KRAMER

USER MANUAL
MODEL:

DSP-62-AEC
Digital Sound Processor

Contents

Introduction

Welcome to Kramer Electronics! Since 1981, Kramer Electronics has been providing a world of unique, creative, and affordable solutions to the vast range of problems that confront the video, audio, presentation, and broadcasting professional on a daily basis. In recent years, we have redesigned and upgraded most of our line, making the best even better!

Getting Started

We recommend that you:

- Unpack the equipment carefully and save the original box and packaging materials for possible future shipment.

- Review the contents of this user manual.

Go to www.kramerav.com/downloads/DSP-62-AEC to check for up-to-date user manuals, application programs, and to check if firmware upgrades are available (where appropriate).

Achieving Best Performance

- Use only good quality connection cables (we recommend Kramer high-performance, high-resolution cables) to avoid interference, deterioration in signal quality due to poor matching, and elevated noise levels (often associated with low quality cables).

- Do not secure the cables in tight bundles or roll the slack into tight coils.

- Avoid interference from neighboring electrical appliances that may adversely influence signal quality.

- Position your Kramer **DSP-62-AEC** away from moisture, excessive sunlight and dust.

Safety Instructions

Caution:
- This equipment is to be used only inside a building. It may only be connected to other equipment that is installed inside a building.

- For products with relay terminals and GPI\O ports, please refer to the permitted rating for an external connection, located next to the terminal or in the User Manual.

- There are no operator serviceable parts inside the unit.

Warning:
- Use only the power cord that is supplied with the unit.

- Disconnect the power and unplug the unit from the wall before installing.

- Do not open the unit. High voltages can cause electrical shock! Servicing by qualified personnel only.

- To ensure continuous risk protection, replace fuses only according to the rating specified on the product label which is located on the bottom of the unit.

Recycling Kramer Products

The Waste Electrical and Electronic Equipment (WEEE) Directive 2002/96/EC aims to reduce the amount of WEEE sent for disposal to landfill or incineration by requiring it to be collected and recycled. To comply with the WEEE Directive, Kramer Electronics has made arrangements with the European Advanced Recycling Network (EARN) and will cover any costs of treatment, recycling and recovery of waste Kramer Electronics branded equipment on arrival at the EARN facility. For details of Kramer's recycling arrangements in your particular country go to our recycling pages at www.kramerav.com/support/recycling.

Overview

Congratulations on purchasing your Kramer **DSP-62-AEC Digital Sound** Processor.

DSP-62-AEC is a member of the Kramer XSPerience family of DSP products. **DSP-62-AEC** is an advanced, professional 6 x 2 audio matrix switcher that includes multi-channel DSP, AEC (Acoustic Echo Cancellation), HDMI de-embedding, and class compliant USB audio interface. The comprehensive and user-friendly graphic interface makes configuring every detail of your audio system intuitive and easy.

DSP-62-AEC provides exceptional quality, advanced and user-friendly operation, and flexible connectivity.

Exceptional Quality

- Advanced Audio Matrix Switcher – Professional, studio grade signal conversion technology.
- Teleconference Optimized – Features AEC (Acoustic Echo Cancellation), that prevents the microphone from picking up the far-end echoed speech, so you only share the audio spoken directly into the microphone.
- Max. Video Resolution – 4K@60Hz (4:4:4).
- Audio De-embedding – De-embeds the audio signal from the HDMI input for routing to any of the outputs.
- Programmable – Supports up to 10 global presets, 10 mixer snapshot presets and 10 mixer presets per system preset.
- HDMI Support – HDR, CEC, ARC, 4K@60Hz, 3D, Deep Color, x.v.Color™, 7.1 PCM, Dolby TrueHD, DTS–HD.

Advanced and User-friendly Operation

- Wide Range of I/O Formats – Includes 2 HDMI inputs, 1 unbalanced stereo analog input & 4 balanced analog audio inputs, 2 balanced analog audio outputs, 1 HDMI output, and 1 bidirectional USB plug & play audio port.
- Reliable PoE (Power over Ethernet) Powering – Accepts power from a remote PoE provider with optional mains powering from connected power adapter.
- Multi-Channel Processing – Provides DSP (Digital Sound Processing) that enables simultaneous processing of all input and output signals.

- Easy Installation – Compact DemiTOOLS® fan-less enclosure for surface mounting, side-by-side mounting of 2 units in a 1U rack space with the recommended rack adapter or fit in a Kramer T-BUS.

- Easy, Cost-Effective Maintenance – LED indicators for main power, line in/out, mic in, clipping, and HDMI input selection, enable easy local maintenance and troubleshooting. Local firmware upgrade via the RS-232 port ensures lasting, field-proven deployment.

- Intuitive and Comprehensive Configuration and Control – Via a powerful, user-friendly graphic interface, set volume (gain and attenuation) and DSP per input; execute routing and select line in, mic in, phantom power or line out on each port; configure master level and more. Control signal routing, volume and other basic settings using API commands via RS-232 communication transmitted by a PC, touch screen system or other serial controller.

Flexible Connectivity

- 2 HDMI inputs with selection buttons on the front panel.

- 1 unbalanced stereo audio source.

- 1 USB Type B bi-directional host port.

- 1 HDMI output.

- 4 balanced audio inputs (mono or mic level).

- One stereo balanced output.

Typical Applications

DSP-62-AEC is ideal for the following typical applications:

- Huddle spaces.

- Small and medium sized meeting rooms.

- Classrooms.

Controlling your DSP-62-AEC

Control your **DSP-62-AEC** directly via the front panel HDMI select buttons:

- By RS-232 serial commands transmitted by a touch screen system, PC, or other serial controller.

- Via the Ethernet using built-in user-friendly Web pages.

Defining DSP-62-AEC Digital Sound Processor

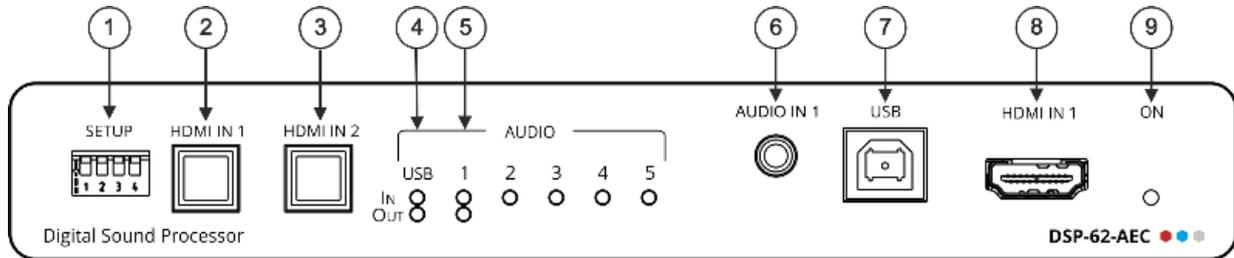This section defines **DSP-62-AEC** front and rear panels.

Figure 1: DSP-62-AEC Digital Sound Processor Front Panel

#	Feature		Function
1	SETUP DIP-switches		For factory use only.
2	HDMI IN 1 Button		Press to select HDMI IN 1 input (on the front panel).
3	HDMI IN 2 Button		Press to select HDMI IN 2 input (on the rear panel).
4	AUDIO LEDs	USB IN	Lights green when an audio signal is detected on the input (for example, if **DSP-62-AEC** operates as a speaker to a PC).
		USB OUT	Lights green when an audio signal is detected on the output (for example, when **DSP-62-AEC** operates as a microphone for a PC).
5		IN (1 to 5)	Lights green when a signal is present, lights red when clipping is detected. For IN 2 to IN 5, when connecting a microphone and setting to Mic mode via the embedded Web pages, lights blue.
		OUT	Lights green when a signal is present, lights red when clipping is detected.
6	AUDIO IN 1 3.5mm Mini Jack		Connect to an unbalanced stereo audio source.
7	USB Type B Bi-Directional Host Port		Connect to an audio source or acceptor.
8	HDMI IN 1 Connector		Connect to an HDMI source.
9	ON LED		Lights green when the device is powered.
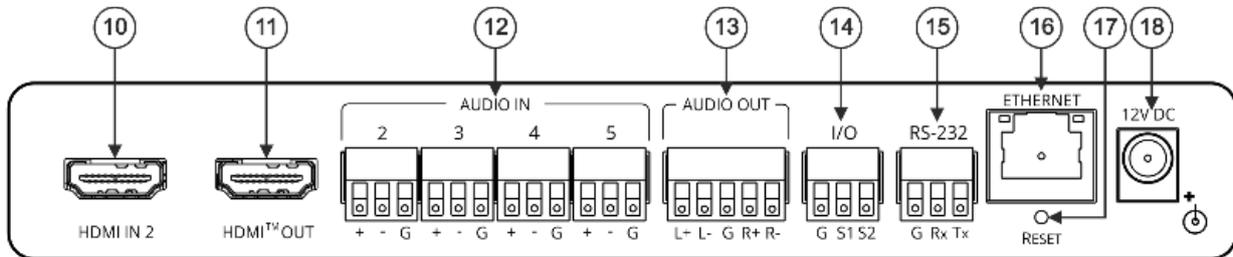
Figure 2: DSP-62-AEC Digital Sound Processor Rear Panel

#	Feature	Function
10	HDMI IN 2 Connector	Connect to an HDMI source.
11	HDMI™ OUT Connector	Connect to an HDMI acceptor.
12	AUDIO IN 3-pin Terminal Block Connectors (2 to 5)	Connect to up to 4 mono balanced audio sources (mono or mic level with selectable 48V).
13	AUDIO OUT 5-pin Terminal Block Connector	Connect to a stereo balanced audio acceptor.
14	I/O Terminal Block Connectors (S1 to S2)	For future use.
15	RS-232 (G, Rx, Tx) Terminal Block Connector	Connect to a PC/serial controller.
16	ETHERNET RJ-45 Connector	Connect to a PC via a LAN.
17	RESET Recessed Button	Press and hold while powering the device to reset to factory default values, including IP settings.
18	12V DC Power Connector	Connect to the power supply and to the mains electricity.

Mounting DSP-62-AEC

This section provides instructions for mounting **DSP-62-AEC**. Before installing, verify that the environment is within the recommended range:

- Operation temperature – 0° to 40°C (32 to 104°F).
- Storage temperature – -40° to +70°C (-40 to +158°F).
- Humidity – 10% to 90%, RHL non-condensing.

Caution:
- Mount **DSP-62-AEC** before connecting any cables or power.

Warning:
- Ensure that the environment (e.g., maximum ambient temperature & air flow) is compatible for the device.
- Avoid uneven mechanical loading.
- Appropriate consideration of equipment nameplate ratings should be used for avoiding overloading of the circuits.
- Reliable earthing of rack-mounted equipment should be maintained.

Mount DSP-62-AEC in a rack:

- Use the recommended rack adapter
  (see www.kramerav.com/product/DSP-62-AEC).

Mount DSP-62-AEC on a surface using one of the following methods:

- Attach the rubber feet and place the unit on a flat surface.
- Fasten a bracket (included) on each side of the unit and attach it to a flat surface. For more information go to www.kramerav.com/downloads/DSP-62-AEC.
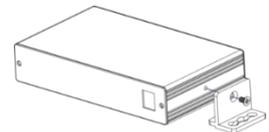
Mount DSP-62-AEC inside a TBUS (for example, the TBUS-10XL):

- Use the designated TBUS frame, to mount **DSP-62-AEC** inside the **TBUS-10XL** (see www.kramerav.com/downloads/TBUS-10XL).

Connecting DSP-62-AEC

Always switch off the power to each device before connecting it to your **DSP-62-AEC**. After connecting your **DSP-62-AEC**, connect its power and then switch on the power to each device.
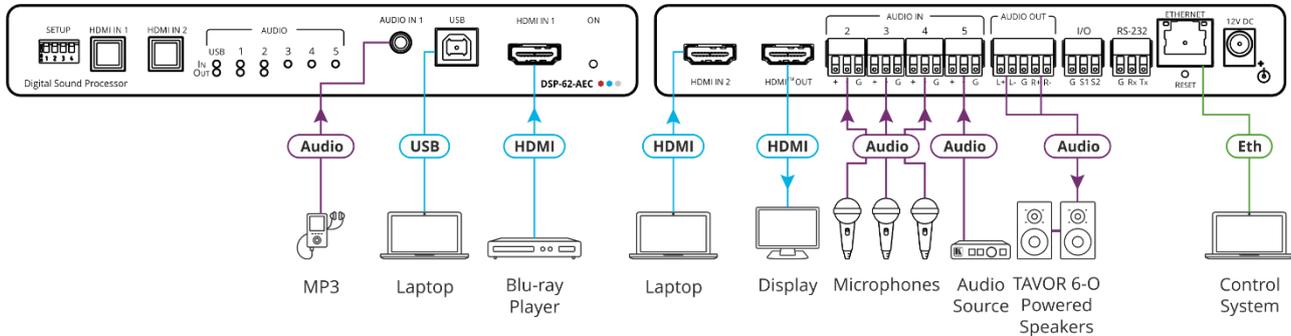
Figure 3: Connecting to the DSP-62-AEC

To connect DSP-62-AEC as illustrated in the example in Figure 3:

1. Connect an unbalanced stereo audio source (for example, an MP3 device) to the AUDIO IN 1 3.5mm mini jack ⑥.

2. Connect a USB source (for example, a laptop) to the USB Host port ⑦.

3. Connect an HDMI source (for example, a Blu-ray player) to the HDMI IN 1 connector ⑧.

4. Connect an HDMI source (for example, a laptop) to the HDMI IN 2 connector ⑩.

5. Connect up to 4 balanced mono audio sources (for example, microphones) to the AUDIO IN 3-pin terminal block connectors ⑫.

6. Connect the HDMI OUT connector ⑪ to an HDMI acceptor (for example, a display).

7. Connect the AUDIO OUT 5-pin terminal block connector ⑬ to a balanced stereo acceptor (for example, a powered speaker).

8. Connect a control system to the ETHERNET RJ-45 port ⑪.

9. Connect the 12V DC power adapter to **DSP-62-AEC** and to the mains electricity (not shown in Figure 3).

Connecting the Output to a Balanced/Unbalanced Stereo Audio Acceptor

The following are the pinouts for connecting the output to a balanced or unbalanced stereo audio acceptor:
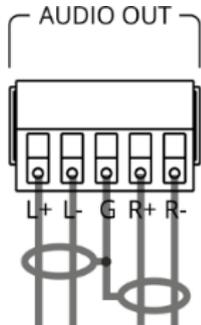
Figure 4: Connecting to a Balanced Stereo Audio Acceptor
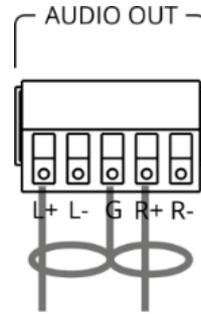
Figure 5: Connecting to an Unbalanced Stereo Audio Acceptor

Connecting to DSP-62-AEC via RS-232

You can connect to **DSP-62-AEC** via an RS-232 connection ⑮ using, for example, a PC.

DSP-62-AEC features an RS-232 3-pin terminal block connector allowing the RS-232 to control **DSP-62-AEC**.

Connect the RS-232 terminal block on the rear panel of **DSP-62-AEC** to a PC/controller, as follows:

From the RS-232 9-pin D-sub serial port connect:

- Pin 2 to the TX pin on the **DSP-62-AEC** RS-232 terminal block

- Pin 3 to the RX pin on the **DSP-62-AEC** RS-232 terminal block

- Pin 5 to the G pin on the **DSP-62-AEC** RS-232 terminal block
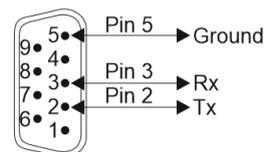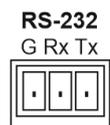
Operating and Controlling DSP-62-AEC

Using the Front Panel

DSP-62-AEC front panel includes:

- Two selection buttons for HDMI IN 1 and HDMI IN 2.

- One USB IN LED ④ to indicate that an audio source is received from the USB host port (green) and one USB OUT ④ LED to indicate that an audio signal is sent to an acceptor.

- 5 Audio IN LEDs to indicate that a signal is present (green), clipping is detected (red), and for LEDs 2 to 5, that a microphone is connected (blue).

- One AUDIO OUT LED to indicate that a signal is present (green) or clipping is detected (red).

- SETUP DIP-switches ① FOR FACTORY USE ONLY.

Operating via Ethernet

You can connect to **DSP-62-AEC** via Ethernet using either of the following methods:

- Directly to the PC using a crossover cable (see Connecting Ethernet Port Directly to a PC on page 9).

- Via a network hub, switch, or router, using a straight-through cable (see Connecting Ethernet Port via a Network Hub on page 11).

ⓘ   If you want to connect via a router and your IT system is based on IPv6, speak to your IT department for specific installation instructions.

Connecting Ethernet Port Directly to a PC

You can connect the Ethernet port of **DSP-62-AEC** directly to the Ethernet port on your PC using a crossover cable with RJ-45 connectors.

ⓘ   This type of connection is recommended for identifying **DSP-62-AEC** with the factory configured default IP address.

After connecting **DSP-62-AEC** to the Ethernet port, configure your PC as follows:

1. Click **Start** > **Control Panel** > **Network and Sharing Center**.

2. Click **Change Adapter Settings**.

3. Highlight the network adapter you want to use to connect to the device and click **Change settings of this connection**.

The Local Area Connection Properties window for the selected network adapter appears as shown in [Figure 6](#).
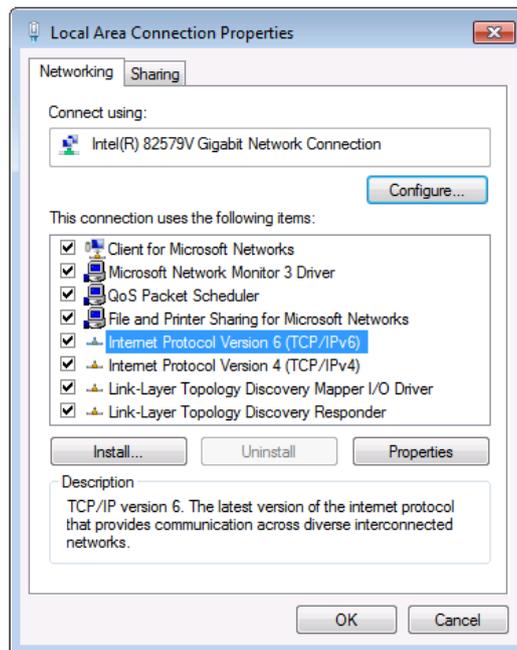
Figure 6: Local Area Connection Properties Window

4. Highlight either **Internet Protocol Version 6 (TCP/IPv6)** or **Internet Protocol Version 4 (TCP/IPv4)** depending on the requirements of your IT system.

5. Click **Properties**.
   The Internet Protocol Properties window relevant to your IT system appears as shown in [Figure 7](#) or [Figure 8](#).
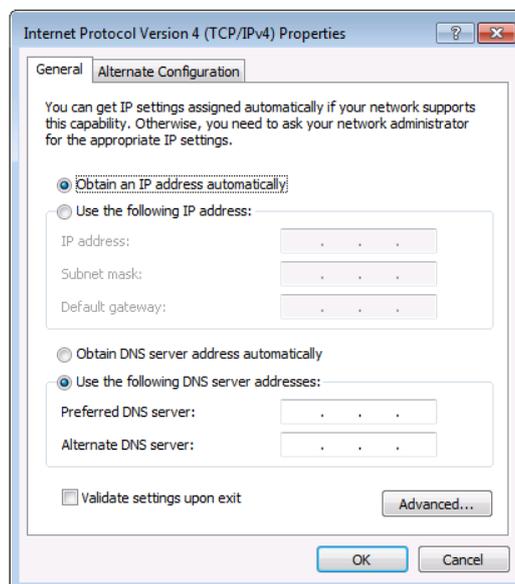
Figure 7: Internet Protocol Version 4 Properties Window
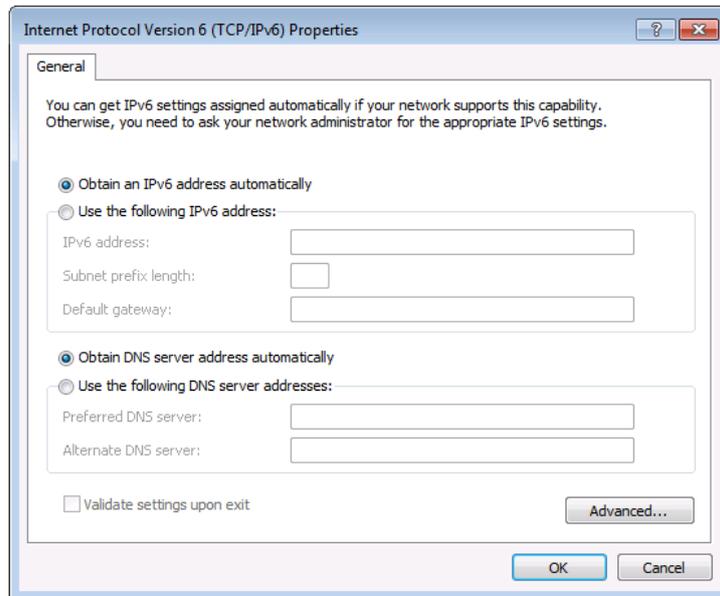
Figure 8: Internet Protocol Version 6 Properties Window

6. Select **Use the following IP Address** for static IP addressing and fill in the details as shown in Figure 9.

   For TCP/IPv4 you can use any IP address in the range 192.168.1.1 to 192.168.1.255 (excluding 192.168.1.39) that is provided by your IT department.
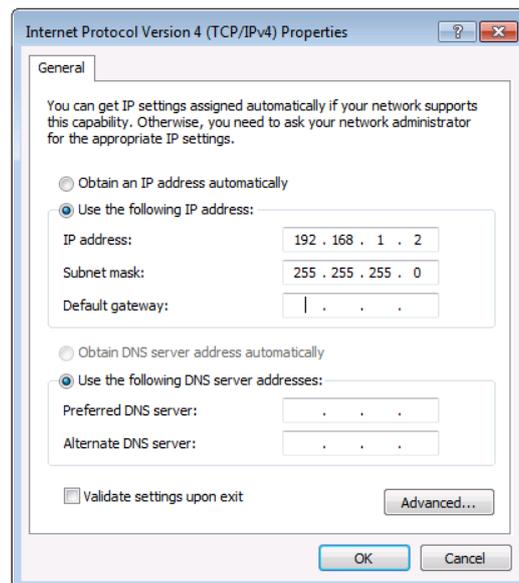
Figure 9: Internet Protocol Properties Window

7. Click **OK**.

8. Click **Close**.

Connecting Ethernet Port via a Network Hub or Switch

You can connect the Ethernet port of **DSP-62-AEC** to the Ethernet port on a network hub or using a straight-through cable with RJ-45 connectors.

Configuring Ethernet Port

You can set the Ethernet parameters via the embedded Web pages.

Using Embedded Webpages

The **DSP-62-AEC** can be operated remotely using the embedded webpages. The webpages are accessed using a Web browser and an Ethernet connection (see Browsing the DSP-62-AEC Webpages on page 12).

Before attempting to connect:

- Perform the procedures in Operating via Ethernet on page 9.

- Ensure that your browser is supported.

The following operating systems and Web browsers are supported:

Operating Systems	Versions
Windows 7	Chrome
Windows 10	Chrome
Mac	Chrome

(i)   Some features might not be supported by some cellphone operating systems.

The **DSP-62-AEC** webpage enables performing the following functions:

- Using the Top Status Bar on page 13.

- Processing Audio Signals on page 16.

- Routing Inputs to Outputs on page 31.

- Mixing Audio Signals on page 35.

- Defining Audio Settings on page 39.

- Defining Video Settings on page 40.

- Defining General Settings on page 41.

- Viewing Device Information on page 50.

(i)   Some of the same tasks can be carried out via DSP, Matrix and Mixer pages, for your convenience. For example, you can link analog input and output pairs through any of these 3 pages.

Browsing the DSP-62-AEC Webpages

To browse the **DSP-62-AEC** webpages:

1. Open your Internet browser.

2. Type the IP Address of the device in the Address bar of your browser. For example, the default IP Address:

   http://192.168.1.39

3. The authentication page appears.

4. Enter the Username and Password (Admin/Admin, by-default):
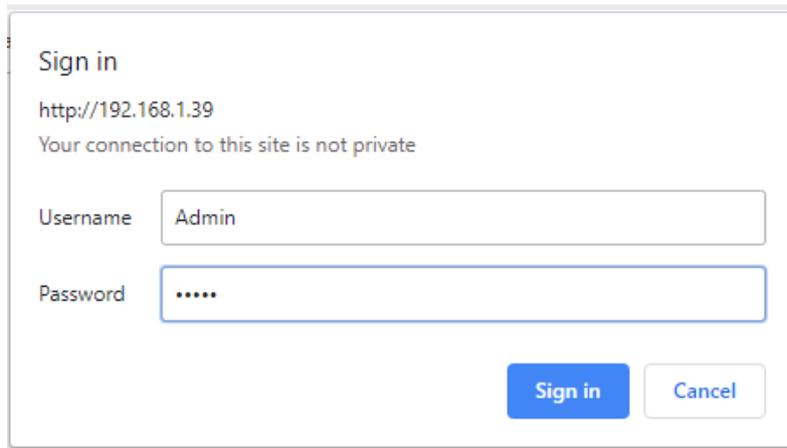
Figure 10: Embedded Webpages Authentication

5. Click **Sign in**.
   The Main webpage appears.
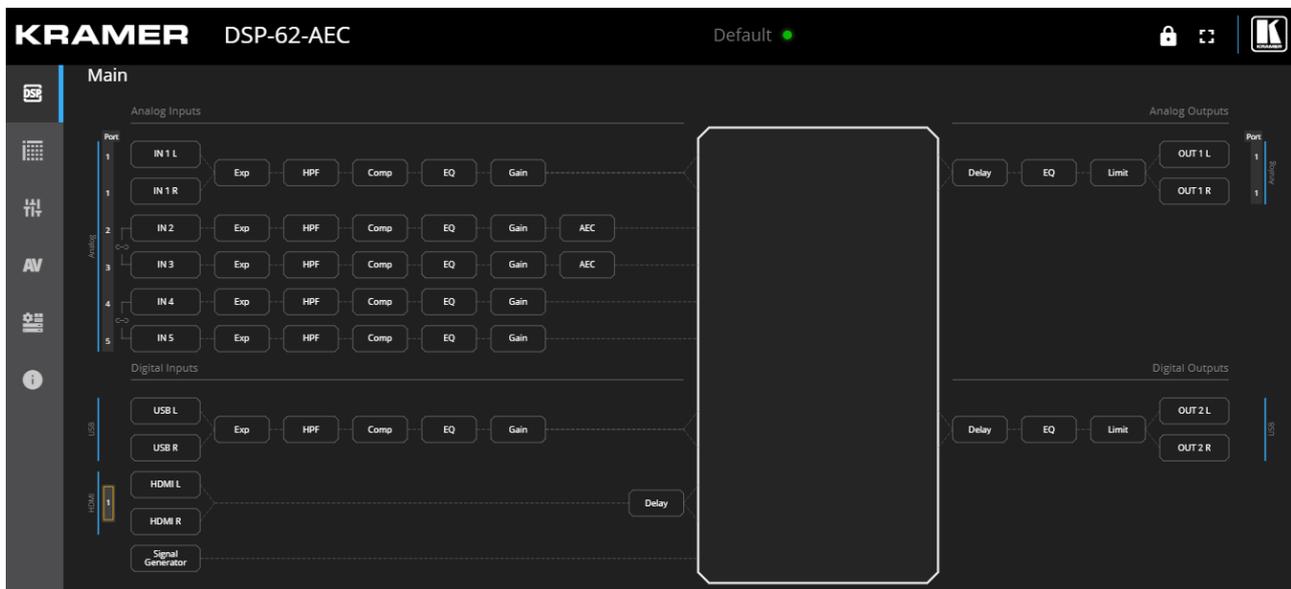
Figure 11: DSP-62-AEC Main Page with the Navigation List on Left

6. Click the desired item in the navigation pane to set and control the device.

Using the Top Status Bar

Use the top status bar to perform the following functions:

- Viewing Current Analog I/O Configuration and Preset Name on page 14.

