword in the command line, use one of the options:

- Type **ColorTheme**
- Type the shortcut **Col**

Description

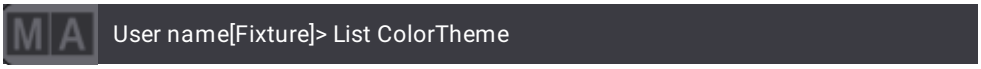
ColorTheme is an object keyword used, for example, to adjust the color theme.

Syntax

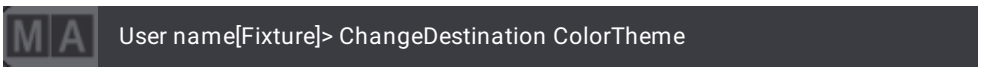
[Function] ColorTheme

Examples

- To list all the elements of the color theme, type:



- To enter the ColorTheme folder in the data structure, type:



8.2.46. Configuration Keyword

To enter the Configuration keyword in the command line, use one of the options:

- Type **Configuration**
- Type the shortcut **Con**

Description

A configuration contains a set of different button, key, fader, and encoder functions. It can be assigned to an executor.

Syntax

Configuration [Number]/[Name] At [Number of Executor]

Example

To assign configuration 2 to executor 101, type:

8.2.47. Console Keyword

To enter the Console keyword in the command line, use one of the options:

- Type **Console**
- Type the shortcut **Cons**

Description

The Console keyword is an object keyword used to display all consoles in the network.

This keyword belongs to a session-related group of keywords. For more information see **Node keyword**, **onPC keyword**, **PU keyword**, **Station keyword**, **Extension keyword**, **Join keyword**, **Leave keyword**, **Invite keyword**, **Dismiss keyword**, **Reboot keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

Console [ID]

Console [Name]

Examples

- To list all consoles that are in the same network, type:

- To invite console "FOH" to the session, type:

8.2.48. Cook Keyword

To enter the Cook keyword in the command line, use one of the options:

- Press MA + Update.
- Type **Cook**

Description

The Cook keyword is a command keyword used to cook the recipes of objects without entering the editor.

Syntax

Cook [object]

Cook [object] (/option)

There are three options:

- **Merge** - Replace cooked data and add new data based on the recipe ingredients, but do not replace non-cooked data.
- **Overwrite** - Replace cooked data and add new data based on the recipe ingredients and replaces non-cooked data.
- **Remove** - Replace all cooked data from the cue part.

Examples

To cook the recipes of preset 1.1, type:

```
User name[Fixture]> Cook Preset 1.1
```

To remove all cooked data from sequence 1, type:

```
User name[Fixture]> Cook Sequence 1 /Remove
```

For more information on Recipes, see **Cue Recipes**.

8.2.49. Copy Keyword

To enter the Copy keyword in the command line, use one of these options:

- Press Copy
- Type **Copy**
- Type the shortcut **Co**

Description

The Copy keyword is a function keyword used to create copies of an object.

If no object type is given and the command line destination is root (no destination), the default object type **– Cue –** is used for this function.

Syntax

Copy [Object_Type] [Source_ID] At [Destination_ID] (Options)

Copy [Object_Type] [Source_ID_List] At [Destination_ID_Start] (Options)

Copy [Object_Type] [Source_ID] At [Destination_List] (Options)

Copy [Object_Type] [Source_ID_List] (Options)

Options

The following command options can be added to the command:

- **/Overwrite** or **/O**:
  Overwrites the existing data at the destination. When copying cue, the option /CopyCueDst should be used in addition to defining how to overwrite the data at the destination.
- **/Merge** or **/M**:
  Merges the data from the source to the destination.
- **/NoConfirm** or **/NC**:
  Suppresses the copy pop-up. If this option is not set, the copy pop-up opens and the options can be manually changed.

When copying cues there are several other options. This is the general cue copy syntax:

Copy Cue [Source_ID] At [Destination_ID] (Options)

These are the additional command options:

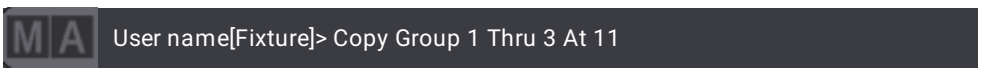
- **/CopyCueSrc** or **/CCS**:
  Defines how the source cue is taken for the copy. One of these values should be defined.
  - **"Content"**: The Content option is forced.
  - **"Status"**: The Status option is forced.
  - **"Look"**: The Look option is forced.
- **/CopyCueDst** or **/CCD**:
  CCD defines what happens with data that is not touched by the copy operation but tracks into the destination cue. Should it still track into that cue (Keep), or should these attributes set to release or default values.
  - **"Keep"**: Merges the data to the destination. Makes the most sense in combination with the overall merge action (see /Merge).
  - **"ForceRelease"**: Sets the data that is part of the destination, but not of the source to Release values in the destination. Makes the most sense in combination with the overall overwrite action (see /Overwrite).
  - **"ForceDefault"**: Sets the data that is part of the destination, but not of the source to default values in the destination. Makes the most sense in combination with the overall overwrite action (see /Overwrite).
- **/CueOnly** or **/CO**:
  Defines if the copy action should be executed with cue only. One of these values could be defined. If nothing is defined then it is the same as "On".
  - **"Off"**: Tracking is used.
  - **"On"**: Cue only function is On.
  - **"OnDefaultNew"**: Cue only is On and unspecified values are set to the default value for the attribute.

Learn more about cue copying in the **Copy Cues topic**.

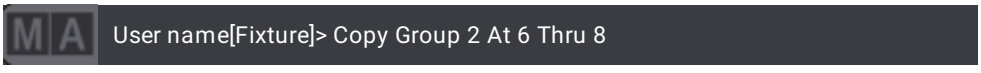
Examples

- To copy group 1 to group 5, type:

- To copy group 1 to group 11; group 2 to group 12; and group 3 to group 13, type:

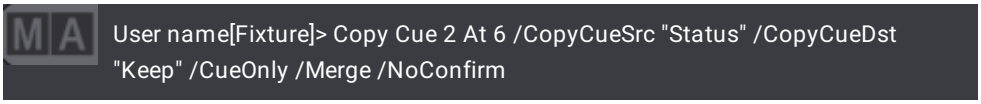


- To copy group 2 to group 6, 7, and 8, type:



- To copy cue 2 to cue 6 of the selected sequence, type:

- To copy cue 2 to cue 6 of the selected sequence with some copy options, type:



- To copy macro 2 to macro 6, type:

8.2.50. Cue Keyword

To enter the Cue keyword in the command line, use one of the options:

- Press Cue
- Type **Cue**

Description

Important:
Cue is the only object type that accepts numerical ID as decimal fractions. The ID allowed for cues ranges from 0.001 to 9999.999. In all other objects, a dot indicates the ID of a parent or a child object.

Cue is an object type holding a look on stage.

Cues are arranged in sequences and are divided into parts.

For more information see the **Cue and Sequence section**.

The **Cue** keyword is an object keyword. Object keywords need a function keyword in front of them to create a complete command. Learn more about this is the **General Syntax Rules** topic.

Cue has a default function called **SelFix**. This selects all the fixtures that have stored values in the cue.

If a sequence is not specified, then the selected sequence is used in the command.

Syntax

[Function] Cue [Cue_ID]

[Function] Sequence [Sequence_ID] Cue [Cue_ID]

[Function] Sequence [Sequence_ID] Cue [Cue_ID] ([Setting] [Setting_Option])

Examples

- To select the fixtures with values stored in cue 3 of a selected sequence, type:

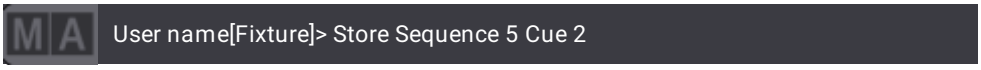
```
User name[Fixture]> Cue 3
```

The default function for **Cue** is SelFix so **Cue 3** is the same as **SelFix Cue 3**.

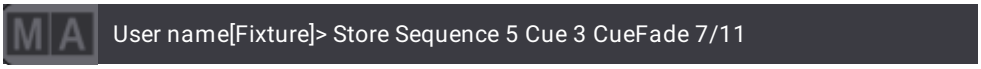
- To delete cue 2.5 in the selected sequence, type:

```
User name[Fixture]> Delete Cue 2.5
```

- To store cue 2 in sequence 5, type:



- To store cue 3 in sequence 5 with a cue fade time of 7 seconds and an outfade of 11, type:



8.2.51. CueAbs Keyword

To enter the CueAbs keyword in the command line, use one of the options:

- Type **CueAbs**
- Type the shortcut **CueA**

Description

CueAbs is a layer keyword. It is used to change the selected layer to the CueAbs layer. This layer can show information in, for instance, the fixture sheet. The layer shows the cue ID number for each parameter that has an absolute value from the active cues.

If a fixture sheet is set to **Auto** layer, then it will automatically change to show the selected layer.

The cue ID format is [Sequence Number].[Cue Number]:[Part Index]

Syntax

CueAbs

Examples

- To select the CueAbs layer, type:

8.2.52. CueDelay Keyword

To enter the CueDelay keyword in the command line, use one of the options:

- Press Time Time (If the Time Key Target is set to Cue. For more information see **User Settings** and **Time Key**.)
- Type **CueDelay**
- Type the shortcut **Cued**

Description

CueDelay can set both the indelay and the outdelay time of a cue. To do so, use a /. See examples further down.

For more information on how to set the delay times in objects see the **Delay Keyword**.

Syntax

CueDelay [Value]

Examples

- To enter a delay of 5 seconds in the current cue of the selected sequence, type:

```
User name[Fixture]> CueDelay 5
```

- To set an indelay of 6 seconds and an outdelay of 12 seconds in the current cue of the selected sequence, type:

```
User name[Fixture]> CueDelay 6/12
```

- To adjust the CueInDelay to 3 seconds, but leave the CueOutDelay as it was, type:

```
User name[Fixture]> CueDelay 3/
```

- To enter a delay of 5 seconds in cues 1 to 4 of the selected sequence, type:

```
User name[Fixture]> Cue 1 Thru 4 CueDelay 5
```

8.2.53. CueFade Keyword

To enter the CueFade keyword in the command line, use one of the options:

- Press Time (If the Time Key Target is set to Cue. For more information see **User Settings** and **Time Key**.)
- Type **CueFade**
- Type **Cuef**

Description

CueFade can set both the infade and the outfade time of a cue. To do so, use a /. See examples further down.

To set the fade times, read more in the **Fade Keyword**.

Syntax

CueFade [Value]

Examples

- To enter a fade of 5 seconds in a cue, type:

```
User name[Fixture]> CueFade 5
```

- To set an infade of 6 seconds and an outfade of 12 seconds in the current cue of the selected sequence, type:

```
User name[Fixture]> CueFade 6/12
```

- To adjust the CueInFade to 3 seconds, but leave the CueOutFade as it was, type:

```
User name[Fixture]> CueFade 3/
```

- To enter a fade of 5 seconds in cues 1 to 4, type:

```
User name[Fixture]> Cue 1 Thru 4 CueFade 5
```

- To enter a fade time of 1 hour 22 minutes 56.3 seconds in cue 1 of the selected sequence, type:

Or press:

Time 1 . . . 2 2 . . 5 6 . 3

- To double the CueFade time, type:

- To subtract 3 seconds from the CueFade time, type:

8.2.54. CueInDelay Keyword

To enter the CueInDelay keyword in the command line, use one of the options:

- Type **CueInDelay**
- Type the shortcut **Cueind**

Description

CueInDelay sets the delay time of a cue.

Syntax

CueInDelay [Value]

Examples

- To enter an indelay of 5 seconds in the current cue of the selected sequence, type:

- To enter an indelay of 5 seconds in cues 1 to 4 of the selected sequence, type:

8.2.55. CueInFade Keyword

To enter the CueInFade keyword in the command line, use one of the options:

- Type **CueInFade**
- Type the shortcut **Cuein**

Description

CueInFade sets the infade time of a cue.

Syntax

CueInFade [Value]

Examples

- To enter an infade of 5 seconds in the current cue of the selected sequence, type:

8.2.56. CueOutDelay Keyword

To enter the CueOutDelay keyword in the command line, use one of the options:

- Type **CueOutDelay**
- Type the shortcut **Cueoutd**

Description

CueOutDelay sets the outdelay time of a cue.

Hint:
CueOutDelay is only used by dimmer parameters that go to a lower value in the cue.

Syntax

CueOutDelay [Value]

Examples

- To enter an outdelay of 5 seconds in the current cue of the selected sequence, type:

```
User name[Fixture]> CueOutDelay 5
```

- To enter an outdelay of 5 seconds in cues 1 to 4 of the selected sequence, type:

```
User name[Fixture]> Cue 1 Thru 4 CueOutDelay 5
```

8.2.57. CueOutFade Keyword

To enter the CueOutFade keyword in the command line, use one of the options:

- Type **CueOutFade**
- Type the shortcut **Cueo**

Description

CueOutFade sets the outfade time of a cue.

Hint:
CueOutFade is only used by dimmer parameters that go to a lower value in the cue.

CueOutFade [Value]

Examples

- To enter an outfade of 5 seconds in the current cue of the selected sequence, type:

```
User name[Fixture]> CueOutFade 5
```

- To enter an outfade of 5 seconds in cues 1 to 4 of the selected sequence, type:

```
User name[Fixture]> Cue 1 Thru 4 CueOutFade 5
```

8.2.58. CueRel Keyword

To enter the CueRel keyword in the command line, use one of the options:

- Type **CueRel**
- Type the shortcut **CueR**

Description

CueRel is a layer keyword. It is used to change the selected layer to the CueRel layer. This layer can show information in, for instance, the fixture sheet. The layer shows the cue ID number for each parameter that has a relative value from the active cues.

If a fixture sheet is set to **Auto** layer, then it will automatically change to show the selected layer.

The cue ID format is [Sequence Number].[Cue Number]:[Part Index]

Syntax

CueRel

Examples

- To select the CueRel layer, type:

8.2.59. CueUpdate Keyword

To enter the CueUpdate keyword in the command line, use one of the options:

- Type **CueUpdate**
- Type the shortcut **CueU**

Description

CueUpdate contains all cues that could be updated. The CueUpdate keyword is an Object keyword.

It needs the **List keyword** to show the list of cues.

It can be prefixed with the **Update** keyword to update an element of the list.

Syntax

List CueUpdate

Update CueUpdate [List_Number]

Example

- To list all the cues that can be updated in the command line feedback, type:

It is the same list visible in the **Update menu**.

- To update the first cue in list, type:

Be aware that the number relates to the first element in the list - it is not the cue ID number.

8.2.60. CurrentUser Keyword

To enter the CurrentUser keyword in the command line, use one of the options:

- Type **CurrentUser**
- Type the shortcut **Cu**

Description

CurrentUser represents the user that is currently logged in.

Syntax

[Function] CurrentUser ["Property"] ["Value"]

Examples

- To list all properties and their values of the current user, type:

```
User name[Fixture]> List CurrentUser
```

- To set the right of the current user to admin, type:

```
User name[Fixture]> Set CurrentUser "Rights" "Admin"
```

8.2.61. CurrentUserProfile Keyword

To enter the CurrentUserProfile keyword in the command line, use one of the options:

- Type **CurrentUserProfile**
- Type the shortcut **CUP**
- Or type the shortcut **CurrentUserP**

Description

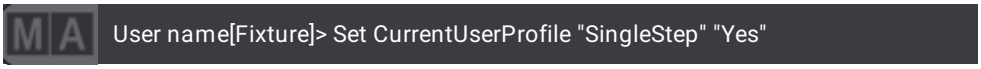
The CurrentUserProfile keyword represents the profile of the current user.

Syntax

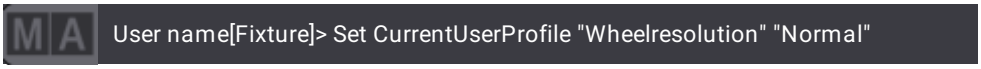
[Function] CurrentUserProfile [Property][Value]

Example

- To set single step in the current user profile, type:



- To set the wheel resolution to normal in the current user profile, type:



8.2.62. Cut Keyword

To enter the Cut keyword in the command line, type **Cut**.

Description

The Cut keyword is a function keyword used to specify the source objects for a two-step action.

Syntax

Cut [Object List]

The given object list is temporarily stored for later use as source object for the following Paste command.

For more information on Cut & Paste see the **Paste Keyword**.

Example

- To prepare the color preset 1 to be moved using the Paste keyword, type:

8.2.63. DataPool keyword

To enter the DataPool keyword in the command line, use one of the options:

- Press MA + Preset + Preset
- Type **DataPool**
- Type the shortcut **Dat**

Description

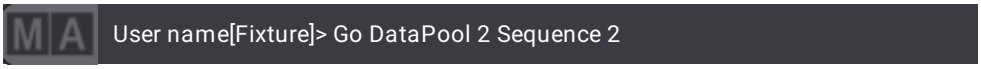
The DataPool keyword is used to address objects outside the pool you are currently in.

Syntax

[Function] DataPool [Number] Sequence [Number]

Example

- To address sequence 2 in the data pool 2, type:



8.2.64. Deceleration Keyword

To enter the Deceleration keyword in the command line, use one of the options:

- Type **Deceleration**
- Type the shortcut **Dec**

Description

The Deceleration keyword is used to set the deceleration curve of a step in the phaser.

For more information see **Phasers**.

Syntax

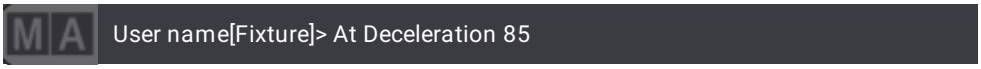
Deceleration

[At] Decelration [Value]

Examples

- To set the layer to deceleration, type:

- To set the deceleration curve of the dimmer to 85, type:



8.2.65. Decimal8 Keyword

To enter the Decimal8 keyword in the command line, use one of the options:

- Type **Decimal8**
- Type the shortcut **Deci**

Description

The Decimal8 keyword is used to set the values of a fixture selection using the 8bit decimal notation.

Syntax

At [Layer] Decimal8 [Value]

Examples

- To set the absolute layer to 255 in decimal8, type:

- To set the pan value in the absolute layer to 128 in Decimal8, type:

8.2.66. Decimal16 Keyword

To enter the Decimal16 keyword in the command line, use one of the options:

- Type **Decimal16**
- Type the shortcut **Decimal1**

Description

The Decimal16 keyword is used to set the values of a fixture selection using the 16bit hexadecimal notation.

Syntax

At [Layer] Decimal16 [Value]

Examples

- To set the dimmer for the selected fixtures on the absolute layer to 65535 in Decimal16, type:

- To set the pan attribute for the selected fixtures on the absolute layer to 32768 in Decimal16, type:

8.2.67. Decimal24 Keyword

To enter the Decimal24 keyword in the command line, use one of the options:

- Type **Decimal24**
- Type the shortcut **Decimal2**

Description

The Decimal24 keyword is used to set the values of a fixture selection using the 24bit decimal notation.

Syntax

At [Layer] Decimal24 [Value]

Examples

- To set the absolute layer to 1677721 in decimal24, type:

- To set the pan value in the absolute layer to 800000 in Decimal24, type:

8.2.68. Default Keyword

To enter the Default keyword in the command line, use one of the options:

- Press MA + .
- Type **Default**
- Type the shortcut **Def**

Description

The Default keyword is used to reset the attributes of your fixture selection to default values. If there is no attribute list, all attributes of the selected fixtures will be set to their default values.

Syntax

Default

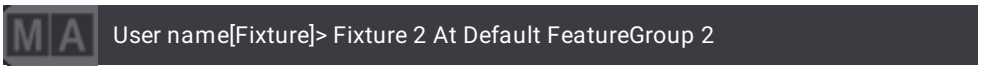
Fixture [ID] At Default

Fixture [ID] At Default [FeatureGroup]

Examples

- To set the dimmer of fixture 1 to its default values, type:

- To set the position attribute of fixture 2 to default, type:



8.2.69. Delay Keyword

To enter the Delay keyword in the command line, use one of the options:

- Press Time Time (If the Time Key Target is set to Fixture. For more information see **User Settings** and **Time Key**.)
- Press MA Time Time Time
- Type **Delay**
- Type the shortcut **Dela**

Description

Delay sets the delay time of an object.

For more information on the delay times for cues see **CueDelay Keyword**.

Syntax

Delay [Value]

Example

- To set an individual delay time of 4 seconds in the current selection, type:

8.2.70. Delete Keyword

To enter the Delete keyword in the command line, use one of the options:

- Press Delete
- Type **Delete**
- Type the shortcut **Del**

Description

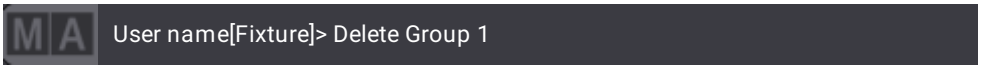
The Delete keyword is a function keyword to remove data from the show file.

Syntax

Delete [Object List]

Example

- To delete group 1 in the group pool, type:



- To delete cue 2 from the selected sequence, type:

8.2.71. DelGlobalVar Keyword

To enter the DelGlobalVar keyword in the command line, use one of the options:

- Type **DelGlobalVar**
- Type the shortcut **Delg**

Description

The DelGlobalVar keyword is used to delete global variables in a show.

Syntax

DelGlobalVar [variable name]

Example

- To delete the global variable Urban Blues 3, type:

8.2.72. DelUserVar Keyword

To enter the DelUserVar keyword in the command line, use one of the options:

- Type **DelUserVar**
- Type the shortcut **Delu**

Description

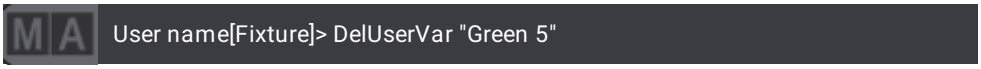
The DelUserVar keyword is used to delete user-specific variables.

Syntax

DelUserVar [variable name]

Example

- To delete the user-specific variable Green 5, type:



8.2.73. Dismiss Keyword

To enter the Dismiss keyword in the command line, use one of the options:

- Type **Dismiss**
- Type the shortcut **Di**

Description

The Dismiss keyword is used to throw stations out of your session.

Syntax

Dismiss [Device Type] [Station List]

Dismiss IP [IP]

Examples

- To dismiss the console with the name "FOH2", type:

- To dismiss the Processing Unit with the number 1, type:

- To dismiss the station with the IP address 192.168.0.10, type:

8.2.74. Display Keyword

To enter the Display keyword in the command line, use one of the options:

- Type **Display**
- Type the shortcut **Disp**

Description

Display is an object keyword used to manage the displays.

Syntax

[Function] Display

[Function] Display [Number]

Examples

- To list all open displays, type:

- To close display 2, type:

- To retrieve display 2, type:

8.2.75. DMXLayer Keyword

To enter the DMXLayer keyword in the command line, use one of the options:

- Type **DMXLayer**
- Type the shortcut **DMXL**

Description

DMXLayer is a layer keyword. It is used to change the selected layer to the DMXLayer. This layer can show information in, for instance, the fixture sheet.

The DMXLayer shows the values of the DMX sheet.

If a fixture sheet is set to **Auto** layer, then it will automatically change to show the selected layer.

Syntax

DMXLayer

Examples

- To select the DMXLayer, type:

8.2.77. DMXAddress Keyword

To enter the DMXAddress in the command line, use one of the options:

- Press MA + X8
- Type **DMXAddress**
- Type the shortcut **DMXA**

Description

The DMXAddress keyword is used to access DMX addresses directly using an absolute numbering method.

Syntax

[Function] DMXAddress [Address]

DMXAddress [Address/DMX List] At (Readout Keyword) [Value List]

Examples

- To select the fixture patched to universe 2, address 1, type:

```
User name[Fixture]> SelFix DMXAddress 513
```

- To output 50% on the third DMX channel of universe 1 using the DMX testing function, type:

```
User name[Fixture]> DMXAddress 3 At 50
```

- To output 42% on the DMX channels 8 to 15 on universe 2 using the DMX testing function, type:

```
User name[Fixture]> DMXAddress 520 Thru 527 At 42
```

- To disable the DMX testing function on all DMX channels of all universes, type:

```
User name[Fixture]> Off DMXAddress Thru
```

8.2.78. DMXReadout Keyword

To enter the DMXReadout keyword in the command line, use one of the options:

- Type **DMXReadout**
- Type the shortcut **DMXR**

Description

The DMXReadout keyword is used to set the readout of DMX values in the user profile. It affects the display of values when editing a fixture type or the normal value in the user profile.

Syntax

DMXReadout [Name]/[ID]

Example

- To set the DMXReadout to Hex8, type:

8.2.79. DMXUniverse Keyword

To enter the DMXUniverse in the command line, use one of the options:

- Press MA + X8 + X8
- Type **DMXUniverse**
- Type the shortcut **DMXU**

Description

The DMXUniverse keyword is used to access DMX universes or all DMX channels of a universe.

Syntax

[Function] DMXUniverse [Universe List](.DMX List)

[Function] DMXUniverse [Universe List](.DMX List) At (Readout Keyword) [Value List]

Examples

- To patch all DMX channels of universe 1 to universe 11, type:

```
User name[Fixture]> Move DmxUniverse 1 At DmxUniverse 11
```

- To select the fixture patched to universe 2.001, type:

```
User name[Fixture]> SelFix DMXUniverse 2.001
```

- To output 50% on the third DMX channel of universe 1 using the DMX testing function, type:

```
User name[Fixture]> DMXUniverse 1.3 At 50
```

- To output 42% on the DMX channels 8 to 15 on universe 2 using the DMX testing function, type:

```
User name[Fixture]> DMXUniverse 2.8 Thru 15 At 42
```

- To disable the DMX testing function on all DMX channels of all universes, type:

8.2.80. Down Keyword

To enter the Down keyword in the command line, use one of the options:

- Press Down
- Type **Down**
- Type the shortcut **Do**

For more information see the **Down Key**.

Description

The Down keyword is a function keyword used to address the subfixture within the main fixture.

Syntax

Down

Example

To select the subfixtures in the selected fixtures in the programmer, type:

8.2.82. Drive Keyword

To enter the Drive keyword in the command line, use one of the options:

- Type **Drive**
- Type the shortcut **Dr**

Description

The Drive keyword is used to select drives in the backup:

- **Save Show**
- **Load Show**
- **Import**
- **Export**

Syntax

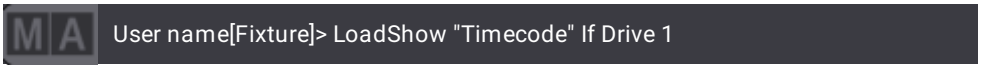
Drive [name]

Drive [ID]

Examples

- To display all available drives, type:

- To load the show "Timecode" from the internal drive, type:



8.2.83. Echo Keyword

To enter the Echo keyword in the command line, use one of these options:

- Type **Echo**
- Type the shortcut **Ec**

Description

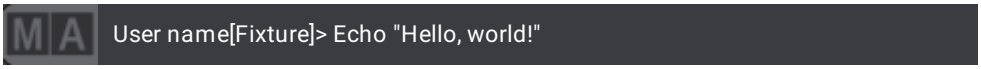
The Echo keyword is used to display feedback in the windows Command Line History and the System Monitor.

Syntax

Echo "Text"

Example

- To feedback "Hello, world!", type:



Result:

8.2.84. Edit Keyword

To enter the Edit keyword in the command line, use one of the options:

- Press Edit
- Type **Edit**
- Type the shortcut **E**

Description

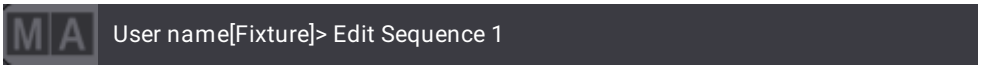
Edit is a function keyword used to modify values.

Syntax

Edit [Object]

Example

- To edit sequence 1, type:



The Edit Sequence pop-up opens.

8.2.85. EditSetting Keyword

To enter the EditSetting keyword in the command line, use one of the options:

- Press Edit Edit
- Type **EditSetting**
- Type the shortcut **EditS**

Description

EditSetting is a function keyword used to modify the object settings.

Syntax

EditSetting [Object]

Examples

- To edit the object settings of group 1, type:

The group settings pop-up opens.

- To edit the settings of object 5 in preset pool 21, type:

The Preset settings pop-up opens.

- To change the Cue Part default, enter EditSetting into the command line and then tap the desired preset. The Preset settings pop-up opens.
- Tap Cue Part, enter the new value and apply it by tapping Please.

8.2.86. Eject Keyword

To enter the Eject keyword in the command line, use one of the options:

- Type **Eject**
- Type the shortcut **Ej**

Description

The Eject keyword is used to eject a USB flash drive.

Syntax

Eject Drive [ID]

Eject Drive [Name]

Example

- To eject drive 2 (USB flash drive) in the console, type:

- To eject drive Enceladus, type:

8.2.87. Effect Keyword

To enter the Effect keyword in the command line, use one of the options:

- Type **F + 7**
- Type **Effect**
- Type the shortcut **Ef**

Description

The Effect keyword is an object keyword used to call fixtures of the ID type Effect in the programmer.

The name of this keyword can change because it is a custom ID type. For more information see **Custom ID Type**.

Syntax

Effect [ID]

Example

- To select the fixture with the Effect ID 10, type:

8.2.88. EndIf Keyword

Important:
If the If keyword is already entered twice in the command line pressing If makes the latter If an EndIf.

To enter the EndIf keyword in the command line, use one of the options:

- Type **EndIf**
- Type the shortcut **End**

Description

EndIf is a helping keyword that indicates the end of an If statement.

It enables If statements to be entered in the middle of a syntax. Upon processing, the If statement is moved to the end of the syntax, and is used as a filter or condition. This enables If syntax to be used in conjunction with pool items.

For more information see **If Keyword**.

Syntax

[Function] If [Object Condition/Filter] EndIf [Object]

Example

- To create preset 1.1 using fixtures of group 5 type:

```
User name[Fixture]> Store If Group 5 EndIf Preset 1.1
```

Result in the command line history:

```
Store Preset 1.1 If Group 5
```

8.2.89. Exchange Keyword

To enter the Exchange keyword in the command line, use one of the options:

- Press Move Move
- Type **Exchange**
- Type the shortcut **Exc**

Description

The Exchange keyword is a function keyword used to swap places.

Syntax

Exchange [Object] [Number] At [Object] [Number]

Example

- To exchange macro 1 for macro 3, type:

8.2.90. Executor Keyword

To enter the **Executor** keyword in the command line, use one of the options:

- Press MA + X16
- Type **Executor**
- Type the shortcut **Ex**

Description

The Executor keyword is an object keyword used as a control handle for other objects.

The default function for Executor objects is **Select**. This means that calling executors without any function specified selects the object assigned to the executor. This selection is now also controllable with the 100 mm fader section.

If you apply a function or reference a property not supported by the Executor object, the command will be passed on to its child: key, fader, or the object assigned to the executor.

Syntax

Executor [Executor_ID]

Select Page [Page_ID] Executor [Executor_ID]

Set Executor [Executor_ID] [Setting] = [Setting_Option]

Settings

The following table displays the settings that can be set using the command line:

Setting	Setting Options	Description
Key	Go+ etc.	Executor key assignment
Fader	Master etc.	Executor fader assignment
Encoder	Master etc.	Executor encoder assignment
EncoderLeft	<<< etc.	Executor encoder assignment when turning the encoder counterclockwise
EncoderRight	>>> etc.	Executor encoder assignment when turning the encoder clockwise
KeyCmd	Go+ etc.	Command run when executor button is pressed
EncoderRightCmd	<<< etc.	Command run when the encoder is turned counterclockwise
EncoderLeftCmd	>>> etc.	Command run when the encoder is turned clockwise

Setting	Setting Options	Description
MAKey	Go+ etc.	Executor key assignment when pressing it together with the MA key
MAFader	Master etc.	Executor fader assignment when pressing the MA key
MAEncoder	Master etc.	Executor encoder assignment when the MA key is pressed
MAEncdoerRight	<<< etc.	Executor encoder assignment when turning the encoder counterclockwise while pressing the MA key
MAEncoderLeft	>>> etc.	Executor encoder assignment when turning the encoder clockwise while pressing the MA key
MAKeyCmd	Go+ etc.	Command run when executor button is pressed together with the MA key
MAEncoderRightCmd	<<< etc.	Command run when the encoder is turned counterclockwise together with the MA key
MAEncoderLeftCmd	>>> etc.	Command run when the encoder is turned clockwise together with the MA key
PrimaryAssignmentChanged		This is information only. See the description below
SecondaryAssignmentChanged		This is information only. See the description below
Width	1-5	Executor width
Height	1-4	Executor height
Object		The object the executor controls
Config		The executor configuration used by the executor
TotalPrimaryAssignmentChanged		This is information only. See the description below
TotalSecondaryAssignmentChanged		This is information only. See the description below

The four information settings are related to changes made in relation to the used executor configuration. Primary assignments are the assignments the executor has when the MA key is not pressed. The Secondary assignments are the assignments the executor has while the MA key is pressed. The two properties beginning with "Total" are for the entire combined executor. This is relevant when the executor is part of a combined executor with more than one executor in height and/or width. The information settings cannot be changed, they are automatically updated by the software.

For setting the executor assignments using the interface, please read the **Assign Object to an Executor topic** and the **Executor Configurations topic**.

Examples

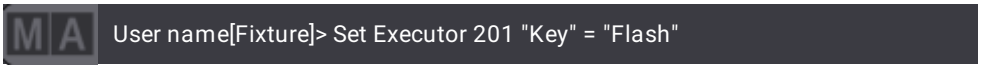
- To remove executor 205 on the current page, type:

It does not delete the object assigned to the executor. It just deletes the assignment.

- To delete cue 3 of the sequence assigned to executor 205, type:

- To select executor 102 on page 4, type:

- To set the setting "Key" of executor 201 to "Flash", type:



For more information see **Executors**.

8.2.91. Export Keyword

To enter the Export keyword in the command line, use one of the options:

- Type **Export**
- Type the shortcut **Exp**

Description

Export is a function keyword to save objects from the current show file as a smaller file without any additional show data.

If no file name is entered in the keyword, the file name will use the name of the object.

By default, files will be exported to the relevant folder within the library folder structure, either on the local drive of the console or onPC station, or on a selected USB drive. For more information about this folder structure, see the **Folder Structure** topic.

Syntax

Export [Object list] ["filename"] (If Drive [Drive number]) (/Option) ("Option value")

Options

The Export keyword supports the following options:

- **/Path:**
  Further defines the folder path where the file is saved.
  Enter a path beginning with a letter or number if the path is incorporated with the default folder structure.
  Enter a path beginning with the forward-slash (/) character if the path begins at the root of the device.
  Entering a path that does not already exist creates the necessary folders.
- **/Gaps:**
  Retains or suppresses empty spaces when exporting a range of pool objects.
  /Gaps or **/Gaps "Yes"** retains empty spaces when exporting a range of pool objects.
  /Gaps "No" suppresses empty spaces when exporting a range of pool objects.

Examples

- To export macro 1 as an .xml file labeled test, type:

- To export macro 1 labeled as the macro name is, type:

- To export macro 1, labeled "test," to the first connected USB drive, type:

- To export macro 1, labeled "test," to a folder labeled "myfavorites" at the root of the first connected USB drive, type:

- To export macros 1 thru 42, maintaining any empty spaces within the range, type:

Hint:
The Import / Export menu offers a graphical user interface for import and export functions. For more information about the Import / Export menu, see the **Import / Export Menu** topic.

8.2.92. Extract Keyword

To enter the **Extract** keyword in the command line, use one of the options:

- Press MA + At + At
- Type **Extract**
- Type the shortcut **Ext**

Description

The Extract keyword is a command keyword used to call the values of presets without references.

Syntax

Extract [Object]

Example

- To call color preset 2, type:

Hint:
Extract needs the complete preset path, e.g., for color preset 2: Extract Preset 4.2.

8.2.93. Extension Keyword

To enter the Extension keyword in the command line, use one of the options:

- Type **Extension**
- Type the shortcut **Exte**

Description

The Extension keyword is an object keyword.

This keyword belongs to a session-related group of keywords.

Syntax

Extension [ID]

Extension [Name]

Examples

To list all extensions that are currently in the same network, type:

```
User name[Fixture]> List Extension
```

To invite extension wing "FOH3" to the station where you execute the command, type:

```
User name[Fixture]> Invite Extension "FOH3"
```

8.2.94. Fade Keyword

To enter the **Fade** keyword in the command line, use one of the options:

- Press Time (If the Time Key Target is set to Fixture. For more information see **User Settings** and **Time Key**.)
- Press MA Time Time
- Type **Fade**

Description

The Fade keyword is a helping keyword to indicate fade times.

As a helping keyword for playback functions (for example Goto), this keyword sets the time used to execute the function.

As a helping keyword for programming functions (for example Store), this keyword sets the fade time of an object.

Used as a starting keyword, Fade applies individual timing in the programmer for the current selection and attributes.

Hint:
As long as the command line starts with a function, the fade keyword and value may appear anywhere in the command line.

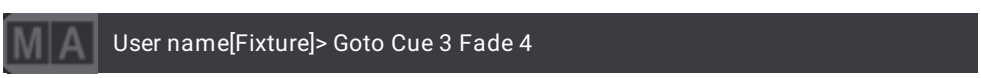
To set the fade times for cues, read more in the **CueFade keyword topic**.

Syntax

Fade [Value List]

Examples

- To crossfade to cue 3 in the selected sequence in 4 seconds, type:



- To set the individual fade time of 2 seconds to the dimmer of the current selection, type:

- To set the dimmer value of the current fixture selection to 50 % and give them an individual fade time of 2 seconds to the dimmer of the current selection, type:

8.2.95. Fader Keyword

The Fader keyword is a synonym for the **Executor keyword**.

Please follow the link above to learn more.

8.2.96. FaderHighlight Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.97. FaderLowlight Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.98. FaderMaster Keyword

To enter the FaderMaster keyword in the command line, use one of the options:

- Type **FaderMaster**
- Type the shortcut **FaderM**

Description

The FaderMaster keyword applies the Master function to an executor.

The master function controls the intensity of the assigned sequence. This may depend on the assigned object.

For more information see the **Executors**.

Syntax

Assign FaderMaster At Executor [Number]

FaderMaster At [Value] (Fade [Time])

Examples

- To assign FaderMaster to executor 204, type:

```
User name[Fixture]> Assign FaderMaster At Executor 204
```

- To move FaderMaster 205 at position 50% in a 5 seconds fade:

```
User name[Fixture]> FaderMaster 205 At 50 Fade 5
```

8.2.99. FaderRate Keyword

To enter the FaderRate keyword in the command line, use one of the options:

- Type **FaderRate**
- Type the shortcut **FaderR**

Description

The FaderRate keyword applies the Rate function to an executor.

Rate divides or multiplies the fade and delay time in a sequence by the value of the fader. If Speed from Rate is on, it is also valid for the speed stored in cues.

For more information see the **Executors**.

Syntax

Assign FaderRate At Executor [Number]

Example

- To assign FaderRate to executor 205, type:

8.2.100. FaderSolo Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.101. FaderSpeed Keyword

To enter the FaderSpeed keyword in the command line, use one of the options:

- Type **FaderSpeed**
- Type the shortcut **FaderS**

Description

The FaderSpeed keyword applies the Speed function to an executor.

It controls the speed of a phaser in a cue.

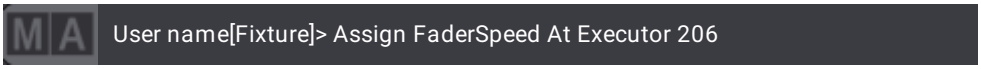
For more information see the **Executors**.

Syntax

Assign FaderSpeed At Executor [Number]

Example

- To assign FaderSpeed to executor 206, type:



8.2.102. FaderTemp Keyword

To enter the FaderTemp keyword in the command line, use one of the options:

- Type **FaderTemp**
- Type the shortcut **FaderT**

Description

The FaderTemp keyword applies the Temp function to an executor.

Temp crossfades the cue on when pulled up, and off when pulled down.

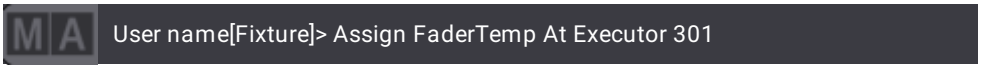
For more information see the **Executors**.

Syntax

Assign FaderTemp At Executor [Number]

Example

- To assign FaderTemp to executor 301, type:



8.2.103. FaderTime Keyword

To enter the FaderTime keyword in the command line, use one of the options:

- Type **FaderTime**
- Type the shortcut **FaderTi**

Description

The FaderTime keyword applies the Time function to a sequence. It is used to overwrite the stored cue part times by setting a time value and activating the time function.

For more information about cue time overwriting, see the **Cue Timing topic**.

Also, see the **Time topic** for how to activate and deactivate the function.

Syntax

Assign FaderTime At Executor [Exectuor_ID]

FaderTime At [Value]

FaderTime Sequence [Sequence_ID] At [Value]

Example

- To assign FaderTime function to executor 207 on the selected page, type:

```
User name[Fixture]> Assign FaderTime At Executor 207
```

- To set the FaderTime value to 50% of the time range for the selected sequence, type:

```
User name[Fixture]> FaderTime At 50
```

- To set the FaderTime value to 10% of the time range for sequence 5, type:

```
User name[Fixture]> FaderTime Sequence 5 At 10
```

8.2.104. FaderX Keyword

To enter the FaderX keyword in the command line, type **FaderX**.

Description

The FaderX keyword represents the Crossfade function of a sequence.

Crossfade gradually activates the next cue of a sequence in accordance with the position of the fader.

For more information see the **Executors**.

Syntax

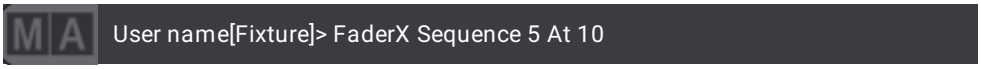
Assign FaderX At Executor [Number]

FaderX Sequence [Sequence_ID] At [Value]

Example

- To assign FaderX as executor 302, type:

- To set the FaderX value to 10% of the fader range for sequence 5, type:



8.2.105. FaderXA Keyword

To enter the FaderX keyword in the command line, type **FaderXA**.

Description

The FaderXA keyword represents the Crossfade A function of a sequence.

Crossfade A gradually fades out dimmer attributes of a current cue in a sequence in accordance with the position of the fader.

For more information see the **Executors**.

Syntax

Assign FaderXA At Executor [Number]

FaderXA Sequence [Sequence_ID] At [Value]

Example

- To assign FaderXA as fader function as executor 303, type:

```
User name[Fixture]> Assign FaderXA At Executor 303
```

- To set the FaderXA value to 10% of the fader range for sequence 5, type:

```
User name[Fixture]> FaderXA Sequence 5 At 10
```

8.2.106. FaderXB Keyword

To enter the FaderXB keyword in the command line, type **FaderXB**.

Description

The FaderXB keyword represents the Crossfade B function of a sequence.

Crossfade B gradually fades in dimmer attributes of the next cue in a sequence in accordance with the position of the fader.

For more information see the **Executors**.

Syntax

Assign FaderXB At Executor [Number]

FaderXB Sequence [Sequence_ID] At [Value]

Example

- To assign FaderXB as fader function to executor 304, type:

```
User name[Fixture]> Assign FaderXB At Executor 304
```

- To set the FaderXB value to 10% of the fader range for sequence 5, type:

```
User name[Fixture]> FaderXB Sequence 5 At 10
```

8.2.107. FeatureGroup Keyword

To enter the FeatureGroup keyword in the command line, use one of the options:

- Press MA + Preset
- Type **FeatureGroup**
- Type the shortcut **Fg**

Description

The FeatureGroup keyword is used to address feature groups.

Syntax

[Function] FeatureGroup [Name of FeatureGroup]/[ID]

Example

- To set the value Remove in all attributes of the FeatureGroup Dimmer in the current selection, type:

8.2.108. Filter Keyword

To enter the Filter keyword in the command line, use one of the options:

- Press Group Group Group
- Type **Filter**
- Type the shortcut **Fil**

Description

The Filter keyword is used to call a filter.

It represents an attribute and a layer filter.

For more information see **Worlds and Filters**.

Syntax

Filter [Name]

Example

- To call filter 4, type:

8.2.109. Fix Keyword

To enter the Fix keyword in the command line, use one of the options:

- Press MA + Pause
- Type **Fix**

Description

Fix is a function keyword to fix executors on a page.

Fix is a toggle function. This means that using Fix without any helping keyword toggles the fixing of the executors on and off.

Syntax

Fix [Executor List]

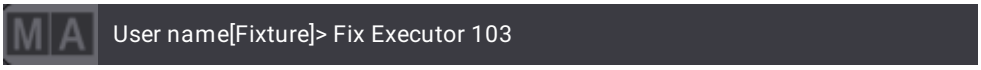
Fix On [Executor List]

Fix Off [Executor List]

Examples

- To fix executor 101 through 105 on the current page, type:

- To toggle executor 103 from changing page, or not to changing page, type:



8.2.110. FixtureClass Keyword

To enter the FixtureClass keyword in the command line, use one of the options:

- Press MA + Fixture + Fixture + Fixture
- Type **FixtureClass**
- Type the shortcuts **FC**

Description

FixtureClass is an object keyword which addresses the fixture classes of a show file.

Syntax

[Function] FixtureClass [ID List/"Name"]

Example

Requirement: Create the class "Spots" in the patch and link fixtures to it.

- To create a group in the group pool object 301 that contains all patched fixtures that are set to class "Spots" within the patch, type:

```
User name[Fixture]> AutoCreate FixtureClass "Spots" At Group 301
```

- To select all fixtures that are set to the class "Spots" within the patch, type:

```
User name[Fixture]> SelFix FixtureClass "Spots"
```

8.2.111. FixtureLayer Keyword

To enter the FixtureLayer keyword in the command line, use one of the options:

- Press MA + Fixture + Fixture
- Type **FixtureLayer**
- Type the shortcuts **FL**

Description

FixtureLayer is an object keyword which addresses the layers of fixtures in a show file.

Syntax

[Function] FixtureLayer [ID List/"Name"]

Example

Requirement: Create the layer "Backtruss" in the patch and link fixtures to it.

- To create single fixture groups starting in group pool object 201 using all patched fixtures that are set to the layer "Backtruss" within the patch, type:

```
User name[Fixture]> AutoCreate FixtureLayer "Backtruss" At Group 201 /Single
```

- To select all fixtures that are set to layer "Backtruss" within the patch, type:

```
User name[Fixture]> SelFix FixtureLayer "Backtruss"
```

8.2.112. Fixture Keyword

To enter the Fixture keyword in the command line, use one of the options:

- Press Fixture
- Type **Fixture**
- Type the shortcut **F**

Description

The Fixture keyword is used as an object keyword to access fixtures that have a fixture ID.

Syntax

Fixture [ID]

Fixture [ID].[Sub ID].[Sub ID]

Examples

- To select fixture 2, type:

- To select the fifth subfixture of the fixture group 10, type:

8.2.113. FixtureType Keyword

To enter the FixtureType keyword in the command line, use one of the options:

- Press MA + Fixture
- Type **FixtureType**
- Type the shortcuts **FT** or **FixtureT**

Description

FixtureType is an object keyword which addresses the fixture types of a show file.

Important:
Most edits and command line actions with the keyword FixtureType has to be done while in the Edit Setup mode. For more information, see **ChangeDestination keyword**.

Syntax

[Function] FixtureType [ID List/"Name"]

Example

Requirement:

- Enter the Patch menu first.
  For more information see **Patch and Fixture Setup**.

- To assign fixture type 2 to fixtures 1 through 4, type:

```
User name@ShowData/Patch/Stages/Stage 1> Assign FixtureType 2 At 1 Thru 4
```

- To select all patched fixtures of fixture type 3, type:

```
User name[Fixture]> SelFix FixtureType 3
```

- To create single fixture groups starting in group pool object 401 using all fixtures of the fixture type 9, type:

```
User name[Fixture]> AutoCreate FixtureType 9 At Group 401 /Single
```

8.2.114. Flip Keyword

To enter the Flip keyword in the command line, use one of the options:

- Type **Flip**
- Type the shortcut **Fli**

Description

The Flip keyword is used to access the different pan/tilt combinations that direct a moving head in the same direction.

Flip adds 180 degrees to the pan value of the fixtures and inverts the tilt angle. If the fixtures reach their physical breakpoint, the pan and tilt values will be set to the smallest possible value.
That is, Flip directs the fixture in the same direction using a different pan/tilt combinations.

Hint:
-If no selection list is entered, Flip is applied to the fixture selection.
-If no number is entered, the function toggles through the different possible combinations.
-The number of combinations depends on the possible degree value the fixture can pan in.

Syntax

Flip [Flip Number] (Selection List)

Examples

- To set the pan and tilt of the fixture selection to the next pan/tilt combination, type:

- To set the pan and tilt of group 7 to the second pan/tilt combination that directs the fixtures in the same direction, type:

8.2.115. Flash Keyword

To enter the Flash keyword in the command line, use one of the options:

- Press MA + >>>
- Type **Flash**
- Type the shortcut **Fla**

Description

The Flash keyword is a function keyword used to temporarily overwrite master level in order to set it to full on executing objects without using times.

If the executor is disabled and Flash is applied, the executor is temporarily activated using zero timing.

Syntax

Flash [Executor List]

Flash On [Executor List]

Flash Off [Executor List]

Examples

- To overwrite the master level of executor 201 and to start in first step, type:

- To recall the overwriting, and set the executor to master fader and disable it, type:

8.2.116. Fog Keyword

To enter the **Fog** keyword in the command line, use one of the options:

- Type **F** + **6**
- Type **Fog**
- Type the shortcut **Fo**

Description

The Fog keyword is an object keyword used to call fixtures of the ID type Fog in the programmer.

The name of this keyword can change because it is a custom ID type. For more information see the **Custom ID Type**.

Syntax

Fog [ID]

Example

- To select the fixture with Fog ID 7, type:

8.2.118. Font Keyword

To enter the Font keyword in the command line, use one of the options:

- Type **Font**
- Type the shortcut **Fo**

Description

Font is an object keyword used, for example, to adjust the font of the console.

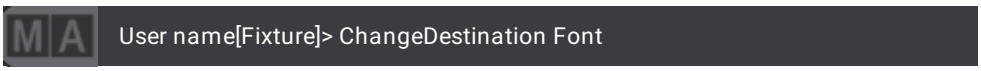
Syntax

[Function] Font

Examples

- To list all the fonts that are used in the console, type:

- To enter the font folder in the data structure, type:



8.2.119. Freeze Keyword

Important:
If Freeze is on, programmer values have a higher priority as playbacks.
Only executor priority **Super** has a higher priority than Freeze.

To enter the Freeze keyword in the command line, use one of the options:

- Press Freeze
- Type **Freeze**
- Type the shortcut **Fr**

Description

Freeze is a function keyword used to change the priority of the programmer.

Freeze is a toggle function. This means that entering Freeze without any helping keyword toggles the Freeze mode on/off.

Syntax

Freeze

Freeze On

Freeze Off

Example

To turn on the freeze mode, type:

```
User name[Fixture]> Freeze On
```

8.2.120. Full Keyword

Important:
Pressing Full directly executes the keyword.

To enter the **Full** keyword in the command line, use one of the options:

- Type **Full**
- Type the shortcut **Fu**

Description

Full is a function keyword used to set the dimmer values to 100 %.

The default function is At. This means that entering Full sets the dimmer attributes of the current selection to 100 %.

Syntax

Full

Example

- To select fixture 1 thru 10 and set the dimmer to 100 %, type:

- To select channel 53 and set the dimmer to 100 %, type:

8.2.121. Gel Keyword

To enter the **Gel** keyword in the command line, use one of the options:

- Press Preset Preset Preset
- Type **Gel**

Description

The keyword Gel provides the opportunity to edit or apply swatch book colors via command line.

Syntax

[Function] Gel "Swatch name"."Gel Name"

[Function] Gel [Swatch Id].[Gel Id]

[Function] Gel [Swatch Id]

[Function] Gel "Swatch name"

Examples

- To set the color of the selected fixtures to Lee's color Mauve, type:

```
User name[Fixture]> At Gel "Lee"."Mauve"
```

To set the color of the selected fixtures to the 44th color of the eighth swatch book, type:

```
User name[Fixture]> At Gel 8.44
```

- To export the "Lee" swatch book, type:

```
User name[Fixture]> Export Gel 8
```

8.2.123. GetGlobalVar Keyword

To enter the GetGlobalVar keyword in the command line, use one of the options:

- Type **GetGlobalVar**
- Type the shortcut **Getg**

Description

The GetGlobalVar keyword is used to display global variables along with their values in the Command Line History.

Syntax

GetGlobalVar [variable name]

Examples

- To display all variables in the Command Line History, type:

- To display all variables beginning with an f, type:

8.2.124. GetUserVar keyword

To enter the GetUserVar keyword in the command line, use one of the options:

- Type **GetUserVar**
- Type the shortcut **Getu**

Description

The GetUserVar keyword is used to display user-specific variables along with their values in the Command Line History.

Syntax

GetUserVar [variable name]

Examples

- To display all user-specific variables in the Command Line History, type:

- To display all user-specific variables beginning with an f, type:

8.2.125. Universal Keyword

To enter the **Universal** keyword in the command line, use one of the options:

- Press Channel twice
- Type **Universal**
- Type the shortcut **Uni**
- Press F 2 on a keyboard

Description

The Universal keyword is an object keyword used to call fixtures of the ID type Universal in the programmer.

Syntax

Universal [ID]

Example

- To select the fixture with Universal ID 1, type:

8.2.126. Go+ Keyword

To enter the Go+ keyword in the command line, use one of the options:

- Press Go+ | Temp
- Type **Go+**
- Type the shortcut **Go**

Description

The Go+ keyword operates as a start signal for commands to be executed.

If the target object has cues, it will proceed to the next cue.

Syntax

Go [Object] [Number]

Examples

- To go to the next cue of executor 101, type:

- To start macro 2, type:

8.2.127. GoStep Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.128. Go- Keyword

To enter the Go- keyword in the command line, use one of the options:

- Press Go- | Top
- Type **Go-**

Description

The Go- keyword is used to activate the previous cue.

If the target object has cues, it will go back to the previous cue.

Syntax

Go- [Object] [Number]

Example

- To go to the previous cue of executor 101, type:

8.2.129. Goto Keyword

To enter the **Goto** keyword in the command line, use one of the options:

- Press Goto
- Type **Goto**
- Type the shortcut **Got**

Description

Goto is a function keyword that is used to jump in a list.

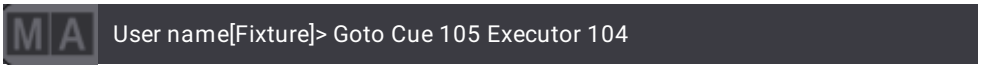
Syntax

Goto [Cue Object] [Executor Object]

Examples

- To go to cue 103 of the selected executor, type:

- To go to cue 105 of executor 104, type:



8.2.130. Grid Keyword

To enter the Grid keyword in the command line, use one of these options:

- Press MA + X3
- Type **Grid**
- Type the shortcut **Gri**

Description

The Grid keyword is used to set the grid cursor or span a grid area in the selection grid window. After the cursor is set, it is possible to arrange fixtures in the selection grid window. To position fixtures with a z-axis in the selection window, use the Grid keyword.

To set the grid cursor using the user interface see **Selection Grid Window**.

The grid command supports a space or a / as delimiters for command line input.

Syntax

Grid [x-axis] / [y-axis] / [z-axis]

Grid [x-axis] [y-axis] [z-axis]

Examples

- To move the blue grid cell cursor in the selection window to column 2, row 2, type:

- To position fixtures in a defined area in the selection window, type:

8.2.131. Group Keyword

To enter the Group keyword in the command line, use one of the options:

- Press Group
- Type **Group**
- Type the shortcut **G**

Description

The Group keyword is an object type that contains a certain selection of fixtures in a specific order along with x-, y-, z-coordinates for the **Selection window**.

The default function of the Group keyword is SelFix. This means that calling groups without any function specified selects the fixtures within the group.

For more information see the **SelFix keyword**.

Syntax

Group [ID]

[Function] Group [ID]

Example

- To select the fixtures stored in group 3, type:

- To list in the command line history all stored groups of the group pool, type:

8.2.132. HardwareKey Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.133. Help Keyword

To enter the Help keyword in the command line, use one of the options:

- Press Help
- Type **Help**
- Type the shortcut **H**

Hint:
Use the context-sensitive help:

-Press and hold Help or tap ? in the control bar and tap a UI element to open the corresponding help topic.

Description

Help is a function used to list all command keywords, list of operators, layer keywords, cue timing keywords, readout keywords, the playback commands, etc. The keywords and what they do is displayed in white, the corresponding shortcuts are displayed in green.

Syntax

Help

Help [Filter]

Examples

- To list all keywords in the Command Line History, type:

- To list all keywords starting with an f, type:

8.2.134. Helplua keyword

To enter the HelpLua keyword in the command line, use one of the options:

- Type **HelpLua**
- Type the shortcut **HelpL**

Description

HelpLua is a function keyword to export a list of all grandMA Lua functions.

To learn more about Lua functions, read the **Plugins** topic.

Syntax

HelpLua

Examples

- To list all grandMA3 specific Lua functions, type:

The Lua functions are exported into the file "grandMA3_lua_functions.txt" in the **grandMA3 library folder**.

8.2.135. Hex8 Keyword

To enter the Hex8 keyword in the command line, use one of the options:

- Type **Hex8**
- Type the shortcut **Hex**

Description

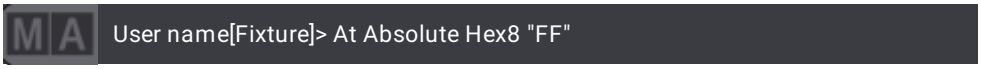
The Hex8 keyword is used to set the values of a fixture selection using the 8bit hexadecimal notation.

Syntax

At [Layer] Hex8 [Value]

Examples

- To set the absolute layer to FF in Hex8, type:



- To set the pan value in the absolute layer to DE in Hex8, type:

8.2.136. Hex16 Keyword

To enter the Hex16 keyword in the command line, use one of the options:

- Type **Hex16**
- Type the shortcut **Hex1**

Description

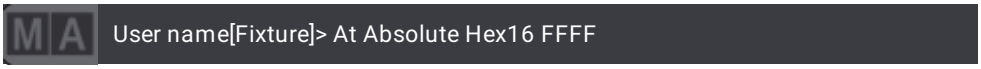
The Hex16 keyword is used to set the values of a fixture selection using the 16bit hexadecimal notation.

Syntax

At [Layer] Hex16 [Value]

Examples

- To set the dimmer for the selected fixtures on the absolute layer to FFFF in Hex16, type:



- To set the pan attribute for the selected fixtures on the absolute layer to 8000 in Hex16, type:

8.2.137. Hex24 Keyword

To enter the Hex24 keyword in the command line, use one of the options:

- Type **Hex24**
- Type the shortcut **Hex2**

Description

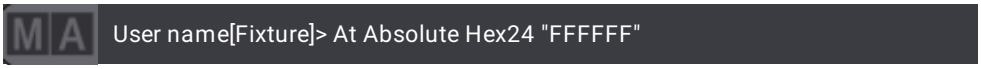
The Hex24 keyword is used to set the values of a fixture selection using the 24bit hexadecimal notation.

Syntax

At [Layer] Hex24 [Value]

Examples

- To set the absolute layer to FFFFFF in Hex24, type:



- To set the pan value in the absolute layer to 800000 in Hex24, type:

8.2.138. Highlight Keyword

To enter the Highlight keyword in the command line, use one of the options:

- Press HighIt
- Type **Highlight**
- Type the shortcut **Hi**

Description

The Highlight keyword is a function used to apply the defined highlight values on fixtures that are selected.

It is a toggle function. This means entering Highlight without a helping keyword enables or disables the Highlight mode.

Hint:
To manually fade from the current value to the highlight value and back, use the Highlight fader.

For information on location see the **HighIt Key**.

Syntax

Highlight

Highlight On

Highlight Off

Example

- To toggle the highlight mode, type:

8.2.139. Houselights Keyword

To enter the **Houselights** keyword in the command line, use one of the options:

- Type **F + 3**
- Type **Houselights**
- Type the shortcut **Ho**

Description

The Houselights keyword is an object keyword used to call fixtures of the ID type Houselights in the programmer.

The name of this keyword can change because it is a custom ID type. For more information see the **Custom ID Type**.

Syntax

Houselights [ID]

Example

- To select the fixture with Houselights ID 5, type:

8.2.140. Hz Keyword

To enter the Hz (Hertz) keyword in the command line, type: **Hz**

Description

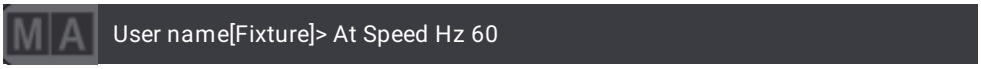
The Hz (Hertz) keyword is used to set the speed of a fixture selection using the unit hertz.

Syntax

[At] [Speed] Hz [Value]

Example

- To set the speed layer to 60 hertz, type:



8.2.141. If Keyword

Important:
To end an if statement in the middle of a longer syntax, use the EndIf keyword.
For more information see **EndIf Keyword**.

To enter the If keyword in the command line, use one of the options:

- Press If
- Type If

Description

The **If** keyword is a function keyword used to deselect fixtures in the entered selection list.

As a helping keyword for other operational functions, If sets a filter for the operation.

As a helping keyword for the Clone function, If sets the scope of the cloning.

For more information see **Clone Keyword**.

Syntax

If [Selection List]

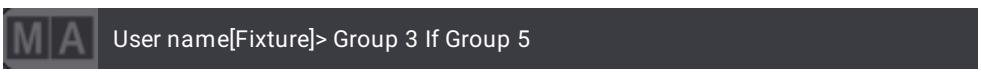
Clone [Selection List] At [Selection List] If [Object List]

[General Syntax] If [Object List] [Attribute List]

Examples

- To deselect fixtures not in group 5, type:

- To select fixtures which are a part of group 3 and group 5, type:



- To delete channel 4 in cue 3, type:

- To delete attibute "Pan" of fixture 4 in cue 3, type:

- To clone fixture 1 to fixture 2 only in sequence 1, type:

8.2.142. IfActive Keyword

To enter the IfActive keyword in the command line, use one of the options:

- Press If If
- Type **IfActive**
- Type the shortcut **Ifa**

Description

IfActive is a function keyword that selects fixtures with active values in the programmer.

Important:
IfActive only works with dimmer values.

If no filter is entered, IfActive will select all fixtures with active values.

If a filter is entered, IfActive will select all fixtures which have this filter and also fixtures that have active values in the programmer.

Syntax

IfActive

IfActive [Filter]

Examples

- To select all fixtures with active values in the programmer, type:

```
User name[Fixture]> IfActive
```

- To select fixtures within group 5 which also have active values in the programmer, type:

```
User name[Fixture]> IfActive Group 5
```

8.2.143. IfOutput Keyword

To enter the IfOutput keyword in the command line, use one of the options:

- Press If
- Type **IfOutput**
- Type the shortcut **Ifo**

Description

IfOutput is a function keyword that selects fixtures based on their current output. This function works with presets and sequences only.

Syntax

IfOutput

IfOutput (object list)

Examples

- To select all fixtures that output the values stored in sequence 1, type:

```
User name[Fixture]> IfOutput Sequence 1
```

- To select all fixtures using the color preset "Green", type:

```
User name[Fixture]> IfOutput Preset "Color"."Green"
```

8.2.144. IfProg keyword

To enter the IfProg keyword in the command line, use one of the options:

- Press If If If
- Type **IfProg**
- Type the shortcut **Ifp**

Description

IfProg is a function keyword that selects fixtures that contain values in the programmer.

Important:
IfProg only works with dimmer values.

If no filter is entered, IfProg selects all fixtures containing values in the programmer.

If a filter is entered, IfProg selects the fixtures which are in the filter and also which contain values in the programmer.

Syntax

IfProg

IfProg (object list)

Examples

- To select all fixtures containing values in the programmer, type:

```
User name[Fixture]> IfProg
```

- To only select the fixtures of group 5, which contain values in the programmer, type:

```
User name[Fixture]> IfProg Group 5
```

8.2.145. Image Keyword

To enter the Image keyword in the command line, use one of the options:

- Type **Image**
- Type the shortcut **Ima**

Description

The keyword Image is an object keyword. It is used to manage images using the command line.

Syntax

Image [Image Pool].[ID/Name]

[Function] Image [Image Pool].[ID/Name]

Examples

- To edit image 1 in the custom image pool, type:

- To delete image 2 in the custom image pool, type:

8.2.146. Import Keyword

To enter the Import keyword in the command line, use one of the options:

- Type **Import**
- Type the shortcut **I**

Description

Import is a function keyword to incorporate small portions of exported show file data, available as .xml files, into the current show file.

The Import command loads data into the specified destination or the current command line destination after any previously occupied objects.

By default, files will be imported from the relevant folder within the library folder structure, either on the local drive of the console or onPC station, or on a selected USB drive. For more information about this folder structure, see the **Folder Structure** topic.

Syntax

Import [Object Type] Library "File Name.xml" (If Drive [Drive Number]) (At [Object Type] [Object Number]) (/Option) ("Option Value")

Options

- **/Path:**
  Further defines the folder path where the source .xml file is saved.
  Enter a path beginning with a letter or number if the path is incorporated with the default folder structure.
  Enter a path beginning with the forward-slash (/) character if the path begins at the root of the device.
- **/Type:**
  When importing, the desired file may be either a "User" file, which has been previously exported; or a "System" file, which is predefined and included with the system software. If this option is not defined within the import syntax, the console will first search the user library for the specified file name. If the file does not exist within the user library, the console will then search within the system files.
  /Type "User" restricts the console to only search within the user library of the selected drive.
  /Type "System" restricts the console to only search within the system files, ignoring the user library.
- **/NoRefresh:**
  When listing the library files of a certain type and/or path, it may take a while to type the path and the options into the command line. When a file shall be imported after a type and/or path has been specified during a previous command, the type and/or path will typically need to be entered again as part of the import command. Using the **/NoRefresh** option, it is not necessary to re-enter /Type and/or /Path as part of the import command.

- **/Gaps:**
  Retains or suppresses empty spaces when importing a range of pool objects, which were previously exported using the **/Gaps** option.
  /Gaps "Yes" or omitting the **/Gaps** option retains empty spaces included in the file when importing a range of pool objects.
  /Gaps "No" suppresses empty spaces when importing a range of pool objects.

Examples

Requirements: Destination is changed to fixture types.

- To import, for example, a MAC Aura XB as a new fixture type in the show file, type:

```
User name@ShowData/Patch/FixtureTypes> Import Library "MAC Aura XB"
```

Requirements: Destination is changed to macros.

- To import the macro color.xml to the pool object following the macro occupied last, type:

```
User name@ShowData/DataPools/Default/Macros> Import Library "color.xml"
```

- To import the same macro to macro 42 in the macro pool without changing the command line destination, type:

```
User name[Fixture]> Import Macro Library "color.xml" At Macro 42
```

Requirements: Destination is changed to macros.

- To import all macros from the library into the show file, type:

```
User name@ShowData/DataPools/Default/Macros> Import Library "*.xml"
```

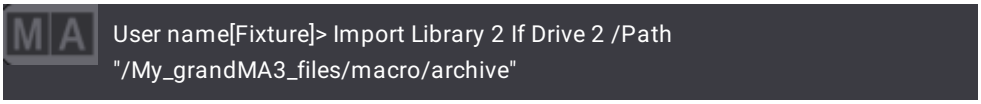
- To use the /NoRefresh option to avoid reentering drive and path specifications:

List all macro libraries within a specific path on a specific drive:

```
User name[Fixture]> List Library If Drive 2 /Path
"/My_grandMA3_files/macro/archive"
```

To import the second library from the list without reentering the drive and path, type:

Without the /NoRefreshOption, the command must also include the same drive and path specifications:



8.2.147. Index Keyword

To enter the Index keyword in the command line, use one of the options:

- Type **Index**
- Type the shortcut **Ind**

Description

Index keyword is a command keyword that addresses values where a property can have more than one value. For example, a fixture with at least two DMX breaks. The patch property has 2 values for this fixture.

The Index Keyword is zero nominated.

Syntax

Set [Object List] ["Property"] Index [Index number] ["Value"]

Example

Requierment:

- Fixture 1 has to have at least two DMX breaks.
  For example, the fixture type LED Wall 20x20

- To set the patch address of the second break to universe 2 address 1 without affecting the first address, type:

8.2.148. Insert Keyword

To enter the **Insert** keyword in the command line, use one of the options:

- Press Copy Copy Copy
- Type **Insert**
- Type the shortcut **Ins**

Description

The Insert keyword is a function keyword to insert pool objects between two other already taken pool objects. The following pool objects will be moved to the next empty destination.

If you entered Insert pressing Copy Copy Copy, you could insert pool objects by tapping them.

Syntax

Insert [Object List] At [Destination Start ID]

Example

- To insert Group 5 between Group 8 and 9, type:

8.2.149. Integrate Keyword

To enter the Integrate keyword in the command line, use one of the options:

- Press MA + At
- Type **Integrate**
- Type the shortcut **Int**

Description

The Integrate keyword integrates step 1 of a preset into other steps of the programmer.

These steps can be stored into presets and cues.

Syntax

Integrate [Object] (Step [Step ID])

Integrate [Object]

Examples

Requirement: Example 1

- There are already active values in step 1 in the programmer

- To integrate step 1 of preset 1.1 into step 2 of the programmer, type:

Requirement: Example 2

- Select fixtures and create preset 4.5
- Follow steps 1 to 4

1. To call preset 4.5 into the programmer, type:

2. To create the next step, type:

3. To integrate step 1 of preset 4.2, type:

4. To store the changes in the preset, type:

8.2.150. Interface Keyword

To enter the Interface keyword in the command line, use one of the options:

- Type **Interface**
- Type the shortcut **Inter**

Description

Interface is an object keyword used to set the network interface using the command line.

For more information see **Interfaces and IP**.

Syntax

[Function] Interface

Interface [Number]

Examples

- To list the network interfaces, enter the Interface folder in the data structure first:

```
User name[Fixture]> ChangeDestination Interface
```

- To list the interfaces, type:

```
User name@Interfaces> List
```

8.2.152. Invert Keyword

To enter the **Invert** keyword in the command line, use one of the options:

- Type **Invert**
- Type the shortcut **Inve**

Description

Invert is a function to invert the selection status of fixtures.

Invert acts in the fixtures that were selected. If a fixture is selected, using this keyword deselects the fixtures. If a fixture was not selected, using the keyword Invert selects the fixtures.

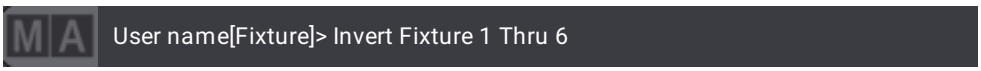
Syntax

Invert [Selection List]

Example

Requirement: Fixtures 1, 3 and 5 are selected

- To invert the select state withtin fixtures 1 to 6, type:



8.2.153. Invite keyword

To enter the Invite keyword in the command line, use one of the options:

- Type **Invite**
- Type the shortcut **Inv**

Description

Invite is a function keyword used to invite other stations into the session.

This keyword belongs to a session-related group of keywords. For more information see **Console keyword**, **Node keyword**, **OnPC keyword**, **PU keyword**, **Station keyword**, **Extension keyword**, **Join keyword**, **Leave keyword**, **Dismiss keyword**, **Reboot keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

Invite [Device Type] [Station List]

Hint:
The device types are: **Console**, **Node**, **onPC**, **PU**, **Station**, **Extension**.

Examples

- To invite console 6 to the session, type:

```
User name[Fixture]> Invite Console 6
```

- To invite the node named "Truss" to the session, type:

```
User name[Fixture]> Invite Node "Truss"
```

- To invite the station with the IP address 192.168.0.10, type:

```
User name[Fixture]> Invite IP 192.168.0.10
```

8.2.154. IP Keyword

To enter the IP keyword in the command line, type **IP**.

Description

IP is an object keyword used to manage devices with IP addresses.

It interacts with other function keywords and the IP address of the corresponding device.

Syntax

[Function] IP [IP Address]

Examples

For information on IP examples see:

- **Dismiss keyword**
- **Reboot keyword**
- **RemoteCommand keyword**
- **Shutdown keyword**

8.2.155. JoinSession Keyword

To enter the JoinSession keyword in the command line, use one of the options:

- Type **JoinSession**
- Type the shortcut **J**

Description

JoinSession is a function keyword used to join a session.

This keyword belongs to a session-related group of keywords. For more information see **Console keyword**, **Node keyword**, **OnPC keyword**, **PU keyword**, **Station keyword**, **Extension keyword**, **Dismiss keyword**, **Invite keyword**, **LeaveSession keyword**, **Reboot keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

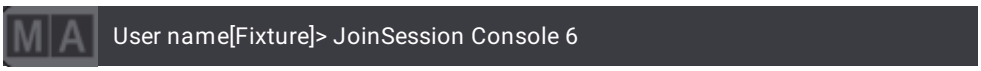
JoinSession IP [Device_IP]

JoinSession [Device_Type] [Device_ID]

Hint:
The device types are: **Console**, **Node**, **onPC**, **PU**, **Station**, **Extension**.

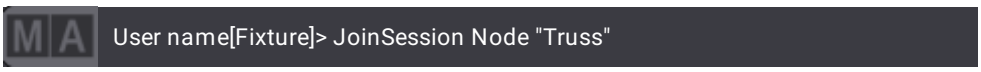
Examples

- To join the session of console 6, type:



- To join the session on station with IP 192.168.10.21:

- To join the session the node named "Truss" is part of, type:



8.2.156. Key Keyword

To enter the Key keyword in the command line, use one of the options:

- Type **Key**
- Type the shortcut **K**

Description

The Key keyword is used to address the network keys.

For more information on what network keys are see **Create a Custom Key**.

Syntax

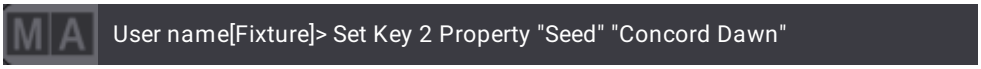
[Function] Key [Number/"Name"] (Property "Property Name" "Value")

Examples

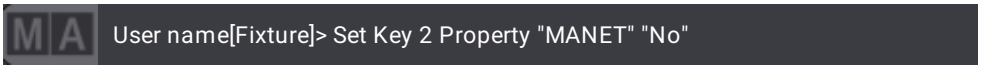
- To store a new key, type:

- To list all keys, type:

- To set a different password for the newly created key, type:



- To not use key 2 for MAnet, type:



8.2.157. Keyboard Keyword

To enter the Keyboard keyword in the command line, use one of the options:

- Type **Keyboard**
- Type the shortcut **Keyb**

Description

Keyboard is a an object keyword used to change the langauge of the on-screen keyboard.

Syntax

[Function] Keyboard

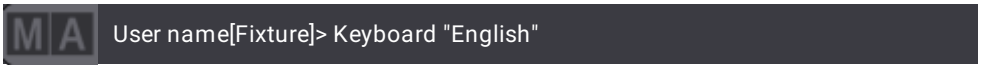
Keyboard [ID]/[Name]

Examples

- To list the keyboard languages, type:

- To set the keyboard to German, type:

- To set the keyboard to English, type:



8.2.158. KeyboardShortcuts Keyword

To enter the KeyboardShortcuts keyword in the command line, use one of the options:

- Type **KeyboardShortcuts**
- Type the shortcut **KeyboardS**

Description

The KeyboardShortcuts keyword is an object keyword that is used to work with keyboard shortcuts using the command line.

Syntax

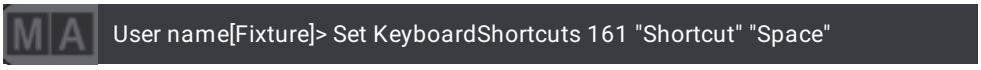
[Function] KeyboardShortcuts [Number]

[Set] KeyboardShortcuts [Number] "Property" "Value"

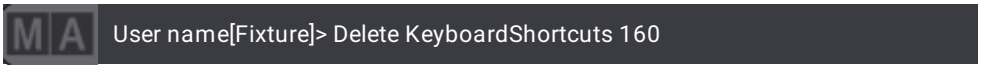
Examples

- To list all keyboard shortcuts, type:

- To assign, for example, the property "Shortcut" to the value "Space", type:



- To delete the keyboard shortcut number 160, type:



8.2.159. Knockin Keyword

To enter the Knockin keyword in the command line, use one of the options:

- Type **Knockin**
- Type the shortcut **Kn**

Description

Knockin is used to take values that are located for example in sequences or presets actively into the programmer.

This is useful to the effect that you can actively take values that have already been stored into the programmer.

Syntax

Fixture [Fixture List] Knockin ["Attribute Name"/Attribute ID]/["Feature Group Name"/Feature Group ID]

Knockin Fixture [Fixture List]

Attribute ["Attribute Name"/ Attribute ID] At Knockin

Examples

- To knock in the attribute Color RGB_R in fixture 1, type:

```
User name[Fixture]> Fixture 1 Knockin Attribute "ColorRGB_R"
```

- To knock in all attributes of fixture 2, type:

```
User name[Fixture]> Knockin Fixture 2
```

- To knockin only the dimmer values of fixtures 11 through 15, type:

```
User name[Fixture]> Fixture 11 Thru 15 At Knockin
```

8.2.160. Knockout Keyword

To enter the Off keyword in the command line, use one of the options:

- Type **Knockout**
- Type the shortcut **Knocko**

Hint:
The Knockout keyword is used as a synonym for the Off Keyword.

Description

The Knockout keyword is used as a function and value keyword to:

- Knock out selections in the programmer
- Knock out active attributes in the programmer

Syntax

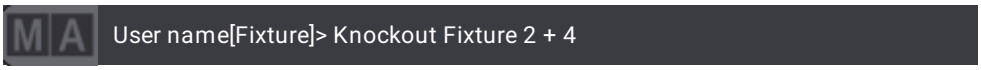
Fixture [Fixture List] Knockout ["Attribute Name"/Attribute ID]/["Feature Group Name"/Feature Group ID]

Knockout Fixture [Fixture List]

Attribute ["Attribute Name"/Attribute ID] At Knockout

Examples

- To knock out the parameters of fixture 2 and 4 in the programmer, type:



- To knockout the attribute Color RGB_R in the selected fixtures, type:

- To knockout the dimmer attribute in fixture 5, type:

8.2.161. Label Keyword

To enter the Label keyword in the command line, use one of the options:

- Press Assign Assign
- Type **Label**
- Type the shortcut **L**

Description

The Label keyword is used to give objects a name.

If multiple objects are labeled, and the name contains a free-standing number, the number will be enumerated for each object.

If you do not label an object, a pop-up appears.

Syntax

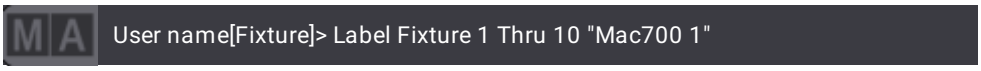
Label [Object List] "Name"

Important:
The name must not contain the following characters: \ " $ & * ? , . ; ^ { | } ~

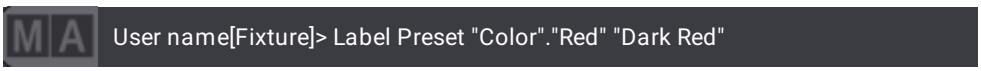
Examples

- To label group 3 "Higgs Boson", type:

- To label fixtures 1 to 10 as "Mac700 1", "Mac700 2" and so on type:



- To rename the color preset "Red" to "Dark Red" type:



8.2.162. Language Keyword

To enter the Language keyword in the command line, use one of the options:

- Type **Language**
- Type the shortcut **Lan**

Description

The Language keyword sets language for the user that is currently logged in.

Syntax

Language [Language List]

Example

- To set the langugage of the current user to German, type:

8.2.163. Layout Keyword

To enter the **Layout Keyword** in the command line, use one of the options:

- Press MA + X4
- Type **Layout**
- Type the shortcut **Lay**

Description

Layout is an object type keyword representing a layout of fixtures and other objects.

Syntax

Layout [ID]

Examples

- To create layout 5 and add fixture selection to this layout, type:

- To add group 5 as a button in layout 4, type:

8.2.164. LearnRate Keyword

To enter the LearnRate keyword in the command line, use one of the options:

- Press Learn
- Type **LearnRate**
- Type the shortcut **Lear**

Description

The LearnRate keyword is a function keyword used to set the rate of cue timings in a sequence.

Syntax

LearnRate [Object List]

Example

- To set the rate of executor 105, type:

8.2.165. LearnSpeed Keyword

To enter the LearnSpeed keyword in the command line, use one of the options:

- Type **LearnSpeed**
- Type the shortcut **LearnS**

Description

The LearnSpeed keyword is a function keyword used to set the speed in phasers.

Syntax

LearnSpeed [Object List]

Example

- To set the speed of executor 101, type:

8.2.166. LeaveSession Keyword

To enter the LeaveSession keyword in the command line, use one of the options:

- Type **LeaveSession**
- Type **Leave**
- Type the shortcut **Le**

Description

LeaveSession is a function keyword used to leave the current session.

Other keywords of this kind are **JoinSession**, **Invite**, **Dismiss**.

Syntax

LeaveSession

Example

- To leave the current session, type:

8.2.167. Library Keyword

To enter the **Library** keyword in the command line, use one of the options:

- Type **Library**
- Type the shortcut **Lib**

Description

Library is an object keyword that is used to access the corresponding library folder down to the root on the hard drive.

Syntax

[Function] Library

Examples

Requirements: Change destination to macros.

For more information see **ChangeDestination Keyword**.

- To list all macros stored in the macro library, type:

- To import the first macro library, type:

Requirements: Change destination to fixture types.

For more information see **ChangeDestination Keyword**.

- To list all fixture types stored in the fixture types library, type:

8.2.168. List Keyword

To enter the List keyword in the command line, use one of these options:

- Press List
- Type **List**
- Type the shortcut **Li**

Description

The List keyword displays show data, including objects and their associated properties, in the command line history.

It is possible to list any object or any property of:

- Cues of the selected executor
- Groups
- Presets

If the list command does not specify any type of object, the data of the current command line destination is displayed.

The List command can also display the contents of library folders on the console, onPC station, or connected USB drives.

Hint:
The /Path option instructs the software to access folders other than the defaults. For more information on the default folder structure, see the **Folder Structure** topic.

The List keyword is a function keyword.

Syntax

List [Object List]

List [Object Type] Library (If Drive [Drive Number]) (/Path [Folder Path]) (/Type [File Type])

Important:
When specifying a type, the word "System" or "User" must be capitalized. For more information on /Type see the **Import Keyword**.

Examples

- To list the first ten fixtures of the Fixture Edit Setup, type:

- To list all existing attributes in the show file, type:

- To list the first 5 groups of the group pool, type:

- To list the macros available to import from the default macro library folder on the first connected USB drive, type:

Example - List Properties of a Parent and Their Children's Properties

Important:
In case the children of an object should have various properties, they will be listed using names and numbers when executing List within the parent object.
To display the properties of a child directly, you have to specifically list the child.
However, the steps described here can also be used in other constellations consisting of a parent and their children.

In the following example we use StationSettings as the parent and DeskLightsCollect as one of their children.

1. Change the destination of the command line first:

```
User name[Fixture]> ChangeDestination StationSettings
```

Result:

```
User name@StationSettings>
```

2. To list the properties of the station settings, type:

```
User name@StationSettings> List
```

The children are now displayed in the command line history:

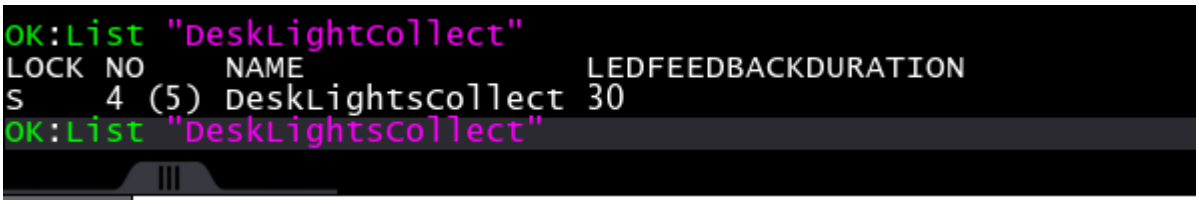
```
OK:ChangeDestination "StationSettings"
LOCK  NO     NAME
      1      TimeConfig
S     2      LocalSettings
S     3 (2)  DefaultDisplayPositionsCollect
S     4 (5)  DeskLightsCollect
      5      Displaysurfaces
      6 (5)  GridColumnRegistry
OK:List
OK:Menu "CommandLineHistory"
```

The children of the Station Settings are listed

3. To list the properties of one child, for example DeskLightsCollect, type:

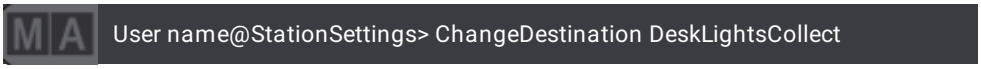
```
User name@StationSettings> List DeskLightsCollect
```

The properties of the child DeskLightsCollect are now displayed in the command line history:



The properties of DeskLightsCollect are listed

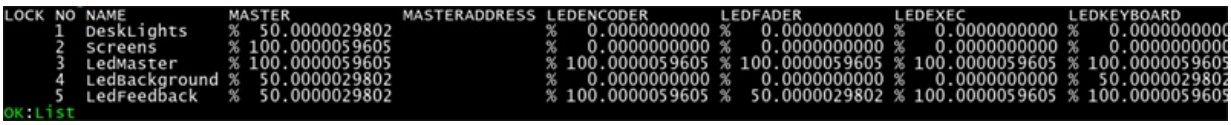
4. To access the folder DeskLightsCollect within Station Settings, type:



Result:

5. To list the properties of the children of DeskLightsCollect, type:

The children's properties are now displayed.



Properties of DeskLightsCollect listed in the command line history

8.2.169. ListOwner Keyword

To enter the ListOwner keyword in the command line, use one of the options:

- Type **ListOwner**
- Type the shortcut **Listo**

Description

ListOwner is used as a troubleshooting keyword in case of a multi-user access conflict.

If a multi-user access conflict occurs, use the ListOwner keyword. It lists the objects that are currently locked for all users in the session, hence causing the conflict.

Syntax

ListOwner

Example

- To display all objects that are locked for all users and that might cause a multi-user conflict, type:

8.2.170. ListRef Keyword

To enter the ListRef keyword in the command line, use one of the options:

- Press MA + List
- Type **ListRef**
- Type the shortcut **Listr**

Description

ListRef is used to display references and/or dependencies in your show file.

Examples of references between objects, only to mention a few:

- Layout to group/macro/fixture
- Preset to cue/preset (embedded preset)

These references and/or dependencies depend on the use of the object in the show file.

Syntax

ListRef [Target]

Example

- To list the references and dependencies of the dimmer preset 1, type:

```
MA  User name[Fixture]> ListRef Preset 1.1
```

The Info pop-up opens listing references and/or dependencies:

References and dependencies in the Info pop-up

8.2.171. Load Keyword

To enter the Load keyword in the command line, use one of the options:

- Press Goto Goto
- Type **Load**

Description

The load keyword is a function keyword used to prepare an executor to jump to another cue rather than jumping to the next cue when a **Go+** is performed on the executor.

Syntax

Load [Executor Object] [Cue Object]

Examples

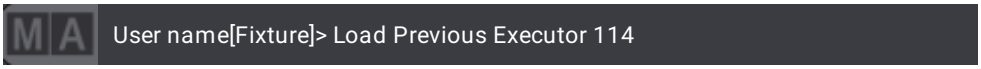
- To load cue 3 on the **selected** executor, type:

Cue 3 is loaded. To indicate that a cue is loaded, the display toggles between Cue 3 and Cue "Name".

- To load cue 5 on executor 114, type:

- To load cue +2 on executor 114, type:

- To load the previous cue on executor 114, type:



8.2.172. Loaded Keyword

Important:
MA + Go+ [large] immediately executes the **Go+ Loaded** command.

To enter the Loaded keyword in the command line, use one of the options:

- Press MA + Goto Goto
- Type **Loaded**

Description

The Loaded keyword is a function keyword used to simultaneously handle several cues in a sequence that are in the Load status.

For more information on Load see the **Load Keyword**.

Syntax

[Function] Loaded

Examples

Requirement: At least one cue has to be in the Load status.

- To start several loaded cues, type:

- To disable sequences that are loaded, type:

8.2.173. LoadShow Keyword

To enter the LoadShow keyword in the command line, use one of the options:

- Type **LoadShow**
- Type the shortcut **Loa**

Description

The LoadShow keyword is a function keyword that loads a show from the folders shows, demo shows, or backup.

To learn more about the folder structure of shows, demo shows, and backups, read the **Show File Handling topic**.

If the show file does not exist in the folder, a new empty show file with the entered name will be created.

If the console is in a session, the show file will be uploaded to all connected stations.

To add the folder location for files, type the command line option /Path. For more information on the default folder structure, see the **Folder Structure** topic.

To distinguish between normal shows, demo shows, and backup files, use the /Type option.

The normal shows are /Type "Shows", demo shows are /Type "Demo", and backup files can be specified with /Type "Backup".

Syntax

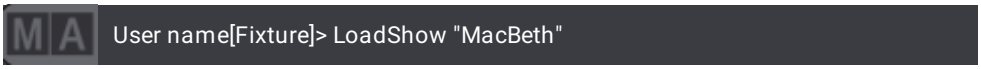
LoadShow ["object name"]

LoadShow ["object name"] (/Path)

LoadShow ["object name"] (/Type)

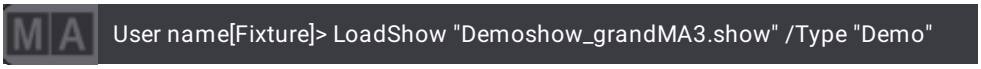
Examples

- To load the show file with the file name "MacBeth", type:



- To create a new show file with the name "William", type:

- To load the demo show from the demo shows folder, type:



Important:
To load the new show again, save the new show file after loading it. For more information see **SaveShow keyword**.

8.2.174. Lock Keyword

To enter the Lock keyword in the command line, use one of the options:

- Type **Lock**
- Type the shortcut **Lo**

Description

The Lock keyword is a function keyword that locks objects.

Objects that are locked are indicated by a padlock.

Syntax

Lock [Object]

Examples

- To lock macro 1, type:

To unlock objects see the **Unlock Keyword**.

- To lock the second color preset 2, type:

- To lock sequence 3, type:

8.2.175. Login Keyword

To enter the Login keyword in the command line, use one of the options:

- Type **Login**
- Type the shortcut **Log**

Description

The Login keyword is a function keyword to log in another user.

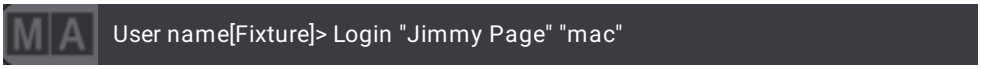
Syntax

Login

Login ["User name"] ["password"]

Example

To log out the current user and log in as user Jimmy Page with the password mac, type:



8.2.176. Logout Keyword

To enter the Logout keyword in the command line, use one of the options:

- Type **Logout**
- Type the shortcut **Logo**

Description

The Logout keyword is a function keyword used to log out the user and change to guest user.

Syntax

Logout

Example

- To log out the user and change to guest user, type:

8.2.177. Lowlight Keyword

To enter the Lowlight keyword in the command line, use one of the options:

- Press MA + HighIt
- Type **Lowlight**
- Type the shortcut **Low**

Description

The Lowlight keyword works in conjuction with Highlight.

Syntax

Lowlight

Example

1. Select a set of fixtures:

2. Apply the highlight values to these fixtures:

3. Apply Next:

Result: The first fixture is now highlighted.

4. To apply the lowlight values on the rest of the fixtures, type:

8.2.178. Lua Keyword

To enter the Lua keyword in the command line, type **Lua**.

Description

The Lua keyword is used to execute commands in the script language Lua.

After entering Lua into the command line, type in the script language Lua version 5.4. The grandMA3 will directly execute the commands written in Lua.

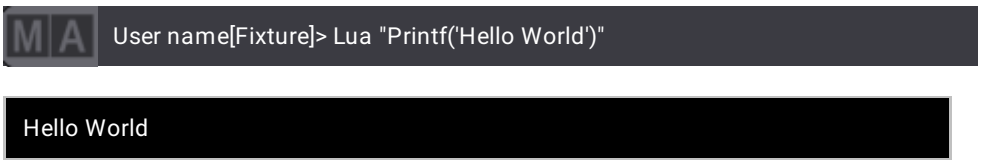
To learn more about the grandMA3 specific Lua functions, read the **Plugins** topic.

Syntax

Lua ["Lua Code"]

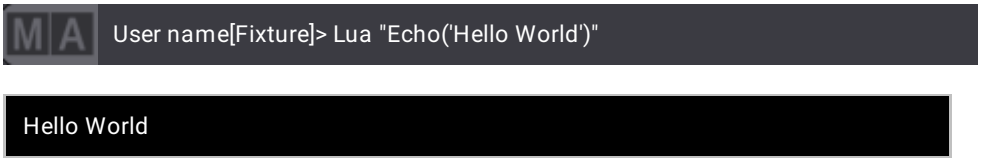
Example

- To output "Hello World" in the Command Line History, type:



will be displayed in the Command Line History.

- To output "Hello World" on the system monitor, type:



will be displayed on the system monitor.

8.2.179. LuaFile Keyword

To enter the LuaFile keyword in the command line, use one of the options:

- Type **LuaFile**
- Type the shortcut **LuaF**

Description

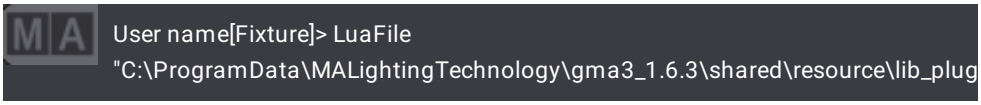
The LuaFile keyword is used to execute a Lua file directly without the need to import it into the grandMA3 show file.

Syntax

LuaFile "AbsolutePathToLuaFile/[filename.lua]"

Example

- To execute the Lua file "execute_example.lua", type:



8.2.180. Macro Keyword

To enter the **Macro** keyword in the command line, use one of the options:

- Press MA + X14
- Type **Macro**
- Type the shortcut **M**

Description

The Macro keyword is an object keyword used to access Macros.

The default function for macro is Go+. This means calling macros without specifying any function starts the macro.

For more information see **Macros**.

Syntax

[Function] Macro [ID]

[Function] DataPool [Data Pool ID] Macro [ID]

Examples

- To start macro 5, type:

```
User name[Fixture]> Macro 5
```

- To set the wait time of macro 3 line 4 to Go, type:

```
User name[Fixture]> Set Macro 3.4 "Wait" "Go"
```

- To store a new empty macro 2, type:

```
User name[Fixture]> Store Macro 2
```

8.2.181. MArker Keyword

To enter the MArker keyword in the command line, use one of the options:

- Type **MArker**
- Type the shortcut **MAr**

Description

The MArker keyword is an object keyword that is used to select fixtures of the ID type MArker.

MArker [ID]

Example

- To select MArker 30, type:

8.2.182. Master Keyword

To enter the **Master** keyword in the command line, use one of the options:

- Type **Master**
- Type the shortcut **Mas**

Description

The Master keyword is an object keyword that is used to assign the different master functions, e.g., for the selected sequence, grand masters, speed masters and playback masters.

Syntax

Assign Master [Object Number] At Executor [Number]

Examples

- To assign the master to the selected sequence on executor 206, type:

- To assign the grand master to executor 207, type:

- To assign the speed master 2 to executor 208, type:

- To assign the playback master 3 to executor 209, type:

For more information on how to assign objects to executors see **Assign Object to an Executor**.

8.2.183. MAtricks keyword

To enter the MAtricks keyword in the command line, use one of the following options:

- Type **MAtricks**
- Type the shortcut **MAt**

Syntax

Set Selection (ID) MAtricks [Property "Property name" Value]

[Function] MAtricks ["Name"/ID]

Properties

The MAtricks are structured like this:

X	Y	Z
XBlock	YBlock	ZBlock
XGroup	YGroup	ZGroup
XWings	YWings	ZWings
XWidth	YWidth	ZWidth
XShuffle	YShuffle	ZShuffle
XShift	YShift	ZShift

Description

The MAtricks keyword acts as an object type.

Used with an ID, MAtricks represents MAtricks objects stored in the MAtricks pool.

With the helping keywords On, Off, and Toggle, the MAtricks of the two selections may temporarily be enabled/disabled.

Furthermore, you can set the values of the MAtricks of the two selections.

Examples

- To set the MAtricks X to 2 for the active selection, type:

- To set the MAtricks XBlock to 4 for selection 2, type:

- To switch the MAtricks off for the active selection, type:

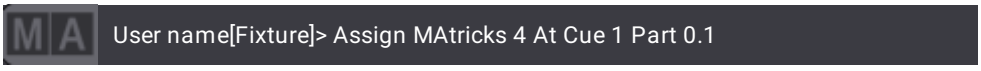
- To toggle MAtricks active for the active selection, press Set or type:

- To reset the MAtricks for the first selection, type:

- To call the first MAtricks object in the MAtricks pool, type:

- To label MAtricks 2 with the name "Great", type:

- To assign the fourth MAtricks object of the pool to the first recipe in cue 1 part 0 of the selected sequence, type:



8.2.184. Media Keyword

To enter the **Media** keyword in the command line, use one of the options:

- Type **F + 5**
- Type **Media**
- Type the shortcut **Med**

Description

The Media keyword is an object keyword used to call fixtures of the ID type Media in the programmer.

The name of this keyword can change because it is a custom ID type. For more information see the **Custom ID Type Topic**.

Syntax

Media [ID]

Example

- To select the fixture with Media ID 2, type:

8.2.186. Menu keyword

To enter the **Menu** keyword in the command line, use one of the options:

- Type **Menu**
- Type the shortcut **Men**

Description

The Menu keyword is an object keyword used to address menus.

Syntax

Menu ["Menu Name"]

[Function] Menu

Examples

- To open the menu Network, type:

- To list all menus, type:

  The command line feedback window lists all menus you can open using this keyword.

- To display the rear panel connectors on the letterbox screens, type:

8.2.187. Mesh3DS keyword

To enter the **Mesh3DS** keyword in the command line, use one of the options:

- Type **Mesh3DS**
- Type the shortcut **Mesh3**

Description

The Mesh3DS keyword is an object keyword used to address meshes.

Syntax

[Function] Mesh3DS (ID)

Examples

- To list all Mesh3DS of the show file, type:

8.2.188. MessageCenter Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.189. Monitor Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.190. Move Keyword

To enter the Move keyword in the command line, use one of the options:

- Press Move
- Type **Move**
- Type the shortcut **Mo**

Description

The Move keyword is a function keyword used to move objects to another location giving them a new object ID.

If the destination is already taken, the object will be moved to the target object. The previous target object will be moved to the next object until an empty object will be occupied.

Syntax

Move [Object] At [Object]

Example

- To move macro 1 to macro 3, type:

8.2.191. MyRunningMacro Keyword

To enter the MyRunningMacro keyword in the command line, use one of the options:

- Type **MyRunningMacro**
- Type the shortcut **MyRunningM**

Description

The MyRunningMacro keyword is used when there are several users in a session.

It allows you to list or disable your own running macros in a show file.

Important:
The numbers of the running macros are not equal to the numbers in the macro pool.

Syntax

[Function] MyRunningMacro [Number]

Example

- To disable the first macro that is running and which you started, type:

```
User name[Fixture]> Off MyRunningMacro 1
```

- To list all your running macros, change destination to the running macros first:

```
User name[Fixture]> ChangeDestination MyRunningMacro
```

Result:

```
User name@Temp/MyRunningMacros>
```

- To list all your running macros, type:

```
User name@Temp/MyRunningMacros> List
```

8.2.192. MyRunningSequence Keyword

To enter the MyRunningSequence keyword in the command line, use one of the options:

- Type **MyRunningSequence**
- Type the shortcut **My**

Description

The MyRunningSequence keyword is used when there are several users in a session.

It allows you to disable your own running sequences in a show file.

Important:
The numbers of the running sequences are not equal to the numbers in the sequence pool.

Syntax

[Function] MyRunningSequence [Number]

Example

- To disable the first sequence that is running and which you started, type:

```
User name[Fixture]> Off MyRunningSequence 1
```

- To list all your running sequences, change destination to the running sequences first:

```
User name[Fixture]> ChangeDestination MyRunningSequence
```

Result:

```
User name@Temp/MyRunningSequences>
```

- To list all your running sequences, type:

```
User name@Temp/MyRunningSequences> List
```

8.2.193. NewShow keyword

To enter the NewShow keyword in the command line, use one of the options:

- Type **NewShow**
- Type the shortcut **New**

Description

The NewShow keyword is a function keyword that is used to create new shows.

Syntax

NewShow "Show Name"

Example

- To create a new show file with the file name "MacBeth", type:

8.2.196. Next Keyword

To enter the **Next** keyword in the command line, use one of the options:

- Press Next
- Type **Next**
- Type the shortcut **N**

Important:
To execute the command directly, press Next.

Description

If no fixtures are selected and the default keyword is Fixture, the Next keyword selects the fixture with the lowest fixture ID.

If only one fixture is selected and the default keyword is Fixture, the Next keyword selects the following fixture.

If multiple fixtures are selected, the Next keyword selects the following fixture one after the other within the selected block of fixtures.

Syntax

Next

Example

- To step through single fixtures in the selected block of fixtures, press Next.

Name	FID	IDType	CID	Dimmer	PanTilt	
				Dim	P	T
AS QWO 1	1	Fixture		60	50	50
AS QWO 2	2	Fixture		60	50	50
AS QWO 3	3	Fixture		60	50	50
AS QWO 4	4	Fixture		60	50	50
AS QWO 5	5	Fixture		60	50	50
AS QWO 6	6	Fixture		60	50	50
AS QWO 7	7	Fixture		60	50	50
AS QWO 8	8	Fixture		60	50	50
AS QWO 9	9	Fixture		60	50	50
AS QWO 10	10	Fixture		60	50	50
AS QWO 11	11	Fixture		0	50	50
AS QWO 12	12	Fixture		0	50	50
AS QWO 13	13	Fixture		0	50	50

Using Next in a selected block of fixtures

8.2.197. Node Keyword

To enter the Node keyword in the command line, use one of the options:

- Type **Node**
- Type the shortcut **Nod**

Description

The Node keyword is an object keyword used to display all nodes in the network

This keyword belongs to a session-related group of keywords. For more information see **Console keyword**, **onPC keyword**, **PU keyword**, **Station keyword**, **Extension keyword**, **Join keyword**, **Leave keyword**, **Invite keyword**, **Dismiss keyword**, **Reboot keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

Node [ID]

Node [Name]

Examples

To list all nodes that are currently in the same network, type:

To invite node "FOH3" into the session, type:

8.2.198. NonDim Keyword

To enter the **NonDim** keyword in the command line, use one of the options:

- Type **F** + **4**
- Type **NonDim**
- Type the shortcut **Non**

Description

The NonDim keyword is an object keyword used to call fixtures of the ID type NonDim in the programmer.

The name of this keyword can change because it is a custom ID type. For more information see the **Custom ID Type Topic**.

Syntax

NonDim [ID]

Example

- To select the fixture with NonDim ID 1, type:

8.2.199. Normal Keyword

Important:
Pressing At twice directly executes the Normal keyword.

To enter the **Normal** keyword in the command line, use one of the options:

- Press At At
- Type **Normal**
- Type the shortcut **No**

Description

Normal is an object keyword used to set the dimmer of the fixtures to the defined normal value.

For information on how to change the normal value see **User Settings**.

Syntax

Normal

Example

- To set the dimmer of the selected fixtures to the normal value, type:

8.2.200. Off Keyword

To enter the Off keyword in the command line, use one of the options:

- Press Off
- Type **Off**
- Type the shortcut **Of**

Description

The Off keyword is used as a function keyword to:

- Stop an executor
- Knock out parameters in the programmer
- Knock out selections in the programmer
- Knock out active attributes in the programmer

Syntax

Off [Object List]

Examples

- To knock out the parameters of fixture 2 and 4 in the programmer, type:

- To knock out the values of sequence 1, type:

- To turn off all sequences containing loaded cues, type:

8.2.201. On Keyword

To enter the On keyword in the command line, use one of the options:

- Press On
- Type **On**

Description

The On keyword is used as a function or as a helping keyword.

Use the On keyword as a function keyword to:

- Start or restart an executor
- Activate selection in the programmer
- Activate attributes in the programmer

Use the On keyword as a helping keyword to:

- Indicate the start of a temporary function
- Enable the state of a toggle function

Syntax

Function keyword:

On [Object List]

Helping keyword:

[Function] On

Examples

- To activate the attributes of fixture 2 and 4 in the programmer, type:

- To turn on sequence 3, type:

8.2.202. OnPC Keyword

To enter the OnPC keyword in the command line, use one of the options:

- Type **OnPC**
- Type the shortcut **OnP**

Description

The OnPC keyword is an object keyword used to display all OnPC stations in the network.

This keyword belongs to a session-related group of keywords. For more information see **Console keyword**, **Node keyword**, **PU keyword**, **Station keyword**, **Extension keyword**, **Join keyword**, **Leave keyword**, **Invite keyword**, **Dismiss keyword**, **Reboot keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

OnPC [ID]

OnPC [Name]

Examples

To list all OnPC stations that are currently in the same network, type:

To invite the OnPC station "FOH3" into the session, type:

8.2.203. OSC Keyword

To enter the **OSC** keyword in the command line, use one of the options:

- Type **OSC**
- Type the shortcut **Os**

Description

The OSC keyword represents the open sound control data.

OSC keyword lists, exports or imports open sound control data in the Remote Inputs.

Syntax

[Function List] OSC

Examples

1. List all OSC data in an interface

1. Change destination to OSC:

```
User name[Fixture]> ChangeDestination OSC
```

Result:

```
User name@ShowData/OSCBase>
```

2. To list OSC data in the interface, type:

```
User name@ShowData/OSCBase> List
```

2. Import OSC data in Remote Input

Requirement: Change destination to OSC.

For more information see **Example 1**.

- To import OSC data, type:

If the target already exists, the import pop-up appears, asking you to choose an option:

Target is occupied

- Choose an option.

The OSC is imported.

8.2.204. Oops Keyword

Hint:
Use Undo as a synonym of the Oops keyword.

To enter the Oops keyword in the command line, use one of the options:

- Press Oops
- Type **Oops**
- Type the shortcut **O**

Description

The Oops keyword is a function keyword used to cancel:

- The last operation made in the command line
- The last fixture selection
- The last operation made in the programmer

Syntax

Oops

Example

- To undo the last operation, type:

8.2.205. Output Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.206. OutputLayer Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.207. Page Keyword

To enter the Page keyword in the command line, use one of the options:

- Press MA + X15
- Type **Page**
- Type the shortcut **P**

Important:
To call a page, the page must exist in the page pool.

Description

The Page keyword is an object keyword used to access pages.

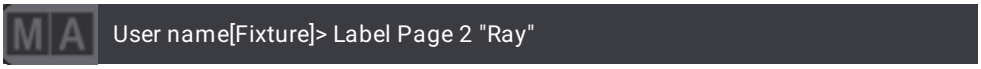
Syntax

[Function] Page [ID]

Examples

- To change to page 2, type:

- To label page 2 as "Ray", type:



- To call the page with the name "Ray", type:

- To change to the next page, type:

8.2.208. Part Keyword

To enter the Part keyword in the command line, use one of the options:

- Press Cue Cue
- Type **Part**
- Type the shortcut **Par**

Description

The Part keyword is an object keyword to segment cues in parts.

Parts are useful for assignment and edit of different timings of groups of fixture parameters.

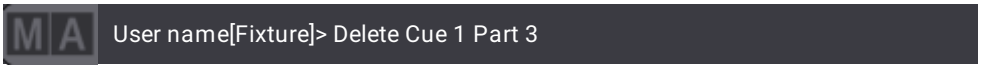
Syntax

[Function] Part [Number]

Examples

- To create a second part of cue 3 in the selected executor, type:

- To delete part 3 of cue 1, type:



8.2.209. Park keyword

To enter the Park keyword in the command line, use one of the options:

- Press Pause Pause
- Type **Park**

Description

The Park keyword is a command keyword to prevent DMX channels of fixtures to change their value.

Syntax

Park [Object]

Examples

- To park fixture 1 with all its attributes, type:

- To park the current selection, type:

It is also possible to enter Park into the command line and tap a cell in the fixture sheet to park a certain attribute.

To unpark fixtures and/or attributes see the **Unpark Keyword**.

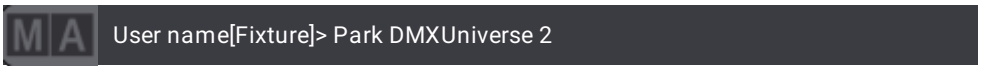
Important:
When parking a fixture and/or attribute it will park the corresponding DMX channel.

- To park fixture 1 at a value 50, type:

Hint:
The command Park Fixture At will park all DMX channels of the fixture to the set value.

- To park only DMX channels for FeatureGroup 1 at 50, type:

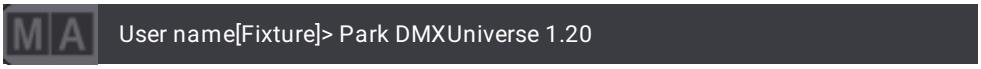
- To park DMX universe 2, type:



To specify the desired universe:

1. Enter the Park keyword in the command line.
2. Tap the universe in the universe pool.

- To park DMX channel 20 on the first universe, type:



To specify the desired DMX channel:

1. Enter the Park keyword in the command line.
2. Tap the channel in the DMX sheet.

Hint:
If there are parked channels in a universe, this will be indicated by a blue **P** icon in the universe pool.

8.2.210. Paste Keyword

To enter the Paste keyword in the command line, use one of the options:

- Press Copy Copy
- Type **Paste**
- Type the shortcut **Pas**

Description

The Paste keyword pastes previously copied content or moves previously cut object lists.

For more information see the **Cut keyword** and the **Copy keyword**.

Syntax

Paste [Starting Object]

Examples

- To move group 1 to group 5, type:

- To copy cue 5 to cue 15, type:

8.2.211. Patch Keyword

To enter the **Patch** keyword in the command line, use one of the options:

- Type **Patch**
- Type the shortcut **Pat**

Description

The Patch keyword is used to edit the patch address of single fixtures or entire fixture selections.

Syntax

Patch Fixture [Number] [Universe.Address]

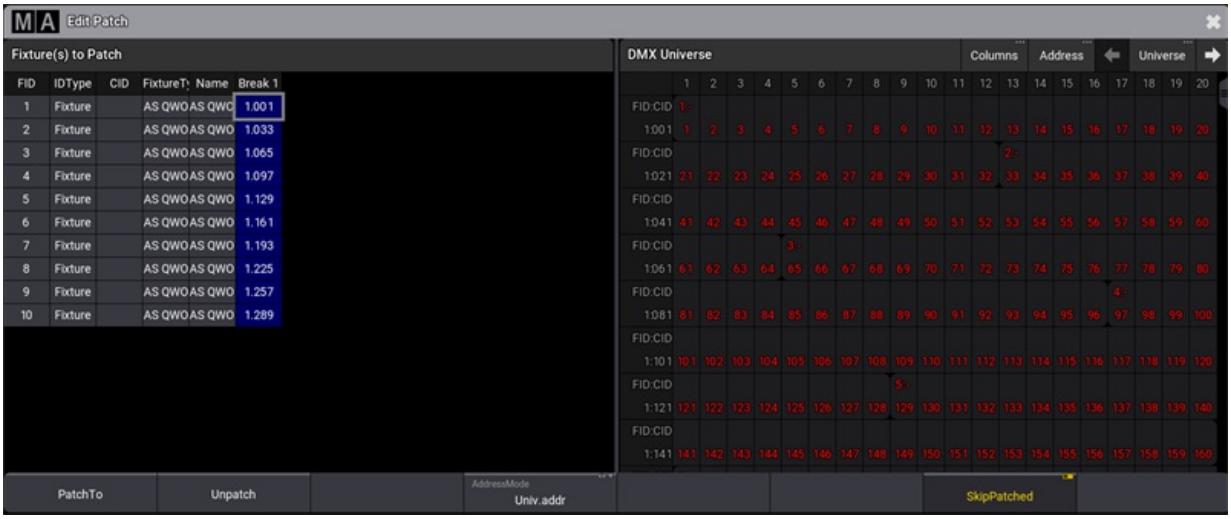
Patch Fixture [Number] Thru [Number]

Example

- To patch fixture 2 to the patch address 123 on universe 3, type:

- To edit the patch address of fixtures 1 through 10, type:

The Edit Patch opens:



Edit the patched fixtures

You can now patch the fixtures to a new patch address or unpatch them using the user interface.

8.2.212. Pause Keyword

To enter the Pause keyword in the command line, use one of the options:

- Press Pause
- Type **Pause**
- Type the shortcut **Pau**

Description

The Pause keyword is a function keyword used to pause:

- Crossfades between steps and cues
- Timecode shows
- Macros

Hint:
Pause is a toggle function between **Pause On** and **Pause Off**.

Syntax

Pause [Object List]

Pause On [Object List]

Pause Off [Object List]

Examples

- To pause an active phaser in the sequence on executor 201, type:

```
User name[Fixture]> Pause Executor 201
```

- To restart the phaser in the sequence of executor 201, type:

```
User name[Fixture]> Pause Executor 201
```

8.2.213. Percent Keyword

To enter the Percent keyword in the command line, use one of the options:

- Type **Percent**
- Type the shortcut **Pe**

Description

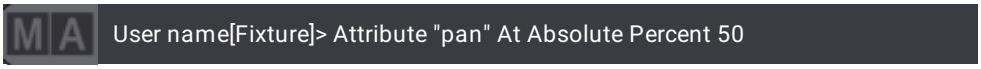
The Percent keyword is used to set the values of a fixture selection using the percent (%) notation.

Syntax

At [Layer] Percent [Value]

Example

- To set the pan value in the absolute layer to 50 in percent, type:



8.2.214. PercentFine Keyword

To enter the PercentFine keyword in the command line, use one of the options:

- Type **PercentFine**
- Type the shortcut **Percentf**

Description

The PercentFine keyword is used to set the values of a fixture selection taking into consideration two decimal places.

Syntax

At [Layer] PercentFine[Value]

Examples

- To set the absolute layer to 50.35 in PercentFine, type:

```
User name[Fixture]> At Absolute PercentFine 50.35
```

- To set the pan value in the absolute layer to 75.50, type:

```
User name[Fixture]> Attribute "pan" At Absolute PercentFine 75.50
```

8.2.215. Phase Keyword

To enter the Phase keyword in the command line, use one of the options:

- Type **Phase**
- Type the shortcut **Ph**

Description

Phase is an object keyword that is used to set the Phase layer.

For more information see **Phasers**.

Syntax

Phase

At Phase [Value] Thru [Value]

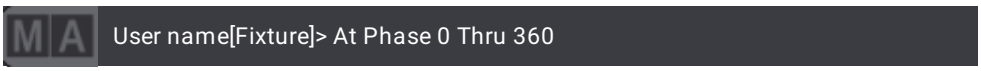
Examples

- To change to the phase layer, type:

Requirement:

In this example, at least 2 steps have to be active in the programmer.

- To set a phase of 0 through 360 in the selected attribute, type:



8.2.216. Physical Keyword

To enter the Physical keyword in the command line, use one of the options:

- Type **Physical**
- Type the shortcut **Phy**

Description

The Physical keyword is used to set the physical values of a fixture selection using the Physical notation. It comprises RPM (rounds per minute), Hz (Hertz), degrees, or intensity.

Syntax

At [Layer] Physical [Value]

Examples

- To set the dimmer value to 1.0 in the absolute layer using Physical, type:

- To set the pan value in the absolute layer to 75.60 degrees using Physical, type:

8.2.217. Plugin Keyword

To enter the Plugin keyword in the command line, use one of the options:

- Press MA + X14 + X14
- Type **Plugin**
- Type the shortcut **Pl**

Description

The Plugin keyword is an object keyword used to access plugins.

The default function of the Plugin keyword is **Go+**.

Syntax

[Function] Plugin [ID]

Examples

- To edit plugin 2, type:

- To label plugin 1 "Weakon," type:

Call a Plugin and Specify an Argument

Requirement:

- Create a plugin that uses an argument when calling a function.

In this example our plugin is plugin 1 of the plugin pool and the argument is called name in the definition of the function.

```
function main(display_handle, name)

  Printf("My name is "..name)

end

return main
```

- To generate the sentence "My name is Richard Roe" in the command line history, type:

```
User name[Fixture]> Plugin 1 "Richard Roe"
```

Result:

```
OK:Plugin 1 "Richard Roe"
My name is Richard Roe
OK:Menu "CommandLineHistory"
```

Respone in the command line history

"My name is Richard Roe" is now displayed in the command line history.

Call a Dedicated LuaComponent

Requirement:

- Create at least two lua components in the plugin.

For more information on how to create lua components see **Plugins**.

- To call the second LuaComponent of plugin 1, type:

```
User name[Fixture]> Plugin 1.2
```

8.2.218. Preset Keyword

To enter the Preset keyword in the command line, use one of the options:

- Press Preset
- Type **Preset**
- Type the shortcut **Pres**

Description

With the Preset keyword you can:

- Select the fixtures stored in a preset
- Apply the At function in the preset within the fixture or channel selection

If no fixtures or channels are selected in the programmer, the default function for preset pool objects will be **SelFix**. For more information see **SelFix Keyword**.

If fixtures or channels are selected in the programmer, the default function will be **At**. For more information see **At Keyword**.

Syntax

Preset [Feature Group].[ID]

Preset [Feature Group]."Name"

Set Preset [Feature Group].[ID] [Parameter] [Parameter Value]

Assign [Object] [ID] At Preset [Feature Group].[ID]

Objects

The following table displays the objects that can be assigned using the **Assign Keyword**.

Object	Object Value	Description
Appearance	"Appearance 1"	Assigns the appearance for the pool object.
InputFilter	"Filter 12"	Assigns the filter for the pool object.
Scribble	"Scribble 1"	Assigns the scribble for the preset pool object.

Parameter

The following table displays the available parameters that can be set using the **Set Keyword**.

Parameter	Parameter Value	Description
Name	"Preset Name"	Sets the preset pool object name.

Examples

- To select the fixtures that can use preset 5 of the dimmer feature group, type:

```
User name[Fixture]> SelFix Preset 1.5
```

- To select the fixtures stored in any preset with the name "DarkRed", type:

```
User name[Fixture]> SelFix Preset *."DarkRed"
```

- To set the name of the preset to the position preset 3, type:

```
User name[Fixture]> Set Preset 2.3 Name "Stage Left"
```

- To assign world 5 to the preset position 4, type:

```
User name[Fixture]> Assign World 5 At Preset 2.4
```

8.2.219. PresetUpdate Keyword

To enter the PresetUpdate keyword in the command line, use one of these options:

- Type **PresetUpdate**
- Type the shortcut **PresetU**

Description

PresetUpdate contains all presets that could be updated at the moment.

It needs the **List keyword** to show the preset list.

The PresetUpdate keyword is an Object keyword.

Syntax

List PresetUpdate

Example

- To list all the presets that can be updated in the command line feedback, type:

It is the same list visible in the **Update menu**.

8.2.220. Press Keyword

To enter the Press keyword in the command line, type **Press**.

Description

The Press keyword is used to simulate the pressed state of a key.

It is possible to assign this state to macros.

For more information see **Macros**.

Syntax

[Function] Press [Executor List]

Example

- To execute Flash on the executor 203, type:

To unpress a key, see **Unpress Keyword**.

8.2.221. Preview Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.224. Previous Keyword

To enter the Previous keyword in the command line, use one of the options:

- Press Prev
- Type **Previous**
- Type the shortcut **Prev**

Description

The Preview keyword has several functions:

- If no fixture is selected, the fixture with the highest ID number will be selected.
- If one fixture is selected, the fixture before this fixture will be selected.
- If MAtricks are active, the Previous keyword has special functions depending on the selected MAtricks options.

Syntax

Previous

Example

- To select the last fixture of the selected fixtures in the selection order, type:

8.2.225. Programmer Keyword

To enter the Programmer keyword in the command line, use one of the options:

- Press MA + Cue
- Type **Programmer**
- Type the shortcut **Prog**

Description

The Programmer keyword represents the programmer playback. It is used to enable, disable, or pause values in the programmer.

Syntax

On Programmer

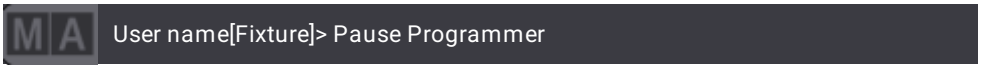
Pause Programmer

Off Programmer

Examples

Requirement: Phasers

- To pause all phasers and transitions running in the programmer, type:



- To create programmer part 42, type:

8.2.226. Property Keyword

To enter the Programmer keyword in the command line, use one of the options:

- Type **Property**
- Type the shortcut **Prop**

Description

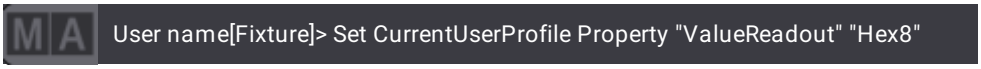
The Property keyword is an object keyword used to communicate to the console that you would like to set a specific property.

Syntax

[Function] [Object List] [Value List] [Property] [List]

Example

- To set the ValueReadout to Hex8 in the current user profile, type:



8.2.227. PU Keyword

To enter the PU keyword in the command line, use one of the options:

- Type **PU**

Description

The PU keyword is an object keyword used to display all Processing Units in the network.

This keyword belongs to a session-related group of keywords. For more information see **Console keyword**, **onPC keyword**, **Node keyword**, **Station keyword**, **Extension keyword**, **Join keyword**, **Leave keyword**, **Invite keyword**, **Dismiss keyword**, **Reboot keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

PU [ID]

PU [Name]

Examples

To list all Processing Units that are currently in the same network, type:

To invite Processing Unit "FOH3" into the session, type:

8.2.228. Pyro Keyword

To enter the Pyro keyword in the command line, use one of the options:

- Type **F** + **8**
- Type **Pyro**
- Type the shortcut **Py**

Description

The Pyro keyword is an object keyword used to call fixtures of the ID type Pyro in the programmer.

The name of this keyword can change because it is a custom ID type. For more information see the **Custom ID Type Topic**.

Syntax

Pyro [ID]

Example

- To select the the fixture with pyro ID 8, type:

8.2.229. Rate1 Keyword

To enter the Rate1 keyword in the command line, use one of the options:

- Press MA + Learn
- Type **Rate1**
- Type the shortcut **Ra**

Description

The Rate1 keyword is a function keyword used to reset the rate of a phaser to 1:1.

Syntax

Rate1 [Object List]

Example

- To reset the rate of executor 105 back to 1:1, type:

8.2.230. Readout Keyword

Important:
Changing the Readout is a user profile setting. That is, the Readout will be automatically set throughout the entire user profile.
For more information see **User Settings**.

Hint:
-The Readout values are:
Decimal8, Decimal24, Hex8, Hex24, Percent, PercentFine, Physical.
-The names of the readout are case-sensitive, e.g., "**H**ex8", "**P**ercent".

To enter the Readout keyword in the command line, use one of the options:

- Type **Readout**
- Type the shortcut **Rea**

Description

The Readout keyword is used to set the readout of fixtures in the usual notation.

Syntax

Readout [Name]

Example

- To set the Readout to Hex8, type:

8.2.231. Reboot Keyword

To enter the Reboot keyword in the command line, use one of the following options:

- Type **Reboot**
- Type the shortcut **R**

Description

The Reboot keyword is a function keyword used to shut down the desired station and boot it up again.

A confirmation pop-up opens on the station in use.

This keyword belongs to a session-related group of keywords. For more information see **Console keyword**, **Node keyword**, **OnPC keyword**, **PU keyword**, **Station keyword**, **Extension keyword**, **Join keyword**, **Leave keyword**, **Invite keyword**, **Dismiss keyword**, **Restart keyword**, **Shutdown keyword**.

Syntax

Reboot IP [IP Address]

Reboot [Device Type] [Station List]

Hint:
The device types are: **Console**, **Node**, **onPC**, **PU**, **Station**, **Extension**.

Examples

- To reboot the connected grandMA3 processing unit 1, type:

- To reboot the connected grandMA3 processing unit called "Stage Right", type:

- To reboot the console with the IP address 192.168.0.4, type:

8.2.232. Recast Keyword

To enter the Recast keyword in the command line, use one of the options:

- Press MA + X1 + X1
- Type **Recast**
- Type the shortcut **Reca**

The Recast keyword is a command keyword used to update fixtures in sequences with changed preset information. Recast works with attributes that were added or removed.

The second use-case for Recast: executor configurations. When an executor configuration is used on several executors, and the handle assignment for one of these execs is changed, the changes are not transmitted automatically to the other executors using this configuration. When storing the changes into the executor configuration, it is possible to recast the executor configuration. All other executors using this configuration will then get the new handle assignment, too. For more information about executor configurations, see the **Executor Configurations** topic.

Syntax

Recast

Recast Preset [ID]

Example

- The dimmer is open and the color is red for 10 fixtures in All preset 21.1. This preset is used in sequence 1. We now add a position to the preset. To recast preset 21.1, type:

8.2.233. Relation Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.234. Relative Keyword

To enter the Relative keyword in the command line, use one of the options:

- Press MA + Time
- Type **Relative**

Description

The Relative keyword is an object keyword used to set the relative layer.

Syntax

Relative

[Attribute "Name"] [At] Relative [Value]

Examples

- To set the layer to relative, type:

- To set the color red to 10 in the programmer, type:

8.2.235. Release Keyword

To enter the **Release** keyword in the command line, use one of the options:

- Press Delete Delete Delete
- Type **Release**
- Type the shortcut **Rel**

Description

The Release keyword enters release values in the programmer depending on specified attributes and the selected fixtures.

Release values that are stored using the merge function release previously tracked values in the tracking list. The fixtures use their default values.

If you release an object, the release is applied to the value layer.

Syntax

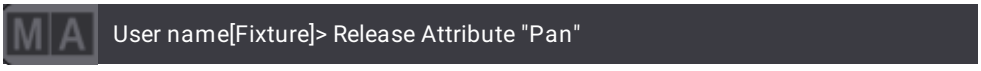
Release [Object List]

Examples

- To enter release values in the programmer for all attributes of the fixture selection in the value layer, type:

Release is displayed in the value layer.

- To enter release values in the programmer for the attribute PAN of the fixture selection in the value layer, type:



- To enter release values in the programmer for fixture 1, type:

8.2.236. ReloadPlugins Keyword

To enter the ReloadPlugins keyword in the command line, use one of the options:

- Type **ReloadPlugins**
- Type the shortcut **ReloadP**
- Or the shortcut **RP**

Description

The ReloadPlugins keyword is a function keyword used to restart the LUA engine.

The restart of the LUA engine is necessary after you did any changes regarding LUA because this could change how the LUA behaves.

You may want to test the integrity of the LUA system to make sure that it behaves as expected also next time you load the show. This is important as the saved show file does not contain a snapshot of the LUA memory, it only contains the inbuilt functions and the code in the defined plugins.

When the show file is loaded, the LUA engine and plugins code are (re)loaded, which may result in a different state than when you powered down the console or saved the show.

Hint:
Double-check the executed command in the system monitor.

Syntax

ReloadPlugins

Example

- To restart the LUA engine after LUA programming, type:

8.2.237. ReloadUI Keyword

To enter the ReloadUI keyword in the command line, use one of the following options:

- Type **ReloadUI**
- Type **Relo**
- Type the shortcut **RU**

Description

The ReloadUI keyword is a function keyword used to refresh the user interface of the console.

Syntax

ReloadUI

Example

- To refresh the user interface of the console, type:

8.2.238. Remote Keyword

To enter the Remote keyword in the command line, use one of the options:

- Type **Remote**
- Type the shortcut **Rem**

Description

The Remote keyword is an object keyword used to access the remote input types.

You can store or delete remote input types and set parameters.

Syntax

[Function] Remote [Remote Input Type ID].[Row Number] ["Parameter"] ["Parameter Value"]

Assign [Object] [Object ID] At Remote [Remote Input Type ID].[Row Number]

The following table displays the available remote input types along with their remote input type ID.

Remote Input Type	Remote Input Type ID
DC Remote	1
MIDI Remote	2
DMX Remote	3

In input type the IDs have to be in an order and have to start with 1.

Parameter

The following table displays the parameters you can set using the command line with the help of the **Set Keyword**.

Hint:
If an option or any other part of the keyword command requires two quotation marks, the outer quotation marks are "+" and the inner quotation marks are '+'.

Parameter	Parameter Value	Description
Lock	"Yes", "No"	Sets the lock status.
Name	"This is the name of the remote"	Sets the name of the remote.
Target	"World", "Sequence", "Macro", "Group", "Menu", "Plugin", "View", "Master"	Sets the target of the action when the contact is active.
Fader	"Master", "X", "Temp", and all the fader functions.	Sets the fader the console should activate.

Parameter	Parameter Value	Description
Key	"Fix", "Select", "SelFix", and all the key functions.	Sets the key the console should activate.
TriggerOn	"0%...100%"	Sets when the trigger will be set to on.
TriggerOff	"0%...100%"	Sets when the trigger will be set to on.
InFrom	"0%...100%"	Sets the starting point of the range of the used incoming signal.
InTo	"0%...100%"	Sets the end point of the range of the used incoming signal.
OutFrom	"0%...100%"	Sets the starting point of the range of the used outgoing signal.
OutTo	"0%...100%"	Sets the end point of the range of the used outgoing signal.
Enabled	"Yes", "No"	Sets the status to enabled or not enabled.
Only for DMX remotes: Address	1.001...1024.512 [universe].[dmx address]	Sets the DMX universe and address.
Only for DMX remotes: Resolution	"8bit", "16bit", "24bit"	Sets the DMX resolution. The DMX channels have to be consecutive for 16 bit and 24 bit.
Only for MIDI remotes: MIDIChannel	"1, 2, 3, ..., 16"	Sets the MIDI channel.
Only for MIDI remotes: MIDIIndex	"1, 2, 3, ..., 128"	Sets the MIDI index.
Only for MIDI remotes: MIDIType	"Note", "NoteAttack", "NoteAttackDecay", "Control"	Sets the MIDI type. Note = MIDI note only NoteAttack = MIDI note and uses the velocity to regulate the master except note off NoteAttackDecay = MIDI note and uses the velocity to regulate the master with note off Control = Control change (CC) messages.

Parameter	Parameter Value	Description
Only for DC remotes: DC start signal	"1, 2, 3, ..., 64"	Sets the DC start signal.

The start signal and the MIDI offset of the desired input console for **DC Remotes** and **MIDI Remotes** have to be set in the **output configuration menu**.

Examples

- To set the key of the first DMX remote to Go+, type:

```
User name[Fixture]> Set Remote 3.1 "Key"  "Go+"
```

- To store a new MIDI remote, type:

```
User name[Fixture]> Store Remote 2.1
```

- To assign sequence 2 to the second DMX remote, type:

```
User name[Fixture]> Assign Sequence 2 At Remote 3.2
```

8.2.239. RemoteHID Keyword

To enter the RemoteHID keyword, use one of the options:

- Type **RemoteHID**

Description

The RemoteHID keyword connects a mouse and/or external keyboard to different stations.

Syntax

RemoteHID IP [IP]

RemoteHID onPC [station name]

Example

- To connect to the station with the IP 192.168.0.4, type:

- To connect to the onPC station that has the name "3D", type:

8.2.240. RemoteCommand Keyword

To enter the RemoteCommand keyword, use one of the options:

- Type **RemoteCommand**
- Type **RemoteC**

Description

The RemoteCommand keyword remotely sends commands to other stations.

Syntax

RemoteCommand IP [IP] ["Command to be Executed"]

RemoteCommand [Station Type] [ID/"Name"] ["Command to be Executed"]

Example

- To remotely execute the command "Call ViewButton 2.1" on the station with the IP address 192.168.0.10, type:

- To remotely lock the desk on the station with the IP address 192.168.0.10, type:

Alternatively type:

For more information on the usage of quotation marks see **General Syntax Rules**.

For more mutual examples see the **Station Keyword**.

8.2.241. Remove Keyword

To enter the **Remove** keyword in the command line, use one of the options:

- Press Delete Delete
- Type **Remove**
- Type the shortcut **Remov**

Description

The Remove keyword enters remove values in the programmer depending on specified attributes and the selected fixtures.

Remove values that are stored using the merge function remove previously stored values.

If a stored value is removed, values from the previous cue will be tracked again.

Syntax

Remove [Object List]

Examples

- To remove values in the selected layer for the dimmer attribute of the current selection, type:

```
User name[Fixture]> Remove
```

- To set values in the feature group Position of the selected layer of the current selection, type:

```
User name[Fixture]> Remove FeatureGroup "Position"
```

8.2.242. Reset Keyword

To enter the Reset keyword in the command line, use one of the options:

- Type **Reset**
- Type the shortcut **Rese**

Description

The Reset keyword is a function keyword used to clear the matricks settings.

Syntax

Reset

Example

- To reset the current matricks settings, type:

8.2.243. Restart Keyword

To enter the Restart keyword in the command line, use one of the options:

- Type **Restart**
- Type the shortcut **Res**

Description

The Restart keyword is a function keyword and used to restart the application. This is the same as close the program and open it again without shutting down the console.

To switch between **Mode2** and grandMA3 mode, use the Restart keyword.

Syntax

Restart (/option)

Restart [device type] [station list] (/option)

Restart IP [IP address] (/option)

Options

The shutdown keyword has the following options:

Option	Option Shortcut	Description
noconfirm	nc	Suppress shutdown confirmation pop-up.

Examples

- Restart the application of the console.

```
User name[Fixture]> Restart
```

- To restart the application of the console without confirmation window, type:

```
User name[Fixture]> Restart /nc
```

- Restart the application of the station with the IP address 192.168.0.32 in the network.

```
User name[Fixture]> Restart IP 192.168.0.32
```

8.2.244. Root Keyword

Important:
We recommend using the index name of the root. If you use root index numbers, double-check after every release if the root index number is still valid. If the root index number has changed, adjust your macros.

To enter the Root keyword in the command line, use one of the options:

- Type **Root**
- Type the shortcut **Ro**

Description

The Root keyword is an object keyword used to access the root in the object tree.

Syntax

Root [root-index]

Root [root-index].[sub-index]

Root [root-index].[sub-index].[sub-sub-index]

Example

- To change the destination back to root, type:

8.2.245. RTChannel

To enter the RTChannel keyword in the command line, use one of the options:

- Type **RTChannel**
- Type the shortcut **RT**

Description

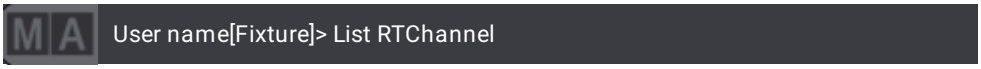
The RTChannel keyword is used to address realtime channels.

Syntax

[Function] RTChannel

Example

- To list all realtime channels in a show file, type:



Realtime channels are also displayed in the Parameter List of the Patch.

For more information see **Patch and Fixture Setup**.

8.2.246. Running Macro Keyword

To enter the RunningMacro keyword in the command line, use one of the options:

- Type **RunningMacro**
- Type the shortcut **RunningM**

Description

The RunningMacro keyword is useful when there are several users in a session.

It allows you to list or disable all running macros in a show file.

Important:
The numbers of the running macros are not equal to the numbers in the macro pool.

Syntax

[Off] RunningMacro [Number]

Example

- To disable the first macro that is currently running, type:

```
User name[Fixture]> Off RunningMacro 1
```

- To list all running macros, change destination to the running macros first:

```
User name[Fixture]> ChangeDestination RunningMacro
```

Result:

```
User name@Temp/RunningMacros>
```

- To list all running macros, type:

```
User name@Temp/RunningMacros> List
```

8.2.247. RunningSequence Keyword

To enter the RunningSequence keyword in the command line, use one of the options:

- Type **RunningSequence**
- Type the shortcut **Run**

Description

The RunningSequence keyword is useful when there are several users in a session.

It allows you to list or disable all running sequences in a show file.

Important:
The numbers of the running sequences are not equal to the numbers in the sequence pool.

Syntax

[Off] RunningSequence [Number]

Example

- To disable the first sequence that is currently running, type:

```
User name[Fixture]> Off RunningSequence 1
```

- To list all running sequences, change destination to the running sequences first:

```
User name[Fixture]> ChangeDestination RunningSequence
```

Result:

```
User name@Temp/RunningSequences>
```

- To list all running sequences, type:

```
User name@Temp/RunningSequences> List
```

8.2.249. SaveShow Keyword

To enter the SaveShow keyword in the command line, use one of the options:

- Type **SaveShow**
- Type the shortcut **Sa**

Description

The SaveShow keyword is a function keyword used to save the current show file.

If you do not enter a new show name, the show will be saved using the name of the current show.

Important:
If there is already a show file with the same show name, the console overwrites the existing show file.

To learn more about the folder structure of shows, demo shows, and backups, read the **Show File Handling topic**.

Syntax

SaveShow "Showname"

Examples

- To save the show as "Rhapsody", type:

```
User name[Fixture]> SaveShow "Rhapsody"
```

- To save the open show, type:

```
User name[Fixture]> SaveShow
```

- To save a show file with the **/Enumerate option**, type:

```
User name[Fixture]> SaveShow /Enumerate
```

8.2.250. ScreenConfig Keyword

Important:
It is possible to create screen configurations per user profile. However, every user of the same user profile can use a different screen configuration.
Per default every user profile has 2 screen configurations:
-Default
-3D

To enter the ScreenConfig keyword in the command line, use one of the options:

- Type **ScreenConfig**
- Type the shortcut **ScreenConf**

Description

The ScreeConfiguration keyword is an object keyword that addresses the different screen configurations of a user profile. For example, one setup for operating the console, and a second setup for a dedicated onPC station.

Syntax

[Function] ScreenConfig [ID/"Name"] ["Property"] ["Property Value"]

ScreenConfig [ID/"Name"]

Examples

- To store a new screen configuration called "Average Joe", type:

```
User name[Fixture]> Store ScreenConfig 3 "Average Joe"
```

- To enter screen configuration 2, type:

```
User name[Fixture]> ScreenConfig 2
```

- To assign the screen configuration "Average Joe" to ViewButton 1.6, type:

```
User name[Fixture]> Assign ScreenConfig "Average Joe" At ViewButton 1.6.
```

For more information on ViewButtons see **Store and Recall Views**.

8.2.251. ScreenContent Keyword

To enter the ScreenContent keyword in the command line, use one of the options:

- Type **ScreenContent**
- Type the shortcut **Scre**

Description

The ScreeContent keyword is used to represent the windows of a display.

Syntax

[Function] ScreenContent

[Function] ScreenContent [ID].[ID] ["Property"] ["Property Value"]

Examples

- To delete all windows on all screens, type:

```
User name[Fixture]> Delete ScreenContent *.*
```

- To delete all windows on screen 1, type:

```
User name[Fixture]> Delete ScreenContent 1.*
```

- To set the width of the first window you created to 12 half columns in screen 1, type:

```
User name[Fixture]> Set ScreenContent 1.1 "W" "12"
```

8.2.252. Scribble Keyword

To enter the Scribble keyword into the command line, use one of the options:

- Press MA + X4 + X4 + X4
- Type **Scribble**
- Type the shortcut **Scr**

Description

Scribbles are little drawings that are used as visualizations instead of labels.

The scribble keyword is an object keyword that is used to access scribbles with a scribble ID.

It is also possible to label scribbles and address them by their label.

For more information about scribbles see **Scribbles**.

Syntax

[Function] Scribble [ID/"Name"] At [Object List]

Examples

- To edit scribble 1, type:

```
User name[Fixture]> Edit Scribble 1
```

The edit scribble pop-up opens.

- To assign scribble 1 to group 5, type:

```
User name[Fixture]> Assign Scribble 1 At Group 5
```

- To delete scribble 1, type:

```
User name[Fixture]> Delete Scribble 1
```

8.2.253. Seconds Keyword

To enter the Seconds keyword in the command line, use one of the options:

- Type **Seconds**
- Type the shortcut **Sec**

Description

The Seconds keyword is used to set the speed of a fixture selection using the unit seconds.

Syntax

[At] [Speed] Seconds [Value]

Example

- To set the speed layer to 20 seconds, type:

8.2.254. Select Keyword

To enter the Select keyword in the command line, use one of the options:

- Press Select
- Type **Select**
- Type the shortcut **Sel**

Description

The Select keyword selects objects as default objects.

Every selected object is indicated by a yellow frame.

- The **selected sequence** is the target for all sequence-related commands, e.g., **Store Cue 4**.
- The **selected sequence** is displayed in the Sequence Sheet, if **Link Selected** is enabeld.
- The **selected camera** is displayed in the 3D View, if the **Camera Selection** is set to **Link Selected**.

Syntax

Select [Executor Object]

Select Camera [ID]

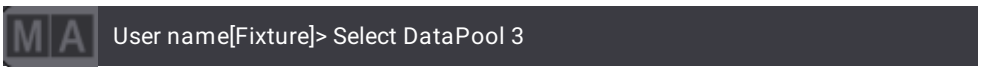
Example

- To select the sequence assigned on executor 105, type:

- To select page 2, type:

- To select world 1, type:

- To select data pool 3, type:



8.2.255. Selection Keyword

To enter the Selection keyword in the command line, use one of the options:

- Press Fixture Fixture
- Type **Selection**
- Type the shortcut **Sel**

Description

The Selection keyword is an object keyword used to represent the current selection of fixtures in the programmer.

Syntax

Selection

Example

- To disable all programmer values of the current fixture selection, type:

8.2.256. SelFix Keyword

To enter the **SelFix** keyword in the command line, use one of the options:

- Press SelFix
- Type **SelFix**
- Type the shortcut **SelF**

Description

The SelFix keyword is a function keyword used to create selections of fixtures in the programmer.

If only fixtures are selected, the SelFix keyword adds additional fixtures to the selection.

If fixtures are selected and activated in the programmer, the SelFix keyword replaces the selection by the SelFix selection.

If the exact same SelFix command is successively used:

- using it the second time activates all attributes of the selected fixtures in the programmer
- using it the third time deactivates all attributes of the selected fixtures in the programmer

SelFix is the default function of most objects, for example, fixture or group or preset.

To clear the selection, press Clear.

Syntax

SelFix [Object List]

Examples

- To select all fixtures or channels stored in a sequence of the executor 101, type:

- To select all fixtures stored in dimmer preset 1.1, type:

8.2.257. Sequence Keyword

To enter the Sequence keyword in the command line, use one of the options:

- Press Sequ
- Type **Sequence**
- Type **Se**

Description

The sequence keyword is an object keyword that is used to select sequences in the sequence pool.

Syntax

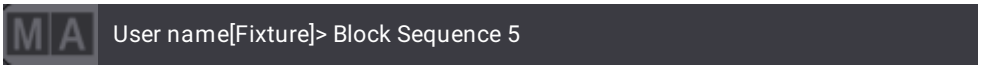
Sequence [ID]

Sequence [Sequence pool].[ID]

Examples

- To select sequence 5 in the sequence pool, type:

- To block all cues in sequence 5, type:



8.2.258. SendOSC Keyword

To enter the **SendOSC** keyword in the command line, use one of the options:

- Type **SendOSC**
- Type the shortcut **Sen**

Description

The SendOSC keyword is a command keyword used to send an OSC command.

To learn more about OSC, read the **Remote In and Out topic**.

Syntax

SendOSC [ID] "/[OSCAddress],[OSC Type],[Value]"

The supported types are:

- Int(i)
- Float(f)
- Blob(b)
- String(s)
- True(T)
- False(F)
- Null(N)
- Impulse(I)
- Timetag(t)

It is not necessary to set a value (Payload) for:

- True
- False
- Null
- Impulse
- Timetag

Hint:
When using the OSC types True, False, Nil/Null, Impulse and Timetag it is not necessary to enter a value.

Hint:
Several values can be sent at once when separating them with commas.

Important:
When addressing an executor, a page must be specified as well.

Instead of using page and executor numbers it is also possible to address them by name.
When addressing executor keys, a value of 0 will be interpreted as not pressed. Values greater 0 will be interpreted as button press.

If a prefix is specified for an OSCData entry, then this very prefix will be added to the sent string when using the OSCSend command.

Hint:
The supported OSC types to control faders, executor knobs, and buttons are: Integer32, Float32, True, False and Nil/Null.
A True will be interpreted as 1, while a False will be interpreted as 0.

Hint:
The addresses defined for Page, Fader, ExecutorKnob, and Key are case-sensitive.

Examples

- To send an OSC command using the first configuration in the OSC menu with integer value 50 to fader 201 on page 1, type:

```
User name[Fixture]> SendOSC 1 "/Page1/Fader201,i,50"
```

- To send an OSC command using the first configuration in the OSC menu with integer value 100 to fader 201 on page 1 and a fade time of 5s, type:

```
User name[Fixture]> SendOSC 1 "/Page1/Fader201,ii,100,5"
```

- To send commands via OSC to a second grandMA3 station, the OSC address /cmd can be used. To store cue 1 via OSC, type:

```
User name[Fixture]> SendOSC 1 "/cmd,s,Store Cue 1"
```

8.2.259. Set Keyword

To enter the Set keyword in the command line, use one of the options:

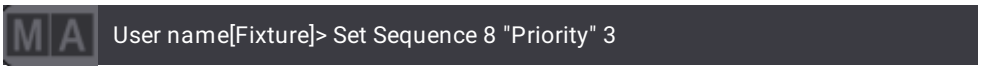
- Press Assign Assign Assign
- Type **Set**

Description

The Set keyword sets values to object properties.

Examples

- To set sequence 8 to priority HTP, type:



8.2.260. SetGlobalVar Keyword

To enter the SetGlobalVar keyword in the command line, use one of the options:

- Type **SetGlobalVar**
- Type the shortcut **Setg**

Description

The SetGlobalVar keyword is used to set global variables in a show.

Syntax

SetGlobalVar "name of variable" "numeric value"

SetGlobalVar "name of variable" "text"

Example

- To set the global variable Urban Blues 3, type:

8.2.261. SetUserVar Keyword

To enter the SetUserVar keyword in the command line, use one of the options:

- Type **SetUserVar**
- Type the shortcut **Setu**

Description

The SetUserVar keyword is used to set user-specific variables.

Syntax

SetUserVar "name of variable" "numeric value"

SetUserVar "name of variable" "text"

Example

- To set the user variable Green to the value 5, type:

8.2.262. Shuffle Keyword

To enter the Shuffle keyword in the command line, use one of the options:

- Press SelFix SelFix
- Type **Shuffle**
- Type the shortcut **Shuf**

Description

Shuffle is a command keyword used to shuffle the order of the fixture selection. Shuffle is part of the MAtricks toolset.

For more information, read the **MAtricks and Shuffle topic**.

Syntax

MAtricks [Axis] [Value]

MAtricks [Axis] +

Hint:
The plus is a replacement for the value. You can either use plus/minus or a value.

Examples

To shuffle the current selection on the y-axis, type:

```
User name[Fixture]> MAtricks "YShuffle" +
```

To shuffle the current selection on the z-axis, type:

```
User name[Fixture]> MAtricks "ZShuffle" +
```

It is also possible to set a certain value to any of the three shuffle settings. To set the shuffle to 4 for the x-axis, type:

```
User name[Fixture]> MAtricks "XShuffle" 4
```

Hint:
When deactivating or resetting the MAtricks, the original selection order will be restored.

8.2.263. Shutdown Keyword

To enter the Shutdown keyword in the command line, use one of these options:

- Type **Shutdown**
- Type the shortcut **Sh**

Description

The shutdown keyword switches off the grandMA3 console or closes the grandMA3 onPC.

A shutdown requires a confirmation in the local station.

The shutdown can be canceled within 10 seconds using a remote station.

This keyword belongs to a session-related group of keywords.

For more information see:

- **Console keyword**
- **Node keyword**
- **OnPC keyword**
- **PU keyword**
- **Station keyword**
- **Extension keyword**
- **Join keyword**
- **Leave keyword**
- **Invite keyword**
- **Dismiss keyword**
- **Reboot keyword**
- **Restart keyword**

Syntax

Shutdown [Device Type] [Station List]

Shutdown IP [IP]

Shutdown /[Option]

Hint:
The device types are: **Console**, **Node**, **onPC**, **PU**, **Station**, **Extension**.

Options

The shutdown keyword has the following options:

Option	Option Shortcut	Description
noconfirm	nc	Suppresses shutdown confirmation pop-up.
noautoclose	noa	Suppresses shutdown timer.

Examples

- To shut down the current station with a timer, type:

- To shut down the current station without confirmation, type:

- To shut down the current station without a timer, type:

- To shut down the station with the IP address 192.168.0.4, type:

8.2.264. SnapDelay Keyword

To enter the SnapDelay keyword in the command line, use one of the options:

- Type **SnapDelay**
- Type the shortcut **Sn**

Description

The SnapDelay keyword sets a snap time.

The snap time is a delay time for attributes that do not fade, for example a gobo wheel or a color wheel.

Syntax

SnapDelay [Value]

Examples

- To set the snap time of the current cue of the selected sequence to 4 seconds, type:

8.2.265. SoftwareImport Keyword

To enter the SoftwareImport keyword in the command line, use one of the options:

- Type **SoftwareImport**
- Type the shortcut **SoftwareI**

Description

The SoftwareImport keyword is a function keyword which is used to import installation packages of a grandMA3 installer permanently to the hard drive of your grandMA3 console or the grandMA3 onPC using a given path.

Syntax

SoftwareImport "release_type_vx.x.x.x.xml;Path/to/the/location/of/the/installer"

Example

- To import the installation packages of the grandMA3 stick v1.6.3.7 on the USB drive "Software" that is recognized as drive D on a Windows computer, and where the files are located within the ma folder, type:

```
User name[Fixture]> SoftwareImport
"release_stick_v1.6.3.7.xml;D:/Software/ma"
```

8.2.266. SoftwareUpdate Keyword

To enter the SoftwareUpdate keyword in the command line, use one of the options:

- Type **SoftwareUpdate**
- Type the shortcut **SoftwareU**

Description

The UpdateSoftware keyword is a function keyword used to update the software of every MA device or program in the network.

For more information on how to update the software and requirements see **Update grandMA3 Consoles**.

Syntax

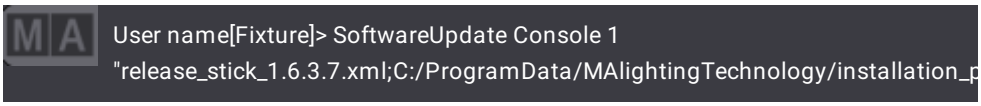
SoftwareUpdate [StationType] [ID/"Name"]
"release_type_x.y.z.a.xml;/Path/to/MALightingTechnology/installation_packages"

Example

Requirement:

1. The grandMA3 onPC runs on Windows®
2. Copy the files of the ma folder of the grandMA3_stick_v1.6.3.7.zip file to C:\ProgramData\MALightingTechnology\installation_packages

- To update the first console within your network to grandMA3 v1.6.3.7, type:



8.2.267. SoundChannel Keyword

To enter the SoundChannel keyword in the command line, use one of the options:

- Type **SoundChannel**
- Type the shortcut **SoundC**

Description

The SoundChannel keyword represents sound codes in the attribute calculators.

Syntax

[Attribute List] [At] SoundChannel [ID/"Name"]

Example

- To set the value for Pan of the selected fixtures to Band1, type:

For more information about using sound input, see the **Sound Window** topic.

8.2.268. Solo Keyword

To enter the Solo keyword in the command line, use one of the options:

- Press Solo
- Type **Solo**
- Type **So**

Description

The Solo keyword is a function keyword used to set the intensity values of fixtures to zero that are not selected.

Only the fixtures that are selected generate visible output on stage.

If Solo is used standalone, it toggles between Solo On and Solo Off.

To manually fade from the current value to solo and vice versa, use the Grand Solo.

Syntax

Solo

Solo On

Solo Off

Example

- To enable Solo, type:

8.2.269. SpecialExecutor Keyword

To enter the SpecialExecutor keyword in the command line, use one of the options:

- Press MA + X16|Exec + X16|Exec
- Type **SpecialExecutor**
- Or type the shortcut **SE**

Description

The SpecialExecutor keyword addresses special executors.

For more information on the configuration see **Executor Configurations**.

Syntax

[Function] SpecialExecutor [ID/"Name"]

Examples

- To open the menu **Special Executor Configuration**, type:

- To assign the functionality of grand master to SpecialExecutor 2, type:

8.2.270. Speed Keyword

To enter the Speed keyword in the command line, type **Speed.**

Description

The Speed keyword is used to set the speed of phasers.

For more information see **Phasers**.

Requirement:
To set values for the speed, there must be at least 2 steps in the programmer.

Syntax

[At] Speed [Value]

Examples

- To set the selected layer to speed, type:

- To set the speed of the dimmer to 70 hertz, type:

8.2.271. Speed1 Keyword

To enter the Speed1 keyword in the command line, use one of the options:

- Type **Speed1**
- Type the shortcut **Spee**

Description

The Speed1 keyword is used to reset the speed to 60 BPM.

Syntax

Speed1 [Executor/Page] [Value]

Example

- To reset the speed master of the sequence that is assigned to executor 201, type:

8.2.272. Stage Keyword

To enter the Stage keyword in the command line, use one of the options:

- Type **Stage**
- Type the shortcut **Sta**

Description

The Stage keyword addresses stages in the 3D space.

Syntax

[Function] Stage [ID/"Name"] ["Property"] ["Value"]

Example

- To store a new stage, type:

- To list all stages in the command line history, type:

- To set the center of stage 1 to 5 meters in the x direction of the Cartesian coordinate system, type:

8.2.273. Station Keyword

To enter the Station keyword in the command line, use one of the options:

- Type **Station**
- Type the shortcut **Stat**

Description

The Station keyword is an object keyword used to display all stations in the network, but you can also **invite** stations into your session or **dismiss** them.

This keyword belongs to a session-related group of keywords.

For more information see:

- **Console keyword**
- **Node keyword**
- **onPC keyword**
- **PU keyword**
- **Extension keyword**
- **Join keyword**
- **Leave keyword**
- **Invite keyword**
- **Dismiss keyword**
- **Reboot keyword**
- **Restart keyword**
- **Shutdown keyword**

Syntax

Station [ID]

Station [Name]

Examples

- To list all stations that are currently in the same network, type:

- To invite station "FOH3" into the session, type:

8.2.274. Step Keyword

To enter the Step keyword in the command line, use one of the options:

- Press MA + X5
- Type **Step**

Description

The Step keyword is a command keyword used to define the step of the At command.

Syntax

[Function] Step [ID]

Example

Requirement:

- Step 1 has to contain information

- To create the next step in the programmer, type:

8.2.275. Stomp Keyword

To enter the Stomp keyword in the command line, use one of the options:

- Press Stomp
- Type **Stomp**
- Type the shortcut **Stom**

Description

The Stomp keyword is a command keyword used to stop a running phaser.

Stomp now knocks in the last used single-step value for the absolute and/or the relative layer of the attribute. This can be a value from the programmer or from a playback. If no single-step value is found it uses the default value.

Syntax

Stomp [Object]

Examples

- To stop the phasers in the selected fixtures, enter:

- Stomp selection:

1. Select the fixtures using any desired method (for example, groups, selection tools in the layout or 3D windows, or the **SelFix** command).
2. Press Stomp to enter the **Stomp** keyword in the command line.
3. Press the Position feature group button in the encoder bar.

8.2.278. Store Keyword

To enter the **Store** keyword in the command line, use one of the options:

- Press Store
- Type **Store**
- Type the shortcut **S**

Description

The Store keyword is a function keyword used to store objects in the show file.

If no object type or destination is given, the object type **Cue** will be used for the selected sequence.

Syntax

Store [Object List] ("Name") (/Option) "Option Value"

Options

The store keyword has the following options.

Option	Option Shortcut	Option Value	Description
CreateSecondCue	Cr	No option value	Stores a cue with the next whole cue number if the sequence has only one cue.
Merge	m	No option value	Merges the new values into the existing values. New values have a higher priority and will overwrite existing values. If no new value is given, the values of the existing cue will stay.
Noconfirm	nc	No option value	Suppresses store confirmation pop-up.
Overwrite	o	No option value	Removes stored values in the cue and stores the new values.
Remove	No shortcut	No option value	Removes stored values in attributes with active programmer values.
Screen	No shortcut	1...6	Stores a view in the view pool with the selected screen.

To learn more about store options, read the **Store Options and Store Preferences** topic.

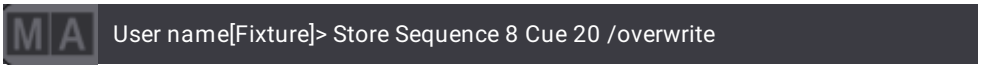
Examples

- To store cue 2 in the selected sequence, type:

For more information see **Store Cues**.

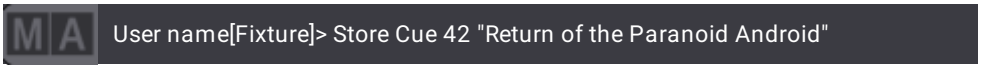
- To store the programmer values as cue 1 through cue 10 and cue 20 through cue 30, type:

- To store the programmer values on the existing sequence 8 cue 20 and use the option overwrite, type:

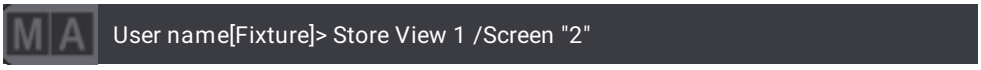


- To shorten the previous example, type:

- To store the programmer values as cue 42 of the selected sequence and directly label it, type:



- To store screen 2 on view pool button 1, type:



8.2.279. Temp Keyword

To enter the Temp keyword in the command line, use one of the options:

- Press MA + Go+
- Type **Temp**
- Type the shortcut **Te**

Description

The Temp keyword is a function keyword that enables an executor as long as you hold the executor key.
The Temp keyword follows cue timing, off timing, and the position of the master fader on the executor.

- To disable the executor, release the executor key.

Syntax

Temp [Object List]

Temp On [Object List]

Temp Off [Object List]

Examples

- To temporarily enable executor 104, type:

- To temporarily disable executor 104, type:

8.2.280. Texture Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.281. Thru Keyword

To enter the Thru keyword in the command line, use one of the options:

- Press Thru
- Type **Thru**
- Type the shortcut **T**

Description

The Thru keyword indicates the range of objects or values.

If the beginning or the end of the range is not defined, the objects or values that were used last are used again.

Syntax

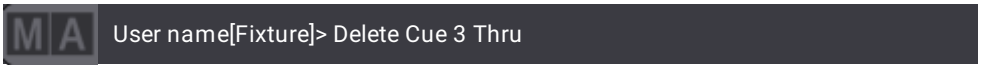
[Beginning of Range] Thru [End of Range]

Examples

- To select fixture 3 through 6, type:

- To select all fixtures starting with the first fixture and ending with fixture 10, type:

- To delete all cues, beginning with cue 3 in the selected sequence, type:



- To select all fixtures in the fixture sheet of the current world, type:

- To disable all executors in the current page, type:

8.2.282. Time Keyword

To enter the Time keyword in the command line, type **Time.**

Description

The Time keyword is used to toggle the time overwrite function on sequences.

For more information see **Cue Timing topic**.

See the **FaderTime keyword** to learn how to set the time.

Requirement:
To toggle the time, there must be a sequence.

Syntax

Time (Value) (Executor)

Time (Value) (Sequence Sequence_ID)

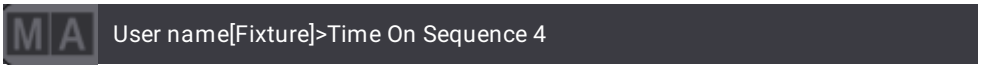
If the sequence ID is not defined then the selected sequence is affected.

Value options are **On** or **Off**.

Examples

- To toggle the time function on the selected sequence, type:

- To enable the time overwrite for sequence 4, type:



8.2.283. Timecode Keyword

To enter the Timecode keyword in the command line, use one of the options:

- Press MA + X6
- Type **Timecode**
- Type the shortcut **TC**

Description

The Timecode keyword is an object keyword used to select the timecode by default.

Using the Timecode keyword you can:

- store
- play (go)
- record
- edit
- label
- set properties of
- rewind (top)

timecode shows.

For more information see **Timecode**.

Syntax

[Function] Timecode [ID/"Name"]

Set Timecode [ID/"Name"] ["Property"] ["Property Value"]

Property

The following table displays the properties that can be assigned using the **Set Keyword**.

Property	Property Value	Description
Name	"Timecode show name"	Set the name of timecode show.
Time		Set position of the time cursor.
Duration	0s to 255h59m58.96s	Set the entire length of timecode show.
Offset	0s to 255h59m58.96s	To move the entire timecode show forward, set an offset in the timecode show.
Loop Mode	Loop Pause Off	Basic settings for Loop and how to pause or stop.
Loop Count	"Endless Repeat" (0), "No Repeat" (1), 2..1000	**Only for timecode shows syncing to the internal clock.** Set if the timecode show runs: Endless (Endless Repeat) Once and stop (No Repeat) A specific number of times (number between 2 and 1000)
TimeMarkers		Timemarkers are used to select the track. They will be stored.
TCSlot	"Internal" (-2), "Link Selected" (-1), "Slot1" (0), "Slot2" (1), "Slot3" (2), etc.	Set the timecode show to a timecode slot.
AutoStart (only available with external timecode slot)	"Yes", "No"	**Only available when syncing to an external source.** If a timecode signal is received, the timecode show switches from the off mode into the play mode.
AutoStop	"Yes", "No"	**Only available when syncing to an external source.** If a timecode signal is received, the timecode show switches from the play mode into the off mode.
User Bits	0 .. FFFFFFFF, 0 .. 4294967296	**Only available when syncing to an external source.** To transmit several kinds of information, for example, a second Timecode Stream, set user bits in hex or decimal. So several incoming Timecode Streams can be discerned.

Examples

- To set timecode show "Intro" to a duration of 55 seconds, type:

```
User name[Fixture]> Set Timecode "Intro" "Duration" "55"
```

- To store a new timecode show called "Napalm Skies" in the timecode pool, type:

- To rename timecode show "Intro" to "Prelude", type:

- To start the timecode show "Prelude", type:

- To rewind the timecode show "Prelude", type:

- To label timecode show 3 in the timecode pool, type:

The label pop-up opens and you can now label the timecode show.

8.2.284. TimecodeSlot Keyword

To enter the TimecodeSlot keyword in the command line, use one of the options:

- Press MA + X6 + X6
- Type **TimecodeSlot**
- Type **Timecodes**

Description

The TimecodeSlot keyword is an object keyword used to address the timecode slots.

For more information, see **What are Timecode Slots**.

Syntax

[Function] TimecodeSlot [ID/"Name"]

Examples

- To select timecode slot 4, type:

- To start the timecode generator of TimecodeSlot 2, type:

- To stop the timecode generator of timecode slot 2, type:

8.2.285. Toggle Keyword

To enter the Toggle keyword in the command line, use one of the options:

- Press and hold MA and press Go+ Go+
- Type **Toggle**
- Type the shortcut **To**

Description

The Toggle keyword is a function and a helping keyword that behaves as an On or Off keyword, depending on the on/off status of the object or function/mode it is applied to.
If object or function/mode is on, it disables it. If object or function/mode is off, it enables it.

For more information see the **On** and **Off Keywords**.

Syntax

Toggle [Object List]

Examples

- To disable the running executor 104, type:

8.2.286. TopUp Keyword

To enter the TopUp keyword in the command line, use one of the options:

- Press Set + Up
- Type **TopUp**
- Type the shortcut **Topu**

Description

The TopUp keyword is a function keyword used to select the highest level in the hierarchy of multi-instance fixtures whenever subfixtures are selected.

Syntax

TopUp

Example

- To jump to the main instance of the fixture within which subfixtures are selected, type:

8.2.287. Top Keyword

To enter the Top keyword in the command line, use one of the options:

- Press MA + Go-
- Type **Top**

Description

The Top keyword is a function keyword used to jump to the beginning of a cue list or to set a timecode marker at the beginning of a timecode show.

Syntax

Top [Executor List]

Top [Timecode List]

Examples

- To jump to the top of the cue list on executor 105, type:

- To jump to the top of the cue list on executor 105 in 3 seconds, type:

- To jump to the top of the Timecode show 1, type:

8.2.288. Transition Keyword

To enter the Transition keyword in the command line, use one of the options:

- Type **Transition**
- Type the shortcut **Tr**

Description

The Transition keyword is used to determine the length of the passage in a step.

For more information see **Phasers**.

Syntax

Transition

[At] Transition [Value]

Examples

- To set the layer to transition, type:

- To shorten the transition of a step to a half, type:

8.2.289. Type Keyword

To enter the Type keyword in the command line, use one of the options:

- Type **Type**
- Type the shortcut **Ty**

Description

The Type keyword is used to address children of an object.

Syntax

[Function] [Object] [Number/"Name"] Type "Name of Type"

Example

- To store cue part 1 in a newly created sequence without specifying a cue number, type:

Hint:
To circumvent the pop-up "Select type of object", press an executor immediately after typing this command.

8.2.290. UIChannel Keyword

To enter the UIChannel keyword in the command line, use one of the options:

- Type **UIChannel**
- Type the shortcut **Ui**

Description

The UIChannel is an object keyword that represents an attribute in the fixture sheet.

Important:
Depending on the fixtures that are patched in the show file, the attributes that are represented in a UIChannel may differ.

Syntax

[List] UIChannel

UIChannel [Number of UIChannel] At [Value]

Examples

- To list all the UIChannels in the command line, type:

```
User name[Fixture]> List UIChannel
```

- To set the UIChannel 9 to a value of 42, type:

```
User name[Fixture]> UIChannel 9 At 42
```

8.2.291. UiGridSelection Keyword

To enter the UiGridSelection keyword in the command line, use one of the options:

- Type **UiGridSelection**
- Type the shortcut **UiG**

Description

UiGridSelection keyword is used by the system to hold information about what objects are selected in a grid window. Grid windows include sheets. It does **not** include the Selection Grid window.

The keyword is primarily used internally by the system, but the selection can be used for normal operations that are done on the selected objects.

Important:
The grid with the selection needs to have the focus for the UiGridSelection command to work.

Syntax

[Function] UiGridSelection

Example

Requirement:

- Some cues in a sequence.
- A macro with a **Copy UiGridSelection** command. This is important to be able to execute the command without moving the focus.

- To copy a selection of the cues to cue 41, using the UiGridCommand do the following steps:

  1. Select the desired cues in the sequence sheet.
  2. Run the macro using the command keys (Important to use the keys to keep the focus in the cue selection).
  3. Now the following command can be entered and executed in the command line:

8.2.292. Unblock Keyword

Important:
To block parameters, use the **Block keyword**.

To enter the Unblock keyword in the command line, use one of the options:

- Type **Unblock**
- Type **UN**
- Type the shortcut **UB**

Description

Unblock is a function to convert blocked values in cues into tracking values.

If the object list does not contain any references to any cues, the Unblock function is applied to the selected sequence.

If unblock does not contain any selection list filter, all fixtures will be used.

If unblock does not contain any attribute list filter, all attributes will be used.

Syntax

Unblock [Object List] If [Selection List] [Attribute List] EndIf

Examples

- To unblock all parameters of the selected sequence, type:

```
User name[Fixture]> Unblock
```

- To unblock pan and tilt of fixture 4 in cue 5 of the selected sequence, type:

```
User name[Fixture]> Unblock Cue 5 If Fixture 4 Feature "Position" EndIf
```

8.2.293. Unlock Keyword

Hint:
A padlock indicates a locked object.

Hint:
To lock objects, use the **Lock Keyword**.

To enter the Unlock keyword in the command line, use one of the options:

- Type **Unlock**
- Type the shortcut **UL**

Description

The Unlock keyword is a function keyword used to unlock objects that were previously locked.

Syntax

Unlock [Object List]

Examples

- To unlock cue 3 to edit the cue, type:

```
User name[Fixture]> Unlock Cue 3
```

- To unlock world 3 to edit the world, type:

```
User name[Fixture]> Unlock World 3
```

- To unlock the macro line 2 of the macro 1, type:

```
User name[Fixture]> UnLock Macro 1.2
```

8.2.294. Unpark Keyword

To enter the Unpark keyword in the command line, use one of the options:

- Press Go+ Go+
- Type **Unpark**
- Type **Unp**

Description

The Unpark keyword is a command keyword used to release a parked DMX channel or universe.

Syntax

Unpark [Object]

Examples

- To unpark fixture 1 with all its attributes, type:

- To unpark the current selection, type:

- To unpark DMX universe 2, type:

If a universe is unparked, the blue **P** icon disappears in the universe pool.

8.2.295. Unpress Keyword

To enter the Unpress keyword in the command line, use one of the options:

- Type **Unpress**
- Type the shortcut **Unp**

Description

The Unpress keyword is used to simulate the unpressed state of a key.

It is possible to assign this state to macros.

For more information see **Macros**.

Syntax

[Function] Unpress [Executor List]

Example

- To disable Flash on the executor 203, type:

8.2.296. Up Keyword

To enter the Up keyword in the command line, use one of the options:

- Press Up
- Type **Up**

Description

The Up keyword is a function keyword used to navigate upward in the fixture structure.

Syntax

Up

Example

- To go back to the main fixture in the fixture sheet in an upward manner, type:

8.2.297. Update Keyword

To enter the Update keyword in the command line, use one of the options:

- Press Update
- Type **Update**
- Type the shortcut **U**

Description

The Update keyword is a function keyword used to update values in their source objects.

Syntax

Update [Object List]

Example

- To update Preset 4.1, type:

8.2.298. UpdateContent Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.299. User1 Keyword

Important:
The **User1** keyword can only be used on the grandMA3 compact, the grandMA3 compact XT, and the grandMA3 onPC command wing where the user key is located on the right of Special Executor 1 and 2.

Hint:
The **User1** keyword must be used as one word.

To enter the **User1** keyword in the command line, use one of the options:

- Press U1
- Type **User1**

Description

The **User1** keyword is a function keyword that toggles between the encoder bar and the Xkeys bar.

Syntax

User1

Example

- To toggle the encoder bar to the Xkeys bar, type:

```
User name[Fixture]> User1
```

8.2.300. User2 Keyword

Important:
The **User2** keyword can only be used on the grandMA3 compact, the grandMA3 compact XT, and the grandMA3 onPC command wing where the user key is located on the right of Special Executor 1 and 2.

Hint:
The **User2** keyword must be used as one word.

To enter the **User2** keyword in the command line, use one of the options:

- Press U2
- Type **User2**

Description

The **User2** keyword is a function keyword that toggles between the encoder bar and the executor bar.

Syntax

User2

Example

- To toggle the encoder bar to the executor bar, type:

8.2.301. User Keyword

To enter the User keyword in the command line, use one of the options:

- Type **User**
- Type the shortcut **Us**

Description

The User keyword is used for login or to change the user.

The User keyword displays the following details regarding the user in the Command Line History:

- Number
- Name
- Scribble
- Appearance
- Password
- Profile
- Screen configuration
- Rights
- Language
- Keyboard

Syntax

User [ID]

Example

- To list the details of all users, type:

8.2.302. UserProfile Keyword

To enter the UserProfile keyword in the command line, use one of the options:

- Type **UserProfile**
- Type **Userp**
- Type the shortcut **UPR**

Description

The keyword UserProfile is an object keyword used to display the user profiles in the command line history.

- Number
- Name
- DMX readout
- Normal value
- Value readout
- Speed readout
- Preset readout
- Resolution of the dimmer wheel
- Mode of the dimmer wheel
- Precise edit
- Single step
- Sync
- Preview
- Screen encoder
- Time key target
- Programmer part
- TC slot
- Overlay fade

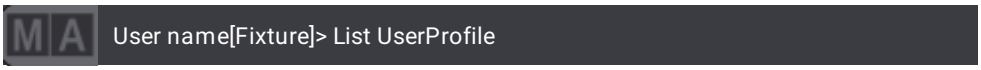
For more information see **Create User**.

Syntax

UserProfile [ID]

Example

- To list all user profiles that are available, type:



8.2.303. Video Keyword

To enter the Video keyword in the command line, use one of the options:

- Type **Video**
- Type the shortcut **Vid**

Description

The Video keyword is an object keyword. It is used to manage videos using the command line.

Syntax

Video [ID/Name]

[Function] Video [ID/Name]

Examples

- To edit video 1 in the video pool, type:

- To delete video 1 in the video pool, type:

8.2.304. Version Keyword

To enter the Version keyword in the command line, use one of the options:

- Type **Version**
- Type the shortcut **Ve**

Description

The Version keyword is a function keyword used to grant access to details of the version in the console or in the onPC software.

It displays the following details in the command line history:

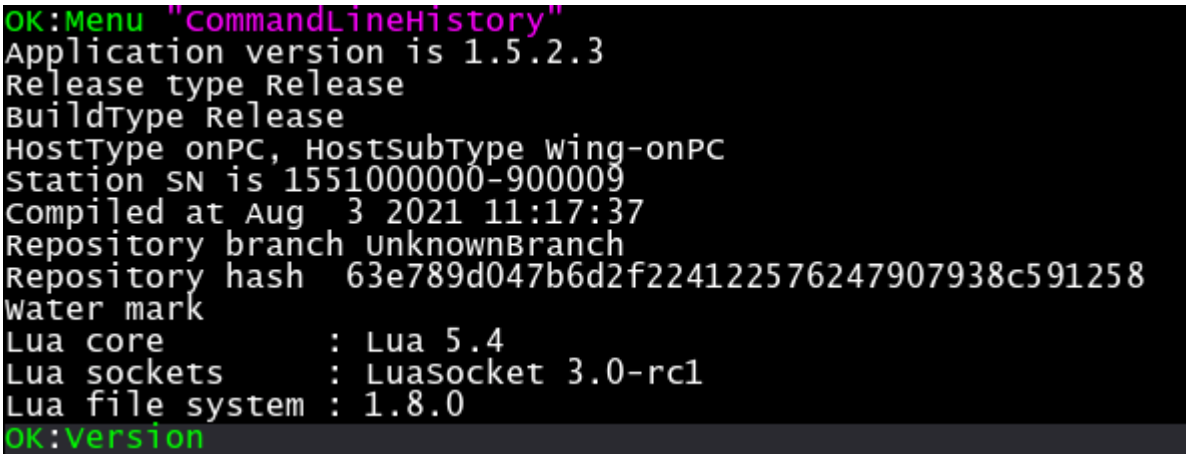
- Version number
- Release type
- Build type
- Code type
- Host type
- Host sub type
- Station serial number
- Date of compilation
- Repository branch
- Repository hash
- Version number of Lua
- Version number of Lua sockets
- Version number of Lua file system

Syntax

Version

Example

- To view the details of the version, type:



Version details in the command line history window

The version details are displayed in the command line history window.

8.2.305. View Keyword

To enter the View keyword in the command line, use one of the options:

- Press MA + X7
- Type **View**
- Type the shortcut **V**

Description

The View keyword calls views on a screen.

The option **/Screen** selects:

- On which screen the view will be called

- or -

- Which screens will be stored in a view

Syntax

View [ID]

View [ID] /Screen "ID"

View [ID] /Option "Option Value"

Options

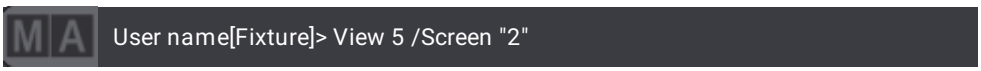
The following table displays the options that can be assigned using the **Set keyword**.

Option	Option Value
Name	View name, e.g., "Stage External"

Examples

- To call view 2 from the view pool on the screen where it is stored in, type:

- T call view 5 on screen 2, type:



- To store the content of screen 2, 3, and 4 in the view object 5, type:

8.2.306. ViewButton Keyword

To enter the ViewButton keyword in the command line, use one of the options:

- Press MA + X7|View + X7|View
- Type **ViewButton**
- Or type the shortcut **ViewB**

Description

The ViewButton keyword is an object keyword used to call the object assigned on the view button.

The call function works only if the assigned object supports it.

For more information see the **Call Keyword**.

Syntax

[Function] ViewButton Screen.ID ["Value"]

Examples

- To assign the user "Eisenstein" to the ViewButton 1.1, type:

```
User name[Fixture]> Assign User "Eisenstein" At ViewButton 1.1
```

- To delete the function assigned on view button 4 view page 1, type:

```
User name[Fixture]> Delete ViewButton 1.4
```

- To label the assigned view button 5 to view page 2, type:

```
User name[Fixture]> Label ViewButton 2.5 "Layout"
```

- To store a view on X1 key, type:

```
User name[Fixture]> Store ViewButton 11.1
```

8.2.307. Width Keyword

To enter the Width keyword in the command line, use one of the options:

- Type **Width**
- Type the shortcut **Wi**

Description

The Width keyword is used to determine the width incorporating transition of the entire step.

For more information see **Phasers**.

Syntax

Width

[At] Width [Value]

Examples

- To set the layer to width, type:

- To create a PWM effect, type:

8.2.308. World Keyword

To enter the World keyword in the command line, use one of the options:

- Press Group Group
- Type **World**
- Or type the shortcut **W**

Description

The World keyword is an object keyword used to call worlds along with their filters, and limit the access to the parameters in the world.

By default, World 1 is always locked and can neither be edited nor deleted. World 1 includes all parameters (fixtures and attributes) of the show.

Syntax

[Function] World [ID/"Name"]

Examples

- To call world 3, type:

- To label the world 3 "All Fixtures", type:

8.2.309. Xkeys Keyword

This is a reserved keyword.

Avoid using this keyword as a name. It might cause unexpected results.

8.2.310. Zero Keyword

To enter the Zero keyword in the command line, use one of the options:

- Press . .
- Type **Zero**
- Type the shortcut **Z**

Description

The keyword Zero is a helping keyword used to set the intensity of the dimmer to zero if fixtures or channels are selected.

Syntax

[Object List] Zero

Examples

- To set the intensity of the selected fixtures or channels to zero, type:

- To set the intensity of fixture 1 trough 10 to zero, type:

8.3. Option Keywords

Option keywords can be seen as criteria. These criteria are used in commands to temporarily determine their result. The result is always a single event - that is, it will not be set permanently.

To make it easier for you to understand how such option keywords work, we will use an everyday situation at a pizza place.

Example

- Pizza order

Let's say you are a regular at your favorite pizza place where you always order your pizza with Mozzarella topping. The owner of the place knows that you always eat this pizza. Whenever you call and say you are having a pizza they already arrange everything and the only thing you have to do is go and pick up your food.

But what if one day you decide to have a Basil topping instead?! Duh dum!

You call your pizza guy and explicitly say that you are having a Basil pizza this time. The next time around your pizza guy will stick to your "default Mozzarella topping", except you explicitly tell them that you would like to have a different topping this time.

Hint:
Mind you, you can always use your favorite dish in this example - burgers, kebab, steamed buns, sauerbraten, varenyky, borscht...

8.3.34. /Merge Option Keyword

To enter the **/Merge** option keyword in the command line, use one of these options:

- Type /**Merge**
- Type the shortcut **/m**

Description

The /Merge option keyword merges new values with existing values.

The most recent values have a higher priority than the preexisting and will effectively overwrite these.

Keywords that use the /Merge option keyword:

- **AutoCreate keyword**
- **Copy keyword**
- **Store keyword**

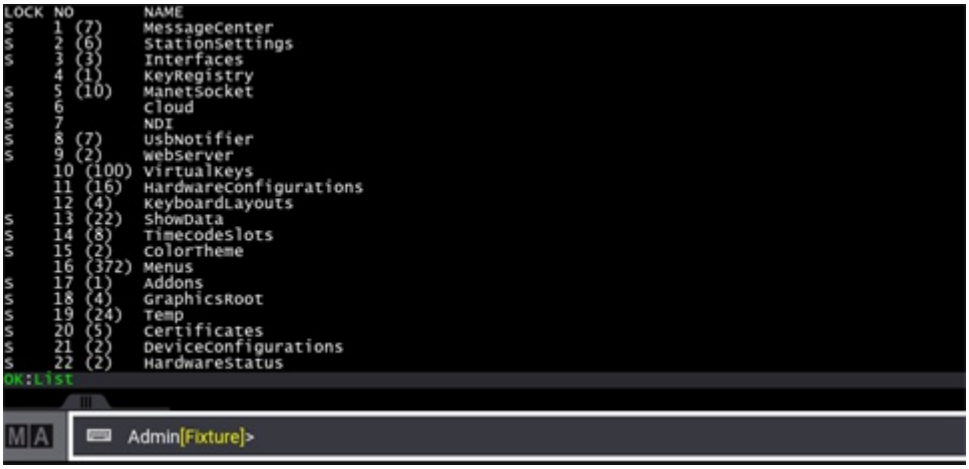
8.4. Extended Command Line Syntax Options

To control the input in the command line use the **List keyword** any time.

