



Axient Digital

Command Strings

Third-party command string information for the Shure Axient Digital wireless system.
Version: 4 (2019-L)

Table of Contents

Axient DigitalCommand Strings	3	Metering Command Strings	11
Device Command Strings	3	Side Channel Command Strings	18
Channel Command Strings	6	Transmitter Slots Command Strings	25

Axient Digital Command Strings

The Axient Digital 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:

GET	Finds the status of a property. After the AMX/Crestron sends a GET command, the system responds with a REPORT string
SET	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.
REP	When the system receives a GET or SET command, it replies with a REPORT command to indicate the status of the property. Important: 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.
SAMPLE	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.

0	All channels
1 through 4	Individual channels

Axient Digital Naming

- AD4Q - Is a 4 channel device (Q - Quad)
- AD4D - is a 2 channel device (D - Dual)
- AD4 channels have 8 transmitter slots
- AD4 channels typically have 1 RF section and 2 Antenna inputs.

Device Command Strings

ALL

Description	Discovery of device properties.
Commands	<code>< GET x ALL ></code> <code>< REP ... ></code>
Variables	When x is zero, the device responds with REP for all device-specific properties and ALL channel-related properties including all metered properties. When x is a channel number, the device responds with REP for all device-specific properties and ALL channel x-related properties including all metered properties.
Notes	None.

DEVICE_ID

Description	Controls the Device ID.
Commands	<code>< GET DEVICE_ID ></code> <code>< REP DEVICE_ID {AD4Q-Ayyyyyyyyyyyyyyyyyyyy} ></code> <code>< SET DEVICE_ID {Name1} ></code> <code>< REP DEVICE_ID {Name1yyyyyyyyyyyyyyyyyyyy} ></code>
Variables	Where the repeating y represents the spaces returned by the device to pad the Device ID to 31 characters.
Notes	The device always responds with 31-character ID. SET accepts 1-8 Characters from the set: A-Z,a-z,0-9,!,"#\$%&'()*+,-./:;<=>?@[\\]^_`~ and space.

ENCRYPTION_MODE

Description	Discovery of the Encryption Mode.
Commands	<code>< GET ENCRYPTION_MODE ></code> <code>< REP ENCRYPTION_MODE ON ></code> or <code>< REP ENCRYPTION_MODE OFF ></code>
Variables	None.
Notes	None.

FLASH

Description	Controls the flash to identify a device or channel.
Commands	< SET FLASH ON > < REP FLASH ON > Device initiates an identify then stops flashing: < REP FLASH OFF > < SET FLASH OFF > < REP FLASH OFF > Note: When used with no channel index the command initiates a Device Identify. When used with a channel index the command initiates a Channel Identify. < SET x FLASH ON > < REP x FLASH ON >
Variables	When used, x is the channel number.
Notes	None.

FW_VER

Description	Discovery of the firmware version.
Commands	Self test passed: < GET FW_VER > < REP FW_VER {2.0.15.2yyyyyyyyyyyyyyyy} > Self test failed: < GET FW_VER > < REP FW_VER {2.0.15.2*yyyyyyyyyyyyyyyy} >
Variables	Where the repeating y represents the spaces returned by the device to pad the response to 24 characters.
Notes	Package version number reported as Maj.Min.Pack.Build.

MODEL

Description	Discovery of the model name of the device.
Commands	< GET MODEL > < REP MODEL {AD4Q-Ayyyyyyyyyyyyyyyyyyyy} >

Variables	Where the repeating y represents the spaces returned by the device to pad the model name to 32 characters.
Notes	The device always responds with a 32-character model name.

QUADVERSITY_MODE

Description	Discovery of Quadversity mode.
Commands	<code>< GET QUADVERSITY_MODE ></code> <code>< REP QUADVERSITY_MODE OFF ></code> or <code>< REP QUADVERSITY_MODE ON ></code>
Variables	None.
Notes	Quadversity Mode - While only applicable to AD4Q, an attempt to GET on an AD4D shall report OFF.

RF_BAND

Description	Discovery of the RF band.
Commands	<code>< GET RF_BAND ></code> <code>< REP RF_BAND {G55yyyyy} ></code>
Variables	Where the repeating y represents the spaces returned by the device to pad the response to 8 characters.
Notes	None.

TRANSMISSION_MODE

Description	Discovery of the transmission mode.
Commands	<code>< GET TRANSMISSION_MODE ></code> <code>< REP TRANSMISSION_MODE STANDARD ></code> or <code>< REP TRANSMISSION_MODE HIGH_DENSITY ></code>
Variables	None.
Notes	None.

Channel Command Strings

AUDIO_GAIN

Description	Control for the channel audio gain.
Commands	< GET x AUDIO_GAIN > < REP x AUDIO_GAIN 030 > There is an offset of 18 so the actual value = 30 - 18 = 12 dB. To set to 22 dB: < SET x AUDIO_GAIN 40 > < REP x AUDIO_GAIN 040 > To decrement the value down 5 dB: < SET x AUDIO_GAIN DEC 5 > < REP x AUDIO_GAIN 035 > To increment the value up 10 dB: < SET x AUDIO_GAIN INC 10 > < REP x AUDIO_GAIN 045 >
Variables	Where x is the channel number.
Notes	Numeric 3 Characters 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

AUDIO_MUTE

Description	Control for the channel audio mute.
Commands	< GET x AUDIO_MUTE > < REP x AUDIO_MUTE OFF > < SET x AUDIO_MUTE ON > < REP x AUDIO_MUTE ON > < SET x AUDIO_MUTE OFF > < REP x AUDIO_MUTE OFF > < SET x AUDIO_MUTE TOGGLE > < REP x AUDIO_MUTE OFF >

Variables	Where x is the channel number.
Notes	TOGGLE switches between ON and OFF.

