



# ULX-D

## ULX-D Command Strings

Third-party command string information for the Shure ULX-D wireless system.  
Version: 0.1 (2022-D)

---

# Table of Contents

<b>ULX-D ULX-D Command Strings</b>	<b>3</b>	<b>Channel Command Strings</b>	<b>9</b>
<b>Device Command Strings</b>	<b>3</b>	<b>Metering Command Strings</b>	<b>12</b>
		<b>Side Channel Command Strings</b>	<b>14</b>

# ULX-D

## ULX-D Command Strings

The ULX-D device is connected via Ethernet to a control system, such as

- AMX, Crestron, or Extron
- Symetrix, Biamp, other digital signal processors (DSP)
- Specialized custom programs

**Connection:** Ethernet (TCP/IP; select "Client" in the AMX/Crestron program)

**Port:** 2202

### Conventions

There are 4 types of strings:

<b>GET</b>	Finds the status of a property. After the AMX/Crestron sends a GET command, the system responds with a REPORT string
<b>SET</b>	Changes the status of a property. After the AMX/Crestron sends a SET command, the system responds with a REPORT string to indicate the new value of the property.
<b>REP</b>	When the system receives a GET or SET command, it replies with a REPORT command to indicate the status of the property.  REP ERR occurs when a command is improperly formatted or when the values are out of range.  <b>Important:</b> With the exception of the metered properties, the device sends a REPORT when a value changes. Thus, it is not necessary to constantly query most device properties.
<b>SAMPLE</b>	Used for metering audio levels.

All messages sent and received are ASCII. Note that the level indicators and gain indicators are also in ASCII

The character "x" in all of the following strings represents the channel and can be ASCII numbers 0 through 4 as in the following table.

<b>0</b>	All channels
<b>1 through 4</b>	Individual channels

### ULX-D Naming

- ULXD4 is a 1-channel device
- ULXD4D is a 2-channel device (D - Dual)
- ULXD4Q is a 4-channel device (Q - Quad)

# Device Command Strings

## ALL

<b>Description</b>	Discovery of all report strings for the selected channel(s).
<b>Commands</b>	<pre>&lt; GET x ALL &gt; &lt; REP ... &gt;</pre>
<b>Variables</b>	<p>When <b>x</b> is zero, the device responds with REP for all device-specific properties and ALL channel-related properties including all metered properties.</p> <p>When <b>x</b> is a channel number, the device responds with REP for all device-specific properties and ALL channel <b>x</b>-related properties including all metered properties.</p>
<b>Notes</b>	<p>This group is only sent once on receipt of the GET ALL command.</p> <p>Examples:</p> <pre>&lt; REP x METER_RATE ##### &gt; &lt; REP x CHAN_NAME {string} &gt; &lt; REP x ENCRYPTION v &gt; &lt; REP x AUDIO_GAIN ## &gt; &lt; REP x GROUP_CHAN gg cc &gt; &lt; REP x FREQUENCY ##### &gt;</pre>

## AUDIO\_SUMMING\_MODE

<b>Description</b>	Controls device-based audio summing mode.
<b>Commands</b>	<pre>&lt; GET AUDIO_SUMMING_MODE &gt; &lt; SET AUDIO_SUMMING_MODE value &gt; &lt; REP AUDIO_SUMMING_MODE value &gt;</pre>
<b>Variables</b>	None.
<b>Notes</b>	<p>Values can be:</p> <p>OFF - Audio summing mode is disabled.</p> <p>1+2 - Audio summing mode is enabled for channel pair 1/2.</p> <p>3+4 - Audio summing mode is enabled for channel pair 3/4.</p> <p>1+2/3+4 - Only applies to quad Rx. Audio summing mode is enabled for channel pairs 1/2 and 3/4. Ignored by device if not valid. Device reports current status without applying any change.</p> <p>1+2+3+4 - Only applies to quad Rx. Ignored by device if not valid. Device reports current status without applying any change.</p>

## DEVICE\_ID

<b>Description</b>	Controls the device ID.
<b>Commands</b>	<pre>&lt; GET DEVICE_ID &gt; &lt; REP DEVICE_ID {yyyyyyyy} &gt;  &lt; SET DEVICE_ID {yyyyyyyy} &gt; &lt; REP DEVICE_ID {yyyyyyyy} &gt;</pre>
<b>Variables</b>	Where the repeating <b>y</b> represents or pads the 8-character string
<b>Notes</b>	A blank string represents an unknown device ID.