For more information about the command line and the Command Line History, read the **command line topic**.

To list all available subfolders, type:

```
User name [Fixture]> List
```

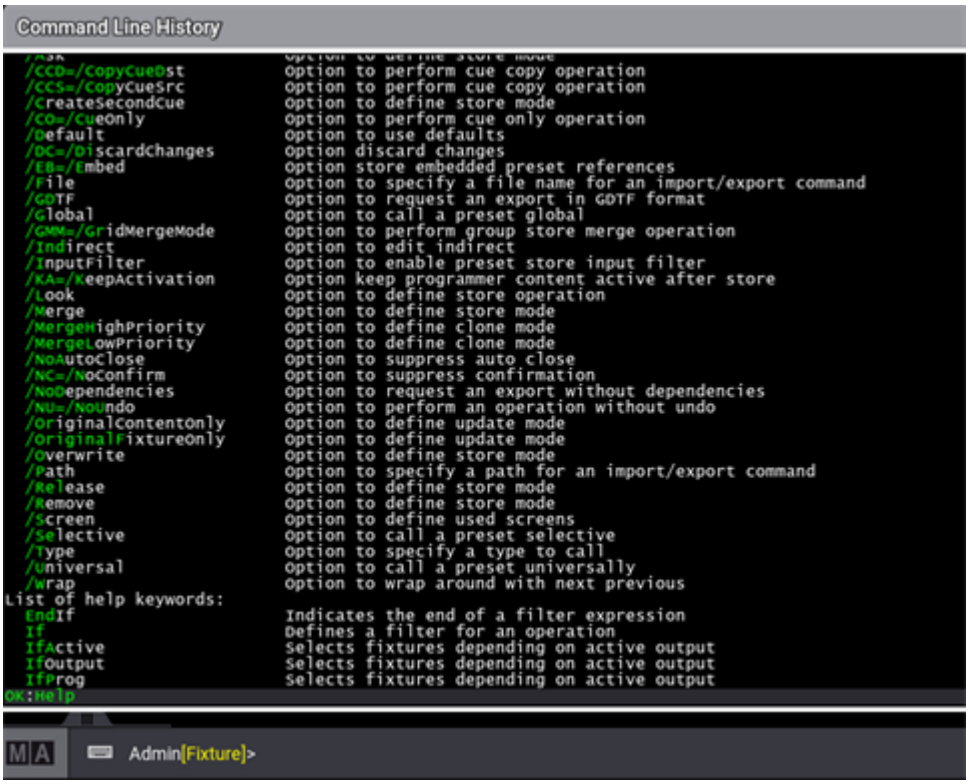


List view in Command Line History

Hint:
When the List command returns multiple objects of different types, the settings for those objects are not displayed. To view the settings of a specific object, include the name or number of the object with the List keyword. For more information, see the **List keyword** topic.

To list all available keywords, type:

```
User name [Fixture]> Help
```



Keyword list in Command Line History

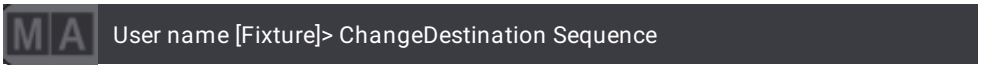
For more information about the help keyword, read the **help keyword topic**.

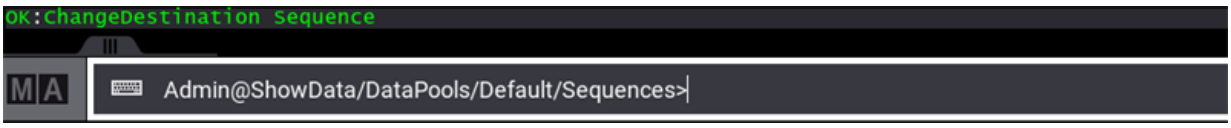
Examples

Change the Value of a Sequence Property

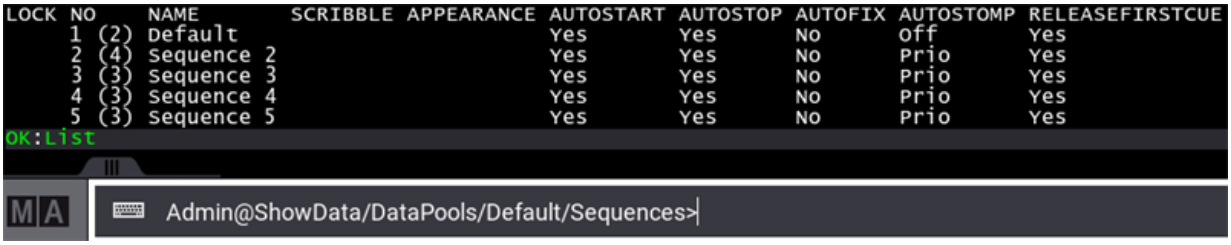
The included data is categorized in directories. In order to change a directory, use the command "Change Destination".

- To change the destination to Sequence, type:

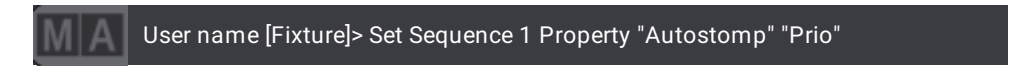


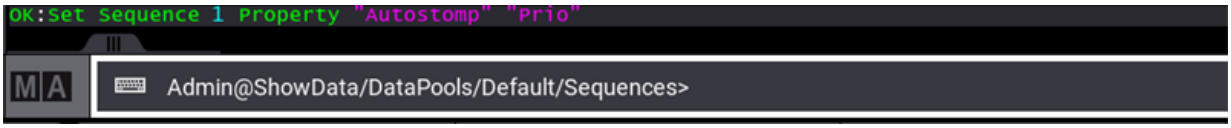


- To display the options in the command line feedback, type:

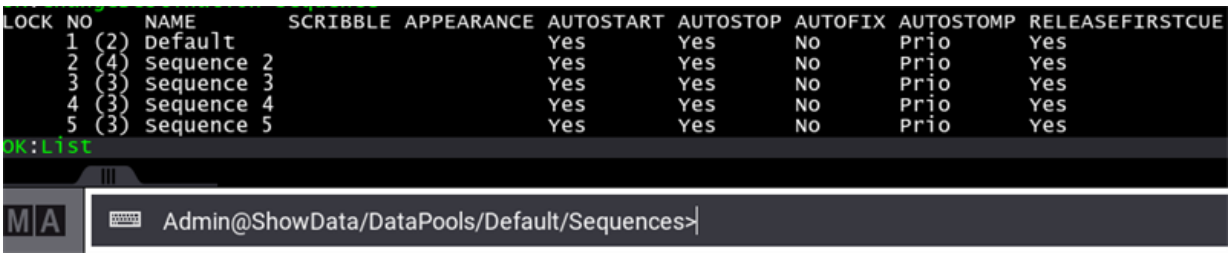


- To change Autostomp from Off to Prio, type:



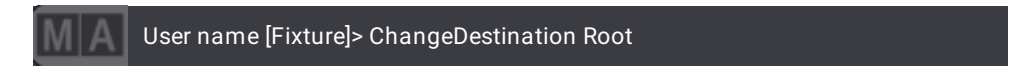


- To doublecheck the options in the command line feedback, type:



Autostomp is now set to Prio.

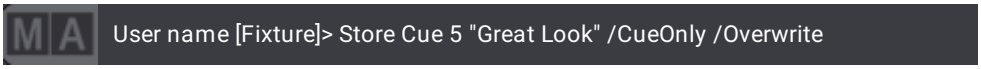
- To return to the destination root, type:

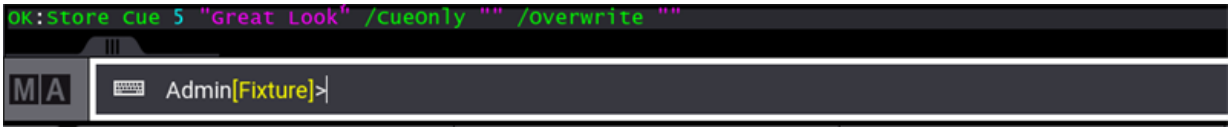


Store a Cue

Using the command line enables to combine a keyword with different options.

To overwrite cue 5 with the current programmer content and rename the cue to "Great Look" and storing the data as cue only, type:





9. Windows, Views, and Menus

Windows are created on the **Screens**. The screens are the monitors. The different sizes of grandMA3 hardware have different amounts of screens. The grandMA3 onPC has quick access to open 5 displays with a large screen area and 2 displays with a small screen area. This offers access to the same primary screens as a grandMA3 full-size. For more information, see **Screen Allocation**.

Most screens are empty in a new show. An empty screen is a blank space where each user can create their own arrangement of windows. This arrangement can be stored as a **View**.

On the right side of each screen, there can be a number of **View Buttons**. Views can be assigned to these buttons. This allows for fast access to recall a stored view or update a view by storing it again.

A new show has six factory-made views assigned to the first six view buttons on each of the big screens. The two smaller screens have four and two views assigned. These can be changed or deleted.

A **Menu** is a big pop-up that covers most of the screen. There are several menus in the software that gives access to the setup of the console, system, fixtures, and much more. For more information, see **Menus**.

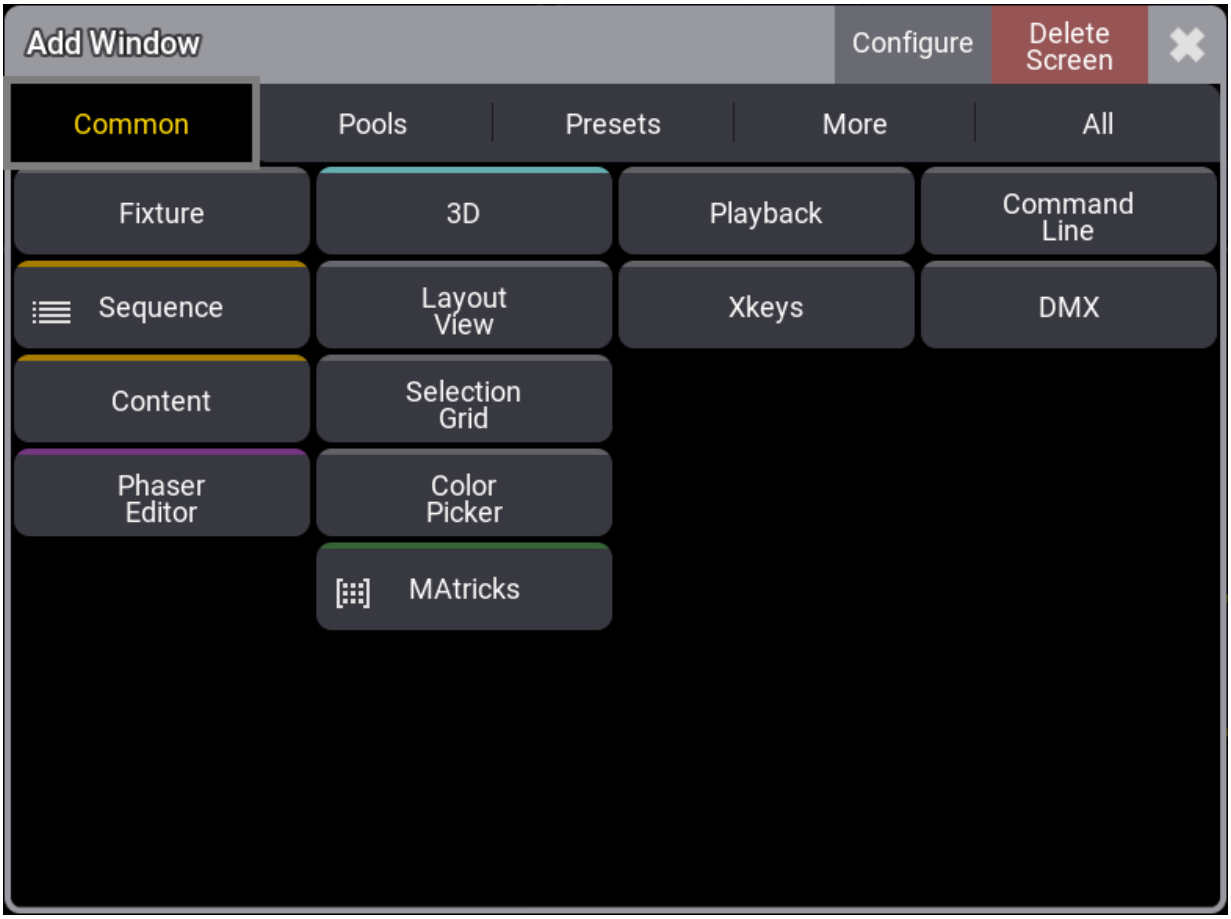
9.1. Add Window

Windows are added to the **user definable areas** on the screens.

The best way to add a window is by tapping inside an empty screen area. The window is added with the upper left corner where the screen is tapped.

Another option is to tap and draw a square on the empty screen to define the window size.

Tap the screen to open the **Add Window** pop-up:



Tap a button with the desired window to create it

Windows are divided into different tabs. The tabs are **Common**, **Pools**, **Presets**, **More**, and **All**. Tapping the different tabs reveals different buttons with the windows for each section.

Tap the desired button to create the window.

The title bar of the pop-up has two buttons. The red Delete Screen button removes all windows from the screen.

The Configure button opens the **Configure Display** pop-up. This can be used to customize some settings regarding tools on the side of the display and change the size of the user area. Read more in the **Configurations**

of Displays topic.

List of current windows that can be created using the **Add Window** pop-up:

Common:

- **Fixture** (Sheet)
- **Sequence** (Sheet)
- **Content** (Sheet)
- **Phaser Editor**
- **3D**
- **Layout View**
- **Selection Grid**
- **Color Picker**
- **MAtricks**
- **Playback**
- **Xkeys**
- **Command Line**
- **DMX**

Pools:

- **Filters**
- **Groups**
- **Layouts**
- **Macros**
- **MAtricks**
- **Worlds**
- **DataPools**
- **Exec Config (Executor Configuration)**
- **Pages**
- **Sequences**
- **Timecodes**
- **Timecode Slots**
- **Universes**
- **Cameras**
- **Plugins**
- Sounds
- **User**
- **Views**
- Gels
- **RenderQualities**
- **Appearances**

- Symbols
- Gobos
- **Images**
- Videos
- Meshes
- **Scribbles**

Presets (**one link to all about presets**):

- Dynamic
- Dimmer
- Position
- Gobo
- Color
- Beam
- Focus
- Control
- Shapers
- Video
- All 1
- All 2
- All 3
- All 4
- All 5

More:

- **AT Filter**
- **Prog. Parts**
- **Smart**
- **Timecode View**
- **Align Bar**
- **Command Wing Bar**
- **Custom/Master Section**
- **Encoder Bar**
- **Selection Bar**
- **Step Bar**
- **Agenda**
- **RDM Devices**
- **Running Playbacks**
- **Sound**
- **Trackpad**
- **Clock**
- **Help**
- **Info**
- **Message Center**
- **System Info**

- **System Monitor**

The **All** tab contains all windows in an alphabetical list that can be filtered.

9.2. Rearrange Windows

Tapping a free space on a screen opens the **Add Window** pop-up. The tapped grid field is used as the upper left corner of the created window. From here the window takes all the available space it can, and becomes as big as possible trying to fill all available space on the screen.

Tap, hold and drag on an empty screen draws a square. Releasing the screen opens the **Add Window** pop-up and the selected window fills the drawn square.

The window size can often be made smaller.

In the lower right corner of each window there is a small resize corner: 

Tap or click and holding this corner allows it to be moved. Moving it resizes the window.

Windows can also be resized by tap and hold in the title bar and meanwhile tap another location on the screen. The window will fill the available space.

Windows can also be moved or resized by moving the entire window. Tap or click and hold the title bar while moving it on the screen will show the grid and the window can be dragged and dropped to a new position.

Windows cannot overlap each other, so it can only be as big as the free space allows. If a window is moved up against another window then it will resize the window.

The windows snap to a dotted grid. This grid is used when windows are created, resized, and moved. There are half size grid dots, but pools and preset windows are limited to a full grid size when resized.

Some windows have a minimum size limit. For instance, a Fixture Sheet window needs to be a minimum of 3 x 2 grid fields.

9.3. Store and Recall Views

The arrangement of windows is called a **View**. Views are stored in a **View Pool**. The pool can be created as a window.

View pool window

Views can be assigned to **ViewButtons** and on **Executors**.

Store Views

A view is stored using the standard store syntax and the **View keyword**. This means that it is possible to use the keys, keys in combination with buttons on the screens, or the command line.

Storing a view opens the **Store View Options** pop-up.

Select what screens to store in a view

This is used to directly label the view and select which screens should be stored in the view. Each screen has a button that can be turned On or Off. The last touched screen is selected as a default. There are two buttons at the bottom that gives fast access to select All or None of the screens. Tap OK to store the window arrangement of the selected screens or press Please to confirm the options.

If the command line is used then the screens can be specified using the **/Screen** options. For more information please read the **Store keyword** and **View keyword** topics.

Store Using Keys

1. Press Store.
2. Press and hold MA while pressing X7 | View.
   This puts the **View keyword** in the command line.
3. Use the numeric keys to type the view number.
4. Execute the command by pressing Please.

5. Select the desired screens in the pop-up and press Please to confirm the options.

If point 3 is skipped (not adding a number), then the first available view is stored.

Store Using a Combination of Keys and Touch Screens

Requirement:

- A visible **View Pool** on one of the screens.

To store a new view:

1. Press Store.
2. Tap an available pool object in the view pool to create a new view or one of the existing pool objects to overwrite the existing view.
3. Select the desired screens in the pop-up and tap OK to confirm the options.

The last touched object (the object with the white frame) can be labeled if you start typing on the keyboard. Any of the labeling methods described in the **Label Pool Objects** topic can also be used.

Store Using Command Line

Storing using the command line is very simple. The two keywords needed are **Store** and **View.**

For example, storing the window arrangement on screens 1 and 2 as view 15 with the "layout" label:

See more in the **View keyword** topic.

Store a View Directly on a View Button

A new view can be stored on a **View Button**. This creates a view in the view pool and immediately assigns it to the view button.

1. Press Store.
2. Tap a view button.
3. Select the desired screens in the pop-up and tap OK to confirm the options.

Now there is a new view in the pool and it is assigned to the tapped view button.

Any of the **described label functions** can be used with the view buttons.

Assign Existing View to a View Button

The above method is for creating a new view and having it available on a view button. Existing views can also be assigned to the view buttons.

Again there are three primary ways to do it: Keys, keys and screens, and command line.

Assign View Using Keys

1. Press Assign.
2. Press and hold MA while pressing X7 | View once.
3. Use the numeric keys to type the view number.
4. Press At.
5. Press and hold MA while pressing X7 | View twice (this gives the **ViewButton** keyword).
6. Use the numeric keys to type the view button number.
7. Execute the command by pressing Please.

Example:

To assign view 4 at view button 7 on screen 1 the following key presses are needed:

Assign | MA + X7 | View | 4 | At | MA + X7 | View | MA + X7 | View | 1 | . | 7 | Please

This is the command result:

Assign View Using Keys and Screens

Requirement:

- A visible **View Pool** on one of the screens and visible view buttons.

To assign a view:

1. Press Assign.
2. Tap the desired view in the view pool.
3. Tap the desired view button.

Assign View Using the Command Line

The **Assign**, **View**, and **ViewButton** keywords are needed for this command.

Assign View [View_ID] At Viewbutton [Display_ID.ViewButton_ID]

Assign Existing View to an Executor

Existing views can be assigned to executors.

Again there are three primary ways to do it: Keys, keys and screens, and command line.

Assign View Using Keys

When objects are assigned to executors on a specific page using the keys and command line, then the object needs to be assigned to the child of the page using the **Page** keyword. The executors are children of the page.

If the object needs to be assigned to executors on the active page, then it can be assigned to the executor using the **Executor** keyword.

1. Press Assign.
2. Press and hold MA while pressing X7 | View once.
3. Use the numeric keys to type the view number.
4. Press At.
5. Press and hold MA while pressing X16 | Exec once.
6. Use the numeric keys to type the executor number (executor on the active page).
7. Execute the command by pressing Please.

Example:

To assign view 9 at executor X1 (executor 291) on executor page 3, the following key presses are needed:

Assign | MA + X7 | View | 9 | At | MA + X15 | Page | 3 | . | 2 | 9 | 1 | Please

This is the command result:

Assign View Using Keys and Screens

Requirement:

- A visible **View Pool** on one of the screens and access to executors.

To assign a view:

1. Press Assign.
2. Tap the desired view in the view pool.
3. Tap the desired executor.

Assign View Using the Command Line

The **Assign**, **View**, and **Page** or **Executor** keywords are needed for this command.

Assign View [View_ID] At Executor [Executor_ID]

Assign View [View_ID At Page [Page_ID].[Executor_ID]

Recall Views

Stored views are recalled to load the view.

If a view is recalled without specifying a destination (which screen should the view be recalled to), then it by default opens on the screen where it was stored. The destination can be set using the /Screen preference. See the example below and learn more in the **View keyword** topic. If the view is stored with two or more screens, then the views are always recalled on the stored screens.

When the view is assigned to a view button, then pressing the view button recalls the view on the same screen as the view button.

Tapping a view in the pool recalls the view on the same screen as the tapped view pool.

Pressing an executor to recall a view uses the default behavior described above.

Recalling the view using the command line without specifying a destination, then it is recalled on the screen where the command line has focus (The command line can be on several screens). Specifying a destination in the command line overrides this default behavior.

(Call) View [View_ID]

(Call) View [View_ID] /Screen "[Screen_ID]"

Update Views

A view is updated simply by storing it again and overwriting the existing view. This can be done directly in the view pool or on the view button using any of the methods described above.

9.4. Remove Windows from a Screen

There are three different ways to remove or delete a window from a screen: A single window can be deleted, all windows on a screen can be deleted, or all windows can be deleted on all screens

This can be done using the command line or the GUI. This topic is about the GUI method. For information about the command method, please see the **Delete keyword** and the **ScreenContent keyword**.

Delete a Single Window on a Screen

Requirement:
A window needs to be on the screen.

1. Tap the MA icon in the windows title bar
   - The settings pop-up appears.
2. Tap the Delete Window button in the title bar of the settings pop-up.

The window is now deleted from the screen.

Delete All Windows From a Screen

There are two methods to do this. Using the **Add Window pop-up** or the **Display pop-up**.

Add Window Method

Requirement:
An empty area on the screen.

1. Tap the empty area on the screen.
2. Tap the Delete Screen button in the **Add Window** pop-up.

All windows are now deleted from the screen.

Display Method

1. Open the Menu and Display pop-up by pressing the Menu key or tap the icon in the **Control Bar**.
2. In the Display pop-up, tap the Delete This Screen button on the screen that needs to be deleted.

All windows are now deleted from the screen.

Delete All Windows From All Screens

1. Open the Menu and Display pop-up by pressing the Menu key or tap the icon in the **Control Bar**.
2. In the Display pop-up, tap the Delete All Screens button on any of the screens.

All windows are now deleted from all screens.

9.5. Window Settings

All windows have settings.

The settings can be accessed by tapping the MA logo in the title bar or title field.

The settings vary depending on the window. The settings are organized in different sections, visualized by different tabs.

The **Display** section looks like this for the **Fixture Sheet**:

Fixture sheet settings - Display tab

There are some common buttons in the settings title bar. The red Delete Window button deletes the window from the screen. The ✖ closes the settings pop-up.

The Save button is used to store the current settings as a user preference. The Load button is used to load stored preferences. Read more **below** to learn more about how to store and load preferences.

The number of tabs changes depending on the window. Some windows have a lot of settings, some have fewer.

Changing a setting does not close the settings pop-up.

Some settings open small selection pop-ups listing the different properties for the setting.

Display

This tab has settings regarding the way the window displays information. The settings here vary a lot depending on the window. The settings that are special for a single window type are described in the topic about the window.

The following is a list of the settings that are shared between two or more windows:

- **#Columns:**
  This input button is used to set how many columns a sheet should display (the settings **Transpose** and **Adjust Columns** must be switched on in all other sheets than the **DMX Sheet**).
- **Adjust Columns:**
  This On/Off button makes a sheet adjust the column width to match the window size and the number of columns.
- **Appearance:**
  Tapping this button opens a **Select Appearance** pop-up that lists all the defined appearances and the possibility to create a new appearance. Selecting one will apply that appearance to the window.
- **Cue Only**:
  It defines if the cue only function is On/Off when editing values in the **Sequence Sheet in track sheet mode** (only visible when that mode is On) and in the **Content Sheet**.
- **Executors**:
  This is an On/Off button in the **Playback window** and the **Xkeys window**. It is used to show or hide the executors in the window. If executors are hidden and labels are shown, then it looks like the executor labels in the Playback Bar on the letterbox screens.
- **Feature Sort**:
  This On/Off button activates feature sorting. The selected feature is moved before the other features in the **Fixture Sheet**, **Sequence Sheet**, and **Content Sheet**.
- **Fixed Target**:
  This setting defines the sequence a sheet displays if the **Link Type** is **Fixed**. Tapping this setting opens an **Assignment Editor** pop-up where a sequence can be selected.
- **Fixture Appearance**:
  This On/Off button shows or hides the appearance of the fixture in the **Fixture Sheet** and the **Content Sheet**.
- **Fixture Sort**:
  This On/Off button activates the sorting of fixtures. The fixtures are sorted in the selection order to the top of the **Fixture Sheet** , **Content Sheet**, and **Sequence Sheet** in track mode.
- **Font Size**:
  This selects the font size in the window. It is a swipe button that opens a list of sizes from 10 to 32. There is also a **Default** property. This is the same as size 18.
- **Frame Readout**:
  This defines the frame readout for this window. It can be used to overwrite the **user profile default** readout.

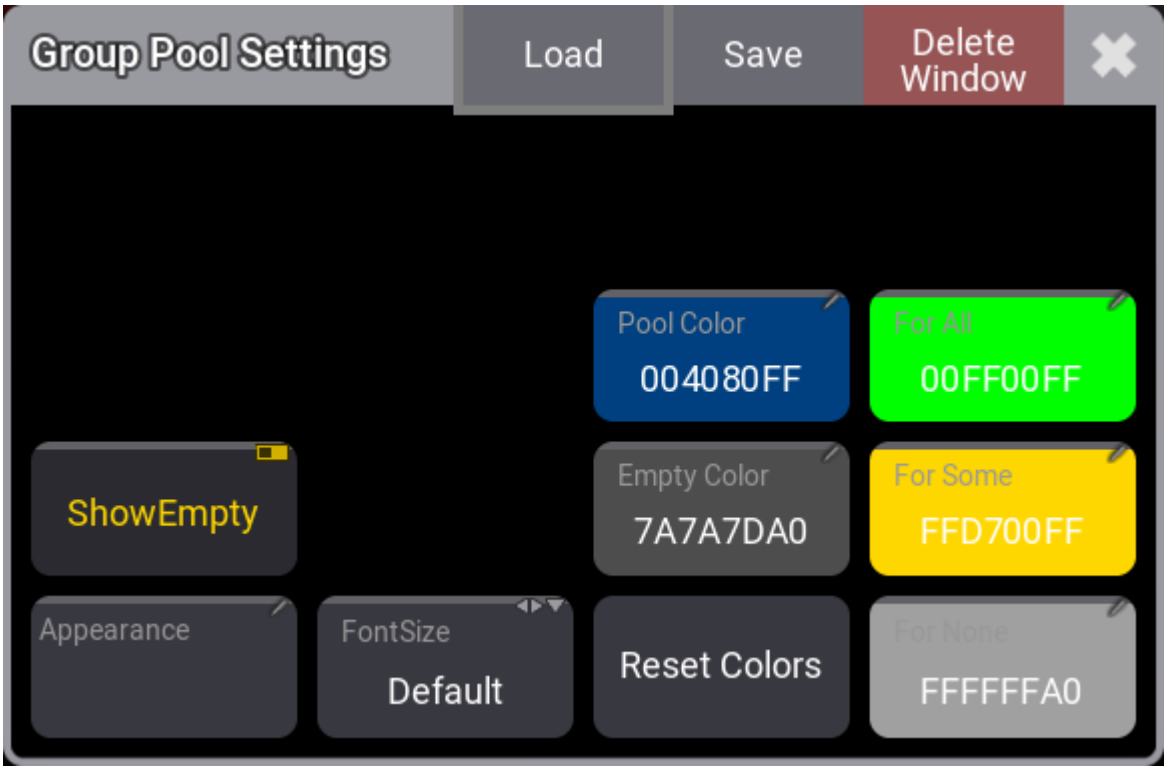
- **Labels**:
  This is an On/Off button in the **Playback window** and the **Xkeys window**. It is used to show or hide the labels in the window.
- **Layer:**
  It selects which layer is displayed in the window. It is a swipe button that opens a list of the layers. A special property is **Auto**. This property makes the window follow the selected layer in the **Encoder Bar**.
- **Layer Toolbar**:
  This On/Off button shows or hides a **layer toolbar** at the bottom with the different Layers. This is a property for the **Fixture Sheet**, **Sequence Sheet**, **Content Sheet**, and **Phaser Editor**.
- **Link Type**:
  This setting defines what sequence is shown in **Sequence Sheets** and **Content Sheets**.
  There are three different link types. This button gives access to choosing the desired one. The options are:
  - **Fixed**:
    The sheet displays the information from a specific sequence. The selection is made in the Sheet Settings. Read about the **Fixed Target** setting above.
  - **Selected**:
    The sheet displays information from the selected sequence.
  - **LastGo**:
    This automatically shows the latest sequence to receive one of the trigger commands (<<<, >>>, Go+, Go-, Goto, Load, On, Select, Top, Temp, Flash, Toggle On, Pause). This includes if the sequence is triggered from a running timecode recording. A sequence can be excluded from LastGo by turning Off the **Include Link Last Go** setting in the **Sequence Settings**. LastGo only shows sequences triggered by the same user profile.
- **Merge Cells**:
  When this is On then the values of a feature are merged to only be shown once if the two or more adjacent values are the same. For instance, if all red, green, and blue values are 100, then there is only shown 100 once.
- **Page:**
  This is used in the **Playback window** , the **Command Wing Bar** window, and the **Xkeys window**. It is used to change which executor page they relate to.
- **Preset**:
  This defines how the preset information is displayed in the **Fixture Sheet** and the **Sequence Sheet**. There are six properties which are different combinations of these three elements:
  - **ID**:
    Shows the ID number of the preset.
  - **Name**:
    Shows the name of the preset.
  - **Value**:
    Shows the values stored in the preset.
- **Readout:**
  This selects the value readout for fixture attributes. It is a swipe button that opens a list of readout types. Here is also an **Auto** property. It makes the sheet follow the selected readout in the **Encoder Bar**.

- **Setup**:
  This changes the **3D**, **Agenda**, **Layout View,** and the **Timecode View** into a mode where the different elements in the two views can be changed.
- **Show Grand Master**:
  This shows or hides the Grand Master section in the **Custom/Master Section** Window, the **Command Wing Bar** window, and the **Encoder Bar** Window.
- **Show Master Section:**
  This shows or hides the Master section in the **Custom/Master Section** Window and the **Command Wing Bar** window.
- **Show Title Bar**:
  This shows or hides the windows title bar in **Layout View**, **Clock**, and **3D**. It is On by default. If it is Off, then the title bar can be shown temporarily by pressing both MA keys in the control area. In grandMA3 onPC the title bar can be temporarily shown by pressing Ctrl + Alt in Windows and Ctrl + Option on Mac.
- **Step:**
  It selects what step to display. Steps are used with **Phasers**. It is a property input button that opens a calculator pop-up. This is a property for the **Fixture Sheet** and **Sequence Sheet**.
- **Time Format**:
  This defines the time format for the **Fixture Sheet** and the **Sequence Sheet** windows. This can be used to select a different format than the **user profile default**.
- **Transpose:**
  This On/Off button flips the columns and rows in windows.
- **View Mode**:
  The view mode defines how the different data and information are displayed in the view. This setting is also a button in the title bar in the two relevant windows.
  The **Phaser Editor** has the following view mode properties: Auto, 2D, 1D, and Step.
  The **Timecode View** has the following view mode properties: Text, Timeline, and Both.
- **Wing ID:**
  Defines which wing the **Command Wing Bar** window and the **Playback** window displays. Tap this setting to open a small **Select WingID** pop-up where the desired wing can be selected.

Hint:
Showing the Layout View or the 3D window in a true full-screen version can be achieved by hiding all other visual elements on the screen using the **Configure Display pop-up** and then turning Off the **Show Title Bar** setting for the window.

Pool Settings

Most pools have the same few settings. Some pools might have extra settings. The Group and Preset pools have three extra settings regarding the colored bar above each pool object. The **Smart window** behaves much like a pool and shares most of the pool settings.



General pool window settings

These are the common settings for pools:

- **Show Empty**:
  This toggle button can hide or show empty pool objects.
- **Appearance**:
  The appearance is applied behind the pool objects.
- **Font Size**:
  There are some different font size properties from 10 to 32. There is also a default property. This is the same as size 18. This simply changes the font size on the pool objects.
- **Pool Color**:
  This is the color for the title button in the pool.
- **Empty Color**:
  This color is applied to empty pool objects.
- **Reset Colors**:
  This resets the colors to the colors in the default color theme.

Preset and group pools also have:

- **For All**:
  This color is used when the preset can be used by all of the selected fixtures.
- **For Some**:
  This color is used when the preset can be used by some of the selected fixtures or when there are more fixtures that can use the preset, than the current selection.
- **For None**:
  This color is used when the preset is not useable by any of the selected fixtures or none of the selected fixtures are in the group.

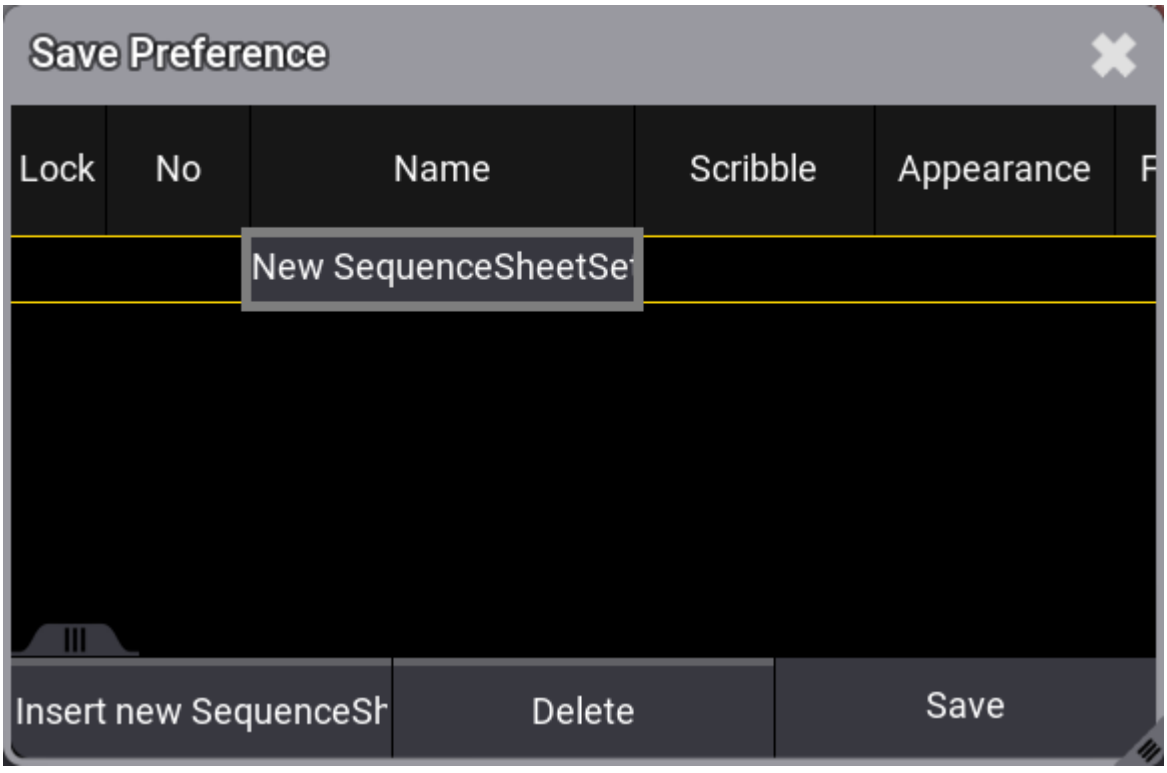
The buttons in the title bar are the same as the ones for other window settings - read above.

Tapping one of the color settings opens an **Edit Color** pop-up.

Store and Load Preferences

Setting preferences can be stored and loaded for each window and pool. These preferences are stored in the user profile. This means that exporting and importing a user profile includes these preferences.

Tapping Save opens a **Save Preferences** pop-up:



Save Preference pop-up

Save a New Preference

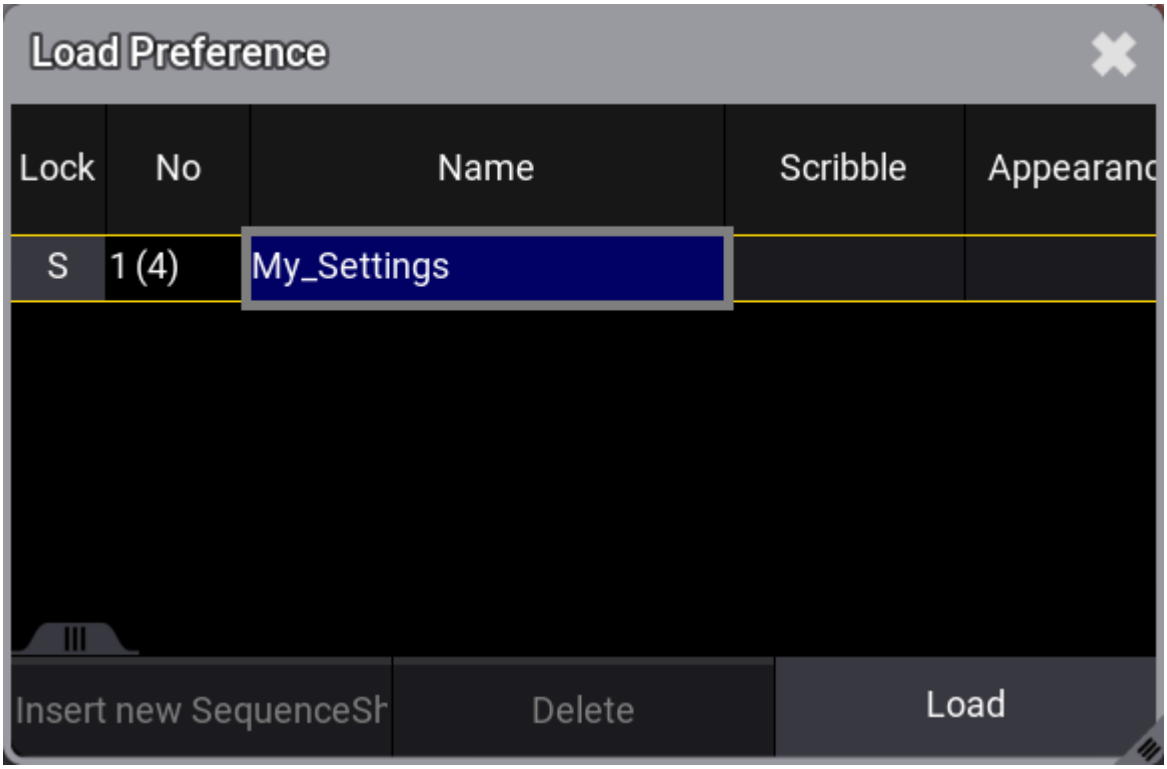
1. Open the **Settings** pop-up.
2. Tap Save in the settings title bar.
3. Tap Insert new [Type] (this button changes a little for each window and the type shows what type of settings are being inserted).
4. Edit the name field and give it a name.
5. Tap Save in the preference pop-up.

Hint:
When saving a new preference, no preference exists yet, and the focus is on "New ...", then it is enough to tap Save to save the preference.

Update the Preference

1. Open the **Settings** pop-up.
2. Tap Save in the settings title bar.
3. Tap the desired preference in the list.
4. Tap Save in the preference pop-up.

Load a Preference



Load preference pop-up

1. Open the **Settings** pop-up.

2. Tap Load in the settings title bar.
3. Select the desired preferences.
4. Tap Load in the preference pop-up to confirm.

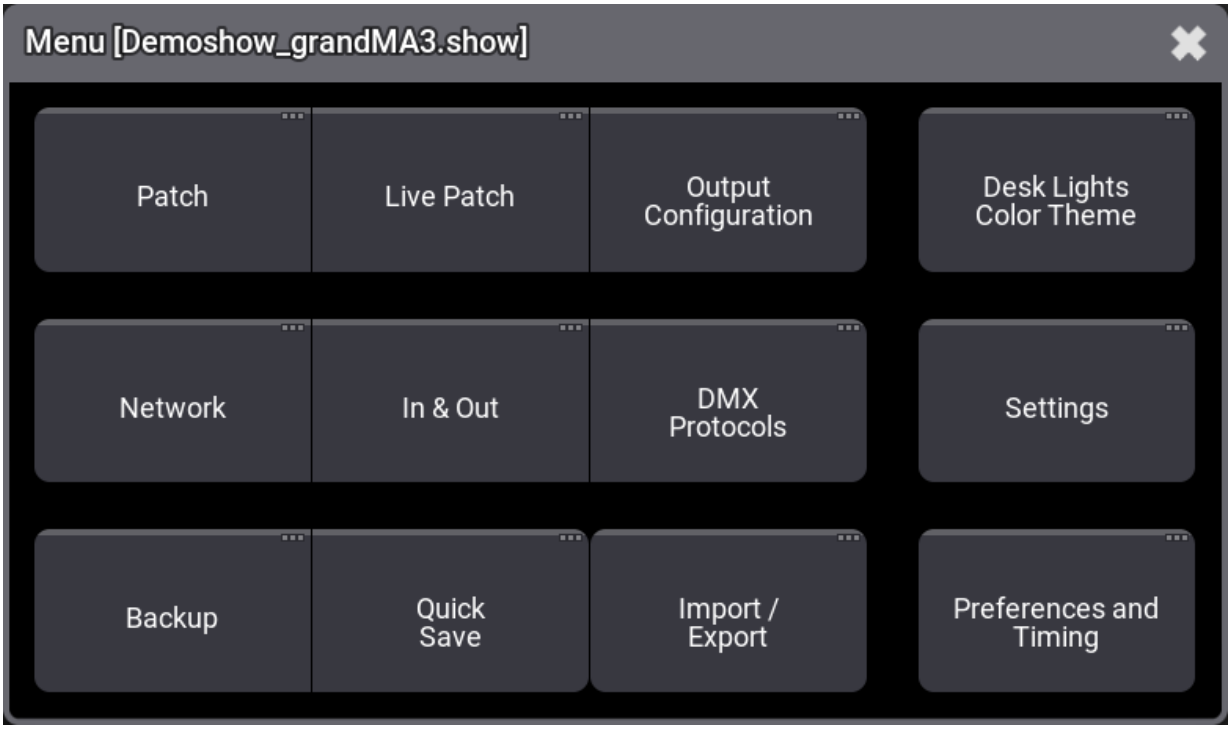
Delete a Preference

1. Open the **Settings** pop-up.
2. Tap Save in the settings title bar.
3. Select the desired preferences.
4. Tap Delete in the preference pop-up.
5. Close the pop-up by tapping the in the upper right corner.

9.6. Menus

There are many different menus in the grandMA3 software. Most are described in relevant sections. For instance, the patch menu is described in the **Patch section**.

There is a **Menu pop-up** that gives access to many of the different menus.



Menu pop-up with buttons that opens a lot of menus

This pop-up can be opened in multiple ways:

- On physical hardware, there is a Menu key. Press this to open the pop-up.
- In the **Control Bar** on the left side of most displays, on the onPC, there is a gear icon. Tap this to open the pop-up.
- It can be opened using the command line:

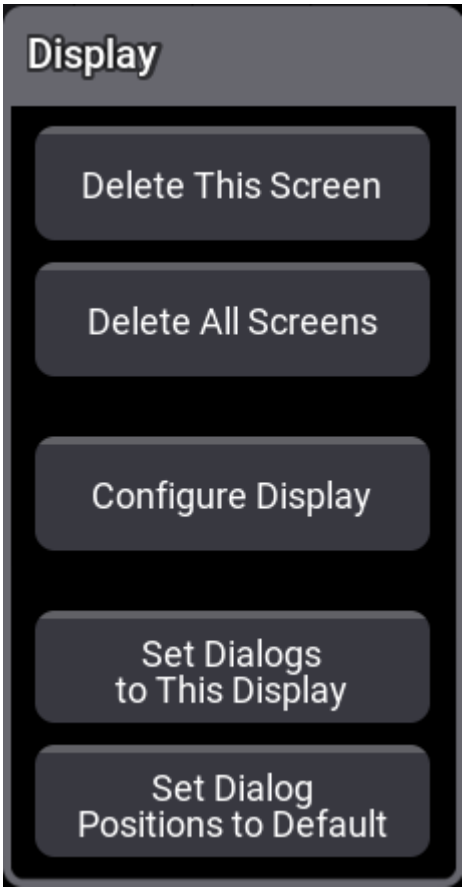
```
User name[Fixture]> Menu "MenuSelector"
```

The menu pop-up gives access to the following menus:

- **Patch**:
  This is where fixtures are added and all about fixture setup - Read more in the **Patch and Fixture Setup section**.
- **Live Patch**:
  This menu gives access to the fixture setup that can be changed without a new show upload - Read more in the **Patch and Fixture Setup section**.
- **Output Configuration**:
  The **output configuration** is used to change the local physical DMX ports on the MA equipment.
- **Desk Lights Color Theme**:
  This opens a pop-up that gives access to **customize the look of the software**.
- **Network**:
  This menu is used to set up all about the network and sessions - read more in the **Networking section**.
- **In & Out**:
  This opens the **In & Out menu** that has settings for DC remote, MIDI remote, DMX remotes, OSC, and PSN.
- **DMX Protocols**:
  This menu is used to set up DMX **network protocols**.
- **Settings**: This opens a small sub-menu that has these options:
  - **User Configuration**:
    This menu is used to set up users and user profiles - read more in the **Single User and Multi-User Systems section**.
  - **Date and Time**:
    This menu is used to change the time and date in the console - read more in the **Date and Time topic**.
  - **USB Configuration**:
    This is used to configure USB devices. This should normally not be changed manually.
  - **Software Update**:
    This menu is used to **update the software**.
  - **Touch Configuration**:
    This menu is used to assign touch screens to the USB inputs.
  - **Extension Configuration**:
    This menu shows the connected grandMA3 extensions.
  - **onPC Settings**:
    This is a pop-up with **onPC settings**.
- **Backup**:
  The backup menu is used to save and load shows - read more in the **Show File Handling topic**.
- **Quick Save**:
  This is not a menu but a shortcut to saving the show with the same show file name.
- **Import / Export**:
  This opens an **Import and Export menu** that can be used to import and export different objects of the show file.
- **Preferences and Timing**:
  This is where the defaults are set up for cues and sequences - read more in the **Cue Timing topic**.

Display Overlay

Opening the **Menu pop-up** also opens a **Display overlay**.



Display overlay in the right hand lower corner of all screens

This pop-up appears on all screens except the letterbox screens.

Most of the actions performed using this pop-up relate to the specific screen where the menu is touched. For instance, deleting a screen (removing the windows from the screen).

- **Delete All Screens**:
  This clears all windows from screens - read more in the **Remove Windows from a Screen topic**.
- **Delete This Screen**:
  This clears all windows from this screen - read more in the **Remove Windows from a Screen topic**.
- **Configure Display**:
  This opens the Configure Display pop-up - read more in the **Configuration of Displays topic**.
- **Set Dialogs to This Display**:
  This makes all menus open on this screen - read more in the **Change Menu Locations topic**.

- **Set Dialog Positions to Default**:
  This resets all menus to open on the default screen - read more in the **Change Menu Locations topic**.

9.7. Change Menu Locations

Some menus can be moved between the screens.

Menus that can be moved have a display icon in the title bar:

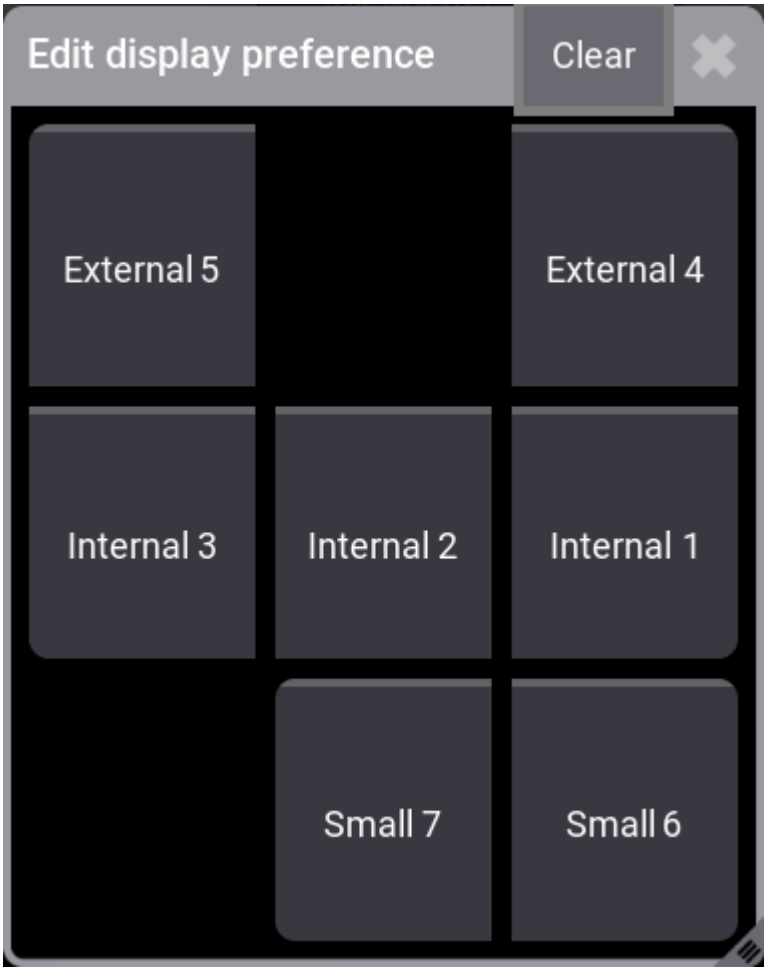
Display icon next to the close X

Move a menu to a different screen

Requirement:

- Open a menu that can be moved. For instance the **Playback Controls** Menu.

1. Tap the display icon in the title bar.
   The **Edit Display Preference pop-up** appears:



Use Edit Display Preference pop-up to select a new location

2. Tap the button corresponding to the desired screen.

Hint:
Tap the Clear button or the  to cancel the move.

Move all menus to a screen

It is possible to select a screen where all moveable menus opens.

1. Open the **Menu and Display pop-up**.
2. In the Display pop-up tap the Set Dialog to This Display button on the screen where the menus should open.

Reset movable menus screen

The screen selection can be reset. This is useful if a show changes from a smaller hardware platform. For instance, if the show is programmed on a grandMA3 full-size and menus are moved to screen 3 and now the show is opened on a grandMA3 light without screen 3.

1. Open the **Menu and Display pop-up**.
2. In the Display pop-up tap the Set Dialog Positions to Default button on the screen where the menus should open.

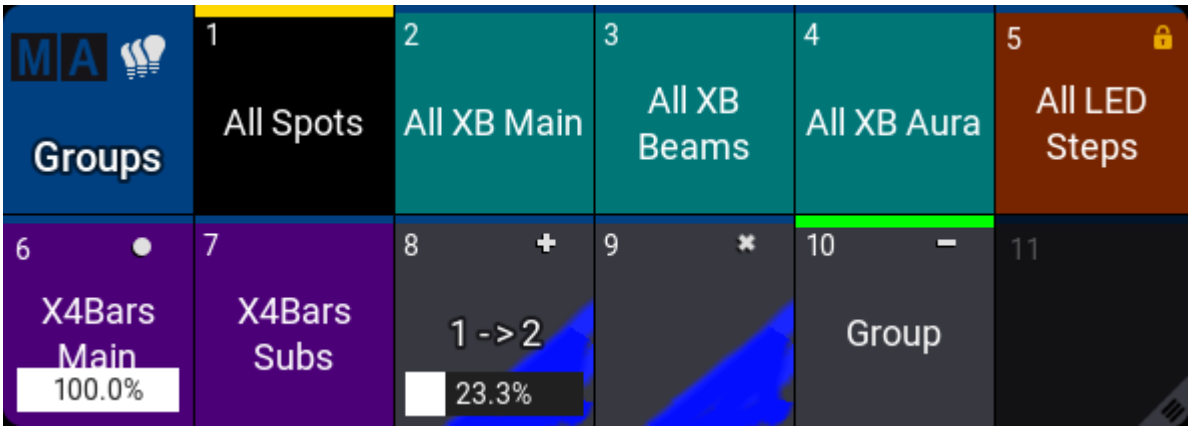
9.8. Pool Windows

Most of the show data in grandMA3 are organized in pools - these pools are a part of a **Data Pool**. So, in essence, everything except the patch and fixture setup are stored inside a Data Pool. This makes it easy to have several shows with the same fixture setup but with different pools of data.

Pools contain a lot of different data. For instance, the **Groups pool** contains information about selections of fixtures. The **Views pool** contains information about the arrangement of windows on a screen.

Pools are made like any other window - read more in the **Add Windows topic**. Pool arrangement and resizing are just like any other window - read more in the **Rearrange topic**.

This is an example of the Groups pool:



Groups pool

Each pool has a title field. In the example above it is the first blue square with the MA logo, the pool icon (if available), and the name of the pool.

Tapping the logo opens the settings for the entire pool - read more about the common window settings in the **Window Settings** topic. For details about pool or preset specific settings read the relevant topics.

Some pools have specific use. This topic and the following subtopics describe the general concepts. Read the specific topics to learn the details for each type of pool.

Pool Object

All other fields except the title field contain a **Pool Object**. If something is stored in the pool object, then it has a lighter gray color (default color) and there is a label (see the example pool above). It can also contain pool specific information like a master level or icons that show information about the content or pool object settings.

The number in the upper left corner of each pool object is the unique pool number identifier. It can be used when the object is called. For instance, using **Group 1** in the command line is the first object in the groups pool. The name of the pool object can also be used to call the group. For example **Group "All Spots"** calls (all) the groups with that name. Asterisk can be used as a joker sign when using object names. Typing **Group "All*"** calls all the

group objects where the name starts with **All**.

Working with or using pool objects can be done in many ways. The examples in the subtopics use the command line, but many of the operations could also be performed using the keys, screens, or any combination thereof.

Some pools have **Selected** pool objects. It is visualized with a yellow frame around the selected pool object.

World 1 is selected

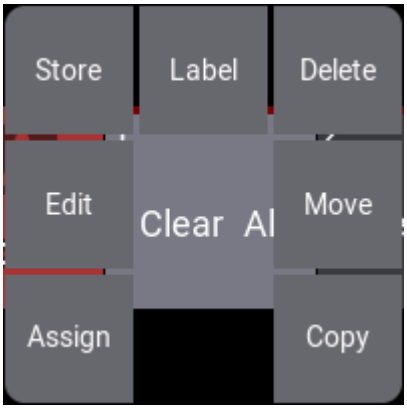
A different pool object can usually be selected by tapping the object in the pool or using the **Select keyword**.

Swipey Commands

Each pool object can open a set of **Swipey Commands**. They give quick access to some common operations with the pool object.

This is how to open the swipey commands:

1. Tap and hold the pool object.
2. Move the finger/cursor outside the pool object while keep pressing the screen.



The open Swipey Commands

3. Keep the screen pressed and move the finger/cursor to the desired command.
4. Release the screen.

These are the available commands with the swipeys:

- **Assign:**
  This adds "Assign" and the pool object type and number in the command line. Waiting for more user input.
- **Edit**:
  This starts the edit mode for the pool object - this is about the content of the pool object.
- **Edit Setting**:
  This opens a small editor that gives access to editing this pool object's settings - this is about the settings for the pool object.
- **Store:**
  This executes "Store" plus the pool object type and number. This is useful for storing new pool objects - for instance, groups.
- **Label**:
  This executes "Label" plus the pool object type and number.
- **Delete:**
  This executes "Delete" plus the pool object type and number - The result is deleting the pool object.
- **Move**:
  This adds "Move" plus the pool object type and number in the command line. Waiting for the user to tap the new location.
- **Copy**:
  This adds "Copy" plus the pool object type and number in the command line. Waiting for the user to tap the new location.

9.8.1. Create Pool Object

This is a general description of how to create pool objects. Please read the topics about each type of pool for details.

The **Store keyword** is used when creating pool objects. This stores the relevant information in the pool object if the relevant information is available.

For instance, to store a preset, there need to be correct values in the programmer. If the relevant values are not available, then an empty pool object is created.

Some pool objects are a bit more complex. For instance, storing a pool object in the image pool. The pool that can be added on a screen, is not just called "Images" as the title suggests. It is actually image "pool" 3.

So storing something on pool object 2 needs the following input:

```
User name[Fixture]> Store Image 3.2
```

But this only stores an empty pool object.

Some objects are better to **Edit** when creating them. This opens the relevant editor to create the object.

Images are an example of such an object.

```
User name[Fixture]> Edit Image 3.2
```

This command creates the object and immediately opens the editor for the image pool object. Read more about images in the **Images section**.

Some pool objects can be created by pressing and holding an empty pool object. If there is relevant information in the programmer, then this might be stored in the new pool object.

For instance, having a selection of fixtures in the programmer and pressing and holding an empty group pool object for about 2 seconds will create a new group with the fixture selection.

9.8.2. Label pool objects

All pool objects can be labeled. The label contains a name, a scribble, and an appearance.

The name can be changed using the **Label** keyword. It can be written in the command line. Simply write **Label** followed by the object you want to name and then the name in quotation marks.

For instance, if group 1 needs to be named "Stage Right Spots" use the following command:

The label command can be accessed by pressing the Assign **key** twice. The syntax is the same no matter how the command is created:

Label [Object_type] [Object_ID] "Object Name"

If the object name is omitted in the command, an **Edit Label** pop-up appears:

Enter a label

The down-pointing triangle opens an on-screen keyboard.

The scribble icon next to the triangle opens the **Scribble pad**.

The last touched pool object has a white frame around it. If the keyboard is used, then the keyboard text input is often interpreted as a new label input and this can be used to label all pool objects. This is the fasted way to label newly created objects.

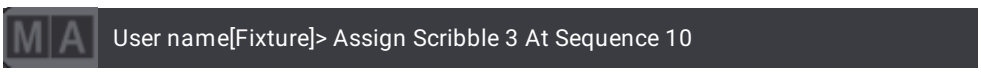
Scribbles

Scribbles are small drawings that can be used together with or instead of a label.

Scribbles from the Scribble pool can be assigned to a pool object. This is the syntax:

Assign Scribble [Scribble_ID] At [Object_Type] [Object_ID]

For instance, if scribble 3 is to be used with Sequence 10, then the following command can be used:



Learn more about scribbles in the **Scribble section**.

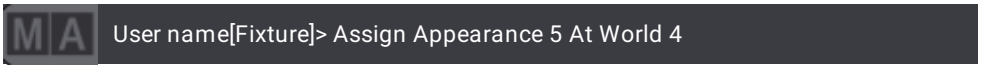
Appearance

An **Appearance** can be assigned to pool objects.

This can also serve as a labeling function. This is the syntax :

Assign Appearance [Appearance_ID] At [Object_Type] [Object_ID]

For instance, if appearance 5 is to be used for world 4, then the following command can be used:



Learn more about appearances in the **Appearance section**.

9.8.3. Move pool objects

Pool objects are moved using the **Move** keyword.

A single object can be moved or it can be a selection of several objects moved at once.

Example

Macro 5 needs to be moved to the empty macro pool object 20:

If the destination is not empty, then the existing object will move one place further. If this object is also occupied, it will move further, and so on until no occupied object has to move anymore. Thus potentially moving all the arranged pool objects.

Important:
Please have a look at the **Exchange keyword** for grandMA2 move functionality.

9.8.4. Insert pool objects

Pool objects can be inserted between other pool objects of the same type. This can be useful for organizing the pools.

It is done using the **Insert** keyword.

The inserted pool object is inserted before the destination object number.

Example

For example inserting view 42 between view 9 and 10:

The old view 10 and other views from 10 to the next empty pool object is then moved one number up.

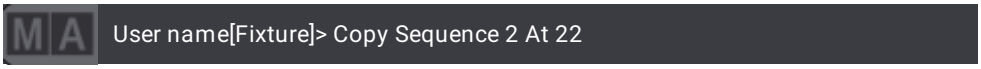
9.8.5. Copy pool objects

Pool objects can be copied to an empty location using the **Copy** keyword.

A single object can be copied or it can be a selection of several objects copied at once. They are positioned at the new location using the selection order.

Example

Copy sequence 2 at pool object 22 using the following command:



If the destination is not empty then the old object is overwritten!

9.8.6. Lock and Unlock pool objects

Lock pool objects

Pool objects can be locked using the **Lock** keyword.

If a pool object is locked then there is a small lock icon in the upper right corner of the pool object.

Example

Locking appearance number 4 is done using the following command:

```
User name[Fixture]> Lock Appearance 4
```

Unlock pool objects

Objects can be unlocked using the **Unlock** keyword. The syntax is the same as locking except the keyword.

Example

Unlocking appearance number 4 is done using this command:

```
User name[Fixture]> Unlock Appearance 4
```

Predefined pool objects are locked from the factory and cannot be unlocked.

9.8.7. Delete pool objects

Removing pool objects can be done using the **Delete** keyword.

Restriction:
When a pool object is deleted, then all links to the object are lost. This means that if the pool object is returned using the oops function, then links to where it was used are not returned even though the pool object is back in the pool.

When an object is deleted, then grandMA3 will try to make everything look the same. For instance, deleting a preset will move the values stored in the preset into the different cues where the preset was used. This is not always possible, for instance, if an executor is controlling a sequence and the sequence is deleted, then it is gone and the executor is empty.

Example

Delete view 42:

```
User name[Fixture]> Delete View 42
```

It can also be a range of numbers for instance:

```
User name[Fixture]> Delete View 42 Thru 90
```

Deleting pool objects can be undone with **Oops**. It will bring back the pool objects but it might not restore links to the object.

For instance, deleting an appearance will remove the link to it where it might be used. Oops'ing the deletion will bring back the appearance, but it will not reassign the appearance where it was used before deletion.

10. Networking

Networking allows for the expansion of a single standalone console to a complete system with DMX nodes and extra processing powers.

There are three Ethernet connectors on the rear panel of all the consoles. They can be used to connect to three different networks.

Networking is use for:

- Connecting several MA3 devices in a session
- Output and input Ethernet-based DMX
- Internet connection

MA Session

Sessions are a way to expand a single console. It allows for connecting multiple consoles in a multi-user setup where several programmers can work in the same show. grandMA3 Processing Units can be added to move DMX calculations away from the consoles and into processors located where needed. grandMA3 xPort Nodes are listening to a session and function as DMX interfaces allowing decentralized synchronized DMX distribution.

A session is needed to connect grandMA3 devices. Read about **sessions here**.

Ethernet DMX

grandMA3 can output DMX using Art-Net and sACN. It is possible to set up a specific Ethernet port to output the network DMX. Read more in the **DMX In and Out of the system topic**.

Internet Connection

The grandMA3 system is designed to be able to connect to the Internet. More specifically connect to World Servers. These servers are able to provide different services to the connected stations.

It is recommended to separate the light network from the internet and use one of the three Ethernet connectors on the back of the MA hardware stations to connect to a network with internet access. This will maintain a separation between the internet and the light network and still provide access to the services on the world server.

For more information see the **World Server topic** and the **Interfaces and IP topic**.

Internal Connections

A console uses internal network connections to have different sections of the internal components connected. This is visible in the Network menu, but should never be changed by the users.

Enabling or Disabling the Network Connection

The network needs to be enabled to communicate. This includes transmitting DMX using Ethernet.

Turning network On or Off is done from the **Network menu** - read about the network menu in the **Sessions topic**.

In the lower right corner of the network menu, there is a button to toggle the network connection.

Network enable button

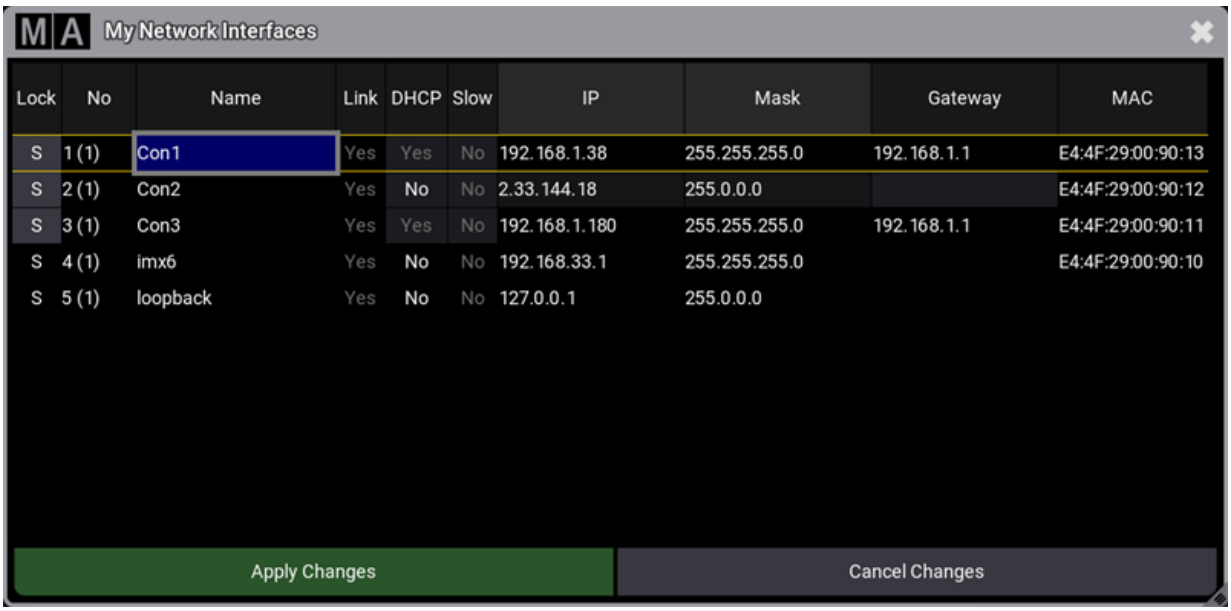
If the icon is red then the network is turned Off. If it is green then it is On.

10.1. Interfaces and IP

The Network Interface menu lists the available interfaces and in the grandMA3 hardware, it can be used to change the settings.

Open the Network Interface Menu

1. Press Menu.
   - Opens the **menu select pop-up**.
2. Tap Network.
   - Opens the Network menu.
3. Tap My Interfaces.
   - Opens the Network Interfaces pop-up:



See the Network Interfaces

Restriction:
If you see an interface with "imx6" then it is for internal communication inside the console. Do not make any changes to this interface.

Every cell with a light gray background can be edited.

The grandMA3 consoles have three network interfaces, with connectors on the rear panel, that can be used to **connect with external equipment**. The connectors are named **Con1**, **Con2**, and **Con3** in the Network Interface menu.

In the grandMA3 onPC, it is not possible to edit the IP addresses in the Network Interface menu. This needs to be done in the computer operating system.

Short Description of the Columns

- **Lock**:
  Changing the value to **Yes** in this cell locks the interface from being edited.
- **No**:
  This is the interface number.
- **Name**:
  This is the name of the interface.
- **Link**:
  This cell says **Yes** if there is an active connection on the connector.
- **DHCP**:
  This cell says **Yes** if DHCP is activated for the interface. No means that DHCP is turned Off, and it is possible to manually set a static IP address information. Read about DHCP below.
- **Slow**:
  This cell says **Yes** if the detected network speed is below 1 Gbit/s.
- **IP**:
  This is the IP address for the interface. If DHCP is set to "No", then this cell can be edited on the console to change the IP address for the interface. CIDR notation is accepted as input to also set the subnet mask.
- **Mask**:
  This is the subnet mask for the interface. If DHCP is set to "No", then this cell can be edited on the console to change the subnet mask for the interface.
- **Gateway**:
  This is the gateway address for the interface. If DHCP is set to "No", then this cell can be edited on the console to change the gateway address for the interface.
- **MAC**:
  This is the MAC address for the interface. This is information only and cannot be edited.

What is DHCP

DHCP is a system where IP addresses are distributed from a DHCP server. If this is On (the field says "Yes"), then it is not possible to set the IP address locally.

Edit the field to turn On or Off DHCP on the console.

Set the IP address in the console

Restriction:
The IP address range 192.168.33.x is not allowed to be used for the network interfaces Con1, Con2, or Con3.

1. Navigate to the Network Interfaces menu (read above).
2. Make sure DHCP is turned Off (the cell says "No") for the network connector (read above).

3. Edit the IP cell and write the new IP address (IPv4 only). The CIDR notation can be used to set the subnet mask while typing the IP address. For example, **192.168.101.11/24** gives the subnet mask to **255.255.255.0**.
4. Edit the Mask cell and write the subnet mask.
5. Optionally edit the Gateway cell to set a gateway address if needed.
6. Tap Apply Changes to use the new settings.

The changed IP address is used without the need for a reboot.

10.2. Session

MA devices are connected in **Sessions.**

Controlling devices (grandMA3 onPC, grandMA3 replay unit, and grandMA3 consoles) are referred to as **Stations**.

Multiple sessions can exist in the same network and are identified by a session name and location.

Hint:
Learn more about different ways the grandMA3 system can work in a networked environment in the **System Overview**.

MA devices recognized in the network can be seen in the network menu.

The menu can be accessed in multiple ways.

Open the Menu Using the Command Line

Type the following command in the command line input and execute it:

```
User name[Fixture]> Menu "Network"
```

Open the Menu from the Menu Pop-up

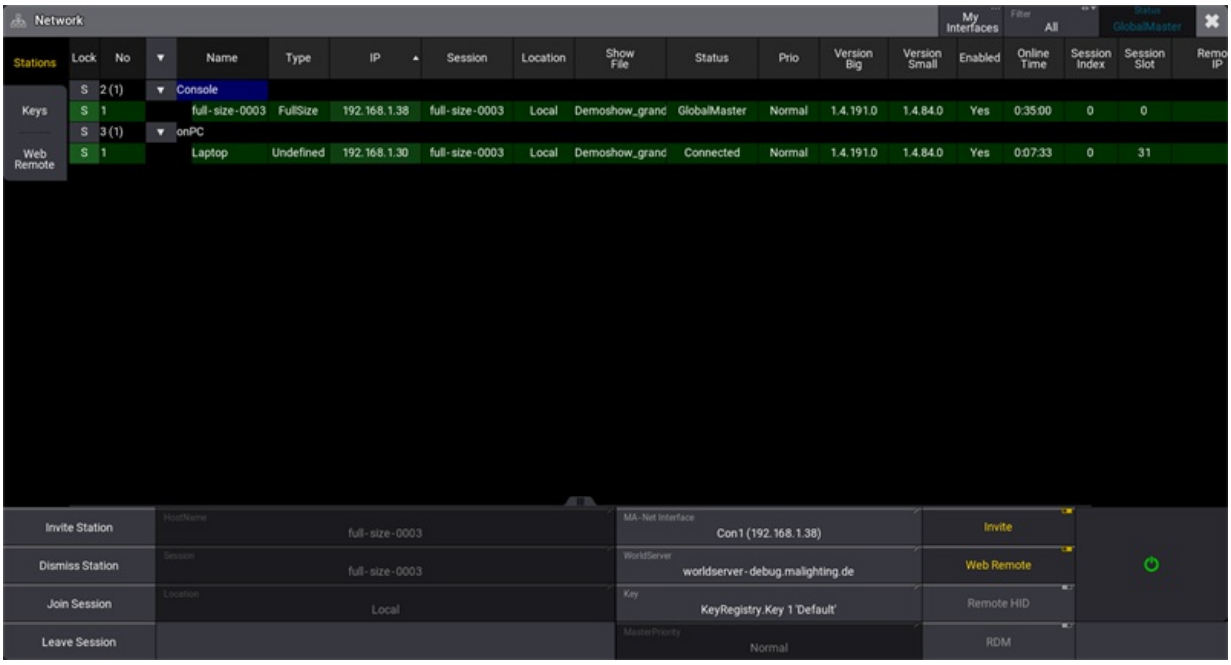
This is a combination of keys and buttons. It executes the same command as above.

1. Press Menu.
2. Tap Network on the menu selection pop-up.

Open the Menu Using the Dedicated Button

The **Command Line** has a dedicated button for the network menu:

Tap this button to open the menu.



The Network menu lists all MA devices in the network

The title bar has the normal close button ( ). Next to this is a button that displays the current status for this device, a filter button, and a button that opens a **My Interfaces pop-up** where the network interfaces are listed and might be modified.

The possible filters are:

- **All**:
  This lists all stations.
- **My Session**:
  This only lists the stations that are in the same session as this station.
- **Not My Session**:
  This lists the stations that are not part of this station's session.
- **Wrong Version**:
  This lists stations that have a different software version number than this station.
- **My Location**:
  This lists the stations that have the same **Location** value. Read about the location setting below.

The tool buttons on the left side have three buttons: Stations, **Keys**, and **Web Remote**.

There are four buttons on the left side at the bottom: **Invite Station**, **Dismiss stations**, **Join Session**, and **Leave Session**.

There are input buttons that change the session settings for the station being operated.

This is a short explanation of each field:

- **HostName**:
  The name for this device. The name can be changed freely. It does not influence the session. It can be used to identify individual devices. It can only be edited when the station is not in a session.
- **Session**:
  The name of the session. There are three elements that need to match before devices can connect to each other in the session. The session name, the **Location**, and the **Key**. It can only be edited when the station is not in a session.
- **Location**:
  The session location. It can be anything it does not need to be the actual location for the session. It is used together with the session name to identify the session. It can only be edited when the station is not in a session.
- **MA-Net Interface**:
  Tapping the Interface field opens a small **Select Interface pop-up**. This lists the possible network interfaces including an Auto option (Read more about this option **below**). Read more about changing the IP addresses in **Interfaces and IP**.
- **WorldServer**:
  This input field is used to define what world server to contact. Read more about it in the **World Server topic**.
- **Key**:
  The key is a password for the session. Tapping this field opens a small **Select Key pop-up** that lists the possible keys - including the option to create a new one. Read more about key creation in **Create a custom key**.
- **Master Priority**:
  This field is used to set the priority for this device. Tapping this field opens the small **Select MasterPriority pop-up**. It can only be edited when the station is not in a session.
  There are several levels of priority: **Never** (not able to be master), **Very Low**, **Low**, **Normal**, and **High**. If stations have the same priority then the station with the lowest slot number becomes master. Slot numbers are automatically assigned depending on the order devices joined the session. The slot number is listed in the station table.

Hint:
Each station in a session has a priority. The session needs to have one station that is the master in the session. If there is only a station with the never priority, then the session is terminated since this station can never become the master.

- **Invite**:
  This On/Off button toggles **Invites**. Disabling this prevents this station from being invited into a session. It is still possible to join a session from this station, but it is an action that needs to be performed locally at the station.
- **Web Remote**:
  This On/Off button toggles the **Web Remote** access. Turning this off prevents web remote connections.
- **Remote HID**:
  This On/Off button toggles **Remote HID**. Learn more about this in the **Remote HID topic**.
- **RDM**:
  This is a global RDM On/Off button. Learn more about RDM in the **RDM section**.

- **Enable button** ():
  This button **enables or disables** the network connections. A green icon means that the station is connected. When the network is Off, then the button shows a small text telling if a new session is created or an existing one is joined, if the network is turned On.

Auto Option in MA-Net Interface Select Interface Pop-up.

The dropdown for selecting the interface used for MA-Net in the network menu has an **Auto** option.

When the MA-Net interface is set to Auto, the grandMA3 software determines the interface to use by itself. In this case, the IP of the selected interface is set into angle brackets on the MA-Net Interface button, for example, <192.168.0.4>.

The automatically determined interface in Auto mode is specified by the order of the following rules:

1. When a Class C IP address (192.168.x.y) is found, the interface with this IP will be taken. No matter the link state of this interface. (With onPC on Mac the interface link state must be active.)
2. When no Class C IP is found, the software searches for a Class B IP (172.16.x.y). Only if the interface has an active link state.
3. If also no Class B IP address is available, a Class A IP in the range of 10 (10.x.y.z) will be searched. Only if the interface has an active link state.
4. If this also fails, a Class A IP in the range of 2 (2.x.y.z) will be searched. Only if the interface has an active link state.
5. If no Class A IP is available, the loopback interface will be used.

It is always possible to change the selected interface from Auto to the preferred interface.

On grandMA3 consoles, Processing Units, and xPort Nodes, the first interface will still be selected as the default interface after a Full Install.

Per default grandMA3 onPC on Windows and macOS will be set to Auto.

The **Stations** view of this menu displays all the MA devices organized in types. The devices with the green background are connected devices.

This is a short description of the columns:

- **Lock**:
  A **Yes** in this column indicates that the row is locked. The fields show "S" if they are unlocked.
- **No**:
  This is the row number. Notice that a row can be a child of a parent object. If there is a number in parentheses, then it is the number of children in the parent object.
- **Name**:
  This is the name of the device or device type.
- **Type**:
  This shows the type for each device. OnPCs might be listed as "Undefined".

- **IP**:
  This is the IP address for the MA-Net interface for the device.
- **Session**:
  This is the name of the session the device is a part of.
- **Location**:
  This is the location for the session. Both session name, location, and key have to match for devices to connect.
- **Show File**:
  This is the name of the active show file in the device.
- **Status**:
  This is the session status for the device.
- **Prio**:
  This is the priority for the station.
- **Version Big**:
  This is the software version for the device. The version "big" number indicates the version number that needs to match for stations and processing units to connect in a session.
- **Version Small**:
  This is the software number that needs to match for grandMA xPortNodes to be able to connect to a session.
- **Enabled**:
  This shows if the network connection is enabled.
- **Online Time**:
  This is the time the device has been online in the network.
- **Session Index**:
  Each session is automatically assigned a number. This is the session number for each device. There is a maximum of 32 sessions in the same network.
- **Session Slot**:
  Each station in a session has a unique slot ID. This is the ID number for each device.
- **Remote IP**:
  Some MA devices are connected to a station. For instance the grandMA3 extension. This field shows the IP address of the station these devices are connected to.
- **Mask**:
  This is the Sub-Net mask of the station.

10.2.1. Create a Session

Requirement:

Network connections need to be made and the correct IP address should be set - For more information see **Interfaces and IP**.

Create a Session with the User Interface

1. Open the Network menu.
2. Tap the buttons at the bottom that needs to be edited.
   - Make sure that the correct interface is selected.
   - Make sure the Session name and Location values are correct.
   - Make sure the correct Key is selected.
3. If the network connection is disabled (the network icon in the **Command Line** is red), then tap the big enable/disable button in the lower right corner of the Network menu.
   - The network icon should become green.

Create a Session with the Command Line

1. Use the **JoinSession keyword** to join existing or create a new session.

10.2.2. Join a Session

Requirement:

Network connections need to be made and the correct IP address should be set - For more information see **Interfaces and IP**. There needs to be a running session that can be joined.

Join a Session Using the User Interface

1. Open the Network menu.
2. Tap the buttons at the bottom that needs to be edited:
   - Make sure that the correct interface is selected.
   - Make sure the correct Key is selected.
3. Tap the station with the desired running session.
4. Tap Join Session.
   - The network icon and the stations should become green.

Join a Session Using the Command Line

1. Use the **JoinSession keyword** to join existing or create a new session.

Restriction:
It is only possible to join a session if the joining station got the same key as the currently active key on the selected station (step 3 above) currently in the session.
This means that the station that wants to join a session selects a station already in the session. The station already in the session got a selected key. The joining station needs to have the same key selected to be able to join the session.

10.2.3. Leave a session

Requirement:

The station needs to be a session member to be able to leave the session.

Leave a Session Using the User Interface

1. Open the **Network menu**.
2. Tap Leave Session or turn the network Off using the network connection button.

Leave a Session Using the Command Line

1. Use the **LeaveSession keyword** to leave the session.

If there are other stations in the session with a priority above never, then the session is still active.

10.2.4. Invite a Station into a Session

Stations can be invited into a session if **EnableInvite** is turned On in the **Network Settings** on the station being invited.

Requirement:

A session needs to be running and the station being operated needs to be part of the session. See how in the **Create a Session topic**.

Using the User Interface

Stations are located and invited using the **Network menu**.

1. Navigate to the Network menu - see how, using the link above.
2. Locate and tap the station in the list of available stations (stations that are in the network, but not connected to a session).
3. Tap Invite Station at the bottom of the Network menu.

Using the Command Line

Stations are invited using the **Invite keyword**.

Restriction:
It is only possible to invite a station into a session if the joining station got the same key, in the key registry, as the one currently selected on the inviting station when the Invite button is tapped.

10.2.5. Dismiss Stations from a Session

Stations that are part of a session, can be dismissed by another station.

Requirement:

A session needs to be running and the station being operated needs to be part of the session. The station to be dismissed also needs to be in the same session. See how to create sessions in the **Create a Session topic**.

Using the User Interface

Stations are located and dismissed using the **Network menu**.

1. Navigate to the Network menu - see how using the link above.
2. Locate and tap the station in the list of stations that are connected to the session (stations with a green background color).
3. Tap Dismiss Station at the bottom of the Network menu.

The selected station is now kicked out of the session.

Using the Command Line

Stations are dismissed using the **Dismiss keyword**.

10.2.6. Create a Custom Key

Keys are passwords for sessions.

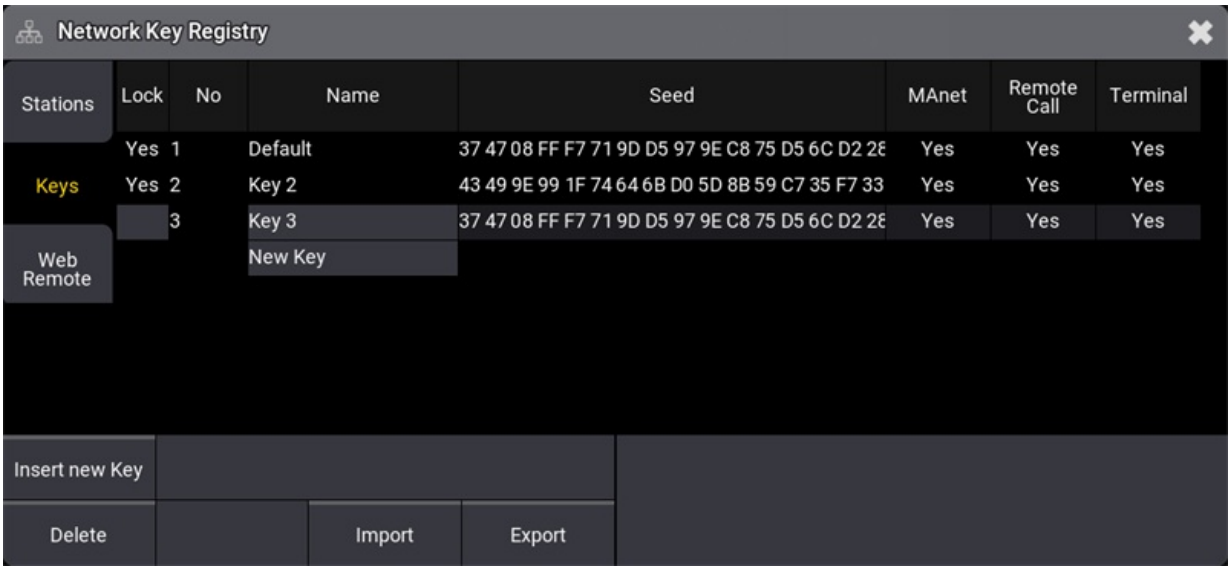
The same default key is a part of every new show file. It is used unless a custom key is created and used instead.

The key is a long hexadecimal hash value generated by a seed word or a set of characters.

Keys can be Inserted/Created, Deleted, Imported, and Exported. Locked keys cannot be edited or deleted.

Keys are selected when a **session is created**.

All the described key operations are done from the **Network Key Registry** menu.



Open the Network Key Registry from the Network menu

Open the Network Key Registry menu

1. Press the Menu key.
2. Tap the Network button.
3. Tap the Keys button on the left menu.

This opens a menu like the one in the image above.

There are a few important columns in the menu:

- **Lock**:
  This column indicates if the key is locked. The default key is locked as a default. When a seed is changed then the key is automatically locked. Edit this field to unlock a key.
- **No**:
  This is the key number.

- **Name**:
  This is a user-friendly name for the key. The name can be edited if the key is unlocked.
- **Seed**:
  The seed is a hash value. This can be edited on unlocked keys. Any text input can be used as a seed to create a hash value. The hash value cannot be manually input as this would create a new hash using the input as a seed. The original seed input needs to be known to be able to manually recreate the hash. The default value for a new key is the same seed value as the default key.
- **MAnet**:
  When this is set to **Yes**, then the key gives access to the MAnet area, this is for instance the possibility to join a session.
- **Remote Call**:
  When this is set to **Yes**, then the key gives access to the Remote Call area, this is for instance the possibility to use the webremote into a station, network updating the software, **RemoteCommand**, and other commands that affect other stations (for instance, reboot and shut down of other devices).
- **Terminal**:
  When this is set to **Yes**, then the key gives access to the Terminal area, this is for instance access to a station's terminal interface.

Important:
Please leave the **MAnet**, **Remote Call**, and **Terminal** columns set to "Yes" for the default key. Changing these might give unexpected results.

Using Keys in Sessions

The key is selected before the session is started. If a custom key is selected then other stations need to have the key in the registry list to be able to join or be invited into a session.

The keys can be exported to a USB memory stick and imported into other stations.

Or if the seed is known, then the custom key can be recreated on other stations using the same seed value.

If a station does not have the key that was used when the session was created, then it cannot be a part of the session. The joining station does not need to have the key selected in the **Network Menu**, but it needs to be in the **Network Key Registry** list.

Requirement for the next few operations:

The **Network Key Registry** menu is opened.

Create a new Key

Keys are created like most new elements.

1. Select the row where the key should be inserted.

2. Tap Insert .
3. (Optional) Edit the **Name** field to give the key a useful name.
4. Edit the **Seed** field and type a word or a random set of characters. This input is converted to a hash number.

Notice that the row is locked as soon as the seed field is changed. It can be unlocked and edited afterward. The lock is to prevent accidental change of the key.

Hint:
The key can be reproduced on other stations if the same seed text is applied.

Delete a Key

A locked key cannot be deleted. Follow these steps to delete a key.

1. Select the row with the key to be deleted.
2. Edit the **Lock** field to unlock the row - the field should be empty for unlocked.
3. Tap Delete .

The key is now deleted.

Export a Key

Keys can be exported and brought to other stations manually.

1. Select the row with the key to be exported.
2. Tap Export .
3. Select the desired drive in the pop-ups title bar.
4. Give the exported file a name.
5. Tap Export in the pop-up.

The key is now exported to the selected drive.

Import a Key

Keys can be imported into a station.

1. Select the "New Key" row.
2. Tap Import .
3. Select the desired drive in the pop-ups title bar.
4. Select the desired file.
5. Tap Import in the pop-up.

The key is now imported to the station.

10.3. Web Remote

The stations can be remote-controlled by any browser connected to the system. This includes WiFi-connected devices.

A maximum of five devices can connect to one station at the same time.

Each remote device connects to a specific station. The remote browser shows the same controls and encoder view at the bottom of the display as the grandMA3 Compact, grandMA3 Compact XT, and grandMA3 onPC display 1. The user-definable area is the same as screen 1 on any consoles or onPC. The web remote will try to log in as the user **Remote** when connecting and the view and windows on screen 1 follow the logged-in user. The web remote can log in as a different user than the one currently logged in on the connected station. Learn more about multiuser setups in the **Single User and Multi User Systems section**.

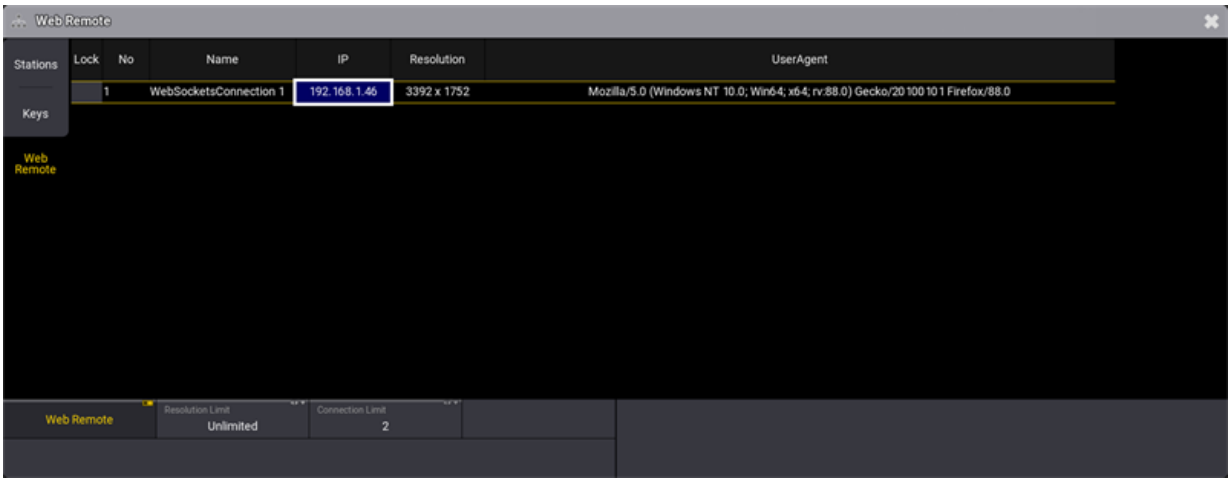
Any window or view that can be created on a console or onPC can be used in the remote. The system will try to provide a high refresh rate of the content, but it is affected by the connection speed and the resolution.

Connect with a Web Remote

Web remotes need to be enabled in the station. This can be done in the **Network Menu** - see the **Session topic** to learn how to access this menu.

In the network menu, there is a toggle button that turns the web remote On or Off.

The same setting is mirrored in the actual **Web Remote** menu that can be accessed by tapping Web Remote on the left side in the network menu.

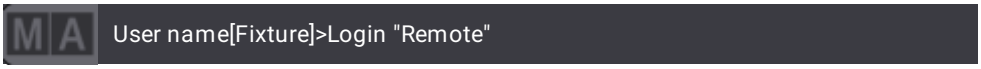


Web Remote menu with one connection

1. Make sure web remote is turned On in the station.
2. Open a browser on a network-connected device.
3. Type the IP address of the station and include **:8080** after the address to specify the port number.
   The result should be a browser window with a grandMA3 display.

Now there is access to remote control the station with the user rights provided by the logged-in user.

A new show file has a user called **Remote**. This uses the default user profile but with a different screen configuration. Remote is also a keyword so if it is desired to log in as this user then type the following in the command line:



Hint:
Putting the user name in quotation marks is important for the software to treat it as a name and not a keyword.

The Web Remote menu shows the connected devices. It is just information about the IP address, operating system, and browser.

There are three different settings that can be changed in the menu.

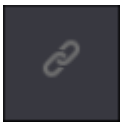
The remote connection can be turned On or Off.

A **Resolution Limit** can be set. There are options to limit it to 480p, 720p, 1080p, or unlimited. This limit is the same for every device that connects. If unlimited is selected, then the resolution is adopted to the device screen.

Connection Limit can be used to set a number between 1 and 5. This is the maximum amount of simultaneous connections allowed.

Linked Command Lines

There is a button on the connected remote device next to the host command line input.



This can be activated to link the remote command line input to the host command line input.

Linked command lines change the background color on both the remote and the host station. It could look like this:

Linked command line example

Notice that the command input in the image above is not a valid command. It is just an example.

Any command input on any of the linked command lines is immediately shown on all the linked command lines where the users are logged in with the same user profile. Any executed commands are executed on the host station.

10.4. SFTP Connection to a Console

The consoles can be accessed using an SFTP client program.

One such program could be FileZilla Client from the **FileZilla Project** (external link to internet webpage).

The SFTP client needs to be installed on a computer in the same network as the console.

To access the folder structure of the console, the IP address of the device is needed. A user name and password is also required. SFTP uses TCP port 22.

To access the grandMA3 area of the consoles hard drive please use **madata** as both username and password.

Hint:
The grandMA2 area of the grandMA3 consoles can also be accessed using SFTP. Please use **data** as both username and password to access this part.

WARNING:
Be very careful about making any changes in the folders or files. This is direct access to the device. Deleting or moving files may cause serious harm to the device.

Once the connection is made, it is possible to browse through the folders inside the device. The first level of folders has three folders. The "actual" folder gives access to the current software version. The "gma3_library" folder contains exported objects including screenshots. The "installation_packages" contains zip files with necessary installation files.

Learn more about the folder structure in the **Folder Structure topic**. Read more about screenshots in the **Screenshots topic**.

11. DMX In and Out

There are several ways to get DMX in and out of the grandMA3 system.

There are DMX ports on stations that can generate DMX as a standalone device.

DMX ports on grandMA3 devices that are a part of a session can also be DMX outputs and inputs.

This is set up from the Output Configuration menu. Read about it in the **DMX Port Configuration topic**.

DMX can also be transferred using standard network protocols. This is **Ethernet DMX**.

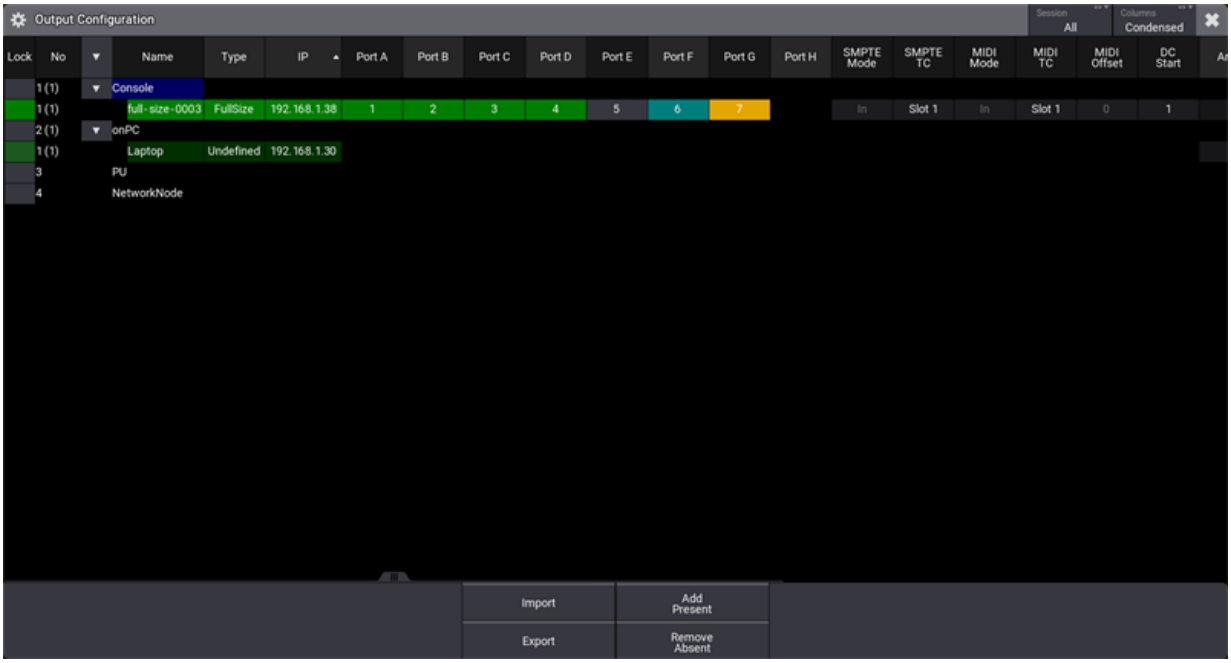
There are different rules for the different kinds of DMX.

Generally, each DMX port on a grandMA3 device can be an input or an output.

11.1. DMX Port Configuration

The DMX outputs and inputs are configured in the **Output Configuration** menu. They are labeled from **A** and alphabetically upwards.

There are also columns for MIDI, SMPTE, DC remote inputs, and Ethernet interfaces, but these elements are not described in this topic. Please read about them in the **Timecode section** and the **Remote In and Out topic**.



Output configuration menu with a console and an onPC

Access the menu by pressing the Menu key and then tap Output Configuration.

-OR-

Use the command line to open the menu:

```
User name[Fixture]> Menu "OutputConfig"
```

See **Ethernet DMX** for how to send and receive DMX using Ethernet.

The menu lists grandMA3 devices. There is a button in the title bar called Session. This filters the list to show **All** devices in the network or only the ones **In Session**.

The devices are organized in Console, onPC, PU, and NetworkNode sections that can be unfolded to see the devices in each category. The section can only be unfolded if there are devices in the category.

The menu has different modes for displaying columns. There is a button in the title bar called Columns. This has three modes:

- **Full**:
  This shows all columns.
- **Condensed**:
  This mode condenses the XLR columns to one for each port and still displays all other columns.
- **XLR Only**:
  This only shows the XLR ports in a condensed version.

DMX Ports

Each DMX port has four different settings (only visible when in **Full** mode):

Mode:

Each port can have four different modes:

- **Off** (gray background color in the cell):
  The port is turned Off.
- **Out** (green background color in the cell):
  DMX is sent out without RDM traffic. The universe is defined in the XLR field. This is the default mode.
- **RDM** (dark sea green background color in the cell):
  DMX is sent out and RDM is active. The universe is defined in the XLR field.
- **In** (yellow background color in the cell):
  DMX is received and merged into the universe specified by the XLR number. The merge uses the priority defined in Prio.

XLR:

This is a number input that defines what universe the port relates to. Fixtures can be patched in universe 1 to 1024.

Merge:

This option is only visible when the port mode is **In**.

The options are:

- **Off**:
  The incoming DMX is not merged.
- **Prio**:
  The priority set in the **Prio** column is used. The highest priority wins.
- **HTP**:
  The highest DMX value is used.
- **LowTP**:
  The lowest DMX value is used.

Prio (Priority):

The priority is used for merging DMX inputs. It is only used and can only be edited when the port mode is **In** and when the merge is set to **Prio**.

Editing this field opens a small select pop-up with the options.

The options are (from highest priority to lowest):

- **Super**:
  This priority is the LTP priority above any other playbacks and even above the programmer.
- **Prog**:
  The Programmer priority is between Super and the other priorities.
- **Swap**:
  Intensity is working as LTP with higher priority than HTP.
- **HTP (Highest Takes Precedence)**:
  The highest intensity value will be used. Other parameters will use LTP.
- **Highest**:
  Highest LTP priority - like LTP but higher than both LTP and High.
- **High**:
  High LTP priority - like LTP but a higher priority than normal LTP.
- **LTP (Latest Takes Precedence)**:
  This is the normal LTP priority. The newest attribute value is prioritized over the old value.
- **Low**:
  Low LTP - This is a lower LTP priority.
- **Lowest**:
  Lowest LTP - This is a lower priority than both LTP and Low.

Important:
The port configuration is stored in each device where it is set.

Import and Export

Each device keeps the port configuration. The configuration can be exported to an XML file and then imported again. This makes it possible to have different configuration files for different shows.

Export a Configuration

- Select the devices that need to be exported.
- Tap the Export button -> a file browser opens.
- Select the desired drive.
- Give the file a name and tap enter/please.

All or several devices can be exported into one file, or files can be created with single devices. It is just a matter of selecting one or several devices before the export.

Import a Configuration

- Select the desired devices.
- Tap the Import button -> a file browser opens.
- Select the desired drive.
- Select the files that have a configuration for the selected devices.

Add and Remove Devices

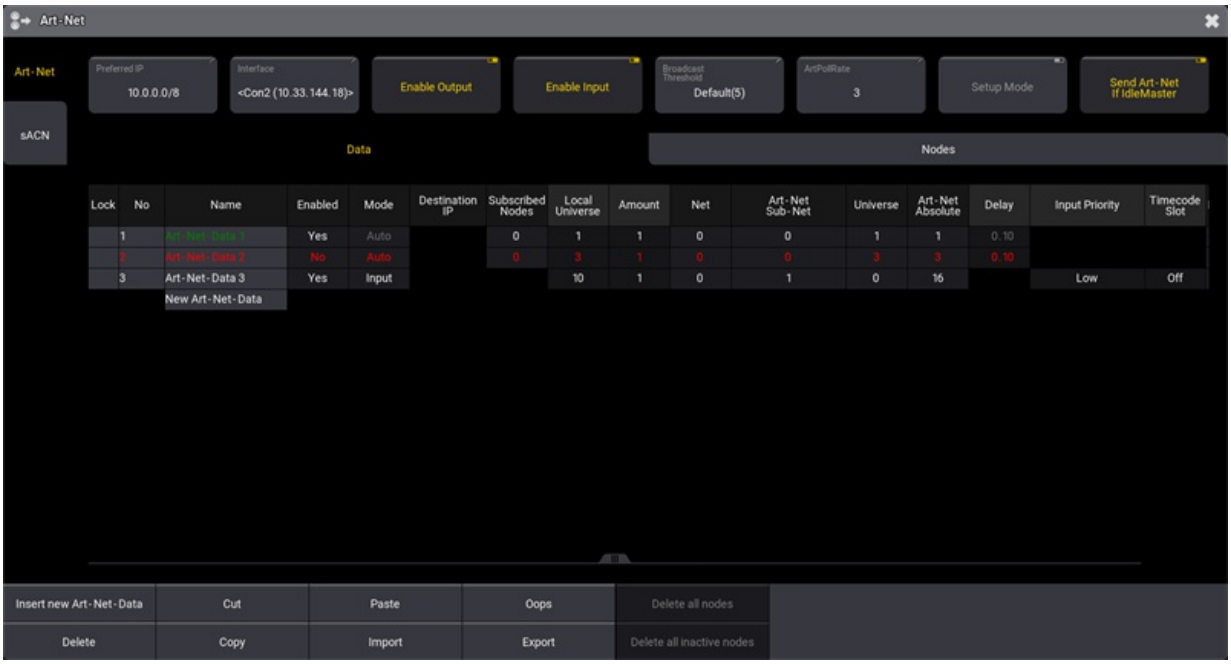
The list of devices is automatically updated with new devices that are present in the network.

Tapping Remove Absent removes devices from the list that are no longer in the network.

11.2. Ethernet DMX

There are currently two supported protocols for sending and receiving DMX using network: sACN and Art-Net.

Both are set up and changed in the **DMX Protocols** menu:



DMX protocol configuration for Art-Net

Access the menu by pressing Menu and then tap DMX Protocols.

Tapping the DMX Protocols button actually opens the menu for the first protocol. In the example above it is the Art-Net menu.

The menus can also be opened using the command line:

For Art-Net:

```
User name[Fixture]> Menu "ArtNet"
```

For sACN:

```
User name[Fixture]> Menu "sACN"
```

On the left side menu, there are two buttons. One for each menu.

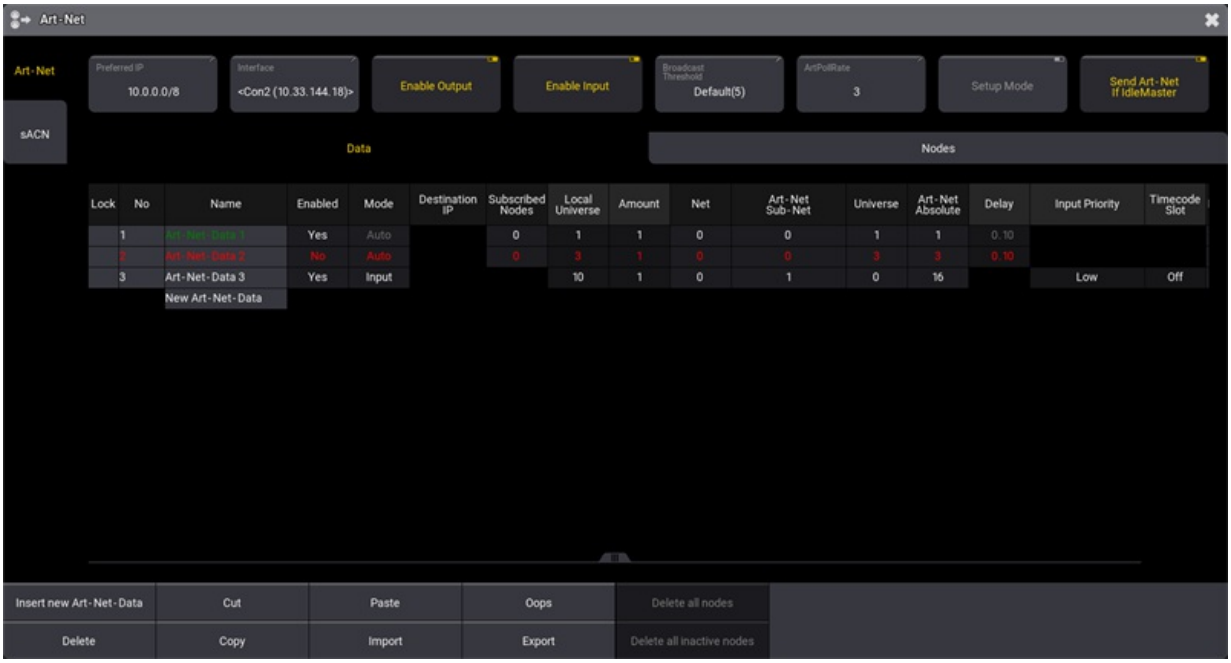
See the subtopics for information about the two protocols.

11.2.1. Art-Net Menu

Art-Net is a royalty-free protocol developed by Artistic Licence (**http://www.artisticlicence.com**).

grandMA3 supports Art-Net 4.

It is configured in the Art-Net menu:



Art-Net configuration menu

See the **Ethernet DMX topic** for information on how to open this menu.

See the **Transmit DMX using Art-Net topic** for an explanation of the steps needed to transmit Art-Net.

Config Buttons

There are some buttons at the top of the menu:

- **Preferred IP**:
  This is the preferred IP address or address range used by the Art-Net protocol. The Interface can be set to **Auto** allowing this setting to select the interface that matches the preferred setting. Tap this input button to open a small number input pop-up. The preferred number has to be input as an IP address with the subnet mask written in CIDR notation. For instance, 10.0.0.0/8 tells the system that it prefers an IP address that starts with 10. The rest of the numbers do not matter (the "/8" is the same as subnet mask 255.0.0.0).
- **Interface**:
  Tap this button to open the Select Interface pop-up where the desired network interface can be selected. This interface will be used for all Art-Net in and out. The **Auto** option can be used to allow the **Preferred IP** setting to select the interface. The interface is written inside "<" and ">" when it is selected by the preferred setting.

- **Enable Output**:
  This On/Off button needs to be On for Art-Net to be transmitted. Data also needs to be configured for output - read more below.
- **Enable Input**:
  This On/Off button needs to be On for Art-Net to be received. Data also needs to be configured for input - read more below.
- **Broadcast Threshold**:
  This input button sets the amount of Art-Net receivers for a universe before the master starts to send the universe as broadcast. This is only valid if the mode is set to **Auto**. If the amount of receivers is below the threshold, then the universe is sent as Unicast. If it is above, then the universe is sent as Broadcast.
- **ArtPollRate**:
  This input sets the time between each ArtPollRequest packet sent by the master station.
- **Setup Mode**:
  This On/Off button is used to toggle the setup mode. This mode can be used to transmit and receive configuration data only. If the output and input are turned Off then only the configuration data is transmitted and received.
- **Send Art-Net If IdleMaster**:
  This On/Off button defines if the station transmits Art-Net data when it is Idle Master. In a session, it is the Global master who transmits the network DMX. If the station is not in a session with other devices, then it is Idle Master. Turning this setting On will make the station output network DMX when it is Idle Master. This needs to be On if a single station is to output networked DMX. Learn more about standalone devices and networked devices in the **System Overview section**.

Important:

- Enabling setup mode allows Art-Net configuration data to be sent and received even when **Enable Output** and **Enable Input** is Off.
- When setup mode is Off and output is enabled, then DMX and configuration data are transmitted.
- When setup mode is Off and input is enabled, then DMX and configuration data are received.

There are two tabs below the buttons. They are **Data** and **Nodes**. Data is used to set up output and input. Nodes can be used to see the discovered nodes in the network. If the nodes support it, then this can also be used to configure the ports on the node.

Data Tab

The data tab is a grid of rows and columns. Each row is an Art-Net configuration. If the row is not valid or not enabled, then the name is in red text. When Art-Net is transmitted, then the text color flashes green.

This is a short description of the columns:

- **Lock**:
  The row can be locked to prevent changes.
- **No**:
  This is the row number.
- **Name**:
  Each row can have a name. This can be used as short info for the row.
- **Enabled**:
  This **Yes** or **No** field is used to enable the row transmitting or receiving Art-Net. No is the default.
- **Mode**:
  The mode defines what the row is doing. There are four options:
  - **Broadcast**:
    This transmits Art-Net using broadcast.
  - **Unicast**:
    This transmits Art-Net using unicast.
  - **Auto**:
    This transmits Art-Net. It uses the **Broadcast Threshold** number to determine if universes should be transmitted using broadcast or unicast. If the number of universe subscribers (determined by ArtPollRequests) are below the threshold, then it is sent using unicast. Is it above the threshold, then it is broadcasted.
  - **Input**:
    The row is receiving Art-Net and merging it into the defined universe.
- **Destination IP**:
  This field is only active if the unicast mode is selected. This is the IPv4 address of the receiving device.
- **Subscribed Nodes**:
  This field is information about the amount of subscribing devices. This is determined using ArtPollRequests.
- **Local Universe**:
  This is the grandMA3 universe to be transmitted or the universe that should receive incoming Art-Net DMX. If the amount is more than one, then this is the first universe in the range.
- **Amount**:
  This is the amount of grandMA3 universes to be transmitted or received.
- **Net**:
  The net number is a value bewteen 0 and 127. There are 128 different nets since Art-Net III. Each net is a complete group of **Sub-Nets** and **Universes**. This allows for addressing a total of 32 768 Art-Net universes. To be compatible with Art-Net I and Art-Net II devices, please use net 0.
- **Art-Net Sub-Net**:
  There are 16 sub-nets in Art-Net. They can be input in decimal numbers from 0 to 15 or 0 to F in hex numbers.
- **Universe**:
  There are 16 universes in each sub-net. They can be input in decimal numbers from 0 to 15 or 0 to F in hex numbers.

- **Art-Net Absolute**:
  This is the absolute universe number. It is calculated based on the net, sub-net, and universe numbers. It can also work the other way. A universe number can be input here and then the net, sub-net, and universe numbers are calculated based on the input.
- **Delay**:
  A delay can be set up between each transmitted universe. This can be helpful for older nodes with slower network cards. Sending many universes at once can flood the node. Adding a small delay helps. This setting is only available for outputting modes.
- **Input Priority**:
  This priority is the priority the received Art-Net packets are merged into the grandMA3 universe. This is no specific Art-Net setting. This is the same priority for normal DMX input and playbacks. This can only be changed when the mode is input.
- **Timecode Slot**:
  Any received ArtTimeCode is sent to the timecode slot number defined here. The mode does not need to be set to Input to receive the timecode, but Input needs to be enabled. This setting can only be changed when the mode is input. There are a total of 8 timecode slots. Read more about them in the **What are timecode slots topic**.
- **Enable RDM**:
  RDM via Art-Net can be enabled for the Art-Net universes specified in the row.

Nodes Tab

Each node detected is a row. Each node has one or more sub rows with "binds". Each bind can only have 4 ports. If a device has more than 4 ports then the device has multiple binds.

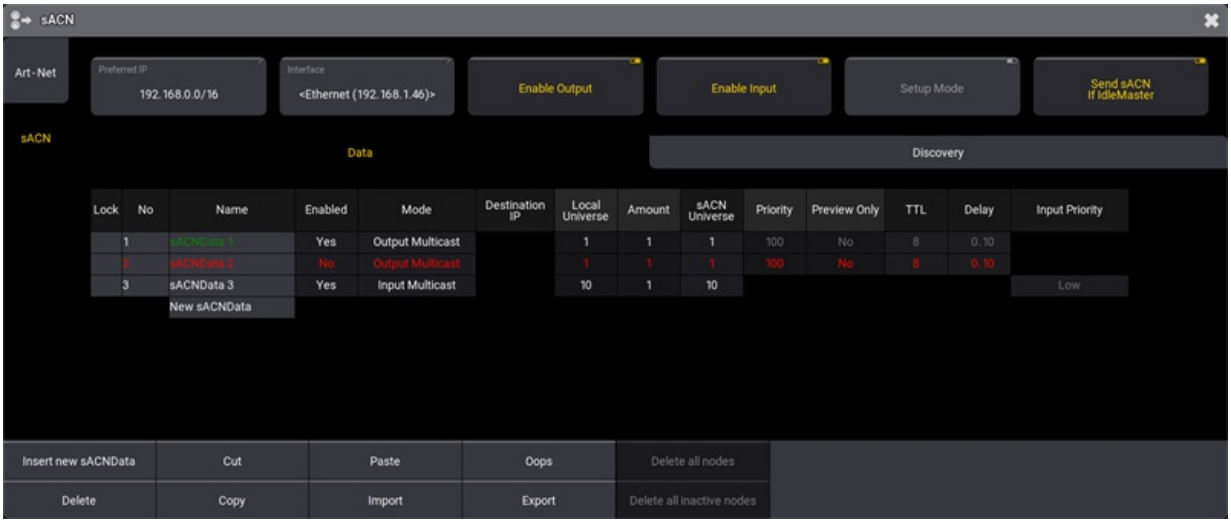
- **Lock**:
  The row can be locked to prevent changes.
- **No**:
  This is the row number.
- **Name**:
  Each row gets a name from the device. The name can be edited if the device supports it.
- **IP**:
  The IP address of the device. The address can be edited if the device supports it.
- **Art-Net Network**:
  Each bind has a net number. The number can be edited if the device supports it.
- **Art-Net Sub-Network**:
  Each bind has a sub-net number. The number can be edited if the device supports it.
- **Output Port**:
  This is the universe number of output ports on the device. The number can be edited if the device supports it.
- **Input Port**:
  This is the universe number of input ports on the device. The number can be edited if the device supports it.

11.2.2. sACN (streaming ACN) Menu

ACN (and streaming ACN) is an ANSI/ESTA international standard. Further readings: **http://en.wikipedia.org/wiki/Architecture_for_control_networks**.

ACN (Architecture for Control Networks) is a suite protocol. It uses a lot of elements that are currently not supported by grandMA3. But the ACN protocols also have a version for transporting DMX data. It is called 'Lightweight streaming protocol for transport of DMX512 using ACN' or more popular "streaming ACN" or "sACN". It is the international standard number E1.31.

It is configured in the sACN menu:



sACN menu

See the **Ethernet DMX topic** for information on how to open this menu.

Config Buttons

There are some buttons at the top of the menu:

- **Preferred IP**:
  This is the preferred IP address or address range used by the sACN protocol. The Interface can be set to Auto allowing this setting to select the interface that matches the preferred setting. Tap this input button to open a small number input pop-up. The preferred number has to be input as an IP address with the subnet mask written in CIDR notation. For instance, 192.168.1.0/24 tells the system that it prefers an IP address that starts with 192.168.1. The last number does not matter (the "/24" is the same as subnet mask 255.255.255.0).
- **Interface**:
  Tap this button to open the **Select Interface pop-up** where the desired network interface can be selected. This interface will be used for all sACN in and out. The **Auto** option can be used to allow the **Preferred IP** setting to select the interface. The interface is written inside "<" and ">" when it is selected by the preferred setting.

- **Enable Output**:
  This On/Off button needs to be On for the master to transmit sACN. Data also needs to be configured for output - read more below.
- **Enable Input**:
  This On/Off button needs to be On for the master to receive sACN. Data also needs to be configured for input - read more below.
- **Setup Mode**:
  This On/Off button is used to toggle the setup mode for nodes. This allows configuration data to set up the nodes, without sending sACN DMX data into the network.
- **Send sACN If Idle Master**:
  This On/Off button defines if the station transmits sACN data when it is Idle Master. In a session, it is the Global Master who transmits the network DMX. If the station is not in a session with other devices, then it is Idle Master. Turning this setting On will make the station output network DMX when it is Idle Master. This needs to be On if a single station is to output networked DMX. Learn more about standalone devices and networked devices in the **System Overview section**.

Important:

- Enabling setup mode allows sACN configuration data to be sent and received even when **Enable Output** and **Enable Input** is Off.
- When setup mode is Off and output is enabled, then DMX and configuration data are transmitted.
- When setup mode is Off and input is enabled, then DMX and configuration data are received.

There are two tabs below the buttons. They are **Data** and **Discovery**. Data is used to set up output and input. Discovery can be used to see the transmitting nodes in the network.

Data Tab

The data tab is a grid of rows and columns. Each row is an sACN configuration. If the row is not valid or not enabled, then the name is in red text. The name text flashes green when sACN data is transmitted.

This is a short description of the columns:

- **Lock**:
  The row can be locked to prevent changes.
- **No**:
  This is the row number.
- **Name**:
  Each row can have a name. This can be used as short info for the purpose of the row.
- **Enabled**:
  This Yes or No field is used to enable the row transmitting or receiving sACN. Yes is the default.

- **Mode**:
  The mode defines what the row is doing. There are four options:
  - **Output Multicast**:
    When choosing Output Multicast, sACN will be sent as multicast to the relevant multicast addresses.
  - **Output Unicast**:
    When choosing Output Unicast, a valid IP address has to be entered in the **Destination IP** column. Universes configured in this row will be sent as unicast to this IP address.
  - **Input Multicast**:
    Input Multicast will join the Multicast group of the relevant DMX Input Universe.
    Input Multicast is limited to max. 10 Universes. If more than 10 rows are configured as Input Multicast, all rows beyond multicast input row 10 will be invalid.
  - **Input Unicast**:
    Input Unicast is not limited and receives sACN data for the relevant universe without joining any multicast group.
- **Destination IP**:
  This field is only active if the output unicast mode is selected. This is the IPv4 address of the receiving device.
- **Local Universe**:
  This is the grandMA3 universe to be transmitted or the universe that should receive incoming sACN DMX. If the amount is more than one, then this is the first universe in the range.
- **Amount**:
  This is the amount of grandMA3 universes to be transmitted or received.
- **sACN Universe**:
  This is the sACN universe number the grandMA3 universes is transmitted to or the universe number that is listened to if Input is selected. If the amount is more than one, then this is the first universe in the range.
- **Priority**:
  The allowed value is 0 to 200. The highest number has the highest priority. The default value is 100. This priority is used for transmitted sACN.
- **Preview Only**:
  sACN data can be sent as preview data. This can, for instance, be used to send DMX to visualizers.
- **TTL** (Time To Live):
  Time To Live is a number used to tell routers and some switches how far through the network the sACN data should be transmitted. This is only relevant for output modes. The default value is 8 and this should usually not be changed.
- **Delay**:
  A delay can be set up between each transmitted universe. This can be helpful for older nodes with slower network cards. Sending many universes at once can flood the node. Adding a small delay helps.
- **Input Priority**:
  This is the priority of the received sACN. sACN input of grandMA3 ignores sACN priorities and uses this priority instead.

Discovery Tab

This tap displays the transmitting nodes in the network. Each node detected is a row. Each node has one or more sub rows with "pages".

- **Lock**:
  The row can be locked to prevent changes.
- **No**:
  This is the row number.
- **Name**:
  Each row gets a name from the device. The name cannot be edited.
- **Universe List**:
  This is a list of the universes the node transmits.

11.2.3. Transmit DMX Using Art-Net

This is a short example of setting up DMX transmission using Art-Net. Learn more about the Art-Net menu in the **Art-Net Menu topic**.

The idea in this example is some lights are patched in universe one to five. The fixtures are connected to Art-Net nodes and the nodes need to get the Art-Net data from the grandMA3. There is also a media server that uses a single universe. It is patched in universe 21 and needs to receive Art-Net universe 21 (hex number 1:5).

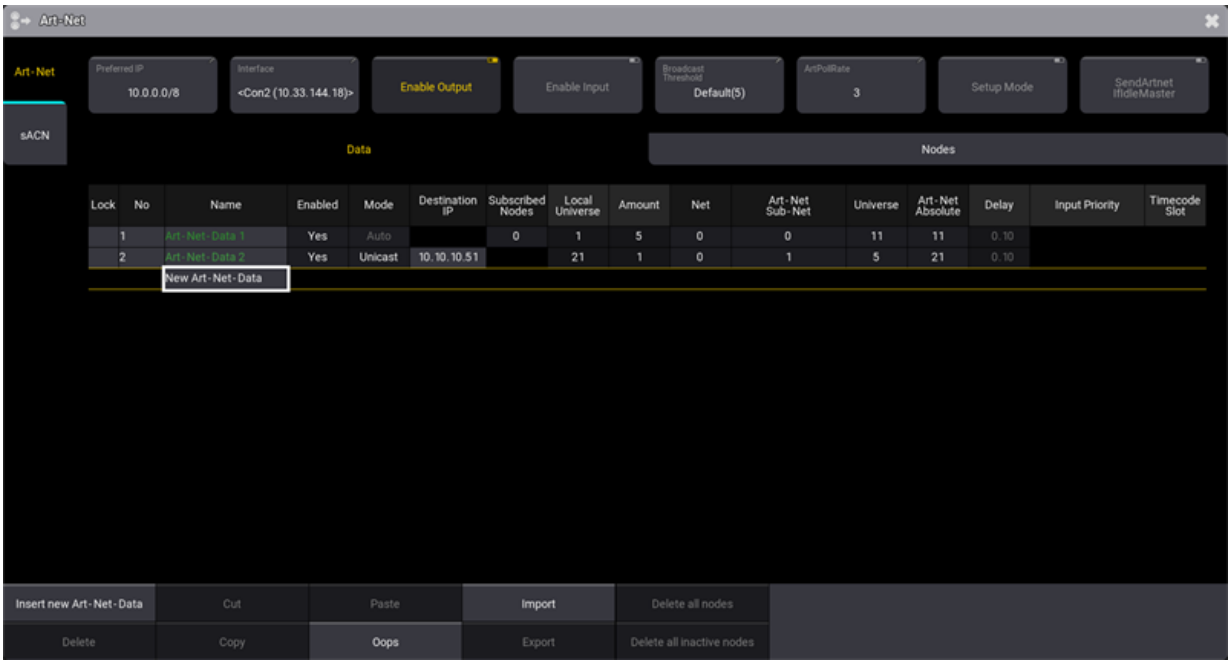
The Art-Net nodes are set up to receive from Art-Net universe 11 (hex number 0:B), so the grandMA3 universe one needs to be sent to this Art-Net universe. The media server is a specific device with a defined Art-Net IP address and this data can be sent using unicast to this device. In this example the media server IP address is set to 10.10.10.51.

The example assumes a patched setup that matches the universes. For demonstration purposes, this is not actually needed. If the grandMA3 onPC is used to try this example, then there might not be parameters to allow actual output.

These are the steps needed:

1. Open the Art-Net menu by pressing Menu and then tap DMX Protocols. Make sure the Art-Net menu is visible by tapping Art-Net.
2. The first line is going to be universe one to five. Edit the following cells to match these options:
    1. Enabled = Yes
    2. Mode = Auto
    3. Local Universe = 1
    4. Amount = 5
    5. Art-Net Absolute = 11
3. A second line is needed for the universe that needs to go to a specific destination. Tap New Art-Net Data below line 1.
4. Tap Insert new Art-Net Data in the button menu at the bottom of the screen.
5. Now the line exists. Edit the new line to match these options:
    1. Enabled = Yes
    2. Mode = Unicast
    3. Destination IP = 10.10.10.51
    4. Local Universe = 21
    5. Amount = 1
    6. Art-Net Absolute = 21 (notice that this automatically calculates the correct Art-Net Sub-Net and Universe number)
6. Finally, tap the Enable Output until it is On to activate the transmission of DMX data via Art-Net

That was it. It should look something like this:



Final Art-Net example setup

If the station is not connected to other stations or nodes in a session, then it is Idle Master. If it is to output Art-Net in a setup like this, then the **Send Art-Net if Idle Master** setting needs to be turned On. If the station is a session with other stations, then it is the Global Master in the session that transmits the Art-Net data.

The menu can be closed by tapping the cross ( ) in the upper right corner.

12. Single User and Multi User Systems

The grandMA3 system can handle small and big systems, with one or many users at the same time.

grandMA3 systems are often referred to as a **Session**.

An Ethernet network can have several sessions running.

A session can have one, some, or many MA devices as members or listeners.

This page is attempt to describe some of the differences between the two primary ways to have a system.

Single User Session

In a single user session there is one operator/programmer. This person might have one or several stations (consoles or onPC).

When there is only one user, then all stations are logged in as the same **User**.

This is also called **Full Tracking Backup**. The stations are all in sync and share the different views. The programmer content is the same on all stations.

This is useful for the single operator. If one station fails, then the operator can move to the other station and continue working without loosing anything.

This operator will usually have full admin rights to the entire system.

Multi User Session

In a multi user session, there are several operators. They might use the same stations and take turns (working in shifts or at different phases of the process). It can also be a system with multiple users working at the same time on different stations.

For this setup the operators need to create more **Users** and most likely more **User Profiles**. Each operator will then log into the station using their User and may control the complete system or just a part of the system.

The stations share the show file and the sequences are all in sync, but the users can have different views on the screens and their programmer is not shared - the output is.

Sometimes systems are a combination of multi users and full tracking backup; There are several operators (Multi User), but each operator has two (or more) stations logged in with their user (Full Tracking Backup).

In a multi user system, it is possible to have users with different operator rights. There are several levels of rights. They span from complete access to only being allowed to change the view.

12.1. Create User

A **User** is also a login name.

Some things are connected to the user. For instance, each user can have different operating rights and different languages. Read more in the **User Settings topic**.

The users are stored in a **User pool**. Read more about creating windows in the **Add Windows topic**.

The User pool with logged-in Admin user

The currently logged-in user has a yellow frame. It is the selected user.

Tapping one of the other users' logs in as this user. If the user has a password then a **Please Login pop-up** appears.

Please Login pop-up asking for password

Important:
The password is case-sensitive.

Type the correct password to log in.

There are always two default users in a show file: **Guest** and **Admin**. They use the same **User Profile**.

User Profiles are the ones that hold information about programmer content, selected sequence, current page, views, cameras, etc.

If two users share the same user profile then they share all the information stored in the user profile.

If the user profiles are different, then two operators can have different selections, programmer values, views. etc. - **Multi User System**.

Each **User** must have a **User Profile** assigned - Read more about this and how to create user profiles in the **User Settings topic**.

Create a new User

The easiest way to create a new user is to edit an empty object in the user pool. For instance, by using swipeys.

This opens the **Edit User pop-up**

Edit User pop-up with a new user

Here the name should be changed as a minimum. This is also the name used when logging in using the **Login keyword**.

The big area on the right side of the pop-up shows the selected Appearance for the user (the image above shows an appearance).

The rest of the settings in the editor is explained in the **User Settings topic**.

12.2. User Settings

There are a lot of settings connected to the **User** and the **User Profile**.

User Settings

There are eight different user settings.

- **Name**:
  This is the name of the user. This is the login name used with the **Login keyword**.
- **Scribble**:
  A user can have a scribble assigned. The scribble is only used in the **User pool**.
- **Appearance**:
  A user can have an appearance. The appearance is used in the user pool and when the station is locked.
- **Password**:
  This is an optional password that can be set to the user. It needs to be provided when the user is logging in.
- **Profile**:
  Each user has a User Profile assigned. Read more **below**.
- **ScreenConfig**:
  Different screen configurations can be created. This is the configuration called when the user logs in.
- **Rights**:
  There are six different levels of rights. Read more **below**.
- **Language**:
  The user can have a language setting. This is used to select one of the available languages. Nothing is currently translated, so changing the language does not change anything.
- **Keyboard**:
  The on-screen keyboard can have different layouts. This setting can be used to select one of the available layouts.

There are two places in the GUI where the settings can be viewed and edited. The settings are the same.

The first place is the Users pool. Editing a pool object opens an editor where the settings can be changed. Learn more about the Users pool in the **Create User topic**.

The second place is the **User Configuration**. Navigate to it using these steps:

1. Press the Menu key.
2. Tap the Settings button.
3. Tap the User Configuration button.

This opens the list of users in the show. Edit any field to change the setting.

User Profile Settings

The user profiles contain most of the settings relevant to the users. The programmer information, views, selected elements, values, preferences, and much more.

Many of these elements are stored and selected by using the software. There are some settings that can be set in the User Profiles menu.

Navigate to the menu:

1. Press the Menu key.
2. Tap the Settings button.
3. Tap the User Configuration button.
4. Tap the Profiles button on the left menu.

There are some settings here that can also be changed relevant places in the software. For instance, the value readout setting can be changed in the **Encoder Bar**.

- **Name**:
  This is the name of the user profile.
- **DMX Readout**:
  This is used to change the readout of DMX values. This is useful when editing **fixture types**.
- **Normal Value**:
  This is the intensity value used when the **Normal keyword** is used - typically by pressing the **At key** twice. It is a DMX value, so the DMX readout setting affects how to input a value in this field.
- **Wheel Resolution**:
  This can be used to change the resolution of the wheels on the consoles. The options are: Coarse, Normal, and Fine.
- **Wheel Mode**:
  This setting change how the wheels work. Read more **below**.
- **Precise Edit**:
  This function is used when editing values in sheets. It can be difficult to hit a field precisely in a sheet with fingers. Turning this Off makes it easier to select a field without accidentally creating a new selection. Learn more about this in the **Gesture topic**.
- **Screen Encoder**:
  When this option is enabled, the rightmost dual encoder becomes a screen encoder. Turning the inner encoder moves the focus in a vertical direction while the outer encoder scrolls in a horizontal direction. To create a lasso selection in a grid, like in the patch menu, press, hold and turning the inner encoder. To edit the currently focussed object, press and release the inner encoder or the dual encoder key.
  The screen encoder label in the encoder displays also on which display the encoder is currently active. This option is enabled by default.
- **Time Key Target**:
  The **Time key** can have two different targets: Cue or Fixture. This defines if the key defaults to keywords relating to cue timing or fixture layers.

- **TCSlot**:
  This displays the currently selected timecode slot. Read more in the **What are Timecode Slots** topic.
- **Value Readout**:
  This is the default value readout. Many sheets can be set to show a readout, this can be a specific readout type or it can follow this default type. This setting can be changed in the **Encoder Bar**.
- **Speed Readout**:
  This is the default speeds readout. The options are Hertz, BPM, and Seconds. Speed is used in **Phasers**.
- **Preset Readout**:
  This changes the way presets are displayed in Fixture and Sequence Sheets. The preset can be displayed with a combination of three elements:
  - **ID**: The ID number of the preset.
  - **Name**: The name of the preset.
  - **Value**: The values in the preset.
- **Overlay Fade**:
  This time sets a fade time used by pop-ups and menus in the user interface. The default time is 250 ms.
- **Time Readout**:
  This is the default time readout for the user profile. Read more about the readout **below**.
- **Frame Readout**:
  This is the default frame readout for the user profile. Read more about the readout **below**.
- **Undo Programmer**:
  This setting is used to set if programmer changes should be oops-able.
- **Undo General**:
  This setting is used to set if general operations (for instance, view changes) should be oops-able.
- **Auto Remove Gaps**:
  This setting is for the selection grid. Please read the **Selection Grid** topic for more information.
- **Mirror SpecialExecutor Pages**:
  This setting is for the custom section of the **grandMA3 extension**. If the setting is "Yes" then the extension has the same assignment as the custom section of the station the extension is connected to. If the setting is "No", then each extension can have its own assignments in the custom section. grandMA3 extensions with the same WingID are always mirrored - they are essentially defined as the same. Learn more about connecting grandMA3 extensions in the **Connect grandMA3 extension topic**.
- **Show Appearance In Cue Input**:
  This setting is used to define if the cue appearances are shown in pop-ups where cues can be selected. For instance, the pop-ups that appear using the **Goto** and **Load** commands without a specified target.
- **Show Settings In Editors**:
  Toggling this shows or hides the settings in some editors. This setting is also in the title bar of the editors where this can be toggled.
- **Exec Config [Object_Type]**:
  There are settings for each object type that can be assigned to executors. These configurations can be found in the **Executor Configurations**. Each object type has a selected executor configuration.

- **Show Connectors**:
  This setting shows or hides the Connector Overlay. Learn more about the Connector Overlay in the **Output Configuration topic**.

User Attribute Preferences

From the profile setting, there is access to the **User Attribute Preferences** by tapping Edit Encoder Bar.

This is used to set the user profiles preferred readout type for the natural readout. It also has different resolution and encoder press resolution multipliers.

Resolution multipliers define the factor by which an encoder is changing a value when the encoder is pressed or in the case of the dual encoders, the difference between the inner and outer ring.

At the top there are four buttons:

- **Time Layer Resolution**:
  Sets the encoder resolution for the time-related layers.
- **Phase Layer Resolution**:
  Sets the encoder resolution for the phaser-related layers.
- **Dual Encoder Factor**:
  This defines the multiplier used by the outer encoder ring in relation to the inner encoder ring.
- **Dual Encoder Press Factor**:
  This defines the multiplier for the outer encoder ring when the dual encoder key is pressed while turning the outer encoder ring.
- **Link Resolution**:
  This setting defines how the encoder resolution is linked between features. The options are **Single** and **Feature Group**. Learn more about this setting in the **Encoder Toolbar topic**.

The rest of the pop-up is a list of all the attributes in rows.

There are three different columns with values that can be changed for each attribute:

- **Natural Readout**:
  The Natural readout allows defining the desired readout type per attribute. For instance, dimmer values are at best displayed in percentage readout, while pan or tilt values are better readable using the physical readout which displays the pan and tilt values as degrees. This setting is used to select the desired readout when Natural is selected in sheets and encoders.
  The user-defined readout has a higher priority than the readout defined for the attribute. The user-defined readout can be linked to the default (from the attribute definition). This option is called Default and the actual readout is shown in angle brackets.
- **Encoder Resolution**:
  The encoder resolution defines how big the change of an attribute value will be when turning a dual encoder by 1 click. This setting can also be changed by pressing MA and tapping the channel function area in the encoder toolbar. Learn how in the **Encoder Resolution topic**.

- **Encoder Press Factor**:
  This is the multiplier used when the inner encoder is pressed and turned for this attribute.

Learn more about the different resolutions available in the **Encoder Resolution topic**.

Wheel Mode

There are four different level wheel modes:

Additive (= default):
Additive keeps the difference between dimmer values until they reach 0% or 100%, using the level wheel. After 0% or 100 % are reached, the values will be leveled out.

Example:
Fixture 1 has a value of 50 and fixture 2 is at 60. When the level wheel is used to take them both up to 100, and then back down, they both go down from 100 at the same time and the same level.

Incremental:
Incremental keeps the difference between the dimmer values always, even if you reached 0% or 100% by using the dimmer wheel.

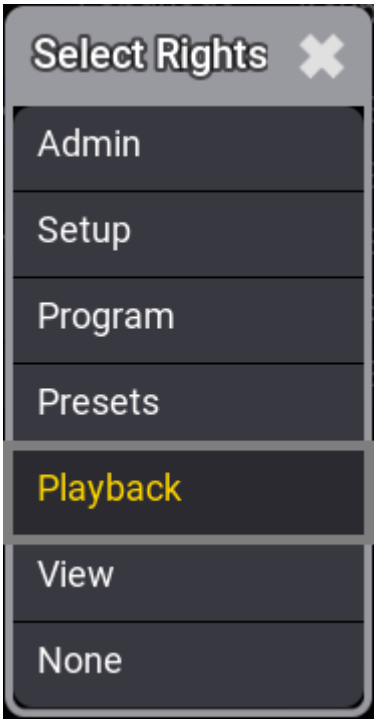
Example:
Fixture 1 has a value of 50 and fixture 2 is at 60. When the level wheel is used to take them both up to 100, and then back down, fixture 1 will start coming down first and then fixture 2 will follow when fixture 1 is at 90.

Prop.+ (Proportional positive):
When using the level wheel to turn up the dimmer values, the difference in the dimmer values will **decrease**.
Turning up to 100% will make all channels reach 100% at the same time.

Prop.- (Proportional negative):
When using the level wheel to turn up the dimmer values, the difference in the dimmer values will **increase**.
Turning down the values will make all channels reach 0% at the same time.

User Rights

There are seven different levels of user rights in the system. They are a user setting - read more above.



Select Rights pop-up is used to select one of the rights levels

The settings are:

- **Admin**:
  This is the right to change everything in the console, system, and show.
- **Setup**:
  This will limit access to some of the elements in the console. There are other console settings that can be accessed.
- **Program**:
  At this level, the user cannot do major changes to the patch. It only gives access to the "Live Patch". Most programming operations can be done.
- **Presets**:
  This level allows for updating existing presets. But the user cannot edit the cue content.
- **Playback**:
  This level allows playback and running a programmed show. But the user cannot store anything.
- **View**:
  With this user right, it is not allowed to use a programmer. The user is allowed to call views and log in as a different user.
- **None**:
  The user is only allowed to log in as a different user.

Time and Frame Readout

The applied values are used most places the time is displayed.

Hint:
The Readout can be changed for individual sheets in the **window settings**.

There is a hierarchy of the readout/format settings. The default is set in the user profile, if this default is changed in a window then the window setting is used.

Timecode is an example of an area that has many layers in the hierarchy. Read more about timecode in the **Timecode section**.

Hint:
The user profile defined time readout is used in the encoder bar even if the readout for an, for instance, fixture sheet is changed separately.

The timing calculator value indicator is aligned with the preselected frame readout.

To provide better optical representation, the trailing zeros from frames or seconds are always suppressed. When the frame readout is set to a frame unit the times will always display trailing zeros.

To easily distinguish between fractions of a second and frames, fractions of a second are separated from second using a dot (.), while frames are separated from seconds using a colon (:).

The time options are:

- **10d11h23m45**:
  The time is separated into days, hours, minutes, and seconds using letters as separators.
- **251h23m45**:
  This is separated into hours, minutes, and seconds using letters as separators. The hour number can become more than 24 if time is more than a day.
- **10.11:23:45**:
  The time is separated into days, hours, minutes, and seconds using a dot and colons as separators.
- **251:23:45**:
  This is separated into hours, minutes, and seconds using colons as separators. The hour number can become more than 24 if time is more than a day.

The frame options are:

- **Seconds**
- **24 fps**
- **25 fps**

- **30 fps**
- **60 fps**

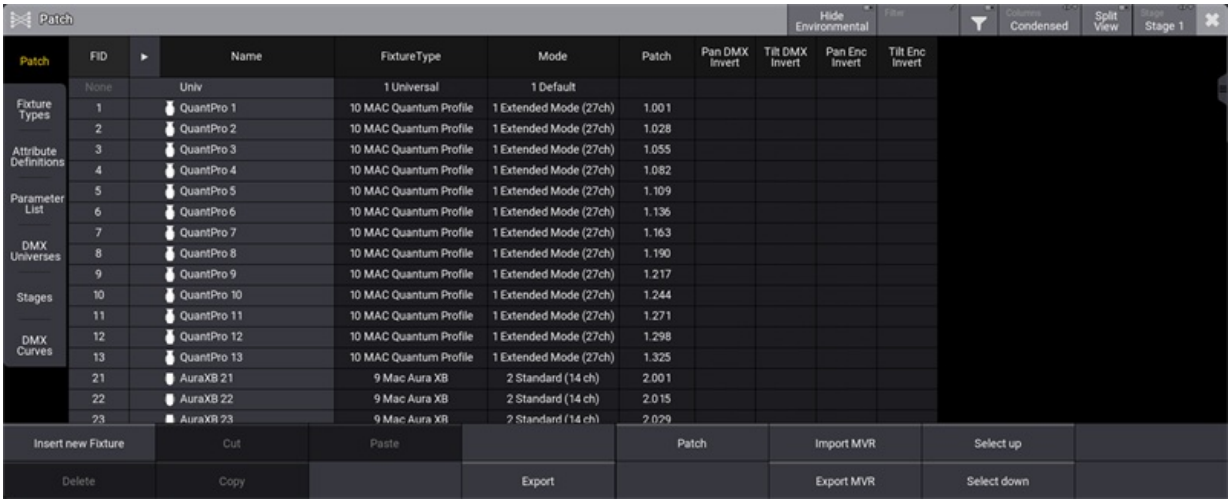
Hint:
If frame readouts (24, 25, 30, 60 fps) are used, fractions are separated by a colon, fractions of seconds are divided by dots.

Fps means frames per second.

13. Patch and Fixture Setup

Fixtures need to be added to the show file before they can be controlled or **operated**.

Fixtures are added to the show using the **Patch** menu.



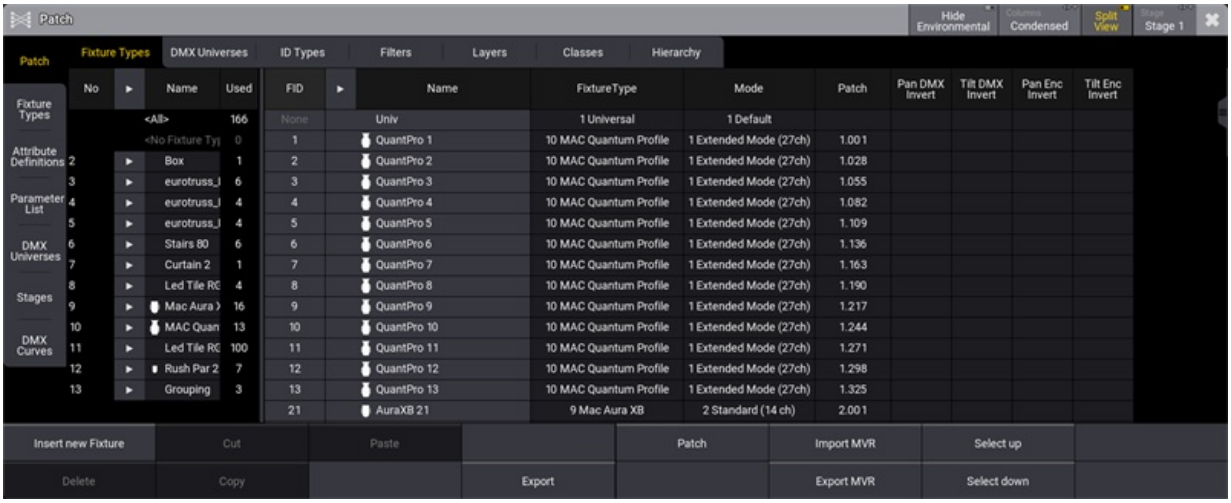
The patch menu with some fixtures patched

The patch menu gives access to **Fixture Types**, **Attribute Definitions**, **Parameter List**, **DMX Universes**, **Stages**, and **DMX Curves**.

The patch menu has two different modes. The image above shows the **Condensed** mode. This mode hides a lot of columns that might not be needed. The other mode is called **Full**. This shows all columns in the patch menu. The mode can be toggled by tapping Columns in the title bar.

The fixture list can also be filtered using the filter settings in the title bar or by activating Split View.

Split view filters the fixtures by different column properties.



The Patch menu in Split View

The different split options appear as tabs that can be selected to filter the fixture list.

The list of fixtures is separated into two sides. The left side is a list of the different elements of the selected split filter. The right side is a list of the fixtures that qualify based on the object selected on the left side.

The tabs and split options are:

- **Fixture Types**:
  The left side lists the different imported fixture types including <All> (all fixtures) and <No Fixture Type> (fixtures without an assigned fixture type). Learn more about fixture types in the **Fixture Types section**.
- **DMX Universes**:
  The left side lists the different DMX universes including <All> (all universes) and <Unpatched> (fixtures without an assigned DMX address). Learn more about DMX universes in the **DMX Universes topic**.
- **ID Types**:
  The left side lists the different defined ID types including <All> (all fixtures) and <No ID> (fixtures without an assigned FID or CID). Learn more about ID types in the **What are Fixtures topic**.
- **Filters**:
  The left side lists the different filters in the Filter pool including <All> (all fixtures). Learn more about filters in the **World and Filters section**.
- **Layers**:
  The left side lists the different defined layers including <All> (all fixtures) and <No Layer> (fixtures without an assigned layer). Learn more about layers in the **Classes and Layers topic**.
- **Classes**:
  The left side lists the different defined classes including <All> (all fixtures) and <No Class> (fixtures without an assigned class). Learn more about classes in the **Classes and Layers topic**.
- **Hierarchy**:
  The left side lists the hierarchical structure of the patch. It lists the parent elements and can be unfolded. The right side displays the elements directly dependent on the selected object on the left side.

13.1. What are Fixtures

Fixtures are the different devices that can be controlled by the grandMA3.

Fixtures are added using the patch menu. Each fixture is a row in the patch menu. Each row has some settings organized in columns. Two of the columns are **Fixture Type** and **Mode**. Fixture Type is the definition or description of the physical fixture and the DMX definition for the fixture might be separated into different modes. Each fixture type can have several different modes. Only one mode can be selected for each fixture in the patch. Read more about editing and creating fixture types in the **Fixture Types section**.

Fixtures contain different **Attributes** or **Parameters**. The parameters can have different values. Individual parameters are the ones manipulated when controlling the fixtures' attributes. For instance, changing the dimmer attribute for ten fixtures changes the dimmer parameter on each of the ten fixtures. Read more about attributes in the **Attribute Definitions topic**.

Fixtures in the grandMA3 are selected using an ID. There are ten different **ID Types**. These ID Types can be used to organize the fixture list. Four of the ID Types are **Universal**, **Fixture**, **Channel**, and **MArker** - they are locked and cannot be renamed. **Universal** is a fixture type that is automatically created and should not be edited manually. The other six ID types can be renamed to match the needs of the show. These are called **Custom ID** because they can be customized. The default names are **Houselights**, **NonDim**, **Media**, **Fog**, **Effect**, and **Pyro**. Each of these is a keyword. Renaming them changes the corresponding keywords. Read about the ID type keywords following these links: **Fixture**, **Channel**, **Universal**, **Houselights**, **NonDim**, **Media**, **Fog**, **Effect**, **Pyro**, and **MArker**.

A fixture can have a **Fixture ID** (FID) and one other **ID Type** (CID) with the same or a different ID number. A fixture needs at least one ID to be **operated**.

Fixtures are added to a stage. In a new empty show, there is a standard stage (Stage 1). More stages can be added to organize the setup. Read more about Stages in the **Stages topic** and about placing fixtures in the **Position Fixtures in the 3D Space topic**.

Fixtures can have **Class** and **Layer** information. These are two different ways to organize or group the fixtures. Read more in the **Classes and Layers topic**. The patch menu can be filtered to hide or show fixtures, for instance, using specific classes or layers.

Another way to organize the patch is to add fixtures as a "child" (structurally below a "parent") to another fixture. A special **Grouping** fixture is very useful for this. It is an "empty" fixture in the generic manufacturer group. This can be useful if there are many fixtures of one type, for instance, 100 LED panels. A grouping fixture would be added to the patch and the LED panels are added inside or below the grouping fixture. The grouping fixture can have an ID. Selecting this ID then controls the LED panels inside or below the grouping fixture. An example of this structure can be seen in the demo showfile.

Fixtures also need to be patched to a DMX address in the grandMA3. If it does not have a DMX address, then the system does not know where to send the parameter values.

Example 1

A row is added to the patch menu. A Robe MegaPointe is assigned to the row. Now the fixture exists in the patch menu. The fixture needs an ID to be able to be selected and controlled. The fixture gets FID (Fixture ID) 1. Now the fixture can be selected using the **Fixture 1** command. For the fixture to react it needs to have the same DMX address as set in the fixture itself and it needs to be connected to a DMX port set to the correct universe. The fixture gets address 3.54. The universe number is 3 and the DMX address in that universe is 54 - this needs to match the fixture setting.

Example 2

A row is added to the patch menu. A generic dimmer (8 bit) is assigned as the mode. The fixture is a part of the building and is used for house lights. The ID Type is changed to **Houselights** and CID (Channel ID / Custom ID depending on the ID type) is changed to 101. It does not get an FID. It can now be selected by pressing Channel three times to get the houselights keyword and the numbers 1 0 1.

The fixture gets the Patch address 1.101 (universe 1, address 101).

Hint:
The channel key is used to get all other ID types than **Fixture**. Pressing it multiple times toggles through the ID types.

13.2. Add Fixtures to the Show using the GUI

This is the process for adding new fixtures (or devices) to the show and the patch, using the GUI.

Quick Steps in a new show:

1. Open the Patch Menu.
2. Follow the steps in the wizard:
   2A. Select the fixture type.
   2B. (optional) Add a custom name.
   2C. Type the quantity.
   2D. Type the first ID number.
   2E. (If in **Full** mode) Select a Layer and class (**None** is an option).
   2F. Type the patch address for the first fixture.
   2G. Tap Apply.
3. Close the Patch menu and tap Ok to save the changes.

These are steps needed for adding the first fixtures to a new show - read below for details.

Quick Steps in a show with existing fixtures:

1. Open the Patch Menu.
2. Select the row where the new fixtures should be inserted and tap the Insert New Fixture button.
3. Make sure the desired source is selected.
4. Select the desired fixture type (use the filter to limit the list) and tap Select.
4. Fill in the rest of the fields in the wizard and tap Create !.
5. Edit any desired field in the patch grid before closing the Patch menu.
7. Close the Patch menu and save the changes.

These are steps needed to add more fixtures in an existing show - read below for details.

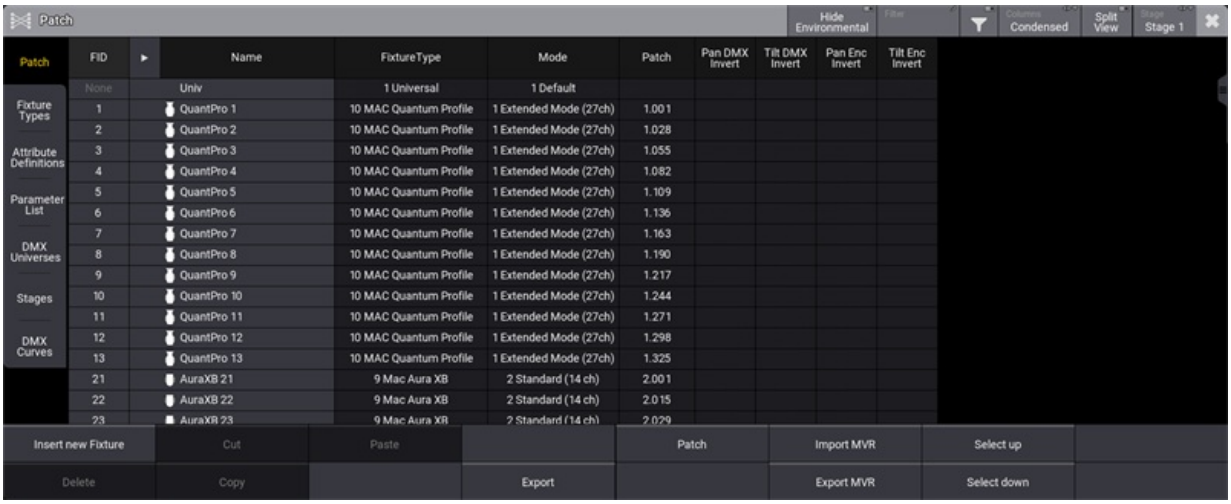
Everything about adding fixtures is done from the Patch menu.

Navigate to the Patch Menu

The Patch menu needs to be open to add fixtures.

1. Press the Menu key.
2. Tap the Patch button in the menu pop-up.

The patch menu is now open.



The open Patch menu - in condensed mode - with some fixtures

When the menu is opened the first time, then a wizard helps to add the first fixtures to the show and instead of the patch menu, there is a guide through the fixture selection and the **Insert New Fixtures** pop-up (see below).

Insert a Device in the Patch

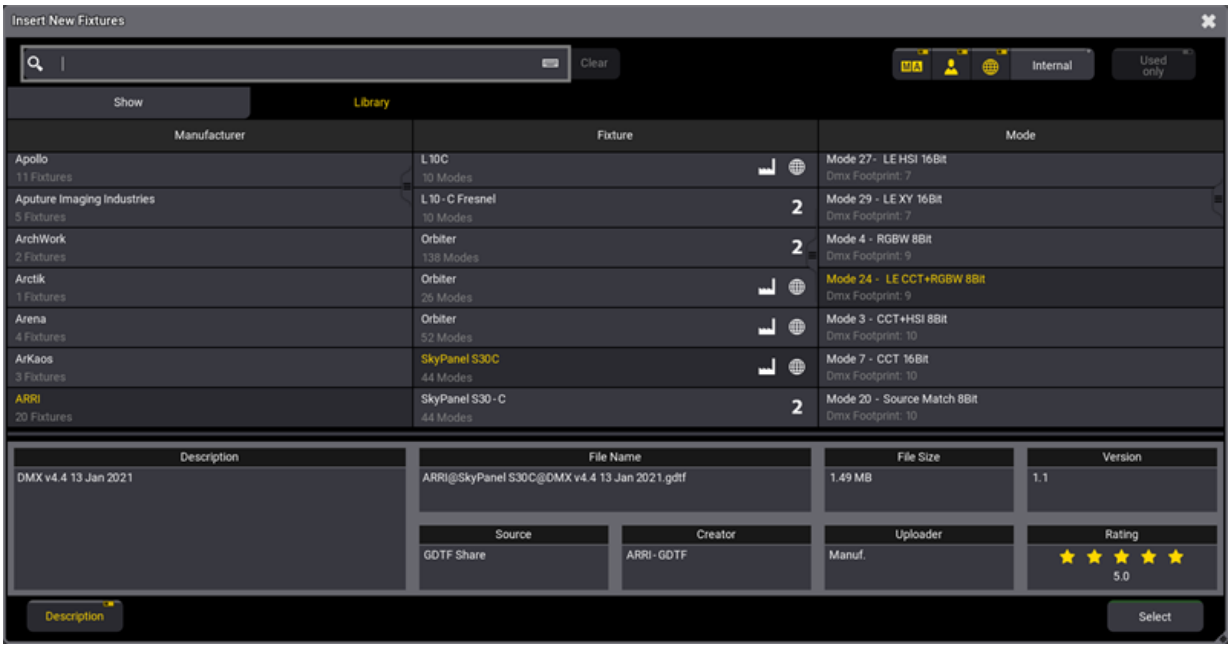
Each fixture needs a row in the patch menu.

The fixtures belong to a "parent" stage object. The default parent is **Stage 1**. Several stages can be created - read more in the **Stage topic**.

The selected stage can be changed using the stage button in the title bar.

1. Select the stage where the fixtures should be added using the button in the title bar.
2. Tap the New Fixture area on the list if it needs to be added to the bottom of the list or tap an existing fixture to insert new fixtures above the selected fixture.
3. Tap Insert new Fixture.

This opens the **Insert New Fixture** wizard pop-up on the **Select DMX Mode to use** part.



Select a fixture in the libraries

Just below the search bar, there are two tabs: **Show** and **Library**. Show is the fixture types already imported into the show file. This list can be limited to only showing the fixture types that are patched. This can be done by activating the Used only button. The **Library** tab lists the different fixture type that can be used. With library selected, there are extra buttons available on the right side.

One button selects the drive. This makes it possible to select the internal drive, previously installed versions library, or an external USB drive. In the image above it is the Internal button. The button is labeled to show the selected drive. Next to this button, there are four small toggle buttons that select different sources.

The sources are MA (grandMA2 converted fixture files and grandMA3 fixture files), User, and Shares (GDTF share and grandMA3 fixture share). The share button is only available when there is an active connection to a World Server (learn more in the **World Server topic**).

Restriction:
grandMA2 fixture files cannot be loaded directly into the grandMA3. The grandMA2 converted source mentioned above is the grandMA2 library converted to match the grandMA3 structure. Fixture files are converted when a grandMA2 file is stored as a grandMA3 file. Learn more in the grandMA2 manual / Using the Backup Menu / Save as grandMA3.

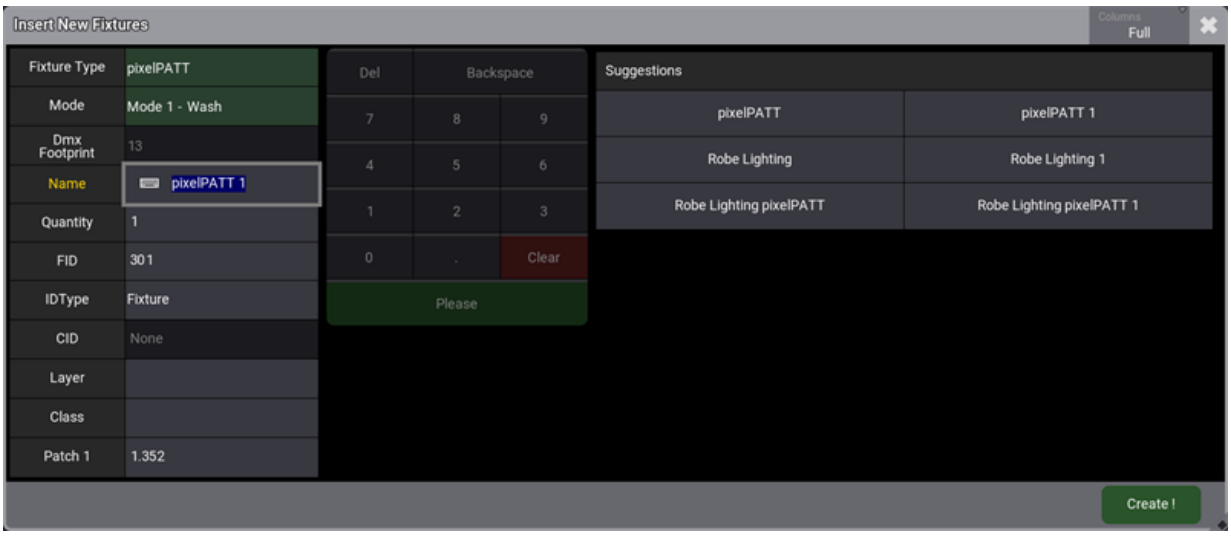
4. Select the desired drive and activate the desired sources.

The fixture libraries are sorted by manufacturers in the left column. Selecting a manufacturer lists the fixtures from that manufacturer in the center column. Selecting a fixture lists the available modes in the right column.

At the bottom, there is a Description button that can show an area with extra information about the selected fixture type file.

5. If the list is long, then it might make sense to filter it by typing a word in the **Search** input. The search is across manufacturers, fixtures, and modes (If a fixture has multiple modes, it will show them all, but only fixtures that include the searched mode).
6. Select the desired fixture and mode from the list and tap Select.

This reveals the **Insert New Fixture Wizard**. This has a list of elements that needs information to add the fixtures to the patch. The wizard comes in two different modes, depending on the **Columns** mode of the patch. This mode can be toggled in the Columns button in the title bar. When the mode is **Full** then there are more elements (ID Type, CID, Layer, and Class). This is the full version of the pop-up:



Fill out all the desired fields in the right column to insert fixtures

7. All the fields in the right column can be filled with information. The required fields have a value suggestion. These can be edited to suit the needs. All the information can also be edited afterward in the patch menu. The area on the right side adapts to help filling out the selected field. It might have suggestions for input.
8. Accept the suggested values, or edit them to suit the needs.
9. Tap Create ! to add the fixtures.

Each new fixture is now represented by a row in the patch menu. Read below for an explanation of each column in the patch menu and also detailed information about changing some of the values.

Short Explanation of Each Column in the Patch Menu

The patch menu has a lot of columns. This is a short explanation of each. Remember that the patch menu has a condensed and full mode. Select Full to see all the columns.

- **FID**:
  This is the Fixture ID of the fixture. Read more in the **Assign an ID to fixtures** below.

- **IDType**:
  This is the ID Type of the fixture. Read more in the **Assign an ID to fixtures** below.
- **CID**:
  This is the fixtures CID. Read more in the **Assign an ID to fixtures** below.
- **Name**:
  This is the name of the fixture. If there are sub-fixtures, then there is a right pointed arrow that can be tapped to unfold the sub-fixtures.
- **FixtureType**:
  This is the name of the selected fixture type.
- **Mode**:
  This is the mode of the selected fixture.
- **Patch**:
  This is the first DMX address of the fixture. Read more in the **Assign DMX address to fixtures** below.
- **Layer**:
  Fixtures can be organized in Layers. This is the layer information for the fixture. Read more in the **Classes and Layers topic**.
- **Class**:
  Fixtures can be organized in Classes. This is the class information for the fixture. Read more in the **Classes and Layers topic**.
- **Pan Offset**:
  This offset value can be used to offset the DMX output for the Pan attribute. The offset value is applied just before the DMX output. This means that the offset is not shown in the programmer or in cues and presets. The offset can be useful if, for instance, a fixture is hung differently than what has been programmed. Especially in a touring situation where it is not a permanent change.
- **Tilt Offset**:
  This is the same as the Pan Offset, but for the tilt attribute. Read above.
- **Pan DMX Invert**:
  The Pan DMX output can be inverted by editing this field. This can be useful depending on how the fixtures are mounted.
- **Tilt DMX Invert**:
  The Pan DMX output can be inverted by editing this field. This can be useful depending on how the fixtures are mounted.
- **Pan Enc Invert**:
  Pan can be inverted for the encoder rotation by editing this field. This can be useful depending on how the fixtures are mounted.
- **Tilt Enc Invert**:
  Tilt can be inverted for the encoder rotation by editing this field. This can be useful depending on how the fixtures are mounted.
- **Pos X**:
  The is the fixtures position on the X-axis in the 3D window.
- **Pos Y**:
  The is the fixtures position on the Y-axis in the 3D window.

- **Pos Z**:
  The is the fixtures position on the Z-axis in the 3D window.
- **Rot X**:
  Is the rotation of the fixture on the X-axis in the 3D window.
- **Rot Y**:
  Is the rotation of the fixture on the Y-axis in the 3D window.
- **Rot Z**:
  Is the rotation of the fixture on the Z-axis in the 3D window.
- **Scale X**:
  This can be used to scale the fixture/object on the X-axis in the 3D window. The scale value is a factor where 1 is the default. A higher number makes the object bigger. A smaller number makes it smaller.
- **Scale Y**:
  This can be used to scale the fixture/object on the Y-axis in the 3D window. The scale value is a factor where 1 is the default. A higher number makes the object bigger. A smaller number makes it smaller.
- **Scale Z**:
  This can be used to scale the fixture/object on the Z-axis in the 3D window. The scale value is a factor where 1 is the default. A higher number makes the object bigger. A smaller number makes it smaller.
- **Gel Color**:
  Here a color can be defined that will be added to the output from the fixture. It is useful for adding gels to conventional fixtures. This is visualized in the console and 3D window.
- **Gel**:
  This column is not implemented.
- **Beam Angle**:
  Here a beam angle can be defined for the fixture. This is useful for conventional fixtures where different angles might be needed. This is visualized in the 3D window.
- **Cast Shadow**:
  Each fixture cast a show or not. Turning this on makes the 3D look more real world, but it also increases the calculations need for the 3D visualization.
- **Ray Traceable**:
  Turning this to Yes makes the fixtures or 3D objects something that can be selected or picked using the follow function in the 3D window.
- **3D Selectable**:
  This is a Yes or No (text is hidden) field. Yes means that the fixture can be selected using the selection tool in the **3D window**. It can bee useful to turn this off (No option) for stage elements that do not need to be controlled.
- **3D Visible**:
  This is a Yes or No (text is hidden) field. Yes means that it is shown in the **3D window**.
- **Target Space**:
  This is used by the MAker fixtures. This is the selected target space for the MArker fixture. It defines the available space wherein fixtures can move their light beams when they select this MAker fixture. Learn more in the **MAker Fixture topic**.

- **Movement Space**:
  This is used by the MAker fixtures. This is the selected movement space for the MArker fixture. It defines the space wherein the MAker fixture can move. Learn more in the **MAker Fixture topic**.
- **Master React**:
  There are three options to this setting: **None**, **Group**, and **Grand**. This defines if the intensity is affected by a master. If Grand is selected, then group masters are also affecting the fixture. If Group is selected then they are not affected by the grandmaster. None makes the fixture ignore both groups and grandmasters.
- **Appearance**:
  An appearance can be assigned to the fixture.

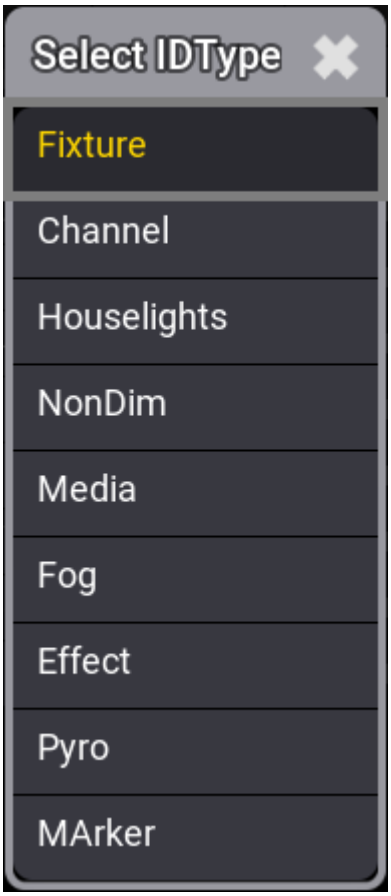
Assign an ID to Fixtures

The fixtures need at least one ID to be selected and controlled.

There are two types of ID numbers for each fixture. Fixtures can have both or just one of the two - but it needs at least one.

The **FID** is the default fixture ID. The number here is used with the **Fixture** keyword.

The **CID** is the second ID. This can be used if the **IDType** is something different than "Fixture". Editing the IDType field opens the **Select IDType** pop-up.



Select the desired IDType

This lists the different IDTypes. Select one that is not "Fixture" to use the CID.

Important:
There is another hidden type called **Universal**. It is a special type that the software automatically creates for a special universal fixture type added to the patch. It cannot be used for normal fixtures.

1. Select the ID fields in one of the two ID columns (FID or CID) for the fixture rows where an ID is to be assigned. The selection order is important.
2. Type a number on the keyboard and assign the number by pressing Please.

Now the fixtures have an ID. The numbers are assigned sequentially based on the selection order.

Assign DMX Address to Fixtures

The fixtures need to be assigned a DMX universe and address before being able to create any output.

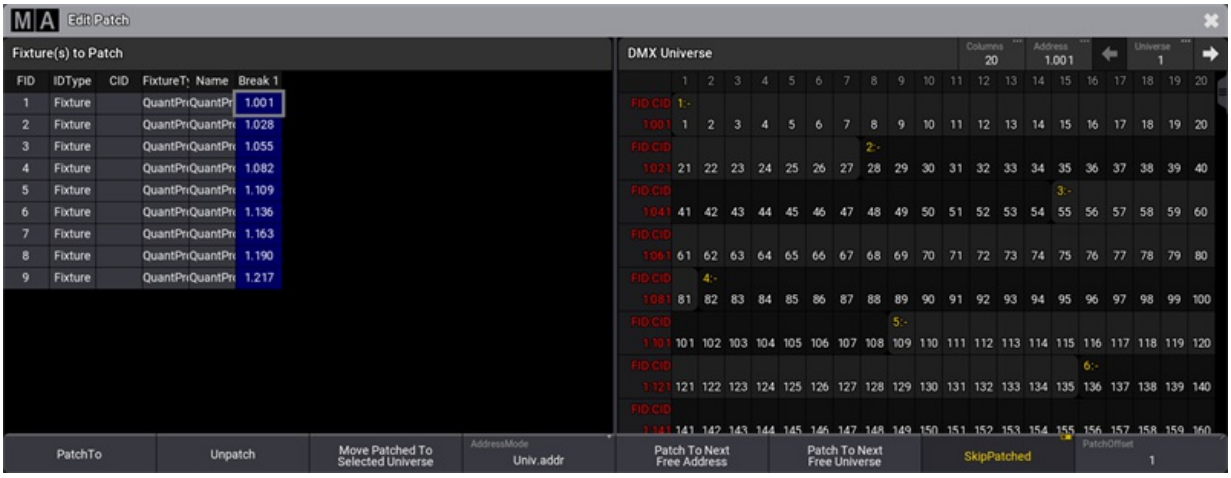
There are two primary options to give the fixtures an address. One is to just type the desired address directly in the patch field. The other is to use the dedicated **Edit Patch** pop-up.

Type the number:

1. Select the fields in the Patch column for the fixture rows where the address is to be assigned. The selection order is important.
2. Type the DMX universe and address separated by a dot (for example **2.1**) on the keyboard and assign the address by pressing Please.

Use the Edit Patch menu:

1. Select the fields in the Patch column for the fixture rows where the address is to be assigned. The selection order is important.
2. Right-click the blue selected areas or tap the Patch button - this opens the **Edit Patch** pop-up.

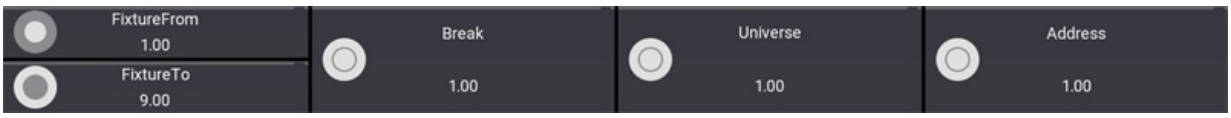


Use the Edit Patch pop-up to get a visual overview of the DMX universe

This pop-up is divided into two sides. On the left side, there are all the selected fixtures. on the right side, there is a view with all the DMX universes and addresses. The purpose of the menu is that fixtures can be selected on the left side and the universe list on the right side shows where there is available space in the universes. There are several ways to navigate universes and assign the fixtures to the selected DMX address.

The numbers on the right side are red if the universe is not granted.

The pop-up also opens a dedicated encoder toolbar:



Edit patch encoder toolbar is used for easy navigation

This toolbar can be used to select fixtures, universe, and address.

This is one way to do it:

1. Select the fixtures in the correct order.

2. Use the encoders to navigate to the desired address.
3. Tap the **Address** encoder to assign the fixtures to the address.

The Edit Patch pop-up has several buttons at the bottom:

- **PatchTo**:
  Opens a small pop-up where the desired patch address can be typed.
- **Unpatch**:
  This removes the patch address from the fixtures.
- **Move Patched To Selected Universe**:
  This moves the fixtures to the selected universe and keeps the DMX address.
- **AddressMode**:
  This toggle button changes how the DMX address is displayed. They can be split up into universes and a range DMX address from 1 through 512 - this option is called **Univ.addr**. The other option is called **Absolute** and it shows the addresses continuously starting from number 1 and upwards.
- **Patch To Next Free Address**:
  This gives the fixtures the next available DMX addresses.
- **Patch To Next Free Universe**:
  This patches the fixtures to the next completely empty universe.
- **SkipPatched**:
  This will skip addresses that have patched fixtures when scrolling through the universes.
- **PatchOffset**:
  This can be used to set the desired number of DMX channels between the fixtures. If the number is lower than the number of channels the fixtures use, then they are patched as close as possible. If the number is more than the used channels then the PatchOffset number is used.

The **Edit Patch** menu auto closes when all the selected fixtures are patched.

Filtering the Patch Menu

Filtering in the patch menu can be useful when there are a lot of fixtures and stage elements.

There are several ways to filter the fixtures in the patch menu.

An existing filter can be assigned using the Filter input in the title bar. Tapping this opens a small select pop-up that lists all existing filters (read about creating a filter in the **Create a Filter topic**). Two other options are **None** (no filter) and **New** (create new filter). Select the desired filter.

Filtering needs to be turned On for the filter to be active. Tap the filter icon button (between Filter and Columns) to turn it On. This button toggles On or Off.

Turning filters On also exposes column filters. An extra row is displayed above the fixture rows just below the column titles. The patch can be filtered by any column value simply by editing the filter field for a column. Only one filter value can be applied to each column.

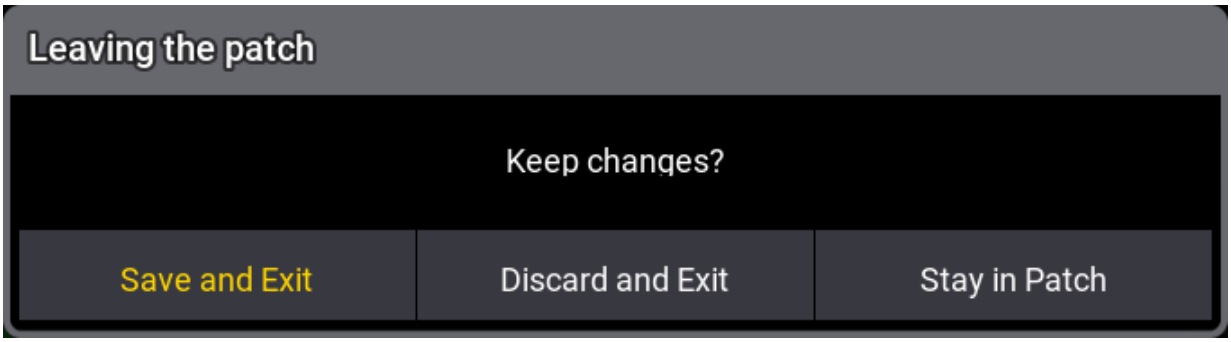
Edit a column filter field to assign a value. Some fields open a small selection pop-up giving a list of available values. Others are input fields where a text needs to be written.

The column filters can be combined with a filter from the pool.

The **Split View** is also a way to filter the patch. Learn more about the split view in the **Patch and Fixture Setup topic**.

Closing the Patch Menu

There is a pop-up asking what to do when exiting the patch menu.



Leave patch pop-up with options

This asks if the changes should be kept.

Tap Save and Exit to save the changes and leave the patch menu. Tap Discard and Exit to cancel any changes and leave the patch menu. Tap Stay in Patch to stay in the patch menu.

13.3. My Virtual Rig

MVR (My Virtual Rig) is a file format that is used to share data for a stage set up between a lighting console, a visualizer, a CAD program or similar tools. This allows transferring parametric and geometric data between different programs.

It is a complementary system to the GDTF files.

MVR is described in detail in the GDTF wiki (external link): **https://gdtf-share.com/wiki/Main_Page**

Import MVR

MVR files can be imported into a show file. This is done from the **Patch menu**. The MVR file needs to be in the correct folder for the software to find it. The folder is [folder with grandMA3]\shared\resource\lib_mvr. It is the same location if the files are on a USB stick.

There is a button at the bottom of the patch menu called Import MVR. Tapping this opens an **Import MVR** pop-up. This is used to browse and select the MVR file. The source drive can be selected in the title bar by tapping the drive selection button (next to the).

Select the desired file in the list and tap Import MVR to import the MVR file.

The content of the MVR file is added to the patch and the information in the MVR files means that there might be added **Layers, Classes**, stage elements, or fixtures.

It is a good idea to store the show file **before** importing the MVR. In case it adds unwanted elements or in worst case corrupts the show.

Export MVR

Exporting the patch to an MVR file is done from the Patch menu. The file is created in the [folder with grandMA3]\shared\resource\lib_mvr folder on the selected drive.

At the bottom of the menu, there is a button called Export MVR. Tapping this opens an **Export MVR** pop-up. This can be used to select the desired drive and give the file a name. It also lists already existing MVR files on the selected drive.

The MVR file contains the entire patch including fixture files, stage elements and organizational elements like the **Classes and Layers**.

It is a good idea to store the show file **before** exporting it to the MVR file.

Important:
In order to export an MVR file or GDTF file that contains user meshes and gobos, save the show file to the local hard drive before exporting.

13.4. Live Patch

The **Live Patch** menu is a version of the **Patch menu** with limited functionality. It looks like the patch menu, but some buttons are missing and some columns have a dark background - these columns cannot be edited.

The idea behind Live Patch is that it is only possible to change the elements that do not require a new show upload to the system.

This means that fixtures/objects cannot be added or removed.

In the live patch it is possible to change the following columns:

- Name
- Patch
- Pan Offset
- Tilt Offset
- Pan DMX Invert
- Tilt DMX Invert
- Pan Enc Invert
- Tilt Enc invert
- Position and Rotation
- Gel Color and Gel
- Beam Angle
- Cast Shadow and Ray Traceable
- 3D Selectable and 3D Visible
- Attached Space and Movements Space
- Master React
- Values for automatic position calculation
- Appearance

There are columns for the four points used for automatic fixture position calculation. Read more about this in the **Position Fixtures in the 3D Space** topic.

Follow the link to the Patch menu (above) to read more about the other columns.

Changes made in the live patch are changed immediately.

The lowest button on the left side gives access to an additional RDM Fixtures menu. This lists fixtures discovered using RDM. Learn more about RDM in the **RDM section**.

13.5. DMX Sheet

The DMX sheet displays the actual DMX output from the system. It shows the result from the sequences, programmer, incoming merged DMX, and any masters that might limit the output.

The sheet can be created as a window on any empty space on the screens. The minimum size is 1.5 squares wide and 2 high. It is created like any other window using the **add windows**.

DMX sheet with patched fixtures

The main part of this window displays a big grid with every DMX address, represented by its own square. This makes it a very long list (all 1024 universes are there).

The first column on the left side is a label for the second column.

There are two ways to see the DMX address: Absolute DMX address and the normal Universe divided. This can be changed in the **Sheet Settings**. The absolute address mode displays the DMX addresses as a continuous number. This means that the first address in the second universe will not be written as "2:001" but as "513" (512 addresses from the first universe + 1 from the second).

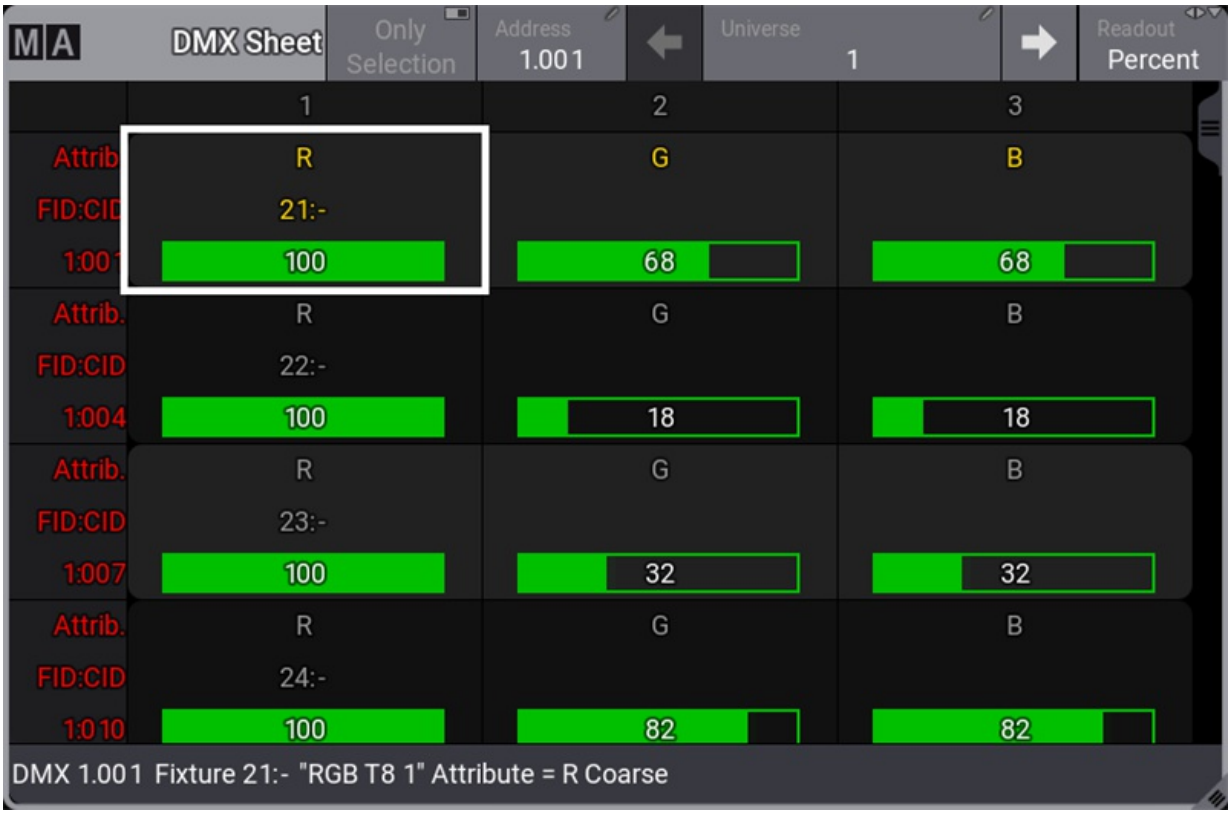
The top row displays the column number. The grid is a matrix, so the column number should be added to the address displayed in the vertical bar (except the first column).

