

# 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:**

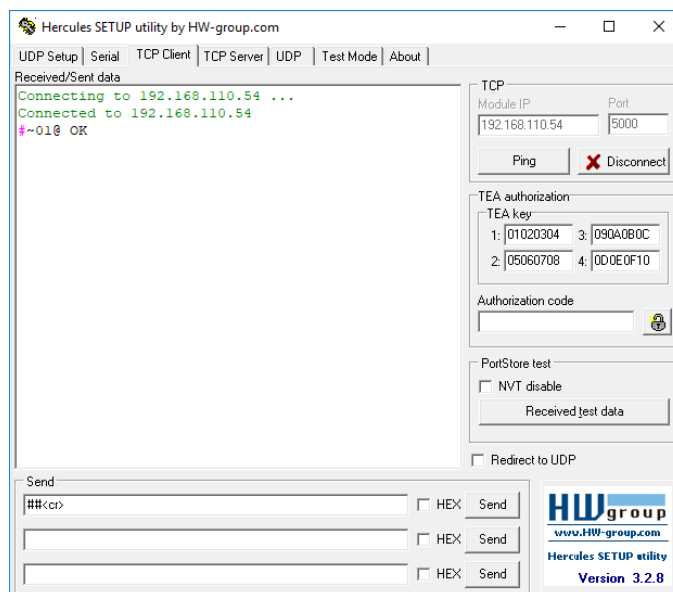
Prefix	Command Name	Constant (Space)	Parameter(s)	Suffix
#	Command	_	Parameter	<CR>