## ENCRYPTION

<b>Description</b>	Controls the encryption mode.
<b>Commands</b>	<pre>&lt; GET ENCRYPTION &gt; &lt; REP ENCRYPTION setting &gt;  &lt; SET ENCRYPTION setting &gt; &lt; REP ENCRYPTION setting &gt;</pre>
<b>Variables</b>	None.
<b>Notes</b>	<p>Valid values for setting are:</p> <p>OFF - Encryption mode is disabled</p> <p>AUTO - Encryption mode is set to automatic control.</p> <p>MANUAL - Encryption mode is set to manual control. Manual must be used to regenerate the encryption key.</p>

## ENCRYPTION\_REGENERATE\_KEY

<b>Description</b>	Controls the regeneration of the encryption key.
<b>Commands</b>	<pre>&lt; SET ENCRYPTION_REGENERATE_KEY value &gt; &lt; REP ENCRYPTION_REGENERATE_KEY value &gt;</pre>
<b>Variables</b>	None.
<b>Notes</b>	<p>This command is only applicable when the encryption is set to MANUAL.</p> <p>Reminder: Sync devices after updating encryption.</p> <p>Values are: REQUESTED, COMPLETED, INACTIVE</p> <p>Examples:</p> <pre>&lt; SET ENCRYPTION_REGENERATE_KEY REQUESTED &gt;</pre>

	<p>SET generates three responses. When the command is accepted:</p> <p>&lt; REP ENCRYPTION_REGENERATE_KEY REQUESTED &gt;</p> <p>When the new key is created:</p> <p>&lt; REP ENCRYPTION_REGENERATE_KEY COMPLETED &gt;</p> <p>When the receiver is ready to accept a new request:</p> <p>&lt; REP ENCRYPTION_REGENERATE_KEY INACTIVE &gt;</p>
--	--

## FLASH

<b>Description</b>	Controls the flash to identify a device or channel.
<b>Commands</b>	<p>&lt; SET FLASH ON &gt; &lt; REP FLASH ON &gt;</p> <p>Device initiates an identify then stops flashing:</p> <p>&lt; REP FLASH OFF &gt;</p> <p>&lt; SET FLASH OFF &gt; &lt; REP FLASH OFF &gt;</p> <p>When used with a channel index, the command initiates a Channel Identify.</p> <p>&lt; SET x FLASH ON &gt; &lt; REP x FLASH ON &gt;</p>
<b>Variables</b>	When used, x is the channel number.
<b>Notes</b>	<p>No REP once the flashing has completed.</p> <p>Flash rate and pattern is fixed and device specific.</p>

## FREQUENCY\_DIVERSITY\_MODE

<b>Description</b>	Controls device-based frequency diversity mode.
<b>Commands</b>	<p>&lt; SET FREQUENCY_DIVERSITY_MODE status &gt; &lt; GET FREQUENCY_DIVERSITY_MODE &gt; &lt; REP FREQUENCY_DIVERSITY_MODE status &gt;</p>
<b>Variables</b>	None.
<b>Notes</b>	<p>Values for the status can be:</p> <p>OFF - Frequency diversity mode is disabled.</p> <p>1+2 - Frequency diversity mode is enabled for channel pair 1/2.</p>

	<p>3+4 - Frequency diversity mode is enabled for channel pair 3/4. Only applies to quad Rx. Ignored by device if not valid. Device reports current status without applying any change.</p> <p>1+2/3+4 - Frequency diversity mode is enabled for channel pairs 1/2 and 3/4. Only applies to quad Rx. Ignored by device if not valid. Device reports current status without applying any change.</p> <p>Reports when channel designator is invalid for the device.</p>
--	--