Each square can display up to three different types of information:

- **Value**:
  The DMX channels value.
- **Attribute**:
  The attribute the DMX channel is controlling.
- **ID**:
  The FID and CID assigned to the fixture patched at the DMX channel.

The three different elements can be turned On or Off individually in the mask settings (read the **Sheets Settings** below for details).

If a fixture uses more than one DMX channel then the rounded corners are on the first and last attribute/DMX channel. The background color alternates between two shades of gray to indicate different fixtures.



Detail of the DMX sheet

In the example above there are four RGB LED fixtures patched. The selected fixture (FID 21) gets a yellow text color in attributes and FID/CID.

If fixtures in the leftmost column have a red text color then the channels/universes are not granted. Read more about getting more parameters in the **Expand the Amount of Parameters topic**.

If the cursor is moved to a square then the information about the square is displayed in a text line at the bottom of the window (see the image above as an example).

The two images show the two different ways the value can be visualized in the DMX sheet. The first is called **Background**, and the second is **Bar**. They can be changed or turned Off in the **Sheet Settings**. It is called **Levelbar**.

The title bar of the window has some buttons.

Only Selection jumps to the universe of the first selected fixture. The one called **Address** displays the number of the highlighted square. Tap this button to get an input box. Typing a new address in this box jumps to the typed address.

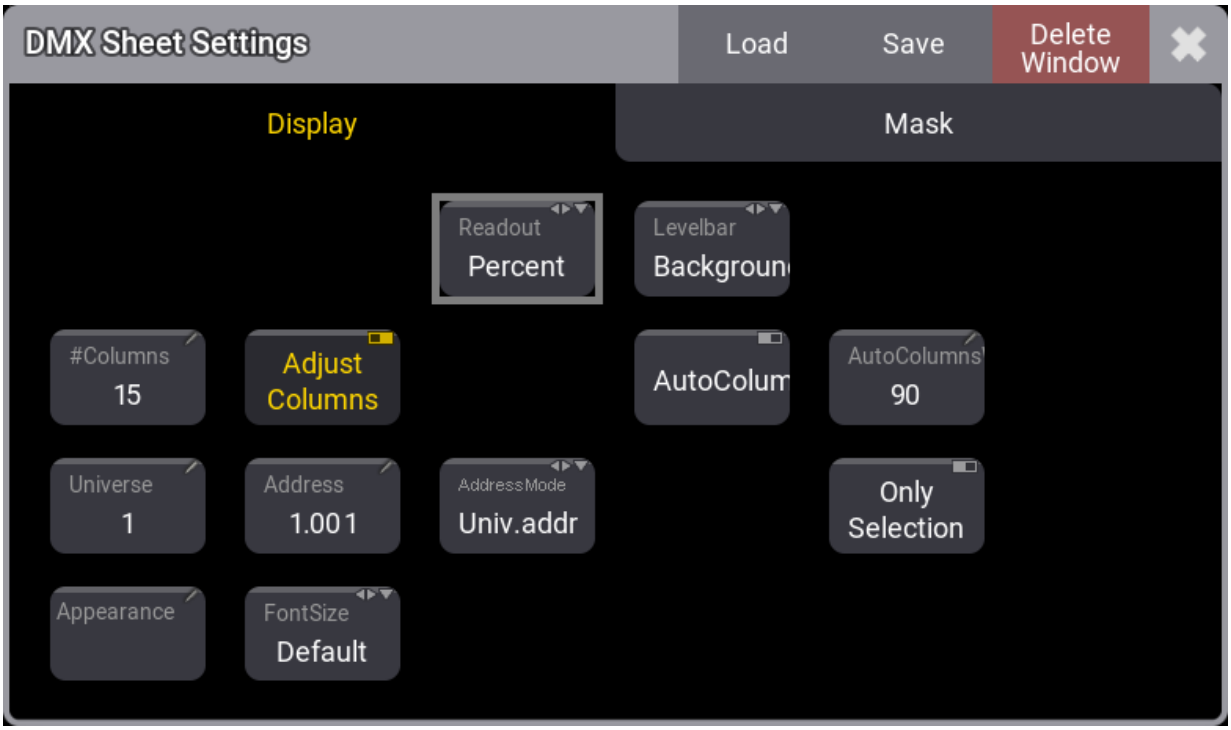
The button called **Universe** displays the current universe number. Tap this button to get an input box. Typing a new number in this box jumps to the universe. The two arrows jump one universe back or forward.

The final button in the title bar is the **Readout**. This can be used to change how the value readout is displayed. The options are Percent, Decimal, or Hex.

Sheet Settings

The DMX sheet has some settings that define how the sheet shows the data.

Tap the MA logo in the upper left corner of the window to open the settings. It opens the Display tab as a default.



DMX Sheet Settings - Display tab

Some of these settings are general settings shared with other windows. They are described in the **Window Settings topic**.

Some settings are only for the DMX Sheet. This is a list of them with a short explanation:

- **Address**:
  This input button opens a calculator that allows you to type the DMX address that the sheet should begin with. This setting is also in the title bar.
- **Address Mode**:
  This toggle button changes how the DMX address is displayed. They can be split up in universes and a range DMX address from 1 through 512 - this option is called **Univ.addr**. The other option is called **Absolute** and it shows the addresses continuously starting from number 1 and upwards.
- **Auto Columns**:
  This On/Off button activates a function that automatically adjusts the number of columns in the DMX Sheet. **Adjust Columns** need to be active for this to work. **#Columns** is automatically adjusted by this function.

- **Auto Columns Width**:
  The **Auto Columns** function uses this value to determine how many columns the sheet can show.
- **Level Bar**:
  This input field is used to change the level bar setting in the sheet. There are three options:
  - **Off**:
    The level is only visualized by the value.
  - **Background**:
    The background changes from dark green to bright green as the value rises.
  - **Bar**:
    The value is visualized as a bar that moves from left to right as the value rises.
- **Only Selection**:
  This On/Off button changes the mode of the DMX sheet. This will limit the DMX sheet to only display the universes of the selected fixtures. This setting is also in the title bar.
- **Universe**:
  This input button can be used to change the universe a DMX sheet should display. There is a special setting called **Selected**. This makes the DMX sheet scroll to the selected universe. This setting is also in the title bar.

The mask tab contains the three On/Off toggle buttons called **Value**, **Attribute**, and **ID**. They are described in the text above.

DMX Tester

The DMX channels can have output from the DMX tester.

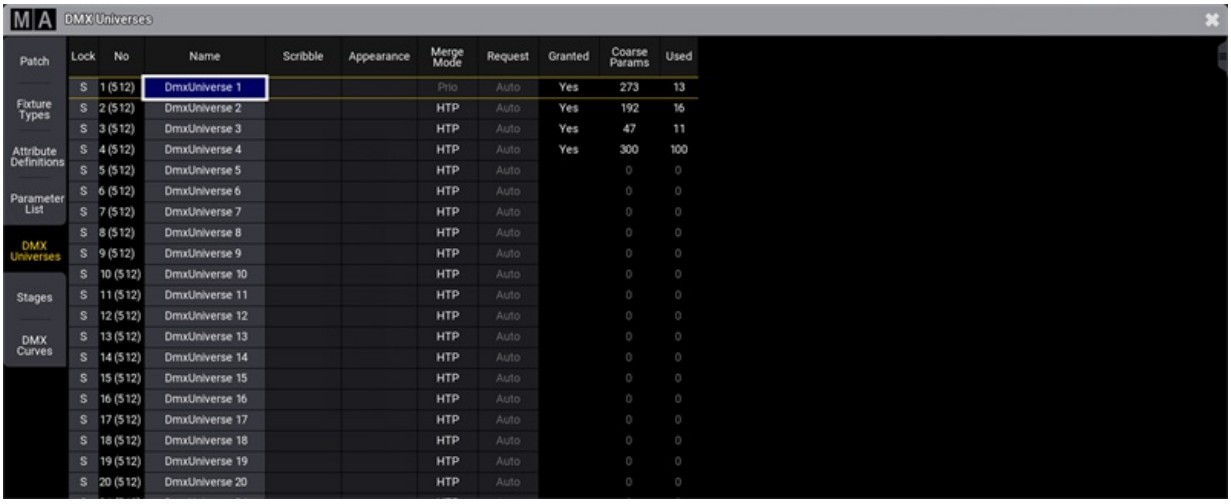
This is currently done using the **DMXAddress** and **DMXUniverse** keywords.

When a DMX channel is affected by the DMX tester, the background color for the value is white.

13.6. DMX Universes

Universe Menu

All the 1024 DMX universes are listed in the DMX Universes menu. This can be accessed by Menu - Patch - DMX Universes.



DMX Universes menu

The menu has several columns showing information about each universe.

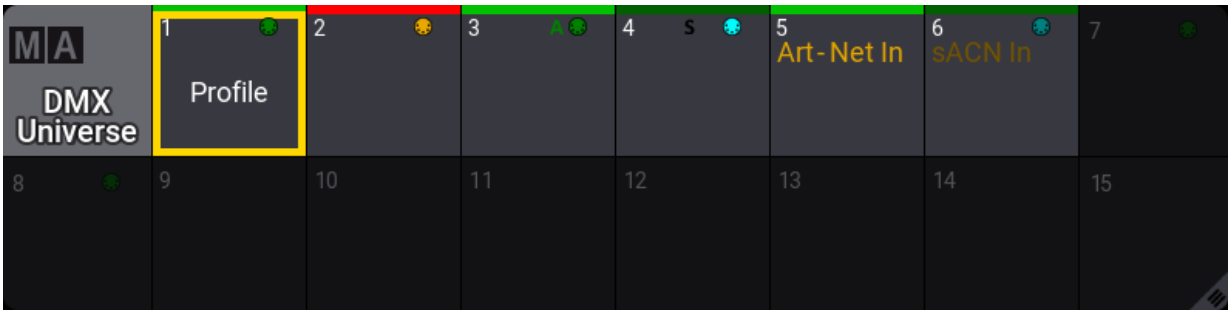
- **Lock**:
  This shows if the row is locked.
- **No**:
  This is the universe number.
- **Name**:
  This is the name of the universe. It can be a good idea to give the universes a name for a better overview.
- **Scribble**:
  A **scribble** can be assigned to the universe.
- **Appearance**:
  An **appearance** can be assigned to the universe.
- **Merge Mode**:
  This is the merge mode used with incoming DMX data. These are described in the **DMX Port Configuration topic**.

- **Request**:
  A universe can be requested or not. Requesting a universe means that there is a desire to send DMX data to the universe. This request might not be granted - normally because there are not enough unlocked parameters (read more in **Expand the Amount of Parameters topic**). The setting has three options:
  - **Auto** (default): This automatically requests the universe if there is something patched to it.
  - **On**: This requests the universe.
  - **Off**: This does not request the universe. It can be useful to turn off the request if there are not enough unlocked parameters but a test is needed for a specific universe. For instance, during preprogramming.
- **Granted**:
  This shows if the universe output is granted. It says "Yes" if the universe is requested and granted. This field is information only.
- **Coarse Params**:
  This is the number of parameters patched to the universe.
- **Used**:
  This is the number of fixtures patched to the universe.

Universe Pool

The universe pool has a pool object for all the 1024 universes that the system can handle.

The pool is a window created like any other window. Learn how in the **Add Windows topic**.



DMX universe pool

Each pool object represents a DMX universe.

The pool object shows some information about the universe.

There is a colored bar at the top of the pool object. This is green if the universe is requested and granted. It flashes bright green when the DMX data is outputting. It is red if it is requested but not granted.

If a universe is assigned to a DMX port, then there is a small port icon. This can have different colors depending on the port is set to output (green), input (orange) or if it is output with RDM enabled (cyan).

There are black indicators to show if the universe is output using sACN (a small "S") or Art-Net (a small "A"). These flashes green when the DMX output values are updated. See universe 3 and 4 in the example image above.

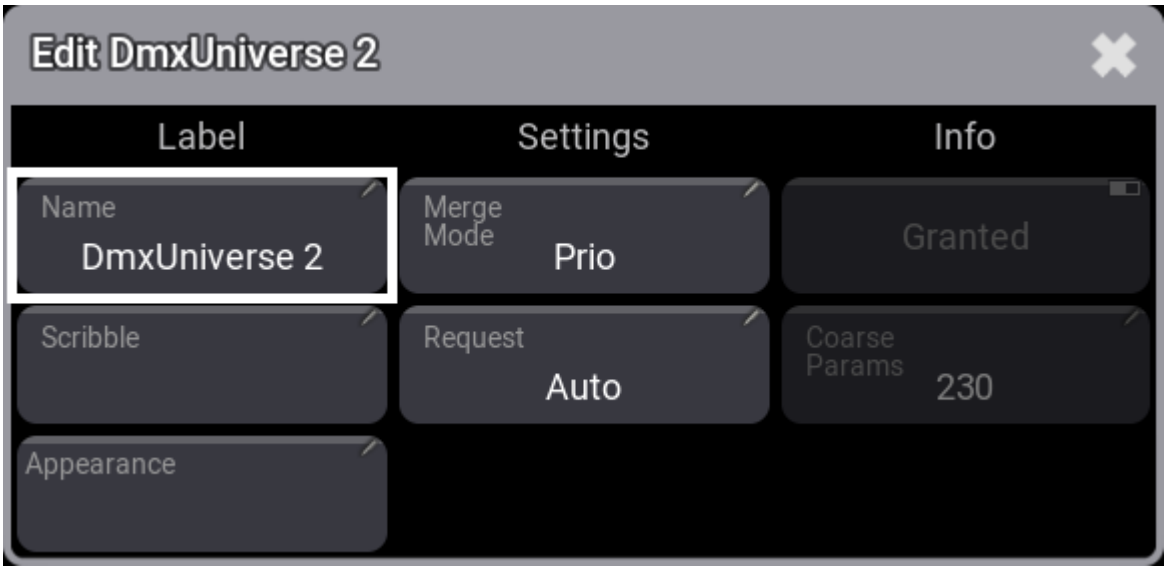
There are also indicators to show if DMX data is merged into the universe from Art-Net or sACN. This is indicated with an orange text: **Art-Net In** and **sACN In**. This is a dark orange text to show the assignment. It flashes orange when there is an incoming signal. See universe 5 and 6 in the example image above.

Read the topics in the **DMX In and Out section** to learn how to set up the DMX ports and how to set up Art-Net and sACN.

Moving a universe in the pool also changes the universe for the fixtures patch in the universe but keeps the DMX start addresses.

A universe can be selected by tapping it in the pool or using the **Select keyword** together with the **DMXuniverse keyword**. The **DMX Sheet** can be set up to only show the selected universe in the sheet settings - using the universe setting.

Editing a pool object opens the Edit DMX Universe pop-up:



Edit DMX Universe pop-up

Here it is possible to edit the same settings as in the DMX Universe menu.

13.7. Remove fixtures from the show

Removing or deleting a fixture row in the **Patch menu** removes all programed information from the show.

It can not be done from the Live Patch menu.

Removing the DMX address or the ID number does not remove the programed value, but the fixture can no longer be edited and it does not produce any output. It is also not visible in the **Fixture Sheet**.

Remove a Fixture from the Patch

Requirement

Have some fixtures already in the patch.

1. Open the Patch Menu
    1. Press Menu
    2. Tap Patch
2. Select the fixtures in the patch
3. Tap Delete
4. Close the Patch menu and tap Ok to accept the changes

The fixtures are now deleted and all programming with those fixtures is lost. It cannot be oopsed.

13.8. Position Fixtures in the 3D Space

The **3D Window** shows a 3D visualization of the fixtures and objects in a 3D space.

The fixtures need to be positioned in the 3D space for this to truly be a powerful tool. The virtual fixtures should be positioned and rotated to match the real-world fixtures.

There are three primary ways to change the fixture position: **Using the patch**, **using the 3D window**, or **position calibration**.

Position Fixtures Using the Patch

The best way to position the fixtures from the patch is from the **Live Patch**:

1. Press Menu.
2. Tap Live Patch.
3. Make sure the menu is in **Full** column mode.

In the patch, there are rows for each fixture and there are columns with position and rotation values:

Pos			Rot		
X	Y	Z	X	Y	Z
0.000	0.000	0.000	0.00	0.00	0.00
0.000	0.000	0.000	0.00	0.00	0.00
0.000	0.000	0.000	0.00	0.00	0.00
0.000	0.000	0.000	0.00	0.00	0.00
0.000	0.000	0.000	0.00	0.00	0.00
0.000	0.000	0.000	0.00	0.00	0.00

Position and Rotation values in the Patch.

New fixtures are always added in the zero point location and with the zero rotation.

The zero points are 0.000 meters for all three position axis (X, Y, and Z) and 0.00° for all three rotation axis.

- X position is usually regarded as the Stage Left and Stage Right indication. A positive value is in the stage left direction.
- Y position is usually the Downstage and Upstage direction. A positive value would move the fixture more upstage.
- Z position is the height. A positive value would move the fixture above the stage.
- X rotation is rotating the fixture around the fixtures' own X-axis. A positive value is rotating the top of the fixture towards downstage.
- Y rotation is rotating around the fixtures' own Y-axis. A positive value rotates the top towards stage left.
- Z rotation is rotating around the fixtures' own Z-axis. A positive value rotates the fixture counter-clockwise seen from the top.

The grandMA3 software currently only works with meters and degrees.

Do the following steps to edit the fixtures position and rotation:

1. Locate the rows for the fixtures that need to be positioned.
2. Edit the needed fields.
3. Type the new position.

Position Fixtures Using the 3D Window

The 3D window can be used to position and rotate the fixtures. The window needs to be in **Setup** mode. This can be changed by tapping Mode in the title bar.

This mode changes the encoder toolbar into this:

Use the encoders to position and rotate fixtures

There are different buttons in this toolbar that changes how the fixtures are affected by the encoders. Some are standard functions that do not change, others change depending on selected functions.

The general workflow is this:

1. Select the fixtures that need to be moved.
2. Change the settings in the toolbar to match the wanted action.
3. Use the encoders to adjust the values.

The values on the encoders show the relevant values depending on the selected fixtures. Turning the encoders changes the values and thus the position or rotation of the selected fixtures.

These are the different standard buttons for position and rotation:

- **Function**:
  This button toggles between two different modes: **Position/Rotation** and **Arrangement**. This changes the function of the encoders to either change the position and rotation of the selected fixtures or to different arrangement types. Read about the Arrangement tool **below**.
- **Axis**:
  The axis is used when a fixture is moved or rotated. The two options here are **Stage** and **Object**. This means that the selected fixtures can be positioned and rotated using their own axis (Object) or the stage or world axis (Stage).
- **Rotation Mode:**
  When multiple fixtures are selected, then rotation mode can be changed between **Single** or **Group**. This defines if the rotation is done for each fixture or if the selection of fixtures is treated as a group, and thus rotated as one object.
- **Calibrate Position**:
  This opens the Position Calibration menu. Read more about it **below**.
- **Change on EncoderEvent**:
  This determines when changes are distributed through the network to other stations. When it is set to **Yes**, then the new position is sent immediately. When it is **No**, then the new information is sent two seconds after the encoders stop turning.
- **Reset Position:**
  Tap this button to reset the fixture position to 0 in all positions.
- **Reset Rotation**:
  Tap this button to reset the rotation to 0 on all axis.

Position Arrangement Tool

The position arrangement tool is a set of three different arrangement or layout types. It can be used to arrange the selected objects in a **Line**, **Grid**, or **Circle**.

The tool is accessed by changing the Function in the encoder toolbar to **Arrangement**. Each of the three layout types has its own encoder functions and their own buttons. There are some common arrangement buttons:

- **Layout Type**:
  There are three types of arrangement types. **Line**, **Grid**, and **Circle**. These different types change the available buttons in the encoder bar and the functions of the encoders.
- **Reset Encoder Values**:
  Resets the values on the encoder to the default values.
- **Apply on Change**:
  This controls if arrangement changes are applied and distributed immediately (Yes) or if they are only marked by a purple indicator of the would-be location (No).
- **Apply**:
  If **Apply on Change** is set to No, then this button needs to be tapped to confirm the new arrangement location.

When an arrangement is being adjusted, then there are purple fixtures in the 3D window with ID numbers to indicate where the fixtures would end up if the arrangement settings are applied.

Line

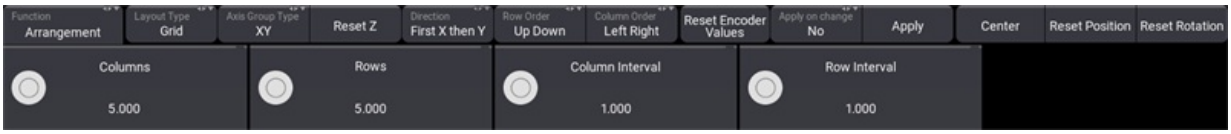
This is used to position the fixture on a single row.

Line Arrangement Encoder Toolbar

The encoders changes to set a start and a length for all three axes. There is a special button called **Line up**. Tap this to align the base of the fixture to match the line.

Grid

The grid arrangement moves the fixtures into rows and columns. It is a 2D grid.



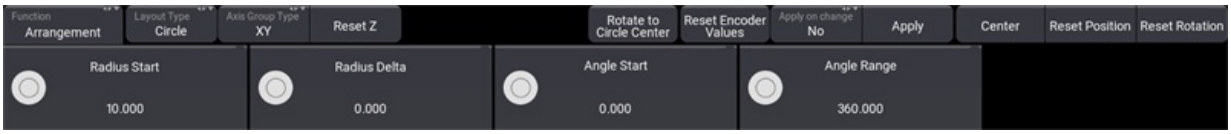
Grid Arrangement Encoder Toolbar

The encoders change to set the number of columns and rows and the interval spacing between the columns and rows. There are some special buttons:

- **Axis Group Type**:
  This sets the orientation of the grid. It shows the two axes that are used as columns and rows.
- **Reset Z** / **Reset X** / **Reset Y**:
  This button resets the position on the third axis. The one not affected by the grid.
- **Direction**:
  This changes the direction of the grid. It changes the order of the two axes.
- **Row Order**:
  Tap this button to reverse the direction of the row.
- **Column Order**:
  Tap this button to reverse the direction of the row.

Circle

The circle arrangement tool is used to position fixtures in circles and spirals.



Circle Arrangement Encoder Toolbar

The encoders change to set the Radius Start size, Radius Delta (sets if the radius changes size), Angle Start, and Angle Range. Angle Start sets where the first fixture is on the circle. Angle Range is used when creating semi-circles (value below 360) or even multiple circles (value above 360).

There are some special buttons:

- **Axis Group Type**:
  This sets the orientation of the circle. It shows the two axes that are used.
- **Reset Z** / **Reset X** / **Reset Y**:
  This button resets the position on the third axis. The one not affected by the circle.
- **Rotate to Circle Center**:
  This rotates the fixture bases to match the circle. Tapping this button several times rotates the fixtures each time to a different rotation.

Position Fixtures Using Position Calibration

The fixture position calibration system calculates the fixture position based on pan and tilt values needed to hit three or four known points on the real-world space. Using only points one, two, and three can be enough but the best result is achieved using all four points.

The four points do not need to be the same for all fixtures. Each fixture can have its own four points. The points are visible in the 3D window as a red, green, blue, and yellow octahedron when the window is in setup mode and the **Position Calibration** pop-up is open.

The pop-up is opened by tapping the Calibrate Position on the encoder bar - read **above**.

Position calibration pop-up with some selected fixtures and their values

The pop-up shows the selected fixtures as rows and four colored sections of columns.

Each color section has X, Y, and Z values for the calibration point and Pan and Tilt values needed to hit the point.

At the bottom, there are buttons to store and recall the pan and tilt values for each point. There is also the Solve needed to start the position calculation.

To record the pan/tilt position that matches the calibration point, tap Store Px. Call Px can be used to recall a stored position to refine it.

Hint:
It is recommended to use four calibration points. Using three calibration points deteriorates the quality of the calibration.

To calibrate fixtures, use this workflow:

1. Open the **Position Calibration** pop-up.
2. Select the fixtures to calibrate.
3. Define the real-world coordinates of the calibration points (P1 - P4) in the position calibration pop-up for each fixture.
4. Position each fixture using pan and tilt to each of the calibration points, and store these positions.
5. Tap Solve.
6. Close the pop-up.

Now the fixtures should move and rotate in the 3D window to match the real-world values.

Using the Command Line

In order to trigger the actions of storing and calling calibration points and solving the calibration, these commands can be used:

To store the position of the selected fixtures to calibration point 1, type:

```
User name[Fixture]> Action "StoreCalibrationPoint1"
```

To re-call the position of calibration point 1 for the actually selected fixtures, type:

```
User name[Fixture]> Action "CallCalibrationPoint1"
```

To calibrate the currently selected fixtures, type:

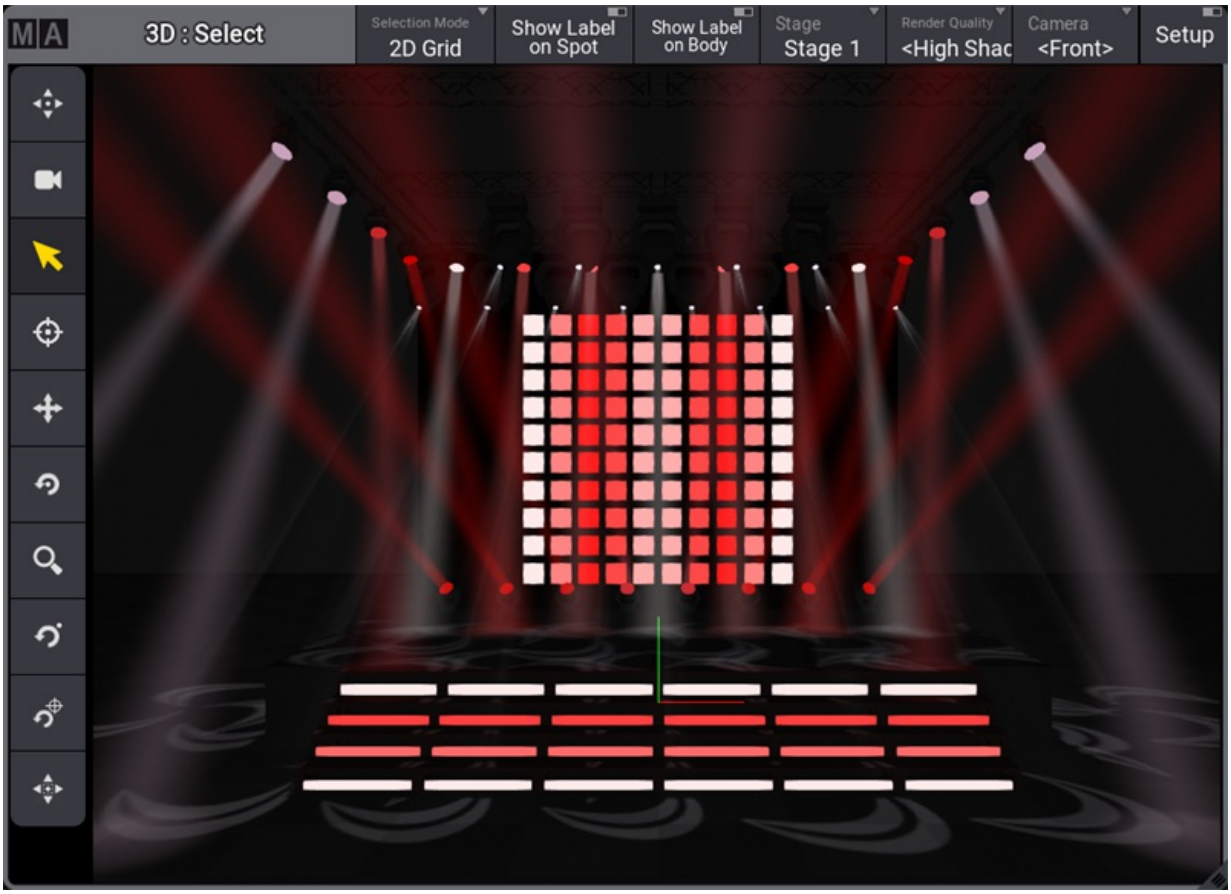
```
User name[Fixture]> Action "SolveCalibration"
```

13.9. 3D

The 3D window shows a 3D visualization of the virtual space where fixtures can be positioned and rotated.

The fixtures can project a light beam that moves and changes color when the values for the fixtures are changed.

All the fixtures and other stage objects can be **positioned and rotated** in the **Patch**, **Live Patch**, or using this window.



The 3D window with some fixtures

The **Stage** object can be visualized as a box or just a floor. The box can be looked into, but not out of. Read more **below**.

There are virtual cameras that are used to see the 3D space. These define the position, direction, and other settings from where and how the 3D space is viewed. The cameras are stored in the **Camera Pool**.

Title Buttons

There are several buttons in the title bar. Most of them are swipe buttons that open a list of the available options.

Three of them are toggle buttons that turn On or Off labels and turn On or Off the Setup mode.

The others give fast access to select Selection Mode, Stage object, Render Quality, and Camera selection.

There can be several stages in the patch. Each 3D window can show one of these stages. Read more about stages in the **Stages topic**.

Read about the settings **below**.

Left Side Tool Buttons

There are some tool buttons on the left side of the window. These tools are used to control what happens when the window is touched and to change camera positions (read more about the cameras below). The ones that change the touch mode are tapped to select the mode and when the window is touched the mode dictates what happens. The selected mode is displayed with a yellow color and in the window title bar. Most of the tools are grayed out and disabled if the selected camera is locked.

This is the explanation of the different tool buttons:

- **(fit):**
  Tapping this button moves the camera to fit the entire stage area into the view.
- **(camera):**
  This button reloads the selected camera to the settings set in the **camera pool object**.
- **(select):**
  Sets the touch mode to **Select**. This is used to select fixtures in the window. The fixtures can be tapped or selected using a selection lasso.
- **(follow):**
  Sets the touch mode to **Follow**. This makes all the selected moving lights point to the stage area that is touched. The follow function obeys the **Align** settings.
- **(move):**
  The mode is set to **Camera Move**. Move means changing the position of the camera without changing the pan and tilt values of the camera. The scroll wheel on a mouse moves the camera forward and backward in the view based on the location of the pointer (not necessarily the center).
- **(rotate center):**
  Sets the mode to **Camera Orbit**. This mode orbits the camera around the center of the window keeping it pointed towards the center. The scroll wheel on a mouse moves the camera closer to or away from the location of the pointer (not necessarily the center).
- **(zoom):**
  Changes the mode to **Camera Zoom**. This mode moves the camera in and out of the 3D window. This can also be done in the other modes using a scroll wheel on a mouse.
- **(rotate pivot):**
  This mode also orbits the camera, but this orbit around a pivot point that does not have to be the center. The pivot point can be set (read about the next button) and is remembered until a new point is set. The scroll wheel functions just like the rotate center mode.

- **(set pivot):**
  This button is used to set a new pivot point in the window. As soon as the point has been set (by clicking or touching the window), then the rotate pivot mode is selected.
- **(fit selected):**
  Tapping this button moves the camera to fit the selected fixtures into the view without orbiting the camera.

Moving the Camera

The cameras are pool objects in the **Camera Pool**. A camera can be moved to different locations and pointed in different directions.

This information can be edited in the pool, but the position and direction are easier to change in the 3D window - read about the left side tool buttons above. There is one camera mode that cannot be accessed in the menu on the left side (Camera Pivot). This can only be changed in the window settings.

Camera Pivot is a lot like camera orbit - the difference is that the camera pivots around the point touched in the window instead of always around the center of the 3D space. A scroll wheel on a mouse moves the camera closer to or away from the location of the pointer.

Editing the camera in the pool allows changing the camera mode and type. Read more in the camera pool topic (link above).

Moving the Fixtures

The fixtures position and rotation can be set in the **patch** or **live patch**. But they can also be positioned live in the 3D window.

Read more about this in the **Position Fixtures in the 3D Space topic**.

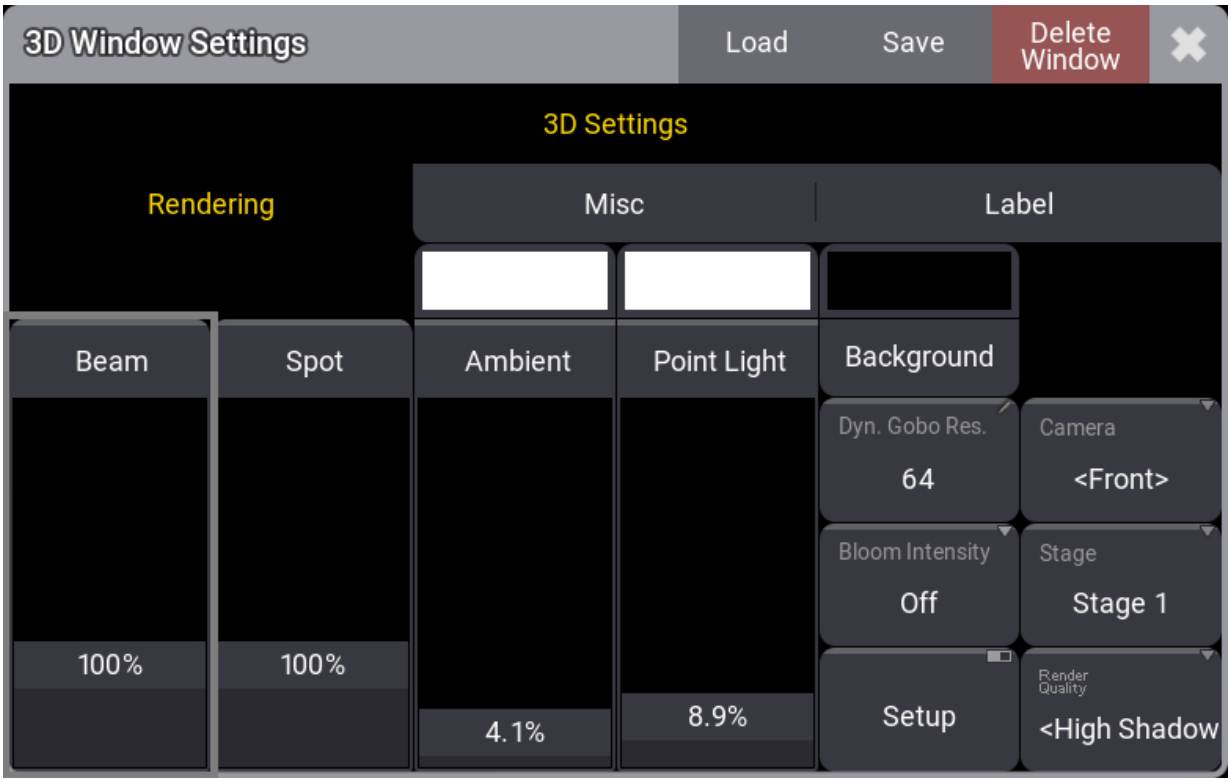
Window Settings

The settings can be opened by tapping the MA logo in the upper left corner of the 3D window.

This opens a settings pop-up. In this pop-up, there are tabs called **Rendering**, **Misc**, and **Label**.

Rendering Settings

The rendering settings are about light levels, colors, and rendering quality.



Rendering tab in 3D Window Settings

There are four on-screen faders:

- **Beam:**
  This is the visibility of the light beam from all fixtures.
- **Spot**:
  This is the general intensity of the visualization of the light beam reflection where it hits a surface.
- **Ambient**:
  This is the ambient light level inside and outside the stage box. It is a very diffuse light that removes some of the contrast in the 3D window.
- **Point Light**:
  This is a light source from the direction of the camera. It is used to light up the elements in the 3D space.

The ambient and point light can be colored, similar to putting a gel in front of the light. The area above the faders can be tapped to open a color selector pop-up.

The **Background** can also be colored. The ambient light needs to be turned up for this color to be visible.

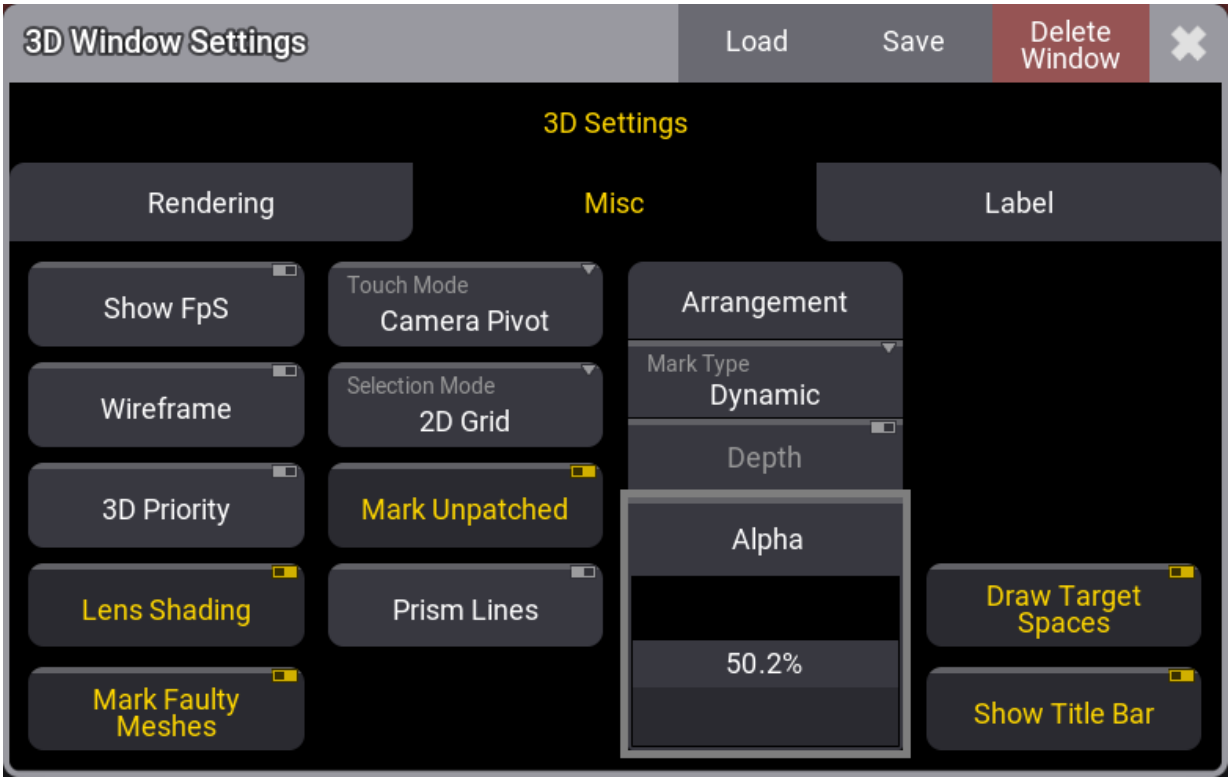
There are several buttons:

- **Dynamic Gobo Resolution (Dyn. Gobo Res.)**:
  This is the resolution of dynamically created gobos.
- **Bloom Intensity**:
  The blooming effect can be turned On or Off. Tapping this button toggles between the two.
- **Setup**:
  Turning setup On makes it possible to move the fixtures and 3D objects using the 3D window and encoder bar. Read more about this in the **Position Fixtures in the 3D Space topic**. This is also a button in the title bar.
- **Camera**:
  This is used to select one of the cameras from the camera pool. If it is set to follow the selected camera then the name is inside angled brackets. Read more about cameras in the **Camera Pool topic**. This is also a button in the title bar.
- **Stage**:
  This selects what stage the 3D windows display. A 3D window can only display one stage at a time. Read more about stages in the **Stages topic**. This is also a button in the title bar.
- **Render Quality**:
  This is used to select one of the render qualities. Read more about render qualities **below**. This is also a button in the title bar.

Some of these are swipe buttons, so remember that the options can be reached easily by swiping out of the button.

Misc Settings

The second tab is called **Misc**.



Misc tab in the 3D Window Settings

The Misc. tab has the following settings:

- **Show FpS:**
  Turning this On displays frames per second information in the upper right corner of the window.
- **Wireframe:**
  Turning this On shows the 3D window as a wireframe instead of a shaded view.
- **3D Priority**:
  Turning this On gives the 3D window a high priority. It is only recommended to turn this On when used on a computer with a high-quality graphics card. It takes away resources from a console interface making it react slower to user input.
- **Lens Shading**:
  This defines whether the selection is shown on the lens of a geometry type “Beam”. The shaded selection is drawn when it is On.
- **Mark Faulty Meshes**:
  This marks meshes that are unloaded or faulty.

- **Touch Mode**:
  Touching and swiping in the 3D window can interact with the window in different ways. This setting defines how. Tapping it toggles through the following options: **Select, Follow, Camera Orbit**, **Camera Zoom**, **Camera Move, Camera Pivot**, and **Camera Set Pivot**. The Camera options move the camera - Read more **above**. The **Select** option is used to select fixtures in the view.
- **Selection Mode:**
  This defines how the fixtures are selected and positioned in the **Selection Grid**. This is also a button in the title bar. The selection mode has two different options:
  - **2D Grid**:
    The 2D grid selection offers two different modes, **Planar** and **Perspective**. Using the lasso selection to select fixtures in the 3D window adds them as a two-dimensional selection to the selection grid. In planar mode, the camera position does not affect the selection order. Only the real position of the fixtures is significant. In perspective mode, the orientation of the camera additionally influences the order of the selection in the selection grid.

    To select fixtures in the 3D window using the planar selection, draw a lasso starting with a horizontal or vertical mouse movement. The color of the lasso changes to green when selection is locked to planar mode. To select fixtures in perspective mode, draw a lasso starting with a diagonal mouse movement. The color of the lasso changes to cyan when the selection is locked to perspective mode.

  - **Linearize**:
    The selection is linearized to only the X-axis of the selection grid depending on which direction the selection lasso was created (top/bottom - left/right). This selection mode is indicated with a yellow lasso.

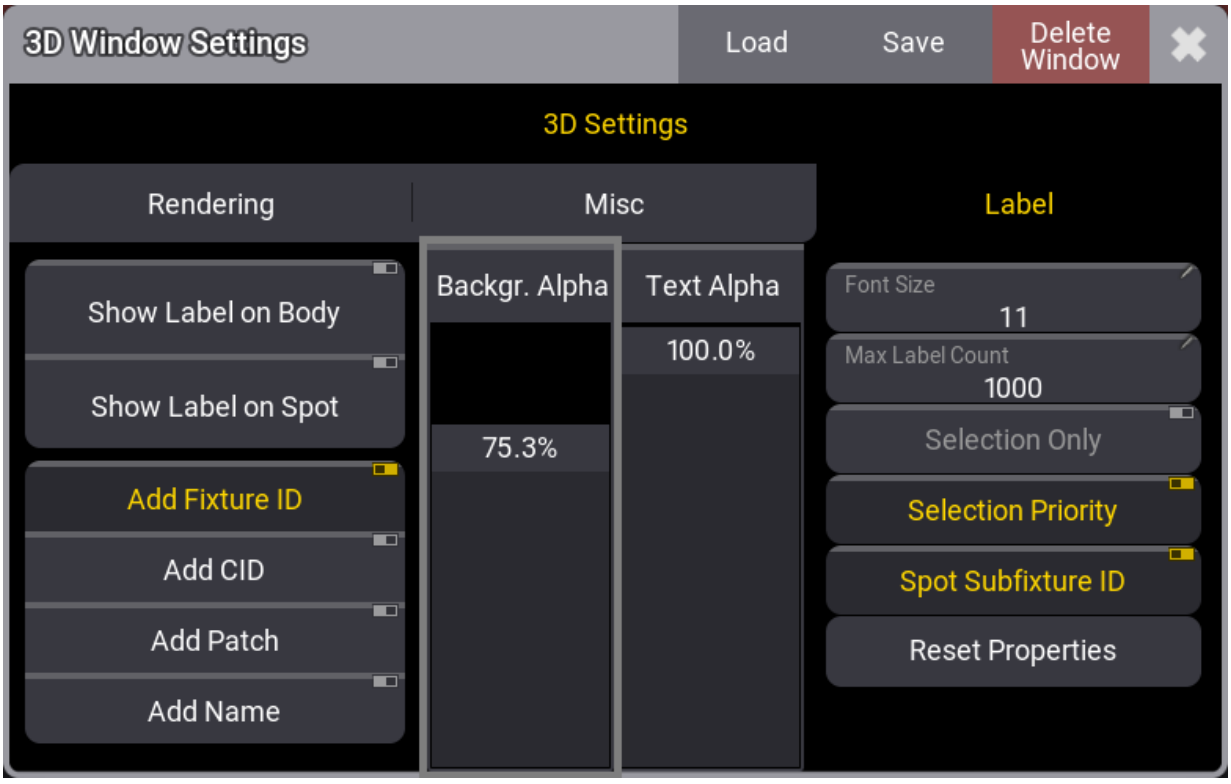
Hint:
The projection distortion of the camera of the 3D window may affect the position of the fixtures in the selection grid if the perspective selection is used. In order to prevent this, use a 2D camera in the 3D window.

- **Mark Unpatched**:
  This marks unpatched fixtures with a dark red color when On.
- **Prism Lines**:
  This shows a line for each prism facet when the **Beam Quality** is set to **Line**.

- **Arrangement**:
  This is a collection of settings regarding the visualization of the purple arrangement "ghosts" markers. They are seen when using the arrangement tool to position fixtures.
  - **Mark Type**:
    There are two options: **Dynamic** or **Small**. Dynamic shows a box matching the size of the fixture. Small is just a small square marker.
  - **Depth**:
    Set if the depth of the 3D space is considered when showing the marks. This could lead to marks being hidden behind objects. If the depth is switched off, the marks are always drawn in front of other objects.
  - **Alpha**:
    Define the transparency of the marker objects.
- **Draw Target Spaces**:
  This setting show or hide a square wireframe for MArker fixtures target spaces.
- **Show Title Bar**:
  This setting shows or hides the title bar for the window.

Label Settings

The tab is called Label. It is settings about the labels that can be turned On or Off for the spots and the fixture bodies.



Label tab in the 3D Window Settings

- **Show Label on Body**:
  This On / Off button enables labels to be drawn on the fixture's body. This is also a button in the title bar.
- **Show Label on Spot**:
  This On / Off button enables labels to be drawn in the center of the fixture's spot. A fixture can also be selected using the rubber band selection on a label. This is also a button in the title bar.
- **Add Fixture ID** / **Add CID** / **Add Patch** / **Add Name**:
  These On/ Off buttons define what kind of information is displayed on the label.
- **Background Alpha**:
  This on-screen fader sets the transparency of the label's background.
- **Text Alpha**:
  This on-screen fader sets the transparency of the text that is displayed in the label.
- **Font Size**:
  This button sets the font size of the text that is displayed in the label. Tapping this opens the small Select Label Font Size pop-up with the size options.
- **Max Label Count**:
  This input field sets the maximum count of labels that are displayed at the same time. If the number of labels exceeds the maximum label count, the labels closest to the camera will be displayed.
- **Selection Only**:
  This On / Off button defines whether labels are displayed for all fixtures or if they are only for fixtures that are selected or partly selected.
- **Selection Priority**:
  This On / Off button defines whether labels of selected fixtures are displayed on top of not selected fixtures.
- **Spot Subfixture ID**:
  This On / Off button defines if the label of the spots shows the Fixture ID of the corresponding sub fixture (On). Otherwise, all spots of a fixture show the Fixture ID of the main fixture (Off).
- **Reset Properties**:
  Tap this button to reset the label properties to their factory defaults.

Render Quality

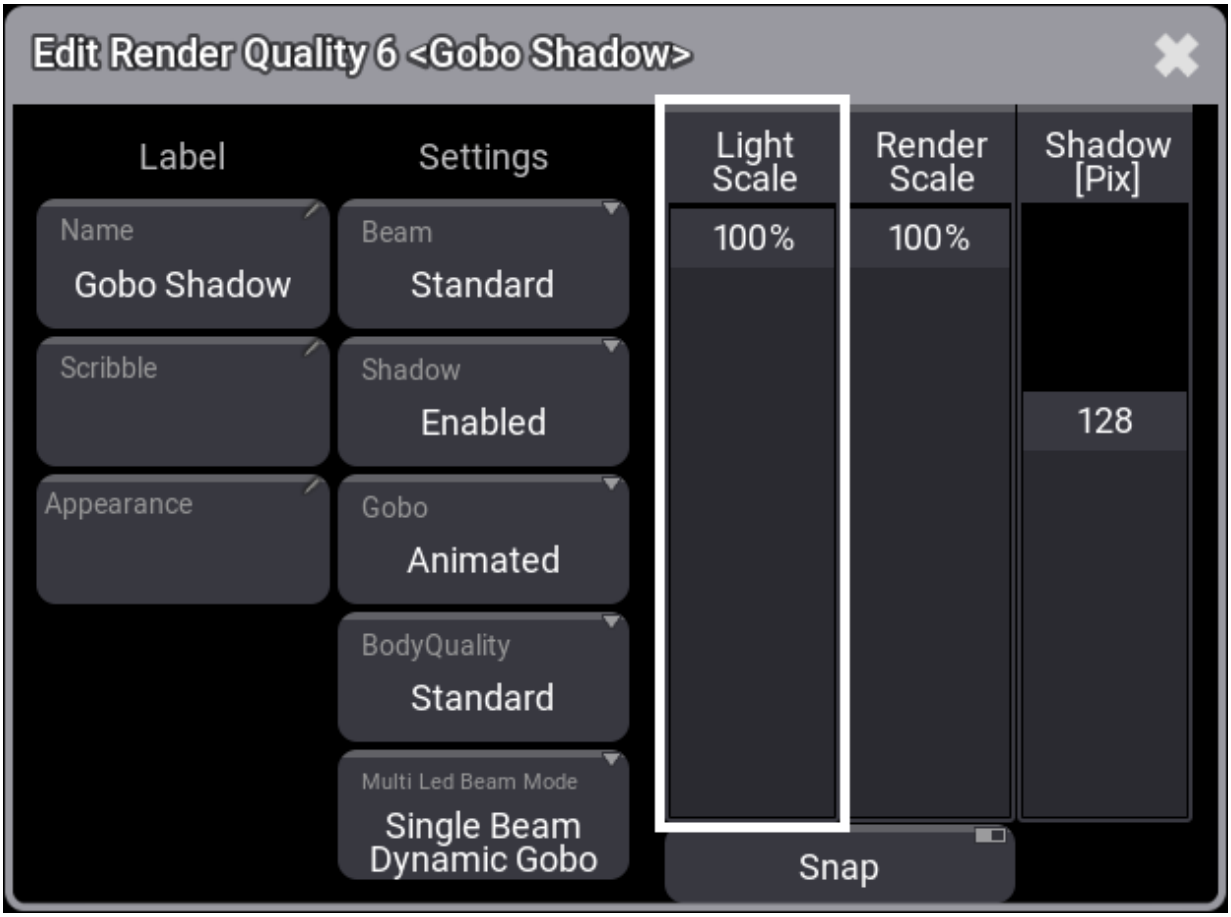
The render quality defines how the light and fixture bodies are rendered in the 3D window.

All the render settings are stored in a Render Quality pool object and can be easily selected in the pool.

RenderQualities pool

There are some predefined render qualities in the pool. There is always a selected object. The 3D can use any of the render qualities or be linked to the selected pool object.

New pool objects can be made and edited to change the render settings.



Render Quality Editor

The editor has the following settings:

- **Name**:
  This is the name of the pool object. Tapping it allows changing the name.
- **Scribble**:
  This is the assigned scribble and tapping it allows select a scribble or create a new one.
- **Appearance**:
  This is the assigned appearance. Tapping it allows select a scribble or create a new one.
- **Beam**:
  This toggles through a list of different simulation qualities that can be selected for the light beam. It goes from low to high quality. The higher quality uses more computer resources.
  - **No Beam**:
    This does not render any beams. This mode does not show any gobos or spot rendering.
  - **Line**:
    This renders a simple line as the beam. This mode does not show any gobos or spot rendering.
  - **Standard**:
    This is a simple light beam from the fixture.
  - **High**:
    This mode adds more reality to the light beam.
  - **High Fancy**:
    This is a more precise calculation of light dissipation.
- **Shadow**:
  Turns On or Off if the beam shall throw shadows when hitting another object. Therefore the 3D object needs the setting **Cast Shadow** enabled within the patch.
- **Gobo**:
  This setting defines if gobos shall be rendered within the beam and spot, or not. It has three different options:
  - **Disabled**:
    Gobos are not rendered.
  - **Enabled**:
    Gobos are rendered.
  - **Animated**:
    This means that in addition to the gobo also gobo animations like gobo shake or gobo wheel spin are rendered. When gobo is only set to enabled, these animations will not be rendered.
- **Body Quality**:
  This defines the quality of the fixture body rendering. It goes from not rendered to ultra render quality. The options are: None, Box, Low, Simple, Standard, High, and Ultra.

- **Multi LED Beam Mode**:
  This defines how multi-emitter fixtures render the light beam. The options are:
  - **Separated Beams**:
    This renders a beam for each emitter.
  - **Single Beam Mean Color**:
    This renders a single beam using the mean color.
  - **Single Beam Dynamic Gobo**:
    This takes the output of the emitters and creates a "virtual dynamic gobo". This is then rendered as a single beam.
- **Light Scale**:
  This changes the resolution of the light rendering in the 3D window continuously. 100% means that the 3D window is rendered with its native resolution. 0% means that the resolution of the light in the 3D window is divided by 5 in width and height. Default: 100%
- **Render Scale**:
  This changes the resolution of the whole rendering in the 3D window continuously. 100% means that the 3D window is rendered with its native resolution. 0% means that the resolution of the 3D window is divided by 5 in width and height. Default: 100%
- **Shadow [Pix]**:
  This changes the resolution of the rendered shadows. The higher this value is, the higher is the resolution of the shadows. This only affects the rendering in Beam Quality "Gobo Shadow", "High Shadow" and "High Shadow Fancy". Default: 128
- **Snap**:
  If "Snap" is enabled, the Render Scale and Light Scale are divided with integer values. The fader then indicates the scale (1/1 to 1/5).

Render Scale and Light Scale can be used in combination. Setting both to 1/2 means that the whole 3D window is rendered with half of its resolution, while the light is rendered with a quarter of its resolution.

Setting Up a 3D Computer

The grandMA3 consoles can show a 3D window with the fixtures and the stage setup. But the consoles are optimized to be the human interface that are used to program the light.

If a high-quality and high framerate 3D render machine is needed then the best solution is to have a high-performing computer with really good graphics cards. This computer needs to have the grandMA3 onPC installed and be optimized for 3D graphics.

The grandMA3 onPC needs to be in a session with the console. The grandMA3 onPC could be logged in as the default 3D user who uses a different **Screen Configuration** (read about screen configurations in the **User Settings** topic) but is the same user profile as the default admin user. The grandMA3 onPC should be logged in with a user that has the same user profile as the user looking at the 3D computer. This ensures that the 3D computer follows the user into preview mode.

The 3D computer could have one big 3D window with all the render settings set to the best quality and the 3D window setting called **3D Priority** should be On (see settings above).

The window can also be set to hide the title bar to maximize 3D space using the **Window Settings**. Other elements like the view buttons, encoder bar, etc. can also be hidden using the **Configure Display pop-up**.

13.10. Camera Pool

The cameras are used in the **3D window** to see the fixtures in the 3D virtual stage.

Multiple cameras can be arranged to see fixtures and 3D objects from different angles and with different camera settings.

The cameras are all in the **Camera Pool**.

Camera pool with some cameras and a selected camera

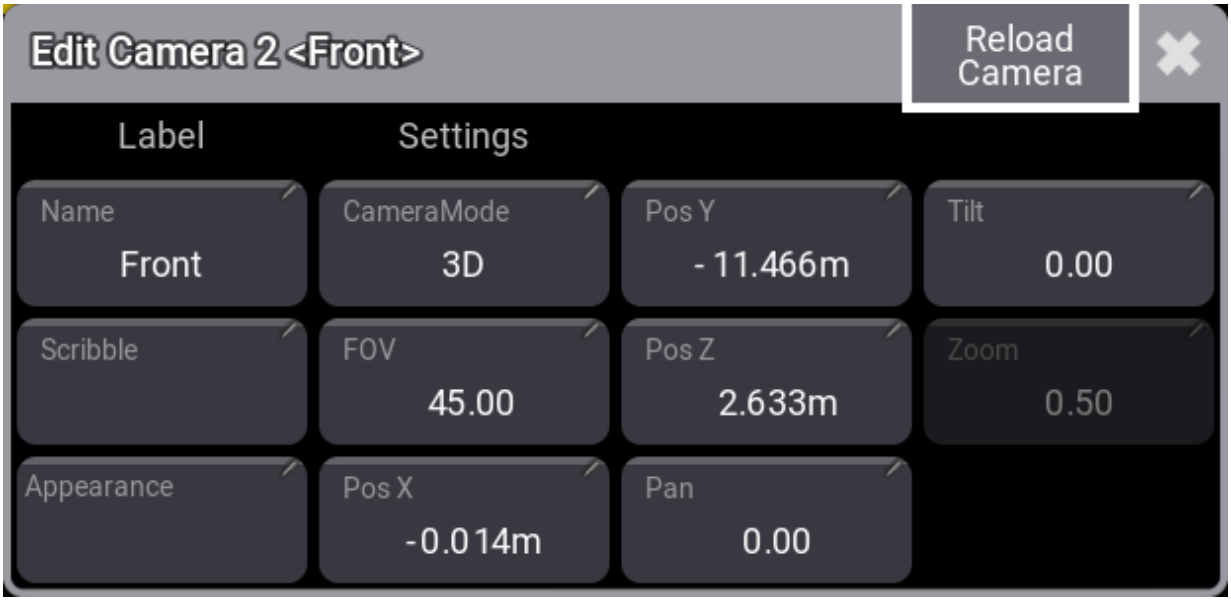
There are some default cameras in the pool in a new show. Most of these can be edited, but the "Auto" camera is locked.

A camera can be selected by tapping it in the pool or by the command line. For example, selecting camera number 3 can be done with this command:

The selected camera has a yellow frame.

Editing an empty pool object creates a new camera that is a copy of the currently selected camera.

Editing an existing camera object opens the **Edit Camera** pop-up:



Edit Camera pop-up for the front camera.

This pop-up contains all the values needed for the camera.

Changing the values here changes the camera.

- **Name:**
  This input box can be used to change the name of the camera.
- **Scribble**:
  A scribble can be assigned to the camera pool object.
- **Appearance**:
  An appearance can be assigned to the camera pool object.
- **Camera Mode:**
  The cameras can have a 3D perspective or a flat 2D projection. Editing this field allows selecting one of the predefined 2D camera angles. Selecting a 2D camera disables some of the other values in this pop-up.
  The 3D camera mode gives access to all values in this pop-up.
- **FOV (Field Of View):**
  The Field of view or Field of vision describes wide or narrow the camera looks at the 3D stage.
- **Pos X:**
  This is the camera position on the X-axis.
- **Pos Y:**
  The camera's position on the Y-axis.
- **Pos Z:**
  The camera's position on the Z-axis.
- **Pan:**
  This value pans the camera. Positive values turn the camera clockwise.

- **Tilt:**
  This tilts the camera up and down. Positive value points the camera down.
- **Zoom:**
  The zoom value is very useful with 2D cameras. The zoom value goes from 0.01 to 1. It goes from narrow to wide.

It might be easier to move the camera and change the pan and tilt values directly in the 3D window. There are nice tools for manipulating the camera. Read more about this in the **3D window** topic.

The values are applied to the camera when Reload Camera is tapped.

Close the pop-up by tapping the in the upper right corner of the pop-up.

13.11. Stages in grandMA3

In grandMA3 there are virtual stages. The fixtures **added to the patch** are placed in a stage. The fixtures can be **positioned** in this 3D virtual stage.

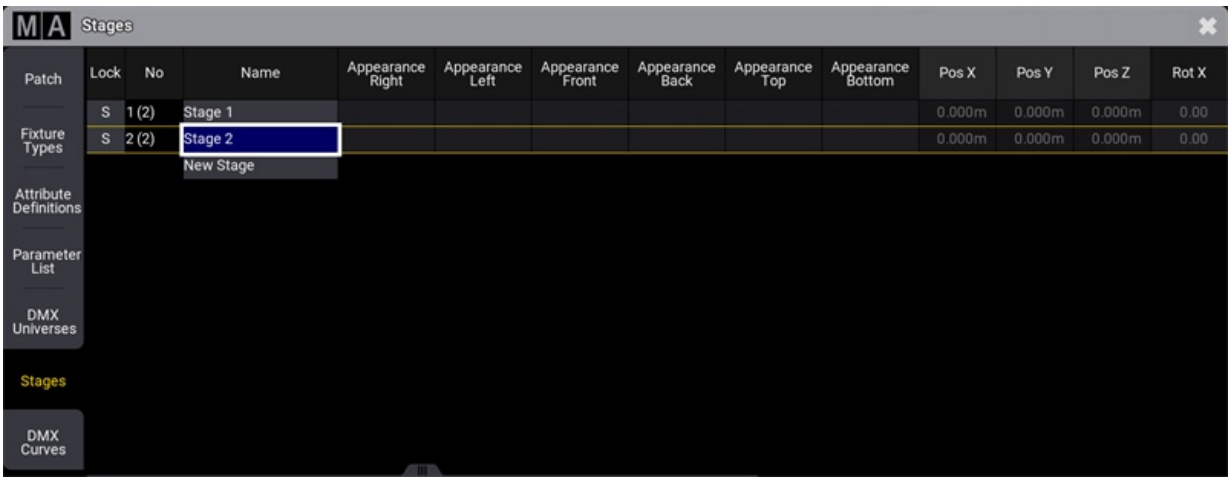
It is possible to create more stages. This could be useful in a house with several physical stages, a television station with several studios, a theme park with several different areas, or for adding a festival rig to a touring show.

The stages have location and dimension information.

Add a Stage in the Patch

1. Press Menu.
2. Tap Patch in the menu pop-up.
3. Tap Stages on the left side.
4. Tap Insert New Stage at the bottom.

A new stage is now added above the line that had focus (yellow frame).



Stages menu with two stages and a global stage

The stages can be named for organizational purposes.

The standard size for a new stage is 30 meters wide (X) and deep (Y) with a zero in the middle. The height (Z) is from zero to 15 meters high.

The stage displayed in the 3D window will automatically expand the visible box, but it will not change the dimension defined for the stage object.

The stages' position and rotation can be changed in the stage setup, but they cannot be visualized together in the same 3D window. The only way to see the stage is in the 3D window and the window can only show one stage at the time. But it is possible to have multiple 3D windows open showing different stages.

The buttons at the bottom make it possible to insert new stages.

The stages can also be deleted. Be careful with this!

Important:
Fixtures are patched inside stages. Deleting a stage also deletes all the fixtures in that stage. Including all the information programmed about those fixtures.

If the stages need to be reorganized then they can be copied, cut, and pasted to move them in the list.

A stage can also be exported and imported. Remember that the stage contains the fixtures. The exported stage object contains the patched fixtures and any other stage objects inside the stage.

For more information about adding fixtures to the stage, please follow the first link at the top.

Edit a Stage

A selected stage in the stage menu can be edited by tapping Edit at the bottom. This opens the **Edit Stage** menu.

Edit stage menu for stage 1

The top part of this edit has input fields for the same settings as the **Stage** menu.

The lower part is used to add and edit a stage **Space** object and stage **Elements**. The elements are flat surfaces that can be combined to create boxes or areas that the light can hit.

A stage space defines a volume in the stage. This space is relevant when using XYZ programming. It defines the size of the space the fixtures can point to. Read more about this in the **XYZ section**.

A stage space can have multiple stage elements inside. These elements are defined by four corners each with an X, Y, and Z coordinate.

Stage spaces and elements can be created, deleted, copied, cut, pasted, oopsed, imported, and exported like many other objects.

Export and Import are especially nice if a lot of time has been spent on creating a 3D set.

Hint:
More complex 3D objects can be added to the Patch. They are found in the "Set" manufacturer.

A new stage always has a Stage Space and a default Stage Element (the "floor").

Adding MArker fixtures automatically adds stage spaces for Target and if needed also for the Movement. Read more about this in the **MArker Fixture topic**.

13.12. Classes and Layers

Fixtures can be organized and grouped in both **Class** and **Layer**.

It is inherited from the MVR (My Virtual Rig) import and from the CAD drawing programs where fixtures can be organized using classes and layers.

In grandMA3 there is no real difference between a class or a layer.

Each fixture can have one of each assigned.

The class and/or layer can be used when applying **filters in the patch**.

Classes and layers can be created manually.

Create a Class or Layer

The following description is for a layer, but the process is the same for classes. Just exchange all the places it says "Layer" with "Class".

1. Press Menu.
2. Tap Patch in the menu pop-up -> this opens the patch menu.
3. Activate Split View in the title bar.
4. Tap Layers.
5. Tap the location in the list where the new layer should be or tap New Layer at the bottom of the list.
6. Tap Insert new Layer.

A new class or layer is now added.

The name can be changed.

Classes and layers can also be created by editing the class or layer field for the fixture. See the description below but instead of selecting an existing class or layer tap New. This creates a new class or layer and the name pop-up appears to give the new class or layer a name.

Assign a Class or Layer to a fixture

Classes and layers can be assigned to fixtures in the **patch menu** when it is in **Full** column mode or when the **Split View** is active in layers or classes.

They also appear as options in the Fixture Wizard (read about it following the link above) when the column mode is full.

Editing the class or layer cells opens small select pop-ups where the existing classes or layers can be selected. There is also the option to select **None**. This can be used to remove a class or layer from a fixture.

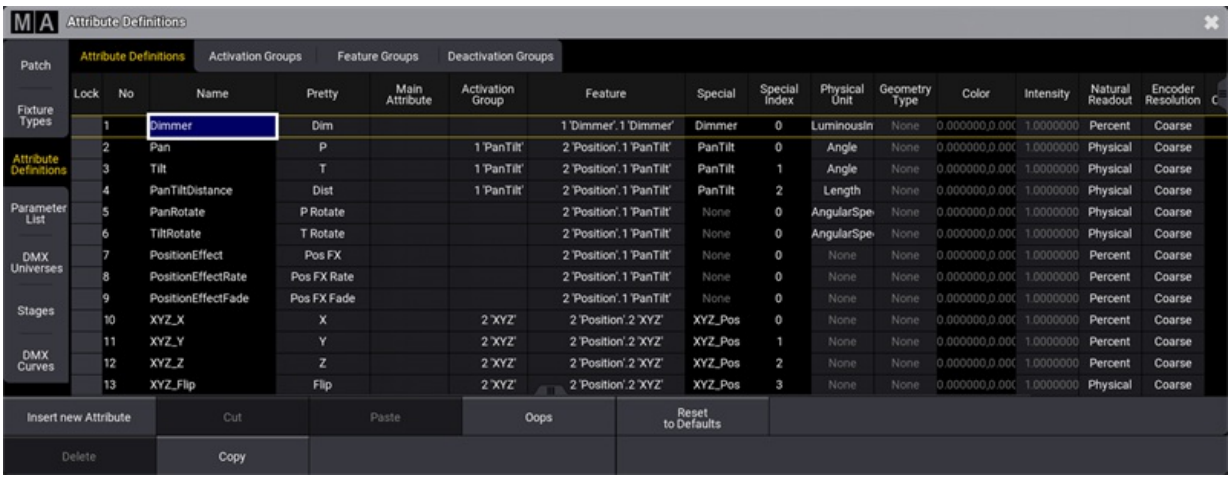
13.13. Attribute Definitions

Attributes are the building blocks of **fixture types**. The same building blocks are used throughout the console and they are what is controlled using the **Encoder bar** when **operating fixtures**.

Attributes definitions describe the relation between Main Attributes and sub-attributes. Attributes are organized in **Activation Groups**, **Feature Groups**, and **Deactivation Groups**.

There is a set of 366 predefined attributes with a complete setup and relations. They can be edited but it is also possible to create custom attributes.

It is all done from the sub menus in the patch menu.



Attribute Definitions

The attribute definitions have the following columns:

- **Lock**:
  This column can be used to lock the rows and protect them from editing.
- **No**:
  This is an auto-generated row number.
- **Name**:
  This name needs to be a unique identifier. When it is typed, it is locked and cannot be edited.
- **Pretty**:
  This is the name displayed above the encoders. This input accepts local country characters if typed using grandMA3 onPC.
- **Main Attribute**:
  This is used to define a hierarchy between the attributes. Editing a cell in this column opens a small **Select Main Attribute** pop-up. Selecting the main attribute makes this attribute a subattribute of the main attribute.
- **Activation Group**:
  Activation groups are used to activate a group of attributes as soon as one of the members in the group is activated. Read more in the **Activation Group topic**.

- **Feature**:
  Features are part of the hierarchy system. Attributes need to be a part of a feature. Read more in the **Feature Group topic**.
- **Special**:
  This is information only. It shows how the attribute is visualized in the **3D window**.
- **Special Index**:
  This is information about the index number for the special value.
- **Physical Unit**:
  The physical units describe the physical properties of the attribute. Read more **below**.
- **Geometry Type**:
  This can be used as a standard geometry type when using the attribute in new Fixture Types. Read more about geometry types in the **Fixture Types topic**.
- **Color**:
  Any attribute can have a color but it is only relevant for the attributes that use "ColorComponent" as the physical unit. Here the color information defines the color of the LED emitter.
- **Intensity**:
  The intensity value can be used to define the relationship between the Color Components. It is used by the color engine to compensate if, for instance, the blue emitters are much brighter than the green. This intensity information can also be defined in the fixture types. The information in the attribute definition is used if nothing is defined in the fixture type.
- **Natural Readout**:
  This is the readout type used when **Natural** readout is selected.
- **Encoder Resolution**:
  This defines the resolution of the encoder controlling the attribute.
- **Log Channels**:
  This is information only. It shows how often the attribute is used in logical channels of fixture types added to the show.
- **Channel Functions**:
  This is information only. It shows how often the attribute is used in channel functions of fixture types added to the show.
- **Hide**:
  This can be used to hide the attribute on the interface. It will not be visible on encoder bars or on any sheets. If the attribute is matched with a DMX, channel then it will still be visible in the DMX sheet.

Add an Attribute Definition

It is recommended to use one of the factory-defined attributes, if possible.

Custom attribute definitions can be created following these steps:

1. Press Menu.
2. Tap Patch.
3. Tap Attribute Definitions on the left side.
4. Select the line in the list, where the new definition should be above.

5. Tap Insert New Attribute at the bottom.
6. Fill in all the relevant cells.
7. Close all the menus and tap Ok to accept the changes to the setup.

Now the definition exists and can be used to create new custom devices.

Physical Units

The physical units are used to describe the physical part of the attribute - if relevant.

For some attributes, this is not relevant. For instance, the selection of a Gobo is linked to an image of the gobo projected, but it is not defined as a physical unit (it is a wheel slot information). Physical unit information is relevant for the attributes that define the rotation of the selected gobo. So the attribute defining the gobo position (index) uses **Angle** as a physical unit. The attribute defining the continuous rotation of the gobo uses **AngularSpeed** as the physical unit.

Editing the cells for physical units (see the column descriptions above) opens the **Select Physical Unit** pop-up.

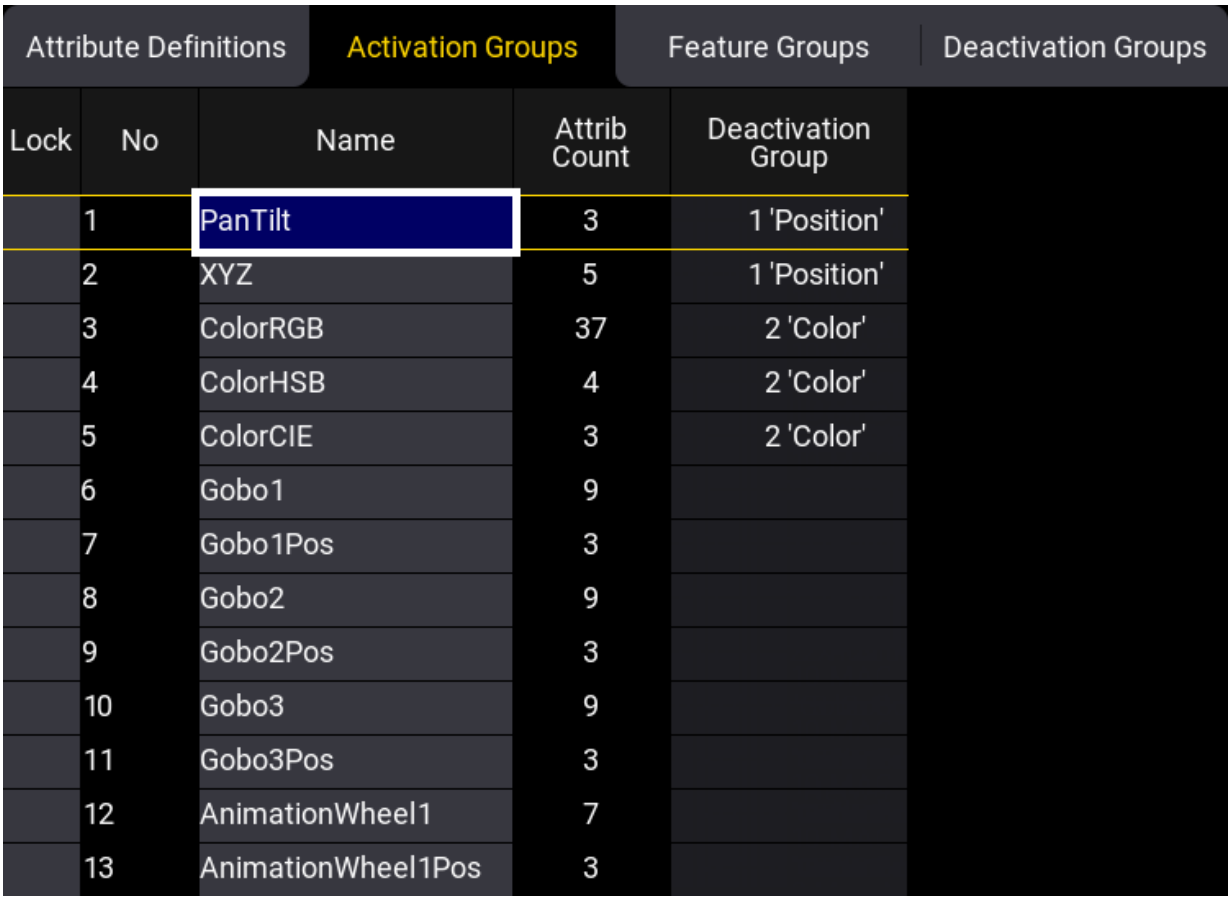
There are many physical units available on the list. Select the one matching the needed unit.

13.13.1. Activation Group

An activation group is used when there are several attributes where it makes sense that they are activated together.

For instance, it often makes sense to store pan and tilt values together. Having them in the same activation group makes this possible. Activating one of them also activates the other.

There are factory-defined groups that are used with the factory-defined attributes. Custom groups can be made and then used in the **Attribute Definitions**. The group needs to be created before it can be used.



The standard Activation Groups

The menu has the following columns:

- **Lock**:
  This can be used to lock the row from being edited.
- **No**:
  This is an auto-generated row number.
- **Name**:
  This is the name of the activation group.

- **Attrib Count**:
  This is a counter showing how many attributes are part of the group.
- **Deactivation Group**:
  This column is used to select a **Deactivation Group** for two or more activation groups. Read more about this in the **Deactivation Group topic**.

Create and use a new Activation Group

Create a new activation group following these steps.

1. Press Menu.
2. Tap Patch.
3. Tap Attribute Definitions on the left side.
4. Tap Activation Groups on the top tabs.
5. Select the line in the list, where the new group should be above.
6. Tap Insert New ActivationGroup at the bottom.
7. Edit the name field to add a descriptive name.
8. Optionally edit the **Deactivation Group** cell to add a deactivation group.
9. Tap Attribute Definitions on the top tabs.
10. Assign the new group to the relevant attributes in the **Attribute Definitions** menu.
11. When finished, close the menus and tap Save and Exit in the pop-up asking if the changes should be kept.

For more information on how to use the **Activation Group** read the **Attribute Definition topic**.

13.13.2. Feature Group

Feature groups are part of the structure and hierarchy of the entire show.

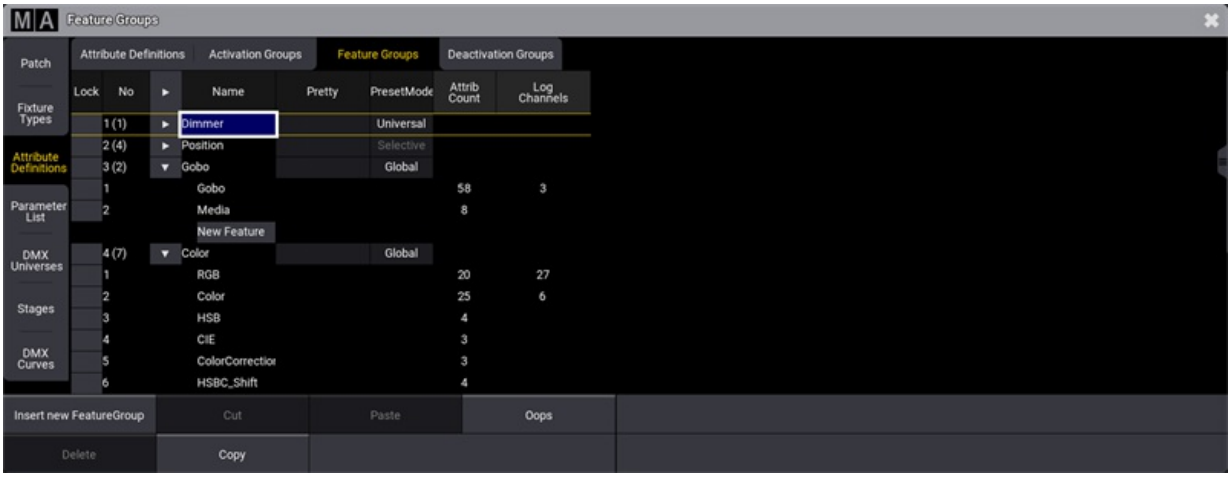
This structure is visible and used in the **Feature Group Control Bar** in the **Encoder Bar**.

A **Feature Group** contains at least one **Feature**.

The feature is assigned to attributes in the **Attribute Definitions**. All attributes need to have an assigned feature.

A feature group automatically has a **Preset Pool** where the attributes in the feature group can store values. Read more about presets in the **Presets topics**.

The feature group and feature need to be created before they can be used.



The standard Feature Groups with Features

The menu has 6 columns:

- **Lock**:
  This can be used to lock the row from being edited.
- **No**:
  This is an auto-generated row number for the feature group. The number in parentheses is the number of features inside the group.
- **Name**:
  This is the name of the feature. This needs to be a unique name.
- **Pretty**:
  This is a custom label for the feature group. It can have local characters or a different spelling. For instance "Color" could be labeled "Colour".
- **Preset Mode**:
  This is the default type used when storing values in the preset pool.
- **Attrib Count**:
  This is information only. This is a counter showing how many attributes use the feature.

- **Log Channels**:
  This is information only. It shows how often the feature is used in logical channels of fixture types added to the show.

Create and use a new Feature Group and Feature

Create a new feature group and feature following these steps.

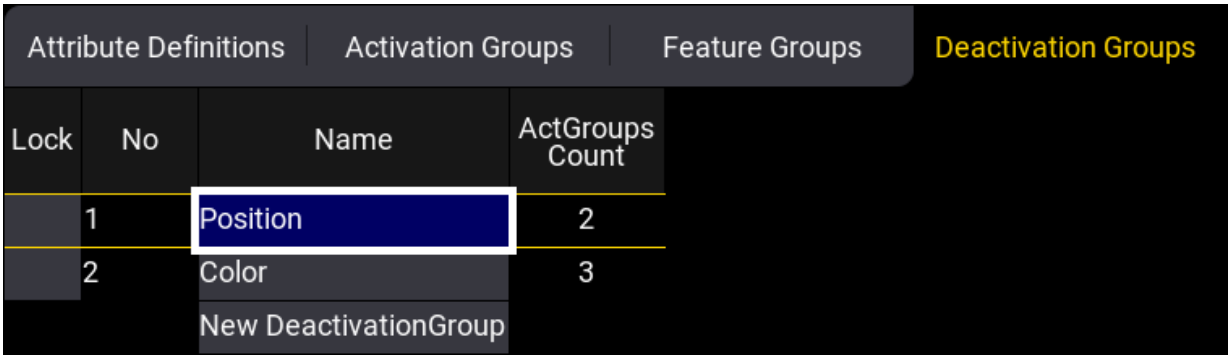
1. Press Menu on the console.
2. Tap Patch to select the patch menu.
3. Tap Attribute Definitions on the left side.
4. Tap Feature Groups on the top tab.
5. Select the row in the list, where the new group should be above.
6. Tap Insert New FeatureGroup at the bottom.
7. Fill in the name of the group.
8. Tap the triangle icon ▶ next to the new group to expand the group.
9. Tap the New Feature area in the new group.
10. Tap Insert and write the name of the feature. Repeat until all the needed features have been created.
11. Tap Attribute Definitions on the top tab.
12. Assign the new features to the relevant attributes.
13. When finished close all the menus and tap Save and Exit in the pop-up asking if the changes should be kept.

For more information on how to use the **Feature Group** read the **Attribute Definition topic**.

13.13.3. Deactivation Group

Deactivation Groups are groups of **Activation Groups**. A deactivation group makes sure that only one set of valid attribute values are active for the actual DMX channels being output. This is to avoid DMX channels getting conflicting information. The deactivation group knocks out other groups of attributes when a value is assigned to a different activation group than the one already active.

For example, when having active pan and/or tilt values in the programmer, the deactivation group "Position" takes care, that these attributes will be knocked out when activating XYZ attributes instead.



Default deactivation groups

By default, there are 2 automatically generated deactivation groups: Position and Color.

The menu has the following columns:

- **Lock**:
  This can be used to lock the row from being edited.
- **No**:
  This is an auto-generated row number.
- **Name**:
  This is the name of the activation group.
- **ActGroups Count**:
  This is the number of activation groups using the deactivation group.

The deactivation groups are assigned to activation groups in the **Activation Groups** tab.

Create and Use a New Deactivation Group

Create a new deactivation group following these steps.

1. Press Menu.
2. Tap Patch.
3. Tap Attribute Definitions on the left side.
4. Tap Deactivation Groups on the top tabs.
5. Select the line in the list, where the new group should be above.
6. Tap Insert New DeactivationGroup at the bottom.

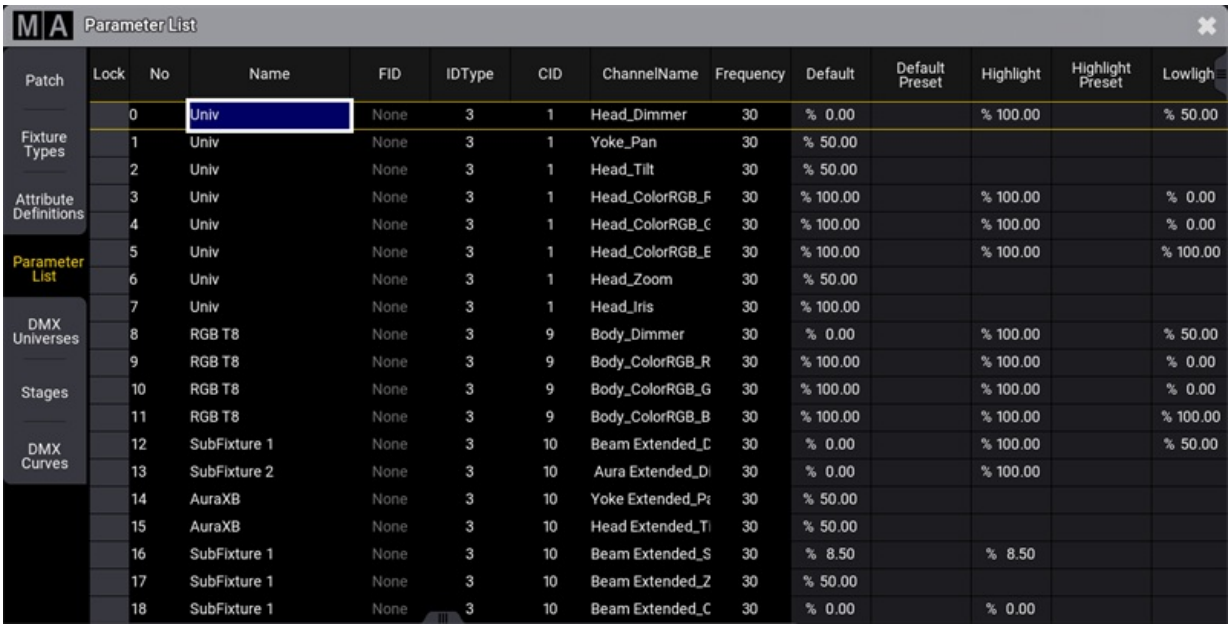
7. Edit the name field to add a descriptive name.
8. Tap Activation Groups on the top tabs.
9. Assign the new deactivation group to the relevant activation groups in the **Activation Groups** menu.
10. When finished, close the menus and tap Save and Exit in the pop-up asking if the changes should be kept.

13.14. Parameter List

The Patch menu gives access to the list of all the parameters by tapping the Parameter List button in the menu on the left of the **Patch Menu**.

These parameters are also called RTChannels (Realtime Channels).

This menu lists all the parameters in the show.



Parameter List menu

The menu is a big table with the parameters in rows. This is a short explanation of the different columns:

- **Lock**:
  This indicates a "Yes" if the row is locked.
- **No**:
  This is the parameter number.
- **Name**:
  This is the name for the fixture.
- **FID**:
  This is the FID of the fixture using the parameter.
- **IDType**:
  This is the ID number of the fixture ID type.
- **CID**:
  This is the CID for the fixture using the parameter.
- **ChannelName**:
  This is the name of the channel in the fixture definition.

- **Frequency**:
  This is the DMX output frequency.
- **Default**:
  This is the default value for the parameter.
- **Default Preset**:
  If the default value is referencing a preset, then the preset number and name are shown here.
- **Highlight**:
  This is the highlight value for the parameter.
- **Highlight Preset**:
  If the highlight value is referencing a preset, then the preset number and name are shown here.
- **Lowlight**:
  This is the lowlight value for the parameter.
- **Lowlight Preset**:
  If the lowlight value is referencing a preset, then the preset number and name are shown here.
- **Coarse**:
  This is the DMX address for coarse control of the parameter.
- **Fine**:
  This is the DMX address for fine control of the parameter.
- **Ultra**:
  This is the DMX address for ultra-fine control of the parameter.
- **DMX Curve**:
  This field can be used to select one of the existing DMX curves. Learn about DMX curves in the **DMX Curve topic**.

Only the default, highlight, lowlight values and presets, and the DMX curves can be changed in this list.

A filtered version of this list is also shown when a fixture is edited. It is filtered to only show the attributes (or RTChannels) for the fixture being edited.

Change the Default, Highlight, and Lowlight Values

The values in the Default, Default Preset, Highlight, Highlight Preset, Lowlight, and Lowlight Preset columns can be edited in this menu or when editing the fixture.

It is also possible to store the programmer content to these elements. It automatically assigns presets that might be in the programmer into the preset column.

It is stored using the **Default**, **Highlight**, and **Lowlight** keywords.

Example:

Store a new default position for a group of fixtures.

1. Clear the programmer for any existing values and selection.

2. Select the desired fixtures.
3. Give them the desired position for instance from a preset.
4. Be careful to not have undesired values in your programmer (for instance, intensity values).
5. Execute this command: **Store Default**.
6. Clear the programmer and verify the new default position.

To reset any of the special values, set the corresponding attributes to Remove values in the programmer and store them to the desired type of special values. Or set any value for the corresponding attributes and do a store remove.

To store remove the currently active values for the default values, type:

```
User name[Fixture]> Store Default /Remove
```

For resetting the special values of highlight and lowlight, follow the example above but replace Default with Highlight, or Lowlight.

In addition, storing Release values to the special values will do the same as Remove.

Important:
For which fixtures the special values will be changed when using the command line approach, the Use Selection setting and the If not empty setting of the Store Options are important.
To learn more about the Use Selection setting and the If not empty setting, read in the **Store Options topic**.

Restrictions:
Presets with more than 1 step ("Phaser presets") will only use step 1 when using them as special values.

13.15. DMX Curves

DMX curves are used by attributes. As a default, they use a linear transition from 0% to 100%. Other DMX curves can be created and assigned to parameters.

There are three different types of DMX curves: MinMax, Switch, and Custom.

Restriction:
A show file can contain 9 999 DMX curves.

Importing fixture types that use special DMX curves adds the curves to the list.

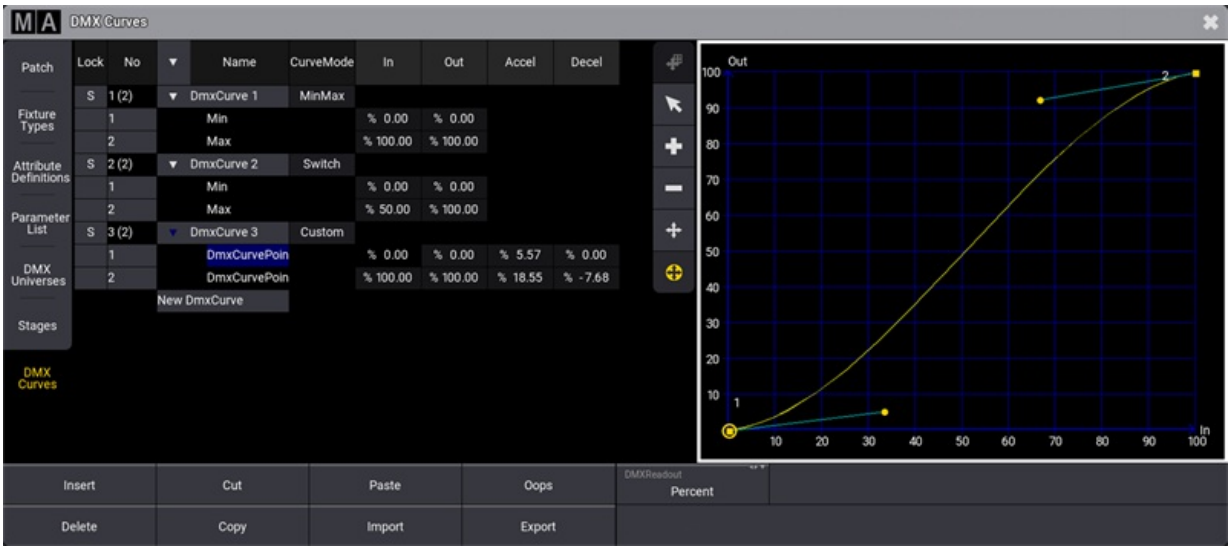
Curve values can be seen and edited in different readouts. These can be changed in the menu at the bottom by tapping the DMXReadout button.

Other buttons at the bottom give access to Cut, Copy and Paste the curves in the list. There is also a button that deletes a curve. If the curve is used, then it might not be possible to delete it.

The curves are displayed, created, and edited in the DMX Curves menu found in the patch.

Open the DMX Curves Menu

1. Press Menu.
2. Tap Patch.
3. Tap DMX Curves on the left side.



DMX Curve menu

The DMX Curve menu is split into left and right sides. The left side is a list of the curves and the right side displays the selected curve.

The curve settings are listed in the left side list. The curves can be unfolded (tap the white right-pointing triangle) and the different points on the curves can be seen and edited in the table mode. In and Out show the relationship between the incoming value and what it is translated to in the curve.

The right side can also be used to edit curves. It has a standard toolset for selection (), addition of point (), removal of point (), move point (), and move point handle (). Not all tools not available for all curve modes.

All DMX curve changes are made inside the patch menu. This means that the menu needs to be closed and changes saved for changes to take effect. Finish all DMX curve changes by leaving and saving the patch changes. Or exit without saving to cancel any changes.

Create a New MinMax Curve

MinMax curves are linear DMX transitions from a minimum to a maximum value. These can be useful to limit output values in for instance dimmers or the pan and tilt movements of fixtures.

1. Open the DMX Curves menu
2. Tap a location in the curves list on the left side where the new curve should be.
3. Tap Insert new DmxCurve at the bottom - the default curve mode for a new curve is **MinMax**.
4. Edit the name to give it a custom name.
5. Edit the Min and Max values to match the needs.

Create a New Switch Curve

Switch curves use two points to create a switch point where the output value instantly changes from one output value to the other.

1. Open the DMX Curves menu
2. Tap a location in the curves list on the left side where the new curve should be.
3. Tap Insert new DmxCurve at the bottom.
4. Edit the CurveMode value to open the small **Select CurveMode** pop-up.
5. Tap Switch in the pop-up.
6. Edit the name to give it a custom name.
7. Edit the Min and Max values of the two points to match the needs.

Having two points in the switch allows for setting a value needed for switching on and another value for switching down. This can help with hysteresis.

Example:

1. Create a new switch curve.
2. Set the **Min** point to In = 70% Out = 0%
3. Set the **Max** point to In = 80% Out = 100%

This creates a switch that needs a value of 80% and above to turn On (output 100%) and a value of 0% to 70% to turn Off (0% output)

Create a New Custom Curve

Custom curves can be used for many different things, for instance, to match the emitter output of LED fixtures.

1. Open the DMX Curves menu
2. Tap a location in the curves list on the left side where the new curve should be.
3. Tap Insert new DmxCurve at the bottom.
4. Edit the CurveMode value to open the small **Select CurveMode** pop-up.
5. Tap Custom in the pop-up.
6. Edit the name to give it a custom name.
7. Points can only be added using the Add tool on the right side.
    1. Tap the Add tool (+)
    2. Tap in the curve where the point should be added
8. Edit the Min and Max values and the Accel (acceleration) and Decel (deceleration) of the points to match the needs. This can be done in the table or by using the graph editor on the right side.

Important:
A curve might give a warning if the output at some point is lower than a previous output value.
Curves with a warning have the name displayed with a red text color.

Export and Import Curves

Curves can be exported to a drive.

1. Select the desired drive.
2. Tap Export.
3. Select the desired drive.
4. Write a name in the name input field.
5. Tap Export in the pop-up.

Curves can be imported to the show.

1. Select a location in the curve list.
2. Tap Import.
3. Select the desired source drive.
4. Select the desired curve.
5. Tap Import in the pop-up.

Assign Curve to a Parameter

The curves can be assigned to parameters in the **Parameter List**.

1. Open the Parameter List in the menu on the left side.
2. Locate the parameter that needs to use the custom curve.
3. Edit the DMXCurve field and select the custom curve in the small select pop-up.

Assign Curve to a Fixture Type

DMX Curves can be assigned to an attribute in a **Fixture Type**.

1. Open the Fixture Types in the menu on the left side.
2. Select the desired fixture in the list.
3. Tap Edit in the menu at the bottom.
4. Tap the DMXModes tab at the top.
5. Edit the DMX Curve field for the desired attribute.
6. Select the wanted curve in the pop-up.
7. Close the FixtureType editor pop-up by tapping the X in the upper right corner.

14. Operate Fixtures

When the fixtures are added and patched in the show, see **Patch and Fixture Setup**, the next step will be to select them and operate them.

Different fixture types can do different things. Most fixtures have some kind of intensity control, but if it is a moving light, it will also have pan and tilt control. If the fixture can change color, it will have a color wheel, an RGB, a CMY, or a scroller. If it is a media server, there are many other settings to control, for instance, which clip to play. All these different control elements are called **Attributes**.

Hint:
It is important to know the patched fixtures and what they can do (what attributes they have). Do not waste a lot of time trying to mix the perfect color if it is a fixture with a fixed color wheel.

One of the primary ways to operate fixtures is via the **encoder bar**.

Encoder bar

This gives fast access to all the attributes available for the selected fixtures.

The **dual encoders** change function depending on the selected **feature group**, for instance, position.

Values that are changed get a red background color. This indicates that the values exist in the **programmer**.

There is a special window for color control - the **color picker**. This is a really nice tool for selecting a color independently of the fixture's actual color system.

The **smart view** gives quick access to the fixture defined values. For instance, most fixtures with a gobo wheel have information about what value is needed to select the gobo slots.

Dimmer

The dimmer is a little special because it is a universal attribute. Almost all fixtures have some intensity control.

There are many ways to set the dimmer value on selected fixtures. Here are some of them:

- Use the level wheel to set the value.
- Use the numeric keys to set a specific value, for example: At 3 0 Please
- Press Full to give the fixture 100% intensity.
- Select **Dimmer** in the encoder bar and use the dual encoders to set a value.
- Select **Dimmer** in the encoder bar and press on the inner ring of the dual encoder. This opens the **calculator**. Use this to set a specific value.
- Edit the value in the fixture sheet to get the calculator to set a value.
- Use a preset.

14.1. Select Fixtures

Fixtures need to be selected to operate them manually and change the values for the different attributes (dimmer, pan, tilt, zoom).

Fixtures can be selected in many different ways:

- Typing the fixture ID using the numeric keys.
- Tapping the fixture in a **fixture sheet**.
- Tapping a **group** containing fixtures.
- Tapping a **preset** containing fixture information.

There are other more complex ways to select fixtures, but these are the most basic ones. One of the two first listed methods needs to be used before groups or presets can be created.

This is an example of how to select fixtures by tapping the fixture ID on the numeric keyboard:

Fixture | 1 | Thru | 5 | Please

This command is visible in the command line input:

This selects all the fixtures with a fixture ID between 1 and 5 (both numbers included).

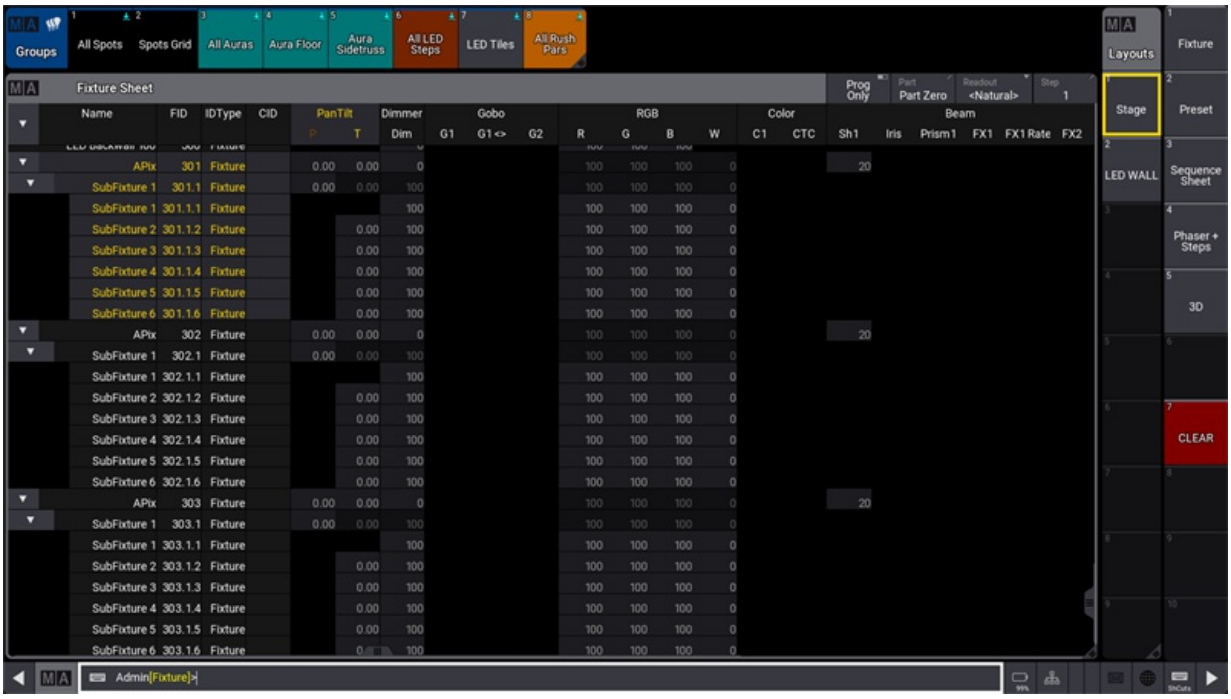
Some keys are used to navigate the sub-selection of fixtures. This function is a part of the **MAtricks** function. The keys have an on-screen version called **Selection Bar**. These can also be used to select fixtures. Read more about it by following the links.

Recursive Selection of Fixtures

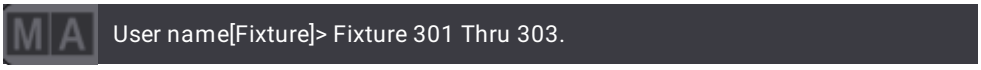
For a fast selection of all subfixtures, a dot (.) can be added to the selection syntax of the main fixture.

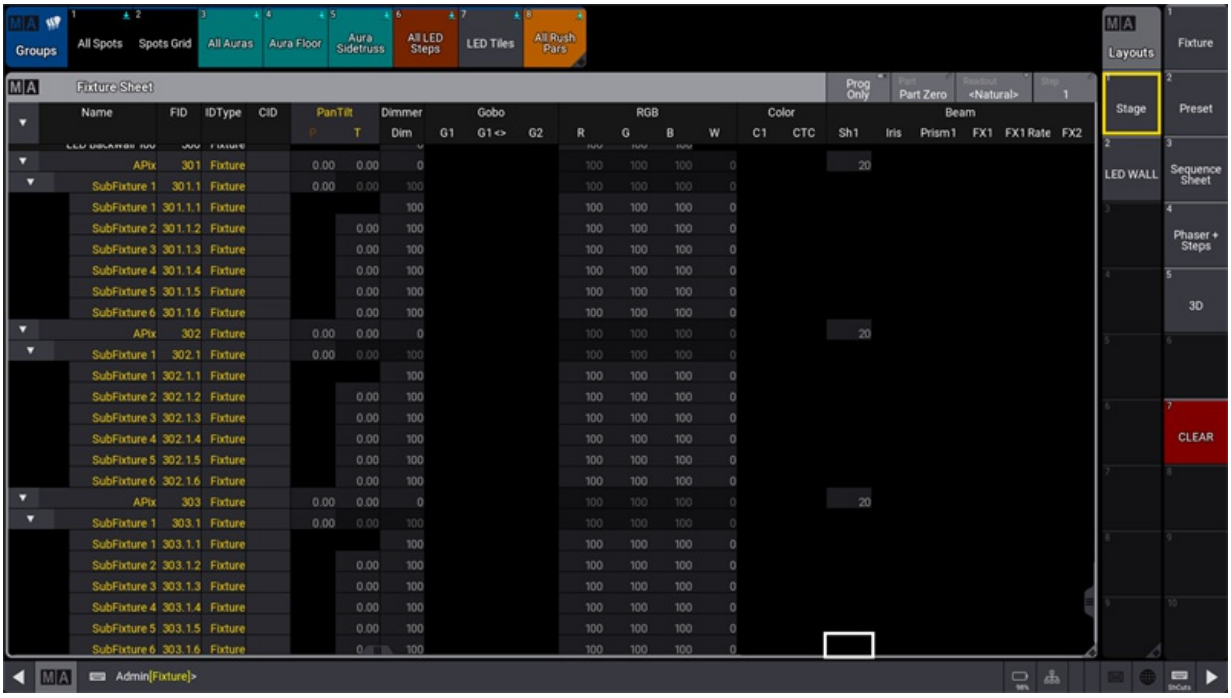
In these examples, we have pathed AlienPix fixtures numbers 301 thru 303.

To select fixture 301 and all its subfixtures, type:



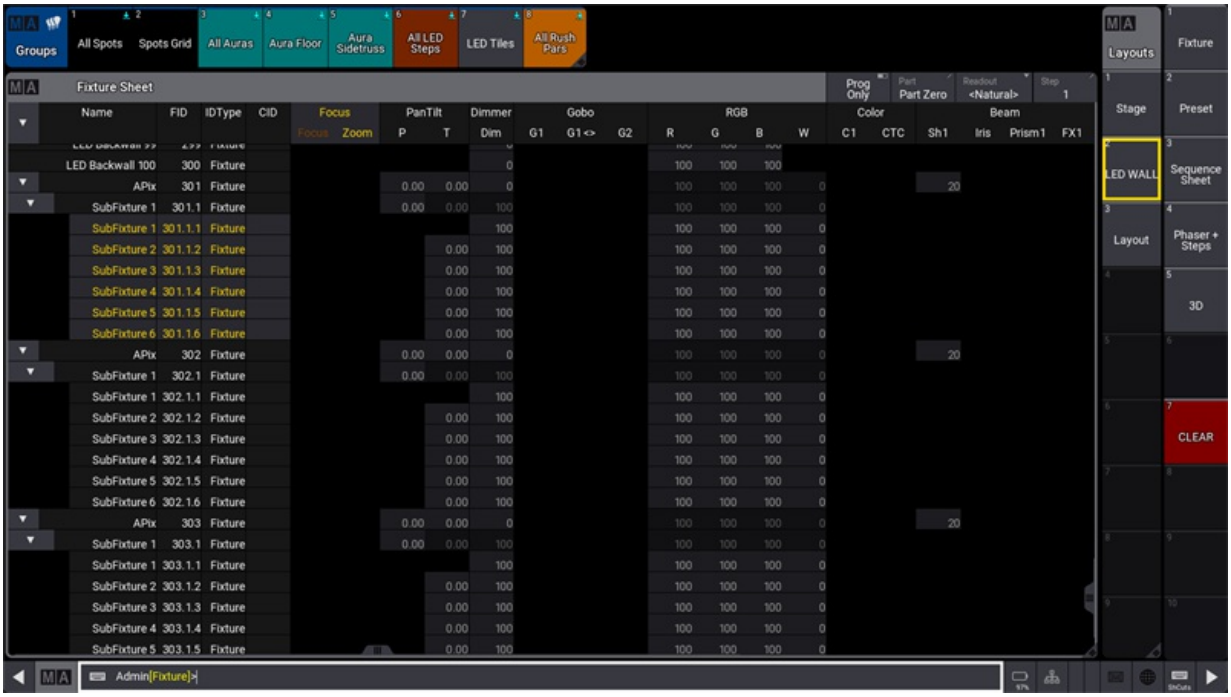
To select all subfixtures of fixtures 301 thru 303, type:





The dot selects all subfixtures downwards from the specified level. Therefore, it is possible to select, e.g., only the pixels of an AlienPix. To do so, type:

User name[Fixture]> Fixture 301.1. Thru



General Selection Syntax

The dot (.) ends the selection on the current level and then steps one level down.

To restart at the top level of the hierarchy within one command, the starting keyword (e.g., Fixture) has to be entered again.

To select subfixtures 8 thru 10 of fixture 1, all subfixtures of fixture 2, and subfixtures 1 to 5 of fixture 3, type:

User name[Fixture]> Fixture 1.8 Thru 10 Fixture 2. Thru Fixture 3.1 Thru 5

To select all subfixtures on the first subfixture level for all fixtures, type:

User name[Fixture]> Fixture Thru.Thru

14.2. What is the Programmer

The programmer is a temporary memory, where the edited values are placed. The values can then be stored or released.

Every user has their own programmer.

The programmer has three levels:

- Selected fixture
- Active programmer values
- Inactive programmer values

The programmer values usually affect the output from the system. But there is a **Blind** function that allows hiding the programmer values from the output.

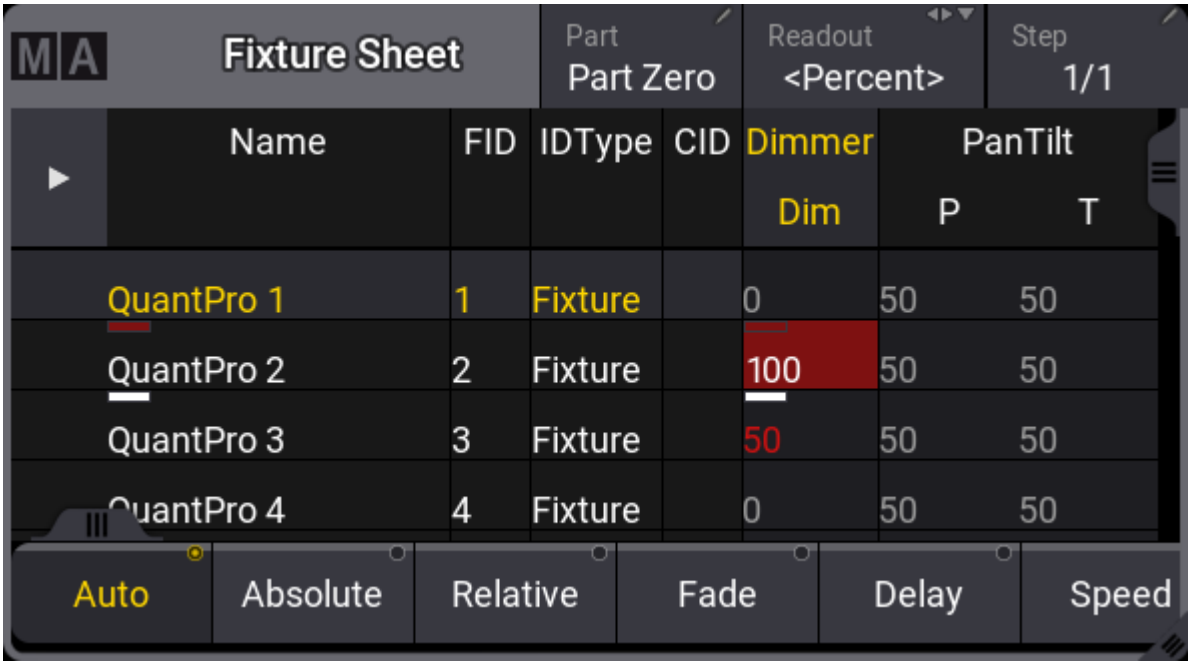
Selected fixtures are the ones that will be affected by encoder input and any intensity changes. Selected fixtures can be identified by having a yellow name and ID text color.

Both active and inactive programmer values can affect the output. The difference is that the active values will be stored using the default **store options** (Use Selection = Active).

Active values have a red background, white value text color, and a red marker in the fixture sheet.

Inactive values have red value text color and a white marker in the fixture sheet.

These are the default system colors. Read more about all the different system colors in the **system (color) topic**.



Values in programmer

When a fixture is selected, it is possible to press Please twice to activate all attributes for the selected fixture. This puts all the current values for all the attributes of the fixture in the programmer. Press Please once more, and the attributes will be inactive values in the programmer.

To release values from the programmer, press Off and then tap the value to deactivate it. Feature groups can also be released from the programmer using Off and then tapping the encoder toolbar.

When there are values in the programmer and a fixture is selected, it is possible to use the clear button to clear each of the three levels:

- Press Clear to deselect the fixture but keep the active and inactive values in the programmer.
- Press Clear again to keep the values in the programmer but as inactive values.
- Press Clear a final time to clear the programmer and release all the values.

Completely clearing the programmer can also be done by holding Clear for more than one second.

The associated keywords are **Clear**, **ClearSelection**, **ClearActive**, and **ClearAll**.

The Clear button in the AT overlay executes the Clear command. For more information, see **At overlay**.

Layers

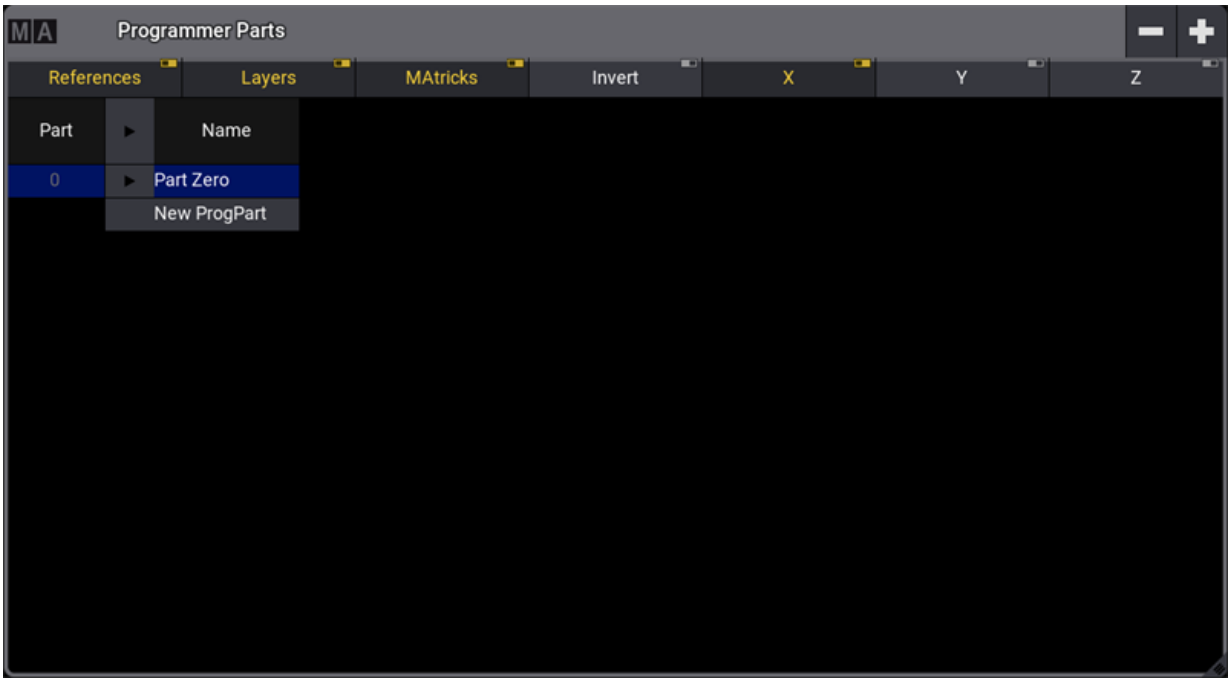
The programmer has several layers. Normal values - for instance, a selected static position - are in the **Absolute** value layer. An absolute value can be affected by a value in the **Relative** layer. For more information, see **Encoder bar**. A dark purple marker indicates relative values. The **Fade** layer can be used to adjust and see individual fade times. Individual fade times are indicated with a green marker. The **Delay** layer can be used to adjust and see individual delay times. An orange marker indicates them. These values are best looked at in the **Sequence Sheet** in tracking mode. Learn more about **Individual Attribute Timing** in Cues and sequences topic.

Programmer Parts

The programmer can have several parts (Maximum: Part Zero + 239 individual parts = 240 parts). The ProgrammerParts window shows all parts of the current programmer. Parts can be labeled here directly.

Using cue parts is a good way to organize your show better. The primary information for every cue is in part 0. Part numbering starts at Part 0. Presets can be called into the programmer parts automatically when calling the preset. This creates then the respective cue parts when storing the programmer content into a cue. Learn more about pool settings at **Preset Pool — Preset Pool Settings**.

To see all active programmer parts, open a Programmer Parts window. For more information, see **Add Windows**.



Programmer parts can be deleted or created using the buttons in the title bar. A part can be selected by tapping the part in the window. A new programmer part can also be created using the command line and the **Programmer keyword**.

To create programmer part 3, which is labeled "Color", type:

```
User name[Fixture]> Store Programmer 3 "Color"
```

Select the programmer part simply by writing the keyword and the number or name.

```
User name[Fixture]> Programmer "Color"
```

When a programmer part is empty, it will be deleted when entering a different cue part.

For example, to store cue 12 with part cue 3, type:

Store | Cue | 1 | 2 | Cue | 3

```
user name[Fixtures]>Store Cue 12 Part 3
```

Freeze

By default, the executors have a higher priority than the programmer. That means that the programmer's values can be changed if a running executor has stored values for the same attributes. To give the programmer a higher priority than the executors, press Freeze. This keeps the adjusted values in the programmer even when the executor is executed.

Blind

Blind is a function that hides programmer values from the output. The blind mode can be toggled On and Off by pressing Blind or by using the **Blind Keyword**.

Turning on blind without any values in the programmer does not change the output. The changing values while in blind do not affect output. If the programmer is cleared before leaving the blind mode, then the output is not affected. Leaving blind with programmer values will take these values to the output. Entering blind mode with values in the programmer removes them from the output, meaning the output is changed.

Program Time

The value changes in the programmer are usually done immediately. But the value change can be done with a timed or manual fade using the **Program Time** function. Read more about it in the **Time Control topic**.

14.3. Fixture Sheet

The fixture sheet shows all the patched objects in the showfile with an FID or a CID. This is normally every object that needs to be controlled in the show. Read the **Patch and Fixture Setup topics** to learn how to add objects to the patch.

The fixture sheet is a window that can be created like any other window. For more information, see **Add windows**.

Fixture sheet in Fixture mode

The sheet has two modes. It can be **Fixture,** or it can be **Channel**. This can be changed in the **window settings**.

Fixture sheet in Channel mode

Hint:
The sheet uses the system colors - read more about them **here**.

Channel mode shows all the objects with an ID and an intensity attribute. It shows only the ID and their intensity values. The **Fixture** mode shows the same objects, but it shows the values of every available attribute.

The currently active world filters the sheet. If this world is different than world one, then the world name and number are displayed in the title bar.

The fixture sheet is divided into rows and columns. In **Fixture** mode, each row represents a fixture or a sub fixture.

The first column is the fixture name. To see the sub fixtures for the current fixture, tap on the arrow on the left side of the fixture name column.

The second column is the FID (Fixture ID). Every fixture patched in the showfile must have a unique FID, CID, or both of them. If a fixture contains sub fixtures, these IDs will be the fixture number, a dot, and a sub number.

The next column shows the IDtype of the fixture. Learn more about the ID types in the **What are Fixtures topic**.

The next column is the CID (Channel ID / Custom ID depending on the ID type). This is the other ID a fixture can have.

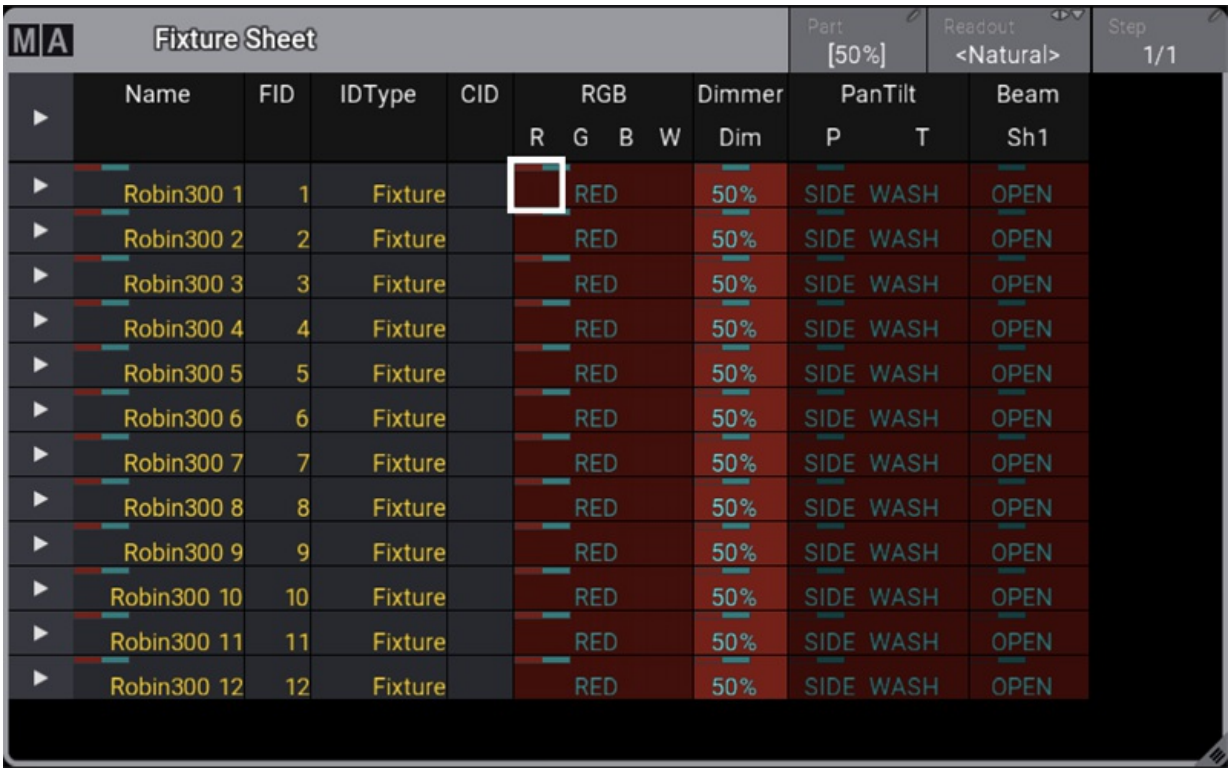
The next columns display the values for the fixture's different attributes, depending on the mask and display settings. Some attributes might have symbols next to them, showing the currently selected gobo or the result of the color attributes.

Channel mode only shows the ID (it prioritizes the CID if available), the intensity value, and a small square showing the combination of current color and intensity.

At the top of the window, there are some buttons with fast access to three different settings. The settings are:

- **Part**:
  This is the Programmer Part - read more about it in the **What is the Programmer topic**.

When the programmer has values from different parts, all cells of attributes that are not part of the currently selected programmer part are displayed in the fixture sheet with a darker background color.



Fixture sheet displaying values for part 1 (Dimmer) with a light red background color.

- **Readout**:
  The readout defines how the attribute values are displayed. The options are:
  - **Auto** - This makes the sheet follow the readout set in the encoder toolbar. Notice that the button in the title bar does not say Auto; it displays the selected readout in angle brackets.
  - **Percent** - This displays the value as a percent from 0% to 100%. This is a whole number. **Percent** Keyword.
  - **PercentFine** - This is also displaying the value in percent. The range is the same, but the resolution is higher. The value has two decimal numbers. **PercentFine** Keyword.
  - **Physical** - This is the physical values defined in the fixture type definition. **Physical** Keyword.
  - **Decimal8** / **Decimal16** / **Decimal 24** - This displays the value in a decimal number in the three available resolutions (256, 65 536, or 16 777 216). **Decimal8** and **Decimal24** keywords.
  - **Hex8** / **Hex16** / **Hex24** - This displays the value in a hexadecimal number in the three available resolutions (FF, FFFF, or FFFFFF). **Hex8** and **Hex24** keywords.
- **Step**:
  This is the step number - learn more about steps in the **Phaser section**.

Hint:
There is a **user profile setting** called **Value Readout**. This is the one being used by the **Auto** option, and it is also the one actually being changed by the encoder toolbar.

To read more about all the other different settings and masks, read the **Window settings** topic.

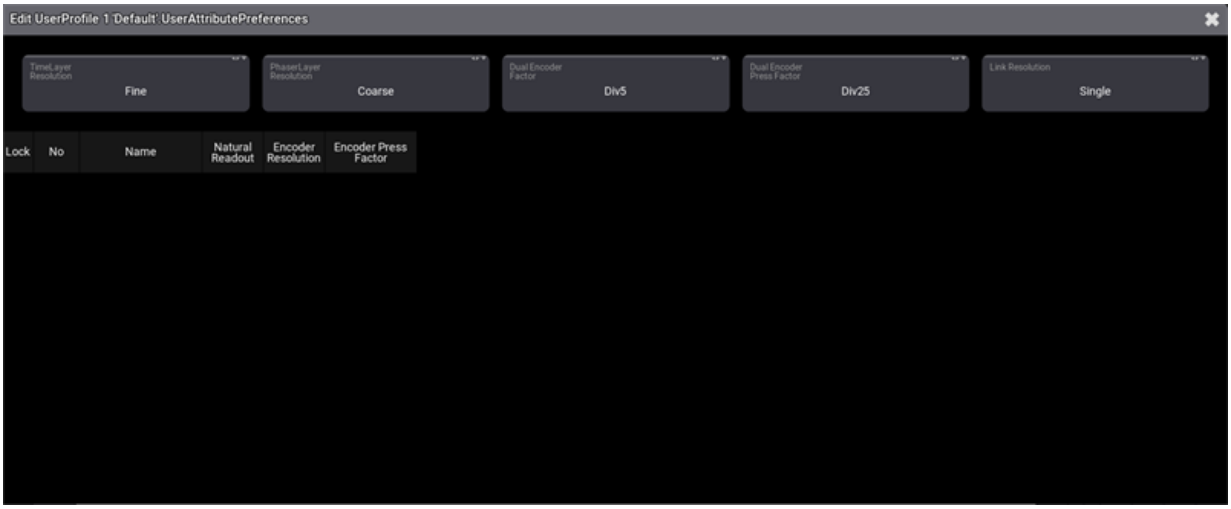
Hint:
It is possible to select fixtures by tapping them in the Fixture Sheet.

14.4. Encoder Resolution

Sometimes precise handling is required, and you may want to change the resolution of an encoder.

Default encoder resolution of attributes can be defined from the user configuration menu:

1. Press Menu, tap Settings, and tap User Configuration. The user configuration menu opens.
2. On the left side of the window, tap Profiles.
3. Tap Edit Encoder Bar. The Default User Attribute Preferences menu opens.



It is also possible to change the encoder resolution directly on an encoder.

To do so:

1. Press and hold MA, this will display the possible resolutions in the channel function area of each encoder.
2. While holding MA, tap Link Resolution and select **Single** or **FeatureGroup**, then tap the resolution area of an encoder, the selected resolution of the encoder will change to the next resolution.
3. When the desired resolution is selected, release MA.

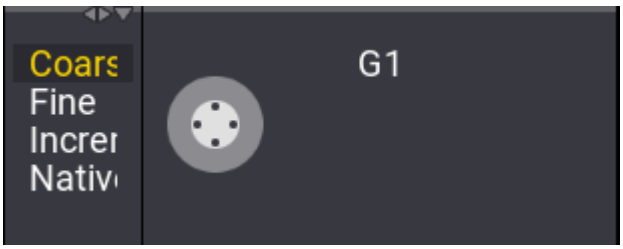
Hint:
- An encoder has 24 clicks for one turnaround.
- 5 turns of an encoder are needed to cross the whole range of the attribute from its minimum to maximum.

The possible resolutions are:

1. **Coarse**: One encoder turn click will change the value depending on the readout:
   - Percent: 1
   - PercentFine: 1
   - Physical: (MaxValue-MinValue)/ (24*5)
     When multiple fixture types are selected, the smallest physical range of a fixture is taken to determine the size of one click. This allows having the same value change when for example, turning the tilt encoder.
   - Dec8: 255/(24*5), therefore 1 click equals 2.125
   - Dec16: 65 535/(24*5), therefore 1 click equals 546.125
   - Dec24: 1 677 216/(24*5), therefore 1 click equals 13 976.8
   - Hex8: 255 (=FF)/(24*5), therefore 1 click equals 2.125
   - Hex16: 65 535 (FFFF)/(24*5), therefore 1 click equals 546.125
   - Hex24: 1 677 216 (=FFFFFF)/(24*5), therefore 1 click equals 13 976.
2. **Fine**: Fine has a 10x finer resolution than coarse.
3. **Increment**: When the resolution is set to Increment, one encoder turn click will change the lowest digit of the displayed readout.
4. **Native**: The value Layers absolute and relative offer this mode to have direct access to the smallest possible value change of the parameter resolution.

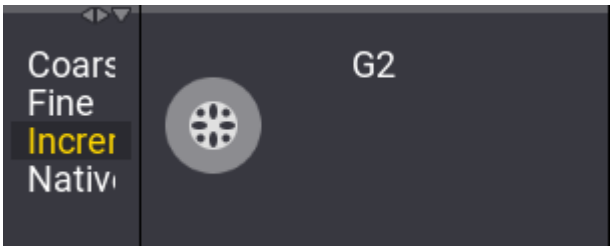
Hint:
When Native mode is selected, if for example, a dimmer channel is based on 8/16/24 bit, one encoder turn always results in one digit change of the DMX output.

An encoder displays the current resolution in the center of the encoder symbol within the encoder bar:

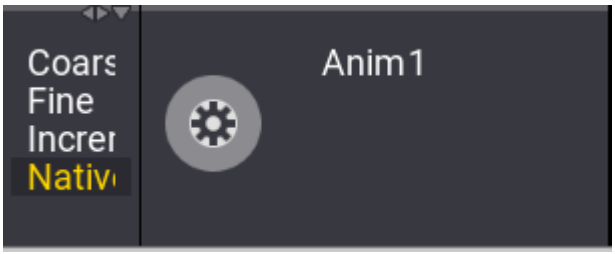


Encoder resolution coarse

Encoder resolution fine



Encoder resolution increment



Encoder resolution native

Furthermore, the factors of value change between turning the inner encoder and all other encoder actions can be defined. See **User settings** for more information.

All factors can be selected from a list of predefined factors:

Predefined Factors	Background Used Factors
Div50	0.02
Div25	0.04
Div10	0.1
Div5	0.2
Div2.55	0.39
Div2	0.5
One	1
Mul2	2
Mul2.55	2.55
Mul5	5
Mul10	10
Mul25	25
Mul50	50
Disable	0

14.5. Using the Color Picker

The **Color Picker** is a window that can be used to select a color in fixtures with a mix color system.

It does not affect color wheels, only color mixing systems.

The color picker provides convenient access to mixing the desired color using several color mixing and selection options. The method used is independent of the fixture's actual color mix system (LED emitters or color subtraction).

Open the Color Picker

The color picker is created like any other window; see **Add windows**. It is found in the **Common** tab.

RGB and HSB color space:

Tap RGB/HSB Space in the title bar of the color picker to switch the color picker's color space. There are four options:

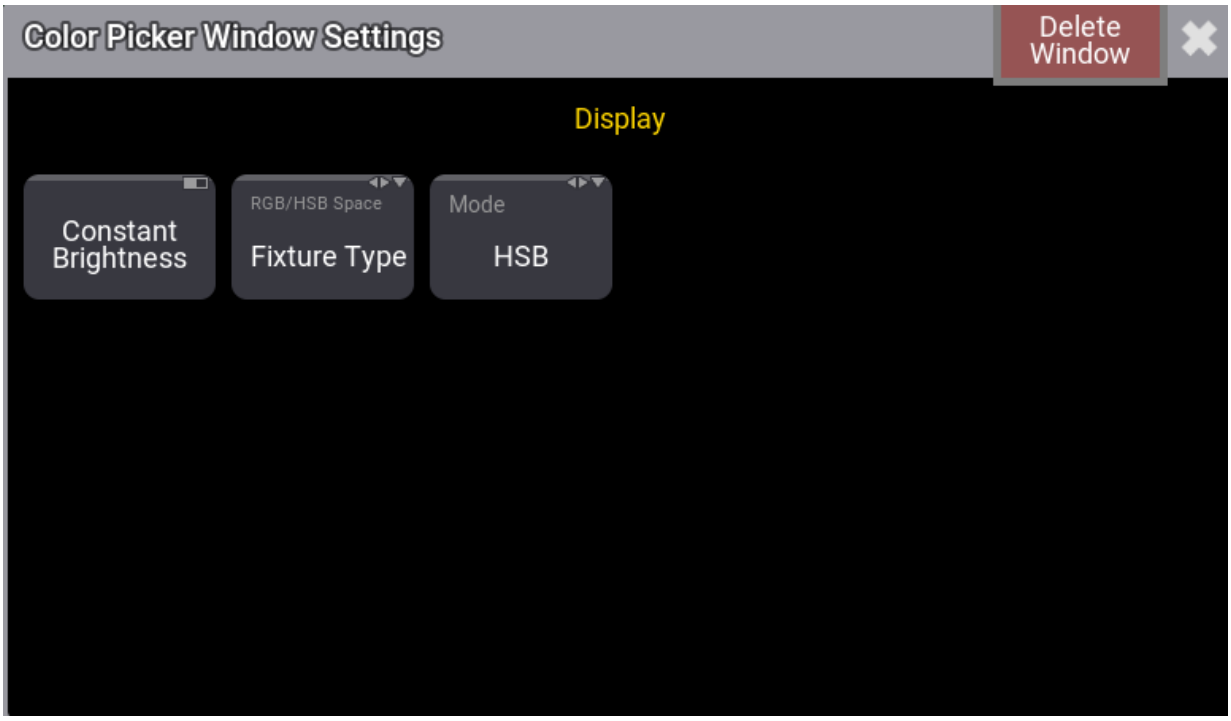
Fixture Type: The color space is defined by the emitters of the respective fixture type.

Standard: Plasa Standard E1.54 for Color Communication in Entertainment Lighting.

Rec.2020: ITU-R BT.2020 or Rec. 2020, is an audiovisual industry standard for ultra high definition (UHDTV).

Rec.709: ITU-R BT 709 or Rec. 709 is an audiovisual industry standard for high definition (HDTV).

Tap MA in the top left corner of the color picker to open the settings.



The mode can be selected using the buttons in the title bar. This is a short description of the different modes.

CIE: A CIE color space area picker with Brightness, Quality, x, and y on-screen faders.

Color Picker: An HSB area with Brightness and Quality on-screen faders.

Fader: On-screen faders to adjust RGB, CMY, HSB, Brightness, and Quality.

Book: This is a swatch book with colors from different filter manufacturers.

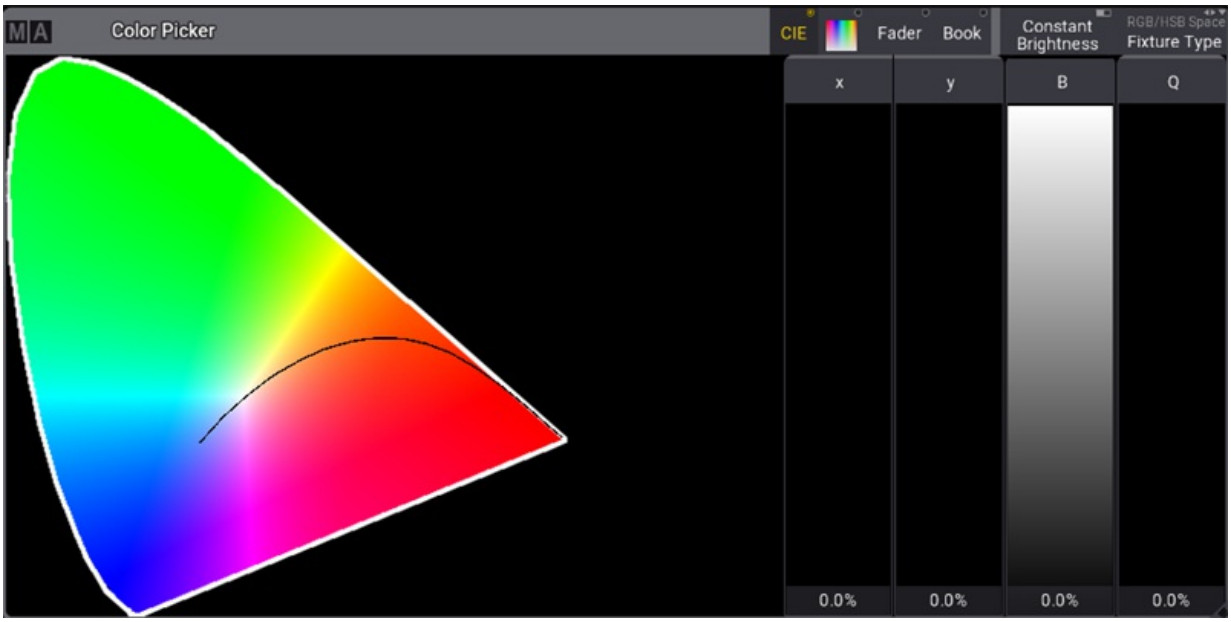
Quality

The Q fader or quality fader is available when the fixtures have a color mix system of more than three colors. It controls how the colors are mixed.
Q at 100 results in kind of small band mixing (the specialized emitters are used). 0% results in a broad band mix. That uses as much emitters as possible to mix the color.

Hint:
The **Align** function can be used together with the color picker.

CIE

The CIE (Commission Internationale de l'éclairage) standard uses a figure that indicates the visible light spectrum.



Color picker with CIE area

The non-shaded part of the figure (the RGB triangle) shows the colors that the specific fixture can mix. Tap inside this area to select a color.

If the fixture has more than 3 mixing colors, the shaded part will become smaller.

The color picker offers the possibility to be used in Constant Brightness Mode. The Constant Brightness mode can be enabled by tapping Constant Brightness in the title bar. The default setting is off.

If the constant brightness is disabled, the selected color is mixed with maximum brightness when the brightness fader is set to 100%. In this case, the fixture's output intensity is not kept constant but changes with the color. If the constant brightness mode is enabled, the maximum overall brightness is limited to the brightness of the fixture's darkest emitter. Changing the color in constant brightness mode does not change the output intensity of the fixture.

Hint:
When enabling the constant brightness mode while the brightness fader is currently positioned above the constant brightness color mixing range, the CONST B fader will become red, showing a value of > 100 %. To ensure constant brightness color mixing, the fader needs to be moved to <= 100 %.

Hint:
The color mixing and the constant brightness mode works the better, the more precise the fixture type's emitter data is.

Except for the Fixture Type color space, the gamut of the selected color space is displayed in the CIE color picker

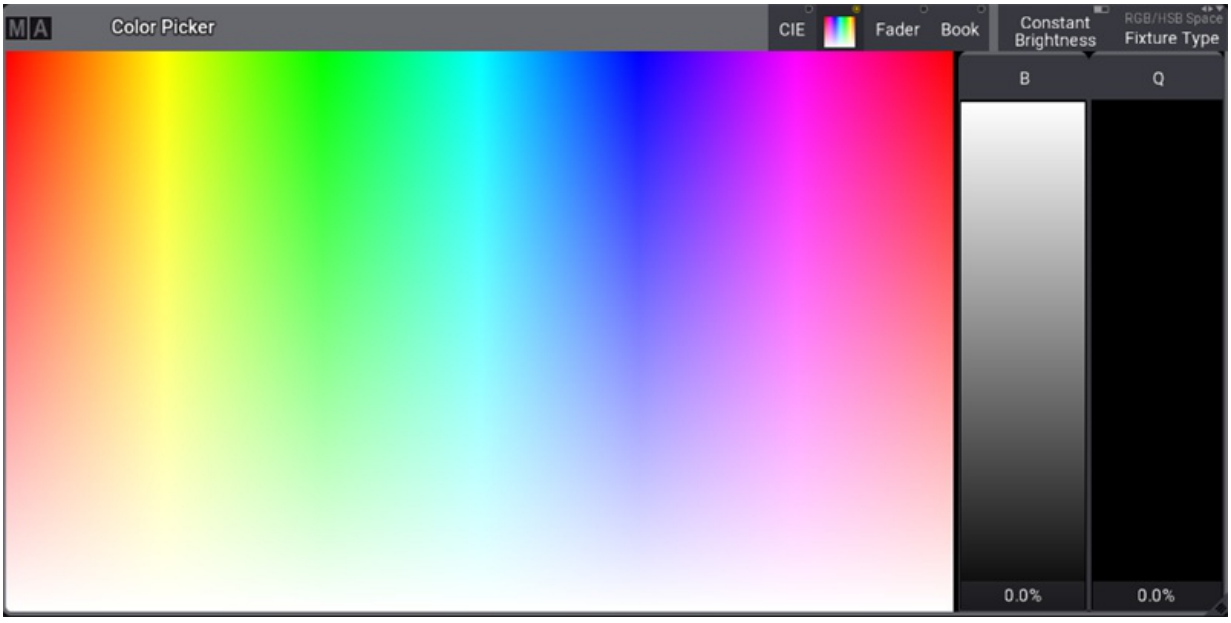
with a white line. The shaded area only depends on the emitters of the fixture. It does not change with the Color Space (only the small white triangle changes with the selected color space). Color mixing in the RGB tab and the HSB Color Picker depends on the color coordinates of the RGB primaries of the selected color space.

Hint:
If a color is picked in the CIE Color picker outside of the gamut of the selected color space, the faders in the RGB tab will show values below 0% or above 100%.

The CIE Color Picker displays the spectral profile (or **curve**) at a specific temperature that corresponds to a specific peak wavelength, and vice versa. As the temperature of the black body increases, the peak wavelength decreases (Wien's Law). The intensity (or flux) at all wavelengths increases as the temperature of the blackbody increases. That is what we call the **black body curve**.

Color Picker

Tap the HSB field symbol in the title bar, also known as a Color Picker, to adjust the color mix.

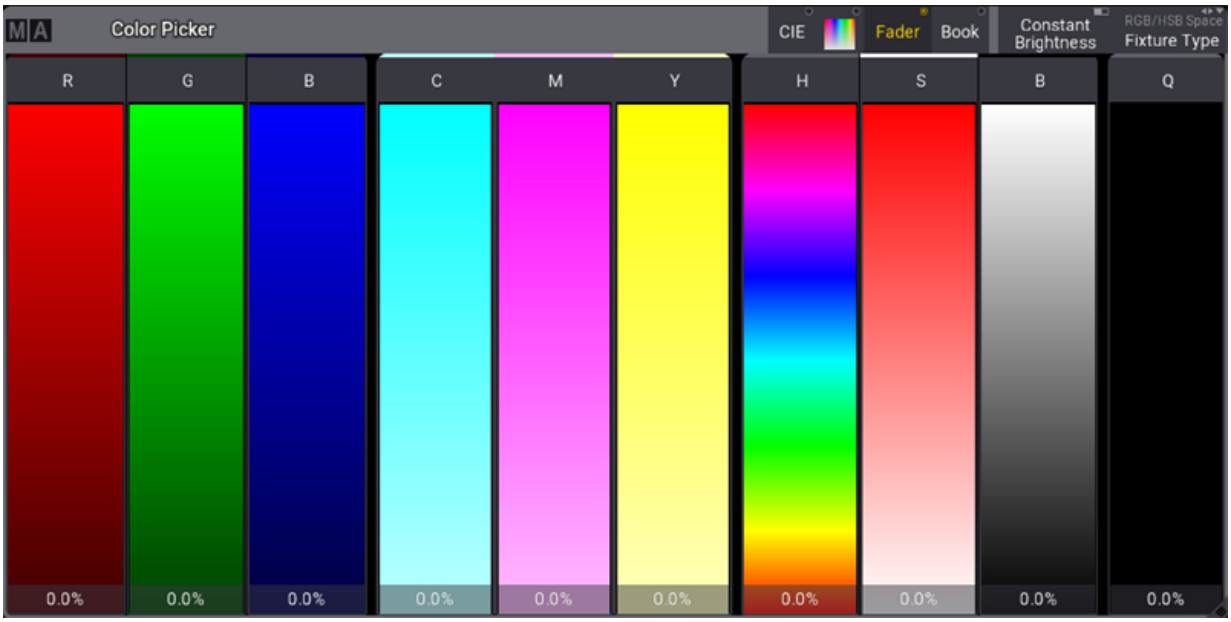


Color picker with HSB area

Here it is possible to tap a color in the HSB field. The x-axis (left/right) is the Hue value. The y-axis (up/down) is the Saturation value, and the B-fader on the right side is Brightness.

Fader

On-screen RGB, CMY, HSB, Brightness, and Quality faders.



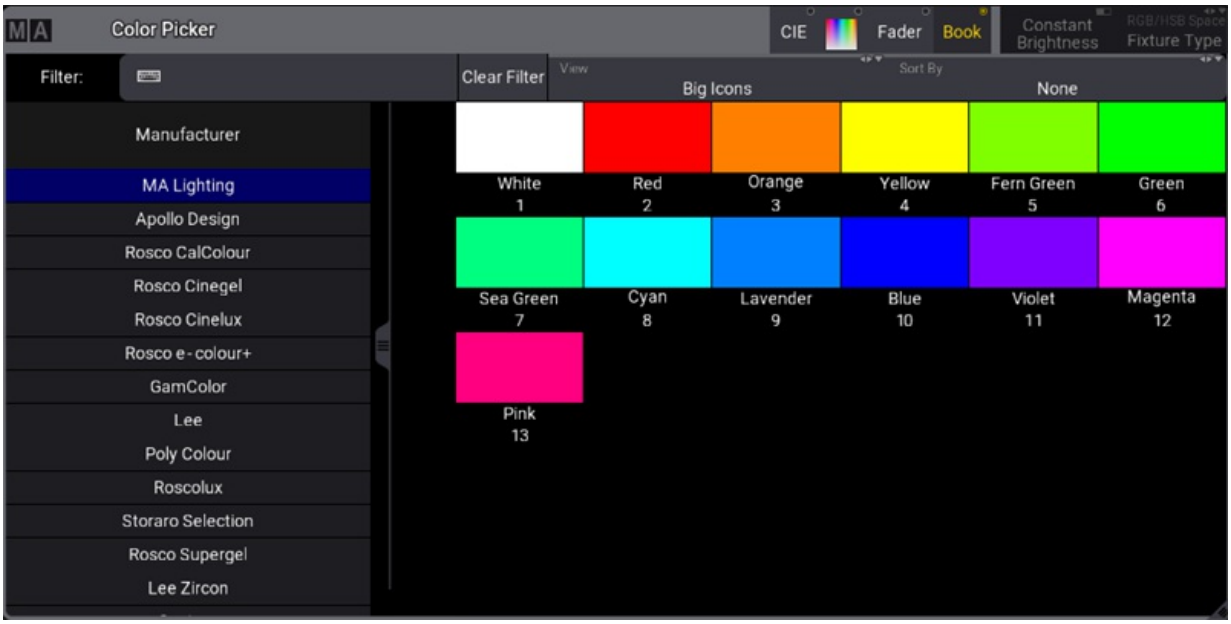
On-screen faders

Here it is possible to adjust the colors using RGB, CMY, or HSB. The Three-color systems are interlinked. This means that adjusting the colors in RGB also moves the CMY and HSB faders.

Book

The swatch book is a library of filter colors from different manufacturers. This can also be accessed using the **Gel keyword**.

The manufacturers are listed on the left side, and the right side displays the manufacturer's colors.



Color picker in book mode

There are three new fields displayed at the top of the window (below the title bar).

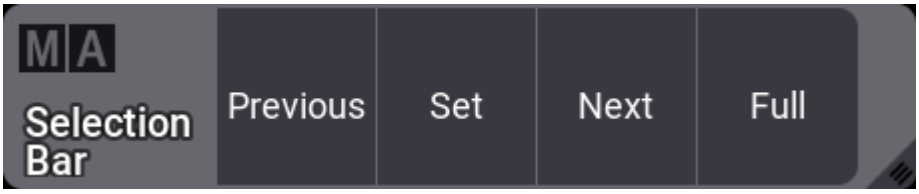
- **Filter** is to enter a name or a key of the color to limit the list of colors.
- **View** defines how the color list is displayed. It is a swipe button with the following options:
  - **Big Icons** - This displays a big color example. Below the color are the name and color key.
  - **List** - This displays a list with Name, Key, and Color columns.
  - **Small Icons** - displays a small color field only.
- **Sort** has three different ways to sort the displayed colors. It has the following options:
  - **None** - the list is not sorted and is displayed in the order in the library.
  - **Key** - Sorted by the color key number.
  - **Name** - The colors are sorted by the color name.

The manufacturer list includes many of the filter manufacturers. It also includes MA Lighting, even though MA does not produce color filters. The MA Lighting library contains all the primary colors and the most used colors.

14.6. Selection Bar

The selection bar is a quick toolbar that gives access to some of the most used hard keys used for selecting fixtures.

It is a window that can be created like any other window. Learn about how in the **Add Windows topic**. It is found in the **More** tab.



Selection Bar

This window can be half field height.

There is only one setting for this window. The background can have an **appearance**.

There are four buttons in the bar:

- **Previous** - Selects the previous fixture.
- **Set** - Resets the MAtricks.
- **Next** - Selects the next fixture.
- **Full** - Gives the selected fixtures a full intensity value on the dimmer.

They behave just like the hard keys. Follow the links to learn about them.

They are also handy when working with **MAtricks**.

14.7. Align

The Align function appears in different places in the software. There is a dedicated **Align key**. There is an **Align keyword**. Align is accessible on the left side of the encoder bar. There is an **Align Bar** that can be created as a window.

Align is used to distribute attribute values between two or more values. There are five different align modes and Off. Read about them below.

The default is a linear transition between the values, but this can also be adjusted. There are four different **Align Transition** options:

- **Linear** (default):
  Spreads the values with the same spacing.
- **Sinus**:
  Spreads the values as if the fixtures were placed on a sinus curve. The values themselves will not represent the sinus form. Depending on the Align mode, this results in smaller value gaps at the beginning and end of the range and bigger gaps towards the center of the range, or vice versa.
- **Slow**:
  The gaps between the values will be small at the beginning of the range and increase towards the end of the range.
- **Fast**:
  The gaps between the values will be big at the beginning of the range and decrease towards the end of the range.

These can be accessed using the **AlignTransition keyword**, the align key, or the **Align Bar**.

The Align function can be used for different attributes. Dimmer, position, and color are the most common. The examples below use tilt or dimmer attributes.

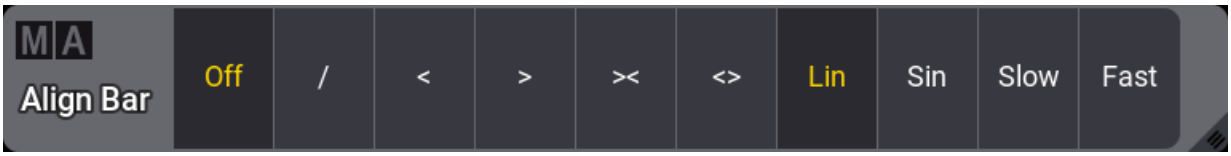
By default, the align mode is set to Off, and the transition is Linear. The result is the encoder will adjust all the selected fixtures equally.

Important:
The selected order of the fixtures is important. The attribute will be adjusted proportionally to the selected order.

The align mode is active until a new attribute is adjusted.

Align Bar

The **Align Bar Window** gives fast access to all the align functions. It has both the align modes and the align transitions.



Align bar with default settings

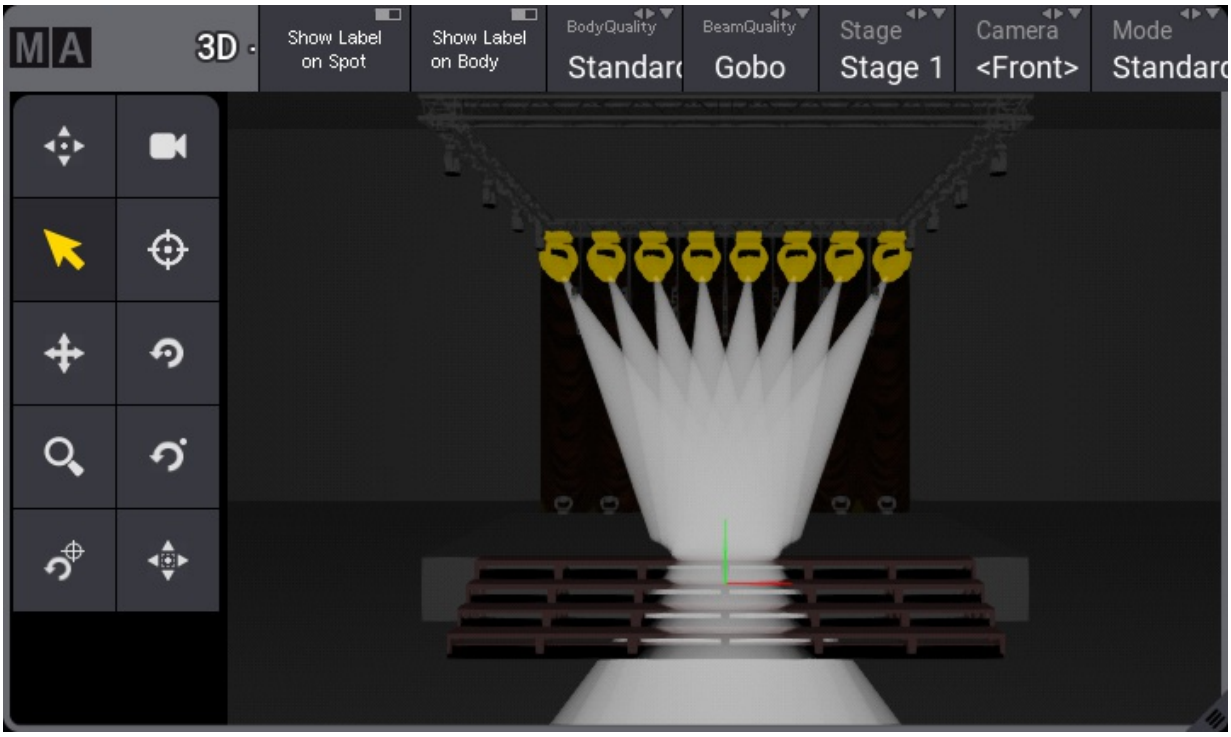
It can be created like any other window using the **Add Window pop-up**. It is found in the **More** tap.

If align is used a lot, then this is a handy window. Read the descriptions above and below to understand the different modes.

Align /

This align mode adjusts the first and last half of the fixtures. The first fixture gets the value from the encoder. This value is then aligned to the center, and then the opposite value is aligned up to the last fixture. The center fixtures do not change value. This mode is great for fanning fixtures.

The example below shows the fixtures turned On and tilted up without align. Then the align mode is selected, and the pan is fanned into the center using the encoder. Turning the encoder the opposite way will make the fixtures fan out.



Align mode / with pan

Align <

This align mode adjusts the last selected fixtures the most. The first fixture does not change value. The values are aligned between the first and last fixtures.

The example below shows fixtures that are selected from left to right. They are all at 0%. Then the align mode is selected, and the dimmer encoder is turned up. The result is apparent in the fixture sheet. The first fixture does not change value. The last selected fixture gets the maximum value from the encoder.

Align mode <

Align >

This align mode is the opposite of the one above. The first selected fixture is affected the most, and the last selected fixture does not change value. The values are aligned between the first and last fixtures.

The example below shows fixtures that are selected from left to right. They are all at 0%. Then the align mode is selected, and the dimmer encoder is turned up. The result is apparent in the fixture sheet - the first fixture changes with the encoder. The last selected fixture does not change value.

Align mode >

Align ><

This align mode adjusts the first and last fixtures the most, and the center selection the least. The value is aligned from the first and last fixtures into the center selection.

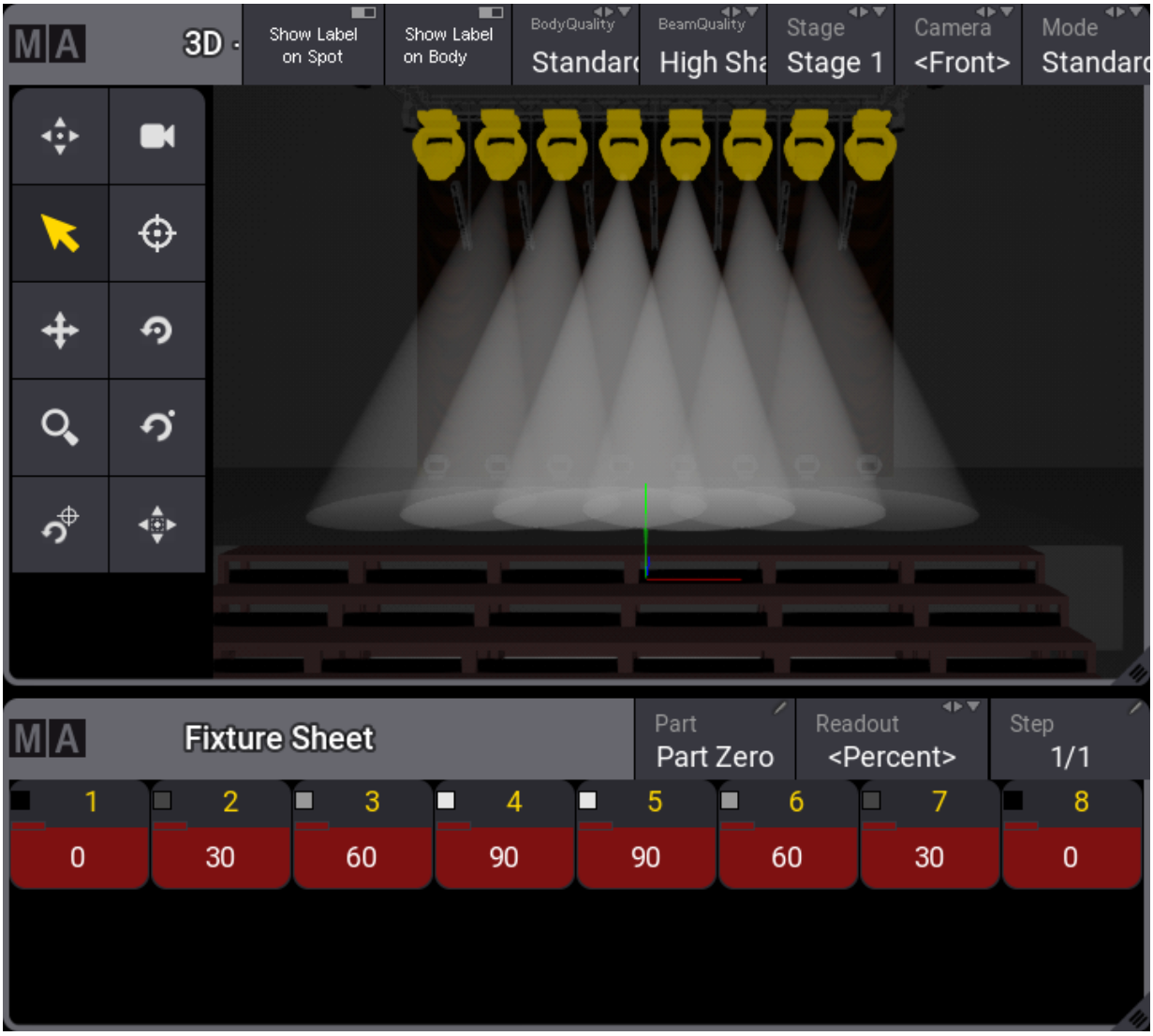
The example below shows fixtures that are selected from left to right. They are all at 0%. Then the align mode is selected, and the dimmer encoder is turned up. The result is apparent in the fixture sheet - the first and last fixture changes with the encoder. The center of the selection does not change the value as much as the first and last.

Align mode ><

Align <>

This align mode does not adjust the first and last fixtures, but the center selection is affected by the encoder. The value is aligned from the center and out to the first and last fixtures in the selection.

The example below shows fixtures that are selected from left to right. They are all at 0%. Then the align mode is selected, and the dimmer encoder is turned up. The result is apparent in the fixture sheet. The first and last fixture does not change with the encoder. The center of the selection changes the value with the encoder.



Align mode <>

Align in Other Ways

It is possible to get the same align functions with the **Calculator** or by the command line.

Tap the encoder belonging to the attribute that shall be aligned, and the calculator appears. It is now possible to enter the align values like the example below.

Calculator

Another way to align attributes is by the command line. Select the fixtures and set the dimmer value like the example below.

14.8. Selection Grid

Fixtures can have information about their position in a 3D selection grid.

Each fixture is symbolized as a box using one space in the grid.

The grid is a way to organize the fixtures in relation to each other but not necessarily their position in the 3D Window.

The grid is used when a range of values are applied. Fixtures positioned at the same X and Y axis but at different Z axis will have the same value applied.

In the first example picture below, the fixtures were selected using a group with the grid information, and then colors were assigned using the **Align** function in the color picker. The value range is applied from the lowest column to the highest column number in the selection window.

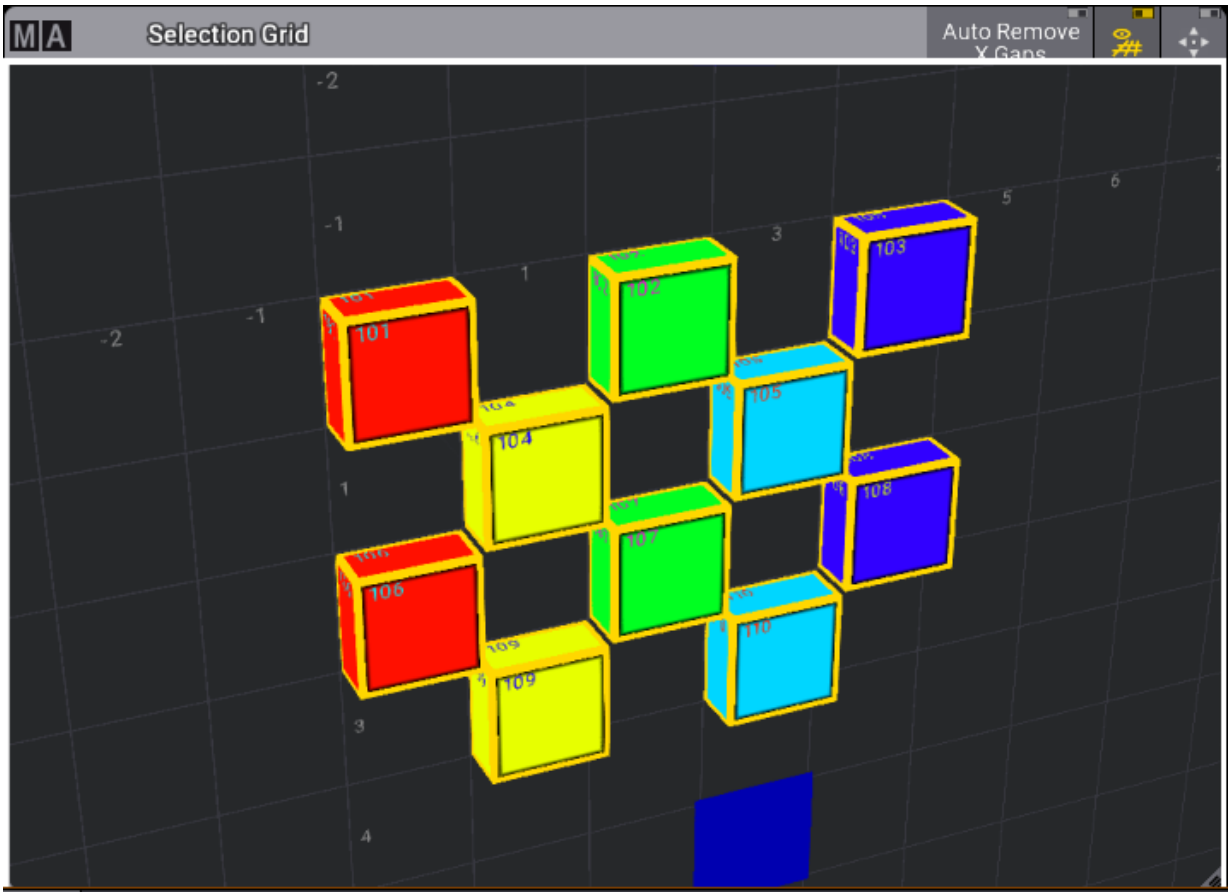
The fixture's position in a grid can be seen in the **Selection Window**.

The title bar has three On/Off buttons:

Auto Remove X Gaps:
When turned on, will remove the gaps in the fixture selection.

Show Grid:
toggles the visibility of the grid.

Fit:
Resets the window to fit all the fixtures in the grid.



Selection grid with 10 fixtures positioned

Grid Cursor

Fixtures are positioned based on the position of the grid cursor. The grid cursor is the blue cells in the grid.

The grid cursor can be moved using the **Grid keyword** (all three axes are possible) or simply by tapping in the selection window (only selects at the 2D XY plane).

For example, moving the cursor to X position 3, Y position 2, and Z position 1, press the following keys:

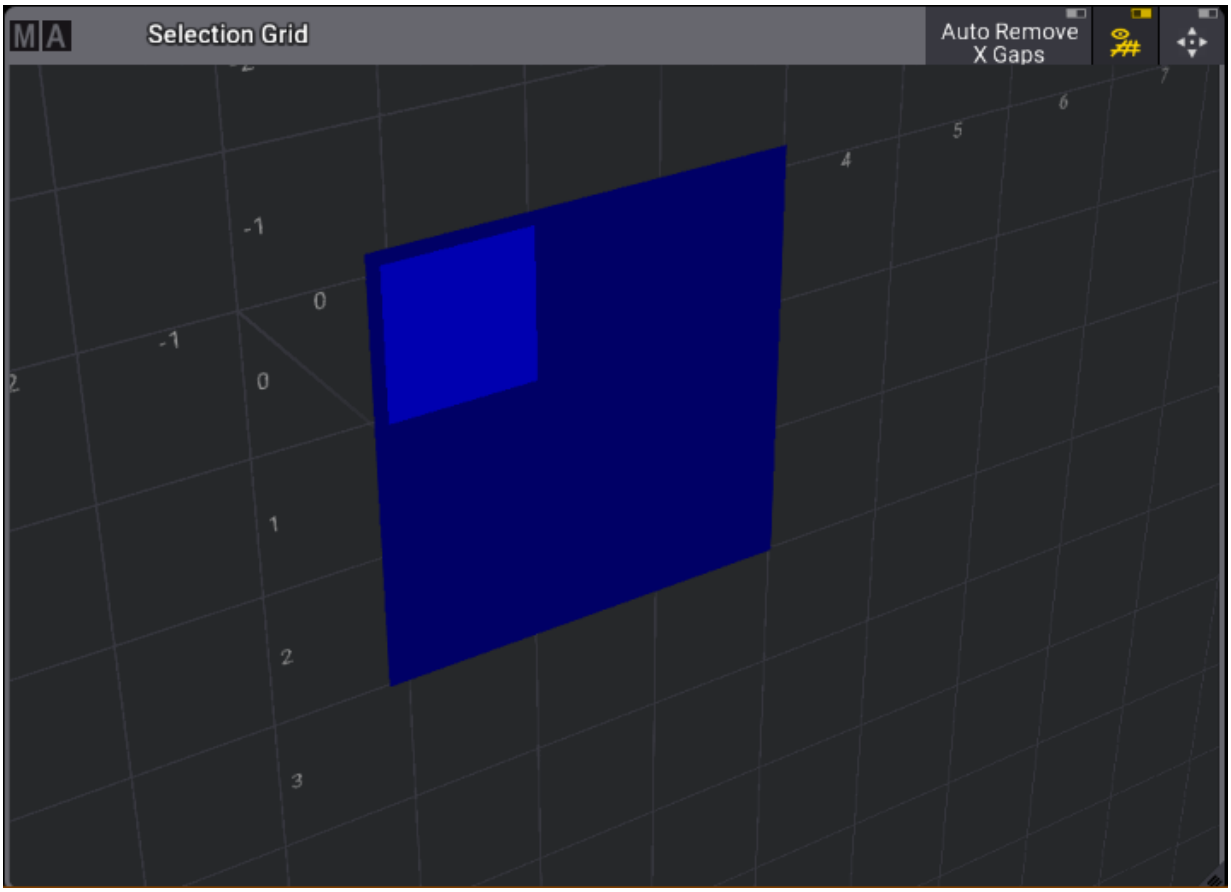
MA + X3 3 / 2 / 1 Please

The Z-axis can be omitted if it is 1.

The grid cursor can be more than one cell. A range of cells can be selected using the **Thru keyword** and specifying the beginning cell and the ending cell.

For example, press the following keys:

MA + X3 1 Thru 3 / 2 Please



Selection grid with a range of cells selected

Selecting fixtures after defining the grid puts the fixtures in the area. If only one cell is selected, then the grid cursor moves to the next free cell in the X-axis and positions horizontally the fixtures from this location. Moving the grid cursor one field "right" (positive X-axis) with every placed fixture.

The grid numbers cannot currently be negative.

Store the Grid

When the fixtures are positioned in a grid, it can be stored in groups, presets, or other objects using the grid information. Use any of the available store methods (buttons, screens, or any combination) to store the group, preset, or another object.

Read more about storing groups in the **Create Groups topic** or more generally about groups in the **Groups topic**.

Read about storing presets in the **Create Presets topic** or, more generally, about presets in the **Presets topic**.

Read more about storing cues in the **Store Cues topic** or generally about **cues and sequences section**.

Adjust the Selection Grid

The grid can be rotated by pressing the window with a single finger and moving the finger around the screen. The same can be achieved by left-clicking a mouse and moving the mouse with the left button pressed.

The grid can be zoomed using a pinch motion with two fingers on the touch screens. A scroll wheel on a mouse also zooms.

The grid can be moved around by touching the screen with two fingers and moving them around the screen. This can be done with a mouse by keeping both the left and right mouse button pressed while moving the mouse.

14.9. Smart View

The **Smart View** is a dynamic pool that gives quick access to fixture defined channel sets.

The channel sets are a part of the fixture types.

It can be created like any other window using the **Add Windows** pop-up. It is found in the **More** tab, where it is called **Smart**.

Smart View showing a gobo wheel

Each pool object represents a channel set defined in the fixture profile.

The channel sets cannot be edited using the smart view.

Tapping one of the objects puts the respective value into the programmer.

The bottom of the smart view shows an attribute selection bar. This makes it easy to select the desired attribute in the currently selected feature group. The example above shows the **Gobo** feature group and the **Gobo Wheel 1** (G1) attribute.

This bar can be enabled or disabled in the **Settings**. This is the only special setting for this window.

15. Scribbles

The grandMA3 has a scribble pool where it is possible to save the scribbles created within the scribble pool itself or when labeling an object.

Important:
Scribbles created during labeling of a pool or any other object are automatically saved in the scribble pool.

Important:
It is not possible to assign a scribble to the appearance pool and vice versa.

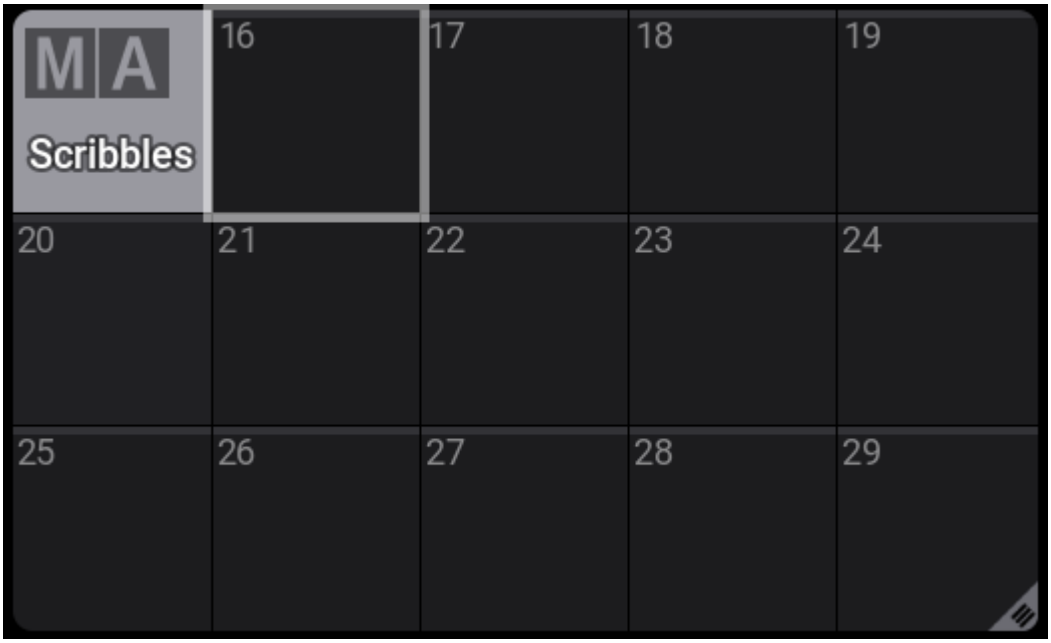
15.1. Create Scribbles

Hint:
It is possible to create scribbles in the scribble pool or when labeling objects.

Create Scribbles in the Scribble Pool

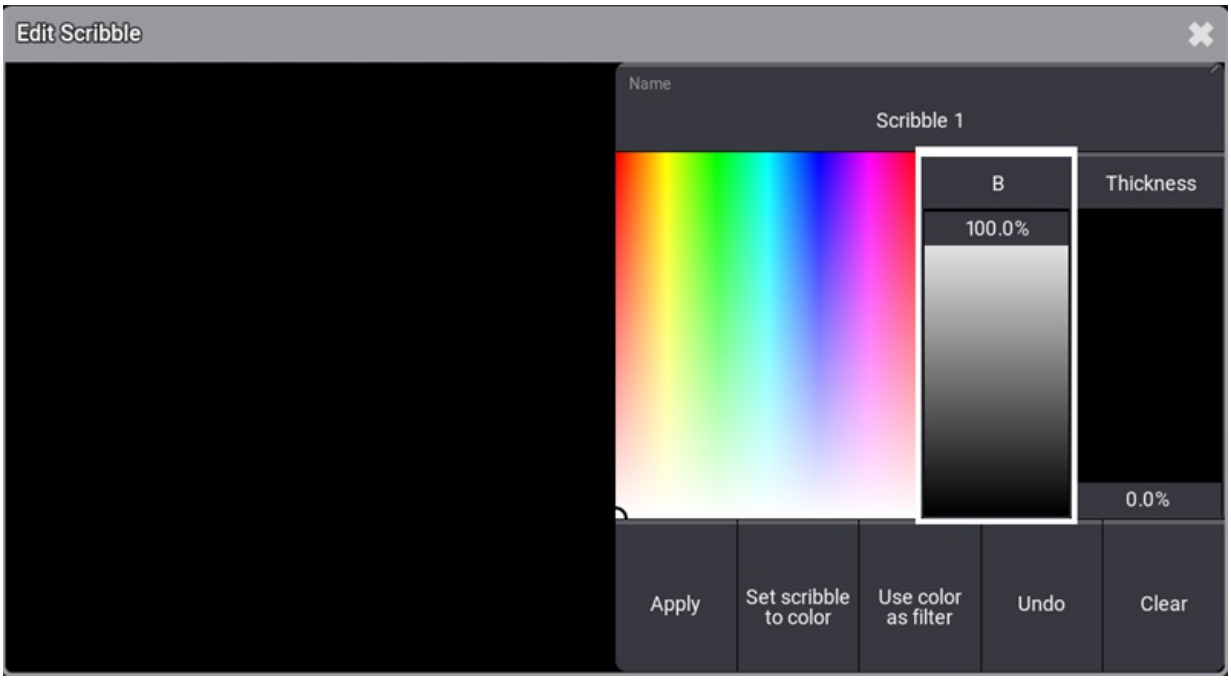
1. Access the scribble pool.
   - Open the **Add Window**.
   - Open Pools.
   - Open Scribbles.

For more information, see **Add Windows**.



Open the scribble pool

2. Store a new scribble.
   - Tap and hold an empty pool object.
   The new scribble is saved.
3. Create a scribble.
   - Press Edit and tap the scribble you saved.
   The **pop-up Edit Scribble** opens.



Pop-up Edit Scribble

4. Label the scribble:
   - Tap Name.
   - The **pop-up Edit Name** opens.
   - Give the scribble a name.
5. Select the color of the scribble.
   - Tap the color picker.
6. Set the brightness of the color.
   - Move the **B** fader.
7. Set the thickness of the scribble.
   - Move the **Thickness** fader.
8. To scribble, move your finger across the scribble pad on the left of the pop-up.
9. To delete the last step made in the scribble, tap Undo.
10. To apply the scribble, tap Apply.

Hint:
The color and the thickness settings of a scribble are saved during the use of a grandMA3 console. Once the console is powered down, these settings are reset.

The scribble is applied to the scribble pool.

Create Scribbles in Other Pool Objects

This example is based on the object of a group pool.

Requirement:

- The group pool is open

1. Save a new pool object in the group pool.
2. Label the new pool object.
   - Press Assign Assign and tap the pool object.
   - The virtual keyboard opens.
3. Tap in the title bar of the keyboard.
   - The virtual keyboard expands and opens the scribble pad and its controls.
4. Follow steps **5 to 8** described in the scribble pool.

The scribble is automatically saved on the next free position in the scribble pool and applied to the group pool object.

Important:
Creating scribbles outside the scribble pool spams the pool because every scribble created with the label function within other pools is saved as a new scribble in the scribble pool.
To learn how to avoid spamming, see **Assign Scribbles**.

15.2. Edit Scribbles

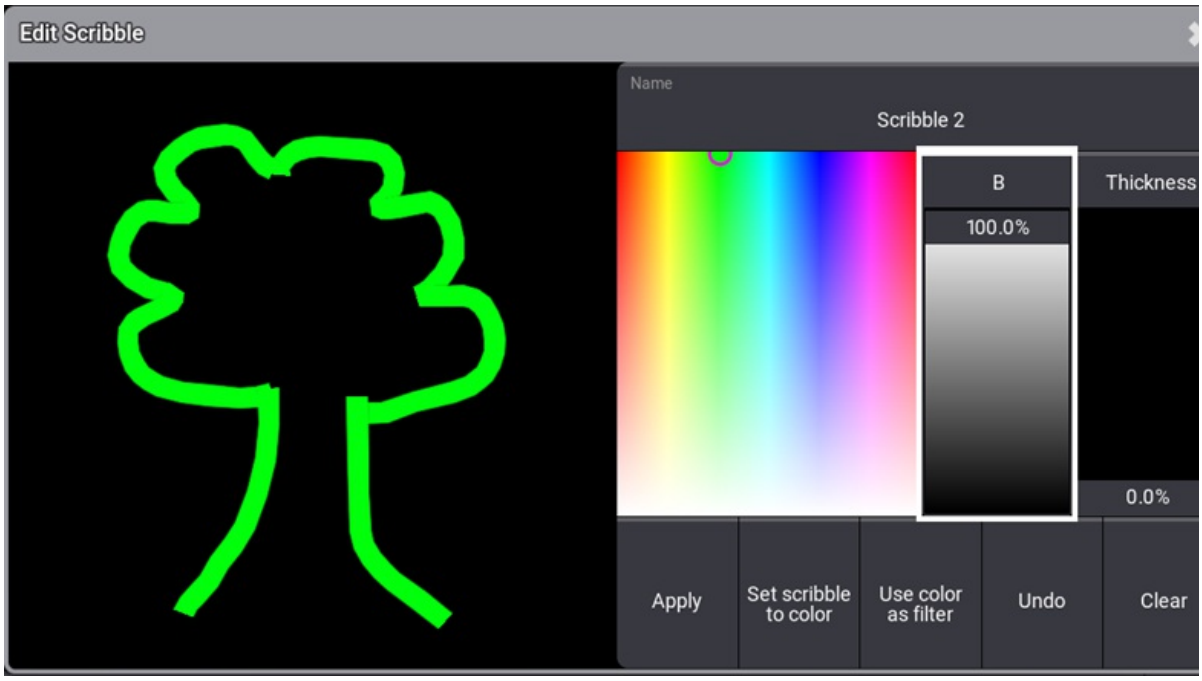
It is possible to edit an existing **Scribble**.

Edit Scribbles in the Scribble Pool

Requirement:

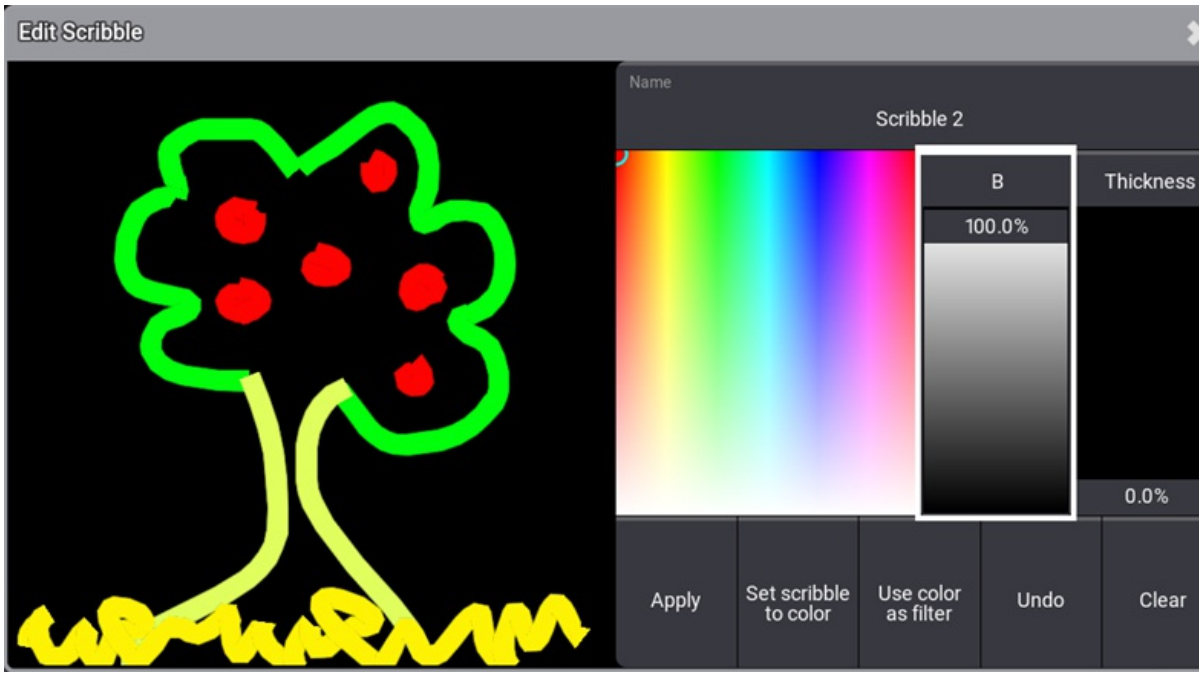
- The scribble is applied in the scribble pool

1. To edit an existing scribble, press Edit and tap it.
   -The window **Edit Scribble** opens.



Edit an existing scribble

2. Adjust the scribble.



Adjustments in the existing scribble

3. Tap Apply.

The edits are saved in the scribble pool.