- **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 the **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
#	<p>Protocol handshaking.</p> <p>① Validates the Protocol 3000 connection and gets the machine number.</p> <p>Step-in master products use this command to identify the availability of a device.</p>	<b>COMMAND</b> <b>#&lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@_OK<CR><LF>		#<CR>
AV-SW-TIMEOUT	Set auto switching timeout.	<b>COMMAND</b> <b>#AV-SW-TIMEOUT_&lt;u&gt;action,time_out&lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@AV-SW-TIMEOUT_<u>action,time_out<CR><LF>	<b>action</b> – 4 – Disable 5V on video output if no input signal detected. <b>time_out</b> – 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: <b>#AV-SW-TIMEOUT_4,5&lt;CR&gt;</b>
AV-SW-TIMEOUT?	Get auto switching timeout.	<b>COMMAND</b> <b>#AV-SW-TIMEOUT?_&lt;u&gt;action&lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@AV-SW-TIMEOUT_<u>action,time_out<CR><LF>	<b>action</b> – 4 – Disable 5V on video output if no input signal detected <b>time_out</b> – Timeout in seconds 30 - 60000	Get the Disable 5V on video output if no input signal detected timeout: <b>#AV-SW-TIMEOUT?_4&lt;CR&gt;</b>
BEACON-INFO?	<p>Get beacon information, including IP address, UDP control port, TCP control port, MAC address, model, name.</p> <p>① There is no Set command. Get command initiates a notification.</p>	<b>COMMAND</b> <b>#BEACON-INFO?_&lt;u&gt;port_id&lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@BEACON-INFO_<u>port_id,ip_string,udp_port,tcp_port,mac_address,model,name<CR><LF>	<b>port_id</b> – ID of the Ethernet port <b>ip_string</b> – Dot-separated representation of the IP address <b>udp_port</b> – UDP control port <b>tcp_port</b> – TCP control port <b>mac_address</b> – Dash-separated mac address <b>model</b> – Device model <b>name</b> – Device name	Get beacon information: <b>#BEACON-INFO?_&lt;u&gt;&lt;CR&gt;</b>
BUILD-DATE?	Get device build date.	<b>COMMAND</b> <b>#BUILD-DATE?_&lt;u&gt;&lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@BUILD-DATE_<u>date,time<CR><LF>	<b>date</b> – Format: YYYY/MM/DD where YYYY = Year MM = Month DD = Day <b>time</b> – Format: hh:mm:ss where hh = hours mm = minutes ss = seconds	Get the device build date: <b>#BUILD-DATE?&lt;u&gt;&lt;CR&gt;</b>
CONF-EXPORT	For factory use only.			
CONF-IMPORT	For factory use only.			
CPEDID	<p>Copy EDID data from the output to the input EEPROM.</p> <p>① Destination bitmap size depends on device properties (for 64 inputs it is a 64-bit word).</p> <p>Example: bitmap 0x0013 means inputs 1,2 and 5 are loaded with the new EDID.</p> <p>In certain products Safe_mode is an optional parameter. See the HELP command for its availability.</p>	<b>COMMAND</b> <b>#CPEDID_&lt;u&gt;src_type,src_id,dst_type,dest_bitmap&lt;CR&gt;</b> or <b>#CPEDID_&lt;u&gt;src_type,src_id,dst_type,dest_bitmap,safe_mode&lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@CPEDID_<u>src_stg,src_id,dst_type,dest_bitmap<CR><LF> ~nn@CPEDID_<u>src_stg,src_id,st_type,dest_bitmap,safe_mode<CR><LF>	<b>src_type</b> – EDID source type (usually output) 0 – Input 1 – Output 2 – Default EDID 3 – Custom EDID <b>src_id</b> – Number of chosen source stage 0 – Default EDID source 1 – Output 1 <b>dst_type</b> – EDID destination type (usually input) 0 – Input 1 – Output 2 – Default EDID 3 – Custom EDID <b>dest_bitmap</b> – 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. <b>safe_mode</b> – 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: <b>#CPEDID_1,1,0,0x1&lt;CR&gt;</b> Copy the EDID data from the default EDID source to the Input: <b>#CPEDID_2,0,0,0x1&lt;CR&gt;</b>
DISPLAY?	Get output HPD status.	<b>COMMAND</b> <b>#DISPLAY?_&lt;u&gt;input_id &lt;CR&gt;</b> <b>FEEDBACK</b> ~nn@DISPLAY_<u>out_id,status<CR><LF>	<b>out_id</b> – Output number 1 – HDMI output <b>status</b> – 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: <b>#DISPLAY?_1&lt;CR&gt;</b>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-AEC	Set DSP field value.	<b>COMMAND</b> <b>#DSP-</b> <b>AEC_</b> <i>field_id</i> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR> <b>FEEDBACK</b> ~nn@DSP-AEC_ <i>field_id</i> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	<b>Field id</b> <ul style="list-style-type: none"> <li>o low-freq</li> <li>o high-freq</li> <li>o bypass</li> </ul> <b>&lt;direction_type&gt;</b> – IN <ul style="list-style-type: none"> <li>▪ <b>&lt;port_type&gt;</b> – <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> </ul> </li> <li>▪ <b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel 2 to 3</li> <li>▪ <b>&lt;signal_type&gt;</b> – <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> </li> <li>▪ <b>&lt;index&gt;</b> – 1</li> </ul> <b>value</b> – For low-freq and high-freq +20Hz to +20kHz (default-150) For bypass 0– Off 1– On	Set bypass on input 3 to off: <b>#DSP-AEC_bypass,IN.ANALOG_AUDIO.3.AUDIO.1,0&lt;CR&gt;</b>
DSP-AEC?	Get DSP field value.	<b>COMMAND</b> <b>#DSP-</b> <b>AEC?</b> <i>field_id</i> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~nn@DSP-AEC_ <i>field_id</i> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	<b>Field id</b> <ul style="list-style-type: none"> <li>o low-freq</li> <li>o high-freq</li> <li>o bypass</li> </ul> <b>&lt;direction_type&gt;</b> – IN <ul style="list-style-type: none"> <li>▪ <b>&lt;port_type&gt;</b> – <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> </ul> </li> <li>▪ <b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel 2 to 3</li> <li>▪ <b>&lt;signal_type&gt;</b> – <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> </li> <li>▪ <b>&lt;index&gt;</b> – 1</li> </ul> <b>value</b> – For low-freq and high-freq +20Hz to +20kHz (default-150) For bypass 0– Off 1– On	get low frequency on input 3: <b>#DSP-AEC?_low-freq,IN.ANALOG_AUDIO.3.AUDIO.1&lt;CR&gt;</b>
DSP-COMP	Set DSP compressor values.	<b>COMMAND</b> <b>#DSP-</b> <b>COMP_</b> <i>field_id</i> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR> <b>FEEDBACK</b> ~nn@DSP-COMP_ <i>field_id</i> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	<b>Field id</b> <ul style="list-style-type: none"> <li>o Attack</li> <li>o Release</li> <li>o Threshold</li> <li>o Ratio</li> <li>o Gain</li> <li>o Bypass</li> </ul> <b>&lt;direction_type&gt;</b> – IN <ul style="list-style-type: none"> <li>▪ <b>&lt;port_type&gt;</b> – <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> <li>o ANALOG_STEREO</li> <li>o USB_B</li> </ul> </li> <li>▪ <b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel For ANALOG_AUDIO – 2 to 5 For USB_B, ANALOG_STEREO – 1</li> <li>▪ <b>&lt;signal_type&gt;</b> – <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> </li> <li>▪ <b>&lt;index&gt;</b> –  For ANALOG_AUDIO – 1  For USB_B, ANALOG_STEREO – 1, 2</li> </ul> <b>value</b> – 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: <b>#DSP-COMP_attack,IN.ANALOG_AUDIO.3.AUDIO.1,15&lt;CR&gt;</b>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-COMP?	Get DSP compressor values.	<p><b>COMMAND</b> #DSP- COMP?_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt; .&lt;signal_type&gt;.&lt;index&gt;&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@DSP-COMP_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</p>	<p><b>Field id</b></p> <ul style="list-style-type: none"> <li>o Attack</li> <li>o Release</li> <li>o Threshold</li> <li>o Ratio</li> <li>o Gain</li> <li>o Bypass</li> </ul> <p>&lt;direction_type&gt; – IN</p> <ul style="list-style-type: none"> <li>▪&lt;port_type&gt; – <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> <li>o ANALOG_STEREO</li> <li>o USB_B</li> </ul> </li> </ul> <p>▪&lt;port_index&gt; – The port number as printed on the front or rear panel For ANALOG_AUDIO – 2 to 5 For USB_B, ANALOG_STEREO – 1</p> <ul style="list-style-type: none"> <li>▪&lt;signal_type&gt; – <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> </li> <li>▪&lt;index&gt; – <ul style="list-style-type: none"> <li>For ANALOG_AUDIO – 1</li> <li>For USB_B, ANALOG_STEREO – 1, 2</li> </ul> </li> </ul> <p>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</p>	Get attack time on input 3: #DSP-COMP?_attack time,IN.ANALOG_AUDIO.3. AUDIO.1<CR>
DSP-DELAY	Set DSP delay.	<p><b>COMMAND</b> #DSP- DELAY_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt; .&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@DSP-DELAY_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</p>	<p><b>Field id</b></p> <ul style="list-style-type: none"> <li>o Delay</li> <li>o Bypass</li> </ul> <p>&lt;direction_type&gt; –</p> <ul style="list-style-type: none"> <li>o IN</li> <li>o OUT</li> </ul> <ul style="list-style-type: none"> <li>▪&lt;port_type&gt; – <ul style="list-style-type: none"> <li>For IN <ul style="list-style-type: none"> <li>o HDMI_AUDIO</li> </ul> </li> <li>For OUT <ul style="list-style-type: none"> <li>o USB_B</li> </ul> </li> </ul> </li> <li>▪&lt;port_index&gt; – The port number as printed on the front or rear panel 1</li> <li>▪&lt;signal_type&gt; – <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> </li> <li>▪&lt;index&gt; – 1 to 2</li> </ul> <p>value – For delay time [ms] 0 to 150 For bypass [ms] 0– off 1– on</p>	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.	<p><b>COMMAND</b> #DSP- DELAY?_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt; .&lt;signal_type&gt;.&lt;index&gt;&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@DSP-DELAY_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</p>	<p><b>Field id</b></p> <ul style="list-style-type: none"> <li>o Delay</li> <li>o Bypass</li> </ul> <p>&lt;direction_type&gt; –</p> <ul style="list-style-type: none"> <li>o IN</li> <li>o OUT</li> </ul> <ul style="list-style-type: none"> <li>▪&lt;port_type&gt; – <ul style="list-style-type: none"> <li>For IN <ul style="list-style-type: none"> <li>o HDMI_AUDIO</li> </ul> </li> <li>For OUT <ul style="list-style-type: none"> <li>o USB_B</li> </ul> </li> </ul> </li> <li>▪&lt;port_index&gt; – The port number as printed on the front or rear panel 1</li> <li>▪&lt;signal_type&gt; – <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> </li> <li>▪&lt;index&gt; – 1 to 2</li> </ul> <p>value – For delay time [ms] 0 to 150 For bypass [ms] 0– off 1– on</p>	Get bypass status on HDMI input 1 to 75: #DSP-DELAY?_bypass,IN.HD MI_AUDIO.1.AUDIO.1<CR>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-EQ	Set DSP equalizer.	<p><b>COMMAND</b></p> <pre>#DSP- EQ_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,band,value&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@DSP-EQ_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,band,value&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>Field_id</b></p> <ul style="list-style-type: none"> <li>o level</li> <li>o freq</li> <li>o qfactor</li> <li>o band-bypass</li> </ul> <p><b>&lt;direction_type&gt; -</b></p> <ul style="list-style-type: none"> <li>o IN</li> <li>o OUT</li> </ul> <p><b>&lt;port_type&gt; -</b></p> <p>For IN</p> <ul style="list-style-type: none"> <li>o ANALOG_STEREO</li> <li>o ANALOG_AUDIO</li> <li>o USB_B</li> </ul> <p>For OUT</p> <ul style="list-style-type: none"> <li>o ANALOG_STEREO</li> <li>o USB_B</li> </ul> <p><b>&lt;port_index&gt; -</b> The port number as printed on the front or rear panel 1 to 5</p> <p><b>&lt;signal_type&gt; -</b></p> <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> <p><b>&lt;index&gt; -</b> 1 to 2</p> <p><b>band -</b></p> <p>For IN</p> <ul style="list-style-type: none"> <li>1 to 4</li> </ul> <p>For OUT</p> <ul style="list-style-type: none"> <li>1 to 8</li> </ul> <p>For bypass</p> <ul style="list-style-type: none"> <li>0- global bypass</li> <li>1,2,3,4 - for each band</li> </ul> <p><b>value -</b></p> <p>For level [dB]</p> <ul style="list-style-type: none"> <li>-24 to +24</li> </ul> <p>For freq [Hz]</p> <ul style="list-style-type: none"> <li>20 to 20K</li> </ul> <p>For qfactor [Oct]</p> <ul style="list-style-type: none"> <li>0.05 to 4</li> </ul> <p>For bypass [ms]</p> <ul style="list-style-type: none"> <li>0- off</li> <li>1- on</li> </ul>	<p>Set EQ level on input 2 to 12:</p> <pre>#DSP-EQ?_level,IN.ANALOG_AUDIO.2.AUDIO.1,3,12&lt;CR&gt;</pre>
DSP-EQ?	Set DSP equalizer.	<p><b>COMMAND</b></p> <pre>#DSP- EQ?_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,band&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@DSP-EQ_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,band,value&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>Field_id</b></p> <ul style="list-style-type: none"> <li>o Level</li> <li>o Freq</li> <li>o Qfactor</li> <li>o Band-bypass</li> </ul> <p><b>&lt;direction_type&gt; -</b></p> <ul style="list-style-type: none"> <li>o IN</li> <li>o OUT</li> </ul> <p><b>&lt;port_type&gt; -</b></p> <p>For IN</p> <ul style="list-style-type: none"> <li>o ANALOG_STEREO</li> <li>o ANALOG_AUDIO</li> <li>o USB_B</li> </ul> <p>For OUT</p> <ul style="list-style-type: none"> <li>o ANALOG_STEREO</li> <li>o USB_B</li> </ul> <p><b>&lt;port_index&gt; -</b> The port number as printed on the front or rear panel 1 to 5</p> <p><b>&lt;signal_type&gt; -</b></p> <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> <p><b>&lt;index&gt; -</b> 1 to 2</p> <p><b>band -</b></p> <p>For IN</p> <ul style="list-style-type: none"> <li>1 to 4</li> </ul> <p>For OUT</p> <ul style="list-style-type: none"> <li>1 to 8</li> </ul> <p>For band-bypass</p> <ul style="list-style-type: none"> <li>0- global bypass</li> <li>1,2,3,4 - for each band</li> </ul> <p><b>value -</b></p> <p>For level [dB]</p> <ul style="list-style-type: none"> <li>-24 to +24</li> </ul> <p>For freq [Hz]</p> <ul style="list-style-type: none"> <li>20 to 20K</li> </ul> <p>For qfactor [Oct]</p> <ul style="list-style-type: none"> <li>0.05 to 4</li> </ul> <p>For bypass [ms]</p> <ul style="list-style-type: none"> <li>0- off</li> <li>1- on</li> </ul>	<p>Get global bypass state for EQ on input 2:</p> <pre>#DSP-EQ?_bypass,IN.ANALOG_AUDIO.2.AUDIO.1,0&lt;CR&gt;</pre>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-EXP	Set DSP expander.	<p><b>COMMAND</b></p> <pre>#DSP- EXP_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt; signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@DSP-EXP_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_i ndex&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>Field id</b></p> <ul style="list-style-type: none"> <li>o Attack</li> <li>o Release</li> <li>o Threshold</li> <li>o Ratio</li> <li>o Bypass</li> </ul> <p><b>&lt;direction_type&gt;</b> – IN</p> <p><b>&lt;port_type&gt;</b> –</p> <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> <li>o ANALOG_STEREO</li> <li>o USB_B</li> </ul> <p><b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel For ANALOG_AUDIO – 2 to 5 For USB_B, ANALOG_STEREO – 1</p> <p><b>&lt;signal_type&gt;</b> –</p> <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> <p><b>&lt;index&gt;</b> – 1</p> <p><b>value</b> –</p> <p>For attack [ms] 0 to 100</p> <p>For release time [ms] 0 to 10K</p> <p>For threshold [dB] -100 to 0</p> <p>For ratio [1 to 100]:1</p> <p>For gain compensation [dB] -100 to +15</p> <p>For bypass [ms] 0– off 1– on</p>	<p>Set attack time on input 2 to 50:</p> <pre>#DSP-EXP_attack,IN.ANALOG_AUDIO.2.AUDIO.1,50&lt;CR&gt;</pre>
DSP-EXP?	Get DSP expander.	<p><b>COMMAND</b></p> <pre>#DSP- EXP?_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt; signal_type&gt;.&lt;index&gt;&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@DSP-EXP_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_i ndex&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>Field id</b></p> <ul style="list-style-type: none"> <li>o Attack</li> <li>o Release</li> <li>o Threshold</li> <li>o Ratio</li> <li>o Bypass</li> </ul> <p><b>&lt;direction_type&gt;</b> – IN</p> <p><b>&lt;port_type&gt;</b> –</p> <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> <li>o ANALOG_STEREO</li> <li>o USB_B</li> </ul> <p><b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel For ANALOG_AUDIO – 2 to 5 For USB_B, ANALOG_STEREO – 1</p> <p><b>&lt;signal_type&gt;</b> –</p> <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> <p><b>&lt;index&gt;</b> – 1</p> <p><b>value</b> –</p> <p>For attack [ms] 0 to 100</p> <p>For release [ms] 0 to 10K</p> <p>For threshold [dB] -100 to 0</p> <p>For ratio 1 to 100:1</p> <p>For bypass 0– off 1– on</p>	<p>Get attack time on input 2:</p> <pre>#DSP-EXP?_attack,IN.ANALOG_AUDIO.2.AUDIO.1&lt;CR&gt;</pre>
DSP-HPF	Set DSP HPF.	<p><b>COMMAND</b></p> <pre>#DSP- HPF_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt; signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@DSP-HPF_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_i ndex&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>Field id</b></p> <ul style="list-style-type: none"> <li>o freq</li> <li>o bypass</li> </ul> <p><b>&lt;direction_type&gt;</b> – IN</p> <p><b>&lt;port_type&gt;</b> –</p> <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> </ul> <p><b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel 2 to 5</p> <p><b>&lt;signal_type&gt;</b> –</p> <ul style="list-style-type: none"> <li>o AUDIO</li> </ul> <p><b>&lt;index&gt;</b> – 1</p> <p><b>value</b> –</p> <p>For freq [Hz] 20 to 20K</p> <p>For bypass 0– off 1– on</p>	<p>Set bypass status on input 2 to off:</p> <pre>#DSP-HPF_bypass,IN.ANALOG_AUDIO.2.AUDIO.1,0&lt;CR&gt;</pre>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-HPF?	Get DSP HPF.	<pre>COMMAND #DSP- HPF?_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt; signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;  FEEDBACK ~nn@DSP-HPF_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_i ndex&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<pre>Field id o freq o bypass &lt;direction_type&gt; – IN ▪&lt;port_type&gt; – o ANALOG_AUDIO ▪&lt;port_index&gt; – The port number as printed on the front or rear panel 2 to 5 ▪&lt;signal_type&gt; – o AUDIO ▪&lt;index&gt; – 1 value – For freq [Hz] 20 to 20K For bypass 0– off 1– on</pre>	Get bypass status on input 2:  #DSP-HPF?_bypass,IN.ANALOG_AUDIO.2.AUDIO.1<CR>
DSP-INVERT	Set DSP phase inversion state.	<pre>COMMAND #DSP- INVERT_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal _type&gt;.&lt;index&gt;,value&lt;CR&gt;  FEEDBACK ~nn@DSP-INVERT_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt; signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<pre>&lt;direction_type&gt; – IN ▪&lt;port_type&gt; – o ANALOG_AUDIO ▪&lt;port_index&gt; – The port number as printed on the front or rear panel 2 to 5 ▪&lt;signal_type&gt; – o AUDIO ▪&lt;index&gt; – 1 value – 0– off 1– on</pre>	Set phase inversion state input 2 to off:  #DSP-INVERT_IN.ANALOG_AUDIO.2.AUDIO.1,0<CR>
DSP-INVERT?	Get DSP phase inversion state.	<pre>COMMAND #DSP- INVERT?_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signa l_type&gt;.&lt;index&gt;&lt;CR&gt;  FEEDBACK ~nn@DSP-INVERT_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt; signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<pre>&lt;direction_type&gt; – IN ▪&lt;port_type&gt; – o ANALOG_AUDIO ▪&lt;port_index&gt; – The port number as printed on the front or rear panel 2 to 5 ▪&lt;signal_type&gt; – o AUDIO ▪&lt;index&gt; – 1 value – 0– off 1– on</pre>	Get phase inversion state on input 2:  #DSP-INVERT_IN.ANALOG_AUDIO.2.AUDIO.1,0<CR>
DSP-LIMITER	Set DSP limiter.	<pre>COMMAND #DSP- LIMITER_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index &gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;  FEEDBACK ~nn@DSP-LIMITER_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;po rt_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<pre>Field id o Threshold o Bypass &lt;direction_type&gt; – OUT ▪&lt;port_type&gt; – o ANALOG_STEREO o USB_B ▪&lt;port_index&gt; – The port number as printed on the front or rear panel 1 ▪&lt;signal_type&gt; – o AUDIO ▪&lt;index&gt; – 1 to 2 value – For threshold [dB] -100 to 0 For bypass 0– Off 1– On</pre>	Set bypass status on output to ON:  #DSP-LIMITER_bypass,OUT.ANALOG_STEREO.1.AUDIO.1,1<CR>
DSP-LIMITER?	Get DSP limiter.	<pre>COMMAND #DSP- LIMITER?_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_inde x&gt;.&lt;signal_type&gt;.&lt;index&gt;&lt;CR&gt;  FEEDBACK ~nn@DSP-LIMITER_field_id,&lt;direction_type&gt;.&lt;port_type&gt;.&lt;po rt_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,value&lt;CR&gt;&lt;LF&gt;</pre>	<pre>Field id o Threshold o Bypass &lt;direction_type&gt; – OUT ▪&lt;port_type&gt; – o ANALOG_STEREO o USB_B ▪&lt;port_index&gt; – The port number as printed on the front or rear panel 1 ▪&lt;signal_type&gt; – o AUDIO ▪&lt;index&gt; – 1 to 2 value – For threshold [dB] -100 to 0 For bypass 0– Off 1– On</pre>	Get bypass status on output:  #DSP-LIMITER?_bypass,OUT.ANALOG_STEREO.2.AUDIO.1,1<CR>