- Changing Security Settings on page 14.

- Entering/exiting full-screen display view by clicking the display-view icon (⊞ / ⊞).

Viewing Current Analog I/O Configuration and Preset Name

The center of the menu bar in every webpage shows the analog I/O setup, the preset name and the status of the setup.

The indication light displays:

- Green when the actual settings are unmodified.
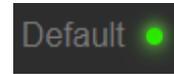
Figure 12: Analog and/or Preset Status Unmodified

- Yellow when actual settings have been modified and not saved into a preset.
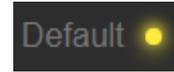
Figure 13: Analog and/or Preset Status modified

To save a modified preset (yellow indication light):

1. Click the preset status area. The A/V settings page appears (see Figure 46).

2. Follow the instructions in Defining Audio Settings on page 39.

Changing Security Settings

You can easily disable or enable the webpages security using the lock icon. When security is disabled, you do not need to enter a password to access the webpages. When security is enabled, you do. For information about the default login credentials, see Default Communication Parameters on page 53. For information about changing the default login credentials, see Setting Access Security on page 47.

To disable security settings:

1. Click the lock icon (🔒) indicating that security is enabled.
The following message appears:
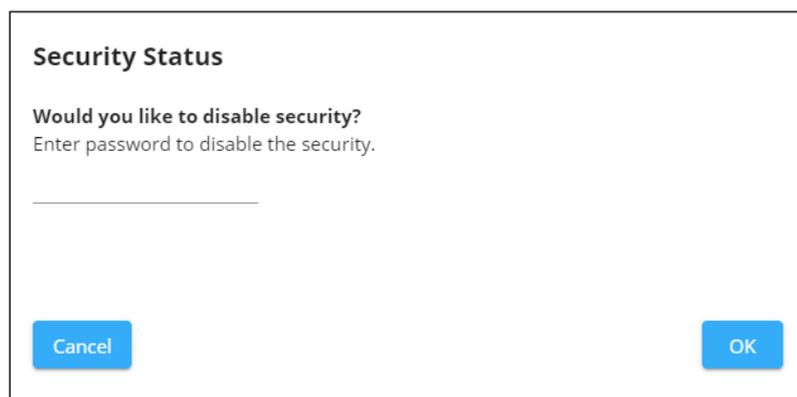
Figure 14: Disabling Security Message

2. Type the current password (Admin, by default).

3. Click **OK**.

Security is disabled.

To enable security settings:

- Click the security disabled icon (🔓).

Viewing the Matrix Area

The matrix area in the DSP page shows the inputs that are currently routed to the outputs.
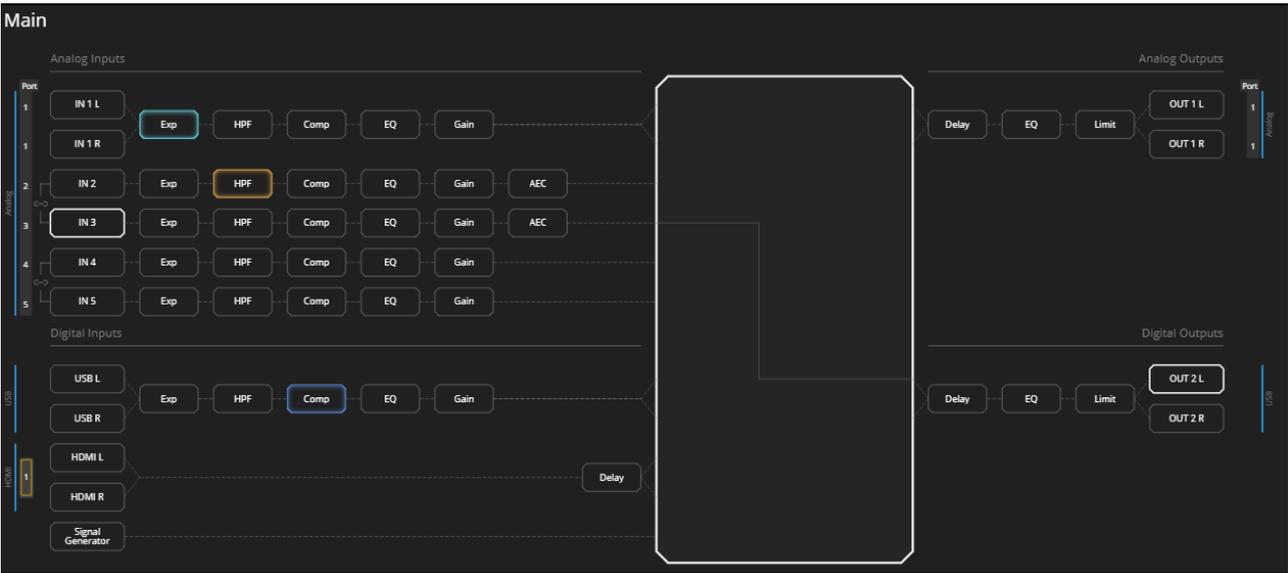
Figure 15: DSP Page – Matrix Area

Clicking an IN or OUT button or a signal processing module (for example, Exp), highlights the routing path.
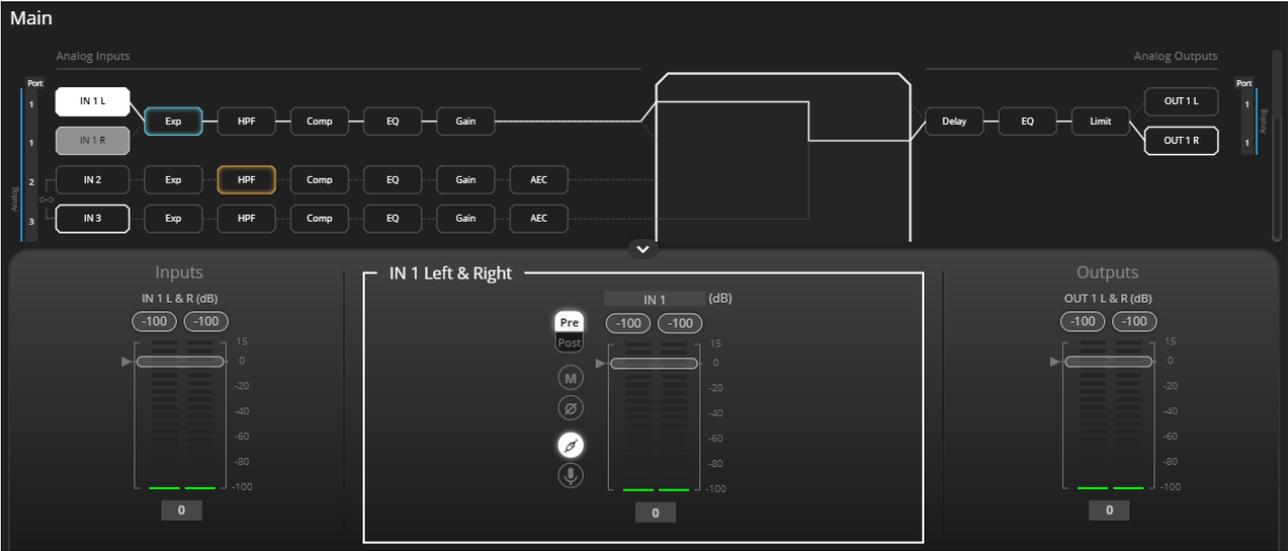
Figure 16: Matrix Area – Routing Path

When opening the processing view, the sliders of the Inputs routed to the outputs appear.
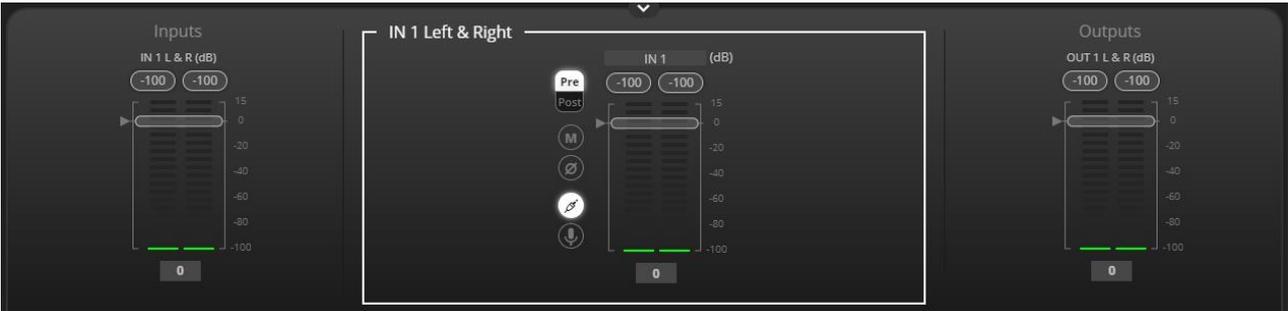
Figure 17: Processing View – Inputs Routed to Outputs

Processing Audio Signals

Use the DSP page to process the input and output signals and present an overall view of your session, including analog and digital in-out connections (in the Matrix area), using pre-matrix and post-matrix modules.

In general:

- Click the Matrix area to enter the Matrix page (see).

- Click an input, output or any module to open its process view and configure that item.

The DSP page enables performing the following functions:

-

-

Linking Analog Inputs and Outputs

Analog inputs and outputs can be linked in predefined pairs to balance stereo analog sources and acceptors. When linked, signal chain modules are set for both channels simultaneously.

To link an analog audio pair:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

   (i) You can also link audio analog audio pairs via the Matrix page, and Mixer page.

2. Click the link on the side of the ports (IN 4 and IN 5 in this example).
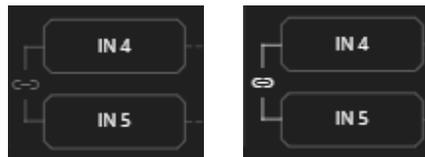
Figure 18: DSP Page – Linking Analog Audio Ports

The selected inputs are linked.

Processing a Signal

Access processing view by clicking an input / output button or a filtering tool in the DSP session view. Use processing view to configure the selected audio signal.
Different port types have different processing modules.

In general:

- Toggle the ⏻ (off) / 🟢 (on) button to enable/disable a processing module.
  The module is enabled while it is set to On and disabled when set to Off.

- Change a port name by clicking the name area and entering the new one.

- In the processing view, the module appears at the center and input/output volume sliders appear to the left/right (for further information, see Input / Output Channels Operation on page 17).

- Adjust configuration knob by clicking and holding the mouse then moving it up or down, or enter the parameter value below the knob and press **Enter** on your keyboard to apply.

- Reset a configuration knob to its default parameter value, by clicking the mouse within the knob area while pressing **Ctrl** on your keyboard.

- The parameter value always appears below the knob or slider.

- A selected input or output button appears with a white rim.

- A selected processing tool button appears with a distinctive color.

- An enabled processing tool button appears with a distinctively colored rim.

Processing modules enable performing the following functions:

- Input / Output Channels Operation on page 17.

- Pre-Matrix Signal Processing on page 18.

- Post-Matrix Signal Processing on page 28.

Input / Output Channels Operation

This section describes the function of the input and output sliders (the examples in this section, showing the inputs, apply also to outputs).

Level Measurement Indicators:

The audio signal enters the digital system at a certain level and is measured in dBFS units (dB relative to full scale, the maximum value).

- **Maximum level indicator** – shows the highest registered level (in RMS) and can change only if a higher level is detected.
  Click the indicator to reset to the current maximum value.

- **0dBFS** – refers to the maximum signal level that can enter the system. signal levels higher than the system limit are clipped.

- **Current maximum level indicator** – displays the current maximum level and holds it until a higher value is detected.
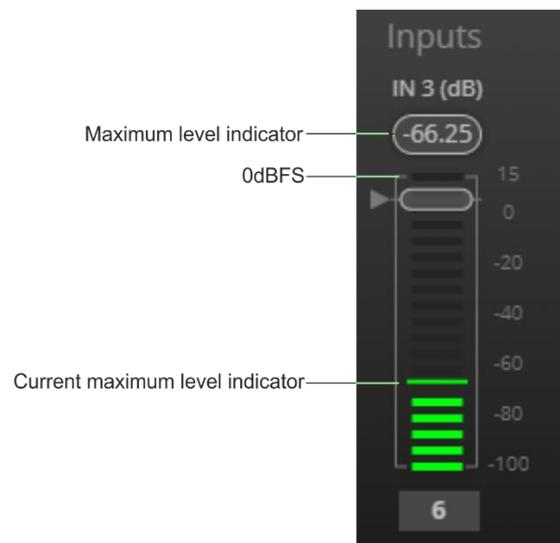
Figure 19: Level Measurement Indicators

Gain/Attenuation Fader

- **Maximum level** – 15dB is the maximum gain.
- **Unity gain** – when volume fader is set to 0dB, the input level is not changed.
- **Volume fader** – slide to increase or decrease the audio level.
- **Minimum level** – -100dB is the maximum attenuation.
- **Current fader position** – shows the current position of the fader. You can also type the desired volume level into this box and press **Enter** on your PC.
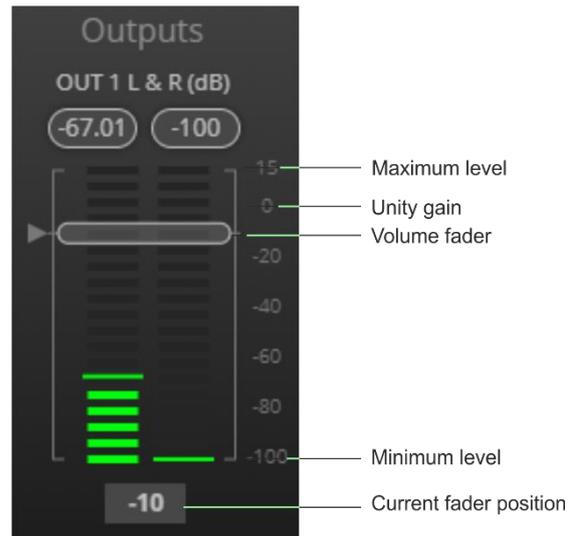
Figure 20: Channel Fader

Pre-Matrix Signal Processing

This section describes the input pre-matrix signal processing of the input audio signal. The input fader always appears to the left.

Pre-matrix enables adjusting the following parameters:

-
-
-
-
-
-
-
-
-
-
-

Adjusting Analog Input Parameters

See to understand the function of the slider. IN 1 is used as an example in this section.

To adjust analog input parameters:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.
2. Click **IN 3**.
   The IN 3 processing page appears.
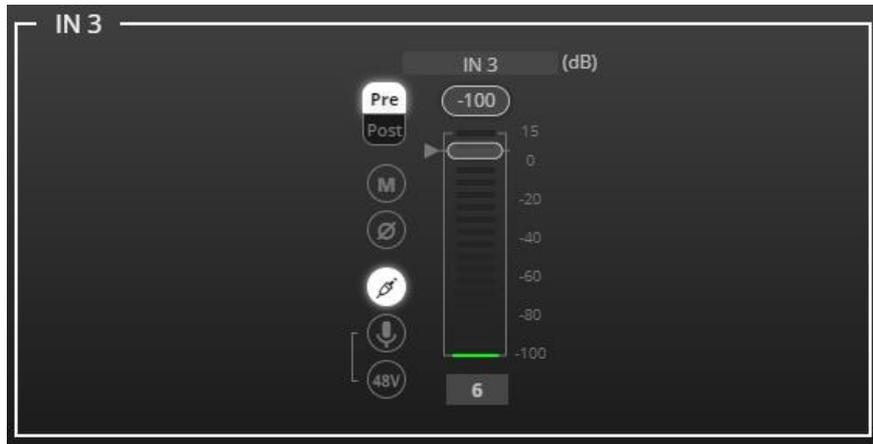
Figure 21: Processing View – Processing Analog Audio Input

3. Perform the following actions:

- Change port name.

- Move the fader to adjust the audio input level.

- Select **Pre** or **Post** to set the signal volume before or after using the pre-matrix modules.

- Toggle [M] / [M] to mute / unmute the input audio, respectively.

- Click [Ø] to inverse polarity (used for troubleshooting).

- Click [Ø] to select audio line in.

- Click [🎤] to select dynamic microphone and [48V] to select condenser microphone (the title IN changes to MIC).

Analog input parameters are adjusted.

Adjusting USB Digital Input Parameters

The USB input signal settings include pre and post processing volume settings and a mute button. The HDMI input signal settings include a mute button but not pre and post volume settings since the signal is not processed.

See to understand the function of the slider.

To adjust the digital USB input parameters:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **USB L/R**.
   The USB Left & Right input processing page appears.
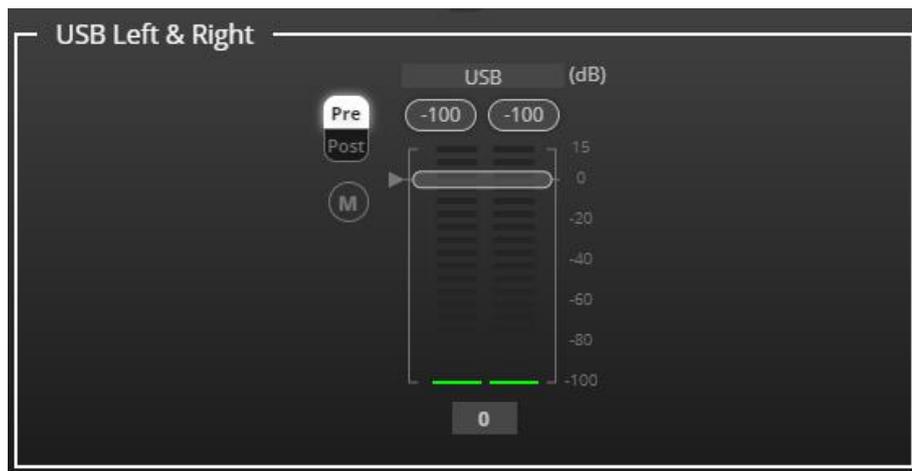
Figure 22: Processing View – Processing USB Input

3. Perform the following actions:

   ▪ Change port name.

   ▪ Move the volume fader to set the left and right audio levels (both sliders are identical).

   ▪ Select **Pre** or **Post** to set the signal volume before and after using the pre-matrix modules.

   ▪ Toggle 🔘 / 🔴 to mute / unmute the input audio, respectively.

Audio parameters are adjusted.

Adjusting HDMI Digital Input Parameters

The HDMI input signal settings include an HDMI selection drop-down box and a mute button (but no pre and post volume settings since the signal is not processed).

See Input / Output Channels Operation on page 17 to understand the function of the slider.

To adjust the digital HDMI input parameters:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **HDMI L/R**.
   The HDMI Left & Right input processing page appears.
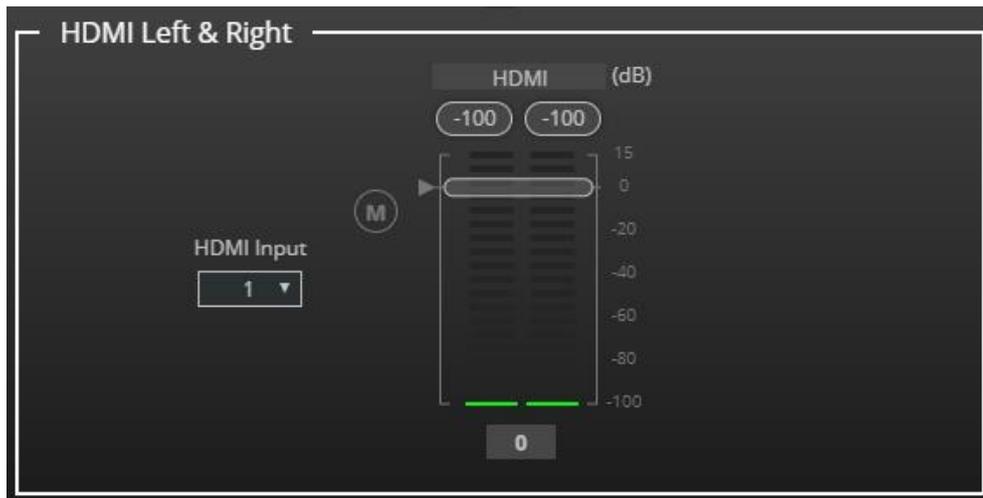
Figure 23: Processing View – Processing HDMI Digital Input

3. Select the HDMI input (1 or 2).

   ⓘ   Only one HDMI port can be active at a time.

4. Perform the following actions:
   - Change port name.
   - Move the volume fader to set the left and right audio levels (both sliders are identical).
   - Toggle ⓜ / 🅜 to mute / unmute the input audio, respectively.

HDMI audio parameters are adjusted.

Adjusting Signal Generator Input Parameters

Use the signal generator to test the output audio signals.

See Input / Output Channels Operation on page 17 to understand the function of the slider.

To adjust the signal generator parameters:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **Signal Generator**.
   The Signal Generator processing page appears.

Figure 24: Processing View – Processing Signal Generator

3. Set the following:

   ▪ Adjust the signal **Level (dB)**.

   ▪ When in **Tone** mode, adjust the signal **Frequency (Hz)**.

   ▪ Select **Pink** noise if required.

   Signal generator parameters are adjusted.

Using Expander Module

Use the Expander module to increase the difference in loudness between the quieter and louder sounds, so that the quiet sounds (usually background noises) become quieter while the loud sounds become louder. The levels of audio signals that fall below the set threshold level are reduced.

To adjust the expander module:

1. In the Navigation pane, click **DSP**. The DSP (Main) page opens.

2. Click **Exp**.
   The button turns light blue and the Expander module page appears.

3. Click the Off button ⏻ Off . The Exp module turns on ⏻ On .

Figure 25: Processing View – Expander Module

4. Define the following:

   ▪ **Threshold (dB)** – Decreases the volume of audio signals that are below the threshold level.

   ▪ **Attack Time (ms)** – Sets the response speed of the expander to signal levels above the threshold.

   ▪ **Release (ms)** – Sets the response speed of the expander to signal levels below the threshold.

5. Open the **Ratio** drop-down box to set the extent to which the volume is decreased. The higher the ratio the more the audio level below the threshold is lowered.

ⓘ   The Expansion (dB) indicates the amount of expansion in a dB scale.

Expander settings are adjusted.

Using HPF (High Pass Filter) Module

A High Pass Filter passes signals that are higher than a certain cut-off frequency. Frequencies under the cut-off frequency are attenuated. Use the HPF module to cut off low frequencies and let higher frequencies pass.

To adjust the HPF:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **HPF**. The button turns light orange and the High Pass Filter module page appears. The left side shows the input volume slider.

3. Click the Off button. The High Pass Filter module turns on.
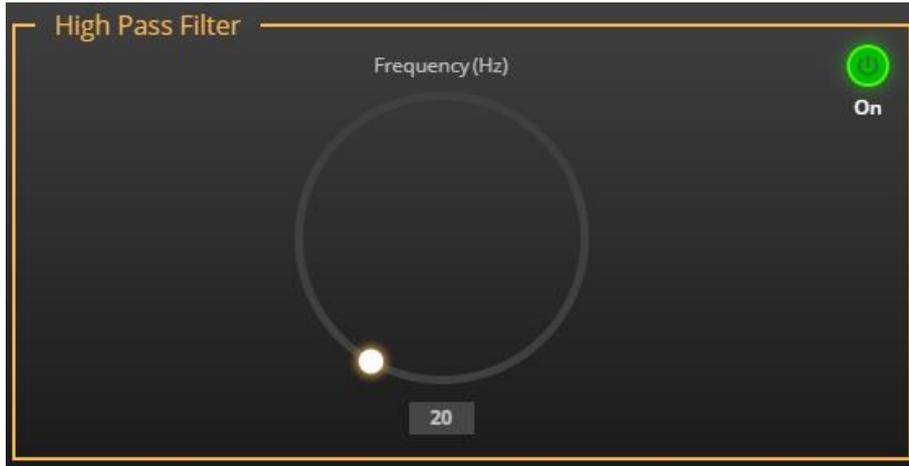
Figure 26: Processing View – HPF Module

4. Set the cut-off frequency.

HPF is adjusted.

Using Compression Module

Use the Compressor module to reduce the signal dynamic range which is the difference between the loudest and quieter sounds (for example, the difference between a scream and a whisper), making the sound seem more natural.

To adjust the compressor settings:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **Comp**.
   The button turns blue and the Compressor module pane appears.

3. Click the Off button. The Comp module turns on.
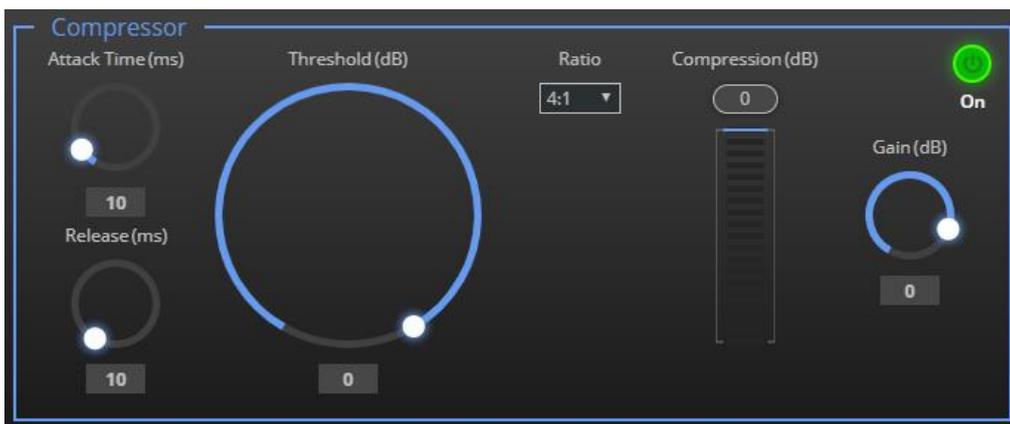
Figure 27: Processing View – Compressor Module

4. Set the following:

  - **Threshold (dB)** – The level that the signal needs to rise above in order for the compressor to begin working. If a signal is too low or does not cross the threshold, the compressor allows the signal to pass through unchanged.

    - **Attack Time (ms)** – The response speed of the compression to signal levels above the threshold.

    - **Release (ms)** – The response speed of the compressor to signal levels above the threshold.

5. Open the **Ratio** drop-down box to set the extent to which the gain is decreased.

6. Set the gain to compensate for the attenuation caused by compression.

The Comp settings are adjusted.