Edit Scribbles in Other Pools

This example is based on the group pool.

Requirement:

- The scribble is assigned to a group pool object

1. To edit an existing scribble in the group pool, press Assign Assign and tap the pool object.
   -The Window **Edit Name** opens. Tap in the title bar to open the scribble area if needed.
2. Adjust the scribble.
3. Tap Apply.

The edits are automatically saved in the scribble pool.

15.3. Assign Scribbles

You can assign existing scribbles in the scribble pool to other pools objects.

Hint:
To avoid spamming the scribble pool, create the scribbles in the scribble pool, and then assign them to other pools objects.

This example is based on a group pool object.

Requirement:

- A scribble was created in the scribble pool.

To assign a scribble to a group pool object:

1. Press Assign.
2. Tap the scribble in the scribble pool.
3. Tap the pool object you would like to assign the scribble to.

The scribble is assigned to the group pool object.

15.4. Delete Scribbles

Delete Scribbles in the Scribble Pool

It is possible to delete scribbles using one of the options:

1. To delete single scribbles:

- Press Delete and tap the scribble you would like to delete.

-or-

- Use the Swipey.
  For information see **Pool Windows – Swipey**.

> **Important:**
> If you assigned scribbles to other pool objects, deleting single scribbles in the scribble pool deletes the links in all pools.

2. To delete several scribbles, type in the command line:

```
User name[Fixture]> Delete Scribble 1 Thru 5
```

Deletes scribbles 1 to 5.

3. To delete all scribbles of the scribble pool, type in the command line:

```
User name[Fixture]> Delete Scribble 1 Thru
```

16. Images

All imported images are in the **Image Pool**. For more information, see **Pool windows**.

Important:
The overall size of the media pools has a maximum of 200 MB.

We advise that you keep the image pool as small as possible. For example, keep the maximum image size below 64MB. Do not exceed an image resolution higher than 1920 x 1080.

Images can be used for appearances. Read more in the **Create appearances** topic.

Example of the image pool

Edit an image pool object using one of two options:

- Press Edit and then tap the desired pool object.
- Open the swipey commands on the pool object and choose Edit.

Import Predefined Images

Some predefined test pattern images can be used. Custom images can also be imported - read **below**.

Workflow:

Images can be imported using the command line.

1. Navigate to the custom image pool:

```
User name[Fixture]> ChangeDestination Image "custom"
```

2. Now import the predefined library images:

```
User name@ShowData/ImagePools/Custom> Import lib "*.xml"
```

This command imports the image files from MA into the image pool.

3. Return to the command line root:

```
User name@ShowData/ImagePools/Custom> ChangeDestination Root
```

Hint: Use syntax shortcut "CD" for "ChangeDestination"

Import Images Using Image Pool

Workflow:

Images can be imported using the Image pool.

1. Edit an empty pool object.
2. Tap the Import button.
3. Tap Internal in the top-right of the title bar to change the drive to the desired source.
4. Select the desired image.
5. Tap Import.
6. Close the **Edit Image** pop-up.
7. Before closing the Edit Image pop-up, it is possible to edit the name.

By default, the grandMA3 software uses the **gma3_library/media/images** folder for the import and export of images. For more information on the default folder structure, see the **Folder Structure** topic.

Delete an Image from the Pool

Deleting images is like deleting any other pool object. The image disappears in any appearance where it might be used.

There are three common ways to delete images.

Delete an Image Using the Command Line

The important keyword for this is: **Delete**.

This is the syntax for deleting a single image:

Delete Image image_number

It is also possible to delete a range of images using the standard range syntax (Thru, +, and -).

For example, if image 4 needs to be deleted:

```
User name[Fixture]> Delete Image 4
```

Or if image 5 to 10 needs to be deleted:

```
User name[Fixture]> Delete Image 5 Thru 10
```

Delete an Image Using the Image Pool on a Screen

Requirement:

A visible image pool on one of the screens.

1. Press the Delete button.
2. Tap the image in the pool.

The image is deleted.

Delete an Image Using the Swipey Commands

Requirement:

A visible image pool on one of the screens.

1. Tap and hold the image you wish to delete.
2. Swipe out of the pool object without releasing the screen.
3. Swipe to the Delete swipey and release the screen.

The image is deleted.

16.1. Screenshots

Screenshots can be created by pressing the Print Screen key or shortcut F11 on a build-in or external keyboard. A green flash on all screens indicates that the screenshot is taken.

An image from each screen is stored as individual png files in the image folder. The file names are generated using this template: YYYYMMDD_hhmmss_displayX.png

The images can be imported into the **image pool** or copied out of a console using an SFTP connection. For more information, see **Networking — SFTP connection**.

Restriction:
There is no keyword or command to create screenshots.

Storing Screenshots on a USB Stick

Screenshots can be stored on a USB memory stick. They are stored as png files.

The USB needs to be the selected drive when Print Screen is pressed.

A list of the drives can be seen in the **Command Line History window** by running the following command:

It is drive number two if there are no other USB devices connected.

Select the drive using this syntax: **Select Drive [drive_number]**

Now press Print Screen.

Bring the USB to a computer. Screenshots are saved to the **/gma3_library/media/images** folder. On USB drives, the **gma3_library** folder is located in the **grandMA3** folder.

Copy Screenshots using SFTP

When accessing the internal drive of the console using SFTP, the **gma3_library** folder is located at the root of the visible folder structure.

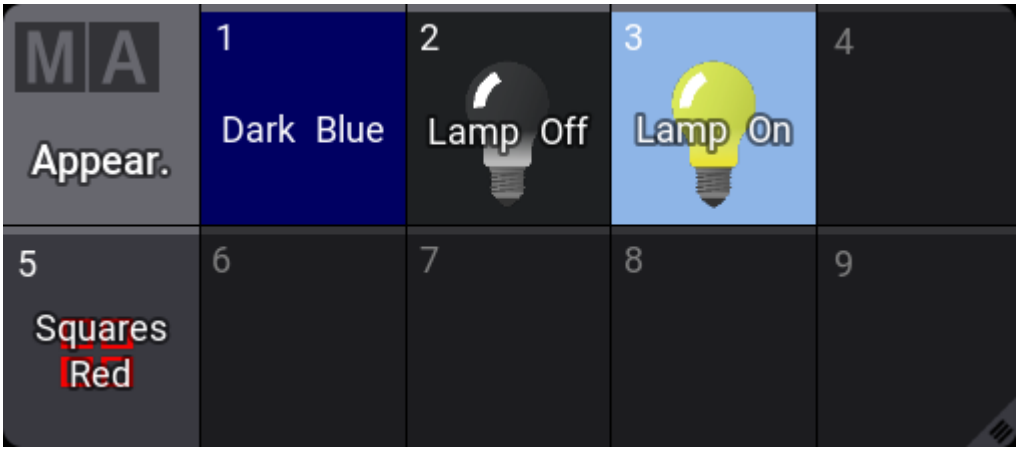
Open an **SFTP connection**, navigate to the images folder, and copy the relevant files from the console.

For more information on folder structure, see the **Folder Structure** topic.

17. Appearances

Appearances are sets of looks that can be assigned to pool objects, presets, view buttons, or windows.

All appearances are stored in the **Appearance Pool**. This can be created like any other window - learn how in the **Add Windows topic**.



Example of the Appearance pool

Images can be a big part of an appearance. To use images, they need to be imported into the **Image Pool**. Read about the images in the **Images topic**.

17.1. Create appearances

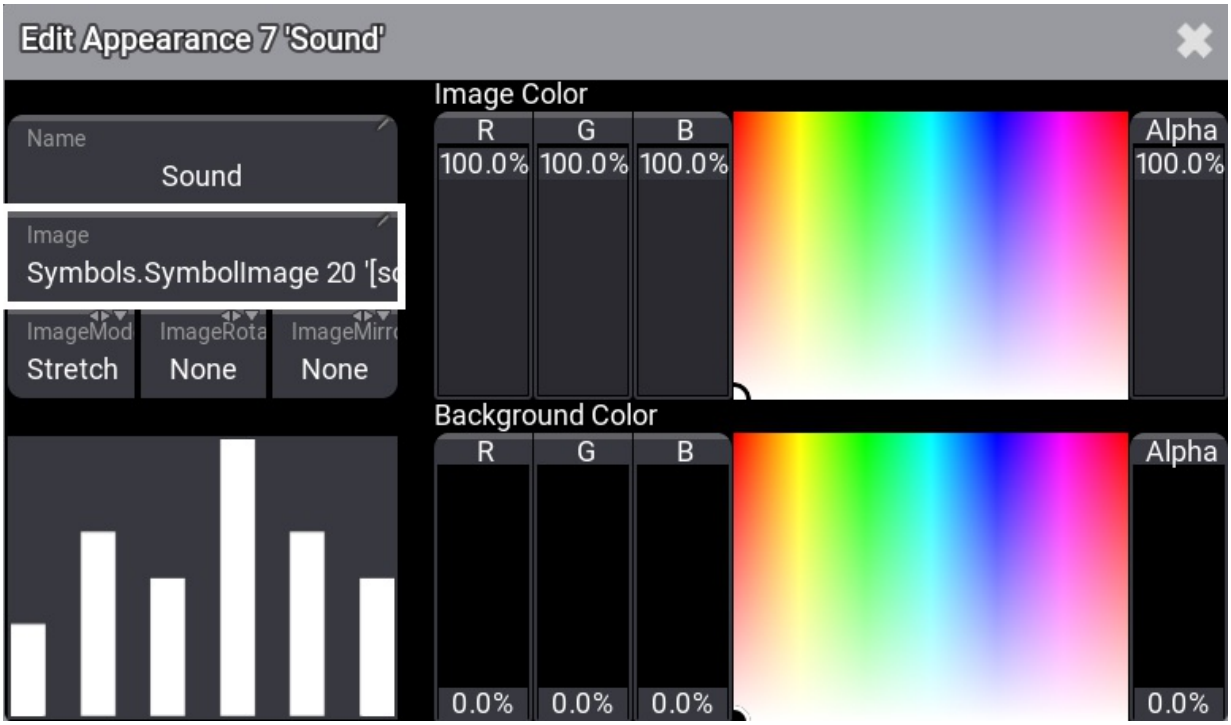
Edit an empty pool object in the **Appearance Pool** to start to create a new appearance.

Edit an appearance pool object using one of these 3 options:

- Press Edit and then tap the desired appearance object.
- Open the swipey commands on the pool object and choose Edit.
- Using the command line: **Edit Appearance [ID]**.

If an existing pool object is edited, then the editor opens with the current state of the appearance.

This is the **Appearance Editor**:



There are five input boxes, two sets of faders with a color picker to adjust color, and finally a preview area that displays a preview of the appearance.

Setting a Background Color

The **Background Color** fader and color picker can be used to change the background of the appearance. The default background is transparent (**Alpha** fader at 0%).

Turning up the alpha and using the R (red), G (green), and B (blue) faders make it possible to mix any color in the RGB range. The color picker can also be tapped to select a color.

Giving the Appearance a Name

Many editors and pop-ups have a **Name** input field. A name can be set using the **Label keyword**.

Appearances can be labeled like any **other pool object**.

It is also possible to give the appearance a name in the editor:

1. Tap the name input box.
2. Write a new name using the Edit Name pop-up.

Restriction:
Appearances cannot have a Scribble in the label.

Adding an Image to the Appearance

Appearances can use an image. The image is placed in front of the background but behind a label.

1. Tap the **Image** input box to add an image. This opens the **Select Image pop-up** where all the images are listed—Tap ImageSource in the title bar, to select a new source.
2. Choose the desired image source.
3. Tap the desired image in the pop-up.

The images have different modes. Their modes define how the image is adapted to the aspect ratio.

These are the different modes:

- **Stretch**
  The image is stretched to fit the appearance area.
- **Bar**
  The image is fitted to the appearance area without changing the aspect of the image.
- **Crop**
  The image is fitted to fill the entire appearance area. The aspect of the image is kept, but it is chopped.
- **Tile**
  The image is tiled and chopped to fit the entire appearance area. The aspect of the image is kept.

Do the following to change the mode:

1. Tap the **Mode** swipe button. This toggles through different modes. Or swipe the button to open the **Select Image Mode pop-up**.
2. Tap the desired mode in the pop-up.

The color of the image can be adjusted using the top set of fader and color picker.

This means that the same image can be used on different appearances but with different color hues and transparency.

Restriction:
Color can only be added to an image. This means that a black image cannot be changed, but a white image can be changed.

17.2. Use Appearances

Appearances can be used in many places.

Almost all pool objects can have an appearance. Many windows can also have an appearance.

The appearance needs to be **created** before it can be assigned. And the object that it is assigned to needs to exist before anything can be assigned to it.

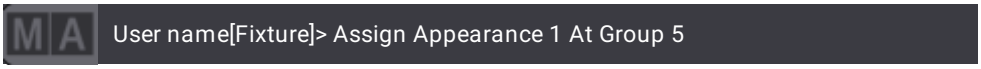
Using Appearances on Objects

Appearances can be assigned to objects using the following syntax:

Assign Appearance appear_number At object_type object_number

This can be used on pool objects.

For example, assigning appearance 1 at group 5 could be done like this in the command line:



This can also be done using the GUI.

The Appearance pool and the Groups pools should be visible on the screens for this.

1. Tap, hold, and slide out the finger on appearance 1.
2. From the swipey commands, move the finger to the Assign button and release.
3. Tap group number 5.

Editing many objects also gives access to an appearance button that can be tapped to select one of the existing appearances.

Using Appearance on Windows

Many windows can have an appearance assigned.

The best way is to open the **settings for the window** and edit the **Appearance** setting in the **Display** tab.

17.3. Delete Appearances

Deleting appearances can be done like **any other pool object**.

If the appearance is used somewhere, then a warning pop-up opens; press Ok to delete.

The object that used the appearance is reset to **No Appearance**.

Oopsing the deletion will reassign the appearance.

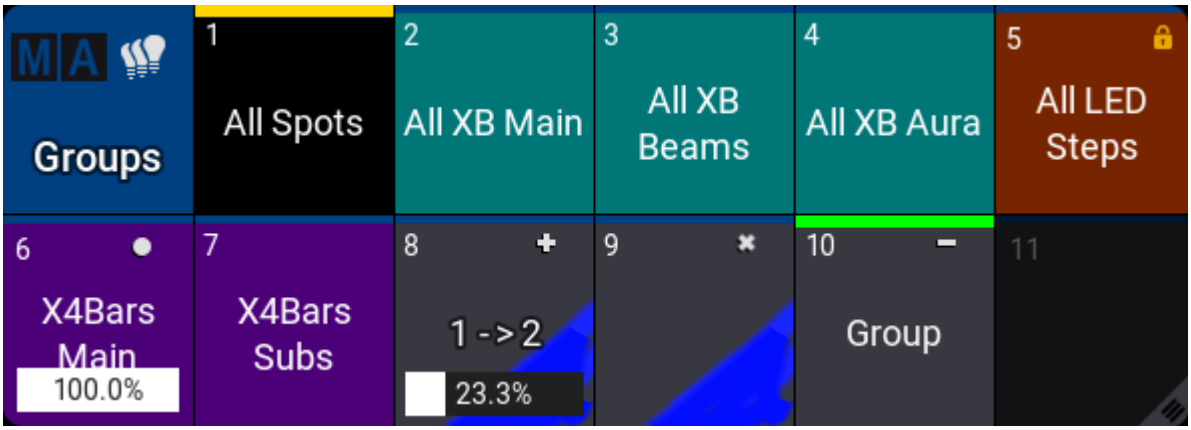
18. Groups

Groups contain a selection of fixtures. This includes the order and grid position of the selection.

This can be used as a programming tool, where it is a fast way to select the fixtures. Grid information is stored in the groups - read more about grids in the **Selection Grid Window topic**.

Groups can also be used as masters. There are four different types of group masters. Read more in the **Group Master topic**.

Groups are organized in a pool - read more about pools in the **Pool Windows topic**.



Groups pool example

Information about the groups is not stored in cues or presets. Only the values applied to the fixtures are stored in cues and presets.

The next topics explore the creation of groups.

18.1. Create Groups

Groups are created by storing the group while having a selection of fixtures. Groups are stored in the **Group Pool**.

The groups store the selection of fixtures, the grid information, and the fixtures' selection order (this is also a grid).

Grid information is a 3D position information that can indicate the position relationship between the fixtures. It is not the location on the 3D stage. For more information, see **Selection Grid topic**.

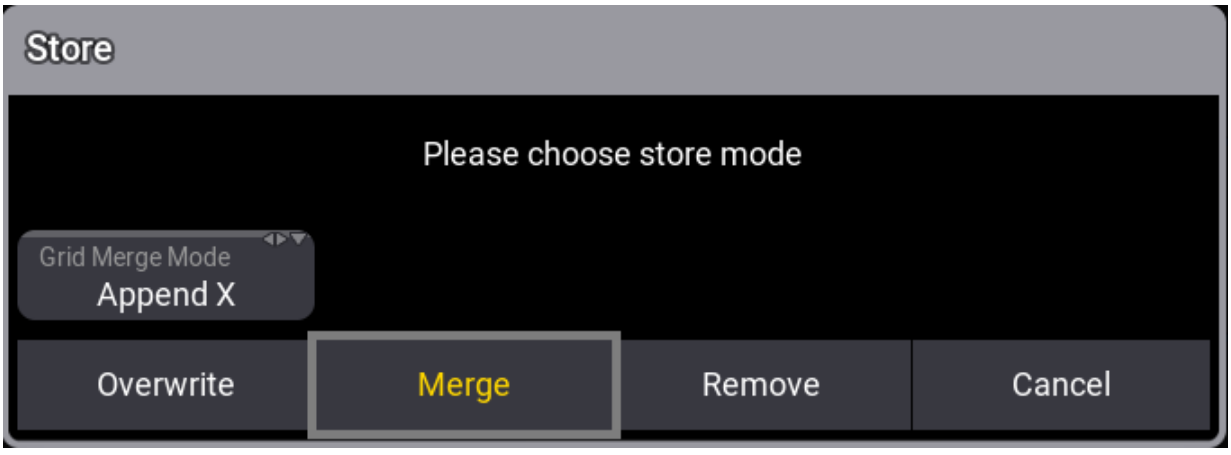
The order and grid information is important for ranged value input or when creating **Phasers**.

Important:
Groups do not store values! - Only the fixture selection, order, and grid.

These are the steps needed to create a new group:

1. Please make sure the correct fixtures are selected and they are in the correct order.
2. Store the group using a syntax like this: **Store Group [my_group_number]**. The two needed keywords are **Store** and **Group**.

When storing to an existing group the store mode pop-up opens:



When Grid Merge Mode is set to **Append X** (this is the default setting), the selected fixtures will be added to the next available X coordinate.

Read the **Selection Grid topic** for information about positioning the fixtures in a grid before storing the group.

Existing groups can be edited using the **Edit Group pop-up**.

It is not required, but it is a good idea to label the group. It can be done using any of the methods described in **Label pool objects**, or the Edit Group pop-up.

18.2. Edit Groups

Groups can be edited using the **Edit Group pop-up**.

It opens when a group is edited using the **EditSetting keyword**. For example, by pressing Edit twice and then tap the desired group in the group pool.

The editor has 2 main columns, **Label**, and **Setting**.

There is **Master** fader on the right. This is used when the **Mode** is one of the four group master types. Read more in the **Group Masters** topic.

Here is a description of the left buttons under the **Label** column:

Name:
This is where the name of the group can be edited.

Hint:
A name is changed using the **Label** command. The **Label keyword** is used to change the name of an object using the command line.

Scribble and **Appearance** can be assigned using these two buttons. For more information, see **Scribble** and **Appearances**.

Here is a description of the right buttons under the **Settings** column:

Lock:
When toggled on, Name, Scribble, Appearance, Mode, and Move Grid Cursor are disabled.

Mode:
This button is used to select one of the four group master mode functions (Read more in the Group Masters topic - link above).

Move Grid Cursor:
When turned off, the cursor will stay at the position it was when the group was called in the programmer.
When turned on, the cursor will move to the next X grid position.

Tap List References to open an **Info pop-up** that shows a list of the group's references and dependencies.

Tap Edit to go into the **Edit** mode (Fixture sheet frame turns green). Now it is possible to change the fixture selection. Press the Off key, then select the fixtures you want to remove from the group. From the swipey commands, move your finger to the store command. A pop-up menu will ask you to Overite, Merge, Remove, or cancel. Use Esc to exit the **Edit** mode without changes.

18.3. Delete Groups

Groups can be deleted like **any other pool object**.

It does not have any effect on programmed cues or any other element. The only exception is if the group is assigned to an executor directly and used as the base for a **Group Master**, then the executor will be empty after the deletion of the source group. Read more in the **Group Master topic**.

18.4. Group Masters

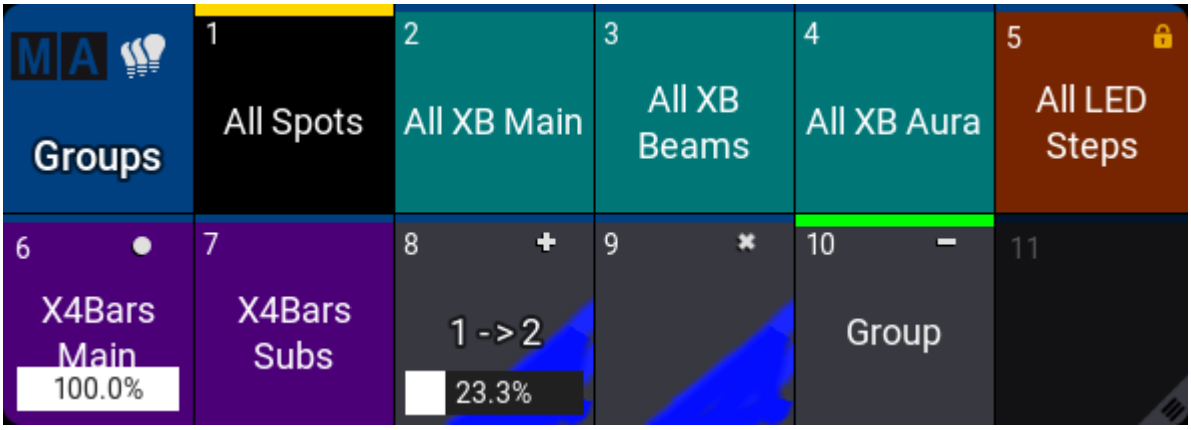
Groups can be masters for the fixtures in the group.

If the value is adjusted often, it can be advantageous to assign the group to an executor. Learn how in the **Assign Object to an Executor topic**.

There are four different kinds of group masters:

- **Positive** - This is an HTP group master. It can be used to limit intensity output. It is indicated by a small plus icon in the upper right corner of the pool object.
- **Negative** - This is a LoTP group master. It can also be used to limit intensity output. It is indicated by a small minus icon in the upper right corner of the pool object. Read a description of the difference between positive and negative masters below.
- **Scaling** - This scales the intensity output. For example, if the fixture is at 50% and the group master is at 50%, then the output is 25%. It is indicated by a small x icon in the upper right corner of the pool object.
- **Additive** - This master does not limit output but adds output as HTP merge with values from programmer and playbacks. It is indicated by a small circle icon in the upper right corner of the pool object.

The difference between the positive and negative masters are relevant when several groups contain the same or overlapping fixtures. A negative master has priority over a positive. For example, if two groups contain a fixture with 100% output and if one group is a positive master on 80% and the other group is a negative master on 60%, then the output is 60% (limited by the negative master). If the negative master is turned up, the output stops at 80% when the positive master becomes valid.



Group pool with masters

Groups with a mode assigned have a horizontal bar that displays the master level if it is relevant.

The master mode can be selected using the **Edit Group pop-up**. Read more about the pop-up in the **Edit Groups topic**.

The mode can also be assigned using the **Set keyword**.

Example

Using the command line to set the master mode of group 4 to positive:

The mode type is capital sensitive. This means that writing "positive" fails, but "Positive" works.

Hint:
If the mode is not specified, then a pop-up appears, allowing you to choose a mode from a list.

Delete a Group Master

Deleting a group assign to an executor does not delete the group from the pool.

If a group is deleted in the group pool, then it is deleted from the show.

Restriction:
If the group is limited by the master, when it is deleted, then the fixtures stay limited. This means that a new group needs to be created with the fixtures to remove the limitation!
If the group is removed from an executor, then it still exists in the group pool, and any master level set using the executor is still valid in the pool.

19. Presets

A preset can hold information about attribute and timing values for a selection of fixtures and may be referenced and re-used in cues or in the programmer.

The principle of presets is to store a labeled reference in a cue, rather than the actual value itself. Or use presets for busking, where presets are called into the programmer or played back on the fly.

Updating the preset means that the cues do not need to be updated since they reference the preset and not the actual preset content.

Presets are marked with a cyan marker. This maker is visible on cues that use preset and on the values themselves, for instance, in the tracking sheet or the fixture sheet. Read more in the **Marker topic**.

Hint:
Using presets when programming facilitates the work, especially when working with a show which is used in different locations and for various customers.

Preset Pools

Presets are stored in pools. There are pools for each feature group. These pools have a default input filter that only allows the values for that feature group to be stored in the preset pool. There are also five preset pools called "All 1" to "All 5". These do not have a default input filter and can be used to store any attribute values. Learn more about the pools in the **Preset Pool topic**.

Preset Modes

Presets modes are used when the preset is stored (or updated) and when the preset is called.

There are three different preset modes Selective, Global, and Universal.

Each preset got letters showing the preset mode.

Example:

Preset 4 has all three modes stored but only show U and S.

The three modes are:

- **Selective (S)**:
  The data will be added as selective data for each fixture that has active data in the programmer.
- **Global (G)**:
  The data will be added mainly as global data. If there are several fixtures of the same fixture type but with different values, then the global data will be determined by average, and selective data will be added for the other fixtures which have divergent data.
- **Universal (U)**:
  Data will be added as global data, and the preset mode will be set to Universal.

When a preset is stored or updated, then there are two more options:

- **Auto**:
  When updating or storing into an existing preset, the preset mode of the preset will be respected.
  In the case of global preset mode, selective data will be added to the preset when at least one fixture that can use the preset is active with new values.
  When creating a new preset Auto mode will take the mode defined by the pool and use the rules for each mode.
- **Force Global**:
  Data will be added as global data, and untouched existing selective data will be discarded within the preset for the fixtures of the same fixture type.
  Force Global will discard the selective data when updating a preset or when storing with the merge option into an existing preset.

Each preset can store all three modes or some combinations of modes.

Each preset pool has a setting for a default mode. This mode is indicated by one of the letters in the upper right corner of the pool title field.

Position preset pool title field with Selective as default mode for the pool

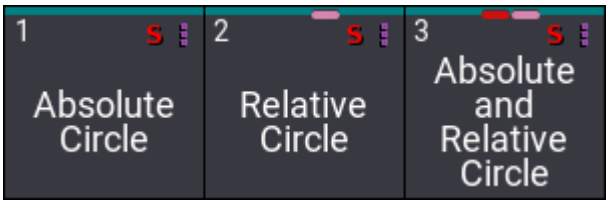
This default can be used when storing the preset or a different mode can be used.

Learn more in the **Create New Presets topic** and the **Use Preset topic**.

Absolute and Relative Values

Preset can store absolute and/or relative values. Relative values are often used with multistep presets but it is not limited to this use.

Example:



Preset with absolute, relative, and both types of values

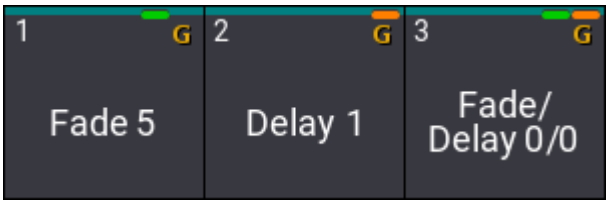
Presets with only absolute values do not have a marker. The absolute marker is a square dark red marker with rounded corners. Relative values are indicated by a dark pink square with rounded corners. Presets can contain both absolute and relative values - they will have both markers in the pool object.

Learn more about the different programmer layers in the **What is the Programmer topic**. Learn more about the markers in the **Markers topic**.

Timing Values

Timing values can be stored in presets. The timing layers are Fade and Delay.

Example:



Preset with fade, delay, and both types of values

The Fade marker is green. The Delay marker is orange.

Timing values can be combined with absolute and relative values.

Embedded Presets

A preset can contain a link to a different preset. It is the same as cues storing a reference to a preset - just for other presets. This can be useful for instance when a show is built using a set of general presets and other presets are created using these as building blocks.

Example:

In this example, there is a universal dimmer preset that has values for the universal fixture. This is used to create a new global preset for fixtures types. This is again used to create a selective preset for only some fixtures. If the fixture type change, then the global preset needs to be updated. The selective preset links to the global preset values and do not need to be updated. The universal fixture is unaffected by the fixture change.

Preset 1 is used in an embedded preset. Preset 2 contains embedded data and is also used in a different embedded preset. Preset 3 contains embedded data.

The "source" preset has a downward pointing arrow that points to a line. This indicates that this preset is referenced by other presets (or cues). The preset with embedded data has an icon with an arrow and a line above to indicate that this preset uses referenced data. Preset with embedded data that is also referenced has the arrow and a line above and below the arrow.

If a new name is not defined then the name is also referenced (referenced names are in square brackets). This means that changing the name of the source preset also updates the name of the new preset. Learn more in the **Create New Presets** topic.

Recipe Presets

Presets can have recipe information. A recipe is one or more lines with information about a group, preset, MAtricks, individual fade, delay, speed, and phase values.

This information can be used to "cook" values into the preset or programmer. If the source information changes then the preset can be cooked again to reflect the changes. For instance, if a group is used in a recipe and it changes after the initial cooking, then the preset can be cooked again and it will now reflect the changes to the group. Learn more in the **Recipe Presets** topic.

Example:

Preset with recipe information

A small pot icon indicates a preset with recipes.

MAgic Presets

MAgic presets are presets where value points in a range are defined. This range can then be applied to a dynamic selection of fixtures. For instance, two fixtures are stored in a MAgic preset with dimmer values of 0% and 100%. MAgic presets should be stored as a selective preset, but that does not mean that only the selected fixtures can use it - it only means that they contain values for a selection of fixtures. The MAgic preset uses the values to define the points in a range. A different fixture selection can be made and the range of values between the points are assigned to the fixtures when the MAgic preset is called into the programmer. These values are calculated based on the points and then taken into the programmer as hard values. There is no reference back to the MAgic preset.

There can be up to five defined points in the range (on each axis in the selection grid).

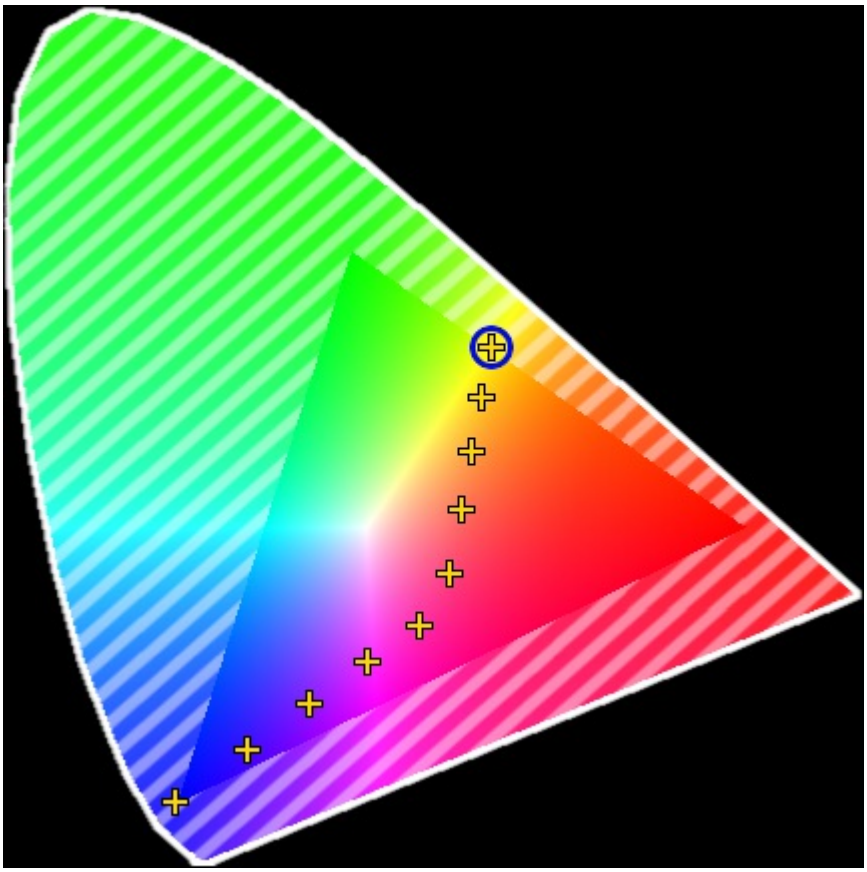
MAgic presets have a small wizard hat icon in the preset pool object

Position preset with MAgic information

Example:

A path needs to be defined to control a color range from blue to yellow. The wish is to have the range go through the red area in a CIE color picker - instead of going through white. Three fixtures are needed to define the range.

The first fixture is blue, the second is magenta, and the third is yellow. This is then stored as a MAgic preset. Now this range through three points can be used by multiple fixtures to create the desired path.



Result of a three-point MAgic color preset applied to multiple fixtures

Learn the details on how to create this MAgic preset in the **Create New Presets** topic.

Multistep Presets

Presets can contain values in multiple steps (Phasers). Presets with only one step usually have the values stored in step one. If a preset has multiple steps then they are called multistep or Phaser presets.

Example:

Multistep preset

They have a three-dot icon to indicate multistep values.

MAtricks Presets

MAtricks information can be stored in presets. Having MAtricks information in the programmer and storing a preset will add the MAtricks information into the preset (if the **Store Settings** allow it).

Example:

Preset with MAtricks information

The MAtricks icon is a green icon with 9 dots in a grid.

Filtered Presets

The entire preset pool has a default input filter - except the All preset pools. The default filter is on feature group preset pools. They automatically filter the attributes in the feature group.

The input filter can be changed to a custom filter for any preset pool including the All presets. If the input filter is different than the default, then there is an input filter icon on the preset pool title field.

Besides an entire preset pool having an input filter it is also possible to have an input filter on individual presets. Learn more about input filters in the **Edit Presets** topic.

Example:

Preset with input filter

The filter icon is a gray filter icon with a small right pointed arrow.

19.1. Preset Pools

Important:
If pools are new to you, then please read the **Pool Windows topics** first.

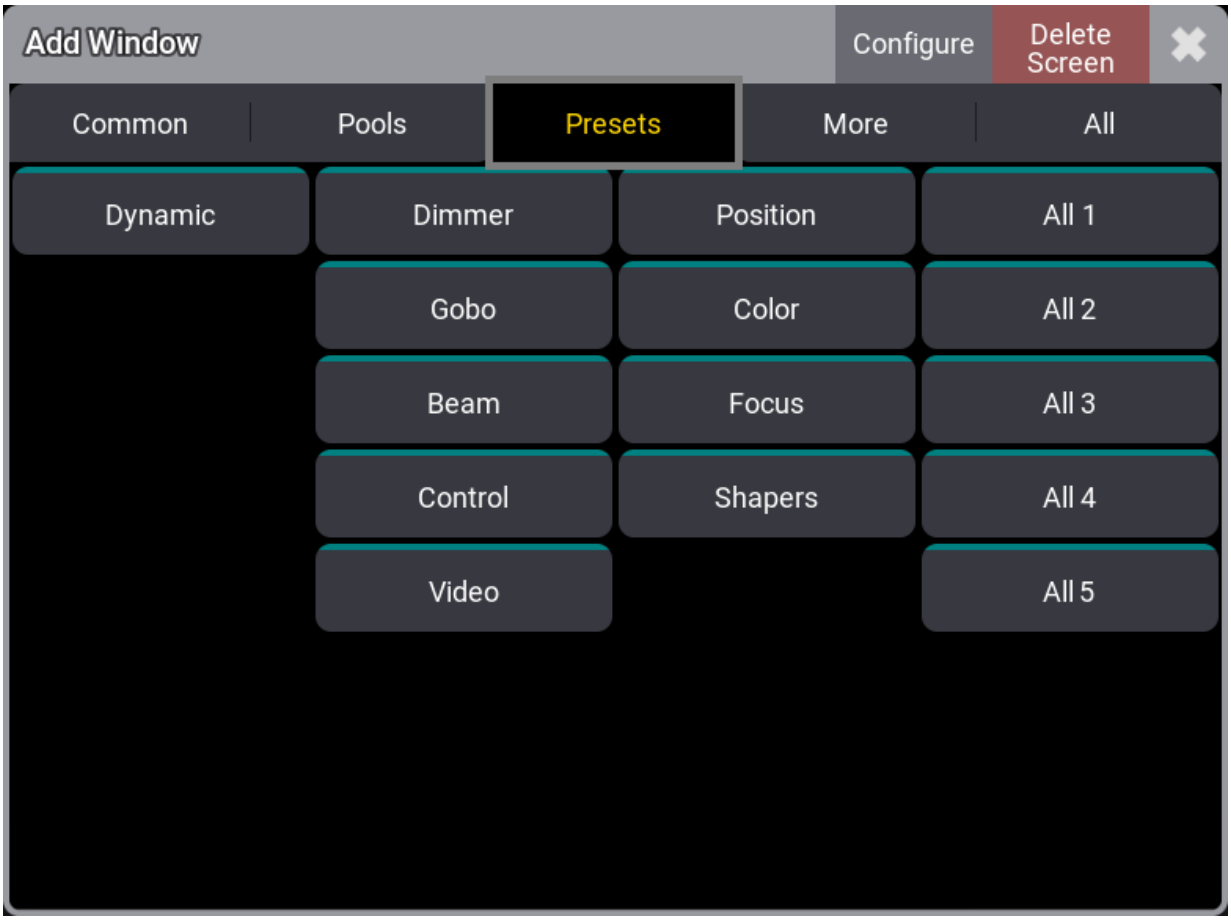
There is a preset pool for each of the **Feature Groups** in the show. A show with no custom feature groups will have nine preset pools named after the default feature groups. For instance, Dimmer. These pools have the option to have a feature group filter besides the standard input filter (read more below).

Hint:
Create your own preset pools using Feature Groups. For more information see **Feature Group**.

Besides the feature group preset pools, there are five "All" preset pools. These do not filter based on feature groups. The "All" preset pools can be labeled to match the needs. For instance one of the pools could be used for storing multistep presets or different parts of the show.

There is a special preset pool called "Dynamic". This is not a preset pool in itself. It automatically changes between the feature group preset pools based on the selected feature group in the **Feature Group Control Bar**.

The preset pools are created like any other windows, in the user-defined areas, using the **Add window pop-up**. They are all under the **Presets** tab:



Open the Presets

The preset pools look and behave like many other pools but there are small differences. Read more about pools in general in the **Pool Windows topics**.

The default preset mode for the preset pool is indicated by a letter in the upper right corner of the pool title field. Read more about the preset modes in the **Preset topic** and below for information about setting the default.

There is a colored indicator bar at the top of each pool object. If nothing is selected then it is colored like the pool color. If there is a selection but none of the currently selected fixtures can use a preset then the pool object is dimmed. If a preset can be used by all the fixtures currently selected, then the colored bar is green. If the preset is only valid for some of the selected fixtures then it is yellow. These are the default colors. They can be edited in the pool settings (see below).

Presets in the All 2 pool

The light blue number below the pool number is the number of fixtures currently using the preset. See preset 2 in the example image above.

Preset Pool Settings

Many of the preset pool settings are described in the general **Window Settings topic**.

There are however some settings that are special for the preset pools.

Enter the settings by tapping the MA logo in the pool title field.

This is an example of the preset pool settings.

Settings for the All1 preset pool window

These are the special settings for presets:

- **Name**:
  The five All preset pools can be named. The other pools are named from the feature group.
- **Input Filter**:
  This can be used to select an input filter for the entire pool. **Worlds and Filters** can be used. An input filter blocks some elements. The blocked elements cannot be stored in the pool. Tap this to open an **Assignment Editor**. Here filters or worlds can be selected or it can be set to Empty.
- **Preset Mode**:
  This sets the default preset mode when storing, for the entire pool. A different mode can always be specified when storing. Learn more about preset modes in the **Presets topic**.

- **Cue Part**:
  By default, presets call their values into programmer part 0. Use this setting to specify a different programmer part for presets for this pool. This setting can also be changed for individual presets in the pool. Learn more about programmer parts in the **What is the Programmer topic**.

The settings described above (except the preset mode) can also be set for individual presets. The individual preset settings have a higher priority than the preset pool setting.

19.2. Create New Presets

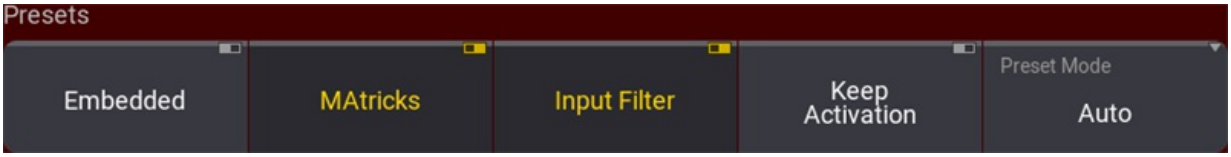
Please read the general **Preset** topic and the **Preset Pool** topic before this topic to get an understanding of the presets and how they are organized.

Creating and using presets requires some fixtures patched in a show file or the universal fixture in the patch for general universal presets.

For more information on how to patch fixtures see **Add Fixtures to the Show**.

When presets are stored then there are some defaults that are used unless other choices are actively made.

Some of these choices are accessible in the **Store Options** pop-up. This is opened by keeping Store pressed for approximately 2 seconds. The pop-up has a section regarding the presets:



The preset section in Store Options pop-up

Hint:
Changing a setting here without saving this as the new preference, makes the change valid for the next store action only. Learn more about it in the **Store Options and Preferences** topic.

The preset mode is one of the options. The preset pool has a default setting but a different mode can be chosen by tapping Preset Mode until it has the desired mode. Learn about the presets modes in the **Presets** topic and the **Preset Pools** topic.

Keep Activation means that when the preset is stored then the newly created preset is active in the programmer, ready to be stored in, for instance, a cue. Turning this setting Off stores the preset, but leaves the values as inactive values in the programmer.

MAtricks allows storing MAtricks settings in the preset.

Read about **Embedded** and **Input Filter** below.

Input Filter

The **Input Filter** setting in the store options defines if the input filtering in preset pools are used. If this option is active, then it is only possible to store attribute values inside the preset pools if it is allowed by the filter. If the input filter setting is empty, then there is a Feature Group filter applied. This means that only attributes of the corresponding feature group can be stored in the preset pool. For instance, pan attributes can be stored in the Position preset pool, but not in the Color preset pool.

The All preset pools do not have any feature group filtering. This means that any attribute can be stored. Regardless of the Input Filter setting.

Custom filters and worlds can be assigned to the preset pool or individual presets using the preset settings or individual preset options. Filters and worlds can also be assigned using the following syntax: **Assign Filter [filter_ID] At Preset [preset_ID]**. Substitute filter with world if a world needs to be assigned as an input filter.

Learn more about filters and worlds in the **Worlds and Filters** section. Learn about the preset pool settings in the **Preset Pools** topic.

Simple Static Preset

Storing a preset with simple static values is very easy:

1. Select some fixtures.
2. Give some of the fixture attributes a value in the programmer.
3. Store this to the relevant empty pool object.

This is the general workflow.

The programmer values are stored in the preset if they are allowed to pass the input filter (read above).

The best way to work with presets is to have the desired preset pool available on a touch screen. Then the pool can simply be tapped to store and call the presets.

It is considered a static preset if it only contains values in **step** one. This means that the values are not changing. Read the **Phaser** topic to learn what steps are.

Global Preset Data

The global data is stored on real fixtures within the patch.

When storing new global presets the global data will be created within the new preset automatically. When all data for the fixture type is the same, the first fixture from the stored data will be used to hold the global data within the preset.

Each fixture type gets its own global data value so the global value can be adjusted per fixture type.

When adding selective data to the preset for one of the fixtures that hold the global data, or when deleting the fixture from the show, the global data will be moved to the first patched fixture for the fixture type. In the case that the global data was held before by the first fixture, the data will be moved to the next patched fixture of this fixture type.

When editing a preset with global data, the fixtures with the global data in the programmer will display a yellow square marker in the top right corner of the attribute cell within the fixture sheet.

Important:
When converting old show files from grandMA3 v1.5 or prior to grandMA3 v1.6 or later the data from the global fixture type object will be migrated to the first patched fixture of the same fixture type that is not already holding selective data.

The global value is determined by the average across all values of attributes within the same activation group:
(Value of fixture 1 + Value of fixture 2 + Value of fixture 3 + ... value of fixture n) / Number of used fixtures = Average value
The value that is closest to the calculated average value will be the global value. The fixture with this value will then hold the global value. In the case of color, all color attributes are handled together when choosing the global value.

Furthermore, the method to determine the global data across several attributes is now taking all attributes of the same activation group into account. This will result in taking all attributes of the same activation group of one fixture, and not taking the different attributes from different fixtures.

Universal Preset Data

Storing a universal preset using a real patched fixture will store global data for the fixture types active when the preset was stored.

Universal presets are more versatile and can use the data from a real fixture with global values when the **Universal Fixture** has no data inside the preset.

This has the added benefit that it is possible to create universal presets using a real fixture from the stage, in the case that the universal fixture has these attributes.

When calling a universal preset the software uses global fixture type data first and then uses the data from the universal fixture. When no data exist for the universal fixture the first fixture with global data is used instead.

Hint:
When calling universal color it will be transformed through the color engine. If for instance, a 7-color LED is used to create a color and call that data universal, a similar color will be called on an RGBA fixture.

Embedded Preset

The concept of embedded preset is explained in the **Presets** topic.

This is the workflow:

1. Select the desired fixtures.
2. Recall an existing preset so the preset is in the programmer.
3. Store a new preset with the Embedded store option active.

A recipe preset could be an alternative to an embedded preset. Learn more about them in the **Recipe Presets** topic.

Store Selective Preset

1. Select one or more fixtures.
2. Give the fixture attributes some values in the programmer.
3. Press and hold Store until the **Store Settings** open.
4. Tap Preset Mode until the mode is **Selective**.
5. Tap the desired preset pool object - make sure it is a valid pool object for the attribute values.

Store Global Preset

Attribute values need to be tagged with a global flag to be stored as global data.

1. Select one fixture from one or more fixture types.
2. Give the fixture attributes some values in the programmer.
3. Press and hold Store until the **Store Settings** open.
4. Tap Preset Mode until the mode is **Global**.
5. Tap the desired preset pool object - make sure it is a valid pool object for the attribute values.

Store Universal Preset

1. Select one or more fixtures or the universal fixture.
2. Give the fixture attributes some values in the programmer.
3. Press and hold Store until the **Store Settings** open.
4. Tap Preset Mode until the mode is **Universal**.
5. Tap the desired preset pool object - make sure it is a valid pool object for the attribute values.

Values that are a part of the universal fixture type are stored in this fixture. Attributes outside the scope of the universal fixture type are stored as global values.

MAgic Presets

The concept of MAgic presets is explained in the **Presets** topic.

Important:
The MAgic presets only work if the stored fixtures have different grid positions on the relevant axes and the best result is if they start at grid position 0 and the fixtures are next to each other. This does not need to match the real-world position it is only for the individual positions between the fixtures.

Important:
Calling a MAgic preset in the programmer does not create a link to the preset. The fixture values are calculated based on the selection and they are stored as hard values. The only way to keep a reference to the MAgic preset is if the preset is used in a recipe. Learn more about recipes in the **Recipe Preset topic.**

This is the general workflow for creating a MAgic preset:

1. Select all the needed fixtures.
2. Use Next to give the first fixture values to match the first point in the range.
3. Press Next to select the next fixture and give it values for the next point in the range.
4. Repeat step 3, if needed, to have a maximum of five fixtures (on each axis in the selection grid).
5. Select all the used fixtures using Set.
6. Store the preset in an appropriate preset pool.
7. Edit the preset settings and tap MAgic to turn the MAgic function On.

The new preset can now be used by as many fixtures as needed.

Learn more about editing the **Edit Presets topic**.

Known Limitation:
Storing MAgic preset only works with selective data as you cannot define multiple points across universal or global fixtures.

Example:

A path needs to be defined to control a color range from blue to yellow. The wish is to have the range go through the magenta/red area in a CIE color picker - instead of going through white or the blue/green area. Three fixtures are needed to define the range.

Requirement:

- Patch a number of fixtures (10 or more) with color mixing possibilities.
- Create a view with the color picker (in CIE mode) and the All1 preset pool.

Follow these steps:

1. Select three of the fixtures.
2. Turn the fixture intensity to 100%.
3. Press Next to select the first of the three fixtures.
4. Tap the blue area in the color picker.
5. Press Next to select the second fixture.
6. Tap the magenta area in the color picker.
7. Press Next to select the third fixture.
8. Tap the yellow area in the color picker.

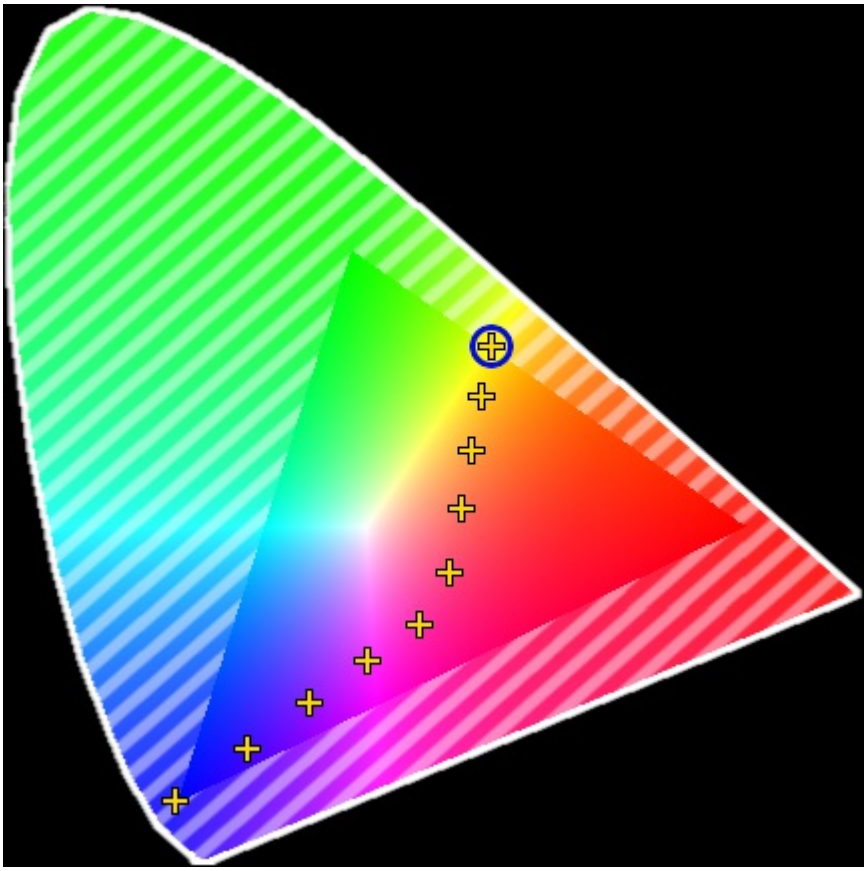
9. Press Set to select all three fixtures again.
10. Press Store.
11. Tap an empty pool object in the All1 preset pool.
12. Use the **swipey menu** to select Edit Settings.
13. Tap MAgic to turn On the option and close the settings.

Now there is a preset that looks something like this:

MAgic preset in the preset pool

14. Clear the programmer.
15. Select all the fixtures.
16. Tap the MAgic preset.

The result should look similar to this in the color picker:



Result of the MAgic preset used on multiple fixtures

The dimmer values are not needed for this example, it just makes it easier to see the result.

Multistep Preset

Multistep presets are presets that contain values in more than one step. It is explained in a little bit more detail in the **Presets** topic. For an explanation of steps, please read the **Phasers** section.

With Phaser information in the programmer, a preset can be stored. This will then be a multistep preset.

Hint:
Preset can be used for more advanced functions with **Recipes**. Please read the **Recipe Presets** topic for more about recipes.

19.3. Recipe Presets

Recipes can be a very useful tool for touring shows or when the show changes a lot.

Recipes can be stored in cue parts and presets. A cue part or preset can contain multiple recipe lines describing what should happen based on a set of information. The recipe can be used to "cook" values into the cue part, preset, or programmer. This cooked data is marked by a small pot icon and the cooked data can easily be removed again if needed.

A recipe line can contain information about a group, preset, MAtricks, individual fade, delay, speed, and phase values.

Values from recipes can be combined with conventionally stored values.

The rest of this topic explores using recipes in presets.

The flexibility in the recipe system allows for a variety of uses. The recipes could be used to:

- Create recipe presets for groups with references to other presets for a flexible fixture setup.
- Create "template" presets with ranged values that can be applied to a flexible selection of fixtures.

Adding Recipe Lines

Recipe lines are added to presets using the **Edit Preset Object** pop-up. This can be accessed using the **Swipey** on a preset pool object.

Edit options for a preset

The middle part of this pop-up is about the recipes.

The line with On/Off toggle buttons near the top of the editor is used to hide or show different element columns for the recipes.

- **References**:
  These are the columns for defining a Group, Preset, and MAtricks.
- **Layers**:
  These are the timing layers. X, Y, or Z also needs to be active for any of the layer columns to be shown.
- **MAtricks**:
  These are the MAtricks columns. X, Y, or Z also needs to be active for any of the layer columns to be shown.
- **Invert**:
  These are the Invert columns. X, Y, and Z have invert columns as well and might need to be active to show the desired columns.
- **X**, **Y**, and **Z**:
  Each of the grid axes has columns. If the recipe only uses one or two axes, then the columns can be limited to only show the relevant columns.

Hint:
All the MAtricks columns and settings are described in the **MAtricks and Shuffle** topics.

Create a Recipe

Tap Insert New Recipe to create a new recipe line.

It is almost always relevant to at least add a preset reference value. To do this tap and hold the field in the recipe row in the preset column.

This opens a **Preset Pool** selection pop-up. Here it is possible to navigate through the existing presets and select the desired preset. It does not have to be in the same feature group and the recipe preset.

Add the desired values in the other columns.

If the recipe contains a group, then the preset is automatically cooked when the edit pop-up is closed.

Example

We want a preset that can be used to take the current selection of fixtures to a new position in two wings and with a ranged time.

Requirement:

Have a show with some moving heads patched and placed in a row. The show also needs a position preset and dimmer preset where the fixtures are at full. This example uses the Demoshow.

Follow these steps to create the recipe preset:

1. Tap an empty position preset pool object and swipe the finger outside the preset to open the Swipey menu.
2. Select Edit Setting in the Swipey menu.
3. Tap Insert New Recipe to get a recipe line.

4. Tap and hold the first field in the preset column.
5. Tap Dimmer and then the preset where the intensity is at full (Open).
6. Tap and hold New Recipe to create a second line.
7. Press Assign.
8. Tap the position preset (Roof Floor) in the position preset pool.
9. Tap the empty recipe line to assign the preset to the line.
10. Select both recipe lines in the XWings column (if the column is not visible then make sure both "X" and "MAtricks" are active in the row with toggle buttons).
11. Edit this value so it says "2" in the XWings.
12. Select both rows in the "Fade From X" column and set the value to "1".
13. Select both rows in the "Fade To X" column and set the value to "3".

It should look like this:

The finished recipe

14. Close the editor.
15. Select some fixtures (for instance, group 1).
16. Tap the new recipe preset and see the fixture move while they turn on.

If the move and fade should start from the center then open the edit preset object pop-up again and set the InvertStyle to All and the InvertX to Yes for both recipe lines.

Working with Recipe Presets

There are three suggested workflows with presets using recipes:

1. Preset containing values and recipe.
2. Preset containing no values and recipe with group values.
3. Preset containing no values and recipe without group values.

Preset Containing Values and Recipe

The values of the Preset take precedence and the recipe line will not be loaded in the programmer (**Programmer Parts window**).

The intention is that the workflow is assuming that recipes are used to cook values into the presets, therefore it is not desired that the recipes are in the programmer so that it is NOT stored into a cue. Only the values of the preset go to the cue.

Hint:
If you want to quickly edit the recipe lines live in the programmer, Edit the preset. This will load the recipe into the programmer so it is possible to edit the recipes and update the preset. The recipe changes made in the programmer will also update the recipe lines in the preset.

Preset Containing No Values and Recipe With Group Values

Since the preset has no values, of course, no values will be directly loaded in the programmer. The recipe will be loaded into the programmer (**Programmer Parts window**) and therefore the values are cooked into the programmer. The cue can then be stored, which will store the cooked values and recipe to the destination cue part.

This workflow does NOT follow the current selection. A group is assigned to the recipe, so the cooking within the programmer will always refer to the group, not the current selection.

Think of these as recipe templates that allow to quickly build recipes into cues. It is the intention to use recipes within the cues to re-cook cues from venue to venue in a touring show.

Hint:
Using Presets this way is only a quick way to build recipes into cues. After the recipes are in the cues, you are not using the preset anymore because you should now only work with the recipes in the cues. Think of it as a "Recipe Template Preset".

Preset Containing No Values and Recipe Without Group Values

As there are no values and no group, this recipe works with the current selection of fixtures. The recipe is not loaded into the programmer. Instead, the recipe preset link and MAtricks are used with the current selection of fixtures to cook values directly to the programmer.

This is a great way to have a bunch of template objects in the preset pools that allow to quickly call complex looks based on MAtricks ranges.

Additional Cue Recipe Workflow

Recipes from the Programmer Parts window will store to the destination cue and will respect the parts.

Example 1:

1. In the Programmer Parts window build some recipes in Part 1, 5 and 7.
2. Store Cue 10.
3. These Recipes will go to CueParts 1, 5, and 7 inside Cue 10.

Example 2:

1. In the Programmer Parts window build some Recipes in Part 1, 5 and 7.
2. Select Part 5.
3. Store Cue 11 Part 23.
4. Only the Recipes of selected ProgrammerPart 5 will go to CuePart 23 of Cue 11.

19.4. Use Preset

Presets are often used for live playback or as building blocks in cues. Tapping a preset to get the values uses the **At** keyword.

Calling Preset Into the Programmer

The first step is often to select the desired fixtures and then calling the preset into the programmer, no matter if the preset is used for live busking or in a cue list.

The workflow is the same for every kind of preset (MAgic, Recipe, Multistep, standard single step, etc.)

If the programmer does not have a fixture selection and a preset is tapped, then the first tap selects all the fixtures who can use the preset. In this case, it works a little like a group.

Tapping a preset with a fixture selection in the programmer calls the preset if it is valid for the selected fixtures.

Timing values can be stored in the preset and calling a timed preset uses the stored timing to fade into the values.

Another way to use timing with presets is the **Programmer Time** master. Learn more about the master in the **Time Control** topic.

Preset with stored timing values have a higher priority than the **Programmer Time** master and the stored timing will be used when the preset is called.

Known Limitation:
Calling a MAgic preset into the programmer will extract the data and it will not reference the preset. It is recommended to use MAgic presets in combination with recipes to maintain referenceable data.

Store Presets in Cues

When a preset is in the programmer, then it can be stored into cues. Learn more about storing cues in the **Store Cues** topic.

A reference to the preset is stored for the specific fixture attributes. This means that the value stored in the preset is not stored into the cue, but a reference to the preset is stored. So if the value stored in the preset is changed after the cue is stored, then the cue still looks in the preset to get the value when the cue is played back and the values in the preset will be used.

If attributes are added to or deleted from the preset after it is used. And the stored cues referencing the preset need to reflect this new change, then the preset needs to be recast. This can be done by using the **Recast** keyword or by the **Edit Preset Object pop-up**. Recasting a preset removes or adds attributes to the fixtures in the cue.

When presets are used in cues and later deleted, then the preset values are transferred to the cue.

Assign Presets to Attributes in the Tracking Sheet

When a value is edited in the **Track Sheet**, then the available presets can be chosen in the **Calculator**.

Extract Preset Values

Preset values can be extracted to the selected fixtures. The preset needs to be valid for the selected fixtures. The **Extract** keyword is used for this.

The values stored in the preset will be pulled into the programmer without a reference to the preset. The values are then like any other normal programmer values.

19.5. Edit or Update Presets

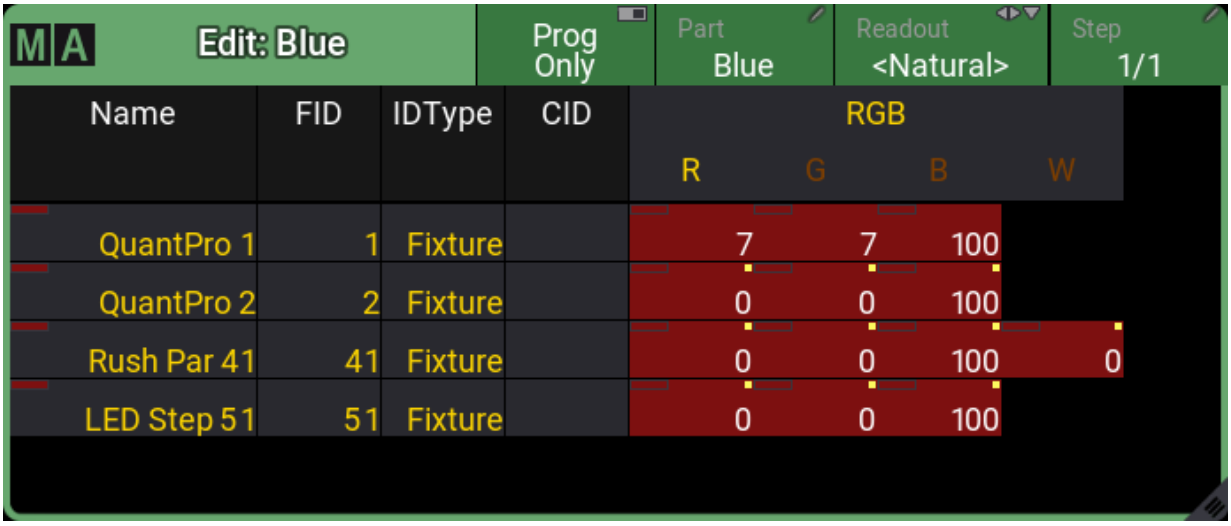
Edit Existing Preset

The existing presets can be edited. The values in the preset can be seen in a **Fixture Sheet** in **Fixture** mode.

> **Important**:
> Editing a preset call the values into the programmer and this is sent to output unless **Blind** mode is activated first.

1. Press Edit.
2. Tap the preset to be edit - this changes the frame around the fixture sheet to a green color and the fixtures are selected and the values are pulled into the programmer as active values. Edit starts blinking.

The sheet shows the fixtures that can use the preset and the attribute values.

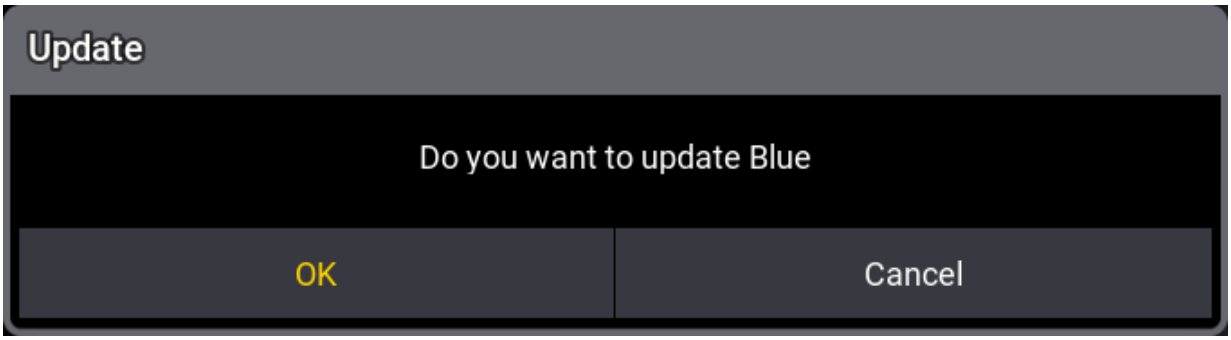


Fixture sheet in edit mode

The example above is a **Global** and **Selective** preset. The global data is marked with a yellow marker in the upper right corner of the attribute values. Any fixture with that fixture type gets the values from the global fixture type. The selective data are values without the yellow marker.

3. Edit the values in the programmer. Update lights up.
4. When the values are correct press Update.

This opens a pop-up asking for confirmation to update the preset.



Pop-up asking for update confirmation

5. Tap OK to confirm the update. Cancel will return to the update menu and not update the preset.

Hint:
To deactivate the edit mode without saving anything, press Esc. Remember that the values are still in the programmer.

Update Existing Preset

A preset that was called into the programmer, where the values have been changed after it was called, can be updated.

Update flashes when there are values that can be updated. Tap the key to open the **Update Menu**.

The presets that can be updated are listed on the left side of the menu.

The **Update Mode** is relevant when updating presets. There are two modes:

- **Original Content Only**:
  If a preset is updated with this mode, then only existing attribute values are updated.
- **Add New Content**:
  This mode update existing attribute values and adds any new attribute values to the preset.

The mode can be toggled by tapping UpdateMode.

The **Preset Mode** defines how the preset values are updated. The modes are described in the **Preset topic**.

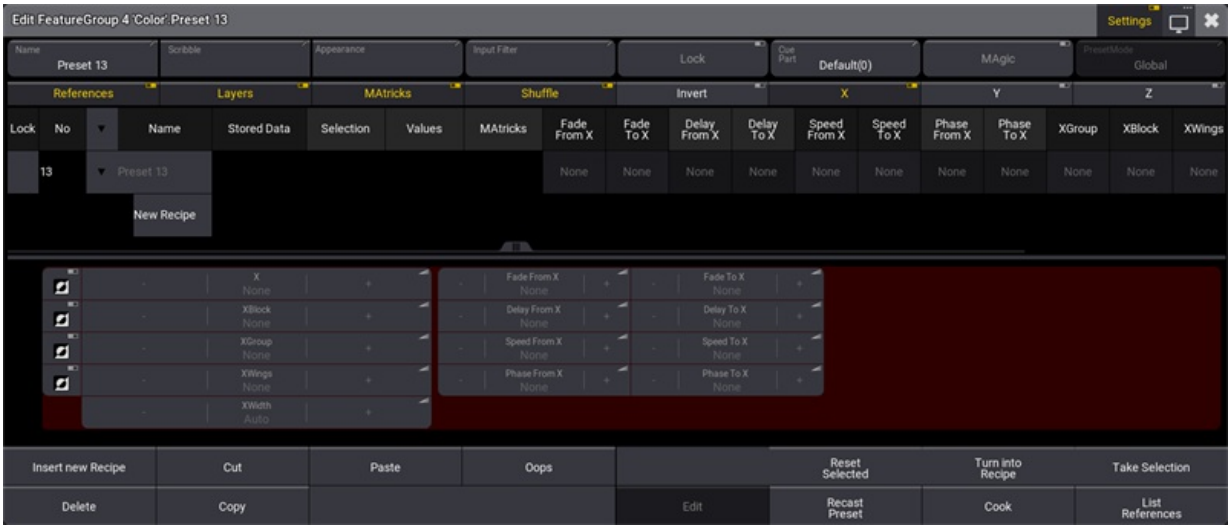
The mode set here is not linked to the mode selected in the store settings.

When the correct modes are selected then the desired preset can be updated by tapping it in the menu.

Recipe presets that reference the updated preset might need to be recooked to look as expected and cues might need to be recast for the expected result.

Edit the Preset Object

The preset object can be edited using the edit pop-up. It can be accessed using the **Swipey** menu or the **EditSetting keyword**.



Edit preset object including the settings

The settings in the top line are the same options as the ones for the entire pool, but here they can be set for the specific preset. Settings on a preset have a higher priority than the pool settings. Learn about the settings in the **Preset Pool** topic.

There is an extra setting here called MAgic. It is a toggle button that changes the behavior when the preset is called into the programmer.

Turning on **MAgic** is used for MAgic presets. Learn more about creating and using MAgic preset in the **Create New Presets topic**.

The settings on the top can be shown or hidden using the Settings toggle button in the title bar.

The rest of the editor is for recipe lines. Learn about these in the **Recipe Presets** topic.

20. Worlds and Filters

Worlds and filters can be used as tools for programming, playback, or filtering information in some windows.

Worlds are used to limit access to fixtures and attributes.

Filters are used to prevent attributes to pass a filter. Typically in store, update, and recall actions but also to filter what is displayed in some windows.

The selected world and filter always dictate what is possible using the programmer.

Worlds and filters can be assigned to some objects. For instance, a sequence or a preset. The world or filter then dictates what can be played back from or stored into the object.

Worlds

Worlds contain information about fixtures and attributes.

Worlds are used to prevent access. Fixtures and attributes not in the active world are removed in some windows and cannot be used in programmer actions.

They are stored in the Worlds pool. This can be created like any window using the **Add Window** pop-up.

World pool with world 1 selected

There is always a selected world. The selected world has a yellow frame around it.

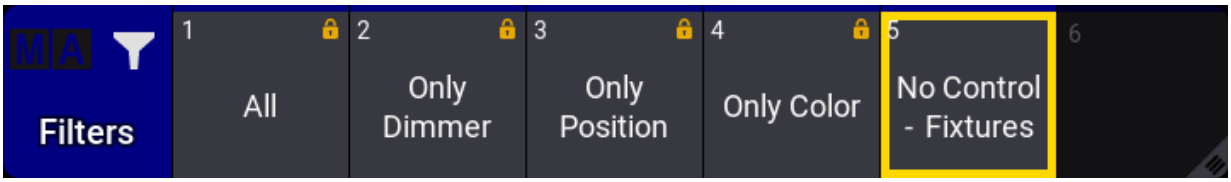
There is a default world from the factory. It is always world number 1 and it is called "Full". This automatically contains all fixtures and all attributes. This world is locked and cannot be edited.

Filters

Filters contain information about attributes and patch information about the fixture (Name, ID Type, Fixture Type, Layer, and Class).

Filters are used to block values from being stored or recalled. For instance, assigning a filter that blocks dimmer values as a playback filter on a sequence, prevents the dimmer values from being played back from that sequence. Assigning a filter that blocks certain ID Types to a **Layout** will hide these fixtures in the layout view.

Filters are stored in the Filters pool. This can be created like any windows using the **Add Window** pop-up.



Filter pool with filter 5 selected

There is always a selected filter. The selected filter has a yellow frame around it.

A new show will have some default filters from the factory. The first one is called "Open". It contains all attributes and layers. If the attribute structure changes, then this filter is automatically updated. The filter is locked and cannot be edited.

The other filters are named after the location they are automatically assigned. For instance, the **Sequence Sheet** filter (number four) is assigned to the sequence sheet in the predefined view. These filters are not locked and can be changed.

If a different filter than number 1 is the selected filter, then the **At key** flashes to indicate that there is an active filter.

Read more about worlds and filters in the subtopics.

20.1. AT Filter

The **AT Filter window** can be used to see and change the current filter settings.

The principle is that turned Off elements are blocked and do not pass the filter.

Attributes with a yellow text and a yellow bar above the text are turned On and they will allow information to pass through the filter.

There are three column groups. The left column displays the **Feature Groups**. The next column is the **Features** in the feature groups. The last group is the **Attributes** inside the features. Read more about the feature and attribute structure in the **Feature Group topic**.

The elements in this list change with the fixture setup. If fixtures are added with more attributes, then these attributes are added to the list.

The window can be created like any other window using the **Add Window pop-up**. It is in the **More** tap.

AT filter window with all selected

The image above shows all attributes and layers selected (yellow text).

The title bar in the **AT filter window** has some buttons:

- **Line Height**:
  This swipe button changes the spacing above and below the text in the main window. There is a **Default** value. It corresponds to size 50.
- **Select All**:
  Tapping this button select all attributes and layers.
- **Select None**:
  Tapping this deselects all attributes but does not change the layer selection.

The main part of the window displays all the attributes and the structure.

The structure dictates that if a feature group is Off, then all features inside are also Off. If a feature is Off then all attributes inside are Off.

If a single attribute is On, then both the feature and feature group is On.

The **AT Filter** window follows the selected filter in the **Filter Pool**.

Filter 1 in the filter pool is called **Open**. This contains all attributes. It is automatically updated if the structure changes. It is locked and cannot be changed.

Temporary Version

The **AT Filter** window exist as a temporary pop-up version that can be opened by tapping At Filter in the **At overlay**.

The functions are the same in the pop-up as in the window.

20.2. Create a World

Worlds are created in the programmer and stored in the world pool.

Worlds may be thought of as matrix with rows (fixtures) and columns (attributes), and being in a world may block programmer access to rows and/or columns in sheets.

The created world will contain the rows of the current fixture selection and the columns of any active attribute. If no attributes were active, all columns/attributes are included in the created world. If no fixtures were selected, then all fixtures are a part of the world.

Examples

Requirement:

Have a world pool and **fixture sheet** visible on a screen.

Create a World with Fixture 1 to 5

1. Clear the programmer.
2. Select fixtures 1 thru 5 (or any 5 fixtures in your show).
3. Press Store followed by an empty world in the **World Pool**.
4. Tap the newly created world.

Now there is only programming access to fixture 1 thru 5, which is reflected by the fixture sheet. All attributes are available.

Create a World with Fixtures and Attributes

Worlds do not contain any values, but active attribute values are needed to store the attribute information for filtering in the world.

Example with five fixtures and only one attributes:

1. Tap world 1 (Full) in the **World Pool**.
2. Clear the programmer.
3. Select fixtures 1 thru 5 (or any 5 fixtures in your show).
4. Give the fixtures a dimmer value - the values do not matter.
5. Press Store followed by tapping an empty world in the **World Pool**.
6. Tap the new world.

Look in a fixture sheet. It only displays the selected fixtures and the dimmer attribute.

Store a World with All Fixtures but Only Some Attributes

If the world is stored without selected fixtures then all fixtures can be used.

Example with only position attributes:

1. Tap World 1 (Full) in the **World Pool**.
2. Clear the programmer.
3. Select fixtures 1 thru 5 (or any 5 fixtures in your show).
4. Give the fixtures pan and tilt values - the values do not matter.
5. Press Clear once to clear the fixture selection but keep the pan and tilt values active.
6. Press Store followed by tapping an empty world in the **World Pool**.
7. Tap the new world.

Now the fixture sheet lists all the fixtures, but only the pan and tilt columns.

As usual, it is really a good idea to name the worlds as soon as they are created.

Edit a World

The content of the world can be edited using the following steps:

1. Press Edit followed by the desired world. This takes it into edit mode.
2. Change the programmer selection to match the desired fixtures and attributes.
3. Press Update followed by the world.
4. Confirm the update by tapping **OK** in the update pop-up.

The world options can also be edited. This is done using the following steps:

1. Use the swipey menu on the desired world.
2. Select the Edit Option button in the swipey. This opens an **Edit World** pop-up.
3. Make the desired changes.
4. Close the pop-up.

The edit world pop-up gives access to the following options:

- **Name**:
  This is the name of the world.
- **Appearance**:
  This can be used to select an appearance to the world.
- **Scribble**:
  This can be used to select or create a scribble for the world.
- **Master**:
  This can be used to limit the intensity output of the fixtures in the world.

There are also two buttons. Edit Content sets the world in edit mode as described above. List Reference opens another pop-up that lists the different elements that use the world.

20.3. Create a Filter

Filters are stored in the filter pool. There are two main ways to create filers.

Edit one of the pool objects. This opens the **Edit Filter pop-up.** Or store a new filter pool object with an active filter.

The **AT Filter window** is a nice tool to see the current filter settings.

Create Filter From the Pool

Editing a filter pool object opens a menu that looks almost the same as the **AT Filter** window.

Edit Filter pop-up

The only difference between them is some extra input fields and the fixture patch information filter at the bottom.

The extra fields at the top can be used to edit the name, appearance, and scribble. At the bottom, there is also a List References button that opens a list of elements that relates to this filter. For instance, if the filter is assigned to a preset.

The extra patch information filters at the bottom offer to add information from the patch as filters. The different information is Name, ID Type, Fixture Type, Layer, and Class. Read more about these elements in the **Patch and Fixture Setup section**. Values can be added to each element type. These values can be used as a positive or negative filter. Meaning that fixtures with the value can pass the filter or be blocked by the filter. Next to each of the elements is a - button. Tapping this toggles it On or Off. If this is white, then it is positive and fixtures with the values can pass. If it is yellow then it is negative (On) and fixtures with the values are blocked. For example, if the **ID Type** has **Fixture** and **Channel** values and the - is white, then the filter allows fixtures with the two ID types

to pass the filter. Other ID types are blocked. These filter options make it possible to have very precise filters.

The title bar has an extra button that allows this pop-up to be moved to a different display.

Follow these steps to create a filter by editing the pool object:

1. Edit a filter using swipeys, typing a command (**Filter keyword**), or using the keys (**Group key**).
2. Make sure the desired attributes and layers are active.
3. Add the desired patch information filter values.
4. Optionally give it a name by editing the **Name** input field.
5. Optionally assign an **Appearance** and **Scribble** by editing the input fields.
6. Close the editor by tapping the  in the upper right corner.

Using the edit function is the easiest way to update or change an existing filter.

Create Filter From Active Filter

The **AT Filter** window is needed for this. Also, the **Filter** pool is nice to have visible.

1. Change the filter settings using the **AT Filter** window.
2. Store a new filter by pressing Store followed by tapping an empty pool object or specify the filter using keys or commands (use links above).

20.4. Use a World or Filter

There are several scenarios where worlds and filters can be useful.

When Programming

Many actions obey the world and filtering. For instance store and delete.

When any store or delete action is used, then the action is filtered through the selected world and filter. Only the fixtures and attributes passing the filtering can be affected.

For instance, having filters for each feature might help to program different values into different cues or cue parts.

In Windows

Some windows can have a filter assigned as a mask for the window. This will hide the blocked elements from the view.

Windows also obey the selected world and hide fixtures and attributes that are not in the world.

The Fixture Sheet and Sequence Sheet also have the option to turn on the **Filter Toolbar**. This toolbar is the patch elements filter - learn more about it in the **Create a Filter topic**. These filter options can be part of a filter from the pool, but they can also be a local sheet option in this toolbar.

The filters from the pool need to be assigned using the Settings pop-up. Read more in the **Window Settings topic**.

On Sequences

Sequences can use worlds and filters as input and output filters.

A world or filter can be assigned as an input filter. This limits what can be stored in the sequence.

Worlds and filters can also be assigned as an output filter. This limits which fixtures and attributes the sequence sends values to during playback.

Read more about assigning filters in the **Sequence Settings topic**.

In Preset Pools

Preset pools and single presets can use worlds and filters as input filters. This limits what can be stored in the preset pool or the single presets. This can be especially interesting for the All preset pools.

Read more about assigning the filters to presets in the **Preset section**.

20.5. Delete a World

Worlds can be deleted like most other pool objects.

Deleting the world does not delete the fixtures or attributes from the show. It is only the world that is deleted.

If a world is assigned as input or output filter to a sequence or preset pool, then this is also removed when the world is deleted.

Oopsing the deletion brings back the world and also the filter settings for the sequence and preset pool.

The general syntax for deleting a world is:

Delete World [World_ID]

The world_id can be the world number in the pool or the world name. It can also be a range of worlds.

World number 1 is locked from the factory and cannot be deleted.

20.6. Delete a Filter

Filters can be deleted like most other pool objects.

Deleting the filter does not delete the attributes from the show. It is only the filter that is deleted.

If a filter is assigned as input or output filter to a sequence or preset, then this is also removed when the filter is deleted.

Oopsing the deletion brings back the filter and also the filter settings for the sequence and preset.

The general syntax for deleting a filter is:

Delete Filter [Filter_ID]

The filter_id can be the number in the pool or the name. It can also be a range of filters.

Filter number 1 is locked from the factory and cannot be deleted.

21. MAtricks and Shuffle

MAtricks is a tool that can be used to divide a selection of fixtures into sub selections.

The general workflow is that a selection of fixtures is selected, then different MAtricks settings are applied to select fixtures inside the main selection.

For example, ten fixtures are selected and you want to step through these ten fixtures one at a time to do a position correction. MAtricks is the tool used to do this.

The selection can be shuffled using a set of shuffle tools. Read more **below**.

MAtricks Tools

One of the ways to work with MAtricks is the MAtricks toolbar or window.

The MAtricks window is separated into 3 sections. One for each axis. The X-axis section has a red background, the Y-axis section has a blue background, and the Z-axis section has a green background.

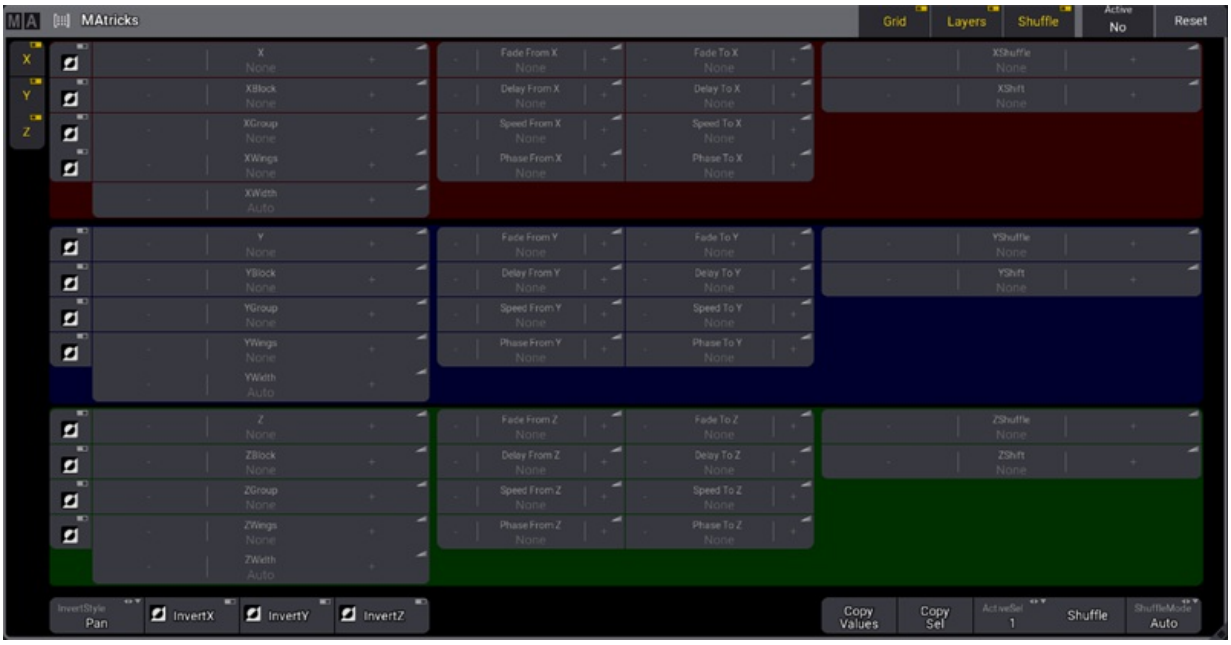
Tap X, Y, or Z in the toolbar on the left side to display or hide the corresponding axis.

Each axis section has its own properties that are grouped together.

Here are the groups and their properties:

- **Grid**: Axis (X, Y, Z), Block, Group, Wings, and Width.
- **Layers**: Fade From/To, Delay From/To, Speed From/To, and Phase From/To.
- **Shuffle**: Shuffle and Shift.

Tap Grid, Layers, or Shuffle in the title bar to display or hide the corresponding group.



MAtricks tools in a window

The above is the MAtricks tools available in a window that can be created like any other window.

There is an overlay version of the window that can be opened by tapping MAtricks in the standard **Encoder bar**.

The two versions have the exact same buttons and options. Read the next topics for details about the different options.

Dimensions and Selection Grid

The MAtricks toolbar shows that there are a lot of settings that can be applied to X, Y, and Z. These are the three dimensions used by the **Selection Grid**.

This means that if the fixtures are in a 3-dimensional grid selection, then the MAtricks tool can be used in all three dimensions.

MAtricks Pool

The different settings that can be made in the MAtricks tool can be stored in the **MAtricks Pool**.

This pool can be created like any other window.

MAtricks pool with some store MAtricks

This pool works just like most pools in the grandMA3. The two most used functions are storing the MAtricks settings and recalling them.

Store a New Pool Object

This is the process for storing a MAtricks pool object.

The easiest way to do this is by long-pressing an empty pool object. This stores the current MAtricks settings (even no settings are stored).

The general syntax uses the **Store keyword** and the **Matricks keyword**: **Store Matricks [number]**

Simple example

This simple example uses ten fixtures.

Select the ten fixtures without any specific grid information. Press Highlight to turn the intensity on

This is how it looks in the **Selection Grid** and the **MAtricks** window:

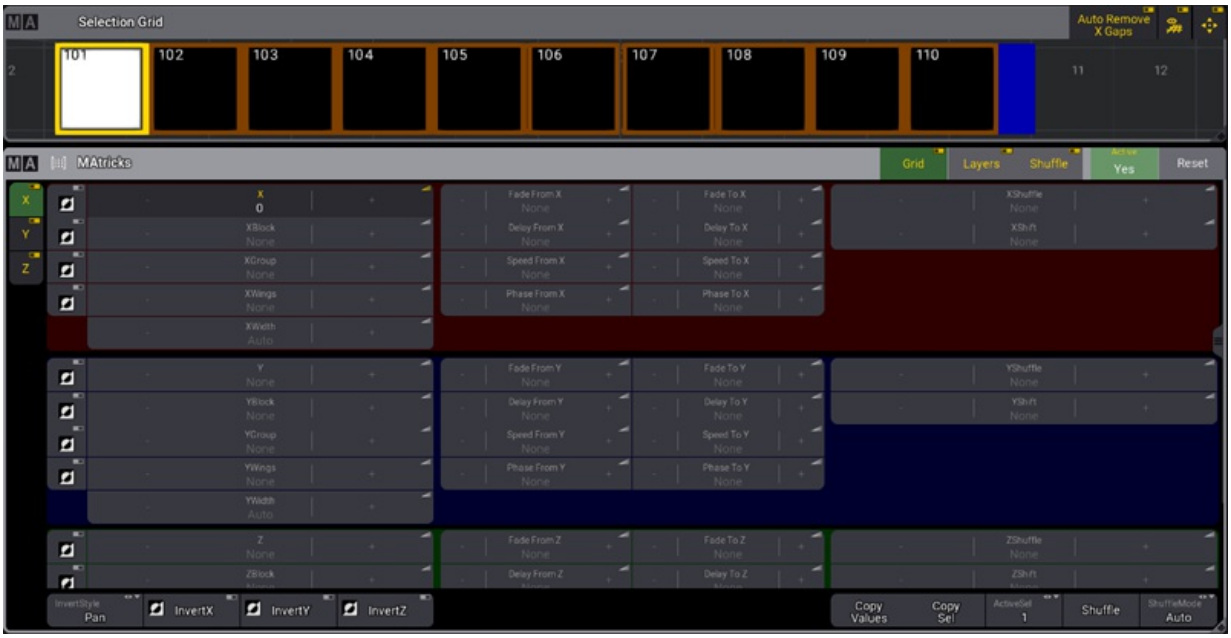
Ten fixtures without any MAtricks settings

These ten fixtures are now in one row on the X-axis.

Press Next. This is the same as tapping the + in the X setting.

Hint:
It is also possible to tap, hold, and slide your finger across any value area to change the value.

This is the result:



Ten fixture with X 0

Notice that the MAtricks is now active - the Active button in the title bar is On. This can be used to toggle the MAtricks tool On or Off without resetting the MAtricks settings.

The first fixture is now outputting lights, and the others are not. The selection frame around the first fixture is the usual yellow color. The rest have a darker yellow selection color. This indicates that they are part of the bigger selection and will output again if the MAtricks is turned Off or reset.

Try pressing Next to step through the selection. Pressing Prev (previous) goes through the selection in the opposite direction.

Notice how the X number in the MAtricks tool updates with the key presses. There is no difference between using the - and + buttons in the X-axis in the MAtricks tool or the keys.

A specific X number can be reached using the command line. For instance, if X needs to be 6, then the following command can be used:

```
User name[Fixture]>MAtricks 'X' 6
```

When you are comfortable with Next and Prev, then press Set.

The result is the same as tapping Reset in the MAtricks tool title bar and all fixtures again outputting light.

Multiple Selection

Each user profile has two different fixture selections called selection 1 and selection 2.

The current active selection can be seen and changed on the ActiveSel button in the MAtricks tool. Tapping this button changes between the two selections.

The selection can also be changed using the **Selection keyword**.

For example, the command needed to change to selection 2 is:

A selection of fixtures can be copied from one to the other by tapping Copy Sel. It copies the currently active selection to the other.

Values can be cloned from one selection to the other by tapping Copy Values. This applies the values from the currently active selections to the other selection.

A **ClearAll** command clears both selections. **ClearSelection** only clears the active selection.

Shuffle Selection

The **Shuffle keyword** allows shuffling the selection order in a random sort.

By default, Shuffle will randomize the selection order on all 3 axes in the **Selection Grid**.

Each shuffle setting per axis can be set to a value from 0 (=None) up to 32 767. Each value represents a different shuffled selection order. When selecting the very same amount of fixtures, again and again, the same shuffle value will result in the same shuffled selection order. This can be useful when a specific nice shuffled selection order is desired for the same number of fixtures. In this case, apply the same shuffle value in the MAtricks when having the same number of fixtures selected. The fixtures are then shuffled the exact same way.

Tapping Shuffle in the MAtricks tool shuffles all three axes in the Selection Grid by adding a random number in XShuffle, YShuffle, and ZShuffle.

To shuffle on a single axis, enter a shuffle value for the desired axis in the MAtricks tool or tap the + or - in the axis until there is a wanted shuffle result.

On the left side of the MAtricks tool, there is a Shuffle Mode with three available modes:

- **Auto**:
  When doing shuffle only for one dimension, this behaves like **Linked**. Shuffling on two or three axes behaves like **Unlinked**.
- **Linked**:
  All fixtures that are placed on the same position along the axis that will be shuffled but have a different position on the other axis will keep their alignment along the other axis.
- **Unlinked**:
  The fixtures placed on other axes than the axis that will be shuffled but have the same position on the shuffled axis will be shuffled independently.

Tap ShuffleMode to switch between these three modes.

Hint:
When deactivating or resetting the MAtricks, the original selection order will be restored.

Since the shuffle is part of the MAtricks, they are also stored in a MAtricks pool object like any other MAtricks setting.

See shuffle examples in the **Shuffle topic**.

Shift Selection

The shift setting in the MAtricks tool allows for shifting the current selection within the selection grid positions. This can be done per axis in the grid. Therefore, change the values for XShift, YShift or ZShift.
Positive values shift to the right (x-axis), to the bottom (y-axis), and the front (z-axis). According to this, negative values shift in the opposite direction.

Invert Options

Invert will define on which axis in the selection grid the values should be inverted when turning the encoder or applying a range of values.
The fixtures that will be inverted are displayed with a green font in the fixture sheet, a green body color in the 3D window, and a green border in the layout window and the selection grid window.

The following buttons can be found at the bottom left of the screen:

- InvertStyle: Defines if Invert shall be applied to Pan, Tilt, Pan and Tilt, or All attributes.
- InvertX: Inverts the overall invert of the current individual inverts per MAtrick property on the X axis.
- InvertY: Inverts the overall invert of the current individual inverts per MAtrick property on the Y axis.
- InvertZ: Inverts the overall invert of the current individual inverts per MAtrick property on the Z axis.

Grid properties can also be inverted by tapping  .

Hint:
When using Align in combination with Invert, the alignment is still based on the arrangement of fixtures inside the selection grid; however, the aligned values will be inverted.

Restriction:
At the moment, Align only works with the X axis.

21.1. MAtricks Blocks

The blocks function in the MAtricks creates blocks of fixtures of the specified size.

This treats blocks of fixtures as one fixture.

It is better explained with examples.

Example - Ten Fixtures One Axis

In this example, there are ten fixtures (101 through 110).

They are selected from 101 to 110 without any specific grid information. Highlight is activated.

Since there is no grid information, then the fixtures are only on one axis - the X axis.

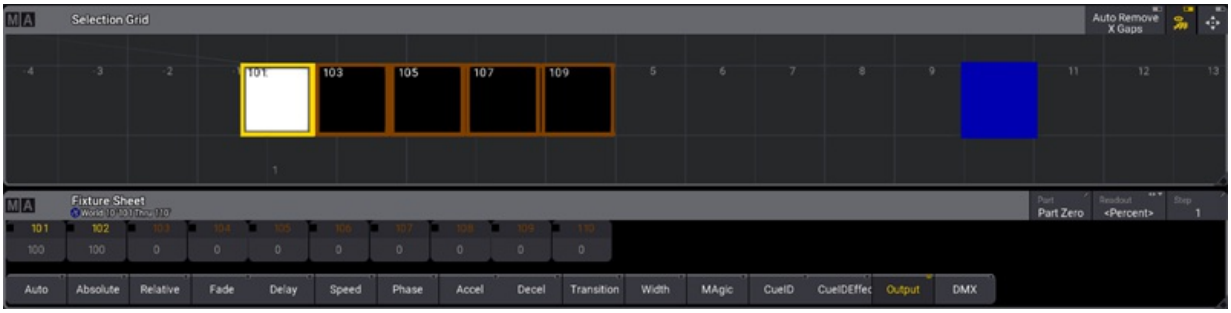
Press Next to make the MAtricks X value 0.

It looks like this in the Selection Grid and Fixture Sheet (Channel SheetMode and Output layer selected) window:

Ten fixtures with MAtricks X at 0 - No Block

Now tap + in the XBlock in the MAtricks window to set the value to two.

Now it looks like this:



Ten fixtures with MAtricks X at 0 and XBlock at 2

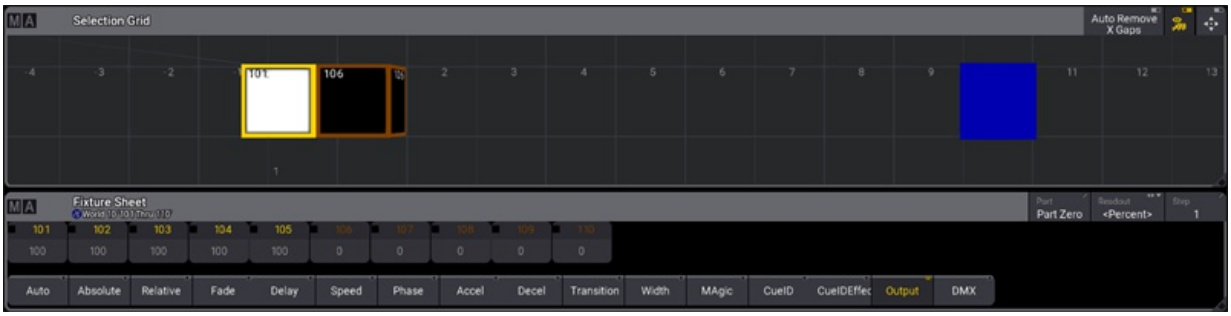
The selection grid now only shows five boxes. This is because fixtures 101 and 102 are blocked together and are in the first position. They are the ones currently outputting light. This can be seen in the fixture sheet.

Using Next and Prev jumps through the ten fixtures blocked together two and two.

Increasing the block size increases the number of fixtures together in a block. It decreases the number of visible boxes in the selection grid and the useful MAtricks X values.

For instance, if the block is increased to five, then there are only two boxes in the grid and only X = 0 and X = 1 actually selects the fixtures.

It looks like this:



Ten fixtures with MAtricks X at 0 and XBlock at 5

21.2. MAtricks Groups

Groups in the MAtricks are used to separate the selection into the number of groups set.

It alternates through the selection putting fixtures into each group.

It is best explained with an example.

Example - Ten Fixtures One Axis

In this example, there are ten fixtures (101 through 110).

They are selected from 101 to 110 without any specific grid information. Highlight is activated.

Since there is no grid information, then the fixtures are only in one axis - the X axis.

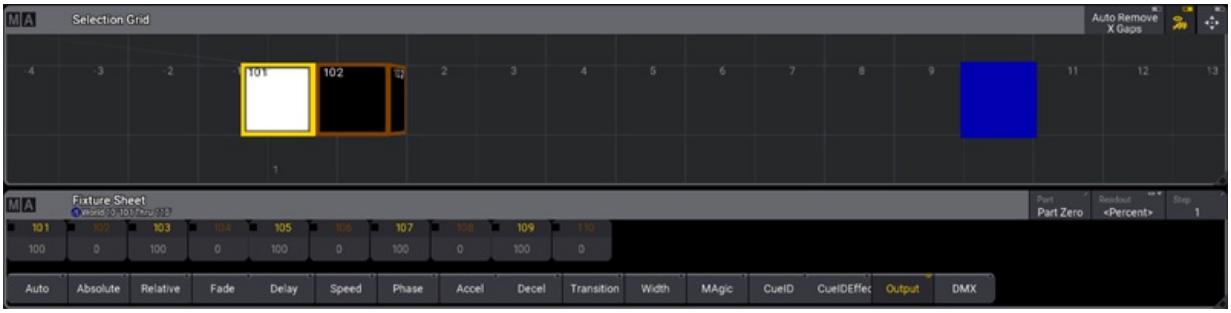
Press Next to make the MAtricks X value 0.

It looks like this in the Selection Grid and Fixture Sheet (Channel SheetMode and Output layer selected) window:

Ten fixtures with MAtricks X at 0 - No Group

Now tap + in the XGroup in the MAtricks window to set the value to two.

Now it looks like this:



Ten fixtures with MAtricks X at 0 - XGroup at 2

Notice how the ten fixtures are moved into two boxes in the selection grid.

The fixture sheet shows how the fixtures are interleaved into the two groups. The first fixture goes into the first group, the second fixture in the next group, the third fixture in the first group, and so on until there are no more fixtures.

The Next and Prev can be used to change the MAtricks X values. Values 0 and 1 are used to select the two groups.

21.3. MAtricks Wings

Wings in the MAtricks are used to separate the selection into the number of wings set and select devices from each wing from opposite directions.

It is better explained with examples.

Example - Ten Fixtures One Axis

In this example, there are ten fixtures (101 through 110).

They are selected from 101 to 110 without any specific grid information. Highlight is activated.

Since there is no grid information, then the fixtures are only in one axis - the X axis.

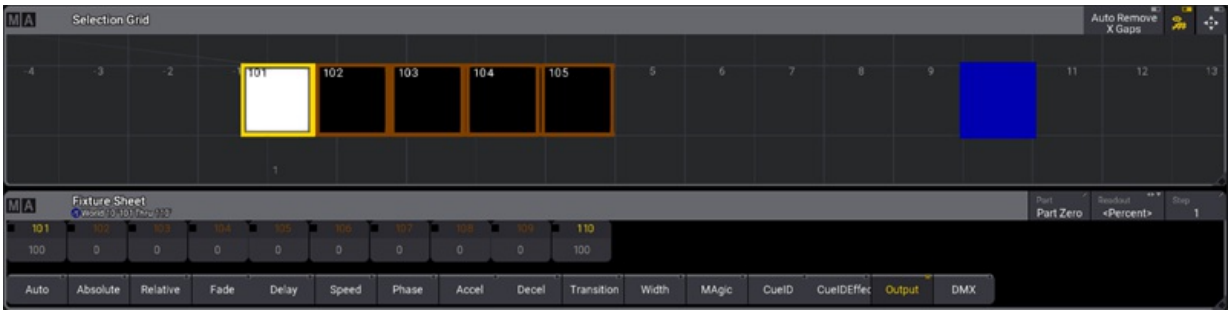
Press Next to make the MAtricks X value 0.

It looks like this in the Selection Grid and Fixture Sheet (Channel SheetMode and Output layer selected) window:

Ten fixtures with MAtricks X at 0 - No Wings

Now tap + in the XWings in the MAtricks window to set the value to two.

Now it looks like this:



Ten fixtures with MAtricks X at 0 and XWings at 2

The selection is split into two wings and each end of the selection is currently On.

Use Next to change the X number up. This moves the highlighted selection towards the "center" of the selection. Prev moves out again.

21.4. MAtricks Width

Width can appear to give the same result as groups. The difference is that groups place the grouped fixtures in the same grid positions where width moves the fixtures out on the next axis.

This makes it possible to combine several axes in the MAtricks selection.

It can be a great tool for fixtures arranged in grids.

Having the fixtures in an MAtricks with a width and storing a group in the **Group pool** also stores the current selection grid setup.

The MAtricks width is best explained with an example.

Example - Ten Fixtures One Axis

In this example, there are ten fixtures (101 through 110).

They are selected from 101 to 110 without any specific grid information. Highlight is activated.

Since there is no grid information, then the fixtures are only in one axis - the X axis.

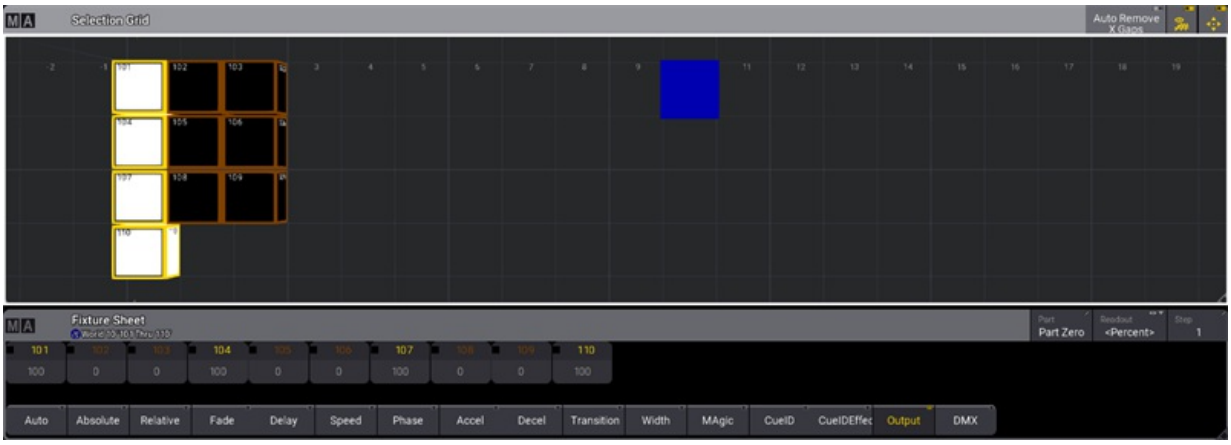
Press Next to make the MAtricks X value 0.

It looks like this in the Selection Grid and Fixture Sheet (Channel SheetMode and Output layer selected) window:

Ten fixtures with MAtricks X at 0 - No Width

Now tap + three times in the XWidth in the MAtricks window to set the value to three.

Now it looks like this:

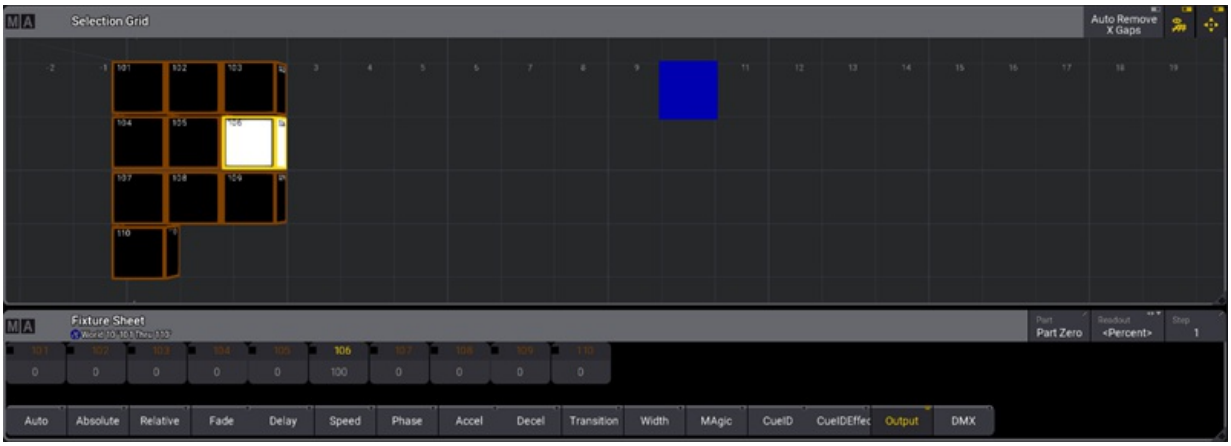


Ten fixtures with MAtricks X at 0 - XWidth at 3

All fixtures in the first X column is now highlighted.

This arrangement makes it possible to select a specific fixture or columns and rows of fixtures using a combination of X and Y values.

Highlighting fixture 106 in the current arrangement can be achieved by having X = 2 and Y = 1. It looks like this:



Ten fixtures with MAtricks X at 2, Y at 1, and XWidth at 3

Try to play around with different X and Y values.

21.5. MAtricks Shuffle

Shuffle is used to shuffle the selection order of the current fixture selection.

The shuffle function is described in more detail in the **MAtricks and Shuffle topic**.

This example uses the **MAtricks window**, the **Selection Grid window**, the **Fixture Sheet window,** the **Group Pool**, the **Color Preset Pool**, and the **All 1 Preset Pool**. It is useful to have them visible on the screens.

This is an example of a shuffle on two axes.

The main setup is 20 fixtures with color mix and four global color presets (Red, Green, Blue, and Magenta).

1. Fixture 101 to 105 is given the red color.
2. Fixture 106 to 110 is green.
3. Fixture 111 to 115 is blue.
4. Fixture 116 to 120 is magenta.
5. They are selected from 101 to 120.
6. Use the MAtricks tool to set XWidth to 5. This arranges the fixtures nicely in a 5 x 4 grid.
7. Store a group with this grid arrangement.
8. A dimmer range of 25% to 100% is applied.
9. This is stored in an All preset.

It looks like this:

Initial shuffle setup

Imagine this is how the fixtures are arranged in the real-world and this now needs to be randomly applied in both X and Y axes.

Tap Shuffle in the MAtricks tool until a desired pattern is achieved.

Notice in the fixture sheet that this does not change the actual output of the fixtures. It only changes how they are selected.

The values need to be reapplied to get the output looking like the random selection.

To get the dimmer applied simply press At 2 5 Thru 1 0 0 Please.

The Y value makes it easy to reapply the colors.

Tap + in the MAtricks Y so the value is 0. Now apply the red color preset. Tap + again to select the next row and apply the next color preset. Continue doing this for the last two rows.

Press Clear once and then tap the group stored at point seven in the initial setup.

The result could look something like this:

Fixture grid after the shuffle

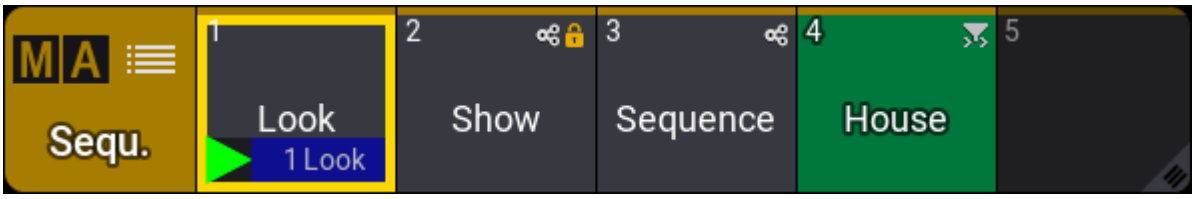
22. Cues and Sequences

Fixture values can be stored in **Presets** or in **Cues**.

Cues are organized in **Sequences**.

Executors can control and playback sequences. The sequence is played back from the sequence pool. The executors are handles for the sequence.

The sequences are all in the **Sequence Pool**. This is created like any other window - see more in the **Add Window topic**.



Sequence pool with sequence 1 running a cue

The sequence contains information about the cues and how to transition from one cue to another.

Cues can contain fixture values (sometimes called "hard values"), references to presets, and recipes. Read the **Store Cues topic** to learn about making cues.

All cues have a minimum of one step. If there is more than one step, then it is a **Phaser**. Follow the phaser link to read about multi-step cues.

Cues have **Cue Parts**. All values are actually stored in the parts. All cues automatically have part 0. Other parts can be created, but it is not necessary. Creating several parts allows separating values into different sub cues. The parts can have different timing or properties, but parts are connected to the primary (parent) cue. This means that all the cue parts in a cue trigger together with the primary cue.

The cues in a sequence can be seen in a **Sequence Sheet**. Read more about looking at the sequence in the **Look at Cues and Sequences topic**.

Sequences have a lot of different settings. Read the **Sequence Settings topic** for more information.

Filters

The sequence can have a world or a filter assigned as an input filter and also as an output filter.

If there is an input filter then there is a small icon () in the pool object. The input filter only allows the data in the world or filter to be stored in the sequence.

If there is an output filter assigned then there is a small icon () in the pool object. The output filter allows only the fixtures and attributes in the world or filter to be played back from the sequence.

See the example image above. Sequence four is called "House". It has both an input and output filter assigned and this is a combined filter icon (). It also has a green **Appearance**.

Read more about filtering sequences in the **Worlds and Filters section**.

Shared Data

Two sequences can be linked and share the cue data. If a sequence share data with another sequence, then it has a small icon () in the pool object. See sequences two and three in the example image above.

Shared data means that the cue content and the cue settings (for example fade times) are the same. Changing one of the two sequences will also change the other sequence. For example, creating cues, changing cue content, or deleting cues.

The sequence settings can be different. Different executors can control the different linked sequences and playback different cues in the two (or more) linked sequences.

This can be useful if during rehearsal a video programmer and a lighting programmer want to work with one sequence, but needs to be able to run different cues. Then relevant input and output filters can be assigned to the two shared sequences. Two different executors can be assigned to control the two linked sequences. Now one sequence can playback and store video data and the other can run and store light data. All values are stored in the same linked sequence. So when the show is ready, then an operator can remove the filters and just playback one sequence with all the data.

22.1. What is Tracking

Tracking is the principle to store only the **changes** in the cues.

If a fixture is turned On in blue color, it will stay like this until it is told to change. It does not matter how many cues there are between being told to turn On and the cue where it is told to turn Off.

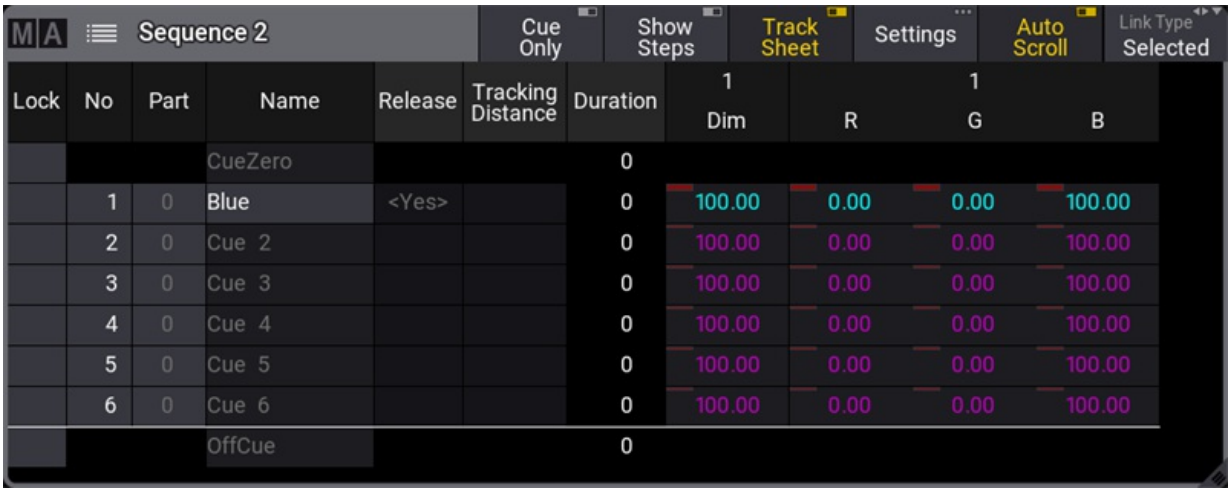
Instead of each cue between turning On and Off having the information that the fixture should be On in blue color, it is only needed to store it once - from there it is tracked.

Quite simply, tracking is the idea that once a parameter is set to a level, it stays there until it is told to go somewhere else.

Example

Fixture number 1 is stored in blue color, at 100 percent dimmer value in cue number 1. If there are 6 cues and fixture number 1 does not get any new information, then it will stay at 100% in blue color in all 6 cues.

The best way to see this information is by turning the **Sequence Sheet** into tracking sheet mode. This can be done by tapping Track Sheet in the title bar.



A sequence with 6 cues - fixture 1 stored values in cue 1

Notice the difference in text color between the values in cue 1 and the others. The magenta text color indicates that the values are not actually stored in the cues, but it is a tracked value from a previous cue.

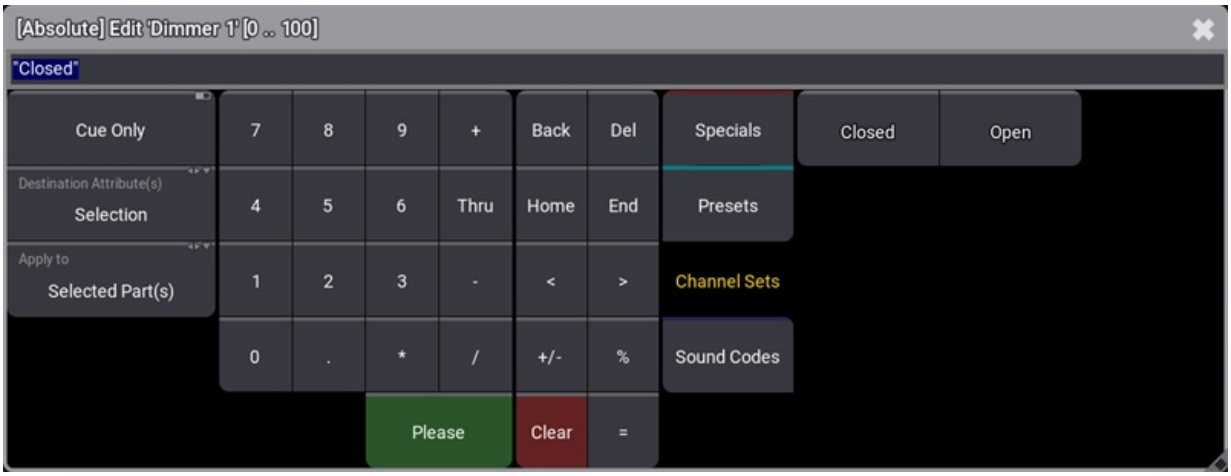
Now change the color of the fixture to red and set the dimmer to 100% again.

Store this in cue 4 **and** 5 (using the Merge option), now it looks like this:

A sequence with 6 cues - fixture 1 has a new color in cue 4 and 5 - blocked values

The text color of the dimmer value is now white in cue 4 and 5. All the color values are also white in cue 5 and the green attribute is also white in cue 4. This indicates that the values are stored here, but it is the same value as the tracked value. This is called "Blocked" values. This means that if any value is changed in any previous cue, then it would still be 100% in cue number 4. It is currently blocking the tracked value from cue 1 (the same value) and the value stored in cue 5 is tracking to the end.

Values can be edited directly in the tracking sheet. Edit the dimmer value in cue 3 by right-clicking with a mouse or the two-finger edit gesture on a touchscreen. This opens the **calculator**.



Calculator editor

Select Channel Sets and tap Closed.

This closes the calculator and changes the dimmer value to 0 in cue 3 in the sequence.

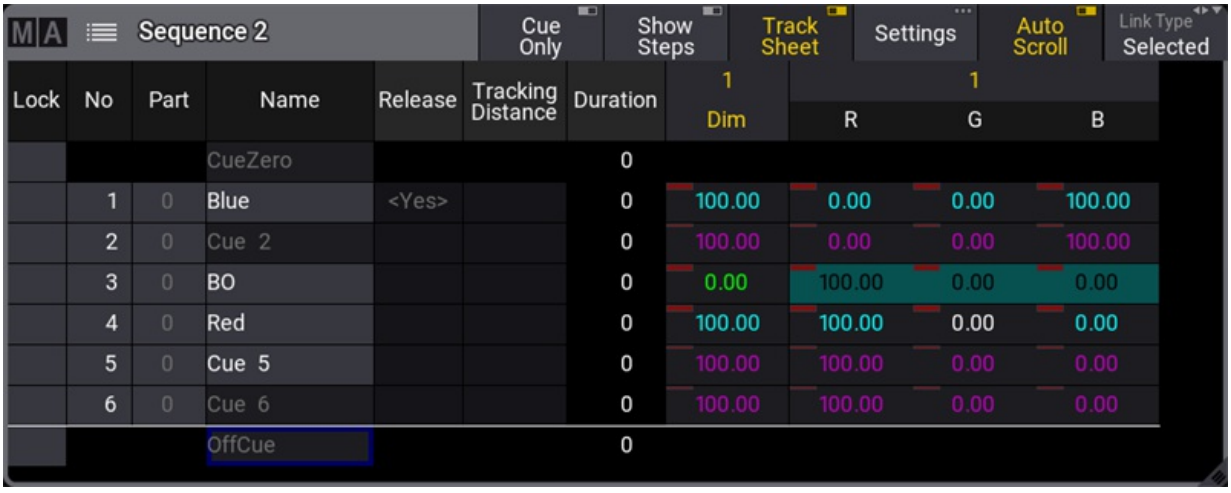
A sequence with 6 cues - fixture 1 has a new dimmer value in cue 3

The dimmer value is now green in cue 3. This indicates that the dimmer value is now stored with a lower value than the previous cue. It is only dimmer values that show this green indication for a lower value.

The deep-sea green background color on the color values indicates a move in black (MIB) prepositioning of the color values. Read more about this in the **Move In Black topic**.

The dimmer value in cue 4 is now cyan indicating a higher value than the previous cue.

The redundantly stored values in cue 5 can be removed from the cue by removing the stored values - this is also called "unblocking".



A sequence with 6 cues - unblocked sequence

There are several ways to do this.

One option is to have the values for fixture number 1 as active values in the programmer (it does not matter what the values are) and store cue 5 with the "Remove" option. This would remove the values from the cue.

A second option is using a command like this:

This removes all redundant blocked values from the entire sequence number 2. Notice that unblocking the entire sequence also unblock the green color value of 0. This might not be the desired result.

A third option is to unblock cue 5. This can be done by adding "cue 5" to the command above or by editing a value in cue 5. It does not matter what value all that is needed is the calculator opened with a value from cue 5.

On the left side of the calculator change the Destination Attributes(s) value to **All Fixtures** and the Apply to to **Selected Part(s)**. Now tap Specials and then Unblock. This unblocks all attributes for all fixtures in the selected cue parts. The selection here is the selected cue when the calculator was opened.

Tracking Distance

Tracking values can have a distance. This can be changed in the **Tracking Distance** column.

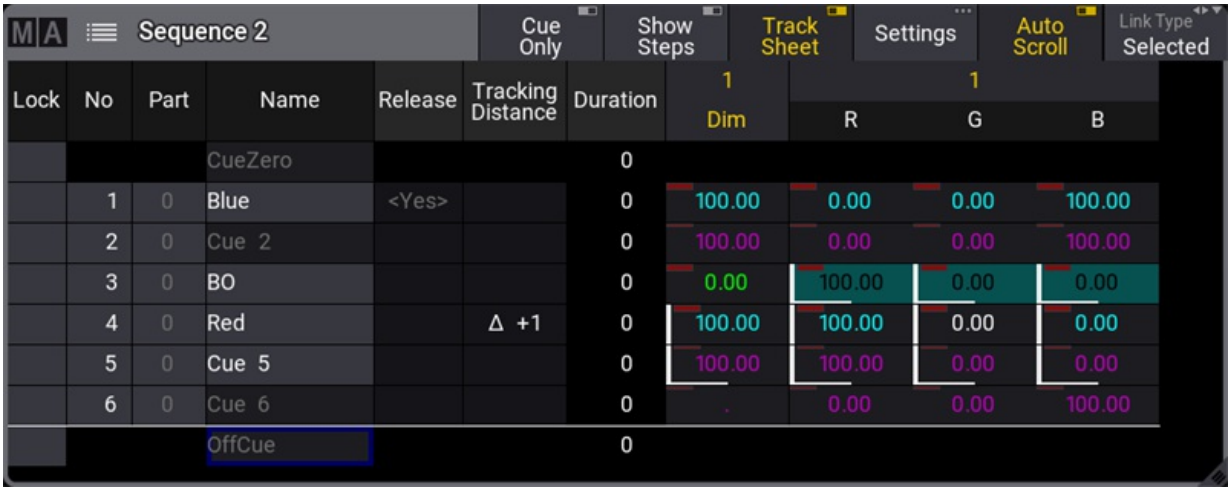
There are two types of tracking distances: Track a number of cues (delta) or track until a specific cue.

Both types release the attribute when the tracking distance ends. This will return the attribute to a previous tracked value or it will release it from the sequence. See the examples below.

The attributes affected and the number of cues the tracking distance is valid is displayed by a vertical white line next to the value.

Examples

This is an example of a sequence with fixture 1 at 100% intensity and in blue color in cue 1. The dimmer is set to 0 in cue 3. In cue 4 the color is changed to red and the tracking distance is set to Δ+1 by editing the cell.

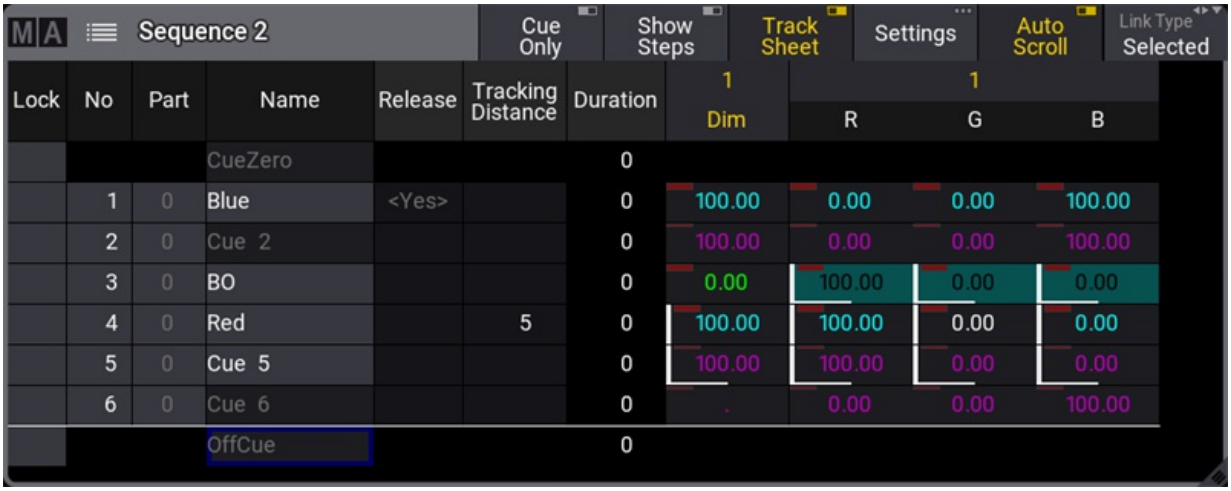


A sequence with 6 cues - Delta tracking distance of +1 in cue 4

As a result, the red color is tracking from the stored cue and one more whole cue number. Then the fixtures return to the previous tracked values. MIB is active for cue 4, so the color prepositions in cue 3.

The same result can be achieved by setting an absolute number. The number indicates what specific cue number the tracking values track to.

In this example, the tracking distance is set to cue number 5 in cue number 4.



A sequence with 6 cues - Cue 4 with a tracking distance until cue number 5

The result is the same as the delta example. The difference is that delta will look at the whole cue number and track the amount of whole cue numbers based on the delta value. Decimal cues numbers are ignored by delta tracking distance. The specific tracking distance cue number will track until that cue.

If for instance, cue 4.5 and 4.6 are added then the tracking distance will still stop the tracking in cue 5 with both delta +1 and the specific number. Delta does not count cue 4.5 and 4.6 since they are part of the same whole number as cue 4.

Important:
If you use delta tracking distance spanning cues that are not whole numbers (decimal numbers), then renumbering the cues in the sequence will change what cue the values are actually tracking to. The renumber function does count the actual amount of cues (including decimal numbers) whereas the tracking distance does not.

The values are released after the tracking distance. The result is that the fixture goes back to the values it had in the cue before the tracking distance. It is indicated as tracked values.

In the example, it gets the color tracked from cue 1 and the dimmer value from cue 3.

If there are no other values from previous cues, then the result is that the fixture is released from the sequence until it gets some values from a cue

Cue Zero

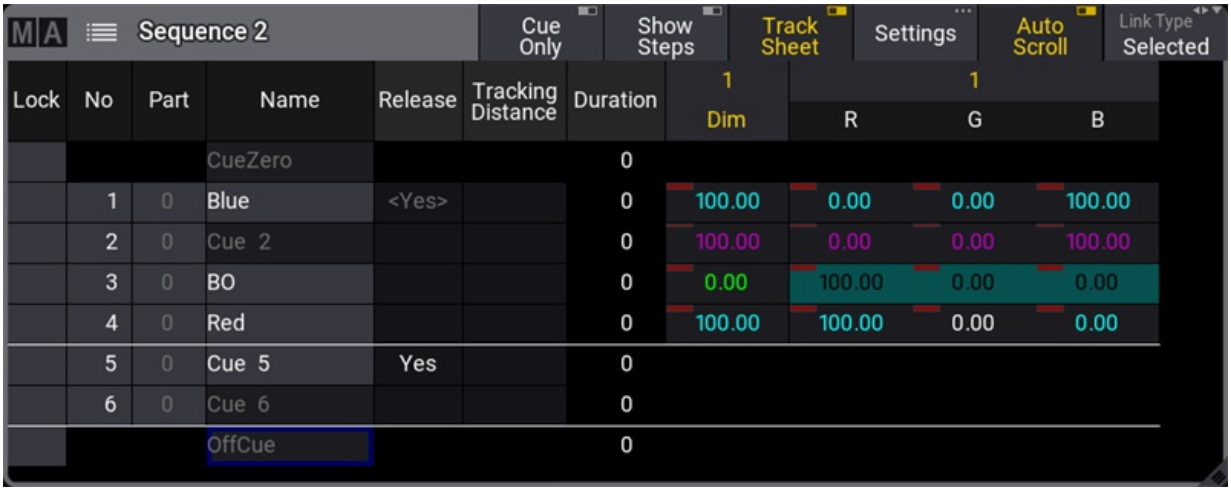
Cue Zero is an automatically created empty cue. Cue Zero can be modified or stored into using the cue name.

Off Cue

The Off cue has timing information that is used then the sequence is turned Off.

Release

The sequence sheet has a **Release** column. Editing a field in this column changes the cell to **Yes**. This means that tracked values will be released from this cue and forward.



A sequence with 6 cues - cue 1 and 5 with release

It makes a white horizontal line above the released cue. This indicates that tracked values are stopped and if there are not any other values in the sequence, then the attribute is released from the sequence - empty black cells.

When an attribute is released in the sequence, then it is the same as the sequence no longer sending any information to the attribute.

If a different sequence is sending values to the attribute, then these values are used. Sequence priority can be important in this case - read about priorities in the **Play back Cues topic**.

If the attributes do not receive values from any sequence or the programmer, then values go back to the default values for the attributes.

It is also possible to store values using the **Release** option. This empties the relevant fields in the sequence tracking sheet.

Turning Tracking On or Off

In the sequence settings, there is the option to turn On or Off the tracking function.

When tracking is Off, the tracked values are gone, and where the value was tracked it will be released instead.

Looking at the sequence used in the example above, it would look like this without tracking:

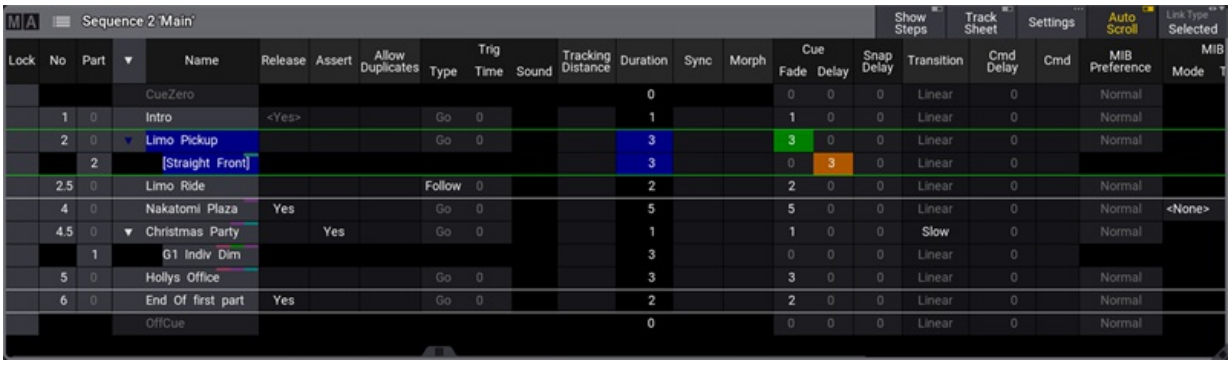
Sequence with Tracking turned Off

Fixtures 1 would only be On in cue 1 in blue. In cue 2 it would return the default values. In cue 4 it would turn On in red. In cue 5 the color returns to the default. The zero value stored in cue 3 is possibly redundant since it is often the default dimmer value.

22.2. Look at Cues and Sequences

The different sequences are best seen in the **Sequence Pool**.

The best way to see the cues inside a sequence is the **Sequence Sheet**:



Sequence sheet for a sequence called "Main" - Cue 2 is active

The sheet shows the cues and cue parts in rows. The different cue settings are in columns. Read below for a description of the different columns.

The purpose of this sheet is to see the cues in a sequence. It also shows the active cue with a green frame. Cue fades and delays are visualized with moving bars while the fades are running.

The Sheet can be created as a window in a view using the **Add Window pop-up** or it can be displayed as a temporary pop-up by editing a sequence pool object.

The sheet shows a lot of different markers and colors - read more about all these in the **Markers topic**.

The sheet can display a value tracking section. In this section, all the stored attribute values are displayed. This can be a very handy tool to see the flow of an attribute through the cues. These values can also be edited directly in the sheet.

Most other fields in the sheet can be edited directly. This can affect the look of the show. For instance, the cue fade and delay times are stored when the cue is created. The default timing is used if nothing else is defined. The cue timing can be edited in this sheet.

Tapping this sheet to give it focus changes the encoder toolbar to make it easy to edit the cue timings. Read more about the toolbar **below.**

Title Bar

The left side of the title bar has the MA logo. Tap this to open the settings for the sheet. Read about them **below**.

Next to the logo is the sequence number and name. Information about the active world (if different than world 1) is displayed below the Sequence number and name.

On the right side of the title bar are some buttons:

- **Cue Only**:
  This button is visible when the **Track Sheet** option is On. It defines if the cue only function is On/Off when editing values in the track sheet.
- **Show Steps**:
  This On/Off button toggles displaying the cue steps in the value tracking section.
- **Track Sheet**:
  This On/Off button toggles the value tracking part of the sheet. Read more below.
- **Settings**:
  This opens the sequence settings. Here it is possible to change different settings about the sequence. Read more in the **Sequence Settings topic**.
- **Auto Scroll**:
  This On/Off button toggles the automatic scrolling function. This will scroll the sheet to keep the active cue visible in the window.
- **Link Type**:
  There are three different link types. This button gives access to choosing the desired one. The options are:
  - **Fixed**:
    The sheet displays the information from a specific sequence. The section is made in the Sequence Sheet Settings. Read more **below**.
  - **Selected**:
    The sheet displays information from the selected sequence.
  - **LastGo**:
    This automatically shows the latest sequence to receive one of the trigger commands (<<<, >>>, Go+, Go-, Goto, Load, On, Select, Top, Temp, Flash, Toggle On, Pause). This includes if the sequence is triggered from a running timecode recording. A sequence can be excluded from LastGo - read about the **Sequence Settings**. LastGo only shows sequences triggered by the same **user profile**.

Main Part of the Sheet

The main part of the sheet is below the title bar. Here is the sheet with rows and columns.

There is a sequence sheet setting that defines how the cue timing is displayed in the sheet. The option is called **Condensed Timing**. The cue fade and delay are actually four different times because there are both fade and delay for values fading in (or up) and for dimmer values fading down. The condensed view shows this in two columns where the time can be separated by a slash (/). The value on the left is the InFade. the value on the right is the OutFade. The list below shows the uncondensed timing columns. Read more in the **Cue Timing topic**.

This is a short description of each of the possible columns in the sheet:

- **Lock**:
  Changing the value to **Yes** in this cell locks the cue from being edited.
- **No**:
  This is the cue number.

- **Part**:
  This shows the cue part number.
- **Name**:
  This is the cue (part) name. If the cue contains part cues, then there is an arrow allowing to fold and unfold the parts. In the example above there are cue parts in cues 2 and 4.5. It is unfolded so the parts can be seen.
- **Release**:
  Changing the value to **Yes** in this cell makes the cue release tracked values.
- **Assert:**
  Changing the value to **Yes** in this cell makes the cue assert tracked values.
- **Allow Duplicates**:
  If several parts of the same cue are to contain values for the same attributes, use **Allow Duplicates** to enable this function. Absolute and relative values in multiple parts will use the value with the highest cue part number.
- **Trig Type**:
  There are five different trigger types. Editing this cell opens a small select pop-up with the five different options:
  - **Go**:
    The cue needs a Go command to be triggered.
  - **Time**:
    The cue is triggered a set time after the previous cue is triggered. The time is set in the **Trig Time** column.
  - **Follow**:
    A follow cue is triggered when the previous is completely done with the cue **Duration** (which includes all individual timing).
  - **Sound**:
    This will trigger the cue using a sound as the trigger. It is possible to choose one of 22 different frequency areas in the **Trig Sound** column. Learn more about sound input in the **Sound Window topic**.
  - **BPM**:
    This will trigger the cue using the beats in the sound input. This can become useful with several cues being triggered by the BPM (beats per minute).
- **Trig Time**:
  The values stored here are only used if the trigger is **Time**.
  If the trigger is **Time**, then the time in the cell will be used. The time starts counting down when the previous cue is triggered.
- **Trig Sound**:
  This setting defines the sound used to trigger the cue when the **Trig Type** is sound.
- **Tracking Distance**:
  The tracking distance sets how many cues a value should track. If the cell is empty then it tracks until changed. Read more about tracking distance in the **What is Tracking topic**.
- **Duration**:
  This is the overall cue time transition time. It is a combination of the longest fade time and any delays. This is the time used with the Follow trigger. The cell cannot be edited. It always shows the complete transition time.

- **Sync**:
  Synchronizes the fixtures of the phaser. For example, if fixtures join the already running phaser, they will be synchronized with the fixtures already running. Learn more about sync in the **Phaser topic**.
- **Morph**:
  If the property is enabled and the phase of the fixtures changes from one cue to the next, they will stay on track and morph into their new phase value. If the option is disabled, the fixtures will take the direct way to their new position of the phase.
- **CueIn Fade**:
  This is the fade time for all non-snap attributes and dimmer values that goes up in value.
- **CueIn Delay**:
  This is the delay or wait time between the trigger and the actual cue in fade begins.
- **CueOut Fade**:
  This is the fade time for dimmer values that goes down in value. The default values for this are the same as the cue in fade time - it is linked to the cue in fade with the **None** value.
- **CueOut Delay**:
  This is the delay for the **Out Fade** (only dimmer values). Its default value is the same as the cue in delay value.
- **Snap Delay**:
  This can be used to control when "snap" attributes change values.
- **Transition**:
  This can be used to select a transition path for the fade. The different options are described in the **Cue Timing topic**.
- **"Preset type" Fade**:
  Each preset type has columns called the name of the preset type followed by "Fade". This uses the cue in fade as a default. It can be changed to give all values in the preset type a different fade time. All fixtures with new values in this preset type will use this timing for the attributes in the preset type.
- **"Preset type" Delay**:
  Each preset type has columns called the name of the preset type followed by "Delay". This uses the cue in delay as a default. It can be changed to give all values in the preset type a different delay time. All fixtures with new values in this preset type will use this timing.
- **Cmd Delay**:
  This will add a delay between the triggering of the cue and the execution of the command. See CMD just below.
- **Cmd**:
  CMD is "Command". Commands (like the ones written in the command line input) can be written in the cues. They are executed on the GlobalMaster, IdleMaster, or Standalone station when the cue is triggered.
- **MIB Preference**:
  This is used to define if the cue is good for the MIB function. Read more in the **Move In Black topic**.
- **MIB Mode**:
  This sets the MIB mode. This can only be edited if MIB is possible. Read more in the **Move In Black topic**.
- **MIB Target**:
  Defines a cue where the MIB should be performed if possible. This can only be edited if MIB is possible. Read more in the **Move In Black topic**.

- **MIB MultiStep**:
  Defines what should happen with phasers in the MIB. This can only be edited if MIB is possible. Read more in the **Move In Black topic**.
- **MIB Fade**:
  Sets the MIb fade time. This can only be edited if MIB is possible. Read more in the **Move In Black topic**.
- **MIB Delay**:
  Sets the MIB delay time. This can only be edited if MIB is possible. Read more in the **Move In Black topic**.
- **Indiv Fade**:
  This is "Individual Fade". It displays the time for attributes that have individual stored fade times. This cell cannot be edited.
- **Indiv Delay**:
  This is "Individual Delay". It displays the time for attributes that have individual stored delay times. This cell cannot be edited.
- **Indiv Duration**:
  This is "Individual Duration". It displays the overall time for attributes that have individual stored fade and delay times. This cell cannot be edited.
- **Speed Master**:
  A Speed Master can be assigned to the cue or cue part by editing this cell. The speed of the cue or cue part is only controlled by the assigned speed master. A speed master assigned to the sequence has a lower priority and does not influence a cue or cue part with a different assigned speed master.
- **Speed Scale**:
  This can be used to scale the speed of the cue or cue part. A speed scale assigned to a cue or cue part has a higher priority that a speed scale assigned to the sequence.
- **Appearance**:
  An Appearance can be assigned to the cue or cue part. It is connected with a sequence sheet setting called **CuePart Appearance**, which defines how the appearance is displayed. Read more **below**.

All cells with a light or dark gray background color can be edited and the value in the field can be changed. Fields with a black background cannot be edited.

Track Sheet Mode

The sequence sheet can be in **Track Sheet** mode. This can be changed in the title bar and in the settings.

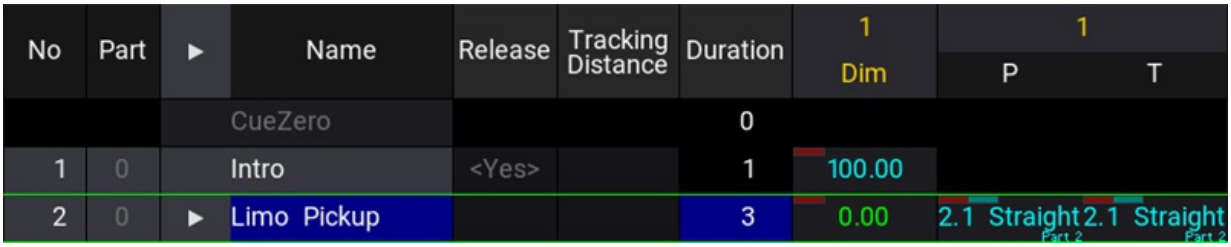
Sequence Sheet in Track Sheet mode

The default for this mode is to hide most of the normal columns and show attribute data in columns instead.

The attribute values and markers have different colors indicating different statuses like the tracking status. Read about the colors and markers in the **Colors topics**. Read about tracking in the **What is Tracking topic**.

The values can be edited in the sheet. Cue Only can be activated in the title bar (an extra button that appears when **Track Sheet** is On). This makes edited values follow the **cue only rules**.

When cues with multiple parts are expanded to show all the parts, then it is easy to see exactly what parts have stored which values. When the cues are collapsed to only show one row, then the values from the parts are shown in this one row with a small text telling what part the values come from.



Collapsed cue with multiple parts

This example is the same cue 2 as the image above. The only difference is that the cue is collapsed to only show one row. Notice the small text showing that the pan and tilt values are from part 2.

The **Layer Toolbar** can be turned On in the settings. This can be really useful when there is a desire to edit or look at values in other layers.

Editing a value opens the Calculator, where a new value can be selected from Presets, Channel Sets, Specials, or simply typed.

The **calculator** has some special functions in the track sheet that defines what is changed and there are special buttons that give access to block, unblock, and extract presets.

Extract presets will remove the link to a preset and store the current preset values directly in the cue.

Sequence Sheet Settings

The sheets have a lot of settings. They are accessed by tapping the MA logo in the upper left corner of the window.

Sequence Sheet Settings - Display tab

Some of them are general settings that are shared with other windows. Read about them in the **Window Settings topic**.

There are two tabs in the settings: **Display** and **Mask**.

This is a description of the display settings that are unique to the sequence sheet.

- **Auto Scroll:**
  This On/Off button activates the auto-scrolling function. This is the same setting as the button in the title bar. Read more in the description above.

- **Condensed Timing**:
  This toggles if the timing columns are displayed condensed or of all four timing columns are visible. Read about the main part of the sheet above.
- **Countdown**:
  A cue timing countdown can be displayed while the fade is running. This setting has three options:
  - **Off**:
    There is no countdown in any of the timing columns. They always display the set times.
  - **Duration**:
    The duration column displays a countdown while the cue transition is running.
  - **All**:
    The duration and cue timing columns display a countdown while the cue transition is running.
- **Cue Part Appearance**:
  This defines how the cue part appearance is displayed in the sheet. The options are:
  - **Off**:
    Cue part appearance is not displayed.
  - **Number**:
    The appearance is only shown on the cue number column.
  - **Num+Name**:
    The appearance is displayed in the number and name columns.
  - **All**:
    The appearance is displayed on all columns.
- **Selection Only**:
  This On/Off button is a mask function that hides fixtures not currently selected in the programmer. This is valid when the **Track Sheet** mode is On.

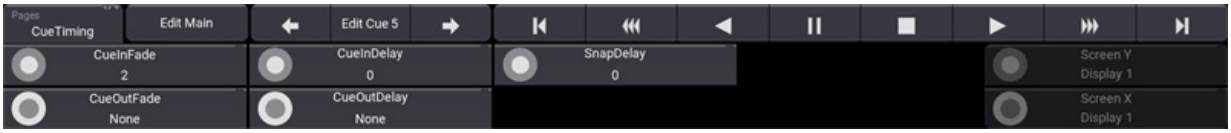
The mask settings unique for the sequence sheets are:

- **Cmd:**
  This On/Off button shows or hides the group of Cmd (command) columns.
- **Cue Settings**:
  This On/Off button shows or hides the cue settings columns.
- **Cue Timing**:
  This On/Off button shows or hides the cue timing columns.
- **Filter**:
  An existing filter can be chosen to filter the content of the sheet.
- **Filter Toolbar**:
  This setting show or hide a filter toolbar that can be used to filter the content of the sheet.
- **MIB Settings**:
  This On/Off button shows or hides the MIB columns.
- **Preset Timing:**
  This On/Off button shows or hides the preset timing columns.

- **Show Recipes**:
  This On/Off button shows or hides the cue recipes at the bottom of the Sequence sheet. Read more about it in the **Cue Recipes topic**.
- **Show Steps**:
  This On/Off button shows or hides the cue steps in the Sequence sheet. This is useful with **Track Sheet** On.
- **Track Sheet:**
  This On/Off button shows or hides the values and tracking information for each attribute in the sheet.

Sequence Edit Toolbar

The encoder toolbar changes when the sequence sheet has focus. The sheet can get focus by tapping the sheet.



Sequence Edit Toolbar - Basic Timing page

There are several pages with many of the different settings for the cues. The pages can be changes using the swipe button in the upper left corner in the toolbar (see the image above).

The top row in the toolbar gives access to select a cue. There are also playback controls that can be used to run cues. Read more in the **Play Back Cues topic**.

Turning the two rings on the encoders changes the respective values for the selected cue in the sheet.

The lower row is the outer ring of the dual encoder. The middle row is the inner ring of the dual encoders.

22.3. Content Sheet

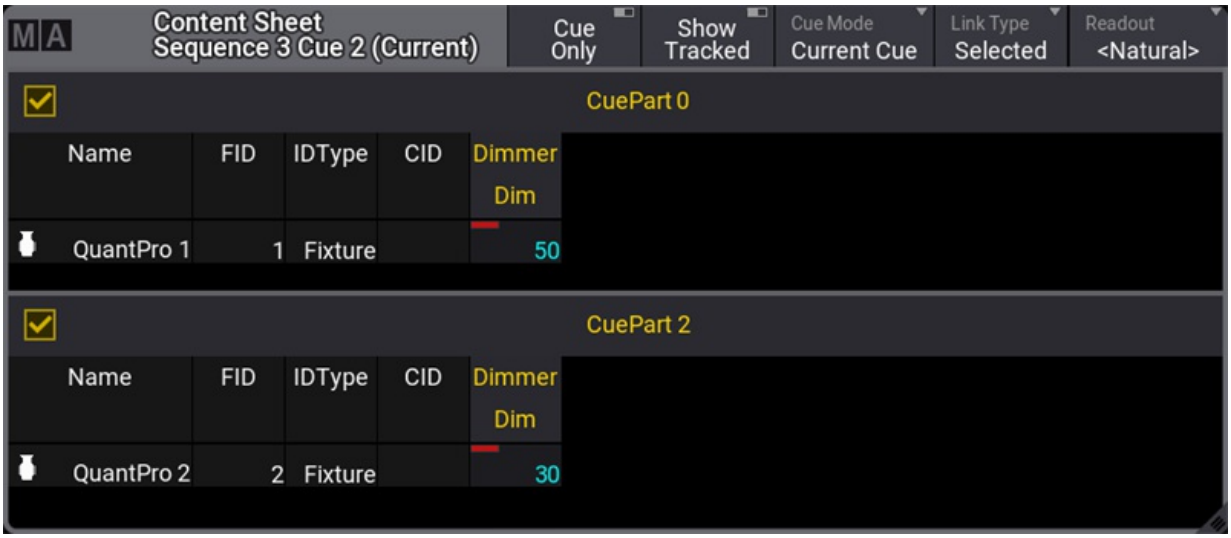
The Content Sheet is used to see the fixtures and values stored in cues. It looks a lot like the **Fixture Sheet**, but it has a masking function that only displays what is stored in a cue part. It is like a combined Sequence Sheet in tracking mode and a Fixture Sheet.

It could look like this:

Content Sheet

The Title Bar shows the sequence ID and the cue ID of the cue being shown. If a world other than the default Full world is selected, then this is also displayed in the title bar (Small World icon with a name and number next to it).

The parts of the cue are shown in a frame for each part. This frame can be toggled On or Off by tapping the title bar of the part frame part. The checkmark in the part frame title bar indicates the show/hide status. A cue with multiple parts can look like this:



Content Sheet with multiple parts

The values displayed in the sheet can be edited directly in the sheet, just like in the Sequence Sheet in **Track sheet mode**.

The title bar has multiple buttons. These are settings that can also be changed in the window settings. They can be opened by tapping the MA logo in the upper left corner of the window.

Some settings are shared with other windows and the description of these settings can be found in the **Windows Settings topic**.

Turning On Cue Only will make the editions using the cue only option. Learn more about Cue Only in the **Store Cues** topic.

The Link Type defines what sequence the sheet is showing. It is possible to select a specific sequence. Read about the Link Type and Fixed Target setting in the Window Settings topic (link above).

The following settings are unique for the content sheet:

Cue Mode

There are three different cue modes options for the sheet:

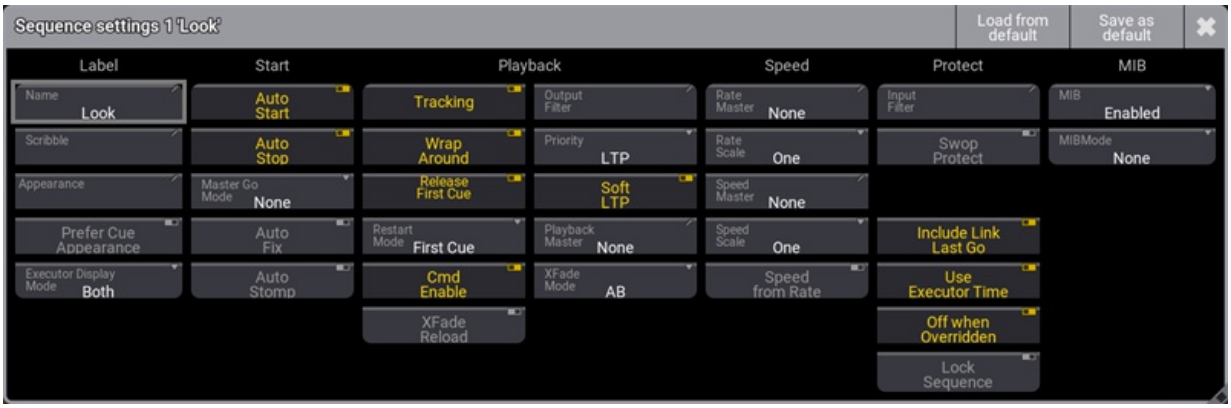
- **Current Cue:**
  This will make the sheet display the information related to the current active cue.
- **Previous Cue:**
  This will display the values from the previous cue. This is the last cue that was active even if the sequence is jumped.
- **Next Cue:**
  This displays the values for the next cue if a Go+ is performed on the sequence. If a cue is "Loaded" then this cue will be displayed.

Show Tracked

This setting is a toggle. Turning it On shows all the values tracked from previous cues. Turning it Off hides the tracked values and only shows the values that are stored in the cue.

22.4. Sequence Settings

Each sequence has a lot of different settings. The settings can be accessed from the top bar of each **sequence sheet** or from the **executor assign menu**. Each of these two locations has a button called Settings. Tap this button to open the settings.



Sequence settings pop-up

The settings are divided into different sections. Label, Start, Playback, Speed, Protect, and MIB. Read about the different settings in each section below.

Each user profile has default sequence settings. These settings are used when a new sequence is created. The default can be loaded and stored from this pop-up.

The title bar has two buttons that are relevant. The left one is Load from default. Tapping this loads the stored sequence defaults. Next to this is Save as default. Tapping this stored the current settings as the new default.

Label

This group is about the name and look of the sequence.

Name

Edit this input field to change the name of the sequence.

Scribble

This scribble field can be used to select or create a scribble for the sequence. This scribble is visible in the sequence pool and on the executor label.

Appearance

This appearance field can be used to select or create an appearance for the sequence. This appearance is visible in the sequence pool and on the executor label.

Prefer Cue Appearance

When this option is enabled, and the current cue has an appearance, the cue appearance will be displayed on the executor or in the layout, instead of the sequence appearance.

Executor Display Mode

The executor display mode defines how the sequence will be displayed on an executor:

- **Data only**:
  Only the cues with their appearances will be displayed. The cue appearance is only displayed in the line of the cue, and not in the background.
- **Appearance only**:
  Only the sequence or cue appearance will be displayed. No cue names, fade bar, etc. will be displayed. This can be handy for sequences with only one cue, or for sequences with only color cues or gobo cues.
- **Both**:
  Each cue line displays its cue appearance, and the sequence appearance or the appearance of the current cue will be displayed in the background of the executor.

Start

The settings in this section are about starting and stopping the sequence.

Auto Start

The **Auto Start** feature switches the executor 'On' when the master is moved from zero.

Auto Stop

The **Auto Stop** feature switches the executor 'Off' when the master is moved down to zero.

Master Go Mode

The **Master Go** function is active if auto stop is turned Off. Tapping this button opens the **Select Master Go Mode** pop-up.

There are four options here. They all take effect when the master fader is moved from zero and up. The options are:

- **None**:
  The cue is still running.
- **Go**:
  It executes a Go.
- **On**:
  The current cue is reloaded (fading in again).
- **Top**:
  The first cue is activated.

Auto Fix

The **Auto Fix** feature can be activated for each executor. It will automatically **Fix** active executors and keep them visible even when pages are changed. The executor is automatically unfixed when it is switched off.

Auto Stomp

If auto stomp is On, then an **absolute** value from a cue will stomp a phaser running from a different playback.

Playback

The playback settings are about running the cues. Running or playing back cues is described in the **Play Back Cues topic**.

Tracking

This turns On or Off value tracking in the sequence. Read more in the **What is Tracking topic**.

Wrap Around

Wrap around allows the sequence to return to the top/first cue if a Go (forward) command is performed after the last cue in the sequence is reached.

Release First Cue

This setting defines if the first cue releases tracking values. These tracking values can come from the last cue if **Wrap Around** is active. If **Release First Cue** is On then it adds a **<Yes>** to the **Release** column in the first cue of the sequence. Learn more about the different columns in the **Look at Cue and Sequences topic**. The first cue can manually be set to release by editing the field in the sequence.

Restart Mode

There are three different restart modes:

- **First Cue**:
  This always restarts the sequence with the first cue.
- **Current Cue**:
  This restarts the sequence with the cue where it was when the sequence was turned Off.
- **Next Cue**:
  This restarts the sequence with the next cue based on where it was when the executor was turned Off.

Cmd Enable

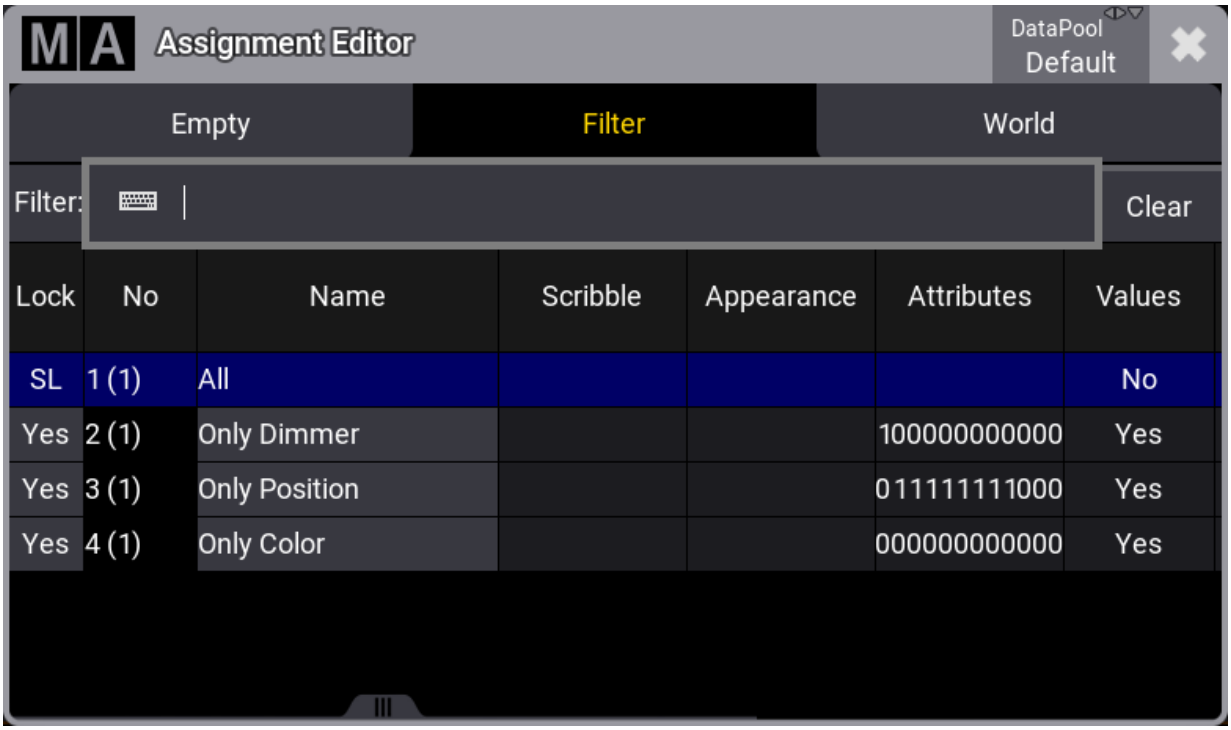
This option disables the execution of the commands in the CMD columns in a sequence. The stored commands are not deleted or removed, they are just disabled.

XFade Reload

When this option is enabled, the Xfader needs to be pulled back to 0 after completing a crossfade in order to do the next crossfade.

Output Filter

The output filter can have a **Filter or World** assigned. Tapping the button opens the **Assignment Editor** pop-up:



Tap a world or filter in the list to apply an output filter

The editor has three tabs. **Empty** can be used to select no filtering. **Filter** and **World** are lists of each type. Each lists the possible choices of their type. Tap the desired filter or world to apply it.

The fixtures and attributes in the world or filter are allowed to pass the output filter and can be output from the sequence. The same sequence can be played back from several executors and each executor share the output filter settings. If different output filters are needed, then the sequence should be linked or shared with another sequence. The other sequence can have different output filter settings. Read about linked sequences in the **Cues and Sequences topic**.

If the sequence has an output filter applied, then there is a small output filter icon ( ) in the sequence pool button.

Priority

This is the priority of the sequence. The priorities are described in the **Play Back Cue topic**.

Soft LTP

The Soft LTP function is described in the **Play Back Cue topic**.

Playback Master

Here it is possible to select a **Playback Master**. It functions as a sub-master that can be shared by multiple sequences.

XFade Mode

This is used to set how the two CrossfadeA/XFadeA and CrossfadeB/XFadeB faders work. There are two crossfade modes:

- **Split**:
  The dual crossfaders work as masters for the current/next cue.
- **AB**:
  The dual crossfaders work as crossfaders for increasing/decreasing values.

Speed

Speed and rate can be used to adjust the stored times without having to reprogram the show. For instance, a cue can have a fade time of 5 seconds. A rate master can then adjust the fade time live while playing it back. All sequences have their own individual rate and speed, but they can be linked to a global master. This master can then adjust the timing for multiple sequences at the same time. Read more details in the **Speed Masters topic**.

Rate Master

The sequence has a rate master. It can be linked to a shared global speed master or it can have an individual rate master. Tapping this button will open the **Select Rate Master pop-up**.

In the pop-up it is possible to select the **None** option for having an individual rate master for the sequence or select one of the global speed masters.

Rate Scale

Enabling this binds the rate to defined steps instead of a variable value. Tapping this button opens the **Select Rate Scale pop-up**.

Choosing one of the steps in the pop-up selects the multiplier or divider. This multiplies or divides the rate by the selected factor.

Speed Master

The sequence has a speed master. It can be linked to a shared global speed master or it can have an individual speed master. Tapping this button will open the **Select Speed Master pop-up**.

In this pop-up, it is possible to select the **None** option for having an individual speed master for the sequence or select one of the global speed masters.

Speed Scale

If a sequence is assigned to a global speed master (read above) then it can be useful to adjust a speed scale. Tapping this button will open the **Select Speed Scale pop-up**.

In the pop-up, it is possible to select one of the multipliers or dividers. This multiplies or divides the speed by the selected factor.

Speed from Rate

This links the speed to follow the rate.

Protect

This group of settings is used to protect the sequence from different actions.

Input Filter

The playback filter is described in a little bit more detail in the **Cues and Sequences topic**. Tapping this button opens the **Assignment Editor pop-up**: See above about the output filter

The fixtures and attributes in the world or filter are allowed to pass the filter and can be stored in the sequence.

Swop Protect

Activating this option protects this sequence from the Swop function.

Include Link Last Go

This setting is On as default. When it is set to Off, playing back a sequence will not trigger the LinkLastGo functionality in the **sequence sheet**.

Use Executor Time

This makes the executor playback cues using the stored timing. If this is turned on then it is affected by the **Exec Time** master fader, who overwrites the timing.

Off when Overridden

The Off when Overridden function allows that a sequence is automatically turned Off if another sequence has taken control with all the attributes in the sequence = this executor does not control any attributes.

Lock Sequence

The sequence is locked against changes when this is On. It can still be played back.

MIB

This group is about MIB settings for the sequence. Read more about MIB in the **Move In Black topic**.

MIB

Enable, disable, or force MIB for the sequence. The options are:

- **Enabled**:
  MIB will be performed according to the cue and cue part MIB settings.
- **Never**:
  MIB will never be performed for this sequence. All cue and cue part-specific MIB settings will be ignored.
- **Force Early**:
  Forces the early MIB for all cues that can perform MIB as soon as the dimmer is closed. Further MIB settings specified per cue or cue part will be ignored.
- **Force UponGo**:
  MIB is forced to be executed with the next cue transition after the dimmer is closed. For all cues that can perform MIB. Further MIB settings specified per cue or cue part will be ignored.
- **Force Late**:
  Forces the MIB latest in the cue before the dimmer opens again. For all cues that can perform MIB. Further MIB settings specified per cue or cue part will be ignored.

MIB Mode

This setting defines which MIB mode will be executed when doing a MIB. The **MIB Mode** per cue or cue part needs to be set to **Default**. The options are:

- **None**
- **Early**
- **UponGo**
- **Late**

Learn about the mode types in the MIB topic (link above).

22.5. Store Cues

Storing a cue is the default **Store** action in grandMA3.

This means that if nothing else is defined and Please is pressed right after Store, then a new cue is stored in the selected sequence.

Or press Store, followed by pressing the executor button where the sequence, with the cue, should be stored.

If there are active values in the programmer, then they are stored into the cue, but programmer values are not needed to store cues.

Of course, there are more details about storing cues, so please keep reading.

Store a New Cue on an Empty Executor

If a cue is stored on an empty executor, then the grandMA3 software automatically stores the cue in a new sequence and assigns this executor to control the sequence using the default settings.

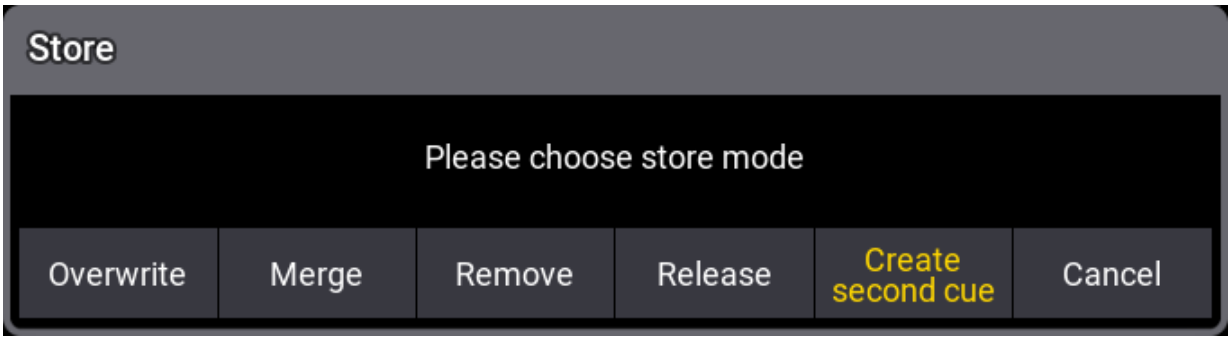
Going back to the second example at the top of this topic: just pressing the Store button and then an executor button on the empty executor is enough. The software assumes the desired action is storing a cue.

This will automatically be cue number 1 - nothing else was specified.

Storing cues obey **Worlds and Filters**. This enables control of what is stored. Worlds and filters can also be assigned to the sequence both as an input but also as an output filter - independently of each other. This will function as an input or output filter allowing only the elements in the world or filter to be stored in or played back from the sequence.

Store the Second Cue

If the store function is used again on the same sequence - without adding cue number details - then the grandMA3 does not know what should happen and a pop-up appears, giving different choices.



Please choose store mode pop-up with "Create second cue" option

Tapping Create Second Cue will store a cue with the next whole number.

Overwrite and **Merge** options are explained below. **Remove** and **Release** are described in the **What is Tracking topic**. **Cancel** does not store anything.

Cue Numbers

When a cue is stored, it is possible to specify a cue number. This is done by using the following syntax: **Store Cue [Cue_Number]**. It is also possible to specify a sequence or an executor, using the keys in the command section, while storing:
Store Cue [Cue_Number] Sequence [Sequence_Number] or
Store Cue [Cue_Number] Executor [Executor_Number].

Notice that the cues are stored in the sequence. Using the executor number will store the cue into the sequence the executor is controlling.

Cue numbers have three decimal numbers. If all are zero then they are not displayed. But cue number "42" is the same as cue number "42.000" - it is not "42 thousand", it is "42 point 0 0 0". The currently highest cue number that can be stored is "999 999.999". The lowest number that can be stored is "0.001".

Restriction:
Storing nearly 1 billion cues will completely fill the memory and make the show file VERY big.
The software will cancel the store process before the system crashes, but almost any operation after this will make the software shut down!!

The software hides the trailing decimal zeros, but they are still there. This means that cue "5.2" is after cue "5.11" because they are actually cue "5.200" and "5.110".

It is not limited to only store a single cue number at a time. It can just as easily be a range of numbers - this means that it is possible to use Thru, +, and - buttons to create number ranges to be stored.

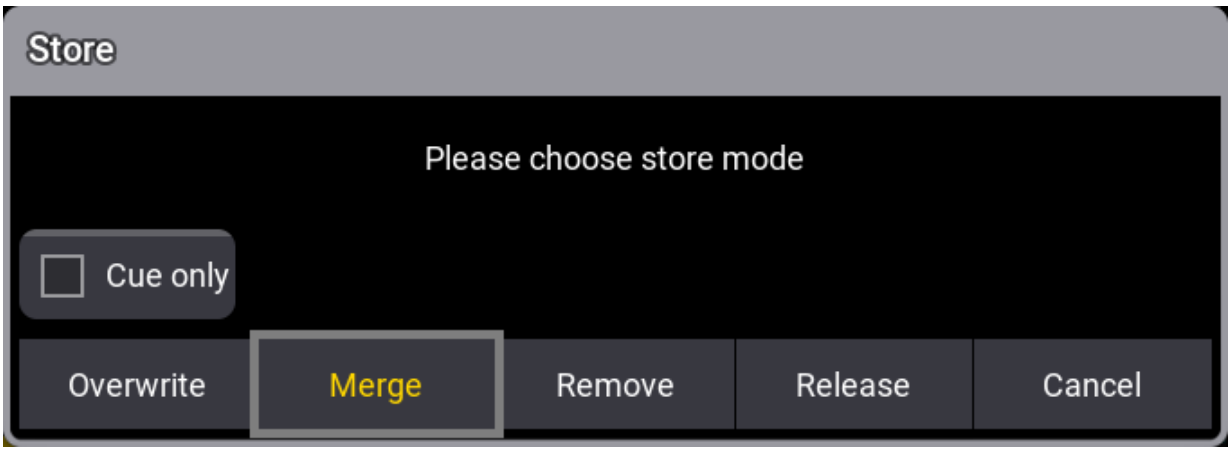
If you are going to be working with or adding a lot of cues to the same sequence, then it can be a good idea to select the sequence.

This can be done easily by pressing Select and then one of the buttons associated with the executor controlling the sequence or tap the sequence in the sequence pool.

If no sequence or executor is defined in the store command, then the selected sequence is used.

Store Into Cues that are Not Empty

If the store operation is used to store into already existing cues then a pop-up like this appears:



Please choose store mode pop-up

It only appears if the store options do not specify what should happen.

Have a look at the **Store Options and Defaults topic** for information about specifying this while storing.

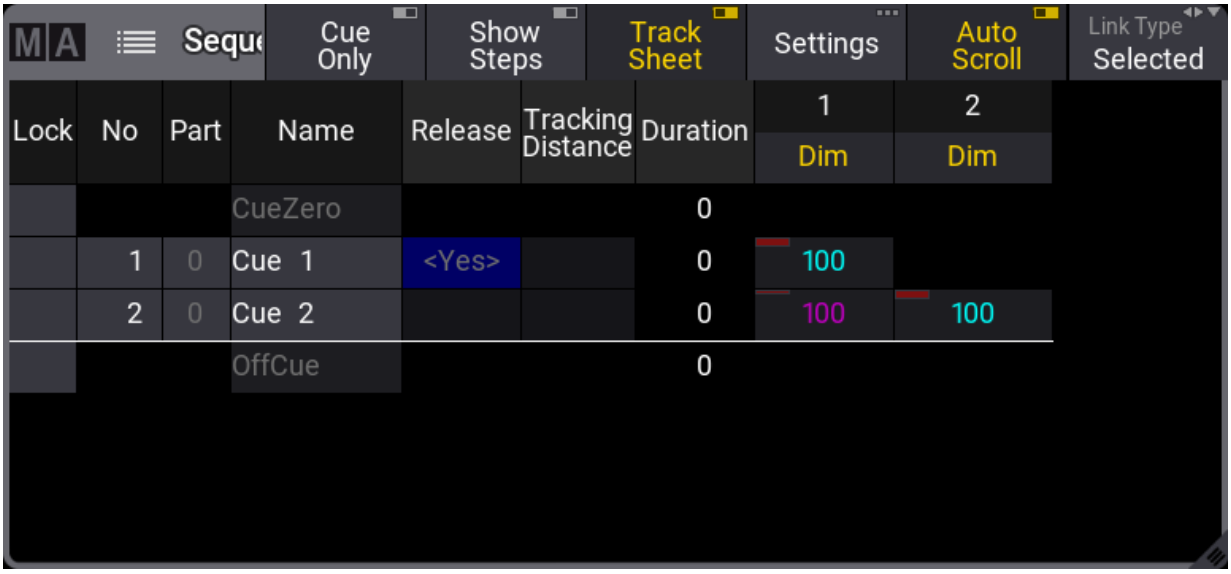
Remove and **Release** are described in detail in the **What is Tracking topic**. **Cue Only** is described **below**.

The two relevant options are:

- **Overwrite:**
  This will remove what is already stored in the cue and only store the new values.
- **Merge:**
  This will merge the new values into the existing values. New values have a higher priority and will overwrite existing values.

Examples

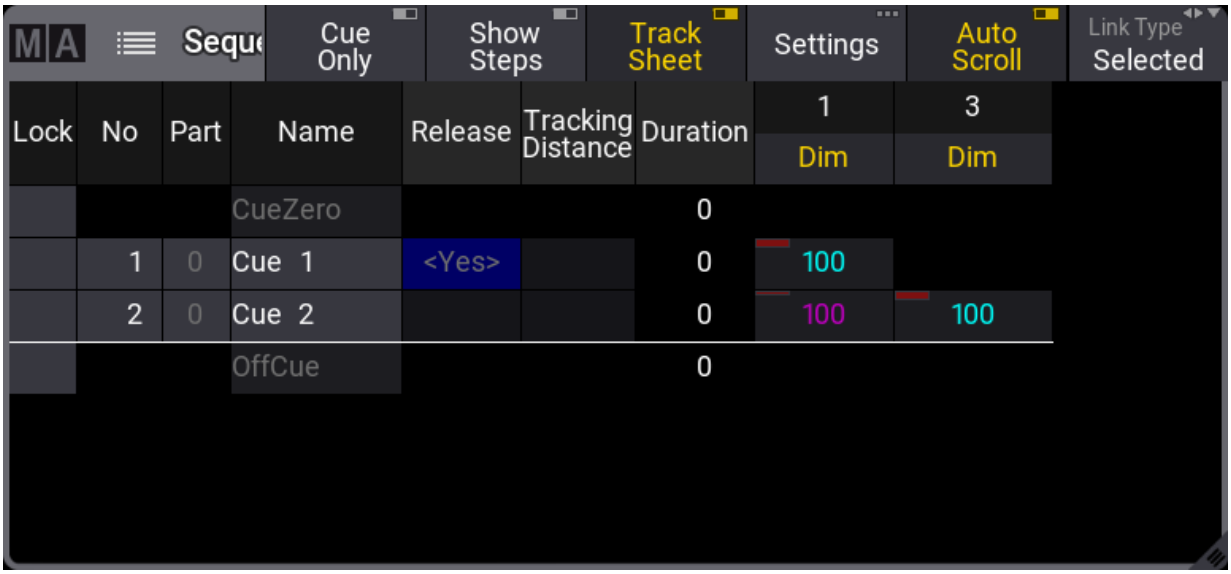
In the following examples we have a sequence with the following two cues:



The magenta value for fixture 1 in cue 2 is a tracked value.

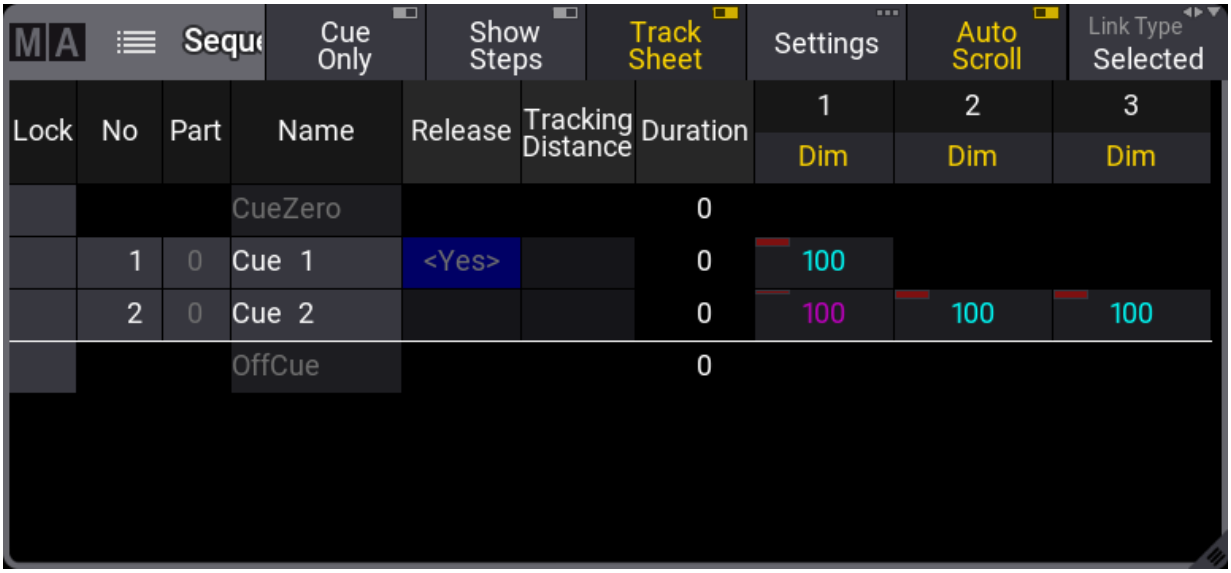
Now we turn on fixture 3 at 100 % and store this into cue 2.

This is the result if **Overwrite** is chosen:



Now fixture 2 is gone. This is because it only had values stored in cue number 2. The dimmer value of fixture 1 is not affected because it is a tracked value.

If we had chosen **Merge** instead it would have looked like this:



Now the value from fixture 3 is added to the existing values.

Store Cues with Timings

When storing a cue it is possible to also store the different cue timings. This is described in detail the **Cue Timings topic**, but here is the short version.

The Time button will add different timing keywords to the command when storing.

For instance, storing cue 4 with a fade time of 6 seconds and a delay of 1 second, the following keys can be pressed:

Store Cue 4 Time 6 Time 1 Please

This is the result in the command line feedback:

```
OK : Store Cue 4 CueFade "6" CueDelay "1"
```

Pressing the Time button repeatedly will change what timing keyword it adds.

Adding and Using Cue Labels

A cue can be given a name - using the **label keyword** - while it is stored. This is the syntax: **Store Cue [Cue_Number] "My Cue Name"**.

The keyboard is needed for writing this. The quotation marks are needed to tell the software that this is text - then it is not interpreted as a command.

Labels can also automatically be numerated while storing. Have a look at this command:

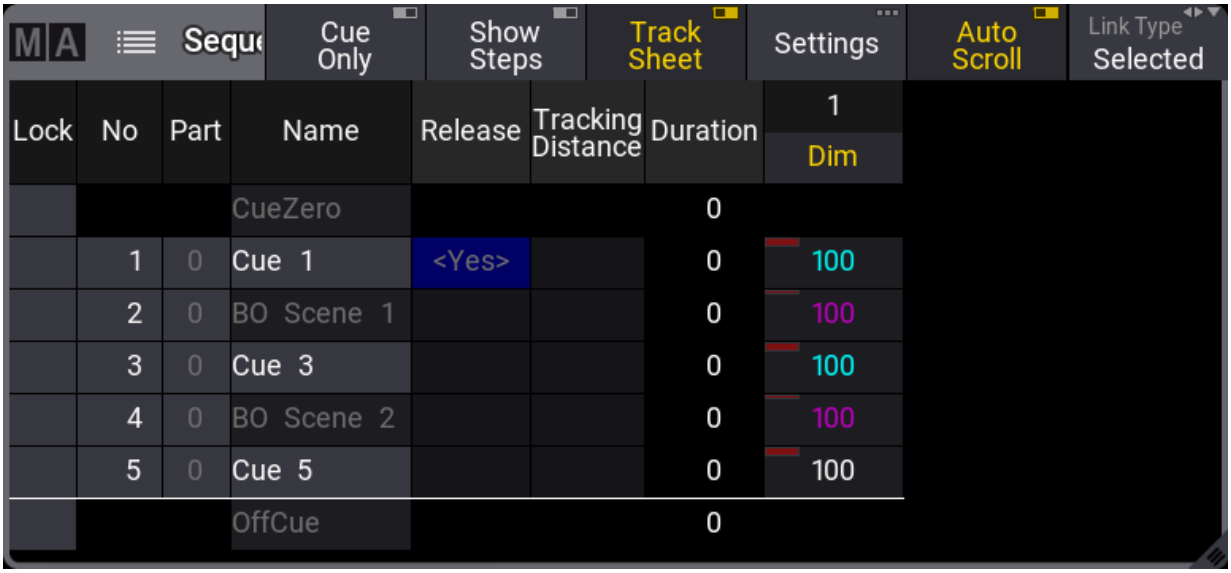
```
User name[Fixture]> Store Cue 2 + 4 "BO Scene 1"
```

This will not label both cues 2 and 4 the same - it will add 1 to the number for each cue. The result is that cue 2 is called "BO Scene 1" and cue 4 is "BO Scene 2". This enumeration only works if the number is the last part of the label and if there is a space between the last word and number.

Cue labels can be used when storing. This means that if there are several cues whose labels start with "BO", then it is possible to store into all these cues in one operation using BO plus an asterisk. See the following example.

Example

This is the cue sequence and content before storing:

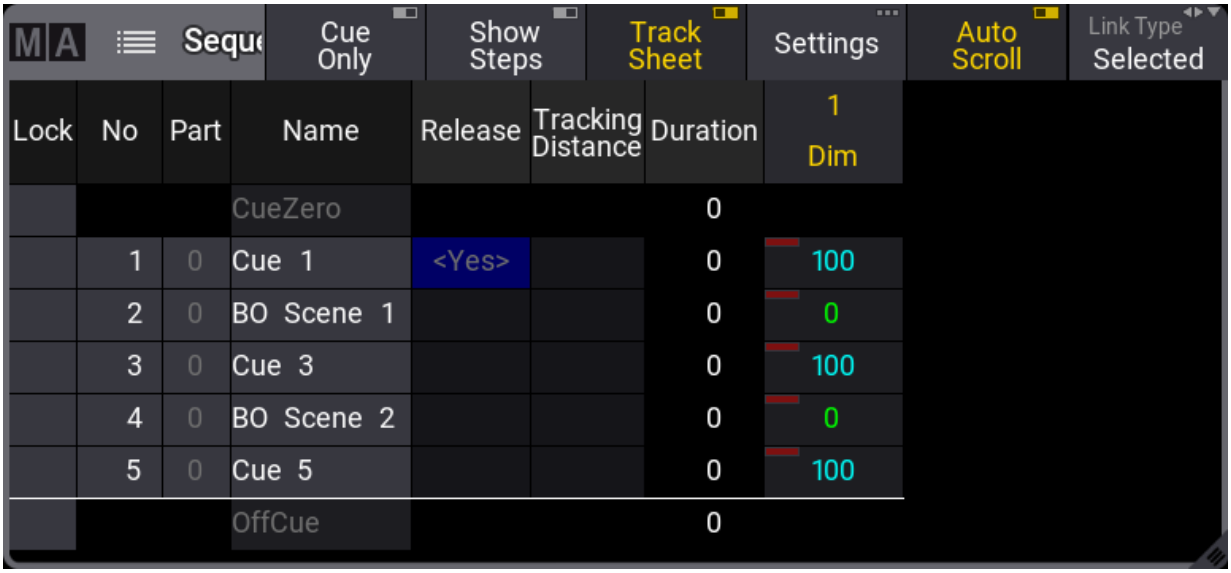


Notice that cues 3 and 5 are **blocked**. This means that 100% is stored in the cues even though it is currently not necessary.