Function	Description	Syntax	Parameters/Attributes	Example
DSP-METER?	Read DSP meters.	<b>COMMAND</b> #DSP- <b>METER</b> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,type<CR>  <b>FEEDBACK</b> ~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>
DSP-POST	Set DSP post volume faders/mute.	<b>COMMAND</b> #DSP- <b>POST</b> _field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR>  <b>FEEDBACK</b> ~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.	<b>COMMAND</b> #DSP- <b>POST?</b> _field_id,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR>  <b>FEEDBACK</b> ~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>



Function	Description	Syntax	Parameters/Attributes	Example
DSP-SIG-GEN	Set DSP signal generator.	<b>COMMAND</b> <b>#DSP-SIG-</b> <b>GEN_</b> <u>field_id</u> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR>  <b>FEEDBACK</b> ~nn@DSP-SIG- <b>GEN?</b> <u>field_id</u> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	<b>Field id</b> 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 <b>value</b> – 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: <b>#DSP-SIG-GEN_</b> <u>mode</u> ,IN.GENERATOR.1.AUDIO.1,2<CR>
DSP-SIG-GEN?	Get DSP signal generator.	<b>COMMAND</b> <b>#DSP-SIG-</b> <b>GEN?</b> <u>field_id</u> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR>  <b>FEEDBACK</b> ~nn@DSP-SIG- <b>GEN?</b> <u>field_id</u> ,<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,value<CR><LF>	<b>Field id</b> 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 <b>value</b> – 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: <b>#DSP-SIG-GEN?</b> <u>mode</u> ,IN.GENERATOR.1.AUDIO.1<CR>
EDID-AUDIO	Set audio capabilities for EDID.	<b>COMMAND</b> <b>#EDID-AUDIO_</b> <u>input_id</u> ,audio_format<CR>  <b>FEEDBACK</b> ~nn@EDID-AUDIO_<u>input_id</u>,audio_format <CR><LF>	<b>input_id</b> – 1 – HDMI IN 1 2 – HDMI IN 2 <b>Audio_format</b> – Audio block added to EDID: 0 – Auto 1 – LPCM 2CH	Set HDMI IN 1 audio capabilities for EDID (LPCM 6CH): <b>#EDID-AUDIO_</b> <u>1</u> ,2<CR>
EDID-AUDIO?	Get audio capabilities for EDID.	<b>COMMAND</b> <b>#EDID-AUDIO?</b> <u>input_id</u> <CR>  <b>FEEDBACK</b> ~nn@EDID-AUDIO_<u>input_id</u>,audio_format<CR><LF>	<b>input_id</b> – 1 – HDMI IN 1 2 – HDMI IN 2 <b>Audio_format</b> – Audio block added to EDID: 0 – Auto 1 – LPCM 2CH	Get HDMI IN 1 audio capabilities for EDID: <b>#EDID-AUDIO?</b> <u>1</u> <CR>
EDID-CS	Set EDID color space. ⓘ Set command might change the current EDID.	<b>COMMAND</b> <b>#EDID-CS_</b> <u>input_id</u> ,ColSpace<CR>  <b>FEEDBACK</b> ~nn@EDID-CS_<u>input_id</u>,ColSpace<CR><LF>	<b>input_id</b> – 1 <b>ColSpace</b> – Color space 0 – RGB 4 – auto	Set HDMI IN 1 EDID color space to RGB (enabled): <b>#EDID-CS_</b> <u>1</u> ,0<CR>
EDID-CS?	Get EDID color space. ⓘ Get command might change the current EDID.	<b>COMMAND</b> <b>#EDID-CS?</b> <u>input_id</u> <CR>  <b>FEEDBACK</b> ~nn@EDID-CS_<u>input_id</u>,ColSpace<CR><LF>	<b>input_id</b> – 1 <b>ColSpace</b> – Color space 0 – RGB 4 – auto	Get EDID color space: <b>#EDID-CS?</b> <u>1</u> <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).	<b>COMMAND</b> <b>#ETH-PORT_</b> <u>portType</u> ,ETHPort<CR>  <b>FEEDBACK</b> ~nn@ETH-PORT_<u>portType</u>,ETHPort<CR><LF>	<b>portType</b> – TCP/UDP <b>ETHPort</b> – TCP/UDP port number (0 – 65535)	Set the Ethernet port protocol for TCP to port 12457: <b>#ETH-PORT_</b> <u>0</u> ,12457<CR>