## FW\_VER

<b>Description</b>	Discovers the firmware version.
<b>Commands</b>	<p>Self test passed:</p> <p>&lt; GET FW_VER &gt;</p> <p>&lt; REP FW_VER {ver} &gt;</p>
<b>Variables</b>	None.
<b>Notes</b>	<p>Version is a 24-character string.</p> <p>Package version number reported as Maj.Min.Pack.Build with values from 00000-65535. Example: 65535.65535.65535.65535*</p> <p>If the version is followed by an * the previous firmware update was incomplete. Please use Shure Update Utility to re-update the device.</p>

## HIGH\_DENSITY

<b>Description</b>	Controls high-density mode.
<b>Commands</b>	<p>&lt; SET HIGH_DENSITY value &gt;</p> <p>&lt; GET HIGH_DENSITY &gt;</p> <p>&lt; REP HIGH_DENSITY value &gt;</p>
<b>Variables</b>	None.
<b>Notes</b>	<p>Values can be:</p> <p>OFF - High-density mode is disabled</p> <p>ON - High-density mode is enabled</p>

## MODEL

<b>Description</b>	Discovery of the device model name.
--------------------	-------------------------------------

<b>Commands</b>	< GET MODEL > < REP MODEL {string} >
<b>Variables</b>	None.
<b>Notes</b>	Format: 32-character string.

## NA\_DEVICE\_NAME

<b>Description</b>	Discovers the Dante device name on dual and quad devices.
<b>Commands</b>	< GET NA_DEVICE_NAME > < REP NA_DEVICE_NAME {string} >
<b>Variables</b>	None.
<b>Notes</b>	<p>The device always responds with 31-character ID. In these examples, the repeating y pads these strings, representing white space.</p> <p>Getting the Dante Device Name from a ULXD4Q that uses Yamaha naming conventions:</p> <pre>&lt; GET NA_DEVICE_NAME &gt; &lt; REP NA_DEVICE_NAME {Y001-Shure-ULXD4Q-435577yyyyyy} &gt;</pre> <p>Getting the name from a device changed by Dante Controller to "GuitarRig3:"</p> <pre>&lt; GET NA_DEVICE_NAME {GuitarRig3} &gt; &lt; REP NA_DEVICE_NAME {GuitarRig3yyyyyyyyyyyyyyyyyyyy} &gt;</pre>

## SCAN\_LOCK

<b>Description</b>	Controls the scan lock.
<b>Commands</b>	< GET SCAN_LOCK > < REP SCAN_LOCK OFF > < SET SCAN_LOCK ON > < REP SCAN_LOCK ON >
<b>Variables</b>	None.
<b>Notes</b>	<p>When SCAN_LOCK is ON:</p> <p>The front panel scan button is ignored.</p> <p>This device prevents any changes if another device deploys a group scan.</p>



## SYNC\_LOCK

<b>Description</b>	Controls the sync lock.
<b>Commands</b>	<pre>&lt; GET SYNC_LOCK &gt; &lt; REP SYNC_LOCK OFF &gt; &lt; SET SYNC_LOCK ON &gt; &lt; REP SYNC_LOCK ON &gt;</pre>
<b>Variables</b>	None.
<b>Notes</b>	<p>When SYNC_LOCK is ON:</p> <p>The front panel sync buttons are ignored. This device prevents IR Sync functions.</p>

## Channel Command Strings

## AUDIO\_GAIN

<b>Description</b>	Control for the channel audio gain.
<b>Commands</b>	<pre>&lt; GET x AUDIO_GAIN &gt; &lt; REP x AUDIO_GAIN ### &gt;  &lt; SET x AUDIO_GAIN # &gt; &lt; REP x AUDIO_GAIN ### &gt;  To decrement the value: &lt; SET x AUDIO_GAIN DEC # &gt; &lt; REP x AUDIO_GAIN ### &gt;  To increment the value: &lt; SET x AUDIO_GAIN INC # &gt; &lt; REP x AUDIO_GAIN ### &gt;</pre>
<b>Variables</b>	Where x is the channel number and # represents dB.
<b>Notes</b>	<p>000 to 060 in increments of 1 The values REPorted and SET are offset by 18 Actual range: -18 to 42 dB in 1 dB steps</p>