Using Equalizer Module

Use the Equalizer module to change the balance of different frequency components in the audio signal.

To adjust the equalizer:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **EQ**.
   The button turns orange and the Equalizer processing page appears.

3. Click the Off button [Off]. The Equalizer module turns on [On].

Figure 28: Processing View – Equalizer Module

4. Perform the following actions for each of the 4 bands:

   - Click **BYPASS** to ignore a band.

   - Adjust the band **Frequency (Hz)**.

   - Set **Bandwidth (Oct)** to set the range of frequencies around the selected frequency.

   - Set the bandwidth audio **EQ Level (dB)**.

Equalizer settings are adjusted.

Using Gain Module

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **Gain**.
   The button turns violet and the Gain processing page appears.
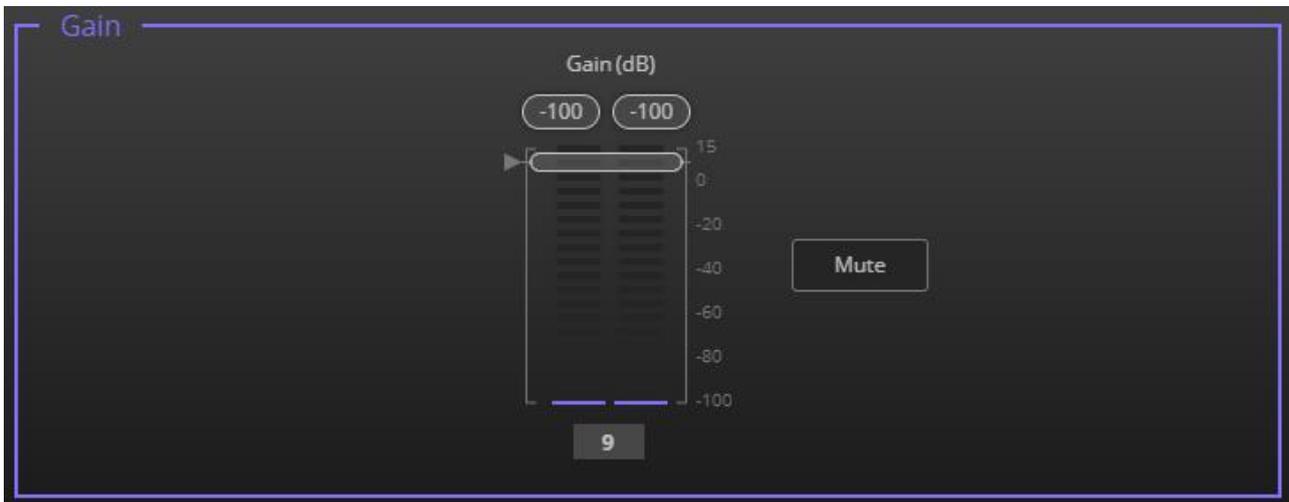
Figure 29: Processing View – Gain Module

3. Perform the following actions:

   ▪ Set gain.

   ▪ Click **Mute** if required.

Gain is adjusted.

Using AEC Module

ⓘ Before enabling AEC, make sure to disable AEC in the software you are using (if it includes this feature).

The AEC (Acoustic Echo Cancellation) module is a learning filter algorithm that, when enabled, filters the unwanted echoes in the room, such as room speakers.

To enable/disable AEC delay:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **AEC**.
   The button turns turquoise, and the Acoustic Echo Cancellation processing page appears.

Figure 30: Processing View – AEC Module

3. Click Enable to apply AEC to the signal.

AEC is enabled.

Using Delay Module

Set the delay to accommodate the audio to the listeners distance from the speakers. Delay time tool converts the delay in milliseconds to meters, feet, and samples.

To adjust the delay:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **Delay**.
   The button turns green and the Equalizer processing page appears.

3. Click the Off button [Off] . The Delay module turns on [On] .

Figure 31: Processing View – Delay Module

4. Set the delay.

Delay setting is adjusted.

Post-Matrix Signal Processing

DSP-62-AEC enables performing post-matrix signal processing to outputs, including:

- Using Delay Module on page 28.
- Using Post Matrix Equalizer Module on page 30.
- Using Limit Module on page 31.

Setting Analog Audio Output Parameters

The analog (AUDIO OUT 5-pin terminal block connector ⑬) and USB ⑦ output signal settings are identical. Analog is used as an example in this section.

See Input / Output Channels Operation on page 17 to understand the function of the slider.

To adjust the audio outputs:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **OUT 1 L or OUT 1 R**.
   The Dante processing page appears.

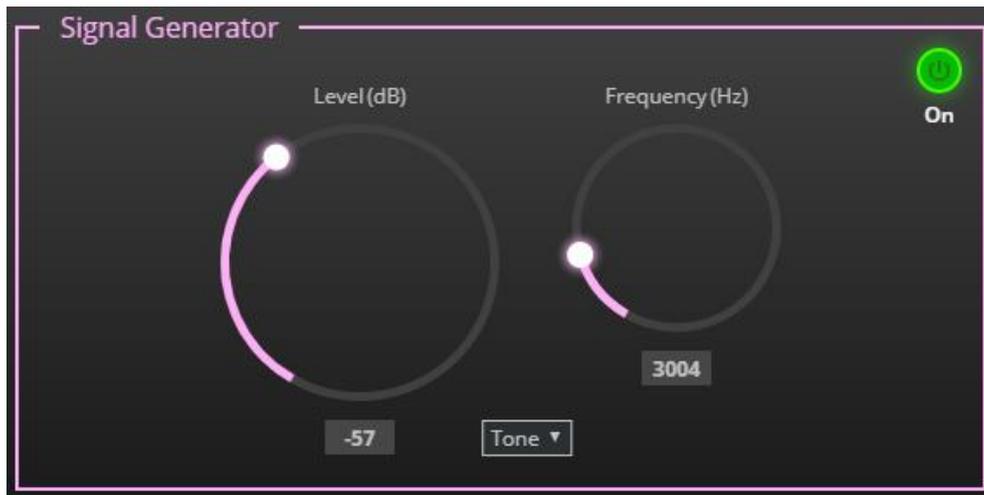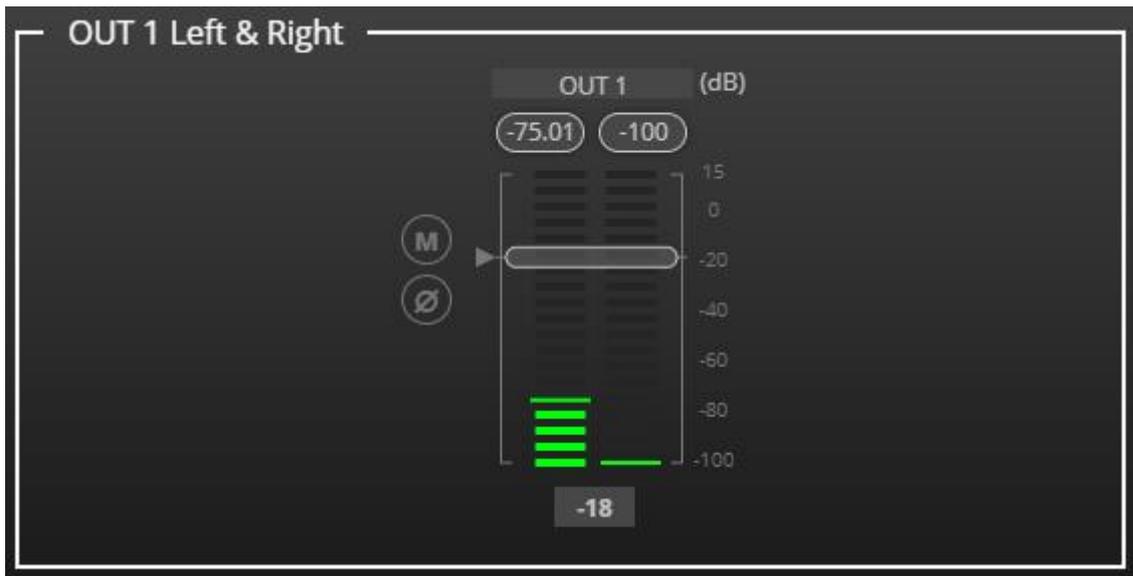
Figure 32: Processing View – Processing Digital Input

3. Perform the following actions:

   ▪ Move the volume fader to set the output audio level (both sliders are identical).

   ▪ Toggle ⓜ / ⓜ to mute / unmute the output audio, respectively.

   ▪ Click ⌀ to inverse polarity (used for troubleshooting).

Audio outputs are adjusted.

Using Post Matrix Equalizer Module

Use the Equalizer module to change the balance of different frequency components in the audio signal.

To adjust the equalizer:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **EQ**.
   The button turns orange and the Equalizer processing page appears.

3. Click the Off button ⏻ . The Equalizer module turns on ⏻ .

Figure 33: Processing View – Processing Output Equalizer

4. Perform the following actions for each of the 8 bands:

   - Click **BYPASS** to ignore that band.

   - Set the band frequency (Hz).

   - Set the audio level (dB).

   - Set the bandwidth (Oct).

Equalizer settings are adjusted.

Using Limit Module

Use the Limiter tool to limit the signal level to the specified threshold, reducing the gain above the threshold. A limiter can boost the volume of a certain sound.

To adjust the limiter:

1. In the Navigation pane, click **DSP**. The DSP (Main) page appears.

2. Click **Limit**. The button turns purple and the Limiter processing page appears.
   The right side shows the output volume slider.

3. Click the Off button [Off]. The Limiter module turns on [On].

Figure 34: Processing View – Limiter Module

4. Set the **Threshold**.
   Note the **Gain Reduction** meter as you change the threshold.

Limiter settings are adjusted.

Routing Inputs to Outputs

Click a cross-point to connect any inputs to any of the outputs via the Matrix page; set the connection volume, link analog input and output pairs and select the outputs to the amplifier.

DSP-62-AEC enables performing the following functions:

- [Connecting Inputs to Outputs](#) on page .
- [Setting Cross-Point Volume](#) on page .
- [Linking Analog Pairs](#) on page .

Connecting Inputs to Outputs

To route an input or several inputs to an output:

1. In the Navigation pane, click **Matrix**. The Matrix page appears.
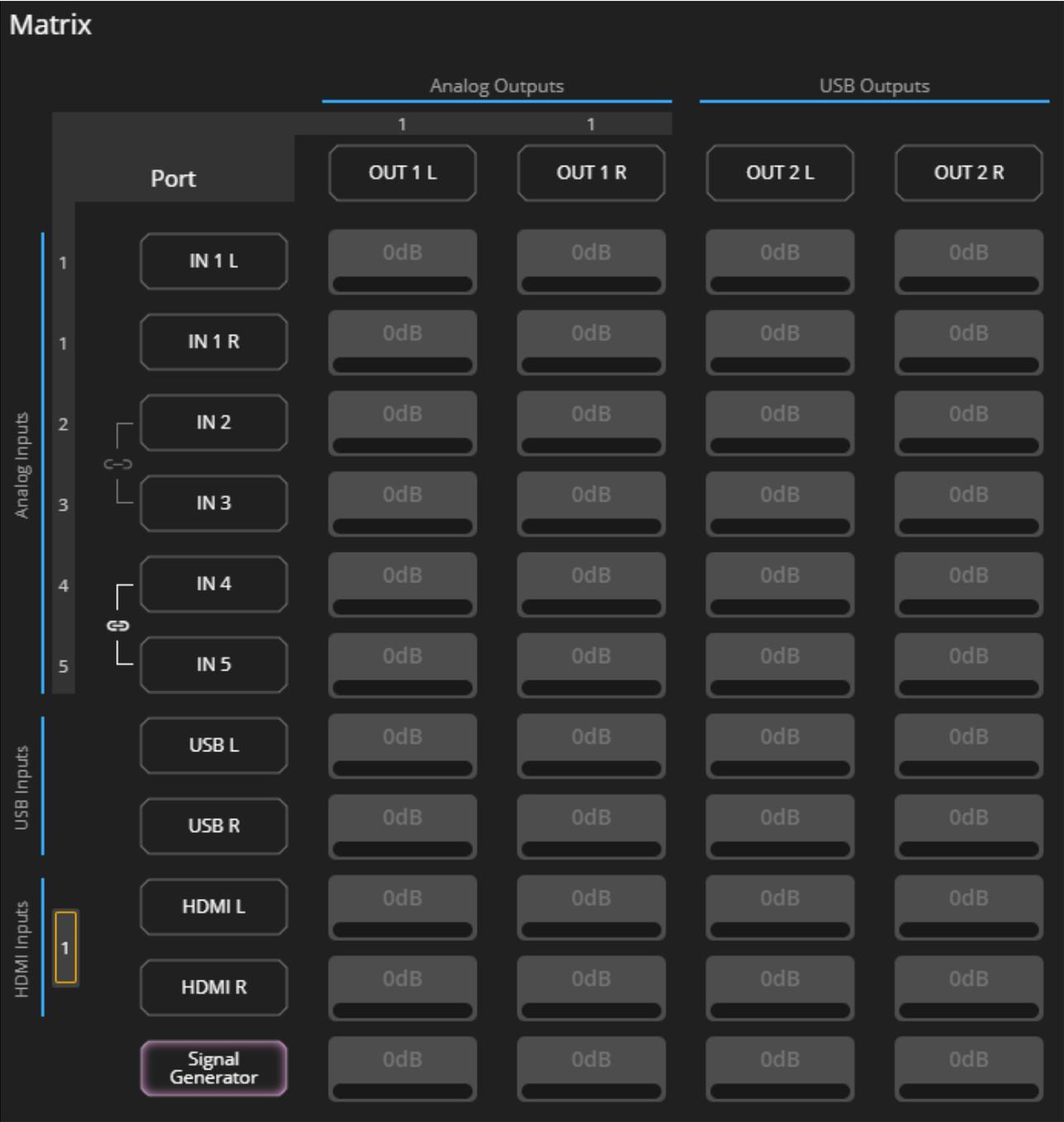
Figure 35: Matrix Page

2. Click an in-out cross-point (for example, IN 2 input and OUT 1 L and R outputs). The black cross-points turn green.
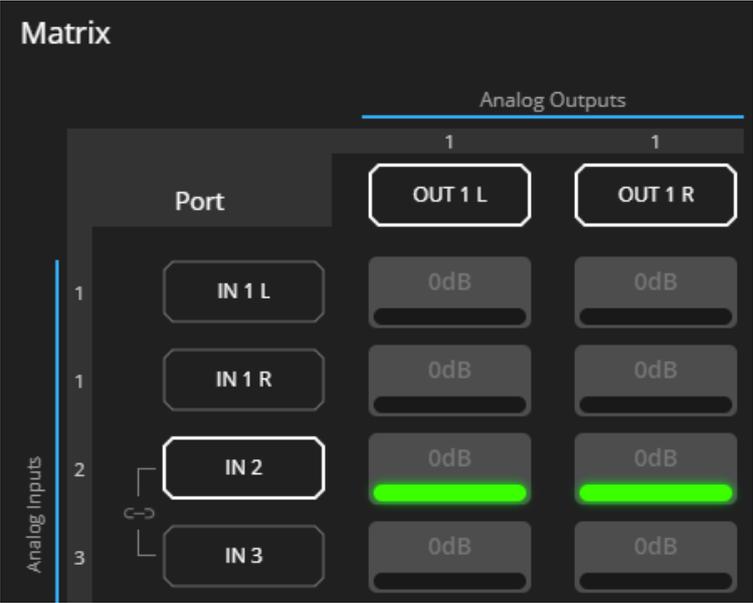
Figure 36: Matrix Page – In-Out Cross-Points

3. Click any other cross-points (one input to output/s or several inputs to output/s).

Figure 37: Matrix Page – Multiple Input-Output Cross-Points

Selected inputs are routed to selected outputs.

ⓘ   You can also select an audio signal generator for testing.

Setting Cross-Point Volume

Set the cross-point volume separately for each in-out connection.

To set the cross-point volume:

1. In the Navigation pane, click **Matrix**. The Matrix page appears.

2. Click the volume area (0dB, by default).
   The volume window appears.
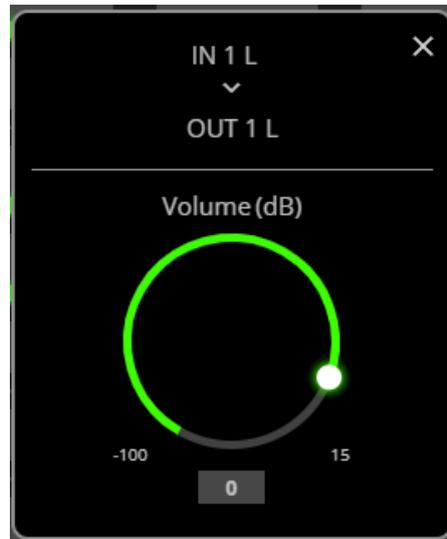
Figure 38: Matrix Page – Setting Cross-Point Volume

3. Set the cross-point volume (using the knob or entering the value and pressing **Enter** on your keyboard). The cross-point volume is set and appears at the cross-point.
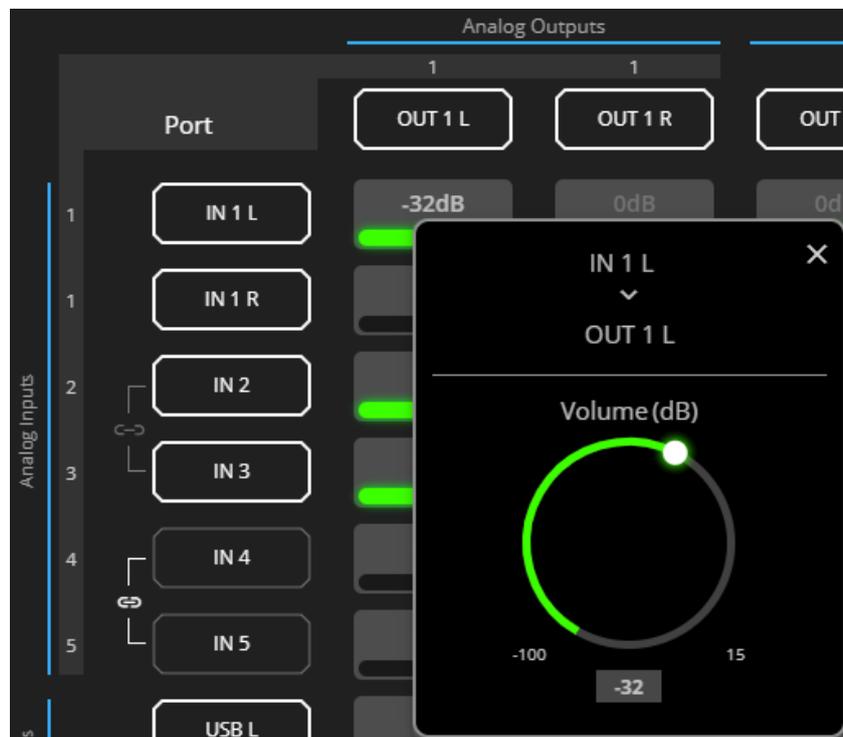
Figure 39: Cross-Point Volume Value

Audio volume is set at the cross-point.

Linking Analog Pairs

To link analog input or output pairs, see <u>Linking Analog Inputs and Outputs</u> on page <u>16</u>.

Mixing Audio Signals

Mix the audio signals and store/recall mixing snapshots via the Mixer page.

DSP-62-AEC enables performing the following tasks:

- <u>Defining Input and Output Parameters</u> on page <u>35</u>.
- <u>Defining Snapshots</u> on page <u>36</u>.

Defining Input and Output Parameters

Set audio parameters for each input and output.

To set input/output parameters:

1. In the Navigation pane, click **Mixer**. The Mixer page appears.
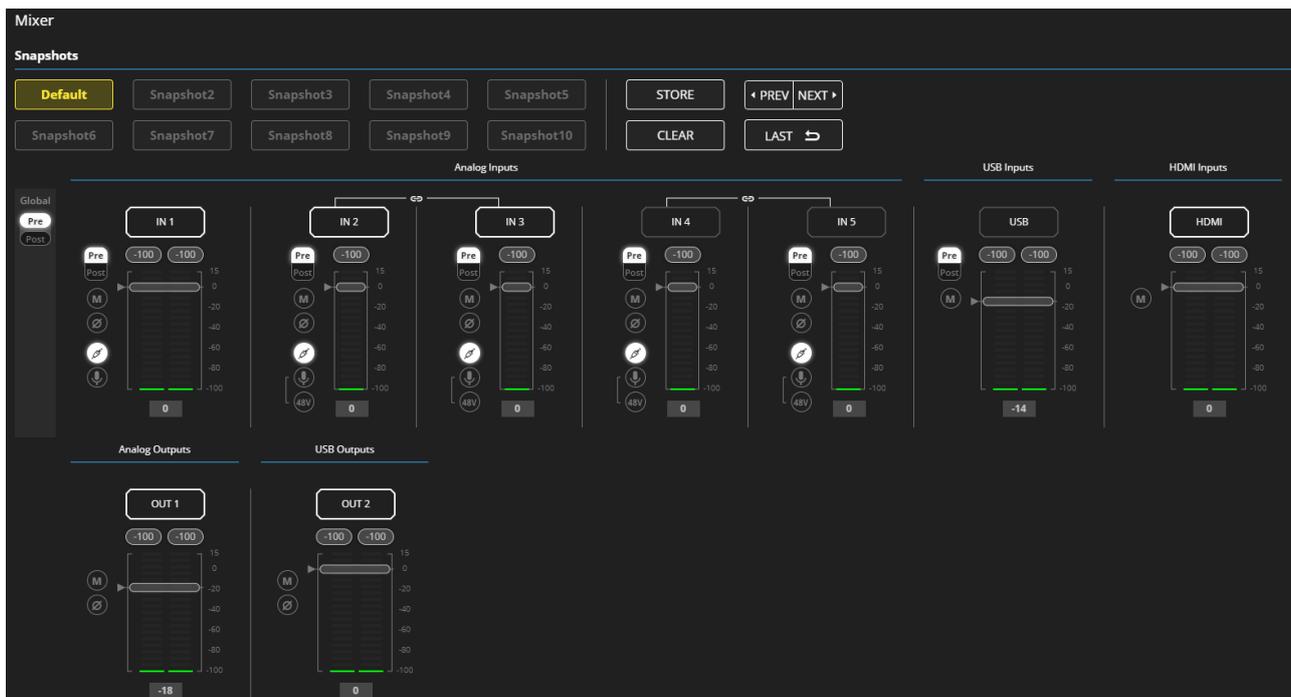
Figure 40: Mixer Page

ⓘ An input/output frame with a white rim indicates that this input/output is currently connected to an output/input, respectively.

2. Use the slider or enter the desired value and press **Enter** (on your PC) to set the volume.
   View the current gain and the input/output name (see <u>Input / Output Channels Operation</u> on page <u>17</u>).

3. Set the following:

- Select **Pre** or **Post** to set the signal volume before and after using the modules.
- Toggle ![M] / ![M] to mute / unmute the input audio, respectively.
- Click ![Ø] to inverse polarity (used for troubleshooting).

For analog audio inputs only:

- Click ![icon] to select audio line in.
- Click ![mic] to select dynamic microphone and ![48V] to select condenser microphone.
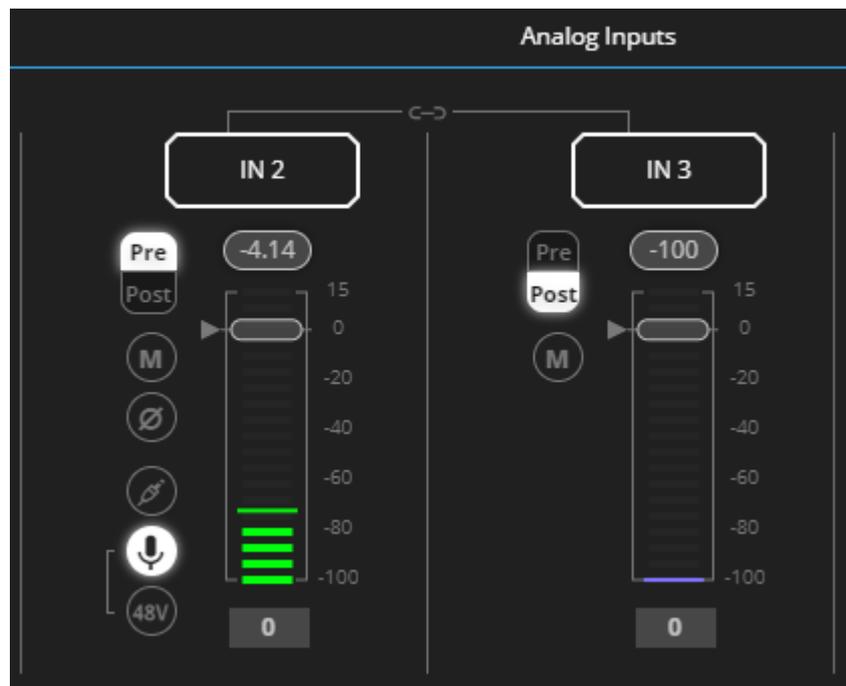
Figure 41: Mixer Page – Analog Audio Settings

Audio parameters are defined.

Defining Snapshots

Store a snapshot (inputs and outputs) to store the current configuration state, recall a snapshot, set to default or clear a snapshot.

Storing Snapshots

To store a snapshot:

1. In the Navigation pane, click **Mixer**. The Mixer page appears.

2. Set input and output mixers.

ⓘ When the parameters change, the Default button turns yellow. Click **Default** to restore default settings.
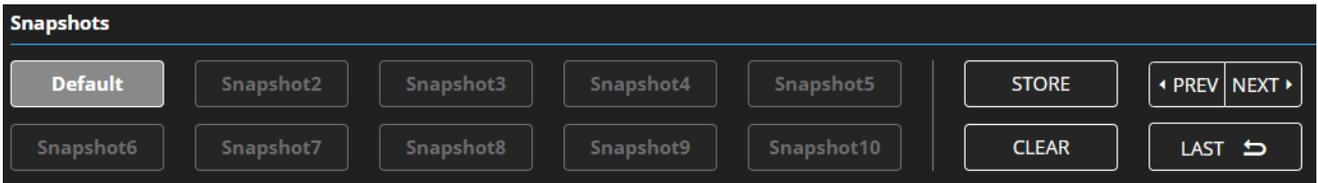
Figure 42: Mixer Page – Snapshots

3. Click **Store**. The Snapshot buttons turn green.
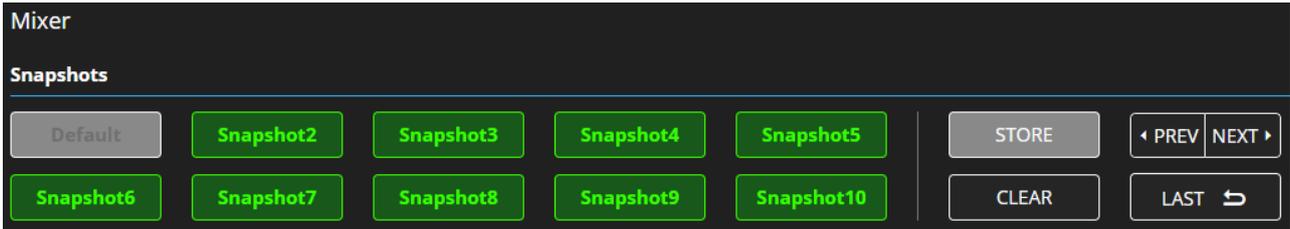
Figure 43: Mixer Page – Storing Snapshots

4. Click a Snapshot button to complete the action (for example, **Snapshot 1**).
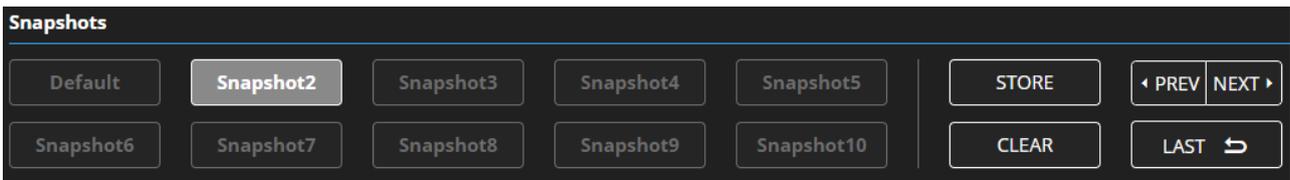
Figure 44: Mixer Page – Selecting a Snapshot

The current configuration is stored to Snapshot 1.