Function	Description	Syntax	Parameters/Attributes	Example
ETH-PORT?	Get Ethernet port protocol.	<b>COMMAND</b> #ETH-PORT?_portType<CR> <b>FEEDBACK</b> ~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.  ① 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.	<b>COMMAND</b> #FACTORY<CR> <b>FEEDBACK</b> ~nn@FACTORY_OK<CR><LF>		Reset the device to factory default configuration: #FACTORY<CR>
FILE-HANDLED	For factory use only.			
FEATURE-LIST?	Get feature state according to the feature ID.	<b>COMMAND</b> #FEATURE-LIST?_feature_id<CR> <b>FEEDBACK</b> ~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.	<b>COMMAND</b> #FW-TYPE?_<CR> <b>FEEDBACK</b> ~nn@FEATURE-LIST_fw_type<CR><LF>	Fw_type – 0 – Application 1 – Safe mode (kboot)	Get the current FW type status: #FW-TYPE?_<CR>
GPIO-CFG	Set HW GPIO configuration.	<b>COMMAND</b> #GPIO-CFG_HwGpioNum,HwGpioType,HwGpioDir,Pullup<CR> <b>FEEDBACK</b> ~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.	<b>COMMAND</b> #GPIO-CFG_HwGpioNum <CR> <b>FEEDBACK</b> ~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	Set HW GPIO state.  ① 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.	<b>COMMAND</b> #GPIO-STATE_HwGpioNum,HwGpioState<CR> <b>FEEDBACK</b> ~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?	Get HW GPIO state.  ① 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).	<b>COMMAND</b> #GPIO-STATE?_HwGpioNum<CR> <b>FEEDBACK</b> ~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>

Function	Description	Syntax	Parameters/Attributes	Example
GPIO-STEP	<p>Set HW GPIO step.</p> <p>ⓘ In digital mode the response is 2. In analog mode the response is 1 to 100. In other modes an error is returned</p>	<p><b>COMMAND</b></p> <pre>#GPIO-STEP_#HwGpioNum,Step&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@GPIO-STEP_#HwGpioNum,NumOfStep,CurrentStep&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>HwGpioNum</b> – Hardware GPIO number (1 to 2)</p> <p><b>NumOfStep</b> – The configuration step – See note in description.</p> <p><b>CurrentStep</b> – The actual step depending on the measured voltage</p>	<p>Set GPIO 2 (set to Analog In) configuration step to 38mV:</p> <pre>#GPIO-STEP_2,38&lt;CR&gt;</pre>
GPIO-STEP?	<p>Get HW GPIO step.</p> <p>ⓘ In digital mode the response is 2. In analog mode the response is 1 to 100. In other modes an error is returned</p>	<p><b>COMMAND</b></p> <pre>#GPIO-STEP?_#HwGpioNum&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@GPIO-STATE_#HwGpioNum,NumOfStep,CurrentStep&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>HwGpioNum</b> – Hardware GPIO number (1 to 2)</p> <p><b>NumOfStep</b> – The configuration step – See note in description.</p> <p><b>CurrentStep</b> – The actual step depending on the measured voltage</p>	<p>Get GPIO 2 configuration:</p> <pre>#GPIO-STEP?_2&lt;CR&gt;</pre>
GPIO-THR	<p>Set HW GPIO voltage levels.</p>	<p><b>COMMAND</b></p> <pre>#GPIO-THR_#HwGpioNum,LowLevel,HighLevel&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@GPIO-THR_#HwGpioNum,LowLevel,HighLevel&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>HwGpioNum</b> – Hardware GPIO number (1 to 2)</p> <p><b>LowLevel</b> – Voltage 500 to 28000 millivolts</p> <p><b>HighLevel</b> – Voltage 2000 to 30000 millivolts</p>	<p>Set GPIO 2 to a low level of 800mV and a high level of 2200mV:</p> <pre>#GPIO-THR_2,800,2200&lt;CR&gt;</pre>
GPIO-THR?	<p>Get HW GPIO voltage levels that were set.</p>	<p><b>COMMAND</b></p> <pre>#GPIO-THR?_#HwGpioNum&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@GPIO-THR_#HwGpioNum,LowLevel,HighLevel&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>HwGpioNum</b> – Hardware GPIO number (1 to 2)</p> <p><b>LowLevel</b> – Voltage 500 to 28000 millivolts</p> <p><b>HighLevel</b> – Voltage 2000 to 30000 millivolts</p>	<p>Get GPIO 2 voltage levels:</p> <pre>#GPIO-THR?_2&lt;CR&gt;</pre>
GPIO-VOLT?	<p>Get active voltage levels of HW GPIO.</p> <p>ⓘ This command is not available in digital out mode.</p>	<p><b>COMMAND</b></p> <pre>#GPIO-VOLT?_#HwGpioNum&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@GPIO-VOLT_#HwGpioNum,Voltage &lt;CR&gt;&lt;LF&gt;</pre>	<p><b>HwGpioNum</b> – Hardware GPIO number (1 to 2)</p> <p><b>Voltage</b> – Voltage 0 to 30000 millivolts</p>	<p>Get GPIO 2 configuration:</p> <pre>#GPIO-STEP?_2&lt;CR&gt;</pre>
HDCP-MOD	<p>Set HDCP mode.</p> <p>ⓘ Set HDCP working mode on the device input:</p> <p>HDCP supported - HDCP_ON [default].</p> <p>HDCP not supported - HDCP OFF.</p> <p>HDCP support changes following detected sink - MIRROR OUTPUT.</p> <p>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.</p>	<p><b>COMMAND</b></p> <pre>#HDCP-MOD_#inp_id,mode&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@HDCP-MOD_#inp_id,mode&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>inp_id</b> – Input number: 1 – HDMI IN 1 2 – HDMI IN 2</p> <p><b>mode</b> – HDCP mode: 0 – HDCP Off 3 – HDCP defined according to the connected output</p>	<p>Set the input HDCP-MODE of HDMI IN 1 to Off:</p> <pre>#HDCP-MOD_1,0&lt;CR&gt;</pre>
GPIO-MOD?	<p>Get HDCP mode.</p> <p>ⓘ Set HDCP working mode on the device input:</p> <p>HDCP supported - HDCP_ON [default].</p> <p>HDCP not supported - HDCP OFF.</p> <p>HDCP support changes following detected sink - MIRROR OUTPUT.</p>	<p><b>COMMAND</b></p> <pre>#HDCP-MOD?_#inp_id&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@HDCP-MOD_#inp_id,mode&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>inp_id</b> – Input number: 1 – HDMI IN 1 2 – HDMI IN 2</p> <p><b>mode</b> – HDCP mode: 0 – HDCP Off 3 – HDCP defined according to the connected output</p>	<p>Get the input HDCP-MODE of HDMI IN 1:</p> <pre>#HDCP-MOD?_1&lt;CR&gt;</pre>
HELP	<p>Get command list or help for specific command.</p>	<p><b>COMMAND</b></p> <pre>#HELP&lt;CR&gt;</pre> <pre>#HELP_#cmd_name&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <p>1. Multi-line:</p> <pre>~nn@Device_#cmd_name,#cmd_name...&lt;CR&gt;&lt;LF&gt;</pre> <p>To get help for command use: HELP (COMMAND_NAME)&lt;CR&gt;&lt;LF&gt;</p> <pre>~nn@HELP_#cmd_name:&lt;CR&gt;&lt;LF&gt;</pre> <pre>description&lt;CR&gt;&lt;LF&gt;</pre> <pre>USAGE:usage&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>cmd_name</b> – Name of a specific command</p>	<p>Get the command list:</p> <pre>#HELP&lt;CR&gt;</pre> <p>Get help for AV-SW-TIMEOUT:</p> <pre>HELP_av-sw-timeout&lt;CR&gt;</pre>