CHAN_NAME

Description	Control for the channel name.
Commands	<pre>< GET x CHAN_NAME > < REP x CHAN_NAME {yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy} > < SET x CHAN_NAME {Lead Vox} > < REP x CHAN_NAME {Lead Voxyyyyyyyyyyyyyyyyyyyyyyyy} ></pre>
Variables	Where x is the channel number. Where the repeating y represents or pads the 31-character string from the set: A-Z,a-z, 0-9,!"#\$\$%&'()*+,-./:;<=>?@[\\]^_`~ and space, that is, {1234567890123456789012345678901}.
Notes	SET only supports 8 characters. The device always responds with a 31-character name.

ENCRYPTION_STATUS

Description	Discovery of the encryption status.
Commands	<pre>< GET x ENCRYPTION_STATUS > < REP x ENCRYPTION_STATUS OK ></pre> When a mismatched transmitter is detected: <pre>< REP x ENCRYPTION_STATUS ERROR ></pre>
Variables	Where x is the channel number.
Notes	It is unnecessary to continuously query this parameter. The receiver automatically sends a REPort when interference is detected.

FREQUENCY

Description	Controls frequency settings.
Commands	<pre>< GET x FREQUENCY > < REP x FREQUENCY 0578350 > < SET x FREQUENCY 602125 ></pre>

	< REP x GROUP_CHANNEL {--,--yyyy} > (Note 1) < REP x FREQUENCY 0602125 >
Variables	Where x is the channel number. Where the repeating y represents the spaces returned by the device to pad the response to 7 characters.
Notes	Frequency and Group Channel FREQUENCY - Device always returns a 7-character, numeric string GROUP_CHANNEL - Device always returns a 10-character string Range and Step per the RF Band - GROUP_CHANNEL and FREQUENCY are related: - Setting FREQUENCY results in the GROUP_CHANNEL value reverting to "--,--" if not already indicating "--,--" in addition to the FREQUENCY value. - Setting GROUP_CHAN results in the corresponding FREQUENCY value being reported in addition to the GROUP_CHAN value. Commands: Starting from a default condition: <pre>< GET x GROUP_CHANNEL > < REP x GROUP_CHANNEL {1,1yyyyyy} > < GET x FREQUENCY > < REP x FREQUENCY 0606025 ></pre> SET the FREQUENCY to some new value: <pre>< SET x FREQUENCY 620000 > < REP x GROUP_CHANNEL {--,--yyyy} > < REP x FREQUENCY 0620000 ></pre> Similarly, when setting GROUP_CHANNEL, the corresponding FREQUENCY is reported: <pre>< SET x GROUP_CHANNEL {6,6} > < REP x FREQUENCY 0614650 > < REP x GROUP_CHANNEL {6,6yyyyyy} ></pre> - For channels operating in FD-C mode, there is a second FREQUENCY2 and GROUP_CHANNEL2 that behave in the same manner as FREQUENCY. <pre>< GET x FD_MODE > < REP x FD_MODE FD-C ></pre> <pre>< GET x FREQUENCY > < REP x FREQUENCY 0578350 ></pre> <pre>< GET x FREQUENCY2 > < REP x FREQUENCY2 0578850 ></pre>

FD_MODE

Description	Discovery of the type of frequency diversity.
Commands	<pre>< GET x FD_MODE > < REP x FD_MODE OFF > < GET x FD_MODE > < REP x FD_MODE FD-C > < GET x FD_MODE > < REP x FD_MODE FD-S ></pre>
Variables	Where x is the channel number.
Notes	FD-C : Combining mode FD-S : Selection mode

GROUP_CHANNEL

Description	Controls the group channel mappings.
Commands	<pre>< GET x GROUP_CHANNEL > < REP x GROUP_CHANNEL {1,1yyyyyyy} > < SET x GROUP_CHANNEL {6,100} > < REP x FREQUENCY 0652875 > < REP x GROUP_CHANNEL {6,100yyyyy} ></pre>
Variables	Where x is the channel number. Where the repeating y represents the spaces returned by the device to pad the response to 10 characters.
Notes	For channels operating in FD-C mode, GROUP_CHANNEL2 corresponds to the second Group Channel value Device always returns a 10-character string Refer to the Group/Channel mappings corresponding to the RF Band and Transmission Mode of the device You must parse the "," from within the reported value Characters before the "," are the Group ID Characters after the "," are the Channel ID The value: "--,--" is the wildcard indicating no GROUP_CHANNEL value is set You cannot SET to "--,--" Note: GROUP_CHANNEL and FREQUENCY are related as described in FREQUENCY.

INTERFERENCE_STATUS

Description	Discovery of interference.
Commands	< GET x INTERFERENCE_STATUS > < REP x INTERFERENCE_STATUS NONE > When interference is detected: < REP x INTERFERENCE_STATUS DETECTED > For FD-C channels only: < GET x INTERFERENCE_STATUS2 > < REP x INTERFERENCE_STATUS2 NONE > When interference is detected: < REP x INTERFERENCE_STATUS2 DETECTED >
Variables	Where x is the channel number.
Notes	It is unnecessary to continuously query this parameter. The receiver automatically sends a REPort when interference is detected.