## AUDIO\_MUTE

<b>Description</b>	Controls the channel audio mute output.
--------------------	---

<b>Commands</b>	<pre>&lt; GET x AUDIO_MUTE &gt; &lt; REP x AUDIO_MUTE OFF &gt;  &lt; SET x AUDIO_MUTE ON &gt; &lt; REP x AUDIO_MUTE ON &gt;  &lt; SET x AUDIO_MUTE OFF &gt; &lt; REP x AUDIO_MUTE OFF &gt;  &lt; SET x AUDIO_MUTE TOGGLE &gt; &lt; REP x AUDIO_MUTE ON &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel number.
<b>Notes</b>	TOGGLE switches between ON and OFF.

## CHAN\_NAME

<b>Description</b>	Control for the channel name.
<b>Commands</b>	<pre>&lt; GET x CHAN_NAME &gt; &lt; REP x CHAN_NAME {yyyyyyyy} &gt;  &lt; SET x CHAN_NAME {yyyyyyyy} &gt; &lt; REP x CHAN_NAME {yyyyyyyy} &gt;</pre>
<b>Variables</b>	<p>Where <b>x</b> is the channel number.</p> <p>Where the repeating <b>y</b> represents or pads the 8-character string from the set: A-Z,a-z, 0-9,!"#\$\$%&amp;'()*+,-./:;&lt;=&gt;@[ ]^_`~ and space.</p>
<b>Notes</b>	The device always responds with a 8-character name.

## ENCRYPTION\_WARNING

<b>Description</b>	Discovers an encryption mismatch status.
<b>Commands</b>	<pre>&lt; GET x ENCRYPTION_WARNING &gt; &lt; REP x ENCRYPTION_WARNING status &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel number.
<b>Notes</b>	<p>Values for status are:</p> <p>OFF - No mismatch detected</p>

	ON - Mismatch detected.
--	-------------------------

## FREQUENCY

<b>Description</b>	Controls frequency settings.
<b>Commands</b>	<pre>&lt; SET x FREQUENCY ##### &gt; &lt; GET x FREQUENCY ##### &gt; &lt; REP x FREQUENCY ##### &gt; &lt; REP x GROUP_CHAN gg,cc &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel, <b>#####</b> is the frequency in KHz, <b>gg</b> is group, <b>cc</b> is channel.
<b>Notes</b>	Invalid group or channel are reported as "--". Group and channel report only sent on change.

## GROUP\_CHAN

<b>Description</b>	Controls the group and channel.
<b>Commands</b>	<pre>&lt; SET x GROUP_CHAN gg,cc &gt; &lt; GET x GROUP_CHAN &gt; &lt; REP x FREQUENCY ##### &gt; &lt; REP x GROUP_CHAN gg,cc &gt;</pre>
<b>Variables</b>	Where: <b>x</b> is the channel number. <b>#####</b> is the frequency in KHz (6 digits long) <b>gg</b> is a two-character group <b>cc</b> is the two-character channel.
<b>Notes</b>	GETs RF Group & Channel setting for channel x. REPorts group and channel and frequency for channel x. Frequency only reported on change. Invalid group or channel is reported as "--"

## NA\_CHAN\_NAME

<b>Description</b>	Discovers Network Audio (Dante) channel name for dual and quad devices.
--------------------	---

<b>Commands</b>	<pre>&lt; GET x NA_CHAN_NAME &gt; &lt; REP x NA_CHAN_NAME {yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy} &gt;</pre>
<b>Variables</b>	<p>Where <b>x</b> is the channel number.</p> <p>Where the repeating <b>y</b> pads the 31-character channel name with spaces.</p>
<b>Notes</b>	None.