Function	Description	Syntax	Parameters/Attributes	Example
HDCP-STAT?	<p>Get HDCP signal status.</p> <p>① Output stage (1) – get the HDCP signal status of the sink device connected to the specified output.</p> <p>Input stage (0) – get the HDCP signal status of the source device connected to the specified input.</p>	<p><b>COMMAND</b></p> <p>#HDCP-STAT?_stage,stage_id&lt;CR&gt;</p> <p><b>FEEDBACK</b></p> <p>~nn@HDCP-STAT_stage,stage_id,status&lt;CR&gt;&lt;LF&gt;</p>	<p>stage – Input/Output</p> <p>0 – Input</p> <p>1 – Output</p> <p>stage_id – Number of chosen stage for the input stage</p> <p>1 – HDMI IN 1</p> <p>2 – HDMI IN 2</p> <p>For the output stage</p> <p>1 – HDMI OUT</p> <p>status – Signal encryption status - valid values On/Off</p> <p>0 – HDCP Off</p> <p>1 – HDCP On</p>	<p>Get the output HDCP-STATUS of HDMI IN:</p> <p>#HDCP-STAT?_0,1&lt;CR&gt;</p>
HELP	<p>Get command list or help for specific command.</p>	<p><b>COMMAND</b></p> <p>#HELP&lt;CR&gt;</p> <p>#HELP_command_name&lt;CR&gt;</p> <p><b>FEEDBACK</b></p> <p>1. Multi-line:</p> <p>~nn@Device_command,_command...&lt;CR&gt;&lt;LF&gt;</p> <p>To get help for command use: HELP (COMMAND_NAME)&lt;CR&gt;&lt;LF&gt;</p> <p>~nn@HELP_command:&lt;CR&gt;&lt;LF&gt;</p> <p>description&lt;CR&gt;&lt;LF&gt;</p> <p>USAGE: usage&lt;CR&gt;&lt;LF&gt;</p>	<p>command – Name of a specific command</p>	<p>Get the command list:</p> <p>#HELP&lt;CR&gt;</p> <p>To get help for AV-SW-TIMEOUT:</p> <p>HELP_AV-SW-TIMEOUT&lt;CR&gt;</p>
LOGIN	<p>Set protocol permission.</p> <p>① 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.</p> <p>The permission system works only if security is enabled with the "SECUR" command.</p> <p>It is not mandatory to enable the permission system in order to use the device.</p> <p>In each device, some connections allow logging in to different levels. Some do not work with security at all.</p> <p>Connection may logout after timeout.</p>	<p><b>COMMAND</b></p> <p>#LOGIN_login_level,password&lt;CR&gt;</p> <p><b>FEEDBACK</b></p> <p>~nn@LOGIN_login_level,password_OK&lt;CR&gt;&lt;LF&gt;</p> <p>or</p> <p>~nn@LOGIN_ERR_004&lt;CR&gt;&lt;LF&gt;</p> <p>(if bad password entered)</p>	<p>login_level – Level of permissions required ( User or Admin)</p> <p>password – Predefined password (by PASS command). Default password is an empty string</p>	<p>Set the protocol permission level to Admin (when the password defined in the PASS command is 33333):</p> <p>#LOGIN_Admin,33333&lt;CR&gt;</p>
LOGIN?	<p>Get current protocol permission level.</p> <p>① For devices that support security, LOGIN allows the user to run commands with an End User or Administrator permission level.</p> <p>In each device, some connections allow logging in to different levels. Some do not work with security at all.</p> <p>Connection may logout after timeout.</p> <p>The permission system works only if security is enabled with the "SECUR" command.</p>	<p><b>COMMAND</b></p> <p>#LOGIN?_&lt;CR&gt;</p> <p><b>FEEDBACK</b></p> <p>~nn@LOGIN_login_level&lt;CR&gt;&lt;LF&gt;</p>	<p>login_level – Level of permissions required (User or Admin)</p>	<p>Get current protocol permission level:</p> <p>#LOGIN?&lt;CR&gt;</p>
LOGOUT	<p>Cancel current permission level.</p> <p>① Logs out from End User or Administrator permission levels to Not Secure.</p>	<p><b>COMMAND</b></p> <p>#LOGOUT&lt;CR&gt;</p> <p><b>FEEDBACK</b></p> <p>~nn@LOGOUT_OK&lt;CR&gt;&lt;LF&gt;</p>		<p>#LOGOUT&lt;CR&gt;</p>

Function	Description	Syntax	Parameters/Attributes	Example
MODEL?	Get device model.  ① 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.	<b>COMMAND</b> #MODEL?_<CR> <b>FEEDBACK</b> ~nn@MODEL_<model_name><CR><LF>	<b>model_name</b> – String of up to 19 printable ASCII chars	Get the device model: #MODEL?_<CR>
NAME	Set machine (DNS) name.  ① 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).	<b>COMMAND</b> #NAME_<machine_name><CR> <b>FEEDBACK</b> ~nn@NAME_<machine_name><CR><LF>	<b>machine_name</b> – 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>
NAME?	Get machine (DNS) name.  ① 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).	<b>COMMAND</b> #NAME?_<CR> <b>FEEDBACK</b> ~nn@NAME_<machine_name><CR><LF>	<b>machine_name</b> – 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.  ① Factory default of machine (DNS) name is "KRAMER_" + 4 last digits of device serial number.	<b>COMMAND</b> #NAME-RST<CR> <b>FEEDBACK</b> ~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.  ① Parameters, [DNS1] and [DNS2] are optional.  ② For Backward compatibility, the <b>id</b> parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.  ③ 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.	<b>COMMAND</b> #NET-CONFIG_<id>,<ip>,<net_mask>,<gateway>,[<DNS1>],[<DNS2><CR> <b>FEEDBACK</b> ~nn@NET-CONFIG_<id>,<ip>,<net_mask>,<gateway><CR><LF>	<b>id</b> – 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.... <b>ip</b> – Network IP <b>net_mask</b> – Network mask <b>gateway</b> – 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.	<b>COMMAND</b> #NET-CONFIG?_<id><CR> <b>FEEDBACK</b> ~nn@NET-CONFIG_<id>,<ip>,<net_mask>,<gateway><CR><LF>	<b>id</b> – 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.... <b>ip</b> – Network IP <b>net_mask</b> – Network mask <b>gateway</b> – Network gateway	Get network configuration: #NET-CONFIG?_<id><CR>






Function	Description	Syntax	Parameters/Attributes	Example
NET-DHCP	<p>Set DHCP mode.</p> <p>① Only 1 is relevant for the mode value. To disable DHCP, the user must configure a static IP address for the device.</p> <p>Connecting Ethernet to devices with DHCP may take more time in some networks.</p> <p>To connect with a randomly assigned IP by DHCP, specify the device DNS name (if available) using the <b>NAME</b> command. You can also get an assigned IP by direct connection to USB or RS-232 protocol port, if available.</p> <p>For proper settings consult your network administrator.</p> <p>① For Backward compatibility, the <b>id</b> parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.</p>	<p><b>COMMAND</b></p> <pre>#NET-DHCP_id,mode&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-DHCP_id,mode&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>id</b> – 0</p> <p><b>mode</b> –</p> <p>1 – Try to use DHCP. (If unavailable, use the IP address set by the factory or the <b>NET-IP</b> command).</p>	<p>Enable DHCP mode for port 1, if available:</p> <pre>#NET-DHCP_1,1&lt;CR&gt;</pre>
NET-DHCP?	<p>Get DHCP mode.</p> <p>① For Backward compatibility, the <b>id</b> parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.</p>	<p><b>COMMAND</b></p> <pre>#NET-DHCP?_id&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-DHCP_id,mode&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>id</b> – 0</p> <p><b>mode</b> –</p> <p>0 – Do not use DHCP. Use the IP set by the factory or using the <b>NET-IP</b> or <b>NET-CONFIG</b> command.</p> <p>1 – Try to use DHCP. If unavailable, use the IP set by the factory or using the <b>NET-IP</b> or <b>NET-CONFIG</b> command.</p>	<p>Get DHCP mode for port 1:</p> <pre>#NET-DHCP?_1&lt;CR&gt;</pre>
NET-GATE	<p>Set gateway IP.</p> <p>① 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.</p>	<p><b>COMMAND</b></p> <pre>#NET-GATE_ip_address&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-GATE_ip_address&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>ip_address</b> – Format: xxx.xxx.xxx.xxx</p>	<p>Set the gateway IP address to 192.168.0.1:</p> <pre>#NET-GATE_192.168.000.001&lt;CR&gt;</pre>
NET-GATE?	<p>Get gateway IP.</p> <p>① A network gateway connects the device via another network and maybe over the Internet. Be aware of security problems.</p>	<p><b>COMMAND</b></p> <pre>#NET-GATE?_&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-GATE_ip_address&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>ip_address</b> – Format: xxx.xxx.xxx.xxx</p>	<p>Get the gateway IP address:</p> <pre>#NET-GATE?_&lt;CR&gt;</pre>
NET-IP	<p>Set IP address.</p> <p>① For proper settings consult your network administrator.</p>	<p><b>COMMAND</b></p> <pre>#NET-IP_ip_address&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-IP_ip_address&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>ip_address</b> – Format: xxx.xxx.xxx.xxx</p>	<p>Set the IP address to 192.168.1.39:</p> <pre>#NET-IP_192.168.001.039&lt;CR&gt;</pre>
NET-IP?	<p>Get IP address.</p>	<p><b>COMMAND</b></p> <pre>#NET-IP?_&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-IP_ip_address&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>ip_address</b> – Format: xxx.xxx.xxx.xxx</p>	<p>Get the IP address:</p> <pre>#NET-IP?_&lt;CR&gt;</pre>
NET-MAC?	<p>Get MAC address.</p> <p>① For backward compatibility, the <b>id</b> parameter can be omitted. In this case, the Network ID, by default, is 0, which is the Ethernet control port.</p>	<p><b>COMMAND</b></p> <pre>#NET-MAC_id,mac_address&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-MAC_id,mac_address&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>id</b> – 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...</p> <p><b>mac_address</b> – Unique MAC address. Format: XX-XX-XX-XX-XX-XX where X is hex digit</p>	<pre>#NET-MAC?_id&lt;CR&gt;</pre>
NET-MASK	<p>Set subnet mask.</p> <p>① For proper settings consult your network administrator.</p>	<p><b>COMMAND</b></p> <pre>#NET-MASK_net_mask&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-MASK_net_mask&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>net_mask</b> – Format: xxx.xxx.xxx.xxx</p>	<p>Set the subnet mask to 255.255.0.0:</p> <pre>#NET-MASK_255.255.000.000&lt;CR&gt;</pre>
NET-MASK?	<p>Get subnet mask.</p>	<p><b>COMMAND</b></p> <pre>#NET-MASK?_&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@NET-MASK_net_mask&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>net_mask</b> – Format: xxx.xxx.xxx.xxx</p>	<p>Get the subnet mask:</p> <pre>#NET-MASK?&lt;CR&gt;</pre>
PASS	<p>Set password for login level.</p> <p>① The default password is an empty string.</p>	<p><b>COMMAND</b></p> <pre>#PASS_login_level,password&lt;CR&gt;</pre> <p><b>FEEDBACK</b></p> <pre>~nn@PASS_login_level,password&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>login_level</b> – Level of login to set (End User or Administrator).</p> <p><b>password</b> – Password for the <b>login_level</b>. Up to 15 printable ASCII chars</p>	<p>Set the password for the Admin protocol permission level to 33333:</p> <pre>#PASS_Admin,33333&lt;CR&gt;</pre>