UNREGISTERED_TX_STATUS

Description	Discovery of an unregistered transmitter.
Commands	< GET x UNREGISTERED_TX_STATUS > < REP x UNREGISTERED_TX_STATUS OK > A REP when the condition is occurring: < REP 1 UNREGISTERED_TX_STATUS ERROR >
Variables	Where x is the channel number.
Notes	It is unnecessary to continuously query this parameter. The receiver automatically sends a REPort when an unregistered transmitter is detected.

Metering Command Strings

The majority of properties generate REP (Report) messages when their values change (for example, Frequency, Channel Name, and so on.)

For attributes such as audio meters, RF meters, channel quality 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.
- Combine the metered attributes into a single SAMPLE message per channel.

for example,

```
< SAMPLE chNum ALL qual audBitmap audPeak audRms rfAntStats rfBitmapA rfRssiA rfBitmapB rfRssiB >
```

returns

```
< SAMPLE 1 ALL 005 000 045 062 BB 31 099 31 085 >
```

where each field is documented and easy to parse.

- Generate periodic SAMPLE messages at the interval set via the METER_RATE.

Note: To turn off sampling, use

```
< SET x METER_RATE 00000 >
```

where **x** is the channel number.

The following sections detail METER_RATE and SAMPLE followed by the set of metered attributes.

METER_RATE

Description	Controls the meter rate.
Commands	<pre>< GET x METER_RATE > < REP x METER_RATE 00000 ></pre> <pre>< SET x METER_RATE 01000 > < REP x METER_RATE 01000 ></pre> Note: This results in one SAMPLE every second.
Variables	Where x is the channel number.
Notes	Numeric, 5-character fixed output. 00000 - Metering OFF (default) 00100 to 65535 - The interval of each SAMPLE report in milliseconds. For example, 00100 - Sample every 100 millisecond (10 samples per sec) 01000 - Sample every second 05000 - Sample every 5 seconds

SAMPLE

Description	Specifies which of the Metering Commands you want to sample.
Commands	Standard Channel: Quadversity = OFF, FD = OFF or FD-S:

< SAMPLE chNum ALL qual audBitmap audPeak audRms rfAntStats rfBitmapA rfRssiA rfBitmapB rfRssiB >

< SAMPLE 1 ALL 005 031 102 102 BB 31 086 31 065 >

Quadversity Channel: Quadversity = ON, FD = OFF

< SAMPLE chNum ALL qual audBitmap audPeak audRms rfAntStats rfBitmapA rfRssiA rfBitmapB rfRssiB rfBitmapC rfRssiC rfBitmapD rfRssiD >

< SAMPLE 1 ALL 005 031 102 102 BBBB 31 083 31 068 31 069 31 072 >

FD-C Channel: Quadversity = OFF, FD = FD-C

< SAMPLE chNum ALL qual audBitmap audPeak audRms rfAntStatsF1 rfBitmapF1A rfRssiF1A rfBitmapF1B rfRssiF1B rfAntStatsF2 rfBitmapF2A rfRssiF2A rfBitmapF2B rfRssiF2B >

< SAMPLE 1 ALL 005 031 102 102 BB 31 082 31 060 BB 31 082 31 060 >

Quadversity and FD-C Channel: Quadversity = ON, FD = FD-C

Sample input:

Note: The sample input that follows is actually 1 line of input spaced to show correlation.

< SAMPLE chNum ALL qual audBitmap audPeak audRms
rfAntStatsF1 rfBitmapF1A rfRssiF1A rfBitmapF1B rfRssiF1B rfBitmapF1C rfRssiF1C
rfBitmapF1D rfRssiF1D
rfAntStatsF2 rfBitmapF2A rfRssiF2A rfBitmapF2B rfRssiF2B rfBitmapF2C rfRssiF2C
rfBitmapF2D rfRssiF2D >

Sample response:

< SAMPLE 1 ALL 005 031 102 102 BBBB 31 084 31 065 31 070 31 070 BBBB 31 084 31 065 31 070 31 070 >

Variables

Key mapping:

Key	Corresponding command string for value format	Num Char	Notes
qual	CHANNEL_QUALITY	3	
aud- Bitmap	AUDIO_LED_BITMAP	3	Use for simple UIs as this is a direct mirror of the front panel LEDs.
audPeak	AUDIO_LEVEL_PEAK	3	
audRms	AUDIO_LEVEL_RMS	3	
rfAntStats	ANTENNA_STATUS	2	Quadversity: XX for Quad = OFF XXXX for Quad = ON FD-C: Append F2's set of RF
rfBitmap	RSSI_LED_BITMAP	2	Use for simple UIs as this is a direct mirror of the front panel LEDs.

	Key	Corresponding command string for value format	Num Char	Notes
				See the command example for how this scales to antenna A-D.
	rfRssi	RSSI	3	See the command example for how this scales to antenna A-D.
Notes	None.			

ANTENNA_STATUS

Description	Discovery of antenna status.
Commands	Normal channel, both LEDs lit: < GET x ANTENNA_STATUS > < REP x ANTENNA_STATUS BB > Quadversity example where antenna A-Blue, B-Red, C-Off, D-Blue: < GET x ANTENNA_STATUS > < REP x ANTENNA_STATUS BRXB >
Variables	Where x is the channel number.
Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Each character represents an antenna ABCD, where the value for each can be: X - Off R - Red B - Blue Example of ANTENNA_STATUS with QUADVERSITY_MODE=OFF: BB Example of ANTENNA_STATUS with QUADVERSITY_MODE=ON: BBBB

AUDIO_LED_BITMAP

Description	Discovery of the Audio LED bitmap.
Commands	< GET x AUDIO_LED_BITMAP > < REP x AUDIO_LED_BITMAP 031 >
Variables	Where x is the channel number.

Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Numeric, 3-character fixed output Each bit corresponds to an LED from bottom to top. LEDs 1-4 - Green LEDs 5-7 - Amber LED 8 - Red Example values: 0 = b00000000 - all off 1 = b00000001 - first LED on ... 131 = b10000011 - after OL that is held, and the 2 bottom green LEDs are on ... 255 = b11111111 - all LEDs on
--------------	---

AUDIO_LEVEL_PEAK

Description	Discovery of peak audio level.
Commands	< GET x AUDIO_LEVEL_PEAK > < REP x AUDIO_LEVEL_PEAK 102 >
Variables	Where x is the channel number
Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Numeric, 3-character fixed output Units: dBFS The actual value = the reported value - 120 Value range as reported over command strings: 000 - 120 Value range after conversion to the actual value: -120 - 0 dBFS The AD4 values fall in the range of about -100 to 0 dBFS.

AUDIO_LEVEL_RMS

Description	Discovery of the RMS audio level.
Commands	< GET x AUDIO_LEVEL_RMS > < REP x AUDIO_LEVEL_RMS 102 >
Variables	Where x is the channel number.

Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Format: Numeric, 3-character fixed output Units: dBFS The actual value = the reported value - 120 Value range as reported over command strings: 000 - 120 Value range after conversion to the actual value: -120 - 0 dBFS The AD4 values fall in the range of about -100 to 0 dBFS.
--------------	--

CHAN_QUALITY

Description	Discovery of the channel quality.
Commands	< GET x CHAN_QUALITY > < REP x CHAN_QUALITY 005 >
Variables	Where x is the channel number.
Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Numeric, 3-character fixed output 000 - 005 255 = Unknown

RSSI

Description	Discovery of the RSSI.
Commands	For a normal channel (Quadversity = OFF) < GET x RSSI 0 > < REP x RSSI 1 083 > < REP x RSSI 2 064 > For a Quadversity channel (Quadversity = ON) < GET x RSSI 0 > < REP x RSSI 1 083 > < REP x RSSI 2 064 > < REP x RSSI 3 082 > < REP x RSSI 4 071 >
Variables	Where x is the channel number.
Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Numeric, 3-character fixed output per antenna Units: dBm

	The actual value = the reported value - 120 Value range as reported over command strings: 000 - 120 Value range after conversion to the actual value: -120 - 0 dBm Channel Number, Antenna Index 1 : A 2 : B 3 : C 4 : D
--	--

RSSI_LED_BITMAP

Description	Discovery of the RSSI LED bitmap.
Commands	Antenna A, 5 LEDs on < GET x RSSI_LED_BITMAP 1 > < REP x RSSI_LED_BITMAP 1 31 > Example of all for a Quad=OFF < GET x RSSI_LED_BITMAP 0 > < REP x RSSI_LED_BITMAP 1 63 > < REP x RSSI_LED_BITMAP 2 63 > Example of all for a Quad=ON < GET x RSSI_LED_BITMAP 0 > < REP x RSSI_LED_BITMAP 1 63 > < REP x RSSI_LED_BITMAP 2 63 > < REP x RSSI_LED_BITMAP 3 63 > < REP x RSSI_LED_BITMAP 4 63 >
Variables	Where x is the channel number.
Notes	This is a metered property using SAMPLE; it does not generate a report on a value change. Numeric, 2-character fixed output per antenna: Represents a bitmap of which of the RSSI LEDs are ON from bottom to top. LEDs 1-5 : Amber LED 6 : Red (for Overload and RF Pad engaged indications) Example Variables 00 = b000000 = All LEDs off 01 = b000001 = LED1 ON (bottom) ...

31 = b0111111 = LEDs 1-5 ON
32 = b1000000 = LED6 ON (top, OL, all others OFF)
 ...
63 = b1111111 = All LEDs on
01 = b0000001 = LED1 ON (bottom)

Channel Number, Antenna Index

1 : A
 2 : B
 3 : C
 4 : D

Side Channel Command Strings

TX_BATT_BARS

Description	Discovery of the transmitter Battery Bars.
Commands	< GET x TX_BATT_BARS > < REP x TX_BATT_BARS 255 > Report when data becomes known: < REP x TX_BATT_BARS 004 >
Variables	Where x is the channel number.
Notes	Numeric, 3-character fixed output 000 to 005 255 - Unknown

TX_BATT_CHARGE_PERCENT

Description	Discovery of the transmitter battery charge.
Commands	< GET x TX_BATT_CHARGE_PERCENT > < REP x TX_BATT_CHARGE_PERCENT 255 > Report when data becomes known: < REP x TX_BATT_CHARGE_PERCENT 088 >

Variables	Where x is the channel number.
Notes	Numeric, 3-character fixed output 000 to 100 - Percent 255 - Unknown

TX_BATT_CYCLE_COUNT

Description	Discovery of the transmitter Battery Cycle Count.
Commands	< GET x TX_BATT_CYCLE_COUNT > < REP x TX_BATT_CYCLE_COUNT 65535 > Report when data becomes known: < REP x TX_BATT_CYCLE_COUNT 00019 >
Variables	Where x is the channel number.
Notes	Numeric, 5-character fixed output 00000 to 65534 - Number of cycles 65535 - Unknown