With an active value of 0% for fixture 1, use the keyboard to type the following command:

```
User name[Fixture]> Store Cue "BO*" /Merge
```

This is the result:



The two cues whose label begins with BO now got the new value.

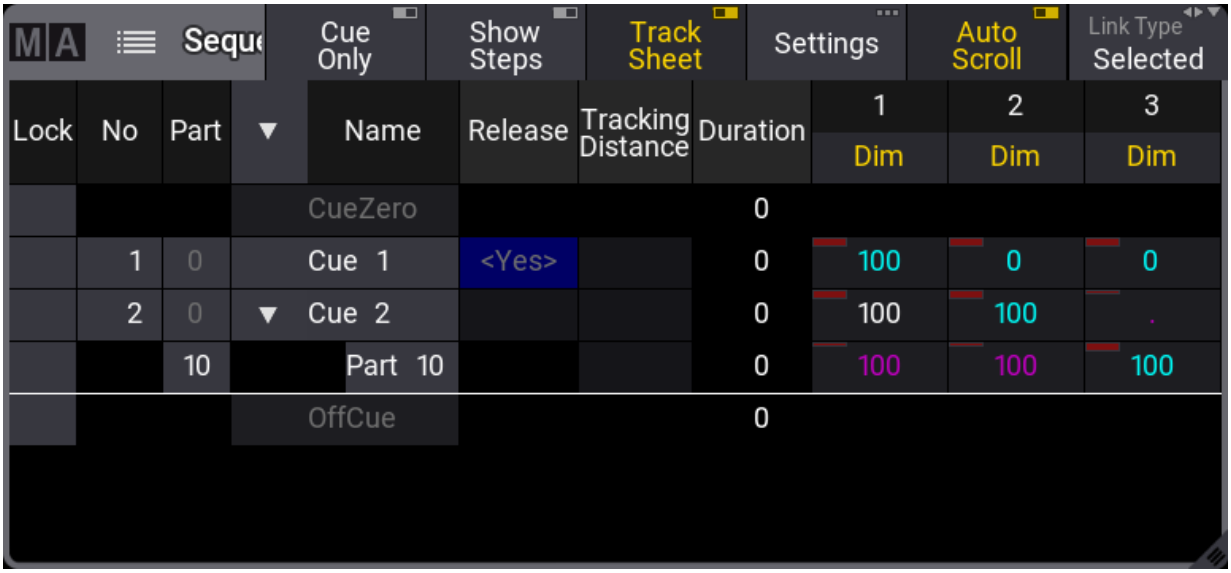
Store Cue Parts

Besides the many cues that could be stored in each sequence, it is also possible to store 256 **Cue Parts** to each cue.

Values are always stored in a cue part and often when we talk about a cue, we are actually talking about all the parts in a cue.

Cue parts are a division of the cue. Cue part 0 is always created with the cue and this part has all the values unless a different part is created or specified when storing.

Have a look at this example:



Notice that the value for fixture 3 is stored in cue 2 part 10. It does not have a value in cue 2 part 0. And the values from fixtures 1 and 2 are in part 0 and therefore do not have any value in cue 2 part 10.

Storing something in a part is almost as easy as storing the main cue. Using the example above the keypresses would be:

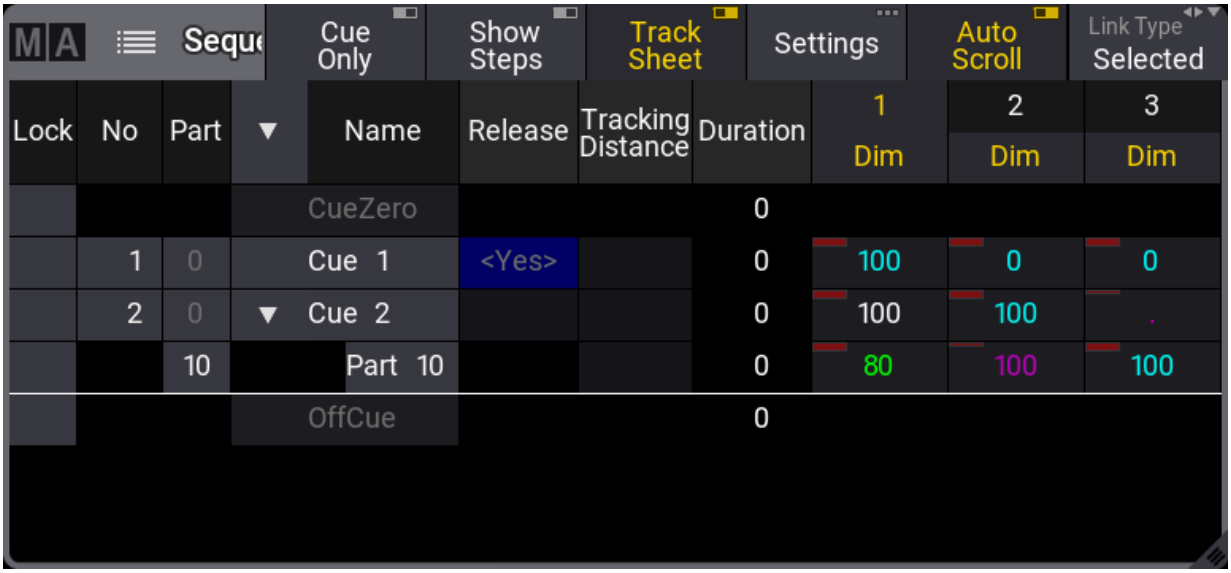
Store Cue 2 Cue 1 0 Please

The second press on the cue key will result in the **Part keyword** and the command line feedback looks like this:

```
OK: Store Cue 2 Part 10
```

As a default, it is only possible for an attribute to be stored in one cue part per cue. But this can be changed so an attribute can have values in multiple cue parts in the same cue. If **Allow Duplicates** is turned On for a cue then attributes can be stored in all parts of the cue.

In this example **Allow Duplicates** is set to Yes for cue 2 and fixture 1 was values in both part 0 and part 10:



The cue part with the highest number takes precedence and in the example above fixture 1 will end at 80% output when cue 2 is triggered.

Allow Duplicate is a column in the **Sequence Sheet**.

Cue Only

Cue Only is a **store option** for cues. When storing into a cue with the Cue Only, the tracked values will be blocked in the next cue or cue part to preserve the previous look on stage. The actual values of the programmer will only be stored in the target cue or the cue part.

Cue only can be used when storing into existing cues - except the last cue. There is no hard key for cue only so it needs to be activated otherwise. These are the following options:

- As command line option /CueOnly or /CO after the normal **Store keyword**.
- In the store options: To open the store options, press and hold Store for at least 2 seconds. Then turn On cue only by tapping Cue Only.
- In the **Store Cue pop-up**. When storing onto an existing cue, the Store Cue pop-up will appear and ask whether to Overwrite, Merge, Remove, Release, or Cancel the current store operation. It is also possible to decide if the cue is to be stored with active cue only within the pop-up. Remove and Release can be used for Cue Only. The pop-up only appears if the desired cue is not the last one in the sequence.

When storing cue only, the grandMA3 software decides, on the basis of the 3 following rules, in which cue part it will block the original values:

1. The default cue part is cue part 0.

2. If the next cue already contains a part that has the same name as the cue part the original value is coming from, the blocked value will be used in this part.
3. If the cue already contains attributes of the same feature group, the previous values will be blocked in this cue.

If none of the rules turn out to be true for phaser values, a new cue part will be created where the previous values will be blocked.

Using Command Line Input to Add More While Storing

In the example using the cue labels to store, there was a command that showed some of the other possibilities while storing cues.

The command line gives access to all the store options that can be found in the GUI Store Options - read about them in the **Store Options and Defaults topic**.

All the different elements are described in the **Store Keyword topic**.

Examples

The following are just a few extra command line examples showing some of the possibilities while storing.

```
User name[Fixture]> Store Cue 1.2 Sequence 4
```

Stores cue number 1.2 in sequence 4.

It does not matter if you write sequence or cue first. So this could also have been:

```
User name[Fixture]> Store Sequence 4 Cue 1.2
```

The commands can often be written shorter in the command line input. See some examples in the **General Syntax Rules topic**.

Read the topics about each keyword to see the short version for the keyword.

```
User name[Fixture]> Store Cue 42 "Al Powell arrives at the Plaza" CueFade 6/3
/merge
```

This will store the cue with a name, it is merged, and it is also stored with an in-fade of six seconds and an out-fade of three seconds. Read more about the store options in the **Store options and defaults topic**.

Store Remove

A version of storing is the **Store Remove**, where Remove is selected in the store pop-up (described above).

This will remove the stored values for the attributes that currently have active values in the programmer.

The actual values in the programmer are irrelevant in this case. They are simply an indicator of what attributes should be removed from the cue.

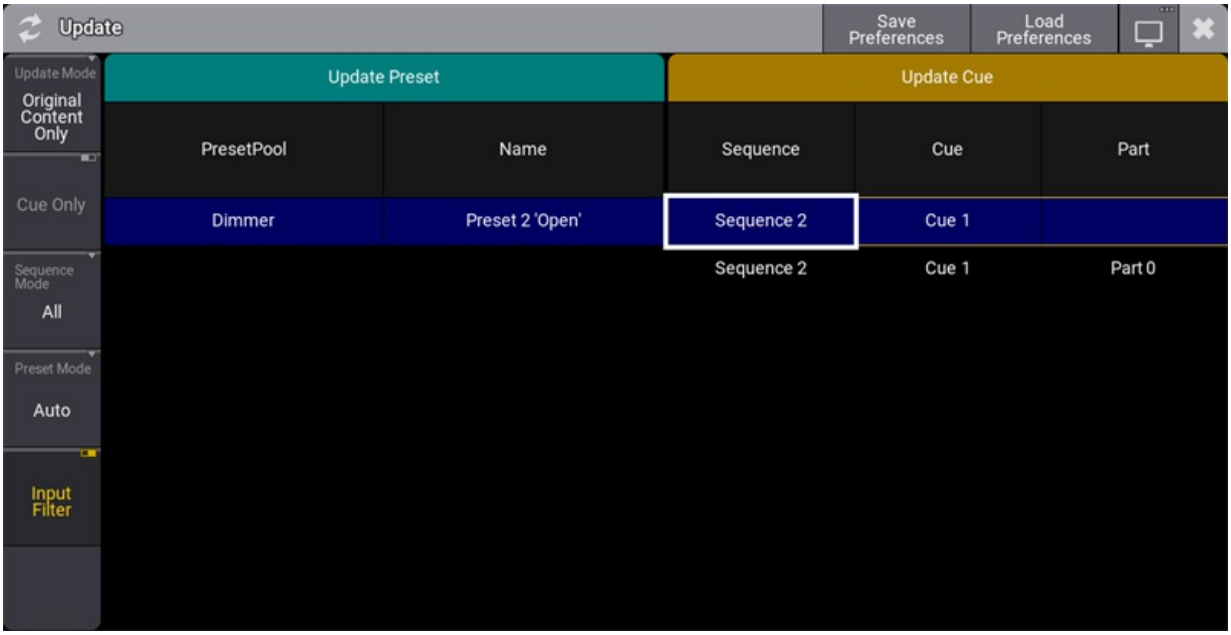
22.6. Update Cues

The stored values in a cue can always be changed by storing new values in them.

But if there are active cues and active values in the programmer, then it is also possible to update the cue or cue part.

The **Update key** flashes when this scenario is valid.

Pressing Update opens the **Update Menu**.



The Update Menu with possible update locations

The right side of this menu lists the possible cue (parts).

To update the cue part with the current settings, just tap the desired cue part on the list. This immediately updates the stored values in the selected location.

There are two options that are relevant to the update process. **Update Mode** and **Cue Only**.

Update Mode:

The mode toggles between two options. **Original Content Only** updates only the already existing values at their original location - possibly a previous cue. **Add New Content** updates the active cue with all the programmer values (filter and worlds are still respected), possibly adding not only new values but also adding new content to the cue.

Cue Only:

Cue only can be toggled On or Off. If it is Off, then the values are updated using normal tracking behavior. If it is On, then the updated values are updated as cue only changes. Read more about cue only in the **Store Cues topic**.

Sequence Mode:

The sequence mode is a filter that can be used to filter the list of possible cue parts. The options are:

- **All**:
  This shows all possible cue parts from all active sequences.
- **Selected**:
  This only shows the cue part from the selected sequence.
- **Last Called**:
  This shows the cue part from the last called sequence.

Preset Mode:

This does not have anything to do with updating the cue parts. But it is also possible to update presets if there are called presets in the cue parts and the active values are relevant for the presets. Valid presets are listed on the left side of the update menu.

This setting sets what mode will be used when updating the preset. The options are **Selective**, **Global**, **Universal**, and **Default**.

Learn more about these options and presets in the **Preset section**.

Input Filter:

This setting is used when updating presets. It can be toggled to turn On or Off the input filters when updating preset values. Learn more about these options in the **Preset Section**.

22.7. Copy Cues

Copy to an Empty Destination

Cues can be copied to a new cue.

The **Copy keyword** is needed for this operation. The general syntax is **Copy Cue [Source_cue_ID] At [New_destination_cue_id]**.

There are different options defining what and how the cue is copied. These options appear in a pop-up when the copy command is executed.

Copy cue to a new empty location

Source Cue has three different radio buttons that are mutually exclusive:

- **Content**:
  Copies only the values stored in the source cue.
- **Status**:
  Copies the status of the source cue. Status includes the values stored in the cue and all tracked values from former cues.
- **Look**:
  Copies all values of attributes for fixtures where the dimmer is open in the source cue, no matter if the open dimmer value is set in this cue, or if it tracks into the source cue. All other fixtures used in the source sequence will get only their dimmer copied at 0.

Tracking into Destination Cue also has three radio buttons that are mutually exclusive. This setting defines what should happen with values, that track into the new copied cue, from previous cues.

- **Keep**:
  The tracking values are kept and track into the new cue.
- **Force Release**:
  Tracking values are forced to the "Release" special value.
- **Force Default**:
  Tracking values are forced to the default value.

Cue Only is another three radio buttons. Read more about cue only in the **Store Cues topic**.

- **Off**:
  The **Cue Only** function is Off and values will track onward.
- **On**:
  Cue Only is on.
- **On (Default New)**:
  Cue only is on. Attributes that do not have previous value to return to will be set to the default value.

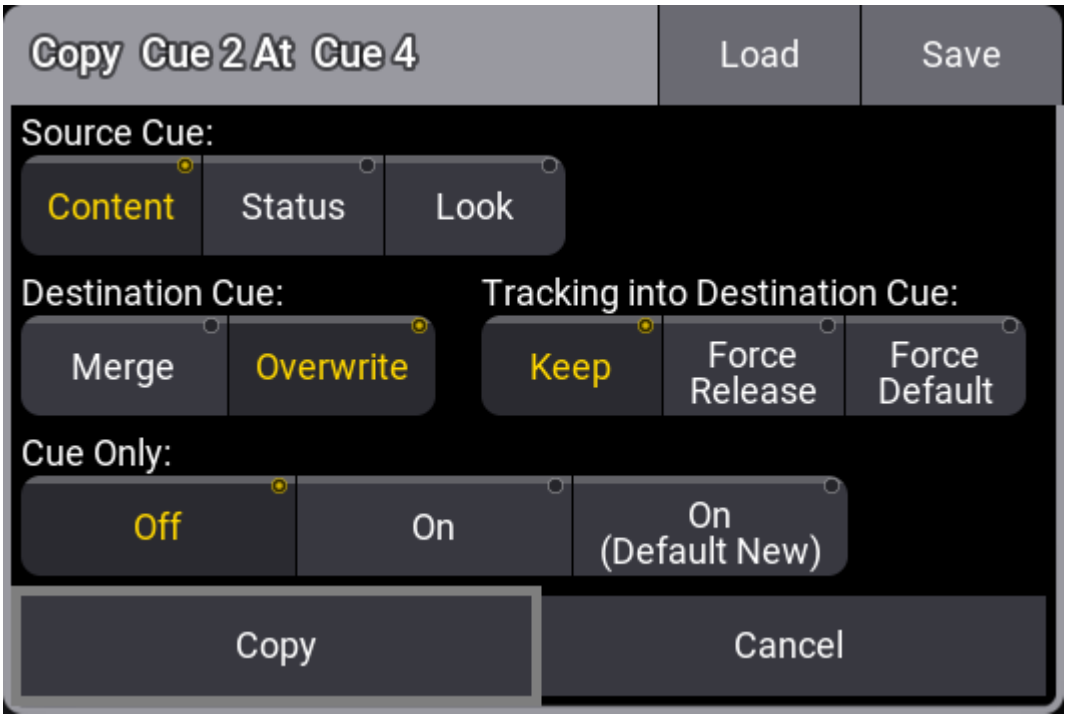
Finally, there are two execution buttons:

- **Copy**:
  Copies the source to the destination using the settings above.
- **Cancel**:
  Cancels the cue copy operation and closes the pop-up.

Copy to an Existing Destination

Copying a cue to an existing cue uses the same syntax except the destination is an existing cue.

The pop-up look s a little different:



Copy cue to an existing cue

There is another option called Destination Cue. This defines what should happen with values existing in the destination cue. Values from the source cue will copy into the cue. This option defines what happens with values in the destination cue that is not defined by the source cue.

- **Merge**:
  Merges the data of the source to the destination. Existing data in the destination will be kept as long as both, the source and destination do not have data of the same attribute. If existing data in the destination is also part of the source, the data from the source wins and overwrites the destination.
- **Overwrite**:
  Overwrites the cue with the source cue data and deletes values that are not defined by the source cue.

Copy of Ranges

When copying ranges of cues these rules apply:

- When a natural order of cues is selected by using **Thru**, the software attempts to maintain all gaps from the source range at the destination.

Example:

Precondition: Cues 1, 2, and 4 exist.

Result: New cues 11, 12, and 14.

- If a cue exists already in the range of the destination spot (even if it would fit into a gap), the software suppresses the range at the destination, so that the original order of cues is uninterrupted. The software tries to set whole cue numbers. If this is not possible it appends dotted cue numbers. The software tries to use as less as possible decimals (1 -> .1 -> .01 -> .001).

Examples:

Precondition: Cues 1, 2, 4, and 12 exist.

Result: New cues 11, 11.1, and 11.3. The gap between cues 2 and 4 is maintained, but shifted, as the start of the destination is 11 and not 11.1.

Precondition: Cues 1, 2, 4, and 11.1 exist.

Result: New cues 11, 11.01, and 11.03. The gap between cues 2 and 4 is maintained, but shifted, as the start of the destination is 11 and not 11.1.

- When the source range of cues is selected in a reversed order by using **Thru**, the software ignores the gaps from the source range when creating the destination range. The same rule for the resulting cue numbers at the destination from above applies here as well.

Example:

Precondition: Cues 1, 2, and 4 exist.

Result:
The content, status, or look of cue 4 is copied to cue 11.
The content, status, or look of cue 2 is copied to cue 12.
The content, status, or look of cue 1 is copied to cue 13.

- In the case of defining the source range by using **+**, the arising gaps between the single cues can be maintained if the cues are selected in ascending order. If the order is reversed or mixed, then the gaps are suppressed. The same rule for the resulting cue numbers at the destination from above applies here as well.

Examples:

Precondition: Cues 1, 2, and 4 exist.

Result: New cues 11, and 14

Precondition: Cues 1, 2, and 4 exist.

Result: New cues 11, 12, and 13.

- When the source list is generated by using a combination of **Thru** and **+**, the above rules apply. The thru-part of the source list uses the rules of thru, while the +-part uses the +-rules.

Copy Paste

The copy function places data in temporary memory. This can be pasted to a destination using the **Paste keyword**.

The principle is that a cue is first copied into the memory and then pasted to a destination.

This process does not provide the same options as described above in the normal copy syntax. Paste uses content as the source and will do a standard copy or merge.

Cut

Cues can be cut and pasted as a means to move a cue. Use the **Cut keyword** instead of copy. This does not give the same options as the copy function.

A cue needs to be cut first and then pasted into a destination.

22.8. Cue Recipes

Recipes can be used in cues and presets. See the **Recipe Preset** topic to learn the basics of recipes and about using recipes in presets. It is a good idea to read that topic before this one.

This topic is about recipes in cues.

A recipe can contain multiple lines describing what should happen based on a set of information. The recipe "cooks" values into the cue.

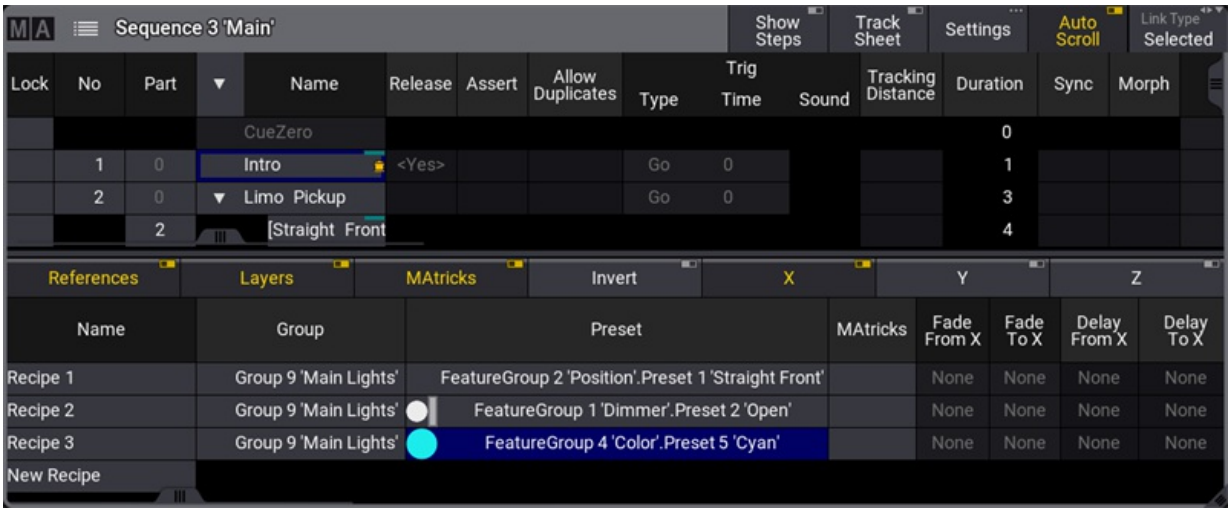
A recipe line can contain information about a group, preset, MAtricks, individual fade, delay, speed, and phase values.

Values from recipes can be combined with conventionally stored values.

Adding Cue Recipes

Recipes are added to each cue part in a sequence.

The best way to access the recipes is by turning On the Show Recipes mask in a **Sequence Sheet**:



Sequence sheet with cue recipes showing

This gives access to adding, editing, and deleting recipe lines.

Showing the recipe lines in the sequence sheet adds a filter line that allows filtering of the different elements in the recipe. This line can be moved up or down by tap and hold the line and slide it up and down. Release the screen at the desired location.

Learn more about this line and the different columns in the **Recipe Presets** topic.

The cells in the lines can be edited and more lines can be added by tapping and holding the New Recipe.

Each line can contain a set of information.

The MAtricks column and the individual MAtricks columns only take effect when there is ranged data from, for instance, a **MAgic preset**.

Making changes to the recipe line automatically cooks the line using the merge option. Recipe lines without a group do not auto-cook.

In that case, the updated values might need to be recooked. This can be done using the **Cook keyword**. An entire sequence can be cooked in one command.

The general syntax for cook is: **Cook [object] (/option)**

There are three options:

- **Merge** - Default if nothing else is specified. Replace cooked data and add new cooked data based on the recipe ingredients, but do not replace non-cooked data.
- **Overwrite** - Removes previously cooked data and adds cooked data based on the recipe ingredients and replaces non-cooked data.
- **Remove** - Removes all cooked data from the cue part.

A cue with recipe information gets the small pot icon in the name column. It does not indicate whether there are cooked values or not.

22.9. Store Settings and Store Preferences

Some settings are used when cues or presets are stored.

Each user has their own preferred settings.

During the store, it is also possible to add commands that will use a specific set of store settings.

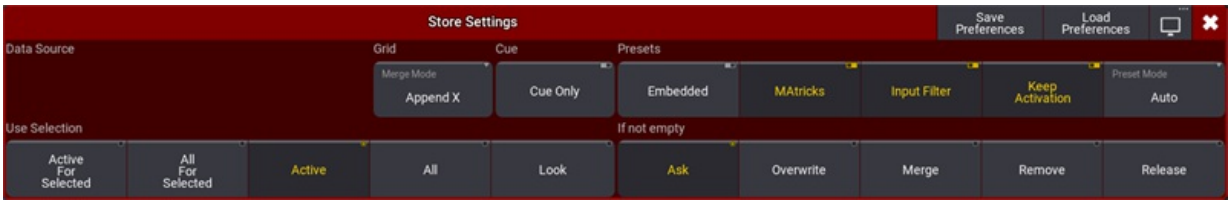
In this topic, the settings regarding storing cues are described. Some of these settings will impact how other cues will respond. Elements regarding tracking are described in the **What is Tracking topic**.

Temporary Store Settings

The store settings can be opened as a temporary version.

This is done by pressing and holding the Store key for approximately one second.

It appears as a pop-up on screen 1 (default location).



Store Settings pop-up showing the current settings

There are different areas with different settings: Data Source (currently empty), Grid, Cue, Presets, Use Selection, and If not empty.

In the title bar, there are three buttons. Tap Save Preferences to store the current settings as the new default user preferences. Tap Load Preferences to load the stored user preferences values into the store settings pop-up. The last button is used to **change the menu location**.

Grid

There is one setting in the grid section. It is Merge Mode. This is relevant for all elements that store the fixture grid position - Learn more about the grid in the **Selection Grid topic**.

The **Merge Mode** is used when an object (including fixture grid position) is merged into another object of the same type. For instance, a group merged into another group or a preset merged into another preset.

There are two options that the button toggles between:

- **Append X**:
  The fixtures are merged into an offset X position in the selection grid. The merged fixtures (source fixtures) are positioned after the last existing fixture (destination fixtures). They are appended to the existing fixtures X position.
- **Off**:
  The fixtures are merged into the grid positions where they are originally stored.

Cue

There is one toggle button here: Cue Only. This can be turned On or Off.

The principle behind cue only is that changes are stored in the specified cues without tracking values into the next cues.

Learn more about **Cue Only** in the **Store Cues topic**.

Presets

These are the settings used when storing presets.

They are described in the **Create Presets topic** or in the other **Preset topics**.

Use Selection

This decides what values, from the source, will be used when storing.

- **Active For Selected**:
  Stores only all active attributes but only for the fixtures selected in the programmer, also with respect for the source selected above.
- **All For Selected**:
  Stores all attributes of the fixtures selected in the programmer, also with respect to the source selected above.
- **Active** - This is the factory default setting:
  Stores the values that are active in the programmer, but with respect for the source selected above.
- **All:**
  Stores all attributes for all fixtures.
- **Look**:
  Look stores all dimmer attributes and for the fixtures that have dimmer values above 0, all attributes will be stored.

If not Empty

This defines the store method used if values are stored into an existing cue.

- **Ask** - This is the factory default setting:
  This means that none of the other options below are used and a pop-up asks the user what to do.
- **Overwrite**:
  All existing data is deleted and the current source and selection are used to store new values.
- **Merge**:
  This will merge the new values into the existing values. New values have a higher priority and will overwrite existing values.
- **Remove**:
  This will remove the stored values for the attributes using the current **Use Selection** setting.
- **Release**:
  A special release value is stored. The actual values in the source are not relevant, but the selection is used to define where the release value is stored.

Cue Preferences

There is a set of timing preferences used when cues are created. They can be seen and edited in the Menu -> Preferences and Timing.

Here are the general cue timings and the preset timings. Read more about these different timings and what they do in the **Cue Timing topic**.

22.10. Play back cues

Cues are stored in a Sequence. All sequences are stored in a sequence pool.

When a cue is active, then the sequence is active in the pool.

It is not possible to run a cue without playing back the sequence.

Executors are controls that allow for easy hands-on playback of the sequences. Executors can playback or trigger other things than sequences as well.

Sequences do not need to be associated with an executor to playback cues.

Relevant playback commands

There are many keywords that can be used for playback operations. They are all listed using the **Help** command.

For sequence playback, there are some very common keywords (read details about them by following the links):

- **Go+**
  Use this to trigger the next cue with a "Go" trigger or to specify a cue (**Go+ Cue cue_number**).
- **Go-**
  Use this to trigger the previous cue.
- **Goto**
  Use this command to go to a specific cue. The specified cue is triggered when the command is executed. Please note that this command also asserts tracked values, so the visual result can be different than a normal Go command.
- **Load**
  This is used to preload cues. The cue is then ready to be triggered via a Go+ command. Several cues in different sequences can be loaded and triggered together with the **Go+ Loaded** syntax.
- **Pause**
  Executing this command will pause all running fades, delays, and phasers. Effectively halting all values where they currently are. The fade and delay are resumed also using the Pause command.
- **Top**
  This keyword is used to trigger the first cue

But there are many more that can be useful.

These functions can be assigned to executor keys for easy access (See how in the **Assign Object to an Executor topic**), but they can also always be sent directly to the sequence. For instance, the following command can be used if you want to go to cue 4 in sequence 6:

The order of sequence and cue does not matter. So it could also be:

The system interprets it as in the last example. The Command Line History window shows this response:

If an executor is handling a sequence then the playback commands can also be sent to the executor.

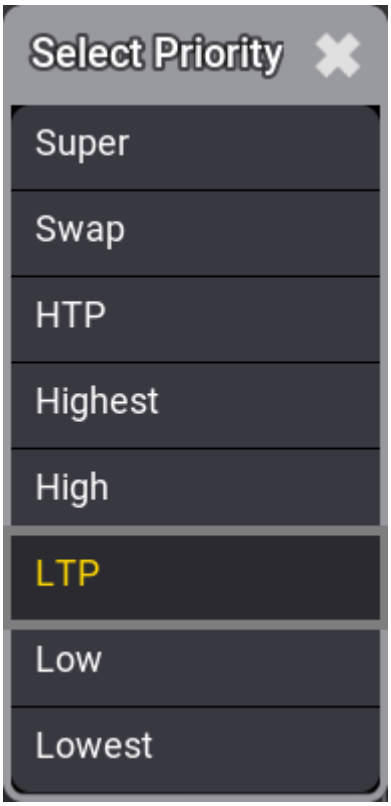
This will send the go command to the object assigned to Executor 101. If this object is a sequence then the next cue will be triggered.

Sequence Priorities

When several sequences are affecting the same fixture attributes then priorities become important. The priority setting is a property of the sequence.

These settings can be opened by pressing the Assign key and then one of the executors keys where there the sequence is assigned. This opens the **Assign Menu**. On the right side of the menu, there are some buttons. Please tap Edit Settings. These are all the settings available for a sequence. The settings can also be opened by tapping Settings in the title bar of a sequence sheet. All the settings are discussed in the **Sequence Settings topic**.

The priority setting can be found in the **Playback** group of settings. Tapping it toggles through the different properties, swiping it opens a small select pop-up like this:



Select Priority pop-up

This is the list of possible priority properties. This is a short explanation of the priorities:

- **Super:**
  This priority is the LTP priority above any other playbacks and even above the programmer.
- **Swap:**
  Intensity is working as LTP with higher priority than HTP.
- **HTP (Highest Takes Precedence):**
  The highest intensity value will be used. Other parameters will use LTP.
- **Highest**:
  Highest LTP priority - Like LTP but with the highest possible LTP priority.
- **High:**
  High LTP priority - like LTP but a higher priority than normal LTP.
- **LTP (Latest Takes Precedence):**
  This is the normal LTP priority. The newest attribute value is prioritized over the old value.
- **Low:**
  Low LTP - This is a lower LTP priority.
- **Lowest**:
  Lowest LTP - This is the lowest possible LTP priority.

LTP is one of the most used priority settings. There are five different levels of LTP priority to be able to give different sequences different priorities.

The list in the pop-up is also a prioritized list, where Super at the top has the absolute highest priority. This also means that HTP sequences have a higher priority than LTP sequences.

Soft LTP

This option is related to how intensities change from one sequence to another when the values transfer from the original sequence to the new one.

When an attribute has a value from a cue and another cue in a different sequence with the same LTP priority is beginning to send new values to the same attribute, then Soft LTP might influence how the transition from the old value to the new value happens. This is only relevant when the new value is applied by moving the master of the new sequence. In this case, the attribute can jump from the old value to the new value and immediately have the value as it should have according to the master position (Soft LTP Off), or it can start to fade from the old value to the new value using the master position as a crossfader (Soft LTP On).

Important:
Both sequences need to be active and both need to send value to the same attributes. **Auto Start** and **Auto Stop** are turned On by default for sequences, if these properties are turned Off, then the sequences need to be turned On manually for the SoftLTP function to affect the transition.
If the Master faders are both at 100%, then SoftLTP does not have a function.

Example

1. Patch a fixture with a dimmer attribute.
2. Give the dimmer a 50% value and store this in a new sequence - call this sequence "Original".
3. Give the dimmer a value of 100% and store this as a new cue in a new sequence. Call this sequence "New".
4. Assign an executor fader as the master for sequence "New".
5. Make sure **Auto Start** and **Auto Stop** are active for sequence "New".
6. Open a fixture sheet and make sure it is set to show the **Output** layer.
7. Run the cue on sequence "Original" and have the master at zero for sequence "New". The dimmer value should now be 50%.
8. Now move the master slowly up from zero and see the output value change.

Soft LTP On: The dimmer will fade from 50% to 100% while the master moved up to 100 (fade from the "Original" value to the "New" value).

Soft LTP Off: The dimmer will jump down to the value dictated by the master position and fade up together with the master (jump from the "Original" value to the "New" value based on fader position).

9. Now try to move the fader slowly down to zero.

Soft LTP On: The dimmer fades from 100% to 50% (from the "New" value to the "Original" value).

Soft LTP Off: The dimmer value will fade down to 0% with the master and when the sequence turns Off then the dimmer will jump to 50% (Fade down with "New" master position - jump to "Original" value when "New" is turning Off).

22.11. Move In Black

Move in black (MIB) is a function where tracking sequences look ahead and preposition attributes of fixtures that are fading the dimmer in from zero, to automatically prevent transitions where the fixture would move the attributes into position, while the fixture is fading in.

MIB is enabled on a cue part basis, by giving the MIB-property of the cue part a value which tells the console when and how it should do the prepositioning.

There are several options and properties that modify MIB behavior. Two of them are **MIB Fade** and **MIB Delay** timing properties.

MIB Fade is the fade time of attributes that will be positioned by MIB. It is available in different places. There is an order of importance which MIB fade time to apply:

1. Per cue part
2. Per attribute in a fixture type
3. Global in the show file

When a MIB fade time is specified in the cue part, the individual attribute MIB fade time will be ignored. When the MIB fade time of the cue is set to default the global MIB fade time will be applied, unless an attribute has an individual MIB fade time set.

MIB Delay is the delay time the attributes wait from having the dimmer closed until the MIB fade will be performed. The MIB delay time can be specified per cue part or global in the show file. The same ruleset as for MIB fade applies: When the MIB delay time of the cue is set to default the global MIB delay time will be applied.

When the MIB fade and/or MIB delay is performed between cues, the MIB times specified in the (future) cue part where the dimmer opens again will be applied. For example, a fixture is moving in cue 3 to be ready for cue 5. The MIB times specified in cue 5 is used for the MIB.

Cue MIB Settings

The sequence sheet can display several columns for the different MIB settings. All MIB related columns can be displayed or hidden by the MIB Settings mask within the **Sequence Sheet Settings**. Read more about the sequence sheet in the **Look at Cues and Preferences**.

- **MIB Preference**:
  This is a specification of the suitability of a cue for MIB. The MIB modes Early, UponGo, and Late will prioritize the cue which has the highest rated suitability and will choose this cue for executing the MIB. MIB mode Defined does not respect the MIB preference. The preference is a percentage number from 0(never) to 100(best).
  Edit the cell to type a number or select one of the following preference options:
  - **Never(0)**:
    An MIB will never be performed.
  - **Worst(1)**:
    If no other options, then this cue will be used.
  - **Bad(25)**:
    It is not optimal but better than the two others.
  - **Normal(50)**:
    This is the default value.
  - **Good(75)**:
    This is a better cue than normal.
  - **Best(100)**:
    This is the optimal cue to perform the MIB.
- **MIB Mode**:
  Defines how early or late the MIB shall be performed per cue part.
  - **Default**:
    Performs MIB corresponding to the MIB mode setting of the sequence setting **MIB Mode**.
  - **None**:
    MIB will not be performed for this cue.
  - **Defined**:
    A specific cue can be defined in the **MIB Target** column where the MIB is to be performed. The MIB is performed when the specified cue is active.
  - **Early**:
    Performs the MIB as soon as the dimmer is closed. Typically after the cue transition has finished.
  - **UponGo**:
    Performs the MIB with the next cue transition after the dimmer has closed. The MIB executes with the cue after **Early** would have triggered the MIB.
  - **Late**:
    Performs the MIB latest in the cue before the dimmer opens again.
- **MIB Target**:
  A specific cue where MIB is performed for this cue part, see MIB Mode above. When setting an MIB target the **MIB Mode** will be changed to **Defined**, and vice versa when changing an MIB Mode that is not **Defined** will remove the **MIB Target**.

- **MIB MultiStep**:
  It is possible to decide whether a phaser where the fixtures are already prepositioned, shall keep running with the closed dimmer, or if they shall be paused. This can prevent unwanted noise and movement of stepper motors for prepositioned fixtures that are running a phaser, especially when it would be disturbing for the audience. The two options are:
  - **Running**:
    A phaser is running with a closed dimmer.
  - **Paused**:
    A phaser will be prepositioned but does not start running until opening the dimmer.
- **MIB Fade**:
  The MIB fade time per cue part. It can be a set time or default. Default takes the global show file MIB fade time or the attribute MIB fade time.
- **MIB Delay**:
  The MIB delay time per cue part. It can be a set time or default. Default takes the global show file MIB delay time or the attribute MIB delay time.

The sequence sheet hides those MIB cells, that are not considered for the different combinations of MIB settings or if a cue or cue part is not suitable for MIB.

Global MIB Settings

To change the global MIB Preferences go to Menu - Preferences and Timing.

In the Timing tab, there is a section called MIB Timing.

This defines the default **MIB Fade** and **MIB Delay** times. This value is input as time. For more information about MIB fade and MIB delay, please read above.

The property **MIB Transition** defines which transition type will be applied to the fade of MIB. Read more about the different types of transitions in the **Cue Timing topic**.

The MIB Transition can only be defined for all MIB fades globally in the show file. It is not possible to define a different transition type for a single MIB fade per cue or cue part.

In the Cue tab, there is a section called **MIB Preferences**.

Here it is possible to change the defaults that will be set to a new cue when it is stored.

- **MIB Mode**:
  The MIB Mode used when storing a new cue that is able to execute MIB.
- **MIB Fade**:
  The MIB fade time used when storing a new cue that is able to execute MIB. Default uses the time set in the timing tab.

- **MIB Delay**:
  The MIB delay time used when storing a new cue that is able to execute MIB. Default uses the time set in the timing tab.
- **MIB Dynamic**:
  The MIB dynamic mode used when storing a new cue that is able to execute MIB.

For more information about these settings, read above.

Sequence MIB Settings

There are sequence-wide MIB settings within the sequence settings. Read more about the settings in the **Sequence Settings topic**.

The purpose of the sequence MIB settings is to have the option to overwrite the cue-based MIB settings with the MIB settings.

When setting the MIB to any option except Enabled, the sequence sheet displays the corresponding setting in the MIB Mode column extended by an exclamation mark, for instance, Never!

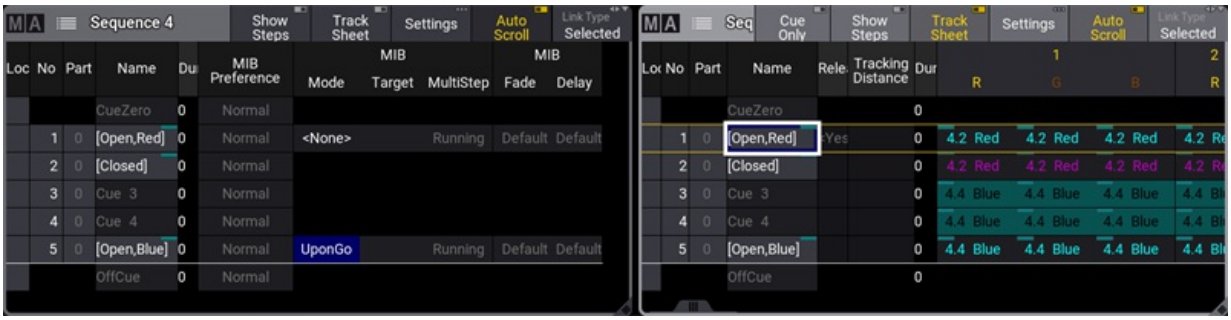
The MIB Mode setting is used to select the default MIB mode used when doing an MIB. The MIB Mode per cue or cue part needs to be set to Default.

For more information about the different MIB Mode types per sequence, please read above.

MIB Color Indicators

Attribute values will be displayed with special colors within the sequence sheet in track mode to show when the MIB is performed.

In this example, the tracking sheet view shows when the MIB is performed.



Two versions of the sequence sheet showing MIB data and the Track Sheet

The fixtures need to be ready for cue 5. In cue 5 the MIB Mode is set to "UponGo". This means that when the fixtures fade to 0% in cue 2, they are ready to MIB with the next cue trigger. When cue 3 is activated they will perform the MIB for cue 5.

The default color indicators are a deep-sea green background and black text color. Read more about the MIB colors in the fixture sheet and other different grandMA3 colors in the **Colors topics**.

22.12. Cue Timing

Each cue part has a lot of timing information.

If nothing is defined on cue creation, then the default times are used. The default can be changed. Read more about defaults in the **Store Options and Defaults topic**.

Any of the cue timings can be changed at any point using the command line or the GUI (**Sequence Sheet**).

There are a lot of elements that affect how a fixture changes values and they can be divided into different groups:

- **General cue times**
- **Preset Timing**
- **Individual Attribute times**
- **Executor Time at point of cue execution**
- **Dynamic changed Rate**

This is also the priority list from lowest to highest priority.

There is also the **cue transition** that defines how the values are changing from one to another.

General Cue Times

There are six different general cue timings. Each has its own column in the Sequence Sheet when condensed timing is turned off - read about it in the **Look at Cues and Sequences topic** :

- **Cue In Fade**
  This is the fade time used for all intensity values changing from a lower value and going to a higher and any other attribute changing value. It starts when the cue is triggered and after the Cue In Delay has counted down.
- **Cue In Delay**
  This is the delay before the fade. This defines a countdown time between the cue is triggered and when the fade should begin. The In Delay affects the In Fade. The default value is 0, meaning that there is no delay.
- **Cue Out Fade**
  The out fade is used by intensity values fading from a higher value to a lower. It is executed after the Cue Out Delay. The default value is the in fade, meaning that it is the same as whatever the cue in fade is.
- **Cue Out Delay**
  This is the delay time for the Out Fade. It can be used to delay when intensity values should start to fade down in value. The default value is in delay, meaning that it is the same as the Cue In Delay value.
- **Snap Delay**
  Some attributes are defined to Snap. This means that they do not fade from one value to another. They change values as fast as possible. This can make sense for attributes like gobos. This delay is used to delay when the snap is performed. It makes it possible to have the fixture fade out before snapping to a new gobo. This delay affects all snap attributes stored in the cue. It can be overwritten by individual attribute timing.

- **Cmd Delay (Command Delay)**
  This delays the execution of a command. This is the only place where this delay can be defined. There are no individual timing for this.

These values can be changed using this syntax:

Cue cue_number CueInFade new_cue_time

Just use the relevant keyword: **CueInFade**, **CueOutFade**, **CueInDelay**, **CueOutDelay**, **SnapDelay**, or **CmdDelay**.

There are two more options for setting the cue fade and delay. They are called **CueFade** and **CueDelay**. They can be used to set the four cue fade and delay times mentioned above. They are used to set the in and out time using just one keyword.

The in and out time is separated by a slash. The time before the slash is the **In** time. The time after the slash is the **Out** time. It is not necessary to specify both in and out. The slash can be used to set just one of them. The desired time has to be on the correct side of the slash.

If only one time is set without any slash, then the in time gets the actual value and the out time is linked to the in time - it is technically set to time "None". The result is that, for instance, the fade in and fade out will be the same.

For example, setting the in fade to 5 seconds and the out fade to 8 seconds in cue 3 could be done with the following keystrokes:

Cue | 3 | Time | 5 | / | 8 | Please

The command line would read:

If only one time is given then both in and out will use the time. For instance,

This makes cue 4 use 7 seconds to both fade in and out.

They can also be used to address only the in or out time by adding the slash and a number to the relevant side. For instance, setting the CueOutFade to 3 and the CueInDelay to 1 on the currently active cue:

Both CueFade and CueDelay can be addressed using the **Time key**. Read more about delays by following the links above to the keywords.

Preset Timings

Each preset type has its own fade and delay time. There are columns for each of these in the Sequence Sheet.

These times are used by all fixtures changing the values of that preset type. Overwriting the general cue timing for that preset type.

Important:
Intensity values changing to a lower value, are not affected by the "Fade Dimmer" time. They are controlled by the "Cue Out Fade".

The preset type fade and delay default is general cue timing. This means that they are the same as the time for the Cue In Fade and Cue In Delay.

Individual Attribute Timing

Each individual attribute can have an individual fade and/or delay time. This is called individual time. These individual times are selected using the **programmer** and added to the cue when it is **stored**. They can also be edited in the sequence sheet when it is in **tracking mode**.

There are two columns in the sequence sheet called **Indiv Fade** and **Indiv Delay**, they display the highest of the individual times. The column called **Indiv Duration** shows the complete individual time (delay + fade).

Executor Time

There is an **Executor Time** master fader. It can be seen and changed in the **Master Controls pop-up**. It can also be **assigned to physical controls or executors**.

If this function is turned On, then the set time will be used instead of the stored fade and delay timing. The **Executor Time** faders position is registered when the cue is triggered. The fader can be moved afterward without this affecting the running cue fade. Read more in the **Time Control topic**.

Sequences can be protected from this function by turning Off the **Use Executor Time** setting in **Sequence Setting**.

Rate Time

The **Rate** allows the timings to be dynamically adjusted while the cue fade is running. It does not change the stored times, it simply adjusts the time to be faster or slower.

Moving the fader up makes the stored times smaller, making the fade faster. Pulling the fader down makes it go slower, extending the stored times.

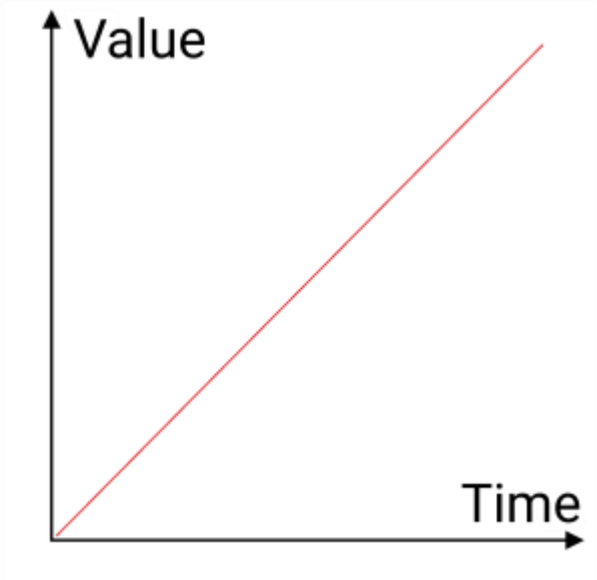
The rate can be reset using the **Rate1 keyword**.

Cue Transition

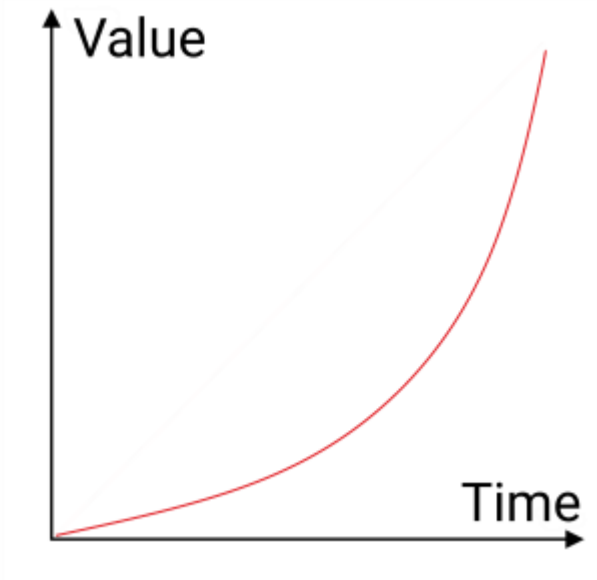
The cue part property **Transition** allows modifying the path values use during the fade from one cue to another cue. The transition can be set per cue part.

There are nine different transition forms available. They are visualized with these images.

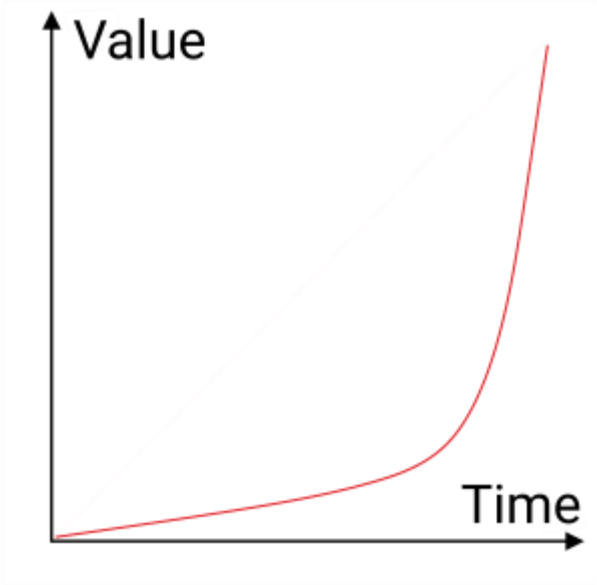
- **Linear**:
  This is the default transition. There is no acceleration or deceleration of the values.



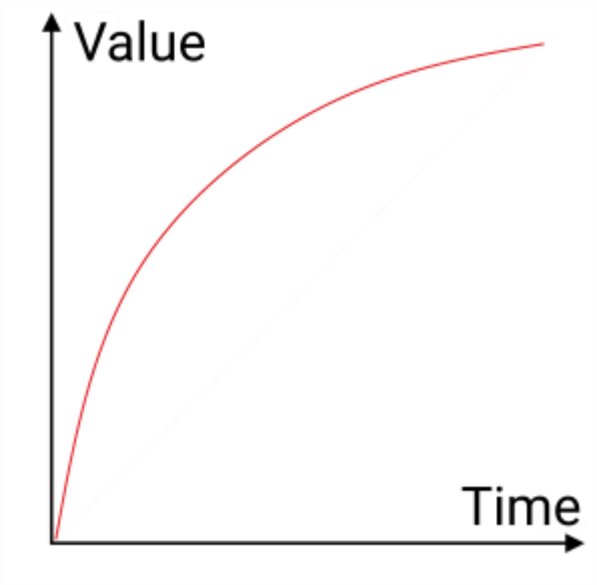
- **Slow**:
  This uses a slow acceleration and fast deceleration.



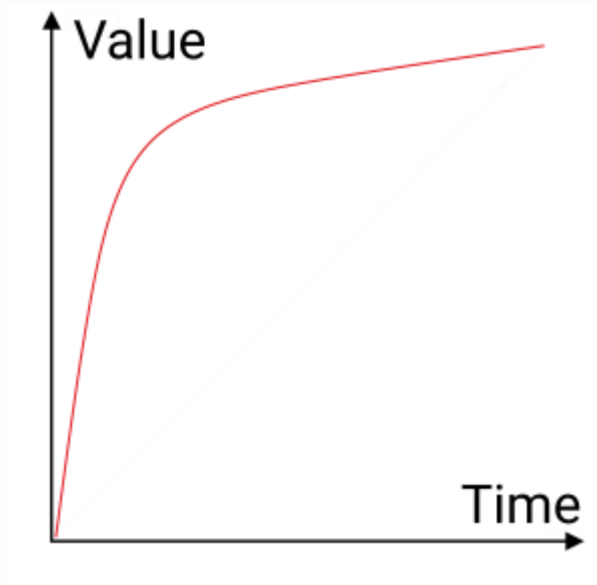
- **Slow+**:
  This is an exaggerated version of **Slow**.



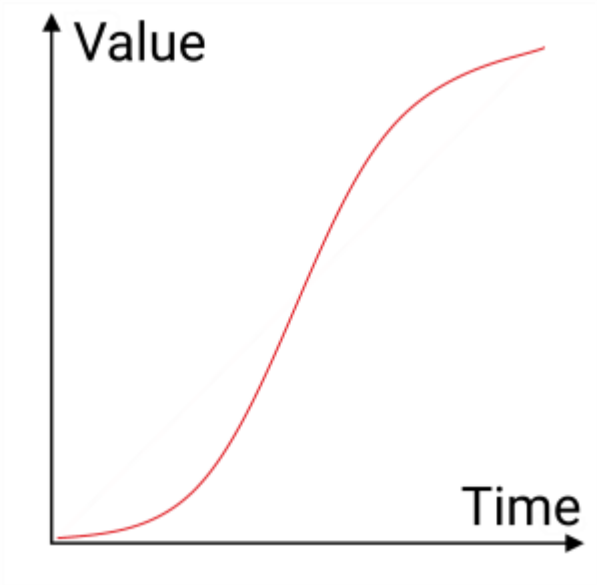
- **Fast**:
  This is a fast acceleration and slow deceleration.



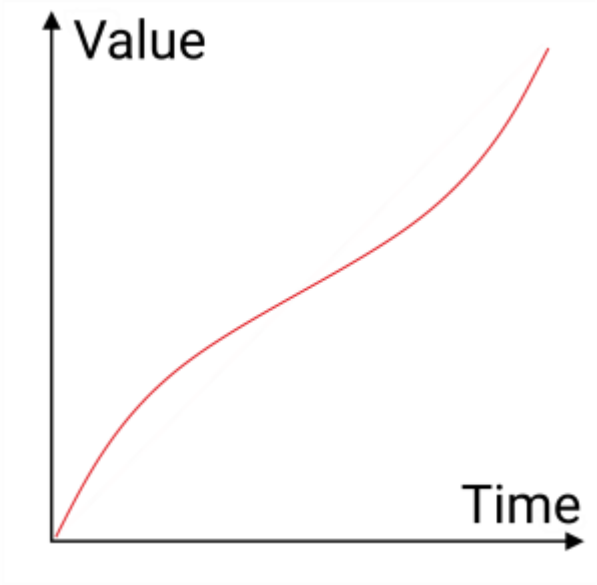
- **Fast+**:
  This is an exaggerated version of **Fast**.



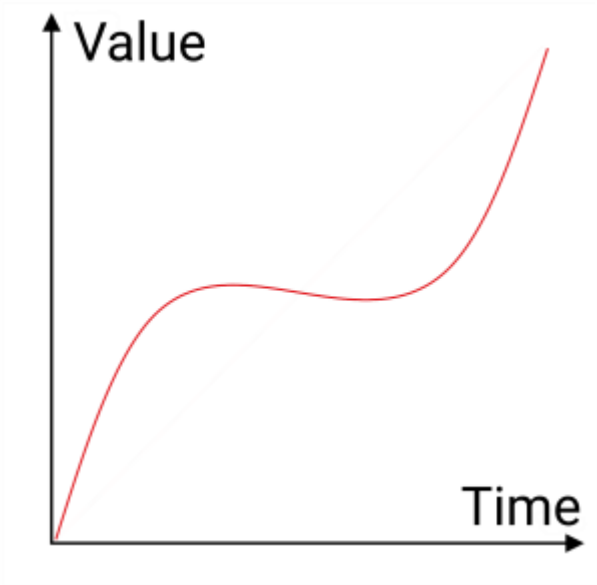
- **SCurve**:
  This uses slow acceleration and slow deceleration.



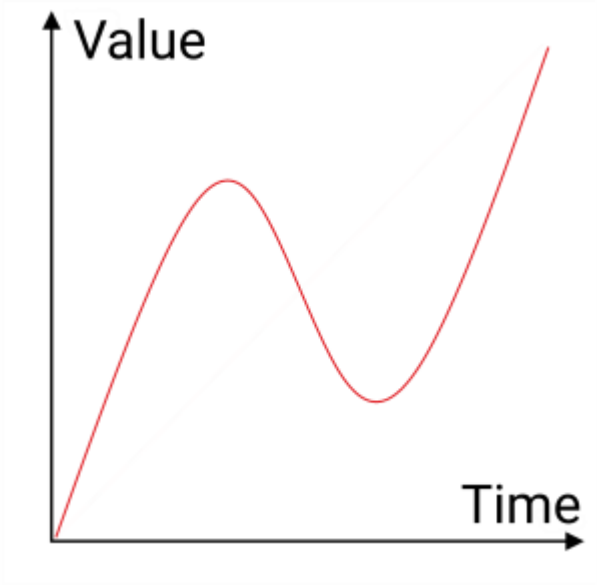
- **Swing-**:
  This uses fast acceleration and fast deceleration.



- **Swing**:
  This is a slight exaggeration of **Swing-**. The values change slows down in the middle of the transition.



- **Swing+**:
  This is an exaggeration of **Swing**. The values go backward in the middle before accelerating forward again.



22.13. Renumber Cues

Cues can be renumbered. This cannot be used to change the order of the cues.

Single cues can be renumbered but it can also be a range of cues.

It is not possible to give the same cue number to two or more cues - they must be unique.

Setting the cue number to a previous number that is already used is prevented by the system.

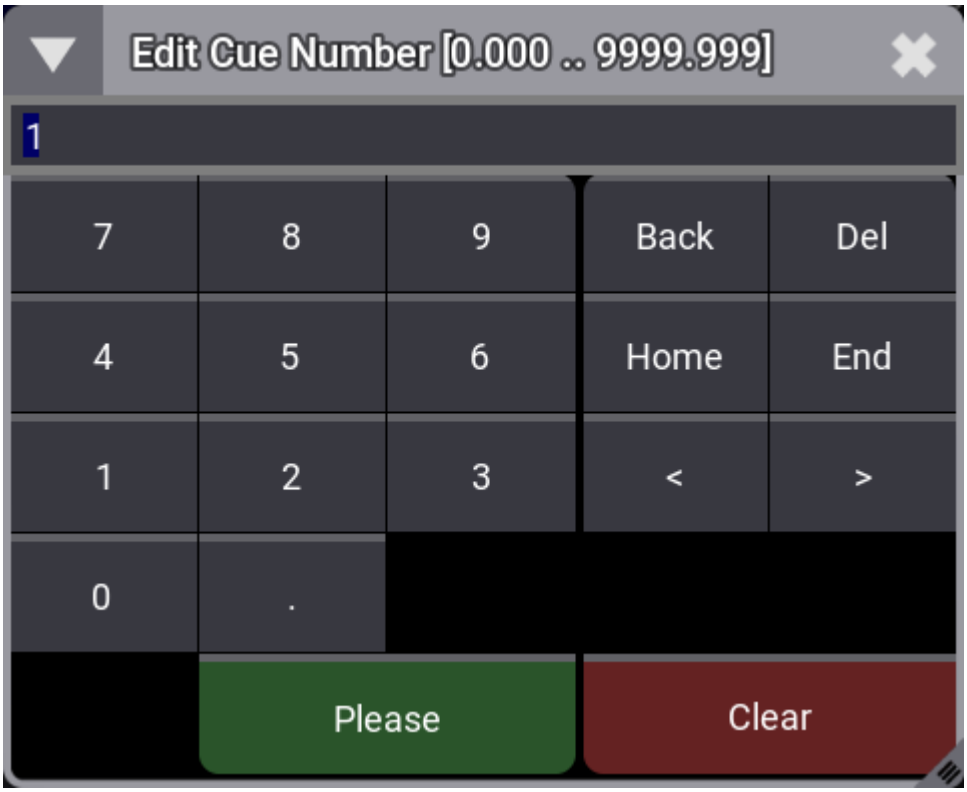
Entering a cue number that is used later in the sequence, all following cues will be renumbered until no collision with existing cue numbers occurs.

Requirement:

Have a sequence with cues.

Procedure:

1. Open **sequence sheet window**.
2. Select the cues numbers that are to be renumbered using the number column in the sheet.
3. Press Edit and tap inside the cue selection.
   This opens an Edit Cue Number pop-up.



Edit cue number pop-up

4. Type the new cue number.

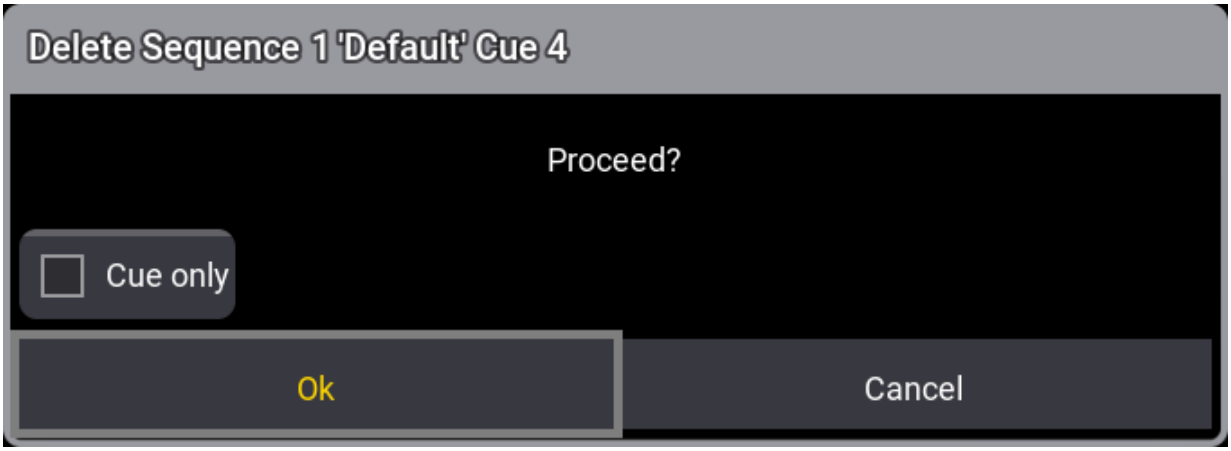
If several cues are to be renumbered then use the lasso selection when tapping the cue numbers.

22.14. Delete Cues

Cues can be deleted using the **Delete keyword**.

This can be accessed using the command line input or the **Delete key**.

Deleting a cue opens a pop-up asking for confirmation.



Delete Cue pop-up

There is a small **Cue Only** checkbox. Checking this deletes the cue using the **cue only** principle.

Tap Ok to delete the cue and close the pop-up.

Tapping Cancel closes the pop-up and does not delete the cue.

Deleting the last cue in a sequence does not open the pop-up, it simply deletes the cue.

23. What are Executors

The executors are handles and controls to other objects.

They are often used to control sequences, but they can control other objects.

The idea is that an object (for instance a sequence) is assigned to the executor. The executor can then control the object. Several executors can control the same object.

A sequence assigned to an executor is actually running cues from the sequence pool. In essence, the executor is manipulating or sending commands to the sequence in the pool.

Executors are physical keys (executor buttons), knobs (executor knobs), and faders (executor faders) on the grandMA3 hardware. They can also be represented as on-screen virtual executors - these can be viewed and operated in the **Playback Window** (see below).

Read more about the physical executor hardware in the **Executor Elements topic**.

There are 4 rows of executors. The bottom row is numbered from 101 and up. This row only has a single executor button. The row above is numbered from 201 and up. This row has an executor button and an executor fader. The next row is from 301 and up. This row has an executor button and a rotating executor knob. The top row is from 401 and up. This row also has an executor button and a rotating executor knob.

The Xkeys are also executors. The executor buttons labeled X1 to X8 are executor 291 til 298. The executor buttons labeled X9 til X16 are executors 191 til 198.

Executors are organized in wings. There are vertically 15 columns of executors on a wing. They are organized in sections of 5. There are 60 (15 columns x 4 rows) executors on a wing. There are 6 wings for a total of 360 (6 wing x 60 executors) executors plus the 16 from the Xkeys - the total is 376.

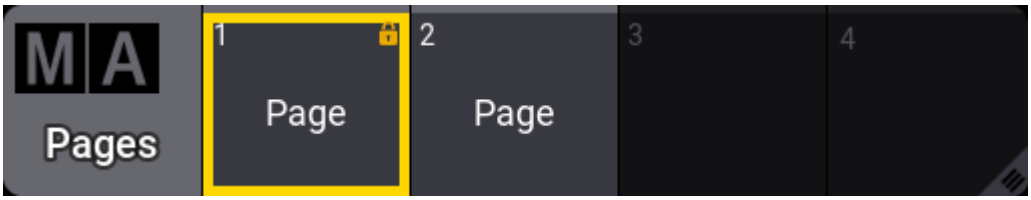
The physical device might have fewer executors, then the amount existing in the software. For instance, the grandMA3 compact console has 2 x 5 columns of executors.

The executors can also be used from the command line using the **Executor keyword**.

Executor Pages

The 376 executors are just on one page. It is possible to create up to 9 999 executor pages. Each page has its own setup of executors. Executors that are active on one page are still active when the selected page is changed to another.

The executor pages can be seen and labeled in the **Page Pool** - it can be **created as a window**.



Page pool with page 1 selected.

The page pool is also available as a pop-up, that can be opened by tapping the middle part of the page selector.

Tap the middle part of the page selector to open the page pool pop-up

There are no functional differences between the pool as a pop-up or as a window.

Pages are automatically created when the Page+ and Page- keys are used to change through the pages. The up and down arrows in the page selector can also be used to change the page just like the physical keys.

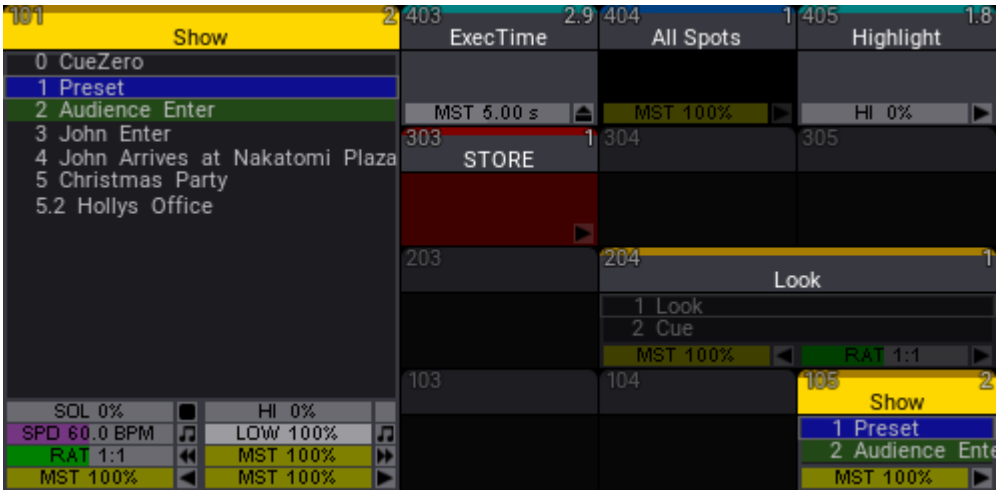
A page can also be created using the following syntax:

Store Page page_number ["page_name"]

The page name is optional in the syntax above. Read more in the **Page keyword** topic.

Executor Labels

The letterbox screens above the executors show the labels for the executors. The bottom of the left command screen shows the labels for the Xkeys.



Example of the playback bar for the first section of 5 x 4 executors

The executor labels above the executors are a part of the Playback Bar. Read more bout them in the **Playback bar topic**.

The labels display how the encoders are grouped, what object they are controlling, and what handle function each executor has.

Read more about how to change all these things in the **Assign Object to an Executor topic**.

The labels take appearance and scribbles from the object assigned. For example, if a sequence has a special appearance then this appearance is also displayed in the playback bar.

Playback Window

The playback window is a virtual representation of the physical executors.

Playback window showing all four rows and both labels and executor handles

The window can automatically adjust the number of visible executors based on the width of the window and the selected amount of sections. This means that a window can minimum have 5 x 4 executors and a maximum of 15 x 4 (one wing). The minimum size of the window to display a section of 15 executors is 10 squares wide on a

screen.

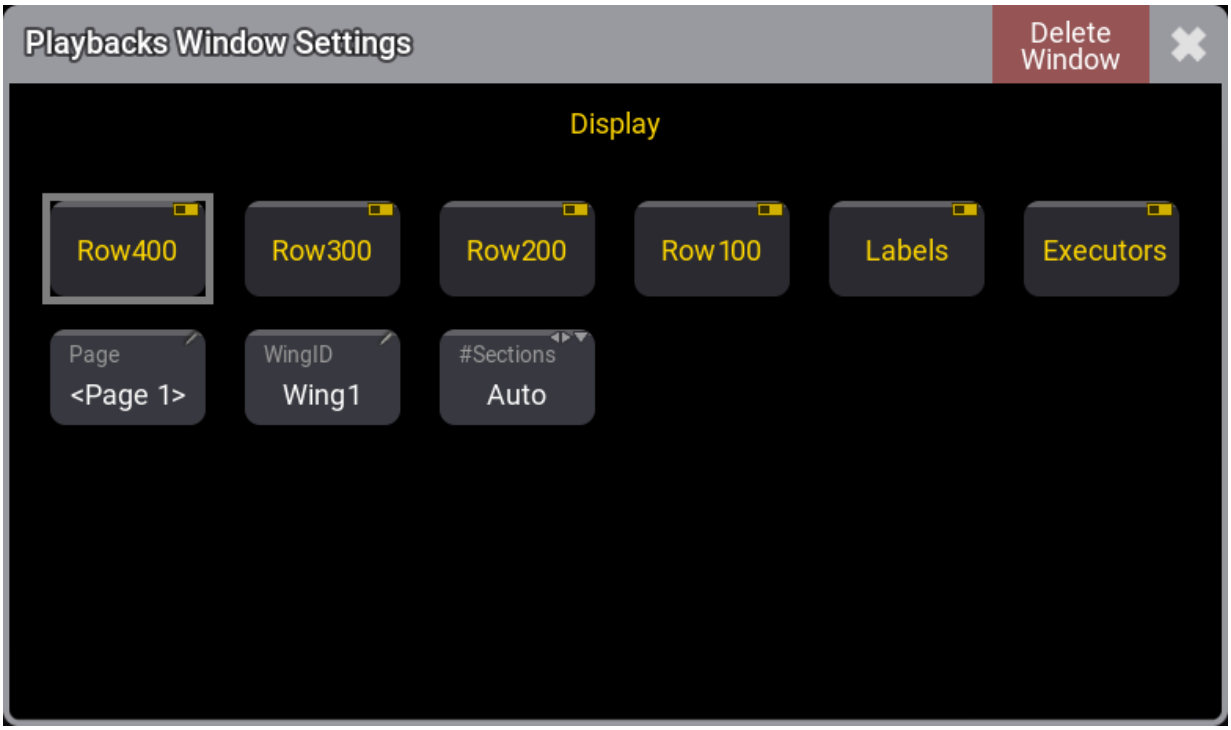
The virtual executors can be operated on the screen.

The title bar has two buttons. Page and WingID. They relate to which executor page and wing are shown. They are described in the **Window Settings** topic.

Playback Window Settings

The settings allow for customization of the window.

The settings can be accessed by tapping the MA logo in the upper left corner.



Playback window settings

There are toggle buttons that can show or hide the four executor rows. It can also show or hide the labels and the executor handles. This makes it very flexible. For instance, it could just show the labels for the 200 row of executors or just the executor handles for the 100 row executors.

The Page and WingID buttons are the same as the ones in the title bar (see description above). They are also described in the **Window Settings** topic.

The #Sections setting is used to select how many sections of five columns the window should display. The properties are:

- **Auto**:
  The window displays as many sections as possible based on the width of the window and a minimum width of each column.
- **1**:
  The window only shows the first section of executors - 5 columns.
- **2**:
  The window shows the first and second sections of executors - 10 columns.
- **3**:
  The window shows all three sections of executors in the wing - 15 columns.

The properties with a specified number of sections will always show the selected amount even when the window width makes it hard to use.

23.1. Assign Object to an Executor

Many objects can be assigned to an executor. The executor is a physical key, fader, or knob that controls the assigned object. The physical devices can also be represented as on-screen controllers.

Quick Steps:
1. Press Assign.
2. Tap what should be assigned or press the relevant keys.
3. Press the executor where it should be.

These are the simplest steps needed for assigning something to an executor - read below for details.

Assigning Objects Using Keys and Pools

It is easy to assign something to an executor.

Press Assign followed by the desired object and then finally the executor where it should be assigned to.

Here are three variations on how it works. The examples use sequences, but it can be any of the allowed types:

Example 1

Using only the keys to assign sequence 3 at executor number 105 on the current page, type:

Assign | Sequ | 3 | At | MA | + | X16 \| Exec | 1 | 0 | 5 | Please

Example 2

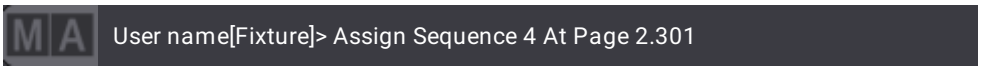
It is also possible to use a combination of keys and pools.

Having a pool visible on one of the screens makes it possible to combine keys presses with pool selection.

1. Tap and swipe out of the sequence pool object that should be assigned.
2. Swipe to the Assign option and release the screen.
3. Press one of the keys associated with the desired executor.

Example 3

Pressing the keys puts keywords into the command line. This means that it is also possible to type it as a command line input.



This command will assign sequence 4 to executor 301 on executor page 2. The page keyword needs to be used when addressing executors on specific pages. The page needs to exist before it can be addressed.

Assign Objects Using the Assign Menu

The **Assign** menu offers a visual approach to assign something to the executors. Use the assign menu by selecting the executor first and then select the object.

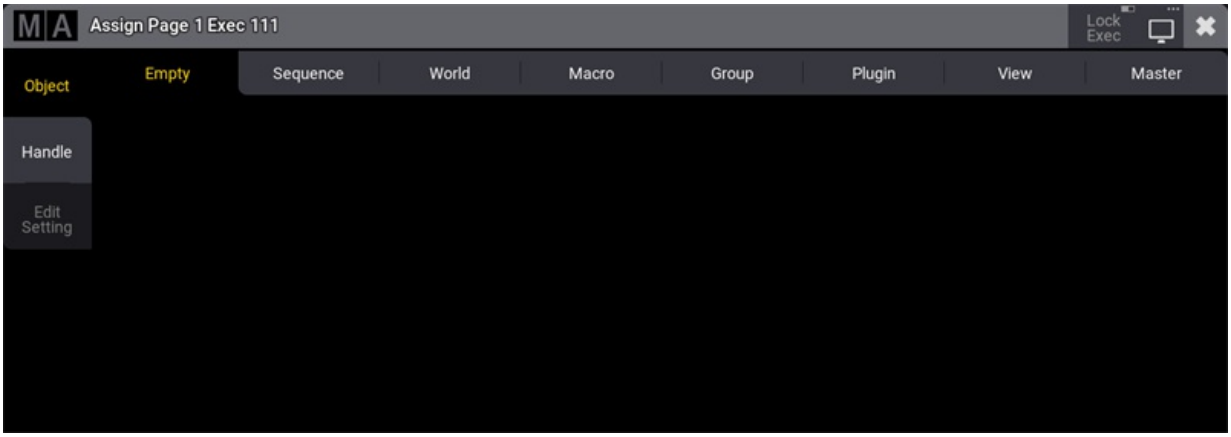
There are two main ways to open the Assign Menu.

- Press Assign followed by any of the keys associated with the desired executor.

-Or-

- Tap the executor label in the **Mini Executor Bar** (do not do a long press - it opens the editor instead).

This is the assign menu:



Assign Menu - Object page

The title bar has a toggle button called Lock Exec. This can be used to lock the executor from changes. It functions normally, the lock only prevents making changes to the executor.

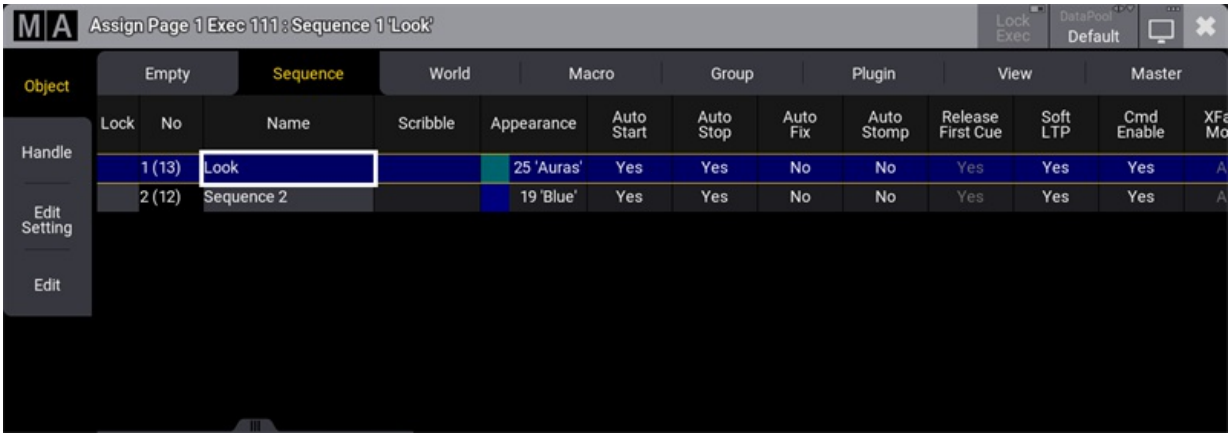
This is the **Object** page of the assign menu. This page is selected by tapping Object on the right side.

The top has several tabs. One for each object type that can be assigned to the executor and a special one used to select an **Empty** object. The other tabs open a selection list. The list will contain the possible objects in that type.

The different types are:

- **Sequence**
- **World**
- **Macro**
- **Group**
- **Plugin**
- **View**
- **Master**

Tap Sequence to open the list of possible sequences. It could look like this:



Assign menu in the sequence object tab

Each of the seven allowed types that can be assigned to an executor provides a list of the available objects. The DataPool in the title bar makes it easy to select an object from a different data pool.

Select the desired object by tapping it.

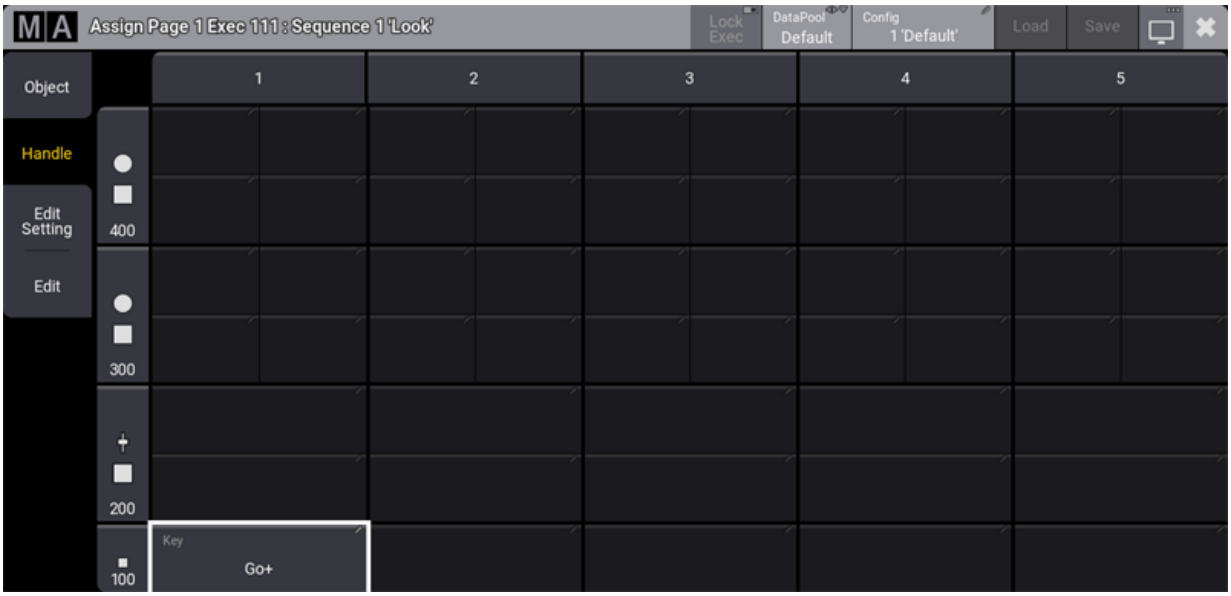
The default handle configuration is applied to the executor. This including executor expansion can be defined on the handle page.

Change Key Function and Executor Size

When the executor has something assigned, then it is possible to change the functions assigned to the executor keys, faders, and knobs.

If continuing to work in the assign menu, then tap Handle on the left side.

It could look like this:



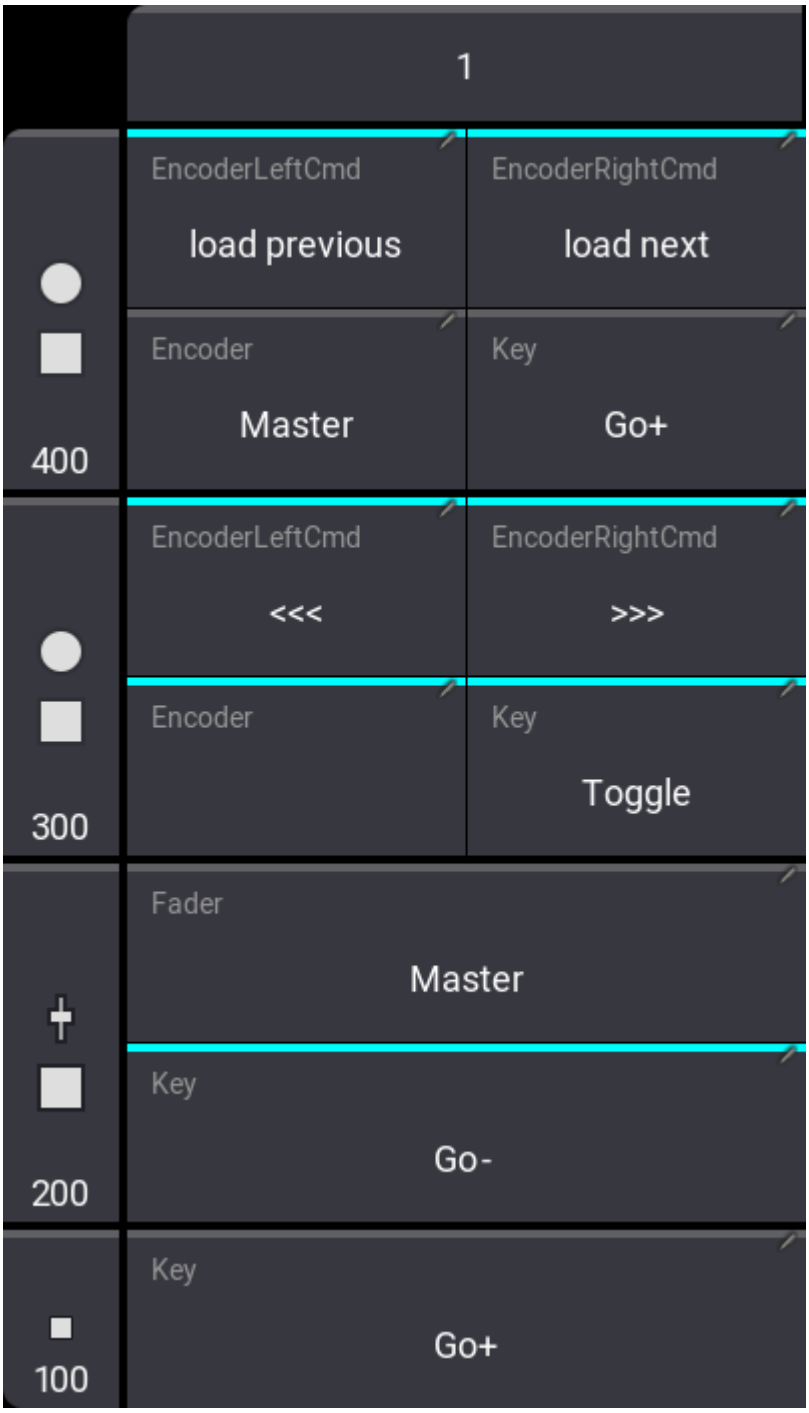
Assign menu on the Handle page

Executors can be expanded if the executors above and to the right are empty. They are grouped into sections of 5 in the width and this grouping cannot be crossed. For instance, executors 104 and 105 cannot be extended into executor 106.

They can also be extended upwards if there is room. The example image above shows executor 111 with room above and to the right. This executor can be extended to a width of all 5 columns and a height that covers all four rows of executors.

It is expanded by tapping the column numbers on the top and the row numbers on the left. When it is expanded then more buttons get available for the executor.

The 100 row only offers a single executor button. The 200 row has a button and a fader. Both the 300 and 400 rows have a button and an encoder knob. If an executor is extended to cover all four rows then it could look like this:



Executor with all four rows in height

The keys, faders, and encoders can have different functions. The options might vary depending on the object assigned to the executor.

Tap any of the available buttons to get a small select pop-up with the available options.

The **Encoder Left Cmd** and **Encoder Right Cmd** open a pop-up that allows any command to be typed. There is even an option where it can be different commands if the MA key is pressed while the encoder knob is turned.

Hint:
The encoder commands are mutually exclusive with the normal encoder function. In the example image above there is both an encoder function and a command function. The encoder commands have a higher priority and overwrite the normal encoder function.

The normal key options are:

- **Empty**:
  The key has no function.
- **<<< (GoFastBack)**:
  This jumps one cue back without cue timing.
- **>>> (GoFastForward)**:
  This jumps one cue forward without cue timing and will not trigger other cues.
- **Black**:
  This turns off the intensity of the executor as long as the button is pressed.
- **Flash**:
  This turns the executor on and sets a virtual intensity master at 100, as long as you have the button pressed. When the button is released it will go back to the status before the button was pressed.
- **Go+**:
  This executes a "Go". It uses fade and delay times.
- **Go-**:
  This fades backward using fade and delay times.
- **LearnSpeed**:
  This sets the speed. By pressing it at least two times it automatically adjusts the speed.
- **Off**:
  This turns the executor Off.
- **On**:
  This turns the executor On.
- **Pause**:
  This holds an active fade and/or delay.
- **Rate1**:
  This resets rate fader.
- **Select**:
  This selects the executor.
- **SelFix**:
  This selects the fixtures used in the object assigned to the executor.
- **Speed1**:
  This resets speed fader to the default speed.

- **Swop:**
  Swop temporarily overrides the master level of executors to full and sets all other master levels to zero. Executors can be protected against the swop.
- **Time**:
  This toggles the time function for the executor. It overwrites the stored cue part times, when it is On.
- **Temp**:
  This turns the executor on as long as it is pressed. The temp function uses the fade times from the cues and it uses the level set by the intensity master.
- **Toggle**:
  This turns an active executor Off and an inactive executor On.
- **Top**:
  This fades to the first cue in an assigned sequence.

These main functions are keywords - they can be found in the **All keywords topics** where there are detailed descriptions of the keywords.

The functions can also be assigned using the keys or commands. Not all functions have a physical key that can be used, then it can be done by writing the command using the keyboard.

Key Example

If the desired function has a physical key, then it is very easy to assign it to an executor key. Simply press Assign, then the function key you want, and finally the executor key where you want the function.

Example 1

For instance, to assign the **Off** function to the key associated with executor number 101, you need to press the following keys:

Assign Off – (the executor key in the lower-left corner on consoles)

Example 2

Assign the **Fix** function to an executor key

Assign MA + Pause [desired executor key]

Command Example

Functions can be assigned using the command line.

This is the general syntax:

Assign [Function] at [Location]

The location needs to be a specific physical key associated with an **Executor Page**.

To assign the Pause function to the top key associated with executor 5 on executor page 8 you would need to type the following command

Change Fader Function

It might be possible to change the function of the fader. This depends on what type of object an executor has. They are changed just as the executor keys - by tapping the onscreen representation in the assign menu (see above).

The **Select Fader** pop-up lists the possible options.

- **Empty**:
  Fader has no function.
- **Master**:
  Controls the intensity. The keyword is **FaderMaster**.
- **X (crossfade)**:
  Crossfades between two cues. Current cue and next cue. The current cue will change when the fader reaches the other end position from where it started. The keyword is **FaderX**.
- **XA (crossfade A)**:
  Is the first of a two-fader manual crossfade between two cues. Works along with XB. The crossfade function can be changed in the **Sequence settings**. The keyword is **FaderXA**.
- **XB (crossfade B)**:
  Is the second of a two-fader manual crossfade between two cues. Works along with XA. The crossfade function can be changed in the **Sequence settings**. The keyword is **FaderXB**.
- **Temp**:
  Crossfades the cue on when pulled up, and off when pulled down. The keyword is **FaderTemp**.
- **Rate**:
  Divides the fade and delay time in a sequence by the value of the fader. If **Speed from Rate** is on, it is also valid for phaser speed stored in cues - see **Sequence Settings topic**. The keyword is **FaderRate**.
- **Speed**:
  Controls the phaser speed in a cue. The keyword is **FaderSpeed**.
- **Time**:
  Sets the time for the executor time overwrite. The keyword is **FaderTime**.

The above functions are described in more detail in each of their keyword topics.

The functions can also be assigned using the command line.

The syntax is the same as the key functions. Except "empty" - see examples below.

Examples

Example 1

To assign the rate function to executor number 209 on the current page, you will need to type the following command:

Example 2

If you want to assign the "Empty" function to the executor fader, then it is necessary to use a different syntax. This example assigns empty to the left-most fader on page 1:

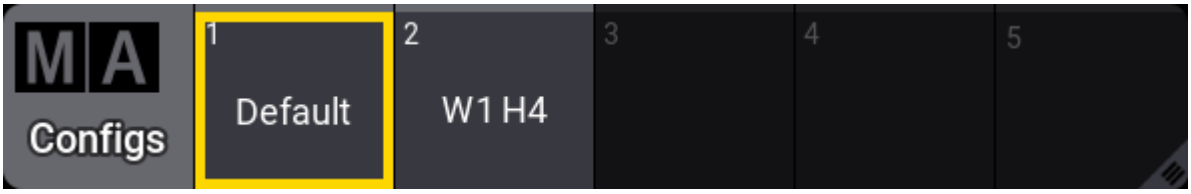
Save the Key and Fader Assignment

The current assignment of key and fader functions can be saved into a pool of different configurations. Read more in the **Executor Configurations topic**.

23.2. Executor Configurations

Executor key and fader configurations can be saved and reused. Read the **Assign Object to an Executor topic** for information on how to change the assignment.

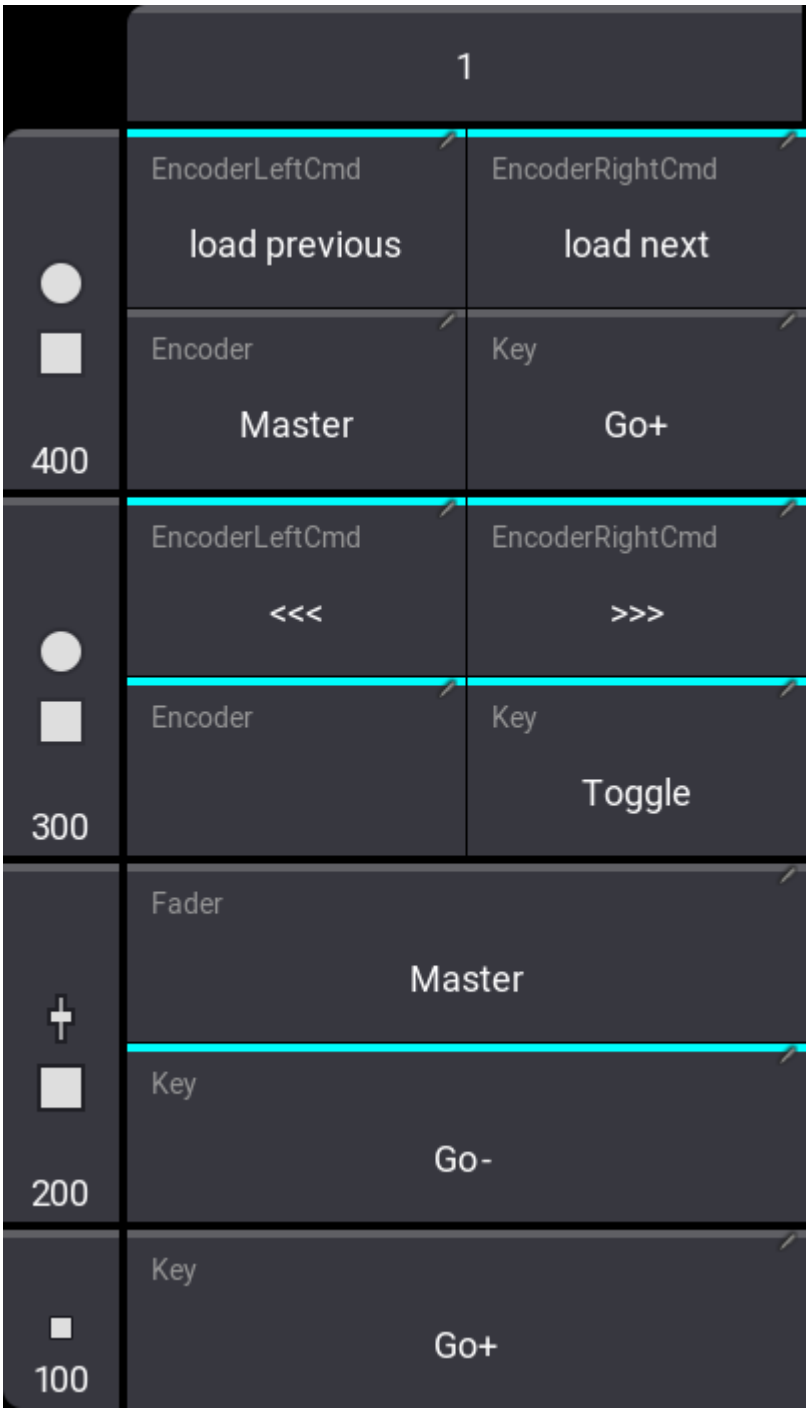
The configurations are stored in an Exec Config pool. This can be created as any other window. Read the **Add Window topic** to learn how.



Executor Configuration pool with a selected configuration

When an object is assigned to a new executor, then the selected configuration is used.

This configuration can be changed in the assign menu.



Key and Fader configuration with unsaved changes

In the example above, there are several Keys and Encoders that have a different assignment than the original saved configuration. The different assignments are marked with a cyan color bar.

Restriction:
Be aware that the image above shows an encoder with both an encoder function and encoder turn commands. These are mutually exclusive functions and the commands have higher priority. The normal encoder function is ignored.

In the title bar of the assign menu there are three buttons called Config, Load, and Save.

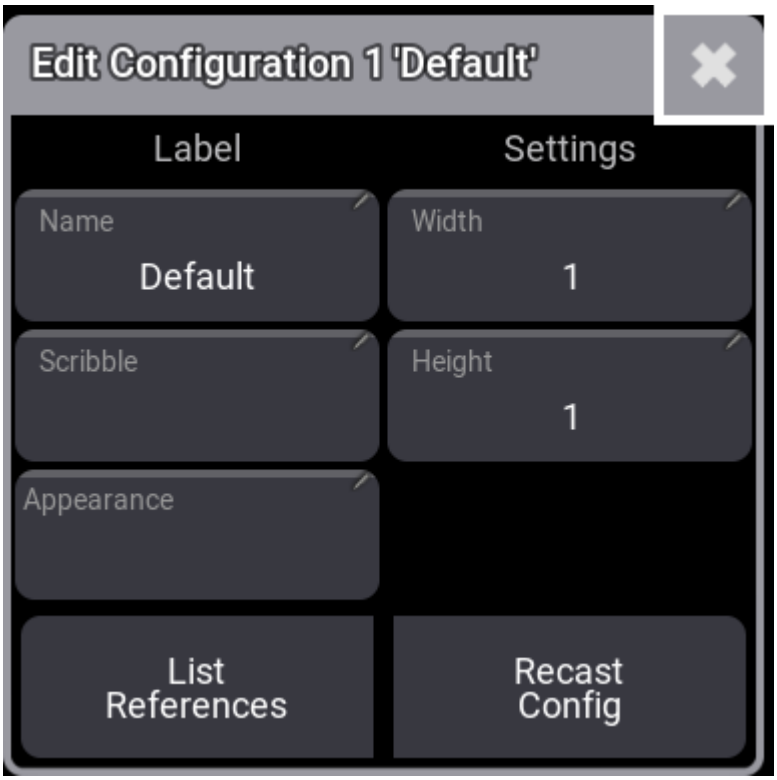
Title bar of the assign menu

Config shows the currently selected configuration and tapping it opens a small select pop-up where any of the existing configurations can be selected or a new one can be created.

Hint:
Creating a new config this way will discard all changes that were made until then.

Load and Save are used to load the selected configuration onto the keys and faders. Save can be used to save the currently assigned functions to the selected configuration.

Editing the configuration pool object opens an editor like this:



Edit configuration pop-up for the first pool object

Here it is possible to change the name of the configuration.

This is also where the width and height are set for the configuration. It is not shown here, but the configuration also knows the starting row for the configuration. This means that a configuration of 1x1 can be different for the rows.

The configuration can also have an appearance and scribble assigned. This is only visible in the pool.

Tapping List References opens an info pop-up that shows the objects that references the configuration and what other objects this configuration might depend on.

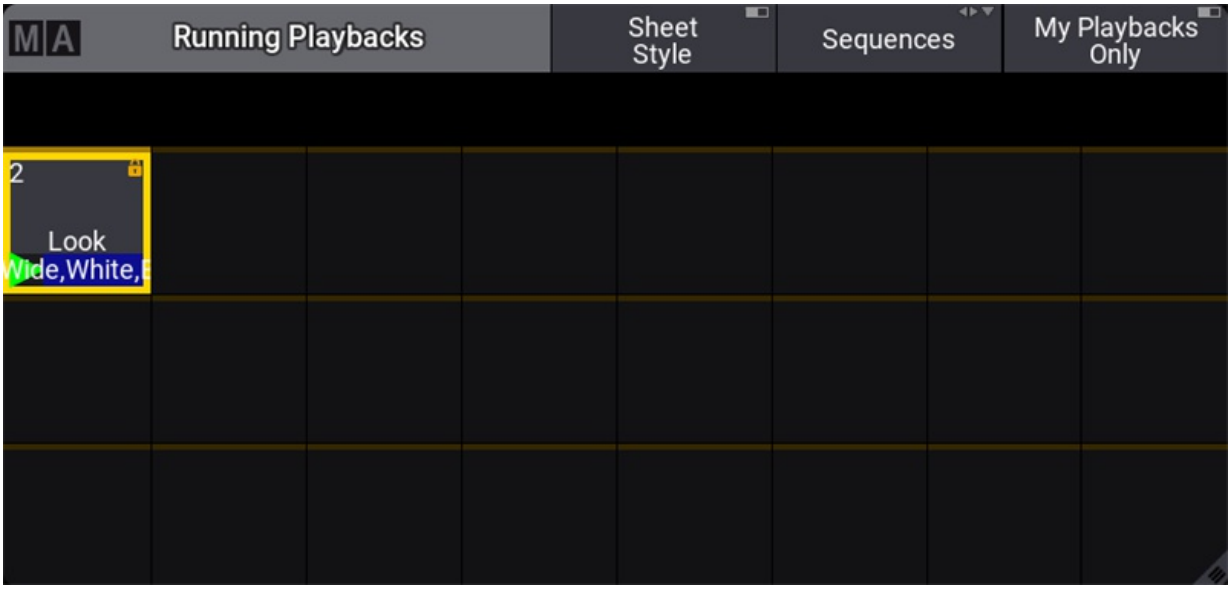
Recast Config is used when a configuration has been saved with changes and these changes also should be applied to the other executors that use this configuration. Saving a change to a configuration does not automatically apply the changes to other executors.

23.3. Running Playbacks

There are two ways to see the different running playbacks. This can be useful when trying to find out what is running and also to quickly turn off different playbacks.

Running Playback Window

It can be a window on a screen. This is called **Running Playbacks** and can be **created like any other window**. It is in the "More" tab.



Running playback window with a sequence running cue 1

Each running object is displayed as a pool object and can be interacted with using some commands. For instance, it is possible to use the **Off keyword** to turn off a sequence, timecode, or macro by pressing Off and then tap the object in the window.

This window can show three different types of objects: Sequences, Timecodes, and Macros.

Toggle between them by tapping Sequence, Timecodes, or Macros in the title bar. This is actually a swipe button, so it can also be pressed and swiped out to select one of the options.

It is possible to mask the objects in the window to only display the playbacks triggered by your user. This can be turned On or Off by tapping My Playbacks Only in the title bar.

The last button in the title bar is Sheet Style. This turns On or Off the sheet style for this window. The default style is pool style like the image above. Sheet style could look like this:

Running playbacks window in sheet style - sequence 2 running

Currently, this mode shows fewer details, as it does not display the current cue number for the sequences.

All operations are the same no matter what style is chosen.

The only other option for the **Running Playbacks** window is the possibility to change the font size. These settings can also be found in the **Running Playback Settings** opened by tapping the MA logo in the upper left corner of the title bar.

Off Menu

The other way to see running playbacks is to open a temporary version of the window. This is called the **Off Menu**. It is opened by pressing Off twice:

Off Menu with a sequence running cue 1

This is the same as the **Running Playbacks** window with a few exceptions.

When the menu is in **Pool Style** there are two buttons at the bottom of the pop-up and the pool style has a permanent Off function, meaning that tapping any object instantly turns the object off.

The button called All Sequences Off in the image above changes depending on the settings. This means that it says All Macros Off if the pool displays the macros and similar for timecode. It also changes from **All** to **My** if the pool only displays your sequences or macros. Tapping it does what the button says.

Tap Everything Off to immediately turn all playback off.

In **Sheet Style**, there is an extra button and Off is not the default function. This means that it is possible to select multiple rows in the sheet and then tap the extra Off Selected button to turn off the selection.

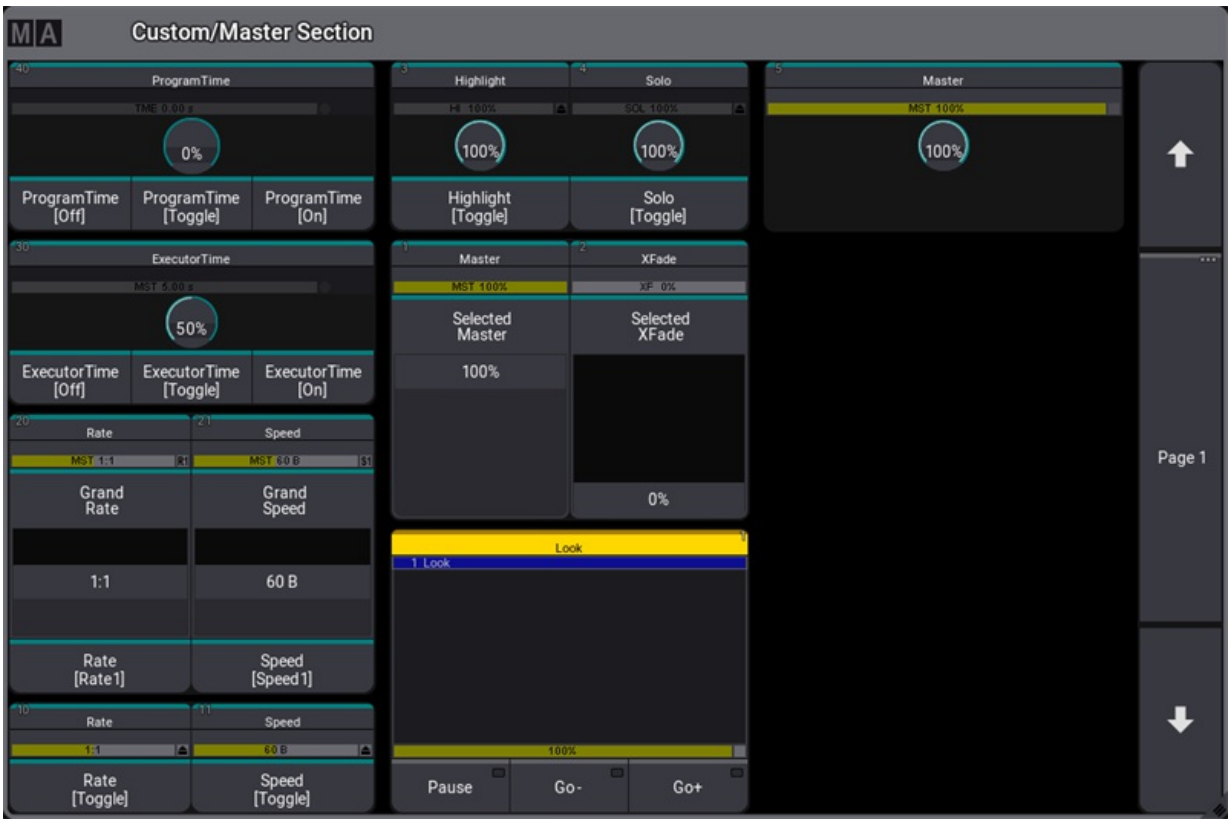
The pop-up can be closed like any other pop-up by tapping the in the upper right corner.

23.4. Special Executors

Special executors include the grand master knob as well as the keys, knobs, faders, encoders, and wheels found in the custom area and master area. By default, they are assigned to control a useful assortment of masters. As with other executors, changing the assignment of special executors allows control of many types of objects including sequences, masters, plugins, groups, macros, worlds, and views.

The master area is always displayed to the right of the playback bar on screen 9. The master area is also displayed to the right of the optional playback bar on screen 2. The custom area is always displayed to the right of the playback bar on screens 10, 11, and 12. The custom area is also displayed to the right of the optional playback bar on screens 3, 4, and 5. For more information on screen allocation, see the **Screen Allocation** topic.

The **Custom/Master Section** window displays the current assignment and status of the special executors in the two custom areas, the master area, and the grand master knob. This window can also include on-screen copies of the special executor encoders, wheels, knobs, and keys; as well as the Go+, Go-, and Pause keys for the selected sequence. Tap the icon in the control bar to open the **Custom/Master Section** overlay temporarily. The **Custom/Master Section** window is also available under the **More** and **All** tabs in the **Add Window** pop-up.



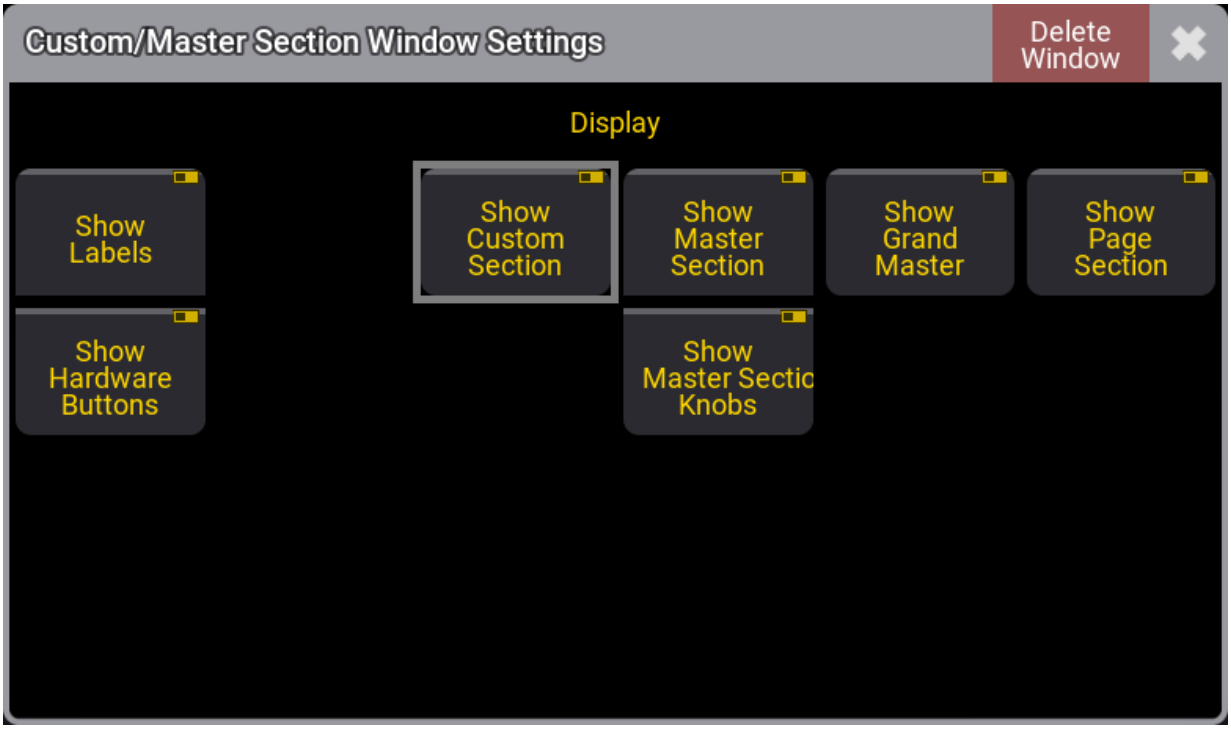
Custom/Master Section window

The Custom/Master Section window contains three main areas:

- Custom area encoders and wheels displayed on the left side of the window when all sections are visible.
- Master area, including the Go+, Go-, and Pause buttons for the selected executor, displayed center of the window when all areas are visible.
- Grand master, displayed on the right side of the window when all sections are visible.

These three main areas can include labels and a visualization of the associated hardware. When both labels and hardware are visible, the label for each special master appears directly above the associated hardware. In addition, page navigation buttons can appear along the right edge of the window.

Tap MA in the upper left corner of the window to open the Window Settings pop-up.



Custom/Master Section Window Settings pop-up

Under the Display tab, tap Show Labels to show or hide the labels for all visible sections. Tap Show Hardware Buttons to show or hide the hardware visualizations for all visible sections. Tap one of the remaining buttons along the top row to show or hide the corresponding area of special executors or page navigation controls.

Hint:
The Show Hardware Buttons and Show Master Section settings must all be enabled in order for the **Default Playback** buttons to be visible.

Hint:
As elements are hidden, the remaining visible elements automatically rescale to fill the available space within the window.

Tap the label of the **Default Playback** area to open the **Edit Sequence** overlay for the selected sequence.

Tap the label of any special executor to open the **Assign Menu** for that special executor. The options available in the **Assign Menu** for a special executor are the same as the options for any other executor. For more information on the assignment process, see the **Assign Object to an Executor** topic.

Special Executor 5 and the Grand Master

By default, the grand master is assigned to special executor 5. Press Assign then press the special executor 5 knob to open the **Assign Menu** for special executor 5. Use the Assign Menu to assign any desired object to special executor 5.

Hint:
Since the hardware associated with special executor 5 consists of only a knob with no additional key, it is possible to assign a normal key function to the press of the knob.

When any object other than the grand master is assigned to special executor 5, the area of the encoder bar that normally shows the grand master adjusts to show both the grand master and a label for the object assigned to special master 5.

Encoder bar showing both sequence 1, assigned to special master 5, and the grand master.

25. Masters

Masters are physical representations of various timing and level overrides that exist in the software.

Masters can be assigned to fader executors or any of the special executors in the **Master section**, **Custom section**, or the **Grand Master**.

They can all be assigned using the Assign Menu. Tap Object in the top-left corner, then Master at the top of the menu to reveal a tree structure separated into the four groups of Masters.

They can also be assigned and adjusted using the **Master keyword**.

For more information on assigning objects to executors, see the **Assign Object to an Executor topic**.

Quick access to all of the selected masters and grand masters is available in the master controls menu. For more information on this menu, see the **Master Controls topic**.

There are no settings for masters, and they have a limited selection of functions available as key assignments.

Please read about the available masters in the following subtopics.

25.1. Selected masters

The selected masters replicate the 11 masters that each executor with a sequence can have assigned.

These masters can be assigned to executors and special executors. They will adjust the levels and timing of the selected sequence.

Read about the functions of the masters in the **Assign Object to an Executor topic**.

The 11 masters are:

- **Master 1.1 'Master'**
- **Master 1.2 'XFade'** - this is also called Crossfade
- **Master 1.3 'XFadeA'** - this is also called CrossfadeA
- **Master 1.4 'XFadeB'** - this is also called CrossfadeB
- **Master 1.5 'Temp'**
- **Master 1.6 'Rate'**
- **Master 1.7 'Speed'**
- **Master 1.8 'Highlight'**
- **Master 1.9 'Lowlight'**
- **Master 1.10 'Solo'**
- **Master 1.11 'Time'**

25.2. Grand masters

The Grand Masters are a selection of 14 different masters that can be assigned to executors and special executors.

They are:

- **Master 2.1 'Grand'**
- **Master 2.2 'World'**
- **Master 2.3 'Highlight'**
- **Master 2.4 'Lowlight'**
- **Master 2.5 'Solo'**
- **Master 2.6 'Rate'**
- **Master 2.7 'Speed'**
- **Master 2.8 'ProgramTime'**
- **Master 2.9 'ExecutorTime'**
- **Master 2.10 'ProgramSpeed'**
- **Master 2.11 'Blind'**
- **Master 2.12 'SoundOut'**
- **Master 2.13 'SoundIn'**
- **Master 2.14 'SoundFade'**

Grand is the grand master. This has a dedicated virtual fader, but can also be assigned to any executor or special executor.

The **World** master functions like a grand master but only for fixtures included in the selected world. For more information on worlds, see the **Worlds and Filters topic**.

The **Highlight** master inhibits the effect of the highlight function on selected fixtures. It offers proportional control over dimmer as well as all other highlighted attributes. It also works as an inhibitor for the lowlight function. For more information on highlight, see the **Highlight keyword topic**.

The **Lowlight** master inhibits the effect of the lowlight function on highlighted fixtures. It offers proportional control over all lowlighted attributes. For more information on lowlight, see the **Lowlight keyword topic**.

The **Solo** master inhibits the effect of the solo function on selected fixtures. It offers proportional control over dimmer attributes. For more information on solo, see the **Solo keyword topic**.

The **Speed** master controls the overall speed of any phaser running in the console, whether it is playing back from the programmer or a sequence.

The **ProgramTime** master modifies the fade time used by the programmer. For more information on program time, see the **Time Control topic**.

The **ExecutorTime** master overrides stored cue timing. For more information on exec time, see the **Time Control topic**.

The **ProgramSpeed** master controls the overall speed of any phaser running from the programmer.

The **Blind** master inhibits the effect of the blind function. It offers proportional control over dimmer as well as all other attributes. For more information on blind, see the **Blind keyword topic**.

The **SoundIn** master inhibits incoming audio signals. For more information about using sound input, see the **Sound Window** topic.

The **SoundFade** master adjusts how quickly the console responds to changes in incoming audio signals. For more information about using sound input, see the **Sound Window** topic.

25.2.1. Time Control

When enabled, the Program Time and Executor Time masters can override playback timing within the programmer and executors, respectively, with a time range of 0 seconds to 10 seconds. They can be assigned to executors as well as special executors in the **Master area**, **Custom area**, or the **Grand Master**. This gives physical control of their activation and level. Quick access to these masters is available in the master controls menu. For more information on this menu, see the **Master Controls topic**.

Program Time

Program Time adds a fade time to changes of attribute values in the programmer. Set the master to the desired value and activate Program Time by pressing On or Toggle on an executor assigned to the program time master, or tap Prog Time in the master controls menu so there is a green marker. The set time will be used for all value changes in the programmer except for those made by turning an encoder or the dimmer wheel.

To deactivate program time, press Off or Toggle on an executor assigned to the program time master, or tap Prog Time again in the master controls menu. The green marker will disappear.

Hint:
Recalling presets that include individual fade time and delay time will continue to use their stored times.
Presets with no individual fade time will respect the program time setting.

Executor Time

Executor Time overrides cue fade times during sequence playback, regardless of whether the sequence is running from an executor or directly from the sequence pool. Set the master to the desired value and activate Executor Time by pressing On or Toggle on an executor assigned to the executor time master, or tap Executor Time in the master controls menu so there is a green marker. Cues stored with only basic timing will change values using the specified executor time. Individual fade and delay and times will continue to playback as stored. Cues with follow or time triggers will continue to use the stored triggers.

To deactivate executor time, press Off or Toggle on an executor assigned to the executor time master, or tap Executor Time again in the master controls menu. The green marker will disappear.

If the Use Executor Time option in the sequence settings menu is enabled, the sequence will follow the executor time master. Disabling Use Executor Time allows all of the cues in the sequence playback using their stored timing. For more information on the sequence settings menu, see the **Sequence Settings** topic.

25.3. Speed Masters

15 different Speed Masters can be assigned to executors and special executors.

This group of masters also includes the BPM Master. This master adjusts the expected speed of the incoming audio. The console will adjust to the actual detected Beats Per Minute closest to this setting. For more information about using sound input, see the **Sound Window** topic.

Speed masters can be assigned to sequences. This makes it possible to sync multiple sequences to the same speed.

Read more about assigning speed masters to executors in the **Assign Object to an Executor** topic. Read about assigning a Speed Master to a sequence in the **Sequence Settings** topic.

25.4. Playback Masters

50 different Playback Masters can be assigned to executors and special executors.

Playback Masters can be assigned as playback settings of sequences. The masters function as an inhibitive dimmer control for the assigned sequences. All sequences with the same playback master assigned can be simultaneously inhibited by that master.

Read more about assigning playback masters to executors in the **Assign Object to an Executor** topic. Read about assigning a playback master to a sequence in the **Sequence Settings** topic.

26. Recipes

Recipes can be a very useful tool for touring shows or when the show changes a lot.

Recipes can be stored in cue parts and presets.

A recipe contains one or multiple recipe lines describing what should happen based on a set of information. The recipe can be used to "cook" values into the cue part, preset, or programmer. This cooked data is marked by a small pot icon and the cooked data can easily be removed again if needed.

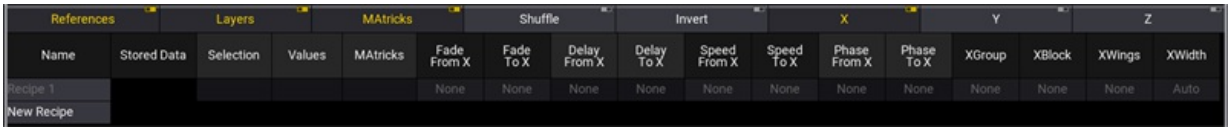
A recipe line can contain information about a selection of fixtures, group, preset, MAtricks, individual fade, delay, speed, and phase values.

Values from recipes can be combined with conventionally stored values.

The flexibility in the recipe system allows for a variety of uses. The recipes could be used to:

- Create recipe presets for groups with references to other presets for a flexible fixture setup.
- Create "template" presets with ranged values that can be applied to a flexible selection of fixtures.
- Cues that contain a recipe on how different elements create the desired look.

This is a recipe line:



Empty recipe line

There is a top bar with toggle buttons that can show or hide different elements of the recipe.

- **References**:
  These are the columns for referenced data.
  - **Selection**:
    This is the selection of fixtures using this recipe line. If the selection is a group, then the Group ID is displayed. If the selection is from the programmer, then it says **<Recipe>**.
  - **Values**:
    This is the value reference used in the recipe line. If the value is a preset, then the preset ID is shown. If the value is from programmer values, then it says **<Recipe>**.
  - **MAtricks**:
    This is a reference to an existing MAtricks pool object. Having a reference to an existing MAtricks adds referenced values in the MAtricks columns.

- **Layers**:
  These are the timing layers. X, Y, or Z also needs to be active for any of the layer columns to be shown.
  - **Fade From / Fade To**:
    The two fade values allow spreading the fade time over a range.
  - **Delay From / Delay To**:
    The two delay values allow spreading the delay time over a range.
  - **Speed From / Speed To**:
    The two speed values allow spreading the speed values over a range.
  - **Phase From / Phase To**:
    The two phase values allow spreading the phase values over a range.
- **MAtricks**:
  These are the MAtricks columns. X, Y, or Z also needs to be active for any of the layer columns to be shown.
  - **Group**:
    The number of groups the selection is split into.
  - **Block**:
    The number of fixtures blocked together.
  - **Wings**:
    The number of wings the selection is split into.
  - **Width**:
    This changes the width of the selection in the **Selection Grid**.
- **Shuffle**:
  These are the shuffle columns. X, Y, or Z also needs to be active for any of the layer columns to be shown.
  - **Shuffle**:
    This value can be set to shuffle the selection order.
  - **Shift**:
    This value can be set to shift the selection in the selection grid.
- **Invert**:
  These are the Invert columns. X, Y, and Z have invert columns as well and might need to be active to show the desired columns.
  - **Invert Style**:
    This defines if Invert is applied to Pan, Tilt, Pan and Tilt, or All attributes.
  - **Invert, Inv, InvB, InvG, InvW**:
    These are Yes/No inverting settings for each of the MAtricks elements.
- **X**, **Y**, and **Z**:
  Each of the grid axes has columns. If the recipe only uses one or two axes, then the columns can be limited to only show the relevant columns.

Besides the columns mentioned above, there is also the **Name** column that can be used to give the recipe line a name. Finally, the **Stored Data** column shows the type of values (Universal, Global, or Selective) stored if it is from the programmer.

Hint:
The MAtricks and Shuffle settings are described in detail in the **MAtricks and Shuffle section**.

Recipes do not need to have information in all columns, but they can have information in all. They often only have information in a few columns.

Read the **Cue Recipe** and **Preset Recipe** topics for details on how to create and use each of the two.

27. Phasers

Phasers are the effects and chasers of grandMA3, creating dynamic output from a single preset or cue.

Phasers change the output for attributes using a set of information in two or more steps. A normal, static cue or preset only contains one step of information. Adding additional steps creates a phaser.

Example

A phaser has two steps with an absolute dimmer value of 100% in step 1 and an absolute dimmer value of 0% in step 2. With all other phaser layers at their default values, the resulting output is a smooth, repeating fade of the dimmer output from 100% to 0% and back to 100%.

Phaser Layers

Each phaser step can include attribute value information. These values can be an absolute value (for instance, a dimmer value of 50%) or a relative value (for instance, a dimmer value of -20%). Steps can contain both absolute and relative values simultaneously.

Each phaser step also includes additional layers, which define the overall width of the step, the percentage of the step dedicated to transitioning to the new value, and the amount of acceleration and deceleration used when changing to the new value.

Each phaser additionally includes layers, which affect all steps of the phaser. Each attribute of each fixture can have its own speed, phase, and measure value in a phaser, but these values will be the same for all steps in each phaser.

The optional Measure layer defines the number of beats in the repeating phaser loop. For example, a phaser with a measure of 4 beats and a speed of 120BPM will repeat every 2 seconds.

Hint:
When using the Measure layer, multiple combinations of step widths can produce the exact same result. In this case, it may be easiest to think of the width as a percentage of one beat in the measure.

All phaser layers, as well as a step selection tool, appear in the **Encoder Toolbar** any time attributes are mapped to the encoders. Phaser layers are indicated by a dark purple color. Phaser layers behave just like other attribute layers when activating, deactivating, storing, and clearing. For more information about attribute encoders and layers, see the **Operate Fixtures** topic. The Phaser Editor provides a graphical view and powerful manipulation of phaser layer information.

Steps

The grandMA3 offers multiple tools and methods for creating, selecting, and deleting phaser steps.

One intuitive method to quickly create steps based on presets involves the Step key. To create steps using this method, press and hold the Step key, then tap a preset for each desired step.

Example

To create a 3-step color phaser from white to red to blue:

1. Select the desired fixtures.
2. Press and hold the Step key.
3. Tap the white color preset.
4. Tap the red color preset.
5. Tap the blue color preset.
6. Release the Step key.

When using this method, the software will only create the next step when tapping a new preset, which contains any of the same attributes as the current step. This behavior allows each step to reference multiple presets as long as those presets apply to different attributes.

Example

To create a 2-step phaser, where each step contains both color and position presets:

1. Select the desired fixtures.
2. Press and hold the Step key.
3. Tap the desired color preset for step 1.
4. Tap the desired position preset for step 1. This preset is added to step 1.
5. Tap the desired color preset for step 2. Step 2 is automatically created.
6. Tap the desired position preset for step 2. This preset is added to step 2.
7. Release the Step key.

The Step key can also quickly change which preset is referenced by a step.

Example

Start with the 3-step color example above, but afterward, change the blue step to yellow:

1. Follow all directions in the 3-step color example above.
2. Press and release Step
3. Press and release 3
4. Tap the yellow color preset.

Delete specific steps from the programmer using standard syntax with the Delete keyword. For example, to delete step 3, type:

Delete all steps in the programmer by deleting the **programmer**.

An easy way to do this (especially on grandMA3 onPC) is to tap Delete Steps in the **At Overlay**.

Stomp

The stomp function stops a phaser, running either in the programmer or from a playback, and resolves it to a single, static step. The console calls and activates the last static output value for each stomped attribute into step 1 of the programmer. All other steps of the stomped phaser are discarded.

Sync

The Sync option ensures that phasers recall with predictable timing and phase offsets.

When calling phasers into the programmer, the Sync option is available in the **Encoder Bar**.

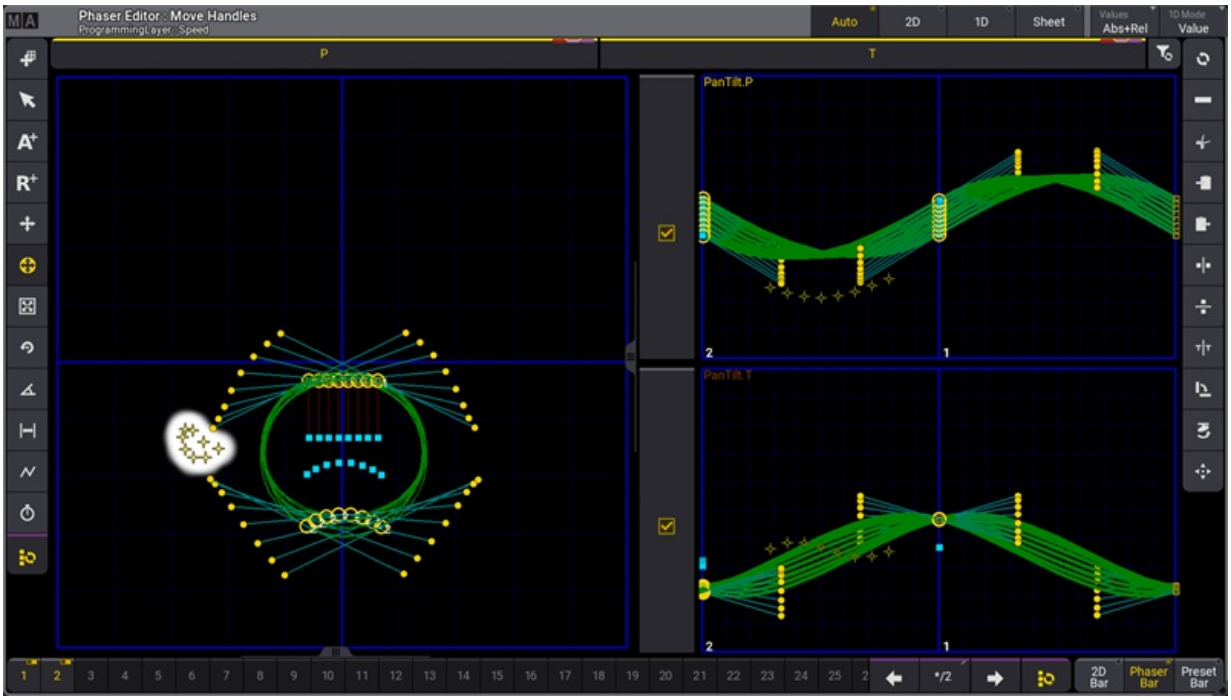
When recalling phasers stored in cues, the Sync option is available as a setting for each cue part in the Sequence Sheet. For more information about settings in cues and cue parts, see the **Look at Cues and Sequences** topic.

27.1. Phaser Editor

The **Phaser Editor** is a diverse tool for manipulating phasers.

It offers multiple means of visualizing running phasers as well as tools for dynamically creating and editing phasers. It can also be used as a simple trackpad to adjust the position attributes of fixtures.

Open a storable **Phaser Editor** window using the **Add Window pop-up**. Alternatively, open a temporary version of the **Phaser Editor** by tapping the Phaser button in the **Encoder Bar**.



Create and manipulate phaser steps in the phaser editor

The image above shows the Phaser Editor in the Auto view mode. Choose between multiple available view modes using the radio buttons in the title bar or the drop-down button in the Window Settings pop-up. Read below for images and descriptions of the other available view modes.

The blue grid on the left represents the entire pan range (horizontally) and the entire tilt range (vertically). This is called the 2D layout.

The smaller blue grids on the right are 1D layouts. Each attribute with more than one step in the phaser is displayed in its own 1D layout. The bold, vertical, blue lines in the 1D layouts represent beats, based on the speed of the phaser. The example above shows pan and tilt attributes, each with two steps.

When fixtures are selected, they are visualized by small yellow cross-hairs. The beams can be represented in the 2D layout by toggling the Show Beams button in the Window Settings pop-up. When Show Beams is enabled, the intensity and color of each fixture are visualized as a larger circle behind the corresponding yellow cross-hair.

Adding points in the 2D layout adds steps to the programmer. The first point moves the fixtures to the specified position. Each subsequent point adds another step to the programmer.

When a step contains both an Absolute point (shown as a yellow or cyan-filled square) and a Relative point (shown as a hollow, yellow or cyan square), a thin red line connects the relative point and the related absolute point. The cyan-colored points show that the value is from a preset. Yellow-colored points are hard values from the programmer.

Selected points have a yellow circle around them.

A green line describes the path of the fixtures.

A row of buttons immediately below the title bar includes one button for each attribute included in the active phaser. These buttons function as a direct link to the **At Filter**. Phaser adjustments apply to any attributes with a yellow bar and yellow text. Attributes with a grey bar and grey text will ignore adjustments. Tap the button immediately to the right of the last attribute button in this bar to enable or disable all attributes in the bar.

Hint:
If any or all attributes in the bar are disabled, the first tap of the button to the right of the attributes enables all displayed attributes.
If all displayed attributes are already enabled, tapping this button disables them all.

The tool buttons on the left side are mouse or touch tools, and the tool buttons on the right are operational functions and shortcuts - read more below. The right side tool buttons might change depending on the selected tool on the left side.

Step Bar

The Step Bar appears across the bottom of the Phaser Editor. The Step Bar is a quick step selection tool. This element is optional in the Phaser Editor window, and it is enabled by default. Tap the Step Bar button in the Window Settings pop-up to show or hide the Step Bar. A storable window version of the Step Bar is also available in the **More** tab in the **Add Windows pop-up**.

Step Bar window with step one selected

Each step has a small square, which displays a status of empty, deselected, or selected.

Empty steps are steps that do not have any values. They have a dark background color.

Deselected steps have values in them, but they are not selected and are not affected by step-specific adjustments. Deselected steps have a light gray background color and an On/Off toggle icon in the upper-right corner.

Selected steps have values in them and are affected by step-specific adjustments. Selected steps have yellow text and a yellow On/Off toggle icon in the upper-right corner.

Tapping any of the steps toggles the selected status.

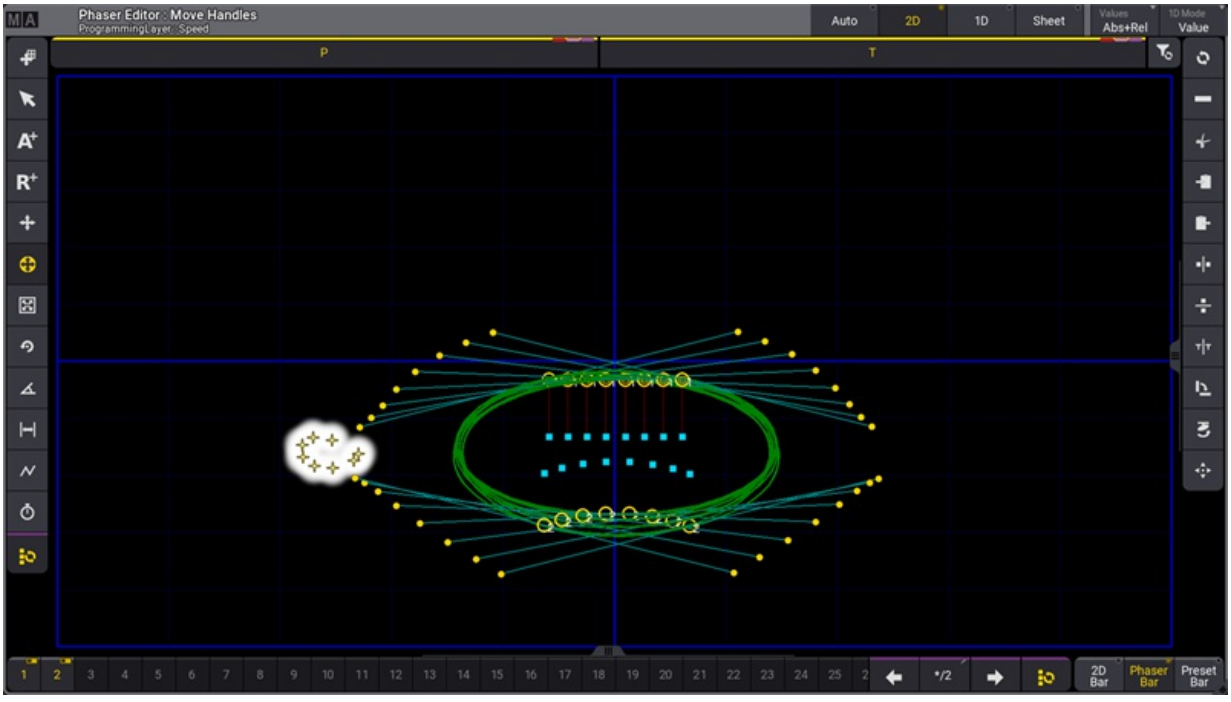
Additional controls appear on the right side of the Step Bar. These controls include left and right arrows used to change between single steps. An informative field appears between the arrows, showing which step is currently selected and the number of steps that contain values in the active phaser. If multiple steps are selected, the first selected step is shown with two dots after it (for example, 1../3). If all steps are selected, the first number is replaced with an asterisk (for example, */3).

Tap the button to the right of the right arrow in the Step Bar to select all steps. Tapping this selects all not-empty steps if only one step is currently selected. If more than one step is already selected, then only step 1 is selected. Tapping the button executes this command: **Step Toggle Executor**.

The Step Bar in the Phaser Editor window includes three radio buttons. Tap one of these buttons to call up the desired encoder toolbar. The Preset Bar opens the standard attribute **Encoder Toolbar** with access to all layers and all attributes. See below for Images and descriptions of the 2D Bar and Phaser Bar.

Restriction:
The storable window version of the Step Bar does not include the encoder toolbar radio buttons.

2D View Mode



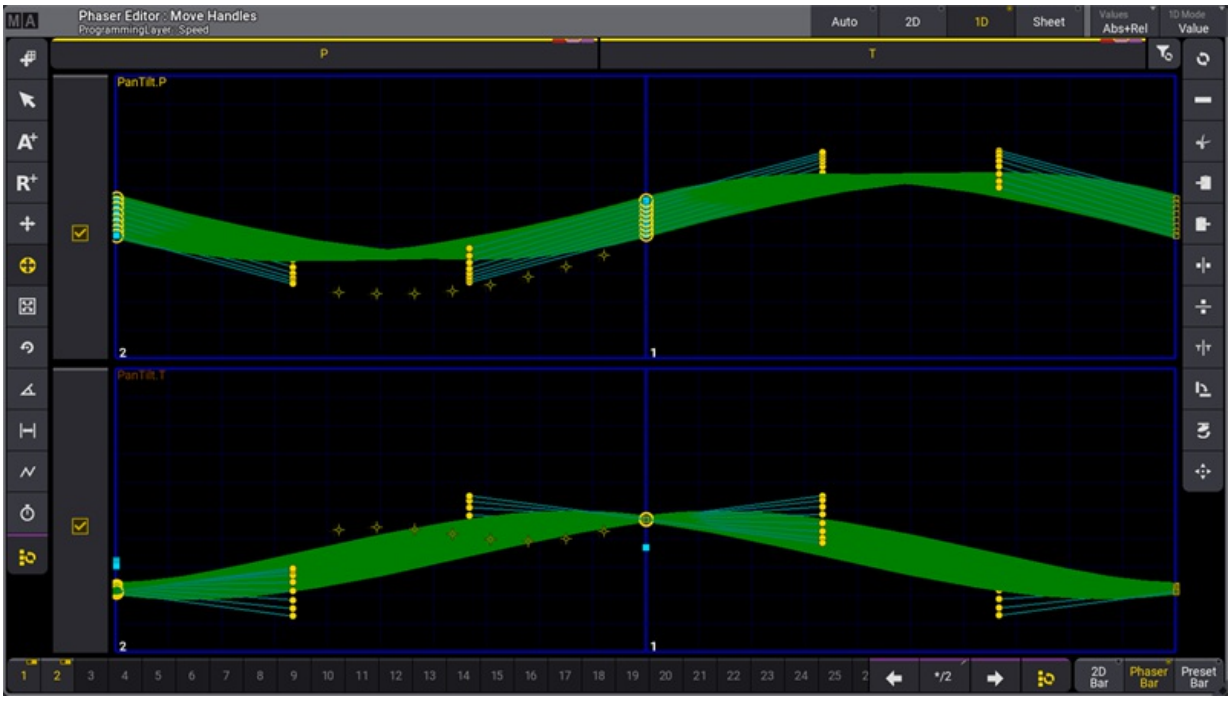
Phaser Editor in 2D View Mode

The 2D view mode removes other layout elements and expands the 2D layout to cover the main area of the window. This view mode is helpful when working with position phasers.

The Values button in the title bar allows or prevents the display and editing of absolute points or relative points. Tap to cycle through the available options or tap and swipe to open a list of available options. Options include:

- None - Displays and allows editing only of handles.
- Absolute - Displays and allows editing of absolute points and their handles.
- Relative - Displays and allows editing of relative points and their handles.
- Abs+Rel - Displays and allows editing of absolute and relative points, as well as their handles.

1D View Mode



Phaser Editor in 1D View Mode showing Values

The 1D view mode removes other layout elements and expands the 1D layouts to cover the main area of the window. This view mode displays a layout for every attribute included in the active phaser.

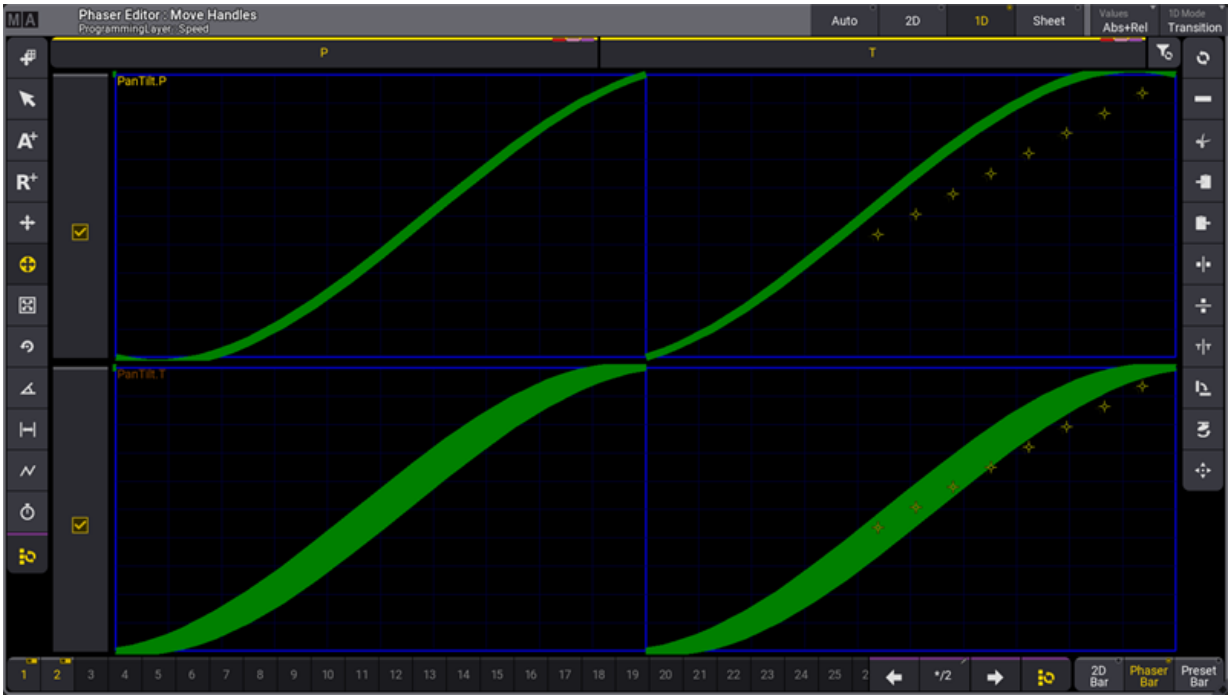
To change the height of the attribute lines, tap the MA logo in the upper-left corner of the window, then tap LineHeight in the **Window Settings** menu, enter the desired height, and press Please.

The checkboxes to the left of the attribute lines represent the status of those attributes in the **At Filter**. These checkboxes relate directly to the attribute buttons across the top of the Phaser Editor. A yellow checkmark indicates that adjustments are allowed on the corresponding attribute. An empty, grey box indicates that

adjustments are not allowed.

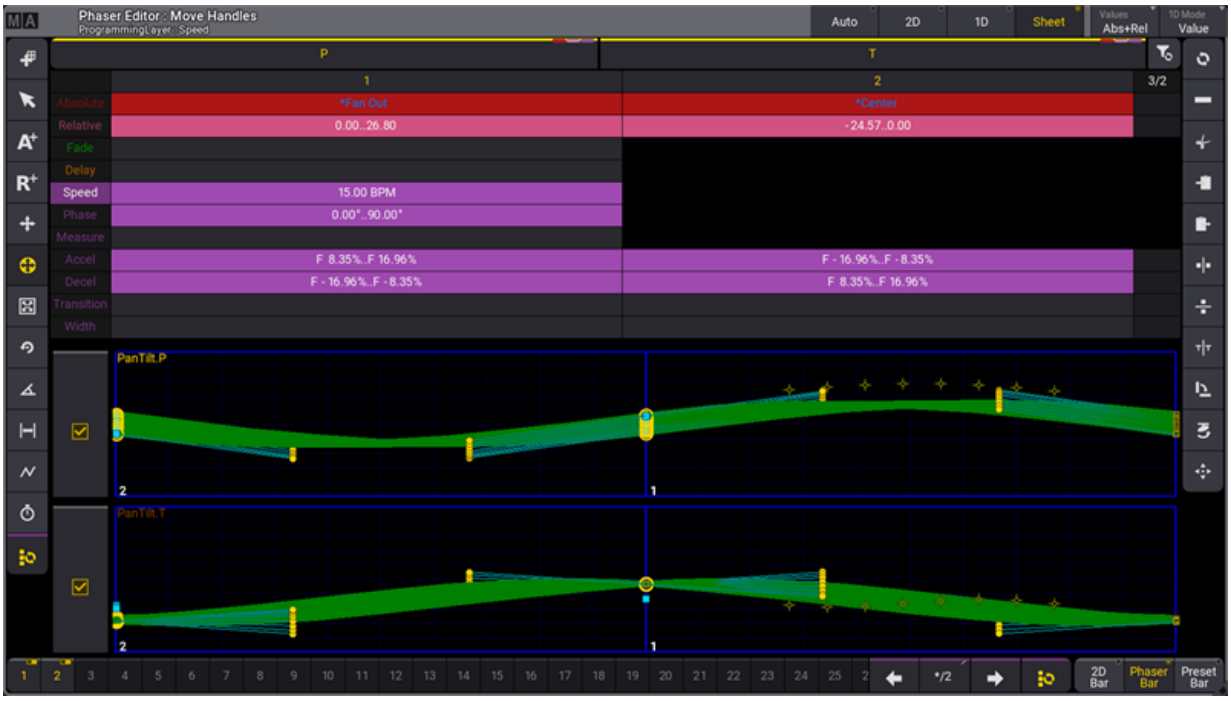
Tap the 1D Mode button at the right end of the title bar to toggle between the two available 1D drawing modes. These modes change the vertical scale of the 1D layouts.

- Value - In this mode, the vertical scale of each 1D layout equals the full range of values available for the displayed attribute.
- Transition - In this mode, the bottom of the vertical axis represents the beginning value for each step, and the top represents the ending value for the step.



Phaser Editor in 1D View Mode showing Transitions

Sheet View Mode

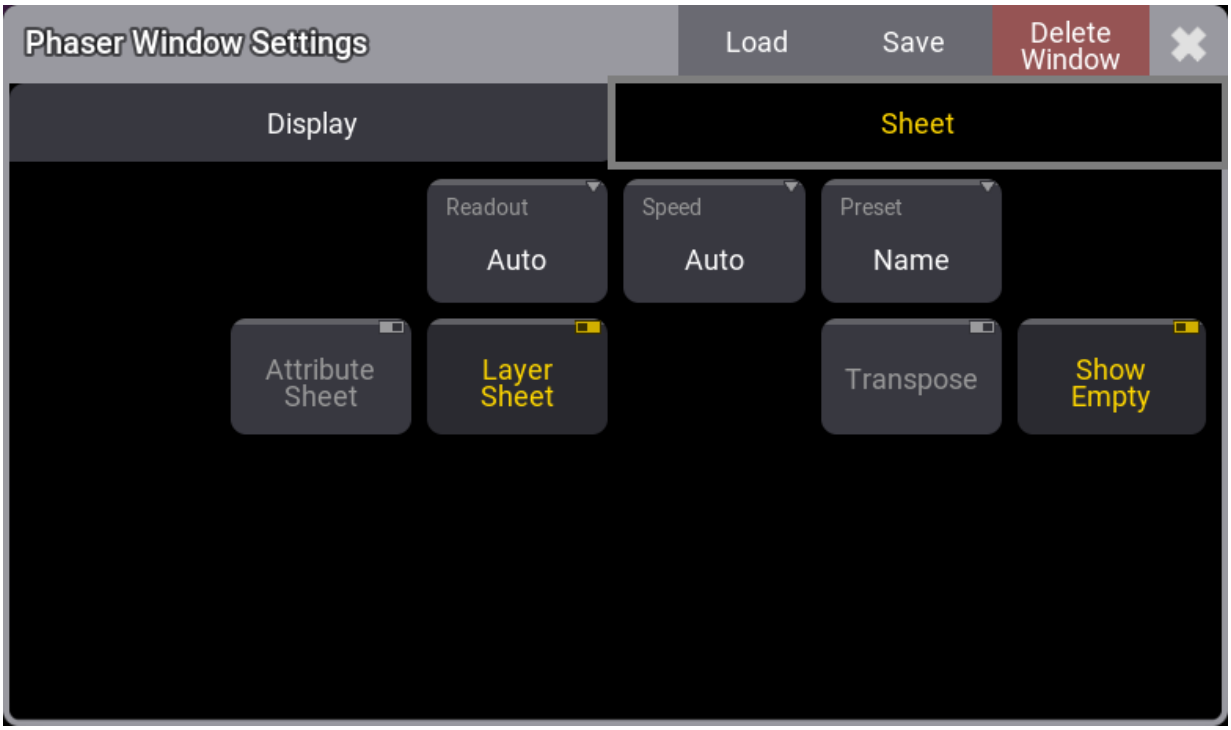


Phaser Editor displayed using Step view mode

The step view mode displays phaser steps using a configurable spreadsheet format similar to the fixture sheet. Beneath all of the relevant spreadsheet data, this mode also includes 1D layouts for each attribute included in the phaser.

The default configuration of the sheet view mode arranges phaser steps as columns and layers as rows. Layers that affect all steps of the phaser appear only under step 1. Each cell in the sheet displays an overview of all values for the layer in that step. In cases where the cell includes multiple values or presets, the sheet displays the lowest and highest values in the cell separated by two dots.

The Sheet tab of the Window Settings pop-up includes some common **Window Settings** as well as some that are specific to the Sheet view mode of the Phaser Editor.



Sheet Tab of the Phaser Window Settings Pop-up

- Attribute Sheet - Tap to show or hide individual rows for each attribute included in the phaser. If Layer Sheet is enabled, each layer will include all of these individual layers.
- Layer Sheet - Tap either show all layers simultaneously or hide most layers, showing only the layer that is currently accessible on the attribute encoders.

Tools

The tool buttons on the left side are:

Phaser Editor left side toolbar

- **1 - Move Area**:
  Moves the entire blue pan/tilt square to show relative values that can be outside the position range.
- **2 - Select**:
  Selects single steps or uses lasso selection for multiple steps. The selection can be filtered to **Absolute** and **Relative** steps by toggling the respective buttons in the title bar.
- **3 - Add Absolute**:
  Adds a step using absolute value. The encoder bar follows the absolute layer. A filled rectangle represents absolute values. The **Single Step Mode** is automatically activated as long as you hold a finger or the mouse cursor in the editor while adding an absolute point.
- **4 - Add Relative**:
  Adds a step using a relative value. The encoder bar follows the relative layer. An empty rectangle represents relative values. The **Single Step Mode** is automatically activated as long as you hold a finger or the mouse cursor in the editor while adding a relative point.
- **5 - Move Point**:
  Moves selected steps. It is also possible to use the Align function. The **Single Step Mode** is automatically activated as long as you hold a finger or the mouse cursor in the editor while moving a single point.
- **6 - Move Handles (The yellow highlight shows this is the current tool.)**:
  Move handles of selected steps. It is also possible to use the **Align** function. Each point/step has two handles to influence acceleration and deceleration at the same time. Each handle has a cyan line connected to the point it controls. Move both handles of a point by touching the point and dragging in the desired direction. A handle can be moved individually by touching and moving the handle (small yellow circle).
- **7 - Change Size**:
  Enlarges/reduces all selected steps evenly from the center point of the selection. It is also possible to use the **Align** function.
  Pressing MA while changing the size enlarges/reduces all selected steps from the center point of the selection on the horizontal or vertical axis.
- **8 - Change Rotation**:
  Rotates the selected steps around the center of the selection. It is also possible to use the **Align** function.
  Pressing MA while changing the rotation, rotates the selected steps around the center of the selection in 5-degree increments.
- **9 - Change Phase**:
  Opens an overlay with yellow dots symbolize the value of the phase for each fixture. Here the phase value can be shifted independently from the selected steps for the selected fixtures. The **At** filter can be used to move phases for individual attributes. It is also possible to use the **Align** and the **Transition** mode functions. If several attributes are displayed along with different phases, the selected feature groups are displayed in a brighter color. If the fixtures are allocated to grid positions, the values of the phases can be vertically or diagonally aligned. Selecting this also changes the functions on the right side to quick selections for alignment. This changes the tool buttons on the right side to give quick access to standard phase values and a Reset and Invert button.

- **10 - Change Width:**
  The width can be changed with this tool. It changes the functions on the right side to shortcuts of different percentages. If all steps are selected and the Speed shortcut is tapped, then the movement speed is equalized between the steps. Default is that each step has the same width with no regard to the distance the fixtures are moving in the step. Equalizing the step width makes the fixture move at the same speed by adjusting the width. There is a Reset shortcut that resets the width. Default is a width of 100%. This means that each step has the same width with no regard to the travel distance of the fixtures in each step.
- **11 - Select Form:**
  Selecting a form overwrites the values of the transition layer, accel layer and, decel layer to create the desired effect. Available forms are listed in the toolbar on the right side and include Rectangle, Sawtooth, Sine and Cosine.
- **12 - Change Speed:**
  Tap to display various speed tools in the right side toolbar. Available speed tools include functions which multiply or divide the current speed of the phaser by a given amount. Special speed options include:
  - **Loop:** Tap Loop to multiply the current phaser speed by the number of steps currently in the phaser.
  - **Fixture:** Tap Fixture to divide the current phaser speed by the number of fixtures in the current selection.
- **13 - Select all Steps**:
  This is not a tool like the others. Tapping this toggles between all steps selected or the selection of single steps. It keeps the previously selected tool active.

Hint:
Some tools don't require the user to precisely touch points in the 2D and 1D layouts in order to adjust them. Instead, make sure the desired steps are enabled in the **Step Bar**, and use the blue grid areas in the **Phaser Editor** as trackpads to make the desired adjustments.

The standard operational functions on the right side are:

- **1 - Reset**:
  Reset spline of the selected steps.
- **2 - Delete**
  Deletes the selected steps.

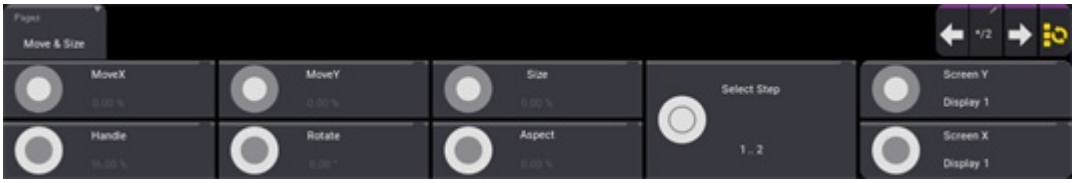
- **3 - Cut Programmer**:
  Removes all programmer values in the currently selected fixtures and moves them to the clipboard.
- **4 - Copy Programmer**:
  Copies all programmer values in the currently selected fixtures and moves them to the clipboard.
- **5 - Paste Programmer**:
  Inserts all values from the clipboard to the currently selected fixtures in the programmer.
- **6 - Mirror X**:
  Mirrors all selected points on the x-axis.
- **7 - Mirror Y**:
  Mirrors all selected points on the y-axis.
- **8 - Mirror Time**:
  Swaps the order of the steps. Select all steps first. This is the same as reversing the direction of the movement.
- **9 - Swap XY**:
  Swaps the X and Y axes of the selected steps.
- **10 - Flip**:
  Flips the position of the selected fixtures.
- **11 - Reset Zoom**:
  Resets the area with the blue square.

Phaser Encoder Toolbar

Depending upon the type of encoder toolbar selected with the radio buttons in the bottom-right corner of the **Phaser Editor**, the encoder toolbar changes to display helpful tools, which are specific to the manipulation of Phaser data.

2D Bar

To enable the 2D Bar, tap 2D Bar in the bottom-right corner of the Phaser Editor.



2D Phaser Encoder Bar

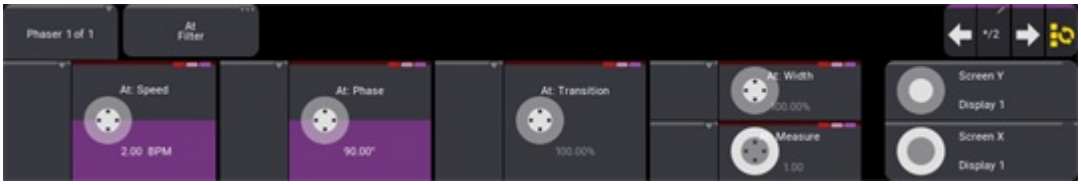
This encoder toolbar includes the following encoder controls:

- **MoveX:** Turn the inner encoder to adjust the X position in the 2D Phaser grid of all steps currently enabled in the **Step Bar**.
- **MoveY:** Turn the inner encoder to adjust the Y position in the 2D Phaser grid of all steps currently enabled in the **Step Bar**.
- **Size:** Turn the inner encoder to adjust size of all steps currently enabled in the **Step Bar**. This allows encoder access to the **Change Size** tool mentioned above.

- **Handle:** Turn the outer encoder to adjust the handle length of all steps currently enabled in the **Step Bar**. This allows encoder access to the **Move Handles** tool mentioned above.
- **Rotate:** Turn the outer encoder to rotate all steps currently enabled in the **Step Bar**. This allows encoder access to the **Change Rotation** tool mentioned above.
- **Aspect:** Turn the outer encoder to adjust the aspect ratio of all steps currently enabled in the **Step Bar**. This tool simultaneously adjusts positions of points as well as length and direction of handles in order to create a wider or taller version of the current 2D form.
- **Select Step:** Turn the inner or outer encoder to select individual steps. Press the inner encoder or the outer encoder key to toggle between selecting all steps and a single step.

Phaser Bar

To enable the Phaser Bar, tap Phaser Bar in the bottom-right corner of the Phaser Editor.



Phaser Bar

This encoder toolbar allows for adjustment of **Speed**, **Phase**, **Transition**, **Width,** and **Measure**; based upon the current fixture and step selection, as well as the current status of the **At Filter**.

27.2. Create a Sinus Dimmer Phaser

This topic details a couple of examples of how to create a sinus dimmer phaser using keys and buttons as well as the phaser editor and encoder bar. There are many ways to reach the same goal in grandMA3. Many of the steps in the examples below are interchangeable while producing identical results. These examples uses a combination of keys and the calculator interface. If you are working with the onPC software, it might be easier to type commands into the command line rather than use the on-screen keys.

It is recommended to read the **Phasers topic** and the **Phaser Editor topic** before this topic.

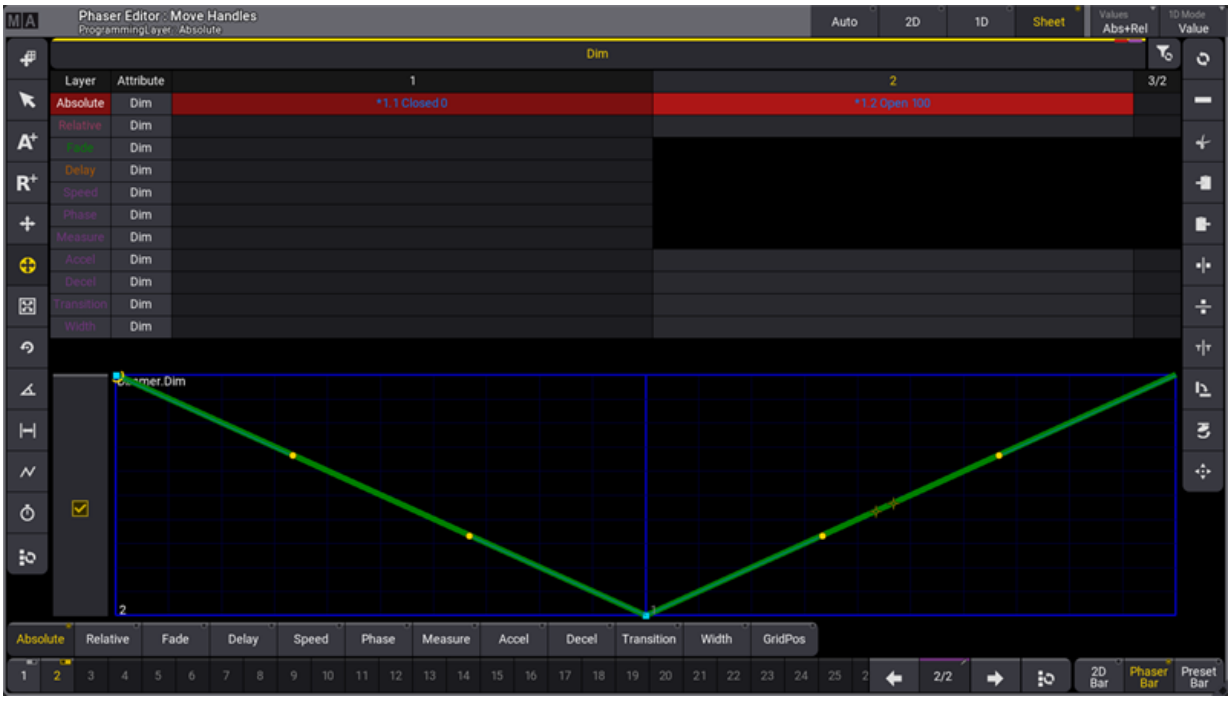
Requirements:

- Have a show with some lights patched.
- An open **Phaser Editor** window with the Step Bar enabled is recommended, though it is also possible to use the temporary Phaser Editor.
- Arranging the fixtures in the 3D window can be useful, but it is not a requirement.
- As it is highly recommended to use presets while programming, an open Dimmer preset pool is recommended with presets for dimmer values of 100% and 0%.

Quickly Create Steps and Apply Adjustments with the Phaser Editor

1. Select the desired fixtures. For example: Fixture Thru Please
2. Choose a dimmer preset for step one: Press and hold Step and tap the preset with the 0% dimmer value.
3. Choose a dimmer preset for step two: While still holding Step, tap the preset with the 100% dimmer value.

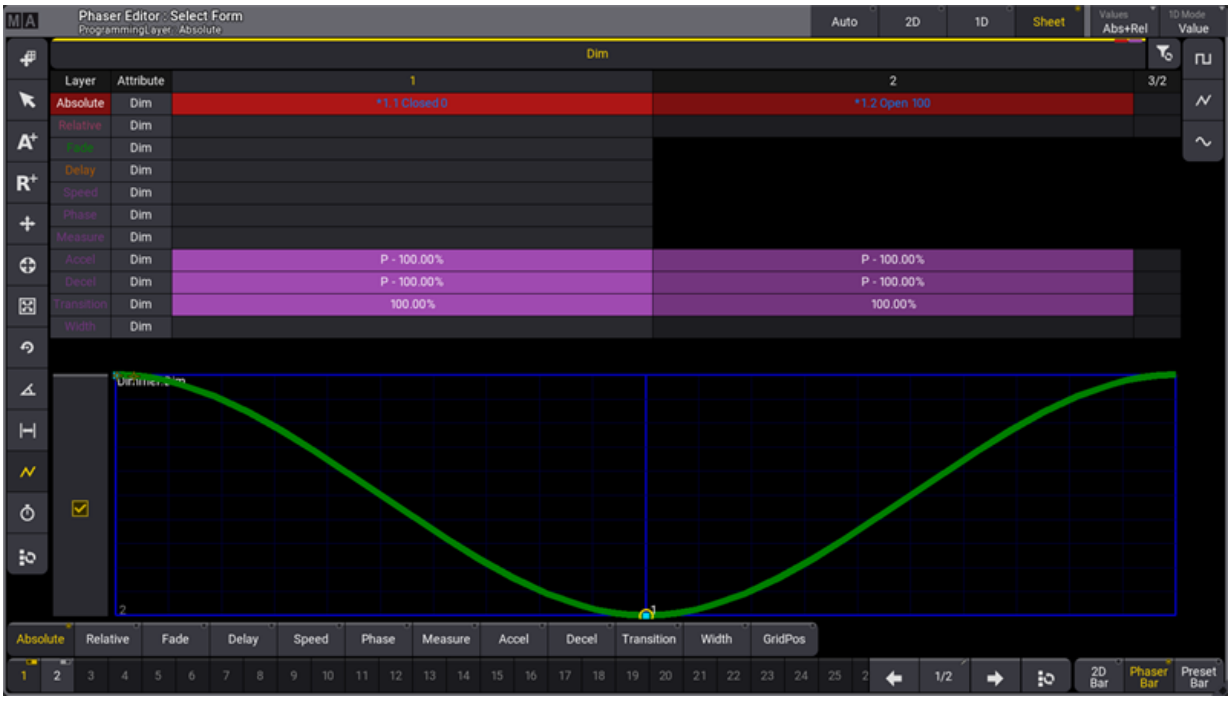
This is the base for the phaser. There are now two steps with a dimmer preset of 0% in the first step and a preset of 100% in the second step. All of the fixtures are changing together from step one to two with a linear fade (not yet using a sine curve). The Sheet view mode of the **Phaser Editor** shows:



Linear Dimmer Phaser with No Phase Distribution

Adjust the curve of both steps using tools in the Phaser Editor:

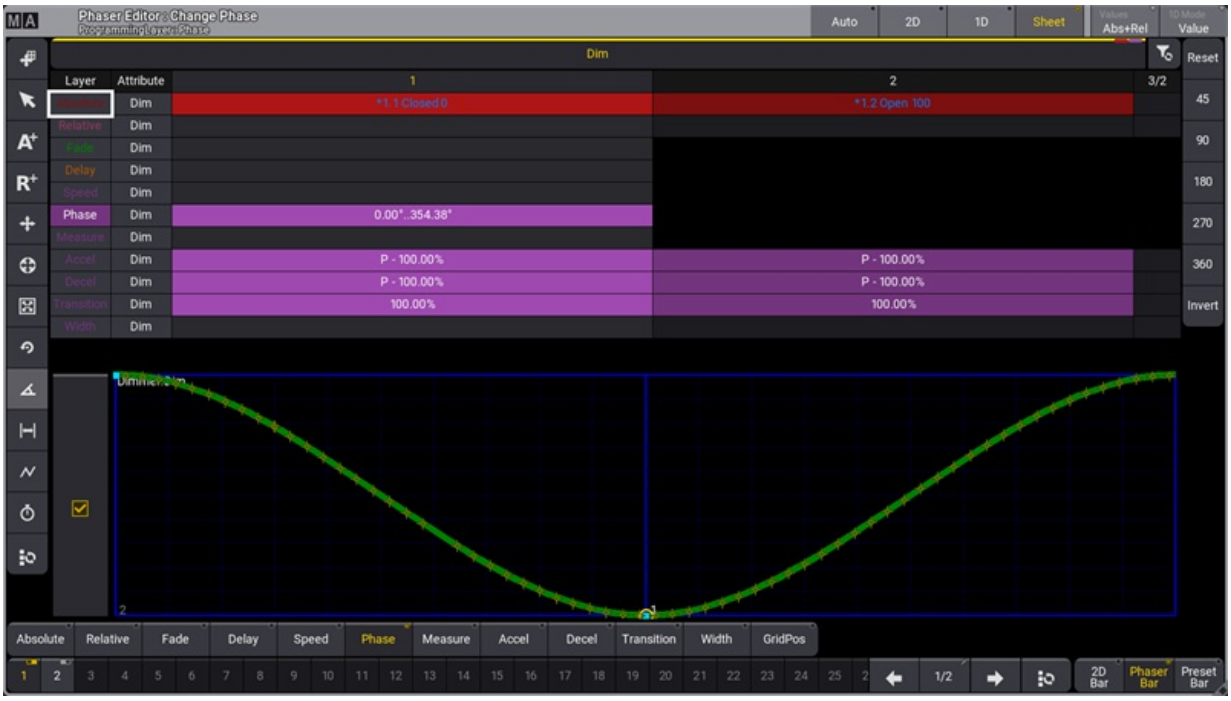
4. Select all steps by tapping the button at the right end of the Step Bar or at the bottom of the left-hand tool bar.
5. Tap to select the Select Form tool () in the left-hand tool bar.
6. In the right-hand tool bar, tap the sine wave button () to select the sine wave form.



Sinus Dimmer Phaser with No Phase Distribution

Adjust the phase of all fixtures using tools in the Phaser Editor:

7. Tap to select the Edit Phase ( ) tool in the left-hand tool bar.
8. In the right-hand tool bar, tap the 360 button to evenly distribute the selected fixtures over the wave form.



Sinus Dimmer Phaser with Full Phase Distribution

Store the phaser into a preset or a cue.

Alternate Methods to Create a Sinus Dimmer Phaser

Follow these steps to create a sinus dimmer phaser:

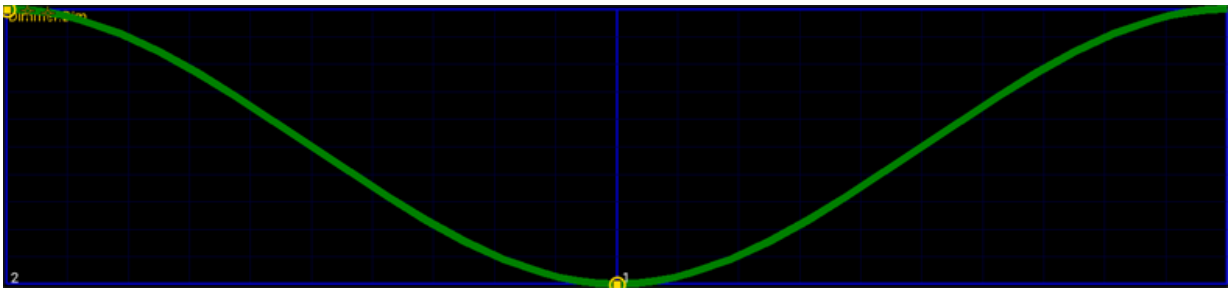
1. Select the desired fixtures. For example: Fixture Thru Please.
2. Set them at a dimmer value of 0: At 0 Please.
3. Create step 2 and select it: MA + Next (Next Step).
4. Set the fixtures at full: Full.

This is the base for the phaser. There are now two steps with a dimmer value of 0 in the first step and a value of 100 in the second step. All the fixtures are changing together from step one to two in a linear direction and do not fade using a sine curve. It looks like this in the **Phaser Editor**:

The curve is corrected by adjusting the acceleration and deceleration in both steps.

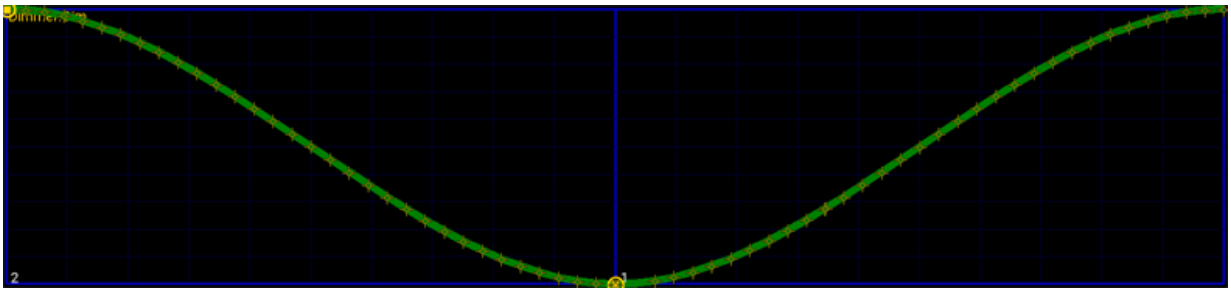
5. Select both steps: MA + Set (Step Toggle).
6. The steps need to have an accelerate value of -100. This is done using the **Encoder Toolbar**.
    1. Tap Dimmer to make sure it is selected.
    2. Tap Accel to select the acceleration layer.
    3. Tap the left attribute encoder to open the calculator.
    4. Tap +/- 1 0 0 Please. This gives a known and precise value. If **Handles** are used then it might not be as precise as typing a value.
7. Do the same for the Decel layer.

Now the attribute accelerates out of and decelerates into each step. The curve now looks like this:



8. The final thing to do is to spread out the fixtures using the phase value. This is also done from the encoder toolbar.
    1. Tap Phase to select the phase layer.
    2. Tap the left attribute encoder to open the calculator.
    3. Tap 0 Thru 3 6 0 Please.

Now the fixtures are spread out equally over the entire phase of the phaser loop.



This is a sinus phaser that can be stored in a cue or preset.

It is created using a specific selection of fixtures so remember to select the desired **Preset Mode** when **storing a preset**.

27.3. Create a Simple Circle Phaser

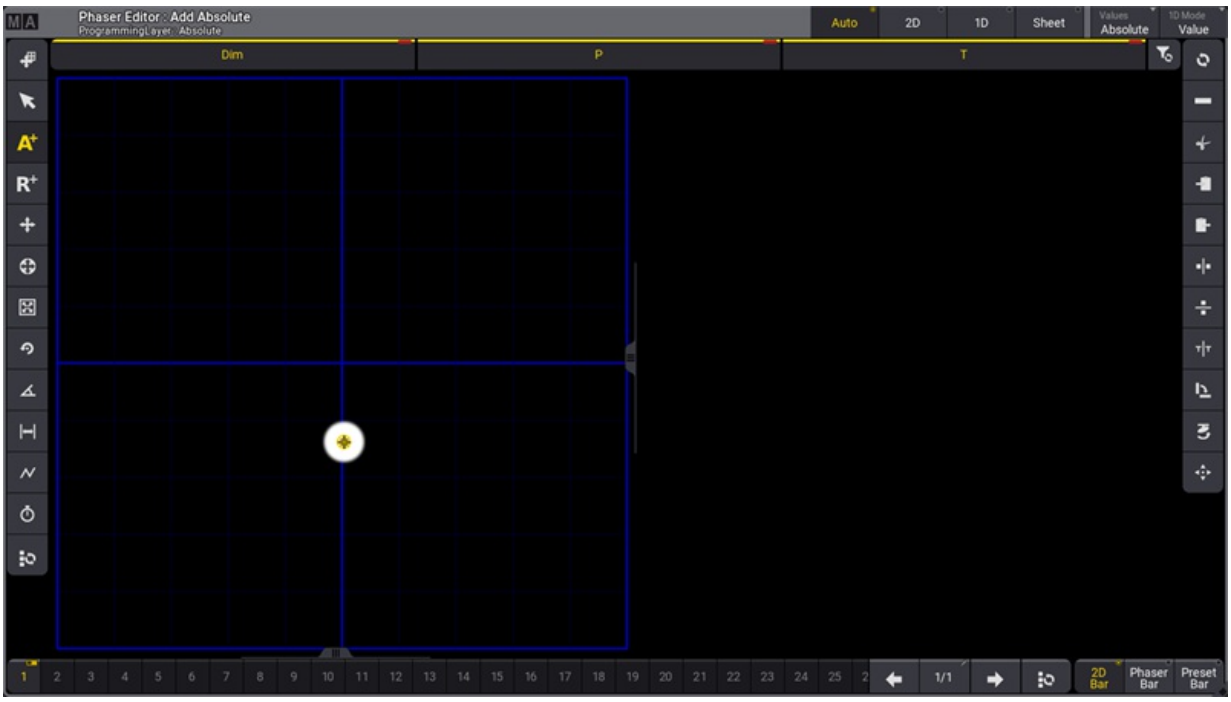
This topic presents an example of one method for creating a simple circle phaser. It uses absolute position values and will always move the lights in the specified circle. See the **Create Circle Phaser Around Position** topic for an example that covers creating a relative phaser circle, which can be used with different absolute base positions.

Requirements:

- Have a show with some moving lights patched - for instance, the demo show.
- An open **Phaser Editor** window is needed.
- Arranging the fixtures in the **3D window** can be useful, but it is not a requirement.

Follow these steps to create a circle:

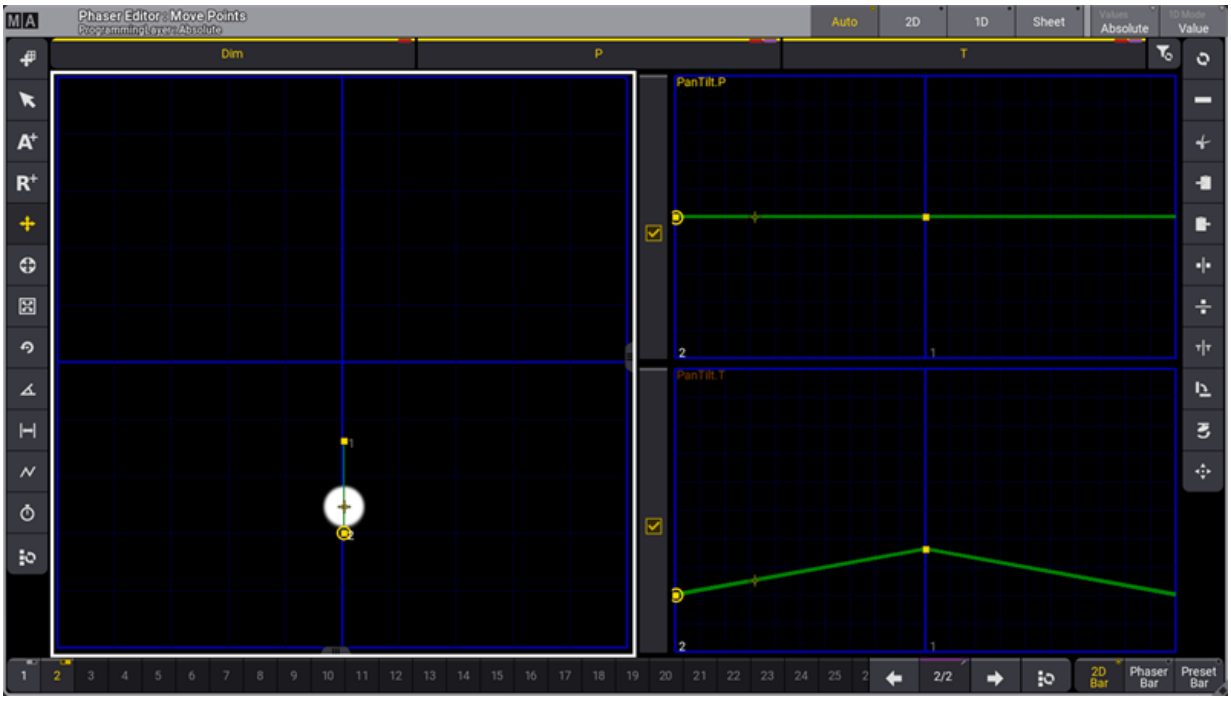
1. Select the desired fixtures and bring their dimmers to full.
   The Phaser Editor shows a white beam in the center of the blue 2D grid.
2. Tap the A+ button on the left of the editor.
3. Tap somewhere in the blue grid; for instance, just below the center.



Phaser Editor with fixtures turned on and one absolute position

This creates the first absolute point. The selected fixtures move to this static position. This is an absolute position value in step one.

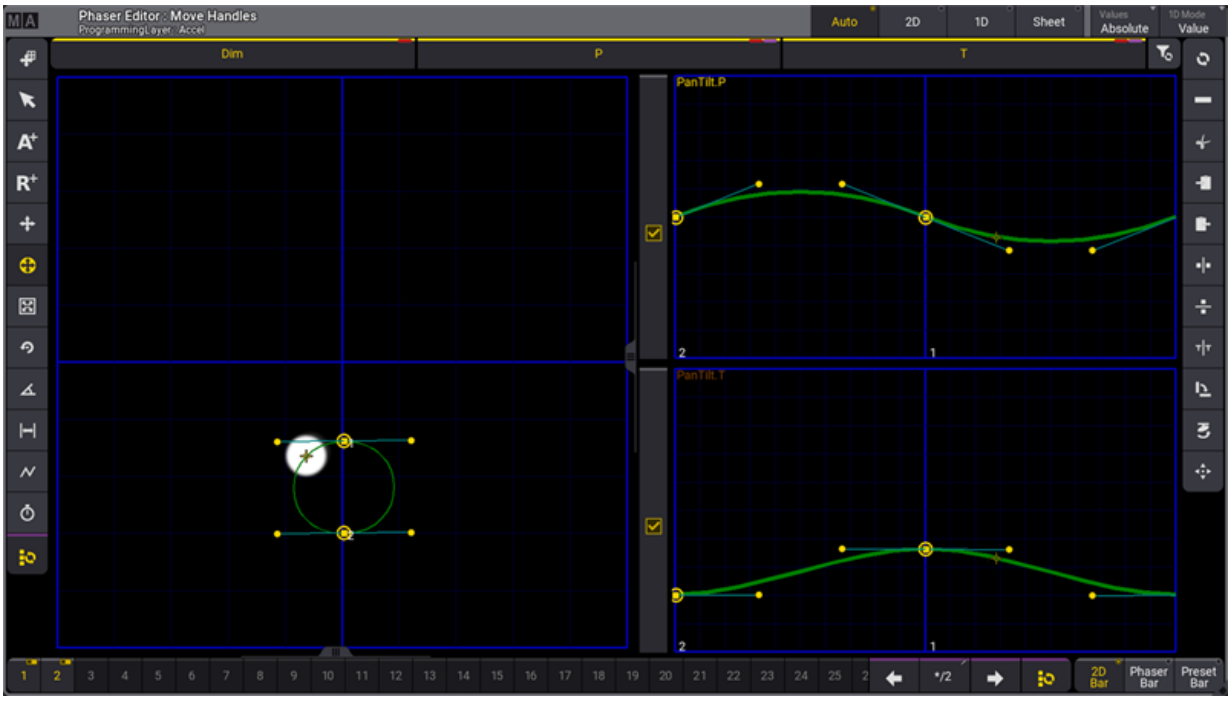
4. Tap somewhere else in the blue grid to create a second absolute position. Please tap 90 degrees vertically to the first point and do not cross the horizontal blue line - otherwise, the circle becomes a figure 8.



Two absolute positions - one in each step

Now, the fixtures move between the two absolute points (two steps).