Function	Description	Syntax	Parameters/Attributes	Example
PASS?	Get password for login level.  ① The default password is an empty string.	<b>COMMAND</b> #PASS?_login_level<CR> <b>FEEDBACK</b> ~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.  ① 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.	<b>COMMAND</b> #PORTS-LIST?_<CR> <b>FEEDBACK</b> ~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.	<b>COMMAND</b> #PROT-VER?_<CR> <b>FEEDBACK</b> ~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.  ① To avoid locking the port due to a USB bug in Windows, disconnect USB connections immediately after running this command. If the port was locked, disconnect and reconnect the cable to reopen the port.	<b>COMMAND</b> #RESET<CR> <b>FEEDBACK</b> ~nn@RESET_OK<CR><LF>		Reset the device: #RESET<CR>
SECUR	Start/stop security.  ① The permission system works only if security is enabled with the "SECUR" command.	<b>COMMAND</b> #SECUR_security_mode<CR> <b>FEEDBACK</b> ~nn@SECUR_security_mode<CR><LF>	security_mode – 0 – OFF (disables security) 1 – ON (enables security)	Enable the permission system: #SECUR_0<CR>
SECUR?	Get current security state.  ① The permission system works only if security is enabled with the "SECUR" command.	<b>COMMAND</b> #SECUR?_<CR> <b>FEEDBACK</b> ~nn@SECUR_security_mode<CR><LF>	security_mode – 0 – OFF (disables security) 1 – ON (enables security)	Get current security state: #SECUR?_<CR>
SIGNAL?	Get input signal status.	<b>COMMAND</b> #SIGNAL?_inp_id<CR> <b>FEEDBACK</b> ~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>
SIGNALS-LIST?	Get signal ID list of this machine.  ① 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.	<b>COMMAND</b> #SIGNALS-LIST?_<CR><LF> <b>FEEDBACK</b> ~nn@SIGNALS-LIST_[<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,<,><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> – ○ AUDIO ○ 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.	<b>COMMAND</b> #SN?_<CR> <b>FEEDBACK</b> ~nn@SN_serial_number<CR><LF>	serial_number – 14 decimal digits, factory assigned	Get the device serial number: #SN?_<CR>

Function	Description	Syntax	Parameters/Attributes	Example
TIME?	<p>Get device time and date.</p> <p>ⓘ The year must be 4 digits.</p> <p>The device does not validate the day of week from the date.</p> <p>Time format - 24 hours.</p> <p>Date format - Day, Month, Year.</p>	<pre>COMMAND #TIME?_&lt;CR&gt; FEEDBACK ~nn@TIME_&lt;day_of_week&gt;,&lt;date&gt;,&lt;time&gt;&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>day_of_week</b> – One of {SUN,MON,TUE,WED,THU,FRI,SAT}</p> <p><b>date</b> – Format: YYYY/MM/DD where            YYYY = Year            MM = Month            DD = Day</p> <p><b>time</b> – Format: hh:mm:ss where            hh = hours            mm = minutes            ss = seconds</p>	Get device time and date: #TIME?<CR>
TIME-LOC?	<p>Get local time offset from UTC/GMT.</p> <p>ⓘ 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.</p> <p>TIME command sets the device time without considering these settings.</p>	<pre>COMMAND #TIME-LOC?_&lt;CR&gt; FEEDBACK ~nn@TIME-LOC_&lt;UTC_off&gt;,&lt;DayLight&gt;&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>UTC_off</b> – Offset of device time from UTC/GMT (without daylight time correction)</p> <p><b>DayLight</b> –            0 – no daylight saving time            1 – daylight saving time</p>	Get local time offset from UTC/GMT: #TIME-LOC?<CR>
TIME-SRV	<p>Get time server.</p> <p>ⓘ This command is needed for setting UDP timeout for the current client list.</p>	<pre>COMMAND #TIME-SRV_&lt;mode&gt;,&lt;time_server_IP&gt;,&lt;time_server_Sync_Hour&gt;&lt;CR&gt; FEEDBACK ~nn@TIME-SRV_&lt;mode&gt;,&lt;time_server_ip&gt;,&lt;time_server_Sync_Hour&gt;,&lt;server_status&gt;&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>mode</b> – On/Off            0 – Off            1 – On</p> <p><b>time_server_ip</b> – Time server IP address</p> <p><b>time_server_Sync_Hour</b> – Hour in day for time server sync</p> <p><b>server_status</b> –            0 – Off            1 – On</p>	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?	<p>Get time server.</p> <p>ⓘ This command is needed for setting UDP timeout for the current client list.</p>	<pre>COMMAND #TIME-SRV?_&lt;CR&gt; FEEDBACK ~nn@TIME-SRV_&lt;mode&gt;,&lt;time_server_ip&gt;,&lt;time_server_Sync_Hour&gt;,&lt;server_status&gt;&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>mode</b> – On/Off            0 – Off            1 – On</p> <p><b>time_server_ip</b> – Time server IP address</p> <p><b>time_server_Sync_Hour</b> – Hour in day for time server sync</p> <p><b>server_status</b> –            0 – Off            1 – On</p>	Get time server: #TIME-SRV?<CR>
VERSION?	<p>Get firmware version number.</p>	<pre>COMMAND #VERSION?_&lt;CR&gt; FEEDBACK ~nn@VERSION_&lt;firmware_version&gt;&lt;CR&gt;&lt;LF&gt;</pre>	<p><b>firmware_version</b> – XX.XX.XXXX where the digit groups are: major.minor.build version</p>	Get the device firmware version number: #VERSION?_<CR>
X-AUD-LVL	<p>Set audio level of a specific signal.</p> <p>ⓘ This is an Extended Protocol 3000 command.</p>	<pre>COMMAND #X-AUD-LVL_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,&lt;audio_level&gt;&lt;CR&gt; FEEDBACK ~nn@X-AUD-LVL_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,&lt;audio_level&gt;&lt;CR&gt;&lt;LF&gt;</pre>	<p>The following attributes comprise the signal ID:</p> <ul style="list-style-type: none"> <li>▪ <b>&lt;direction_type&gt;</b> –           <ul style="list-style-type: none"> <li>○ IN</li> <li>○ OUT</li> </ul> </li> <li>▪ <b>&lt;port_type&gt;</b> –           <ul style="list-style-type: none"> <li>○ HDMI_AUDIO</li> <li>○ ANALOG_AUDIO</li> <li>○ ANALOG_STEREO</li> <li>○ USB_B</li> <li>○ GENERATOR</li> </ul> </li> <li>▪ <b>&lt;port_index&gt;</b> – The port number as printed on the front or rear panel</li> <li>▪ <b>&lt;signal_type&gt;</b> –           <ul style="list-style-type: none"> <li>○ AUDIO</li> </ul> </li> <li>▪ <b>&lt;index&gt;</b> – Indicates a specific channel number when there are multiple channels of the same type 1 to 2.</li> </ul> <p><b>audio_level</b> – Audio level in dB (range between -60 to +30) depending of the ability of the product</p>	Set the audio level of analog audio specific signal to 10: #X-AUD-LVL_IN.ANALOG_AUDIO.5.AUDIO.1,10<CR>



Function	Description	Syntax	Parameters/Attributes	Example
X-AUD-LVL?	Get audio level of a specific signal.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-AUD-LVL?<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~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?<u>OUT.ANALOG_AUDIO.1.AUDIO.1<CR>
X-AUD-MODE	Set line/Mic mode.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-AUD-MODE<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,mode<CR> <b>FEEDBACK</b> ~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<u>IN.ANALOG_AUDIO.5.AUDIO.1,2<CR>
X-AUD-MODE?	Get line/Mic mode.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-AUD-MODE?<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~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?<u>IN.ANALOG_AUDIO.5.AUDIO.1<CR>
X-LABEL	Set the port label.  ① Labels are used commonly by webpages.  This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-LABEL<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,label_text<CR><LF> <b>FEEDBACK</b> ~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<u>IN.HDMI.1.VIDEO.1,Blu_ray<CR>