TX_BATT_HEALTH_PERCENT

Description	Discovery of the transmitter Battery Health.
Commands	< GET x TX_BATT_HEALTH_PERCENT > < REP x TX_BATT_HEALTH_PERCENT 255 > Report when data becomes known: < REP x TX_BATT_HEALTH_PERCENT 088 >
Variables	Where x is the channel number.
Notes	Numeric, 3-character fixed output 000 to 100 - Percent 255 - Unknown

TX_BATT_MINS

Description	Discovery of the transmitter battery runtime minutes.
--------------------	---

Commands	< GET x TX_BATT_MINS > < REP x TX_BATT_MINS 65535 > Report when data becomes known (example is 2 hours 5 minutes): < REP x TX_BATT_MINS 00125 >
Variables	Where x is the channel number.
Notes	Numeric, 5-character fixed output 00000 to 65532 - Number of minutes of runtime 65533 - Battery communication warning 65534 - Battery time calculating 65535 - Unknown, or not applicable

TX_BATT_TEMP_C

Description	Discovery of the transmitter battery temperature in Celsius.
Commands	< GET x TX_BATT_TEMP_C > < REP x TX_BATT_TEMP_C 255 > Report as data becomes known or changes: < REP x TX_BATT_TEMP_C 062 >
Variables	Where x is the channel number.
Notes	Numeric, 3-character fixed output 000 to 254 - Temperature 255 - Unknown, or not applicable The actual value = the reported value - 40.

TX_BATT_TEMP_F

Description	Discovery of the transmitter battery temperature in Fahrenheit.
Commands	< GET x TX_BATT_TEMP_F > < REP x TX_BATT_TEMP_F 255 > Report as data becomes known or changes: < REP x TX_BATT_TEMP_F 062 >
Variables	Where x is the channel number.

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

TX_BATT_TYPE

Description	Discovery of the transmitter's battery type.
Commands	< GET x TX_BATT_TYPE > < REP x TX_BATT_TYPE UNKN > Report when data becomes known: < REP x TX_BATT_TYPE LION >
Variables	Where x is the channel number.
Notes	Report responses: LION ALKA NIMH LITH UNKN - Either no transmitter or not supported by transmitter

TX_DEVICE_ID

Description	Discovery of the transmitter's Device ID.
Commands	< GET x TX_DEVICE_ID > < REP x TX_DEVICE_ID {yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy} > Report of a transmitter being received: < REP x TX_DEVICE_ID {LeadVoxyyyyyyyyyyyyyyyyyyyy} >
Variables	Where x is the channel number. Where the repeating y represents or pads the 31-character string
Notes	31-character string - Device ID of the transmitter being received Unknown - blank, all spaces

TX_INPUT_PAD

Description	Discovery of the transmitter Input Pad.
Commands	< GET x TX_INPUT_PAD > < REP x TX_INPUT_PAD 255 > Report when data becomes known: < REP x TX_INPUT_PAD 000 >
Variables	Where x is the channel number.
Notes	Numeric, 3-character fixed output 000 - Input Pad is ON (-12 dB) 012 - Input Pad is OFF (0 dB) 255 - Unknown, or Input Pad is not applicable to the transmitter variant The actual value = the reported value - 12. Input Pad is applicable to AD1, ADX1. All others indicate unknown.

TX_LOCK

Description	Discovery of the transmitter Lock.
Commands	< GET x TX_LOCK > < REP x TX_LOCK UNKNOWN > Report when data becomes known: < REP x TX_LOCK MENU >
Variables	Where x is the channel number.
Notes	Report responses: NONE POWER MENU ALL UNKNOWN

TX_MODEL

Description	Discovery of the transmitter model.
--------------------	-------------------------------------

Commands	< GET x TX_MODEL > < REP x TX_MODEL UNKNOWN > Report of a transmitter being received: < REP x TX_MODEL AD2 >
Variables	Where x is the channel number.
Notes	Report responses: AD1 AD2 ADX1 ADX1M ADX2 ADX2FD UNKNOWN

TX_MUTE_MODE_STATUS

Description	Discovery of the transmitter Mute Mode Status.
Commands	< GET x TX_MUTE_MODE_STATUS > < REP x TX_MUTE_MODE_STATUS UNKNOWN > Report when data becomes known: < REP x TX_MUTE_MODE_STATUS MUTE >
Variables	Where x is the channel number.
Notes	ON - Audio is being transmitted (power switch in "ON" position) MUTE - Audio is being locally muted, but still transmitting RF (power switch is in "OFF" position) UNKNOWN - Either no transmitter or not supported by transmitter

TX_OFFSET

Description	Discovery of the transmitter offset.
Commands	< GET x TX_OFFSET > < REP x TX_OFFSET 255 > Report when data becomes known: < REP x TX_OFFSET 012 >

Variables	Where x is the channel number.
Notes	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_POLARITY

Description	Discovery of the transmitter Polarity.
Commands	< GET x TX_POLARITY > < REP x TX_POLARITY UNKNOWN > Report when data becomes known: < REP x TX_POLARITY POSITIVE >
Variables	Where x is the channel number.
Notes	Report responses: POSITIVE NEGATIVE UNKNOWN - Either no transmitter or not supported by transmitter