## RF\_INT\_DET

<b>Description</b>	Reports when interference is detected.
<b>Commands</b>	<pre>&lt; GET x RF_INT_DET &gt; &lt; REP x RF_INT_DET status &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel number.
<b>Notes</b>	<p>Valid values for status are:</p> <p>NONE - No interference detected</p> <p>CRITICAL - Interference detected</p>

## Metering Command Strings

Most properties generate REP (Report) messages when their values change (for example, Frequency or Channel Name) For audio meters, RF meters, and the like, a REP on each change is inefficient and can flood many simple control systems.

The Shure approach is to use metering to periodically sample your channels and devices:

- You can still use GET to discover a value when necessary.
- The SAMPLE command packs multiple attributes into a single message.
- You can control the interval of the messages by setting the METER\_RATE.

The following sections detail METER\_RATE and SAMPLE followed by the set of metered attributes.

## METER\_RATE / SAMPLE

<b>Description</b>	Controls the meter rate and reports sampled data.
<b>Commands</b>	<pre>&lt; GET x METER_RATE &gt; &lt; SET x METER_RATE ##### &gt; &lt; REP x METER_RATE ##### &gt;</pre>

	< SAMPLE x ALL nn aaa eee >
<b>Variables</b>	<p>Where <b>x</b> is the channel number.</p> <p>Where ##### is the rate in milliseconds.</p> <p>Where <b>nn</b> indicates the diversity LEDs from the receiver. (See RF_ANTENNA)</p> <p>Where <b>aaa</b> is the value of the RF level received on antenna in-use. (See RX_RF_LVL)</p> <p>Where <b>eee</b> is the audio level and is 000-255. (See AUDIO_LVL)</p>
<b>Notes</b>	<p>00100 to 99999 - The interval of each SAMPLE report in milliseconds.</p> <p>00100 - Sample every 100 millisecond (10 samples per sec)</p> <p>01000 - Sample every second</p> <p>05000 - Sample every 5 seconds</p> <p>To turn off metering, use</p> <p>&lt; SET x METER_RATE 00000 &gt;</p>

## RF\_ANTENNA

<b>Description</b>	Discovers RF antenna status and receiver squelch status.
<b>Commands</b>	<p>&lt; GET x RF_ANTENNA &gt;</p> <p>&lt; REP x RF_ANTENNA nn &gt;</p>
<b>Variables</b>	<p>Where <b>x</b> is the channel number.</p> <p>Where <b>nn</b> indicates the RF LEDs from the receiver.</p>
<b>Notes</b>	<p>This is a metered property. It does not generate a report on a value change.</p> <p>Receiver squelch status:</p> <p>AX - Antenna A is on, Antenna B is off</p> <p>XB - Antenna A is off, Antenna B is on</p> <p>AB - Antenna A is on, Antenna B is on (not possible on ULXD4)</p> <p>XX - Antenna A is off, Antenna B is off</p>

## RX\_RF\_LVL

<b>Description</b>	Discovers the RF received level.
<b>Commands</b>	<p>&lt; GET x RX_RF_LVL &gt;</p> <p>&lt; REP x RX_RF_LVL rssi &gt;</p>
<b>Variables</b>	Where <b>x</b> is the channel.

<b>Notes</b>	<p>This is a metered property using SAMPLE; it does not generate a report on a value change.</p> <p>Numeric, 3-character fixed output per antenna</p> <p>Units: dBm</p> <p>The actual value = the reported value - 128</p> <p>Value range as reported over command strings: 000 - 255</p> <p>Value range after conversion to the actual value: -128 - 127 dBm</p>
--------------	---

## AUDIO\_LVL

<b>Description</b>	Discovery of the receiver RMS audio level.
<b>Commands</b>	<pre>&lt; GET x AUDIO_LVL &gt; &lt; REP x AUDIO_LVL ### &gt;</pre>
<b>Variables</b>	Where x is the channel number and ### is a 3-digit number between 000 and 255.
<b>Notes</b>	<p>This is a metered property using SAMPLE; it does not generate a report on a value change.</p> <p>When metering is enabled, this value is sent at the metered sample rate.</p> <p>Units: dBFS</p> <p>The actual value = the reported value - 50</p>