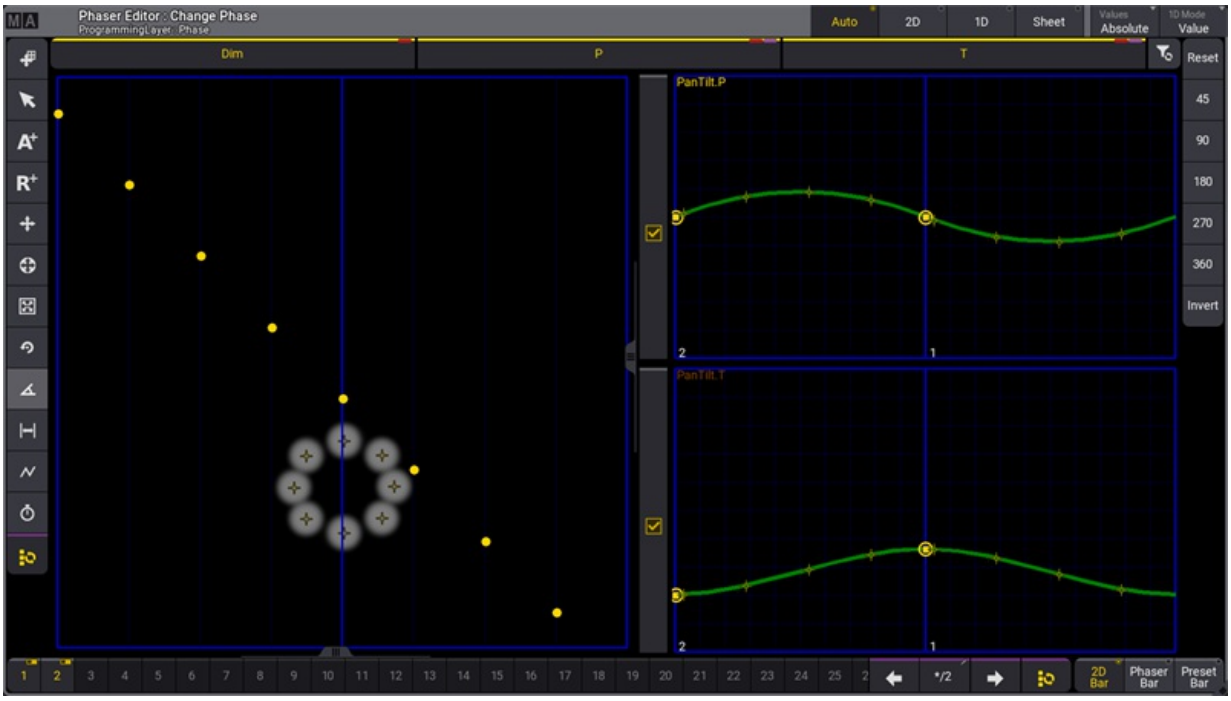
5. Tap the **Select All Steps** button ( ).
   This selects both steps.
6. Tap the **Move Handle** button ( ).
7. Tap one of the two points and drag horizontally.
8. Release the screen when there is a nice circular movement.



The fixtures now move in a circle

9. Tap the **Edit phase** button ( ) on the left menu.
10. Tap 360 in the menu on the right
    This distributes the fixtures evenly along with the form.

The finished result could look like this:



Finished circle phaser

The **Change Phase** part of the phaser editor represents each fixture as a yellow dot in a grid where the horizontal axis is the phase value.

The circle phaser is now finished and it can be stored into a cue or a preset.

Important:
Following the steps above exactly results in an active dimmer value in the first step of the phaser. If you prefer that your circle phaser only contain position values, deactivate the dimmer attribute before storing.

27.4. Create a Circle Phaser Around a Position Preset

This topic is about creating a relative circle phaser. This can be used together with an absolute position - for instance, from a position preset.

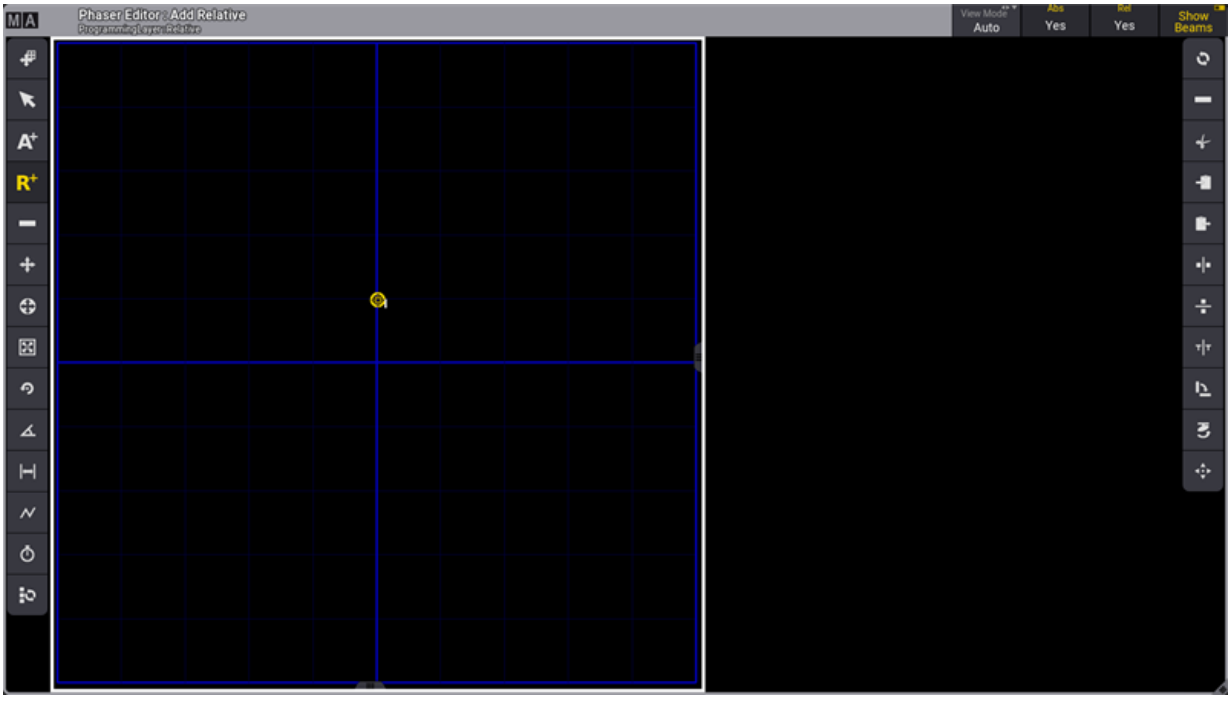
Requirements:

- Have a show with some moving lights patched - for instance, the demo show.
- An open **Step Bar** can be useful.
- An open **Phaser Editor** window is needed.
- Arranging the fixtures in the **3D window** can be useful, but it is not a requirement.

This example shows the recommended workflow when using a relative phaser with a position preset - for instance, the **Center** position preset from the demo show.

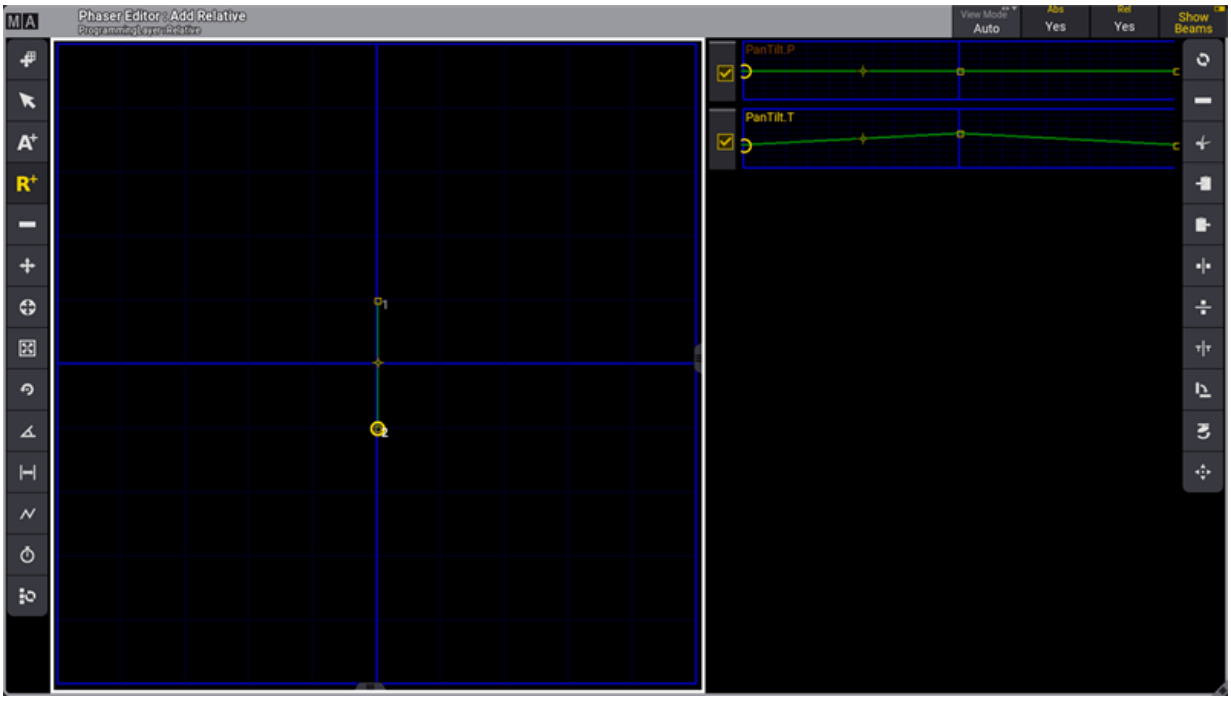
First, create the relative circle preset.

1. Select all the desired fixtures. This could be all fixtures that might want to use the relative circle phaser.
2. Tap R+ on the left side menu in the **Phaser Editor**.
3. Tap just above the center of the blue grid.



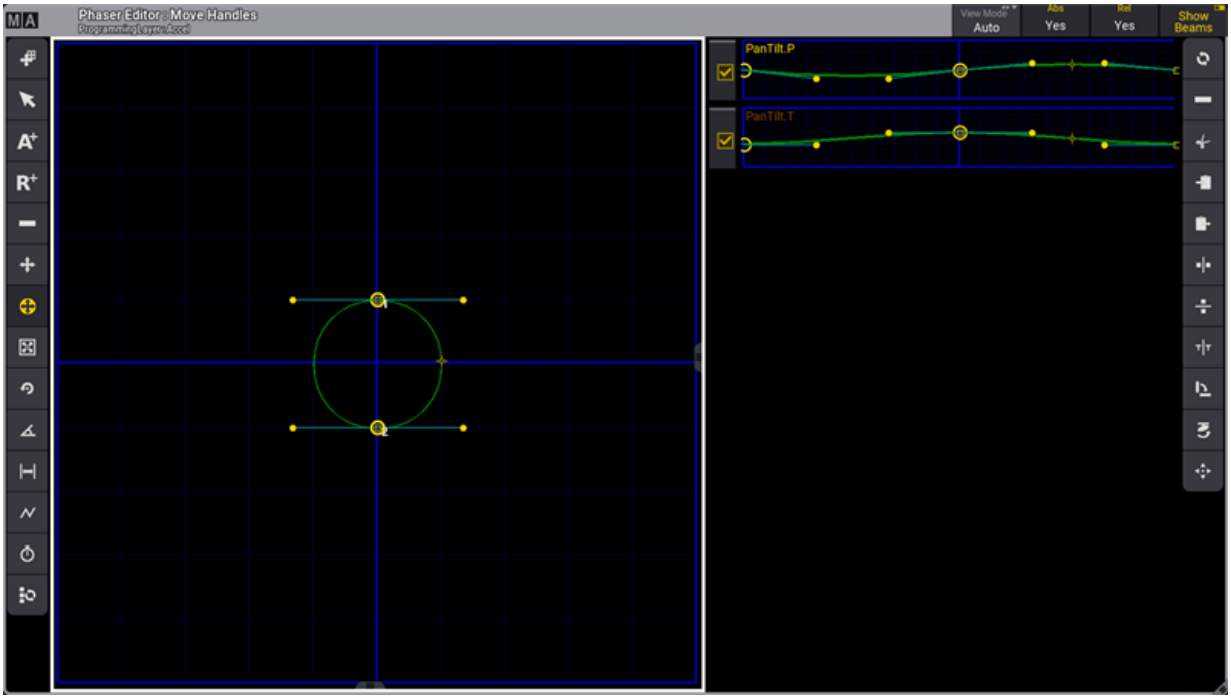
The first point in a relative circle

4. Tap just below the center of the blue grid
   -> now there are two relative points.



Two steps /points are the basis for the circle

5. Tap the **Select All Steps** button (the last button on the left menu or the last button in the step bar).
6. Tap the **Move Handle** button (circle with cross arrows).
7. Tap and move perpendicular from one of the points.
8. Release the screen when there is a nice circular movement.



The finished relative circle

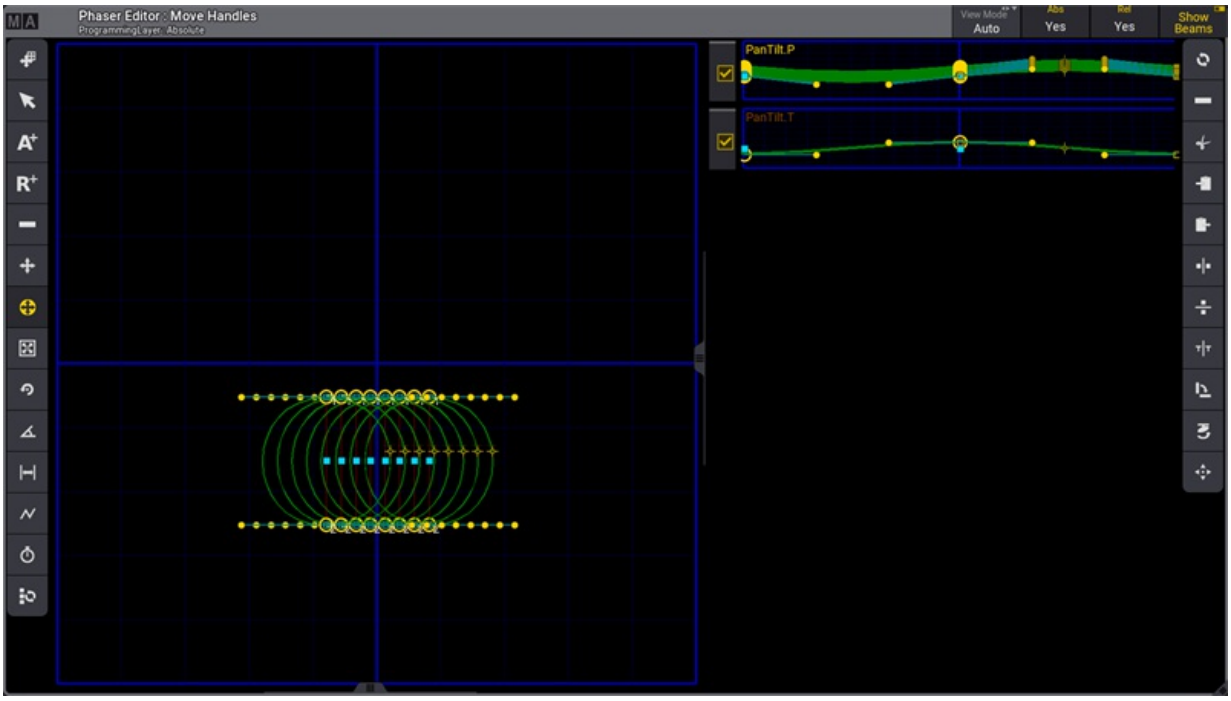
9. Store this as a global preset. This would allow this general circle form to be used in many phasers in the future.

Now we need to integrate the existing position preset (the **Center** position). The **Integrate keyword** is used for this.

1. Select the desired fixtures.
2. Tap the preset with the relative circle.
3. Make sure all steps/points are selected. If necessary, tap the **Select All Steps** button.
4. Press MA + At to get the **Integrate** command.
5. Tap the position preset with the desired position.

Now the relative points in the circle are referencing the position preset.

The result could look like this:



Fixtures moving in a circle around the middle position

This could be stored as a new preset.

27.5. Create Color Rainbow Phaser

This topic is about creating a rainbow color phaser. It creates the rainbow chase using three steps with a primary color in each step.

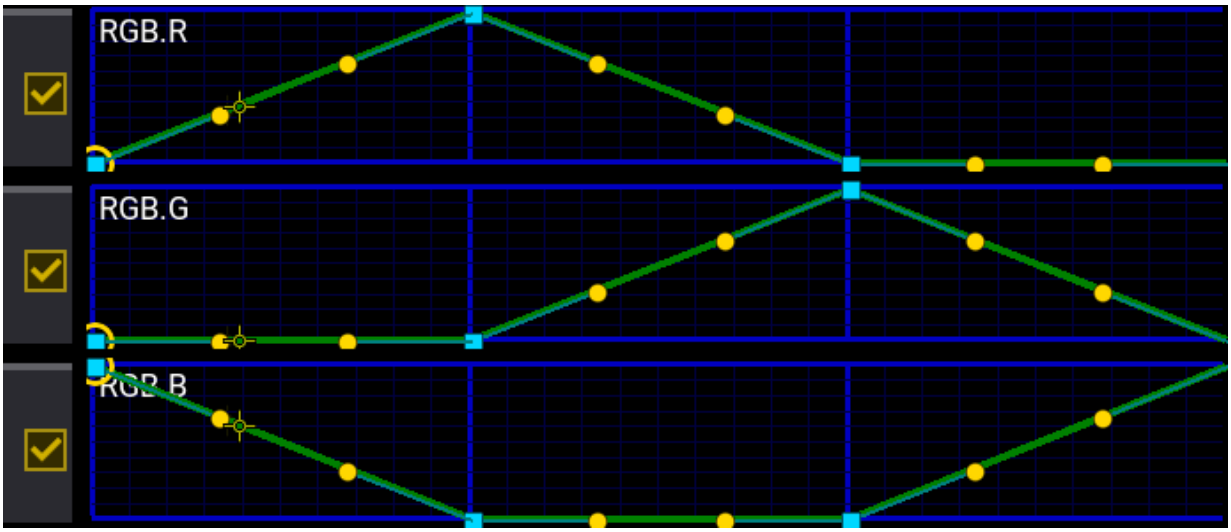
Requirements:

- Have a show with some fixtures that have a color mixing system (not color wheels with static colors) - for instance, the demo show.
- Have three color presets with Red, Green, and Blue colors stored using the color mixing system. These exist in the demo show file.
- An open **Step Bar** can be useful.
- An open **Phaser Editor** window is needed.
- Arranging the fixtures in the **3D window** can be useful, but it is not a requirement.

These are the steps needed to create the phaser.

1. Select the relevant fixtures.
2. Turn on dimmer so the output can be seen.
3. All presets can store phaser data. This means that color presets contain information in step one (as a default), but this needs to be integrated into the relevant phaser steps. The first step is going to be red.
   Press MA + At to get the **Integrate** command, then select the red color preset.
4. Create the second step by tapping the right-pointing arrow in the step bar.
5. Now the green presets need to be integrated into this step.
   Press MA + At to get the **Integrate** command, then select the green color preset.
6. Create the third step using the arrow again.
7. The third step needs to integrate the blue preset. Use the same method as described above.

Now the phaser editor should something like this:



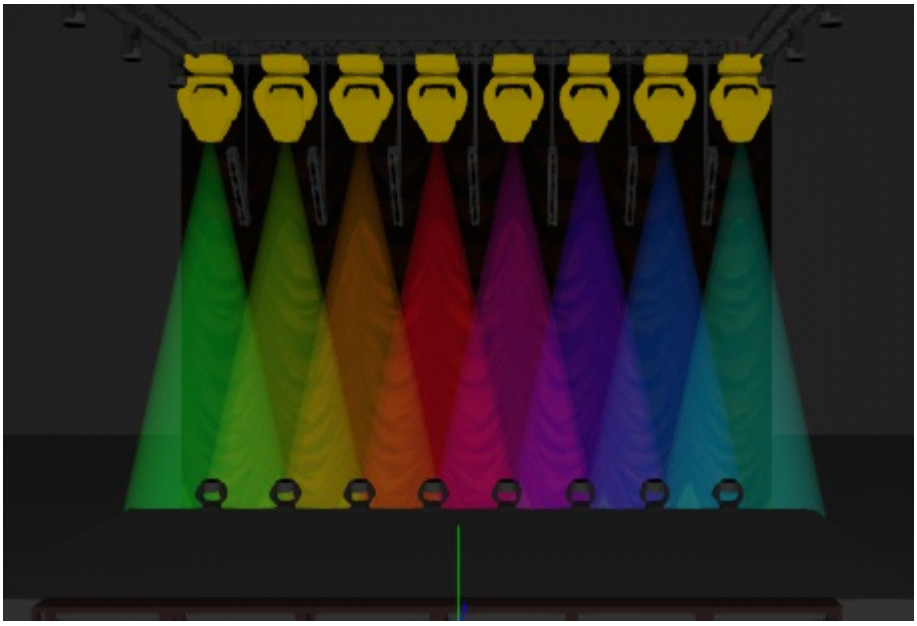
Phaser Editor with a three-step, three-color phaser

All the fixtures now change color together through the rainbow colors. If this is the desired result then store this in a cue or preset.

Phase can be used to spread out the rainbow across the selected fixtures:

1. Select all step by tapping the **Select All Step** button in the step bar.
2. Tap the **Change Phase** button on the left menu in the **Phaser Editor**.
3. Tap 360 on the right menu in the **Phase Editor** to spread the fixtures evenly.

The result could look something like this:



Rainbow chase across a range of fixtures

This can be stored as a preset or in a cue.

28. XYZ

Fixtures can be programmed using XYZ values instead of Pan and Tilt values.

This offers some advantages over Pan Tilt programming. For instance, when fixtures move from one position to another they do it in a straight line from position A to B when using XYZ values. When using Pan/ Tilt the fixtures fade from DMX value A to B, this means that two fixtures often do not follow the same path.

Pan/Tilt values can be mixed with XYZ values in a sequence but it might give unwanted results. It is currently recommended to use either XYZ or Pan/Tilt.

The X, Y, and Z attributes have a range from 0 to 100 % of the space they are using. The default space is the default stage space volume. Read about stage spaces in the **Stages topic**.

XYZ needs to be activated for fixtures that would like to use MArker fixtures.

28.1. Activating XYZ for Fixture Types

XYZ needs to be activated for a fixture type to get access to the XYZ attributes.

Important:
Turning On XYZ adds extra virtual parameters for each fixture using the fixture type mode. These extra virtual parameters do not count against the parameter limit but they are shown in the **System Info window**.

XYZ is turned on in the Fixture Type Editor.

1. Press Menu.
2. Tap Patch.
3. Tap Fixture Types in the menu on the left.
4. Select the fixture type that needs XYZ.
5. Tap Edit in the menu at the bottom.
6. Select the desired DMX Mode, if there are multiple.
7. Turn On XYZ in the menu at the bottom.
8. Close the editor and the patch and save the changes.

Now XYZ can be used for the selected fixture type mode.

XYZ is activated for the selected fixture type mode. So make sure it is activated for the mode that is used.

28.2. MArker Fixture

A fixture type called **MArker** from the manufacturer MA Lighting is part of the fixture library for the MA source.

A MArker is a virtual fixture, that allows operating (for instance, moving around or rotating) objects such as fixtures, actors, or stage props in the 3D stage environment. Learn more bout this in the **3D window**.

It can also function as a target for XYZ-enabled fixtures. Learn how to activate XYZ in the **Activating XYZ for Fixture Types topic**.

MArker fixtures have a **Target Space** in the Stage. The **Space** defines the volume around the MArker. This space is also the boundary for X, Y, and Z attribute values for the fixtures pointing to the MArker. For instance, if the target space is 2 meters wide on the X-axis, then a fixture pointing to this maker can move inside this 2-meter wide space volume. Setting the X attribute to 0 will move the fixture to the minimum X position in the MArkers target space. The default size for a target space is 200 meters on the X and Z axes (very much bigger than the default stage space).

Moving MArker fixture also has a Movement Space. This defines the space volume in which the MArker fixture can be moved. The default values for this space are also 200 meters in X, Y, and Z.

Read more about the spaces in the **Stage topic**.

MArker as an Object Mover

For each object or group of objects that shall be moved separately, a MArker fixture needs to be added to the patch in the **Moving** mode.

As soon as a MArker fixture is patched, other patched fixtures can be added as children of the MArker. After applying these changes to the patch, the MArker fixture can be operated like any other fixture in the show.

The MArker fixture provides these attributes:

- X, Y, and Z for moving the MArker and all its children together along the corresponding axes.
- Rot X, Rot Y, and Rot Z rotate the MArker and all its children together around the corresponding axes.

These attributes are located within the features XYZ and Rotation within the feature group Position in the encoder bar.

Select the MArker fixture and turn the encoders for the described attributes in order to see the MArker and its children moving around in the 3D window. The children of a MArker are always moved around relative to the position set up of the MArker itself.

This means that if the fixtures are already at a height of 5 meters and then attached to a MArker that is at a height of 0 meters, then the MArker needs to move below zero to move the fixtures below 5 meters.

Important:
A MArker fixture needs to have a DMX address patched in order to be able to see its changes within the 3D window.

MArker as a Target

The MArker fixtures can be a target for other fixtures that are XYZ enabled. Read about activation in the **Activate XYZ for Fixture Types topic**.

The MArker fixtures need to have the MArker IDType in the patch and a CID number.

The MArker fixtures can be of the Moving or Still mode. Still markers can be positioned using the patch or any other method described in the **Position Fixtures in the 3D Space** topic. Still MArkers are meant to be used for positions in the 3D space that do not move around. Moving MArkers are meant to be used with positions that can move in the 3D space.

The fixtures with XYZ enabled have a MArker attribute. Giving this attribute a number makes the fixture point to the MArker fixture with the matching number.

Moving the MArker, makes the light fixtures pointing at the MArker move the beams to match the MArker fixtures' position.

If the light fixtures pointing to MArker fixture, are be moved using a different MArker fixture while pointing to a MArker, then the light fixture will try to keep the light beam pointing to the MArker.

Moving the MArkers by a Tracking System

The input of PSN trackers can be linked to moving MArkers. This allows moving around the MArker and its children via an external tracking system.

To do so, set up the number of markers needed, and add the desired children to them within the grandMA3 software.

Then set up the PSN system and configure the **PSN input** within the grandMA3 software. Within the PSN menu, the columns called IDType and ID allows entering the MArker IDType and ID of the MArker fixture that shall be linked to the input of each tracker.

The column DMX Priority defines at which level of the grandMA3 playback priorities the PSN data shall be processed. This allows overwriting the input of the PSN system by using a sequence with a higher priority if needed.

Important:
The DMX universes the marker fixtures are patched to which are receiving data from PSN trackers need to be set to the **merge mode Prio**.

29. Macros

What are Macros

Macros are commands stored in a pool object.

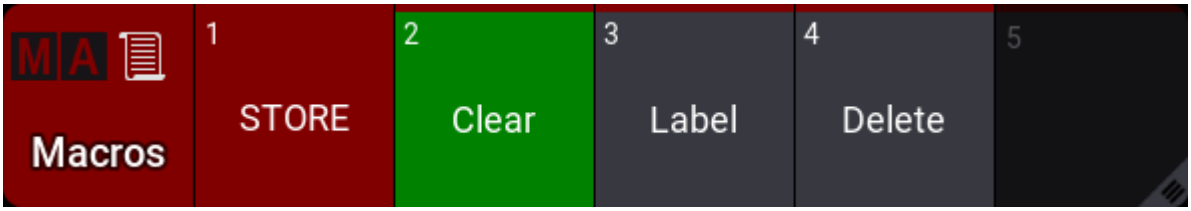
The commands can be simple, very complex, and everything in between.

Macros can make programming faster and more convenient. Macros make it possible to do trivial or complex operations by a push of a button.

Macro Pool

Macros are stored in the **Macro Pool** but can be assigned to physical keys and view buttons. The macro pool is shared between all users in the show file. This means that if UserA stores a macro on pool object number 10, then it will also be available for UserB as macro number 10.

The Macro Pool could look like this:



Macro pool

Elements in a Macro

A macro consists of one or more rows.

Each row has information about a **Command** the row executes.

Each row has several fields that specify how the row is handled.

The **Wait** field can add a wait time before moving to the next row in the macro. This wait time is added after the command in the row is executed. It is relevant if there is more than one row in a macro. The first row will execute, it will wait the specified time and then execute the next row. There are two special wait commands. They are not a time value but a special value.

- **Follow**: This is the same as having a wait time of 0. The next row will be executed as soon as this row has executed the command.
- **Go**: This special value pauses the macro at this row. The macro is paused until it receives a new Go command.

The **Enabled** field indicates if the row is enabled. This is a **Yes** or **No** field. If it is **Yes** then it is executed when the macro row is triggered.

The **AddToCmdline** function makes it possible to append the command from the macro line to existing content in the command line. This is a **Yes** or **No** field. If it is **Yes** then it is appended to the command line.

Execute field defines if the macro row is automatically executed. **Yes** means that the row is executed (automatic please at the end of the row). **No** will put the command in the command line ready for user interaction.

Import Macro Library

There is a factory library of macros that can be imported using the import command.

Read more about importing in the **Import Keyword topic**.

29.1. Create Macros

Macros are created using the GUI editor or the command line. Please read the **Macros topic** for a better understanding of macros.

Create from GUI (Pool)

Creating a macro using the GUI is very similar to editing the macros.

Requirement:

- Have a Macro Pool visible on one of the screens.

This is the creation process:

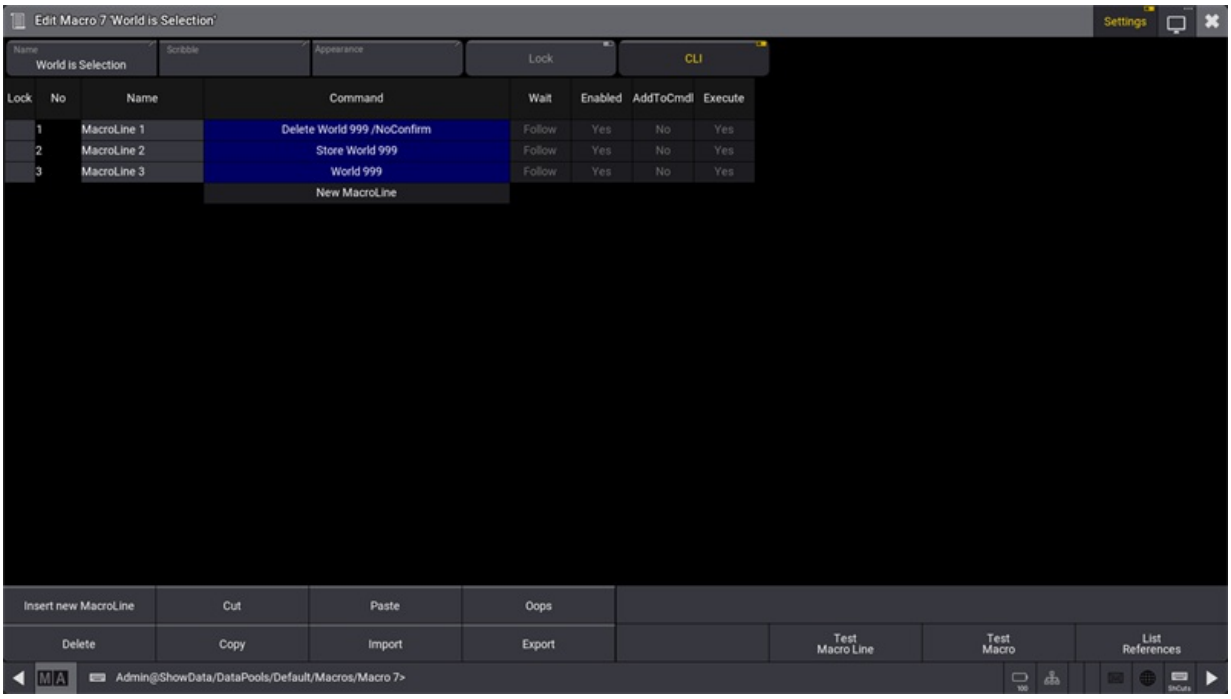
1. Edit an empty pool object - this automatically creates a new macro and opens the **Editor**.
2. Tap the Insert New MacroLine button to add a row in the macro.
3. Edit the **Command** field in the new row - this opens a text input pop-up.
4. Write the command the macro row should perform.
5. Edit the other fields in the row to match the needs (read the Macro topic for explanations).
6. Repeat steps 2 through 5 to add several rows with commands to the macro.
7. Close the editor when done.

There are several buttons at the bottom of the editor. This is a short explanation of the special buttons for this editor:

- **Settings:**
  Tap Settings in the title bar to display the **Name**, **Scribble**, **Appearance**, **Lock**, and **CLI** buttons.
- **Name**:
  Tap this to label a macro. Please read about **label** further down this topic.
- **Appearance**:
  Tap this to add an appearance to the macro.
- **Scribble**:
  Tap this to add a scribble to the macro.
- **Go**:
  Tapping this executes the macro.
- **List References**:
  Tapping this opens a pop-up with a list of objects that reference and depend on the macro.

The new macro pool object can be tapped to execute it.

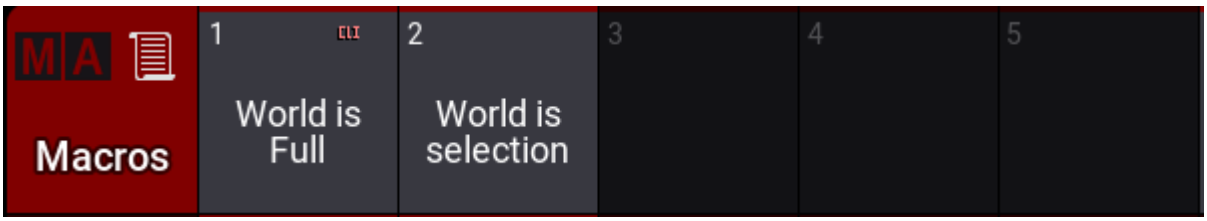
The editor could look like this (the macro is the "World is Selection" macro from the library):



Macro editor with multi-row macro

When CLI is deactivated, tapping the macro pool will still execute it if nothing is in the command line. When you press Edit and tap macro 5 in the macro pool, it will no longer interact with the edit command. To edit the macro, you would have to type:

Hint:
When command-line interaction is disabled, CLI is displayed in red letters on the pool object. Now that commands no longer interact with the command line, so do the swipey commands.



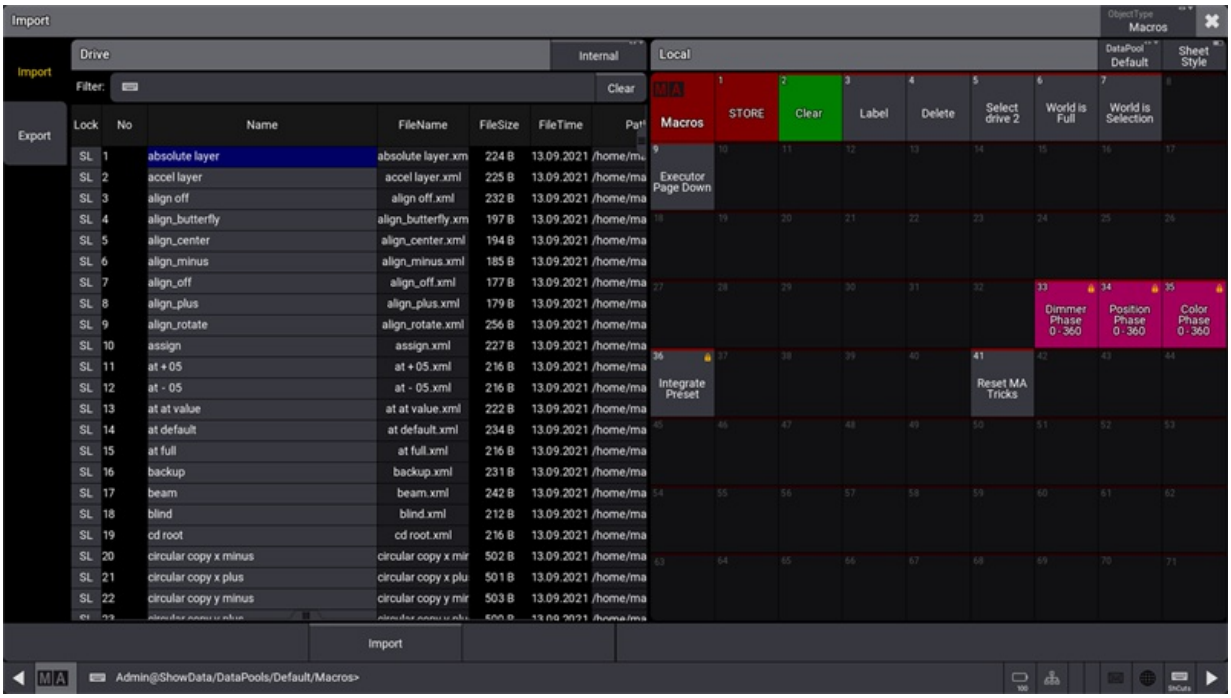
Macro 1 with CLI deactivated

Import Macros from the Library

Predefined macros can be imported from the Library.

1. Press Menu.

2. Tap Import/Export, this opens the Import menu.



Import menu

3. Tap and hold ObjectType in the title bar. This opens the edit object Type menu.
4. Tap Macro.
5. Tap an empty pool object (on the right side of the window) where you want to import the macro. If multiple macros are selected, this is where the first macro of your selection will be imported onward.
6. Tap Import at the bottom of the window.
7. The macro is imported.

Hint:
If no pool object is selected, the macro will be imported at the next available empty pool object.

Import a Macro from the Macro Pool

Requirement:

- Have a Macro Pool visible on one of the screens.

This is the import process:

1. Edit an empty pool object - this automatically creates a new macro and opens the **Editor**.
2. Tap Import at the bottom of the editor - This opens the macro library.
3. From the list, tap the name of the macro you wish to import then tap Import.
4. The macro is imported.
5. Close the editor when done.

Create from Command Line

Macros can be added using the command line. This is relevant when the GUI is not available, for instance, when the software is accessed via command-line only - for instance, using the terminal window.

To see the result in the command line feedback, use the **List** command at any time during the creative process.

Requirement:

- The **command line feedback** needs to be visible.

This is the creation process:

1. Navigate to the macro part of the software:

```
User name[Fixture]>ChangeDestination Macro
```

2. List the existing macros to see empty macro objects:

```
User name@ShowData/DataPools/Default/Macros>List
```

3. Locate an available number (one that is not listed).
4. Store a macro with the number:

```
User name@ShowData/DataPools/Default/Macros>Store [available_number]
```

5. Navigate into the new macro:

```
User name@ShowData/DataPools/Default/Macros>ChangeDestination
[available_number]
```

6. Insert a line into the macro:

```
User name@ShowData/DataPools/Default/Macros/Macro #>Insert
```

7. Add the command to the text field line number using the **Set** command:

```
User name@ShowData/DataPools/Default/Macros/Macro #>Set [line_number]
Command "command inside quotations"
```

8. The wait time can be set in the same manner as the text:

```
User name@ShowData/DataPools/Default/Macros/Macro #>Set [line_number]
Wait "wait time"
```

Hint:
The wait time can be entered as a number in seconds without quotation marks or as text with quotation marks. The two special times called **Follow** and **Go** can be entered without the quotation marks, but the software will add the marks. They are case-sensitive. Typing "go" gives an error. Typing "Go" works.

9. Add more macro lines by repeating steps 6 through 8, but remember to use the correct line number with the Set command.
10. Return to the root of the command line:

```
User name@ShowData/DataPools/Default/Macros/Macro
#>ChangeDestination Root
```

Now there is a new macro that can be used.

Label a Macro

Macro pool objects are labeled like any pool object. Read more in the **Label Pool Objects topic**.

Remember that using unique names makes it possible to run the macros using their names.

29.2. Edit Macros

Editing macros is the same as creating macros except for the macro already exist.

There are two ways to edit them: Using the GUI and using the command line.

Edit Using the GUI

Requirement:

- Have a Macro Pool visible on one of the screens.

This is the process to edit a macro:

1. Tap, hold and swipe out of the macro in the macro pool that is to be edited
2. Swipe to the Edit swipey command and release the screen.
   The editor opens.
3. Edit the fields that are to be changed.
4. Close the editor when the macro is correct.

Please read the **Create Macros** topic to learn about adding lines and labeling a macro.

Macro lines can be deleted by selecting the line and then tap the Delete button in the editor.

Edit Using the Command Line

Requirement:

- The **command line feedback** is very nice to have visible.

Remember that the **List** command displays the content at the current location and it can be used at any time.

This is the editing process:

1. Navigate to the macro that is to be edited:

```
User name[Fixture]>ChangeDestination Macro [macro_number]
```

2. Edit the fields using the **Set** command. This simply overwrites the current content in the field.
3. When done editing, return to the root location:

```
User name@ShowData/DataPools/Default/Macros/Macro
#>ChangeDestination Root
```

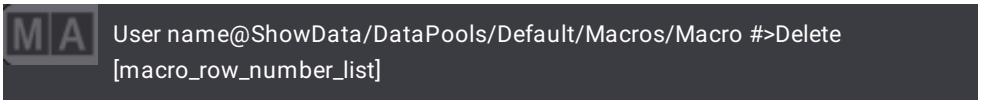
Please read the **Create Macros** topic to learn about adding lines and labeling a macro.

Delete a Macro Row Using the Command Line

Macro rows can be deleted using the command line:

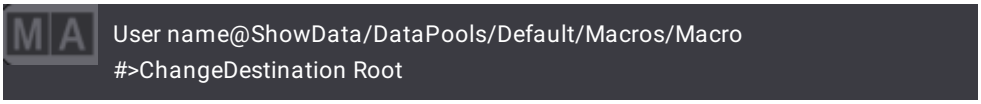
1. Navigate to the macro where the row is to be deleted:

2. Use the **Delete** command followed by the row number:



Hint:
Multiple rows can be deleted by writing a list of numbers.

3. When done deleting rows, return to the root location:



29.3. Assign Macros to Keys and Buttons

The macros can be assigned to executors or view buttons to allow easy access.

The macro is still in the macro pool. The executor or view button simply run the macro in the pool. It is, therefore, a requirement to have the macro in the pool before assigning it anywhere.

There are different ways to assign macros to keys or buttons:

Assign Using the Keys

This is maybe the fastest way to assign the macro when using a console:

1. Press the Assign keys.
2. Press the MA key while pressing the X14 | Macro key.
3. Press the number matching the macro number.
4. Press the key where the macro is to be assigned.

Alternatively, if the macro pool is visible then it can be used:

1. Press the Assign key.
2. Tap the macro in the pool.
3. Press the key where the macro is to be assigned.

Assign Using the Swipey Commands

Requirement

Have the macro pool visible on the screen.

1. Tap and hold the macro pool object.
2. Swipe out of the pool object -> the swipey commands open.
3. Swipe to the Assign button and release the screen.
4. Press the key where the macro is to be assigned.

Assign Using the Assign Menu for Executors

1. Open the **Assign Menu** for the desired Executor.
2. Tap the Object button in the menu on the left.
3. Tap the Marco button at the top of the **Assign Menu**.
4. Tap the desired macro in the list in the menu.
5. Close the **Assign Menu**.

Assign Using the Command Line to Assign to Executors

Make sure input focus is on the command line and use this syntax:

```
User name[Fixture]>Assign Macro [macro_number] At Page [page_number].
[executor_number]
```

The syntax above is universal and allows for the macro to be assigned to a specific page.

If the macro is to be assigned to the current page then the following syntax can be used:

```
User name[Fixture]>Assign Macro [macro_number] At Executor
[executor_number]
```

29.4. Example Macros

This topic have some example macros. They are meant as inspiration for other macros. They show some of the possibilities with macros.

Change User

This macro opens the login window where you can enter a new user name and password.

Lock	No	Name	Command	Wait	Enabled	AddToCmdl	Execute
	1	MacroLine 1	Login	Follow	Yes	No	Yes

World Is Selection

This is a predefined macro that will create a temporary world for the selected fixtures

Lock	No	Name	Command	Wait	Enable	AddToCmdl	Execute
	1	MacroLine 1	Delete World 999 /NoConfirm	Follow	Yes	No	Yes
	2	MacroLine 2	Store World 999	Follow	Yes	No	Yes
	3	MacroLine 3	World 999	Follow	Yes	No	Yes

Block Sequence

This macro will prompt you for a sequence ID and a cue number to block.

Lock	No	Name	Command	Wait	Enable	AddToCmdl	Execute
	1	MacroLine 1	Block Sequence (Sequence #) Cue (Cue #)	Follow	Yes	No	Yes

Hint:
If no cue number is entered, the entire sequence will be blocked.

30. Agenda

The agenda allows you to schedule objects, e.g., sequences, macros, or plugins, to be executed by the console based on the calendar. Commands can also be defined, for example, to shut down the system at a certain time.

You can also schedule events that will repeat every minute, every day, every week, every month, and/or every year, e.g., Saint-Patrick day cue every year.

To learn about the Agenda keyword, see **Agenda**

30.1. View Modes

The agenda window can display the data using five different view modes.

Open an agenda window, see **Add Windows**. From the add windows dialog, tap More, and then tap Agenda.

Agenda window

To change the view, tap View Mode in the title bar.

- **Sheet**: Displays all agenda entries in a spreadsheet format.

Hint:
To sort the agenda entries in the sheet view by name, right-click or tap and hold a column header.

- **Year**: Displays a calendar view of the year. The current date has a light gray background. Days with at least one enabled event are indicated by a green background. Days that have all events disabled are indicated by a bright red background.
- **Month**: Displays all agenda entries of the selected month.
- **Week**: Displays all agenda entries of the selected week.
- **Day**: This mode is divided into two sections. It displays a smaller monthly overview on the left and entries for the selected day on the right.

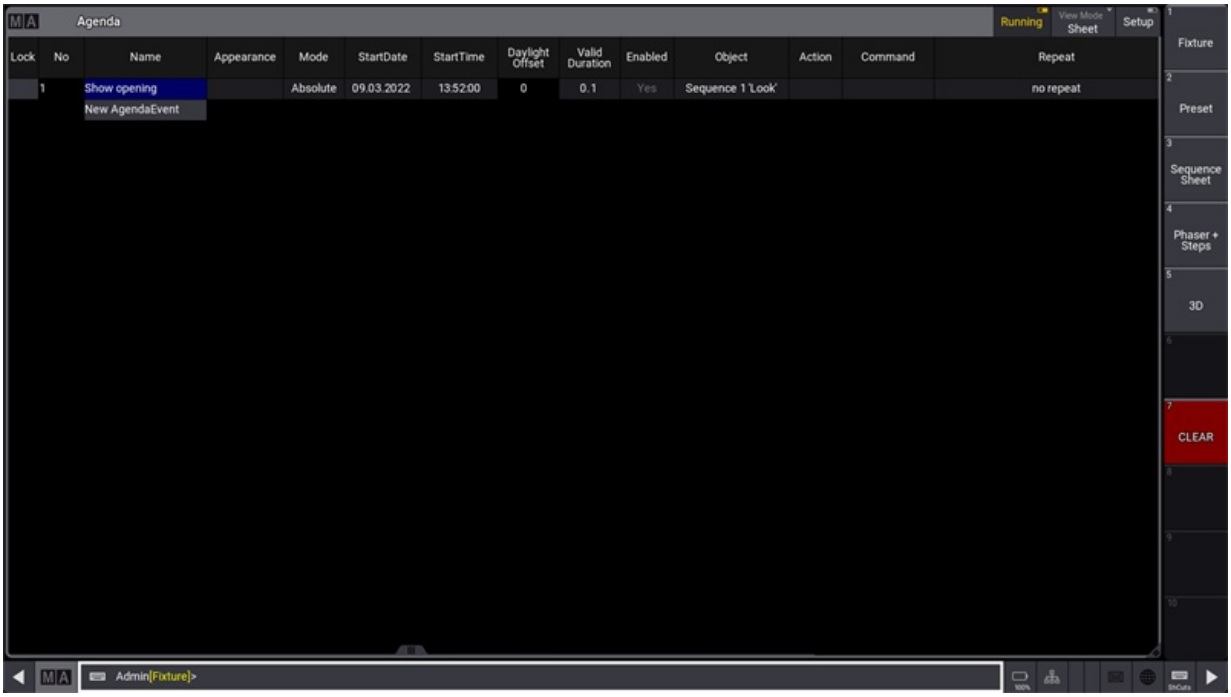
Hint:
- Disabled events are displayed with red font color.
- Repeated events are displayed with on the right side. This applies to the month, week, and day view modes.

30.2. Create an Agenda entry

Create an Agenda Entry Using the Sheet View Mode

1. Tap and hold View Mode in the title bar, then slide your finger in the list and select Sheet.
2. From the **Name** column, right-click or tap and hold New AgendaEvent to create a new event.
3. A new entry is created with the current system date and time.

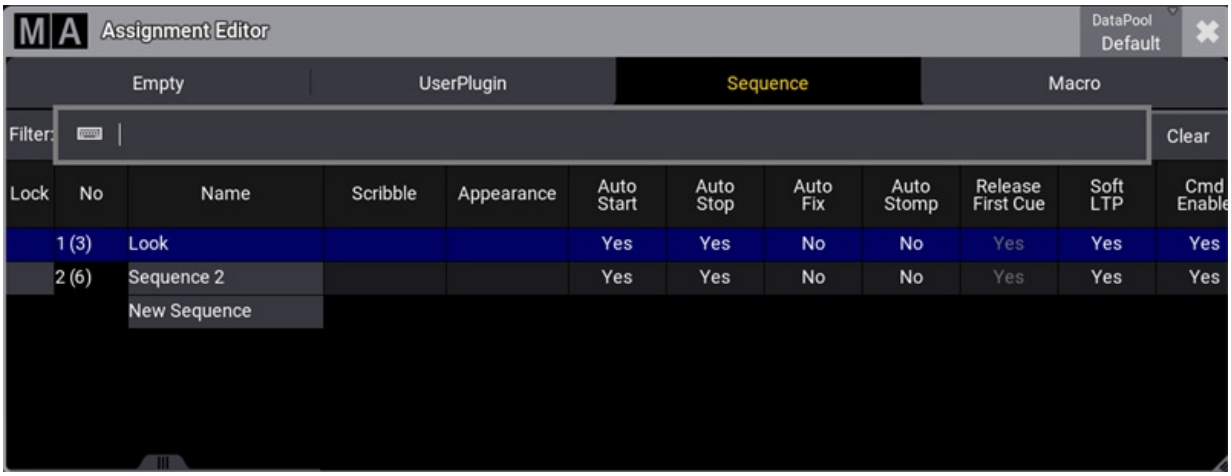
Hint:
It is also possible to create an entry from the **Sheet** layout mode by pressing Edit, then tapping New AgendaEvent.



The following is a list of the properties to be defined by the user:

- **Name**: Enter the name of the event that will be displayed in all layout modes.
- **Appearance**: Assign an appearance to define how the entry looks in the calendar views.
- **Mode**: When set to Absolute, the agenda uses the entered start date and time. Twilight times like Dawn, Sunrise, Sunset, and Dusk can also be used. See **Date and time** to learn how to configure the grandMA3 software to calculate the correct twilight time for your location.
- **StartDate**: Tap and hold a cell to open the **Edit StartDate** pop-up, then edit the day, month, and year. Use Today in the title bar to quickly enter today's date.
- **StartTime**: Tap and hold a cell to open the **Edit StartTime** pop-up, then edit the hours, minutes, and seconds. Use Now in the title bar to quickly enter the current time.

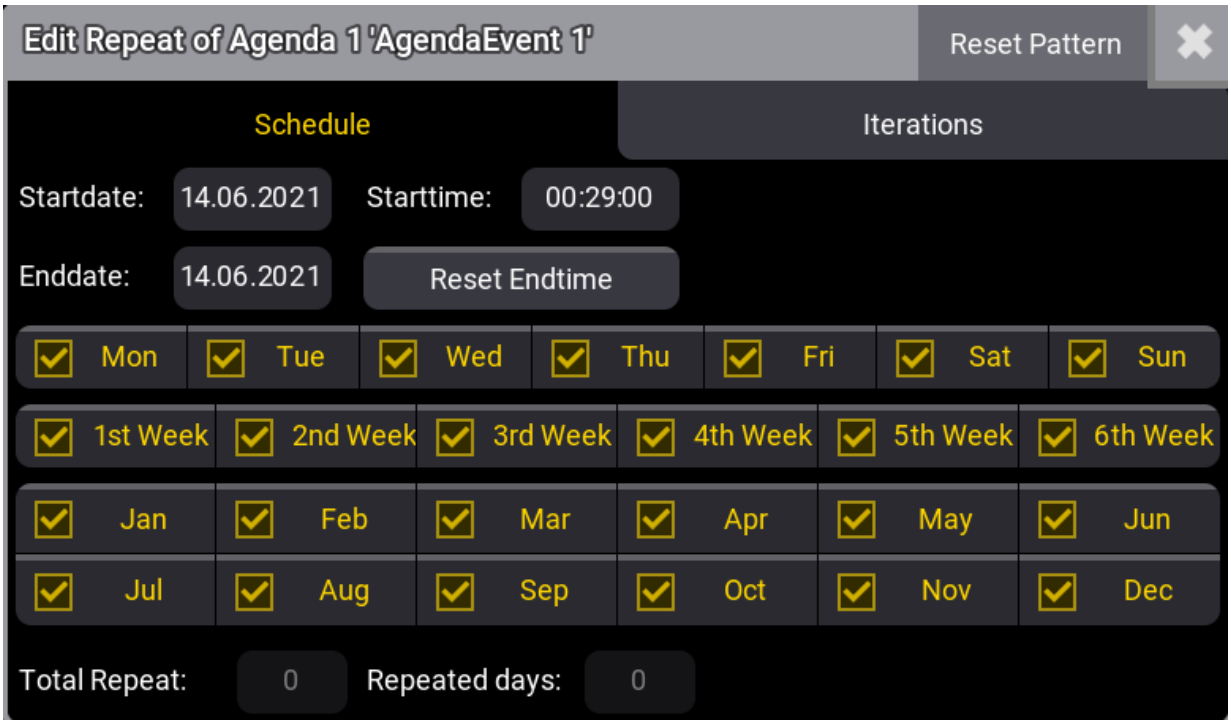
- **Daylight Offset**: When a twilight time is set in the mode column, it defines an offset for an event. For example, if you want an event to start 15 minutes before dawn. See **Date and time** to learn about twilight times.
- **Valid Duration**: Enter a duration value to allow the backcasting of events if the console is not switched on or the agenda is disabled during a scheduled entry.
- **Enabled**: To enable or disable an agenda entry. Disabled entries are displayed with a red color font.
- **Object**: Tap and hold a cell to open the Assignment Editor pop-up, select a **Plugin**, **Macro**, or **Sequence** to be executed.



- **Action**: Tap and hold a cell, then select an action to use when the defined object is executed.
- **Command**: Define a command here to be executed instead of defining an object to be triggered. E.g., Go+ Executor 101.
- **Repeat**: Defines if an entry is repeated and, if so, how often. Tap and hold a cell to open the **Edit Repeat of Agenda** pop-up.

They are two main settings to define repetition of an agenda entry, Schedule and Iterations.

Schedule Settings



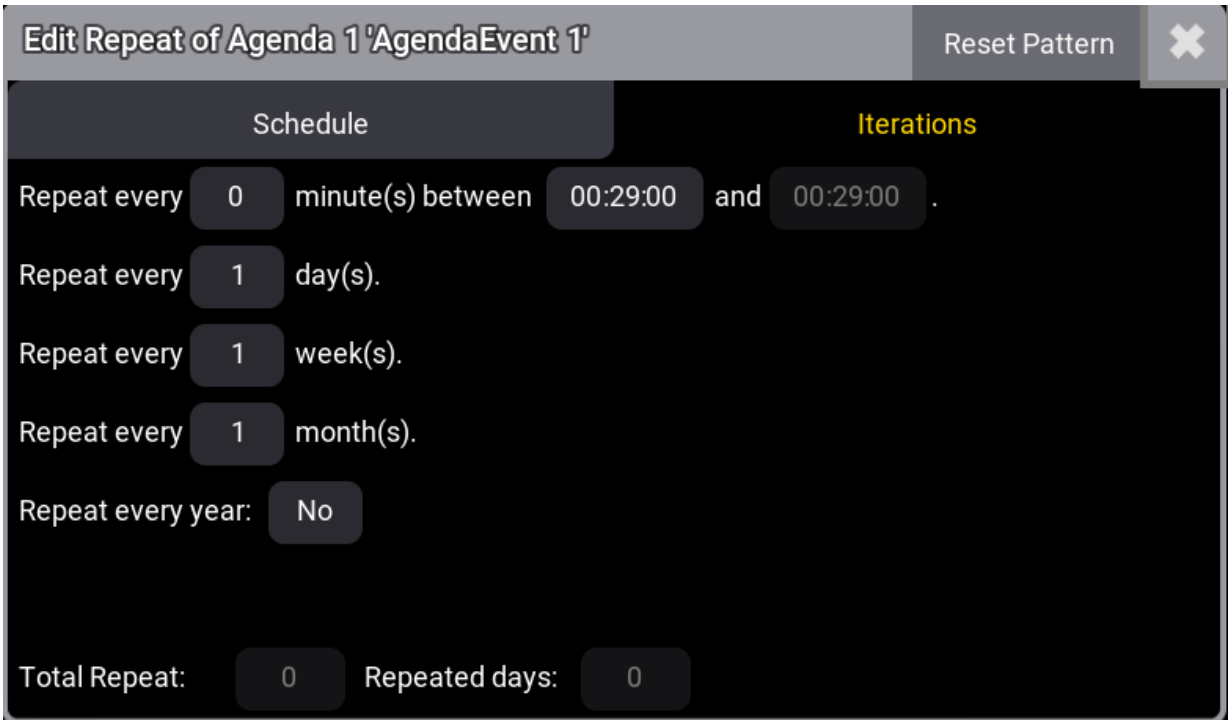
The start date and time are linked to the agenda event and therefore are identical. At least a different end date or end time must be set to get repetitions.

- Tap Reset Pattern in the title bar to reset the repeat settings you made previously.
- Tap Reset Endtime to reset the end date back to the agenda event date.

Tap a day, week, or month cell to enable or disable it. This will define at which days, weeks, or months the event will be repeated.

Hint:
The week settings are the weeks within a full month and not the weeks from the start date onwards.

Iterations Settings



Iterations settings allow defining repeat per minute, days, weeks, and months.

These settings are counting from the start date onwards.
To repeat an event every year again, set **Repeat every year** to yes.

The following is a description of the last five columns of the agenda window, when ViewMode is set to **Sheet**, in the title bar. (These columns can not be edited).

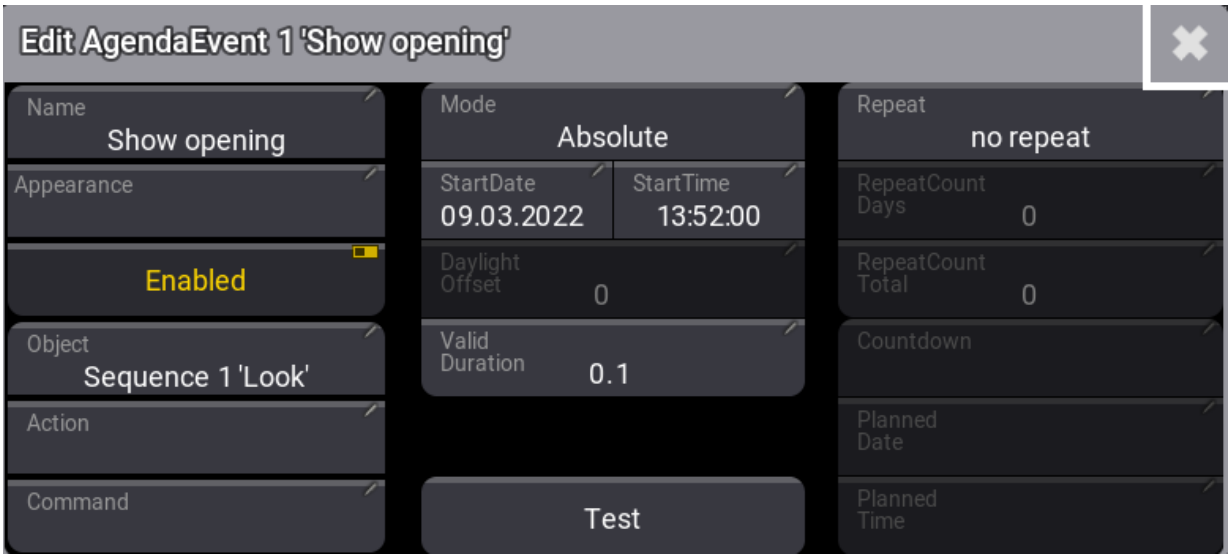
- **Countdown**: Displays the remaining days or time until the next launch of an event.
- **Planned Date**: Displays the next date when the event will be launched.
- **Planned Time**: Displays the next time when the event will be launched,
- **Repeat Count Days** and **Repeat Count Total**: These two columns are similar to the edit repeat pop-up. This is a quick way to verify the repeat pattern.

Tap Test selected in the title bar to execute the selected agenda event immediately. This will verify if the set object or command will be executed correctly.

Tap Delete Old in the title bar to erase all preceded events, including the valid duration.

30.3. Edit an agenda entry

To edit an agenda entry from the **Month**, **Week**, and **Day** layout mode, press Edit, then tap the agenda entry you wish to edit. You can also tap and hold or right-click an agenda entry. The **Edit Agenda Event** pop-up opens.



To learn about the properties that can be defined here, see **Create an agenda entry**.

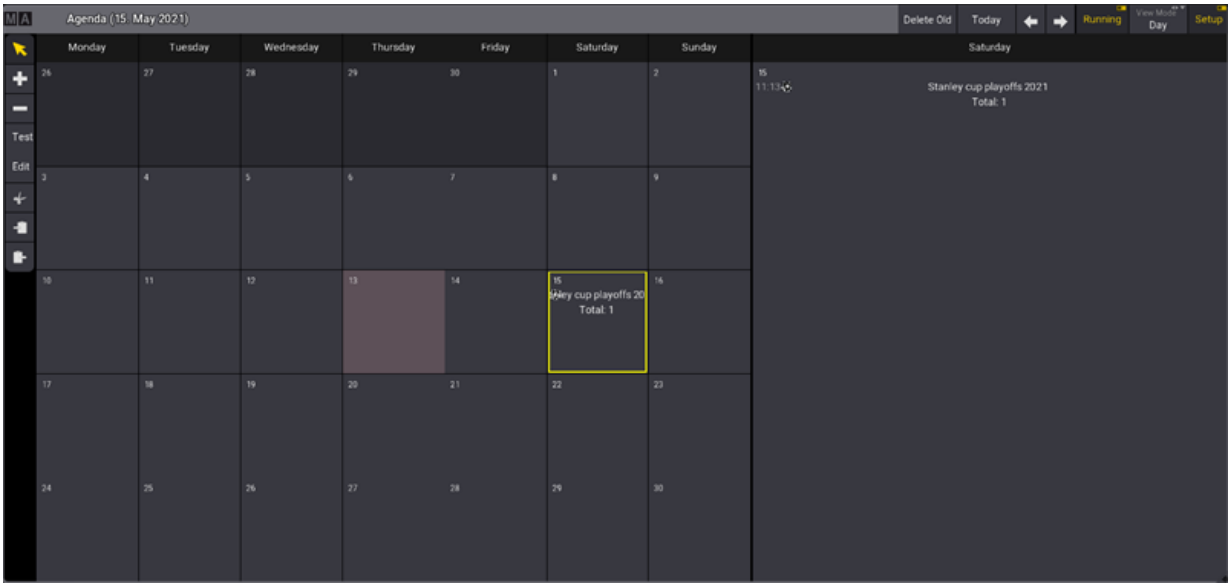
Hint:
Agenda events can also be edited using the toolbar. see, **Agenda toolbar**.

Hint:
Events can be deleted from any view mode by pressing Delete, then tapping the event you wish to delete.

30.4. Agenda Toolbar

Create or edit an agenda entry using the toolbar within the 5 available view modes; **Sheet**, **Year**, **Month**, **Week**, and **Day**. For more information, see **Agenda Modes**.

To enable the toolbar, tap Setup in the title bar.



Agenda day view mode

To select an event, tap in the toolbar, then tap the event you wish to select.

To create an event, tap in the toolbar, then tap a day in the agenda.

To delete an event, tap in the toolbar, then tap the event you wish to delete.

Tap Test in the toolbar, then tap an event to execute it immediately. This will verify if the set object or command will be executed correctly.

Tap Edit in the toolbar, then tap the event you wish to edit. This opens the Edit Agenda pop-up. See **Edit an agenda entry** for more information.

To cut an agenda entry, tap in the toolbar, then tap the entry you wish to cut. Notice that (the paste tool) is selected in the toolbar and awaits for the user to tap a day where to paste the cut entry.

To copy an agenda entry, tap in the toolbar, then tap the day you wish to copy. Notice that (the paste tool) is selected in the toolbar and awaits for the user to tap a day where to paste the copied entry.

Use and in the title bar to change the day, week, month, or year.

Hint:
The day, week, month, or year is displayed in brackets in the title bar according to the selected ViewMode.

Tapping Today in the title bar will set the agenda to today's date.

31. Timecode

Timecode is a time signal used to let off the recordings of a playback or trigger cues.

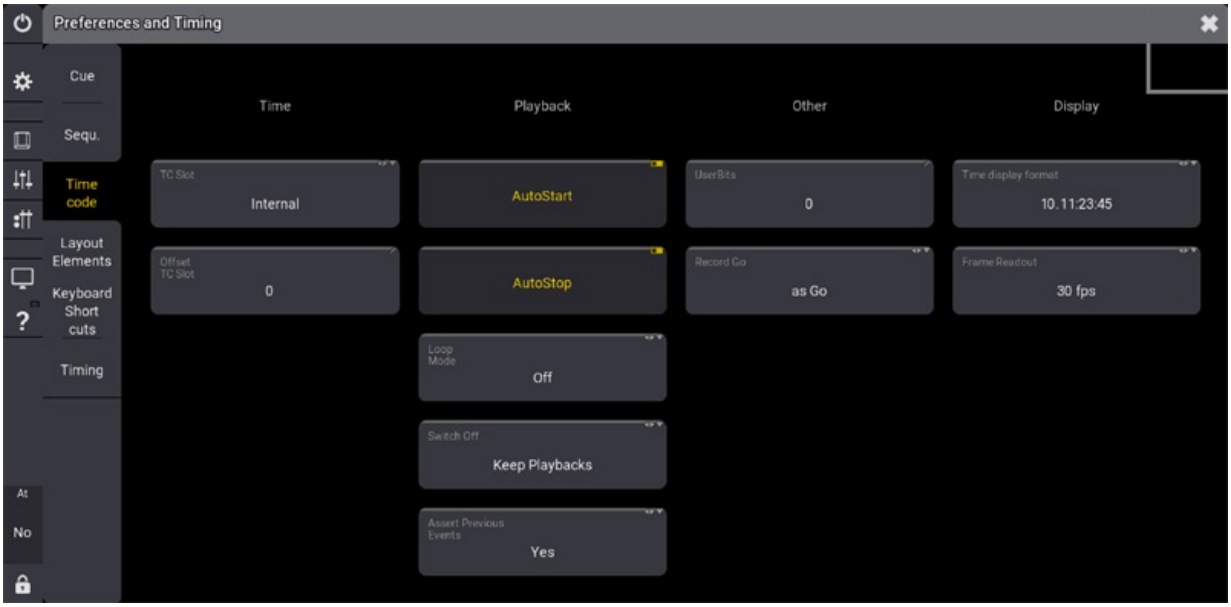
This time signal can either be generated using an external source, (SMPTE), (MIDI), or (Art-Net) timecode or it can be generated internally using the console.

The grandMA3 can receive up to 8 external timecode signals at the same time.

These timecode signals are located in **Timecode Slots**.

Default timecode settings can be set from the Preferences and Timing menu.

Press Menu, then tap Preferences and Timing. On the left side of the window, tap Timecode.



31.1. What are Timecode Slots

A timecode slot is an integrated interface that interprets a timecode signal in hours (h), minutes (m), seconds (s), and frames (f).
The grandMA3 can receive up to 8 different external timecode signals at the same time.

Slots are located in the Timecode Slots pool. For information on Pools, see **Pool windows**.

For information on the timecode slots pool settings, see **Window settings**.

Clock

The clock can be used to display the time of a timecode slot.

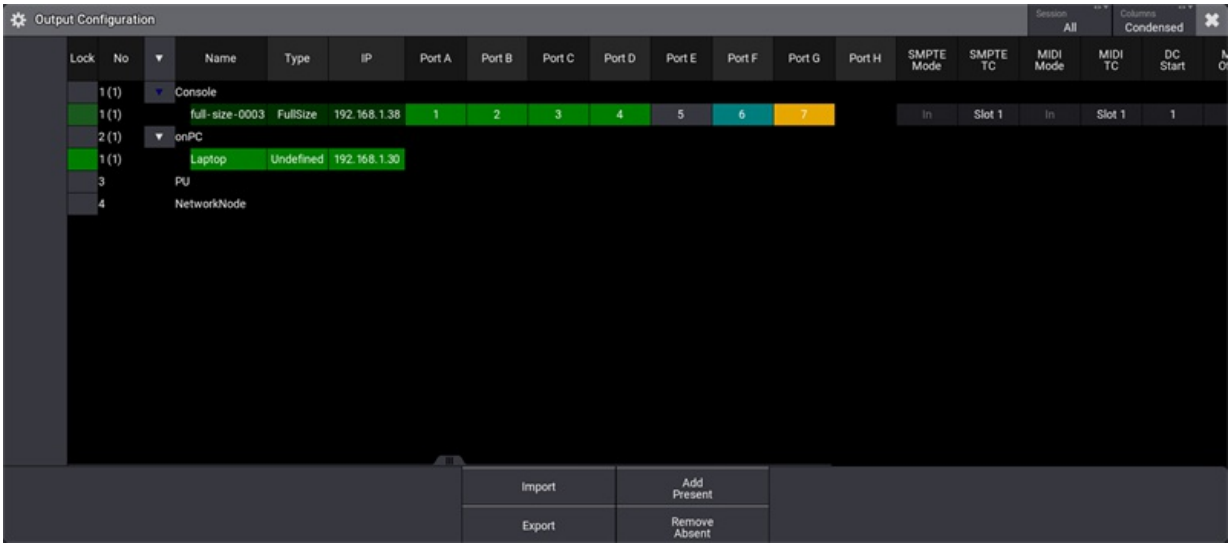
Open the clock window. For more information, see **Add windows**.

For general information on the clock, see **System – Clock**.

1. In the title bar of the clock window, toggle ClockSource to **Timecode**.
2. To select the timecode slot, tap Slot in the title bar of the Clock.
3. Choosing **<Selected>** will display time from the selected slot in the timecode pool.

The clock window buttons can be used to control the timecode slot signal sent to external receptors—media servers, for example.
To do so, press Menu, then tap Output Configuration.

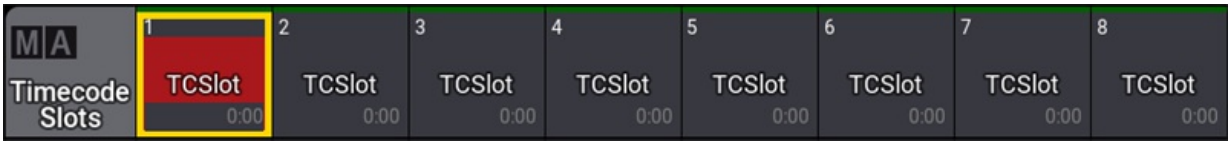
1. Tap and hold a cell in the SMPTE Mode or MIDI Mode column and set it to **Out**.
2. Tap and hold a cell in the SMPTE TC or MIDI TC column to open the drop-down menu.
3. Slide your finger to the desired slot.



Important:
Timecode slot settings are no part of the show file. Therefore, they will not be transmitted within the show file to another console.

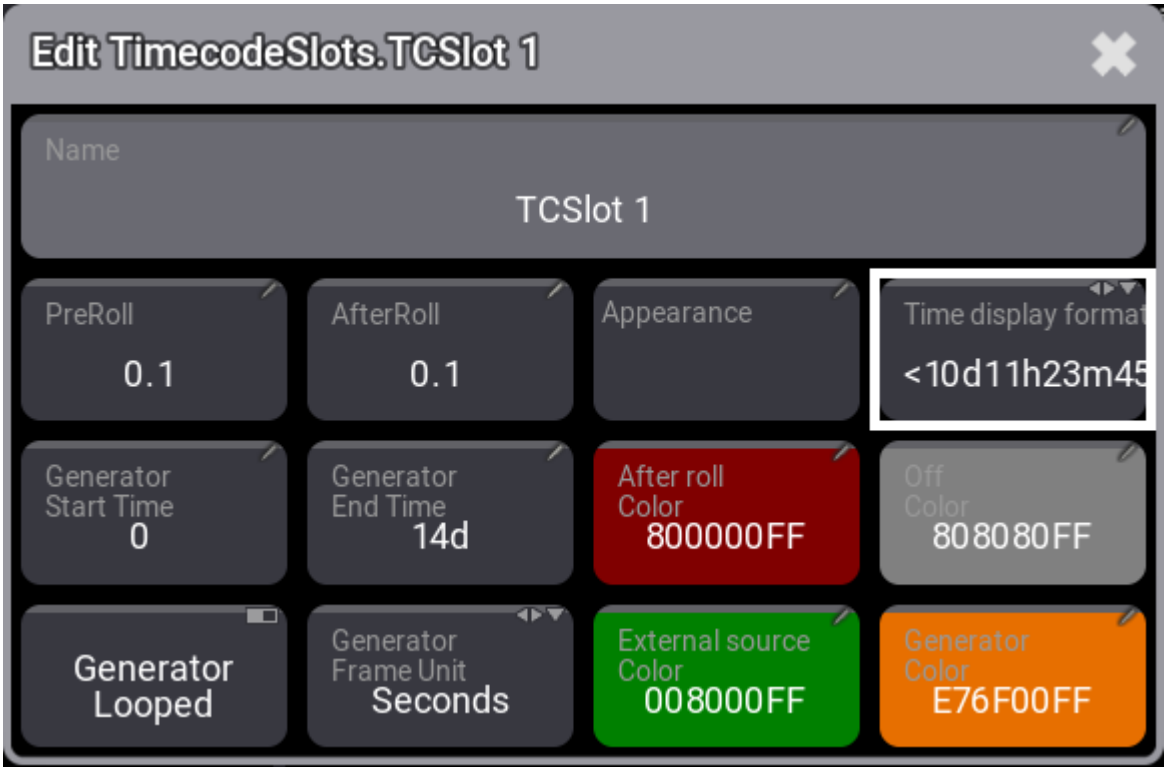
Important:
The 8 timecode slots are hard-coded. That is, they cannot be added, copied, deleted, or moved. However, they can be edited.

Important:
ArtTimeCode can be sent to timecode slots via Art-Net. For more information on the configuration, see **Ethernet DMX – Art-Net Menu**.



Timecode Slots pool

To edit a timecode slot, use the swipey command or press Edit and tap a timecode slot. The pop-up Edit Timecode Slot opens.



Edit a Timecode Slot

Important:
-PreRoll and AfterRoll are primarily used for external signals.
-Generator times are primarily used for internal signals.

PreRoll:
States how long a signal is coming in until the console uses it.

AfterRoll:
If the external signal stops, the timecode goes on internally in the AfterRoll. For example, if 10 seconds were set, the time runs for another 10 seconds, even though the external signal ceased.

Appearance:
Sets the appearance of the object in the timecode slot pool. This appearance will also apply to the background of the cursor button in the title bar of the timecode window. This is a quick visual way of knowing what timecode slot a timecode show is getting the signal from. For more information, see **Timecode settings**.

Time display format:
Select the Time display format, either including days or only in hours and two different formats.

Generator Start Time:
Sets the start time for the internal timecode generator.

Generator End Time:
Sets the time at which the internal timecode generator stops.

AfterRoll Color:
Sets the timecode slot color to indicate that the external signal was interrupted and that the timecode slot is in AfterRoll.

Off Color:
Sets the color of the clock when no signal is coming in. (Clock stopped)

Generator Looped:
Repeats the internal timecode indefinitely.

Generator frame Unit:
Sets at which frame rate the signal is being emitted.

External source Color:
Sets the color of the clock for the external timecode source.

Generator Color:
Sets the color of the clock for the internal timecode source.

31.2. Timecode Settings

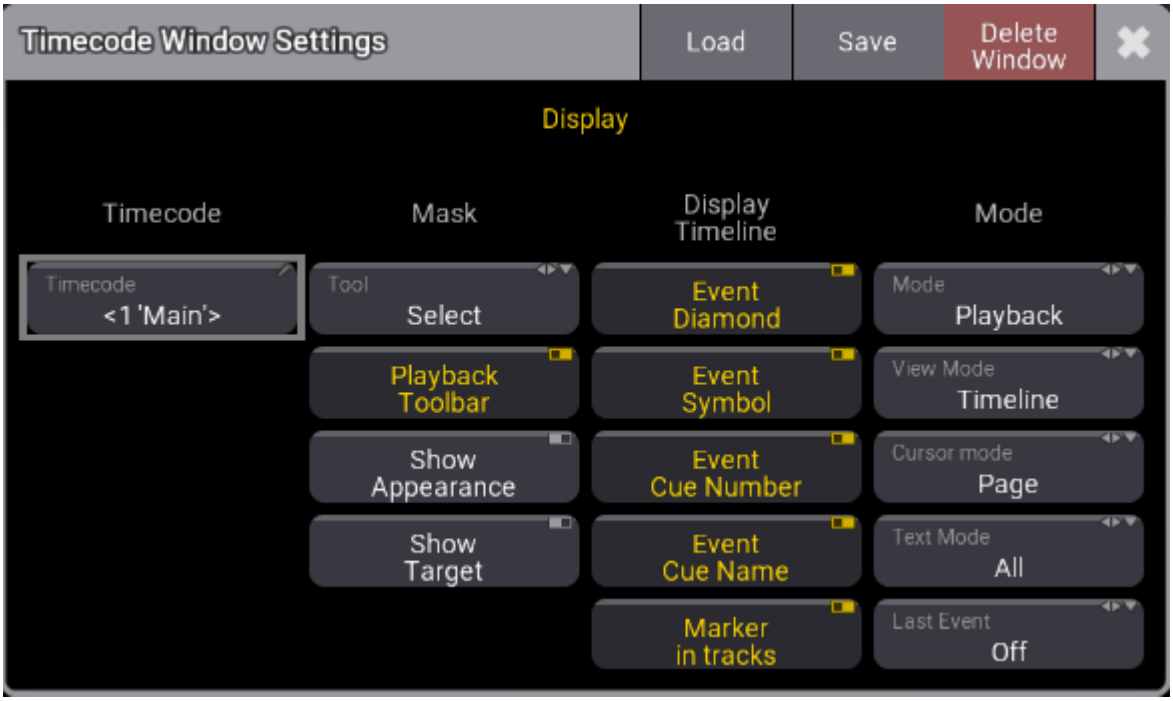
There are two different settings in a timecode:

- Window settings
- Timecode show settings

Open a timecode window. For more information, see **Add windows**.

Open the timecode window settings by taping the MA logo in the title bar.

Timecode Window Settings



Timecode:
Select the timecode show you would like to use here.

The timecode show is initially stored in the timecode pool. For more information, see **Record a Timecode Show**.

Tool:
Toggles between the tool buttons in the toolbar of the timecode view. The toolbar is visible only in setup mode. For more information, see **Time ranges and events**.

Playback Toolbar:
Enables the toolbar to navigate the timeline and to playback or record the selected timecode show.

Show Appearance:
Toggle on to display the appearance column. For more information, see **Toggle View Mode**.

Show Target:
Toggle to display the target column. For more information, see **Toggle View Mode**.

Event diamond, Event Symbol, Event Cue Number, and Event Cue Name:
Toggle on to display event diamonds, event symbols, event cue numbers, and event cue names on the timeline.

Marker in tracks:
Toggle on to display markers on all tracks of a track group. When toggled off, the markers will be shown in the track group section only.

View Mode:
Tap and hold to open the Select ViewMode drop-down list, then change the timecode window view to **Text**, **Timeline,** or **Both**. For more information, see **Toggle View Mode**.

Text Mode:
When View Mode is set to Both or Text, tap and hold to open the Select TextMode drop-down list, then change the text mode to **All, Tracks**, **Selected,** or **Markers** to filter the text display.

Cursor Mode:
When the stopwatch is enabled in the title bar, the cursor will stay visible when the timecode is running.

- **Center:**
  When an external signal is coming in or generated by the console, the cursor (green timeline) will move until it reaches the middle of the visible area of the timeline. The timeline will start moving to keep the cursor centered in the display.
- **Page:**
  When an external signal is coming in or generated by the console, the cursor (green timeline) will move until it reaches the end of the visible part of the timeline, then a new section of the timeline will be displayed. The cursor will move from the beginning to the end of that new section, and this will repeat until the timecode is stopped.

Hint:
The same applies when moving the cursor using the dual encoders. For more information, see **Edit a Timecode Show**.

Last Event:
Defines if the last played back event in the timecode show is selected.

- **Off:**
  The last played back event will not be selected.

- **Track:**
  The last event that was played back in the selected track will be selected.
- **All:**
  Selects the last played back event no matter of which track.

Timecode Show Settings

To open the timecode show settings, tap Settings in the title bar.

Timecode show settings pop-up opens:

Timecode show settings

TC Slot:
Tap and hold to open the Select TCSlot pop-up list, then select **<Internal>**, **<Selected>**, or **TCSlot 1** thru **TCSlot 8**.
To learn more about Timecode Slots, read **What Are Timcecode Slots**.

Offset TC Slot:
Tap to open the calculator, then set the offset for the timecode slot.

Duration:
Tap to open the calculator, then set the duration of the entire timecode show recording.

Cursor:
Displays the cursor at its current position on the timeline. Tap to open the calculator and enter a new position.

Hint:
The selected timecode slot appearance defines the cursor button background color. To learn more about timecode slots, read **What are timecode slots**.

Auto Start:
Starts the timecode show automatically when an external signal is received.

Auto Stop:
Automatically stops the timecode show when an external signal is stoped.

Loop Mode:
When the timecode is internally generated:

- **Loop**:
  The timecode show will repeat playing from the beginning to the end. Press the stop button in the playback bar to stop the timecode show.
- **Pause**:
  The timecode show will pause at the end, press the start button in the playback bar to restart the timecode show from the beginning.
- **Off**:
  The timecode show will not loop or pause.

Switch Off:
Toggles between Playbacks Off and Keep Playback.

- **Playbacks Off:**
  Switches off all playbacks that were started by the timecode show.
- **Keep Playback:**
  Does not switch off the playbacks that were started by the timecode show.

Assert Previous Events:
When set to Yes, events preceding the play head will be asserted.

Time Display Format:
Select the time display format you want to use **<Default>, 10d11h23m45, 25 1h23m45, 10.11:23:45,** or **251:23:45**

Frame Readout:
Select the frame readout between **<Default>**, **Seconds**, **24 fps**, **25 fps**, **30 fps**, or **60 fps**.

UserBits:
Besides the 32 Bit for 8 digit timecode time, timecode executes 32 **User Bits** (8 digit) per frame. Use User Bits to mark a timecode signal. For example, use User Bit 1 for light and User Bit 2 for pyro. This is not available when using an internal timecode source. Do not change this, unless you are expressively told to do so by the timecode supplier.

Record Go:
When set to **as Go**, events will be recorded with a Go+ action, and with a Goto+ action when set to **as Goto (Status)**.

31.3. Track Groups

Timecode shows are arranged in a hierarchical tree structure where **Track Groups** can contain one or more **Tracks**.

Markers can be assigned to the selected track group. Time ranges are assigned to tracks. For more information, see **Time ranges and events**.

To select a track group, press Select and tap the track group you wish to select. The selected track group has a yellow text label.

Track groups can be expanded and collapsed by tapping the arrow on the left-hand side of the track group name.

Track groups can also be selected via the command line. For example,

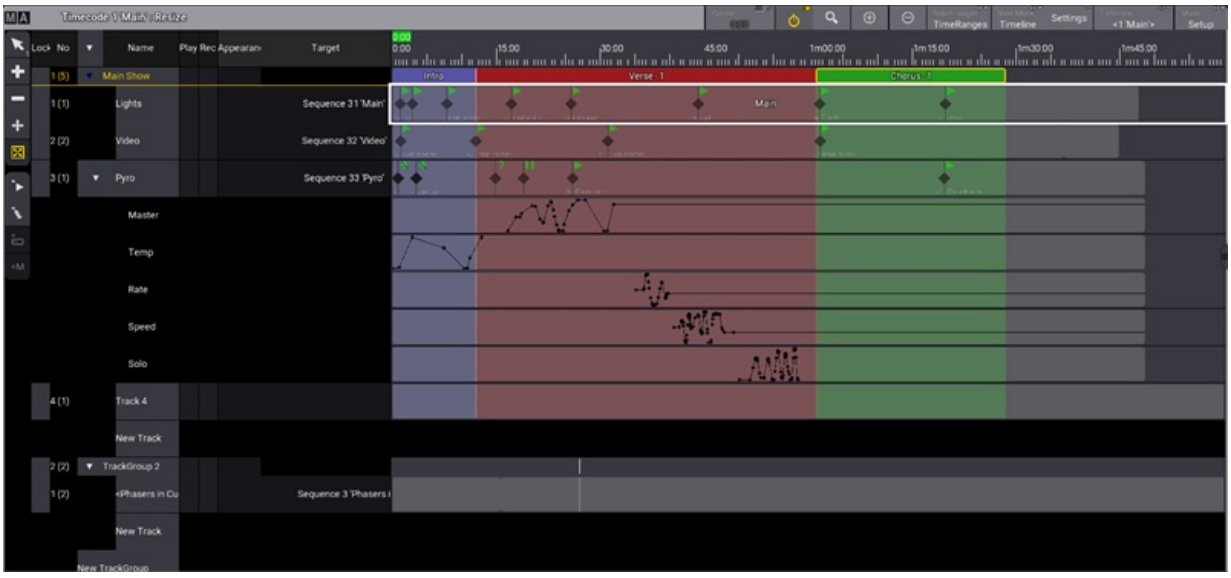
Select track group 2 of timecode 1:

```
User Name[Fixture]> Select Timecode 1.2
```

Tracks can be added to track groups by tapping and holding a cell in the **Target** column. For more information, see **Toggle view mode**.

A track is also added to the selected track group when pressing the corresponding target executor key. For more information, see **Record a timecode show**.

Fader movements are recorded on separated sub-tracks within a track. For example, **Master**, **Rate**, and **Speed**. See the image below.



Recorded fader movements.

31.4. Time Ranges and Events

Time ranges

Time ranges and markers are defined regions drawn on the timeline. They can be labeled and they can get appearances assigned. A marker is created on the timeline section of a track group and will be applied downward to all the tracks in the track group. A time range is created on the timeline section of a track and it is track specific.

Time ranges and markers are tools used to better navigate and visualize the timecode show. They do not affect timecode playback or recording.

When a track is added, a new time range is created for that track and the duration is set to the total length of the timecode show. To set the duration of the timecode show, see **Timecode settings**.

Events

Events are tags positioned on the timeline that can get **Tokens** assigned to.
Tokens can be assigned from the text display of the timecode view or with the event encoder in the encoder bar. For more information, see **Toggle view mode**.

When the cursor (green timeline) reaches an event, the assigned tag is executed.

Edit Time Ranges and Events

Events and time ranges can be selected, added, deleted, moved, resized, and much more using the toolbar.

To enable the timecode toolbar, tap Setup in the title bar.

The toolbar is displayed on the left side of the timecode view and the encoder bar is switched to the timecode encoder bar.

Hint:
Press and hold any tool icon in the toolbar to display a short description in the command line.

Hint:
Some toolbar buttons will be enabled depending on the commands you will execute.

To edit Time Ranges or events using the toolbar, you first need to define the target. To do so, tap and hold Select target in the title bar. This will open the **Selection Target Type** pop-up, then select **Time Ranges** or **Events**.

Tap then tap a time range or an event in the timeline. The selected event is displayed in yellow and the selected time range as a yellow frame.

Hint:
- It is possible to select multiple events or time ranges using the lasso selection method.
- A random selection can be performed by holding down the **Ctrl** key on the keyboard and tapping the desire events or time ranges.

To add an event to a track, tap then tap in the timeline where you want to add the event.
To add a time range, tap then do one of the following:

1. Tap in the timeline where you want to add the time range. When tapping in a **TrackGroup** section of the timeline, a one-frame **Marker** is created. When tapping in a **Track** section of the timeline, a **Time Range** is created from this point to the end of the timecode.
2. Tap in the timeline where you want the time range to start, hold and drag your finger until you have reached the desired length, then release your finger.

Hint:
Events and time ranges can be adjusted in detail and accuracy using the dual encoders. For more information, see **Edit a timecode show**.

Delete an event or a time range by tapping , then tap the event or time range you want to delete.

To move an event or a time range, tap , then tap-hold and drag your selection along the timeline.

Tap , then tap-hold, and drag your finger anywhere on a time range to resize it. Drag to the left to reduce the time range duration or to the right to increase it. This tool is enabled when **Time Ranges** is the selected target.

Events and time ranges can be (cut), (copied), and (pasted).

1. Tap , then tap the event or time range you want to cut or copy.
2. Position the cursor in the timeline where you wish to paste your selection.
3. Select the track where you want to paste your selection.
4. Tap , or then, tap . The selection is cut or copied to the desired track at the cursor position.

Hint:
Events can only be pasted to tracks. Time ranges can be cut or copied from tracks to track groups.

Tap to add an event at the cursor position. This tool is handy to add events while recording.

1. Select **Events** as the target.
2. Select the track you want to add events to.
3. An event is added each time you tap .

Tap to add multiple events at the cursor position.

1. From the title bar, select **Events** as the target.
2. Select the track you want to add events to.
3. Position the cursor in the timeline where you want to add multiple events.
4. Tap , the **Add multiple events** pop-up opens.

5. Enter the number of events you want to add and specify the time interval between them.
6. The events are added.

To add a marker at the cursor position, tap .

1. From the title bar, select **TimeRanges** as the target.
2. Select the track group where you want to add the marker.
3. Position the cursor in the timeline.
4. Tap , the virtual keyboard opens.
5. Enter a name for the new marker and press Please or tap Enter.
6. The new Marker is added.

To quickly add a marker at the cursor position, tap +M. This tool is handy to add markers while recording.

1. From the title bar, select **TimeRanges** as the target.
2. Select the track group where you want to add the marker.
3. A marker is added every time you tap +M.

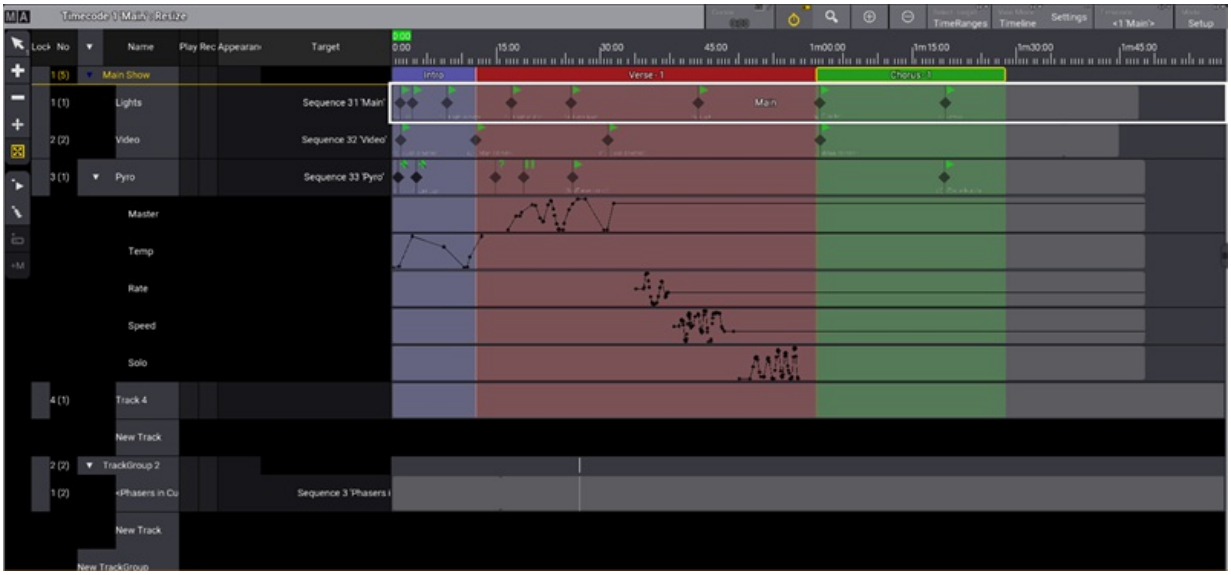
31.5. Toggle View Mode

There are three different view modes of a timecode show.

Tap and hold View Mode in the title bar, then select a mode.

Timeline Mode

Displays the timecode show as a timeline.



Timecode view with view mode set to Timeline

The timecode window's left side is always visible, whether the view mode is set to **Text**, **Timeline**, or **Both**.

Here is a description of that section.

Lock:
Tap and hold a cell in this column and set it to yes to disable editing.

No:
Displays the number of the track. The number of children for the selected track is displayed in brackets.

Name:
Displays the name of the track groups and tracks. Tap and hold a cell in this column to open the keyboard, then set a new name.

Play:
Tap and hold a cell in this column to disable the playback of a track. When disabled, events on that track will not be executed when the timecode is running.

Rec:
Tap and hold a cell in this column to disable the recording of a track. When disabled, events can not be recorded on that track.

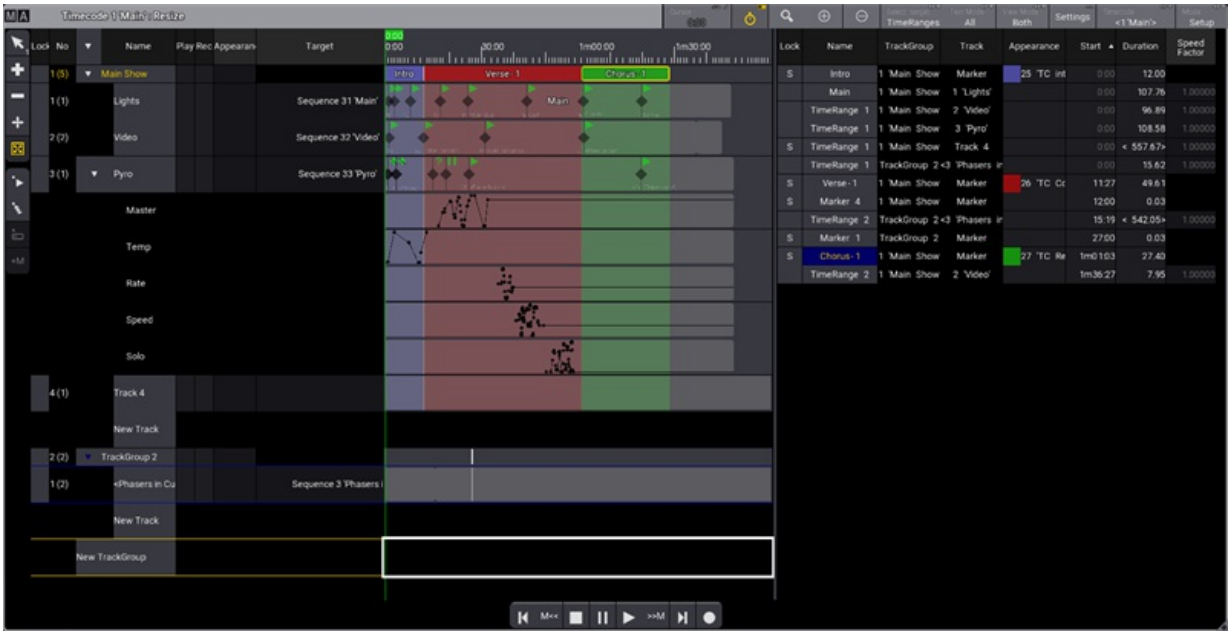
Hint:
The next two columns need to be toggled on to be displayed. For more info, see **Timecode settings**.

Appearance:
Tap and hold a cell in this column. This will open the select appearance pop-up list, then select an appearance. For more information, see **Appearances**.

Target:
Tap and hold a cell in this column to open the assignment editor pop-up menu. Tap Empty, Sequence, Timecode, or TCSlot, then select a target.

Both Mode

Divides the display in two, displaying the timecode show as a timeline on the left side and text on the right.



Timecode view with view mode set to both

Tapping Select target in the title bar will set the focus to **Events** or **TimeRanges**.

Tap Text mode in the title bar to filter the information.:

- **All:**
  - When the target is set to **TimeRanges**, this mode will display information for all markers and time ranges in the timecode show.
  - When the target is set to **Events**, this mode will display information for all events in the timecode show.

- **Tracks:**
  - When the target is set to **TimeRanges**, this mode will display information for markers and time ranges of the selected track.
  - When the target is set to **Events**, this mode will display information for events of the selected track.

- **Selected:**
  - When the target is set to **TimeRanges**, this mode will display information for the selected markers and time ranges.
  - When the target is set to Events, this mode will display information for the selected events.

- **Markers:**
  This mode will display markers information for the selected track group. The target will automatically switch to **TimeRanges**.

This is a description of the column headers in the text display.

Lock:
Set to yes, to disable editing.

Name:
- When the target is set to **Events**, it will display the target's handle that was used to record the event. For more information, see **Executor Configurations**.
- When the target is set to **TimeRanges**, it will display markers, tracks, and track group information.

TrackGroup:
Displays the name of the track groups.

Track:
Displays the name of the tracks.

Time:
Displays in seconds the location of an event on the timeline. Tap and hold a cell in this column to open the calculator, then set a new value.

Token:
Displays the action executed when the event is played back. Tap and hold a cell in this column to open the select token pop-up, then select a token.

Cue Destination:
Displays the cue that will be executed when the event is played back. Tap and hold a cell in this column, then select a new cue.

Status:
Displays the status of an event recorded when an executor was set to flash, black, or temp. For more information, see **Executor Configurations**.

Execute Cmd:
Defines if cue commands are executed or not when an event is played back. Tap and hold a cell in this column and set it to Yes or No. The default is Yes. For more information, see **Cues and sequences — Look at cues in sequences**.

FaderValue:
When recording a track using a fader, the fader movement is recorded in steps between 0% and 100%. This column displays the value of the fader at each step. Tap and hold a cell in this column to open the calculator, then set a new value.

Appearance:
Displays the appearances assigned to markers and time ranges. Tap and hold to open a cell in this column. The select pop-up list will open, then assign a new appearance. For more information, see **Appearances**.

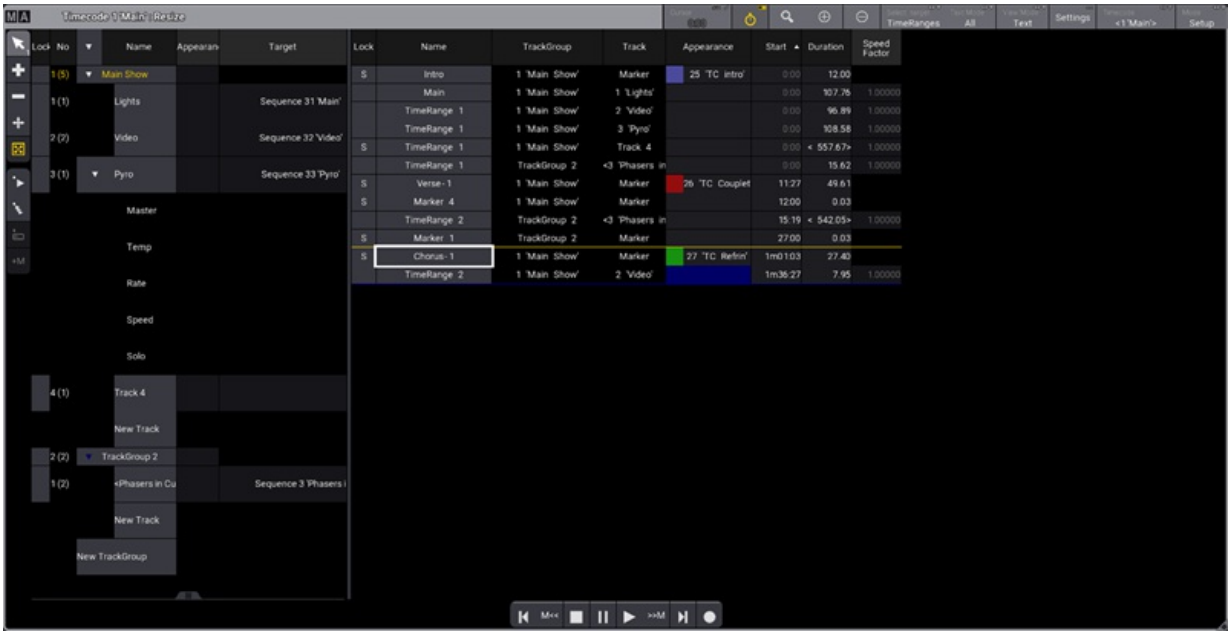
Start:
Displays the start position of markers and time ranges on the timeline. Tap and hold a cell in this column to open the calculator, then set a new value. For more information on **Time display format**, see **Timecode Settings**.

Duration:
Displays in seconds the time duration of markers and time ranges. Tap and hold a cell in this column to open the calculator, then set a new value. The value is displayed in angle brackets < > when the duration is set to the end of the timecode show.

Speed Factor:
Tap and hold to open, then set a value to multiply or divide the speed.

Text Mode

When the timecode view mode is set to **Text**, the timeline is hidden.



Hint:
Only cells with a gray background can be edited.

Hint:
Tap and hold a cell of a column header to sort the display.

31.6. Record a Timecode Show

Requirements:

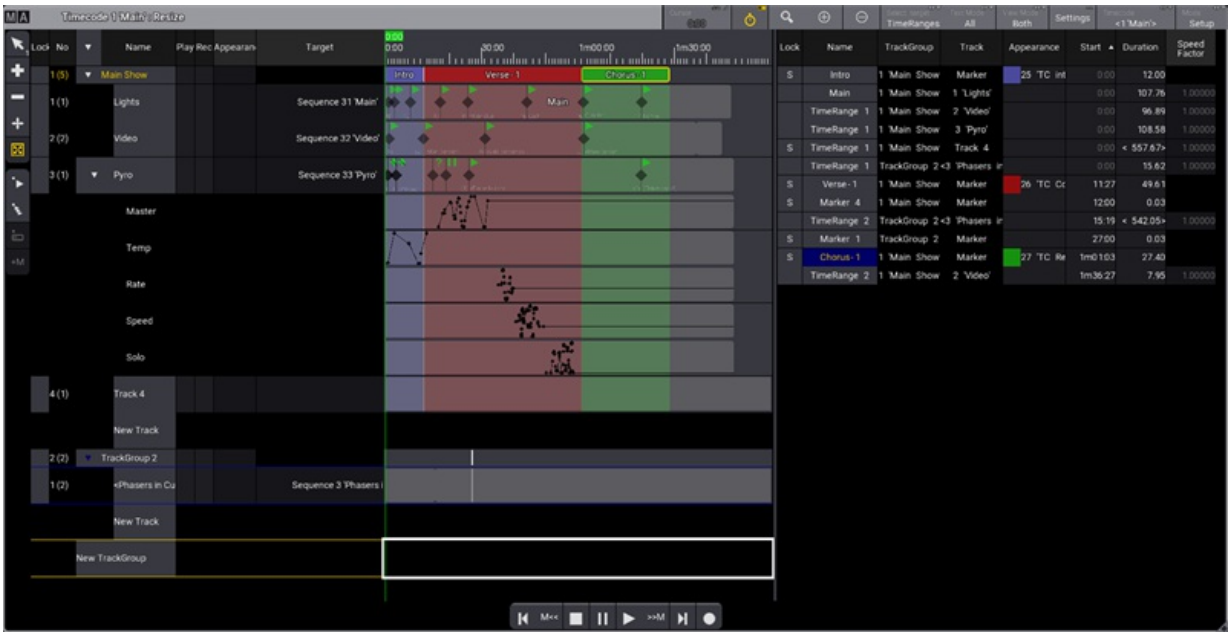
- Patched fixtures. For more information, see **Patch and Fixture Setup**.
- Store values in cues and create a sequence. For more information, see **Cues and Sequences**.
- Open a timecode pool and a timecode view. For more information, see **Add windows**.

Example: Internal Timecode

Important:
This sequence of operations is based on an internally generated timecode.

1. Store a new object in the Timecode Pool and label it.
   For more information, see **Windows, Views, and Menus**.

2. Select the Timecode in the pool.



Timecode window in setup mode

3. In the title bar, tap Mode and set it to **Setup**.
   For more information on different views, see **Toggle view mode**.

4. To enable the follow time cursor, tap the stopwatch in the title bar. When enabled, the view follows the time cursor.
   If it is disabled, the cursor moves beyond the displayed view.

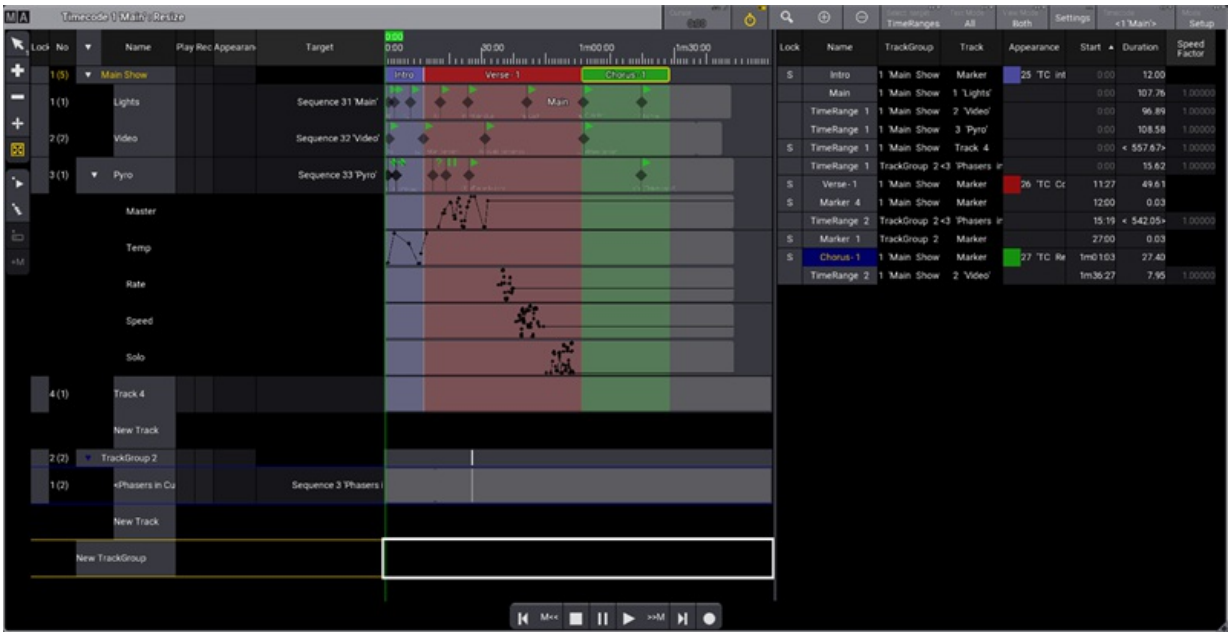
5. In the title bar, tap Settings in the pop-up menu, tap TCSlot and select <Internal>.
   For more information, see **What are timecode slots**.

6. To start recording, tap .
   The time starts to run.

7. To record an event, press the corresponding target executor key. This will create a new track with a new target.

8. To record fader movements, move the corresponding target fader.

9. To finish recording, tap .



Record timecode show

The timecode show is now recorded.

31.7. Record an External Timecode Show

The grandMA3 can receive 3 types of external timecode signals: SMPTE, MIDI (MTC), or Art-Net Timecode (ArtTimeCode).

Requirements:

- Patch fixtures
- Store values in cues and create a sequence. For more information, see **Patch and Fixture Setup** and **Cues and Sequences**.
- A visible timecode window and timecode pool. For more information, see **Add Windows**.

Here are the steps needed to record a timecode show from an externally generated timecode source.

SMPTE:

1. Connect the external SMPTE source to the LTC XLR connector at the back of the console. For more information, see **Device overview**.
2. Press Menu, then tap Output Configuration.
3. In the **SMPTE Mode** column, tap and hold a cell and set the mode to In. (The default mode is In).
4. In the **SMPTE TC** column, tap and hold a cell to open the TC slot pop-up list. (The default is Slot 1).
5. Select the slot that will receive the external SMPTE signal.
6. Store a new object in the Timecode Pool. Label it and select it. For more information, see **Windows, Views, and Menus**.
7. Tap Settings in the title bar of the timecode window. For more information, see **Timecode settings**.
8. Tap TCSlot and select a slot from 1 thru 8. This has to match the slot that was set for **SMPTE TC** in the Output Configuration menu.
9. Tap Mode in the title bar of the timecode view and set it to **Setup**.
10. Assign a new **Target** to a track. For more information, see **Toggle view mode**.
11. Turn on the **Playback Toolbar**. For more information, see **Timecode Settings**.
12. Tap in the Playback Toolbar to start recording.

Hint:
At this point, the timecode show is pending. The cursor will start moving when the external signal is detected.

13. To record a cue, press the corresponding target executor key. For more information, see **Toggle view mode**.
14. To record fader movements, move the corresponding target fader.
15. To finish recording, tap in the toolbar.

MIDI:

1. Connect the external MIDI source to the MIDI In connector at the back of the console. For more information, see **Device overview**.
2. Press Menu, then tap Output Configuration.

3. In the **MIDI Mode** column, tap and hold a cell and set the mode to In. (The default mode is In).
4. In the **MIDI TC** column, tap and hold a cell to open the TC slot pop-up list. (The default is Slot 1).
5. Select the slot that will receive the external MIDI signal.
6. Store a new object in the Timecode Pool. Label it and select it. For more information, see **Windows, Views, and Menus**.
7. Tap Settings in the title bar of the timecode window. For more information, see **Timecode settings**.
8. Tap TCSlot and select a slot from 1 thru 8. This has to match the slot that was set for **MIDI in** the Output Configuration menu.
9. Tap Mode in the title bar of the timecode view and set it to **Setup**.
10. Assign a new **Target** to a track. For more information, see **Toggle view mode**.
11. Turn on the **Playback Toolbar**. For more information, see **Timecode Settings**.
12. Tap in the Playback Toolbar to start recording.

Hint:
At this point, the timecode show is pending. The cursor will start moving when the external signal is detected.

13. To record a cue, press the corresponding target executor key. For more information, see **Toggle view mode**.
14. To record fader movements, move the corresponding target fader.
15. To finish recording, tap in the toolbar.

ArtTimeCode:

1. Connect a cable to one of the 3 Ethernet interfaces at the back of the console.
2. Press Menu, then tap DMX Protocols.
3. Tap Art-Net in the top left corner.
4. In the **Mode column**, tap and hold a cell; from the pop-up list, tap Input.
5. In the **Timecode Slot** Column, tap and hold a cell. This will open the calculator. Enter the slot number that will receive the external signal.
6. Set the Interface IP address and Enable Input. For more information, see **Art-Net menu**.
7. Store a new object in the Timecode Pool. Label it and select it. For more information, see **Windows, Views, and Menus**.
8. Tap Settings in the title bar of the timecode window. For more information, see **Timecode settings**.
9. Tap TCSlot and select a slot from 1 thru 8. This has to match the slot number that was set in the Art-Net configuration menu.
10. Tap Mode in the title bar of the timecode view and set it to **Setup**.
11. Assign a new **Target** to a track. For more information, see **Toggle view mode**.
12. Turn on the **Playback Toolbar**. For more information, see **Timecode Settings**.
13. Tap in the Playback Toolbar to start recording.
14. 

Hint:
At this point, the timecode show is pending. The cursor will start moving when the external signal is detected.

15. To record a cue, press the corresponding target executor key. For more information, see **Toggle view mode**.
16. To record fader movements, move the corresponding target fader.
17. To finish recording, tap  in the toolbar.

31.8. Edit a Timecode Show

Requirement:

- Record a timecode show.

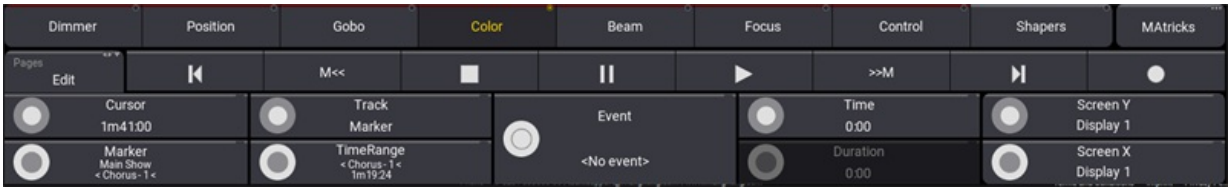
For more information, see **Record a Timecode Show**.

It is possible to edit an existing timecode show to adjust details and accuracy in time.

Important:
A track is created for each sequence, and an event is created for each cue.

1. Tap Mode in the title bar to enter **Setup** mode. The timecode encoder bar is displayed.
2. Use the dual encoders to navigate the tracks and select events or time ranges.

Hint:
It is possible to select multiple events, markers, or time ranges using the lasso selection method. The number on the right of the tool button indicates the number of selected events.



Timecode encoder bar

Edit Using Dual Encoders

Cursor: (First dual encoder)

- Rotate the dual encoder inner ring clockwise or counterclockwise to move the cursor (green timeline) on the time axis.
- Press and rotate the dual encoder inner ring to move the cursor twice as fast.
- Press the dual encoder inner ring to open the calculator and enter a new value.

Marker: (First dual encoder)

- Rotate the dual encoder outer ring clockwise or counterclockwise to select markers on the selected track group.
- Press the dual encoder key, and select a marker from the drop-down list.

Track: (Second dual encoder)

- Rotate the dual encoder inner ring to select a track group or a track.

- Press the dual encoder inner ring, and select a track from the drop-down list.

TimeRange: (Second dual encoder)

- Rotate the dual encoder outer ring to select a time range on the selected track or track group.
- Press the dual encoder key, and select a time range from the drop-down list.

Event: (Third dual encoder)

- Rotate the inner or outer ring of the dual encoder to select an event on the selected track.
- Press the inner ring of the dual encoder and select a token from the drop-down list.
- Press the dual encoder key, and select an event from the drop-down list.

Time: (Fourth dual encoder)

- When an event is selected, rotate the dual encoder inner ring to move the event on the timeline.
- When a marker or a time range is selected, rotate the dual encoder inner ring to adjust the starting point.

Duration: (Fourth dual encoder)

- Rotate the dual encoder outer ring to adjust the duration of the selected marker or time range.

32. Layouts

Layouts are basically two-dimensional drafts where it is possible to arrange fixtures, macros, groups, and other pool objects.

Layouts are created in the layout pool and are displayed and edited in the layout window.

Restriction:
The maximum number of elements per layout is 10 000.

Layouts are stored in the layout pool.

To set the defaults for new layout elements, press Menu, tap Preferences and Timing, then tap Layout Elements on the left side of the window.

32.1. Create a Layout

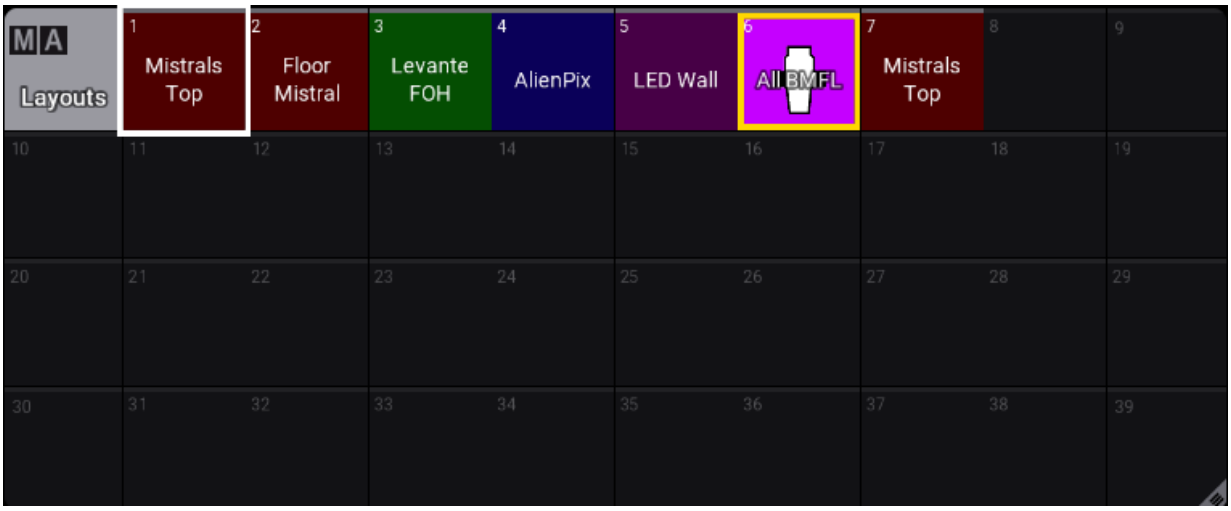
It is possible to create layouts for every need and taste.

Important:
Create a layout in the layout pool and edit the layout in the layout window. For more information, see **Edit layout**.

To add a layout pool window, follow the instructions under **Add windows**.

In the Add Window pop-up:

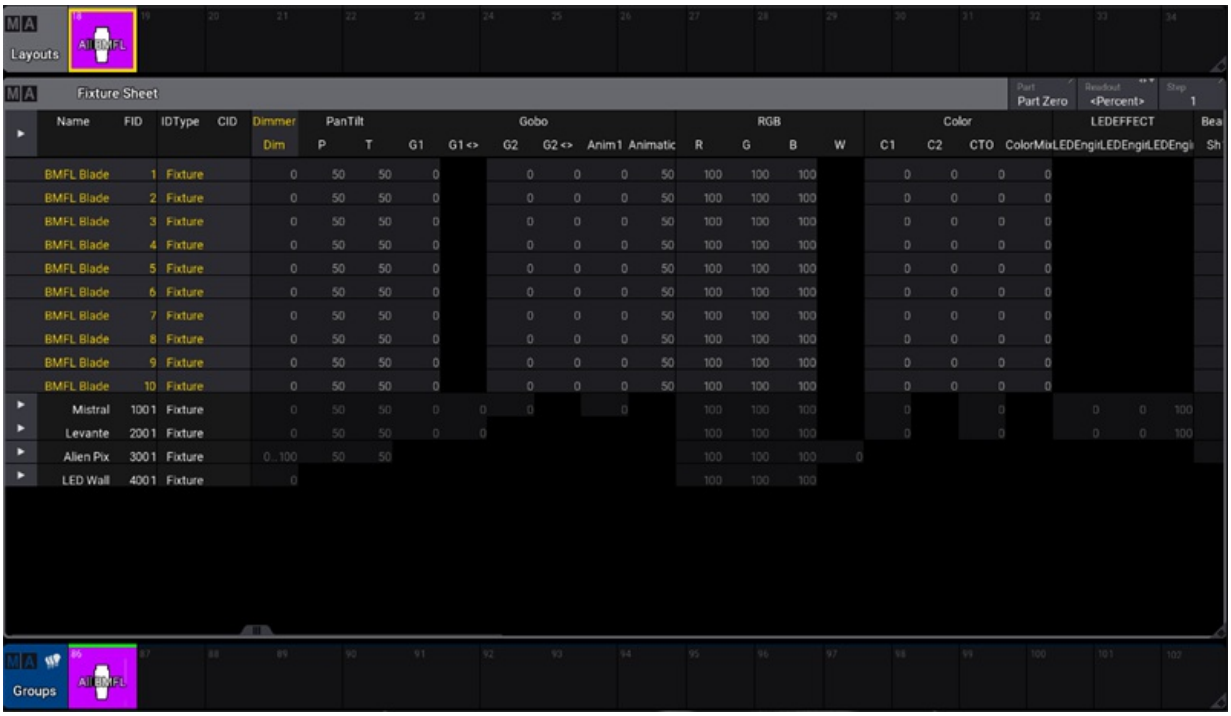
1. Tap Pools.
2. Tap Layouts.
   The layout pool opens.



Layout pool window

3. Select fixtures using the fixture sheet or select a group.

Hint:
The selection grid position of the selected fixtures is assigned to layouts. For more information, see **Selection Grid**.

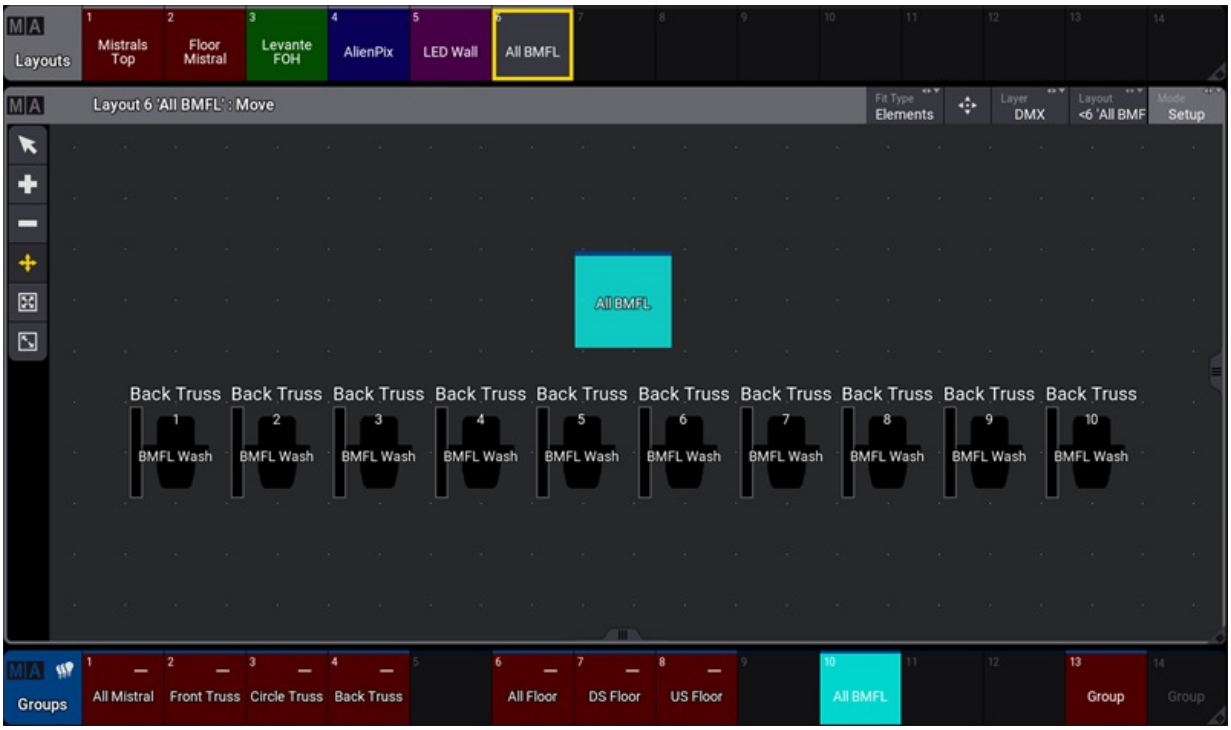


Screen arranged with layout pool, fixture sheet, and group pool

The same fixture can only be assigned once into a layout. A message box will inform the user when he tries to add the same fixture again and cancel this fixture operation.

4. To add the selected fixtures to a layout, press Assign and tap a cell in the layout pool or tap the area in the layout window to which you would like to assign the fixtures.
   The fixtures are added to the layout.
   To add pool elements in the layout window, press, for instance, Assign Group 3 and then tap the area of the layout window to which you would like to assign this group.
   The pool element is added to the layout window.

Hint:
To operate the layouts faster and to have all at a glance, arrange the layout pool and the layout window on one screen.



Screen arranged with layout pool, layout window, and group pool

Tap Mode in the title bar to change the mode to **Setup** enable the toolbar on the left side of the layout view.

Select an Element

1. To select an element, tap .
2. Tap the element you wish to select or use the lasso command to select multiple elements.
3. The element is selected.

Add an Element

1. To add an element in the layout window, tap .
2. Tap an empty space in the layout window.
3. The element is added.
4. To add an object to the element, see **Edit layout view**.

Delete an Element

1. To delete an element in the layout window, tap .
2. Tap the element you want to delete.
3. The element is deleted.

Move an Element

1. To move an element within the layout window, tap .

2. Tap, hold and drag the element to move it.
   The element is moved.

Resize an Element

1. To resize an element, tap .
2. Select the element you would like to resize.
3. Tap, hold and drag to resize.
   The element is resized.

Resize an Element with Fixed Ratio

1. To resize an element and preserving the actual size ratio, tap .
2. Select the element you would like to resize.
3. Tap, hold and drag to resize.
   The element is resized with a fixed ratio.

Layout View Fit Options

In the title bar, tap Fit Type and select the fit option you wish to perform.

- Select Elements to fit all layout elements in the window.
- Select Canvas to fit the entire canvas in the window. If some elements are assigned far outside the canvas, they will not be displayed.
- Selecting Both will fit the canvas and all elements.
- Tap to execute "Zoom to Fit" depending on the selected Fit Type.

Hint:
The canvas fit mode can be modified. For more information, see **Layout View Settings**.

- Create a layout with the command line using the layout keyword. For more information, see **Layout keyword** and **Assign keyword**.

```
User name[Fixture]> Assign Layout 1
```

32.2. Edit Layout

Requirement:
First, create a layout. For more information, see **Create a Layout**.

To edit a layout, use the swipey command or press Edit, then tap the layout pool object you wish to edit.

The Edit Layout overlay opens.

Edit layout window

The following is a description of the softkeys found in the Edit Layout window:

To rename the layout, tap Name, enter the new name, tap Enter on the virtual keyboard, or press Please. For more information, see **Label pool objects**.

To add a scribble to the layout pool object, tap Scribble. For more information, see **Scribbles**.

To add an appearance to the layout pool object, tap Pool Appearance. For more information, see **Appearances**.

To add a canvas appearance to the layout, tap Canvas Appearance. This will apply to the layout window background. For more information, see **Appearances**.

Hint:
If no dedicated canvas appearance is set, the pool appearance will be taken as canvas appearance.

Tap the following keys to set the position and size of the canvas:

- To set the start X position, tap PositionX.

- To set the start Y position, tap PositionY.
- To set the width dimension, tap DimensionW.
- To set the horizontal dimension, tap DimensionH.

When ViewPosActive is switched on, loading a layout again will recall the previously stored view zoom and position values. This button is switched off by default.

Hint:
This feature allows you to store two layouts with the same fixtures but with different positions and zoom values.

ViewPosX, ViewPosY, and ViewPosScale are the stored values for the layout view position and zoom.

Layout Element Defaults

To set the default values for layout elements, press Edit, then tap any layout pool object. This will open the Edit Layout menu, tap Layout Element Defaults in the upper left corner.

Layout element defaults are part of the user profile and will apply every time you assign an object into a layout.

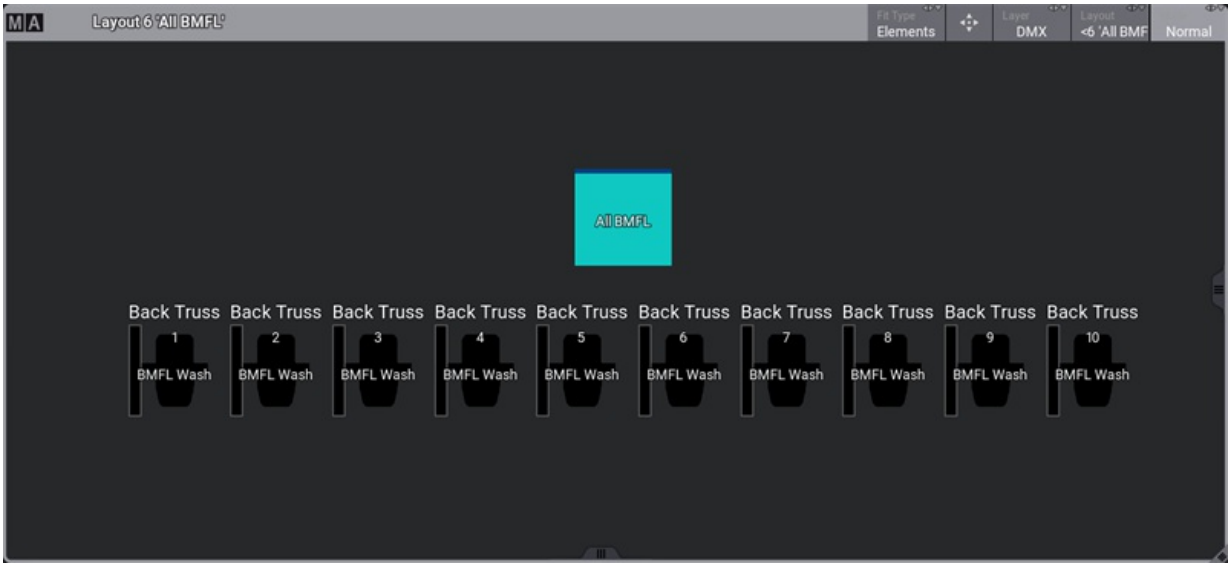
Hint:
This can also be done from the main menu, press Menu, then tap Preferences and Timing. On the left-hand side, tap Layout Elements.

32.3. Layout View Settings

To open the Layout View window, follow the instructions under **Add windows**.

In the Add Window pop-up:

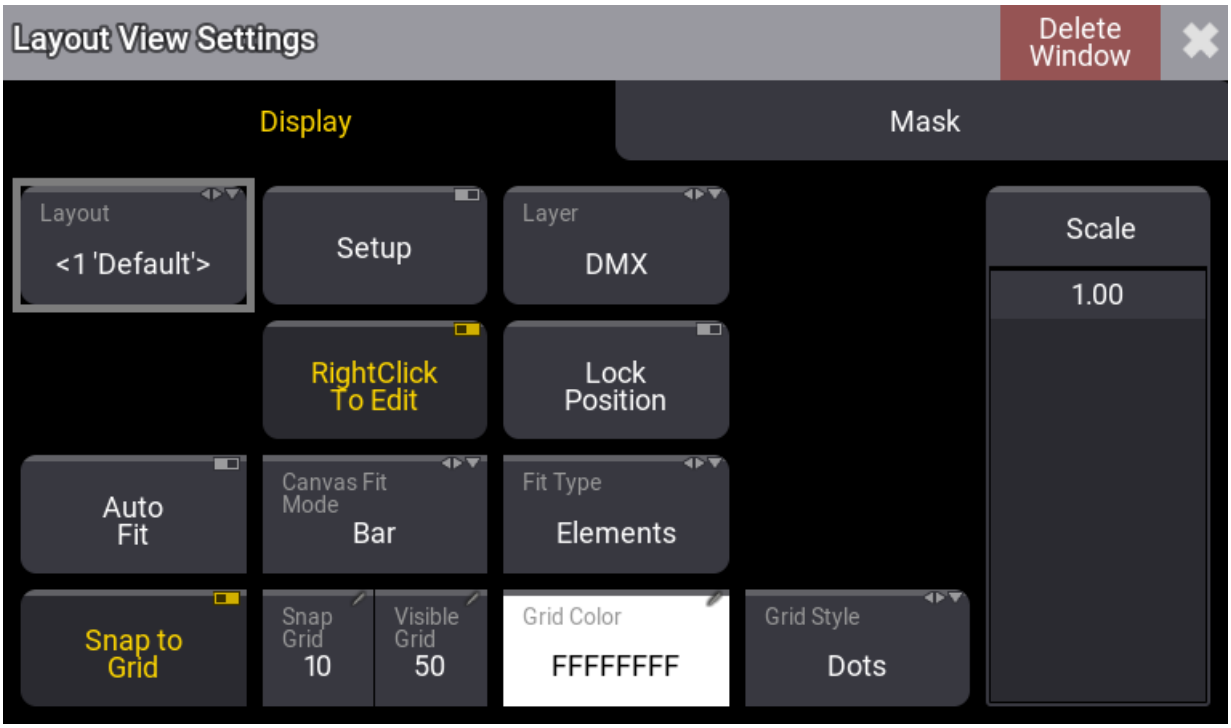
1. Tap Common.
2. Tap Layout View.



Layout view window

Layout View Settings

- To open the Layout View Settings, tap in the upper left corner of the layout window.
  The Layout View Settings pop-up opens.



Layout View Setting Options

This is a short description of each button in the layout view settings:

- **Layout**:
  Link selected, select an existing layout, or create a new one.

- **Layer**:
  Select the layer readout between **DMX**, **Value**, or **Output**.

- **Use RightClick To Edit**:
  When this button is toggled on, it will allow editing of objects assigned to layout elements by right-clicking or using the two-finger edit gesture.
  Toggle this button off to disable the right-click and two-finger edit gesture to prevent accidental editing of objects.

Hint:
To edit an object when this button is toggled off, press Edit and tap the assigned object you wish to edit.

- **Lock Position**:
  Toggle on to protect your layout arrangement.

- **Auto Fit**:
  Toggle on to fit elements according to the window size.

- **Canvas Fit Mode**:
  There are 3 fit modes, **Bar** will fit the entire canvas, **Crop** fits the canvas to its total width, and **Stretch** fits the canvas to its total height.

- **Fit Type**:
  Toggle to fit the **Elements**, the **Canvas**, or **Both**.

- **Snap To Grid**:
  Toggle on for objects to snap to the grid when in setup mode.

- **Snap Grid**:
  Tap to open the calculator, then enter a snap value between 1 and 14 000 pixels.
  Make this number smaller to be as precise as possible when moving elements in the layout view.

- **Visible grid**:
  Tap to open the calculator, then enter a value for the visible grid size.

- **Grid Color**:
  Use the edit grid color popup to create the desired grid color.

- **Grid Style**:
  Select the grid style between **Off**, **Lines**, or **Dots**.

- **Scale**:
  When Auto Fit is off, the fader can be used to scale the layout.

32.4. Edit layout view

To open the layout view, follow the instructions under **Add windows**.

In the Add Window pop-up:

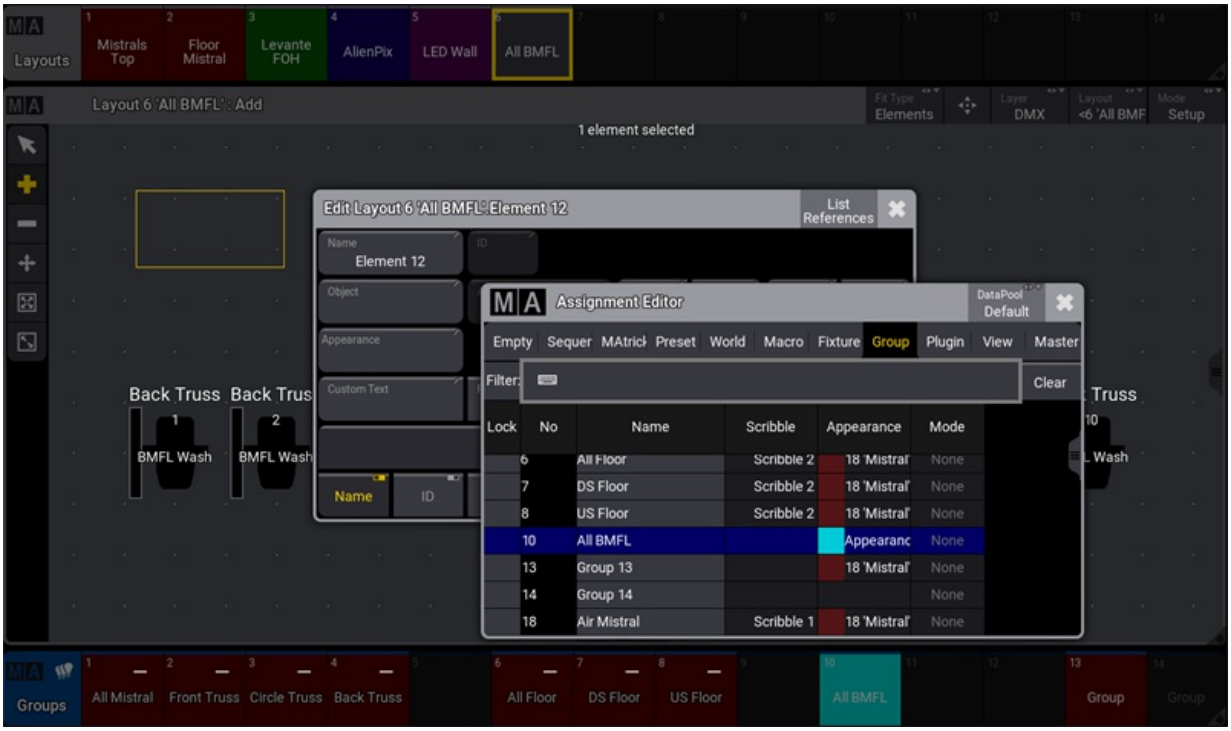
1. Tap Common.
2. Tap Layout View.
   The layout view opens.

> **Important:**
> Setting the Mode to **Setup** in the layout view allows the editing of single layout elements. Set the Mode to **Normal** to allow editing as in the programmer.

Add a Layout Element

In the layout window title bar, tap Mode to enable the setup mode.

- To add a layout element in the layout window, tap +.
- Tap an empty area in the window, a new empty layout element is added.
- To assign a function/object to the element you have created in the layout, tap in the toolbar and open the new element with the **2-finger edit**.
- Tap Object, and the assignment editor opens. Select the desired object type (Fixture, World, Macro, etc.).



Adding an Element Using the GUI

The group pool and the Layout window should be visible on the screen for this.

For example, add group 1 to layout 1:

1. Press Assign.
2. Tap group 1 in the group pool.
3. Tap an empty area in the layout window.

Hint:
This can be done in setup mode and normal mode.

Adding an Element Using the Command Line

```
User name[Fixture]> Assign Macro 1 At Layout 1
```

Layout Toolbar

Use the toolbar to edit elements. For more information, see **Create a layout**

32.5. Edit layout elements

To edit a layout element:

1. Tap Mode to activate the setup mode.
2. Tap and open the element with the **2-finger edit**.

This is a short description of each of the buttons in the edit layout pop-up:

- **Name**:
  Name of the element.
- **ID**:
  - When a fixture is assigned to a layout, the fixture ID is displayed here. For more information, see **What are fixtures**.
  - When a pool object is assigned, ID represents the pool object number.
- **CID**:
  The custom ID number. For more information, see **What are fixtures**.
- **Object**:
  Opens the assignment editor.
- **Action:**
  Select an action to be performed when tapping the element.
- **PaddingTop, Bottom, Left, Right**:
  Padding is the distance between the edge of the element and the content inside the element.
- **PositionX, Y**:
  Specify the position along X and Y in the layout.

- **Appearance**:
  Assign an appearance. For more information, see **Appearances**.
- **PositionW**, **H**:
  This is to adjust the dimensions of the element.
- **Custom Text:**
  Add a custom label to the element.
- **Font Size**, **Text Color**:
  Use these buttons to personalize the custom text size and color.
- **Text Horizontal**:
  The horizontal custom text alignement.
  - **Center**: Will align the text to the horizontal center of the layout element.
  - **Left**: The left end of the text will be aligned with the left edge of the layout element.
  - **Right**: The right end of the text will be aligned with the right edge of the layout element.
- **Text Vertical:**
  The vertical custom text alignment.
  - **Center**: Will align the text to the vertical center of the layout element.
  - **Top**: Places the text within the layout element at the top edge.
  - **Bottom**: Places the text within the layout element at the bottom edge.
  - **Above**: Places the text outside the layout element at the top edge.
- **Border Color** and **BorderSize**:
  Adjust the outline color and size.
- **Element Visible**:
  Toggle to display or hide the element.
- **Name**, **ID**, **CID**, **Bar**, **Value,** and **Border**:
  Toggle to display or hide the name, the fixture ID, the custom ID, the dimmer bar, the output value, and the border of the layout element.
- **Indicator Bar**:
  When a pool object is assigned to a layout (e.g., a group), it defines if the indicator bar will be displayed on top of the element.
- **Selection Relevance**:
  When enabled, the image of the layout element will be colored with the For All or For Some color. For more information, see **Window settings, Pool Settings**.

32.6. Layout encoder bar

Use the layout encoder bar to position and arrange layout elements.

Requirements:

- A visible layout pool and a visible layout window.

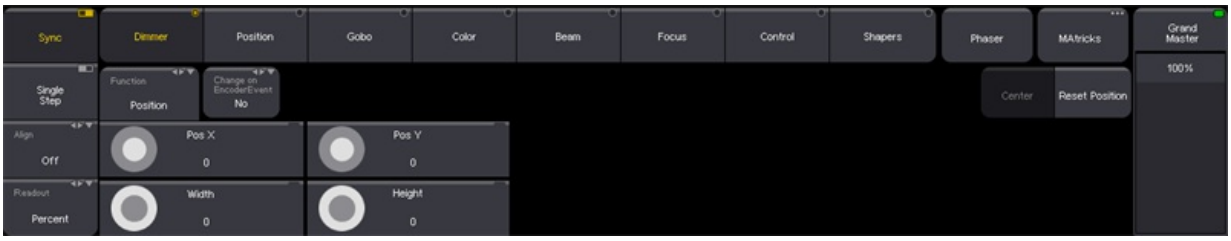
For more information on how to add a window, see **Add windows**.

There are two function settings for the layout encoder bar: **Position** and **Arrangement**.

Layout Encoder Bar Function set to Position

1. Create a layout and select it. See **Create a layout**.
2. Tap Mode in the title bar to enable **Setup**; the layout encoder bar is displayed.
3. Tap Function in the layout encoder bar until it is set to **Position**.
4. Select the layout element (e.g., fixture) you wish to position.
5. Use the encoders to position and/or resize the element.

Hint:
When multiple elements are selected, use the **Align** function combined with the dual encoders to spread out your selection evenly.



Layout encoder bar function set to position

- **Pos X**: Rotate the inner ring of the dual encoder to move the selected element left and right.
- **Pos Y**: Rotate the inner ring of the dual encoder to move the selected element up and down.
- **Width**: Rotate the outer ring of the dual encoder to set the width of the selected element.
- **Height**: Rotate the outer ring of the dual encoder to set the height of the selected element.
- **Change on EncoderEvent**: When set to **Yes**, encoder changes are instantly transmitted to the network. If set to **No**, after stopping to turn an encoder the actual state will be transmitted after 2s.

- **Reset Position**: Tap to position the bottom left corner of the selected elements to Pos X and Pos Y zero.

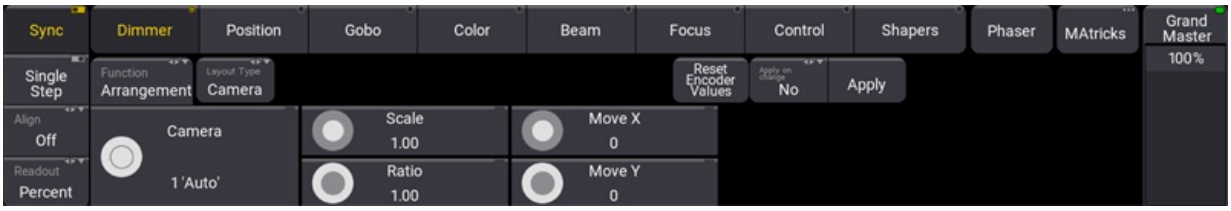
Layout Encoder Bar Function set to Arrangement

1. Create a layout and select it.
2. Tap Mode in the title bar to enable **Setup**; the layout encoder bar is displayed.
3. Tap Function in the layout encoder bar until it is set to **Arrangement**.
4. Select the layout elements (e.g., fixtures) you wish to position.
5. Tap and hold LayoutType, the Select Layout Type drop-down menu opens.
6. Select one of the four arrangement types: **Line**, **Grid**, **Circle**, or **Camera**.

For more information on **Line**, **Grid**, and **Circle** arrangements, see **Position fixtures in the 3D space**.

Camera Arrangement

Arrange your current selection similar to your camera view in the 3D window. Use the encoders to select a camera, set the scale and ratio factor, and move the selected elements.



Layout encoder bar function set to Arrangement

To do so:

1. Tap Function until it is set to **Arrangement**.
2. Tap Layout Type until it is set to **Camera**.

Set the following parameters using the dual encoders:

- **Camera**: Select the desired camera.
- **Scale**: Scales the whole selection in X and Y direction.
- **Ratio**: Defines the ratio of the selection.
- **Move X**: Move the whole selection in the X direction for a more convenient position.
- **Move Y**: Move the whole selection in the Y direction for a more convenient position.

Hint:
Press the first dual encoder inner ring or the first dual encoder key to open the camera calculator. Select a camera from the list or enter a camera number.

3. When you're satisfied with your layout setup, tap Apply to confirm.

Hint:
When Apply on change is active, Apply is disabled, and changes are made in real-time.

33. Plugins

A plugin is a piece of software that adds features to an existing program.

The grandMA3 supports such plugins, thus allowing customization.

For the plugins used in the grandMA3 software, the scripting language Lua is used.

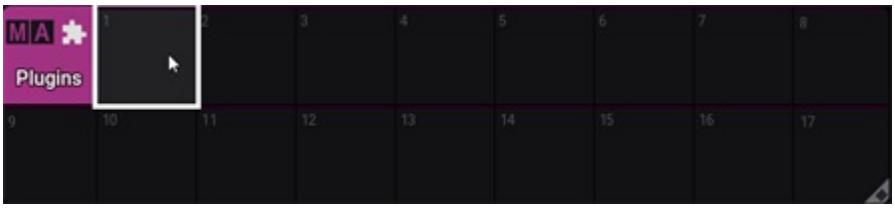
Important:
The creation and use of plugins can go deeper into the system as the “normal” usage of a console. Therefore the technical support team of MA Lighting may not be able to help you in all circumstances when using complex Lua plugins and Lua plugins might have to be rewritten when migrating show files to future grandMA3 software version.

Add a Plugin Pool Window

- To add a plugin pool window, follow the instructions under **Add windows**.

In the Add Window pop-up:

1. Tap Pools.
2. Tap Plugins.
   The plugin pool opens.



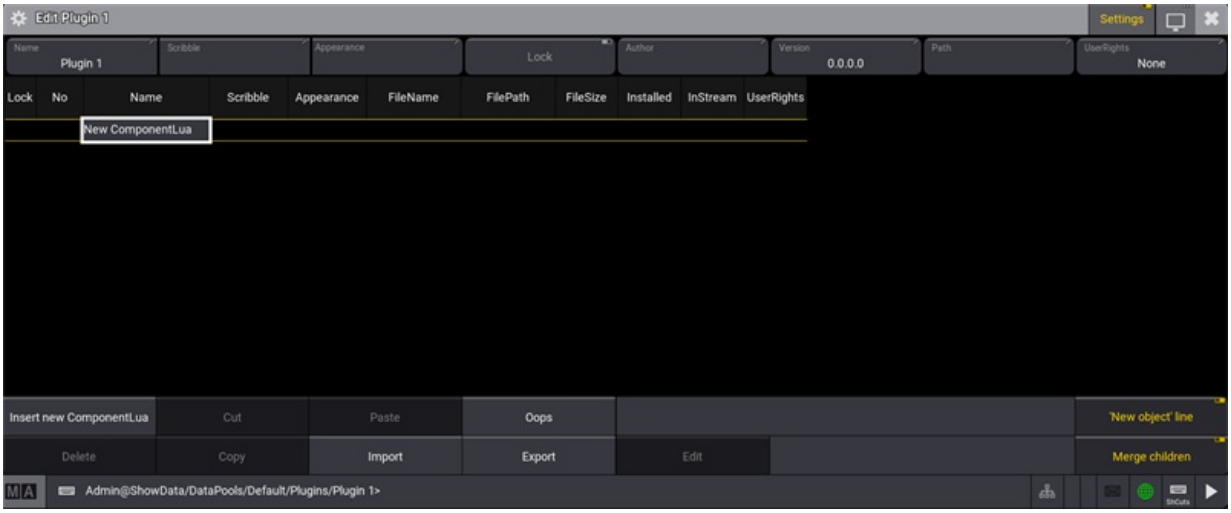
Plugin pool window

- To adjust the plugin pool window settings, follow the instructions under **Window settings**.

Add a Plugin to the Pool

- To add a plugin to the plugin pool, tap an empty pool object and use the **swipey menu** to select Edit.

The Edit Plugin window opens:

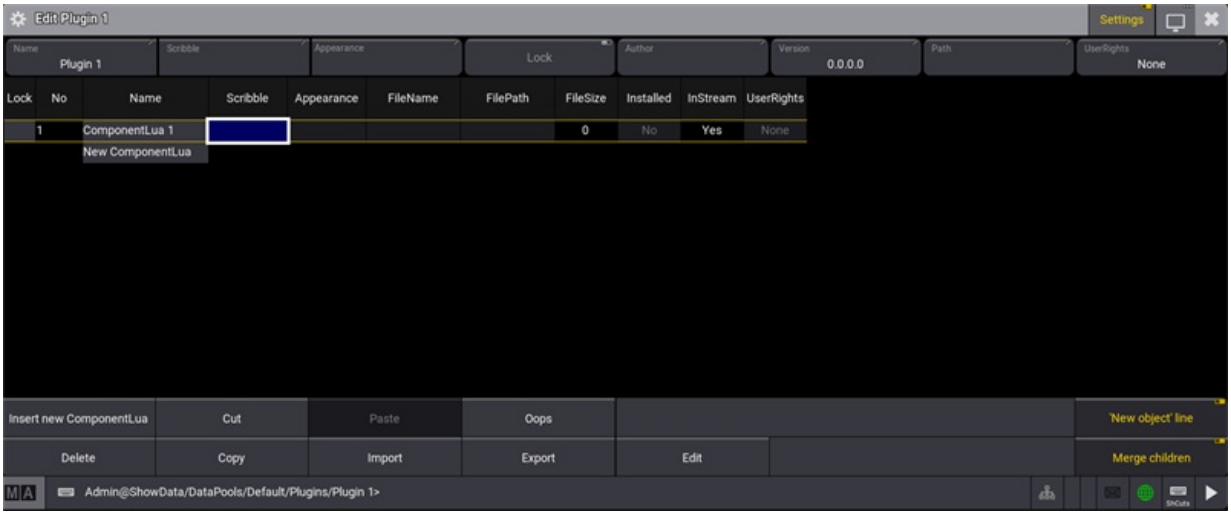


Edit Plugin window

Edit a Plugin

A Lua component is a piece of software code that can be inserted into the console, usually one file.

- To add a new Lua component, tap Insert new ComponentLua.



The new component is added.

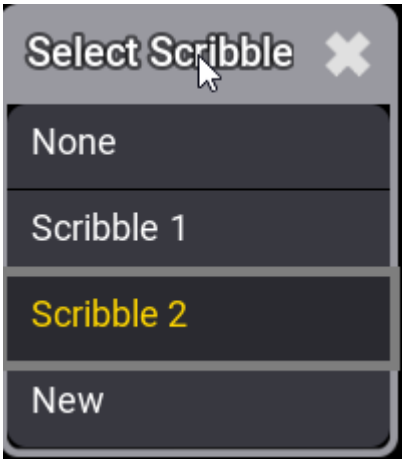
- To add Lua code to the component, tap Edit.

The Text input window opens:

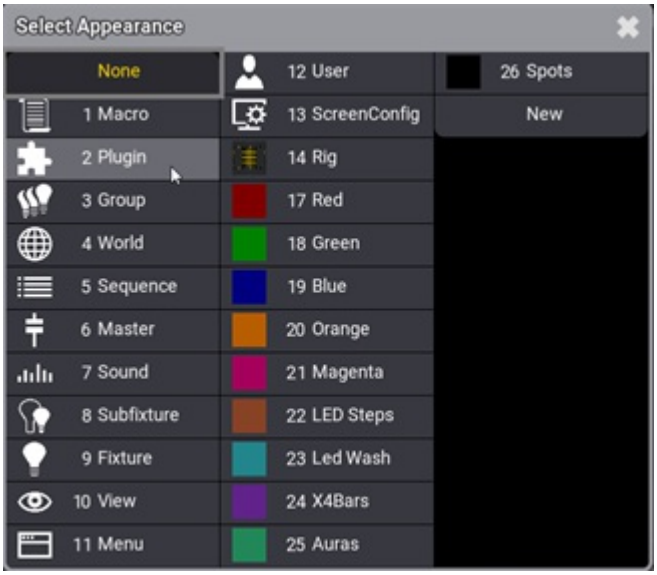
Lua Text input window

- Add text or paste the Lua code into the Text input window.
- To save the Lua plugin, tap Save.

- To add a Scribble, tap Scribble, and select the desired scribble from the drop-down list.



- To add an Appearance, tap Appearance, and select the desired appearance from the drop-down list.

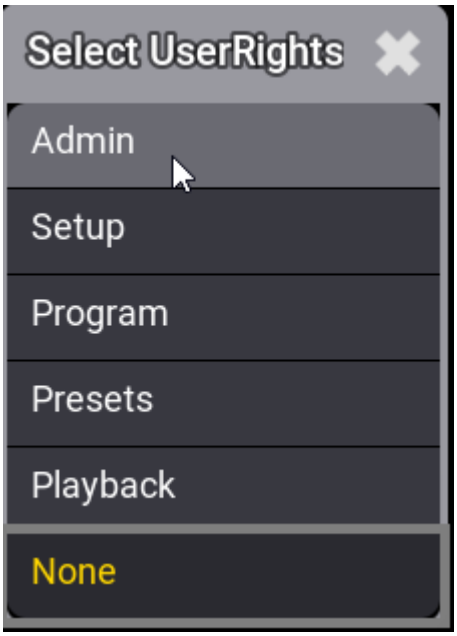


This file is saved in the show file, unless you export it.

- To enable or disable Installed, tap Installed to toggle between Yes and No.
  If Installed is set to Yes, the Lua components will be updated from the file archive they were imported from.
  To update any changes in these Lua components, the **command ReloadPlugins** must be executed.

- In Stream describes if the Lua code will be saved in the show file and be streamed in the session or not.
  In Stream depends on the setting in Installed and cannot be modified by the user.
  In Stream Yes means that the Lua code is saved in the show file and streamed in the session but stays as saved in the show file.
  In Stream No means that the Lua code is locally saved on the hard drive. The content of this Lua code can be updated by the **command ReloadPlugins**.

- To adjust the user rights, tap User Rights, and select the desired user right needed to execute the plugin from the drop-down list.



- To close the Edit Plugin window, tap .

Run a Plugin

- To run a Lua plugin, tap the desired Plugin object.

Import a Plugin

- To import a plugin, first create the plugin with your editor.

Example Plugin

XML file my_plugin.xml:

<?xml version="1.0" encoding="UTF-8"?>

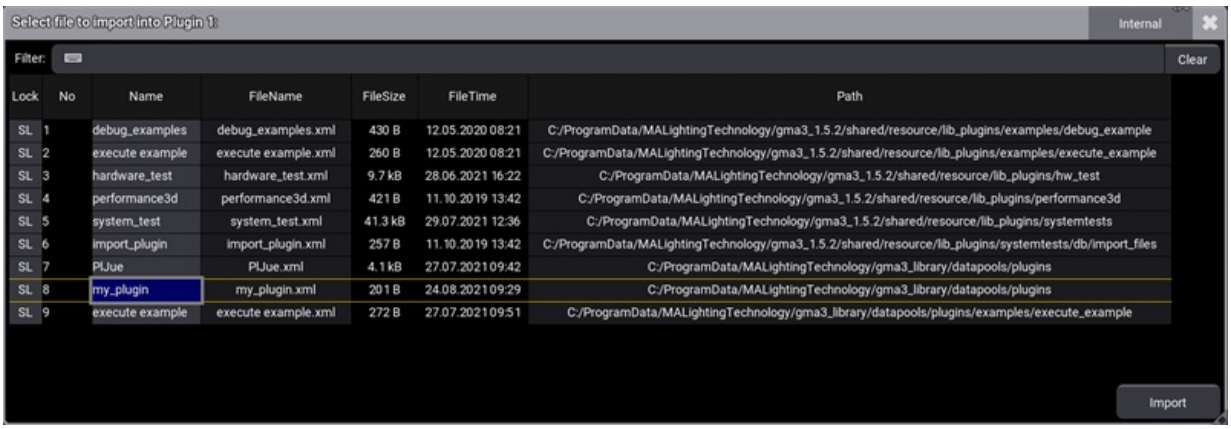
<GMA3><Plugin Path="**/my_plugin**" name="l_mstrongMyPluginstrong"><ComponentLua name="l_mstrongmypluginstr

Lua file my_plugin.lua:

```
local function main()
   Printf("Hello, MA user!")
end

return main
```

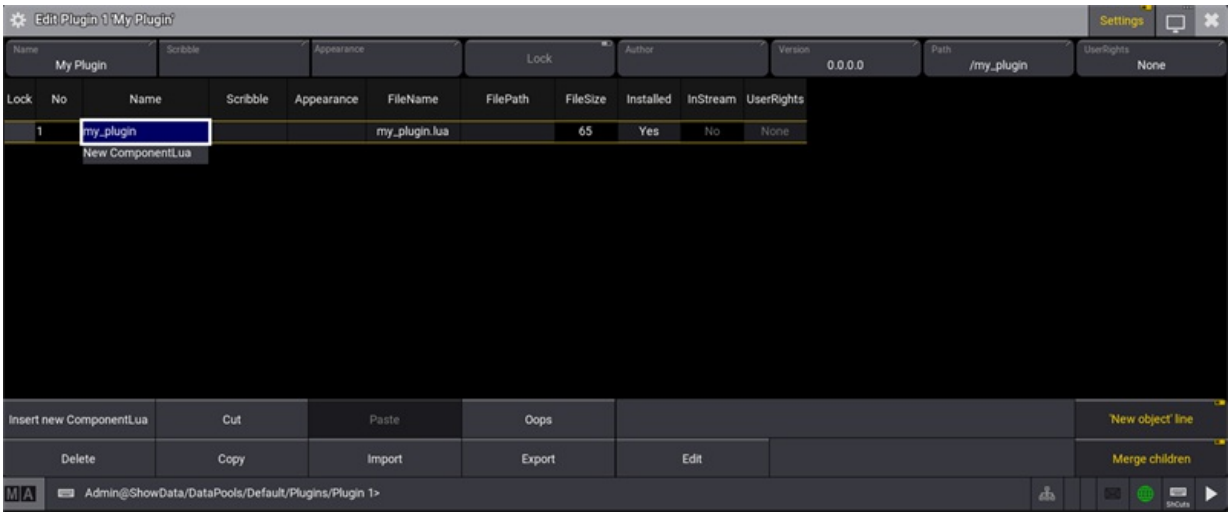
- Save the Lua file and the XML file in the folder C:\ProgramData\MALightingTechnology\gma3_library\datapools\plugins.
- Tap Import.

The Import selection menu opens:



- Select the plugin you want to import and tap Import.

The plugin is imported.



33.1. What is Lua

Lua is a scripting language designed to support general procedural programming. It offers support for object-oriented programming, functional programming, and data-driven programming. Lua is implemented as a library, written in *clean* C (common subset of ANSI C and C++).

To see the list of all grandMA3 Lua functions, read the **Lua Function List** topic.

To learn more about the individual grandMA3 Lua functions, read the subtopics below **Lua Functions**, for example, **Echo**.

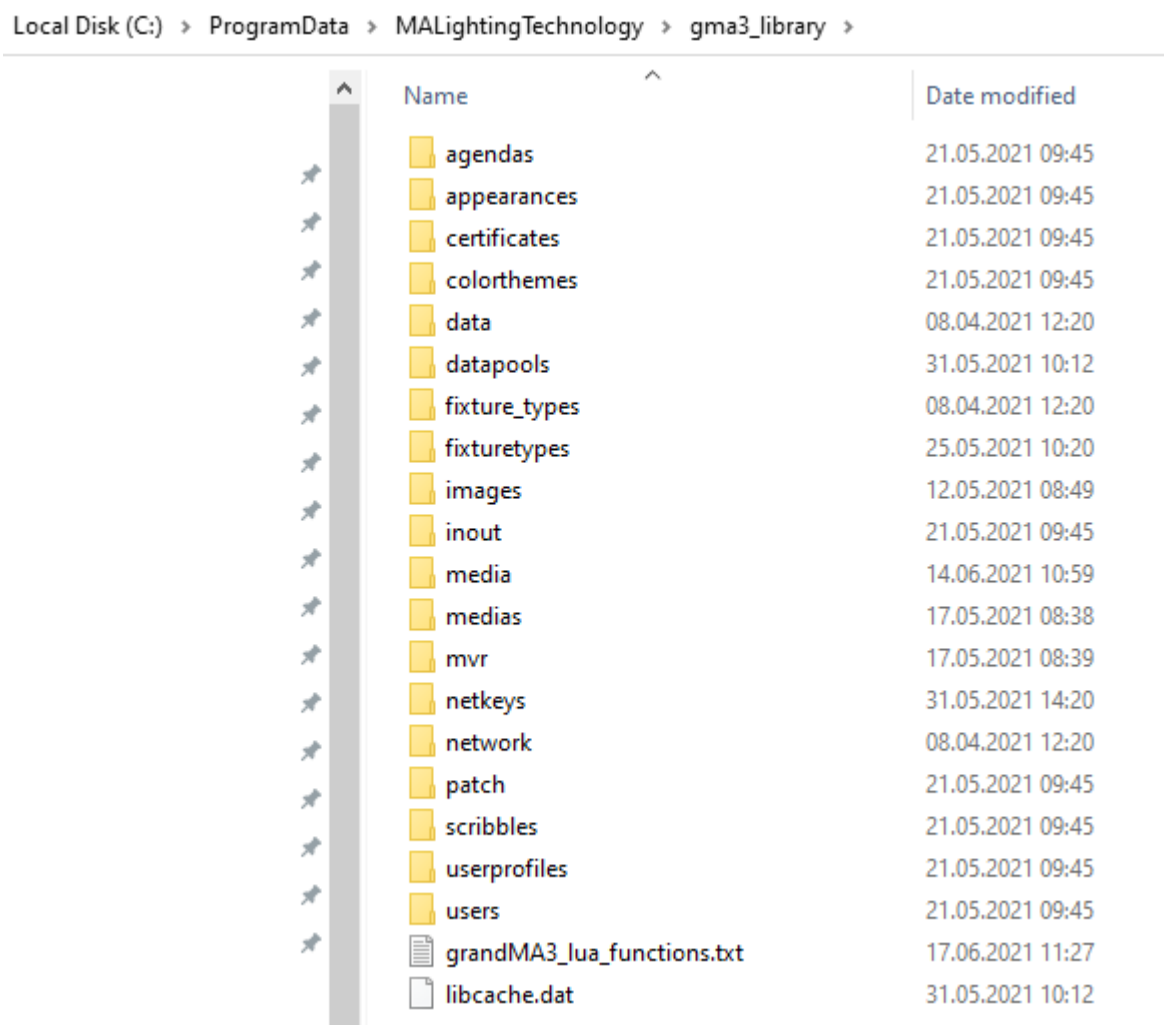
To check which Lua version your device is currently running, use the **Version keyword** in the command line.

For more information on scripting with Lua, see **www.lua.org**.

33.2. Lua Function List

The grandMA3 Lua API includes a variety of Lua functions.

The Lua functions included in the current software version can be exported into a list called "grandMA3_lua_functions.txt" in the gma3_library:



For more information read the **HelpLua keyword** topic.

List of grandMA3 Lua Functions

Object-Free API

Echo(string:format ...): nothing
ErrEcho(string:format ...): nothing
Printf(string:format ...): nothing
ErrPrintf(string:format ...): nothing

Cmd(string:format[,light_userdata:undo], ...): string:command execution result (Ok, Syntax Error, Illegal Command...)
CmdIndirect(string:cmd_to_execute[,light_userdata:undo[,light_userdata:target]]): nothing
CmdIndirectWait(string:cmd_to_execute[,light_userdata:undo[,light_userdata:target]]): nothing
HostOS(nothing): string::OsType
HostType(nothing): string::HostType
HostSubType(nothing): string::HostSubType
SerialNumber(nothing): string::SerialNumber
OverallDeviceCertificate(nothing): pCertificate::OverallCertificate
AddIPAddress(light_userdata: interface handle): nothing
DeleteIPAddress(light_userdata: interface handle): nothing
ReleaseType(nothing): string::release type
Version(nothing): string::version
BuildDetails(nothing): table:build details
GetShowFileStatus(nothing): Enums.ShowFileStatus
CmdObj(nothing): light_userdata:handle
Root(nothing): light_userdata:handle
Pult(nothing): light_userdata:handle
DefaultDisplayPositions(nothing): light_userdata:handle
Patch(nothing): light_userdata:handle
FixtureType(nothing): light_userdata:handle
ShowData(nothing): light_userdata:handle
ShowSettings(nothing): light_userdata:handle
DataPool(nothing): light_userdata:handle
MasterPool(nothing): light_userdata:handle
Programmer(nothing): light_userdata:handle
ProgrammerPart(nothing): light_userdata:handle
Selection(nothing): light_userdata:handle
CurrentUser(nothing): light_userdata:handle
CurrentProfile(nothing): light_userdata:handle
CurrentExecPage(nothing): light_userdata:handle to current ExecPage
SelectedSequence(nothing): light_userdata:handle
GetExecutor(integer:exec number): light_userdata:handle to executor, light_userdata: handle to page
LoadExecConfig(light_userdata: exec handle): nothing
SaveExecConfig(light_userdata: exec handle): nothing
SelectionFirst(nothing): int:first subfixture index,int:x, int:y, int:z
SelectionNext(int:current subfixture index): int: next subfixture index,int:x, int:y, int:z
SelectionCount(nothing): int: amount of selected subfixtures
SelectionComponentX(nothing): int: min,int:max,int:index,int:block,int:group
SelectionComponentY(nothing): int: min,int:max,int:index,int:block,int:group
SelectionComponentZ(nothing): int: min,int:max,int:index,int:block,int:group

GetSubfixtureCount(nothing): int:subfixture count
GetSubfixture(int:subfixture index): light userdata:reference to Subfixture object or nil
GetUIChannelCount(nothing): int:ui channel count
GetRTChannelCount(nothing): int:rt channel count
GetAttributeCount(nothing): int:attribute count
GetUIChannels(integer: subfixture index or light userdata: reference to Subfixture object[,boolean: as handles]): {array of UI channel indices or handles} or nil
GetRTChannels(integer: fixture index or light userdata: reference to Fixture object[,boolean: as handles]): {array of RT channel indices or handles} or nil
GetUIChannel(integer: channel UI index): {ChannelUI descriptor} or nil
GetRTChannel(integer: channel RT index): {ChannelRT descriptor} or nil
GetAttributeByUIChannel(integer: UI channel index): light userdata: reference to attribute or nil
GetAttributeIndex(string:attribute name): int:attribute index
GetUIChannelIndex(int:subfixture index,int:attribute index): int:ui channel index
GetChannelFunctionIndex(int:ui channel index,int:attribute index): int:channel function index
GetChannelFunction(int:ui channel index,int:attribute index): light_userdata: handle
GetSelectedAttribute(): light_userdata:attribute handle
GetTokenName(string:shortName): string:fullName
GetTokenNameByIndex(integer: token index): string:fullName
SetProgPhaser(number:uichannelindex,{[fade:<val>][delay:<val>][speed:<Hz>][repeat:<val>][phase:<val>][preset:lud]{[cfindex:<val>][preset:lud][trans:<val>][width:<val>][accel:<val>][decel:<val>][rel:<val>][abs:<val>]}*}): nothing
SetProgPhaserValue(number:uichannelindex,number:step,{[cfindex:<val>][preset:lud][trans:<val>][width:<val>][accel:<val>][decel:<val>][rel:<val>][abs:<val>]}): nothing
GetProgPhaser(number:uichannelindex,bool:phaser_only): {fade:<val>,delay:<val>,speed:<Hz>,repeat:<val>,phase:<val>,preset:lud,{cfindex:<val>,preset:lud,trans:<val>,width:<val>,accel:<val>,decel:<val>,rel:<val>,abs:<val>}*}
GetProgPhaserValue(number:uichannelindex,number:step): {cfindex:<val>,preset:lud,trans:<val>,width:<val>,accel:<val>,decel:<val>,rel:<val>,abs:<val>}
SetColor(string:colormodel(RGB,xyY,Lab,XYZ,HSB),double:tripel1,double:tripel2,double:tripel3,double: Brightness, double: Quality, bool: const_Brightness): int:flag
GetPresetData(light_userdata:handle(Preset)): array of phaser data
ColMeasureDeviceDarkCalibrate(): int:flag
ColMeasureDeviceDoMeasurement(): table:values
ObjectList(string:address): table of light_userdata:handle
FromAddr(string:address[, light_userdata:base handle]): light_userdata:handle
ToAddr(light_userdata:handle): string:address
IntToHandle(LuaInteger): light_userdata:handle
HandleToInt(light_userdata:handle): LuaInteger
StrToHandle(string: handle in H#... format): light_userdata:handle
HandleToStr(light_userdata:handle): string: handle in H#... format
IsObjectValid(light_userdata:handle): true or nil

Export(string:filename,table:export_data): bool:success
Import(string:filename): table:content
ExportJson(string:filename,table:export_data): bool:success
ExportCSV(string:filename,table:export_data): bool:success
HookObjectChange(function:callback,light_userdata:handle,light_userdata:plugin_handle[,light_userdata:target]): nothing
Unhook(function:callback): nothing
UnhookMultiple(function:callback(can be nil), light_userdata:handle to target(can be nil), light_userdata: handle to context (can be nil)): integer: amount of removed hooks
GetPath(string:path type | int as path type from enum 'PathType'[,bool: create]): string:path
GetPathType(light_userdata:target object[, integer: content type (Enums.PathContentType)]): path type name
GetPathOverrideFor(string:path type | int as path type from enum 'PathType', string:path[,bool: create]): string:overriden path
GetPathSeparator():
FileExists(string:path): boolean:result
SyncFS(nothing): nothing
DirList(string:path[,string:filter(s)]): array of {name:string, size:int, time:int}
StartProgress(string:name): light_userdate:handle
StopProgress(light_userdate:handle): nothing
SetProgressText(light_userdate:handle, string:text): nothing
SetProgressRange(light_userdate:handle, integer:start, integer:end): nothing
SetProgress(light_userdate:handle, integer:value): nothing
IncProgress(light_userdate:handle[, integer:delta]): nothing
GetPropertyColumnId(light_userdata:handle, string:propertyname): LuaInteger
Keyboard(integer: displayIndex, string:type('press','char','release'), (str:char(for type 'char') | str:keycode, bool:shift, bool:ctrl, bool:alt, bool:numlock)): nothing
Mouse(integer: displayIndex, string:type('press','move','release'), (str:button('Left', 'Middle', 'Right' for 'press', 'release') | integer:absX, integer:absY)):
Touch(integer: displayIndex, string:type('press','move','release'), integer:touchId, integer:absX, integer:absY):
Time(nothing): number:time
MouseObj(nothing): light_userdata:mouse object handle
TouchObj(nothing): light_userdata:touch object handle
KeyboardObj(nothing): light_userdata:keyboard object handle
Timer(function:name,number:dt,number:max_count,[function:cleanup],[light_userdata:context object]): nothing
FindBestDMXPatchAddr(light_userdata:patch,integer:starting address,integer:footprint): integer:absolute address
CheckDMXCollision(light_userdata:dmx mode,string:dmx address[,integer:count[,integer:breakIndex]]): boolean:true - no collision, false - collisions
CheckFIDCollision(integer:FID[,integer:count[,integer:type]]): boolean: true - no FID collisions, false - collisions
GetDMXValue(integer:address[,integer:universe, boolean: modePercent]): integer: dmx value
GetDMXUniverse(integer:universe[,boolean: modePercent]): table of integer: dmx values
SetLED(light_userdata:usb device object handle,table:led_values): nothing

GetButton(light_userdata:usb device object handle): table of bool:state
CreateUndo(string:undo text): light userdata: handle to undo
CloseUndo(light userdata: handle to undo): boolean: true if was closed, false - if it's still in use
DeskLocked(): boolean: true if desk is locked
RefreshLibrary(light userdata: Handle):
SelectionNotifyBegin(ligh_userdata:associated context):
SelectionNotifyObject(ligh_userdata:object to notify about):
SelectionNotifyEnd(ligh_userdata:associated context):
GlobalVars(): light userdata: global variables
UserVars(): light userdata: user variables
PluginVars([string: plugin name]): light userdata: plugin variables
AddonVars(string: addon name): light userdata: addon variables
SetVar(light userdata: variables, string:varname,value): bool:success
GetVar(light userdata: variables,string:varname): value
DelVar(light userdata: variables,string:varname): bool:success
CreateNeuralTraining(integer : InputLayerSize, integer: OutputLayerSize, table : TrainingValues, String : filename):
GetSample(string: type ('MEMORY', 'CPU', 'CPUTEMP', 'GPUTEMP', 'SYSTEMP', 'FANRPM')): number: current value in percent
GetFailedFrameCounter(enum: RealtimeSection): integer: current failed frame counter value for a given realtime section
SetRTTimeAllowance(enum: RealtimeSection, integer: new time budget in microseconds): nothing
RTSectionDeltaTime(enum: RealtimeSection): number: smoothed amount of microseconds
AddFixtures({mode:handle to DMX mode, amount:integer [,undo: string][,parent: handle][,insert_index:integer] [,idtype:string][,cid:string][,fid:string][,name:string][,layer:string][,class:string][,patch:{array 1..8: string address}]}): true on success or nil
TextInput([string:title[,string:value[,integer:x[,integer:y]]]]): string:value
PopupInput({title:str,caller:handle,items:table:{{'str'|'int'|'lua'|'handle', name, type-dependent}...},selectedValue:str,x:int,y:int,target:handle,render_options: {left_icon,number,right_icon},useTopLeft:bool,properties:{prop:value}}): string:value
Confirm([string:title[,string:message[,integer:displayIndex]]]): boolean:result
GetDisplayByIndex(integer:display_index): light_userdata:display_handle
GetUIObjectAtPosition(integer:display_index, {x,y}:position): light_userdata:handle to UI object or nil
WaitObjectDelete(light_userdata:handle to UIObject[, number:seconds to wait]): boolean:true on success, nil on timeout
GetFocus(nothing): light_userdata:display_handle
GetFocusDisplay(nothing): light_userdata:display_handle
GetDisplayCollect(nothing): light_userdata:handle to DisplayCollect
FindBestFocus([light_userdata:handle]): nothing
FindNextFocus([bool:backwards(false)[,int(Focus::Reason):reason(UserTabKey)]]): nothing
CloseAllOverlays(nothing): nothing
GetTopModal(nothing): light userdata: handle to top modal overlay

GetTopOverlay(integer:display_index): light userdata: handle to top overlay on the display
WaitModal([number:seconds to wait]): handle to modal overlay or nil on failure(timeout)
SetBlockInput(boolean:block): nothing
FindTexture(string:texture name): light userdata: handle to texture found
MessageBox({title:string,[backColor:string,][,timeout:number (ms)][,timeoutResultCancel:boolean] [,timeoutResultID:number][icon:string,][titleTextColor:string,][messageTextColor:string,] message:string[, display: (integer|lightuserdata)], commands:{array of {value:integer, name:string}}, inputs:{array of {name:string, value:string, blackFilter:string, whiteFilter:string, vkPlugin:string, maxTextLength:integer}}, states:{array of {name:string, state:boolean[,group:integer]}, selectors:{array of {name:string, selectedValue:integer, values:table[,type:integer 0-swipe, 1-radio]} }): {success:boolean, result:integer, inputs:{array of [name:string] = value:string}, states:{array of [name:string] = state:boolean}, selectors:{array of [name:string] = selected-value:integer}}

Object API

ToAddr(light_userdata:handle): string:address
Addr(light_userdata:handle,[light_userdata:base_handle[,bool:force parent-based address]]): text:numeric root address
AddrNative(light_userdata:handle[,light_userdata:base_handle[,bool:escape names]]): text:numeric root address
Index(light_userdata:handle): number:index
Parent(light_userdata:handle): light_userdata:parent_handle
Count(light_userdata:handle): number:child_count
MaxCount(light_userdata:handle): number:child_count
Compare(light_userdata:handle, light_userdata:handle): bool:isEqual, String:whatDiffers
Resize(light_userdata:handle,number size): nothing
Ptr(light_userdata:handle,number:index(1-based)): light_userdata:child_handle
CmdlinePtr(light_userdata:handle,number:index(1-based)): light_userdata:child_handle
Children(light_userdata:handle): table of light_userdata:child_handles
CurrentChild(light_userdata:handle): light_userdata:current child or nil
Create(light_userdata:handle,number:child_index(1-based)[,string:class[,light_userdata:undo]]): light_userdata:child_handle
Append(light_userdata:handle[,string:class[,light_userdata:undo]]): light_userdata:child_handle
Aquire(light_userdata:handle[,string:class[,light_userdata:undo]]): light_userdata:child_handle
Delete(light_userdata:handle,number:child_index(1-based)[,light_userdata:undo]): nothing
Insert(light_userdata:handle,number:child_index(1-based)[,string:class[,light_userdata:undo]]): light_userdata:child_handle
Remove(light_userdata:handle,number:child_index(1-based)[,light_userdata:undo]): nothing
Copy(light_userdata:dst_handle,light_userdata:src_handle[,light_userdata:undo]): nothing
HasParent(light_userdata:handle,handle:object to check): nothing
Changed(light_userdata:handle,string:change level enum name): nothing

IsEmpty(light_userdata:handle): boolean:returns true if objects is considered 'empty'
Set(light_userdata:handle,string:property_name,string:property_value): nothing
SetChildren(light_userdata:handle_of_parent,string:property_name,string:property_value[,bool:recursive (default: false)]): nothing
Get(light_userdata:handle,string:property_name[,enum{Roles}:role]): string:property_value (if 'role' provided - always string)
PropertyCount(light_userdata:handle): number:property_count
PropertyName(light_userdata:handle,number:property_index): string:property_name
PropertyType(light_userdata:handle,number:property_index): string:property_type
IsValid(light_userdata:handle): bool:result
IsClass(light_userdata:handle): string:class_name
GetClass(light_userdata:handle): string:class_name
GetChildClass(light_userdata:handle): string:class_name
GetAssignedObj(light_userdata:handle): light_userdata:handle
HasEditSettingUI(light_userdata:handle): bool:result
HasEditUI(light_userdata:handle): bool:result
GetUIEditor(light_userdata:handle): string:ui_editor_name
GetUISettings(light_userdata:handle): string:ui_settings_name
FindParent(light_userdata:search_start_handle,string search_class_name): light_userdata:found_handle
Find(light_userdata:search_start_handle,string search_name[,string search_class_name]): light_userdata:found_handle
FindRecursive(light_userdata:search_start_handle,string search_name[,string search_class_name]): light_userdata:found_handle
FindWild(light_userdata:search_start_handle,string search_name): light_userdata:found_handle
Import(light_userdata:handle,string:file_path,string:file_name): bool:success
Export(light_userdata:handle,string:file_path,string:file_name): bool:success
GetExportFileName(light_userdata:handle[,bool:camel_case_to_file_name]): string:filename
Load(light_userdata:handle,string:file_path,string:file_name): bool:success
Save(light_userdata:handle,string:file_path,string:file_name): bool:success
CommandCall(light_userdata:handle,light_userdata:dest_handle,bool:focusSearchAllowed(default:true)): nothing
CommandAt(light_userdata:handle): nothing
CommandDelete(light_userdata:handle): nothing
CommandStore(light_userdata:handle): nothing
CommandCreateDefaults(light_userdata:handle): nothing
SetFader(light_userdata:handle,{[double:value[0..100]], [bool:faderDisabled], [string:token(Fader*)]}): nothing
GetFader(light_userdata:handle, {[string:token(Fader*)], [integer:index]}): double:value[0..100]
GetFaderText(light_userdata:handle, {[string:token(Fader*)], [integer:index]}): string:text
HasActivePlayback(light_userdata:handle): boolean:result
InputSetTitle(light_userdata:handle,string:name_value): nothing
InputSetEditTitle(light_userdata:handle,string:name_value): nothing
InputSetAdditionalParameter(light_userdata:handle,string:parameter name,string:parameter value): nothing

InputRun(light_userdata:handle): nothing
InputCallFunction(light_userdata:handle,string:function name[,...parameters to function...]): <depends on function>
InputHasFunction(light_userdata:handle,string:function name): true or nil
AddListStringItem(light_userdata:handle,string:name,string:value[,{[left={...}][right={...}]}:appearance]): nothing
AddListPropertyItem(light_userdata:handle,string:name,string:value,light_userdata:target handle[,{[left={...}][right={...}]}:appearance]): nothing
AddListNumericItem(light_userdata:handle,string:name,number:value[,light_userdata:base handle[,{[left={...}][right={...}]}:appearance]]): nothing
AddListLuaItem(light_userdata:handle,string:name,string:value/function name,lua_function:callback reference[,<any lua type>:argument to pass to callback[,{[left={...}][right={...}]}:appearance]]): nothing
AddListObjectItem(light_userdata:handle,light_userdata:target object[,(string: explicit name[,{[left={...}][right={...}]}:appearance] | enum{Roles}: role [,:boolean: extended_name[,{[left={...}][right={...}]}:appearance])]): nothing
AddListStringItems(light_userdata:handle, table{item={[1]=name,[2]=value},...}): nothing
AddListPropertyItems(light_userdata:handle, table{item={[1]=name,[2]=property name,[3]=target handle},...}): nothing
AddListNumericItems(light_userdata:handle, table{item={[1]=name,[2]=integer:value},...}): nothing
AddListLuaItems(light_userdata:handle, table{item={[1]=name,[2]=value/function name,[3]=callback reference[,[4]=argument of any lua type to pass to callback]},...}): nothing
AddListChildren(light_userdata:handle, light_userdata:parent[,enum{Roles}:role]): nothing
AddListChildrenNames(light_userdata:handle, light_userdata:parent[,enum{Roles}:role]): nothing
AddListRecursiveNames(light_userdata:handle, light_userdata:parent[,enum{Roles}:role]): nothing
RemoveListItem(light_userdata:handle,string:name): nothing
ClearList(light_userdata:handle): nothing
SelectListItemByName(light_userdata:handle,string:name_value): nothing
SelectListItemByValue(light_userdata:handle,string:value): nothing
SelectListItemByIndex(light_userdata:handle,integer:index(1-based)): nothing
IsListItemEnabled(light_userdata:handle,integer:index): nothing
SetEnabledListItem(light_userdata:handle,integer:index[,bool:enable(default:true)]): nothing
IsListItemEmpty(light_userdata:handle,integer:index): nothing
SetEmptyListItem(light_userdata:handle,integer:index[,bool:empty(default:true)]): nothing
GetListItemValueStr(light_userdata:handle,integer:index): string:value
SetListItemValueStr(light_userdata:handle,integer:index,string:value): nothing
GetListItemValueI64(light_userdata:handle,integer:index): integer:value
GetListItemName(light_userdata:handle,integer:index): string:name
SetListItemName(light_userdata:handle,integer:index,string:name): nothing
GetListItemAppearance(light_userdata:handle,integer:index): {left={AppearanceData}, right={AppearanceData}}
SetListItemAppearance(light_userdata:handle,integer:index,{[left={...AppearanceData...}][right={...AppearanceData...}]}): nothing
GetListItemButton(light_userdata:handle,integer:index): light userdata:button or nil if not visible
GetListSelectedItemIndex(light_userdata:handle): integer:1-based index

GetListItemsCount(light_userdata:handle): integer:amount of items in the list
FindListItemByValueStr(light_userdata:handle,string:value): integer:1-based index
FindListItemByName(light_userdata:handle,string:value): integer:1-based index
GridGetBase(light_userdata:handle to UIGrid (or derived)): light_userdata:handle to GridBase
GridGetData(light_userdata:handle to UIGrid (or derived)): light_userdata:handle to GridData
GridGetSelection(light_userdata:handle to UIGrid (or derived)): light_userdata:handle to GridSelection
GridGetSelectedCells(light_userdata:handle to UIGrid (or derived)): array of {r,c,
r_UniqueId,r_GroupId,c_UniqueId,c_GroupId} cells in the selection
GridGetSettings(light_userdata:handle to UIGrid (or derived)): light_userdata:handle to GridSettings
GridGetDimensions(light_userdata:handle to UIGrid (or derived)): {r,c}
GridGetScrollOffset(light_userdata:handle to UIGrid (or derived)): {v = {index,offset}, h={index,offset}}
GridSetColumnSize(light_userdata:handle to UIGrid (or derived), integer: columnId, integer:size in pixels): nothing
GridGetScrollCell(light_userdata:handle to UIGrid (or derived)): {r,c}
GridGetCellData(light_userdata:handle to UIGrid (or derived), {r,c}:cell): {text, color={text,back}}
GridIsCellVisible(light_userdata:handle to UIGrid (or derived), {r,c}:cell): boolean
GridCellExists(light_userdata:handle to UIGrid (or derived), {r,c}:cell): boolean
GridIsCellReadOnly(light_userdata:handle to UIGrid (or derived), {r,c}:cell): boolean
GridScrollCellIntoView(light_userdata:handle to UIGrid (or derived), {r,c}:cell): nothing
GridGetCellDimensions(light_userdata:handle to UIGrid (or derived), {r,c}:cell): {x,y,w,h}
GridGetParentRowId(light_userdata:handle to UIGrid (or derived), integer: rowId): parent row id (integer) or nil (if
there's no parent)
GridsGetExpandHeaderCell(light_userdata:handle to UIGrid (or derived)): {r,c} or nil
GridsGetLevelButtonWidth(light_userdata:handle to UIGrid (or derived), {r,c}:cell): width in pixels or nil
GetOverlay(light_userdata:handle to UIObject): light_userdata:overlay_handle
GetDisplay(light_userdata:handle to UIObject): light_userdata:display_handle
GetDisplayIndex(light_userdata:handle to UIObject): integer:display_index
GetScreen(light_userdata:handle to UIObject): light_userdata:handle
GetUIChildrenCount(light_userdata:handle to UIObject): integer:count
ClearUIChildren(light_userdata:handle to UIObject): nothing
GetUIChild(light_userdata:handle to UIObject, integer:index(1-based)): light_userdata:handle to UIObject
UIChildren(light_userdata:handle to UIObject): array of references to children of passed UIObject
WaitInit(light_userdata:handle to UIObject[, number:seconds to wait[, bool: force to re-init, default - false]]):
boolean:true on success, nil on timeout or if object doesn't exist
WaitChildren(light_userdata:handle to UIObject, integer:expected amount of children, [, number:seconds to wait]):
boolean:true on success, nil on timeout or if object doesn't exist
HookDelete(light_userdata:handle to UIObject, function:callback to invoke on deletion[,any:argument to pass by]):
boolean:true on success, nil on failure
ShowModal(light_userdata:handle,callback:function): nothing
SetPositionHint(light_userdata:handle,integer:x,integer:y): nothing
ScrollIsNeeded(light_userdata:handle,integer:scroll type (see 'ScrollType' enum)): boolean:true if scroll of the
requested type is needed

ScrollDo(light_userdata:handle,integer:scroll type (see 'ScrollType' enum), integer:scroll entity (item or area, see 'ScrollParamEntity' enum), integer: value type (absolute or relative, see 'ScrollParamValueType' enum), number: value to scroll (items - 1-based), boolean: updateOpposite side): boolean:true scroll
ScrollGetInfo(light_userdata:handle,integer:scroll type (see 'ScrollType' enum)): {index(1-based), offset, visibleArea, totalArea, itemsCount, itemsOnPage} or nil
ScrollGetItemSize(light_userdata:handle,integer:scroll type (see 'ScrollType' enum), integer: 1-based item idx): integer:size of the item of nil
ScrollGetItemOffset(light_userdata:handle,integer:scroll type (see 'ScrollType' enum), integer: 1-based item idx): integer:offset of the item or nil
ScrollGetItemByOffset(light_userdata:handle,integer:scroll type (see 'ScrollType' enum), integer: offset): integer:1-based item index
UILGGetColumnWidth(light_userdata:handle to UILayoutGrid, idx:integer): size:integer
UILGGetRowHeight(light_userdata:handle to UILayoutGrid, idx:integer): size:integer
UILGGetColumnAbsXLeft(light_userdata:handle to UILayoutGrid, idx:integer): x:integer
UILGGetColumnAbsXRight(light_userdata:handle to UILayoutGrid, idx:integer): x:integer
UILGGetRowAbsYTop(light_userdata:handle to UILayoutGrid, idx:integer): y:integer
UILGGetRowAbsYBottom(light_userdata:handle to UILayoutGrid, idx:integer): y:integer
OverlaySetCloseCallback(light_userdata:handle to Overlay, callbackName:string[, ctx:anything]): nothing

33.3. Lua Functions

Individual Lua functions are described in the subtopics below, for example, **Printf**.