Function	Description	Syntax	Parameters/Attributes	Example
<b>X-LABEL?</b>	<p>Get the port label.</p> <p> Labels are used commonly by webpages.</p> <p>This is an Extended Protocol 3000 command.</p>	<pre>COMMAND #X-LABEL?_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;&lt;CR&gt;&lt;LF&gt; FEEDBACK ~nn@X-LABEL_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,label_text&lt;CR&gt;&lt;LF&gt;</pre>	<p>The following attributes comprise the signal ID:</p> <ul style="list-style-type: none"> <li>&lt;direction_type&gt; – <ul style="list-style-type: none"> <li>IN</li> <li>OUT</li> </ul> </li> <li>&lt;port_type&gt; – <ul style="list-style-type: none"> <li>o HDMI</li> <li>o HDMI_AUDIO</li> <li>o ANALOG_AUDIO</li> <li>o ANALOG_STEREO</li> <li>o USB_B</li> <li>o GENERATOR</li> </ul> </li> <li>&lt;port_index&gt; – The port number as printed on the front or rear panel</li> <li>&lt;signal_type&gt; – <ul style="list-style-type: none"> <li>AUDIO</li> <li>VIDEO</li> </ul> </li> <li>&lt;index&gt; – Indicates a specific channel number when there are multiple channels of the same type 1 to 2</li> </ul> <p>Label_text – ASCII characters without space</p>	<p>Get the port label for HDMI IN1:</p> <pre>#X-LABEL?_IN.HDMI.1.VIDEO.1&lt;CR&gt;</pre>
<b>X-LINK-GROUP</b>	<p>Set link between signals in a group:</p> <p>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.</p> <p> This is an Extended Protocol 3000 command.</p>	<pre>COMMAND #X-LINK-GROUP_group_id,linked_state&lt;CR&gt; FEEDBACK ~nn@X-LINK-GROUP_group_id,linked_state&lt;CR&gt;&lt;LF&gt;</pre>	<p>group_id –</p> <ul style="list-style-type: none"> <li>1 – Input analog audio 2 and 3.</li> <li>2 – Input analog audio 4 and 5.</li> </ul> <p>Linked_state – OFF/ON (not case sensitive)</p>	<p>Set link for group 1 (AUDIO IN 2 and 3) to OFF:</p> <pre>#X-LINK-GROUP_1,OFF&lt;CR&gt;</pre>
<b>X-LINK-GROUP?</b>	<p>GET LINK-MODE feature:</p> <p>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.</p> <p> This is an Extended Protocol 3000 command. Used essentially by the web command.</p>	<pre>COMMAND #X-LINK-GROUP?_group_id&lt;CR&gt; FEEDBACK ~nn@X-LINK-GROUP_group_id,linked_state&lt;CR&gt;&lt;LF&gt;</pre>	<p>group_id –</p> <ul style="list-style-type: none"> <li>1 – Input analog audio 2 and 3.</li> <li>2 – Input analog audio 4 and 5.</li> </ul> <p>Linked_state – OFF/ON (not case sensitive)</p>	<p>Get the group 1 link status:</p> <pre>#X-LINK-GROUP?_1&lt;CR&gt;</pre>
<b>X-LINK-GROUPS-LIST?</b>	<p>LINK-MODE feature:</p> <p>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.</p> <p>The LINK mode of a group is defined using the command: <b>X-LINK-GROUP</b></p> <p> This is an Extended Protocol 3000 command. Used essentially by the web</p>	<pre>COMMAND #X-LINK-GROUPS-LIST?&lt;CR&gt; FEEDBACK ~nn@X-POE_[[group_id,is_linked,[signal_id,...,signal_id]]&lt;CR&gt;&lt;LF&gt;</pre>	<p>group_id – 1 to 2</p> <p>Linked_state – OFF/ON (not case sensitive)</p> <p>signal_id – includes:</p> <ul style="list-style-type: none"> <li>▪ &lt;direction_type&gt; – <ul style="list-style-type: none"> <li>o IN</li> </ul> </li> <li>▪ &lt;port_type&gt; – <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> </ul> </li> <li>▪ &lt;index&gt; – 2 to 5</li> </ul> <p>&lt;signal_type&gt; – AUDIO</p> <p>&lt;index&gt; – Indicates a specific channel number when there are multiple channels of the same type 1 to 2</p>	<p>Get the link state for all ports:</p> <pre>#X-LINK-GROUPS-LIST?_&lt;CR&gt;</pre> <pre>[[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]]]</pre>
<b>X-MIC-TYPE</b>	<p>Set microphone type.</p> <p> This is an Extended Protocol 3000 command.</p>	<pre>COMMAND #X-MIC-TYPE_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,mic_type&lt;CR&gt; FEEDBACK ~nn@X-MIC-TYPE_&lt;direction_type&gt;.&lt;port_type&gt;.&lt;port_index&gt;.&lt;signal_type&gt;.&lt;index&gt;,mic_type&lt;CR&gt;&lt;LF&gt;</pre>	<p>The following attributes comprise the port ID:</p> <ul style="list-style-type: none"> <li>▪ &lt;direction_type&gt; – <ul style="list-style-type: none"> <li>o IN</li> </ul> </li> <li>▪ &lt;port_type&gt; – <ul style="list-style-type: none"> <li>o ANALOG_AUDIO</li> </ul> </li> <li>▪ &lt;port_index&gt; – The port number as printed on the front or rear panel: 2 to 5.</li> </ul> <p>&lt;signal_type&gt; – AUDIO</p> <p>&lt;index&gt; – Indicates a specific channel number when there are multiple channels of the same type: 1 to 2</p> <p>mic_type – Dynamic/Condenser (not case sensitive)</p>	<p>Set AUDIO IN 2 type to condenser:</p> <pre>#X-MIC-TYPE_IN.ANALOG_AUDIO.3.AUDIO.1,condense&lt;CR&gt;</pre>

Function	Description	Syntax	Parameters/Attributes	Example
<b>X-MIC-TYPE?</b>	Get microphone type.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MIC-TYPE?_<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~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> – ○ IN ▪ <port_type> – ○ 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>
<b>X-MIX-LVL</b>	Set DSP matrix cross-point MIX level in dB.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MIX-LVL_OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index>,dB<CR> <b>FEEDBACK</b> ~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	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>
<b>X-MIX-LVL?</b>	Get DSP matrix cross-point MIX level in dB.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MIX-LVL?_OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~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>
<b>X-MIX-MUTE</b>	Set DSP matrix cross-point mute state.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MIX-MUTE_OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index>,dB<CR> <b>FEEDBACK</b> ~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>

Function	Description	Syntax	Parameters/Attributes	Example
<b>X-MIX-MUTE?</b>	Get DSP matrix cross-point mute state.  ⓘ This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MIX-MUTE?_OUT.<port_type>.<port_index>.<signal_type>.<index>,IN.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~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 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 ▪ <mute_state> – o ON o 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>
<b>X-MUTE</b>	Set the mute state of the signal.  ⓘ This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MUTE_<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,state<CR> <b>FEEDBACK</b> ~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 <b>state</b> – 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>
<b>X-MUTE?</b>	Get the mute state of the signal.  ⓘ This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-MUTE?_<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~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 <b>state</b> – 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>
<b>X-PATTERN</b>	Set a pattern on the selected output.  ⓘ This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-PATTERN_<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_id<CR> <b>FEEDBACK</b> ~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 <b>Pattern_id</b> – 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>

Function	Description	Syntax	Parameters/Attributes	Example
<b>X-PATTERN?</b>	Get the pattern on a selected output.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-PATTERN?_<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~nn@X-PATTERN_<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_id<CR><LF>	The following attributes comprise the signal ID: <ul style="list-style-type: none"> <li>▪ &lt;direction_type&gt; – <ul style="list-style-type: none"> <li>○ OUT</li> </ul> </li> <li>▪ &lt;port_type&gt; – <ul style="list-style-type: none"> <li>○ HDMI</li> </ul> </li> <li>▪ &lt;port_index&gt; – The port number as printed on the front or rear panel</li> <li>▪ &lt;signal_type&gt; – <ul style="list-style-type: none"> <li>○ VIDEO</li> </ul> </li> <li>▪ &lt;index&gt; – Indicates a specific channel number when there are multiple channels of the same type</li> </ul> <b>Pattern_id</b> – pattern ID <ul style="list-style-type: none"> <li>○ 0 : none</li> <li>○ 1 : Color bar</li> <li>○ 2 Blue screen</li> <li>○ 3: Green screen</li> <li>○ 4: Red screen</li> </ul>	Get the pattern on HDMI output: #X-PATTERN?_OUT.HDMI.1.VIDEO.1<CR>
<b>X-PATTERNS-LIST?</b>	Get the pattern list of a selected output.  ① This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-PATTERN-LIST?_<direction_type>.<port_type>.<port_index>.<signal_type>.<index><CR> <b>FEEDBACK</b> ~nn@X-PATTERN-LIST_<direction_type>.<port_type>.<port_index>.<signal_type>.<index>,pattern_list<CR><LF>	The following attributes comprise the signal ID: <ul style="list-style-type: none"> <li>▪ &lt;direction_type&gt; – <ul style="list-style-type: none"> <li>○ OUT</li> </ul> </li> <li>▪ &lt;port_type&gt; – <ul style="list-style-type: none"> <li>○ HDMI</li> </ul> </li> <li>▪ &lt;port_index&gt; – The port number as printed on the front or rear panel</li> <li>▪ &lt;signal_type&gt; – <ul style="list-style-type: none"> <li>○ VIDEO</li> </ul> </li> <li>▪ &lt;index&gt; – Indicates a specific channel number when there are multiple channels of the same type</li> </ul> <b>Pattern_id</b> – pattern ID <ul style="list-style-type: none"> <li>○ 0 : none</li> <li>○ 1 : Color bar</li> <li>○ 2 Blue screen</li> <li>○ 3: Green screen</li> <li>○ 4: Red screen</li> </ul>	Get the pattern list for HDMI OUT: #X-PATTERN-LIST_OUT.HDMI.1.VIDEO.1<CR>
<b>X-PRST-CURR?</b>	Get the current preset loaded per type.  To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b>  This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-PRST-CURR?_<preset_type><CR> <b>FEEDBACK</b> ~nn@X-ROUTE_<preset_type>,[preset_id:name:lock_state]<CR><LF>	<ul style="list-style-type: none"> <li>▪ &lt;preset_type&gt; – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ &lt;preset_id&gt; – preset index</li> <li>▪ &lt;name&gt; – the name of the preset in URL encode format</li> <li>▪ &lt;lock_state&gt; – <ul style="list-style-type: none"> <li>○ ON</li> <li>○ OFF</li> </ul> </li> </ul>	Get current mixer preset: <b>X-PRST-CURR?_IOCONFIG.SYSTEM.MIXER&lt;CR&gt;</b> ~01@X-PRST-CURR IOConfig.SYSTEM.MIXER,[1:Default:ON]<CR><LF>
<b>X-PRST-LOCK</b>	Set LOCK state of a preset per type.  ① 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: <b>X-PRST-TYPES?</b>  This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-PRST-LOCK_<preset_type>,preset_id,lock_state<CR> <b>FEEDBACK</b> ~nn@X-ROUTE_<preset_type>,[preset_id:name:lock_state]<CR><LF>	<ul style="list-style-type: none"> <li>▪ &lt;preset_type&gt; – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ &lt;preset_id&gt; – preset index</li> <li>▪ &lt;lock_state&gt; – <ul style="list-style-type: none"> <li>○ ON</li> <li>○ OFF</li> </ul> </li> </ul>	lock mixer preset 9: <b>X-PRST-LOCK_IOCONFIG.SYSTEM.MIXER,9&lt;CR&gt;</b>
<b>X-PRST-LOCK?</b>	Get LOCK state of a preset per type.  ① 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: <b>X-PRST-TYPES?</b>  This is an Extended Protocol 3000 command.	<b>COMMAND</b> #X-PRST-LOCK?_<preset_type>,preset_id,lock_state<CR> <b>FEEDBACK</b> ~nn@X-ROUTE_<preset_type>,[preset_id:name:lock_state]<CR><LF>	<ul style="list-style-type: none"> <li>▪ &lt;preset_type&gt; – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ &lt;preset_id&gt; – preset index</li> <li>▪ &lt;lock_state&gt; – <ul style="list-style-type: none"> <li>○ ON</li> <li>○ OFF</li> </ul> </li> </ul>	Get lock mixer preset 9 status: <b>X-PRST-LOCK?_IOCONFIG.SYSTEM.MIXER,9&lt;CR&gt;</b>