Clearing Snapshots

To clear a snapshot configuration:

1. In the Navigation pane, click **Mixer**. The Mixer page appears.

2. Click **Clear**. Snapshot buttons turn blue.
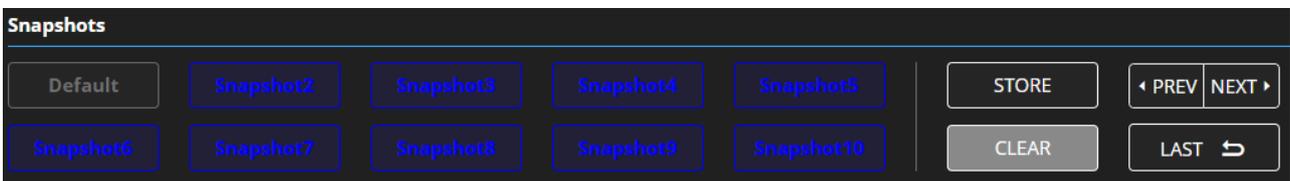
Figure 45: Mixer Page – Clearing a Snapshot

3. Select the snapshot to be cleared. The snapshot cleared returns to its default values.

The snapshot is cleared (reset to factory default values).

Loading Snapshots

To load a snapshot:

1. In the Navigation pane, click **Mixer**. The Mixer page appears.

2. Do any of the following to load the desired snapshot:

   ▪ Click **Snapshot** (Default or 2 to 10).

   ▪ Click **Next** to load the next snapshot configuration.

   ▪ Click **Prev** to load the previous snapshot configuration.

   ▪ Click **Last** to load the latest configured snapshot (clicking **Last** again goes to the previously configured snapshot and so on).

   The selected snapshot is loaded.

Audio and Video Settings

Audio and video settings enable performing the following settings:

-
-

Defining Audio Settings

Set the **DSP-62-AEC** analog audio I/O configuration, system presets and amplifier settings using the A/V Settings page.

> ⓘ Amplifier settings are only relevant to **DSP-62-AEC**.

To define audio settings:

1. In the Navigation pane, click **A/V Settings**. The A/V Settings page appears.
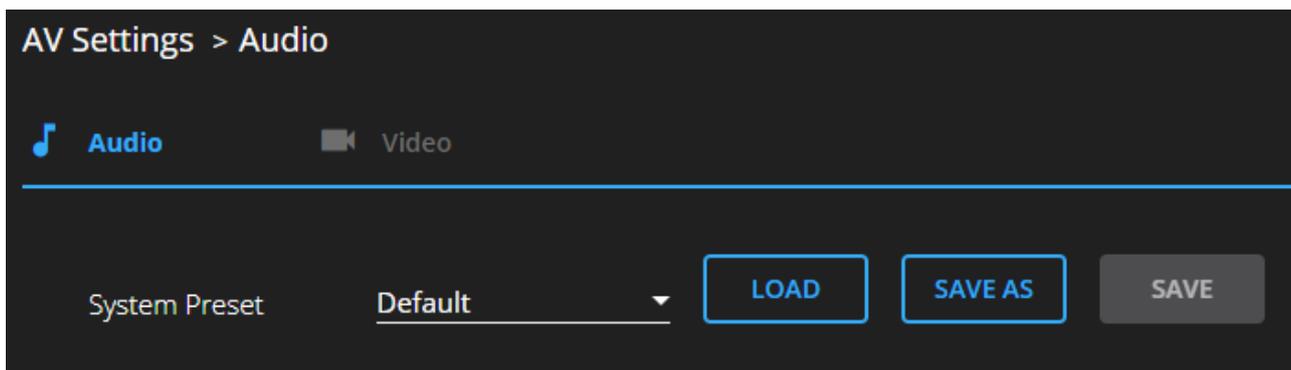
Figure 46: A/V Settings Page

2. In the **System Preset** drop-down box, select a preset (default or System 2 to System 10) and click **LOAD**, **SAVE AS** or **SAVE**.
   The current preset is loaded or saved.

   > ⓘ System Presets contain all the system configuration including Snapshot configuration and excluding IP settings.

Audio settings are defined.

Defining Video Settings

Set the **DSP-62-AEC** HDMI input and output labels, Force RGB and/or Force 2LPCM, and video pattern (if required), using the Video tab in the A/V Settings page.

To define video settings:

1. In the Navigation pane, click **A/V Settings**. The A/V Settings page appears.
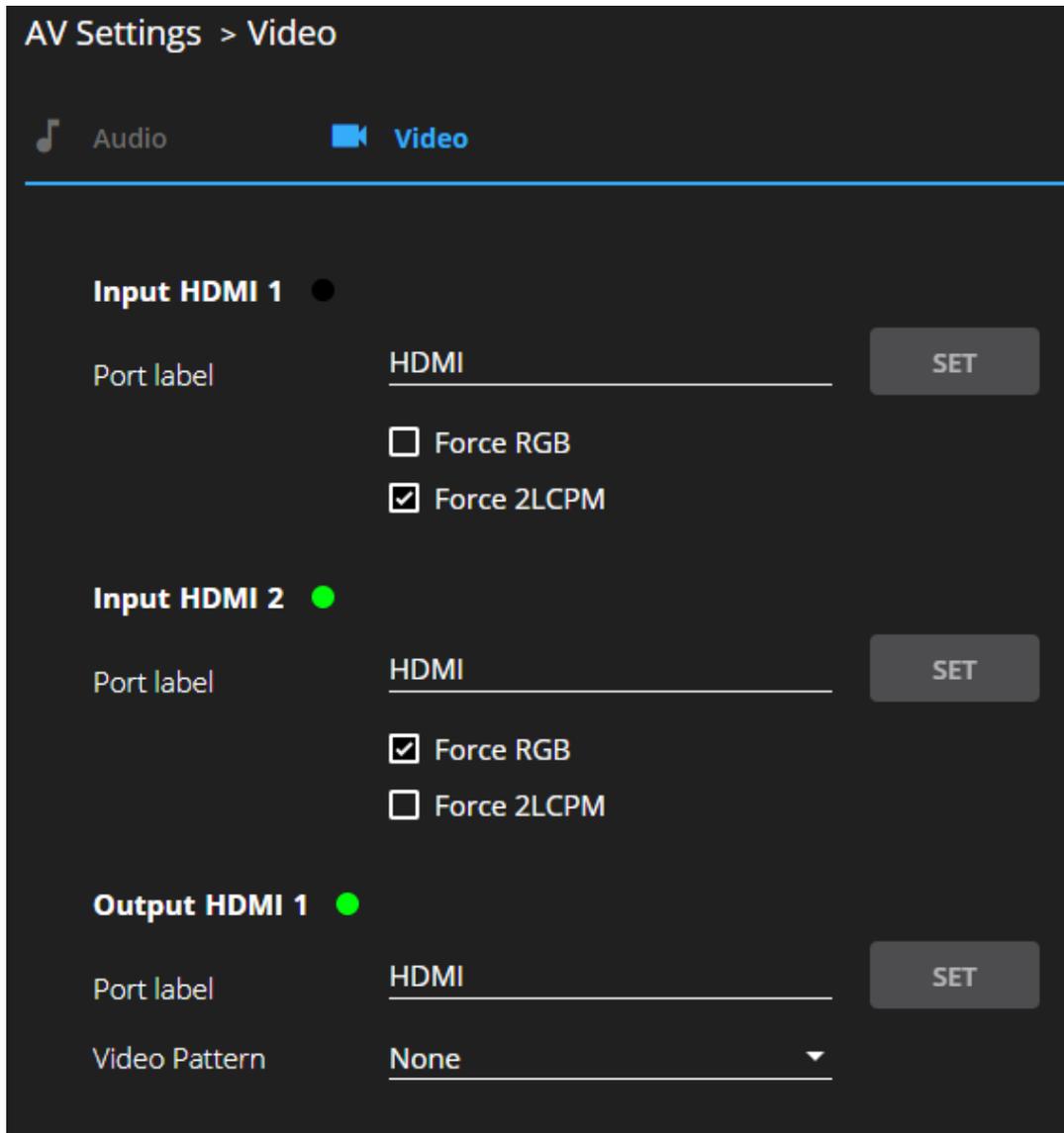
Figure 47: Video Settings Page

2. Select Video tab.

3. Enter HDMI input and output labels then click **Set**.

4. For HDMI input, check/uncheck **Force RGB** and/or **Force 2LPCM**.

5. If required, select a video pattern from the drop-down box.

Video settings are defined.

Defining General Settings

Change the device name, view the model and serial number and firmware version using the General tab in the Settings page, which also enables:

- [Performing Firmware Upgrade](#) on page [42](#).

- [Importing/Exporting Global Settings](#) on page [43](#).

- [Restarting and Resetting the Device](#) on page [44](#).

- [Defining Communication Settings](#) on page [45](#).

- [Setting Access Security](#) on page [47](#).

Performing Firmware Upgrade

Perform **DSP-62-AEC** firmware upgrade via the General tab in the Device Settings page.

To perform firmware upgrade:

1. In the Navigation pane, click **Device Settings**. The General tab in the Device Settings page appears.

Device Settings > General

⚙ **General**   🌐 Network   🛡 Security

General Preferences

Device Name   DSP-62-AE-0141

Model   DSP-62-AEC

Serial Number   7200053200141

Firmware Version   1.0.58843   **UPGRADE**

Global System Settings

IMPORT   **EXPORT**

RESTART   ⚠ **RESET**

Device Restart   Factory Reset

SAVE   CANCEL
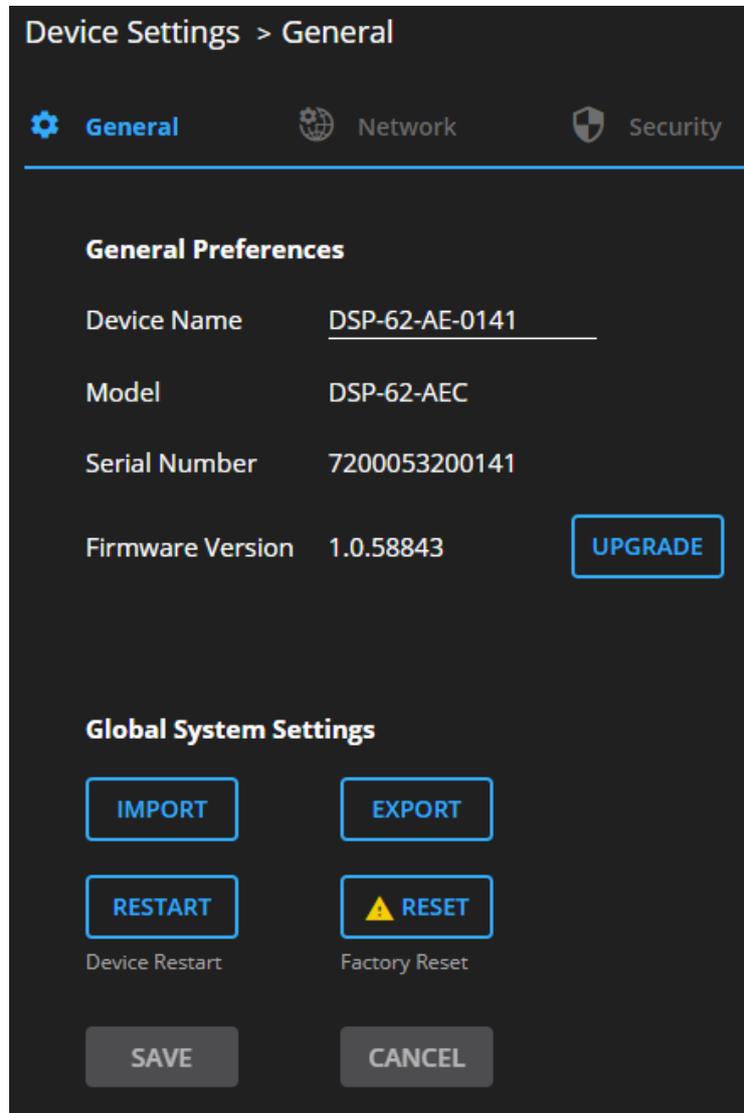
Figure 48: Upgrade Settings Tab – Upgrading the Firmware

2. Click **UPGRADE** and select the new firmware file.
   The following message appears:
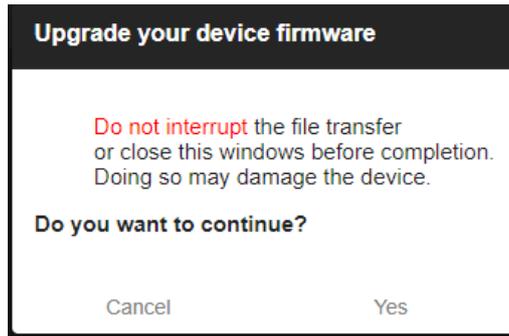
Figure 49: Upgrade Settings Tab – Firmware Upgrade Message

3. Click **Yes**.
   Wait for completion of the upgrade process:
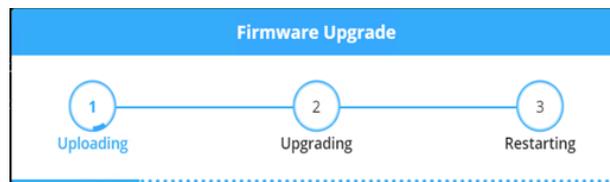
Figure 50: Upgrade Settings Tab – Firmware Upgrade Process

4. Wait for the device to restart.

Firmware upgrade is complete.

Importing/Exporting Global Settings

You can export a Global Settings file to a different **DSP-62-AEC** device or import a file to your device. This feature is used for multi deployment of same-configuration devices.

Global configuration includes all the settings, including System presets and all their Snapshot configurations, excluding IP settings.

To import/export global settings:

1. In the Navigation pane, click **Device Settings**. The General tab appears.

2. In the General tab, in the Global System Settings area:

   - Click **IMPORT** to import a file: select the system setting ".bin" file from the Open window and click **Open**.
     The imported system settings file is uploaded onto the device.

   - Click **EXPORT** to export a file: the current system setting ".bin" file is downloaded onto your PC and can be exported to other devices.
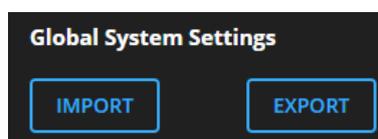
Figure 51: General Settings Tab – Importing / Exporting Global Settings

Global system settings are imported/exported.

Restarting and Resetting the Device

Restart the **DSP-62-AEC** or reset it to its factory default parameters using the Device Settings page.

Restarting the Device

To restart the device:

1. In the Navigation pane, click **Device Settings**. The Device Settings page appears.
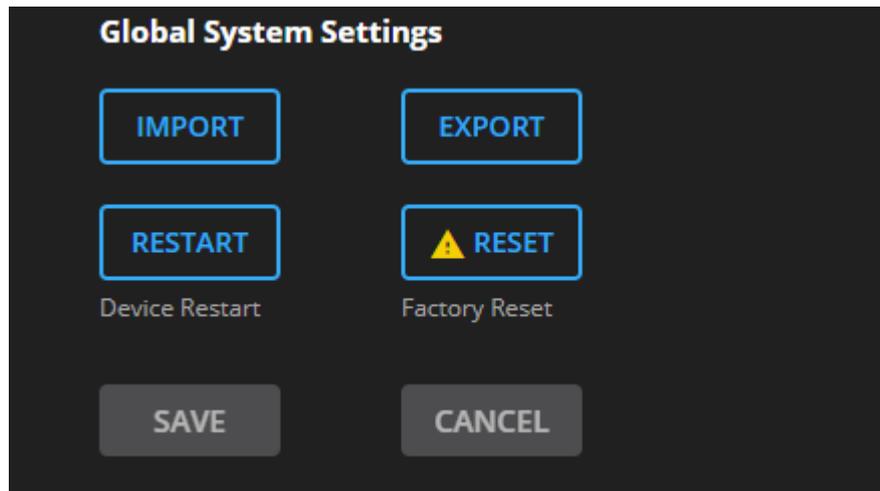
Figure 52: Device Settings Page - Restart

2. Click **Restart**.

Figure 53: Device Restart Window

3. Click **Proceed**. Wait for the device to reload after device restart.

Device has restarted.

Resetting the Device

To reset the device to its default parameters:

1. In the Navigation pane, click **Device Settings**. The Device Settings page appears.

2. Click **Factory reset**. The following message appears:

Device Reset

Do you want to reset the device?
This action resets the device's operation mode and restarts the device.
During restart the device will be temporarily offline.
Do you want to proceed?

Cancel   Procceed
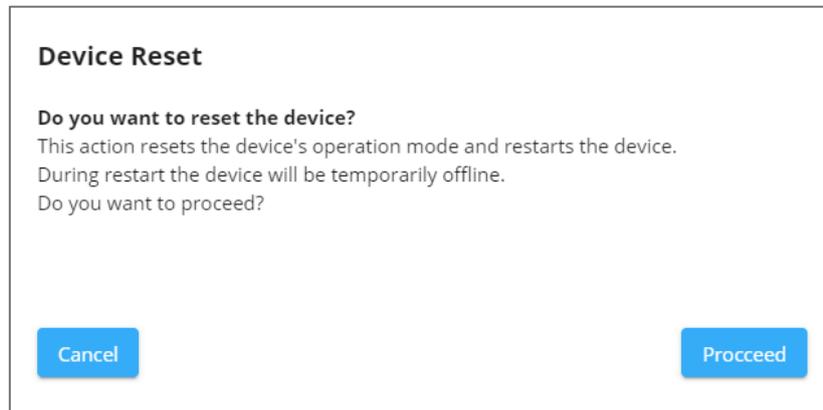
Figure 54: Settings Page – Factory Reset Message

3. Click **Proceed**.

The device is resets to its factory default parameters.

Defining Communication Settings

Set the **DSP-62-AEC** communication parameters, including the IP Address, Mask, gateway and so on using the Communication tab in the Settings page.

DSP-62-AEC enables performing the following functions:

-
-

Setting Parameters when DHCP is On

To set parameters when DHCP is set to On:

1. In the Navigation pane, click **Device Settings**. The General tab in the Device Settings page appears.

2. Select the **Network** tab.
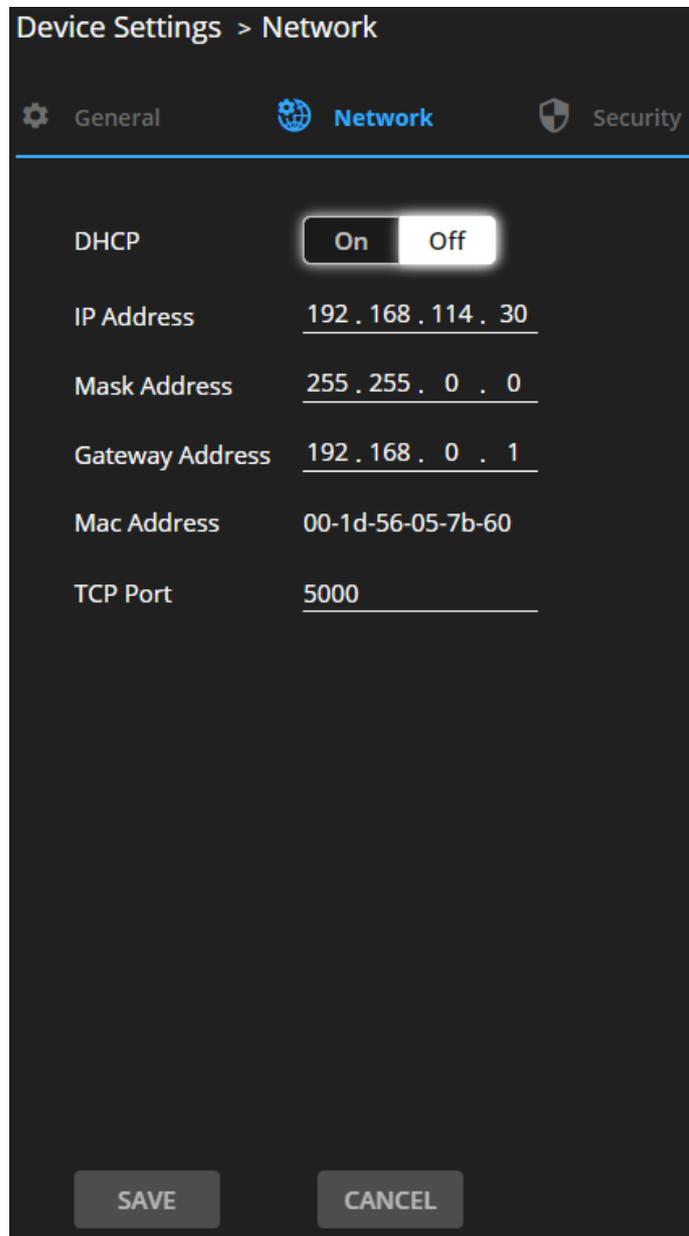
Figure 55: Device Settings Page – Network Tab

3. Take note of the Device Name in the General tab (you will need it when reloading the page).

4. Set DHCP to **ON**.

5. Click **SAVE**.

6. Type the device name in the address bar of your browser to reload the page.
   You can read the new IP address from the Network page.

Parameters are set.

Changing Ethernet Settings

To change the Ethernet settings:

1. In the Navigation pane, click **Device Settings**. The General tab in the Device Settings page appears.

2. Select the **Network** tab.

3. If DHCP is set to Off (default), change any of the parameters (IP Address, Mask Address and/or Gateway Address).

4. If required, change the TCP port number.

5. Click **Save**. A communication error message appears trying to retrieve the connection, when changing any of the addresses.

Communication error

Communication with device has been closed.
Machine is down click OK to reconnect.

OK
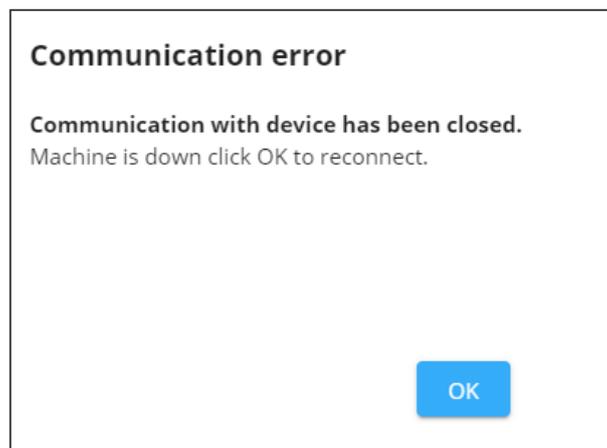
Figure 56: Communication Settings Tab – Communication Error Message

6. Refresh the page and enter the new data or click **OK**.

After changing the IP address, reload the webpage with the new IP address.

If DHCP is On, reload the webpage with the new IP address (see Setting Parameters when DHCP is On on page 46).

Ethernet settings have changed.

Setting Access Security

By default, the webpages are secured and require access permission (user name and password are both **Admin**).

DSP-62-AEC enables performing the following security actions:

- Disabling Security on page 48.
- Enabling Security on page 49.
- Changing the Password on page 49.

Disabling Security

To disable security:

1. In the Navigation pane, click **Device Settings**. The General Settings tab appears.

2. Select the Security tab.
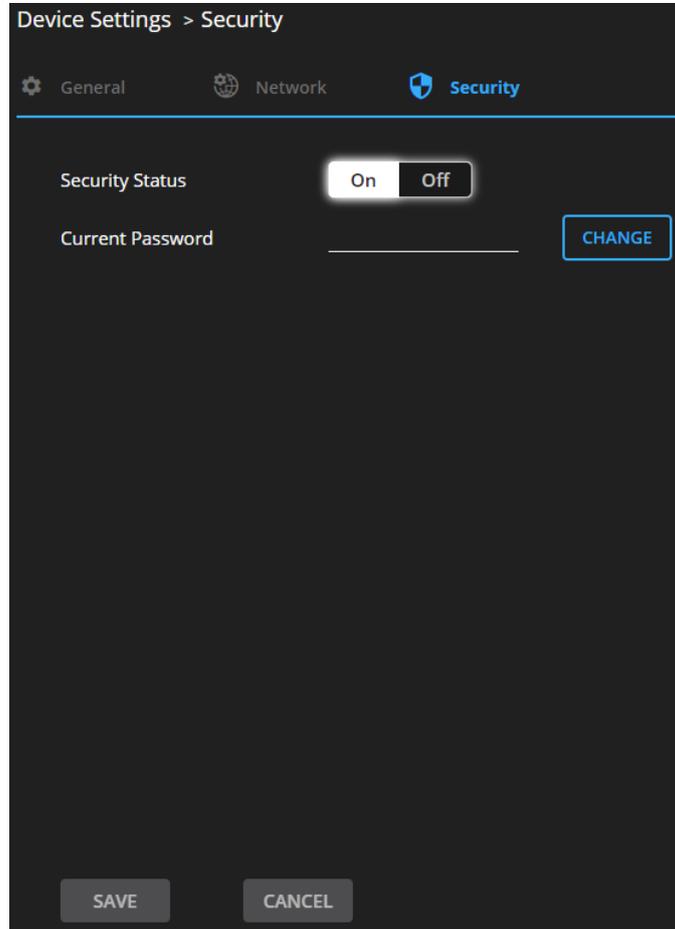
Figure 57: General Settings Tab – Security

3. Click **Off**. The following message appears.
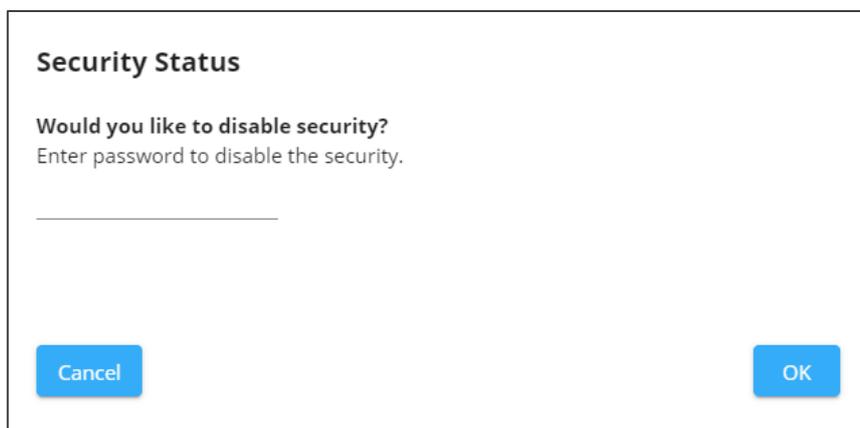
Figure 58: General Settings Tab – Security Message

4. Enter the current password and click **OK**.

Security is disabled. The security-disabled icon appears (🔓).