TX_POWER_LEVEL

Description	Discovery of the transmitter power level.
Commands	< GET x TX_POWER_LEVEL > < REP x TX_POWER_LEVEL 255 > Report when normal power level is decoded: < REP x TX_POWER_LEVEL 010 >
Variables	Where x is the channel number.
Notes	Numeric, 3-character fixed length Transmit power level in mW (for example, 002, 010, or 020) 255 - Unknown

TX_TALK_SWITCH

Description	Discovery of the transmitter Talk Switch activity.
Commands	Before a Talk Switch has been pressed, or no transmitter has been decoded: <pre>< GET x TX_TALK_SWITCH ></pre> <pre>< REP x TX_TALK_SWITCH UNKNOWN ></pre> User presses Talk Switch, there is a report: <pre>< REP x TX_TALK_SWITCH ON ></pre> User releases the Talk Switch, there is a report: <pre>< REP x TX_TALK_SWITCH OFF ></pre> Now that the AD4 has knowledge that a Talk Switch is being used on this session: <pre>< GET x TX_TALK_SWITCH ></pre> <pre>< REP x TX_TALK_SWITCH OFF ></pre> When the Transmitter goes out of range, that session is lost and the state of the Talk Switch becomes unknown. There is a report (along with the reports for battery level unknown, model unknown, and so on): <pre>< REP x TX_TALK_SWITCH UNKNOWN ></pre>
Variables	Where x is the channel number.
Notes	ON - Talk Switch button press has been detected OFF - Talk Switch button has been released UNKNOWN - No transmitter detected and/or no Talk Switch button press has been detected up to this point in the transmitter's current connection. Note: OFF and UNKNOWN are treated the same by the AD4 algorithms with respect to Talk Switch routing.

Transmitter Slots Command Strings

Transmitter slot general description

- Each channel of AD4 has 8 slots where a transmitter can be registered
 - Specify Channel number 0 to interact with ALL 8 slots on a channel
 - Specify Channel number 1 through 8 to interact with that specific slot on a channel
- The SLOT_STATUS lets you know the status and what to expect from further operations
 - EMPTY - No transmitter exists in the SLOT.
GET - You can GET the other properties but those report as "Unknown" values
SET - Not supported and results in **REP ERR**
 - STANDARD - An AD (standard) transmitter is registered in the slot
 - LINKED.INACTIVE - An ADX (enhanced) transmitter is registered in the slot, but is not currently online

- STANDARD & LINKED.INACTIVE:

GET - You can GET more useful data for the SLOT_DEVICE_ID and SLOT_TX_MODEL but the other properties report as "Unknown" values

SET - Not supported and results in **REP ERR**

- LINKED.ACTIVE - An ADX (enhanced) transmitter is registered in the slot and is online.

GET - You can GET more useful data for all SLOT properties

SET - You can SET properties where it is applicable to do so. See the specific command details

Note: As transmitters get linked, unlinked, and moved around there will be corresponding REP events for all of the slot attributes that change. This is a great deal of data, but a control system can readily parse and use the data that it requires.

SLOT_BATT_BARS

Description	Discovery of the transmitter's battery bars by slot.
Commands	Transmitter is LINKED.ACTIVE: <pre>< GET x SLOT_BATT_BARS z > < REP x SLOT_BATT_BARS z 004 ></pre> Reports occur as the battery depletes: <pre>< REP x SLOT_BATT_BARS z 003 > ... < REP x SLOT_BATT_BARS z 002 ></pre> Transmitter is out of range: <pre>< GET x SLOT_BATT_BARS z > < REP x SLOT_BATT_BARS z 255 ></pre>
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 3-character fixed output 000 to 005 - Number of bars 255 - Unknown

SLOT_BATT_CHARGE_PERCENT

Description	Discovery of the battery charge percent by slot.
Commands	Transmitter is LINKED.ACTIVE: <pre>< GET x SLOT_BATT_CHARGE_PERCENT z > < REP x SLOT_BATT_CHARGE_PERCENT z 087 ></pre> Reports occur as the battery is first calculating, then depletes: <pre>< REP x SLOT_BATT_CHARGE_PERCENT z 087 > ... < REP x SLOT_BATT_CHARGE_PERCENT z 086 ></pre>

	... < REP x SLOT_BATT_CHARGE_PERCENT z 085 > Transmitter is out of range, or not applicable < REP x SLOT_BATT_CHARGE_PERCENT z 255 >
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 3-character fixed output 000 to 100 - Percent 255 - Unknown

SLOT_BATT_CYCLE_COUNT

Description	Discovery of the battery cycle count by slot.
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_BATT_CYCLE_COUNT z > < REP x SLOT_BATT_CYCLE_COUNT z 00013 > Transmitter is out of range, or not applicable: < REP x SLOT_BATT_CYCLE_COUNT z 65535 >
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 5-character fixed output 00000 to 65534 - Number of cycles 65535 - Unknown

SLOT_BATT_HEALTH_PERCENT

Description	Discovery of the battery health by slot.
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_BATT_HEALTH_PERCENT z > < REP x SLOT_BATT_HEALTH_PERCENT z 097 > Reports as the Transmitter is ON/OFF: < REP x SLOT_BATT_HEALTH_PERCENT z 097 > ... < REP x SLOT_BATT_HEALTH_PERCENT z 255 > ... < REP x SLOT_BATT_HEALTH_PERCENT z 096 >

Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 3-character fixed output 000 to 100 - Percent 255 - Unknown