## Side Channel Command Strings

### BATT\_BARS

<b>Description</b>	Discovers the battery bars.
<b>Commands</b>	<pre>&lt; GET x BATT_BARS &gt; &lt; REP x BATT_BARS ### &gt;</pre>
<b>Variables</b>	Where x is the channel number and ### are the bars.
<b>Notes</b>	<p>Numeric, 3-character fixed output</p> <p>000 to 005 - remaining charge</p> <p>255 - Unknown</p>

### BATT\_CHARGE

<b>Description</b>	Discovers the charge left in rechargeable batteries.
--------------------	--

<b>Commands</b>	< GET x BATT_CHARGE > < REP x BATT_CHARGE ### >
<b>Variables</b>	Where x is the channel number and ### represents the percent of charge remaining.
<b>Notes</b>	Numeric, 3-character fixed output 000 to 100 - Percent 255 - Unknown

## BATT\_CYCLE

<b>Description</b>	Discovers the number of battery cycles as reported via side channel.
<b>Commands</b>	< GET x BATT_CYCLE > < REP x BATT_CYCLE ##### >
<b>Variables</b>	Where x is the channel number and ##### is the cycle count of full charges.
<b>Notes</b>	Numeric, 5-character fixed output 00000 to 65534 - Number of cycles 65535 - Unknown

## BATT\_HEALTH

<b>Description</b>	Discovers the battery health of rechargeable batteries.
<b>Commands</b>	< GET x BATT_HEALTH > < REP x BATT_HEALTH ### >
<b>Variables</b>	Where x is channel number and ### is the battery health in percent.
<b>Notes</b>	Numeric, 3-character fixed output 000 to 100 - Percent 255 - Unknown

## BATT\_RUN\_TIME

<b>Description</b>	Discovers the minutes of run time remaining in a rechargeable microphone battery.
--------------------	---

<b>Commands</b>	<pre>&lt; GET x BATT_RUN_TIME &gt; &lt; REP x BATT_RUN_TIME ##### &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel number and <b>#####</b> is the remaining run time.
<b>Notes</b>	<p>Valid only for microphone units in use, in-range, and out of charger.</p> <p>Numeric, 5-character fixed output</p> <p>00000 to 65532 - Number of minutes of runtime</p> <p>65533 - Battery communication error</p> <p>65534 - Battery time calculating</p> <p>65535 - Unknown, or not applicable</p> <p>Reports when data becomes known. For example, 2 hours 5 minutes would be:</p> <pre>&lt; REP x BATT_RUN_TIME 00125 &gt;</pre>

## BATT\_TEMP\_C

<b>Description</b>	Discovers the battery temperature in Celsius.
<b>Commands</b>	<pre>&lt; GET x BATT_TEMP_C &gt; &lt; REP x BATT_TEMP_C ### &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel number and <b>###</b> is the temperature offset by 40.
<b>Notes</b>	<p>Numeric, 3-character fixed output</p> <p>000 to 254 - Temperature</p> <p>255 - Unknown, or not applicable</p> <p>The actual value = the reported value - 40.</p>

## BATT\_TEMP\_F

<b>Description</b>	Discovers the battery temperature in Fahrenheit.
<b>Commands</b>	<pre>&lt; GET x BATT_TEMP_F &gt; &lt; REP x BATT_TEMP_F ### &gt;</pre>
<b>Variables</b>	Where <b>x</b> is the channel number and <b>###</b> is the temperature offset by 40.



<b>Notes</b>	Numeric, 3-character fixed output 000 to 254 - Temperature 255 - Unknown, or not applicable The actual value = the reported value - 40.
--------------	--

## BATT\_TYPE

<b>Description</b>	Discovers the battery type.
<b>Commands</b>	< GET x BATT_TYPE > < REP x BATT_TYPE batt >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	GET supports determination of battery remaining minutes.  Values for batt can be:  LION ALKA NIMH LITH WARN - Tx is charging on dock or via USB. Only applies if battery is not LION. UNKN - Either no transmitter or not supported by transmitter