Enabling Security

To enable security:

1. In the Navigation pane, click **Settings**. The General Settings tab appears, displaying the Security area.

2. Click **On**. The full security page appears (see Figure 57).

Figure 59: General Settings Tab – Enabling Security

Security is enabled. The security-enabled icon appears ( ).

Changing the Password

To change the password:

1. In the Navigation pane, click **Settings**. The Settings page appears, displaying the Security area (see Figure 57).

2. Enable security (if disabled).

3. Enter the current password.

4. Click **CHANGE**.
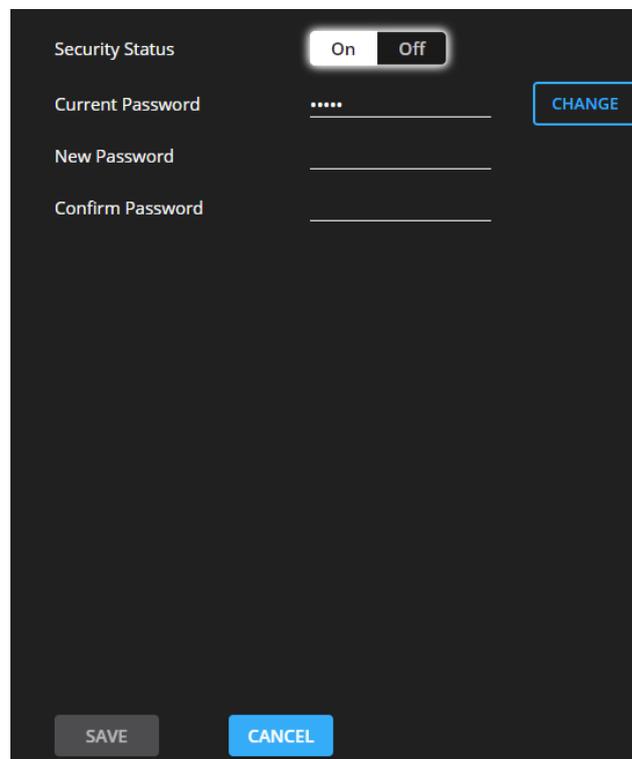
Figure 60: General Settings Tab – Changing the Password

5. Enter the new password or use the suggested password.

6. Click **SAVE**.

The password has changed.

Viewing Device Information

In the Navigation pane, click **About** to view the **DSP-62-AEC** webpage version and Kramer Electronics Ltd. details.
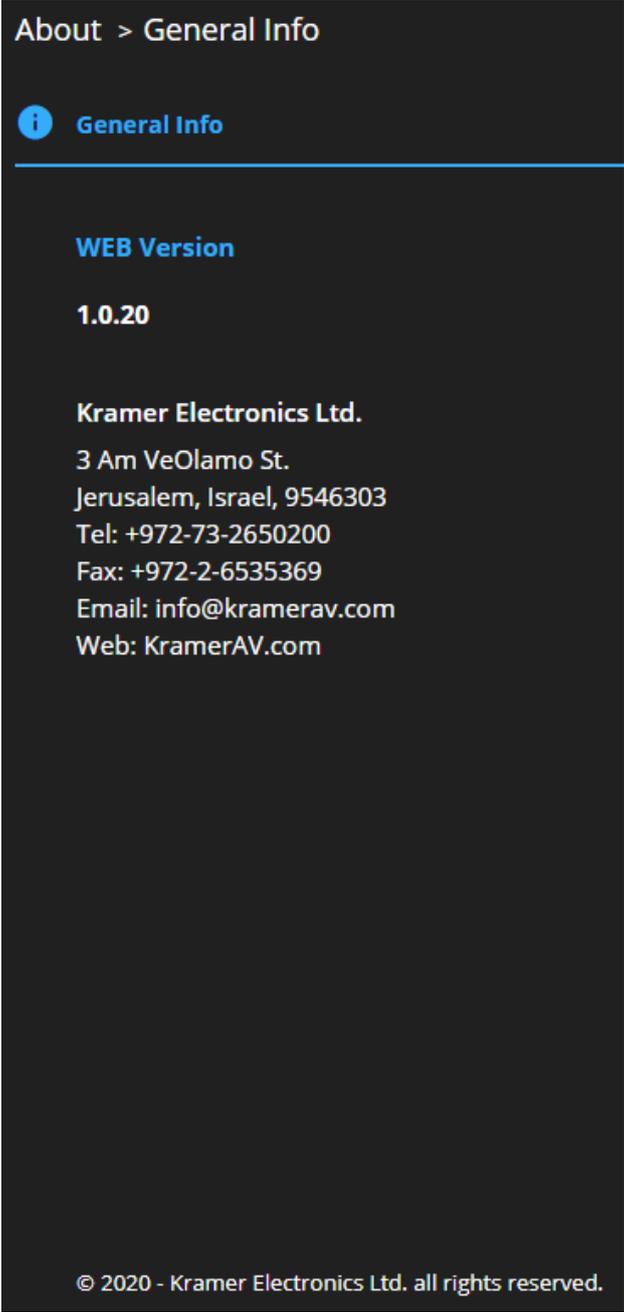
Figure 61: About Page

Upgrading Firmware

Use the Kramer **K-UPLOAD** software to upgrade the firmware via the **DSP-62-AEC** RS-232 port ⑮.

The latest version of **K-UPLOAD** and installation instructions can be downloaded from our website at: www.kramerav.com/support/product_downloads.asp.

ⓘ Note that in order to use the micro USB port, you need to install the Kramer USB driver, available at: www.kramerav.com/support/product_downloads.asp.

Technical Specifications

Inputs	2 HDMI	On female HDMI connectors
	1 Unbalanced Stereo Audio	On a 3.5mm mini jack
	4 Balanced Mono Audio	On 3-pin terminal block connectors
Outputs	1 HDMI	On a female HDMI connector
	1 Balanced Stereo Audio	On a 5-pin terminal block connector
Ports	1 USB Audio	On a USB B connector
	1 RS-232	On a 3-pin terminal block
	1 Ethernet	On an RJ-45 female connector
Video	Max Bandwidth	18Gbps (6Gbps per graphic channel)
	Max Resolution	4K@60Hz (4:4:4)
	Compliance	HDMI and HDCP 2.2
Line/Mic Level Input	Impedance Unbalanced Impedance Balanced Impedance Microphone	7.6kΩ 3.8kΩ 3.8kΩ
	Nominal level Unbalanced Nominal level Balanced	0dBV (0.77Vrms) +6.8dBu (1.54Vrms)
	Maximum level (Balanced)	+8dBu (2Vrms)
	Sensitivity Unbalanced Sensitivity Balanced	Full power @ 0dBV (0.77Vrms) Full power @ +6dBu (1.54Vrms)
	Phantom Power	48 VDC on/off per input
Line Level Output	Impedance Unbalanced Impedance Balanced	500Ω
	Frequency Response	20Hz - 20kHz @ +/-1dB
	S/N Ratio:	>85dB, 20Hz - 20kHz, at unity gain (unweighted)
	Audio THD + Noise:	<0.01%, 20 Hz - 20kHz, at unity gain
	Crosstalk	<-85dB, 20Hz to 20kHz
Controls	Front Panel	DIP-switches, 6 input and 2 output audio status LEDs, 1 power on LED
Control RS-232	Baud Rate	115200
Supported Web Browsers	Windows	Chrome
Power	Consumption	630mA
	Source	PoE or 12V DC, 5A
Environmental Conditions	Operating Temperature	0° to +40°C (32° to 104°F)
	Storage Temperature	-40° to +70°C (-40° to 158°F)
	Humidity	10% to 90%, RHL non-condensing
Regulatory Compliance	Safety	CE, FCC
	Environmental	RoHs, WEEE
Enclosure	Size	DemiTOOLS®
	Type	Aluminum
	Cooling	Convection Ventilation
General	Net Dimensions (W, D, H)	19.1cm x 6cm x 2.7cm (7.5" x 2.4" x 1.1")
	Shipping Dimensions (W, D, H)	34.5cm x 16.5cm x 5.2cm (13.6" x 6.5" x 2.1")
	Net Weight	0.16kg (0.4lbs)
	Shipping Weight	0.6kg (1.3lbs) approx.
Accessories	Included	Power adapter cord

Specifications are subject to change without notice at www.kramerav.com

Default Communication Parameters

RS-232 Control/Protocol 3000	
Baud Rate:	115,200
Data Bits:	8
Stop Bits:	1
Parity:	None
Command Format:	ASCII
Example (adjust analog audio output 1 to "-10dB"):	#x-aud-lvl out.analog_audio.1.audio.1,-10
Ethernet	
To reset the IP settings to the factory reset values go to: Menu->Setup -> Factory Reset-> press Enter to confirm	
Note that **DSP-62-AEC** is configured as DHCP ON mode, so in a network where the device receives an automatic IP, the device is accessible via its name or the IP it received.	
IP Address:	192.168.1.39
Subnet mask:	255.255.0.0
Default gateway:	192.168.0.1
UDP Port #:	50000
TCP Port #:	5000
Default username:	Admin
Default password:	Admin
Full Factory Reset	
Recessed Button	Press and hold while powering the device.
Protocol 3000:	"#factory" and #Reset commands.
Web Pages:	In the Settings page, click Reset.

Default EDID

Monitor
  Model name............... DSP-62-AEC
  Manufacturer............. KMR
  Plug and Play ID......... KMR1200
  Serial number............ 295-883450100
  Manufacture date........ 2018, ISO week 255
  Filter driver............ None

  EDID revision............ 1.3
  Input signal type........ Digital
  Color bit depth.......... Undefined
  Display type............. Monochrome/grayscale
  Screen size.............. 520 x 320 mm (24.0 in)
  Power management......... Standby, Suspend, Active off/sleep
  Extension blocs.......... 1 (CEA/CTA-EXT)

  DDC/CI................... n/a

Color characteristics
  Default color space...... Non-sRGB
  Display gamma............ 2.20
  Red chromaticity......... Rx 0.674 - Ry 0.319
  Green chromaticity....... Gx 0.188 - Gy 0.706
  Blue chromaticity........ Bx 0.148 - By 0.064
  White point (default).... Wx 0.313 - Wy 0.329
  Additional descriptors... None

Timing characteristics
  Horizontal scan range.... 30-83kHz
  Vertical scan range...... 56-76Hz
  Video bandwidth.......... 170MHz
  CVT standard............. Not supported
  GTF standard............. Not supported
  Additional descriptors... None
  Preferred timing......... Yes
  Native/preferred timing.. 1920x1080p at 60Hz (16:9)
    Modeline.............. "1920x1080" 148.500 1920 2008 2052 2200 1080 1084 1089 1125 +hsync +vsync

Standard timings supported

720 x 400p at 70Hz - IBM VGA
720 x 400p at 88Hz - IBM XGA2
640 x 480p at 60Hz - IBM VGA
640 x 480p at 67Hz - Apple Mac II
640 x 480p at 72Hz - VESA
640 x 480p at 75Hz - VESA
800 x 600p at 56Hz - VESA
800 x 600p at 60Hz - VESA
800 x 600p at 72Hz - VESA
800 x 600p at 75Hz - VESA
832 x 624p at 75Hz - Apple Mac II
1024 x 768i at 87Hz - IBM
1024 x 768p at 60Hz - VESA
1024 x 768p at 70Hz - VESA
1024 x 768p at 75Hz - VESA
1280 x 1024p at 75Hz - VESA
1152 x 870p at 75Hz - Apple Mac II
1280 x 1024p at 75Hz - VESA STD
1280 x 1024p at 85Hz - VESA STD
1600 x 1200p at 60Hz - VESA STD
1024 x 768p at 85Hz - VESA STD
800 x 600p at 85Hz - VESA STD
640 x 480p at 85Hz - VESA STD
1152 x 864p at 70Hz - VESA STD
1280 x 960p at 60Hz - VESA STD

EIA/CEA/CTA-861 Information
 Revision number.......... 3
 IT underscan............. Supported
 Basic audio............. Supported
 YCbCr 4:4:4.............. Not supported
 YCbCr 4:2:2.............. Not supported
 Native formats........... 1
 Detailed timing #1....... 1920x1080p at 60Hz (16:10)
   Modeline.............. "1920x1080" 148.500 1920 2008 2052 2200 1080 1084 1089 1125 +hsync +vsync
 Detailed timing #2....... 1920x1080i at 60Hz (16:10)
   Modeline.............. "1920x1080" 74.250 1920 2008 2052 2200 1080 1084 1094 1124 interlace +hsync +vsync
 Detailed timing #3....... 1280x720p at 60Hz (16:10)
   Modeline.............. "1280x720" 74.250 1280 1390 1430 1650 720 725 730 750 +hsync +vsync
 Detailed timing #4....... 720x480p at 60Hz (16:10)
   Modeline.............. "720x480" 27.000 720 736 798 858 480 489 495 525 -hsync -vsync

CE audio data (formats supported)
 LPCM   2-channel, 16/20/24 bit depths at 32/44/48 kHz

CE video identifiers (VICs) - timing/formats supported
 1920 x 1080p at 60Hz - HDTV (16:9, 1:1) [Native]
 1920 x 1080i at 60Hz - HDTV (16:9, 1:1)
 1280 x 720p at 60Hz - HDTV (16:9, 1:1)
 720 x 480p at 60Hz - EDTV (16:9, 32:27)
 720 x 480p at 60Hz - EDTV (4:3, 8:9)
 720 x 480i at 60Hz - Doublescan (16:9, 32:27)
 720 x 576i at 50Hz - Doublescan (16:9, 64:45)
 640 x 480p at 60Hz - Default (4:3, 1:1)
 NB: NTSC refresh rate = (Hz*1000)/1001

CE vendor specific data (VSDB)
 IEEE registration number. 0x000C03
 CEC physical address..... 1.0.0.0
 Maximum TMDS clock....... 165MHz

CE speaker allocation data
 Channel configuration.... 2.0
 Front left/right......... Yes
 Front LFE................ No
 Front center............. No
 Rear left/right.......... No
 Rear center............. No
 Front left/right center.. No
 Rear left/right center... No
 Rear LFE................. No

Report information
 Date generated........... 02/12/2020
 Software revision........ 2.91.0.1043
 Data source.............. File - NB: improperly installed
 Operating system......... 10.0.18363.2

Raw data
00,FF,FF,FF,FF,FF,FF,00,2D,B2,00,12,00,00,00,00,FF,1C,01,03,80,34,20,78,E2,B3,25,AC,51,30,B4,26,
10,50,54,FF,FF,80,81,8F,81,99,A9,40,61,59,45,59,31,59,71,4A,81,40,02,3A,80,18,71,38,2D,40,58,2C,
45,00,A0,5A,00,00,00,1E,00,00,00,FF,00,32,39,35,2D,38,38,33,34,35,30,31,30,30,00,00,00,FC,00,44,
53,50,2D,36,32,2D,41,45,43,0A,20,20,00,00,00,FD,00,38,4C,1E,53,11,00,0A,20,20,20,20,20,20,01,5C,
02,03,1B,C1,23,09,07,07,48,90,05,04,03,02,07,16,01,65,03,0C,00,10,00,83,01,00,00,00,02,3A,80,18,71,
38,2D,40,58,2C,45,00,07,44,21,00,00,1E,01,1D,80,18,71,1C,16,20,58,2C,25,00,07,44,21,00,00,9E,01,

1D,00,72,51,D0,1E,20,6E,28,55,00,07,44,21,00,00,1E,8C,0A,D0,8A,20,E0,2D,10,10,3E,96,00,07,44,21,
00,00,18,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,77

Protocol 3000

Kramer devices can be operated using Kramer Protocol 3000 commands sent via serial or Ethernet ports.

Understanding Protocol 3000

Protocol 3000 commands are a sequence of ASCII letters, structured according to the following.

- **Command format**:

Prefix	Command Name	Constant (Space)	Parameter(s)	Suffix
#	Command	␣	Parameter	**<CR>**

- **Feedback format**:

Prefix	Device ID	Constant	Command Name	Parameter(s)	Suffix
~	nn	@	Command	Parameter	**<CR><LF>**

- **Command parameters** – Multiple parameters must be separated by a comma (,). In addition, multiple parameters can be grouped as a single parameter using brackets ([and]).

- **Command chain separator character** – Multiple commands can be chained in the same string. Each command is delimited by a pipe character (|).

- **Parameters attributes** – Parameters may contain multiple attributes. Attributes are indicated with pointy brackets (<…>) and must be separated by a period (.).

The command framing varies according to how you interface with **DSP-62-AEC**. The following figure displays how the # command is framed using terminal communication software (such as Hercules):
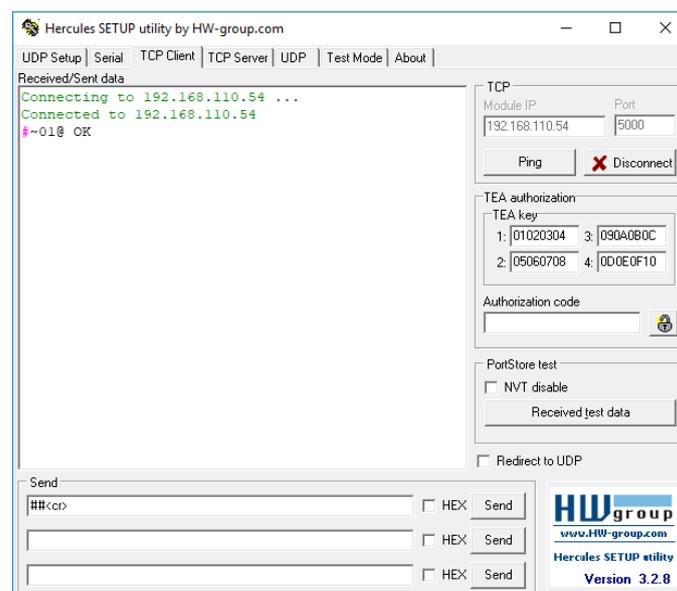
Protocol 3000 Commands

Function	Description	Syntax	Parameters/Attributes	Example
#	Protocol handshaking. ⓘ Validates the Protocol 3000 connection and gets the machine number. Step-in master products use this command to identify the availability of a device.	**COMMAND** **#<CR>** **FEEDBACK** **~nn@␣OK<CR><LF>**		**#<CR>**
AV-SW-TIMEOUT	Set auto switching timeout.	**COMMAND** **#AV-SW-TIMEOUT␣**action,time_out**<CR>** **FEEDBACK** **~nn@AV-SW-TIMEOUT␣**action,time_out**<CR><LF>**	**action** – 4– Disable 5V on video output if no input signal detected. **time_out** – Timeout in seconds 30 - 60000	Set the auto switching timeout to 5 seconds in the event of 5V disable when no input signal is detected: **#AV-SW-TIMEOUT␣**4,5**<CR>**
AV-SW-TIMEOUT?	Get auto switching timeout.	**COMMAND** **#AV-SW-TIMEOUT?␣**action**<CR>** **FEEDBACK** **~nn@AV-SW-TIMEOUT␣**action,time_out**<CR><LF>**	**action** – 4– Disable 5V on video output if no input signal detected **time_out** – Timeout in seconds 30 - 60000	Get the Disable 5V on video output if no input signal detected timeout: **#AV-SW-TIMEOUT?␣**4**<CR>**
BUILD-DATE?	Get device build date.	**COMMAND** **#BUILD-DATE?␣<CR>** **FEEDBACK** **~nn@BUILD-DATE␣**date,time**<CR><LF>**	**date** – Format: YYYY/MM/DD where YYYY = Year MM = Month DD = Day **time** – Format: hh:mm:ss where hh = hours mm = minutes ss = seconds	Get the device build date: **#BUILD-DATE?<CR>**
CONF-EXPORT	For Kramer internal use only.			
CONF-IMPORT	For Kramer internal use only.			
CPEDID	Copy EDID data from the output to the input EEPROM. ⓘ Destination bitmap size depends on device properties (for 64 inputs it is a 64-bit word). Example: bitmap 0x0013 means inputs 1,2 and 5 are loaded with the new EDID. In certain products Safe_mode is an optional parameter. See the HELP command for its availability.	**COMMAND** **#CPEDID␣**src_type,src_id,dst_type,dest_bitmap**<CR>** or **#CPEDID␣**src_type,src_id,dst_type,dest_bitmap,safe_mode**<CR>** **FEEDBACK** **~nn@CPEDID␣**src_stg,src_id,dst_type,dest_bitmap**<CR><LF>** **~nn@CPEDID␣**src_stg,src_id,st_type,dest_bitmap,safe_mode**<CR><LF>**	**src_type** – *EDID source type (usually output)* 0– Input 1– Output 2– Default EDID 3– Custom EDID **src_id** – Number of chosen source stage 0– Default EDID source 1– Output 1 **dst_type** – *EDID destination* type (usually input) 0– Input 1– Output 2– Default EDID 3– Custom EDID **dest_bitmap** – Bitmap representing destination IDs. Format: XXXX…X, where X is hex digit. The binary form of every hex digit represents corresponding destinations. 0– indicates that EDID data is not copied to this destination. 1– indicates that EDID data is copied to this destination. **safe_mode** – 0– device accepts the EDID as is without trying to adjust 1– device tries to adjust the EDID (default value if no parameter is sent)	Copy the EDID data from the Output (EDID source) to the Input: **#CPEDID␣**1,1,0,0x1**<CR>** Copy the EDID data from the default EDID source to the Input: **#CPEDID␣**2,0,0,0x1**<CR>**
DISPLAY?	Get output HPD status.	**COMMAND** **#DISPLAY?␣**input_id **<CR>** **FEEDBACK** **~nn@DISPLAY␣**out_id,status**<CR><LF>**	**out_id** – Output number 1– HDMI output **status** – HPD status according to signal validation 0– Signal or sink is not valid 1– Signal or sink is valid	Get the output HPD status of Output 1: **#DISPLAY?␣**1**<CR>**
DSP-AEC	Set DSP field value.	**COMMAND** **#DSP-AEC␣**field_id,<direction_type>.<port_type>.<port_index>.<signal_type>. <index>,value**<CR>** **FEEDBACK** **~nn@DSP-AEC␣**field_id,<direction_type1>.<port_type1>.<port_index1>.<signal_type1>. <index1>,value**<CR><LF>**	**Field_id** ○ bypass **<direction_type>** – IN ▪ **<port_type>** – ○ ANALOG_AUDIO ▪ **<port_index>** – The port number as printed on the front or rear panel 2 to 3 ▪ **<signal_type>** – ○ AUDIO ▪ **<index>** – 1 **value** – For low-freq and high-freq +20Hz to +20kHz (default-150) For bypass 0– Off 1– On	Set bypass on input 3 to off: **#DSP-AEC␣**bypass,IN.ANALOG_AUDIO.3.AUDIO.1,0**<CR>**

Function	Description	Syntax	Parameters/Attributes	Example

`DSP-AEC?`	Get DSP field value.	**COMMAND**
		`#DSP-AEC?␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>. <index><CR>`
		FEEDBACK
		`~nn@DSP-AEC␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index1>,value<CR><LF>`
			Field_id
			o bypass
			<direction_type> – IN
			▪**<port_type>** –
			o ANALOG_AUDIO
			▪**<port_index>** – The port number as printed on the front or rear panel
			2 to 3
			▪**<signal_type>** –
			o AUDIO
			▪**<index>** –
			1
			value –
			For low-freq and high-freq
			+20Hz to +20kHz (default-150)
			
			For bypass
			0– Off
			1 – On
				get low frequency on input 3:
				`#DSP-AEC?␣low-freq,IN.ANALOG_AUDIO.3.AUDIO.1<CR>`

`DSP-COMP`	Set DSP compressor values.	**COMMAND**
		`#DSP-COMP␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-COMP␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index1>,value<CR><LF>`
			Field_id
			o Attack
			o Release
			o Threshold
			o Ratio
			o Gain
			o Bypass
			<direction_type> – IN
			▪**<port_type>** –
			o ANALOG_AUDIO
			o ANALOG_STEREO
			o USB_B
			▪**<port_index>** – The port number as printed on the front or rear panel
			For ANALOG_AUDIO – 2 to 5
			For USB_B, ANALOG_STEREO – 1
			▪**<signal_type>** –
			o AUDIO
			▪**<index>** –
			For ANALOG_AUDIO – 1
			For USB_B, ANALOG_STEREO – 1, 2
			value –
			For attack time [ms]
			0 to 100
			For release time [ms]
			0 to 10K
			For threshold [dB]
			-100 to 0
			For ratio
			[1 to 100]:1
			For gain compensation [dB]
			-100 to +15
			For bypass [ms]
			0– off
			1– on
				Set attack time on input 3 to 15ms:
				`#DSP-COMP␣attack,IN.ANALOG_AUDIO.3.AUDIO.1,15<CR>`

`DSP-COMP?`	Get DSP compressor values.	**COMMAND**
		`#DSP-COMP?␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR>`
		FEEDBACK
		`~nn@DSP-COMP␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			Field_id
			o Attack
			o Release
			o Threshold
			o Ratio
			o Gain
			o Bypass
			<direction_type> – IN
			▪**<port_type>** –
			o ANALOG_AUDIO
			o ANALOG_STEREO
			o USB_B
			▪**<port_index>** – The port number as printed on the front or rear panel
			For ANALOG_AUDIO – 2 to 5
			For USB_B, ANALOG_STEREO – 1
			▪**<signal_type>** –
			o AUDIO
			▪**<index>** –
			For ANALOG_AUDIO – 1
			For USB_B, ANALOG_STEREO – 1, 2
			value –
			For attack time [ms]
			0 to 100
			For release time [ms]
			0 to 10K
			For threshold [dB]
			-100 to 0
			For ratio
			[1 to 100]:1
			For gain compensation [dB]
			-100 to +15
			For bypass [ms]
			0– off
			1– on
				Get attack time on input 3:
				`#DSP-COMP?␣attack time,IN.ANALOG_AUDIO.3.AUDIO.1<CR>`