SLOT_BATT_MINS

Description	Discovery of the battery runtime by slot.
Commands	Transmitter is LINKED.ACTIVE: <pre>< GET x SLOT_BATT_MINS z > < REP x SLOT_BATT_MINS z 00360 ></pre> Reports occur as the battery is first calculating, then depletes: <pre>< REP x SLOT_BATT_MINS z 65534 > < REP x SLOT_BATT_MINS z 00300 > ... < REP x SLOT_BATT_MINS z 00299 ></pre> Transmitter is out of range: <pre>< GET x SLOT_BATT_MINS z > < REP x SLOT_BATT_MINS z 65535 ></pre>
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 5-character fixed output 00000 to 65532 - Number of minutes of runtime 65533 - Battery communication warning. (Check contacts) 65534 - Battery time calculating 65535 - Unknown, or not applicable

SLOT_BATT_TYPE

Description	Controls the transmitter slot's battery type.
Commands	Transmitter is LINKED.ACTIVE: <pre>< GET x SLOT_BATT_TYPE z > < REP x SLOT_BATT_TYPE z LION ></pre> Transmitter is out of range, or a not-supported by Transmitter:

	<pre>< GET x SLOT_POLARITY z > < REP x SLOT_POLARITY z UNKN > < SET x SLOT_POLARITY z ALKA > < REP ERR ></pre>
Variables	Where x is the channel number and z is the slot number.
Notes	Report responses: LION ALKA NIMH LITH UNKN - Either no transmitter or not supported by transmitter

SLOT_INPUT_PAD

Description	Controls the transmitter's slot input pad.
Commands	Transmitter is LINKED.ACTIVE, input pad not engaged: <pre>< GET x SLOT_INPUT_PAD z > < REP x SLOT_INPUT_PAD z 012 ></pre> To engage the input pad: <pre>< SET x SLOT_INPUT_PAD z 0 > < REP x SLOT_INPUT_PAD z 000 ></pre> Transmitter is out of range, or property not supported by the transmitter variant: <pre>< GET x SLOT_INPUT_PAD z > < REP x SLOT_RF_OUTPUT z 255 > < SET x SLOT_RF_OUTPUT z 0 > < REP ERR ></pre>
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 3-character fixed output 000 - Input Pad is ON (-12 dB) 012 - Input Pad is OFF (0 dB) 255 - Unknown, or Input Pad is not applicable to the transmitter variant The actual value = the reported value - 12. Note: A remotely controllable Input Pad is applicable to ADX1. All others indicate unknown.

SLOT_OFFSET

Description	Controls the transmitter's slot offset.
Commands	Get for Linked.Active transmitter: (default) < GET x SLOT_OFFSET z > < REP x SLOT_OFFSET z 012 > SET for Linked.Active transmitter: to min value: < SET x SLOT_OFFSET z 0 > < REP x SLOT_OFFSET z 000 > SET for Linked.Active transmitter: (demonstrating INC and DEC) < SET x SLOT_OFFSET z INC 5 > < REP x SLOT_OFFSET z 005 > < SET x SLOT_OFFSET z INC 5 > < REP x SLOT_OFFSET z 010 > < SET x SLOT_OFFSET z DEC 2 > < REP x SLOT_OFFSET z 008 > Attempt to GET/SET for an unsupported or out of range transmitter: < GET x SLOT_OFFSET z > < REP x SLOT_OFFSET z 255 > < SET x SLOT_OFFSET z 012 > < REP ERR >
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 3-character fixed Range - 000 to 033 255 - Unknown The actual value = the reported value - 12. The Actual Range is -12 to +21 dB in 1 dB increments.

SLOT_POLARITY

Description	Controls the transmitter's slot polarity.
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_POLARITY z > < REP x SLOT_POLARITY z POSITIVE >

	< SET x SLOT_POLARITY z NEGATIVE > < REP x SLOT_POLARITY z NEGATIVE > Transmitter is out of range, or a not-supported transmitter such as ADX2: < GET x SLOT_POLARITY z > < REP x SLOT_POLARITY z UNKNOWN > < SET x SLOT_POLARITY z POSITIVE > < REP ERR >
Variables	Where x is the channel number and z is the slot number.
Notes	Report responses: POSITIVE NEGATIVE UNKNOWN - Either no transmitter or not supported by transmitter <i>Note: Polarity is only supported by ADX1 and ADX1M.</i>

SLOT_RF_OUTPUT

Description	Controls the transmitter's RF output.
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_RF_OUTPUT z > < REP x SLOT_RF_OUTPUT z RF_ON > < SET x SLOT_RF_OUTPUT z RF_MUTE > < REP x SLOT_RF_OUTPUT z RF_MUTE > Transmitter is out of range: < GET x SLOT_RF_OUTPUT z > < REP x SLOT_RF_OUTPUT z UNKNOWN > < SET x SLOT_RF_OUTPUT z RF_MUTE > < REP ERR >
Variables	Where x is the channel number and z is the slot number.
Notes	Report responses: UNKNOWN RF_ON RF_MUTE