Function	Description	Syntax	Parameters/Attributes	Example
<b>X-PRST-LST?</b>	<p>Get the preset list of a specific preset type.</p> <p><b>i</b> 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.</p> <p>To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b></p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> <b>#X-PRST-LST?</b><u>pres</u>et_type&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-LST,&lt;preset_type,[preset_id:name:lock_state]&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </b></li> <li>▪ <b>&lt;name&gt;</b> – the name of the preset</li> <li>▪ <b>&lt;lock_state&gt;</b> – <ul style="list-style-type: none"> <li>○ ON</li> <li>○ OFF</li> </ul> </li> </ul>	<p>Get the IO configuration list: <b>X-PRST-LST?</b><u>IOCONFIG</u>&lt;CR&gt;</p> <p><b>[[1:4x16:ON],[2:6x14:ON],[3:8x12:ON],[4:10x10:ON],[5:12x8:ON],[6:14x6:ON],[7:16x4:ON]]</b></p>
<b>X-PRST-NAME</b>	<p>Set the name of a preset per type.</p> <p><b>i</b> 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.</p> <p>To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b></p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> <b>#X-PRST-NAME</b><u>pres</u>et_type,preset_id,name&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-NAME,&lt;preset_type,preset_id,name&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </b></li> <li>▪ <b>preset_id – preset index</b></li> <li>▪ <b>name</b> – the name of the preset in URL encode format (no spaces)</li> </ul>	<p>Set the name of a preset (per type): <b>X-PRST-NAME?</b><u>IOCONFIG</u>.SYSTEM.MIXER,9,ROOM1&lt;CR&gt;</p>
<b>X-PRST-NAME?</b>	<p>Get the name of a preset per type.</p> <p><b>i</b> 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.</p> <p>To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b></p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> <b>#X-PRST-NAME?</b><u>pres</u>et_type,preset_id,name&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-NAME,&lt;preset_type,preset_id,name&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </b></li> <li>▪ <b>preset_id – preset index</b></li> <li>▪ <b>name</b> – the name of the preset in URL encode format</li> </ul>	<p>Get the name of a preset (per type): <b>X-PRST-NAME?</b><u>IOCONFIG</u>.SYSTEM.MIXER,9&lt;CR&gt;</p>
<b>X-PRST-RCL</b>	<p>Recall saved preset list per type.</p> <p><b>i</b> 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.</p> <p>To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b></p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> <b>#X-PRST-RCL</b><u>pres</u>et_type,preset_id&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-RCL,&lt;preset_type,preset_id&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </b></li> <li>▪ <b>preset_id – preset index</b></li> </ul>	<p>Recall mixer preset 8: <b>X-PRST-RCL?</b><u>IOCONFIG</u>.SYSTEM.MIXER,8&lt;CR&gt;</p>

Function	Description	Syntax	Parameters/Attributes	Example
X-PRST-RCL-LAST	<p>Recall LAST preset per type, this command just retrieves the last preset loaded from the history of preset activity and RECALLS it.</p> <p>ⓘ 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.</p> <p>To get the list of preset types existing in your product use the command: X-PRST-TYPES?</p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> #X-PRST-RCL-LAST_preset_type &lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-RCL-LAST_preset_type,preset_id&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ preset_id – preset index</li> </ul>	<p>Recall the last mixer preset: X-PRST-RCL-LAST_IOCONFIG.SYSTEM.MIXER&lt;CR&gt;</p>
X-PRST-RCL-NEXT	<p>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.</p> <p>ⓘ 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.</p> <p>To get the list of preset types existing in your product use the command: X-PRST-TYPES?</p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> #X-PRST-RCL-NEXT_preset_type&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-RCL-NEXT_preset_type,preset_id&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ preset_id – preset index</li> </ul>	<p>Recall next mixer preset: X-PRST-RCL-NEXT_IOCONFIG.SYSTEM.MIXER&lt;CR&gt;</p>
X-PRST-RCL-PREV	<p>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.</p> <p>ⓘ 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.</p> <p>To get the list of preset types existing in your product use the command: X-PRST-TYPES?</p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> #X-PRST-RCL-PREV_preset_type&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-RCL-PREV_preset_type,preset_id&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ preset_id – preset index</li> </ul>	<p>Recall previous mixer preset: X-PRST-RCL-PREV_IOCONFIG.SYSTEM.MIXER&lt;CR&gt;</p>
X-PRST-RESET	<p>Reset preset per type</p> <p>ⓘ 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.</p> <p>To get the list of preset types existing in your product use the command: X-PRST-TYPES?</p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> #X-PRST-RESET_preset_type,preset_id&lt;CR&gt;</p> <p><b>FEEDBACK</b> ~nn@X-PRST-RESET_preset_type,preset_id&lt;CR&gt;&lt;LF&gt;</p>	<ul style="list-style-type: none"> <li>▪ preset_type – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ preset_id – preset index</li> </ul>	<p>Reset mixer preset 9: X-PRST-RESET_IOCONFIG.SYSTEM.MIXER,9&lt;CR&gt;</p>

Function	Description	Syntax	Parameters/Attributes	Example
<b>X-PRST-<u>SAVED?</u></b>	<p>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.</p> <p><b>i</b> 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.</p> <p>To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b></p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> <b>#X-PRST-<u>SAVED?</u> preset_type &lt;CR&gt;</b></p> <p><b>FEEDBACK</b> <b>~nn@X-PRST-<u>SAVED?</u> preset_type, saved_status &lt;CR&gt;&lt;LF&gt;</b></p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type</b> – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ <b>Saved_status</b> – preset index <ul style="list-style-type: none"> <li>○ 0 – False (not saved)</li> <li>○ 1 – True (saved)</li> </ul> </li> </ul>	<p>Get saved status of mixer preset: <b>X-<u>SAVED?</u> IOCONFIG.SYSTEM.MIXER &lt;CR&gt;</b></p>
<b>X-PRST-<u>STO</u></b>	<p>Store current changes into a preset (per type).</p> <p><b>i</b> 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.</p> <p>To get the list of preset types existing in your product use the command: <b>X-PRST-TYPES?</b></p> <p>This is an Extended Protocol 3000 command.</p>	<p><b>COMMAND</b> <b>#X-PRST-<u>STO</u> preset_type, preset_id &lt;CR&gt;</b></p> <p><b>FEEDBACK</b> <b>~nn@X-PRST-<u>STO</u> preset_type, saved_status &lt;CR&gt;&lt;LF&gt;</b></p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type</b> – <ul style="list-style-type: none"> <li>○ System Preset – IOCONFIG.SYSTEM</li> <li>○ Snapshot – IOCONFIG.SYSTEM.MIXER</li> </ul> </li> <li>▪ <b>preset_id</b> – preset index</li> </ul>	<p>Store changes into mixer preset 9: <b>X-PRST-<u>STO</u> IOCONFIG.SYSTEM.MIXER, 9 &lt;CR&gt;</b></p>
<b>X-PRST-<u>TYPES?</u></b>	<p>Get the types of presets that the system supports and their hierarchy.</p>	<p><b>COMMAND</b> <b>#X-PRST-<u>TYPES?</u> &lt;CR&gt;</b></p> <p><b>FEEDBACK</b> <b>~nn@X-PRST-<u>TYPES?</u> preset_type &lt;CR&gt;&lt;LF&gt;</b></p>	<ul style="list-style-type: none"> <li>▪ <b>preset_type</b> – <ul style="list-style-type: none"> <li>○ 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.</li> <li>○ 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.</li> </ul> </li> </ul>	<p>Get preset types: <b>X-PRST-<u>TYPES?</u> &lt;CR&gt;</b></p>



---

# 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_TIMEOUT	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