With the Lua keyword, you can execute Lua code via the command line.

For more information, read the **Lua keyword**.

An easy code to be integrated into a Lua plugin is to print "Hello World!" in the Command Line History:

```
local function main()
  Printf("Hello World!")
end

return main;
```

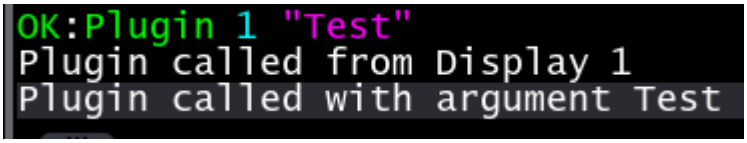
The local function main is the main function. When the plugin is executed, the returned function (e.g. "main") is executed.

Additionally, two more arguments can be added, displayHandle and args:

```
local function main(displayHandle, args)
  Printf("Plugin called from "..displayHandle:ToAddr())
  if args then
    Printf("Plugin called with argument "..args)
  end
end
return main
```

The two arguments have the following meaning:

1. displayHandle
   A reference to the display from which the plugin was called.
2. args
   Is a command line string added to the execution call, for example "Test" in the example below.



33.3.1. Object-Free API

The Lua function list is separated into two parts: Object-free API and **Object API**.

Object-free means that this includes the global API functions.

The Object-free APIs can be found in the **Lua function list**.

33.3.1.1. BuildDetails

Description

The BuildDetails Lua function returns build details of the station like the version and build date.

Syntax

BuildDetails(nothing): table:build details

Example

To print the build details of the station in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("CompileTime: "..BuildDetails().CompileTime)
   Printf("CompileDate: "..BuildDetails().CompileDate)
   Printf("Big Version: "..BuildDetails().BigVersion)
   Printf("Small Version: "..BuildDetails().SmallVersion)
   Printf("HostType: "..BuildDetails().HostType)
   Printf("HostSubType: "..BuildDetails().HostSubType)
end

return main
```

33.3.1.2. Cmd

Description

The CMD Lua function is used to execute a command within the grandMA3 command line, comparable with macros.

Syntax

Cmd(string:format[,light_userdata:undo], ...)

Returns: string:command execution result (Ok, Syntax Error, Illegal Command,...)

Example

To let the command line execute the command "Fixture Thru At 100", create a plugin with this code:

```
local function main()
   Cmd("Fixture Thru At 100")
end

return main
```

33.3.1.3. CmdIndirect

Description

The CmdIndirect Lua function is used to execute a command within the grandMA3 command line, but it does not block the Lua execution and is executed asynchronously. Also, the command is executed by the Main Task instead of the Lua Task.

Syntax

CmdIndirect(string:cmd_to_execute [,light_userdata:undo [,light_userdata:target]])

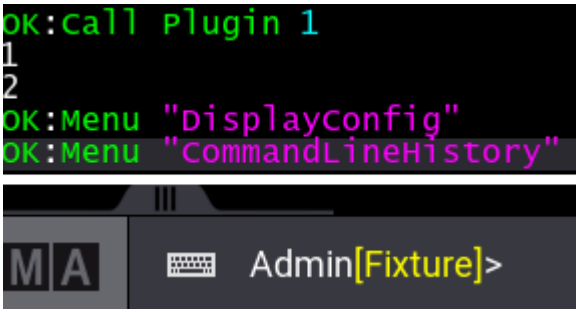
Returns: nothing

Example

To print "1" and "2" in the Command Line History and let the command line open the **Configure Display pop-up**, create a plugin with this code:

```
local function main()
   Printf("1")
   CmdIndirect("Menu DisplayConfig")
   Printf("2")
end

return main
```

The Command Line History shows:



33.3.1.4. CmdIndirectWait

Description

The CmdIndirectWait Lua function is used to execute a command within the grandMA3 command line. It does not block the Lua execution and is executed synchronously, but waits for the command to be executed. Also, the command is executed by the Main Task instead of the Lua Task.

Syntax

CmdIndirectWait(string:cmd_to_execute [,light_userdata:undo [,light_userdata:target]])

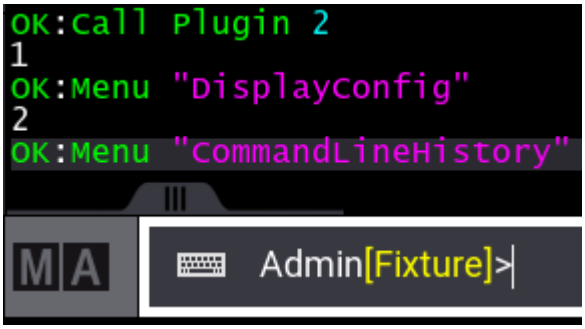
Returns: nothing

Example

To print "1" and "2" in the Command Line History and let the command line open the **Configure Display pop-up**, create a plugin with this code:

```
local function main()
   Printf("1")
   CmdIndirectWait("Menu DisplayConfig")
   Printf("2")
end

return main
```

The Command Line History shows:



33.3.1.5. CmdObj

Description

The CmdObj Lua function is used to return information about the cmdline object as light userdata.

Syntax

CmdObj(nothing): light_userdata:handle

Returns: nothing

Example

To print some values of the cmdline object in the Command Line History, create a plugin with this code:

```
local function main()
   local cmd = CmdObj()
   Printf("Current text in the command line: " ..cmd.cmdtext)
   Printf("Current cmd edit object: " ..tostring(cmd.editobject and cmd.editobject:ToAddr()))
   Printf("Current cmd destination: " ..tostring(cmd.destination and cmd.destination:ToAddr()))
   Printf("Current user of the command line: " ..tostring(cmd.user and cmd.user:ToAddr()))
   Printf("Current profile of the command line: " ..tostring(cmd.profile and cmd.profile:ToAddr()))
   Printf("Current DMX readout: " ..cmd.dmxreadout)
   Printf("Current amount steps: " ..cmd.maxstep)
   Printf("Current selected object: " ..tostring(cmd:GetSelectedObject() and cmd:GetSelectedObject():ToAddr()))
end

return main
```

33.3.1.6. Confirm

Description

The Confirm Lua function provides a confirmation for a query etc.

Syntax

Confirm([string:title[,string:message[,integer:displayIndex]]]): boolean:result

Example

To add a confirmation, create a plugin with this code:

```
local function main()
   if Confirm("Confirm me", "Tap OK") then
      Printf("OK")
   else
      Printf("Cancel.")
   end
end

return main
```

33.3.1.7. DataPool

Description

The DataPool Lua function provides a reference to the currently selected DataPool and is used to read or edit objects within the data pool etc.

Syntax

DataPool(nothing)

Returns: light_userdata:handle

Example

To output the name of the selected data pool and the name of sequence 1 within the selected data pool in the Command Line History, create a plugin with this code:

```
local function main()
   local nameofDatapool = DataPool().Name

   local nameofSeq1 = DataPool().Sequences[1].Name

   Printf("Name of selected data pool: " .. nameofDatapool)

   Printf("Name of sequence 1: " .. nameofSeq1)
end

return main
```

33.3.1.9. Echo

Description

The Echo Lua function is used to print text on the System Monitor.

Syntax

Echo(string:format ...)

Returns: nothing

Example

To print "Hello World!" on the System Monitor, create a plugin with this code:

```
local function main()
   Echo("Hello World!")
end

return main
```

33.3.1.10. ErrPrintf

Description

The ErrPrintf Lua function is used to print a red error message in the Command Line History and System Monitor.

Syntax

ErrPrintf(string:format ...)

Returns: nothing

Example

To print "This is a red error message!" in the Command Line History and System Monitor, create a plugin with this code:

```
local function main()
   ErrPrintf("This is a red error message!")
end

return main
```

33.3.1.11. ErrEcho

Description

The ErrEcho Lua function is used to print a red error message on the system monitor.

Syntax

ErrEcho(string:format ...)

Returns: nothing

Example

To print "This is a red error message!" on the system monitor, create a plugin with this code:

```
local function main()
   ErrEcho("This is a red error message!")
end

return main
```

33.3.1.12. Export

Description

The Export Lua function is used to export a Lua table in XML format.

This Lua function correlates with the **Import Lua function**.

Syntax

Export(string:filename,table:export_data): bool:success

Returns: nothing

Example

To export the table "mySettings.xml", create a plugin with this code:

```
local function main()
   local mySettings = {
      count = 10,
      indices = {1, 2, 3, 4},
      names = "SomeString"
   }
   local exportPath = GetPath(Enums.PathType.Library) .. "/mySettings.xml"
   Export(exportPath, mySettings)

   Printf("Table exported to " .. exportPath)
end

return main
```

The command line history shows:

The exported table looks like this:

```
<?xml version="1.0" encoding="utf-8"?>
<LUA_EXPORT table="true">
    <count integer="10"/>
    <names string="SomeString"/>
    <indices table="true">
        <table_index i="1" integer="1"/>
        <table_index i="2" integer="2"/>
        <table_index i="3" integer="3"/>
        <table_index i="4" integer="4"/>
    </indices>
</LUA_EXPORT>
```

33.3.1.13. GetPath

Description

The GetPath Lua function is used to deliver the path of a grandMA3 folder.

The searched folder is given by the first argument, e.g. by a value from the 'global "Enums.PathType" table', or by text.

Syntax

GetPath(string:path type | int as path type from enum 'PathType'[,bool: create]): string:path

Example

To print the paths of the user macro folder and the shows folder on the system monitor, create a plugin with this code:

```
local function main()
   Printf("Path of user macros is " .. GetPath(Enums.PathType.UserMacros))

   Printf("Path of showfiles is " .. GetPath("shows"))
end

return main
```

33.3.1.14. GetShowFileStatus

Description

The GetShowFileStatus Lua function returns the current device's show file status (e. g. "NoShow", "ShowLoaded", "ShowDownloaded", "DataNegotiationActive").

Syntax

GetShowFileStatus(nothing): Enums.ShowFileStatus

Example

To print the current device's show file status in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("ShowfileStatus: "..GetShowFileStatus())
end

return main
```

33.3.1.15. GetSubfixture

Description

The GetSubfixture Lua function is used to return the handle of the subfixture that is specified by its index.

Syntax

GetSubfixture(int:subfixture index): light userdata:reference to Subfixture object or nil

Returns: nothing

Example

To print the name of the first selected fixture in the Command Line History, create a plugin with this code:

```
local function main()
  local firstSubfixtureIndex = SelectionFirst()

  -- Cancel the plugin if no fixture is selected
  assert(firstSubfixtureIndex,"No fixture selected! Please select a fixture first.")

  local firstSubfixtureHandle = GetSubfixture(firstSubfixtureIndex)
  Printf("Name of first selected fixture: "..firstSubfixtureHandle.name)
end

return main
```

33.3.1.16. GetSubfixtureCount

Description

The GetSubfixtureCount Lua function returns the total number of subfixtures that are patched within the show file.

Syntax

GetSubfixtureCount(nothing): int:subfixture count

Returns: nothing

Example

Hint:
Before executing the plugin below, select one or more fixtures first.

To print the number of subfixtures in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("Total number of subfixtures: " .. GetSubfixtureCount())
end

return main
```

33.3.1.17. HookObjectChange

Description

The HookObjectChange Lua function is used to automatically call a function when an grandMA3 object changes.

Syntax

HookObjectChange(function:callback, light_userdata:handle, light_userdata:plugin_handle[, light_userdata:target]): nothing

Returns: nothing

Example

To call a function every time the content of the sequence pool changes, create a plugin with this code:

```
local pluginName     = select(1,...);
local componentName  = select(2,...);
local signalTable    = select(3,...);
local my_handle      = select(4,...);

local function main()
   local hookObject = DataPool().Sequences;
   local pluginHandle = my_handle:Parent();
   local callbackFunction = function(obj)
      Printf(tostring(obj.name).." changed!")
   end
   HookObjectChange(callbackFunction, hookObject, pluginHandle)

end

return main
```

33.3.1.18. HostOS

Description

The HostOS Lua function returns the operating system of the device where the plugin is executed (e.g. "Windows", "Linux", "Mac").

Syntax

HostOS(nothing): string::OsType

Example

To print the operating system of the device in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("HostOs = "..HostOS())
end

return main
```

33.3.1.19. HostSubType

Description

The HostSubType Lua function returns the host sub type of the station where the plugin is executed (e.g., "FullSize", "Light", "RPU", "onPCRackUnit", "Undefined").

Syntax

HostSubType(nothing): string::HostSubType

Example

To print the host type of the station in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("HostSubType = "..HostSubType())
end

return main
```

33.3.1.20. HostType

Description

The HostType Lua function returns the host type of the device where the plugin is executed (e.g. "Console", "onPC", "PU").

Syntax

HostType(nothing): string::HostType

Example

To print the host type of the device in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("HostType = "..HostType())
end

return main
```

33.3.1.21. Import

Description

The Import Lua function is used to import a Lua table in XML format.

This Lua function correlates with the **Export Lua function**.

Syntax

Import(string:filename): table:content

Returns: nothing

Example

To import the table "mySettings.xml", create a plugin with this code:

```
local function main()
   local mySettings = {
      count = 10,
      indices = {1, 2, 3, 4},
      names = "SomeString"
   }

   local importPath = GetPath(Enums.PathType.Library) .. "/mySettings.xml"
   local importedTable = Import(importPath)
   Printf("Imported table from " .. importPath)
   Printf("Imported count = " .. importedTable.count)
   Printf("Imported names = " .. importedTable.names)
   for index in pairs(importedTable.indices) do
      Printf("Index = " .. index)
   end
end

return main
```

The command line history shows:

33.3.1.22. MessageBox

Description

The MessageBox Lua function is used to create message boxes.

Syntax

MessageBox({title:string,[backColor:string,][,timeout:number (ms)][,timeoutResultCancel:boolean] [,timeoutResultID:number][icon:string,][titleTextColor:string,][messageTextColor:string,] message:string[, display: (integer|lightuserdata)], commands:{array of {value:integer, name:string}}, inputs:{array of {name:string, value:string, blackFilter:string, whiteFilter:string, vkPlugin:string, maxTextLength:integer}}, states:{array of {name:string, state:boolean[,group:integer]}, selectors:{array of {name:string, selectedValue:integer, values:table[,type:integer 0-swipe, 1-radio]} }): {success:boolean, result:integer, inputs:{array of [name:string] = value:string}, states:{array of [name:string] = state:boolean}, selectors:{array of [name:string] = selected-value:integer}}

Examples

To create a message box "Messagebox example", create a plugin with this code:

```
local function main(displayHandle)
  local res =
    MessageBox(
    {
      title = "Messagebox example",
      message = "Edit the following values",
      display = displayHandle.index,
      inputs = {{value = 1, name = "A"}, {value = 2, name = "B"}}
    }
  )

  Printf("A changed to " .. res.inputs["A"])

  Printf("B changed to " .. res.inputs["B"])
end

return main
```

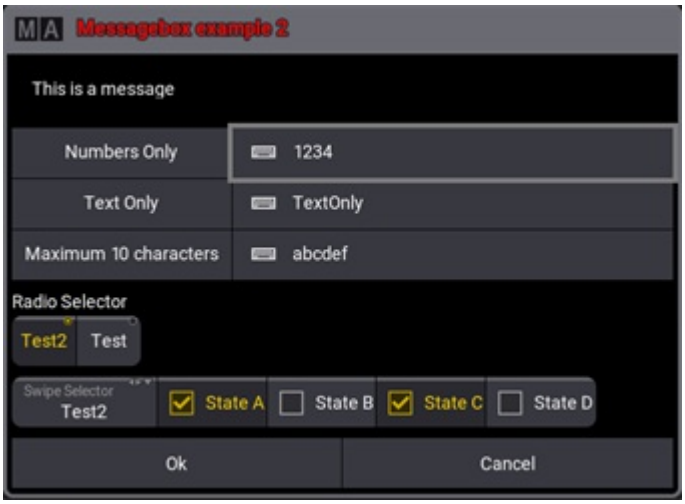
To create a different message box "Messagebox example 2" with a maximum of options, create a plugin with this code:

```
local function main()
  -- create inputs:
  local states = {
```

```
    {name = "State A", state = true, group = 1},
    {name = "State B", state = false, group = 1},
    {name = "State C", state = true, group = 2},
    {name = "State D", state = false, group = 2}
}
local inputs = {
    {name = "Numbers Only", value = "1234", whiteFilter = "0123456789"},
    {name = "Text Only", value = "TextOnly", blackFilter = "0123456789"},
    {name = "Maximum 10 characters", value = "abcdef", maxTextLength = 10}
}
local selectors = {
    { name="l_mSwipeSelector", selectedValue=2, values={["Test"]=1,["Test2"]=2}, type=0},
    { name="l_mRadioSelector", selectedValue=2, values={["Test"]=1,["Test2"]=2}, type=1}
}

-- open messagebox:
local resultTable =
    MessageBox(
    {
        title = "Messagebox example 2",
        message = "This is a message",
        message_align_h = Enums.AlignmentH.Left,
        message_align_v = Enums.AlignmentV.Top,
        commands = {{value = 1, name = "Ok"}, {value = 0, name = "Cancel"}},
        states = states,
        inputs = inputs,
        selectors = selectors,
        backColor = "Global.Default",
        -- timeout = 10000, --[[milliseconds]]
        -- timeoutResultCancel = false,
        icon = "logo_small",
        titleTextColor = "Global.AlertText",
        messageTextColor = "Global.Text"
    }
)

-- print results:
Printf("Success = "..tostring(resultTable.success))
Printf("Result = "..resultTable.result)
for k,v in pairs(resultTable.inputs) do
    Printf("Input '%s' = '%s'",k,v)
end
for k,v in pairs(resultTable.states) do
    Printf("State '%s' = '%s'",k,tostring(v))
end
for k,v in pairs(resultTable.selectors) do
```

```
      Printf("Selector '%s' = '%d'",k,v)
    end
end

return main
```

After this Lua plugin is executed, the following pop-up appears:



33.3.1.23. ObjectList

Description

The ObjectList Lua function generates a list of objects based on a text. The text has the same syntax as the interaction with the command line (e.g. "Fixture 1 Thru 10").

Syntax

ObjectList(string:address)

Returns: table of light_userdata:handle

Example

To create a list of fixtures 1 to 10 and output the name of the first fixture of this list, create a plugin with this code:

```
local function main()
   local myObjects = ObjectList("Fixture 1 Thru 10")

   Printf("Name of Fixture 1 = " .. myObjects[1].name)
end

return main
```

33.3.1.25. PopupInput

Description

The PopupInput Lua function creates a popup input field in the UI, where the user can select an item out of a list of different variables.

Syntax

PopupInput({title:str,caller:handle,items:table:{{'str'|'int'|'lua'|'handle', name, type-dependent}...},selectedValue:str,x:int,y:int,target:handle,render_options:{left_icon,number,right_icon},useTopLeft:bool,properties:{prop:value}}): string:value

Example

To open a popup input and print the selected result in the command line history, create a plugin with this code:

```
local function main(displayHandle)
   local descTable = {
      title = "Demo",
      caller = displayHandle,
      items = {"Select","Some","Value","Please"},
      selectedValue = "Some",
      add_args = {FilterSupport="Yes"},
   }
   local a,b = PopupInput(descTable)
   Printf("a = %s",tostring(a))
   Printf("b = %s",tostring(b))
end

return main
```

33.3.1.26. Printf

Description

The Printf Lua function is used to print text in the Command Line History.

Syntax

Printf(string:format ...)

Returns: nothing

Example

To print "Hello World!" in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("Hello World!")
end

return main
```

33.3.1.27. ProgressBar

Description

The ProgressBar Lua function includes a variety of Lua functions used to show the progress of a certain function.

Example

To execute the progress bar "myProgress", create a plugin with this code:

```
local function main()
   -- create the progress bar:
   local progHandle = StartProgress("myProgress")
   -- set start index and end index of the progress bar:
   local startIdx, endIdx = 1, 3

   -- define the range of the progress bar:
   SetProgressRange(progHandle, startIdx, endIdx)
   for i = startIdx, endIdx do
      -- set the progress state of the progress bar:
      SetProgress(progHandle, i)
      coroutine.yield(1)
   end

   -- remove the progress bar:
   StopProgress(progHandle)

end

return main
```

33.3.1.28. SelectedSequence

Description

The SelectedSequence Lua function is used to address the selected sequence.

Syntax

SelectedSequence(nothing): light_userdata:handle

Returns: nothing

Example

To return the name of the selected sequence in the Command Line History, create a plugin with this code:

```
local function main()

   local selectedSeq = SelectedSequence()

   Printf("Selected sequence: "..selectedSeq.name)
end

return main
```

33.3.1.29. SelectionCount

Description

The SelectionCount Lua function returns the total number of currently selected fixtures.

Syntax

SelectionCount(nothing): int: amount of selected subfixtures

Returns: nothing

Example

Hint:
Before executing the plugin below, select one or more fixtures first.

To print the number of the currently selected fixtures in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("Number of selected fixtures: "..SelectionCount())
end

return main
```

33.3.1.30. SelectionFirst

Description

The SelectionFirst Lua function is used to return the subfixture index of the first selected fixture.

Syntax

SelectionFirst(nothing): int:first subfixture index,int:x, int:y, int:z

Returns: nothing

Example

To print the subfixture index of the first selected fixture in the Command Line History, create a plugin with this code:

```
local function main()
   local subfixtureIndex = SelectionFirst();
   Printf("subfixture selected with index: "..subfixtureIndex)
end

return main
```

33.3.1.31. SelectionNext

Description

The SelectionNext Lua function is used to return the next subfixture index that follows after the subfixtureindex given with the argument.

Syntax

SelectionNext(int:current subfixture index): int: next subfixture index,int:x, int:y, int:z

Returns: nothing

Example

To print the subfixture indices of all fixtures of the current selection in the Command Line History, create a plugin with this code:

```
local function main()
   local subfixtureIndex = SelectionFirst();
   repeat
      Printf("subfixture selected with index: "..subfixtureIndex)
      subfixtureIndex = SelectionNext(subfixtureIndex)
   until not subfixtureIndex;
end

return main
```

33.3.1.32. SerialNumber

Description

The SerialNumber Lua function returns the serial number of the station where the plugin is executed.

Syntax

SerialNumber(nothing): string::SerialNumber

Example

To print the serial number of the station in the Command Line History, create a plugin with this code:

```
local function main()
   Printf("SerialNumber = "..SerialNumber())
end

return main
```

33.3.1.33. ShowData

Description

The ShowData Lua function returns the handle of the ShowData object using the **Dump Lua function**.

Syntax

ShowData(nothing): light_userdata:handle

Example

To print the information of the ShowData object in the Command Line History, create a plugin with this code:

```
local function main()
   ShowData():Dump()
end

return main
```

33.3.1.34. ShowSettings

Description

The ShowSettings Lua function returns the handle of the ShowSettings object using the **Dump Lua function**.

Syntax

ShowSettings(nothing): light_userdata:handle

Example

To print the information of the ShowSettings object in the Command Line History, create a plugin with this code:

```
local function main()
   ShowSettings():Dump()
end

return main
```

33.3.1.35. TextInput

Description

The TextInput Lua function opens a text input field in the UI, the entered text will be returned as a lua variable.

Syntax

TextInput([string:title [, string:value [, integer:x [, integer:y]]]]): string:value

Example

To open a text input and print the entered value in the Command Line History, create a plugin with this code:

```
local function main()
   local input = TextInput("This is the title","The message to be displayed.")
   Printf("You entered this message: %s",tostring(input))
end

return main
```

33.3.1.36. Timer

Description

The Timer Lua function is used to call a Lua function with a time delay.

Syntax

Timer(function:name, number:delaytime, number:max_count, [function:cleanup], [light_userdata:context object]): nothing

Returns: nothing

Example

To call a Lua function three times with a delay of one second between each function call, create a plugin with this code:

```
local function main()
   local function timedFunction()
      Printf("Do something")
   end
   local waitSeconds = 1
   local iterations = 3
   Timer(timedFunction, waitSeconds, iterations);
end

return main
```

33.3.1.37. UserVariables

Description

The UserVariables Lua functions are functions to interact with user variables.

Furthermore, the following variables are available:

GlobalVars
PluginVars

Functions

GetVar: to get the value of a user variable by its key ("myUserVar").

SetVar: to store or change a user variable.

DelVar: to delete a user variable.

UserVars: returns the reference object to user variables.

Example

To set, get and delete user variables, create a plugin with this code:

```
local function main()
  local userVars = UserVars()

  -- store user variable:
  SetVar(userVars, "myUserVar", 666)

  -- get user variable:
  local myVariable = GetVar(userVars, "myUserVar")
  Printf("myUserVar = " .. myVariable)

  -- delete user variable:
  DelVar(userVars, "myUserVar")
end

return main
```

33.3.1.38. Version

Description

The Version Lua function returns the software version.

Syntax

Version(nothing): string::version

Returns: nothing

Example

To print the software version in the Command Line History, create a plugin with this code:

```
local function main()
   Printf(Version())
end

return main
```

33.3.2. Object API

The Lua function list is separated into two parts: **Object-free API** and Object API.

Object API means that these Lua functions are related to an object.

The Object APIs can be found in the **Lua function list** below the heading Object API.

33.3.2.1. Children

Description

The Children Lua function creates a list of children of an object.

Syntax

CurrentChild(light_userdata:handle): light_userdata:current child or nil

Example

To return the children of the first sequence of the selected data pool, create a plugin with this code:

```
local function main()
   local mySequence = DataPool().Sequences[1]
   local cues = mySequence:Children()

   for i = 1, #cues do
      Printf("Sequence 1 Child " .. i .. " = " .. cues[i].name)
   end
end

return main
```

33.3.2.2. Dump

Description

The Dump Lua function returns some information about the object, e.g., the name, class or path of the object, its properties, and children. In the example, the user can input object addresses, e.g., Sequence 1. Then, the Dump() method is called for this object and the information about this object is printed in the Command Line History.

Syntax

Dump(light_userdata:handle): string:information

Example

To print the information of Sequence 1 in the Command Line History, create a plugin with this code:

```
local function main()
   local searchedObject = TextInput("Object Address", "Sequence 1")
   local object = ObjectList(searchedObject)[1]
   object:Dump()
end

return main
```

33.3.2.3. Export

Description

The Export Lua function is used to export an object into an XML file.

Syntax

Export(light_userdata:handle,string:file_path,string:file_name): bool:success

Returns: nothing

Example

To export the current sequence into the file "mySelectedSequence.xml", create a plugin with this code:

```
local function main()
   local selectedSequence = SelectedSequence()
   local path = GetPath(Enums.PathType.UserSequences)
   selectedSequence:Export(path,"mySelectedSequence.xml")
end

return main;
```

33.3.2.4. HasActivePlayback

Description

The HasActivePlayback Lua function is used to return the information if an object has a currently active playback, e.g. if a sequence has a running cue.

Syntax

HasActivePlayback(light_userdata:handle): boolean:result

Returns: Result

Example

To return the information if the selected sequence has an active playback, create a plugin with this code:

```
local function main()
   local selectedSequence = SelectedSequence()
   if selectedSequence:HasActivePlayback() then
      Printf("Sequence '" ..selectedSequence.name.. "' has active playback.")
   else
      Printf("Sequence '" ..selectedSequence.name.. "' has NO active playback.")
   end
end

return main;
```

33.3.2.5. Import

Description

The Import Lua function is used to import an object in XML format.

Syntax

Import(light_userdata:handle,string:file_path,string:file_name): bool:success

Returns: nothing

Example

To import the sequence in the file "mySelectedSequence.xml" to the current sequence, create a plugin with this code:

```
local function main()
   local selectedSequence = SelectedSequence()
   local path = GetPath(Enums.PathType.UserSequences)
   selectedSequence:Import(path,"mySelectedSequence.xml")
end

return main;
```

33.3.2.6. ToAddr

Description

The ToAddr Lua function is used to provide an API object with an address that can be used for further commands.

Syntax

ToAddr(light_userdata:handle): string:address

Example

To return the address of the first sequence of the selected data pool and go it through the grandMA3 command line, create a plugin with this code:

```
local function main()
   local mySequence = DataPool().Sequences[1]

   Cmd("Go %s", mySequence:ToAddr())
end

return main
```

34. Data Pools

All the different show data (except the patch, color theme, and timecode slots) are stored in different pools. They actually all exist as children inside a data pool parent object.

A new show file creates a default data pool. All other show data pool objects are inside this data pool object.

New data pool objects can be created giving a completely new set of pools.

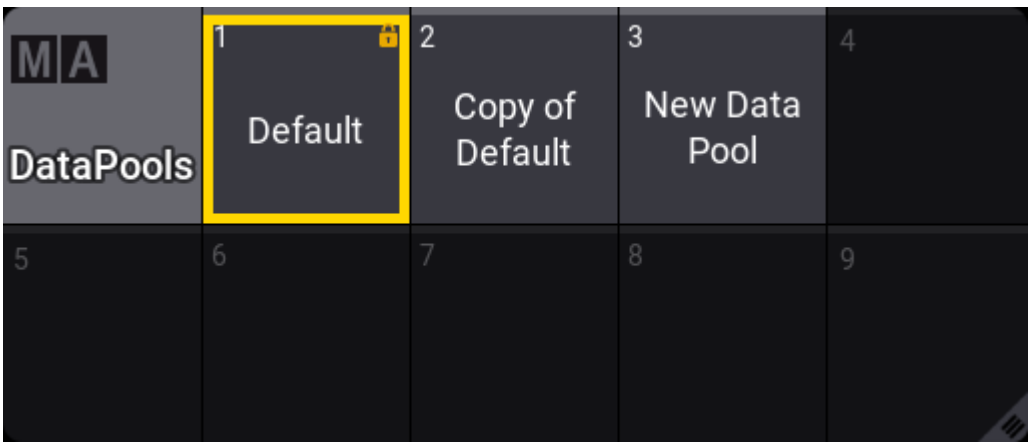
This can be very useful if several shows or acts use the same patch or for instance, each song in a big catalog of a band can be in their own data pool.

These are the elements in the data pools:

- Worlds
- Filters
- Preset Pools
- Groups
- Sequences
- Plugins
- Macros
- MAtricks
- Configurations (executor configuration)
- Pages
- Layouts
- Timecodes

Data Pool Window

The best way to see the different data pools is in the **Data Pool window**. This can be created like any other window - learn how in the **Add Windows topic**.



Data pool with some data pool objects

The pool can be used to select the desired object by tapping it. Any of the different data pool operations mentioned below can be done using the appropriate keyword in combination with the pool window. But it can also be done using the command line and the **DataPool keyword**.

The data pools can also be found in Menu / Settings / User Configuration / Pools. This is a list form of the data pools.

Create a New Data Pool Object

Data pools need to be stored in order to be created. These are different ways to create a new Data Pool.

- Press Store and then tapping an empty pool object.
- Press Store MA + Preset + Preset object number Please.
- Open the swipey commands on an empty pool object and choose Store.
- Using the command line: **Store DataPool [number/name]**

Performing an "edit" command on an empty pool object also creates the data pool object.

Copy a Data Pool Object

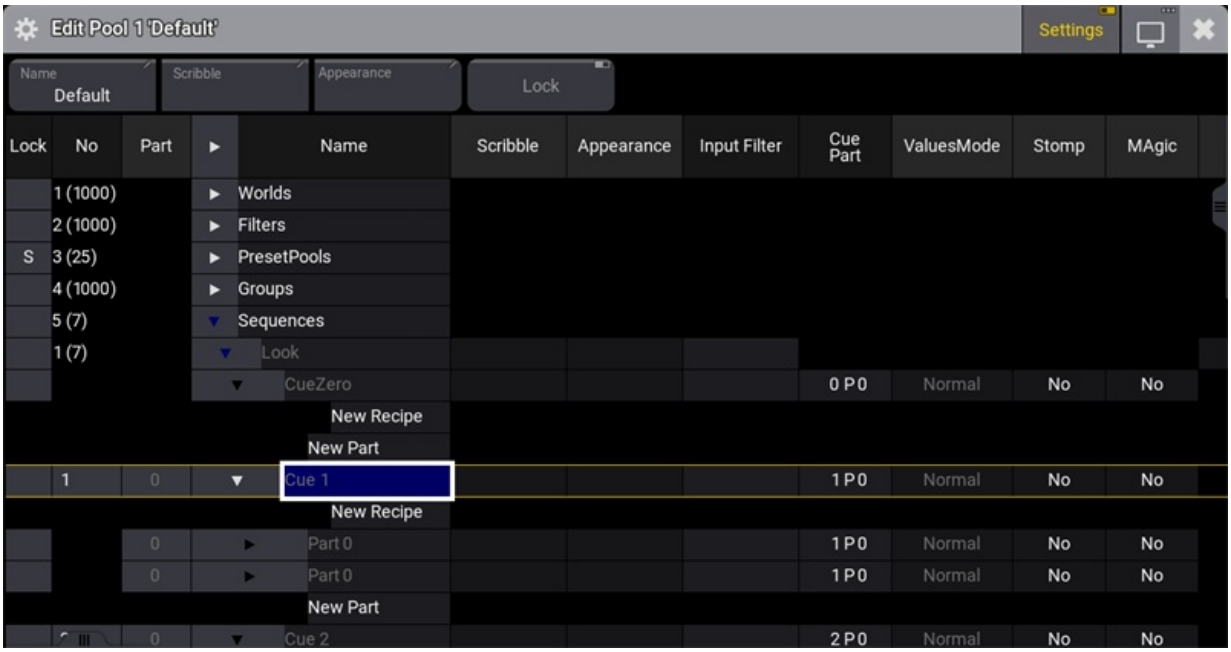
A data pool can be copied. This makes a copy of all the elements in the data pool. The copies a not linked. Making a change in one of the copies after the copy action does not change the other copy.

- Press Copy and then tapping the source pool object and then an empty pool object.
- Press Copy MA + Preset + Preset source object number At destination object number Please.
- Open the swipey commands on the source pool object and choose Copy then tap an empty pool object.
- Using the command line: **Copy DataPool [number/name] At [number/name]**

Hint:
Objects inside a data pool can be copied from one pool to another using the normal **Copy** and **Paste** commands.

Edit a Data Pool Object

Editing a data pool object opens an editor with a structured list of the elements in the data pool. Unfolding the elements in the list shows the settings for the elements.



Data pool in edit mode - Sequence unfolded

This is a deep look into the structure of the data.

Changing a setting here changes the setting for the object.

Edit a pool object using one of these methods:

- Press Edit and then tap the desired pool object.
- Press Edit MA + Preset + Preset object number Please.
- Open the swipey commands on the pool object and choose Edit.
- Using the command line: **Edit DataPool [number/name]**.

Reference Data Pool Objects From a Different Data Pool

A data pool element that exists inside a data pool can be used by other data pools.

For instance, imagine a repertoire theater. All default elements are stored in the data pool one. Including a sequence that controls the house lights. Tonight's show is programmed in data pool two. The sequence can be copied from data pool one, but if changes are made to the sequence later, then it would need to be copied again into the second data pool. Instead, the original sequence can be assigned to an executor in data pool two.

The image of the edit menu above shows that the structure in the data pool is numbered. So sequence 2 in this data pool is data pool object 1.5.2. The first number is the data pool number. The second number specifies that it is a sequence. The third number specifies that it is sequence number 2. If this element needs to be assigned somewhere, then it is the data pool object that needs to be assigned.

Example

Data pool 1 has sequence 2. This needs to be used in data pool 2 on (page 1) executor 201.

Make sure data pool 2 is the selected pool.

The example can be achieved using the command line: **Assign DataPool 1.5.2 At Page 1.201**.

Now the sequence in data pool 1 can be controlled from the executor in data pool 2.

Some menus give access to select the data pool. In the title bar of the menus, there is a button called DataPool that can toggle between the different data pools.

Delete a Data Pool Object

Important:
Deleting a data pool object also deletes everything inside the data pool!

Unlocked data pools can be deleted using any of the following methods:

- Press Delete and then tapping the pool object.
- Press Delete MA + Preset + Preset object number Please.
- Open the swipey commands on the relevant pool object and choose Delete.
- Using the command line: **Delete DataPool [number/name]**

35. System

This section holds system windows that provide information on the system, operational views, and overall system settings.

35.1. Date and Time

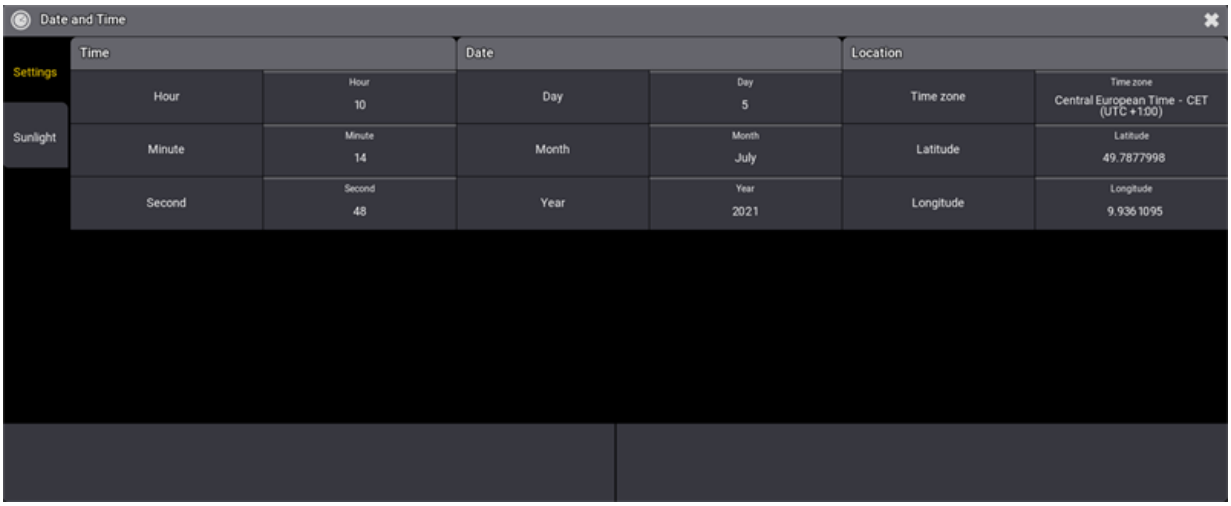
Sets date and time.

To access **Date and Time**:

1. Press Menu
2. Tap Settings.

A drop-down menu opens.

3. Tap Date and Time.
4. Date and Time window opens.



Open Date and Time

Date and Time

1. To set the date and time, tap the buttons with a title bar on the right of the columns Time and Date.
   The calculator opens.
2. Enter values and tap Please.

Date and time are set.

Location

The time zone refers to the international time standard UTC.

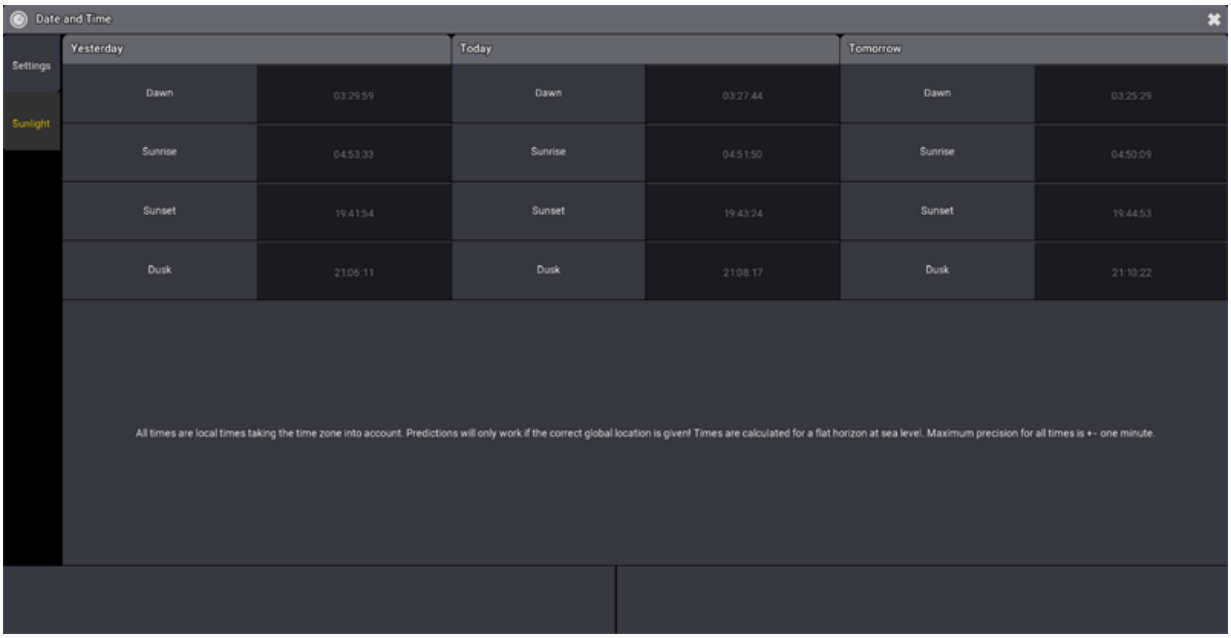
1. To enter the current location, tap each of the buttons –Timezone, Longitude, Latitude – on the right of the column Location.
   The calculator opens.
2. Enter values and tap Please.

Location is set.

Sunlight

The sunlight window, displays yesterday's, today's, and tomorrow's dawn, sunrise, sunset, and dusk.

All times are local times taking the time zone into account. Predictions will only work if the correct global location is given. Times are calculated for a flat horizon at sea level. Maximum precision for all times is +- one minute.



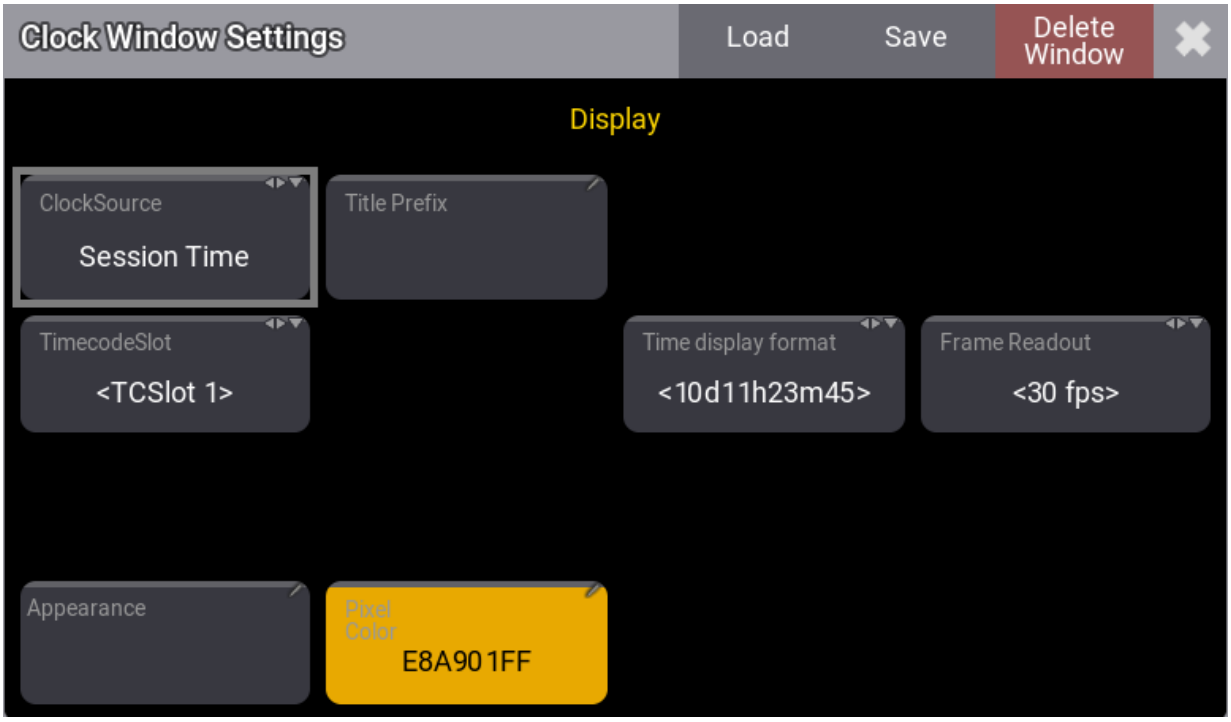
Hint:
The console calculates nautical times for events such as sunrise, dusk, sunset and dawn.

35.2. Clock

1. To open the clock, access **Add Window**.
2. Tap More.
3. Tap Clock.

For information, see **Add Window**.
To learn more about windows, see **Window Settings**.

Tap MA in the left corner of the title bar to open the clock settings.



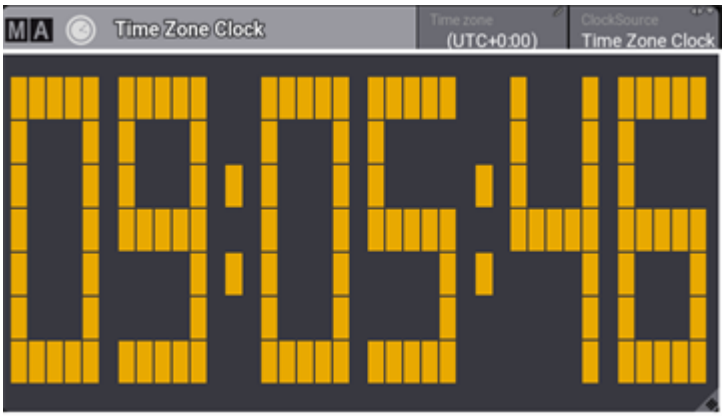
Clock

- The clock is displayed in yellow digits by default.

- It displays the time set in **Date and Time**.
- Tap ClockSource in the title bar to toggle between Session Time, Time Zone Clock, and Timecode. To learn more about timecode settings, see **What Are Timecode Slots**.

Select Time Zone

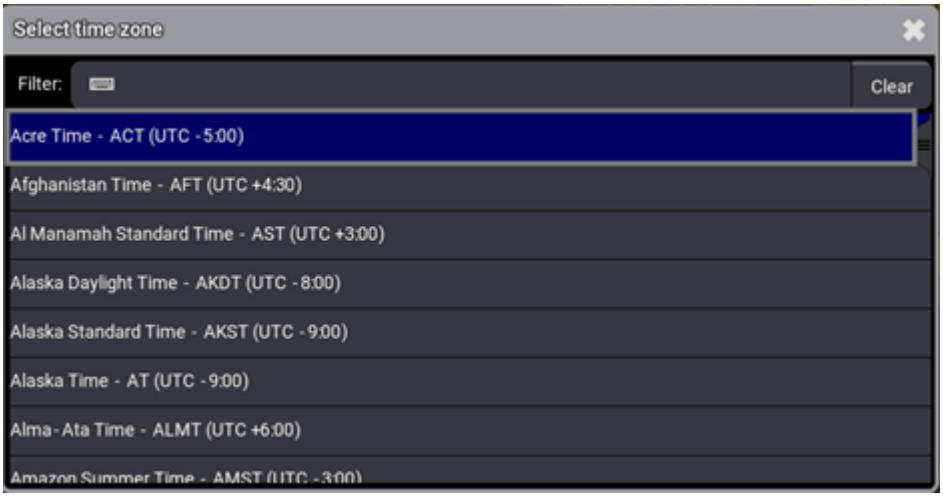
- Set the ClockSource to Time Zone Clock.



Time Zone Clock window

- Tap Time zone.

The Time zone pop-up opens:

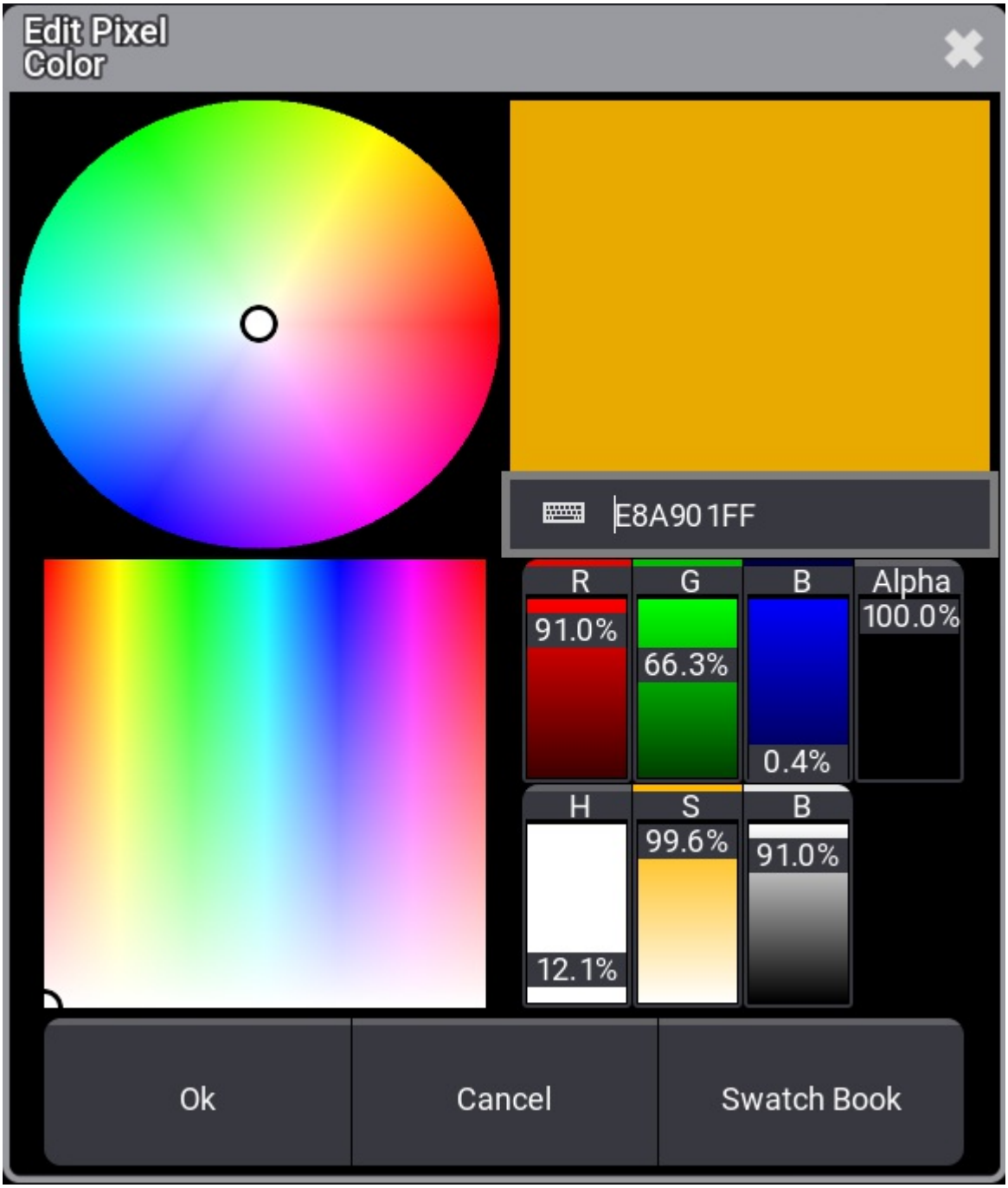


Time zone pop-up

- Select the time zone.

Edit Color of Digits

1. To change the color of the digits, tap MA in the top left corner, then tap PixelColor.
2. The pop-up Edit Pixel Color opens:



Edit color of the digits

3. Tap to select a color in the color picker.

4. Tap Ok.
5. The color of the digits is modified.

Appearance as Background Color

1. Create an appearance in the appearance pool.
   For more information, see **Create Appearances**.
2. Press Assign, tap an appearance in the pool, and tap the clock title bar.
   For more information, see **Use Appearances**.

35.3. Desk Lights

This topic describes the setting of desk lights.

To learn more about color themes, read the **topic Color Themes**.

To access the Desk Lights & Color Theme menu:

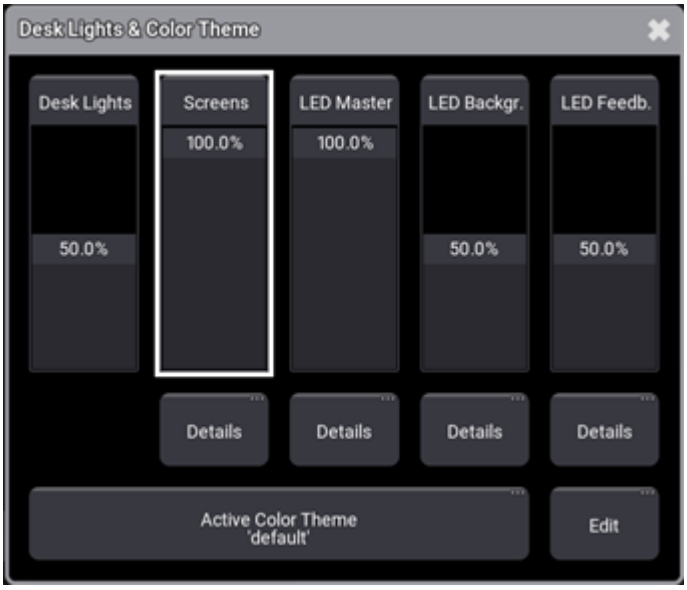
1. Press Menu
2. Tap Desk Lights & Color Theme

Menu window

The Desk Lights & Color Theme menu opens.

Desk Lights & Color Theme menu

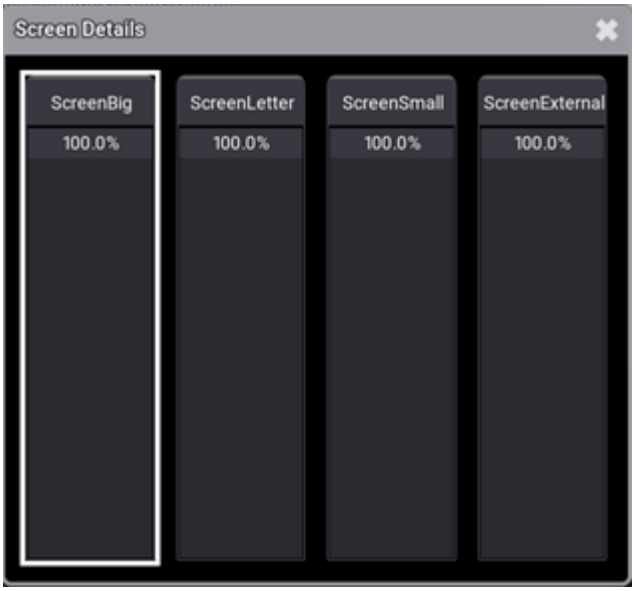
- To adjust the desk light intensity, move the Desk Lights fader up or down.
- To adjust the general brightness of all screens, move the Screens fader up or down.



Screen brightness fader

- To adjust the brightness of individual screens, tap Details below the Screens fader.

The Screen Details pop-up opens.

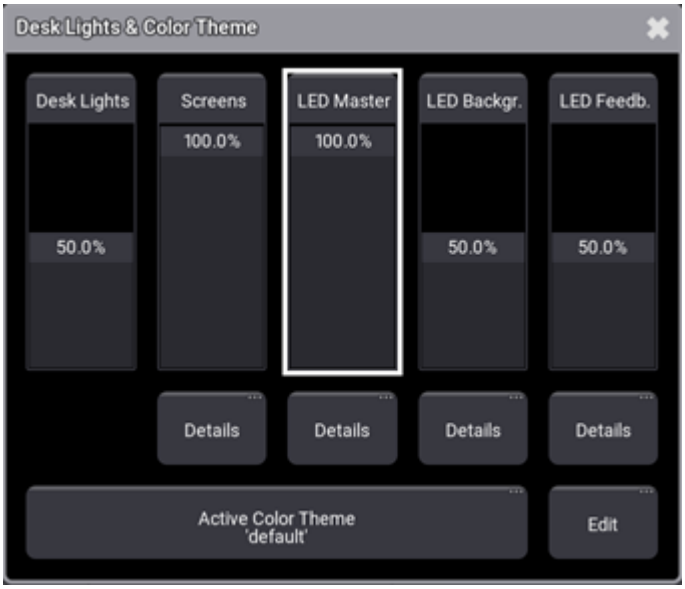


Screen Details pop-up

- To adjust the brightness of the desired screen, move the respective Screen fader up or down.

Restriction:
The Screen External fader currently has no function.

- To adjust the general intensity of all LEDs, move the LED Master fader up or down.



LED Master fader

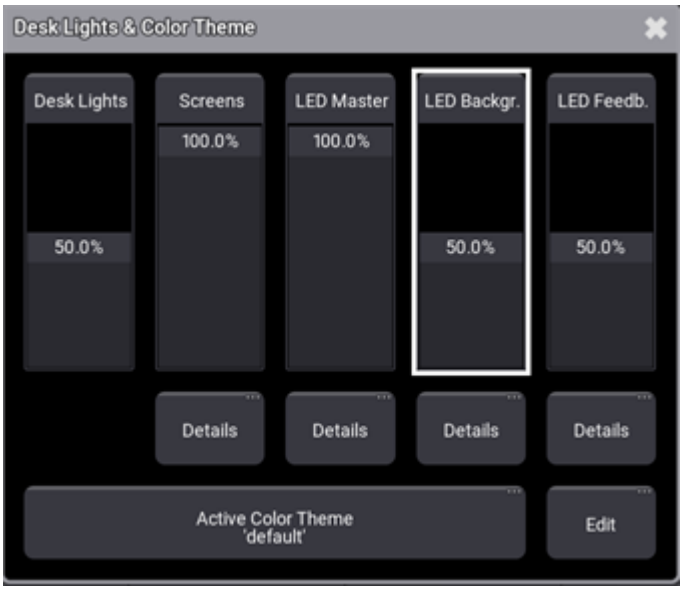
- To adjust the LED intensity of the individual elements, tap Details below the LED Master fader.

The LED Details pop-up opens.

LED Details pop-up

To adjust the intensity of the desired element, move the respective LED fader up or down.

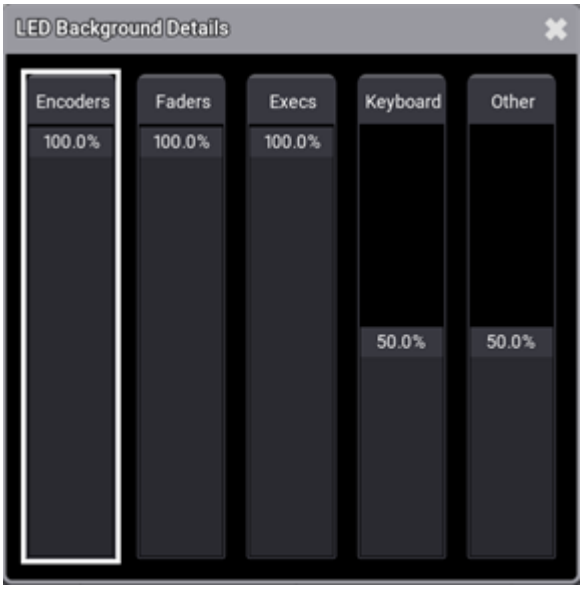
- To adjust the general LED Background intensity, move the LED Background fader up or down.



LED Background fader

- To adjust the LED background intensity of the individual elements, tap Details below the LED Master fader.

The LED Background Details pop-up opens.



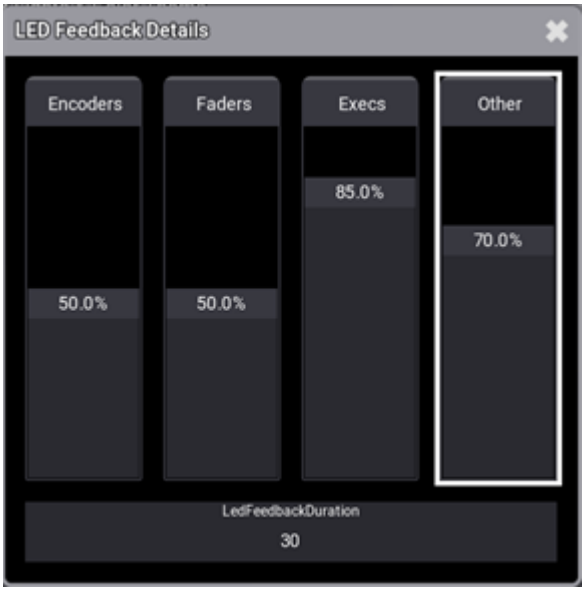
LED Background Details pop-up

- To adjust the general LED Feedback intensity, move the LED Feedback fader up or down.

LED Feedback fader

- To adjust the LED Feedback intensity of the individual elements, tap Details below the LED Feedback fader.

The LED Feedback Details pop-up opens.



LED Feedback Details pop-up

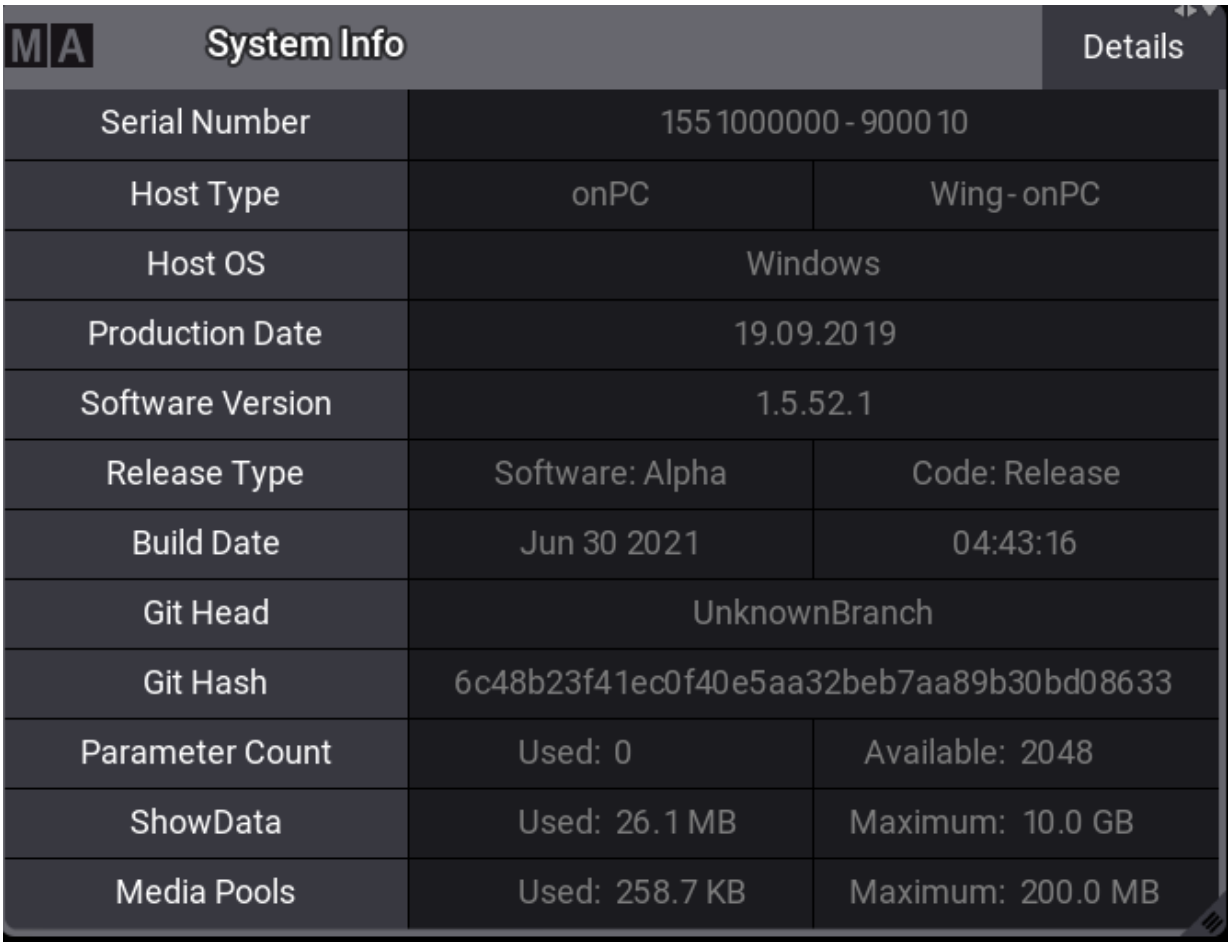
35.5. System Information

System Info is a mixture of system and performance monitoring of the console.

To add a System Info window, follow the instructions under **Add windows**.

In the Add Window pop-up:

1. Tap More.
2. Tap System Info.
   The System Info window opens.



System Info – details on onPC

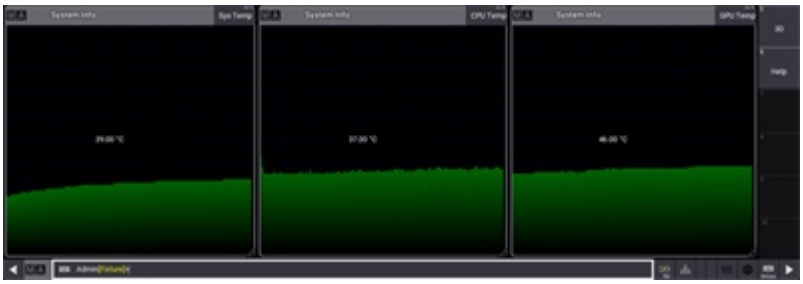
To adjust the System Info window settings, follow the instructions under **Window Settings**.

CPU, GPU, system temperatures and fan speed information can be displayed in the System Info window.

Temperature check

CPU, GPU, and system temperatures can be displayed on:

- full-size
- light
- replay unit
- compact XT



Example of temperature windows

The grandMA3 compact can only display CPU and system temperatures.

These products are not able to display temperature information:

- extension
- processing unit
- All Nodes
- viz-key
- All onPC solution devices

Fan check

The fan view in the System Info window displays the system/CPU cooling fan speed in correlation to its maximum speed.

Fan speeds can be displayed on:

- full-size
- light
- replay unit
- compact (XT)

These products are not able to display fan speed information:

- extension

- processing unit
- All Nodes
- viz-key
- All onPC solution devices

System Info is divided into different views you can select using the button in the title bar on the right.

Toggle between:

Details

Details displays other relevant system info such as the serial number or the build date.

Realtime

Realtime displays the workload of the system in milliseconds per DMX frame.

Timing

Timing displays the time that is required to render screens on the console.

MainLoop, Swap, and End display internal processings and in which displays they take place.

CPU

CPU displays the workload of the main processor.

Memory

Memory displays the RAM of the system in GB.

CPU Temp

CPU Temp displays the current temperature of the console.

GPU Temp

GPU Temp displays the current temperature of the graphics processing unit of the console.

Sys Temp

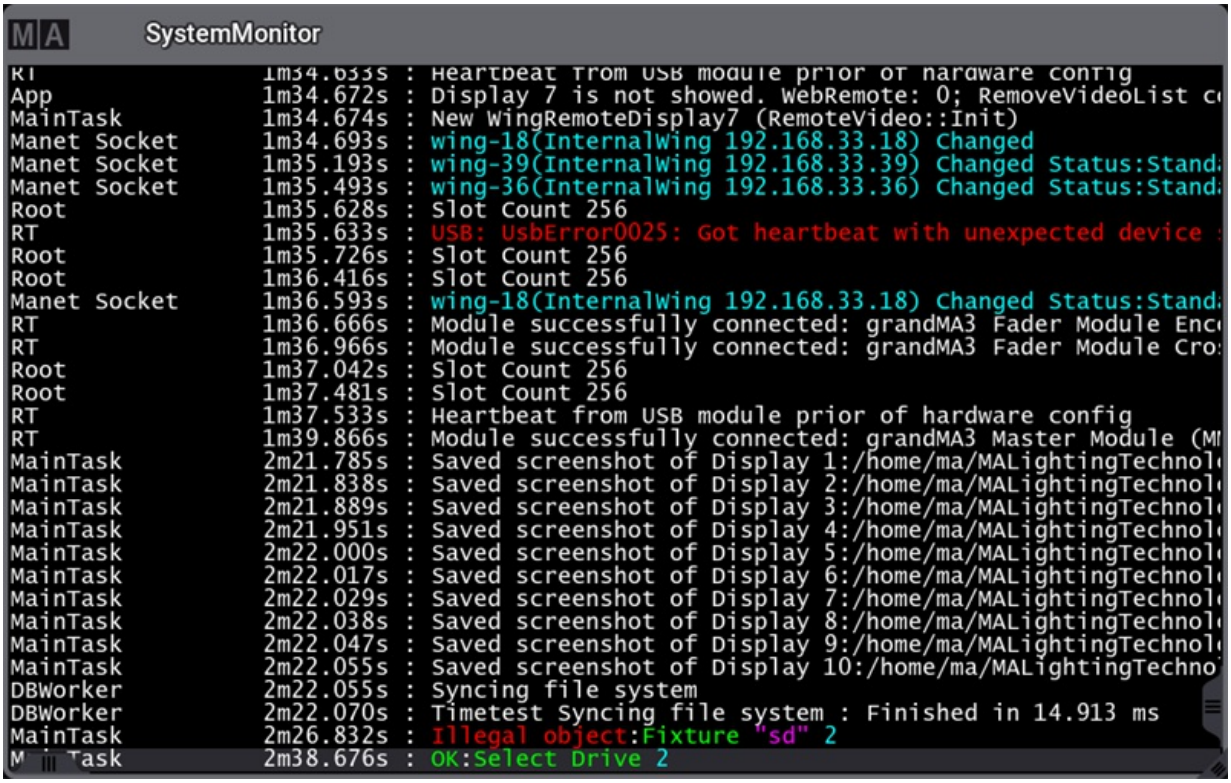
Sys Temp displays the current temperature of the central computer board.

Fan

Fan displays the rotational speed of the fan of the cooling system in RPM.

35.6. System Monitor

The system monitor is a window that can be created like any other window using the **Add Window pop-up**.



System Monitor

It shows what is happening at the station. This includes feedback on user commands. It is a log of the different things happening in the background. It also shows warnings, errors, and changes to the system.

It is a debugging window that can provide useful information if there are any problems with the system.

There are only two settings for the window. Appearance and Front Size. Both are described in the **Window Settings topic**.

The default background is black. The bottom line shows the latest thing that happened.

If the system monitor is scrolled up, then the background changes to a red color to indicate that is is not the latest data and the auto-scrolling function that automatically shows the latest info is not active. Scroll back down to the bottom to reactivate the auto-scroll function.

35.7. Info Window

Open an Info window like any other window. See **Add Window**.

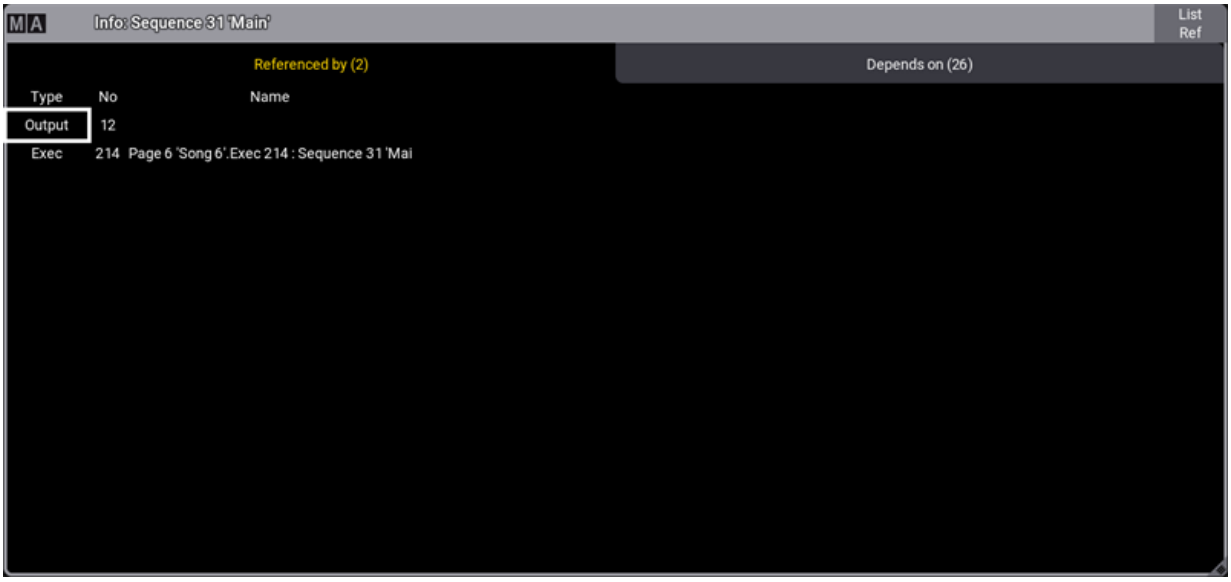
In the Add Window pop-up, tap More, then tap Info.

The Info window opens.

Tap ListRef in the title bar of the info window to enter the **ListRef** keyword in the command line, then tap the desired object you want to see the references of.

For more information, see **ListRef**.

Tap Referenced by to display the references.



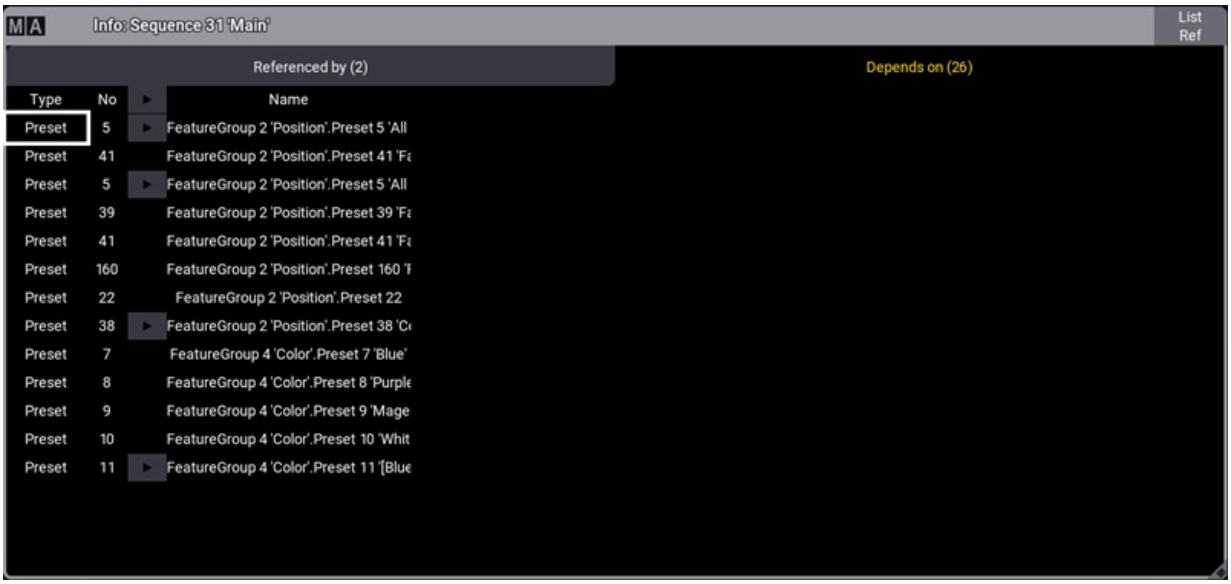
Info window displaying references

Type:
This column displays the targeted object reference type.

No:
This column displays the targeted object reference number.

Name:
This column displays the targeted object reference name.

Tap Depends on to display the dependencies.



Info window displaying dependencies

Type:
This column displays the type of the targeted object dependencies.

No:
This column displays the number of the targeted object dependencies.

Name:
This column displays the name of the targeted object dependencies.

Important:
The results of **ListRef** for a targeted object will stay visible until you perform a new search or change the view. It can not be cleared or deleted.

36. Sound

The grandMA3 system can receive a sound signal. This signal can be used in Phasers and different **Sound Channels** can be used to give value to attributes. Learn more about this in the Sound Window topic.

The grandMA3 onPC can playback sounds that are imported into the **Sounds Pool**. Learn more about this in the Sounds Pool topic.

36.1. Sound Window

The **Sound** window displays an incoming audio signal as a raw waveform. This window also breaks down the strength of that signal into several bands of a few different widths. The **Sound** window also displays calculations performed by the console to find steady beats within the signal.

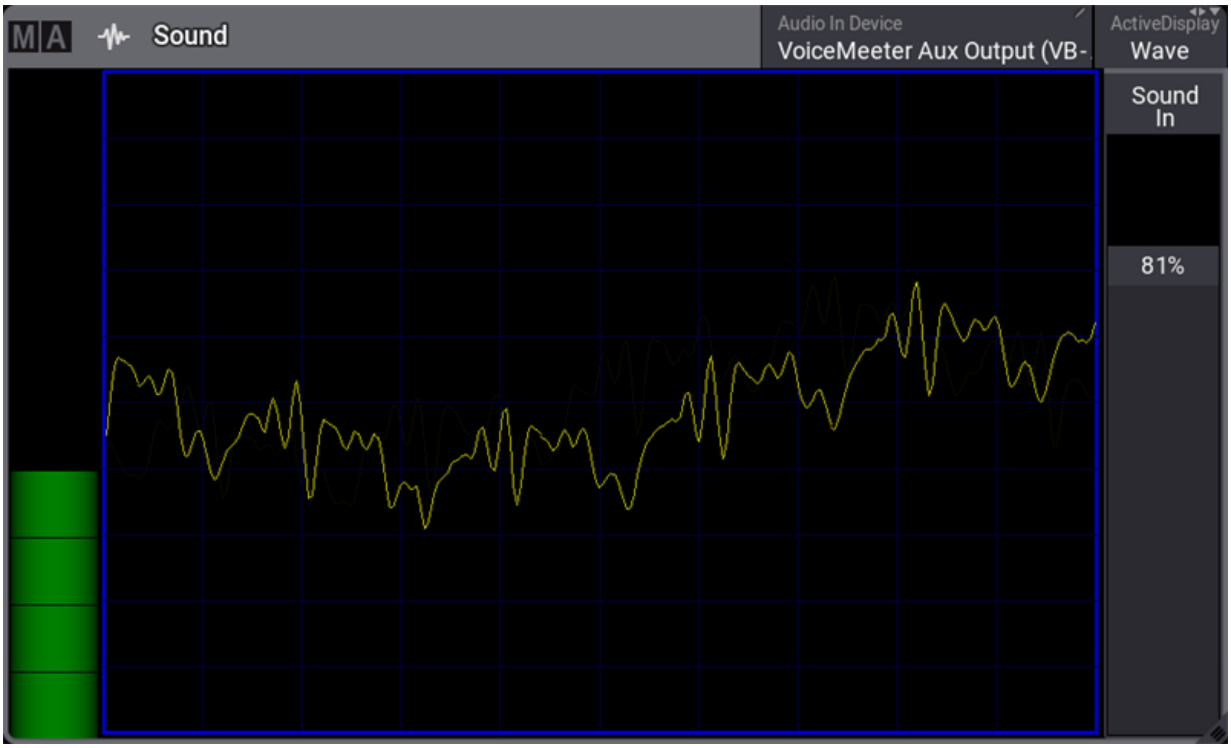
Requirement:
An active sound input is required in order for the console to display useful information in the **Sound** window. For information about connecting an audio input to a console, see the **Connect Audio In** topic. The grandMA3 onPC software uses any sound input hardware designated by the operating system.

The **Add Window** pop-up includes the **Sound** window under the More tab. For more information about the **Add Window** pop-up, see the **Add Windows** topic.

The right corner of the **Sound** window title bar includes an ActiveDisplay button. Tap this button to cycle through available display mode options or tap and swipe to open a pop-up menu containing all of the available display mode options for the **Sound** window.

On the grandMA3 onPC, there is another button in the title bar called Audio In Device. This is a list of available sound devices on the computer that can be used as a sound input device.

Wave Display



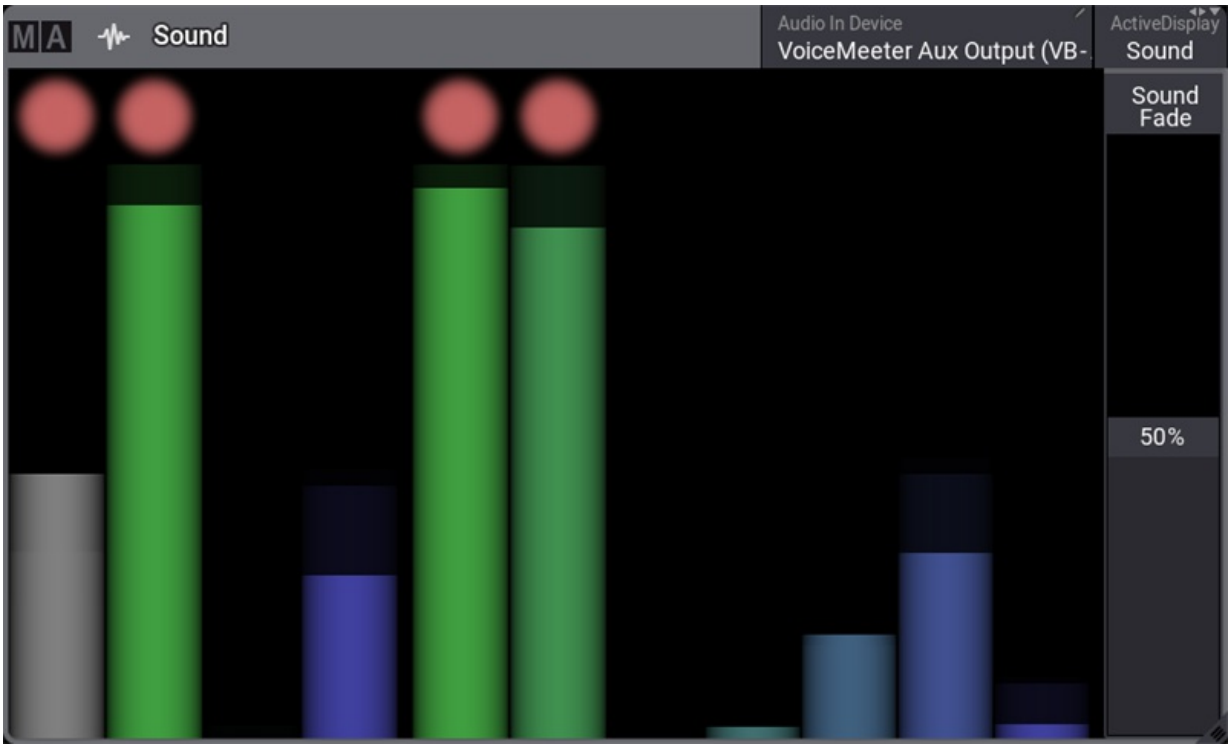
Sound window in Wave display mode

The left side of the window includes a simple VU meter shown as a vertical bar.

The main area of the window displays the raw waveform of the incoming audio signal.

The right side of the window includes quick access to the **Sound In** master. Adjusting the **Sound In** master changes the volume of the input signal. This change is immediately visible in both the VU meter and the waveform. For more information about the **Sound In** master, see the **Grand Masters** topic.

Sound Display



Sound window in Sound display mode

The **Sound** display mode of the **Sound** window displays the incoming audio signal as a series of bars representing the volume of different bands of frequencies. When a given bar reaches a peak, a dot appears above the bar. This dot represents a possible sound trigger based on that frequency band.

The eleven bars shown in the window break down the frequencies of the audio signal in three different ways, displaying all three breakdowns simultaneously. The first bar represents the volume of the whole signal. The next three bars divide the signal into **Bass**, **Mid**, and **High** frequencies. And, the remaining seven bars divide the frequencies equally into narrower bands.

These eleven bars relate directly to the first eleven **Sound Channels** and inversely to the remaining **Sound Channels**. Available **Sound Channels** include:

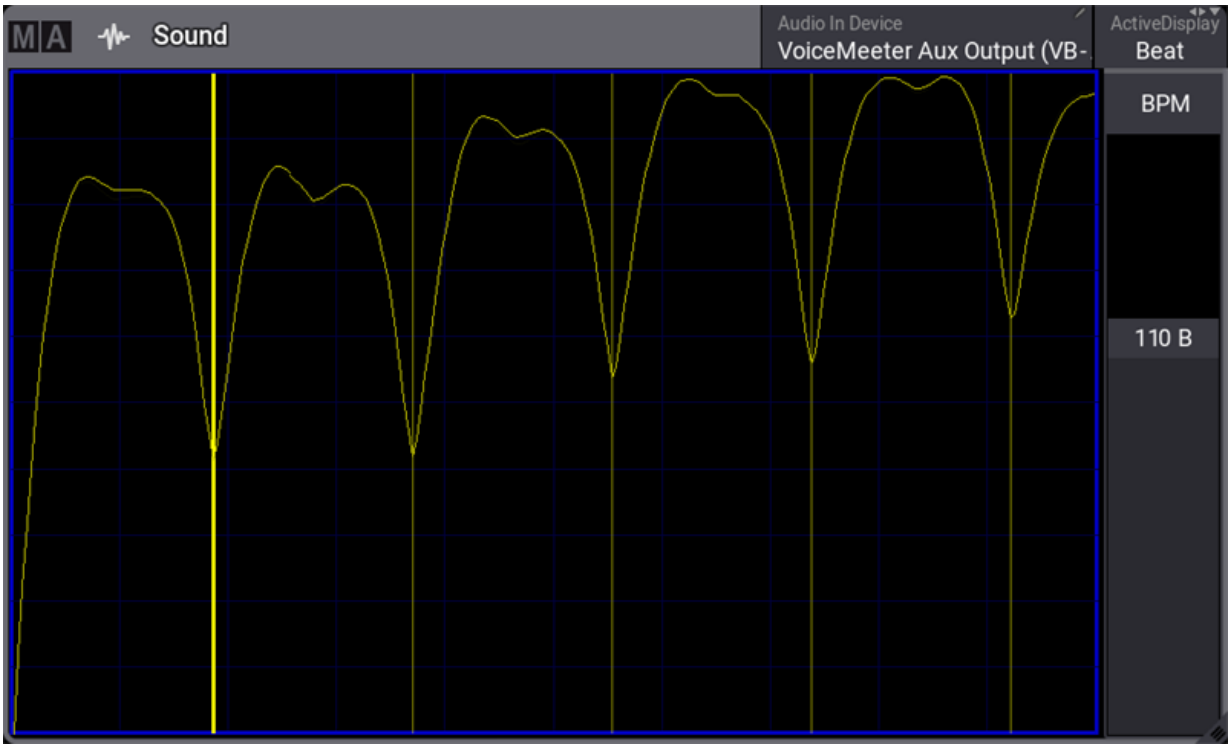
1. **All**

2. **Bass**
3. **Mid**
4. **High**
5. **Band1**
6. **Band2**
7. **Band3**
8. **Band4**
9. **Band5**
10. **Band6**
11. **Band7**
12. **InvAll**
13. **InvBass**
14. **InvMid**
15. **InvHigh**
16. **InvBand1**
17. **InvBand2**
18. **InvBand3**
19. **InvBand4**
20. **InvBand5**
21. **InvBand6**
22. **InvBand7**

Set the value of any parameter to follow one of the **Sound Channels** in the command line, using the **SoundChannel** keyword, or the Sound Codes tab of the **Calculator.**

The right side of the window includes quick access to the **Sound Fade** master. Adjusting the **Sound Fade** master changes the fall-off speed of all of the channels. While a lower number results in more accurate tracking of quickly changing dynamics, the response may appear undesirably erratic. A higher value results in a smoother response. For more information about the **Sound Fade** master, see the **Grand Masters** topic.

Beat Display



Sound window in Beat display mode

The **Beat** display mode of the **Sound** window displays the incoming audio signal after some additional processing. This processing looks for repeating beats. The processed signal appears as a series of hills and valleys. Deeper valleys form where louder pulses in the incoming signal line up more consistently over time.

Dim, yellow, vertical lines appear in the valleys with the most consistent repeated pulses. These lines represent beat candidates, which the **BPM** speed master can follow.

A bold, yellow, vertical line appears when the **BPM** master locks onto a beat. The **BPM** master adjusts automatically to follow any changes detected in the incoming signal.

The right side of the window includes quick access to the **BPM** speed master fader. For more information about the **BPM** speed master, see the **Speed Masters** topic.

36.2. Sounds Pool

Sound files can be imported into the grandMA3 in the Sounds Pool.

MP3 files are currently supported.

Sounds Pool

The sound files in the pool can be played back using a GO+ command and they can be stopped with the Off command. They can also be paused with the Pause command. The paused playback can be resumed using the Pause command again.

Sound pool objects can be assigned to executors and can be played back by the executor. Sounds on executors also give access to adjust the master level for each sound pool object.

The **Sound Out** master needs to be turned up for the sounds to output. This can be done in the **Master Controls**. It can also be assigned to an executor.

On the consoles, the sound output is the S/PDIF Out connector.

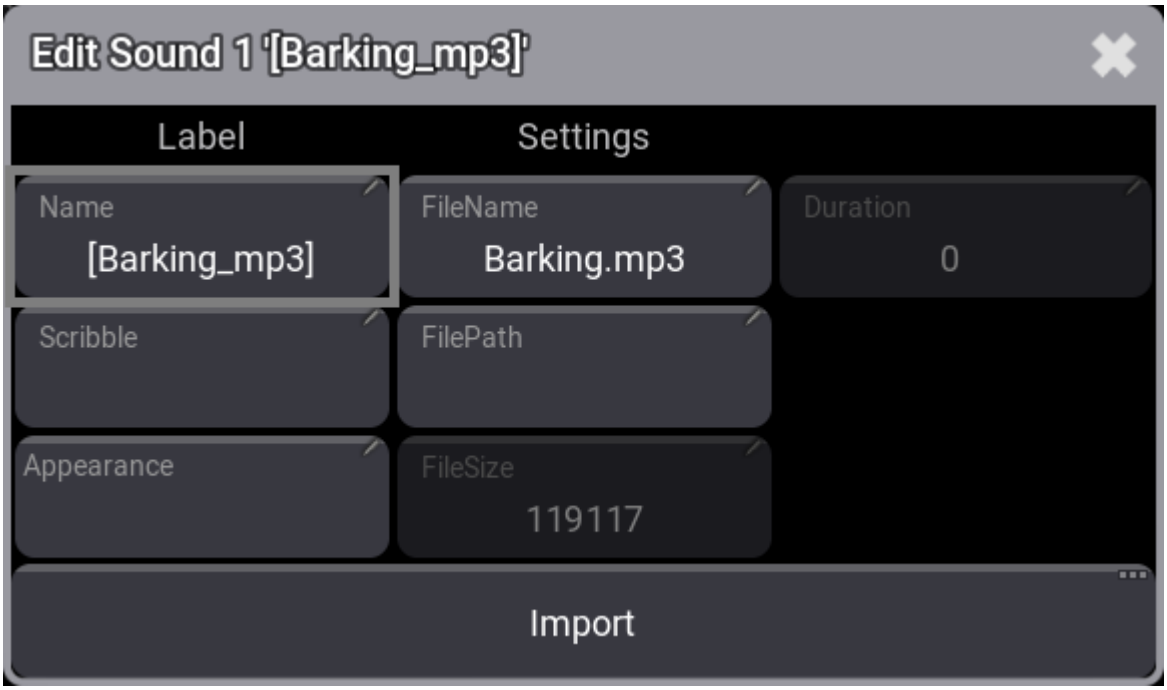
In the grandMA3 onPC, a compatible sound output device needs to be selected. This is done in the **onPC settings menu**.

Restriction:
Third-Party Software needs to be activated and agreed to in Menu / Settings / Software Update / EULA for sound to be played back. The button is in the lower right corner. Turning it On opens a pop-up that needs to be agreed to.

Import Sound

The sound file needs to be in the correct folder to be able to be imported. The folder is /gma3_library/Media/Sounds. Learn more about this in the **Folder Structure topic**.

1. Edit an empty sound pool object. This opens the **Edit Sound** pop-up:



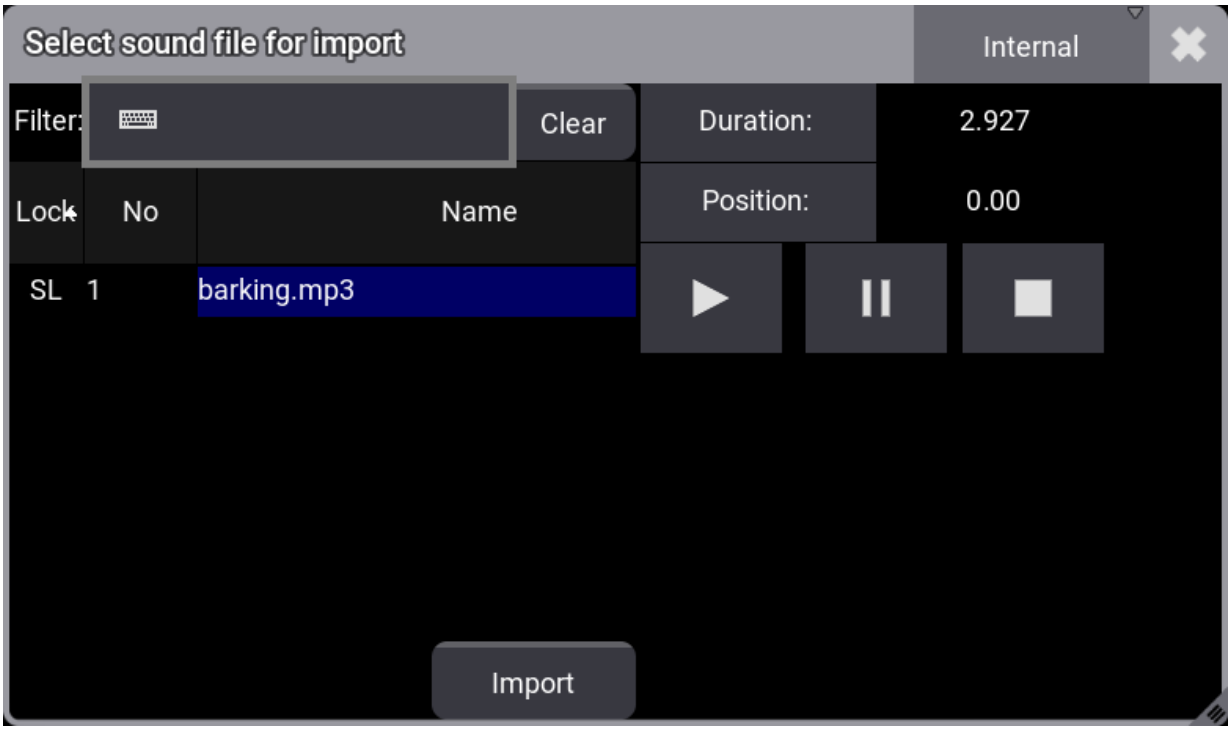
Edit Sound Pop-up - with an imported sound

This editor has the following settings:

- **Name**:
  This is the name for the pool object.
- **Scribble**:
  This is the scribble assigned to the pool object.
- **Appearance**:
  This is the appearance assigned to the pool object.
- **File Name**:
  This is the name of the sound file.
- **File Path**:
  This shows the path for the sound file if it is not in the default folder.
- **File Size**:
  This is only information. This is the file size of the sound file.
- **Duration**:
  This is only information. This shows the duration of the sound file.

2. Tap the Import button to import a sound file.

This opens the **Select Sound File for Import** pop-up.



Select Sound File for Import pop-up with a sound file

The title bar has a button to select the drive with the sound file.

Sound files are shown in a list on the left side.

The right side has information about the selected file, and the possibility to play, pause and stop the playback of the selected file.

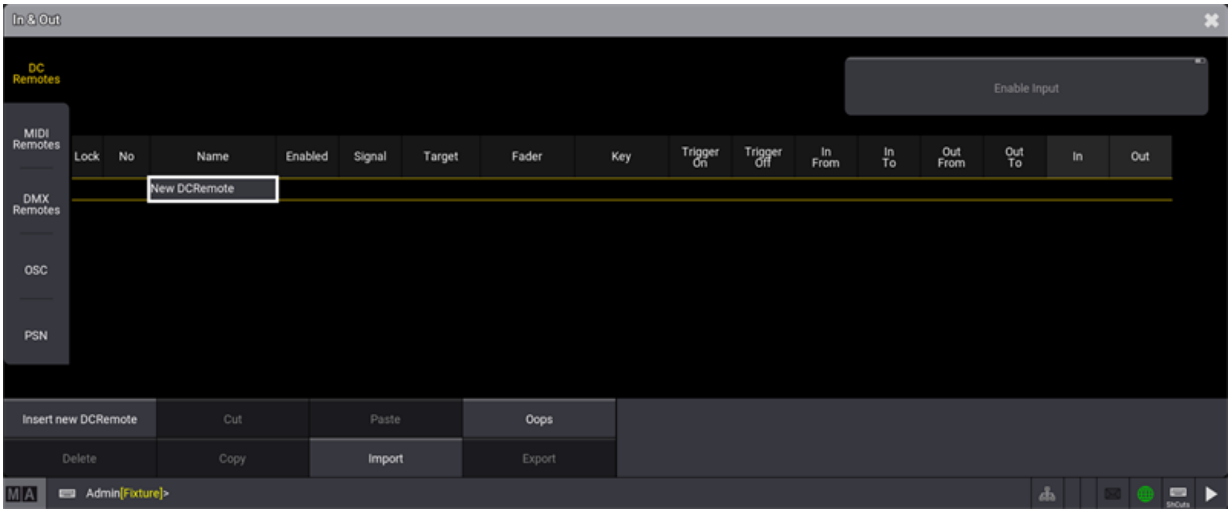
3. Select the desired file in the list.
4. Tap Import. This closes the Select Sound File pop-up.
5. Tap to close the Edit Sound pop-up.

Now the sound is imported and can be played back.

37. Remote In and Out

Remote Inputs are handled through the In & Out menu: DC Remotes, MIDI Remotes, DMX Remotes, OSC, and PSN.

- To open the **In & Out** window, press Menu and then tap In & Out.



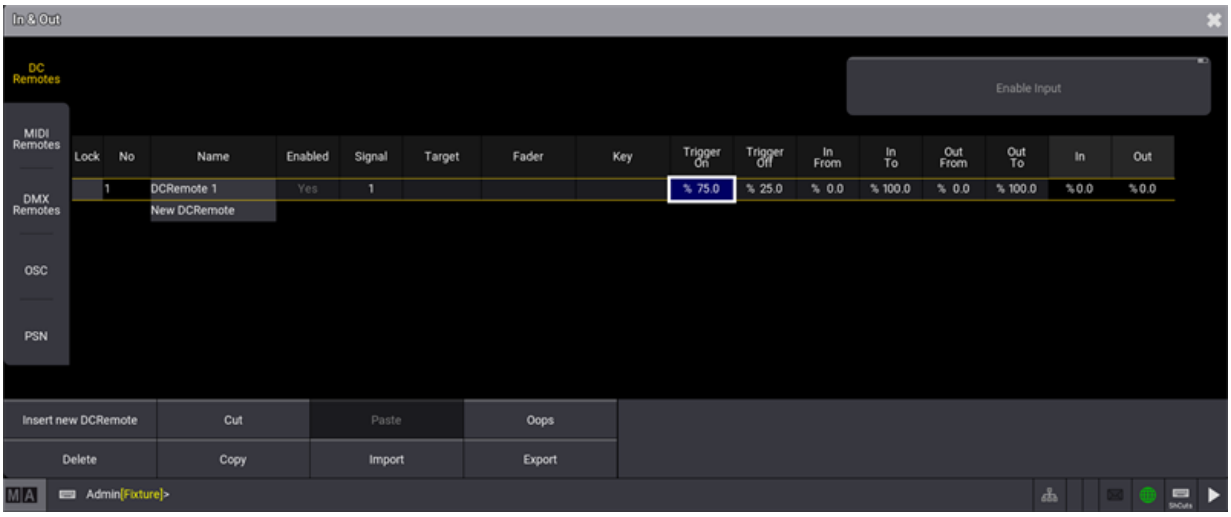
In & Out window

There are five tabs: **DC Remotes**, **MIDI Remotes**, **DMX Remotes**, **OSC**, and **PSN**.

Hint:
The different readout options take effect on the Trigger On/Off, In From/To, Out From/To values.

The common procedure and pop-ups throughout the three tabs DC Remotes, MIDI Remotes, and DMX Remotes are described as follows:

- To add a new element, e.g. DC Remote, tap Insert new DCRemote.

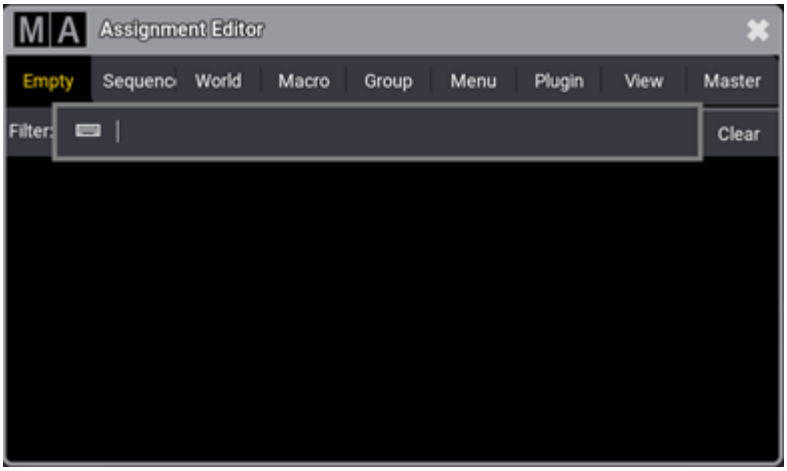


New DC Remote

- To enable or disable the Remote Input, right-click or tap and hold Enabled to toggle between Yes and No.

- To assign the target, tap and hold Target.

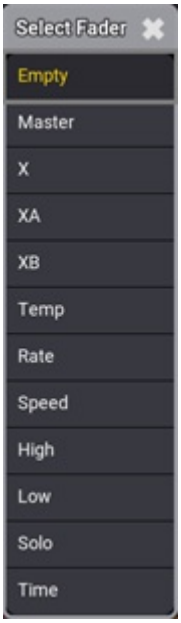
The Assignment Editor window opens:



Assignment Editor

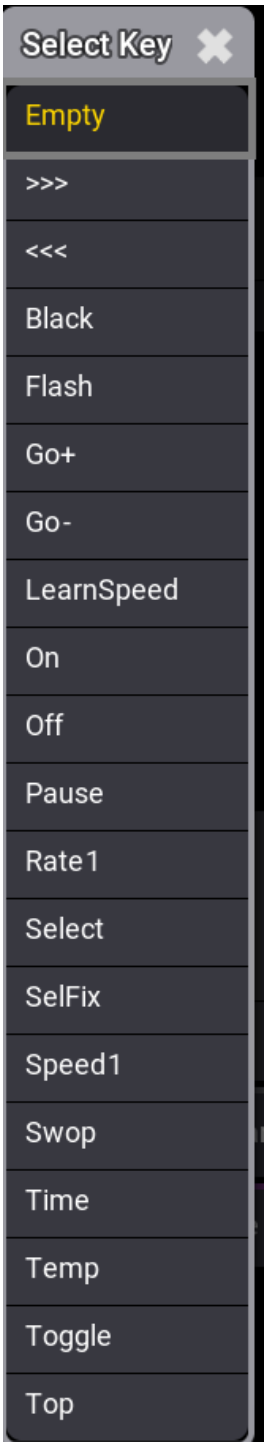
- Select the target, e.g. a sequence.

- To select a fader function, right-click or tap and hold Fader.
  The Select Fader pop-up opens.



Select Fader pop-up

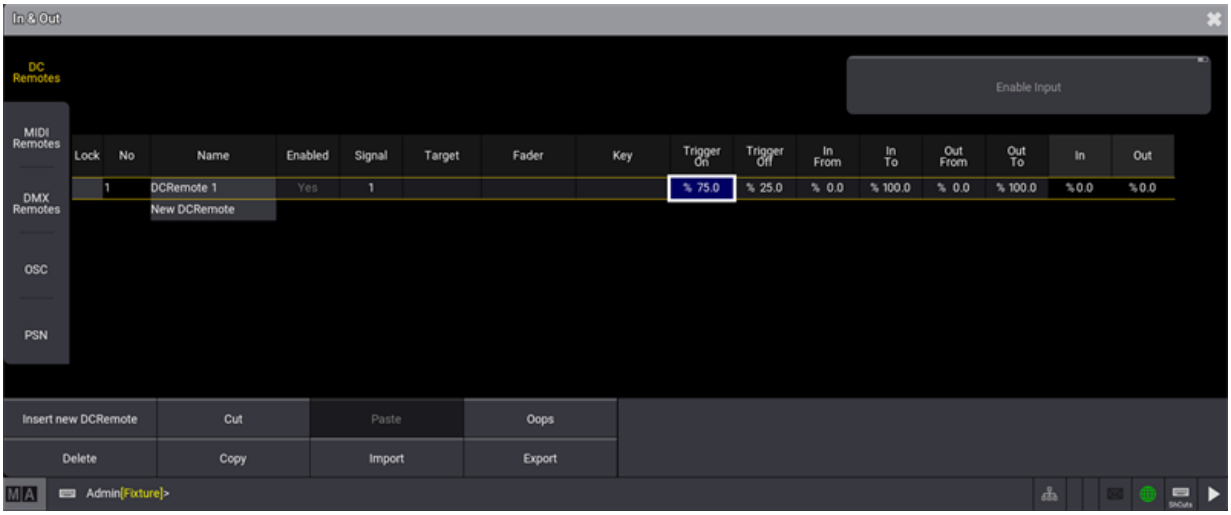
- To select a key function, right-click or tap and hold Key.
  The Select Key pop-up opens.



The Trigger On and Trigger Off values define the range of the key reaction.

If the Trigger On, for example, is set to 75 %, the Remote reacts as soon as the incoming signal exceeds this value.

If the Trigger Off, for example, is set to 25 %, the Remote stops reacting as soon as the incoming signal has fallen below this value.



The In From and In To values define the range of the input signal reaction for the defined fader.

DC Remotes and DMX Remotes thresholds are defined in percent, while MIDI Remotes thresholds are defined between 1 and 127 MIDI velocity.

- To set the In From value to a new value, tap In From .
  The In From pop-up opens.

In From pop-up

- To set the In To value to a new value, tap In To.
  The In To pop-up opens.

In To pop-up

The Out From and Out To values recalculate the input signal range to match up the output signal range. The output singal range defines the range to use of a fader.

If a fader, for example, should not be moved to 100 %, restrict the Out To value to e.g. 90 %.

- Example 1
  - In From/To: 20-80%
  - Out From/To: 0-100%
    - When the incoming signal is 10%, the fader is at 0 %.
    - When the incoming signal is 20%, the fader is at 0 %.
    - When the incoming signal is 30%, the fader is at 17 %.
    - When the incoming signal is 50%, the fader is at 50 %.
    - When the incoming signal is 80%, the fader is at 100 %.
    - When the incoming signal is 90%, the fader is at 100 %.
- Example 2
  - In From/To: 20-80 %.
  - Out From/To: 10-90 %.
    - When the incoming signal is 0%, the fader is at 10 %.
    - When the incoming signal is 20%, the fader is at 10 %.
    - When the incoming signal is 30%, the fader is at 23 %.
    - When the incoming signal is 50%, the fader is at 50 %.
    - When the incoming signal is 80%, the fader is at 90 %.
    - When the incoming signal is 100%, the fader is at 90 %.

The In and Out columns at the end of the sheet display the signal value of the incoming signal (In) and the resulting value for the selected fader function (Out).

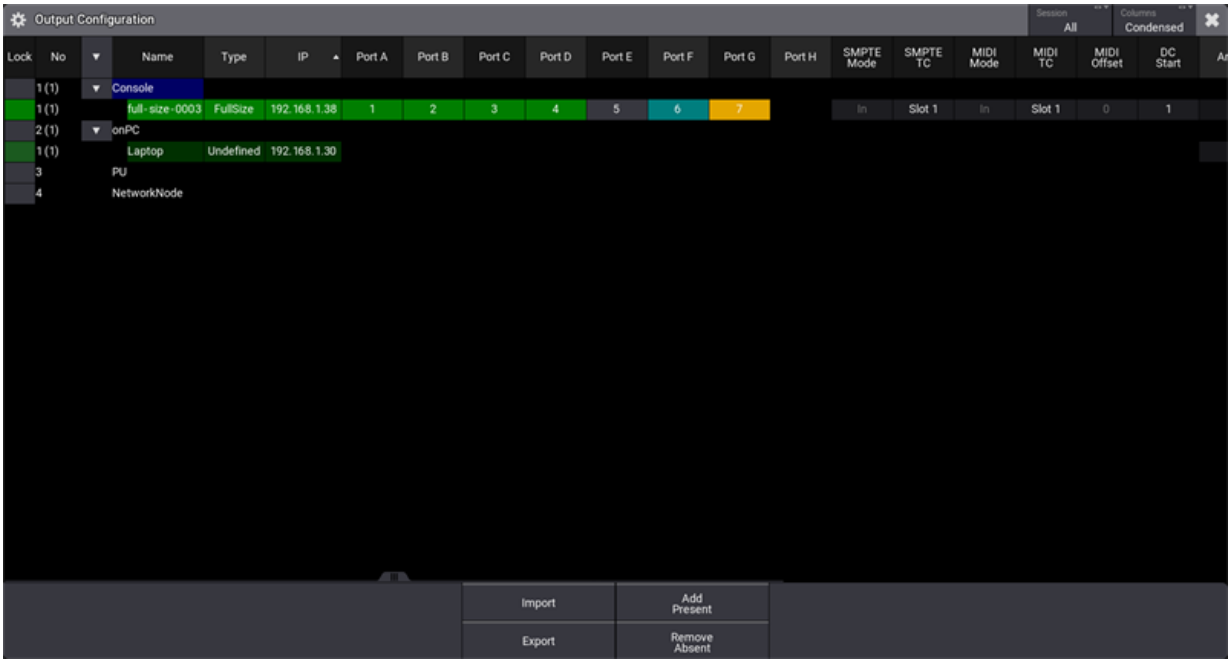
For detailed information about the values, read the **Remote keyword** topic.

37.1. DC Remotes

The **DC Remotes** tab is used to configure the DC Remote Control input on the rear panel.

To learn more about the hardware part of the input, read the **Connect DC Remote In topic**.

- To set the DC start signal, open the **Output configuration** menu:



Output configuration menu with a console and an onPC

Access the menu by pressing the Menu key and then tap Output Configuration.

-OR-

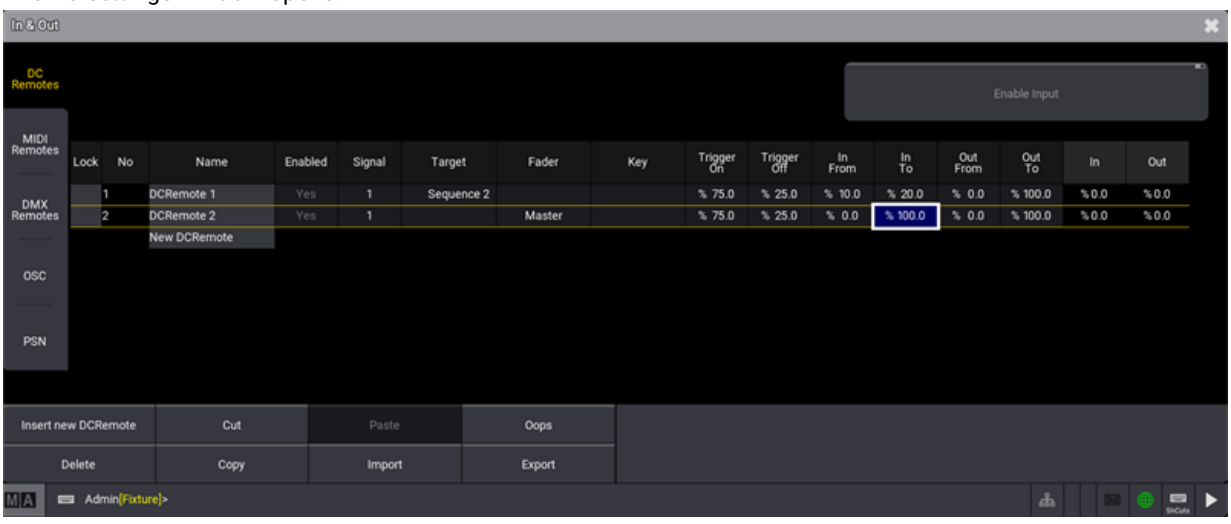
Use the command line to open the menu:

```
User name[Fixture]> Menu "OutputConfig"
```

- To set the DC Start value to a new value, tap DC Start.
  The DC Start pop-up opens:

DC Start pop-up

- To adjust the settings of the DC Remotes, switch to the In & Out window.
- To open the **In & Out** window, press Menu and then tap In & Out.
- Tap DC Remotes.
  The DC settings window opens.



The column Signal corresponds with the DC Start value in the Output Configuration Menu, see above.

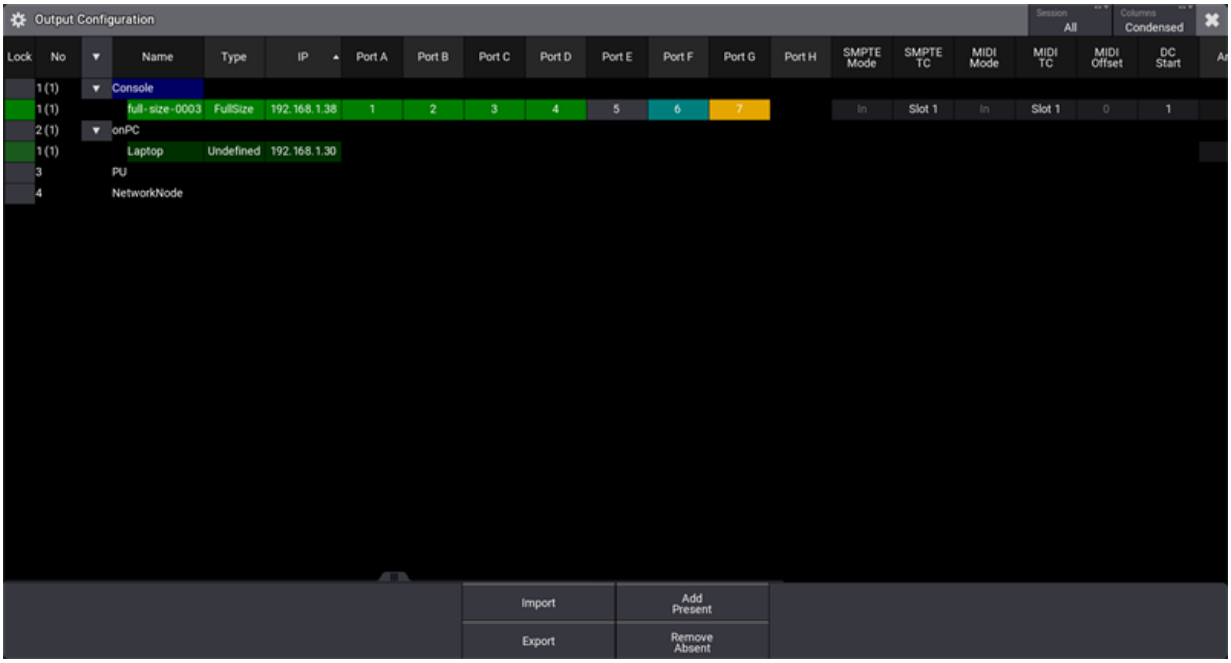
To learn more about the general DC Remotes settings like Name, Enabled, Fader, or Key read the **Remote In and Out topic**.

37.2. MIDI Remotes

The **MIDI Remotes** tab is used to define actions for incoming MIDI notes or MIDI Control Changes (CC).

To learn more about the hardware part of the input, read the **Connect MIDI topic**.

- To set the MIDI Offset, open the **Output configuration** menu:



Output configuration menu with a console and an onPC

Access the menu by pressing the Menu key and then tap Output Configuration.

-OR-

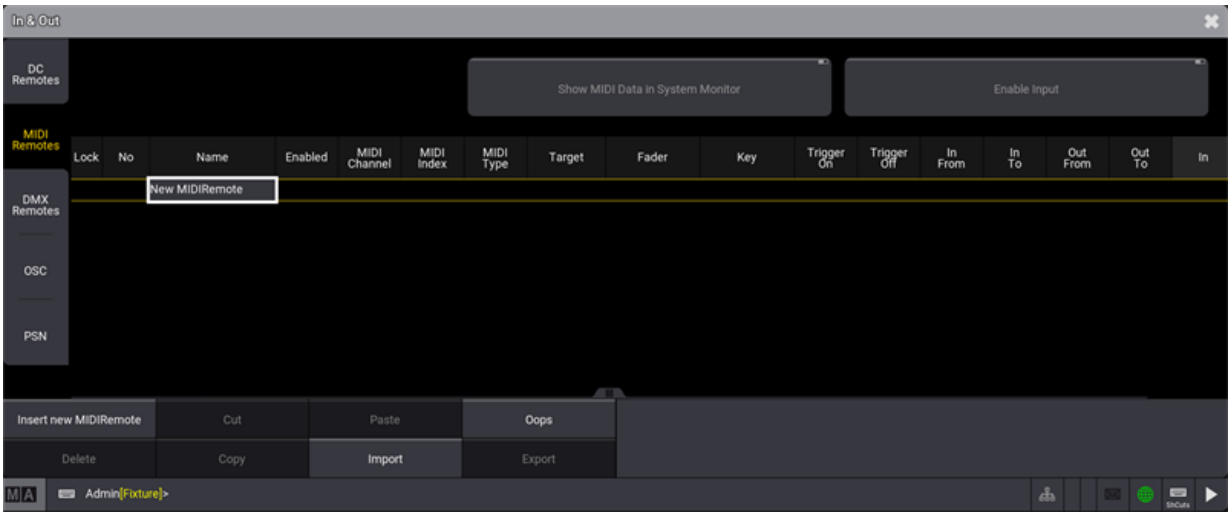
Use the command line to open the menu:

```
User name[Fixture]> Menu "OutputConfig"
```

- To set the MIDI Offset to a new value, tap MIDI Offset.
  The MIDI Offset pop-up opens:

MIDI Offset pop-up

- To adjust the settings of the MIDI Remotes, switch to the In & Out window.
- To open the **In & Out** window, press Menu and then tap In & Out.
- Tap MIDI Remotes.
- The MIDI settings window opens.



MIDI settings window

To learn more about the general DC Remotes settings like Name, Enabled, Fader, or Key read the **Remote In and Out topic**.

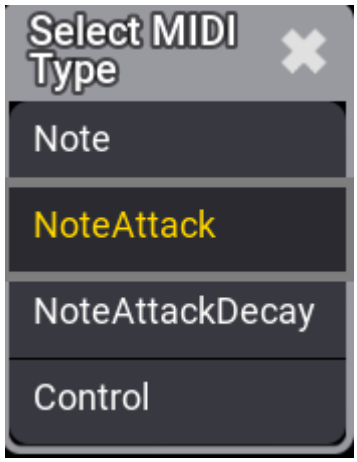
- To set the MIDI Channel value to a new value, tap MIDI Channel.
  The Edit MIDI Channel pop-up opens:

The MIDI channel value features 16 channels that can be controlled individually. The MIDI receiver must use the same MIDI channel as the MIDI sender to understand each other.

- To set the MIDI Index value to a new value, tap MIDI Index.
  The Edit MIDI Index pop-up opens:

The MIDI index value is the MIDI note or the MIDI control change number, depending on the set MIDI type. The MIDI receiver must use the same MIDI index value as the MIDI sender to understand each other.

- To set the MIDI Type value to a new value, tap MIDI Type.
  The Select MIDI Type pop-up opens:



Note: only the MIDI note is analysed.

Note Attack: in addition to the MIDI note, the attack of the Note will be analyzed. The higher the attack, the more a defined fader can be moved.

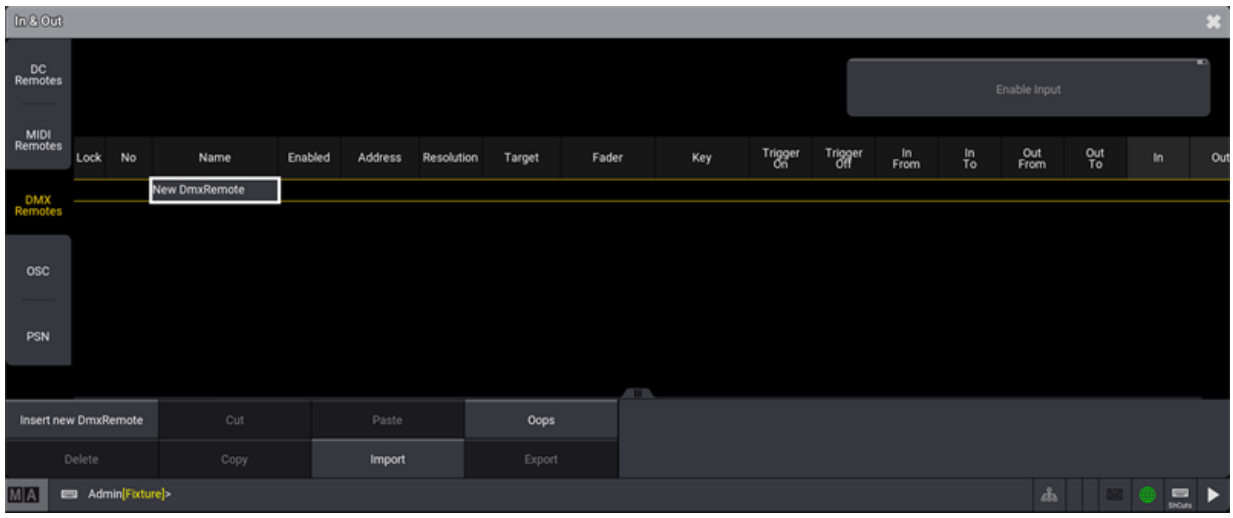
Note Attack Decay: together with the MIDI note, the Attack and Decay of the MIDI Note will be used.

37.3. DMX Remotes

The **DMX Remotes** tab uses DMX channels as remote triggers. The DMX source can also be the console itself.

To learn more about the hardware part of the input, read the **Connect DMX topic**.

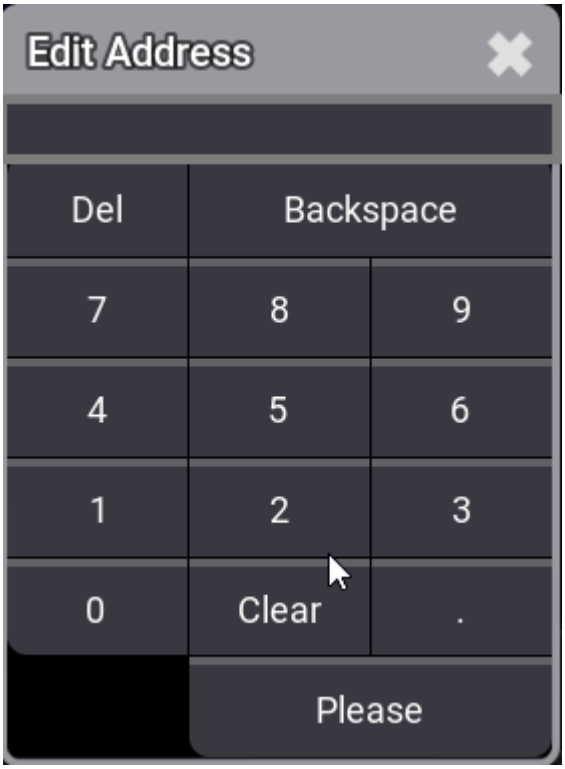
- To adjust the settings of the DMX Remotes, tap DMX Remotes.
  The DMX settings window opens.



DMX settings window

To learn more about the general DC Remotes settings like Name, Enabled, Fader, or Key read the **Remote In and Out topic**.

- To set the DMX address the DMX Remote entry should listen to, tap Address.



DMX address window

- To adjust the DMX resolution, tap Resolution.

DMX resolution window

The DMX Remote can be controlled by 16 bit or 24 bit DMX channels to be able to control it more precisely.

Hint:
For 16 bit and 24 bit, the additional one or two DMX channels need to be patched right behind each other at the controlling DMX device.

37.4. OSC (Open Sound Control)

OSC is a client and server system that defines a message address pattern used to address elements in the receiving server.

The grandMA3 software supports OSC 1.1.

Open Sound Control, or OSC, is a networking protocol used to allow devices of various types to control other devices of other types. OSC messages are human-readable (unlike, for example, MIDI Show Control, or MSC), and follow this general pattern:

"(/prefix)/[OSC Address],[OSC Type],[Value]"

prefix - this is optional, depending on your system setup. It can be used in a more complex OSC network to differentiate messages meant for one set of devices (e.g. lighting consoles) and not others (e.g. sound consoles).

OSC Address - this is the target you are controlling on the receiving device(s), for example /Fader201 would be the address to move the fader for executor 201 in grandMA3. Sometimes the address will be more complex, for example /Page1/Fader201 would be the address to move the fader for executor 201 on page 1 in grandMA3.

OSC Type - this is the type of value you're sending, for example:

i = integer

f = float

s = string

t = true

f = false

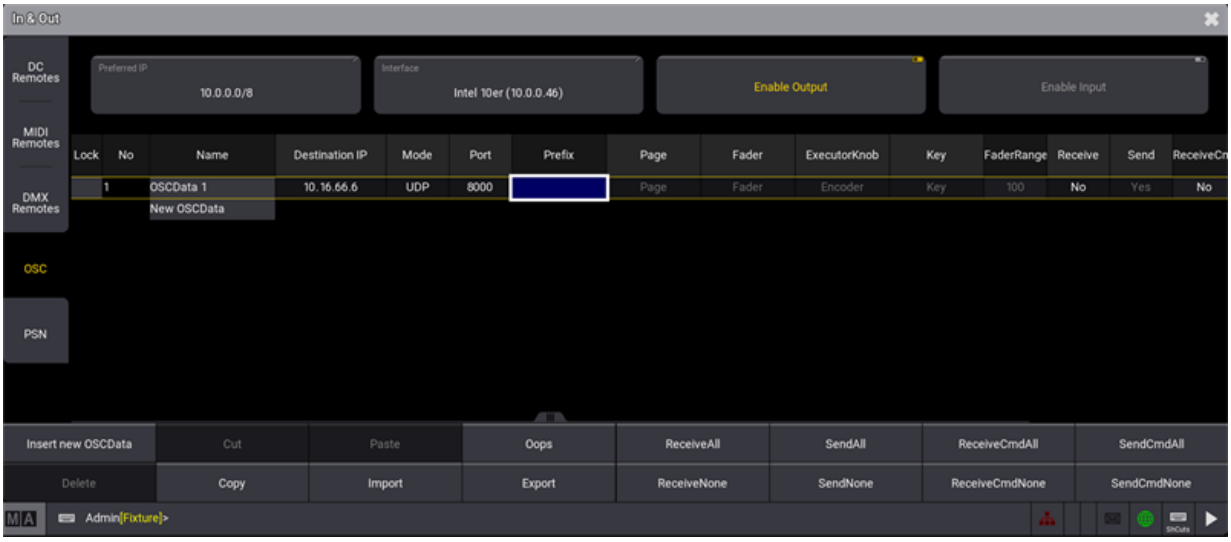
Value - this is the value you are actually sending for the target.

An example OSC command to set the fader for executor 201 to 100 might be:

- "/Page1/Fader201,i,100"
- or with a prefix to specify only, e.g. grandMA3 devices: "/gma3/Page1/Fader201,i,100"

For more information about the OSC address, OSC type, and OSC structure, see **SendOSC keyword**.

- To adjust the settings of OSC, tap OSC in the In & Out window.
  The OSC settings window opens.

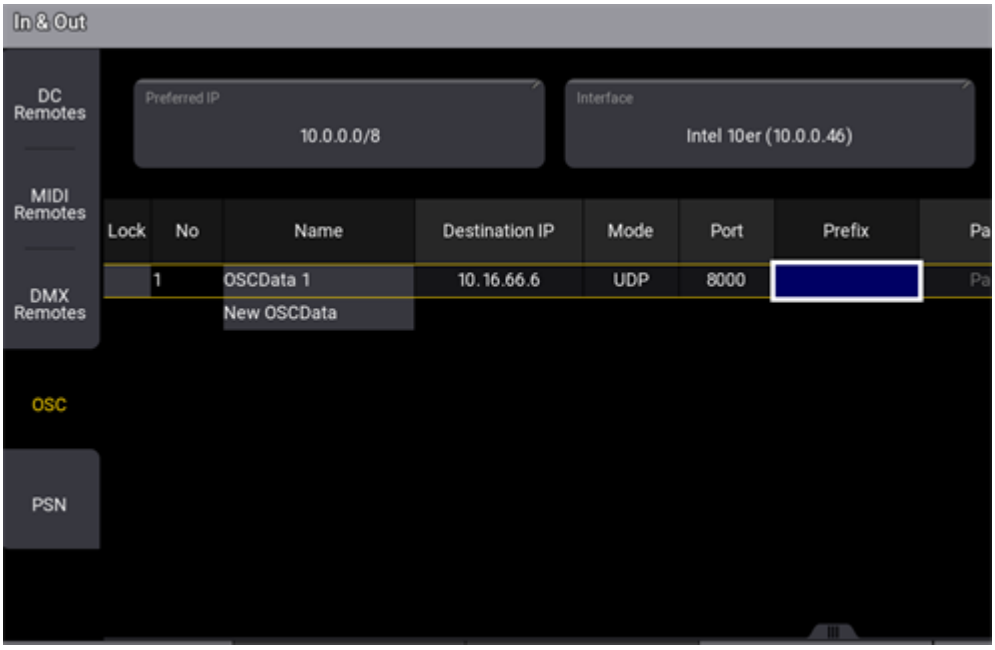


OSC settings window

Hint:
Please note that in the example above, no prefix is defined.

The most important OSC setting is the correct network configuration.

Make sure that the IP address, the network protocol (UDP, TCP), and the port are set correctly.



OSC network configuration

Hint:
Please note that the port configuration is used for sending and receiving OSC data.

When receiving OSC messages, Input will highlight its title bar. When sending OSC messages, the title bar of Output will be highlighted.

Each configuration line for OSC can be used for input and/or output.

These properties can be configured:

- Name: Sets the name for this configuration.
- Destination IP: Sets the IP address for sending OSC data. A specific IP address or a broadcast IP can be set.
- Mode: OSC packets can be sent via UDP or TCP.
- Port: Specifies the network port of the incoming and/or outgoing OSC packets.
- Prefix: A prefix can be set by the user if he needs to. A prefix can be used for example as a criterion for limiting the range of possible receivers, e.g. /lighting would only take packets with /lighting into account, and discard OSC packets with the /sound-prefix.
- Page: Specifies which OSC Address of incoming OSC messages is routed to pages.
- Fader: Specifies which OSC Address of incoming OSC messages is routed to faders.
- ExecutorKnob: Specifies which OSC Address of incoming OSC messages is routed to the mini encoders.
- Key: Specifies which OSC Address of incoming OSC messages is routed to keys.
- FaderRange: Specifies which OSC value range is used for the fader, e.g. FaderRange 255 sets OSC 0-255 to 100%.
- Receive: Specifies if OSC data (but no commands) shall be received.
- Send: Specifies if this OSC configuration sends OSC data (but no commands).
- ReceiveCmd: Specifies if commands for the command line will be received via OSC. This setting is independent of the general receive setting.
- SendCmd: Specifies if commands of the command line will be sent via OSC. This setting is independent of the general send setting.
- EchoInput: Specifies if the input data shall be displayed in the system monitor.
- EchoOutput: Specifies if the output data shall be displayed in the system monitor.

With the buttons
ReceiveAll, ReceiveNone, SendAll, SendNone, ReceiveCmdAll, ReceiveCmdNone, SendCmdAll, and SendCmdNone all OSC configuration lines can be modified together for the properties Receive, Send, ReceiveCMD and SendCmd.

Hint:
The addresses defined for Page, Prefix, Fader, ExecutorKnob, and Key are case-sensitive.

In addition to the example OSC strings mentioned above, the entirety of the grandMA3 command line can be accessed via OSC, using the "/cmd" OSC Address and the string 's' OSC Type. *Note - this requires "ReceiveCmd" to be set to Yes.* For example:

- "/cmd,s,FaderMaster Page 1.201 At 100"
  - Bring fader 201 on page 1 to 100% (same as the examples above but using gMA3 command line syntax instead)
- "/cmd,s,Fixture 1 At 75"
  - Use gMA3 command line syntax to execute the command "Fixture 1 At 75" in the command line
- "/cmd,s,Go+ Exec 402"
  - Trigger whatever is assigned to exec 402
- "/cmd,s,Patch Fixture 1 3.42"
  - Patch fixture 1 to address 42 in universe 3

Examples for using OSC

Common programs to trigger grandMA3:

TouchOSC

TouchOSC is a modular OSC and MIDI control surface for Windows, macOS, and Android by **hexler.net**.

It supports sending and receiving Open Sound Control and MIDI messages over Wi-Fi and CoreMIDI inter-app communication and compatible hardware.

Fader

This example will control the fader on Executor 230 of Page 1:

Notes:

- Assumes the OSCData line on the console has a prefix of "gma3" configured. If the prefix is empty, this would just be /Page1/Fader230.
- Assumes the "Page" and "Fader" cells in the OSCData line on the console are set to "Page" and "Fader" respectively (this is the default).

Executor button

This example will press the button for Executor 230 of Page 1:

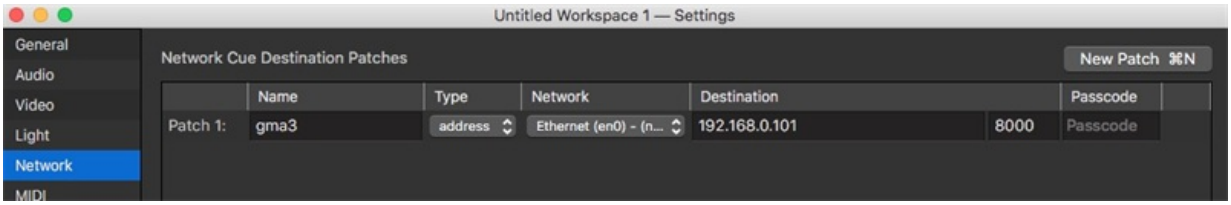
Notes:

- Assumes the OSCData line on the console has a prefix of "gma3" configured. If the prefix is empty, this would just be /Page1/Key230.
- Assumes the "Page" and "Key" cells in the OSCData line on the console are set to "Page" and "Key" respectively (this is the default).
- The {Send on Press} and {Send on Release} settings (not pictured above) should both be enabled/checked.

QLab

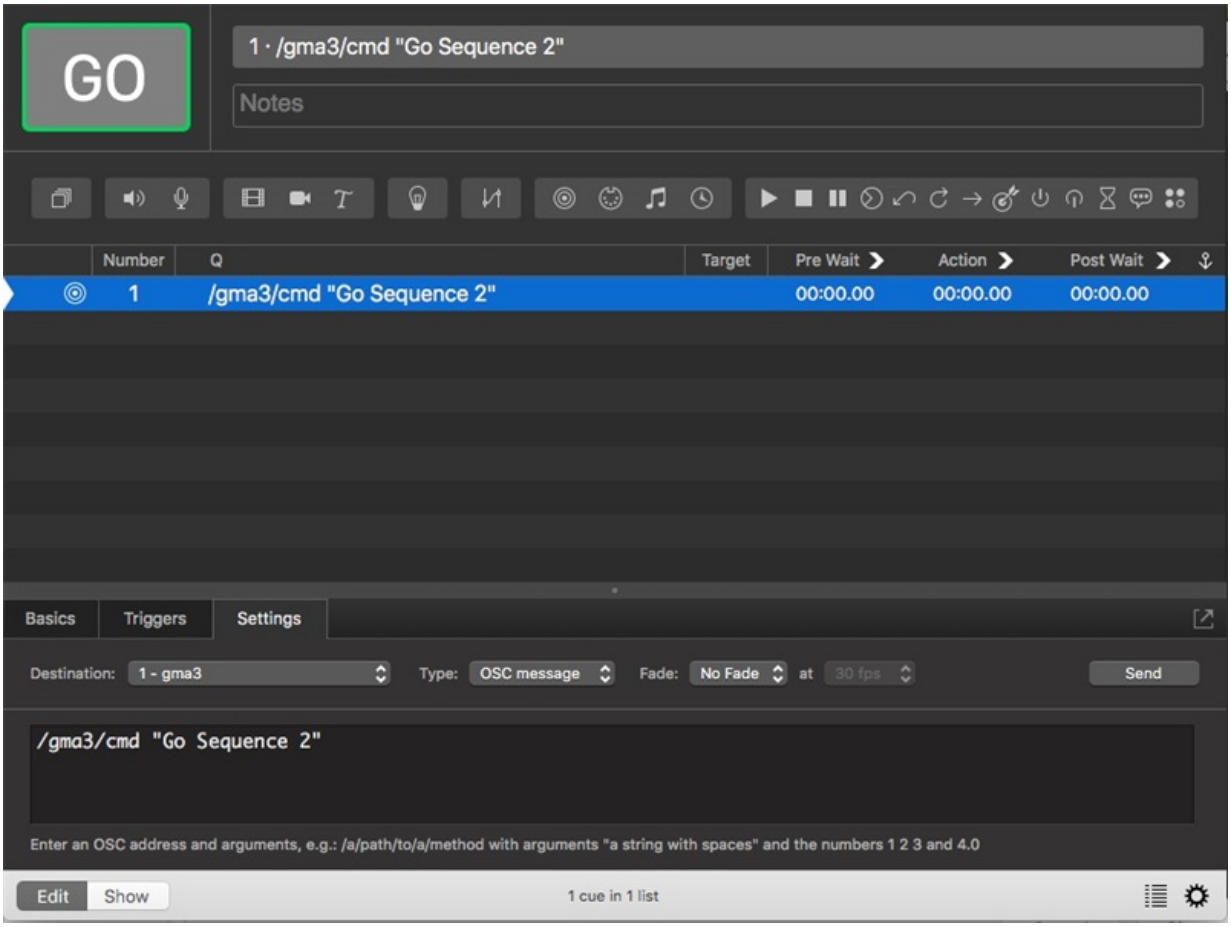
QLab is sound, video, and lighting control for macOS by **qlab.app**.

QLab is fairly simple to use with OSC. These are the QLab network settings in our example:



- Name = something to identify this particular configuration
- Network = the network interface on your computer connected to the grandMA3 system
- Destination
  - IP address of the grandMA3 console
  - Port set in the OSCData line configuration in the console (8000 is default)

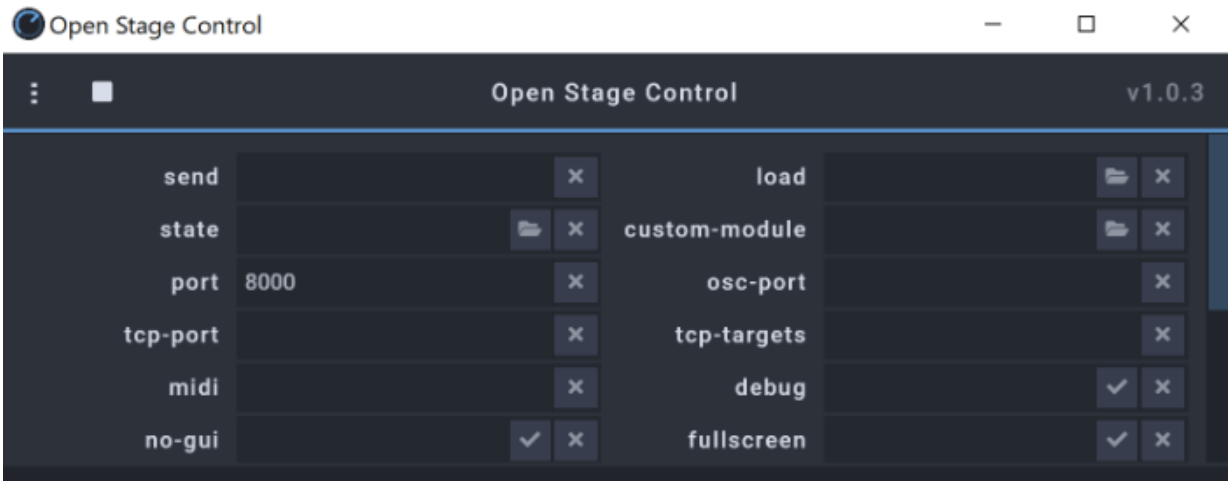
And then we have our cue setup in QLab:



- Destination = configuration as set in the "Network Cue Destination Patches" above
- Type = OSC message
- Enter the desired OSC message
  - OSCAddress - in this example we are sending a command syntax string directly, so the address is: /gma3/cmd
    - This assumes the OSCData line on the console has a prefix of "gma3" configured! If the prefix is empty, this would just be /cmd
    - *Requires "ReceiveCmd" to be enabled for that OSCData line on the console!*
  - Argument - enter the command syntax in quotes, e.g. "Go Sequence 2" (advancing to the next cue in Sequence 2)

Open Stage Control

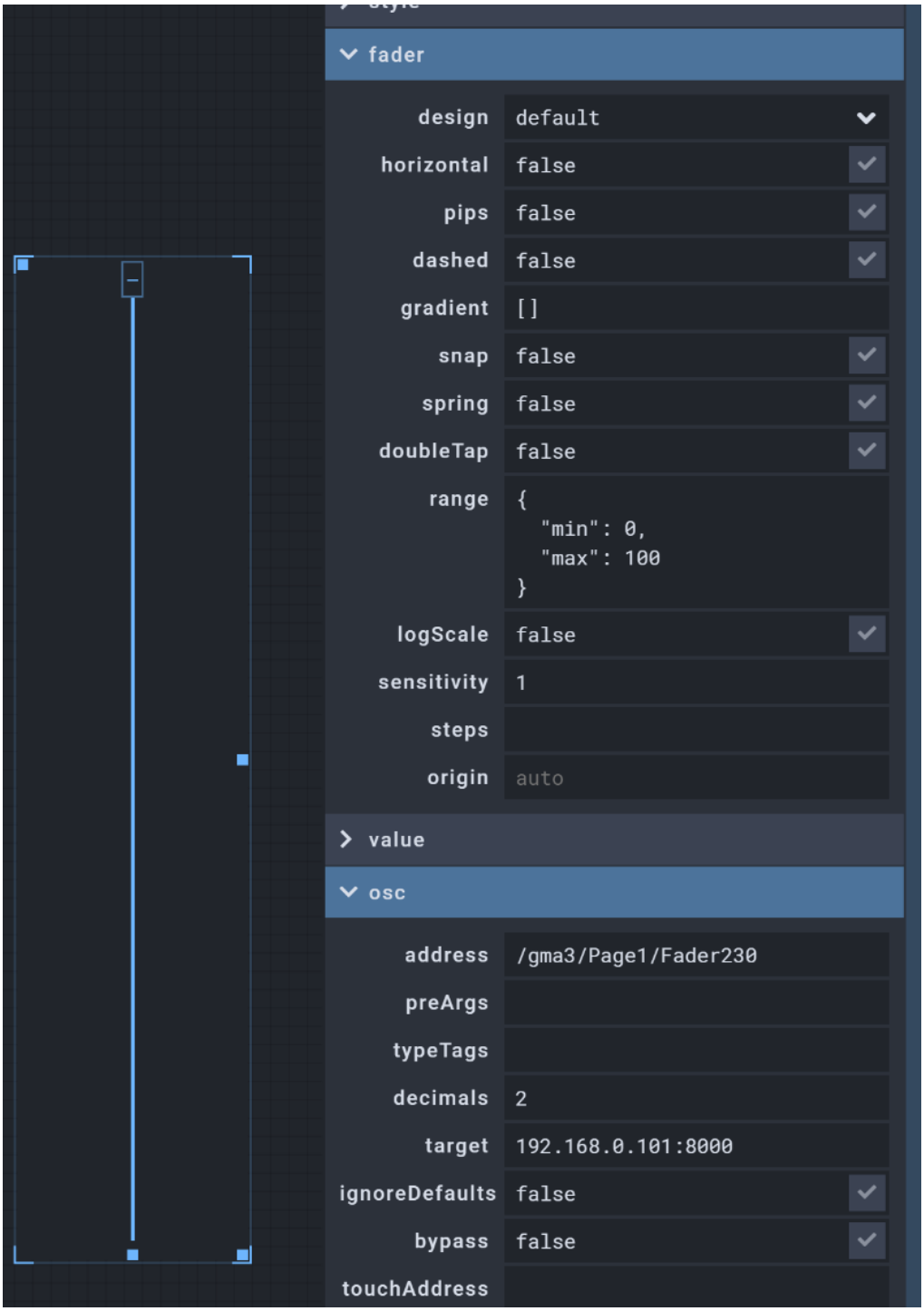
Open Stage Control is a free program you can use to build a simple OSC interface. When you first open it, you'll be prompted for some network configuration settings. Besides telling it which network interface to use on the computer, the only thing to fill in here is the port:



This needs to match the port you've set in the corresponding OSCData line in the console. Port 8000 is the default. Then you can start your Open Stage Control session.

Fader

This example will control the fader for Executor 230 on Page 1:

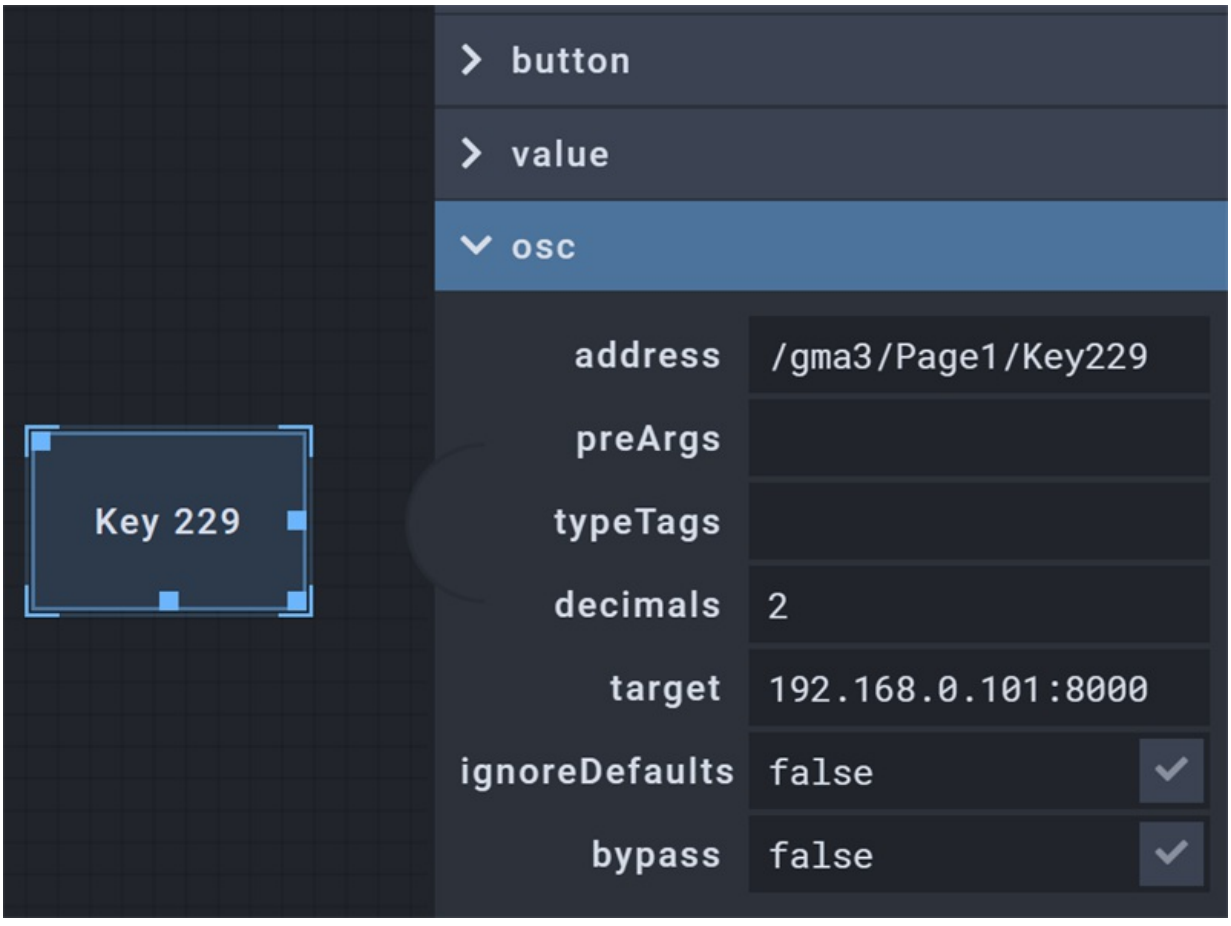


Notes:

- Assumes the OSCData line on the console has a prefix of "gma3" configured. If the prefix is empty, this would just be /Page1/Fader230.
- Assumes the "Page" and "Fader" cells on the OSCData line in the console are set to "Page" and "Fader" respectively (this is the default).
- All of the settings in the picture above are at their defaults except for:
  - Fader settings: Range: change the 'max' to 100 instead of 1
  - OSC settings: Address

Executor button

This example will press the button for Executor 229 on Page 1:



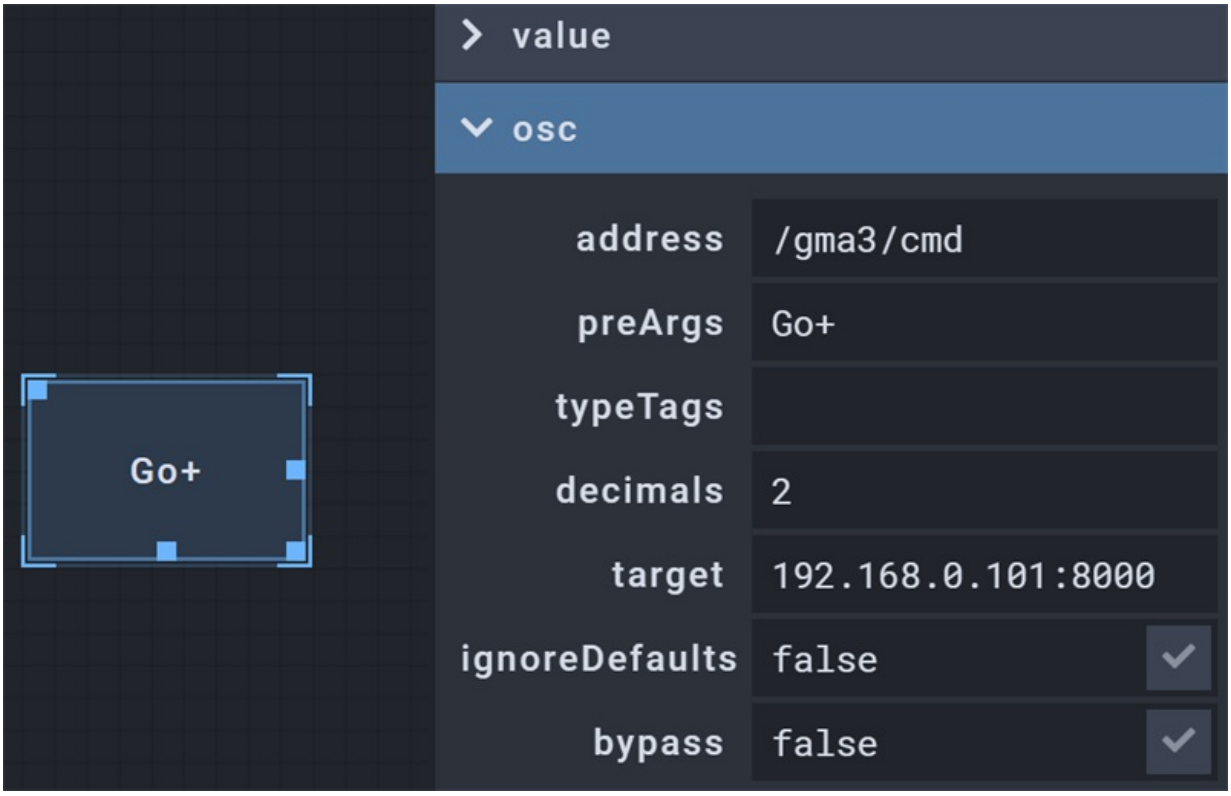
Notes:

- Assumes the OSCData line on the console has a prefix of "gma3" configured. If the prefix is empty, this would just be /Page1/Key229.

- Assumes the "Page" and "Key" cells in the OSCData line on the console are set to "Page" and "Key" respectively (this is the default).
- All of the settings in the picture above are at their defaults except for the address in the OSC settings
- Open Stage Control buttons default to functioning as 'toggle' - you may wish to change this to 'tap'

Command Line Syntax

Here we have a button that will execute command line syntax on the console, in this case triggering the Selected Sequence:



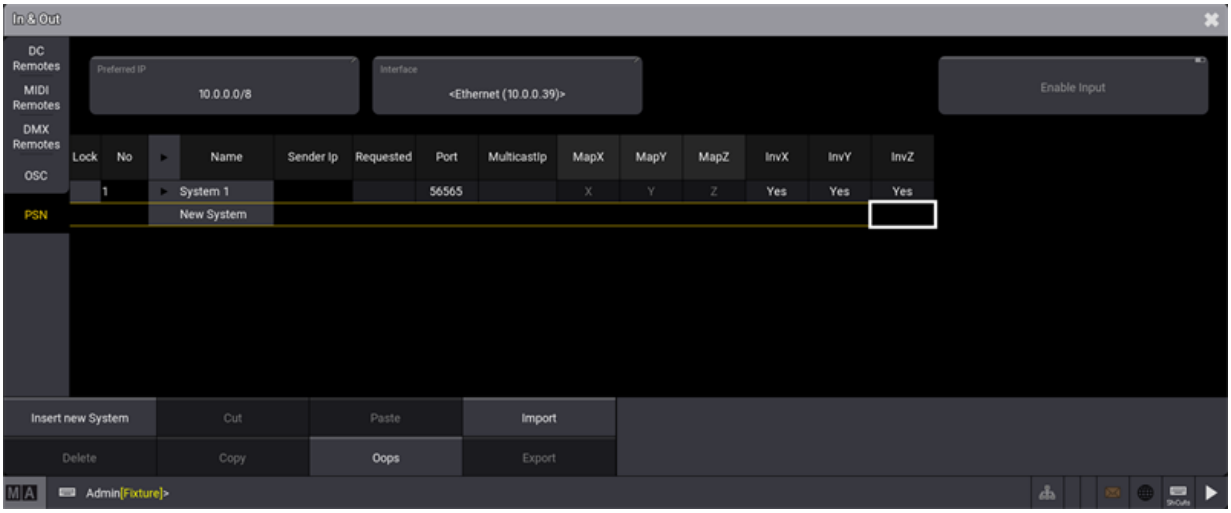
Notes:

- Assumes the OSCData line on the console has a prefix of "gma3" configured. If the prefix is empty, this would just be /cmd.
- *Requires "ReceiveCmd" to be enabled for that OSCData line on the console!*
- All of the settings in the picture above are at their defaults except for:
  - address
  - preArgs - this is where you enter the syntax string you wish to execute
- Open Stage Control buttons default to functioning as 'toggle' - you may wish to change this to 'tap'

37.5. PSN (Posi Stage Net)

grandMA3 stations can receive PosiStageNet (PSN) data.

- To adjust the settings of PSN, tap PSN.
  The PSN settings window opens.



PosiStageNet (PSN) window

When a PSN source is available in the network, adding a new PSN System by tapping Insert new System is required.

Depending on the type of transmitting the data, the user may have to make some settings first: In case of sending the PSN data to a multicast address, the user has to manually add the multicast IP of the PSN system. In most cases, this may be IP 236.10.10.10.
If the sender is transmitting the data via unicast to the selected interface in the PSN menu, the PSN system automatically detects the data.

For each PSN system that is received, the user can map each axis to a different axis and/or invert the incoming data per axis. To do so, tap and hold the cell of an axis or of an invert option. In the case of mapping, a drop-down opens to choose the target axis. Invert just toggles the cell between No (=empty cell) and Yes (= data will be inverted).

As soon as an axis is mapped or inverted, the individual trackers display their values according to the settings made by the user.

38. Remote Device Management (RDM)

Remote Device Management is a protocol that allows bi-directional communication between a grandMA3 device and RDM-ready devices attached to it (= RDM-ready fixtures) over a standard DMX line. ANSI E1.20 - 2010 by PLASA specifies the RDM standard as an extension of the DMX 512-A protocol (ANSI E1.11).

Manual settings, such as adjusting the DMX starting address, are no longer needed. This is especially useful for devices installed in a remote area.

RDM is integrated in DMX without influencing the connections. The RDM data is transmitted via the standard XLR-poles – new DMX cables are not required. RDM-ready and conventional DMX devices can be operated in one DMX line. The RDM protocol sends its own data packages in the DMX512 data feed and does not influence conventional devices.

To be able to use RDM, RDM has to be enabled at two different spots:

1. Globally within the show file. To do so, tap RDM in Menu - Network Menu, or tap RDM in Menu - Live Patch - RDM.
2. Per XLR port that shall use RDM. Therefore, the mode of an XLR port in the Output Configuration needs to be set from Out to RDM.

In RDM mode, DMX data is only sent when there are changes of DMX values. In addition, every 500ms a refreshing packet will be sent so that DMX fixtures will not switch into DMX fail mode.
This RDM output mode allows more time on the DMX line for RDM configuration.

The new RDM Devices window lists all devices that are discovered via RDM. It can be opened via the Add Window dialog and is located in the More-tab.
The same list is also displayed in Menu - Live Patch - RDM.

When an RDM fixture is detected on an XLR port, a new node called RDMPort will be added to the RDM devices list. Within each RDMPort node, all fixtures that are detected via RDM on this physical XLR port are listed. An RDM port is labeled with the IP of the device and the XLR port of the device, e.g., 192.168.0.4 - XLR D. If the device is not available anymore with the port, the font color turns red.
For each different fixture type, detected per RDM, a new node in the RDMFixtureTypes node in the RDM Devices window will be created. Each RDMFixtureType contains general information of the RDM fixtures, that is similar to all fixtures of the same product, e.g., the parameter description, or the available DMX personalities.

The grandMA3 creates RDMFixtureTypes by itself depending on these three parameters: ManufacturerID, DeviceModelID, and SoftwareID.
As soon as the same physical type of lighting fixtures have different software IDs due to different firmware versions, different RDMFixtureTypes are created.

This first implementation of RDM supports these RDM parameters, which can be set by the user:

- IDENTIFY_DEVICE, 0x1000

- DEVICE_LABEL, 0x0082
- DMX_PERSONALITY, 0x00E0
- DMX_START_ADDRESS, 0x00F0
- PAN_INVERT, 0x0600
- TILT_INVERT, 0x0601
- DISPLAY_INVERT 0x0500
- DISPLAY_LEVEL 0x0501
- RESET_DEVICE 0x1001
- FACTORY_DEFAULTS 0x0090
- LAMP_STATE 0x0403

Only the cells of properties that an RDM fixture provides as set-able can be edited in the RDM devices window.

In grandMA3 the RDM communication follows this process:

1. Discovery for new RDM fixtures.
    1. Check if detected fixtures are still available.
    2. Check for new RDM fixtures.
2. Get parameter and sensor data.
3. 1s Pause
4. Start again at 1.

Parameters that are not changing during the runtime of a fixture, e.g., Device Info, are only pulled via RDM once when creating the corresponding RDMFixtureType. All other parameters and sensors are pulled every time in step 2.
As soon as an RDM fixture is not available for 3 discoveries in a row, it will be displayed in red in the list of RDM fixtures.

It is possible to match an RDM fixture with a fixture of the grandMA3 patch. To do so, edit the fixture cell of the desired RDM fixture in the RDM devices window. A pop-up opens and offers all fixtures of the current show file.
In addition, it is also possible to match fixtures within the RDM window in the live patch. In the live patch, it is possible to open the fixture list in the same way as described above, and by selecting any cell of a fixture then tapping Match at the bottom of the window.
To remove a match between an RDM fixture and a grandMA3 fixture, it is possible to tap Unmatch in the RDM window in the live patch or to tap Clear in the match pop-up.

39. Control other MA Devices

This section describes how to control other MA devices via the console.

39.1. Control grandMA3 Nodes

To adjust the settings in the grandMA3 Nodes, it may be convenient to control them from a connected console or onPC.

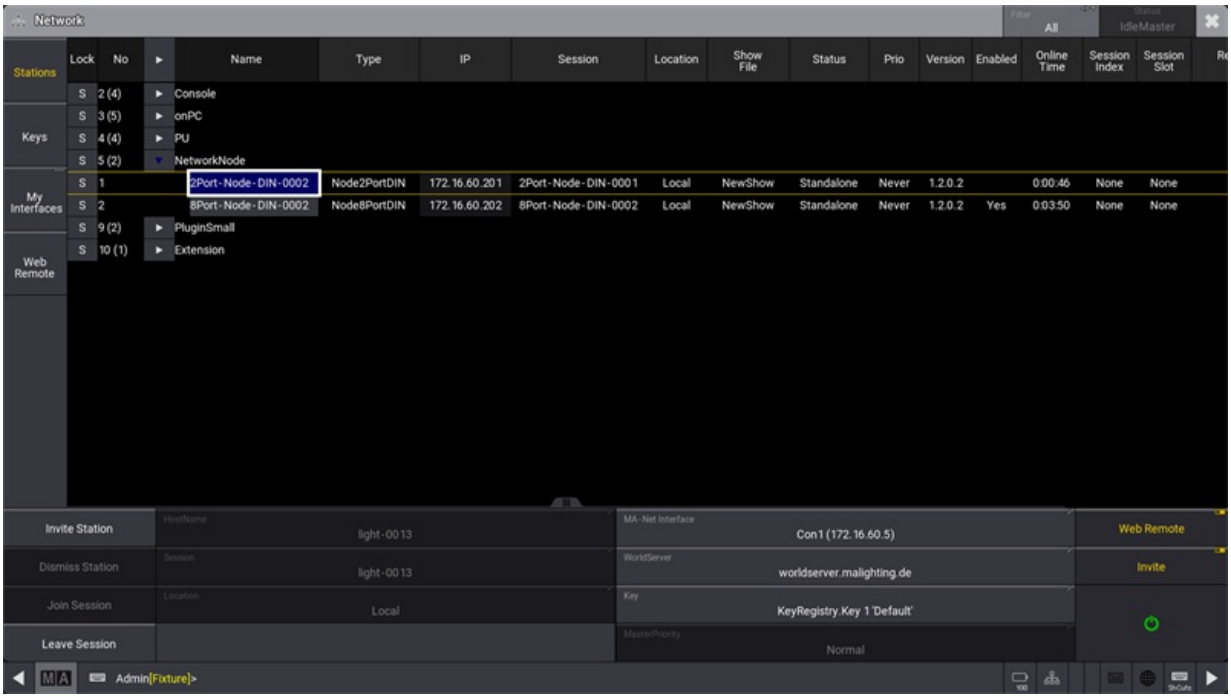
For more information, read the **Session topic** in the **Networking section**.

To connect a grandMA3 Node with the console, make sure that the Node is in grandMA3 mode.

Change Name and Set IP Address

To change the name and to set the IP address of a grandMA3 Node, open the Network Interface Menu:

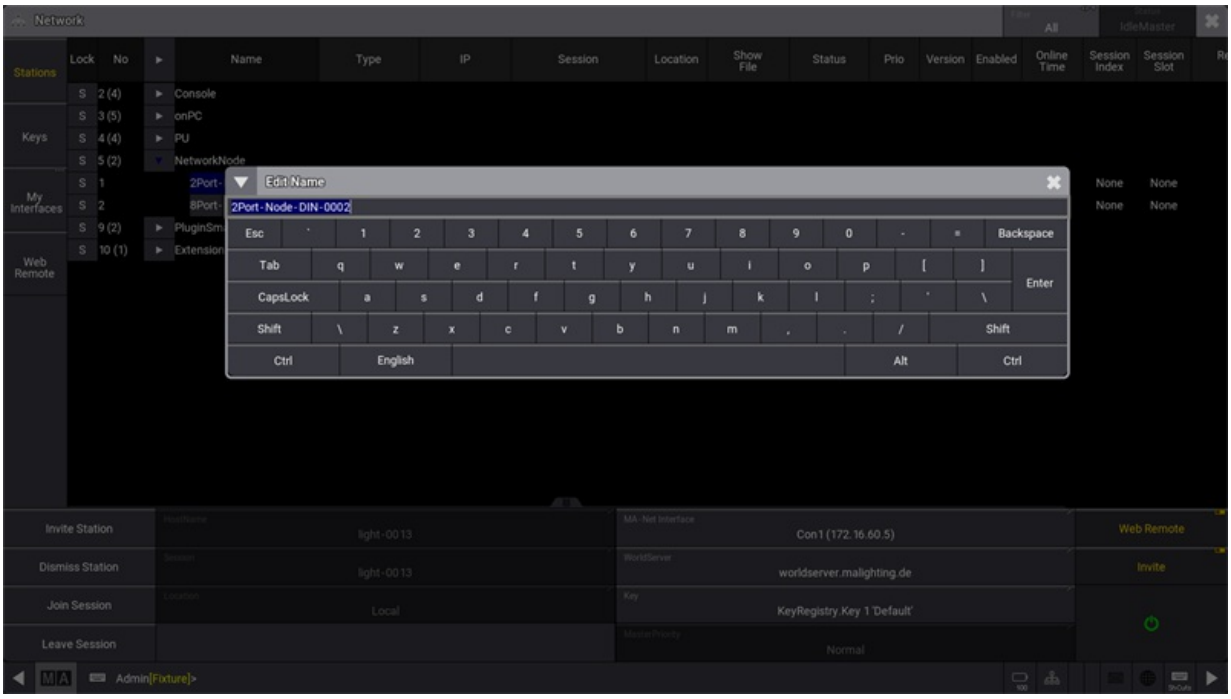
- Press Menu.
  - Opens the **menu select pop-up**.
- Tap Network.
  - Opens the Network menu.



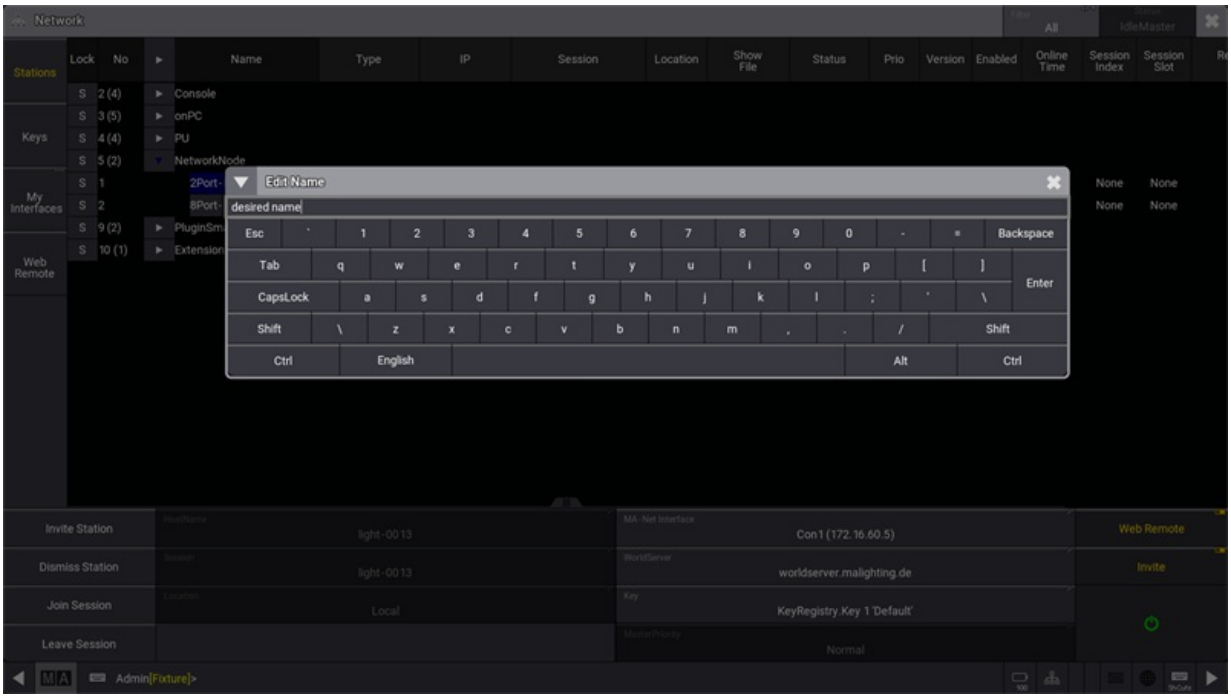
Important:
You can only change the name or IP address of a node if it is not in another session.

To Change the Name

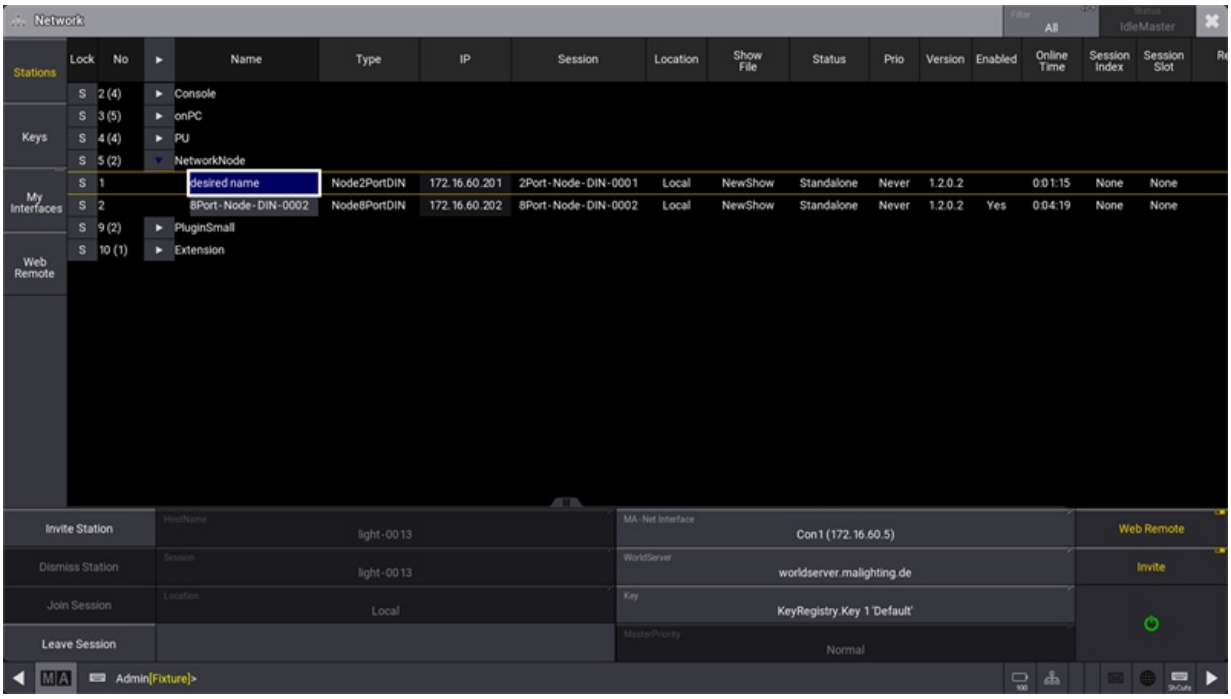
- Edit the name of the respective node by right-clicking with a mouse or by using **gestures**.



- Enter the desired name.

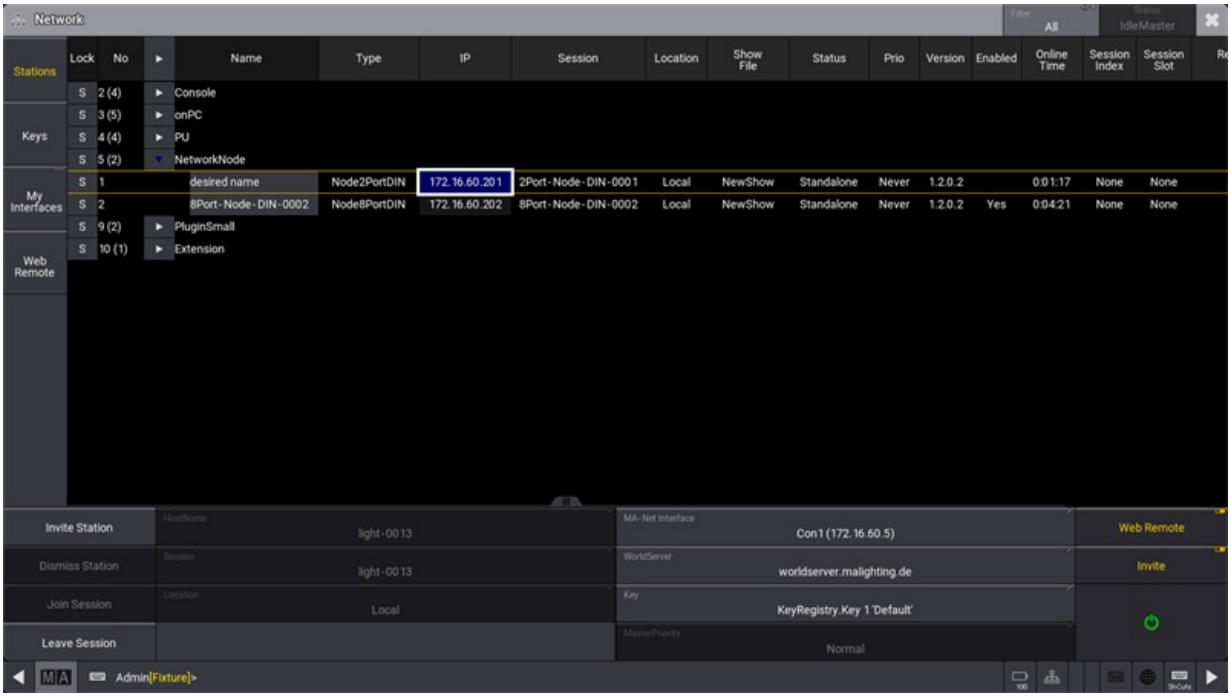


- Tap Apply Changes to confirm the changes.