SLOT_RF_POWER

Description	REports the actual power level of the unit in mW. Note: Use <i>SLOT_RF_POWER_MODE</i> to set the power level
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_RF_POWER z > < REP x SLOT_RF_POWER z 002 > Transmitter is LINKED.INACTIVE or out of range: < GET x SLOT_RF_POWER z > < REP 255 > Reports as the level changes: < REP x SLOT_RF_POWER z 040 > < REP x SLOT_RF_POWER z 010 > < REP x SLOT_RF_POWER z 255 > Attempt to SET SLOT_RF_POWER reports an ERR: < SET x SLOT_RF_POWER z 10 > < REP ERR >
Variables	Where x is the channel number and z is the slot number.
Notes	Use SLOT_RF_POWER_MODE to SET the power level Numeric, 3-character fixed output 002, 010, 020, 035, 040, 050 - Typical values 255 - Unknown

SLOT_RF_POWER_MODE

Description	Controls the logical power level the unit is operating on.
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_RF_POWER_MODE z > < REP x SLOT_RF_POWER_MODE z NORMAL > Using SET (note both responses): < SET x SLOT_RF_POWER_MODE z LOW > < REP x SLOT_RF_POWER_MODE z LOW > < REP x SLOT_RF_POWER z 002 > Transmitter is out of range:

	< SET x SLOT_RF_POWER z LOW > < REP ERR > Reports as the level changes: < REP x SLOT_RF_POWER_MODE z HIGH > < REP x SLOT_RF_POWER z 040 > Unit changes frequency to a more restrictive HIGH Power allowed value: < REP x SLOT_RF_POWER z 020 >
Variables	Where x is the channel number and z is the slot number.
Notes	Report responses: UNKNOWN LOW NORMAL HIGH Note: Some RF Bands or modes do not support HIGH and any attempt to use HIGH returns a REP ERR.

SLOT_SHOWLINK_STATUS

Description	Discovery of the slot showlink status.
Commands	Transmitter is LINKED.ACTIVE: < GET x SLOT_SHOWLINK_STATUS z > < REP x SLOT_SHOWLINK_STATUS z 005 > Transmitter is out of range: < GET x SLOT_SHOWLINK_STATUS z > < REP x SLOT_SHOWLINK_STATUS z 255 > Reports as the level changes: < REP x SLOT_SHOWLINK_STATUS z 003 > < REP x SLOT_SHOWLINK_STATUS z 004 > < REP x SLOT_SHOWLINK_STATUS z 005 >
Variables	Where x is the channel number and z is the slot number.
Notes	Numeric, 3-character fixed output 001 to 005 255 - Unknown

SLOT_STATUS

Description	Discovery of the slot status.
Commands	An empty slot: <pre>< GET x SLOT_STATUS z > < REP x SLOT_STATUS z EMPTY ></pre> Slot 1 with an AD transmitter: <pre>< GET x SLOT_STATUS 1 > < REP x SLOT_STATUS 1 STANDARD ></pre> Slot 2 with an ADX transmitter that is online: <pre>< GET x SLOT_STATUS 2 > < REP x SLOT_STATUS 2 LINKED.ACTIVE ></pre> The transmitter goes offline: (among others) <pre>< REP x SLOT_STATUS 2 LINKED.INACTIVE ></pre> The user registers an AD1 to Slot 2: <pre>< REP x SLOT_STATUS 2 STANDARD ></pre> The user unlinks or cleans up Slot 2 or moves the transmitter: <pre>< REP x SLOT_STATUS 2 EMPTY ></pre>
Variables	Where x is the channel number and z is the slot number.
Notes	Report responses: EMPTY - No transmitter is registered in the slot STANDARD - An AD (standard) transmitter is registered in the slot LINKED.INACTIVE - An ADX (enhanced) transmitter is registered, linked, but the receiver is not connected to the live transmitter at this time LINKED.ACTIVE - An ADX (enhanced) transmitter is registered, linked, and the receiver is connected. You can remotely adjust the transmitter now.

SLOT_TX_DEVICE_ID

Description	Controls the device ID of the transmitter by slot.
Commands	An empty, unregistered slot: <pre>< GET x SLOT_TX_DEVICE_ID z > < REP x SLOT_TX_DEVICE_ID z {yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy} ></pre> A slot with an AD (standard) transmitter:

	<pre>< GET x SLOT_TX_DEVICE_ID z > < REP x SLOT_TX_DEVICE_ID z {AD1yyyyyyyyyyyyyyyyyyyyyyyy} ></pre> An attempt to set the Device ID of an AD transmitter that is not supported: <pre>< SET x SLOT_TX_DEVICE_ID z {Name_1} > < REP ERR ></pre> An attempt to set the Device ID of a Linked.Active ADX transmitter: <pre>< SET x SLOT_TX_DEVICE_ID z {Name_1} > < REP x SLOT_TX_DEVICE_ID z {Name_1yyyyyyyyyyyyyyyyyyyyyyyy} ></pre>
Variables	Where x is the channel number and z is the slot number. Where the repeating y represents the spaces returned by the device to pad the response to 31 characters.
Notes	SET only accepts 1 to 8 characters from the set: A-Z,a-z,0-9,!"#\$\$%&'()*+,-./:;<=>?@[\\]^_`~ and space The device always responds with a 31-character string Empty slot - blank, all spaces

SLOT_TX_MODEL

Description	Discovery of the transmitter slot's model.
Commands	An empty slot: <pre>< GET x SLOT_TX_MODEL z > < REP x SLOT_TX_MODEL z UNKNOWN ></pre> An occupied slot: <pre>< GET x SLOT_TX_MODEL z > < REP x SLOT_TX_MODEL z ADX1 ></pre>
Variables	Where x is the channel number and z is the slot number.
Notes	Report responses: AD1 AD2 ADX1 ADX1M ADX2

	ADX2FD
	UNKNOWN