Function	Description	Syntax	Parameters/Attributes	Example
DSP-DELAY	Set DSP delay.	**COMMAND** #DSP-DELAY␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR> **FEEDBACK** ~nn@DSP-DELAY␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type1>.<index>,value<CR><LF>	**Field_id** o Delay o Bypass **<direction_type>** – o IN o OUT ▪ **<port_type>** – For IN o HDMI_AUDIO○ For OUT o USB_B ▪ **<port_index>** – The port number as printed on the front or rear panel 1 ▪ **<signal_type>** – o AUDIO ▪ **<index>** – 1 to 2 **value** – For delay time [ms] 0 to 150 For bypass [ms] 0– off 1– on	Set delay time on USB channel 1 to 75: #DSP-DELAY␣delay,OUT.USB_B.1.AUDIO.1,75<CR>
DSP-DELAY?	Get DSP delay.	**COMMAND** #DSP-DELAY?␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> **FEEDBACK** ~nn@DSP-DELAY␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	**Field_id** o Delay o Bypass **<direction_type>** – o IN o OUT ▪ **<port_type>** – For IN o HDMI_AUDIO○ For OUT o USB_B ▪ **<port_index>** – The port number as printed on the front or rear panel 1 ▪ **<signal_type>** – o AUDIO ▪ **<index>** – 1 to 2 **value** – For delay time [ms] 0 to 150 For bypass [ms] 0– off 1– on	Get bypass status on HDMI input 1 to 75: #DSP-DELAY?␣bypass,IN.HDMI_AUDIO.1.AUDIO.1<CR>
DSP-EQ	Set DSP equalizer.	**COMMAND** #DSP-EQ␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,band,value<CR> **FEEDBACK** ~nn@DSP-EQ␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,band,value<CR><LF>	**Field_id** o level O freq O qfactor O band-bypass **<direction_type>** – o IN o OUT ▪ **<port_type>** – For IN o ANALOG_STEREO o ANALOG_AUDIO o USB_B For OUT o ANALOG_STEREO o USB_B ▪ **<port_index>** – The port number as printed on the front or rear panel 1 to 5 ▪ **<signal_type>** – o AUDIO ▪ **<index>** – 1 to 2 **band** – For IN 1 to 4 For OUT 1 to 8 For bypass 0– global bypass 1,2,3,4 – for each band **value** – For level [dB] -24 to +24 For freq [Hz] 20 to 20K For qfactor [Oct] 0.05 to 4 For bypass [ms] 0– off 1– on	Set EQ level on input 2 to 12: #DSP-EQ␣level,IN.ANALOG_AUDIO.2.AUDIO.1,?,12<CR>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-EQ?	Set DSP equalizer.	**COMMAND** **#DSP-EQ?**␣field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,band<CR> **FEEDBACK** **~nn@DSP-EQ**␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,band,value<CR><LF>	**Field_id** o Level o Freq o Qfactor o Band-bypass **<direction_type>** – o IN o OUT ▪**<port_type>** – For IN o ANALOG_STEREO o ANALOG_AUDIO o USB_B For OUT o ANALOG_STEREO o USB_B ▪**<port_index>** – The port number as printed on the front or rear panel 1 to 5 ▪**<signal_type>** – o AUDIO ▪**<index>** – 1 to 2 **band –** For IN 1 to 4 For OUT 1 to 8 For band-bypass 0– global bypass 1,2,3,4 – for each band **value –** For level [dB] -24 to +24 For freq [Hz] 20 to 20K For qfactor [Oct] 0.05 to 4 For bypass [ms] 0– off 1– on	Get global bypass state for EQ on input 2: **#DSP-EQ?**␣bypass,IN.ANALOG_AUDIO.2.AUDIO.1,0<CR>
DSP-EXP	Set DSP expander.	**COMMAND** **#DSP-EXP**␣field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR> **FEEDBACK** **~nn@DSP-EXP**␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	**Field_id** o Attack o Release o Threshold o Ratio o Bypass **<direction_type>** – IN **<port_type>** – o ANALOG_AUDIO o ANALOG_STEREO o USB_B **<port_index>** – The port number as printed on the front or rear panel For ANALOG_AUDIO – 2 to 5 For USB_B, ANALOG_STEREO – 1 **<signal_type>** – o AUDIO **<index>** – 1 **value –** For attack [ms] 0 to 100 For release time [ms] 0 to 10K For threshold [dB] -100 to 0 For ratio [1 to 100]:1 For gain compensation [dB] -100 to +15 For bypass [ms] 0– off 1– on	Set attack time on input 2 to 50: **#DSP-EXP**␣attack,IN.ANALOG_AUDIO.2.AUDIO.1,50<CR>

Function	Description	Syntax	Parameters/Attributes	Example

`DSP-EXP?`	Get DSP expander.	**COMMAND**
		`#DSP-`
		`EXP?⎵field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index><CR>`
		FEEDBACK
		`~nn@DSP-EXP⎵field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR><LF>`
			Field_id
			o Attack
			o Release
			o Threshold
			o Ratio
			o Bypass
			<direction_type> – IN
			<port_type> –
			o ANALOG_AUDIO
			o ANALOG_STEREO
			o USB_B
			<port_index> – The port number as printed on the front or rear panel
			For ANALOG_AUDIO – 2 to 5
			For USB_B, ANALOG_STEREO – 1
			<signal_type> –
			o AUDIO
			<index> – 1
			value –
			For attack [ms]
			0 to 100
			For release [ms]
			0 to 10K
			For threshold [dB]
			-100 to 0
			For ratio
			1 to 100:1
			For bypass
			0– off
			1– on
				Get attack time on input 2:
				
				`#DSP-EXP?⎵attack,IN.ANALOG_AUDIO.2.AUDIO.1<CR>`

`DSP-HPF`	Set DSP HPF.	**COMMAND**
		`#DSP-`
		`HPF⎵field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-HPF⎵field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR><LF>`
			Field_id
			o freq
			o bypass
			<direction_type> – IN
			▪**<port_type>** –
			o ANALOG_AUDIO
			▪**<port_index>** – The port number as printed on the front or rear panel
			2 to 5
			▪**<signal_type>** –
			o AUDIO
			▪**<index>** – 1
			value –
			For freq [Hz]
			20 to 20K
			For bypass
			0– off
			1– on
				Set bypass status on input 2 to off:
				
				`#DSP-HPF⎵bypass,IN.ANALOG_AUDIO.2.AUDIO.1,0<CR>`

`DSP-HPF?`	Get DSP HPF.	**COMMAND**
		`#DSP-`
		`HPF?⎵field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-HPF⎵field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR><LF>`
			Field_id
			o freq
			o bypass
			<direction_type> – IN
			▪**<port_type>** –
			o ANALOG_AUDIO
			▪**<port_index>** – The port number as printed on the front or rear panel
			2 to 5
			▪**<signal_type>** –
			o AUDIO
			▪**<index>** – 1
			value –
			For freq [Hz]
			20 to 20K
			For bypass
			0– off
			1– on
				Get bypass status on input 2:
				
				`#DSP-HPF?⎵bypass,IN.ANALOG_AUDIO.2.AUDIO.1<CR>`

`DSP-INVERT`	Set DSP phase inversion state.	**COMMAND**
		`#DSP-`
		`INVERT⎵<direction_type>.<port_type>.<port_index>.<signal type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-INVERT⎵<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			<direction_type> – IN
			▪**<port_type>** –
			o ANALOG_AUDIO
			▪**<port_index>** – The port number as printed on the front or rear panel
			2 to 5
			▪**<signal_type>** –
			o AUDIO
			▪**<index>** – 1
			value –
			0– off
			1– on
				Set phase inversion state input 2 to off:
				
				`#DSP-INVERT⎵IN.ANALOG_AUDIO.2.AUDIO.1,0<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

`DSP-INVERT?`	Get DSP phase inversion state.	**COMMAND**
		`#DSP-`
		`INVERT?`␣`<direction_type>.<port_type>.<port_index>.<signal type>.<index><CR>`
		FEEDBACK
		`~nn@DSP-INVERT`␣`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			`<direction_type>` – IN
			▪**`<port_type>`** –
			  o ANALOG_AUDIO
			▪**`<port_index>`** – The port number as printed on the front or rear panel
			  2 to 5
			▪**`<signal_type>`** –
			  o AUDIO
			▪**`<index>`** – 1
			`value` –
			0– off
			1– on
				Get phase inversion state on input 2:
				
				`#DSP-INVERT?`␣`IN.ANALOG_AUDIO.2.AUDIO.1<CR>`

`DSP-LIMITER`	Set DSP limiter.	**COMMAND**
		`#DSP-`
		`LIMITER`␣`field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-LIMITER`␣`field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			Field_id
			  o Threshold
			  o Bypass
			`<direction_type>` – OUT
			▪**`<port_type>`** –
			  o ANALOG_STEREO
			  o USB_B
			▪**`<port_index>`** – The port number as printed on the front or rear panel
			  1
			▪**`<signal_type>`** –
			  o AUDIO
			▪**`<index>`** – 1 to 2
			`value` –
			For threshold [dB]
			-100 to 0
			For bypass
			0– Off
			1– On
				Set bypass status on output to ON:
				
				`#DSP-LIMITER`␣`bypass,OUT.ANALOG_STEREO.1.AUDIO.1,1<CR>`

`DSP-LIMITER?`	Get DSP limiter.	**COMMAND**
		`#DSP-`
		`LIMITER?`␣`field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR>`
		FEEDBACK
		`~nn@DSP-LIMITER`␣`field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			Field_id
			  o Threshold
			  o Bypass
			`<direction_type>` – OUT
			▪**`<port_type>`** –
			  o ANALOG_STEREO
			  o USB_B
			▪**`<port_index>`** – The port number as printed on the front or rear panel
			  1
			▪**`<signal_type>`** –
			  o AUDIO
			▪**`<index>`** – 1 to 2
			`value` –
			For threshold [dB]
			-100 to 0
			For bypass
			0– Off
			1– On
				Get bypass status on output:
				
				`#DSP-LIMITER?`␣`bypass,OUT.ANALOG_STEREO.2.AUDIO.1<CR>`

`DSP-METER?`	Read DSP meters.	**COMMAND**
		`#DSP-`
		`METER`␣`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,type<CR>`
		FEEDBACK
		`~nn@DSP-METER`␣`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,type,value<CR><LF>`
			`<direction_type>` –
			  o IN
			  o OUT
			▪**`<port_type>`** –
			  o HDMI
			  o HDMI_AUDIO
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			  o GENERATOR
			▪**`<port_index>`** – The port number as printed on the front or rear panel
			  1 to 5
			▪**`<signal_type>`** –
			  o AUDIO
			▪**`<index>`** – 1
			`type` –
			1– Gain.
			2– Post-gain (for output only).
			3– Expander (for input only).
			4– Compressor (for input only).
			5– Limiter (for output only).
			`value` – [dBFS]
				Read the limiter value on the output:
				
				`#DSP-METER`␣`bypass,OUT.ANALOG_STEREO.1.AUDIO.1,5<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

`DSP-POST`	Set DSP post volume faders/mute.	**COMMAND**
		`#DSP-POST_field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-POST_field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			Field_id
			o Level
			o Mute
			\<direction_type\> – IN
			▪ **\<port_type\>** –
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			▪ **\<port_index\>** – The port number as printed on the front or rear panel
			  1 to 5
			▪ **\<signal_type\>** –
			  o AUDIO
			▪ **\<index\>** – 1
			value –
			For level [dB]
			-100 to +15
			For mute
			0– Off
			1– On
				Set mute status on input 2 to ON:
				
				`#DSP-POST_mute,IN.ANALOG_AUDIO.2.AUDIO.1,1<CR>`

`DSP-POST?`	Set DSP post volume faders/mute.	**COMMAND**
		`#DSP-POST?_field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR>`
		FEEDBACK
		`~nn@DSP-POST_field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			Field_id
			o Level
			o Mute
			\<direction_type\> – IN
			▪ **\<port_type\>** –
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			▪ **\<port_index\>** – The port number as printed on the front or rear panel
			  1 to 5
			▪ **\<signal_type\>** –
			  o AUDIO
			▪ **\<index\>** – 1
			value –
			For level [dB]
			-100 to +15
			For mute
			0– Off
			1– On
				Get mute status on input 2:
				`#DSP-POST?_mute,IN.ANALOG_AUDIO.2.AUDIO.1<CR>`

`DSP-SIG-GEN`	Set DSP signal generator.	**COMMAND**
		`#DSP-SIG-GEN_field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR>`
		FEEDBACK
		`~nn@DSP-SIG-GEN_field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>`
			Field_id
			o Mode
			o Tone-freq
			o Tone-lvl
			o Pink-lvl
			o Bypass
			\<direction_type\> – IN
			▪ **\<port_type\>** –
			  o GENERATOR
			▪ **\<port_index\>** – The port number as printed on the front or rear panel
			  1
			▪ **\<signal_type\>** –
			  o AUDIO
			▪ **\<index\>** – 1
			value –
			for mode
			1– Tone
			2– Pink noise
			For tone-freq [Hz]
			20 to 20K
			For tone-level [dB]
			-100 to +15
			For pink-level [dB]
			-100 to +15
			For bypass
			0– Off
			1– On
				Set signal generator to pink noise mode on input 2:
				`#DSP-SIG-GEN_mode,IN.GENERATOR.1.AUDIO.1,2<CR>`

Function	Description	Syntax	Parameters/Attributes	Example
DSP-SIG-GEN?	Get DSP signal generator.	**COMMAND** #DSP-SIG-GEN?␣field_id,<direction_type>.<port_type>.<port_index>.<signal type>.<index><CR> **FEEDBACK** ~nn@DSP-SIG-GEN␣field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	**Field_id** o Mode o Tone-freq o Tone-lvl o Pink-lvl o Bypass **<direction_type>** – IN ▪**<port_type>** – o GENERATOR ▪**<port_index>** – The port number as printed on the front or rear panel 1 ▪**<signal_type>** – o AUDIO ▪**<index>** – 1 **value –** for mode 1– Tone 2– Pink noise For tone-freq [Hz] 20 to 20K For tone-level [dB] -100 to +15 For pink-level [dB] -100 to +15 For bypass 0– Off 1– On	Get signal generator mode on input 2: #DSP-SIG-GEN?␣mode,IN.GENERATOR.1.AUDIO.1<CR>
EDID-AUDIO	Set audio capabilities for EDID.	**COMMAND** #EDID-AUDIO␣input_id,audio_format<CR> **FEEDBACK** ~nn@EDID-AUDIO␣input_id,audio_format <CR><LF>	**input_id** – 1 – HDMI IN 1 2 – HDMI IN 2 **Audio_format** – Audio block added to EDID: 0– Auto 1– LPCM 2CH	Set HDMI IN 1 audio capabilities for EDID (LPCM 6CH): #EDID-AUDIO␣1,2<CR>
EDID-AUDIO?	Get audio capabilities for EDID.	**COMMAND** #EDID-AUDIO?␣input_id<CR> **FEEDBACK** ~nn@EDID-AUDIO␣input_id,audio_format<CR><LF>	**input_id** – 1 – HDMI IN 1 2 – HDMI IN 2 **Audio_format** – Audio block added to EDID: 0– Auto 1– LPCM 2CH	Get HDMI IN 1 audio capabilities for EDID: #EDID-AUDIO?␣1<CR>
EDID-CS	Set EDID color space. ⓘ Set command might change the current EDID.	**COMMAND** #EDID-CS␣ input_id,ColSpace<CR> **FEEDBACK** ~nn@EDID-CS␣ input_id,ColSpace<CR><LF>	**input_id** – 1 **ColSpace** – Color space 0– RGB 4– auto	Set HDMI IN 1 EDID color space to RGB (enabled): #EDID-CS␣1,0<CR>
EDID-CS?	Get EDID color space. ⓘ Get command might change the current EDID.	**COMMAND** #EDID-CS?␣input_id<CR> **FEEDBACK** ~nn@EDID-CS␣input_id,ColSpace<CR><LF>	**input_id** – 1 **ColSpace** – Color space 0– RGB 4– auto	Get EDID color space: #EDID-CS?␣1<CR>
ETH-PORT	Set Ethernet port protocol. ⓘ If the port number you enter is already in use, an error is returned. The port number must be within the following range: 0-(2^16-1).	**COMMAND** #ETH-PORT␣portType,ETHPort<CR> **FEEDBACK** ~nn@ETH-PORT␣portType,ETHPort<CR><LF>	**portType** – TCP/UDP **ETHPort** – TCP/UDP port number (0 – 65535)	Set the Ethernet port protocol for TCP to port 12457: #ETH-PORT␣0,12457<CR>
ETH-PORT?	Get Ethernet port protocol.	**COMMAND** #ETH-PORT?␣portType<CR> **FEEDBACK** ~nn@ETH-PORT␣portType,ETHPort<CR><LF>	**portType** – TCP/UDP 0 – TCP 1 – UDP **ETHPort** – TCP / UDP port number (0 – 65535)	Get the Ethernet port protocol for UDP: #ETH-PORT?␣1<CR>
FACTORY	Reset device to factory default configuration. ⓘ This command deletes all user data from the device. The deletion can take some time. Your device may require powering off and powering on for the changes to take effect.	**COMMAND** #FACTORY<CR> **FEEDBACK** ~nn@FACTORY␣OK<CR><LF>		Reset the device to factory default configuration: #FACTORY<CR>
FILE-HANDLED	For internal use only.			
FEATURE-LIST?	Get feature state according to the feature ID.	**COMMAND** #FEATURE-LIST?␣feature_id<CR> **FEEDBACK** ~nn@FEATURE-LIST␣feature_id,ir_state<CR><LF>	**Feature_Id** – Feature ID 1 – Maestro 2 – Room Controller **Ir_State** – IR Interface 0 – Disabled 1 – Enabled	Get the room controller feature state (for the room controller 1): #FEATURE-LIST?␣1<CR>
FW-TYPE?	Get the current FW type status. Used by Kramer Network and KUpload to identify recovery process.	**COMMAND** #FW-TYPE?␣<CR> **FEEDBACK** ~nn@FEATURE-LIST␣fw_type<CR><LF>	**Fw_type** – 0 – Application 1 – Safe mode (kboot)	Get the current FW type status: #FW-TYPE?␣<CR>

Function	Description	Syntax	Parameters/Attributes	Example

GPIO-CFG	Set HW GPIO configuration.	**COMMAND**
		
		`#GPIO-CFG_HwGpioNum,HwGpioType,HwGpioDir,Pullup<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-CFG_`
		`HwGpioNum,HwGpioType,HwGpioDir,Pullup<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			HwGpioType – Hardware GPIO type
			  0 – analog
			  1 – digital
			HwGpioDir – Hardware GPIO direction
			  0 – input
			  1 – output
			Pullup – Enable/Disable pull-up
			  0 – disable
			  1 – enable
				Set HW GPIO configuration:
				`#GPIO-CFG_1,1,1,1<CR>`

GPIO-CFG?	Set HW GPIO configuration.	**COMMAND**
		
		`#GPIO-CFG?_HwGpioNum <CR>`
		
		FEEDBACK
		
		`~nn@GPIO-CFG_HwGpioNum,HwGpioType,HwGpioDir,Pullup<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			HwGpioType – Hardware GPIO type
			  0 – analog
			  1 – digital
			HwGpioDir – Hardware GPIO direction
			  0 – input
			  1 – output
			Pullup – Enable/Disable pull-up
			  0 – disable
			  1 – enable
				Get HW GPIO configuration:
				`#GPIO-CFG?_1<CR>`

GPIO-STATE
 	
Currently not in use.
	Set HW GPIO state.
	
	ⓘ This GPIO-STATE can only be set in digital out mode and the answer is 0=Low, 1=High. In all other modes an error message is sent.
	
	The device uses this command to notify the user of any change regarding the step and voltage in:
	
	In digital mode the answer is 0 (low), 1 (high).
	
	In analog mode the answer is 0 to 100.
		COMMAND
		
		`#GPIO-STATE_HwGpioNum,HwGpioState<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-STATE_HwGpioNum,HwGpioState<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			HwGpioState – Hardware GPIO state
			  0 – low
			  1 – High
				Set GPIO 2 to high:
				`#GPIO-STATE_2,1<CR>`

GPIO-STATE?
 	
Currently not in use.
	Get HW GPIO state.
	
	ⓘ This GPIO-STATE can only be set in digital out mode and the answer is 0=Low, 1=High. In all other modes an error message is sent.
	
	The device uses this command to notify the user of any change regarding the step and voltage in:
	
	In digital mode the answer is 0 (low), 1 (high).
		COMMAND
		
		`#GPIO-STATE?_HwGpioNum<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-STATE_HwGpioNum,HwGpioState<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			HwGpioState – Hardware GPIO state
			  0 – low
			  1 – High
				Get HW GPIO configuration:
				`#GPIO-STATE?_1<CR>`

GPIO-STEP
 	
Currently not in use.
	Set HW GPIO step.
	
	ⓘ In digital mode the response is 2.
	In analog mode the response is 1 to 100.
	In other modes an error is returned
		COMMAND
		
		`#GPIO-STEP_HwGpioNum,Step<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-STEP_HwGpioNum,NumOfStep,CurrentStep<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			NumOfStep – The configuration step – See note in description.
			CurrentStep – The actual step depending on the measured voltage
				Set GPIO 2 (set to Analog In) configuration step to 38mV:
				`#GPIO-STEP_2,38<CR>`

GPIO-STEP?
 	
Currently not in use.
	Get HW GPIO step.
	
	ⓘ In digital mode the response is 2.
	In analog mode the response is 1 to 100.
	In other modes an error is returned
		COMMAND
		
		`#GPIO-STEP?_HwGpioNum<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-STATE_HwGpioNum,NumOfStep,CurrentStep<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			NumOfStep – The configuration step – See note in description.
			CurrentStep – The actual step depending on the measured voltage
				Get GPIO 2 configuration:
				`#GPIO-STEP?_2<CR>`

GPIO-THR
 	
Currently not in use.
	Set HW GPIO voltage levels.	**COMMAND**
		
		`#GPIO-THR_HwGpioNum,LowLevel,HighLevel<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-THR_HwGpioNum,LowLevel,HighLevel<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			LowLevel – Voltage 500 to 28000 millivolts
			HighLevel – Voltage 2000 to 30000 millivolts
				Set GPIO 2 to a low level of 800mV and a high level of 2200mV:
				`#GPIO-THR_2,800,2200<CR>`

GPIO-THR?
 	
Currently not in use.
	Get HW GPIO voltage levels that were set.	**COMMAND**
		
		`#GPIO-THR?_HwGpioNum<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-THR_HwGpioNum,LowLevel,HighLevel<CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			LowLevel – Voltage 500 to 28000 millivolts
			HighLevel – Voltage 2000 to 30000 millivolts
				Get GPIO 2 voltage levels:
				`#GPIO-THR?_2<CR>`

GPIO-VOLT?
 	
Currently not in use.
	Get active voltage levels of HW GPIO.
	
	ⓘ This command is not available in digital out mode.
		COMMAND
		
		`#GPIO-VOLT?_HwGpioNum<CR>`
		
		FEEDBACK
		
		`~nn@GPIO-VOLT_HwGpioNum,Voltage <CR><LF>`
			HwGpioNum – Hardware GPIO number (1 to 2)
			Voltage – Voltage 0 to 30000 millivolts
				Get GPIO 2 active voltage levels:
				`#GPIO-VOLT?_2<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

HDCP-MOD	Set HDCP mode.
	
	ⓘ Set HDCP working mode on the device input:
	
	HDCP supported - HDCP_ON [default].
	
	HDCP not supported - HDCP OFF.
	
	HDCP support changes following detected sink - MIRROR OUTPUT.
	
	When you define 3 as the mode, the HDCP status is defined according to the connected output in the following priority: OUT 1, OUT 2. If the connected display on OUT 2 supports HDCP, but OUT 1 does not, then HDCP is defined as not supported. If OUT 1 is not connected, then HDCP is defined by OUT 2.
		COMMAND
		`#HDCP-MOD␣inp_id,mode<CR>`
		FEEDBACK
		`~nn@HDCP-MOD␣inp_id,mode<CR><LF>`
			inp_id – Input number:
			  1 – HDMI IN 1
			  2 – HDMI IN 2
			mode – HDCP mode:
			  0 – HDCP Off
			  3 – HDCP defined according to the connected output
				Set the input HDCP-MODE of HDMI IN 1 to Off:
				`#HDCP-MOD␣1,0<CR>`

GPIO-MOD?	Get HDCP mode.
	
	ⓘ Set HDCP working mode on the device input:
	
	HDCP supported - HDCP_ON [default].
	
	HDCP not supported - HDCP OFF.
	
	HDCP support changes following detected sink - MIRROR OUTPUT.
		COMMAND
		`#HDCP-MOD?␣inp_id<CR>`
		FEEDBACK
		`~nn@HDCP-MOD␣inp_id,mode<CR><LF>`
			inp_id – Input number:
			  1 – HDMI IN 1
			  2 – HDMI IN 2
			mode – HDCP mode:
			  0 – HDCP Off
			  3 – HDCP defined according to the connected output
				Get the input HDCP-MODE of HDMI IN 1:
				`#HDCP-MOD?␣1<CR>`

HELP	Get command list or help for specific command.	**COMMAND**
		`#HELP<CR>`
		`#HELP␣cmd_name<CR>`
		FEEDBACK
		1. Multi-line:
		`~nn@Device␣cmd_name,cmd_name…<CR><LF>`
		To get help for command use: HELP (COMMAND_NAME)`<CR><LF>`
		`~nn@HELP␣cmd_name:<CR><LF>`
		`description<CR><LF>`
		`USAGE:usage<CR><LF>`
			cmd_name – Name of a specific command	Get the command list:
				`#HELP<CR>`
				
				Get help for AV-SW-TIMEOUT:
				`HELP␣av-sw-timeout<CR>`

HDCP-STAT?	Get HDCP signal status.
	
	ⓘ Output stage (1) – get the HDCP signal status of the sink device connected to the specified output.
	
	Input stage (0) – get the HDCP signal status of the source device connected to the specified input.
		COMMAND
		`#HDCP-STAT?␣stage,stage_id<CR>`
		FEEDBACK
		`~nn@HDCP-STAT␣stage,stage_id,status<CR><LF>`
			stage – Input/Output
			  0 – Input
			  1 – Output
			stage_id – Number of chosen stage
			for the input stage
			  1 – HDMI IN 1
			  2 – HDMI IN 2
			For the output stage
			  1 – HDMI OUT
			status – Signal encryption status -
			valid values On/Off
			  0 – HDCP Off
			  1 – HDCP On
				Get the output HDCP-STATUS of HDMI IN:
				`#HDCP-STAT?␣0,1<CR>`

Function	Description	Syntax	Parameters/Attributes	Example
LOGIN	Set protocol permission. ⓘ For devices that support security, LOGIN allows the user to run commands with an End User or Administrator permission level. When the permission system is enabled, LOGIN enables running commands with the User or Administrator permission level When set, login must be performed upon each connection The permission system works only if security is enabled with the "SECUR" command. It is not mandatory to enable the permission system in order to use the device In each device, some connections allow logging in to different levels. Some do not work with security at all. Connection may logout after timeout.	**COMMAND** **#LOGIN**␣login_level,password**<CR>** **FEEDBACK** **~nn@LOGIN**␣login_level,password␣OK**<CR><LF>** or **~nn@LOGIN**␣ERR␣004**<CR><LF>** (if bad password entered)	**login_level** – Level of permissions required (User or Admin) **password** – Predefined password (by PASS command). Default password is an empty string	Set the protocol permission level to Admin (when the password defined in the PASS command is 33333): **#LOGIN**␣Admin,33333**<CR>**
LOGIN?	Get current protocol permission level. ⓘ For devices that support security, LOGIN allows the user to run commands with an End User or Administrator permission level. In each device, some connections allow logging in to different levels. Some do not work with security at all. Connection may logout after timeout. The permission system works only if security is enabled with the "SECUR" command.	**COMMAND** **#LOGIN?**␣**<CR>** **FEEDBACK** **~nn@LOGIN**␣login_level**<CR><LF>**	**login_level** – Level of permissions required (User or Admin)	Get current protocol permission level: **#LOGIN?<CR>**
LOGOUT	Cancel current permission level. ⓘ Logs out from End User or Administrator permission levels to Not Secure.	**COMMAND** **#LOGOUT<CR>** **FEEDBACK** **~nn@LOGOUT**␣OK**<CR><LF>**		**#LOGOUT<CR>**
MODEL?	Get device model. ⓘ This command identifies equipment connected to **DSP-62-AEC** and notifies of identity changes to the connected equipment. The Matrix saves this data in memory to answer REMOTE-INFO requests.	**COMMAND** **#MODEL?**␣**<CR>** **FEEDBACK** **~nn@MODEL**␣model_name**<CR><LF>**	**model_name** – String of up to 19 printable ASCII chars	Get the device model: **#MODEL?**␣**<CR>**
NAME	Set machine (DNS) name. ⓘ The machine name is not the same as the model name. The machine name is used to identify a specific machine or a network in use (with DNS feature on).	**COMMAND** **#NAME**␣machine_name**<CR>** **FEEDBACK** **~nn@NAME**␣machine_name**<CR><LF>**	**machine_name** – String of up to 15 alpha-numeric chars (can include hyphen, not at the beginning or end)	Set the DNS name of the device to room-442: **#NAME**␣room-442**<CR>**

Function	Description	Syntax	Parameters/Attributes	Example

NAME?	Get machine (DNS) name.
	