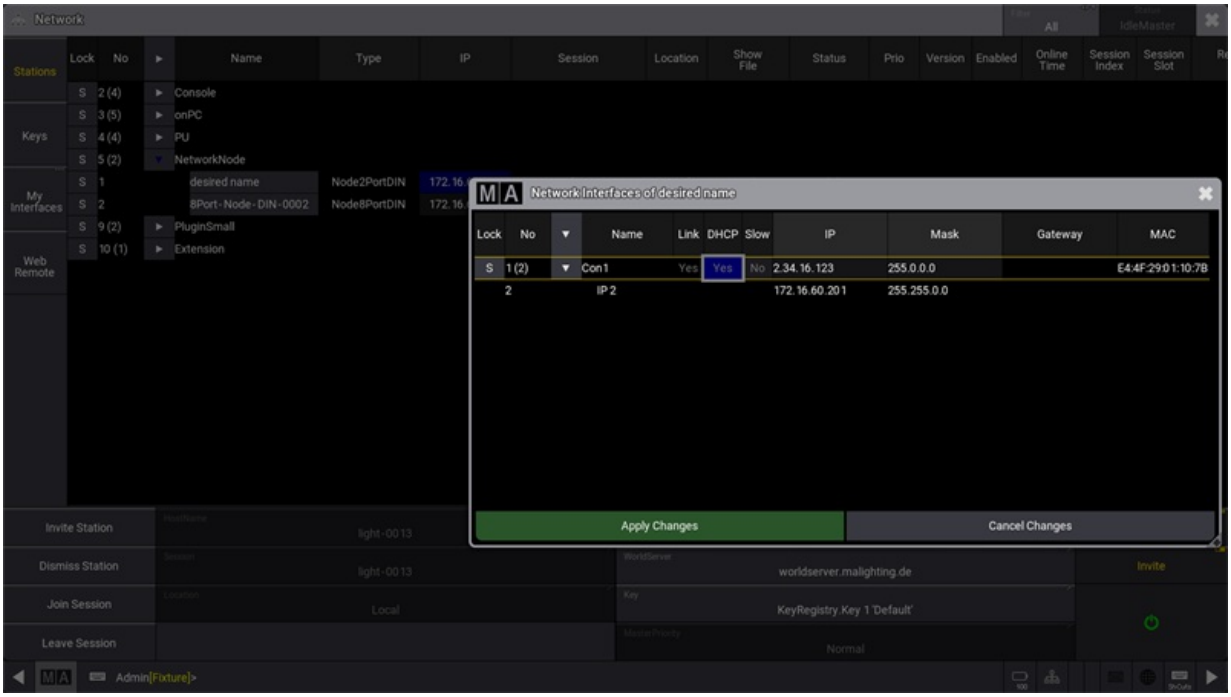


To Change the IP Address

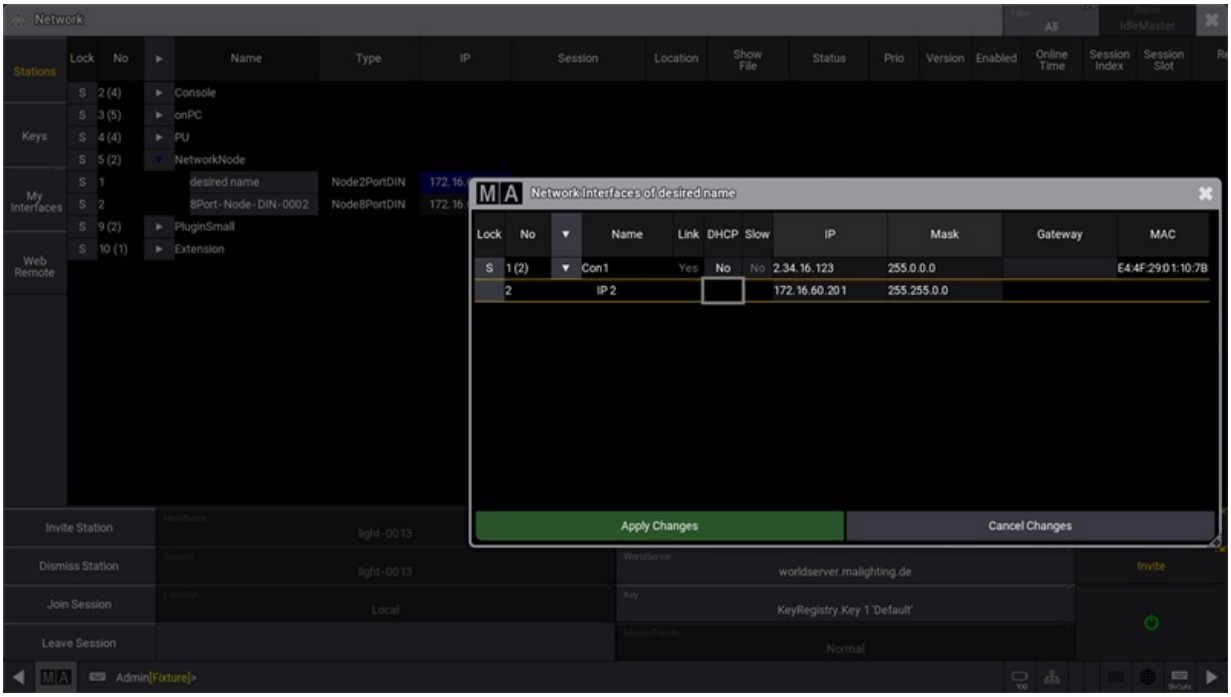
- Edit the IP address of the respective node by right-clicking with a mouse or by using **gestures**.



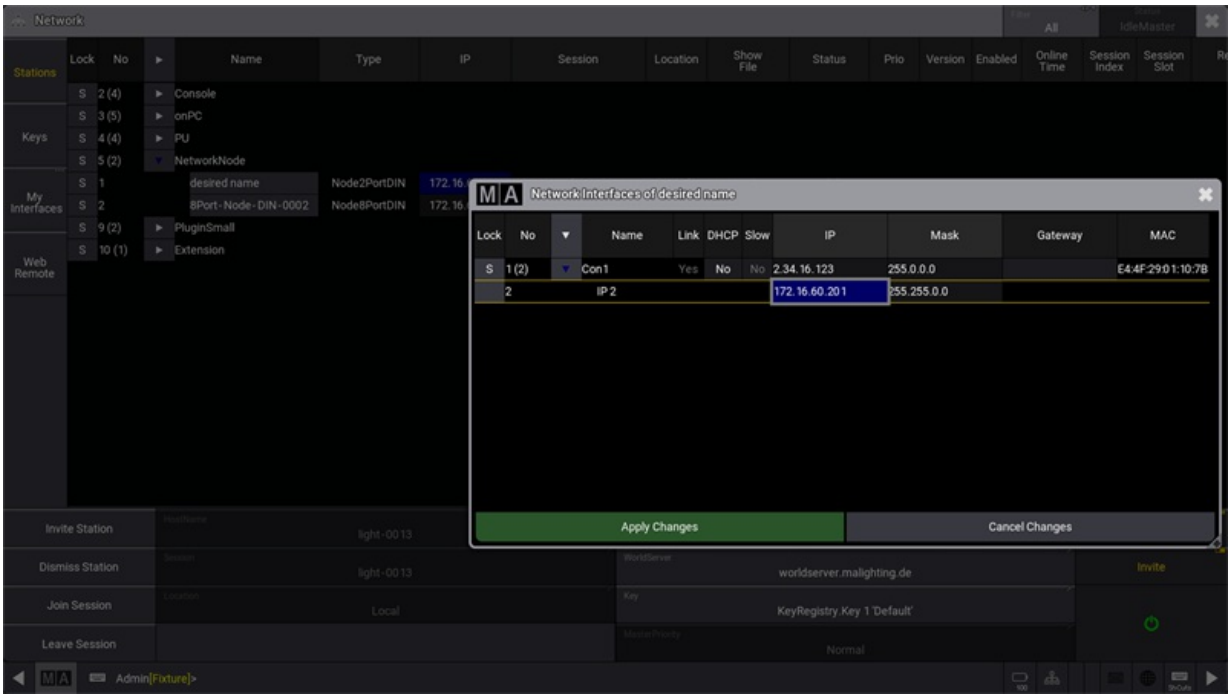
- Edit the DHCP settings by right-clicking the DHCP field with a mouse or by using **gestures**.



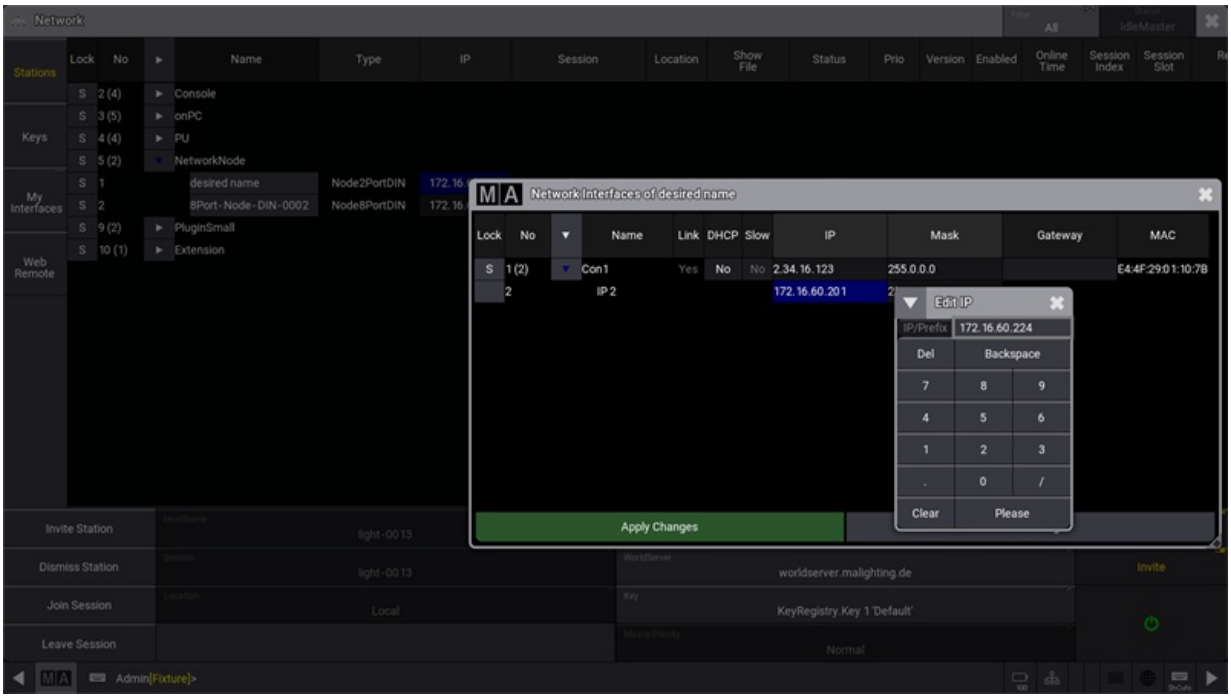
- Set DHCP to Yes or No.



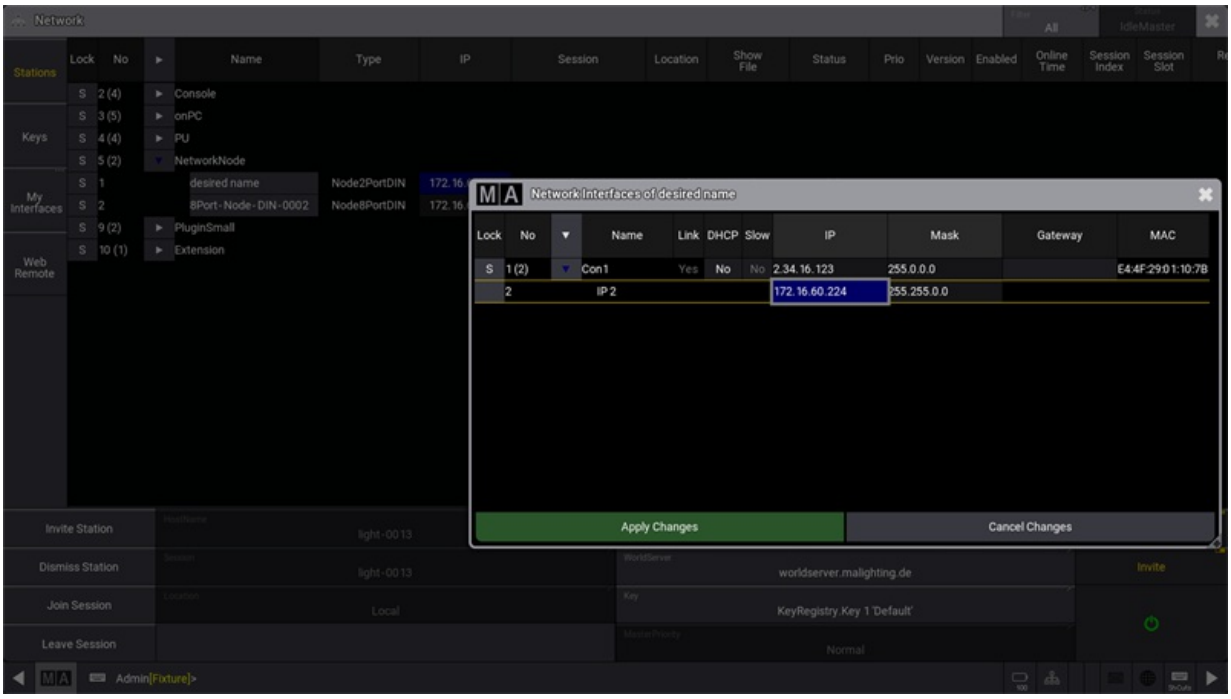
- Edit the IP address of the respective node by right-clicking with a mouse or by using **gestures**.



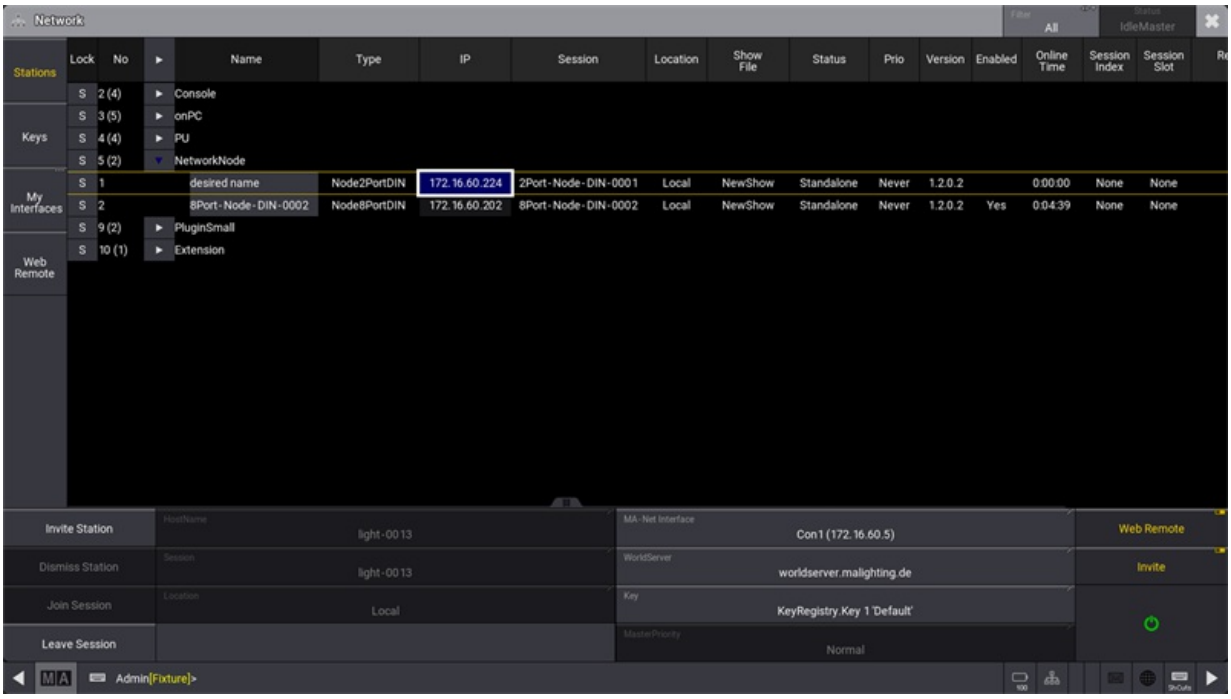
- Enter the new IP address (here 172.16.60.224).



- Tap Apply Changes to confirm the changes.



- The new IP address is now set.



For more information about the network menu, read the **Network Update topic** in the **Update the Software section**.

Hint:
You can also update the Nodes via **manual update**.

Change the Output Configuration

To adjust the output configuration settings in the grandMA3 Nodes, it may be convenient to control them from a connected console or onPC.

For more information, read the **DMX port configuration topic** in the **DMX In and Out section**.

To connect a grandMA3 Node with the console, make sure that the node is in grandMA3 mode.

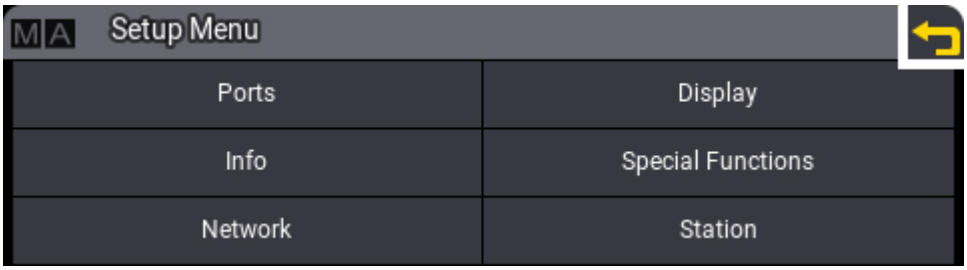
Change Modes

From grandMA3 to Mode2

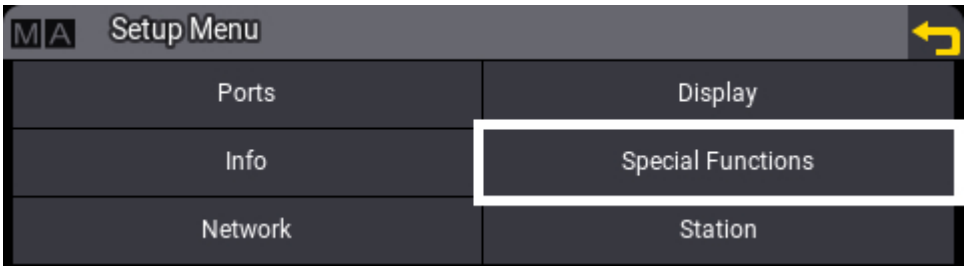
- To change the Node from grandMA3 to Mode2, follow these instructions:

grandMA3 screen:

- To switch to Mode2, press the rotary knob:

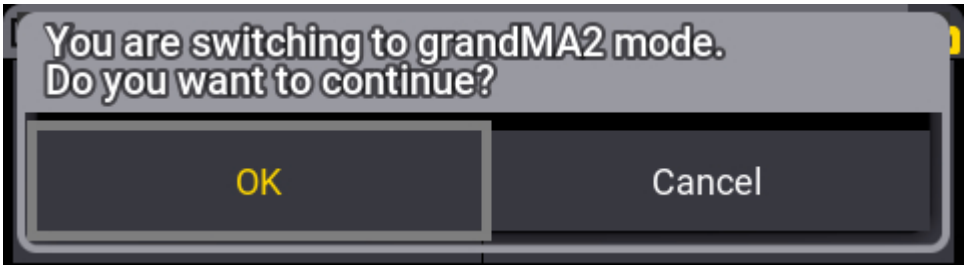


- Turn the rotary knob to select Special Functions and press the rotary knob to confirm:



- Select Switch to grandMA2 and press the rotary knob to confirm:

- Select OK and press the rotary knob to confirm:



The Node reboots into Mode2.

For more information about changing to grandMA3 Mode2, read the **Mode2 topic** in the section grandMA3 Mode2 of the **grandMA2 User Manual**.

39.2. Control an MA Network Switch

The MA Network Switch is the perfect device for a full lighting control solution. The combination of console, switch, and networking devices provides the easiest solution even for complex lighting systems.

You can control the MA Network Switch by using the web interface or directly with a grandMA3 console in Mode2 or a grandMA2 onPC. For more information about how to control the MA Network Switch by using the web interface see the **MA Network Switch manual**.

To learn how to control the MA Network Switch from a grandMA3 console in **Mode2** or a grandMA2 onPC, read the topic **Control the MA Network Switch** in the **grandMA2 User Manual**. There, you will find out how to add the MA Network Switch to the network configuration, configure ports, or edit groups and presets.

For more information about changing to grandMA3 Mode2, read the **Mode2 topic** in the section grandMA3 Mode2 of the **grandMA2 User Manual**.

39.3. RemoteHID

The Remote HID feature allows using the local HID devices (typically mouse and keyboard) to control other grandMA3 stations that are available in the network as if the local HID devices are connected to the remote device.

A use case for this feature is controlling onPC stations that are used for visualization right at a console. Instead of placing several mice and keyboards on the table, only one set of mouse and/or keyboard is needed.

Restriction:
It is not a remote desktop feature. The display of the remote device needs to be visible.

To control another station from the local console, follow these steps:

1. Enable RemoteHID in Menu - Network menu on the remote station.
2. Execute the RemoteHID command on the local station:

RemoteHID IP [remote_ip_address]

- Or -

RemoteHID [remote_station_type] ["remote_hostname"]

3. When the connection has been established, the screen of the local station changes to olive green. During the connection, only the executors including the 100mm faders and Go+[Large], Go-[Large], and Pause[Large] remain usable on the local station. All other elements are blocked.

The remote function can be ended using one of the following options:

- Press MA + MA + Off
- Use the keyboard shortcut Shift + Ctrl + Alt + E

Example

A preprogramming setup where there are a console and a powerful graphics computer running grandMA3 onPC optimized for 3D visualization. They are connected in a session on a local LAN network.

The grandMA3 onPCs hostname is "3D" and RemoteHID is enabled.

From the console, there is a need to change a setting in the 3D window on the grandMA3 onPC.

In the command line type:

Now the mouse and keyboard connected to the console can be used to operate the computer.

Finish the connection by pressing MA + MA + Off

Important:
grandMA3 onPC on Windows controlled via RemoteHID, is not only limited to control the grandMA3 onPC application. Furthermore, the mouse and keyboard can also access remotely the entire system.

Restriction:
RemoteHID can only be used on grandMA3 consoles and grandMA3 onPC on Windows.

Restriction:
At the moment, it is not possible to use the touch displays of grandMA3 consoles to control the mouse remotely.

40. Update the Software

Every grandMA3 device is delivered with the latest version of the grandMA3 software.

All devices in a network with software versions higher than 1.0 can see and update each other.

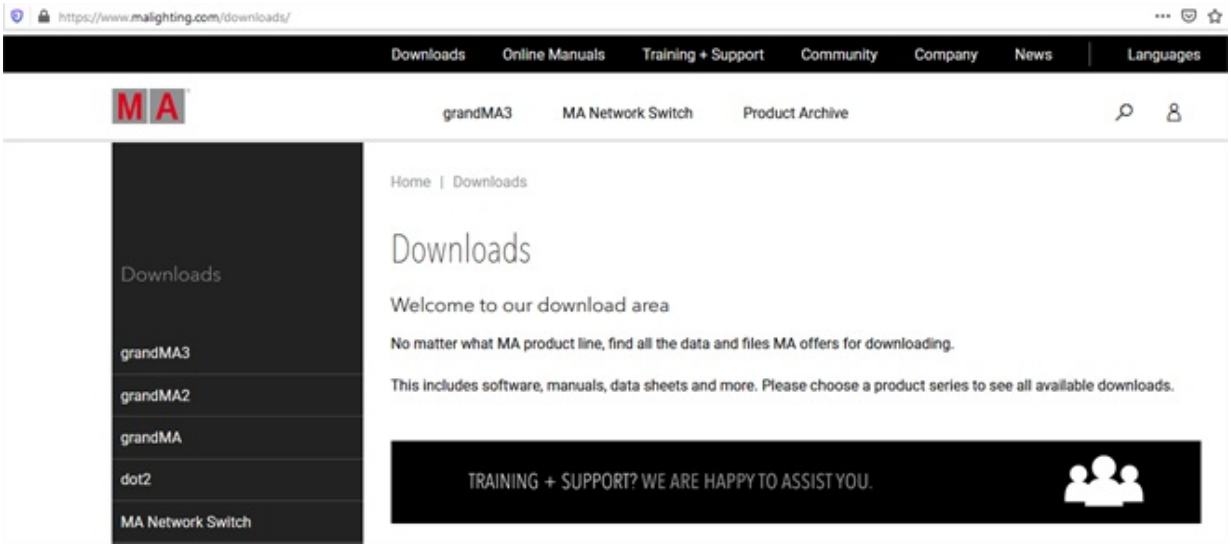
This is to give you information on how to update or downdate the grandMA3 software if needed.

To check which version your device is currently running, use the **Version keyword** in the command line (does not work with PUs and Nodes).

Check for Updates and Download the Latest Version

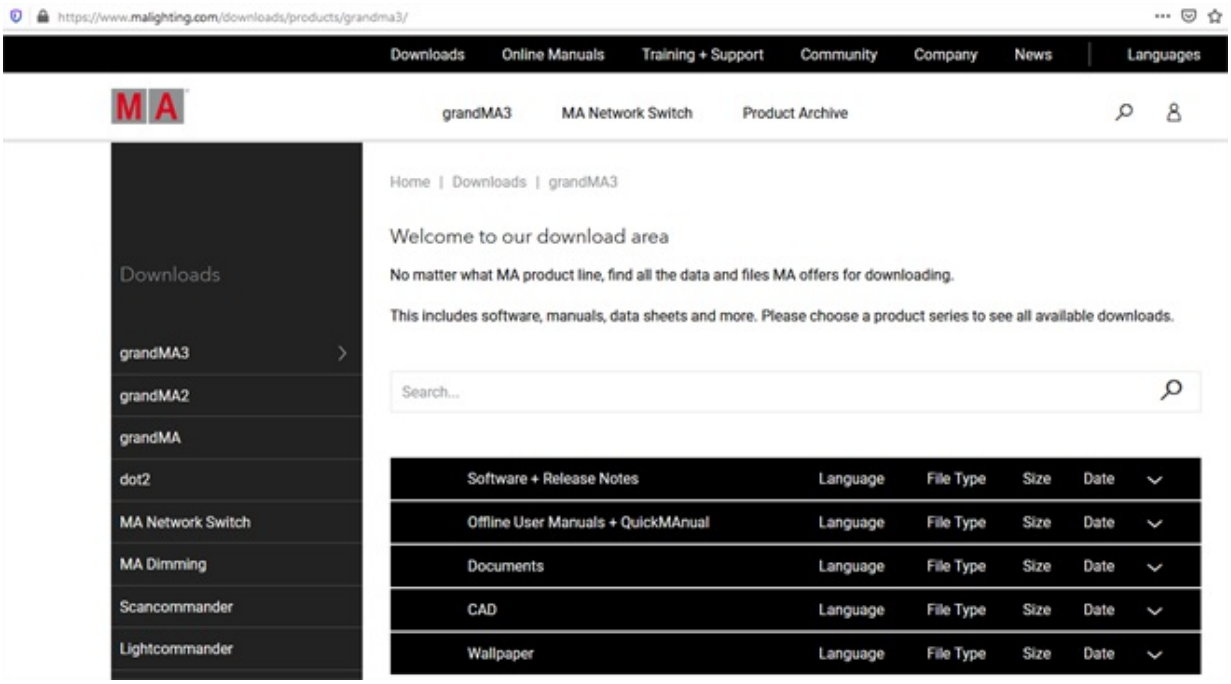
To check if there is a new grandMA3 software update:

1. Go to **www.malighting.com**, click **Downloads.**



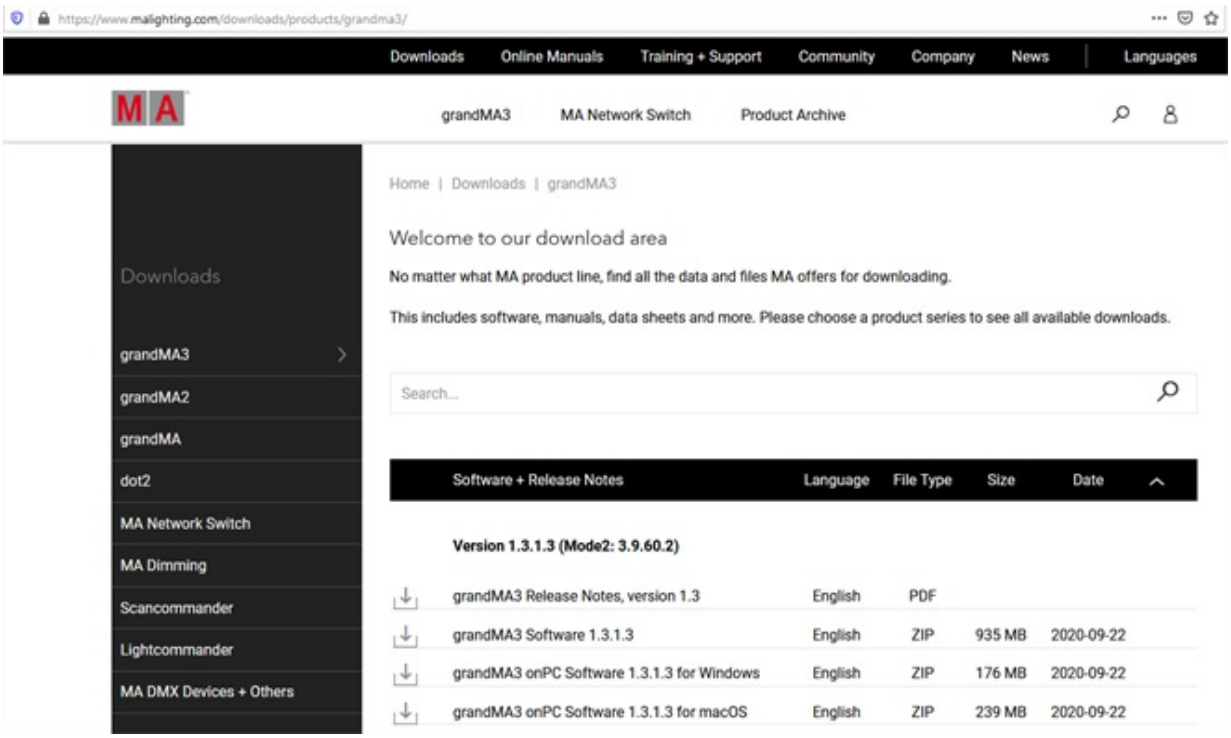
Website malighting.com Downloads

2. Click **grandMA3** in the bar on the left.



grandMA3 section

3. Click **Software + Release Notes** to find the latest software version.



grandMA3 Software section

4. Click the current version to download the desired installation package.

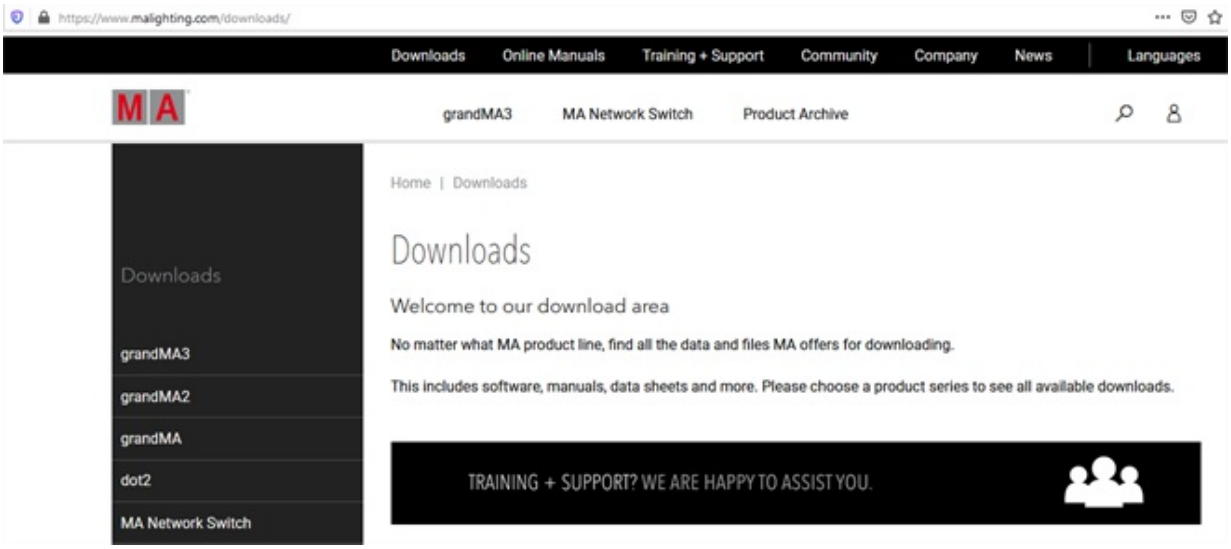
Important:
Every grandMA3 device except with Windows or macOS operating system uses the same software and needs the main installation package grandMA3 Software.
To update the grandMA3 software on a PC or laptop, choose the installation package grandMA3 onPC Software for Windows or grandMA3 Software for macOS.
For network updates via PC or laptop, you also need to download the installation package grandMA3 onPC Software for the connected grandMA3 devices.

The download process starts.

Download an Older Version

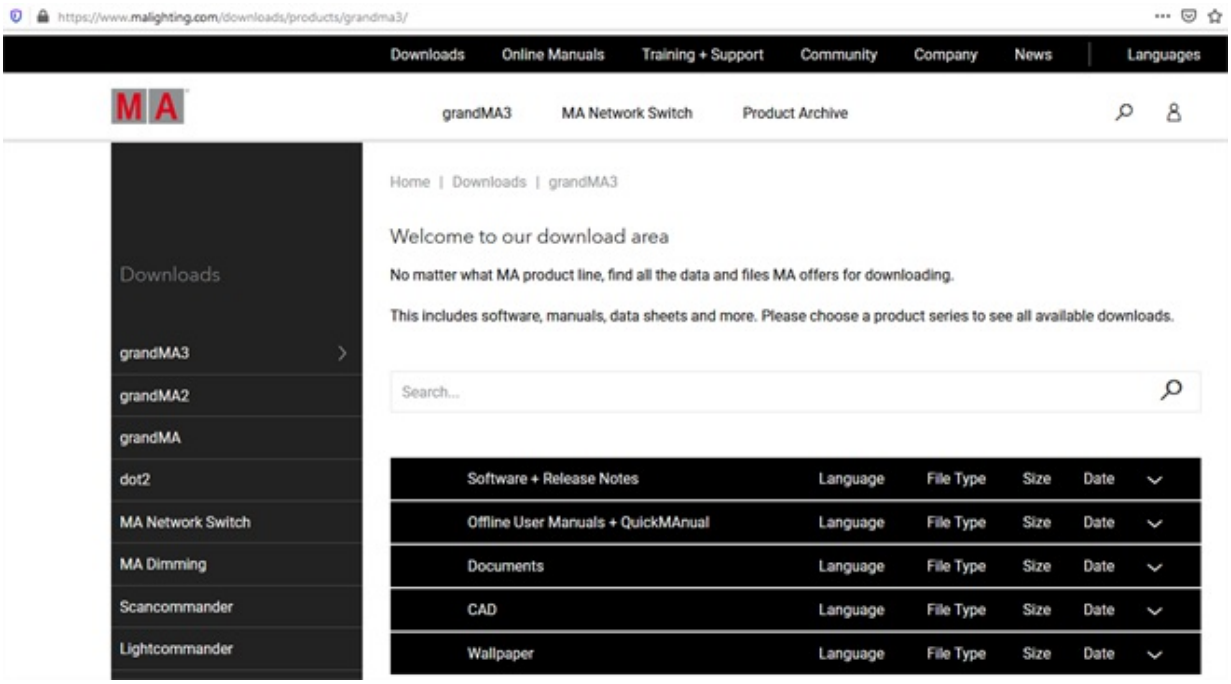
To downdate a grandMA3 device to an older software version, e.g. in a permanent installation, follow these steps:

Go to **www.malighting.com**, click **Downloads.**



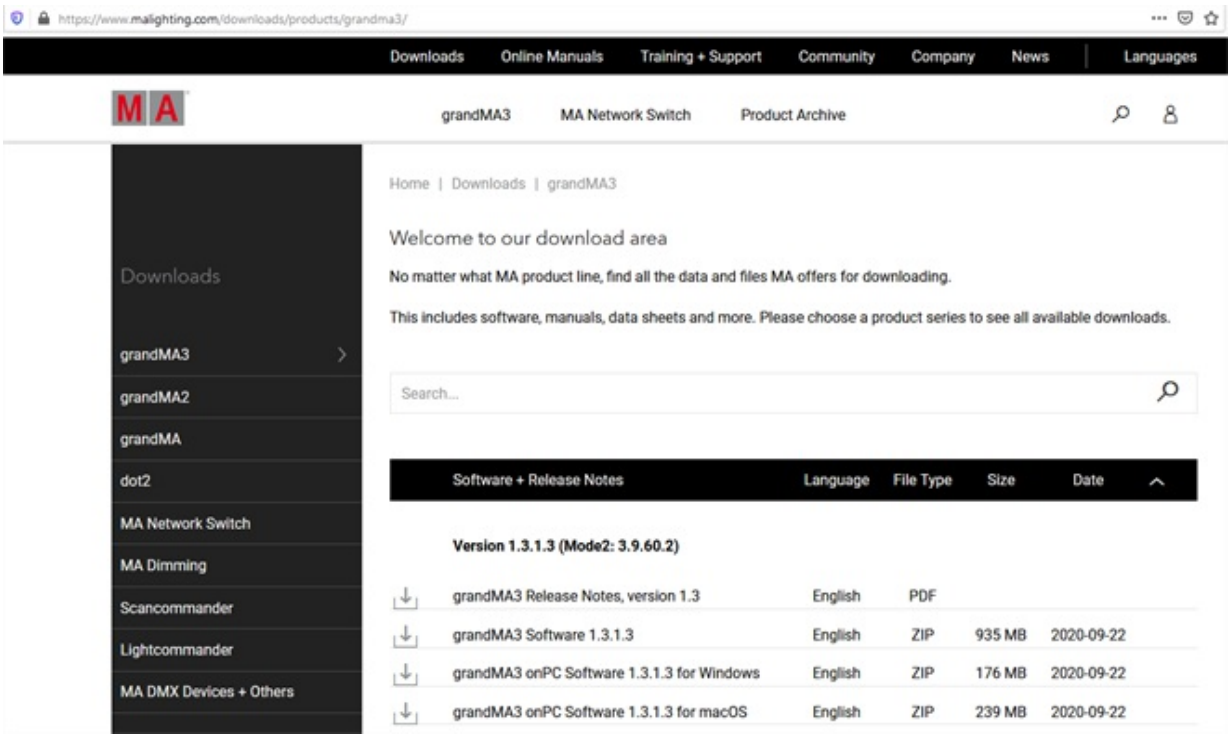
Website malighting.com Downloads

2. Click **grandMA3** in the bar on the left.



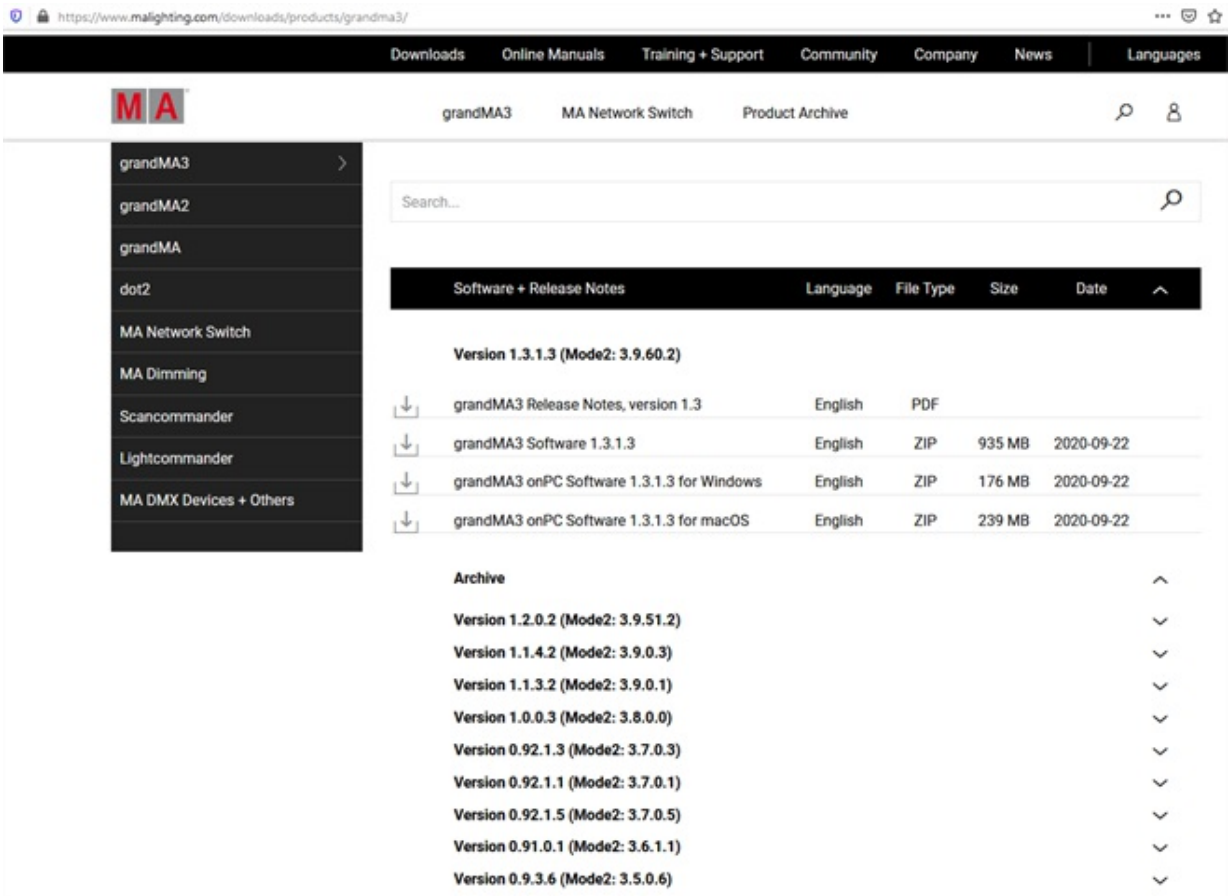
grandMA3 section

3. Click **Software + Release Notes**.



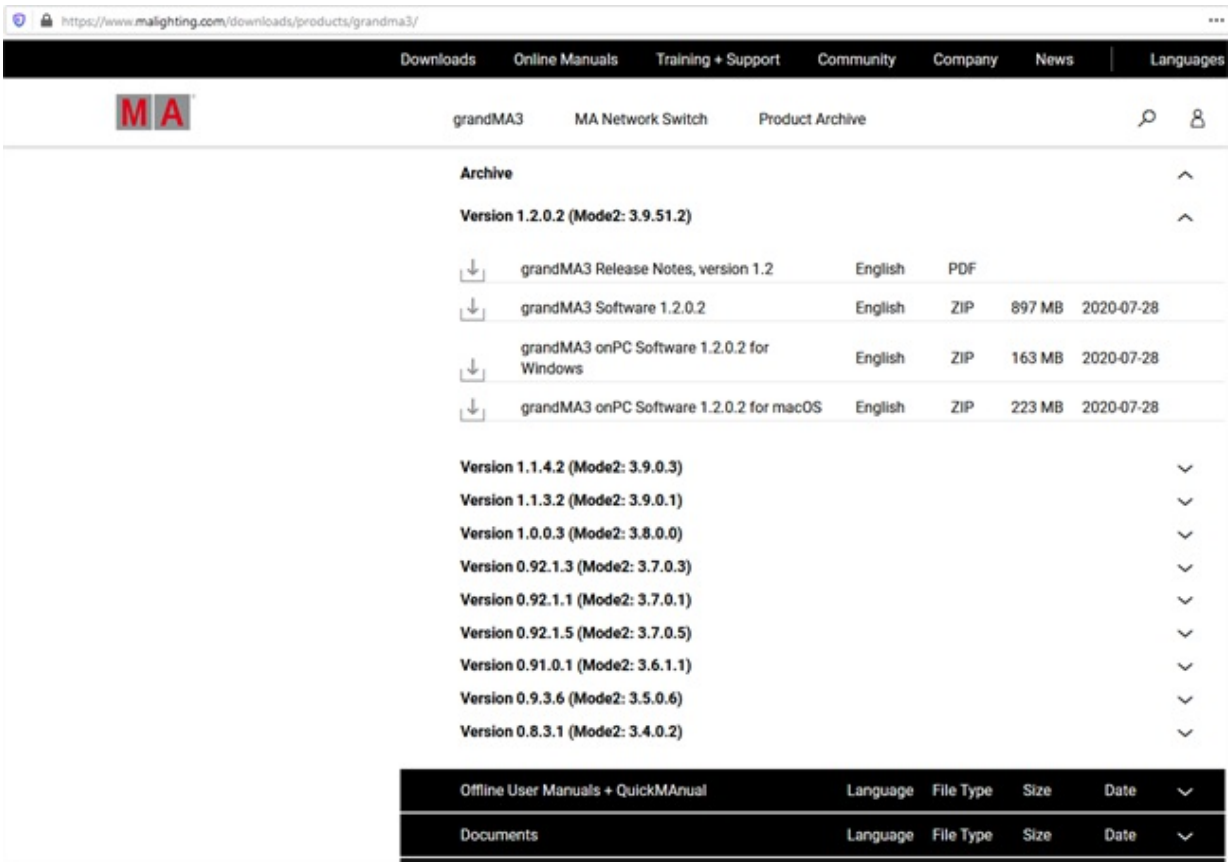
grandMA3 Software section

4. Scroll down and click **Archive** to find older software versions.



grandMA3 Software archive

5. To open the folder, click the desired version, e.g. Version 1.2.0.2.



grandMA3 Software archive versions

6. Click the desired installer version to download the required installation package.

The download process starts.

Available Installation Packages

There are two installation packages available:

- grandMA3_vx.x.x.x_full_package.zip

Use the full package to update the grandMA3 installations:
Console, RPU, processing unit, onPC (Windows and macOS), viz-key (Windows and macOS), xPort Node, and I/O node.

- grandMA3_vx.x.x.x_onPC_package.zip

Use the onPC package to update the grandMA3 onPC software installations:
onPC (Windows and macOS), and viz-key (Windows and macOS), xPort Node, and I/O node.

Store the Installation Packages

All installation packages for grandMA3 software are included in one zip-file.

Extract the entire data from the installation package zip-file.
The folders EFI, ma, and other data included are extracted.

For Windows systems, copy the files from the ma folder into the directory C:\ProgramData\MALightingTechnology\installation_packages.

For macOS systems, copy the files from the ma folder into the directory ~/MALightingTechnology/installation_packages.

Follow all the onscreen instructions that appear during the update.

40.1. Update grandMA3 Consoles

This topic describes the software update with additional options like factory reset etc.

For a simple software update via the user interface or UI, follow the instructions in the **Network Update topic**.

Important:
- The folders EFI, ma, and the update.scr file have to be directly accessible on the flash drive and must not be located in an extra folder.
- The USB flash drive's data system has to be FAT32.

Download the Software Package

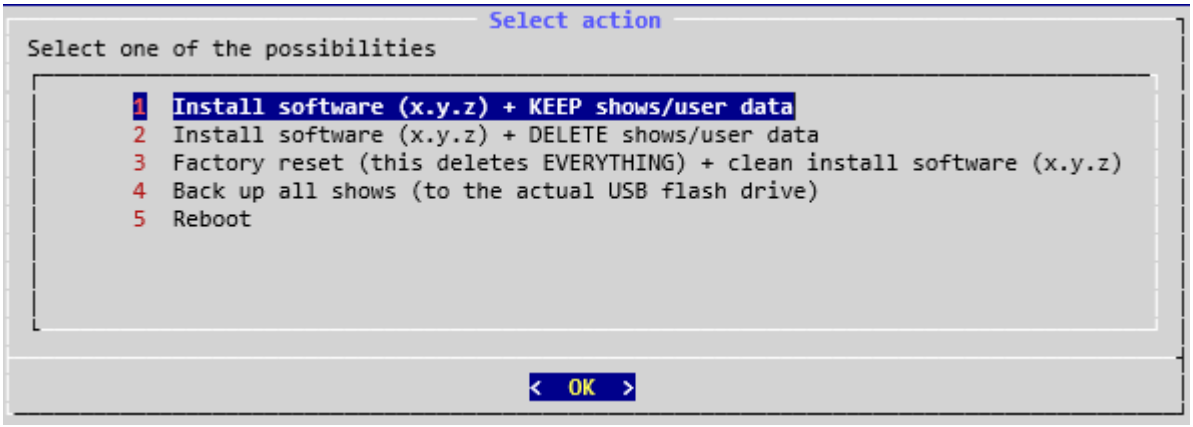
1. To update a grandMA3 device, download the latest software version from **www.malighting.com**.
   The required installer is called grandMA3 Software x.x.x.x.
2. Extract the zip file "grandMA3_stick_v.x.x.x.x.zip" and copy the folders EFI, ma, and the update.scr file into the root directory of your USB flash drive.
3. Insert the USB flash drive in a USB port.

Install the Software Package on the Console

1. Turn off the grandMA3 device.
2. For devices without an integrated keyboard (e.g. grandMA3 compact console), connect an external keyboard with a USB port.
3. Turn on the grandMA3 device.
4. Press the key 8/F8 on the internal or the external keyboard several times.
   The **Boot Manager** opens.
5. Scroll down to EFI USB Device using the arrow keys.
6. Press Enter on the (external) keyboard.
   The console starts to boot. The EULA screen opens.
7. Accept the EULA.

Important:
If the USB flash drive contains more than 1 version, select the version you would like to install first.

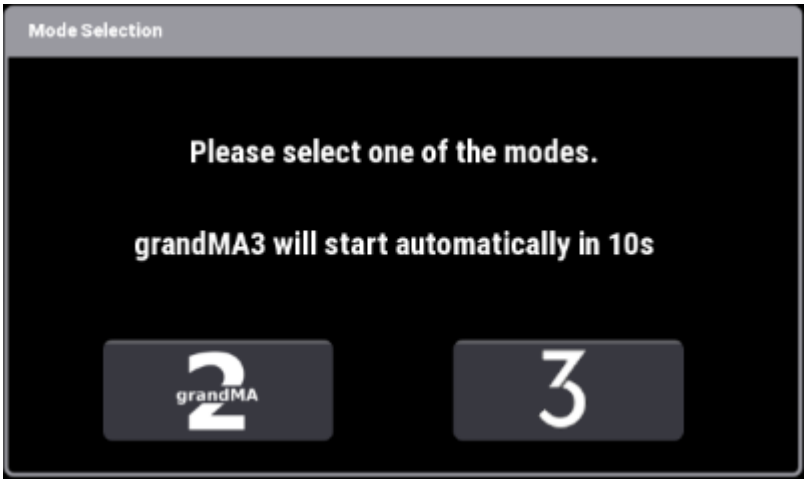
The Install Selector dialog appears:



- Select option 1 to keep the shows and user data.
- Select option 2 to delete the shows and user data.
- Select option 3 for a complete factory reset to clean the system before installing the software.
- Select option 4 to save the shows on the USB flash drive.
- Select option 5 to reboot the grandMA3 device.

1. Press Enter on the (external) keyboard.
   Wait for completion. The grandMA3 device reboots.
2. Remove the USB flash drive.

The Mode Selection dialog appears.



Select mode

3. Select one of the modes.

For more information about grandMA3 mode2, read the **Mode2 topic** in the section grandMA3 Mode2 of the **grandMA2 User Manual**.

If no selection is made, the device automatically starts in grandMA3 mode.

Screens 1, 2, 3 are initializing.

The letterbox screens go into self-test mode (red, green, blue, white, and black color changer).

The command screens stay black.

It can take several seconds for them to start initializing.

The installation is complete.

40.2. Update grandMA3 Nodes

Important:
We recommend formatting with every update.
Whenever formatting, the IP address will be reset back to default (DHCP).

Important:
-The folders EFI, ma, and the update.scr file have to be directly accessible on the flash drive and must not be located in an extra folder.
-The USB flash drive's data system has to be FAT32.

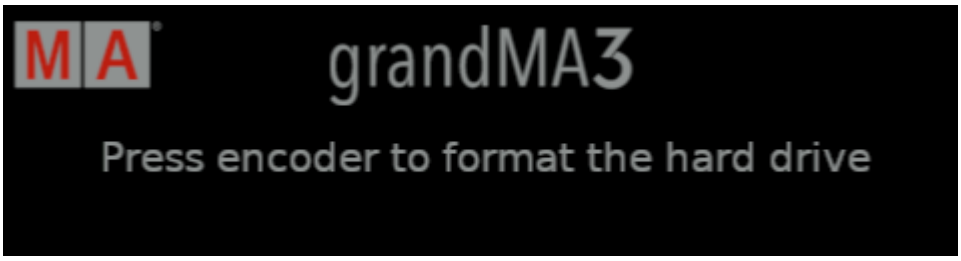
Requirement:

1. To update a grandMA3 device, download the latest software version from **www.malighting.com**.
   The required installer is called grandMA3 Software x.x.x.x.
2. Extract the zip file "grandMA3_stick_v.x.x.x.x.zip" and copy the folders EFI, ma, and the update.scr file into the root directory of your USB flash drive.
3. Insert the USB flash drive in the device's USB port.
4. Turn off the grandMA3 device.
5. Turn on the grandMA3 device.
6. Follow the onscreen instructions during the update process.

Update notification

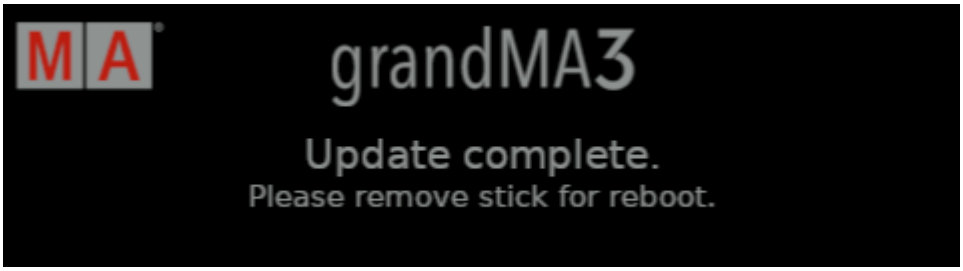
- Press the rotary knob.

Corrupted installer package



Format the hard drive

grandMA3 xPort Node is updating



Update completed

40.3. Update grandMA3 onPC Windows Hardware

Updating the onPC Software

Updating the grandMA3 onPC software for the grandMA3 onPC windows hardware like command wing XT or rack-unit requires a normal grandMA3 onPC software update.

For more information, see the **Network update topic**.

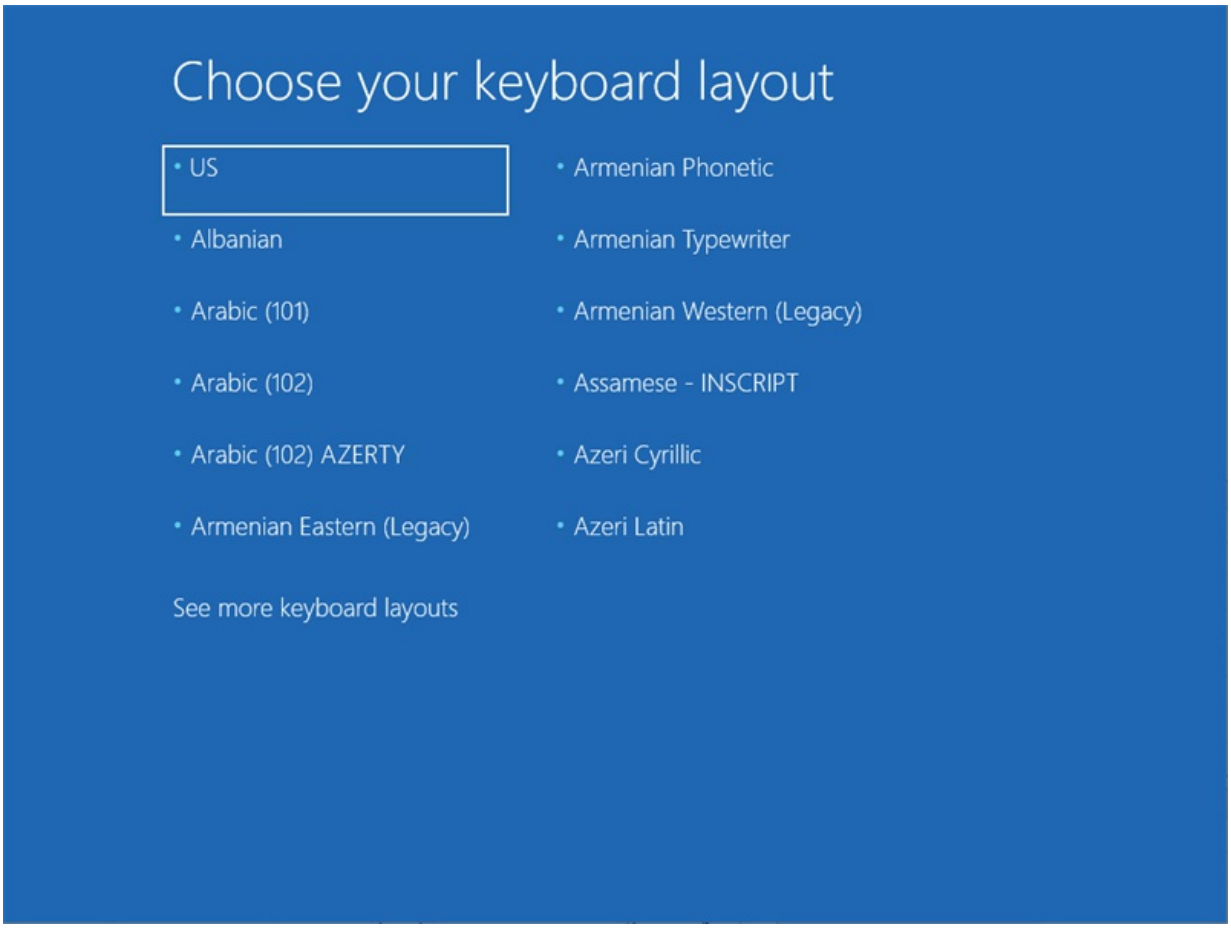
Important:
Select the Windows installer version!

Reset to Factory Defaults

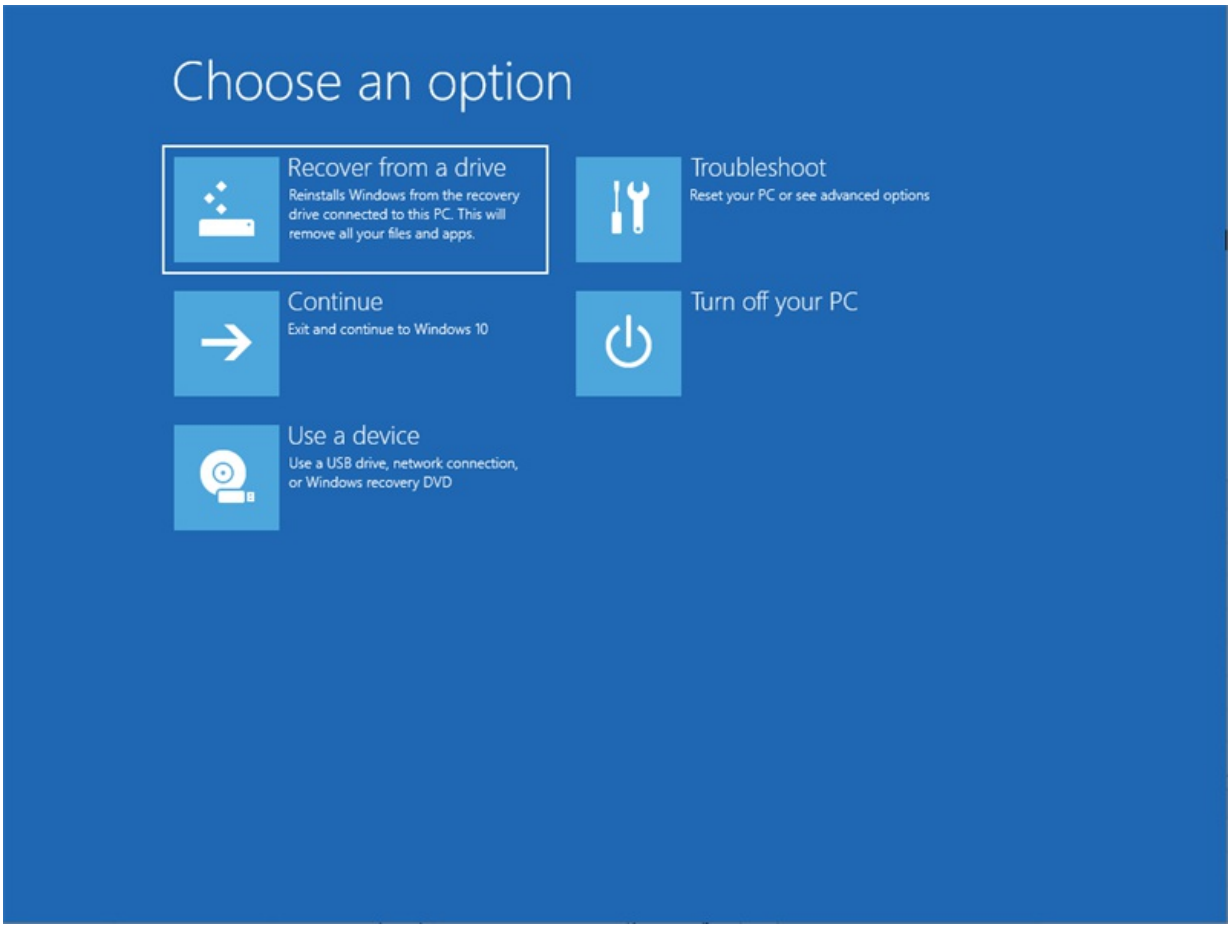
To reset the grandMA3 onPC command wing XT or rack-unit to factory defaults, it is necessary to install the full software image onto the hard drive including all operating system and grandMA3 onPC software information.

The installer package always installs the latest version of the onPC software and the Windows updates!

- To update, download the latest software version from **www.malighting.com**.
  The required installer is called grandMA3 onPC Windows Hardware Image x.x.x.x.
- Save the full software image on your hard drive.
- Extract the entire data from the zip file and copy it onto a USB flash drive (FAT32 formatted, minimum 8 GB).
- Insert the USB flash drive.
- Power up the grandMA3 command wing XT or rack-unit.
- If necessary, tap F8 to choose Boot device and select boot from USB.

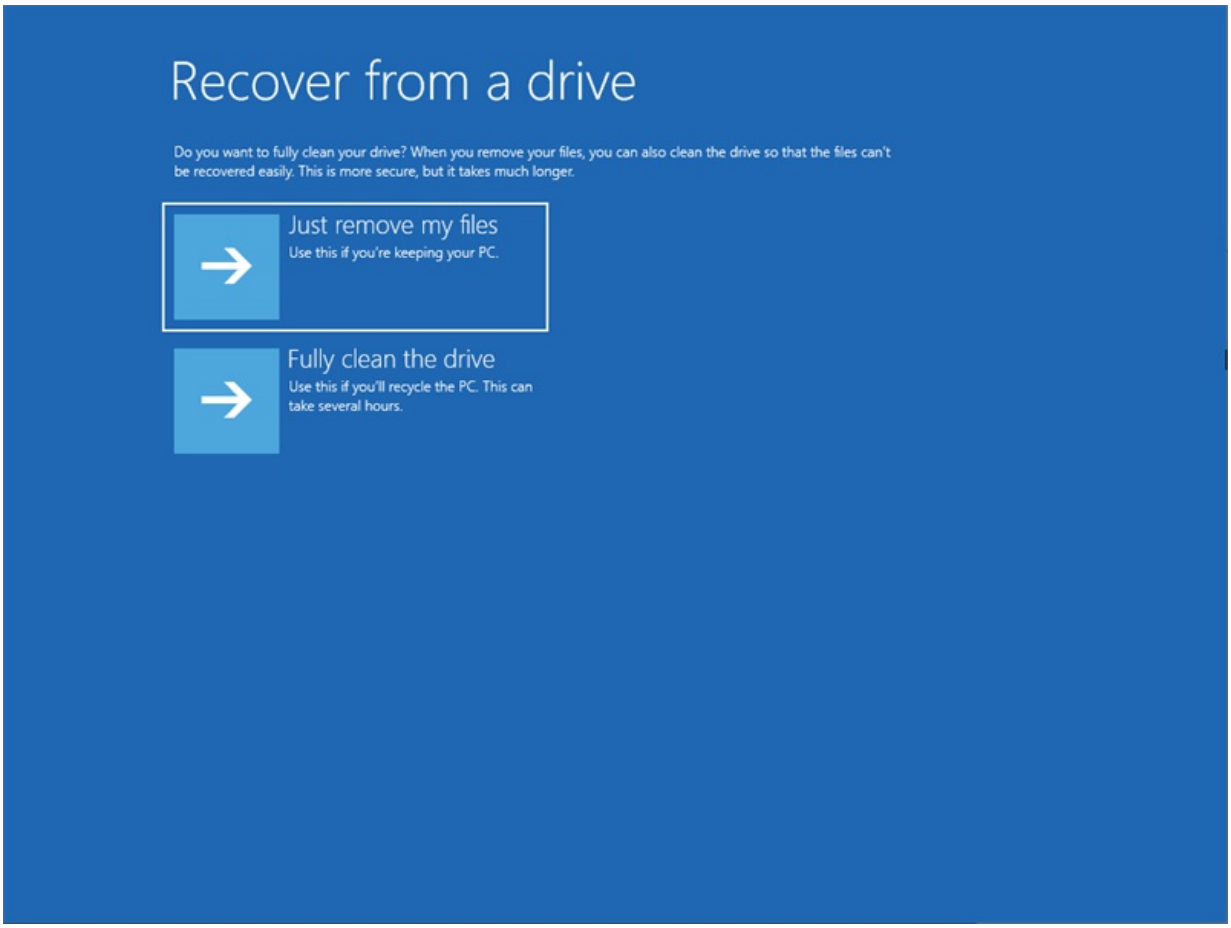


- Choose a keyboard layout: (US).

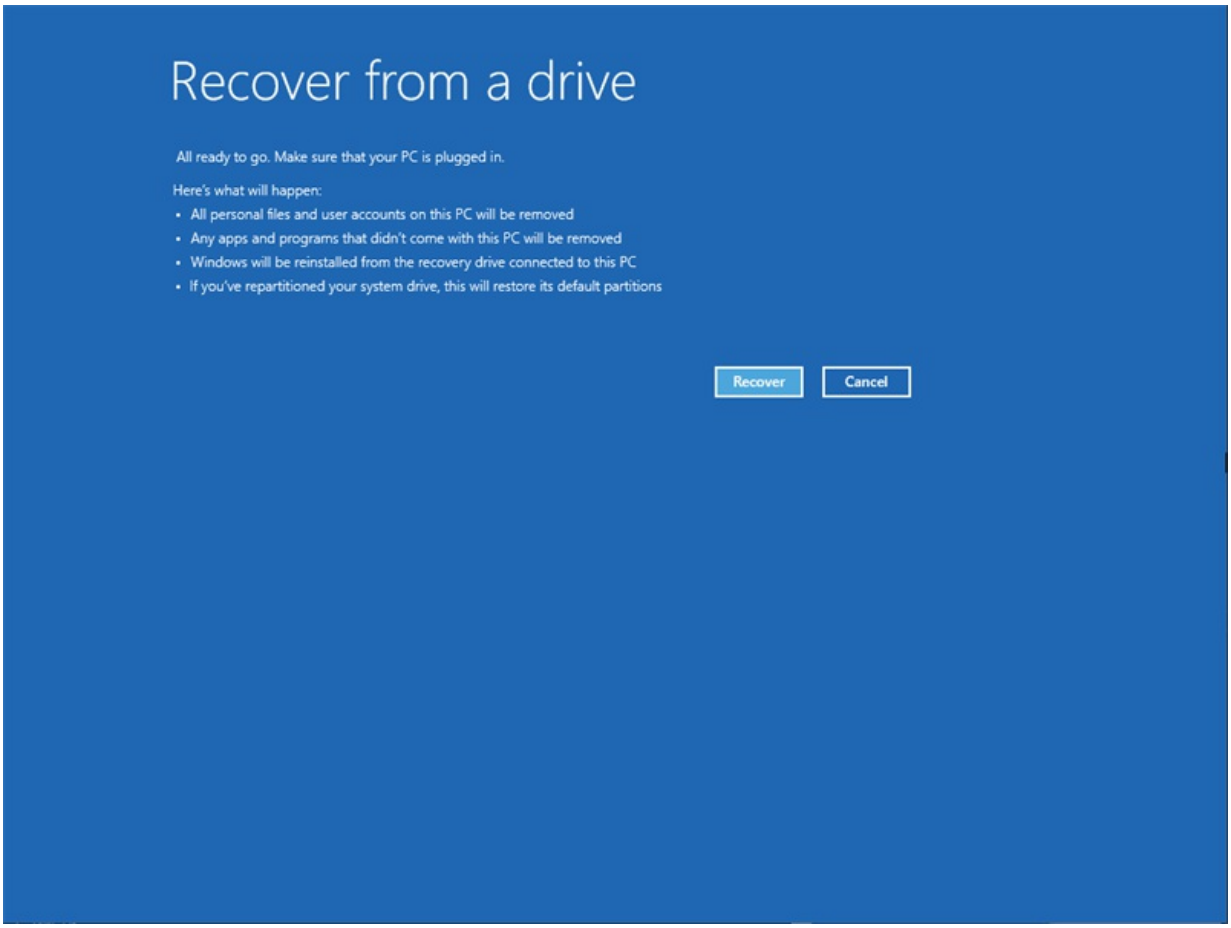


- Choose an option: Recover from a drive.

Hint:
If the "Recover from a drive" icon does not appear, the USB flash drive may be faulty.
Try another USB flash drive and make sure that the files are properly unzipped.



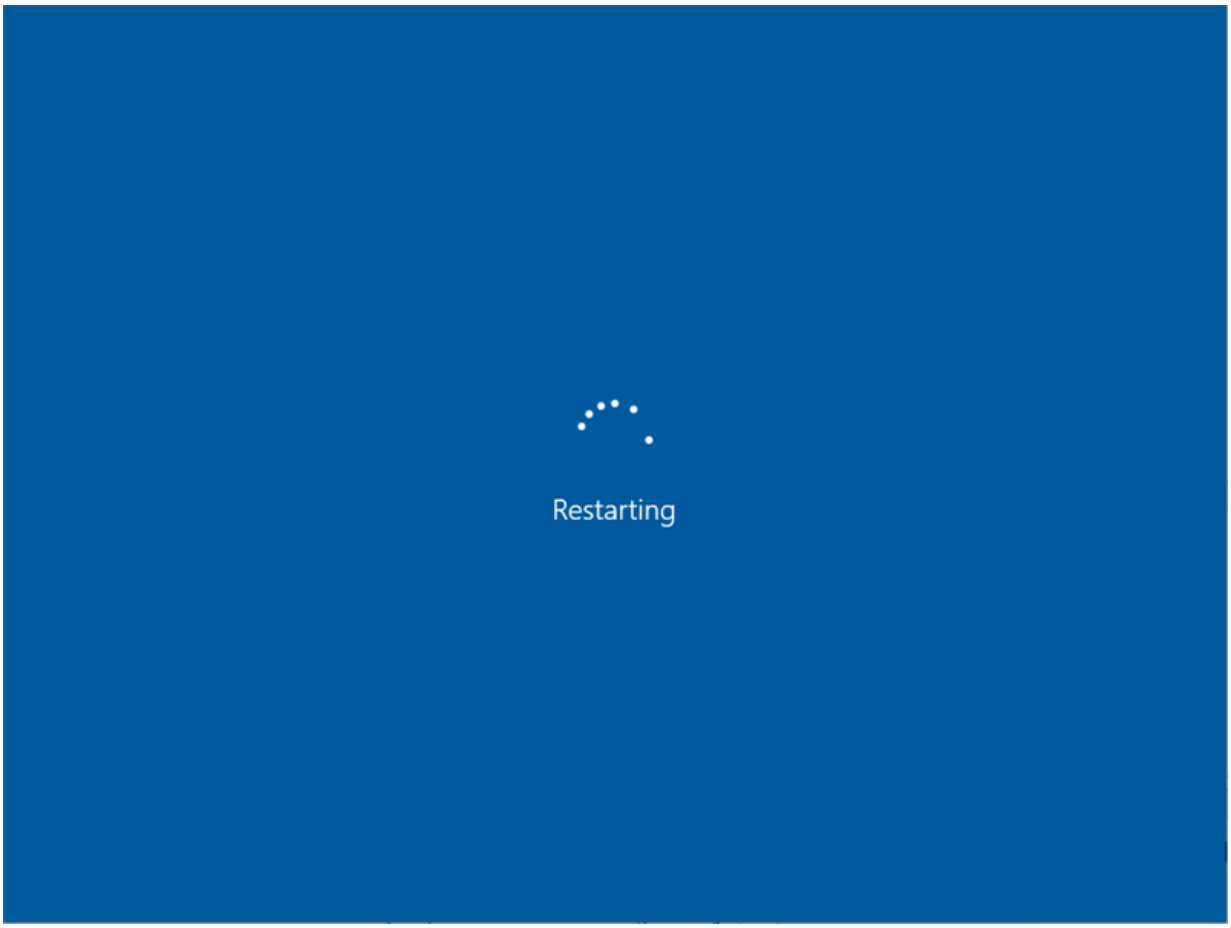
- Select Recover from a drive: Just remove my files.

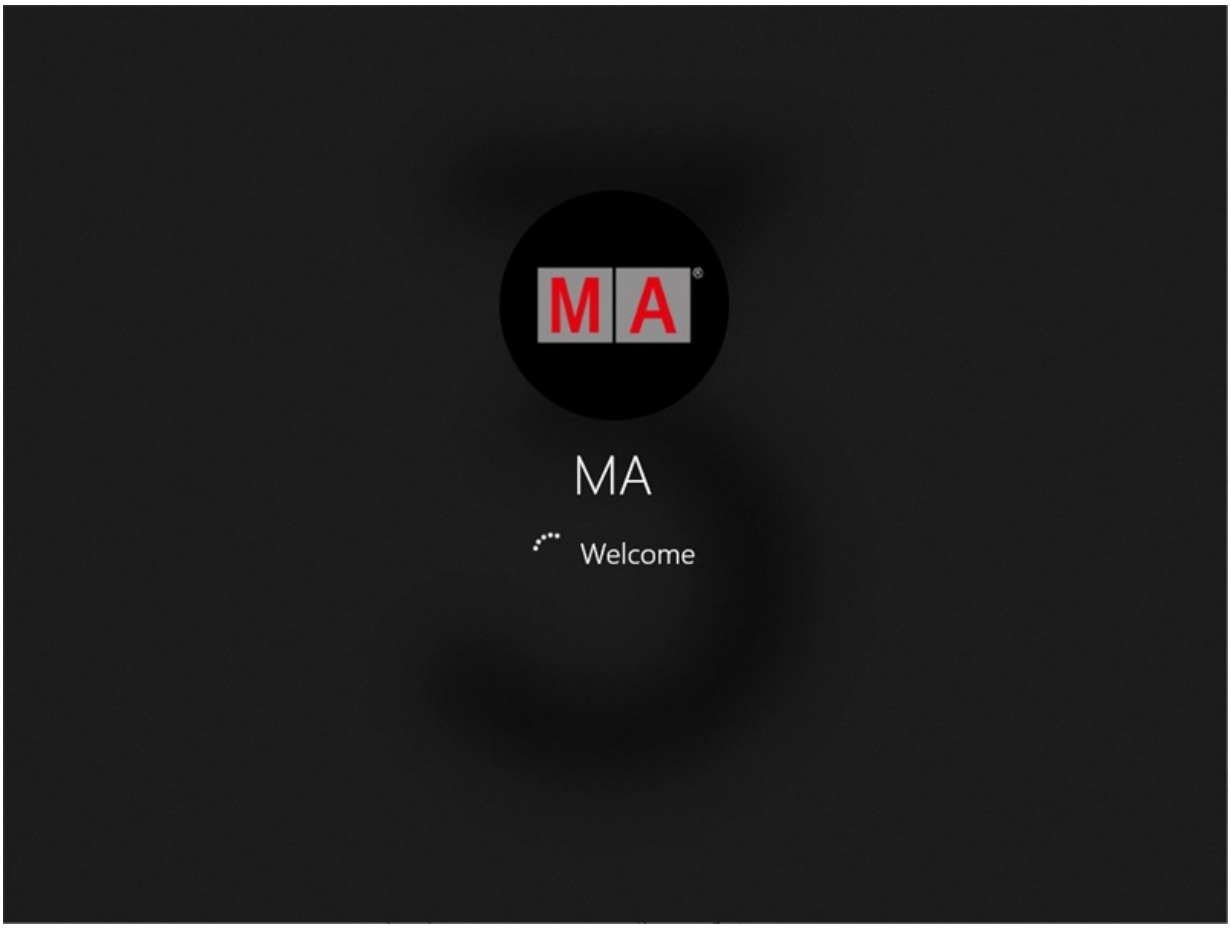


- Recover from a drive: Select Recover.

- Win 10 Licence Agreement: Click Accept.

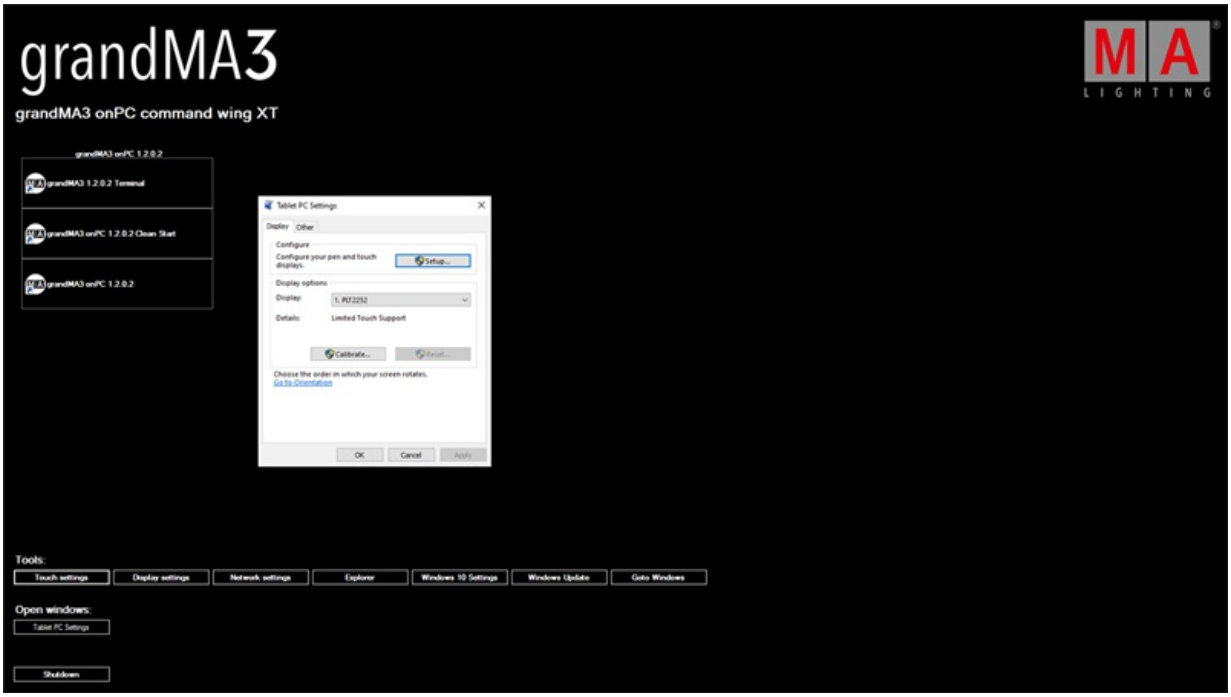
The system restarts several times.



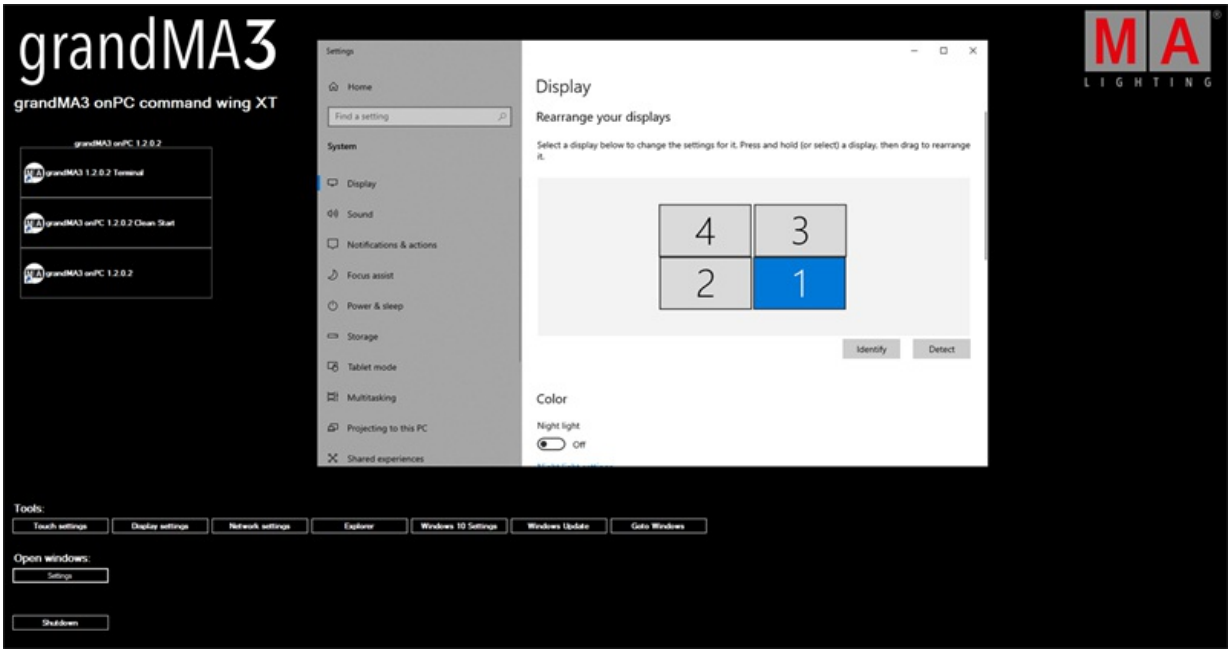


The MA Shell Launcher starts for the first time.

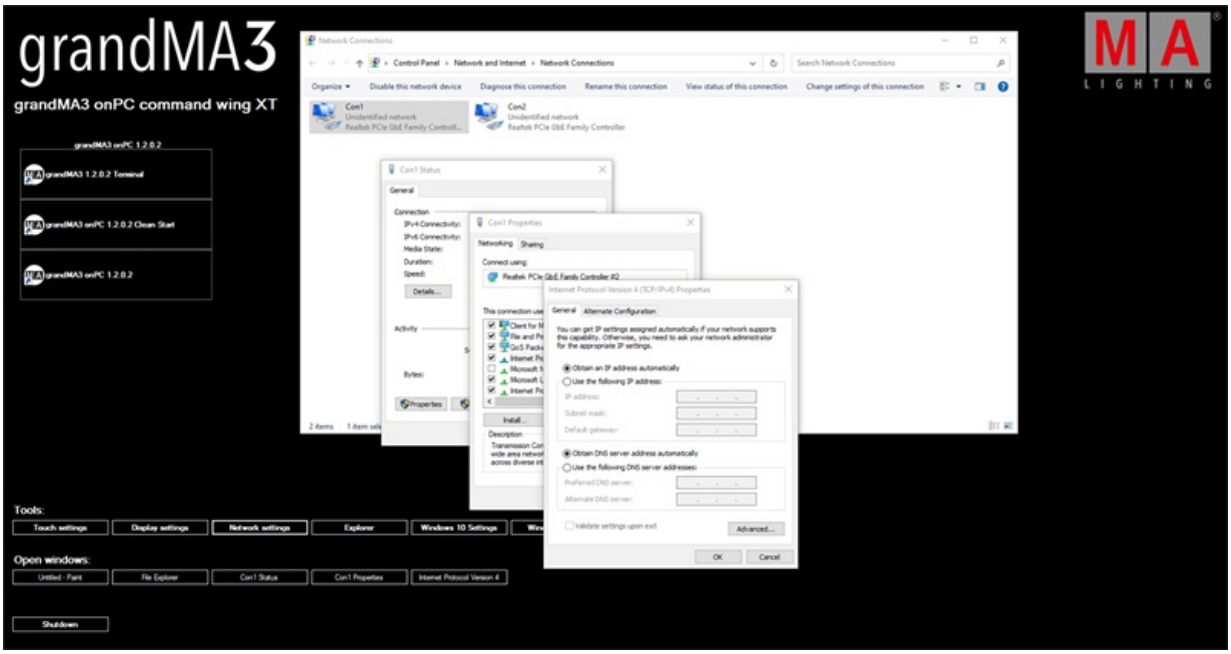
Configure the Settings



- Configure the touch screens.

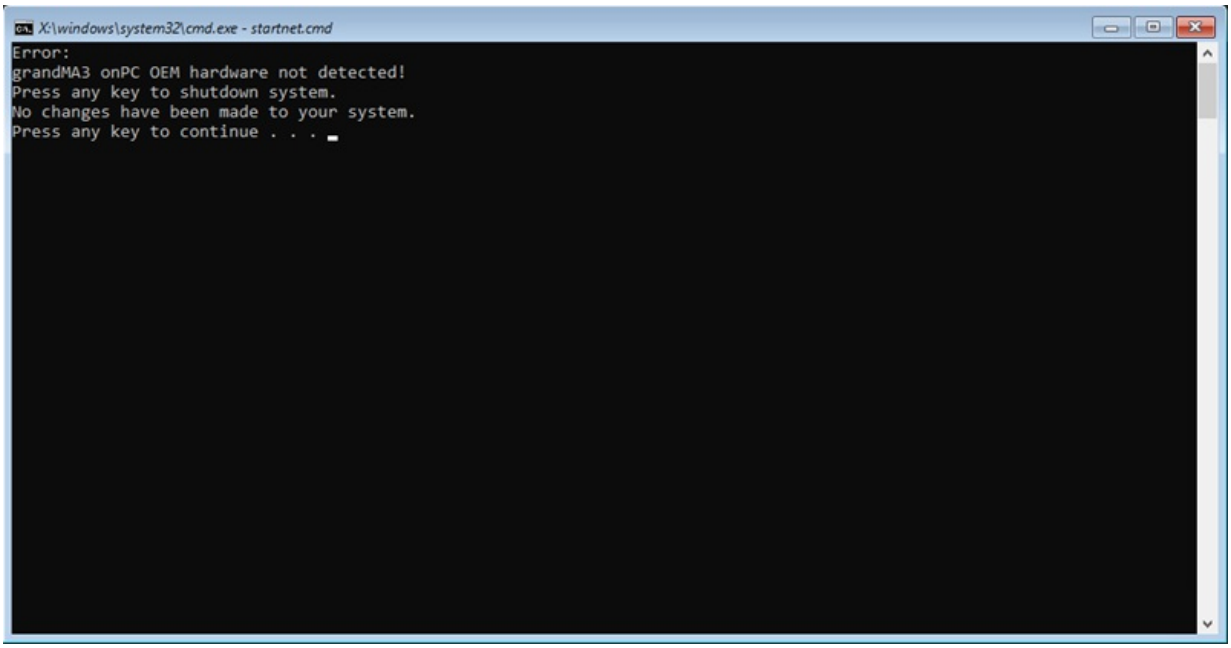


- Set the desired screen configuration.



- Set the desired IP address configuration.

Hint:
If a wrong mainboard is installed, the following error message pops up.



40.4. Update grandMA3 viz-key

Important:
To configure the third-party visualizer with viz-key support, or to install, update or downdate the viz-key connectivity software, use a grandMA3 console or your PC or laptop with the grandMA3 onPC software installed.

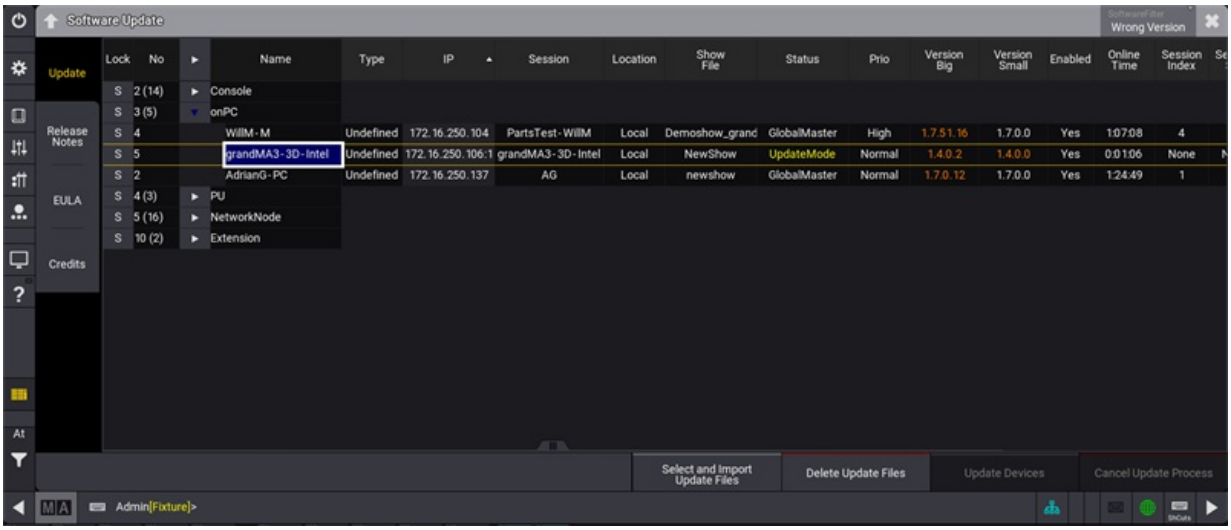
Important:
-The ma folder has to be directly accessible on the flash drive and must not be located in an extra folder.
-The USB flash drive's data system has to be FAT32.
-Alternatively, you can copy the contents of the ma folder into the directory installation_packages on your PC or laptop.
For Windows systems, copy the files from the ma folder into the directory C:\ProgramData\MALightingTechnology\installation_packages.
For macOS systems, copy the files from the ma folder into the directory ~/MALightingTechnology/installation_packages.

Important:
The viz-key must be connected with the laptop or PC where the third-party visualizer must be running in the update mode.

Accessible stations are located and invited using the **Network menu**.

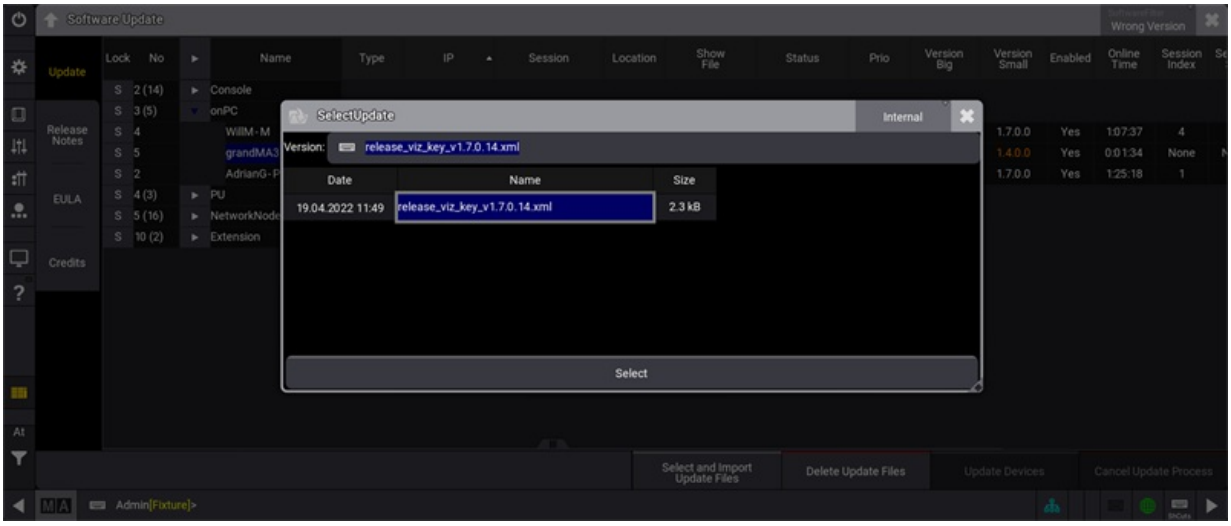
1. Download the latest viz-key version from **www.malighting.com**. For more information, see **update the software**.
2. The required installer is called grandMA3 Software x.x.x.x for grandMA3 viz-key.
3. Extract the zip file grandMA3_viz_key_vx.x.x.x.zip and copy the ma folder into the root directory of your USB flash drive.
4. Insert the USB flash drive in the grandMA3 onPC or console USB port.
5. To access Software Update, tap
6. Tap Settings.
7. Tap Software Update.

The Software Update window opens:



Software Update window

8. Tap Select and Import Update Files.
   The selection window opens:



Select Update pop-up

9. Select the location that contains the update files (internal or any plugged-in external device). Select the release_viz_key_vx.x.x.x file.
   The selected update file is displayed at the title bar of the software update window.
10. Tap Select.
    The pop-up closes and the End User License Agreement (EULA) opens.
    Confirm the End User License Agreement (EULA).
11. Select the third-party visualizer with viz-key support. The selected devices turns into bright blue.
12. Tap Update Devices.
13. The software update starts copying files.
14. Once the file is transferred, restart the third-party visualizer with the viz-key support software.

Hint:
To learn more on how to connect, visit **https://www.malighting.com/viz-key**.

40.5. Network update

To update one or several stations in a network use software update.

Stations are located and invited using the **Network menu**.

Important:
Make sure that all devices you want to update are in the same network (check network adapter).
To learn more about network settings, read the Networking, **Interfaces and IP** topic.

Hint:
It is also possible to select multiple devices with the lasso selection or by holding Ctrl and selecting the devices.

Important:
Make sure that you only select devices of the same operating system.

Preparation

- Make sure that the grandMA3 devices are connected with the grandMA3 console, laptop, or PC using the etherCON/RJ45 connector.
- Download all required installation packages from **www.malighting.com**, **Downloads.**
- Copy all installation packages for grandMA3 software into the root directory of your USB flash drive.
- Insert the USB flash drive in the device's USB port.
- For Windows systems, copy the files from the ma folder into the directory C:\ProgramData\MALightingTechnology\installation_packages.
- For macOS systems, copy the files from the ma folder into the directory ~/MALightingTechnology/installation_packages.
- Optional: to copy the installation packages on the hard drive of a grandMA3 device, tap Import Selected Update File. The files are copied into the directory installation_packages.

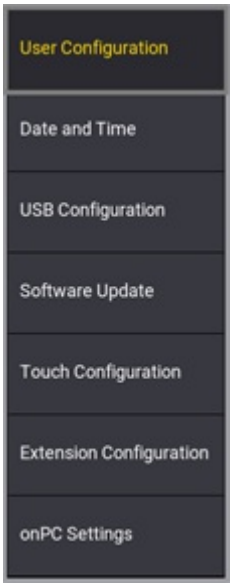
Procedure

To properly update the grandMA3 device, follow all the onscreen instructions that appear during the update.

To access Software Update:

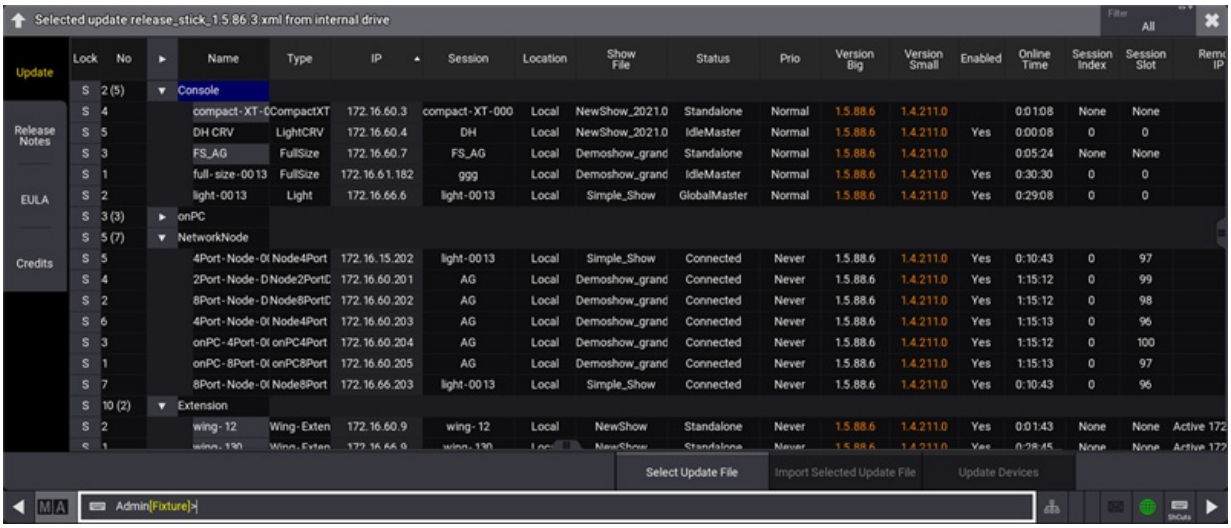
1. Tap .
2. Tap Settings.

A drop-down menu opens:



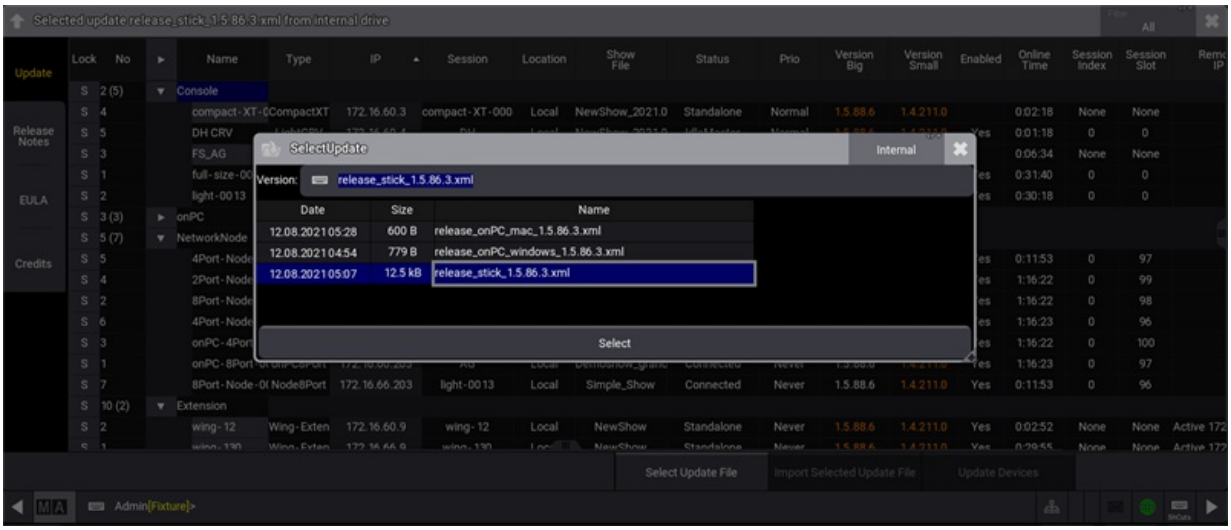
Settings pop-up

3. Tap Software Update.
   The Software Update window opens:



Software update window

4. Tap Select Update File.
   The selection window opens:



Selection window

6. Select the location that contains the update files (internal or any plugged-in external device). Select the update file you want to execute.
   The selected update file will be displayed in the title bar of the software update menu.
7. Tap Select.
   The pop-up closes and the End User License Agreement (EULA) opens.
8. Confirm the End User License Agreement (EULA).
9. Select the desired device(s). The selected devices are marked in blue.
10. Tap Update Devices.
11. The software update starts copying files.
12. The devices reboot.

Hint:
It is also possible to update your **onPC**, **consoles**, **xPort Nodes**, **onPC command wing XT**, and **onPC rack-unit** manually.

40.6. Troubleshooting Update Process

During the update process, different problems may occur.

Corrupted Files on USB Flash Drive

Corrupted installer package

If the installer package is corrupted, try the following:

- Remove the USB flash drive from the grandMA3 device.
- Shutdown the grandMA3 device.
- Format the USB flash drive (FAT32).
- Copy the folders EFI, ma, and the update.scr file into the root directory of your USB flash drive.
- Update the grandMA3 device with the USB flash drive.

If the error message shows up again, try the following:

- Use a different USB flash drive.
- Format the USB flash drive (FAT32).
- Start a new download of the latest software version from **www.malighting.com**.
- Extract the zip file before copying the folders EFI, ma, and the update.scr file into the root directory of your USB flash drive.

Corrupted Show File

If the grandMA3 device does not stop updating, the show file saved on the grandMA3 device may be corrupted.

If the show file is corrupted, try the following:

- Remove the USB flash drive from the grandMA3 device.
- Reboot the grandMA3 device.
- Save the corrupted show file to another USB flash drive.
- Delete the corrupted show file from the grandMA3 device.
- Update the grandMA3 device with the other USB flash drive.

Individual Hardware Freezes

If the grandMA3 device starts updating, but fails to update one of the hardware sections, try the following:

- Remove the USB flash drive from the grandMA3 device.
- Reboot the grandMA3 device.

If the measures above do not help, try the following:

- Save any show file to another USB flash drive.
- Delete any show file from the grandMA3 device.
- Update the grandMA3 device with the other USB flash drive.

41. Fixture Types

Fixture types are used to visualize and control real-life fixtures onstage.

A fixture type is a footprint of a real fixture, smoke machine, laser, media server, dimmer, or anything to be controlled and visualized with the grandMA3.

It describes its modes, channels, and its physical properties.

You can find fixture types in the grandMA3 library.

Fixture types are also available for downloads from MA fixture share and GDTF share.

For more information, see **https://www.malighting.com/training-support/fixture-shares/**.

41.1. Import Fixture Types

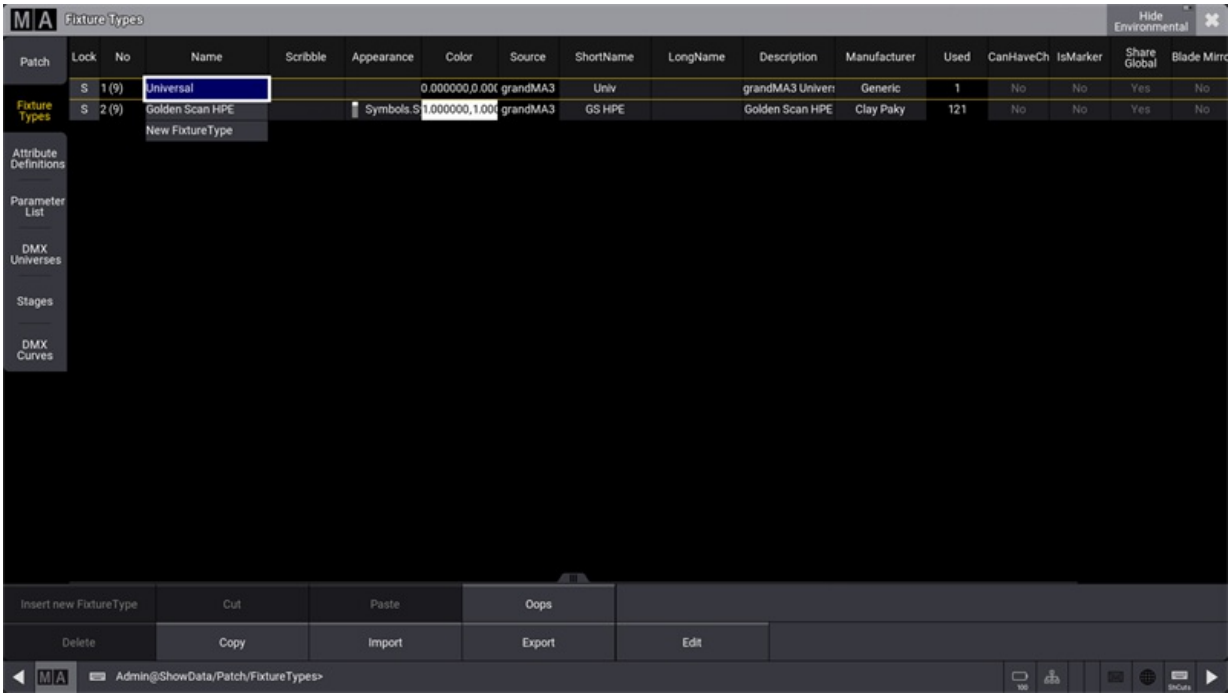
It is possible to import fixture types from these libraries:

- **grandMA3** fixtures
- Converted **grandMA2** fixtures
- Fixtures using **GDTF** format

Requirement:

- Access Patch

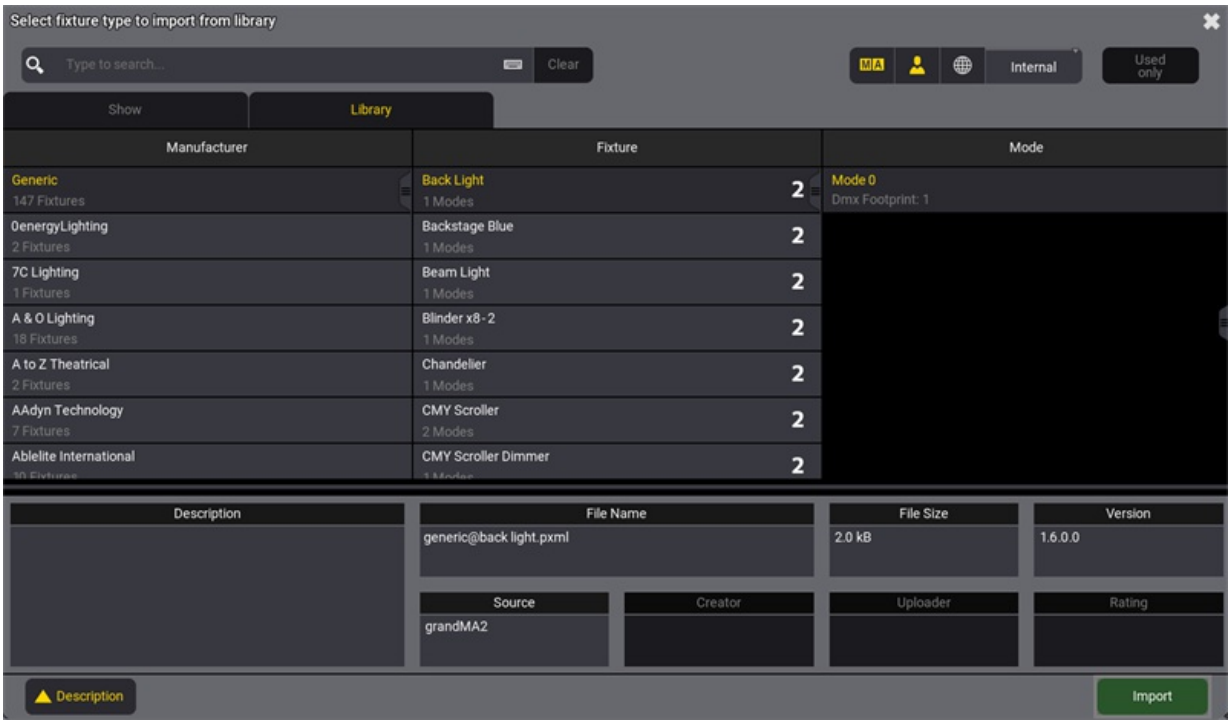
1. Tap Fixture Types in the bar on the left of the patch dialog.
   The window Fixture Type opens.



Open Fixture Types

2. Tap Import .

The pop-up **Select fixture type to import** opens.



For a detailed description of this window see, **Add fixtures to the show**.

Hint:
The **Show** tab and the Used only button are available only when inserting a new fixture type from the patch menu.

41.1.1. Import GDTF

In grandMA3, it is possible to import a GDTF file (General Device Type Format) that contains the description of a fixture type.

The GDTF file is a zip file containing:

- Description
- Geometry data
- Gobo images

For a detailed description of GDTF, see this tutorial **gdtf-share.com/wiki/GDTF_File_Tutorial** and **gdtf-share.com/wiki/GDTF_File_Description**.

Import GDTF in the grandMA3 Console

1. Download the GDTF file on **www.gdtf-share.com**.
2. The file is located in the download folder.
3. Copy the GDTF file to a USB flash drive. The default path for GDTF fixture types is:
   /grandMA3/gma3_library/fixturetypes/gdtf
4. Insert the USB flash drive into the console.
5. Open the patch dialog and tap Fixture Types.
6. Tap Import.
   The **Select Fixture Type to Import pop-up** opens.

1. Tap the library button GDTF.
   The button is highlighted in yellow.
2. In the upper right corner of the title bar, select the USB flash drive that contains the GDTF files.
3. Select the fixture type you wish to import.
4. Tap Import.

The fixture is imported to the fixture type list and can now be patched.

For more information on how to patch, see **Add Fixtures to the Show**.

To learn how to import fixtures from the online **garandMA3 Fixture Share** and the **GDTF Share** libraries, see **Import GDTF**.

Hint:
You can proceed the same way to import a GDTF file to grandMA3 onPC if you insert a USB flash drive in the computer running the grandMA3 onPC software. If you want to import GDTF files to your computer's hard drive, see the following topic Import GDTF in grandMA3 onPC.

Import GDTF in the grandMA3 onPC

1. Follow steps **1 and 2** described in **Import GDTF in the grandMA3 Console**.
2. Copy the GDTF file to the appropriate folder on the hard drive. The default GDTF folder is found along the following path from the main MALightingTechnology folder: **MALightingTechnology/gma3_library/fixturetypes/gdtf**
   For more information on the default folder structure and where to find the main MALightingTechnology folder, see the **Folder Structure** topic.
3. Open the patch dialog.
4. Tap Fixture Types.
5. Tap Import.
   Select Fixture Type to Import pop-up opens.

6. Select Internal in the upper right corner of the title bar.
7. Select a fixture type and tap Import.

The fixture is imported to the fixture type list and can now be patched.

41.2. Build Fixture Types

grandMA3 allows building your own fixture types.

For more information on how to build fixture types, see the topics below.

41.2.1. Insert Fixture Types

To create a new fixture type, insert a fixture type with its basic data.

Important:
Make sure always to save the settings you made; otherwise, they will get lost.
How to save settings in Fixture Types:
1. Leave the Patch.
2. The **pop-up Leaving the patch** appears.
3. Tap, Ok.
4. Save the show file.

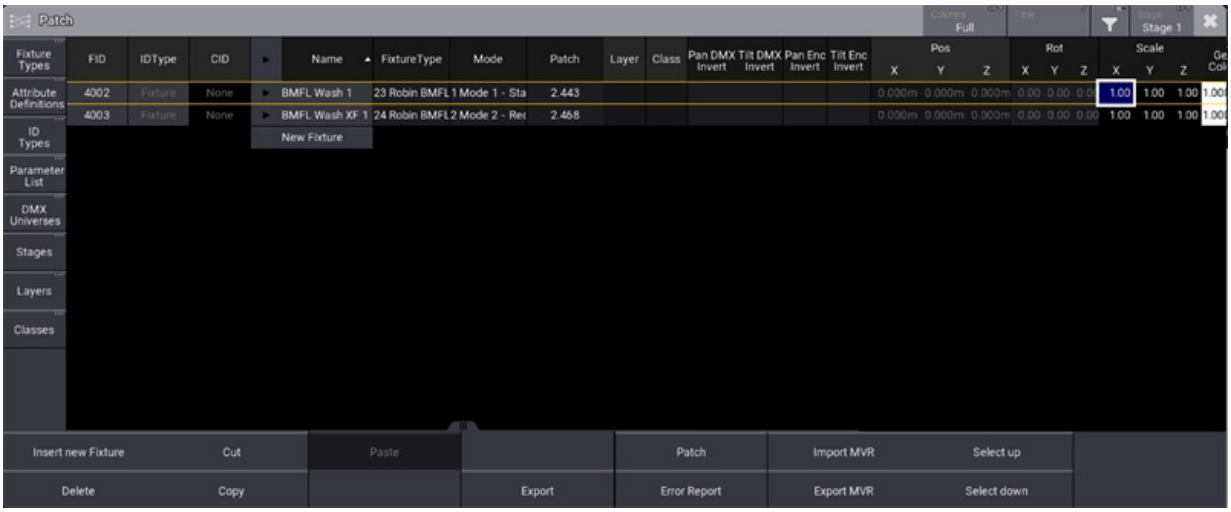
For more information, see **SaveShow Keyword**.

1. To open the menu, press Menu, or tap .
   The **Menu** window opens.

Open the pop-up Menu

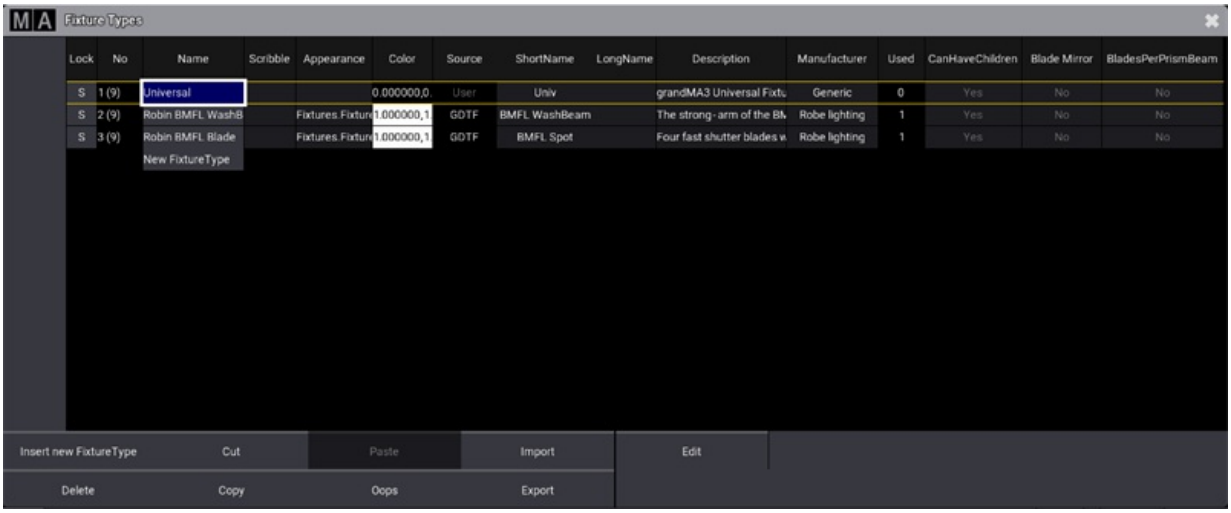
2. Tap Patch.
   The **Patch** window opens.

Hint:
If no fixtures are patched, the pop-up **Select DMX Mode to Use** opens. Close the dialog to proceed.



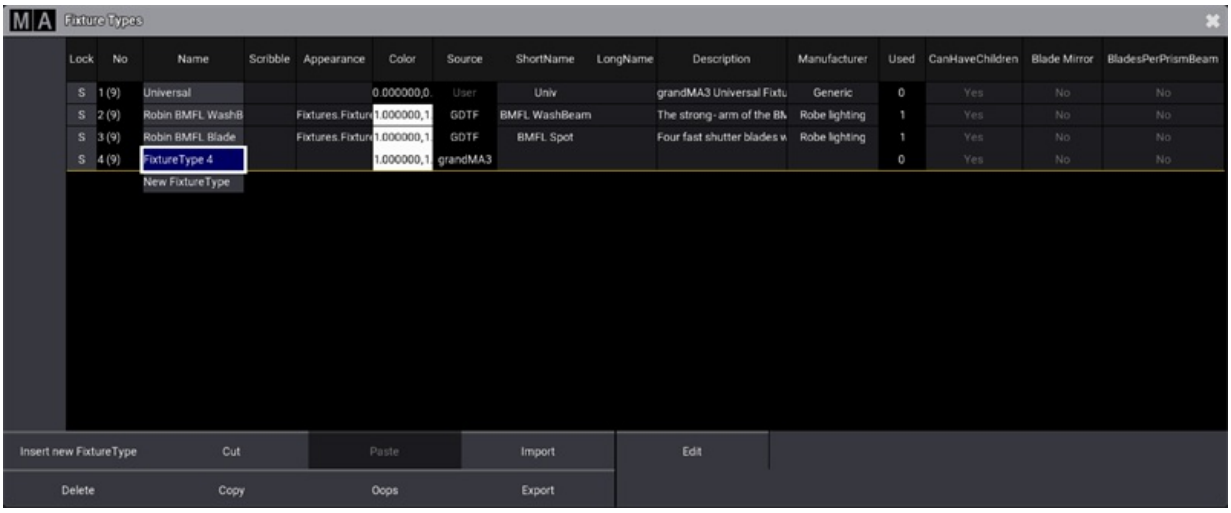
Open the patch menu

3. Tap Fixture Types .
   The **Fixture Types** menu opens.



Open the window Fixture Types

4. In the column **Name,** tap New Fixture Type.
5. Tap Insert New Fixture Type in the bottom left corner.



Insert a fixture type

A new fixture type row is inserted into the sheet. In the example above, **FixtureType 4**

6. To edit or enter the basic data of the fixture type, fill out these cells:
   Press Edit and tap the cell or tap and hold it.

Column	Description
Name	Official name of the fixture type.
Scribble	Selects scribble.
Appearance	Selects appearance.
ShortName	Short name of the fixture type.
Description	Description of the fixture type.
Manufacturer	Manufacturer of the fixture type.

41.2.2. Insert DMX Modes

A DMX mode consists of one or several DMX channels.

The DMXMode is the parent, and the DMXChannel is its child.

Requirement:

- Insert fixture type first.

For more information, see **Insert Fixture Types**.

Example:

- How to create a Basic moving head.

Important:
The build of a fixture type is based on the hierarchic structure of parent-child. A parent comes first, and the child follows its parent. There may be several children.

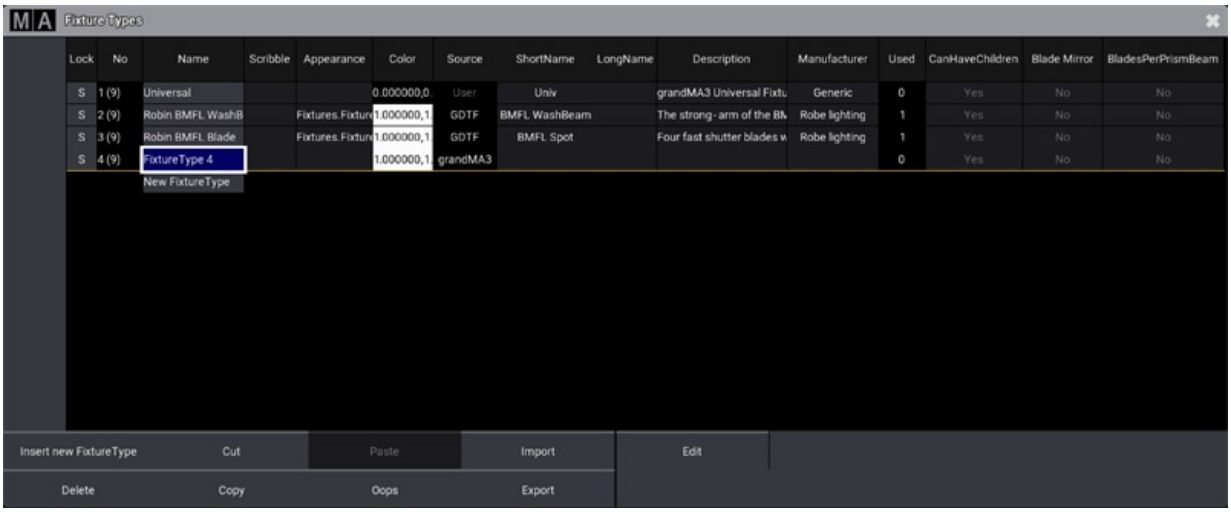
Hint:
Many manufacturers provide DMX charts that define channels.
Use a DMX chart to enter the channels in the fixture type table.

This basic moving head is based on this chart:

Relative Patch Address	Function
1	Pan (8 bit) Pan movement by 540°
2	Pan Fine (16 bit)
3	Tilt (8 bit) Tilt movement by 270°
4	Tilt Fine (16 bit)
5	Dimmer
6	Red
7	Green
8	Blue

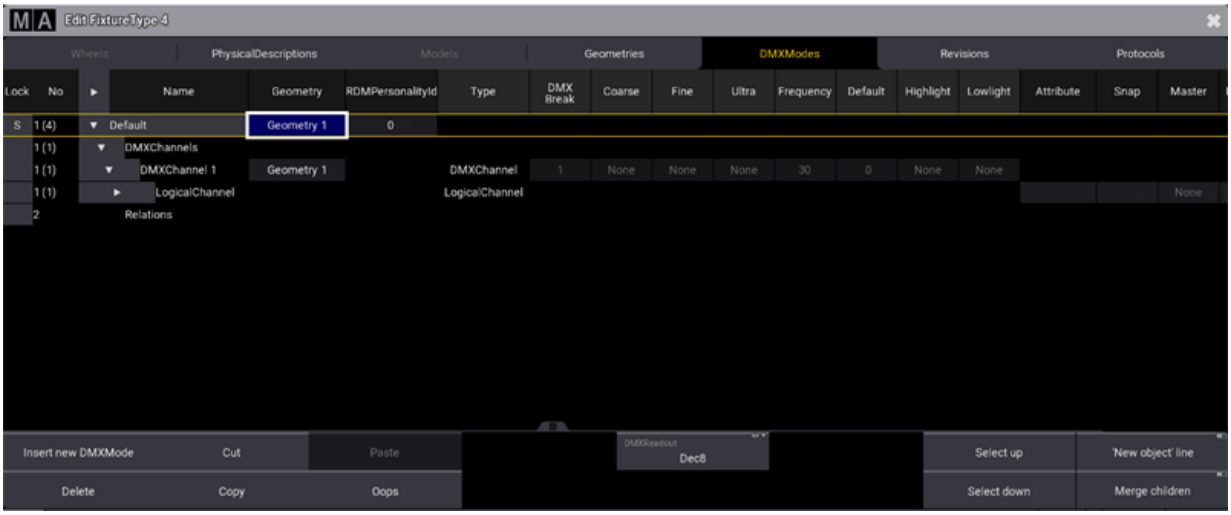
Important:
-To back up data during or after the build of fixture types, leave the **Fixture Type Editor** and tap Export in the **Fixture Types** menu.
An xml. file is exported. It is possible to import this file into the show file.
-Or save the show file. Leave the **Patch** menu and save all changes.

1. Set the focus to the fixture type.



Hint:
To add children in the window **Edit Fixture Type**, tap and hold **'New object' line**.

2. Tap Edit.
The window **Edit FixtureType** opens, and the focus is automatically set to the column **DMX Modes.**



Insert DMXMode

3. For a clear overview, tap to disable **Merge children** in the lower right corner of the window.

4. In the column **Name,** tap and hold the cell Default.
   -The **pop-up Edit Name** opens.
   -Rename Default to Mode 1.

Edit name of Default

Enter DMXChannel 1 – Pan

To enter the first channel:

1. Expand Mode 1:
   -Tap ▶.
2. In the row DMXChannel 1, enter:
   -**Coarse**: 1
   -**Fine**: 2
   -**Default**: 50 %
   -**Highlight**: none
3. Expand DMXChannel 1 for further entries in LogicalChannel:
   -Tap ▶ in the row DMXChannel 1.
4. In the row LogicalChannel select:
   -**Attribute**: Pan
5. The DMXChannel 1 is renamed to Geometry 1_Pan.
   -**Master**: none
6. Expand LogicalChannel for further entries in ChannelFunction:
   -Tap ▶ in the row LogicalChannel.

7. In the row ChannelFunction select:
   -**Attribute**: Pan
   -**Physical From**: -270
   -**Physical To**: 270

Hint:
As soon as you select an attribute, the DMX Channel is renamed after the attribute – **Geometry_Attribute**.

DMXChannel1 is entered.

Enter DMXChannel 2 – Tilt

To enter the second channel:

1. Tap Geometry 1_Pan.
2. Tap to enable **'New object' line.**
3. Tap New DMXChannel and tap Insert.
   New DMXChannel is renamed to DMXChannel 2.
4. To enter the second channel, enter in the row DMXChannel 2:
   -**Coarse**: 3
   -**Fine**: 4
   -**Default**: 50 %
   -**Highlight**: none
   -Open LogicalChannel. For more information, see **step 3** in the type LogicalChannel.
5. -**Attribute**: Tilt
6. DMXChannel 2 is renamed after the attribute.
   -**Master**: none
7. Open ChannelFunction. For more information, see **step 5** in the type ChannelFunction.
   -**Attribute**: Tilt
   -**Physical From**: -135
   -**Physical To**: 135

DMXChannel 2 is entered.

Enter DMXChannel 3 – Dimmer

To enter the third channel:

1. Start with steps **1 to 3**, as described in DMXChannel 2.
2. To enter the third channel, enter in the row DMXChannel 3:
3. -**Coarse**: 5
   -**Default**: 0 %
   -**Highlight**: 100 %
   -Open LogicalChannel. For more information see **step 3** in the type LogicalChannel.

4. -**Attribute**: Dimmer
5. DMXChannel 3 is renamed after the attribute.
6. To control the value of the attribute using the Grand Master, tap in the column **Master.**
   -The **pop-up Select Master** opens.

Select Master

   -Tap Grand.
7. Open ChannelFunction. For more information, see **step 5** in the type ChannelFunction.
   -A**ttribute**: Dimmer
   -**Physical From**: 0
   -**Physical To**: 1

DMXChannel 3 is entered.

Enter DMXChannel 4 – Red

1. Start with steps **1 to 3**, as described in DMXChannel 2.
2. To enter the fourth channel, enter in the row DMXChannel 4:
3. -**Coarse**: 6
   -**Default**: 100 %
   -**Highlight**: 100 %
   -Open LogicalChannel. For more information, see **step 3** in the type LogicalChannel.
4. -**Attribute**: ColorRGB_R
5. DMXChannel 4 is renamed after the attribute.
6. **Master**: none
7. Open ChannelFunction. For more information see **step 5** in the type ChannelFunction.
   -**Attribute**: ColorRGB_R
   -**Physical From**: 0
   -**Physical To**: 1

DMXChannel 4 is entered.

Enter DMXChannel 5 – Green

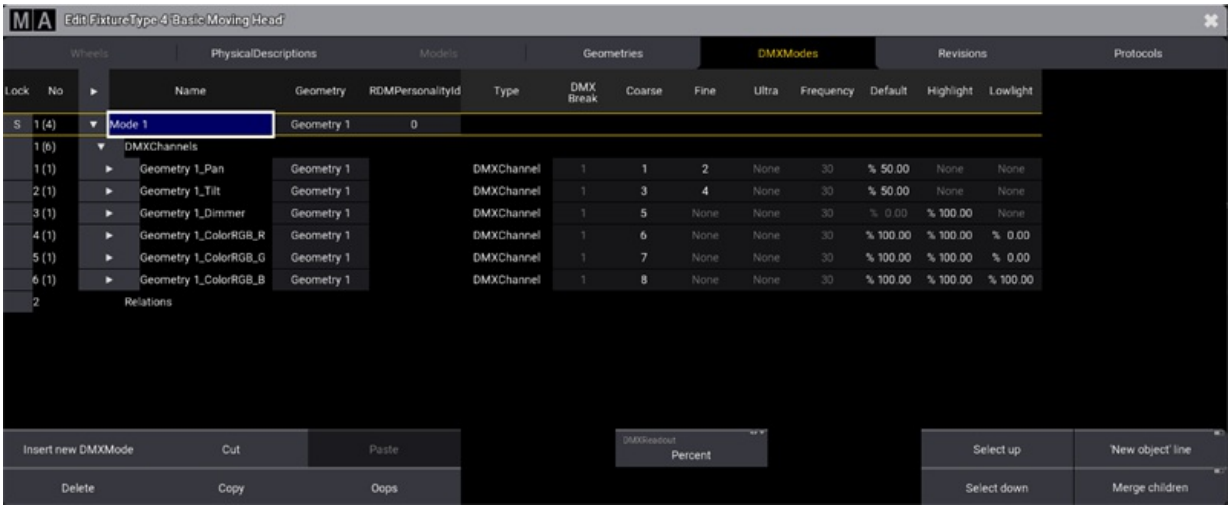
1. Start with steps **1 to 3**, as described in DMXChannel 2.
2. To enter the fifth channel, enter in the row DMXChannel 5:
3. -**Coarse**: 7
   -**Default**: 100 %
   -**Highlight**: 100 %
   -Open LogicalChannel. For more information, see **step 3** in the type LogicalChannel.
4. -**Attribute**: ColorRGB_G
5. DMXChannel 5 is renamed after the attribute.
6. **Master**: none
7. Open ChannelFunction. For more information see **step 5** in the type ChannelFunction.
   -**Attribute**: ColorRGB_G
   -**Physical From**: 0
   -**Physical To**: 1

DMXChannel 5 is entered.

Enter DMXChannel 6 – Blue

1. Start with steps **1 to 3**, as described in DMXChannel 2.
2. To enter the sixth channel, enter in the row DMXChannel 6:
3. -**Coarse**: 8
   -**Default**: 100 %
   -**Highlight**: 100 %
   -Open LogicalChannel. For more information, see **step 3** in the type LogicalChannel.
4. -**Attribute**: ColorRGB_B
5. DMXChannel 6 is renamed after the attribute.
6. **Master**: none
7. Open ChannelFunction. For more information, see **step 5** in the type ChannelFunction.
   -**Attribute**: ColorRGB_B
   -**Physical From**: 0
   -**Physical To**: 1

DMXChannel 6 is entered, and DMX Mode 1 is inserted.



Insert Mode 1

41.2.3. Insert Geometries

Geometry is the physical description of parts of the device.

Requirement:

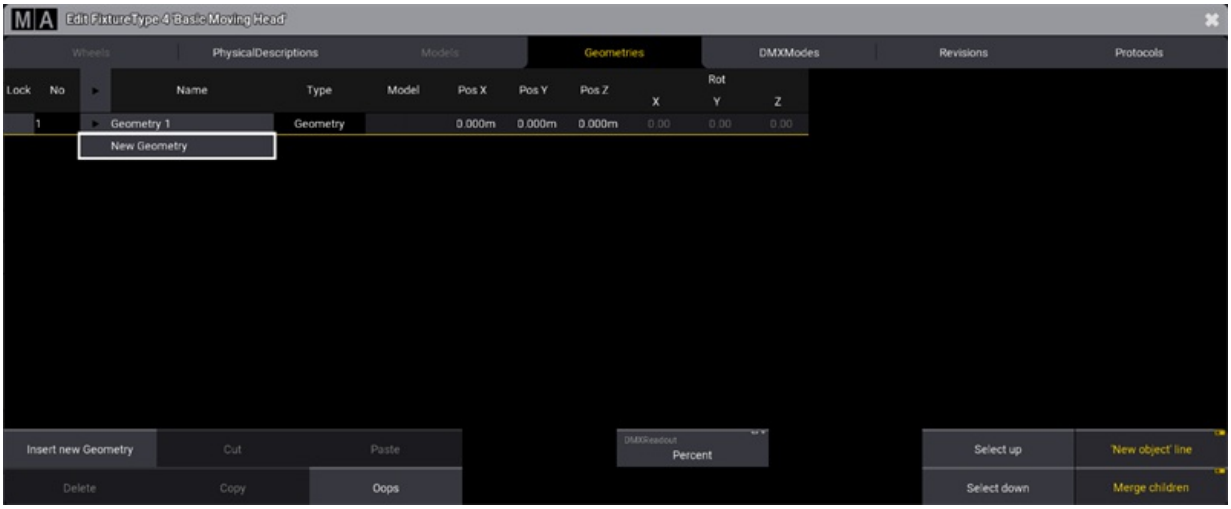
- Insert DMXModes

For more information, see **Insert DMX Modes**.

Geometries are built upon DMX Modes.

1. Tap the tab Geometries.

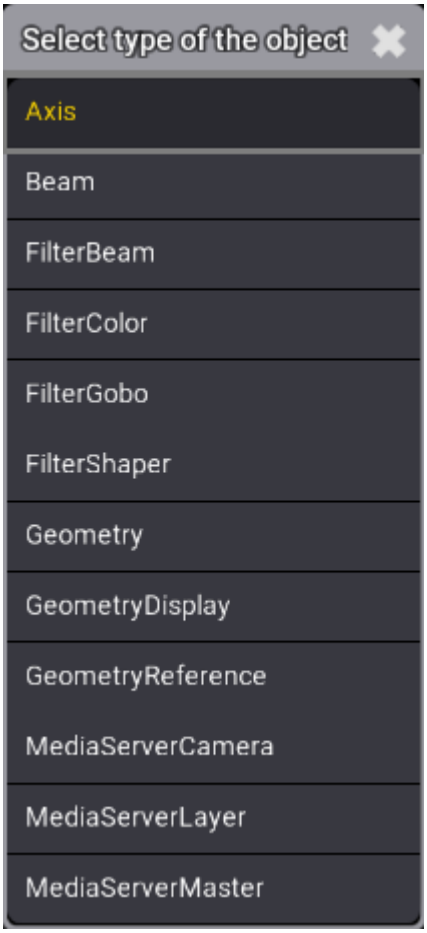
The window Geometries opens.



Open Geometries

2. Select Geometry 1 and rename it Base.
3. Tap to enable 'New Object' line.
4. Open Base and select the Base´s child New Geometry.
5. Tap Insert.

The **pop-up Select type of the object** opens.



Select type

- Select **Axis**.
- Axis is displayed in the column Type.

6. In the column Name, rename Axis 1 to Yoke.

- Press Edit and tap Axis 1, or press and hold Axis 1.
- The virtual keyboard opens.
- Enter Yoke.

7. Expand the cell Yoke.

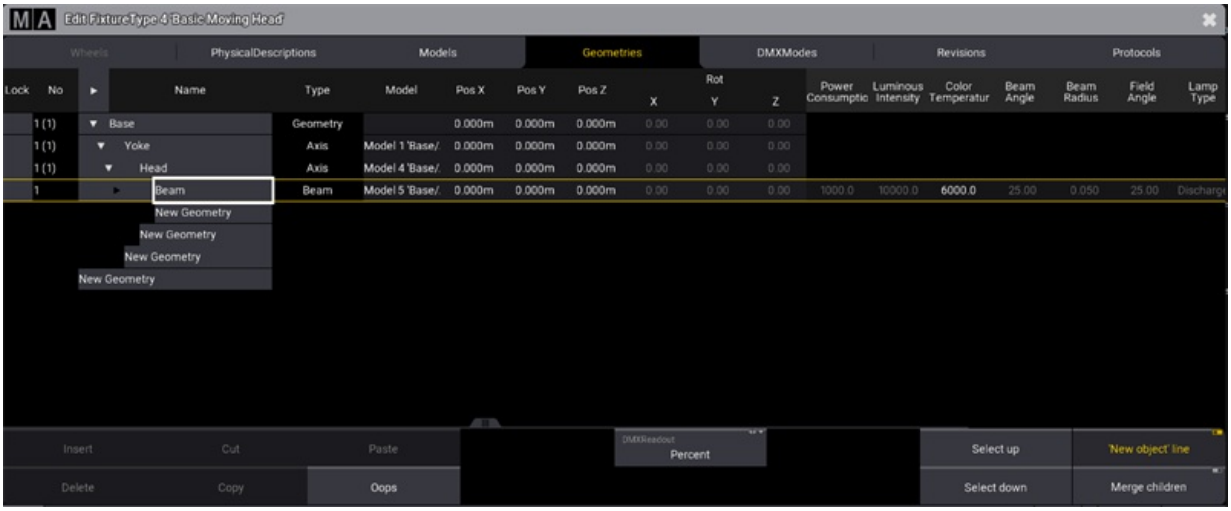
- Yoke´s child New Geometry opens.

- Repeat steps **5 to 6** and rename Axis 1 to Head.

9. Expand the cell Head.

- Head´s child New Geometry opens.
- Repeat steps **5 to 6** selecting Beam.
- Rename Beam 1 to Beam.

Geometries are inserted.



Insert geometry and its children

41.2.4. Insert Models

Each geometry has a separate description of the model.

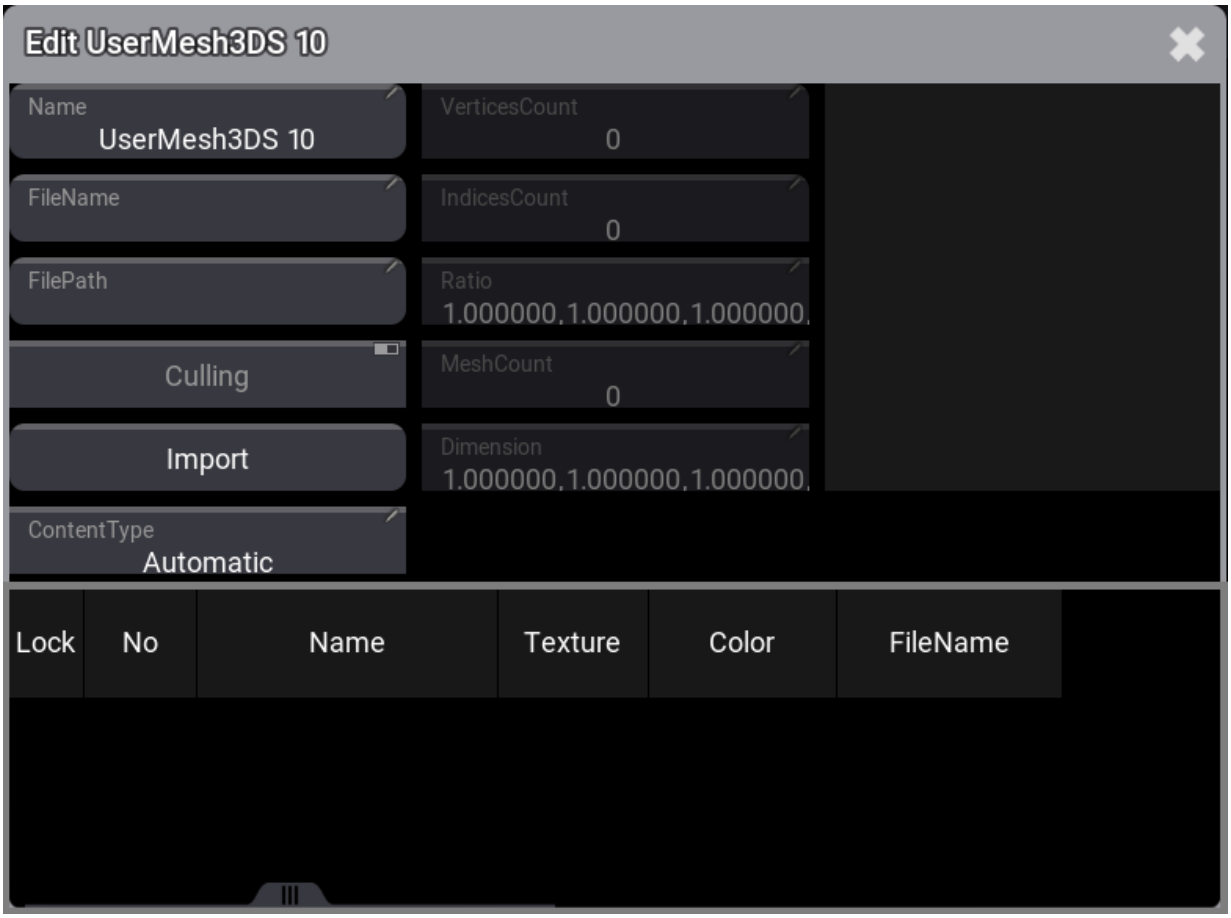
A mesh is a 3ds file (3D image format used by Autodesk 3D Studio).

Meshes are stored in the Meshes pool after they have been imported to the show. See **Add Fixtures To The Show**.

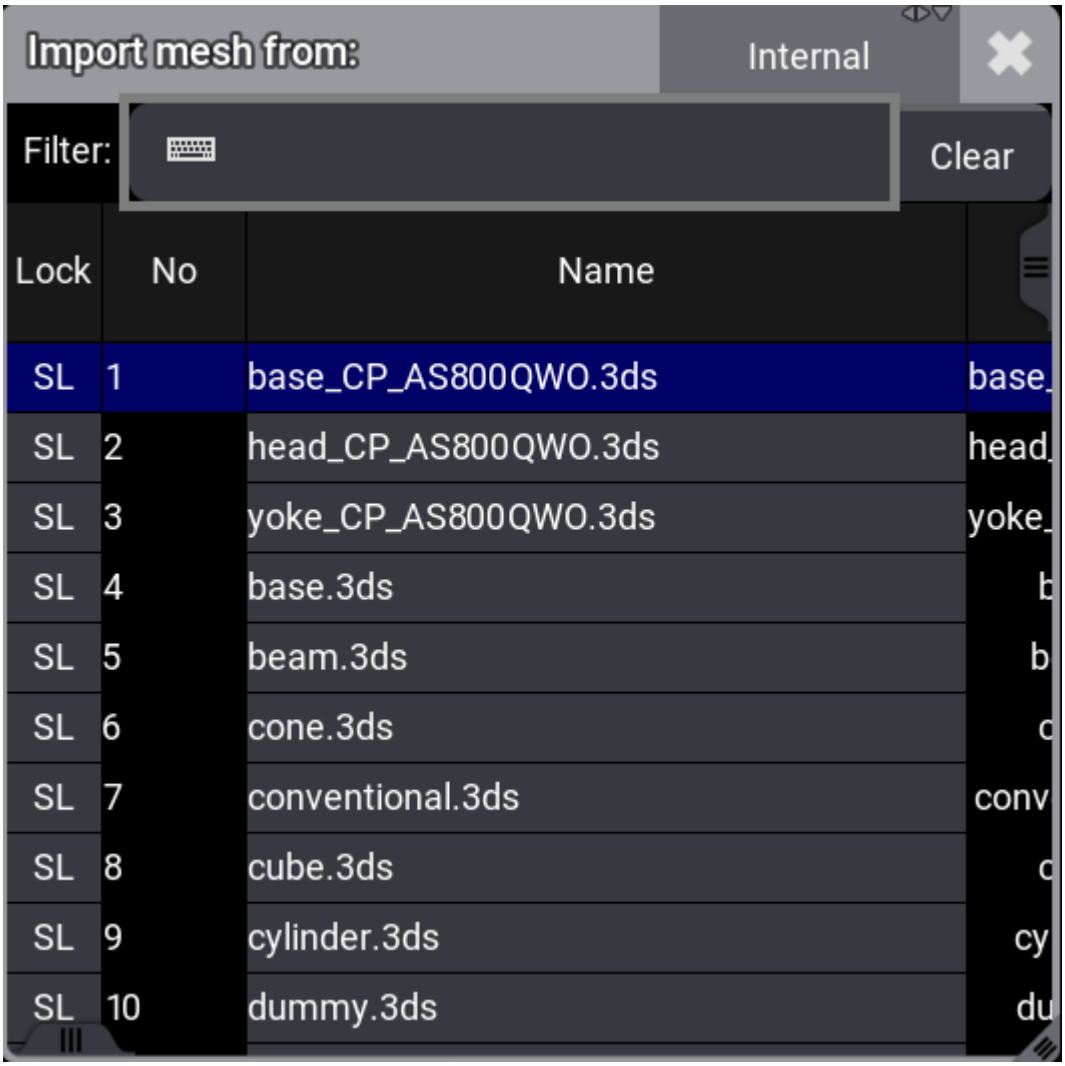
Meshes Pool 1 thru 9 locked

Meshes can also be imported when editing a mesh pool object:

1. Edit an empty mesh pool object using the 2 finger gesture.
2. The **Edit UserMesh3DS** menu opens.



3. Tap Import, the **Import mesh from** menu opens.



4. Select the source from the title bar.
5. Tap the mesh you want to import.

Important:
After they have been imported, meshes must be link to a model of a fixture type.

Hint:
Meshes are automatically added to the mesh pool after importing an MVR file or when a fixture type is imported to the show. For more information, see **MVR**.

Requirement:

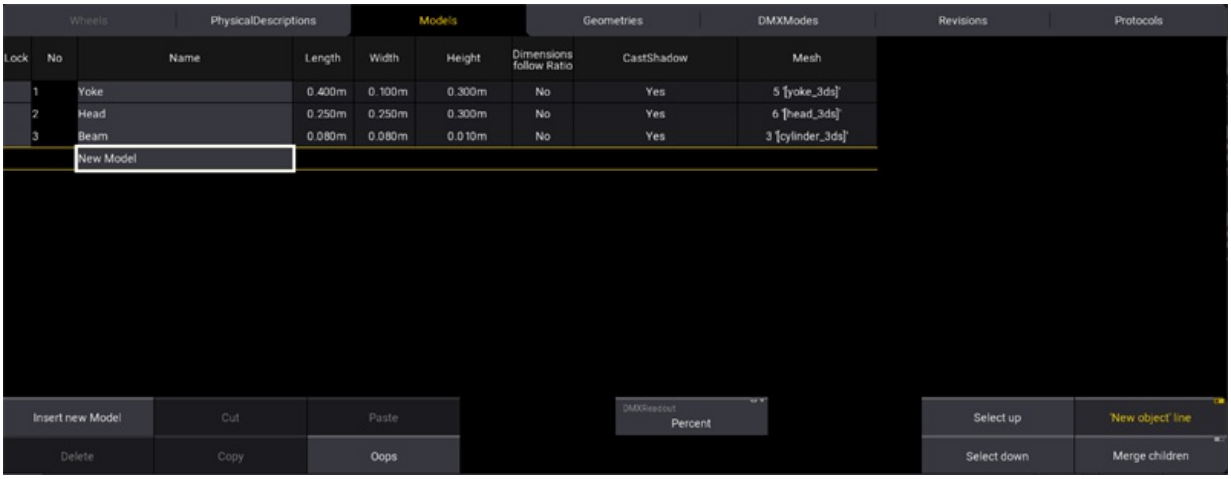
- Insert Geometries

For more information, see **Insert Geometries**.

Models are built upon Geometries.

1. Tap the tab Models.

The menu Models opens.



Open Models

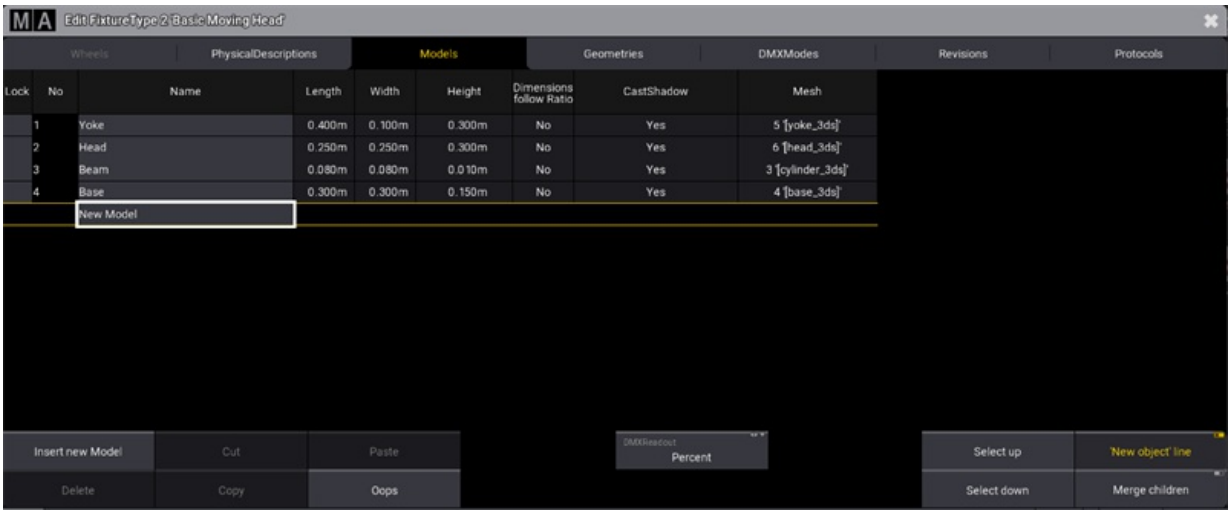
2. Select New Model and tap Insert new Model.

- Model 4 is added.

3. Rename Model 4 Base and set its **Mesh** to [base_3ds].
4. Rename Base/Axis1 Yoke and set its **Mesh** to [yoke_3ds].
5. Rename Base/Yoke/Axis 1 Head and set its **Mesh** to [head_3ds].
6. Rename Base/Yoke/Head/Beam 1 Beam and set its **M**esh to [cylinder_3ds].
7. Enter the measurements of the fixture.

Important:
So that the models do not automatically resize themselves, set **Dimensions follow Ration** to **No** before entering the measurements.

Name	Length	Width	Height
Yoke	0.4 m	0.1 m	0.3 m
Head	0.25 m	0.25 m	0.3 m
Beam	0.08 m	0.08 m	0.01 m
Base	0.3 m	0.3 m	0.150 m



Insert models and the measurements

The models are automatically taken over to the Geometries in the column Model, except for Base as it was inserted directly in the models.

Link Base to the model Base. For more information, see **Link Models to Geometries**.

41.2.5. Link Models to Geometry

The offset of the position depends on the measurements of the models.

Requirement:

- Insert geometries and models

For more information, see **Insert Geometries** and **Insert Models**.

1. Tap the tab Geometries.
2. To link the model Base to geometries, tap, and hold the cell in the column Model.

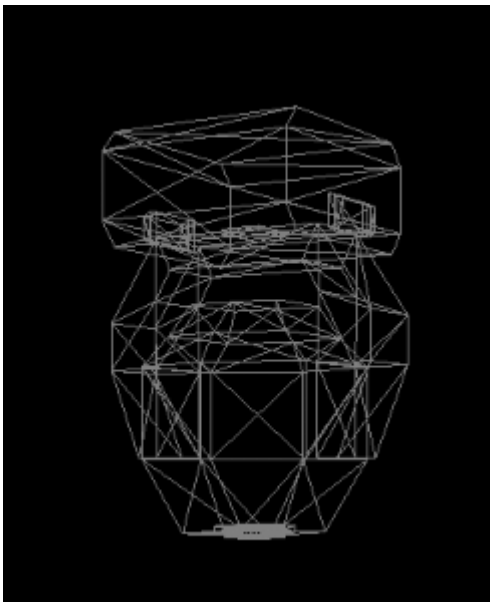
- Pop-up Select Model opens.
- Select Base.

3. Set the offset of position in single geometries.

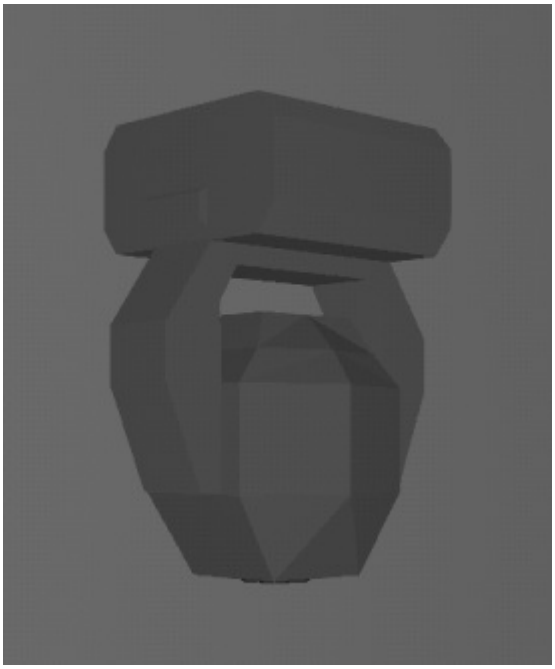
- In Base, leave the default value of Pos Z.
- In Yoke, set Pos Z to -0.265 m.
- In Head, set Pos Z to -0.100 m.
- In Beam, set Pos Z to -0.150 m.

Set the offset of Pos Z

This is the result in the 3D window.



Basic moving head displayed in wireframe in the 3D window



Basic moving head rendered in the 3D window

41.2.6. Link DMX Modes to Geometries

After linking models to geometries, link DMX channels and DMX modes to geometries.

Requirement:

- Link models to geometries

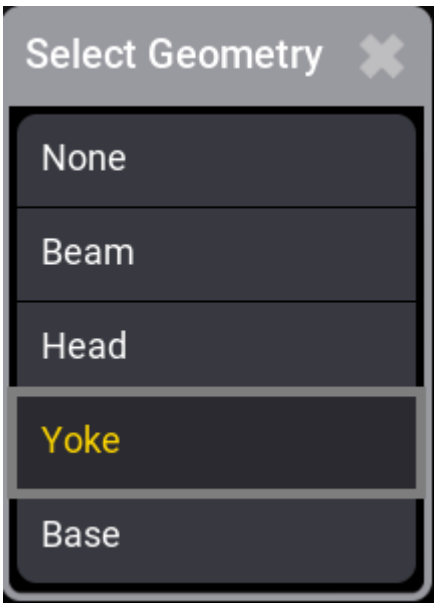
For more information, see **Link Models to Geometries**.

1. Tap the tab DMXModes.

Edit the column Geometry:

2. In the row Yoke_Pan, tap and hold or press Edit and tap the cell.
   Select geometry Yoke.

The **pop-up Select Geometry** opens.

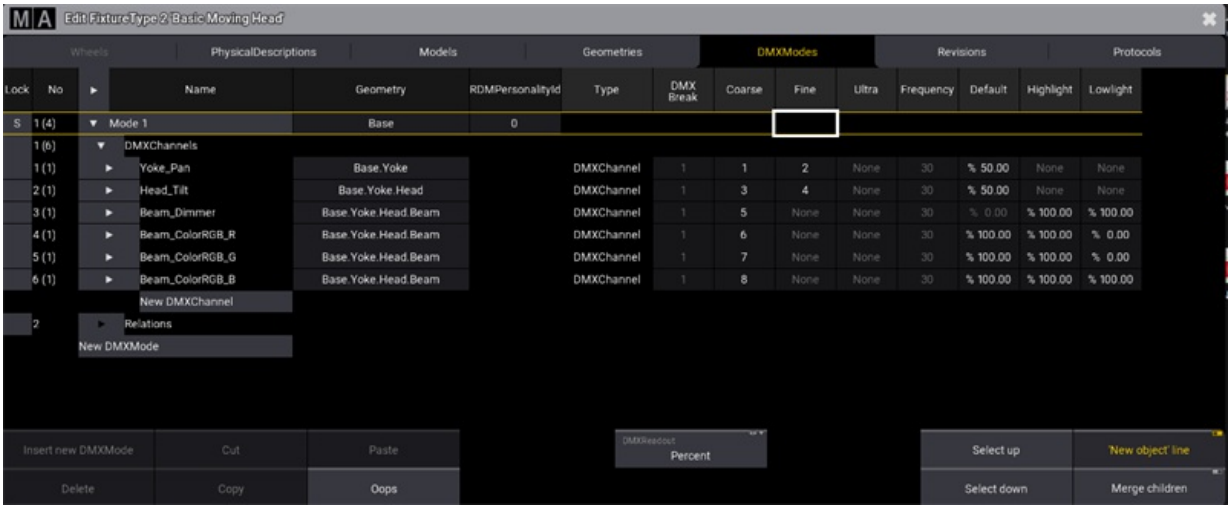


Select geometry

3. In the row Head_Tilt, tap and hold the cell.
   Select geometry Head.
4. In the row Beam_Dimmer, tap and hold the cell.
   Select geometry Beam.
5. In the rows Base_ColorRG_R, Base_ColorRGB_G, and Base_ColorRGB_B, tap and hold the cell.
   Select geometry Beam.

6. In row Mode 1, tap and hold the cell.
   Select geometry Base.

DMX channels and DMX modes are now linked to geometries.



Link geometries

41.3. Export Fixture Types

It is possible to export already existing fixture types and fixture types you built.

Important:
Export grandMA3 fixture types to the internal drive or an external USB drive.

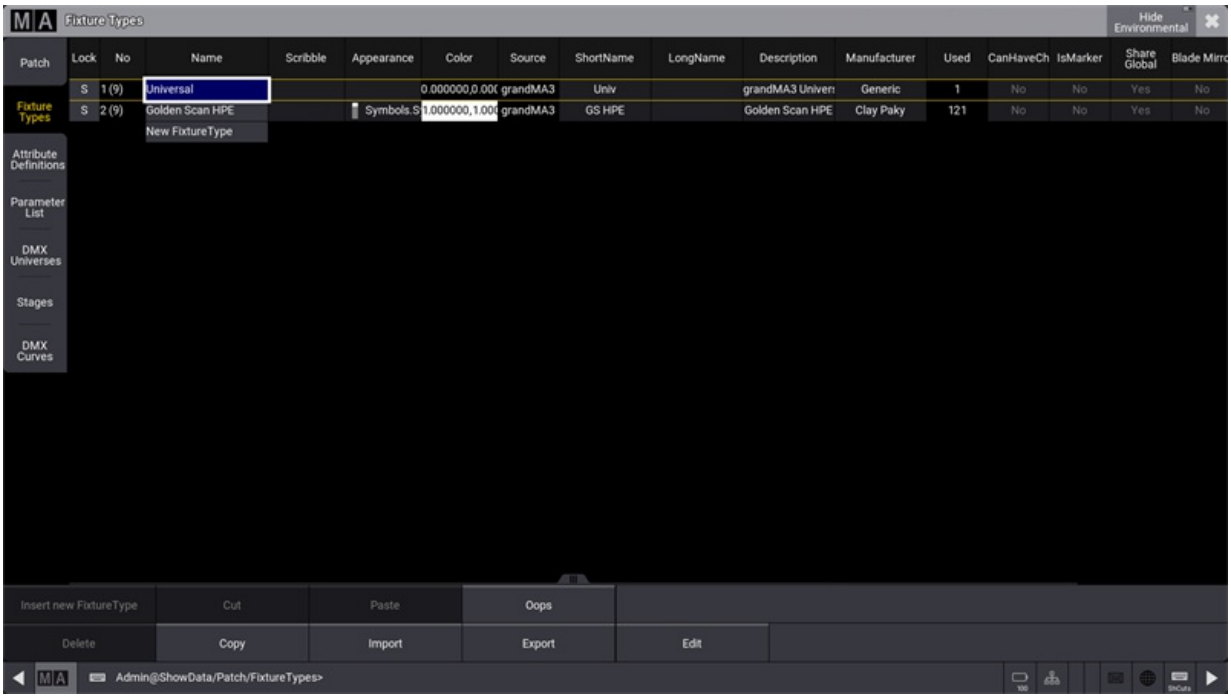
Export from these libraries:

- **User** fixtures
- Fixtures using **GDTF** format

Requirement:

- Access Patch

1. Tap Fixture Types in the bar on the left of the patch dialog.
   The **Fixture Types** menu opens.



Open Fixture Types

2. Select the fixture type.
3. Tap Export.
   The **pop-up Export Fixture Type opens**.

Export a fixture type

4. From the title bar right corner, select the destination drive.
5. Enter a name.

Hint:
If you do not enter a name, the fixture type file will be exported as a default name. That is, manufacturer@name.xml.

6. Tap Export.

- When the selected drive is internal, the fixture type is exported to the computer local drive **(C:)\ProgramData\MALightingTechnology\gma3_library\fixture_types**.
- When the selected drive is external, the fixture type is exported to the external USB Drive **(E:)\grandMA3\library\fixture_types.**

41.3.1. Export GDTF

It is also possible to export GDTF files.

Important:
Export fixture types using GDTF format to the internal drive or an external drive.

For a detailed description of GDTF, see this tutorial **gdtf-share.com/wiki/GDTF_File_Tutorial** and **gdtf-share.com/wiki/GDTF_File_Description**.

Export GDTF from the grandMA3 Console or the grandMA3 onPC

1. Insert a USB flash drive in a grandMA3 console or your PC or laptop.
2. Open the patch dialog and tap Fixture Types.
3. Tap Export.
   Export Fixture Type pop-up opens.
4. Tap GDTF.
   The button is highlighted in yellow.
5. From the title bar in the right corner, select Internal or USB drive.
6. Enter a name and tap Export.

Hint:
If you do not enter a name, the GDTF file will be exported as a default name. That is, manufacturer@name.gdtf.

Export GDTF

The fixture type is exported and located on the USB flash drive in the following folder: **grandMA3/gma3_library/fixturetypes/gdtf**. For more information on the default folder structure, see the **Folder Structure** topic.

Feel free to upload the GDTF file to **www. gdtf-share.com**.

42. File Management

In addition to saving and loading complete show files, the grandMA3 software supports the exchange and portability of smaller portions of show data; for example, a collection of macros or even a single preset.

To maintain a predictable organizational structure, the software uses well-defined sets of folders within the hard drive of the console, the hard drive of a computer running grandMA3 onPC, and any USB drive connected to either a console or an onPC station.

The following topics cover the export and import functions available within the graphical user interface of the grandMA3 software, as well as the default folder structure created and utilized by the software. For more information about the handling of complete show files, see the **Show File Handling** topic.

Important:
The file name must not contain the following characters: \ " $ * ? ^ | / : < > `

42.1. Import / Export Menu

The **Import / Export** menu presents a graphical workflow for exporting objects from the current show file as a smaller file with a minimum of additional show data.

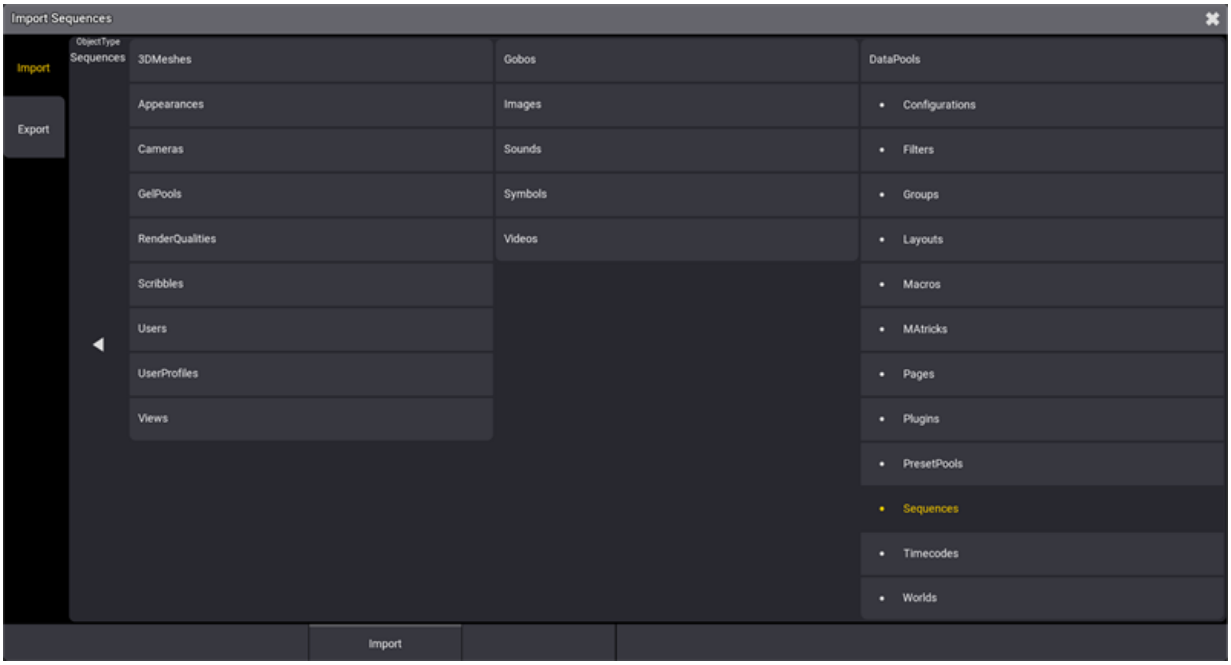
Hint:
When exporting objects that reference other objects in the show, the referenced objects are automatically included in the export. For example, when exporting a macro with a defined appearance, the exported file includes the referenced appearance.

This menu also allows the import of objects from these smaller files into the current show file. For example, this menu provides an easy way to copy a user profile or selection of macros from one show file to another.

Open the **Import / Export** menu using the following steps:

1. Press the Menu key or tap the icon near the top of the **Control Bar**.
2. Tap Import / Export along the bottom row of buttons in the **Menu** pop-up.

Once the main **Import / Export** menu is open, tap the Import or Export tab in the upper-left corner to access the desired interface.

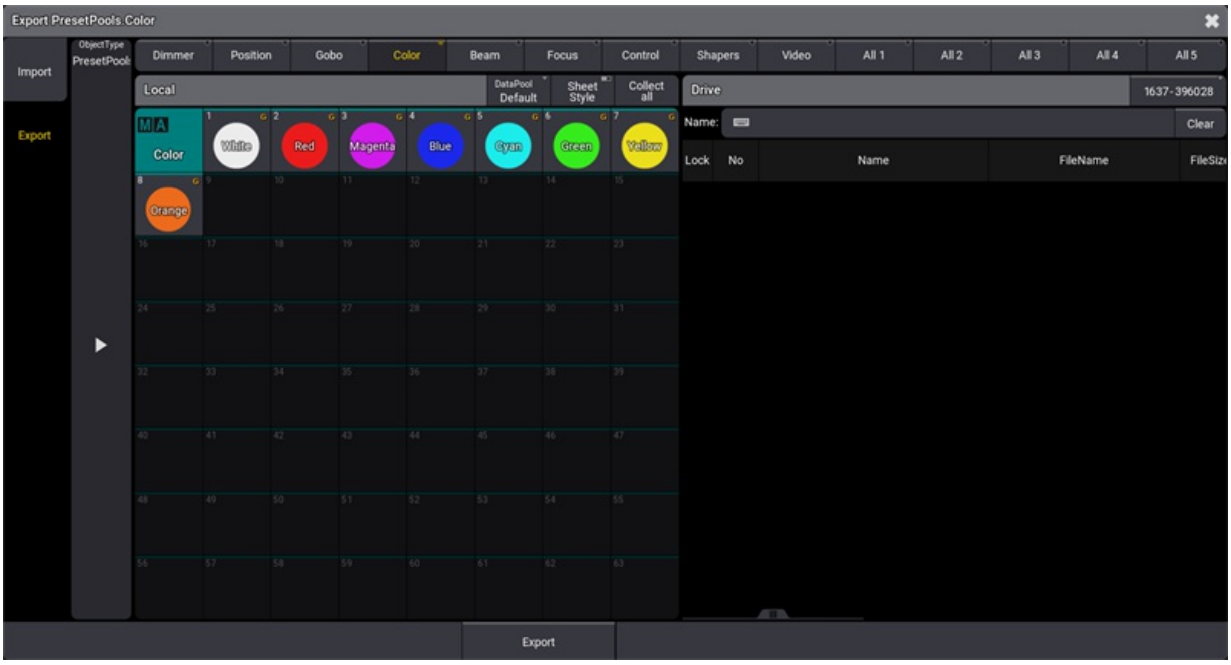


Import / Export menu showing Object Type selection

Object Types

These menus allow for the import and export of a wide variety of object types. Tap ObjectType on the left side of the main area to open or close the full list of available object types:

- **3D Meshes**
- **Appearances**
- **Cameras**
- **Configurations**
- **Data Pools**
- **Filters**
- **Gel Pools**
- Gobos
- **Groups**
- **Images**
- **Layouts**
- **Macros**
- **MAtricks**
- **Pages**
- **Plugins**
- **Preset Pools**
- **Render Qualities**
- **Scribbles**
- **Sequences**
- Sounds
- Symbols
- **Timecodes**
- **Users**
- **User Profiles**
- **Views**
- Videos
- **Worlds**



Export menu prepared to export color presets

The main area of both the **Export** and **Import** menus shows the source of the data on the left and the destination on the right. Therefore, the **Export** menu displays the specified object types from the current show file in the area on the left, in an area titled **Local**, and the contents of the relevant folder of the selected drive on the right, in an area titled **Drive**. In the **Import** menu, these two areas appear on opposite sides.

Drive Selection

To choose a different destination drive for the exported data, tap the Internal button at the right side of the title bar of the **Drive** section or swipe to open a list of available drives. In the **Import** menu, this drive setting defines which drive the console uses as the data source.

Local Section Title Bar Tools

The title bar of the **Local** section includes a few buttons with helpful tools.

- Tap DataPool to cycle through available data pools within the current show file or swipe to open a list of data pools. This option is only available when importing or exporting an **ObjectType** which is incorporated into a **DataPool** within the show file. These **ObjectTypes** are listed below DataPools in the right column of the ObjectType section.
- Tap Sheet Style to display objects in the **Local** section as a sheet with relevant details for each object. Tap Sheet Style again to view objects as they appear in their pools. Some objects, like UserProfiles, never appear in pools. Therefore they will always appear in the sheet format regardless of the state of the Sheet Style button.
- Tap Collect all to select all objects of the current type for export. Once tapped, this button changes to read Uncollect all. Tap this button to deselect all objects. These functions do not appear in the **Import** menu.

Object Sub-Types

Some entries in the ObjectType selection include several collections of objects, which appear as different pools. These include:

- PresetPools: The **Import / Export** menu handles and displays each preset pool individually.
- GelPools: The **Import / Export** menu handles and displays each gel book as an individual pool.

With the PresetPools or GelPools ObjectType selected, a row of radio buttons appears across the top of the main area showing all of the relevant object sub-types. Tap one of these buttons to display the desired pool below.

Multiple Object Selection

Tap any object in the displayed pool to select it for export. A brown border appears around each selected object. To select multiple objects in a pool, tap each of the desired objects. Tap any selected object to deselect it.

When Sheet Style is enabled, select multiple adjacent objects by drawing a lasso over all desired objects. To select just one object, tap only the desired object. To toggle selection for any objects in the sheet without changing the selection status of any other objects, hold Ctrl on the keyboard while tapping the desired objects in the sheet.

Export Workflow

1. Tap the Export tab in the upper-left corner.
2. Set the desired ObjectType.
3. Set the desired destination drive.
4. Select all of the desired objects to export from the **Local** area.
5. Enter the desired file name in the Name field of the **Drive** area.
6. Tap Export at the bottom of the menu.

Import Workflow

1. Tap the Import tab in the upper-left corner.
2. Set the desired ObjectType.
3. Set the desired source drive.
4. Select all of the desired files to import from the **Drive** area.
5. Select the desired destination in the pool in the **Local** area.
6. Tap Import at the bottom of the menu.

Hint:
Certain object types (including **Gobos**, **Images**, **Symbols**, and **Videos**) can be included in **Appearances**. To automatically create new **Appearances** based on these objects as they are importing, toggle on the Create Appearances option before tapping Import.

Import and Export Using Command Line Syntax

For information about exporting and importing show data using command line syntax without the use of these menus, see the **Export keyword** and **Import keyword** topics.

42.2. Folder Structure

The grandMA3 software uses a default set of folders to help maintain a predictable structure when saving data to any internal or external drives.

Hint:
This topic, and other topics in the manual, lists a particular folder name as **gma3_x.y.z**. In this name, the "x.y.z" portion takes the place of the actual installed version number. If the installed version is v1.4.2.1, the name of this folder is **gma3_1.4.2**.

Main Folder

The main folder where all grandMA3 user data is stored by default appears using different labels depending upon the type of drive. This folder can be found under the following names and locations:

On USB Drives:

The main folder on any USB drive connected to either a console or a computer running grandMA3 onPC appears at the root of the drive with the name **grandMA3**.

The full folder structure is created once the drive is connected to the station and selected in the system. There are many ways to select the drive, including the Backup menu, any menu allowing import or export, or the **Select Drive [Drive Number]** command. For more information on the **Drive** keyword, see the **Drive keyword** topic.

On Internal Drives of Computers Running grandMA3 onPC:

The full folder structure is created as part of the software installation process.

The main folder on the internal drive of any computer running grandMA3 onPC appears with the name **MALightingTechnology**.

On computers running macOS, the main folder is found at the following location:
[System HD]/Users/[User Name]/MALightingTechnology

On computers running Windows, the main folder is found at the following location:
C:\ProgramData\MALightingTechnology

On the Internal Drive of the Console:

When logged into the console via SFTP, the console presents the main grandMA3 folder as the root directory, using the name **/**. For more information about SFTP, see the **SFTP Connection** topic.

Shows and Backups

Show files and backup show files are saved in dedicated folders, which appear in the following locations:

On USB Drives:

grandMA3/shared/shows
grandMA3/shared/backups

On Internal Drives of Computers Running grandMA3 onPC:

MALightingTechnology/gma3_x.y.z/shared/shows
MALightingTechnology/gma3_x.y.z/shared/backups

On the Internal Drive of the Console:

/actual/shared/shows
/actual/shared/backups

Warning:
Moving, altering, or deleting any files other than show files or backup show files within in the **gma3_x.y.z** folder of an onPC station or the **actual** folder of a console may make the software unstable or render the system unable to boot.

Library

When exporting smaller portions of data from the show file (including objects such as fixture profiles, user profiles, and macros), these files are saved into the **gma3_library** folder of the selected drive unless an alternate destination is specified during the export. Most object types have dedicated default sub-folders within the library.

The complete list of automatically generated library folders and sub-folders includes:

- agendas
- appearances
- certificates
- colorthemes

- datapools
  - executorconfigurations
  - executorpages
  - filters
  - groups
  - layouts
  - macros
  - matricks
  - plugins
  - presets
  - sequences
  - timecodes
  - worlds
- fixturetyperesources
  - gobos
  - meshes
- fixturetypes
- inout
  - artnet
  - dcremotes
  - dmxremotes
  - midiremotes
  - osc
  - outputconfigurations
  - sacn
- media
  - images
  - sounds
  - symbols
  - videos
- mvr
- netkeys
- patch
  - dmxcurves
  - stages
- scribbles

- userprofiles
  - cameras
  - renderqualities
  - screenconfigurations
  - viewbuttons
  - views
- users

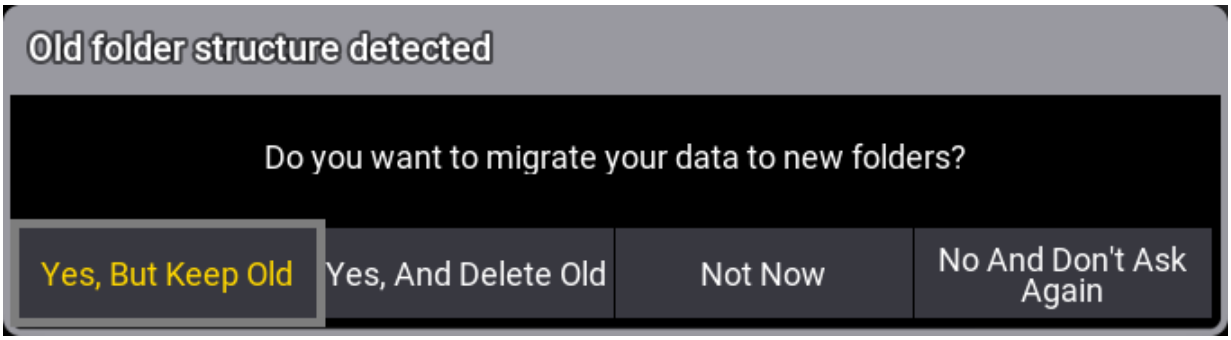
Exceptions to the Default Library Structure

Exporting any objects, which don't have a dedicated sub-folder within the gma3_library, saves the files directly to the **gma3_library** folder unless an alternate destination is specified during the export.

Exporting either the parent of multiple objects, which each have their own dedicated folders, (for example, exporting DMXProtocols as one file rather than separate exports of Art-Net and sACN) or children of objects where those children do not have their own dedicated folder (for example, exporting one or more cues rather than a whole sequence), automatically adds a descriptive sub-extension to the file name while saving the file to the most appropriate folder.

Automatic Folder Structure Migration

The default folder structure used by version 1.5 and later of the grandMA3 software is different than the structure used in previous versions. When the software recognizes a USB drive, which includes the older structure but not the newer structure, the software presents a pop-up with options for automatically migrating the existing data to the new structure.



Old folder structure pop-up

Options in the pop-up include:

- Yes, But Keep Old : Tap to copy files to the new structure while also maintaining copies in the old structure. This allows both older and newer software versions to continue using the drive.
- Yes, And Delete Old : Tap to move files to the new structure and delete the old structure.

- Not Now : Tap to make no changes to the current structure. The next time the USB drive is recognized by the software, the same pop-up will appear.
- No And Don't Ask Again : Tap to make no changes to the current structure. The software will not present this pop-up when the USB drive is recognized in the future.

43. Shut down the System

Before shutting down the system, save the show file via the **Backup menu**.

To start the shutdown procedure, use one of the following options:

Use the software UI

- Tap the icon at the top of the control bar.

The shutdown menu opens:

- To shut down, tap the icon.
- To restart, tap the icon.
- To lock the desk, tap the icon. For more information, see **desk lock**.

Use the power key

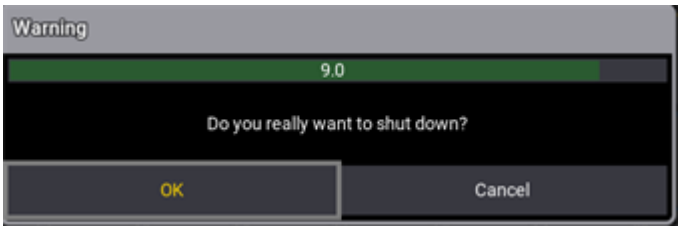
- Press the **power key** on the front panel.

Use a keyword

- **Reboot keyword**
- **Restart keyword**
- **Shutdown keyword**

Shutdown warning pop-up

One of the following warning pop-ups appears:



- Tap OK.

The grandMA3 device shuts down.

Hint:
When using the **Shutdown keyword**, you can choose an option without a shutdown pop-up or without timer.

- To shut down the current station without confirmation, type:

```
User name[Fixture]> Shutdown /nc
```

- To shut down the current station without timer, type:

```
User name[Fixture]> Shutdown /noautoclose
```

45. Troubleshooting

In case your grandMA3 device does not work as expected, the topics below include helpful general troubleshooting steps as well as detailed solutions to some possible issues.

45.1. Clean Start

A clean start clears the current show and all user settings as the grandMA3 software boots. This may be a helpful troubleshooting step in cases where the software boots normally but encounters an issue when the show file loads.

Clean Start a grandMA3 console, Replay Unit, or Processing Unit

Important:
This process relies on the Ctrl or control key commonly found at or near the bottom-left or bottom-right corner of an alpha-numeric keyboard.
For devices without an integrated keyboard, an external USB keyboard is required.

To perform a clean start on a grandMA3 console, replay unit, or processing unit:

1. Make sure the integrated keyboard is functional or connect an external USB keyboard.
2. Boot or reboot the device, watching for the mode selection pop-up.
3. When the mode selection pop-up appears, press and hold the Ctrl key.
4. While holding the Ctrl key, tap the desired mode in the mode selection pop-up.
5. Once the mode selection pop-up disappears, release the Ctrl key.
6. The device completes the boot process with no show file loaded and all user settings reset to factory defaults.

Clean Start grandMA3 onPC

- To perform a clean start on a Windows system, run the grandMA3 onPC x.x.x.x Clean Start app from the MA Lighting folder.
- To perform a clean start on a macOS system, press and hold the option key on the keyboard and then open the grandMA3 onPC application. An alert pop-up appears, confirming the clean start operation. Click Continue to proceed with the clean start. Otherwise, click Cancel.

The software starts with no show file loaded and all user settings reset to factory defaults.

44. Index_