## TX\_DEVICE\_ID

<b>Description</b>	Discovers the transmitter's Device ID.
<b>Commands</b>	< GET x TX_DEVICE_ID > < REP x TX_DEVICE_ID {yyyyyyyy} >
<b>Variables</b>	Where x is the channel number.  Where the repeating y represents or pads the 8-character string
<b>Notes</b>	8-character string - Device ID of the transmitter being received  Blank, all spaces - Unknown, not applicable, or there is no transmitter

## TX\_FW\_VER

<b>Description</b>	Discovers the transmitter firmware version.
--------------------	---

<b>Commands</b>	< GET x TX_FW_VER > < REP x TX_FW_VER {ver} >
<b>Variables</b>	Where x is the channel.
<b>Notes</b>	Package version number reported as Maj.Min.Pack.Build. If the version is followed by an * the previous firmware update was incomplete. Please use Shure Update Utility to re-update the device. The REP returns a blank, padded string if the firmware version is unknown or if there is no transmitter.

## TX\_MENU\_LOCK

<b>Description</b>	Discovers the transmitter menu lock status.
<b>Commands</b>	< GET x TX_MENU_LOCK > < REP x TX_MENU_LOCK status >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	Valid values for status are: ON - Transmitter menu is locked OFF - Transmitter menu is unlocked UNKN - Transmitter menu status is unknown, not applicable, or there is no transmitter

## TX\_MUTE\_BUTTON\_STATUS

<b>Description</b>	Discovers the transmitter mute button status of ULXD6 and ULXD8 transmitters.
<b>Commands</b>	< GET x TX_MUTE_BUTTON_STATUS > < REP x TX_MUTE_BUTTON_STATUS status >
<b>Variables</b>	Where x is the channel index.
<b>Notes</b>	Values for status are: PRESSED - Mute button is on, audio is muted RELEASED - Mute button is off, audio is unmuted UNKN - Button status is unknown, not applicable, or there is no transmitter

## TX\_MUTE\_STATUS

<b>Description</b>	Discovers the transmitter mute status of the ULXD6 and ULXD8 transmitters.
--------------------	--

<b>Commands</b>	< GET x TX_MUTE_STATUS > < REP x TX_MUTE_STATUS status >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	Values for status are:  ON - Tx is muted. OFF - Tx is not muted. UNKN - Unknown, not applicable, or there is no transmitter.

## TX\_OFFSET

<b>Description</b>	Discovery of the transmitter offset.
<b>Commands</b>	< GET x TX_OFFSET > < REP x TX_OFFSET ### >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	Numeric, 3-character fixed  Range - 000 to 033  255 - Unknown  The actual value = the reported value - 12. This means that the actual range is -12 to +21 dB in 1 dB increments.

## TX\_POWER\_SOURCE

<b>Description</b>	Discovers the power source of the ULXD6 and ULXD8 transmitters.
<b>Commands</b>	< GET x TX_POWER_SOURCE > < REP x TX_POWER_SOURCE value >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	Values can be:  BATTERY - Powered by batteries EXTERNAL - Power source is external UNKNOWN - Power Source is unknown, not applicable or there is no transmitter

## TX\_PWR\_LOCK

<b>Description</b>	Discovers the transmitter power lock status.
<b>Commands</b>	< GET x TX_PWR_LOCK > < REP x TX_PWR_LOCK status >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	Values for status can be:  ON - Power lock is on and the power switch is disabled OFF - Power lock is off (default) UNKN - Power lock status is unknown

## TX\_RF\_PWR

<b>Description</b>	Discovers the transmitter RF power.
<b>Commands</b>	< GET x TX_RF_PWR > < REP x TX_RF_PWR pwr >
<b>Variables</b>	Where x is the channel number.
<b>Notes.</b>	Values for pwr can be:  LOW NORMAL HIGH UNKN

## TX\_TYPE

<b>Description</b>	Discovers the transmitter type.
<b>Commands</b>	< GET x TX_TYPE > < REP x TX_TYPE model >
<b>Variables</b>	Where x is the channel number.
<b>Notes</b>	Values for model can be:  QLXD1 QLXD2 ULXD1 ULXD2 ULXD6 ULXD8

---

	UNKN - Either no transmitter or not supported by transmitter
--	--