	ⓘ The machine name is not the same as the model name. The machine name is used to identify a specific machine or a network in use (with DNS feature on).
		COMMAND
		`#NAME?␣<CR>`
		FEEDBACK
		`~nn@NAME␣machine_name<CR><LF>`
			machine_name – String of up to 15 alpha-numeric chars (can include hyphen, not at the beginning or end)	Get the DNS name of the device:
				`#NAME?␣<CR>`

NAME-RST	Reset machine (DNS) name to factory default.
	
	ⓘ Factory default of machine (DNS) name is "KRAMER_" + 4 last digits of device serial number.
		COMMAND
		`#NAME-RST<CR>`
		FEEDBACK
		`~nn@NAME-RST␣OK<CR><LF>`
				Reset the machine name (S/N last digits are 0102):
				`#NAME-RST␣KRAMER_0102<CR>`

NET-CONFIG	Set a network configuration.
	
	ⓘ Parameters, **[DNS1]** and **[DNS2]** are optional.
	
	ⓘ For Backward compatibility, the **id** parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.
	
	ⓘ If the gateway address is not compliant to the subnet mask used for the host IP, the command will return an error. Subnet and gateway compliancy specified by RFC950.
		COMMAND
		`#NET-CONFIG␣id,ip,net_mask,gateway,[DNS1],[DNS2]<CR>`
		FEEDBACK
		`~nn@NET-CONFIG␣id,ip,net_mask,gateway<CR><LF>`
			id – Network ID–the device network interface (if there are more than one). Counting is 0 based, meaning the control port is '0', additional ports are 1,2,3….
			ip – Network IP
			net_mask – Network mask
			gateway – Network gateway
				Set the device network parameters to IP address 192.168.113.10, net mask 255.255.0.0, and gateway 192.168.0.1:
				`#NET-CONFIG␣0,192.168.113.10,255.255.0.0,192.168.0.1<CR>`

NET-CONFIG?	Get a network configuration.	**COMMAND**
		`#NET-CONFIG?␣id<CR>`
		FEEDBACK
		`~nn@NET-CONFIG␣id,ip,net_mask,gateway<CR><LF>`
			id – Network ID–the device network interface (if there are more than one). Counting is 0 based, meaning the control port is '0', additional ports are 1,2,3….
			ip – Network IP
			net_mask – Network mask
			gateway – Network gateway
				Get network configuration:
				`#NET-CONFIG?␣id<CR>`

NET-DHCP	Set DHCP mode.
	
	ⓘ Only 1 is relevant for the mode value. To disable DHCP, the user must configure a static IP address for the device.
	
	Connecting Ethernet to devices with DHCP may take more time in some networks.
	
	To connect with a randomly assigned IP by DHCP, specify the device DNS name (if available) using the **NAME** command. You can also get an assigned IP by direct connection to USB or RS-232 protocol port, if available.
	
	For proper settings consult your network administrator.
	
	ⓘ For Backward compatibility, the **id** parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.
		COMMAND
		`#NET-DHCP␣id,mode<CR>`
		FEEDBACK
		`~nn@NET-DHCP␣id,mode<CR><LF>`
			id – 0
			mode –
			1– Try to use DHCP. (If unavailable, use the IP address set by the factory or the NET-IP command).
				Enable DHCP mode for port 1, if available:
				`#NET-DHCP␣1,1<CR>`

NET-DHCP?	Get DHCP mode.
	ⓘ For Backward compatibility, the **id** parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.
		COMMAND
		`#NET-DHCP?␣id<CR>`
		FEEDBACK
		`~nn@NET-DHCP␣id,mode<CR><LF>`
			id –0
			mode –
			0– Do not use DHCP. Use the IP set by the factory or using the **NET-IP** or **NET-CONFIG** command.
			1– Try to use DHCP. If unavailable, use the IP set by the factory or using the **NET-IP** or **NET-CONFIG** command.
				Get DHCP mode for port 1:
				`#NET-DHCP?␣1<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

NET-GATE	Set gateway IP.
	
	ⓘ A network gateway connects the device via another network and maybe over the Internet. Be careful of security issues. For proper settings consult your network administrator.
		COMMAND
		`#NET-GATE␣ip_address<CR>`
		FEEDBACK
		`~nn@NET-GATE␣ip_address<CR><LF>`
			`ip_address` – Format: xxx.xxx.xxx.xxx	Set the gateway IP address to 192.168.0.1:
				`#NET-GATE␣192.168.000.001<CR>`

NET-GATE?	Get gateway IP.
	
	ⓘ A network gateway connects the device via another network and maybe over the Internet. Be aware of security problems.
		COMMAND
		`#NET-GATE?␣<CR>`
		FEEDBACK
		`~nn@NET-GATE␣ip_address<CR><LF>`
			`ip_address` – Format: xxx.xxx.xxx.xxx	Get the gateway IP address:
				`#NET-GATE?␣<CR>`

NET-IP	Set IP address.
	
	ⓘ For proper settings consult your network administrator.
		COMMAND
		`#NET-IP␣ip_address<CR>`
		FEEDBACK
		`~nn@NET-IP␣ip_address<CR><LF>`
			`ip_address` – Format: xxx.xxx.xxx.xxx	Set the IP address to 192.168.1.39:
				`#NET-IP␣192.168.001.039<CR>`

NET-IP?	Get IP address.	**COMMAND**
		`#NET-IP?␣<CR>`
		FEEDBACK
		`~nn@NET-IP␣ip_address<CR><LF>`
			`ip_address` – Format: xxx.xxx.xxx.xxx	Get the IP address:
				`#NET-IP?␣<CR>`

NET-MAC?	Get MAC address.
	ⓘ For backward compatibility, the `id` parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.
		COMMAND
		`#NET-MAC?␣id<CR>`
		FEEDBACK
		`~nn@NET-MAC␣id,mac_address<CR><LF>`
			`id` – Network ID–the device network interface (if there are more than one). Counting is 0 based, meaning the control port is '0', additional ports are 1,2,3….
			`mac_address` – Unique MAC address. Format: XX-XX-XX-XX-XX-XX where X is hex digit
				`#NET-MAC?␣id<CR>`

NET-MASK	Set subnet mask.
	ⓘ For proper settings consult your network administrator.
		COMMAND
		`#NET-MASK␣net_mask<CR>`
		FEEDBACK
		`~nn@NET-MASK␣net_mask<CR><LF>`
			`net_mask` – Format: xxx.xxx.xxx.xxx	Set the subnet mask to 255.255.0.0:
				`#NET-MASK␣255.255.000.000<CR>`

NET-MASK?	Get subnet mask.	**COMMAND**
		`#NET-MASK?␣<CR>`
		FEEDBACK
		`~nn@NET-MASK␣net_mask<CR><LF>`
			`net_mask` – Format: xxx.xxx.xxx.xxx	Get the subnet mask:
				`#NET-MASK?<CR>`

PASS	Set password for login level.
	
	ⓘ The default password is an empty string.
		COMMAND
		`#PASS␣login_level,password<CR>`
		FEEDBACK
		`~nn@PASS␣login_level,password<CR><LF>`
			`login_level` – Level of login to set (End User or Administrator).
			`password` – Password for the *login_level*. Up to 15 printable ASCII chars
				Set the password for the Admin protocol permission level to 33333:
				`#PASS␣Admin,33333<CR>`

PASS?	Get password for login level.
	
	ⓘ The default password is an empty string.
		COMMAND
		`#PASS?␣login_level<CR>`
		FEEDBACK
		`~nn@PASS␣login_level,password<CR><LF>`
			`login_level` – Level of login to set (End User or Administrator).
			`password` – Password for the *login_level*. Up to 15 printable ASCII chars
				Get the password for the Admin protocol permission level:
				`#PASS?␣Admin<CR>`

PORTS-LIST?	Get the port list of this machine.
	
	ⓘ The response is returned in one line and terminated with <CR><LF>.
	
	The response format lists port IDs separated by commas.
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#PORTS-LIST?␣<CR>`
		FEEDBACK
		`~nn@PORTS-LIST␣[<direction_type>.<port_type>.<port_index>,..]<CR><LF>`
			The following attributes comprise the port ID:
			▪ `<direction_type>` –
			 ○ IN
			 ○ OUT
			▪ `<port_type>` –
			 ○ HDMI
			 ○ HDMI_AUDIO
			 ○ ANALOG_AUDIO
			 ○ ANALOG_STEREO
			 ○ USB_B
			 ○ GENERATOR
			▪ `<port_index>` – The port number as printed on the front or rear panel
				Get the ports list:
				`#PORTS-LIST?␣<CR>`

PROT-VER?	Get device protocol version.	**COMMAND**
		`#PROT-VER?␣<CR>`
		FEEDBACK
		`~nn@PROT-VER␣3000:version<CR><LF>`
			`version` – XX.XX where X is a decimal digit	Get the device protocol version:
				`#PROT-VER?␣<CR>`

RESET	Reset device.	**COMMAND**
		`#RESET<CR>`
		FEEDBACK
		`~nn@RESET␣OK<CR><LF>`
				Reset the device:
				`#RESET<CR>`

SIGNAL?	Get input signal status.	**COMMAND**
		`#SIGNAL?␣inp_id<CR>`
		FEEDBACK
		`~nn@SIGNAL␣inp_id,status<CR><LF>`
			`inp_id` – Input number
			1 – HDMI IN1
			2 – HDMI IN2
			`status` – Signal status according to signal validation:
			0 – Off
			1 – On
				Get the input signal lock status of HDMI IN 1:
				`#SIGNAL?␣1<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

SIGNALS-LIST?	Get signal ID list of this machine.
	
	ⓘ The response is returned in one line and terminated with <CR><LF>.
	
	The response format lists signal IDs separated by commas.
	
	This is an Extended Protocol 3000 command.
		COMMAND
		#SIGNALS-LIST?␣<CR><LF>
		FEEDBACK
		~nn@SIGNALS-LIST␣[<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,..]<CR><LF>
			The following attributes comprise the signal ID:
			▪ **<direction_type>** –
			  o IN
			  o OUT
			▪ **<port_type>** –
			  o HDMI
			  o HDMI_AUDIO
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			  o GENERATOR
			▪ **<port_index>** – The port number as printed on the front or rear panel
			▪ **<signal_type>** –
			  o AUDIO
			  o VIDEO
			▪ **<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2
				Get signal ID list:
				#SIGNALS-LIST?␣<CR>

SN?	Get device serial number.	COMMAND #SN?␣<CR> FEEDBACK ~nn@SN␣**serial_number**<CR><LF>	**serial_number** – 14 decimal digits, factory assigned	Get the device serial number: #SN?␣<CR>
TIME?	Get device time and date. ⓘ The year must be 4 digits. The device does not validate the day of week from the date. Time format - 24 hours. Date format - Day, Month, Year.	**COMMAND** #TIME?␣<CR> **FEEDBACK** ~nn@TIME␣day_of_week,date,time<CR><LF>	**day_of_week** – One of {SUN,MON,TUE,WED,THU,FRI,SAT} **date** – Format: YYYY/MM/DD where YYYY = Year MM = Month DD = Day **time** – Format: hh:mm:ss where hh = hours mm = minutes ss = seconds	Get device time and date: #TIME?<CR>
TIME-LOC?	Get local time offset from UTC/GMT. ⓘ If the time server is configured, device time calculates by adding UTC_off to UTC time (that it got from the time server) + 1 hour if daylight savings time is in effect. TIME command sets the device time without considering these settings.	**COMMAND** #TIME-LOC?␣<CR> **FEEDBACK** ~nn@TIME-LOC␣UTC_off,DayLight<CR><LF>	**UTC_off** – Offset of device time from UTC/GMT (without daylight time correction) **DayLight** – 0– no daylight saving time 1– daylight saving time	Get local time offset from UTC/GMT: #TIME-LOC?<CR>
TIME-SRV	Get time server. ⓘ This command is needed for setting UDP timeout for the current client list.	**COMMAND** #TIME-SRV␣mode,time_server_IP,time_server_Sync_Hour<CR> **FEEDBACK** ~nn@TIME-SRV␣mode time_server_ip time_server_Sync_Hour server_status<CR><LF>	**mode** – On/Off 0– Off 1– On **time_server_ip** – Time server IP address **time_server_Sync_Hour** – Hour in day for time server sync **server_status** – 0– Off 1– On	Set time server with IP address of 128.138.140.44 to ON: #TIME-SRV␣1,128.138.140.44,0,1<CR>
TIME-SRV?	Get time server. ⓘ This command is needed for setting UDP timeout for the current client list.	**COMMAND** #TIME-SRV?␣<CR> **FEEDBACK** ~nn@TIME-SRV␣mode,time_server_ip,time_server_Sync_Hour,server_status<CR><LF>	**mode** – On/Off 0– Off 1– On **time_server_ip** – Time server IP address **time_server_Sync_Hour** – Hour in day for time server sync **server_status** – 0– Off 1– On	Get time server: #TIME-SRV?<CR>
VERSION?	Get firmware version number.	**COMMAND** #VERSION?␣<CR> **FEEDBACK** ~nn@VERSION␣firmware_version<CR><LF>	**firmware_version** – XX.XX.XXXX where the digit groups are: major.minor.build version	Get the device firmware version number: #VERSION?␣<CR>

Function	Description	Syntax	Parameters/Attributes	Example

X-AUD-LVL	Set audio level of a specific signal.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-AUD-LVL␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,audio_level**\<CR>**
		FEEDBACK
		~nn@X-AUD-LVL␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,audio_level**\<CR>\<LF>**
			The following attributes comprise the signal ID:
			▪ **\<direction_type>** –
			  o IN
			  o OUT
			▪ **\<port_type>** –
			  o HDMI_AUDIO
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			  o GENERATOR
			▪ **\<port_index>** – The port number as printed on the front or rear panel
			▪ **\<signal_type>** –
			  o AUDIO
			▪ **\<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2.
			audio_level – Audio level in dB (range between -60 to +30) depending of the ability of the product
				Set the audio level of analog audio specific signal to 10:
				#X-AUD-LVL␣IN.ANALOG_AUDIO.5.AUDIO.1,10**\<CR>**

X-AUD-LVL?	Get audio level of a specific signal.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-AUD-LVL?␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>**\<CR>**
		FEEDBACK
		~nn@X-AUD-LVL␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,audio_level**\<CR>\<LF>**
			The following attributes comprise the signal ID:
			▪ **\<direction_type>** –
			  o IN
			  o OUT
			▪ **\<port_type>** –
			  o HDMI_AUDIO
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			  o GENERATOR
			▪ **\<port_index>** – The port number as printed on the front or rear panel
			▪ **\<signal_type>** –
			  o AUDIO
			▪ **\<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2.
			audio_level – Audio level in dB (range between -60 to +30) depending of the ability of the product
				Get the audio level of a specific signal:
				#X-AUD-LVL?␣OUT.ANALOG_AUDIO.1.AUDIO.1**\<CR>**

X-AUD-MODE	Set line/Mic mode.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-AUD-MODE␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,mode**\<CR>**
		FEEDBACK
		~nn@X-AUD-MODE␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,mode**\<CR>\<LF>**
			The following attributes comprise the signal ID:
			▪ **\<direction_type>** –
			  o IN
			▪ **\<port_type>** –
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			▪ **\<port_index>** – The port number as printed on the front or rear panel 1 to 5.
			▪ **\<signal_type>** –
			  o AUDIO
			▪ **\<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2.
			mode –
			1 – Line
			2 – Mic
				Set AUDIO IN 5 to Mic mode:
				#X-AUD-MODE␣IN.ANALOG_AUDIO.5.AUDIO.1,2**\<CR>**

X-AUD-MODE?	Get line/Mic mode.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-AUD-MODE?␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>**\<CR>**
		FEEDBACK
		~nn@X-AUD-MODE␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,mode**\<CR>\<LF>**
			The following attributes comprise the signal ID:
			▪ **\<direction_type>** –
			  o IN
			▪ **\<port_type>** –
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			▪ **\<port_index>** – The port number as printed on the front or rear panel 1 to 5.
			▪ **\<signal_type>** –
			  o AUDIO
			▪ **\<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2.
			mode –
			1 – Line
			2 – Mic
				Get AUDIO IN 5 to audio mode:
				#X-AUD-MODE?␣IN.ANALOG_AUDIO.5.AUDIO.1**\<CR>**

Function	Description	Syntax	Parameters/Attributes	Example
X-LABEL	Set the port label. ⓘ Labels are used commonly by webpages. This is an Extended Protocol 3000 command.	**COMMAND** **#X-LABEL␣direction_type>.<port_type>.<port_index>.<signal_type>.<index>,label_text<CR><LF>** **FEEDBACK** **~nn@X-LABEL␣direction_type>.<port_type>.<port_index>.<signal_type>.<index>,label_text<CR><LF>**	The following attributes comprise the signal ID: ▪ **<direction_type>** – 　o IN 　o OUT ▪ **<port_type>** – 　o HDMI 　o HDMI_AUDIO 　o ANALOG_AUDIO 　o ANALOG_STEREO 　o USB_B 　o GENERATOR ▪ **<port_index>** – The port number as printed on the front or rear panel ▪ **<signal_type>** – 　o AUDIO 　o VIDEO ▪ **<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2 **Label_text** – ASCII characters without space	Set the port label for HDMI IN1: **#X-LABEL␣**IN.HDMI.1.VIDEO.1,Blu_ray**<CR>**
X-LABEL?	Get the port label. ⓘ Labels are used commonly by webpages. This is an Extended Protocol 3000 command.	**COMMAND** **#X-LABEL?␣direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR><LF>** **FEEDBACK** **~nn@X-LABEL␣direction_type>.<port_type>.<port_index>.<signal_type>.<index>,label_text<CR><LF>**	The following attributes comprise the signal ID: **<direction_type>** – 　IN 　OUT **<port_type>** – 　o HDMI 　o HDMI_AUDIO 　o ANALOG_AUDIO 　o ANALOG_STEREO 　o USB_B 　o GENERATOR **<port_index>** – The port number as printed on the front or rear panel **<signal_type>** – 　AUDIO 　VIDEO **<index>** – Indicates a specific channel number when there are multiple channels of the same type 1 to 2 **Label_text** – ASCII characters without space	Get the port label for HDMI IN1: **#X-LABEL?␣**IN.HDMI.1.VIDEO.1**<CR>**
X-LINK-GROUP	Set link between signals in a group: Get linked signals groups info, this is a way to define group of signals for which any action made on one of them will be applied to all the members of the group. ⓘ This is an Extended Protocol 3000 command.	**COMMAND** **#X-LINK-GROUP␣group_id,linked_state<CR>** **FEEDBACK** **~nn@X-LINK-GROUP␣group_id,linked_state<CR><LF>**	**group_id** – 1 – Input analog audio 2 and 3. 2 – Input analog audio 4 and 5. **Linked_state** – OFF/ON (not case sensitive)	Set link for group 1 (AUDIO IN 2 and 3) to OFF: **#X-LINK-GROUP␣**1,OFF**<CR>**
X-LINK-GROUP?	GET LINK-MODE feature: Get linked signals groups info, this is a way to define group of signals for which any action made on one of them will be applied to all the members of the group. ⓘ This is an Extended Protocol 3000 command. Used essentially by the web command.	**COMMAND** **#X-LINK-GROUP?␣group_id<CR>** **FEEDBACK** **~nn@X-LINK-GROUP␣group_id,linked_state<CR><LF>**	**group_id** – 1 – Input analog audio 2 and 3. 2 – Input analog audio 4 and 5. **Linked_state** – OFF/ON (not case sensitive)	Get the group 1 link status: **#X-LINK-GROUP?␣**1**<CR>**

Function	Description	Syntax	Parameters/Attributes	Example

X-LINK-GROUPS-LIST?	LINK-MODE feature:
	
	Get linked signals groups info, this is a way to define group of signals for which any action made on one of them will be applied to all member of the group.
	The LINK mode of a group is defined using the command: **X-LINK-GROUP**
	
	ⓘ This is an Extended Protocol 3000 command. Used essentially by the web
		COMMAND
		#X-LINK-GROUPS-LIST?\<CR>
		FEEDBACK
		~nn@X-LINK-GROUPS-LIST␣[[group_id,is_linked,[signal_id,...,signal_id]]**\<CR>\<LF>**
			group_id – 1 to 2
			Linked_state – OFF/ON (not case sensitive)
			signal_id – includes:
			▪ **\<direction_type>** –
			  o IN
			▪ **\<port_type>** –
			  o ANALOG_AUDIO
			▪ **\<index>** – 2 to 5
			▪ **\<signal_type>** –
			  AUDIO
			\<index> – Indicates a specific channel number when there are multiple channels of the same type 1 to 2
				Get the link state for all ports:
				#X-LINK-GROUPS-LIST?␣\<CR>
				
				[[1,OFF,[IN.ANALOG_AUDIO.3.AUDIO.1,IN.ANALOG_AUDIO.2.AUDIO.1]],[2,OFF,[IN.ANALOG_AUDIO.5.AUDIO.1,IN.ANALOG_AUDIO.4.AUDIO.1]]

X-MIC-TYPE	Set microphone type.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-MIC-TYPE␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,mic_type**\<CR>**
		FEEDBACK
		~nn@X-MIC-TYPE␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,mic_type**\<CR>\<LF>**
			The following attributes comprise the port ID:
			▪ **\<direction_type>** –
			  o IN
			▪ **\<port_type>** –
			  o ANALOG_AUDIO
			▪ **\<port_index>** – The port number as printed on the front or rear panel: 2 to 5.
			\<signal_type> –
			  AUDIO
			\<index> – Indicates a specific channel number when there are multiple channels of the same type: 1 to 2
			mic_type – Dynamic/Condenser (not case sensitive)
				Set AUDIO IN 2 type to condenser:
				#X-MIC-TYPE␣IN.ANALOG_AUDIO.3.AUDIO.1,condenser\<CR>

X-MIC-TYPE?	Get microphone type.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-MIC-TYPE?␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>**\<CR>**
		FEEDBACK
		~nn@X-MIC-TYPE␣\<direction_type>.\<port_type>.\<port_index>.\<signal_type>.\<index>,mic_type **\<CR>\<LF>**
			The following attributes comprise the port ID:
			▪ **\<direction_type>** –
			  o IN
			▪ **\<port_type>** –
			  o ANALOG_AUDIO
			▪ **\<port_index>** – The port number as printed on the front or rear panel: 2 to 5.
			\<signal_type> –
			  AUDIO
			\<index> – Indicates a specific channel number when there are multiple channels of the same type: 1 to 2
			mic_type – Dynamic/Condenser (not case sensitive)
				Get MIC 3 type:
				#X-MIC-TYPE?␣IN.MIC.3\<CR>

X-MIX-LVL	Set DSP matrix cross-point MIX level in dB.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		#X-MIX-LVL␣OUT.\<port_type>.\<port_index>.\<signal_type>.\<index>,IN.\<port_type>.\<port_index>.\<signal_type>.\<index>,dB**\<CR>**
		FEEDBACK
		~nn@X-MIX-LVL␣OUT.\<port_type>.\<port_index>.\<signal_type>.\<index>,IN.\<port_type>.\<port_index>.\<signal_type>.\<index>,dB**\<CR>\<LF>**
			The following attributes comprise the primary signal ID (suffix 1) and follower signal ID (suffix 2 or greater):
			▪ **\<direction_type>** – IN
			  o IN
			  o OUT
			▪ **\<port_type>** –
			  o HDMI
			  o HDMI_AUDIO
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			  o GENERATOR
			▪ **\<port_index>** – The port number as printed on the front or rear panel
			▪ **\<signal_type>** –
			  o AUDIO
			▪ **\<index>** – Indicates a specific channel number when there are multiple channels of the same type
				Set analog audio output 1 and USB-B (R) cross-point level to -25dB:
				#X-MIX-LVL␣OUT.ANALOG_STEREO.1.AUDIO.1,IN.USB_B.1.AUDIO.2,-25\<CR>

Function	Description	Syntax	Parameters/Attributes	Example

`X-MIX-LVL?`	Get DSP matrix cross-point MIX level in dB.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-MIX-LVL?␣OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index><CR>`
		FEEDBACK
		`~nn@X-MIX-LVL␣OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index>,dB<CR><LF>`
			The following attributes comprise the primary signal ID (suffix 1) and follower signal ID (suffix 2 or greater):
			▪ **`<direction_type>`** – IN
			  ○ IN
			  ○ OUT
			▪ **`<port_type>`** –
			  ○ HDMI
			  ○ HDMI_AUDIO
			  ○ ANALOG_AUDIO
			  ○ ANALOG_STEREO
			  ○ USB_B
			  ○ GENERATOR
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  ○ AUDIO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
				Get analog audio output 1 and HDMI audio cross-point level:
				`#X-MIX-LVL?␣OUT.ANALOG_STEREO.1.AUDIO.1,IN.HDMI_AUDIO.1.AUDIO.1<CR>`

`X-MIX-MUTE`	Set DSP matrix cross-point mute state.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-MIX-MUTE␣OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index>,dB<CR>`
		FEEDBACK
		`~nn@X-MIX-MUTE␣OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index>,<mute_state><CR><LF>`
			The following attributes comprise the primary signal ID (suffix 1) and follower signal ID (suffix 2 or greater):
			▪ **`<direction_type>`** – IN
			  ○ IN
			  ○ OUT
			▪ **`<port_type>`** –
			  ○ HDMI
			  ○ HDMI_AUDIO
			  ○ ANALOG_AUDIO
			  ○ ANALOG_STEREO
			  ○ USB_B
			  ○ GENERATOR
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  ○ AUDIO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			▪ **`<mute_state>`** –
			  ○ ON
			  ○ OFF
				Mute analog audio output 1 and HDMI audio cross-point:
				`#X-MIX-MUTE␣OUT.ANALOG_STEREO.1.AUDIO.1,IN.HDMI_AUDIO.1.AUDIO.1,ON<CR>`

`X-MIX-MUTE?`	Get DSP matrix cross-point mute state.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-MIX-MUTE?␣OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index><CR>`
		FEEDBACK
		`~nn@X-MIX-MUTE␣OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index>,<mute_state><CR><LF>`
			The following attributes comprise the primary signal ID (suffix 1) and follower signal ID (suffix 2 or greater):
			▪ **`<direction_type>`** – IN
			  ○ IN
			  ○ OUT
			▪ **`<port_type>`** –
			  ○ HDMI
			  ○ HDMI_AUDIO
			  ○ ANALOG_AUDIO
			  ○ ANALOG_STEREO
			  ○ USB_B
			  ○ GENERATOR
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  ○ AUDIO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			▪ **`<mute_state>`** –
			  ○ ON
			  ○ OFF
				Get analog audio output 1 and HDMI audio cross-point mute state:
				`#X-MIX-MUTE?␣OUT.ANALOG_STEREO.1.AUDIO.1,IN.HDMI_AUDIO.1.AUDIO.1<CR>`

`X-MUTE`	Set the mute state of the signal.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-MUTE␣<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,state<CR>`
		FEEDBACK
		`~nn@X-MUTE␣<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,state<CR><LF>`
			The following attributes comprise the signal ID:
			▪ **`<direction_type>`** –
			  ○ IN
			  ○ OUT
			▪ **`<port_type>`** –
			  ○ HDMI
			  ○ HDMI_AUDIO
			  ○ ANALOG_AUDIO
			  ○ ANALOG_STEREO
			  ○ USB_B
			  ○ GENERATOR
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  ○ VIDEO
			  ○ AUDIO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			`state` – OFF/ON (not case sensitive)
				Set the mute state of analog audio (L) output to off:
				`#X-MUTE␣OUT.ANALOG_STEREO.1.AUDIO.2,OFF<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

X-MUTE?	Get the mute state of the signal.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-MUTE?`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>`**`<CR>`**
		FEEDBACK
		`~nn@X-MUTE`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,state`**`<CR><LF>`**
			The following attributes comprise the signal ID:
			▪ **`<direction_type>`** –
			  o IN
			  o OUT
			▪ **`<port_type>`** –
			  o HDMI
			  o HDMI_AUDIO
			  o ANALOG_AUDIO
			  o ANALOG_STEREO
			  o USB_B
			  o GENERATOR
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  o VIDEO
			  o AUDIO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			`state` – OFF/ON (not case sensitive)
				Get the mute state of analog audio (L) output to off:
				`#X-MUTE?`‿`OUT.ANALOG_STEREO.1.AUDIO.2`**`<CR>`**

X-PATTERN	Set a pattern on the selected output.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PATTERN`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_id`**`<CR>`**
		FEEDBACK
		`~nn@X-PATTERN`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_id`**`<CR><LF>`**
			The following attributes comprise the signal ID:
			▪ **`<direction_type>`** –
			  o OUT
			▪ **`<port_type>`** –
			  o HDMI
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  o VIDEO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			`Pattern_id` –pattern ID
			  o 0 : none
			  o 1 : Color bar
			  o 2 Blue screen
			  o 3: Green screen
			  o 4: Red screen
				Set the pattern on HDMI OUT to pattern 2 (blue screen):
				`#X-PATTERN`‿`OUT.HDMI.1.VIDEO AUDIO.1,2`**`<CR>`**

X-PATTERN?	Get the pattern on a selected output.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PATTERN?`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>`**`<CR>`**
		FEEDBACK
		`~nn@X-PATTERN`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_id`**`<CR><LF>`**
			The following attributes comprise the signal ID:
			▪ **`<direction_type>`** –
			  o OUT
			▪ **`<port_type>`** –
			  o HDMI
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  o VIDEO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			`Pattern_id` –pattern ID
			  o 0 : none
			  o 1 : Color bar
			  o 2 Blue screen
			  o 3: Green screen
			  o 4: Red screen
				Get the pattern on HDMI output:
				`#X-PATTERN?`‿`OUT.HDMI.1.VIDEO.1`**`<CR>`**

X-PATTERNS-LIST?	Get the pattern list of a selected output.
	
	ⓘ This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PATTERN-LIST?`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>`**`<CR>`**
		FEEDBACK
		`~nn@X-PATTERNS-LIST`‿`<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_list`**`<CR><LF>`**
			The following attributes comprise the signal ID:
			▪ **`<direction_type>`** –
			  o OUT
			▪ **`<port_type>`** –
			  o HDMI
			▪ **`<port_index>`** – The port number as printed on the front or rear panel
			▪ **`<signal_type>`** –
			  o VIDEO
			▪ **`<index>`** – Indicates a specific channel number when there are multiple channels of the same type
			`Pattern_id` –pattern ID
			  o 0 : none
			  o 1 : Color bar
			  o 2 Blue screen
			  o 3: Green screen
			  o 4: Red screen
				Get the pattern list for HDMI OUT:
				`#X-PATTERNS-LIST?`‿`OUT.HDMI.1.VIDEO.1`**`<CR>`**

Function	Description	Syntax	Parameters/Attributes	Example

X-PRST-CURR?	Get the current preset loaded per type.
	
	To get the list of preset types existing in your product use the command:
	　X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PRST-CURR?␣preset_type<CR>`
		FEEDBACK
		`~nn@X-PRST-CURR␣<preset_type,[preset_id:name:lock_state]<CR><LF>`
			▪ **preset_type** –
			　○ System Preset – IOCONFIG.SYSTEM
			　○ Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **<preset_id>** –preset index
			▪ **<name>** – the name of the preset in URL encode format
			▪ **<lock_state>** –
			　○ ON
			　○ OFF
				Get current mixer preset:
				`X-PRST-CURR?␣IOCONFIG.SYSTEM.MIXER<CR>`

X-PRST-LOCK	Set LOCK state of a preset per type.
	
	ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	　X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PRST-LOCK␣preset_type,preset_id,lock_state<CR>`
		FEEDBACK
		`~nn@X-PRST-LOCK␣<preset_type,[preset_id:name:lock_state]<CR><LF>`
			▪ **preset_type** –
			　○ System Preset – IOCONFIG.SYSTEM
			　○ Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **<preset_id>** –preset index
			▪ **<lock_state>** –
			　○ ON
			　○ OFF
				lock mixer preset 9:
				`X-PRST-LOCK␣IOCONFIG.SYSTEM.MIXER,9<CR>`

X-PRST-LOCK?	Get LOCK state of a preset per type.
	
	ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	　X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PRST-LOCK?␣preset_type,preset_id,lock_state<CR>`
		FEEDBACK
		`~nn@X-PRST-LOCK␣<preset_type,[preset_id:name:lock_state]<CR><LF>`
			▪ **preset_type** –
			　○ System Preset – IOCONFIG.SYSTEM
			　○ Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **<preset_id>** –preset index
			▪ **<lock_state>** –
			　○ ON
			　○ OFF
				Get lock mixer preset 9 status:
				`X-PRST-LOCK?␣IOCONFIG.SYSTEM.MIXER,9<CR>`

X-PRST-LST?	Get the preset list of a specific preset type.
	
	ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	　X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PRST-LST?␣preset_type<CR>`
		FEEDBACK
		`~nn@X-PRST-LST␣<preset_type,[preset_id:name:lock_state]<CR><LF>`
			▪ **preset_type** –
			　○ System Preset – IOCONFIG.SYSTEM
			　○ Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **<name>** – the name of the preset
			▪ **<lock_state>** –
			　○ ON
			　○ OFF
				Get the IO configuration list:
				`X-PRST-LST?␣IOCONFIG<CR>`
				`[[1:4x16:ON],[2:6x14:ON],[3:8x12:ON],[4:10x10:ON],[5:12x8:ON],[6:14x6:ON],[7:16x4:ON]]`

X-PRST-NAME	Set the name of a preset per type.
	
	ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	　X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-PRST-NAME␣preset_type,preset_id,name<CR>`
		FEEDBACK
		`~nn@X-PRST-NAME␣preset_type,preset_id,name<CR><LF>`
			▪ **preset_type** –
			　○ System Preset – IOCONFIG.SYSTEM
			　○ Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **preset_id** – preset index
			▪ **name** – the name of the preset in URL encode format (no spaces)
				Set the name of a preset (per type):
				`X-PRST-NAME␣IOCONFIG.SYSTEM.MIXER,9,ROOM1<CR>`

Function	Description	Syntax	Parameters/Attributes	Example

X-PRST-NAME?	Get the name of a preset per type.
	
	(i) this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		#X-PRST-NAME?␣preset_type,preset_id,name**<CR>**
		FEEDBACK
		~nn@X-PRST-NAME␣preset_type,preset_id,name**<CR><LF>**
			▪ **preset_type** –
			  o System Preset – IOCONFIG.SYSTEM
			  o Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **preset_id** – preset index
			▪ **name** – the name of the preset in URL encode format
				Get the name of a preset (per type):
				X-PRST-NAME?␣IOCONFIG.SYSTEM.MIXER,9**<CR>**

X-PRST-RCL	Recall saved preset list per type.
	
	(i) this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		#X-PRST-RCL␣preset_type,preset_id**<CR>**
		FEEDBACK
		~nn@X-PRST-RCL␣preset_type,preset_id**<CR><LF>**
			▪ **preset_type** –
			  o System Preset – IOCONFIG.SYSTEM
			  o Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **preset_id** – preset index
				Recall mixer preset 8:
				X-PRST-RCL␣IOCONFIG.SYSTEM.MIXER,8**<CR>**

X-PRST-RCL-LAST	Recall LAST preset per type, this command just retrieves the last preset loaded from the history of preset activity and RECALLs it.
	
	(i) this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		#X-PRST-RCL-LAST␣preset_type **<CR>**
		FEEDBACK
		~nn@X-PRST-RCL-LAST␣preset_type,preset_id**<CR><LF>**
			▪ **preset_type** –
			  o System Preset – IOCONFIG.SYSTEM
			  o Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **preset_id** – preset index
				Recall the last mixer preset:
				X-PRST-RCL-LAST␣IOCONFIG.SYSTEM.MIXER**<CR>**

X-PRST-RCL-NEXT	Recall NEXT preset per type, this command increments by one the current preset id loaded and loads it. If the index is the highest, recall will fail.
	
	(i) this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system.
	
	To get the list of preset types existing in your product use the command:
	X-PRST-TYPES?
	
	This is an Extended Protocol 3000 command.
		COMMAND
		#X-PRST-RCL-NEXT␣preset_type**<CR>**
		FEEDBACK
		~nn@X-PRST-RCL-NEXT␣preset_type,preset_id**<CR><LF>**
			▪ **preset_type** –
			  o System Preset – IOCONFIG.SYSTEM
			  o Snapshot – IOCONFIG.SYSTEM.MIXER
			▪ **preset_id** – preset index
				Recall next mixer preset:
				X-PRST-RCL-NEXT␣IOCONFIG.SYSTEM.MIXER**<CR>**

Function	Description	Syntax	Parameters/Attributes	Example
X-PRST-RCL-PREV	Recall previous preset per type, this command increments by one the current preset id loaded and loads it. If the index is the lowest, recall will fail. ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system. To get the list of preset types existing in your product use the command: X-PRST-TYPES? This is an Extended Protocol 3000 command.	**COMMAND** **#X-PRST-RCL-PREV␣**preset_type**<CR>** **FEEDBACK** **~nn@X-PRST-RCL-PREV␣**preset_type,preset_id**<CR><LF>**	▪ **preset_type** – o System Preset – IOCONFIG.SYSTEM o Snapshot – IOCONFIG.SYSTEM.MIXER ▪ **preset_id** – preset index	Recall previous mixer preset: **X-PRST-RCL-PREV␣IOCONFIG.SYSTEM.MIXER<CR>**
X-PRST-RESET	Reset preset per type ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system. To get the list of preset types existing in your product use the command: X-PRST-TYPES? This is an Extended Protocol 3000 command.	**COMMAND** **#X-PRST-RESET␣**preset_type,preset_id**<CR>** **FEEDBACK** **~nn@X-PRST-RESET␣**preset_type,preset_id**<CR><LF>**	▪ **preset_type** – o System Preset – IOCONFIG.SYSTEM o Snapshot – IOCONFIG.SYSTEM.MIXER ▪ **preset_id** – preset index	Reset mixer preset 9: **X-PRST-RESET␣IOCONFIG.SYSTEM.MIXER,9<CR>**
X-PRST-SAVED?	Get SAVED status for a preset type. This flag indicates to the WEB if a change have been made since the last RECALL and has not been saved. ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system. To get the list of preset types existing in your product use the command: X-PRST-TYPES? This is an Extended Protocol 3000 command.	**COMMAND** **#X-PRST-SAVED?␣**preset_type**<CR>** **FEEDBACK** **~nn@X-PRST-SAVED␣**preset_type,saved_status**<CR><LF>**	▪ **preset_type** – o System Preset – IOCONFIG.SYSTEM o Snapshot – IOCONFIG.SYSTEM.MIXER ▪ **Saved_status** – preset index o 0 – False (not saved) o 1 – True (saved)	Get saved status of mixer preset: **X-PRST-SAVED?␣IOCONFIG.SYSTEM.MIXER<CR>**
X-PRST-STO	Store current changes into a preset (per type). ⓘ this is an extended preset command using preset type as first parameter. This is used essentially when we have different types of Presets inside the same system. To get the list of preset types existing in your product use the command: X-PRST-TYPES? This is an Extended Protocol 3000 command.	**COMMAND** **#X-PRST-STO␣**preset_type,preset_id**<CR>** **FEEDBACK** **~nn@X-PRST-STO␣**preset_type,saved_status**<CR><LF>**	▪ **preset_type** – o System Preset – IOCONFIG.SYSTEM o Snapshot – IOCONFIG.SYSTEM.MIXER ▪ **preset_id** – preset index	Store changes into mixer preset 9: **X-PRST-STO␣IOCONFIG.SYSTEM.MIXER,9<CR>**

Function	Description	Syntax	Parameters/Attributes	Example

X-PRST-TYPES?	Get the types of presets that the system supports and their hierarchy.	**COMMAND**
		`#X-PRST-TYPES?␣<CR>`
		FEEDBACK
		`~nn@X-PRST-TYPES␣preset_type <CR><LF>`
			• **preset_type** –
			 ○ IOCONFIG.SYSTEM – used for system preset per IOConfig, we have 10 preset banks per IOConfig setup, Preset #1 is the default system preset for this setup and is READ ONLY, Preset #2 is used for the first user system preset, Preset #3 for the second etc.
			 ○ IOCONFIG.SYSTEM.MIXER – used for a Mixer snapshot of a specific system preset per IOConfig. There are 10 MIXER snapshots per System presets in each IOConfig setup, Snapshot #1 is the default MIXER snapshot and is READ ONLY. Snapshot #2 is used for the first user Mixer snapshot, Snapshot #3 for the second etc.
				Get preset types:
				`X-PRST-TYPES?␣<CR>`

X-ROUTE	Send routing command to matrix.
	
	ⓘ It is recommended to use the command `#SIGNALS-LIST` to get the list of all signal IDs available in the system and which can be used in this command.
	
	Video 1 is the default port in this command and is implied even if not written.
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-ROUTE␣<direction_type1>.<port_type1>.<port_index1>.<signal_type1>.<index1>,<direction_type2>.<port_type2>.<port_index2>.<signal_type2>.<index2><CR>`
		FEEDBACK
		`~nn@X-ROUTE␣<direction_type1>.<port_type1>.<port_index1>.<signal_type1>.<index1>,<direction_type2>.<port_type2>.<port_index2>.<signal_type2>.<index2><CR><LF>`
			The following attributes comprise the output signal ID (suffix 1) and input signal ID (suffix 2):
			▪ **<direction_type>** –
			 ○ OUT
			 ○ IN
			▪ **<port_type>** –
			 ○ HDMI
			▪ **<port_index>** – The port number as printed on the front or rear panel
			▪ For output – 1
			▪ For input – 1 and 2
			▪ **<signal_type>** –
			 ○ VIDEO
			▪ **<index>** – Indicates a specific channel number when there are multiple channels of the same type
				Route HDMI IN 1 to HDMI OUT 1:
				`#X-ROUTE␣OUT.HDMI.1.VIDEO.1,IN.HDMI.1.VIDEO.1<CR>`

X-ROUTE?	Get routing status.
	
	ⓘ It is recommended to use the command `#SIGNALS-LIST` to get the list of all signal IDs available in the system and which can be used in this command.
	
	VIDEO.1 are the default **<signal_type>** and **<index>** in this command and are implied even if not written.
	
	This is an Extended Protocol 3000 command.
		COMMAND
		`#X-ROUTE?␣<direction_type1>.<port_type1>.<port_index1>.<signal_type1>.<index1><CR>`
		FEEDBACK
		`~nn@X-ROUTE␣<direction_type1>.<port_type1>.<port_index1>.<signal_type1>.<index1>,<direction_type2>.<port_type2>.<port_index2>.<signal_type2>.<index2><CR><LF>`
			The following attributes comprise the output signal ID (suffix 1) and input signal ID (suffix 2):
			▪ **<direction_type>** –
			 ○ OUT
			 ○ IN
			▪ **<port_type>** –
			 ○ HDMI
			▪ **<port_index>** – The port number as printed on the front or rear panel
			▪ For output – 1
			▪ For input – 1 and 2
			▪ **<signal_type>** –
			 ○ VIDEO
			▪ **<index>** – Indicates a specific channel number when there are multiple channels of the same type
				Get the routing status:
				`#X-ROUTE?␣OUT.HDMI.1.VIDEO.1<CR>`

Result and Error Codes

Syntax

In case of an error, the device responds with an error message. The error message syntax:

- **~NN@ERR XXX<CR><LF>** – when general error, no specific command

- **~NN@CMD ERR XXX<CR><LF>** – for specific command

- **NN** – machine number of device, default = 01

- **XXX** – error code

Error Codes

Error Name	Error Code	Description
P3K_NO_ERROR	0	No error
ERR_PROTOCOL_SYNTAX	1	Protocol syntax
ERR_COMMAND_NOT_AVAILABLE	2	Command not available
ERR_PARAMETER_OUT_OF_RANGE	3	Parameter out of range
ERR_UNAUTHORIZED_ACCESS	4	Unauthorized access
ERR_INTERNAL_FW_ERROR	5	Internal FW error
ERR_BUSY	6	Protocol busy
ERR_WRONG_CRC	7	Wrong CRC
ERR_TIMEDOUT	8	Timeout
ERR_RESERVED	9	(Reserved)
ERR_FW_NOT_ENOUGH_SPACE	10	Not enough space for data (firmware, FPGA…)
ERR_FS_NOT_ENOUGH_SPACE	11	Not enough space – file system
ERR_FS_FILE_NOT_EXISTS	12	File does not exist
ERR_FS_FILE_CANT_CREATED	13	File can't be created
ERR_FS_FILE_CANT_OPEN	14	File can't open
ERR_FEATURE_NOT_SUPPORTED	15	Feature is not supported
ERR_RESERVED_2	16	(Reserved)
ERR_RESERVED_3	17	(Reserved)
ERR_RESERVED_4	18	(Reserved)
ERR_RESERVED_5	19	(Reserved)
ERR_RESERVED_6	20	(Reserved)
ERR_PACKET_CRC	21	Packet CRC error
ERR_PACKET_MISSED	22	Packet number isn't expected (missing packet)
ERR_PACKET_SIZE	23	Packet size is wrong
ERR_RESERVED_7	24	(Reserved)
ERR_RESERVED_8	25	(Reserved)
ERR_RESERVED_9	26	(Reserved)
ERR_RESERVED_10	27	(Reserved)
ERR_RESERVED_11	28	(Reserved)
ERR_RESERVED_12	29	(Reserved)
ERR_EDID_CORRUPTED	30	EDID corrupted
ERR_NON_LISTED	31	Device specific errors
ERR_SAME_CRC	32	File has the same CRC – not changed
ERR_WRONG_MODE	33	Wrong operation mode
ERR_NOT_CONFIGURED	34	Device/chip was not initialized

The warranty obligations of Kramer Electronics Inc. ("Kramer Electronics") for this product are limited to the terms set forth below:

What is Covered

This limited warranty covers defects in materials and workmanship in this product.

What is Not Covered

This limited warranty does not cover any damage, deterioration or malfunction resulting from any alteration, modification, improper or unreasonable use or maintenance, misuse, abuse, accident, neglect, exposure to excess moisture, fire, improper packing and shipping (such claims must be presented to the carrier), lightning, power surges, or other acts of nature. This limited warranty does not cover any damage, deterioration or malfunction resulting from the installation or removal of this product from any installation, any unauthorized tampering with this product, any repairs attempted by anyone unauthorized by Kramer Electronics to make such repairs, or any other cause which does not relate directly to a defect in materials and/or workmanship of this product. This limited warranty does not cover cartons, equipment enclosures, cables or accessories used in conjunction with this product.

Without limiting any other exclusion herein, Kramer Electronics does not warrant that the product covered hereby, including, without limitation, the technology and/or integrated circuit(s) included in the product, will not become obsolete or that such items are or will remain compatible with any other product or technology with which the product may be used.

How Long this Coverage Lasts

The standard limited warranty for Kramer products is seven (7) years from the date of original purchase, with the following exceptions:

1. All Kramer VIA hardware products are covered by a standard three (3) year warranty for the VIA hardware and a standard three (3) year warranty for firmware and software updates; all Kramer VIA accessories, adapters, tags, and dongles are covered by a standard one (1) year warranty.
2. Kramer fiber optic cables, adapter-size fiber optic extenders, pluggable optical modules, active cables, cable retractors, ring mounted adapters, portable power chargers, Kramer speakers, and Kramer touch panels are covered by a standard one (1) year warranty. Kramer 7-inch touch panels purchased on or after April 1st, 2020 are covered by a standard two (2) year warranty.
3. All Kramer Calibre products, all Kramer Minicom digital signage products, all HighSecLabs products, all streaming, and all wireless products are covered by a standard three (3) year warranty.
4. All Sierra Video MultiViewers are covered by a standard five (5) year warranty.
5. Sierra switchers & control panels are covered by a standard seven (7) year warranty (excluding power supplies and fans that are covered for three (3) years).
6. K-Touch software is covered by a standard one (1) year warranty for software updates.
7. All Kramer passive cables are covered by a lifetime warranty.

Who is Covered

Only the original purchaser of this product is covered under this limited warranty. This limited warranty is not transferable to subsequent purchasers or owners of this product.

What Kramer Electronics Will Do

Kramer Electronics will, at its sole option, provide one of the following three remedies to whatever extent it shall deem necessary to satisfy a proper claim under this limited warranty:

1. Elect to repair or facilitate the repair of any defective parts within a reasonable period of time, free of any charge for the necessary parts and labor to complete the repair and restore this product to its proper operating condition. Kramer Electronics will also pay the shipping costs necessary to return this product once the repair is complete.
2. Replace this product with a direct replacement or with a similar product deemed by Kramer Electronics to perform substantially the same function as the original product. If a direct or similar replacement product is supplied, the original product's end warranty date remains unchanged and is transferred to the replacement product.
3. Issue a refund of the original purchase price less depreciation to be determined based on the age of the product at the time remedy is sought under this limited warranty.

What Kramer Electronics Will Not Do Under This Limited Warranty

If this product is returned to Kramer Electronics or the authorized dealer from which it was purchased or any other party authorized to repair Kramer Electronics products, this product must be insured during shipment, with the insurance and shipping charges prepaid by you. If this product is returned uninsured, you assume all risks of loss or damage during shipment. Kramer Electronics will not be responsible for any costs related to the removal or re-installation of this product from or into any installation. Kramer Electronics will not be responsible for any costs related to any setting up this product, any adjustment of user controls or any programming required for a specific installation of this product.

How to Obtain a Remedy Under This Limited Warranty

To obtain a remedy under this limited warranty, you must contact either the authorized Kramer Electronics reseller from whom you purchased this product or the Kramer Electronics office nearest you. For a list of authorized Kramer Electronics resellers and/or Kramer Electronics authorized service providers, visit our web site at www.kramerav.com or contact the Kramer Electronics office nearest you.

In order to pursue any remedy under this limited warranty, you must possess an original, dated receipt as proof of purchase from an authorized Kramer Electronics reseller. If this product is returned under this limited warranty, a return authorization number, obtained from Kramer Electronics, will be required (RMA number). You may also be directed to an authorized reseller or a person authorized by Kramer Electronics to repair the product.

If it is decided that this product should be returned directly to Kramer Electronics, this product should be properly packed, preferably in the original carton, for shipping. Cartons not bearing a return authorization number will be refused.

Limitation of Liability

THE MAXIMUM LIABILITY OF KRAMER ELECTRONICS UNDER THIS LIMITED WARRANTY SHALL NOT EXCEED THE ACTUAL PURCHASE PRICE PAID FOR THE PRODUCT. TO THE MAXIMUM EXTENT PERMITTED BY LAW, KRAMER ELECTRONICS IS NOT RESPONSIBLE FOR DIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY BREACH OF WARRANTY OR CONDITION, OR UNDER ANY OTHER LEGAL THEORY. Some countries, districts or states do not allow the exclusion or limitation of relief, special, incidental, consequential or indirect damages, or the limitation of liability to specified amounts, so the above limitations or exclusions may not apply to you.

Exclusive Remedy

TO THE MAXIMUM EXTENT PERMITTED BY LAW, THIS LIMITED WARRANTY AND THE REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, REMEDIES AND CONDITIONS, WHETHER ORAL OR WRITTEN, EXPRESS OR IMPLIED. TO THE MAXIMUM EXTENT PERMITTED BY LAW, KRAMER ELECTRONICS SPECIFICALLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IF KRAMER ELECTRONICS CANNOT LAWFULLY DISCLAIM OR EXCLUDE IMPLIED WARRANTIES UNDER APPLICABLE LAW, THEN ALL IMPLIED WARRANTIES COVERING THIS PRODUCT, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, SHALL APPLY TO THIS PRODUCT AS PROVIDED UNDER APPLICABLE LAW.

IF ANY PRODUCT TO WHICH THIS LIMITED WARRANTY APPLIES IS A "CONSUMER PRODUCT" UNDER THE MAGNUSON-MOSS WARRANTY ACT (15 U.S.C.A. §2301, ET SEQ.) OR OTHER APPLICABLE LAW, THE FOREGOING DISCLAIMER OF IMPLIED WARRANTIES SHALL NOT APPLY TO YOU, AND ALL IMPLIED WARRANTIES ON THIS PRODUCT, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR THE PARTICULAR PURPOSE, SHALL APPLY AS PROVIDED UNDER APPLICABLE LAW.

Other Conditions

This limited warranty gives you specific legal rights, and you may have other rights which vary from country to country or state to state.

This limited warranty is void if (i) the label bearing the serial number of this product has been removed or defaced, (ii) the product is not distributed by Kramer Electronics or (iii) this product is not purchased from an authorized Kramer Electronics reseller. If you are unsure whether a reseller is an authorized Kramer Electronics reseller, visit our web site at www.kramerav.com or contact a Kramer Electronics office from the list at the end of this document.

Your rights under this limited warranty are not diminished if you do not complete and return the product registration form or complete and submit the online product registration form. Kramer Electronics thanks you for purchasing a Kramer Electronics product. We hope it will give you years of satisfaction.

KRAMER

⚠ **SAFETY WARNING**
Disconnect the unit from the power supply before opening and servicing

For the latest information on our products and a list of Kramer distributors, visit our website where updates to this user manual may be found.

We welcome your questions, comments, and feedback.

www.KramerAV.com
info@KramerAV.com